Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

В. С. Муха

СЛУЧАЙНЫЕ ПРОЦЕССЫ. ПРАКТИКУМ

Рекомендовано УМО вузов Республики Беларусь
по образованию в области информатики и радиоэлектроники
в качестве пособия для студентов,
получающих высшее образование по специальности 1-53 01 02
«Автоматизированные системы обработки информации»

Минск БГУИР 2012

УДК 519.216(076)
ББК 22.171я73
М92

Р е ц е н з е н т ы:

кафедра компьютерных технологий и систем Белорусского государственного университета (протокол №7 от 31.01.2012 г.);

профессор кафедры математического моделирования и анализа данных Белорусского государственного университета, доктор физико-математических наук, профессор Е. Е. Жук

Муха, В. С.

М92 Случайные процессы. Практикум : пособие / В. С. Муха. – Минск : БГУИР, 2012. – 56 с. : ил.
ISBN 978-985-488-862-0.

Содержит описание семи практических занятий по темам дисциплины «Случайные процессы». При выполнении занятий предполагается использование системы Matlab с пакетами прикладных программ по статистике «Statistics Toolbox» и символьным вычислениям «Symbolic Toolbox». Описания содержат достаточно подробную теоретическую часть, относящуюся к каждой теме, описание средств системы Matlab для решения поставленных задач, указания к порядку выполнения работы и варианты индивидуальных заданий.

Для специальности 1-53 01 02 «Автоматизированные системы обработки информации».

УДК 519.216(076)
ББК 22.171я73

ISBN 978-985-488-862-0

Введение

Пособие предназначено для выполнения практических занятий по дисциплине «Случайные процессы» для студентов специальности 1-53 01 02 «Автоматизированные системы обработки информации» (АСОИ). Данная дисциплина является дисциплиной по выбору студентов и имеет небольшой объем: 16 часов лекций и 18 часов практических занятий. При таком объеме возникает проблема выбора разделов для изучения из огромной и весьма развитой теории случайных процессов. Тематика и способ проведения практических занятий, предложенные в пособии, адаптированы автором к специфике специальности АСОИ.

Во-первых, предпочтение отдано изучению теории случайных процессов с дискретным временем, в частности, теории марковских процессов. Теория марковских процессов является основой таких разделов исследования операций, как теория массового обслуживания и динамическое программирование. Кроме того, теория случайных процессов с дискретным временем ориентирована на компьютерную обработку данных.

Во-вторых, практические занятия проводятся не в аудитории с доской и мелом, а в компьютерном классе. Это должно способствовать приобретению практических навыков в программировании методов обработки информации и разработке систем обработки информации.

Вместе с тем изучение дисциплины не должно ограничиваться только программированием, несмотря на то что современное программное обеспечение позволяет выполнять практически все необходимые как численные, так и аналитические расчеты. «Универсальный студент», из которого впоследствии вырастет «универсальный специалист», а не «программист-кодировщик», должен обогащать свою память знанием теории с использованием в том числе ручки и бумаги.

Тема 1. Моментные функции случайных процессов

1.1. Цель работы

1. Ознакомление с параметрической формой задания случайных процессов.
2. Приобретение навыков расчета математических ожиданий, дисперсий, ковариационных и взаимных ковариационных функций случайных процессов.

1.2. Теоретические положения

Математическое ожидание $a_\xi(t)$, дисперсия $d_\xi(t)$, ковариационная функция $R_\xi(t_1,t_2)$ случайного процесса $\xi(t)$ определяются выражениями:

$$a_\xi(t) = E\big(\xi(t)\big),$$

$$d_\xi(t) = E\Big((\xi(t) - a_\xi(t))^2\Big),$$

$$R_\xi(t_1,t_2) = E\Big((\xi(t_1) - a_\xi(t_1))(\xi(t_2) - a_\xi(t_2))\Big).$$

Взаимные ковариационные функции двух процессов $\xi(t)$ и $\eta(t)$ определяются выражениями:

$$R_{\xi,\eta}(t_1,t_2) = E\Big((\xi(t_1) - a_\xi(t_1))(\eta(t_2) - a_\eta(t_2))\Big),$$

$$R_{\eta,\xi}(t_1,t_2) = E\Big((\eta(t_1) - a_\eta(t_1))(\xi(t_2) - a_\xi(t_2))\Big).$$

Здесь E – символ математического ожидания по соответствующему распределению.

Если случайные процессы $\xi(t)$ и $\eta(t)$ определены своими конечномерными плотностями вероятностей, то указанные моментные функции рассчитываются как следующие интегралы:

$$a_\xi(t) = \int_{-\infty}^{\infty} x f_\xi(x,t)dx,\ a_\eta(t) = \int_{-\infty}^{\infty} x f_\eta(x,t)dx,$$

$$d_{\xi}(t) = \int_{-\infty}^{\infty} (x - a_{\xi}(t)) f_{\xi}(x,t) dx,$$

$$d_{\eta}(t) = \int_{-\infty}^{\infty} (x - a_{\eta}(t)) f_{\eta}(x,t) dx,$$

$$R_{\xi}(t_1,t_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - a_{\xi}(t_1))(y - a_{\xi}(t_2)) f_{\xi}(x,y,t_1,t_2) dxdy,$$

$$R_{\eta}(t_1,t_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - a_{\eta}(t_1))(y - a_{\eta}(t_2)) f_{\eta}(x,y,t_1,t_2) dxdy,$$

$$R_{\xi,\eta}(t_1,t_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - a_{\xi}(t_1))(y - a_{\eta}(t_2)) f_{\xi,\eta}(x,y,t_1,t_2) dxdy,$$

$$R_{\eta,\xi}(t_1,t_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - a_{\eta}(t_1))(y - a_{\xi}(t_2)) f_{\eta,\xi}(x,y,t_1,t_2) dxdy.$$

Здесь $f_{\xi}(x,t)$, $f_{\eta}(x,t)$ – одномерные плотности вероятностей случайных процессов $\xi(t)$ и $\eta(t)$ соответственно; $f_{\xi}(x,y,t_1,t_2)$, $f_{\eta}(x,y,t_1,t_2)$ – двухмерные плотности вероятностей случайных процессов $\xi(t)$ и $\eta(t)$ соответственно; $f_{\xi,\eta}(x,y,t_1,t_2)$, $f_{\eta,\xi}(x,y,t_1,t_2)$ – взаимные двухмерные плотности вероятностей случайных процессов $\xi(t)$ и $\eta(t)$ соответственно.

Если же случайные процессы $\xi(t)$ и $\eta(t)$ заданы в параметрической форме, т.е. в виде известных функций $\xi(t,\lambda)$ и $\eta(t,\mu)$ векторных случайных параметров $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$, $\mu = (\mu_1, \mu_2, \ldots, \mu_k)$, то указанные моментные функции рассчитываются как следующие интегралы:

$$a_{\xi}(t) = \int_{-\infty}^{\infty} \xi(t,z) f_{\lambda}(z) dz, \; a_{\eta}(t) = \int_{-\infty}^{\infty} \eta(t,v) f_{\mu}(v) dv,$$

$$d_{\xi}(t) = \int_{-\infty}^{\infty} (\xi(t,z) - a_{\xi}(t)) f_{\lambda}(z) dz,$$

$$d_{\eta}(t) = \int_{-\infty}^{\infty} (\eta(t,v) - a_{\eta}(t)) f_{\mu}(v) dv,$$

$$R_\xi(t_1,t_2)=\int_{-\infty}^{\infty}(\xi(t_1,z)-a_\xi(t_1))(\xi(t_2,z)-a_\xi(t_2))f_\lambda(z)dz,$$

$$R_\eta(t_1,t_2)=\int_{-\infty}^{\infty}(\eta(t_1,v)-a_\eta(t_1))(\eta(t_2,v)-a_\eta(t_2))f_\mu(v)dv,$$

$$R_{\xi,\eta}(t_1,t_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\xi(t_1,z)-a_\xi(t_1))(\eta(t_2,v)-a_\eta(t_2))f_{\lambda,\mu}(z,v)dzdv,$$

$$R_{\eta,\xi}(t_1,t_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\eta(t_1,v)-a_\eta(t_1))(\xi(t_2,z)-a_\xi(t_2))f_{\lambda,\mu}(z,v)dzdv.$$

Здесь $f_\lambda(z)$, $f_\mu(v)$ – плотности вероятностей случайных векторов λ и μ соответственно; $f_{\lambda,\mu}(z,v)$ – совместная плотность вероятности случайных векторов λ и μ; $z=(z_1,z_2,...,z_m)$; $v=(v_1,v_2,...,v_k)$; $dz=dz_1dz_2\cdots dz_m$; $dv=dv_1dv_2\cdots dv_k$.

Если случайные векторы λ и μ независимые, то $f_{\lambda,\mu}(z,v)=f_\lambda(z)f_\mu(v)$.

Как свойства приведенных выше интегралов легко могут быть получены следующие свойства моментных функций. Если $\zeta(t)=\xi(t)\pm\eta(t)$, то

$$a_\zeta(t)=a_\xi(t)\pm a_\eta(t),$$

$$d_\zeta(t)=d_\xi(t)+d_\eta(t)\pm 2R_{\xi,\eta}(t,t),$$

$$R_\zeta(t_1,t_2)=R_\xi(t_1,t_2)+R_\eta(t_1,t_2)\pm R_{\xi,\eta}(t_1,t_2)\pm R_{\eta,\xi}(t_1,t_2).$$

Если случайные процессы $\xi(t)$ и $\eta(t)$ независимы, то в приведенных выше формулах $R_{\xi,\eta}(t_1,t_2)=R_{\eta,\xi}(t_1,t_2)=0$.

Если случайные процессы $\xi(t)$ и $\eta(t)$ независимы и $\zeta(t)=\xi(t)\eta(t)$, то

$$a_\zeta(t)=a_\xi(t)a_\eta(t),$$

$$d_\zeta(t)=d_\xi(t)d_\eta(t),$$

$$R_\zeta(t_1,t_2)=R_\xi(t_1,t_2)R_\eta(t_1,t_2).$$

Примерами случайных процессов являются метеорологические процессы (температура атмосферного воздуха, атмосферное давление, относительная влажность воздуха, направление и скорость ветра и др.).

1.3. Программные средства для выполнения работы

1.3.1. Символьное вычисление интегралов

Переменные или объекты объявляются в Matlab символьными с помощью функции **syms**. Например,

syms x1 x2 … создает группу символьных объектов.

syms x1 x2 … positive создает группу действительных положительных объектов.

syms x1 x2 … real создает группу символьных объектов с вещественными значениями, т.е. с нулевой мнимой частью.

syms x1 x2 … unreal создает группу чисто формальных символьных объектов без дополнительных свойств, т.е. страхует, чтобы эти объекты были ни действительными, ни положительным. Эту функцию можно использовать для отмены задания вещественных и положительных объектов.

Отметим, что символьные переменные в функции **syms** разделяются пробелами.

Пример 1.1. В результате выполнения программы

syms x1 x2 … real

x=[x1,x2];

будут созданы символьные положительные скалярные переменные x1, x2 и символьный массив (вектор-строка) x с положительными компонентами.

Для вычисления указанных в разд. 1.2 интегралов можно воспользоваться функцией символьного интегрирования **int**.

int(s) – возвращает символьное значение неопределенного интеграла от символьного выражения или массива символьных выражений **s** по переменной, которая автоматически определяется функцией **findsym**. Если **s** – константа, то вычисляется интеграл по переменной **'x'**;

int(s,a,b) – возвращает символьное значение определенного интеграла на отрезке интегрирования **[a,b]** от символьного выражения или массива символьных выражений **s** по переменной, которая автоматически определяется функцией **findsym**. Пределы интегрирования **a, b** могут быть как символьными, так и числовыми, как конечными, так и бесконечными (inf);

int(s,v) – возвращает символьное значение неопределенного интеграла от символьного выражения или массива выражений **s** по переменной **v**;

int(s,v,a,b) – возвращает символьное значение определенного интеграла от символьного выражения или массива символьных выражений **s** по переменной **v** с пределами интегрирования **[a,b]**.

Пример 1.2. Программа

```
syms alpha u
y2=int(sin(alpha*u),alpha)
```

возвращает следующий результат:

```
y2 =
-cos(alpha*u)/u
```

1.3.2. Символьные подстановки

Часто в символьное выражение вместо некоторых переменной или выражения необходимо подставить новую переменную или выражение. В Matlab это можно сделать с помощью функции **subs.**

y=subs(s,old,new) заменяет в символьном выражении **s** старую переменную или выражение **old** на новую переменную или выражение **new.**

Пример 1.3. Программа

```
syms t tau
y=sin(t);
y1=y*subs(y,t,t+tau)
```

возвращает следующий результат:

```
y1 =
sin(t)*sin(t+tau)
```

Отметим, что если все символьные переменные в символьном выражении **s** заменить на численные значения, то функция **y=subs(s,old,new)** возвратит численный результат.

В качестве переменной **old** может использоваться символьный массив, а в качестве переменной **new** – символьный или численный массив.

1.3.3. Построение графиков функций одной переменной

Для построения графика функции одной переменной используется функция **plot**. Вызов этой функции осуществляется командой

plot(x,y,s)

где **x, y** – одномерные массивы одинаковой размерности; **x** – массив значений аргумента функции $y = f(x)$; **y** – массив значений функции $y = f(x)$; **s** – строковая константа, определяющая цвет линии, маркер узловых точек и тип линии. Эта константа может содержать от одного до трех символов.

Цвет линии определяется символами **y** (желтый), **m** (фиолетовый), **c** (голубой), **r** (красный), **g** (зеленый), **b** (синий), **w** (белый), **k** (черный).

Тип узловой точки определяется символами **.** (точка), **o** (окружность), **x** (крестик), + (плюс), ***** (звездочка), **s** (квадрат), **d** (ромб), < > ^ (треугольники различной направленности), **p** (пятиугольник), **h** (шестиугольник).

Тип линии определяется символами - (непрерывная), : (короткие штрихи), -. (штрихпунктир), -- (длинные штрихи).

Символьную константу **s** можно опустить. В этом случае по умолчанию используется непрерывная линия желтого цвета.

Для построения в одном окне нескольких графиков можно использовать команду

plot(x1,y1,s1,x2,y2,s2,x3,y3,s3,…)

Команда **grid on** добавляет на график сетку. Команда **hold on** позволяет добавлять в то же графическое окно графики новых функций.

Пример 1.4

```
x=0:0.1:2*pi;
y1=sin(x);
y2=cos(x);
plot(x,y1,'k-o',x,y2,'r--*')
grid on
```

Созданный с помощью программы **plot** график можно скопировать в буфер Clipboard, активизировав в пункте **Edit** главного меню графического окна команду **Copy Figure** с целью его дальнейшего редактирования в каком-либо графическом редакторе, например Paint.

1.3.4. Построение графиков функций двух переменных

Ковариационная функция нестационарного случайного процесса и взаимная ковариационная функция двух совместно нестационарных случайных процессов являются функциями двух переменных. Построение графика функции двух переменных $z = f(x, y)$ выполняется в Matlab с помощью функций **meshgrid** и **mesh**. Специфика построения таких графиков в Matlab требует не просто задания ряда значений *x* и *y*, т. е. векторов **x** и **y**, а определения двухмерных массивов **X** и **Y**. Для создания таких массивов служит функция **meshgrid**.

[X,Y]=meshgrid(x,y) – преобразует область, заданную векторами **x** и **y**, в двухмерные массивы **X** и **Y**, которые могут быть использованы для вычисления значений функции двух переменных и построения трехмерных графиков. Эта функция формирует массивы **X** и **Y** таким образом, что строки выходного массива **X** являются копиями вектора **x**, а столбцы выходного массива **Y** – копиями вектора **y**.

mesh(X,Y,Z,C) – выводит в графическое окно сетчатую поверхность с цветами узлов поверхности, заданных массивом **C**.

mesh(X,Y,Z) – аналог предшествующей команды при **C=Z** с использованием функциональной окраски, при которой цвет задается высотой поверхности.

Пример 1.5

```
[x,y]=meshgrid(-2:0.1:3,-4:0.1:4);
z=x.^2+y.^2;
mesh(x,y,z)
```

В результате выполнения этой программы на экран будет выведен график функции двух переменных $z = x^2 + y^2$.

В приведенном примере массив значений функции двух переменных формируется посредством использования поэлементных операций с массивами. Использование поэлементных операций предполагает наличие навыков их использования, т.е. четкого представления того, что происходит при их применении. Если уверенности в правильности использования поэлементных операций нет, то массив значений функции можно сформировать путем применения вложенных циклов **for** и индексированных переменных. Так, предыдущий пример можно оформить следующим образом:

```
x1=-2:0.1:3;
y1=-4:0.1:4;
[x,y]=meshgrid(x1,y1);
nx=length(x1);
ny=length(y1);
for i=1:nx
 for j=1:ny
  z(j,i)=x1(i)^2+y1(j)^2;
 end;
end;
mesh(x,y,z)
```

1.3.5. Моделирование случайных чисел

Случайные числа из равномерного распределения $u(a,b)$, плотность вероятности которого имеет вид

$$f_{\xi}(x)=\begin{cases}\dfrac{1}{b-a}, & a<x<b,\ a,b\in R,\ a<b,\\ 0, & x\le a,\ x\ge b,\end{cases}$$

моделируются с помощью функции **y=unifrnd(a,b).**

Непрерывный случайный вектор $\overline{\xi}=(\xi_1,\ldots,\xi_m)$ называется распределенным по многомерному нормальному (гауссовскому) закону, если его плотность вероятности имеет вид

$$f_{\overline{\xi}}(X)=\frac{1}{\sqrt{(2\pi)^m|R|}}\exp(-\frac{1}{2}\varphi(X)),$$

где

$$\varphi(X)=(X-A)^T R^{-1}(X-A).$$

Здесь приняты следующие обозначения: $X^T=(x_1,\ldots,x_m)$ – вектор-строка аргументов плотности вероятности; $A^T=(a_1,\ldots,a_m)$ – вектор-строка параметров; $R=(R_{i,j}),\ i,\ j=\overline{1,m},$ – симметричная положительно определенная $(m\times m)$-матрица параметров; R^{-1} – матрица, обратная матрице *R*; $|R|$ – определитель матрицы *R*. Символ *T* означает транспонирование, так что X и A – векторы-столбцы. Параметры *A* и *R* распределения являются математическим ожиданием и ковариационной матрицей вектора $\overline{\xi}$ соответственно.

Если случайный вектор $\overline{\xi}=(\xi_1,\ldots,\xi_m)$ распределен по нормальному закону, то любой его подвектор (включая отдельную компоненту) также распределен по нормальному закону. Распределение этого подвектора получается вычеркиванием в исходном распределении из векторов $\overline{\xi}=(\xi_1,\ldots,\xi_m)$, $A^T=(a_1,\ldots,a_m)$ ненужных компонент и из матрицы R строк и столбцов, соответствующих ненужным компонентам.

Символически многомерное нормальное распределение обозначается как $f_{\xi}(X)=N(A,R)$.

Если $\bar{\xi} = (\xi_1, \xi_2)$ – двухмерный случайный вектор, распределенный по нормальному закону, то мы имеем

$$X^T = (x_1, x_2), \quad A^T = (a_1, a_2), \quad R = \begin{pmatrix} R_{1,1} & R_{1,2} \\ R_{2,1} & R_{2,2} \end{pmatrix},$$

$$|R| = R_{1,1}R_{2,2} - R_{1,2}R_{2,1}, \quad R^{-1} = \frac{1}{|R|}\begin{pmatrix} R_{2,2} & -R_{1,2} \\ -R_{2,1} & R_{1,1} \end{pmatrix},$$

$$\varphi(X) = (X - A)^T R^{-1}(X - A) =$$

$$= R_{2,2}(x_1 - a_1)^2 - 2R_{1,2}(x_1 - a_1)(x_2 - a_2) + R_{1,1}(x_2 - a_2)^2.$$

Если здесь обозначить $R_{1,1} = \sigma_1^2$, $R_{2,2} = \sigma_2^2$ и выразить коэффициент ковариации $R_{1,2} = R_{2,1} = \mathrm{cov}(\xi_1, \xi_2)$ через коэффициент корреляции $r_{1,2}$ по формуле

$$R_{1,2} = R_{2,1} = r_{1,2}\sigma_1\sigma_2,$$

то функцию $\varphi(X)$ можно представить в виде

$$\varphi(x_1, x_2) = \frac{1}{(1 - r_{1,2}^2)}\left(\frac{(x_1 - a_1)^2}{\sigma_1^2} - 2r_{1,2}\frac{(x_1 - a_1)}{\sigma_1}\frac{(x_2 - a_2)}{\sigma_2} + \frac{(x_2 - a_2)^2}{\sigma_2^2}\right),$$

а плотность вероятности двухмерного нормального распределения – в виде

$$f_{\bar{\xi}}(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1 - r_{1,2}^2}}\exp\left(-\frac{1}{2}\varphi(x_1, x_2)\right).$$

Функция **r=mvnrnd(mu,sigma,cases)** возвращает матрицу **r** случайных чисел, выбранных из многомерного нормального распределения с вектором средних **mu** и ковариационной матрицей **sigma**. Параметр **cases** является количеством строк в **r** (количеством многомерных случайных чисел).

1.4. Порядок выполнения работы

1.4.1. Рассчитать функции математического ожидания, дисперсии и ковариационной функции случайного процесса $\zeta(t)$, выбранного из табл. 1.1 в соответствии со своим вариантом задания. Вид составляющих случайного процесса

$\zeta(t)$ и распределения их параметров заданы в табл. 1.2. Параметры $\lambda = (b, c, d)$, φ, ω предполагаются случайными величинами, независимыми в совокупности. В символьных расчетах параметры b, c, d, φ, ω рассматривать как символьные переменные.

1.4.2. В одно графическое окно вывести 10 реализаций случайного процесса $\zeta(t)$ и функции математического ожидания и дисперсии, моделируя случайные значения параметров b, c, d, φ, ω и выбирая по своему усмотрению значения неслучайных параметров a_b, a_c, a_d, A, B, σ^2, σ_b^2, σ_c^2, σ_d^2, ω_1 и интервал времени $t_0 \le t \le t_1$.

Таблица 1.1

Варианты случайных процессов $\zeta(t)$ для расчетов

№ варианта	Процесс $\zeta(t)$	№ варианта	Процесс $\zeta(t)$
1	$\xi(t)+\eta(t)$	11	$\xi(t)\eta(t)$
2	$\xi(t)+\theta(t)$	12	$\xi(t)\theta(t)$
3	$\xi(t)+\psi(t)$	13	$\xi(t)\psi(t)$
4	$\xi(t)+\chi(t)$	14	$\xi(t)\chi(t)$
5	$\eta(t)+\theta(t)$	15	$\eta(t)\theta(t)$
6	$\eta(t)+\psi(t)$	16	$\eta(t)\psi(t)$
7	$\eta(t)+\chi(t)$	17	$\eta(t)\chi(t)$
8	$\theta(t)+\psi(t)$	18	$\theta(t)\psi(t)$
9	$\theta(t)+\chi(t)$	19	$\theta(t)\chi(t)$
10	$\psi(t)+\chi(t)$	20	$\psi(t)\chi(t)$

1.4.3. В отдельное графическое окно вывести график ковариационной функции случайного процесса $\zeta(t)$ при выбранных значениях неслучайных параметров a_b, a_c, a_d, A, B, σ^2, σ_b^2, σ_c^2, σ_d^2, ω_1.

1.4.4. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, численные и графические результаты работы программ, краткие выводы.

Таблица 1.2

Составляющие случайного процесса $\zeta(t)$, заданного в табл. 1.1

<table>
<tr><th>№ процесса</th><th>Вид процесса</th><th>Случайный параметр</th><th>Обозначение распределения параметра</th></tr>
<tr><td>1</td><td>$\xi(t) = b$</td><td>b</td><td rowspan="3">$\lambda = (b, c, d)$,
$f_\lambda(\bar{x}) = N(a_\lambda, R_\lambda)$,
$a_\lambda = (a_b, a_c, a_d)$,
$R_\lambda = \begin{bmatrix} \sigma_b^2 & 0 & 0 \\ 0 & \sigma_c^2 & 0 \\ 0 & 0 & \sigma_d^2 \end{bmatrix}$</td></tr>
<tr><td>2</td><td>$\eta(t) = ct$</td><td>c</td></tr>
<tr><td>3</td><td>$\theta(t) = \sigma^2 \exp(-dt)$</td><td>$d$</td></tr>
<tr><td>4</td><td>$\psi(t) = A\cos(\omega_1 t + \varphi)$</td><td>$\varphi$</td><td>$U(-a, a)$</td></tr>
<tr><td>5</td><td>$\chi(t) = B\cos\omega t$</td><td>ω</td><td>$U(0, a)$</td></tr>
</table>

1.5. Контрольные вопросы

1. Что такое случайный процесс (определение случайного процесса)?
2. Какие классы случайных процессов вы знаете?
3. Что такое конечномерные распределения случайного процесса?
4. Что такое математическое ожидание и дисперсия случайного процесса?
5. Что такое ковариационная функция случайного процесса?
6. Что такое взаимная ковариационная функция двух случайных процессов?
7. Какие свойства имеет ковариационная функция случайного процесса?

8. Какие свойства имеет взаимная ковариационная функция двух случайных процессов?

9. Запишите формулы для расчета моментных функций случайного процесса.

10. Объясните полученные графики реализаций, математического ожидания и дисперсии случайного процесса, основываясь на математической модели случайного процесса вашего варианта.

11. Объясните полученный график ковариационной функции случайного процесса исходя из свойств ковариационной функции.

Тема 2. Моделирование дискретных случайных величин

2.1. Цель работы

1. Приобретение навыков моделирования полной группы случайных событий.

2. Приобретение навыков моделирования случайных чисел из дискретных распределений.

2.2. Теоретические положения

2.2.1. Моделирование полной группы случайных событий

Случайные события $A_i \subseteq F$, $P(A_i) = p_i$, $i = \overline{1,k}$, заданные на вероятностном пространстве (Ω, F, P), образуют полную группу событий, если выполняются условия:

$$\text{а) } \bigcup_{i=1}^{k} A_i = \Omega; \quad \text{б) } A_i \bigcap A_j = \text{O},\ i \neq j; \quad \text{в) } P(A_i) \geq 0,\ i = \overline{1,k}; \quad \text{г) } \sum_{i=1}^{k} P(A_i) = 1.$$

Для моделирования полной группы случайных событий будем использовать базовую случайную величину α, т.е. непрерывную случайную величину, распределенную равномерно в интервале $(0,1)$. По вероятностям p_i случайных событий A_i определим числа

$$S_0 = 0,\ S_i = \sum_{j=1}^{i} p_j, \tag{2.1}$$

а события A_i определим как следующие множества:

$$A_i = \left(S_{i-1} \leq \alpha < S_i\right),\ i = \overline{1,k}. \tag{2.2}$$

Построенные таким образом события удовлетворяют сформулированным выше условиям а) – г). Действительно,

$$P(A_i) = \int_{S_{i-1}}^{S_i} f_\alpha(x)dx = \int_{S_{i-1}}^{S_i} dx = S_i - S_{i-1} = p_i,\ i = \overline{1,k}.$$

Алгоритм, моделирующий отдельное случайное событие A_i (его номер i), состоит в выполнении следующих шагов.

1. Расчет чисел $S_0, S_1, ..., S_k$ по формуле (2.1).

2. Обращение к датчику базового случайного числа и получение псевдослучайного числа α.

3. Сравнение α с величинами $S_0, S_1, ..., S_k$ и выбор номера события i, удовлетворяющего условию (2.2).

Для моделирования результатов n независимых испытаний, в каждом из которых может появиться одно из событий $A_1, A_2, ..., A_k$, пп. 2, 3 алгоритма необходимо повторить n раз.

2.2.2. Моделирование случайных чисел из дискретных распределений

Пусть ξ – дискретная случайная величина, принимающая значения $x_1, x_2, ..., x_k$ с вероятностями $p_1, p_2, ..., p_k$. Моделировать значения этой случайной величины можно с помощью алгоритма моделирования полной группы случайных событий, изложенного в п. 2.2.1. Действительно, между значениями x_i дискретной случайной величины и случайными событиями A_i полной группы событий, $i = \overline{1,k}$, существует взаимно-однозначное соответствие. Поэтому для моделирования дискретной случайной величины ξ к алгоритму, приведенному в п. 2.2.1, следует добавить четвертый пункт.

4. Выбор значения случайной величины x по полученному номеру i случайного события: $y_j = x_i$, $i = \overline{1,k}$, $j = \overline{1,n}$.

2.3. Программные средства для выполнения работы

Случайное число из равномерного в промежутке $(0,1)$ распределения моделируется с помощью функции **unifrnd.** Команда **y=unifrnd(a,b)** выдает случайное число из равномерного в промежутке (a,b) распределения.

Моделировать случайные числа (выборки) из дискретных распределений в Matlab позволяет функция **randsample**.

y=randsample(k,n) возвращает $(1\times n)$-вектор значений без повторений **y**, выбранных из дискретного распределения с равновозможными значениями $1,2,\ldots,k$. При отсутствии повторений **n** должно быть меньше или равно **k**.

Пример 2.1. Программа

y=(randsample(10,8))'

выдает результат

y =

4 3 6 9 7 10 8 2

y=randsample(population,n) возвращает **n** значений без повторений в **y**, выбранных из дискретного распределения с равновозможными значениями, определенными в **population.** При таком обращении к функции **randsample** ее параметры **population** и **y** являются символьными массивами (строками символов). Здесь также должно быть $n \le k$, где k – число символов в символьной строке **population.**

Пример 2.2. Программа

population='abcdefghij'

y=randsample(population,8)

выдает результат

population =

abcdefghij

y = edfcgiba

y=randsample(...,replace) возвращает выборку с повторениями, если параметр **replace** равен **true**, или без повторений, если параметр **replace** равен **false** (по умолчанию).

Пример 2.3. Программа

```
population='abcdefghij'
y=randsample(population,12,true)
```

выдает результат

```
population =
abcdefghij
y =
ebghdaehhjid
```

y=randsample(...,true,w) возвращает выборку с повторениями из распределения, заданного вектором вероятностей **w** и строкой значений **population**. Этот вариант обращения не поддерживает выбор без повторения.

Пример 2.4. Программа

```
population='abcde'
w=[0.1 0.3 0.5 0.02 0.08];
y=randsample(population,12,true,w)
```

выдает результат

```
population =
abcde
y =
ccceeabcbcac
```

Замечание. Использование функции **randsample** в виде

y=randsample(n,k,true,w)

позволяет моделировать выборки из произвольных дискретных распределений с числовыми значениями $1,2,\ldots,k$. При необходимости моделирования выборки из распределения с произвольными числовыми значениями $x_1, x_2, \ldots, x_k$ полученный вектор $y=(y_j)$ необходимо пересчитать в вектор $z=(z_j)$ по формуле $z_j = x_{y_j}$, $j=\overline{1,n}$.

2.4. Порядок выполнения работы

2.4.1. Написать m-файл-функцию для моделирования полной группы случайных событий. Входными параметрами этой функции должен быть вектор вероятностей случайных событий $p=(p_1, p_2,\ldots,p_k)$ и количество независимых испытаний n, а выходным – вектор смоделированной последовательности событий $e=(e_1,e_2,\ldots,e_n)$.

2.4.2. Использовать написанную в п. 2.4.1 функцию для моделирования последовательности независимых испытаний для случайных событий. В качестве вектора вероятностей случайных событий взять вектор p из табл. 2.1. По достаточно большой последовательности испытаний рассчитать оценки вероятностей этих событий.

2.4.3. Написать m-файл-функцию для моделирования выборки объема n из дискретного распределения с возможными значениями $x=(x_1,x_2,\ldots,x_k)$ и их вероятностями $p=(p_1, p_2,\ldots,p_k)$. Входными параметрами функции должны быть векторы $x=(x_1,x_2,\ldots,x_k)$, $p=(p_1, p_2,\ldots,p_k)$ и объем выборки n, а выходным – выборка возможных значений $y=(y_1, y_2,\ldots,y_n)$. В этой функции использовать функцию, написанную в п. 2.4.1.

2.4.4. Использовать написанную в п. 2.4.2 функцию для моделирования выборки из дискретного распределения, приведенного в табл. 2.1. По выборке достаточно большого объема рассчитать оценки среднего значения и дисперсии дискретного распределения и сравнить их с соответствующими теоретическими характеристиками.

2.4.5. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, смоделированные при их работе последовательности событий и чисел и рассчитанные значения, краткие выводы.

Таблица 2.1

Варианты заданий для моделирования случайных событий и дискретных случайных величин

№ варианта	Вектор вероятностей p, вектор значений x	№ варианта	Вектор вероятностей p, вектор значений x
1	$p=(0.1 \quad 0.6 \quad 0.3)$ $x=(1 \quad 6 \quad 3)$	11	$p=(0.2 \quad 0.3 \quad 0.5)$ $x=(0 \quad 1 \quad -1)$
2	$p=(0 \quad 0.5 \quad 0.5)$ $x=(-1 \quad 1 \quad 2)$	12	$p=(0.5 \quad 0.5 \quad 0)$ $x=(1 \quad 3 \quad 4)$
3	$p=(0.3 \quad 0.3 \quad 0.4)$ $x=(2 \quad 1 \quad -1)$	13	$p=(0.2 \quad 0.3 \quad 0.5)$ $x=(0 \quad 1 \quad -1)$
4	$p=(0 \quad 0.3 \quad 0.7)$ $x=(-1 \quad 0 \quad 1)$	14	$p=(0.3 \quad 0 \quad 0.7)$ $x=(-2 \quad -1 \quad 0)$
5	$p=(0.3 \quad 0.7 \quad 0)$ $x=(5 \quad 6 \quad 7)$	15	$p=(0.2 \quad 0.4 \quad 0.4)$ $x=(5 \quad 4 \quad 3)$
6	$p=(0.4 \quad 0.2 \quad 0.4)$ $x=(2 \quad 3 \quad 4)$	16	$p=(0.4 \quad 0.2 \quad 0.4)$ $x=(-3 \quad -2 \quad -1)$
7	$p=(0 \quad 0.4 \quad 0.6)$ $x=(0 \quad 1 \quad 2)$	17	$p=(0.2 \quad 0.3 \quad 0.5)$ $x=(-1 \quad 0 \quad 1)$
8	$p=(0 \quad 0.4 \quad 0.6)$ $x=(-1 \quad 0 \quad 3)$	18	$p=(0 \quad 0.2 \quad 0.8)$ $x=(1 \quad -1 \quad 0)$
9	$p=(0 \quad 0.2 \quad 0.8)$ $x=(1 \quad -1 \quad 0)$	19	$p=(0 \quad 0.2 \quad 0.8)$ $x=(1 \quad -1 \quad 0)$
10	$p=(0.1 \quad 0.7 \quad 0.2)$ $x=(-3 \quad -4 \quad -5)$	20	$p=(0.6 \quad 0 \quad 0.4)$ $x=(1 \quad 2 \quad 5)$

2.5. Контрольные вопросы

1. Что такое вероятностное пространство?
2. Что такое полная группа случайных событий?
3. Что такое случайная величина (определение случайной величины)?
4. Что такое закон распределения случайной величины?
5. Каким образом конструктивно задается закон распределения непрерывной, дискретной случайной величины?
6. Как найти среднее значение (математическое ожидание) и дисперсию случайной величины?
7. Как получить оценки среднего значения (математического ожидания) и дисперсии случайной величины?
8. Что такое моменты случайной величины?

Тема 3. Моделирование цепей Маркова

3.1. Цель работы

Приобретение навыков моделирования последовательностей случайных состояний, подчиненных марковской зависимости.

3.2. Теоретические положения

Будем рассматривать дискретную случайную последовательность ξ_i, $i = 0,1,2,...$, т.е. случайный процесс с дискретным множеством состояний $E = \{E_1, E_2,...,E_k\}$ или $E = \{E_1, E_2,...,E_k,...\}$ и дискретным временем $T = \{t_1, t_2,...,t_m,...\}$. Обычно при рассмотрении дискретной последовательности ξ_i говорят о некоторой системе, которая может находиться в одном из состояний $E = \{E_1, E_2,...,E_k,...\}$ и переходить из одного состояния в другое в дискретные моменты времени.

Определение. Дискретная случайная последовательность ξ_i, $i = 0,1,2,...$, называется цепью Маркова, если вероятность $p_{i,j}^{s,s+1}$ того, что в момент времени $s+1$ система будет находиться в состоянии E_j, зависит от того, в каком состоянии E_i система находилась в предыдущий момент времени s, и не зависит от того, в каких состояниях она находилась в более ранние моменты времени $s-1, s-2,...,0$:

$$p_{i,j}^{s,s+1} = P(\xi_{s+1} = E_j / \xi_s = E_i) = P(\xi_{s+1} = E_j / \xi_s = E_i, \xi_{s-1} = E_k,...,\xi_0 = E_l).$$

Вероятность

$$p_{i,j}^{s,s+1} = P(\xi_{s+1} = E_j / \xi_s = E_i)$$

– это условная вероятность того, что в момент времени $s+1$ система будет находиться в состоянии E_j при условии, что в предыдущий момент s она находилась в состоянии E_i. Эта вероятность называется вероятностью перехода из состояния E_i в состояние E_j за один шаг для моментов времени $s, s+1$.

Цепь Маркова называется конечной, если множество E ее состояний конечное.

Цепь Маркова называется однородной, если условная вероятность $p_{i,j}^{s,s+1}$ не зависит от момента времени s. В этом случае такая вероятность обозначается как $p_{i,j}$ и называется вероятностью перехода из состояния E_i в состояние E_j за один шаг. Мы будем рассматривать только однородные цепи Маркова.

Вероятности перехода $p_{i,j}$ образуют матрицу

$$P = (p_{i,j}),\ i, j = 1,2,\dots, \qquad (3.1)$$

которая называется матрицей вероятностей перехода однородной цепи Маркова. Элементы этой матрицы удовлетворяют следующим условиям:

$$0 \le p_{i,j} \le 1,$$

$$\sum_{j=1}^{\infty} p_{i,j} = 1,\ i = 1,2,\dots .$$

Первое условие является естественным свойством любой вероятности, а второе является следствием того, что система обязательно перейдет за один шаг в иное состояние или останется в прежнем. Это условие означает, что сумма элементов каждой строки матрицы вероятностей перехода P равна единице.

Пример 3.1. Урновая модель. Имеем две урны. В первой из них находятся 1 белый и 2 черных шара, во второй – 1 белый и 5 черных шаров. Начиная с первой урны, наугад вынимаем шар за шаром, причем если был вынут белый шар, то следующий шар вынимаем из первой урны, а если черный – то из второй. Каждый вынутый шар тут же возвращаем в урну, из которой он был вынут. Нас может интересовать, например, вероятность вынуть белый (или черный) шар на n-м шаге.

Описанная в данном примере последовательность испытаний образует однородную цепь Маркова с двумя состояниями: E_1 – вынутый шар белый и E_2 – вынутый шар черный. Вероятности перехода за один шаг имеют следующий смысл: $p_{1,1}$ – вероятность вынуть белый шар после предыдущего белого, т.е. вероятность вынуть белый шар из первой урны; $p_{1,2}$ – вероятность вынуть черный шар после предыдущего белого, т.е. вероятность вынуть черный шар из первой урны; $p_{2,1}$ – вероятность вынуть белый шар после предыдущего черного, т.е. вероятность вынуть белый шар из второй урны; $p_{2,2}$ – вероятность вынуть черный шар после предыдущего черного, т.е. вероятность вынуть черный шар из второй урны. Таким образом, матрица вероятностей перехода за один шаг имеет вид

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} \\ p_{2,1} & p_{2,2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{1}{6} & \frac{5}{6} \end{pmatrix}. \tag{3.2}$$

Пример 3.2. Случайное блуждание на прямой с поглощающими экранами. Пусть некоторая частица находится на действительной прямой и движется по ней под воздействием случайных толчков, которые возникают в моменты $t_0, t_1, \ldots$. Частица может находиться в точках $a, a+1, \ldots, a+m, \ldots, b$. В точках a и b размещаются поглощающие стенки (экраны). Из каждой точки, кроме точек a и b, частица перемещается в правую точку с вероятностью p и в левую с вероятностью $q = 1 - p$. При достижении стенок (точек a и b) частичка прилипает к ним.

В данном примере рассмотрена система, которая может находиться в одном из состояний $E_1 = a$, …, $E_k = b$. Если система находится в состоянии $E_1 = a$ или $E_k = b$, то она с вероятностью 1 остается в этих состояниях. Если же система находится в одном из промежуточных между a и b состояний $a + m$, то она с вероятностью p переходит в правое состояние $a + m + 1$ и с вероятностью

$q = 1 - p$ – в левое состояние $a + m - 1$. Матрица вероятностей перехода этой цепи Маркова имеет вид

$$P = \begin{pmatrix} 1\,0\,0\,0 \ldots 0 \\ q\;0\;p\;0 \ldots 0 \\ 0\;q\;0\;p \ldots 0 \\ \vdots \\ 0\,0\,0\,0 \ldots 1 \end{pmatrix}.$$

Состояния a и b системы называются поглощающими.

В данном примере нас может интересовать, например, вероятность прилипания частицы к стенке в точке b.

Этот пример имеет также иную интерпретацию. Некий игрок, играя с одним партнером, выигрывает и проигрывает определенную сумму с вероятностями p и $q = 1 - p$ соответственно. Суммарный капитал обоих игроков равен b. Игра продолжается до тех пор, пока капитал нашего игрока не уменьшится до нуля ($a = 0$) или не возрастет до b, т.е. до того времени, пока один из игроков не разорится. Нас может интересовать вероятность разорения нашего игрока и распределение вероятностей на протяжении игры. Такая интерпретация задачи случайных блужданий называется классической задачей о разорении.

3.3. Программные средства для выполнения работы

Для моделирования цепей Маркова необходимо, прежде всего, уметь моделировать случайные числа (выборки) из дискретных распределений. В Matlab это позволяет делать функция **randsample**, описание которой приведено в подразд. 2.3.

Для отображения на графике процесса смены состояний системы, описываемой цепью Маркова, можно воспользоваться функцией **stairs**:

stairs(y) строит лестничный (ступенчатый) график по значениям элементов вектора **y**;

stairs(x,y) строит лестничный график по значениям элементов вектора **y** в точках скачков, определенных в **x**. Значения **x** должны располагаться в возрастающем порядке.

Для построения графика смены состояний цепи Маркова необходимо в **stairs(y)** в качестве **y** использовать вектор последовательности состояний.

3.4. Порядок выполнения работы

3.4.1. Написать m-файл-функцию для моделирования состояний цепи Маркова. Входными параметрами этой функции выбрать вектор состояний, вектор начальных вероятностей, матрицу вероятностей перехода и число шагов (тактов). Использовать эту функцию для моделирования реализации состояний длительностью 50 – 100 тактов. Реализацию вывести в графическом окне в виде графика. Вариант описания цепи Маркова взять из табл. 3.1.

Указание. Начальное состояние моделируется с использованием вектора вероятностей начального состояния, а последующие – с использованием соответствующих векторов-строк матрицы вероятностей перехода.

3.4.2. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, смоделированную при их работе последовательность состояний цепи Маркова (реализацию случайной последовательности), краткие выводы.

Таблица 3.1

Варианты описаний цепей Маркова

№ варианта	Матрица вероятностей перехода P, вектор вероятностей начальных состояний A, вектор состояний E	№ варианта	Матрица вероятностей перехода P, вектор вероятностей начальных состояний A, вектор состояний E
1	2	3	4
1.	$P=\begin{pmatrix}0 & 0.5 & 0.5\\ 0.5 & 0 & 0.5\\ 0.5 & 0.5 & 0\end{pmatrix}$ $A=(0.1\quad 0.6\quad 0.3)$ $E=(1\quad 6\quad 3)$	11	$P=\begin{pmatrix}0.7 & 0.1 & 0.2\\ 0.2 & 0.5 & 0.3\\ 0 & 0.4 & 0.6\end{pmatrix}$ $A=(0.2\quad 0.3\quad 0.5)$ $E=(0\quad 1\quad -1)$
2.	$P=\begin{pmatrix}0.5 & 0.3 & 0.2\\ 0.4 & 0.4 & 0.2\\ 0 & 0.3 & 0.7\end{pmatrix}$ $A=(0\quad 0.5\quad 0.5)$ $E=(-1\quad 1\quad 2)$	12	$P=\begin{pmatrix}0.5 & 0.5 & 0\\ 0 & 0.5 & 0.5\\ 0.5 & 0 & 0.5\end{pmatrix}$ $A=(0.5\quad 0.5\quad 0)$ $E=(1\quad 3\quad 4)$
3.	$P=\begin{pmatrix}0.1 & 0.2 & 0.7\\ 0.3 & 0.6 & 0.1\\ 0 & 0.4 & 0.6\end{pmatrix}$ $A=(0.3\quad 0.3\quad 0.4)$ $E=(2\quad 1\quad -1)$	13	$P=\begin{pmatrix}0.4 & 0.2 & 0.4\\ 0.3 & 0.7 & 0\\ 0 & 0.2 & 0.8\end{pmatrix}$ $A=(0.2\quad 0.3\quad 0.5)$ $E=(0\quad 1\quad -1)$
4.	$P=\begin{pmatrix}0.2 & 0.4 & 0.4\\ 0.8 & 0.2 & 0\\ 0.7 & 0 & 0.3\end{pmatrix}$ $A=(0\quad 0.3\quad 0.7)$ $E=(-1\quad 0\quad 1)$	14	$P=\begin{pmatrix}0.6 & 0.4 & 0\\ 0 & 0.4 & 0.6\\ 0.3 & 0.3 & 0.4\end{pmatrix}$ $A=(0.3\quad 0\quad 0.7)$ $E=(-2\quad -1\quad 0)$

Продолжение табл. 3.1

1	2	3	4
5.	$P=\begin{pmatrix} 0.6 & 0 & 0.4 \\ 0.4 & 0.6 & 0 \\ 0.4 & 0.2 & 0.4 \end{pmatrix}$ $A=(0.3 \quad 0.7 \quad 0)$ $E=(5 \quad 6 \quad 7)$	15	$P=\begin{pmatrix} 0 & 0.6 & 0.4 \\ 0.4 & 0 & 0.6 \\ 0.2 & 0.2 & 0.6 \end{pmatrix}$ $A=(0.2 \quad 0.4 \quad 0.4)$ $E=(5 \quad 4 \quad 3)$
6.	$P=\begin{pmatrix} 0.1 & 0.5 & 0.4 \\ 0.4 & 0.2 & 0.4 \\ 0.2 & 0 & 0.8 \end{pmatrix}$ $A=(0.4 \quad 0.2 \quad 0.4)$ $E=(2 \quad 3 \quad 4)$	16	$P=\begin{pmatrix} 0.1 & 0.5 & 0.4 \\ 0.5 & 0.2 & 0.3 \\ 0.2 & 0.8 & 0 \end{pmatrix}$ $A=(0.4 \quad 0.2 \quad 0.4)$ $E=(-3 \quad -2 \quad -1)$
7.	$P=\begin{pmatrix} 0.2 & 0.8 & 0 \\ 0.2 & 0 & 0.8 \\ 0 & 0.2 & 0.8 \end{pmatrix}$ $A=(0 \quad 0.4 \quad 0.6)$ $E=(0 \quad 1 \quad 2)$	17	$P=\begin{pmatrix} 0.2 & 0 & 0.8 \\ 0 & 0.2 & 0.8 \\ 0.2 & 0.8 & 0 \end{pmatrix}$ $A=(0.2 \quad 0.3 \quad 0.5)$ $E=(-1 \quad 0 \quad 1)$
8.	$P=\begin{pmatrix} 0 & 0.2 & 0.8 \\ 0.2 & 0 & 0.8 \\ 0.2 & 0.8 & 0 \end{pmatrix}$ $A=(0 \quad 0.4 \quad 0.6)$ $E=(-1 \quad 0 \quad 3)$	18	$P=\begin{pmatrix} 0.4 & 0 & 0.6 \\ 0 & 0.4 & 0.6 \\ 0.4 & 0.6 & 0 \end{pmatrix}$ $A=(0 \quad 0.2 \quad 0.8)$ $E=(1 \quad -1 \quad 0)$
9.	$P=\begin{pmatrix} 0.1 & 0.4 & 0.5 \\ 0.4 & 0.5 & 0.1 \\ 0.2 & 0.8 & 0 \end{pmatrix}$ $A=(0 \quad 0.2 \quad 0.8)$ $E=(1 \quad -1 \quad 0)$	19	$P=\begin{pmatrix} 0.4 & 0.1 & 0.5 \\ 0.5 & 0.4 & 0.1 \\ 0.4 & 0.2 & 0.4 \end{pmatrix}$ $A=(0 \quad 0.2 \quad 0.8)$ $E=(1 \quad -1 \quad 0)$

Окончание табл. 3.1

1	2	3	4
10.	$P=\begin{pmatrix} 0.4 & 0 & 0.6 \\ 0 & 0.4 & 0.6 \\ 0.4 & 0.6 & 0 \end{pmatrix}$ $A=(0.1 \quad 0.7 \quad 0.2)$ $E=(-3 \quad -4 \quad -5)$	20	$P=\begin{pmatrix} 0.4 & 0 & 0.6 \\ 0 & 0.4 & 0.6 \\ 0.4 & 0.6 & 0 \end{pmatrix}$ $A=(0.6 \quad 0 \quad 0.4)$ $E=(1 \quad 2 \quad 5)$

3.5. Контрольные вопросы

1. Что такое дискретная случайная последовательность?
2. Что такое цепь Маркова (определение)?
3. Что такое матрица вероятностей перехода цепи Маркова и какими свойствами она обладает?
4. Какая цепь Маркова называется однородной?
5. Докажите адекватность полученной реализации ее математической модели (исходным данным).

Тема 4. Динамика вероятностей состояний цепей Маркова

4.1. Цель работы

Приобретение навыков расчета вероятностей перехода и безусловных вероятностей цепей Маркова для произвольного числа шагов.

4.2. Теоретические положения

Определение цепи Маркова приведено в подразд. 3.2. Сейчас нас будут интересовать вероятности перехода системы, которая образует однородную цепь Маркова, из состояния E_i в состояние E_j за n шагов. Обозначим эти вероятности как $p_{i,j}(n)$ и назовем матрицу

$$P_n = \left(p_{i,j}(n)\right),\ i, j = 1,2,\dots, \tag{4.1}$$

матрицей вероятностей перехода за n шагов. Понятно, что $P_1 = P$, где P – матрица вероятностей перехода за один шаг (4.1).

Рассмотрим переходы системы на протяжении последовательных n шагов. Пусть P_n – матрица вероятностей перехода за эти n шагов, P_m – матрица вероятностей перехода за первые m шагов, $m < n$, P_{n-m} – матрица вероятностей перехода за оставшиеся $n-m$ шагов. Тогда выполняется следующее равенство:

$$P_n = P_m P_{n-m},\ 0 < m < n. \tag{4.2}$$

Действительно, переход за n шагов возможен только через одно из состояний на m-м шаге. Поэтому по формуле полной вероятности получим

$$p_{i,j}(n) = \sum_{\nu=1}^{\infty} p_{i,\nu}(m)\, p_{\nu,j}(n-m).$$

Последнее выражение есть не что иное, как формула умножения матриц (4.2).

Равенство (4.2) называется уравнением Чепмена–Колмогорова.

Из уравнения (4.2) при $n = 2$ получим, что

$$P_2 = P_1 P_1 = P_1^2 = P^2.$$

При $n = 3$ находим

$$P_3 = P_1 P_2 = P_2 P_1 = P_1^3 = P^3.$$

Вообще, при любом n имеем

$$P_n = P^n, \tag{4.3}$$

т.е. матрица вероятностей перехода за n шагов равна n-й степени матрицы вероятностей перехода за один шаг.

Для цепи Маркова важно знать абсолютные (безусловные) вероятности состояния системы на любом n-м шаге

$$a_i(n) = P(\xi_n = E_i),\ i = 1,2,\dots .$$

Эти вероятности образуют матрицу-строку (вектор-строку) A_n^T безусловных вероятностей системы для момента времени n:

$$A_n^T = (a_1(n), a_2(n), \dots) = (a_i(n)),\ i = 1,2,\dots . \tag{4.4}$$

Элементы вектора A_n^T удовлетворяют при любом n очевидным условиям:

$$0 \le a_i(n) \le 1,$$

$$\sum_{i=1}^{\infty} a_i(n) = 1.$$

Для полного описания однородной цепи Маркова необходимо знать матрицу вероятностей перехода P и вектор безусловных вероятностей A_0^T для начального момента времени $n = 0$. Этих данных достаточно, чтобы найти вектор безусловных вероятностей A_n^T для любого n-го шага с помощью формулы

$$A_n^T = A_0^T P_n. \tag{4.5}$$

Действительно, попадание системы в состояние E_j на n-м шаге возможно при ее выходе в начальный момент времени из одного из своих состояний и переходе за n шагов в состояние E_j. В этом случае применима формула полной вероятности

$$a_j(n)=\sum_{\nu=1}^{\infty}a_\nu(0)\,p_{\nu,j}(n),\ \ j=1,2,\dots,$$

которая в векторно-матричной форме имеет вид (4.5).

В качестве примера рассмотрим урновую модель примера 5.1 (см. подразд. 3.2) и найдем вероятности вынуть белый и черный шары во втором испытании. Поскольку выбор шаров начинается с первой урны, то вектор безусловных вероятностей для начального испытания состоит из вероятностей вынуть белый и черный шары из первой урны:

$$A_0^T=\left(\frac{1}{3},\frac{2}{3}\right).$$

По формуле (4.3) найдем матрицу вероятностей перехода за два шага:

$$P_2=P^2=\begin{pmatrix}\frac{1}{3} & \frac{2}{3}\\ \frac{1}{6} & \frac{5}{6}\end{pmatrix}\begin{pmatrix}\frac{1}{3} & \frac{2}{3}\\ \frac{1}{6} & \frac{5}{6}\end{pmatrix}=\begin{pmatrix}\frac{2}{9} & \frac{7}{9}\\ \frac{7}{36} & \frac{29}{36}\end{pmatrix}.$$

Наконец, по формуле (4.5) найдем вектор безусловных вероятностей на втором шаге:

$$A_2^T=A_0^T P_2=\left(\frac{1}{3},\frac{2}{3}\right)\begin{pmatrix}\frac{1}{3} & \frac{2}{3}\\ \frac{1}{6} & \frac{5}{6}\end{pmatrix}=\left(\frac{11}{54},\frac{43}{54}\right).$$

Таким образом, при втором испытании (начиная с нулевого) мы с вероятностью $\frac{11}{54}$ вынем белый шар, и с вероятностью $\frac{43}{54}$ – черный.

4.3. Программные средства для выполнения работы

Matlab позволяет выполнять умножение матриц так же просто, как и скалярных величин. Так, команда **c=a*b** возвращает матрицу **c**, равную произведению матриц **a** и **b**. Возможно возведение квадратной матрицы в степень: **c=a^n**.

Для построения графиков изменения вероятностей в зависимости от числа шагов используется функция **plot** (см. п. 1.3.3).

4.4. Порядок выполнения работы

4.4.1. Рассчитать матрицы вероятностей перехода, векторы безусловных вероятностей и среднее состояние цепи Маркова для последовательности шагов $1 \ldots n$. Вариант описания цепи Маркова взять из табл. 3.1.

4.4.2. Построить графики изменения вероятностей первой строки матрицы вероятностей перехода, безусловных вероятностей и среднего состояния, рассчитанных в п. 4.4.1.

4.4.3. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, полученные численные и графические результаты, краткие выводы.

4.5. Контрольные вопросы

1. Запишите уравнение Чэпмена–Колмогорова для цепи Маркова.
2. Что такое безусловные вероятности состояний цепи Маркова?
3. Как рассчитывается вектор безусловных вероятностей цепи Маркова?
4. Как рассчитывается математическое ожидание (среднее состояние) цепи Маркова?
5. Опишите характер изменения графиков, полученных в п. 4.2.2.

Тема 5. Предельные вероятности состояний цепей Маркова

5.1. Цель работы

Приобретение навыков расчета предельных вероятностей цепей Маркова.

5.2. Теоретические положения

Часто нас интересует поведение системы, описываемой цепью Маркова, на достаточно длительном промежутке времени. Это значит, что нас интересует поведение вероятностей перехода цепи Маркова за n шагов $p_{i,j}(n)$ при $n \to \infty$. В некоторых случаях эти вероятности сходятся при $n \to \infty$ к некоторым предельным значениям, которые называются предельными вероятностями. Условия существования предельных вероятностей определяет следующая теорема.

Теорема 5.1 (Маркова). Если существует такое $s > 0$, что все $p_{i,j}(s) > 0$, то существуют такие числа p_j^*, $j = 1,2,\ldots,k$, что независимо от индекса i выполняются следующие соотношения:

$$\lim_{n\to\infty} p_{i,j}(n) = p_j^*,\ j = 1,2,\ldots,k, \tag{5.1}$$

$$\sum_{j=1}^{k} p_j^* = 1.$$

Физический смысл этой теоремы состоит в том, что вероятности $p_{i,j}(n)$ перехода из состояния E_i в состояние E_j за n шагов при $n \to \infty$ не зависят от состояния E_i, из которого был начат переход. Система как бы забывает о своем состоянии в далеком прошлом.

Пример 5.1. Применима ли теорема о предельных вероятностях к цепи Маркова с матрицей вероятностей перехода

$$P=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}?$$

Поскольку для такой цепи

$$P_{2m}=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix},\ P_{2m+1}=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix},\ m=1,2,\dots,$$

то для каждого s матрица P_s имеет нулевые элементы. Условия теоремы не выполняются, так что мы не можем утверждать, что предельные вероятности существуют.

Объединим предельные вероятности p_j^*, $j=1,2,\dots,k$, в вектор-строку предельных вероятностей

$$(p^*)^T=(p_1^*,p_2^*,\dots,p_k^*). \tag{5.2}$$

Тогда выражение (5.1) можно записать в следующей векторно-матричной форме:

$$\lim_{n\to\infty} P_n=\Pi^*, \tag{5.3}$$

где Π^* – $(k\times k)$-матрица предельных вероятностей. Все строки матрицы Π^* одинаковы и совпадают с вектором-строкой $(p^*)^T$ (5.2).

Теорема 5.2. Если для цепи Маркова существует вектор-столбец предельных вероятностей p^* (5.2), то он удовлетворяет следующей системе линейных алгебраических уравнений:

$$P^T p^*=p^*, \tag{5.4}$$

$$\sum_{j=1}^{k} p_j^*=1. \tag{5.5}$$

Действительно, запишем для цепи Маркова уравнение Чэпмена–Колмогорова (4.2) в виде

$$P_{n+1}=P_nP$$

и найдем предел обеих частей при $n\to\infty$. Поскольку при этом $P_{n+1}=\Pi^*$, $P_n=\Pi^*$, то получаем уравнение

$$\Pi^* = \Pi^* P,$$

из которого следует (5.4).

Рассмотрим также безусловные предельные вероятности

$$a_j^* = \lim_{n\to\infty} a_j(n),$$

образующие вектор-строку безусловных предельных вероятностей $(A^*)^T = (a_1^*, a_2^*, ..., a_k^*)$, так что

$$(A^*)^T = \lim_{n\to\infty} A_n. \tag{5.6}$$

Теорема 5.3. Если существует вектор предельных вероятностей P^* (5.2), то он является и вектором безусловных предельных вероятностей:

$$A^* = P^*. \tag{5.7}$$

Чтобы получить равенство (5.7), запишем соотношение (4.5), взяв предел от обеих его частей:

$$\lim_{n\to\infty} A_n^T = A_0^T \lim_{n\to\infty} P_n.$$

Учитывая обозначения пределов (5.3), (5.6), получим

$$(A^*)^T = A_0^T \Pi^*.$$

Поскольку $a_1(0) + a_2(0) + ... + a_k(0) = 1$ и матрица Π^* состоит из одинаковых строк, то легко понять, что $A_0^T \Pi^* = (P^*)^T$, и равенство (5.7) доказано.

Пример 5.2. Найти предельные вероятности для цепи Маркова из примера 3.1 подразд. 3.2 на урновую модель.

Поскольку матрица вероятностей перехода этой цепи (3.2) имеет вид

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} \\ p_{2,1} & p_{2,2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{1}{6} & \frac{5}{6} \end{pmatrix},$$

т.е. не содержит нулевых элементов, то эта цепь удовлетворяет теореме 5.1 и предельные вероятности существуют.

Система уравнений (5.4), (5.5) для предельных условных вероятностей в данном примере имеет вид

$$\frac{1}{3}p_1^* + \frac{1}{6}p_2^* = p_1^*,$$
$$\frac{2}{3}p_1^* + \frac{5}{6}p_2^* = p_2^*,$$
$$p_1^* + p_2^* = 1.$$

Поскольку первые два уравнения являются линейно зависимыми, то, отбрасывая первое из них, получим два уравнения с двумя неизвестными:

$$\frac{2}{3}p_1^* - \frac{1}{6}p_2^* = 0,$$
$$p_1^* + p_2^* = 1.$$

Отсюда получаем $p_1^* = 0{,}2$, $p_2^* = 0{,}8$. Такими же будут и безусловные предельные вероятности $a_1^* = 0{,}2$, $a_2^* = 0{,}8$. Это значит, что после продолжительного числа экспериментов мы будем на каждом шаге вынимать белый шар с вероятностью $a_1^* = 0{,}2$ и черный шар – с вероятностью $a_2^* = 0{,}8$.

5.3. Программные средства для выполнения работы

Для выполнения задания необходимо уметь решать систему линейных алгебраических уравнений, т.е. систему вида

$$ax = b,$$

где $a = (a_{i,j})$, $i, j = \overline{1,n}$ – квадратная матрица коэффициентов системы; $b = (b_i)$, $i = \overline{1,n}$ – вектор-столбец свободных членов; $x = (x_i)$, $i = \overline{1,n}$ – вектор-столбец неизвестных. В Matlab это можно сделать как методом обращения матрицы коэффициентов, т.е. с помощью команды **x=a^-1*b**, так и методом исключения Гаусса, т.е. с помощью команды левого деления **x=b\a**. Метод Гаусса является более предпочтительным с точки зрения быстродействия.

5.4. Порядок выполнения работы

5.4.1. Рассчитать предельные вероятности для цепи Маркова, описание которой взять из табл. 3.1. Убедиться в том, что полученные вероятности совпадают с вероятностями, полученными для той же цепи в теме 4.

5.4.2. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, рассчитанные при их работе предельные вероятности, краткие выводы.

5.5. Контрольные вопросы

1. Что такое безусловные и условные предельные вероятности состояний цепи Маркова?

2. Что такое матрица предельных вероятностей и вектор предельных вероятностей состояний цепи Маркова?

3. Как рассчитывается вектор предельных вероятностей состояний цепи Маркова?

4. Какие методы решения систем линейных алгебраических уравнений вы знаете?

Тема 6. Пуассоновский поток заявок в системах массового обслуживания

6.1. Цель работы

Приобретение навыков моделирования реализаций пуассоновского потока заявок в системах массового обслуживания, исследование его свойств и характеристик.

6.2. Теоретические положения

Потоком однородных событий (заявок) называется конечная или счетная последовательность τ_n случайных величин, определенная на одном и том же вероятностном пространстве, при условии, что в любой фиксированный интервал времени (a,b) с вероятностью 1 попадает конечное число этих величин.

Если фиксированный момент времени t совпадает сразу с r элементами последовательности τ_n, то будем говорить, что в момент t происходит r событий потока.

Если $\tau_n \le \tau_{n+1}$ – два упорядоченных элемента последовательности, то будем говорить, что в полуинтервале $[\tau_n, \tau_{n+1})$ происходит одно событие τ_n.

Поток заявок в системе массового обслуживания (СМО) называется простейшим, или пуассоновским, если он обладает свойствами *стационарности, ординарности, отсутствия последействия*.

Стационарность потока означает, что для любой группы из конечного числа n непересекающихся отрезков времени вероятность появления в них соответственно $k_1, k_2, \ldots, k_n$ заявок зависит только от этих чисел и длин указанных промежутков времени, но не зависит от их расположения на оси времени. В частности, вероятность p_k появления k заявок на отрезке $[T, T+t]$ не зависит от T и является функцией только k и t: $p_k = p_k(k,t)$.

Ординарность потока означает практическую невозможность появления двух или более заявок в один и тот же момент времени. В математической форме ординарность записывается следующим образом:

$$\frac{p_{>1}(h)}{h} \underset{h \to 0}{\to} 0,$$

или, иначе,

$$p_{>1}(h) = o(h), \qquad (6.1)$$

где $p_{>1}(h)$ – вероятность появления на отрезке длиной h двух и более заявок. Отсюда сразу следует, что для простейшего потока

$$p_{>1}(0) = 0. \qquad (6.2)$$

Отсутствие последействия состоит в том, что вероятность поступления k заявок в течение отрезка времени $[T, T+t]$ не зависит от того, сколько заявок и как поступали до этого отрезка. Иначе говоря, количества заявок (случайные величины), поступающие в непересекающиеся отрезки времени, независимы в совокупности.

Приведенные три свойства являются характеристическими. Другие свойства простейшего потока являются их следствием.

Прежде всего отметим, что для простейшего потока

$$\frac{p_1(h)}{h} \underset{h \to 0}{\to} \lambda,$$

или, иначе,

$$p_1(h) = \lambda h + o(h), \qquad (6.3)$$

где $\lambda = \text{const}$. Это значит, что вероятность поступления ровно одной заявки на отрезке времени длиной h пропорциональна длине этого отрезка. Из (6.3) следует, что

$$p_1(0) = 0. \qquad (6.4)$$

Величина λ в выражении (6.3) называется параметром простейшего потока. Физический смысл этого параметра – среднее число заявок, поступающих в единицу времени. Она называется интенсивностью простейшего потока заявок.

Параметр λ простейшего потока является также средней длиной отрезка времени между соседними заявками.

Вероятность отсутствия заявок на отрезке времени длиной t определяется выражением

$$p_0(t) = e^{-\lambda t}. \qquad (6.5)$$

Выражение (6.5) определяет также вероятность того, что случайный отрезок времени ξ между соседними заявками простейшего потока будет не меньше t (рис. 6.1):

$$P(\xi \geq t) = p_0(t) = e^{-\lambda t}.$$

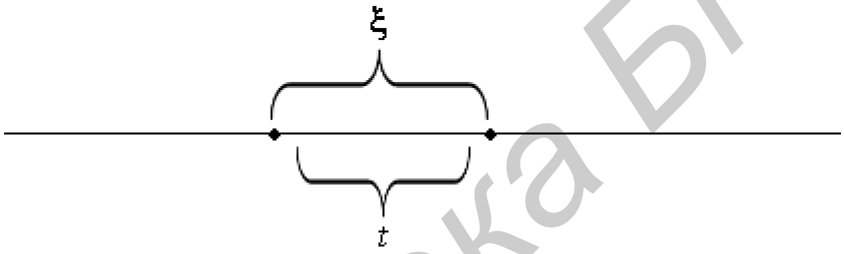

Рис. 6.1. К закону распределения отрезка времени между соседними заявками простейшего потока

Тогда

$$P(\xi < t) = F_\xi(t) = 1 - e^{-\lambda t}, \qquad (6.6)$$

а это есть функция распределения случайного отрезка времени ξ между соседними заявками простейшего потока. Плотность вероятности этого распределения равна

$$f_\xi(t) = \frac{d}{dt} F_\xi(t) = \lambda e^{-\lambda t}. \qquad (6.7)$$

Распределение вида (6.6), (6.7) называется экспоненциальным, т.е. отрезок времени ξ между соседними заявками простейшего потока является случайной величиной, распределенной по экспоненциальному закону $E(\lambda)$ (6.6).

Для вероятностей $p_k(t)$ простейшего потока справедлива система дифференциальных уравнений

$$\frac{dp_k(t)}{dt} = -\lambda p_k(t) + \lambda p_{k-1}(t),\ k = 1,2,\ldots; \tag{6.8}$$

$$\frac{dp_0(t)}{dt} = -\lambda p_0(t). \tag{6.9}$$

Как решение этих дифференциальных уравнений можно получить, что вероятность поступления k заявок в отрезке времени t для простейшего потока определяется выражением

$$p_k(t) = \frac{(\lambda t)k}{k!} e^{-\lambda t},\ k = 0,1,2,\ldots. \tag{6.10}$$

Формула (6.10) представляет собой известное распределение Пуассона с параметром λt. Таким образом, число заявок в отрезке времени t для простейшего потока подчиняется пуассоновскому распределению $\Pi(\lambda t)$.

Теорема 6.1. Если $\tau_1, \tau_2, \ldots, \tau_n, \ldots$ – моменты последовательных заявок простейшего потока, начиная с любого момента времени t_0, то случайный вектор $(\tau_1, \tau_2, \ldots, \tau_n)$ имеет плотность вероятности вида

$$\prod_{i=1}^{n} \left[\lambda e^{-\lambda(t_i - t_{i-1})}\right] = \lambda^n e^{-\lambda(t_n - t_0)}. \tag{6.11}$$

Последнее означает, что интервалы времени τ_1, $\tau_2 - \tau_1$, …, $\tau_n - \tau_{n-1}$ между последовательными заявками являются независимыми случайными величинами, распределенными по экспоненциальному закону с параметром λ.

Теорема 6.2. При условии, что число событий простейшего потока в интервале (a,b) равно n ($\xi = n$), моменты этих событий $\tau_1, \tau_2, \ldots, \tau_n$ независимы и равномерно распределены в интервале (a,b).

Теоремы 6.1 и 6.2 определяют два алгоритма моделирования простейшего потока заявок: 1) как независимых интервалов времени между заявками, распределенных по экспоненциальному закону (теорема 6.1); 2) как независимых моментов заявок, равномерно распределенных в интервале (a,b) (теорема 6.2).

Алгоритм 1

1. Определяем интервал (a,b), параметр λ.

2. Моделируем независимые последовательные интервалы времени $\Delta_1 = \tau_1 - a$, $\Delta_2 = \tau_2 - \tau_1$, …, распределенные по экспоненциальному закону с параметром λ, до тех пор, пока $\tau_m = a + \sum\limits_{i=1}^{m} \Delta_i \le b$, $m = 1,2,\ldots$, $\tau_0 = a$. Величины $\tau_1, \tau_2, \ldots$ здесь являются моментами появления заявок. Их число заранее не определено.

Алгоритм 2

1. Определяем интервал (a,b), параметр λ.

2. Моделируем число заявок n потока как случайное число из пуассоновского распределения (6.10) $\Pi(\lambda(b-a))$.

3. Моделируем n независимых случайных чисел $z_1, z_2, \ldots, z_n$ из равномерного в (a,b) распределения.

4. Сортируем случайные числа $z_1, z_2, \ldots, z_n$ в порядке возрастания, в результате чего получаем последовательные моменты времени $\tau_1, \tau_2, \ldots, \tau_n$ поступления заявок.

5. Рассчитываем интервалы времени между заявками $\Delta_1 = \tau_1 - a$, $\Delta_2 = \tau_2 - \tau_1$, … $\Delta_n = \tau_n - \tau_{n-1}$.

6.3. Программные средства для выполнения работы

Для выполнения задания необходимо уметь моделировать случайные числа из равномерного, экспоненциального и пуассоновского распределений.

Случайные числа из равномерного распределения $u(a,b)$, плотность вероятности которого имеет вид

$$f_\xi(x)=\begin{cases}\dfrac{1}{b-a}, & a<x<b,\ a,b\in R,\ a<b,\\ 0, & x\le a,\ x\ge b,\end{cases}$$

моделируются с помощью функции **y=unifrnd(a,b).**

Случайные числа из экспоненциального распределения $E(\lambda)$, плотность вероятности которого в Matlab определяется в виде

$$f_\xi(x)=\begin{cases}\lambda^{-1}e^{-\lambda^{-1}x}, & x\ge 0,\ \lambda>0,\\ 0, & x<0,\end{cases}$$

моделируются с помощью функции **y=exprnd(lambda).**

Дискретная случайная величина ξ называется распределенной по пуассоновскому закону, если она принимает значения из счетного множества $\{0,1,2,\ldots,m,\ldots\}$ с вероятностями

$$P(\xi=m)=\frac{a^m}{m!}e^{-a},\ m=0,1,2,\ldots,$$

где $a\ge 0$ – параметр распределения. Распределение Пуассона обозначается как $\Pi(a)$. Случайные числа из пуассоновского распределения $\Pi(a)$ моделируются с помощью функции **y=poissrnd(a).**

Для графического представления результатов моделирования используется функция **plot** (см. п. 1.3.3).

Для построения гистограммы распределения используется функция **hist.**

n=hist(y) распределяет элементы вектора **y** на 10 интервалов одинаковой длины и возвращает количество элементов, попавших в каждый интервал, в виде вектора **n**. Если **y** – матрица, то **hist** работает со столбцами.

n=hist(y,l), где **l –** скаляр, использует **l** интервалов одинаковой длины.

n=hist(y,x), где **x** – вектор, возвращает количество элементов вектора **y**, попавших в интервалы с центрами, заданными вектором **x**. Число интервалов в этом случае равно числу элементов вектора **x**.

[n,x]=hist(…) возвращает числа попаданий в интервалы (в векторе **n**), а также положения центров интервалов (в векторе **x**).

hist(…) строит гистограмму без возвращения параметров, т. е. строит прямоугольники высотой

$$h_i = m_i,$$

где m_i – число элементов, попавших в i-й интервал, $i = \overline{1, l}$.

6.4. Порядок выполнения работы

6.4.1. Выполнить моделирование реализаций простейшего потока двумя приведенными в подразд. 6.2 алгоритмами. Вывести графическую иллюстрацию потоков в виде нанесенных на ось абсцисс моментов появления заявок и концов промежутка (a, b). Параметры потока для моделирования взять из табл. 6.1.

6.4.2. Вывести гистограмму промежутка времени между соседними заявками потока и его плотность вероятности. Для согласования масштабов гистограммы и генеральной плотности вероятности необходимо генеральную плотность вероятности умножить на коэффициент

$$k = n\frac{\Delta_{(n)} - \Delta_{(1)}}{l},$$

где n – объем выборки; l – число интервалов разбиения выборки при построении гистограммы; $\Delta_{(1)}$, $\Delta_{(n)}$ – минимальный и максимальный из интервалов времени между заявками соответственно.

6.4.3. Получить оценку $\widehat{\lambda}$ параметра λ потока.

6.4.4. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, полученные при их работе графические и численные результаты, краткие выводы.

Таблица 6.1

Параметры случайных потоков для моделирования

№ варианта	Параметр λ, интервал (a,b)	№ варианта	Параметр λ, интервал (a,b)
1	$\lambda=1,\ a=0,\ b=100$	11	$\lambda=0{,}1,\ a=0,\ b=650$
2	$\lambda=2,\ a=0,\ b=90$	12	$\lambda=0{,}2,\ a=0,\ b=550$
3	$\lambda=3,\ a=0,\ b=80$	13	$\lambda=0{,}3,\ a=0,\ b=500$
4	$\lambda=4,\ a=0,\ b=70$	14	$\lambda=0{,}4,\ a=0,\ b=450$
5	$\lambda=5,\ a=0,\ b=60$	15	$\lambda=0{,}5,\ a=0,\ b=400$
6	$\lambda=6,\ a=0,\ b=50$	16	$\lambda=0{,}6,\ a=0,\ b=350$
7	$\lambda=7,\ a=0,\ b=40$	17	$\lambda=0{,}7,\ a=0,b=300$
8	$\lambda=8,\ a=0,\ b=30$	18	$\lambda=0{,}8,\ a=0,\ b=200$
9	$\lambda=9,\ a=0,\ b=25$	19	$\lambda=0{,}9,\ a=0,\ b=150$
10	$\lambda=8,\ a=0,\ b=20$	20	$\lambda=0{,}8,\ a=0,\ b=170$

6.5. Контрольные вопросы

1. Дайте определение пуассоновского потока заявок.

2. Каким образом математически формализуется свойство ординарности пуассоновского потока заявок?

3. В чем состоит физический смысл параметра λ пуассоновского потока заявок?

4. Какие вероятностные распределения связаны с пуассоновским потоком заявок?

5. Поясните два способа получения оценки $\hat{\lambda}$ параметра λ пуассоновского потока заявок.

6. Приведите графический пример реализации пуассоновского потока требований как случайного процесса.

Тема 7. Случайный двоичный сигнал

7.1. Цель работы

Изучение цепей Маркова с непрерывным временем на примере случайного двоичного сигнала.

7.2. Теоретические положения

Рассмотрим конечную цепь Маркова с непрерывным временем и двумя состояниями E_1 и E_2. Пусть эти состояния имеют численные значения $E_1 = -1$ и $E_2 = 1$, причем вероятность перехода за малое время Δt в другое состояние определяется условием

$$p_{1,2}(\Delta t) = p_{2,1}(\Delta t) = \lambda \Delta t + o(\Delta t). \tag{7.1}$$

Такой процесс называется случайным телеграфным или случайным двоичным сигналом. Одна из реализаций этого процесса приведена на рис. 7.1.

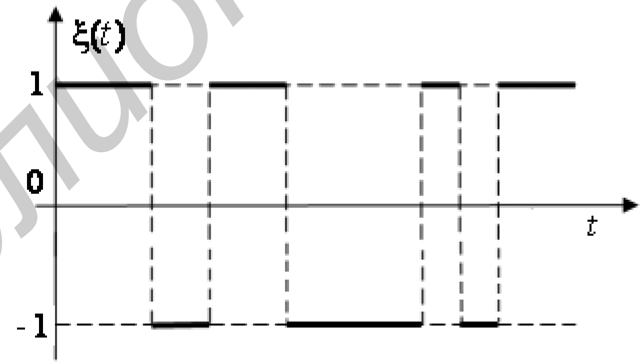

Рис. 7.1. Реализация случайного телеграфного сигнала

Условие (7.1) означает, что инфинитезимальная матрица процесса имеет следующий вид:

$$Q = \begin{bmatrix} -\lambda & \lambda \\ \lambda & -\lambda \end{bmatrix}. \quad (7.2)$$

Матрица вероятностей перехода $P(t)$ имеет размер 2х2:

$$P(t) = \begin{bmatrix} p_{1,1}(t) & p_{1,2}(t) \\ p_{2,1}(t) & p_{2,2}(t) \end{bmatrix}.$$

Найдём эту матрицу как решение прямой системы дифференциальных уравнений

$$P'(t) = P(t)Q$$

с начальными условиями

$$P(0) = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Известно, что решением этих уравнений является матричная экспонента, которая определятся как следующий ряд:

$$P(t) = P(0)e^{Qt} = I(I + Qt + \frac{1}{2!}(Qt)^2 + \frac{1}{3!}(Qt)^3 + \ldots). \quad (7.3)$$

Отсюда получаем, что

$$p_{1,1}(t) = p_{2,2}(t) = \frac{1}{2} + \frac{1}{2}e^{-2\lambda t},$$

$$p_{1,2}(t) = p_{2,1}(t) = \frac{1}{2} - \frac{1}{2}e^{-2\lambda t}. \quad (7.4)$$

Чтобы убедиться в справедливости последних формул, необходимо разложить эти функции в ряды Тейлора в окрестности нуля и сравнить их с рядом (7.3) для матрицы $P(t)$.

Для случайного телеграфного сигнала существуют предельные вероятности

$$p_j^* = \lim_{t \to \infty} p_{i,j}(t).$$

Из формул (7.4) при $t \to \infty$ получаем

$$p_1^* = p_2^* = \frac{1}{2}.$$

Эти вероятности будут для больших моментов времени и безусловными:

$$a_1^* = P(\xi(t)=1) = \frac{1}{2},$$

$$a_2^* = P(\xi(t)=1) = \frac{1}{2}.$$

Теперь можно найти математическое ожидание и ковариационную функцию телеграфного сигнала. Для больших моментов времени (для стационарного состояния) получим

$$E(\xi(t)) = 1 \cdot P(\xi(t)=1) - 1 \cdot P(\xi(t)=-1) = \frac{1}{2} - \frac{1}{2} = 0, \quad (7.5)$$

$$R_\xi(\tau) = E\big(\xi(t)\xi(t+\tau)\big) =$$

$$= 1 \cdot 1 \cdot a_1^* \cdot p_{1,1}(\tau) + 1 \cdot (-1) \cdot a_1^* \cdot p_{1,2}(\tau) + (-1) \cdot 1 \cdot a_2^* \cdot p_{2,1}(\tau) + (-1) \cdot (-1) \cdot a_2^* \cdot p_{2,2}(\tau) =$$

$$= \frac{1}{4} + \frac{1}{4}e^{-2\lambda\tau} - \left(\frac{1}{4} - \frac{1}{4}e^{-2\lambda\tau}\right) - \left(\frac{1}{4} - \frac{1}{4}e^{-2\lambda\tau}\right) + \frac{1}{4} + \frac{1}{4}e^{-2\lambda\tau} = e^{-2\lambda\tau}, \ \tau > 0. \quad (7.6)$$

Учитывая, что ковариационная функция является четной, получаем выражение ковариационной функции для любых τ:

$$R_\xi(\tau) = e^{-2\lambda|\tau|}.$$

7.3. Программные средства для выполнения работы

Для автоматизации расчетов удобно пользоваться функцией **expm.**

y=expm(x) вычисляет матричную экспоненту e^x, где x – $(n \times n)$-матрица-аргумент; y – $(n \times n)$-матрица-функция.

Пример 7.1. Программа

```
clc
clear
x=[-1,2;3,-2];
y=expm(x)
```

выдает результат

```
y =
   1.6383    1.0800
   1.6200    1.0983
```

Если **x** – символьная $(n \times n)$-матрица, то **y=expm(x)** вычисляет $(n \times n)$-матрицу **y** в символьном виде.

Пример 7.2. Программа

```
clc
clear
syms t
syms lam mu positiv
x=[-lam,lam;mu,-mu];
y=expm(x*t)
```

выдает результат

```
y =
[ (mu+lam*exp(-t*lam-t*mu))/(lam+mu), -lam*(exp(-t*lam-t*mu)-1)/(lam+mu)]
[  -mu*(exp(-t*lam-t*mu)-1)/(lam+mu), (lam+mu*exp(-t*lam-t*mu))/(lam+mu)]
```

7.4. Порядок выполнения работы

7.4.1. Предложить и запрограммировать алгоритм моделирования реализаций случайного двоичного сигнала, описанного выше.

7.4.2. Рассчитать среднее значение и ковариационную функцию для стационарного состояния описанного выше сигнала в предположении, что состояния E_1 и E_2 принимают различные числовые значения c_1 и c_2, и вероятности перехода за малое время Δt в другое состояние определяются условиями

$$p_{1,2}(\Delta t) = \lambda \Delta t + o(\Delta t); \quad p_{2,1}(\Delta t) = \mu \Delta t + o(\Delta t),$$

т.е. инфинитезимальная матрица имеет вид

$$Q = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}. \qquad (7.7)$$

7.4.3. Оформить полученные результаты в виде отчета. Отчет должен содержать титульный лист с темой практического занятия, номером варианта и данными о студенте (ФИО, номер группы), исходные данные к варианту задания, коды написанных программ, полученные при их работе графические и численные результаты, краткие выводы.

7.5. Контрольные вопросы

1. Что такое матричная экспонента?

2. Запишите аналоги выражений (7.5), (7.6) для расчета математического ожидания и ковариационной функции описанного в работе сигнала в предположении, что состояния E_1 и E_2 принимают различные числовые значения c_1 и c_2 и инфинитезимальная матрица имеет вид (7.2).

3. Запишите матричную экспоненту (7.3) для случая инфинитезимальной матрицы (7.7) и выпишите ее элементы.

Литература

1. Гнеденко, Б. В. Курс теории вероятностей. – 10-е изд., доп. / Б. В. Гнеденко. – М. : URSS, 2011. – 488 с.

2. Тихонов, В. И. Марковские процессы / В. И. Тихонов, В. А. Миронов. – М. : Сов. радио, 1977. – 488 с.

3. Коваленко, И. Н. Теория вероятностей и математическая статистика / И. Н. Коваленко, А. А. Филиппова. – М. : Высш. шк., 1982. – 256 с.

4. Гнеденко, Б. В. Введение в теорию массового обслуживания. – 5-е изд., испр. / Б. В. Гнеденко, И. Н. Коваленко.– М. : URSS, 2010. – 400 с.

5. Маталыцкий, М. А. Теория массового обслуживания и ее применения / М. А. Маталыцкий, О. М. Тихоненко, А. В. Паньков. – Гродно: ГрГУ, 2008. – 771 с.

6. Боровков, А. А. Теория вероятностей. – 5-е изд. / А. А. Боровков. – М. : URSS, 2009. – 656 с.

7. Крамер, Г. Стационарные случайные процессы / Г. Крамер, М. Лидбеттер. – М. : Мир, 1969. – 398 с.

8. Острем, К. Ю. Введение в стохастическую теорию управления / К. Ю. Острем. – М. : Мир, 1973. – 324 с.

9. Чистяков, В. П. Курс теории вероятностей / В. П. Чистяков. – М. : Дрофа, 2007. – 254 с.

10. Муха, В. С. Теория вероятностей / В. С. Муха. – Минск : БГУИР, 2001. – 167 с.

11. Муха, В. С. Методическое пособие по курсу «Вероятностные процессы в АСУ». Ч. 1 / В. С. Муха. – Минск : МРТИ, 1987. – 60 с.

12. Муха, В. С. Методическое пособие по курсу «Вероятностные процессы в АСУ». Ч. 2 / В. С. Муха. – Минск : МРТИ, 1987. – 60 с.

13. Харин, Ю. С. Практикум на ЭВМ по математической статистике / Ю. С. Харин, М. Д. Степанова. – Минск : Университетское, 1987. – 304 с.

14. Дьяконов, В. П. Матлаб 5.0/5.3. Система символьной математики / В. П. Дьяконов, И. В. Абраменкова. – М. : Нолидж, 1999. – 640 с.

15. Кетков, Ю. Л. Matlab 6.x: программирование численных методов / Ю. Л. Кетков, А. Ю. Кетков, М. М. Шульц. – СПб. : БХВ-Петербург, 2004. – 672 с.

Содержание

Св. план 2012, поз. 23

Учебное издание

Муха Владимир Степанович

СЛУЧАЙНЫЕ ПРОЦЕССЫ. ПРАКТИКУМ

Пособие

Редактор Т. Н. Крюкова
Корректор Е. Н. Батурчик
Компьютерная верстка Ю. Ч. Клочкевич

Подписано в печать 19.06.2012.	Формат 60x84 1/16.	Бумага офсетная.
Гарнитура «Таймс».	Отпечатано на ризографе.	Усл. печ. л.
Уч.-изд. л. 3,0.	Тираж 100 экз.	Заказ 49.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6