Л. ЛАБОРАТОРНЫЕ РАБОТЫ ПО КУРСУ «МОДЕЛИ НАДЕЖНОСТИ АСОИУ»

1. Имитационные модели МП
2. Невосстанавливаемые системы

3. Аналмтика восстанавливаемых систем

4. Расчет над ПО

Л.1. Марковские цепи

Задание 1

Написать программу генерации выборочных траекторий процессов состояний Марковской цепи со случайной матрицей переходных вероятностей.

Для различных ограничений на длину реализации построить графики:

- оценок частот пребывания в каждом состоянии;
- гистограмму частот пребывания в состояниях.

Найти теоретические значения стационарных вероятностей МЦ. Сравнить статистические и теоретические значения.

Теоретические положения 1

Обозначим через $\xi=(C,\Pi,P)$ однородную Марковскую цепь с множеством состояний $C=\{c_i\}$, с вектор-строкой начальных вероятностей $\Pi=\|p_i\|$ и матрицей переходных вероятностей $P=\|p_{ij}\|$

$p_{ij}\geq 0$, $\Sigma_j\, p_{ij}=1$, $p_{ij}=P\{\xi_1=j|\xi_0=i\}= ... =P\{\xi_n=j|\xi_{n-1}=i\}$,

Матрицей переходных вероятностей за k шагов называется матрица

$P^{(k)}=\|p_{ij}^{(k)}\|$, где $p_{ij}^{(k)}=P\{\xi_k=j|\xi_0=i\}$

Состоянием цепи в момент k называется вектор-строка вероятностей

$\Pi^{(k)}=\|p_i^{(k)}\|$, $p_i\geq 0$, $\Sigma_i\, p_i=1$,

где $p_i^{(k)}=P\{\xi_k=i\}$ - вероятность нахождения цепи в момент времени k в состоянии i.

Переходную вероятность также называют одношаговой переходной, поскольку она описывает изменение состояния системы между последовательными моментами времени t_{n-1} и t_n.

Двухшаговая вероятность перехода определяется как вероятность того, что система S, находясь в состоянии e_i, переходит в состояние e_j в результате двух последующих испытаний, где переход из e_i в e_j за два шага может пройти через некоторое промежуточное состояние e_q $(q=1,2,\ldots,n)$. Например, из e_i система может перейти в состояние e_1 , а затем уже в e_j , или из e_i перейти сначала в e_q , а затем в e_j и т.д. по схеме «или»:

$$p_{ij}^{(2)} = \sum_{q=1}^{n} p_{iq} p_{qj}$$

В общем случае $p_{ij}^{(m)}$ - m-шаговая вероятность перехода есть вероятность p_{ij} того, что система, находясь в состоянии e_i в результате m последующих шагов оказывается в состоянии e_j.

Стационарная вероятность состояния МЦ - это предельная доля пребывания МЦ в данном состоянии.

Пример 1

Наброски программы оценки стационарных вероятностей (MarC_Sim_Com.m)

```
% Марковская цепь
clear all;    n=5;  N=100;
% Формирование случайной матрицы переходных вероятностей MP
MP1=rand(n,n);   SMP1=sum(MP1);
for i=1:1:n      MP(:,i)=MP1(:,i)/SMP1(i);  end;  MP=MP';  MPW=MP;
for i=2:1:n     MPW(:,i)=MPW(:,i)+MPW(:,i-1);  end; % MPW - матрица сумм
SS=zeros(n,N);  S1=1;  SS(1,1)=1;  % сброс статистики
% цикл повторных реализаций марковской цепи
for i=1:1:N
   R=rand; S2=0;  j=1;
   while S2==0
      if R<=MPW(S1,j)  S2=j;    end;
       j=j+1;
   end
   S1=S2;      SS(S1,i)=1;
end
% Статистические значения стационарных вероятностей
for i=2:1:N     SS(:,i)=SS(:,i)+SS(:,i-1);  end;
for i=2:1:N     SS(:,i)=SS(:,i)/i;  end;
figure('Position',[100 200 500 300]);
t=1:1:N; figure(1);  plot(t, SS, 'LineWidth',3);
figure('Position',[100 200 500 300]);
figure(2);  histfit(SS(1,:),100);
%Теоретические значения стационарных вероятностей
```

```
MP_th=MP-eye(5,5);  for i=1:1:n  MP_th(i,5)=1; end;
P_th=[0 0 0 0 1]*MP_th^(-1);  % решение системы
P_all=[P_th' SS(:,N)]  % результат сравнительного анализа
```

Результаты моделирования

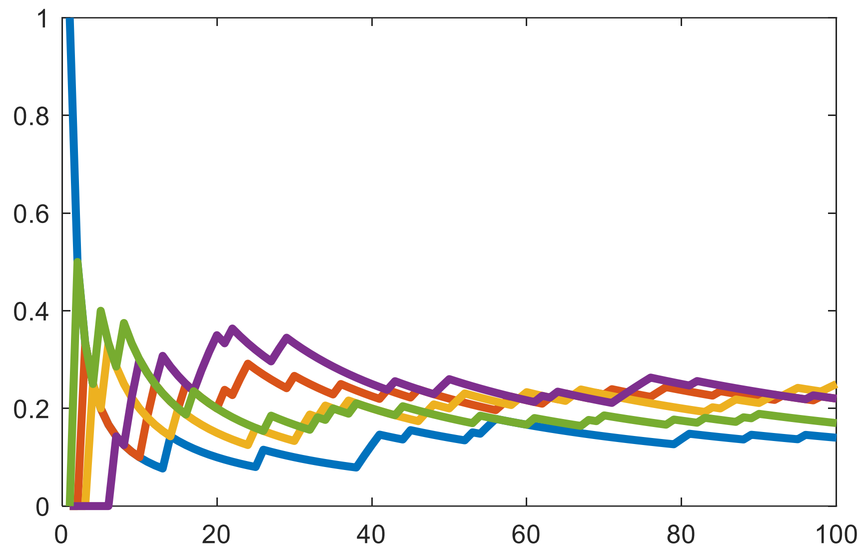


Рисунок Л.1. - Выборочные траектории частот состояний МЦ

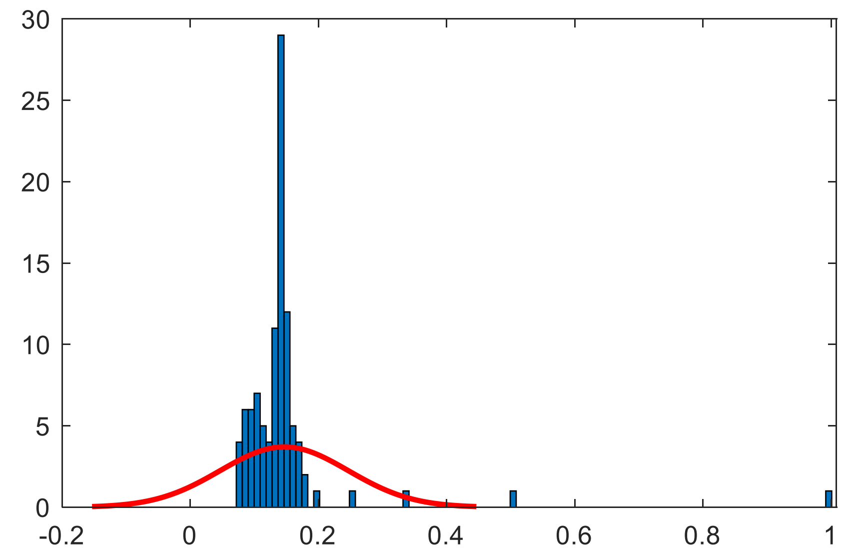


Рисунок Л.2. - Гистограмма распределения состояний МЦ

Таблица Л.1. - Теоретические и статистические значения стационарных вероятностей

Теоретические	Стат. N=1000		Теоретические	Стат. N=100000
0.3330	0.3087		0.1746	0.1747

0.1340	0.1500		0.2283	0.2282
0.1930	0.1697		0.2233	0.2230
0.1500	0.1570		0.1869	0.1868
0.1910	0.2146		0.1869	0.1873

Задание 2

Написать программу расчета вероятностей перехода в поглощающие состояния (на примере задачи о разорения в виде модели МЦ случайного блуждания с поглощающими экранами).

Для различных значений общего капитала и различных распределений начального капитала построить графики вероятности разорения одного из игроков.

Пример 2

Наброски программы (MarC_Sim_Bluz.m)

```
% Марковская цепь с безвозвратными состояниями
% "задача о разорении игрока"
clear all;
K=11;       % размерность цепи
p=0.5;  q=1-p; % вероятности переходов в соседние состояния
x0=6;        % начальное состояние (начальный капитал)
N=10000;     % количество повторных реализаций
% формирование матрицы переходных вероятностей
MP=zeros(K,K); MP(1,1)=1; MP(K,K)=1;
for i=2:1:K-1 MP(i,i-1)=q; MP(i,i+1)=p;  end;
N1=0; NK=0; % Обнуление результатов статистики
% Цикл повторных разыгрываний цепи
for n=1:1:N
x=x0;
while (x~=1 & x~=K)
if rand>q  x=x+1;  else  x=x-1; end;
end
if x==1 N1=N1+1; else NK=NK+1; end;
end;
PP1=N1/N; % результат - вероятность разорения
c=eig(MP);  % собственные числа
        Собственные значения
        1.0000
        1.0000
        -0.951
        -0.809
```

```
-0.587
-0.309
-2.264
0.9510
0.8090
0.3090
0.5877
```

Задание 3

Написать программу генерации матрицы переходных вероятностей для Марковской цепи с двумя и тремя поглощающими состояниями. Рассчитать собственные значения матрицы переходных вероятностей. Сделать вывод о количестве единичных собственных значений

Теоретические положения 3

e=eig(A) возвращает вектор-столбец, содержащий собственные значения квадратной матрицы A;

[V,D]=eig(A) возвращает диагональный матричный D из собственных значений и матричного V чьи столбцы являются соответствующими правыми собственными векторами, так, чтобы A*V = V*D;

[V,D,W]=eig(A) возвращает полный матричный W чьи столбцы являются соответствующими левыми собственными векторами, так, чтобы W'*A = D*W'.

Задача о собственных значениях должна определить решение уравнения $A\,v = \lambda\,v$, где A является n- n матрица, v является вектор-столбцом длины n, и λ является скаляром. Значения λ, которые удовлетворяют уравнению, являются собственными значениями. Соответствующие значения v, которые удовлетворяют уравнению, являются правыми собственными векторами. Левые собственные вектора, w, удовлетворяют уравнению $w\,'A = \lambda\,w'$.

Пример 3

Наброски программы (MarC_Sim_Pogl.m)

```
% Марковская цепь с безвозвратными состояниями
clear all;    n1=5; n2=7;
MP1=rand(n1,n1);   SMP1=sum(MP1);
for i=1:1:n1      MP1(:,i)=MP1(:,i)/SMP1(i);  end;  MP1=MP1';
MP2=rand(n2,n2);   SMP2=sum(MP2);
for i=1:1:n2      MP2(:,i)=MP2(:,i)/SMP2(i);  end;  MP2=MP2';

SS=zeros(n1+n2, n1+n2);
SS(1:n1,1:n1)=MP1;
SS(n1+1:n1+n2, n1+1:n1+n2)=MP2;
 e=eig(SS);
0.999999999999999 + 0.00000000000000i
-0.286389780032213 + 0.00000000000000i
0.389404947933979 + 0.00000000000000i
0.0709570848864506 + 0.0463531749905185i
0.0709570848864506 - 0.0463531749905185i
0.999999999999999 + 0.00000000000000i
0.0291293938834328 + 0.209383535991948i
0.0291293938834328 - 0.209383535991948i
-0.0781613562833926 + 0.00000000000000i
0.0418966356369767 + 0.00000000000000i
0.0632424424806436 + 0.0523989973251727i
0.0632424424806436 - 0.0523989973251727i
```

Вопросы по лабораторной

Л.2. Марковский процесс с дискретными состояниями

Задание

Расчет переходного режима непрерывного Марковского процесса на основе имитационного моделирования.

Разработать имитационную модель по данным переходов между состояниями для своего варианта. Построить графики переходных режимов для различных начальных состояний. Найти стационарные вероятности.

Вариант задания включает матрицу интенсивности переходов **(MPr.xlsx)** и соответствующий граф переходов (**Схемы ОТН.docx**).

Таблица Л.2. - Вариант задания для непрерывного МП

Вариант	Схема	Матрица				
Nвар	Ncx		1	2		9
		1	λ_{11}	λ_{12}		λ_{19}
		2	λ_{21}	λ_{22}		λ_{29}
		9	λ_{91}	λ_{92}		λ_{99}

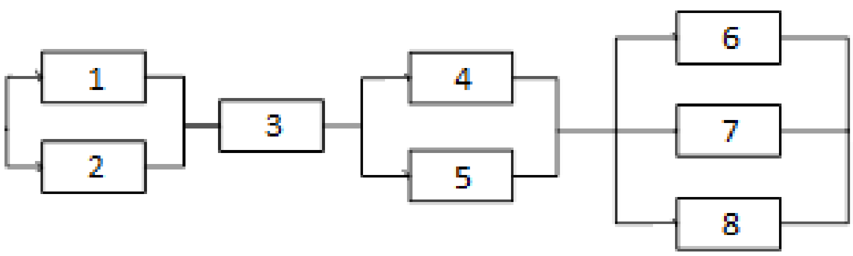


Рисунок Л.3. - Пример графа переходов

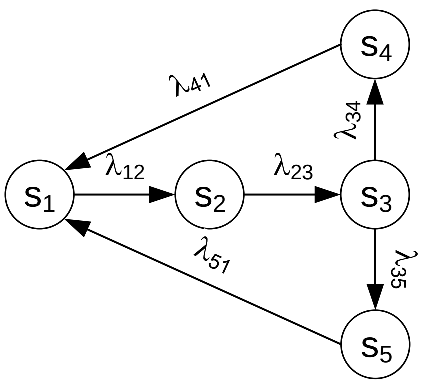
В качестве начального состояния выбирается первое (1). Интенсивности переходов, которых нет на диаграмме переходов не использовать или сделать нулевыми.

Теоретические положения

Пример

Пусть нормально работающая система ПА (состояние e_0) подвергается простейшему потоку отказов с интенсивностью λ_{01} переходя в новое состояние e_1, в котором она некоторое время может работать с необнаруженным отказом. Как только отказ обнаруживается (интенсивность обнаружения λ_{12}), производится осмотр ПА (состояние e_2). В результате осмотра, ПА либо направляется в ремонт (состояние e_3) с интенсивностью λ_{23}, либо списывается и заменяется новым (состояние e_4) с интенсивностью λ_{24}. Из состояния e_3 с интенсивностью λ_{30} и из состояния e_4 с интенсивностью λ_{40} ПА переходит в рабочее состояние e_0. Найти распределение вероятностей состояний для любого момента времени и финальные вероятности состояний.

Решение: Марковский процесс с дискретными состояниями иллюстрируется с помощью размеченного графа состояний. Граф состояний для сформулированной задачи приведен на рисунке.



Написать программу имитационной модели для генерации выборочных траекторий Марковского процесса с заданными интенсивностями переходов. Найти стационарные вероятности пребывания в каждом состоянии

Для определенности придадим параметрам следующие значения:

λ_{12}=0.5, λ_{23}=2, λ_{34}=1.5, λ_{35}=1.5, λ_{41}=0.8, λ_{51}=2.

Имитационная модель оценки переходных режимов (MarP_Sim_Ex.m)

```
% Непрерывная марковская цепь
clear all;    Ns=5;  N=500000;
MP=[0     0.5  0   0     0 ;
        0     0     2   0     0;
        0     0     0   1.5  1.5;
        0.8  0     0   0     0;
        2     0     0   0     0];
SS=zeros(N+1,3);  % Время                 Состояние     Пребывание
Tsys=0;                        SS(1,1)=Tsys;    SS(1,2)=1;      SS(1,3)=1;
for t=1:1:N
    Smin=0; Tmin=10^20; Sw=SS(t,2);
    for s=1:1:Ns
        if MP(Sw,s)==0 T=10^20;     else T=random('exp',(1/MP(Sw,s)),1,1); end;
        if T<Tmin  Smin=s; Tmin=T;   end;
    end
     SS(t,3)=Tmin;  Tsys=Tsys+Tmin;  SS(t+1,1)=Tsys; SS(t+1,2)=Smin;
end
TTT=sum(SS(:,3));  Tstat=zeros(Ns,1);  %  Обработка статистики
for t=1:1:N
  Sw=SS(t,2);  Tstat(Sw)=Tstat(Sw)+SS(t,3);
end
Tstat=Tstat/Tsys;
```

Результат расчета

```
0.539021625699355
0.135479766050578
0.0903043676489653
0.167406685944481
0.0677875546566019
```

Кроме стационарных вероятностей построить графики выборочных траекторий процесса изменения состояний и относительной доли пребывания в каждом состоянии.

Типовые вопросы

1. В чем различие классического и статистического определения вероятности события?
2. Как определяется безусловная и условная вероятность события.
3. Какие события называются совместными и несовместными? Приведите примеры.

4. Какие два события можно назвать зависимыми, какие - независимыми? Приведите примеры.
5. Какие матрицы называются стохастическими?
6. Сформулируйте определение собственных чисел и собственных векторов матрицы.
7. Дайте определение спектра матрицы.
8. Приведите общий вид характеристического уравнения.
9. Какой процесс называют случайным? Приведите примеры.
10. В чем состоит Марковское свойство случайного процесса?
11. Какие разновидности Марковского процесса вы знаете?
12. Какие процессы носят название Марковских цепей?
13. Алгебраическая и геометрическая интерпретация Марковских цепей.
14. В чем смысл элементов переходной матрицы?
15. Как определить переходную матрицу спустя m шагов?
16. Для чего задается вектор начального состояния системы?
17. Как определить вектор состояния системы спустя m шагов?
18. Какие компоненты включает размеченный граф стохастической системы?

Л.3. Подбор распределения модели надежности и оценка параметров

Задание 1.

Подобрать распределение и найти его параметры.

Теоретические положения

Функция распределение Вейбулла

$$f(x)=a\, b^{-e}\, x^{e-1}\, e^{-(n/b)},\ x>0. \qquad (Л.1)$$

где a — параметр масштаба; b — параметр формы.

Характеристики распределение Вейбулла

$$M\xi=\frac{b/a}{\Gamma(1/a)},\ D\xi=\frac{b^2}{a}\left[2\Gamma(2/a)-\frac{1}{a}\Gamma(1/a)\right]^2. \qquad (Л.2)$$

Плотность распределения Вейбулла

$$f(t)=\frac{b}{a}\left(\frac{t}{a}\right)^{b-1}\boldsymbol{exp}\left(-\left(\frac{t}{a}\right)^{b}\right). \qquad (Л.3)$$

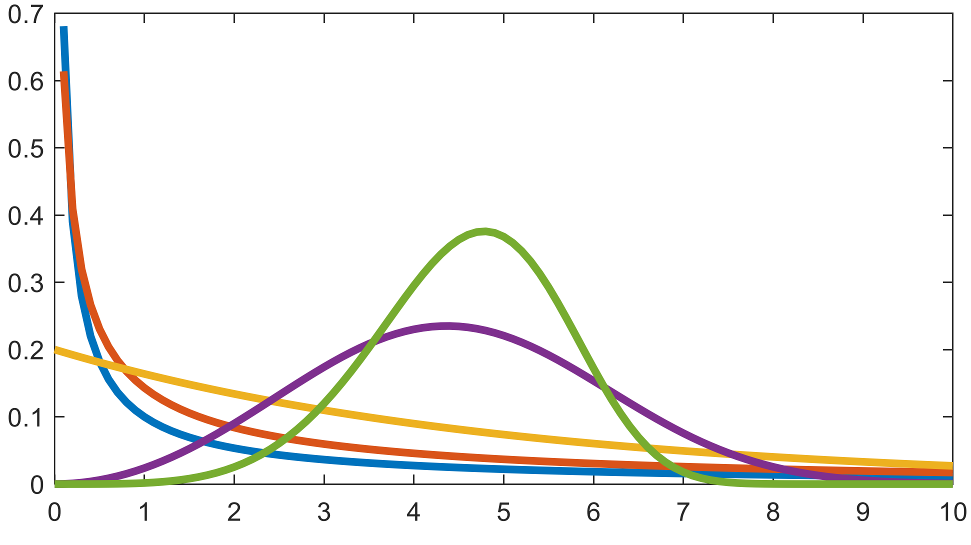


Рисунок Л.4. - Плотности распределения Вебулла с параметрами

ParmR=[0.3 0.5   1 3 5]

Синий – красный - желтый – фиолетовый - зеленый

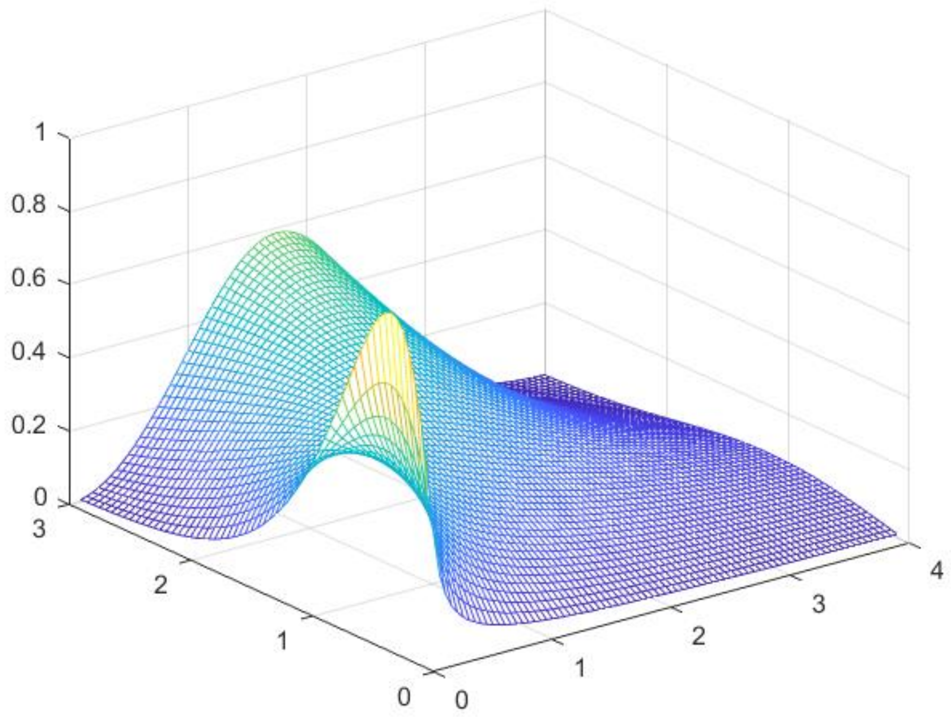


Рисунок Л.5. - Плотность распределения Вебулла с параметрами 1 и 0.3

Основная гипотеза и простейшие критерии её проверки

Первым шагом выполняется проверка выборки на нормальность с помощью двух простейших критериев: по доверительным интервалам для ax* и ex* и Жарка-Бера. Для проверки возьмём один, самый жёсткий уровень значимости.

Программа 1

```
clear all % очистить память
sf='D:\Iglin\Matlab\ContData\xrayl.txt'; % имя файла данных
x=load(sf); % вводим ИД
x=sort(x(:)); % переформатировали столбец и рассортировали
n=length(x); % количество данных
xmin=x(1); % минимальное значение
xmax=x(n); % максимальное значение
Mx=mean(x); % математическое ожидание
f=n-1; % число степеней свободы
Dx=var(x); % дисперсия
Sx=std(x); % среднеквадратичное отклонение
Ax=skewness(x,0); % несмещенная асимметрия
Ex=kurtosis(x,0)-3; % несмещенный эксцесс
```

```
Medx=median(x); % медиана
Rx=range(x); % размах выборки
p=[0.9;0.95;0.99;0.999]; % задаём доверительные вероятности
q=1-p; % уровни значимости
Da=6*(n-1)/(n+1)/(n+3); % дисперсия Ax
De=24*n*(n-2)*(n-3)/(n+1)^2/(n+3)/(n+5); % дисперсия Ex
Axd=[p,Ax-(Da./q).^0.5,Ax+(Da./q).^0.5]';  % формула (10.19)
Exd=[p,Ex-(De./q).^0.5,Ex+(De./q).^0.5]';  % формула (10.20)
fprintf('Простейший тест: q=%2.0f%% - ',q(1)*100);
if prod(Axd(2:3,1))|prod(Exd(2:3,1)), % простейший критерий
  fprintf('отвергаем основную гипотезу.\n')
else
  fprintf('принимаем основную гипотезу.\n')
end
fprintf('Тест Жарка-Бера: q=%2.0f%% - ',q(1)*100);
if jbtest(x,q(1)), % критерий Жарка-Бера
  fprintf('отвергаем основную гипотезу.\n')
else
  fprintf('принимаем основную гипотезу.\n')
end
```

```
Простейший тест: q=10% - отвергаем основную гипотезу.
Тест Жарка-Бера: q=10% - отвергаем основную гипотезу.
```

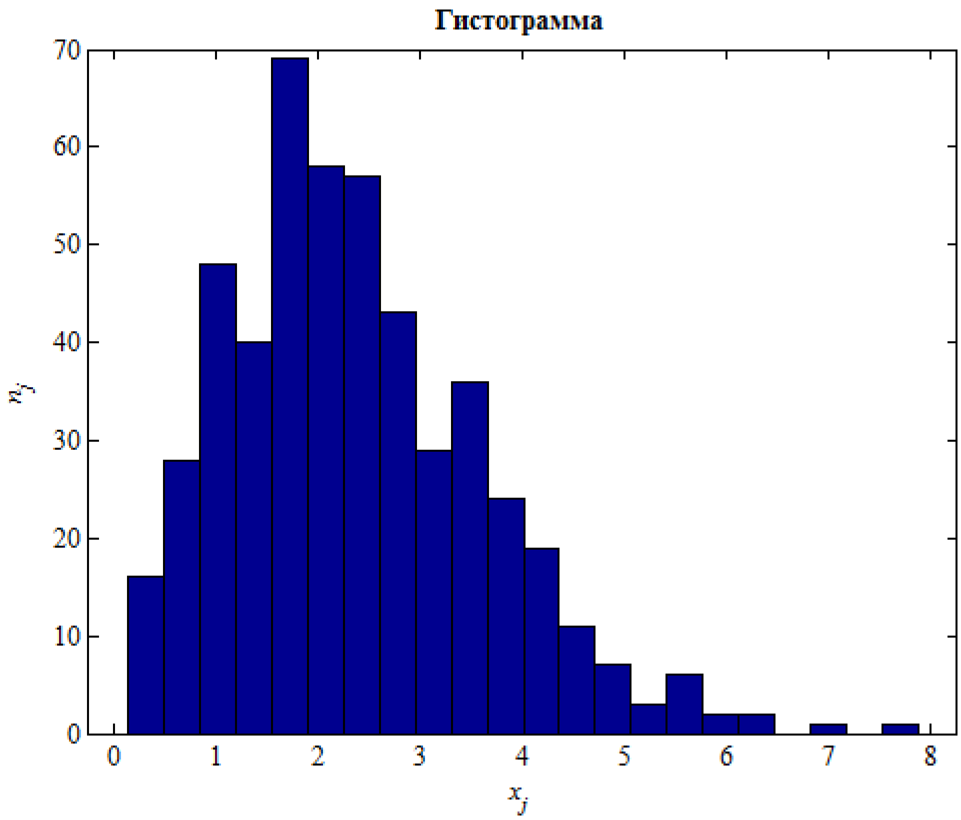
Подбор вида распределения

Выполним очередной фрагмент ИДЗ по обработке массива данных. Зададим количество интервалов. Найдём ширину каждого интервала (она обозначена в программе идентификатором d). Будем предполагать распределение непрерывным, поэтому построим гистограмму.

Программа 2

```
clear all % очистить память
sf='D:\Iglin\Matlab\ContData\xrayl.txt'; % имя файла данных
x=load(sf); % вводим ИД
x=sort(x(:)); % переформатировали столбец и рассортировали
n=length(x); % количество данных
xmin=x(1); % минимальное значение
xmax=x(n); % максимальное значение
Mx=mean(x); % математическое ожидание
f=n-1; % число степеней свободы
Dx=var(x); % дисперсия
Sx=std(x); % среднеквадратичное отклонение
Ax=skewness(x,0); % несмещенная асимметрия
Ex=kurtosis(x,0)-3; % несмещенный эксцесс
Medx=median(x); % медиана
Rx=range(x); % размах выборки
p=[0.9;0.95;0.99;0.999]; % задаём доверительные вероятности
q=1-p; % уровни значимости
k=round(n^0.5); % число интервалов для построения гистограммы
d=(xmax-xmin)/k; % ширина каждого интервала
del=(xmax-xmin)/20; % добавки влево и вправо
xl=xmin-del;
```

```
xr=xmax+del; % границы интервала для построения графиков
fprintf('Число интервалов k=%d\n',k)
fprintf('Ширина интервала h=%14.7f\n',d)
figure % создаем новую фигуру
hist(x,k) % построили гистограмму
set(get(gcf,'CurrentAxes'),...
  'FontName','Times New Roman Cyr','FontSize',10)
title('\bfГистограмма') % заголовок
xlim([xl xr]) % границы по оси OX
xlabel('\itx_{j}') % метка оси x
ylabel('\itn_{j}') % метка оси y
```




```
Число интервалов k=22
Ширина интервала h=     0.3514961
```

Подбор параметров распределения

Продолжим выполнение ИДЗ "Обработка массива данных". Найдём параметры теоретического распределения по ПМП для всех подходящих непрерывных распределений, реализованных в MATLAB. При выполнении своего варианта ИДЗ обратите внимание: для некоторых наборов данных отдельные виды распределений нужно исключить. Так, β-распределение не подходит, если $\exists\ x_i \notin\ [0;\ 1]$; экспоненциальное и рэлеевское распределения не подходят, если есть отрицательные x_i, и т.д.

Программа 3

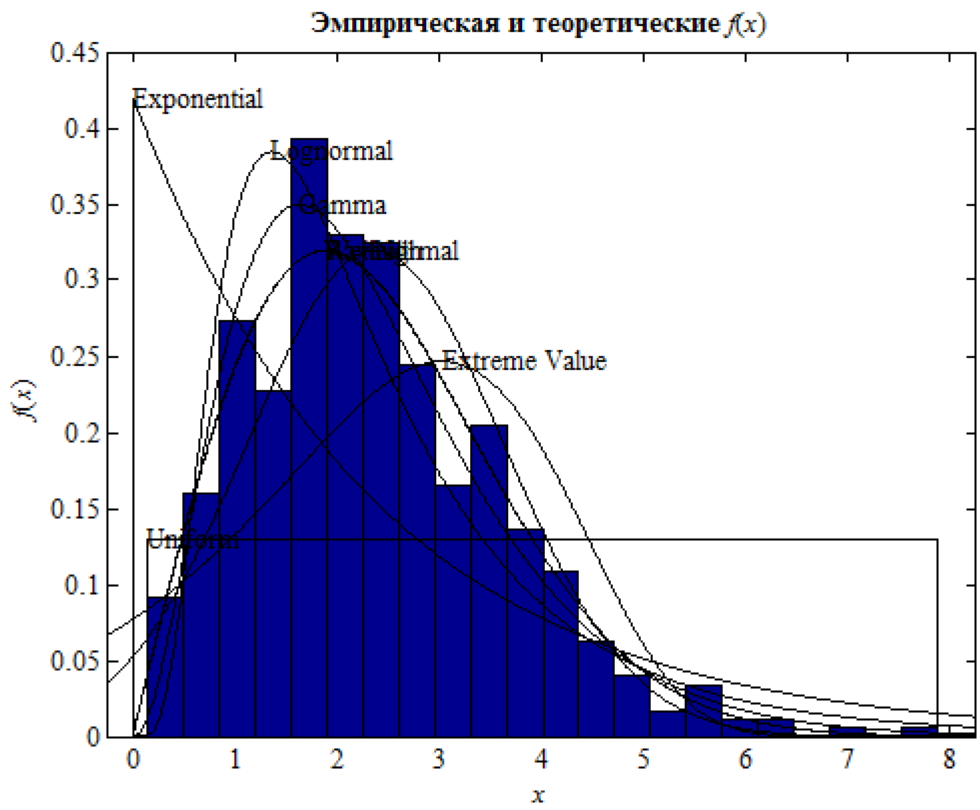
```
dlist={'Extreme Value'; 'Exponential'; 'Gamma'; 'Lognormal'; ...
  'Normal'; 'Rayleigh'; 'Uniform'; 'Weibull'};
dlistr={'гамбеловское'; 'экспоненциальное'; ...
  'гамма -'; 'логнормальное'; 'нормальное'; 'рэлеевское'; ...
  'равномерное'; 'вейбулловское'};
dparname={{'mu' 'sigma'};{'mu'};{'a' 'b'};{'mu' 'sigma'};...
  {'mu' 'sigma'};{'b'};{'a' 'b'};{'a' 'b'}};
ndist=length(dlist); % количество распределений
for idist=1:ndist, % подбираем параметры для всех распределений
  phatone=mle(x,'distribution',dlist{idist});
  phat{idist}=phatone; % запомнили
end
disp('Параметры различных распределений по ПМП:')
for idist=1:ndist, % печатаем параметры для всех распределений
  fprintf('%s распределение:',dlistr{idist});
  parname=dparname{idist}; % список параметров
  phatone=phat{idist}; % значения параметров
  for ipar=1:length(parname), % печатаем параметры
    fprintf('  %s=%14.10f;',parname{ipar},phatone(ipar));
  end
  fprintf('\n');
end
```

```
Параметры различных распределений по ПМП:
гамбеловское распределение:  mu=  3.0396037771;  sigma=  1.4934684657;
экспоненциальное распределение:  mu=  2.3764047033;
гамма - распределение:  a=  3.2163611628;  b=  0.7388488366;
логнормальное распределение:  mu=  0.7021526965;  sigma=  0.6256024600;
нормальное распределение:  mu=  2.3764047033;  sigma=  1.2496641734;
рэлеевское распределение:  b=  1.8985467943;
равномерное распределение:  a=  0.1477582345;  b=  7.8806725958;
вейбулловское распределение:  a=  2.6828440974;  b=  1.9927897171;
```

Построим на одном графике эмпирическую и все подбираемые теоретические плотности распределения. График эмпирической плотности распределения (11.1) отличается от гистограммы множителем nh по оси ординат, где n – число экспериментальных данных, а h – ширина интервала при построении гистограммы. Если все ординаты на гистограмме разделить на это число, то площадь под гистограммой станет равной 1. В MATLAB Statistics Toolbox это делает функция ecdfhist Она строит нормированную нужным образом гистограмму. Для неё нужно предварительно построить выборочную функцию распределения с помощью функции ecdf. Теоретические плотности распределения строим с помощью функции pdf, задавая нужный вид распределения и его параметры. Над каждой теоретической кривой надпишем вид распределения, чтобы его легче было сравнивать с эмпирическим.

Программа 4

```
[Fi,xi]=ecdf(x); % эмпирическая функция распределения
figure % создаем новую фигуру
ecdfhist(Fi,xi,k) % построили нормированную гистограмму
xpl=linspace(xl,xr,1000); % абсциссы для графиков
hold on % задержка для рисования на одном графике
for idist=1:ndist, % рисуем теоретические распределения
  phatone=phat{idist}; % значения параметров
  com=['pdf(''' dlist{idist} ''',xpl']; % команда
  for ipar=1:length(phatone), % добавляем параметры
    com=[com ',' sprintf('%d',phatone(ipar))];
  end
  com=[com ')']; % сформировали команду
  ypl=eval(com); % выполнили команду - вычислили f(x)
  plot(xpl,ypl,'k-') % добавили на график
  [ym,iym]=max(ypl); % максимум на графике
  h=text(xpl(iym),ym,dlist{idist}); % название распределения
  set(h,'FontName','Times New Roman Cyr','FontSize',10)
end
hold off % выключили задержку
set(get(gcf,'CurrentAxes'),...
  'FontName','Times New Roman Cyr','FontSize',10)
title('\bfЭмпирическая и теоретические\rm \itf\rm(\itx\rm)') % заголовок
xlim([xl xr]) % границы по оси OX
xlabel('\itx') % метка оси x
ylabel('\itf\rm(\itx\rm)') % метка оси y
```



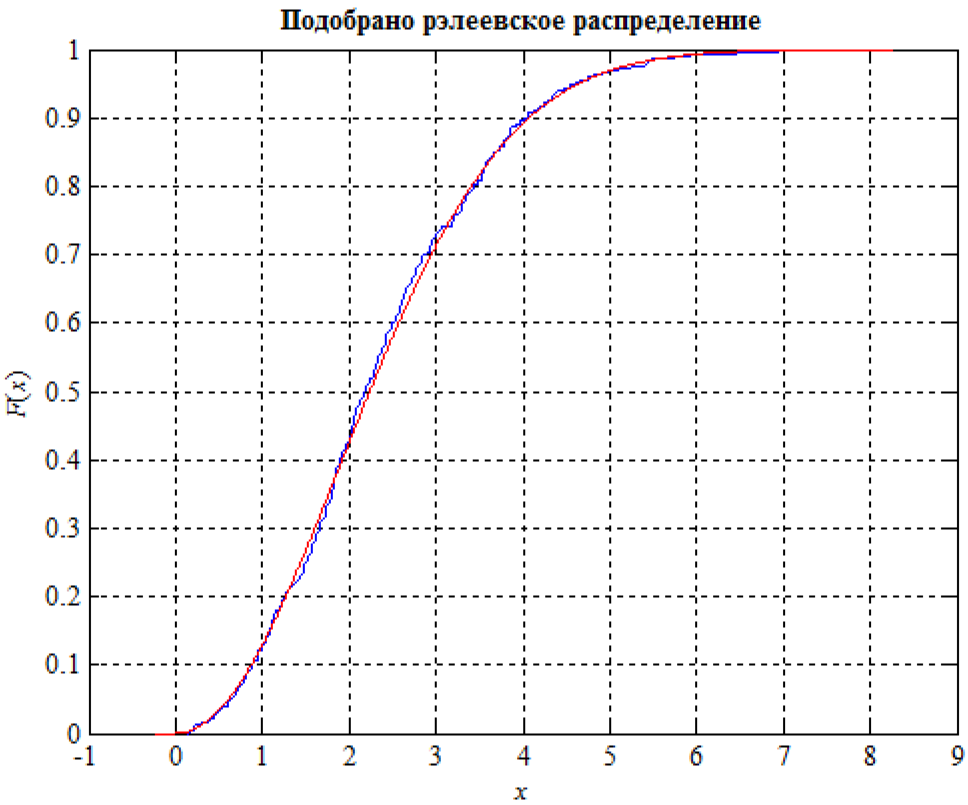
Проверка правильности подбора - Критерий согласия Колмогорова

В MATLAB проверка по критерию согласия Колмогорова осуществляется с помощью функции kstest. Продолжим выполнение ИДЗ по обработке массива данных. В предыдущем фрагменте мы взяли все возможные непрерывные распределения, реализованные в MATLAB, в качестве предполагаемых теоретических. Проверим, какое них лучше всего согласуется с нашей выборкой по критерию согласия Колмогорова. Для него критическое значение (уровень значимости) q будет максимальным. Поэтому проведём проверку по критерию Колмогорова со всеми распределениями, и выберем то, для которого q максимальное. Покажем на одном рисунке выборочную функцию распределения F*(x) и наиболее подходящую теоретическую. График выборочной функции распределения рисует функция cdfplot, а теоретическую строим по точкам xi.

Программа 5

```
clear all % очистить память
sf='D:\Iglin\Matlab\ContData\xrayl.txt'; % имя файла данных
x=load(sf); % вводим ИД
x=sort(x(:)); % переформатировали столбец и рассортировали
n=length(x); % количество данных
xmin=x(1); % минимальное значение
xmax=x(n); % максимальное значение
Mx=mean(x); % математическое ожидание
f=n-1; % число степеней свободы
Dx=var(x); % дисперсия
Sx=std(x); % среднеквадратичное отклонение
Ax=skewness(x,0); % несмещенная асимметрия
Ex=kurtosis(x,0)-3; % несмещенный эксцесс
Medx=median(x); % медиана
Rx=range(x); % размах выборки
p=[0.9;0.95;0.99;0.999]; % задаём доверительные вероятности
q=1-p; % уровни значимости
dlist={'Extreme Value'; 'Exponential'; 'Gamma'; 'Lognormal'; ...
  'Normal'; 'Rayleigh'; 'Uniform'; 'Weibull'};
dlistr={'гамбеловское'; 'экспоненциальное'; ...
  'гамма -'; 'логнормальное'; 'нормальное'; 'рэлеевское'; ...
  'равномерное'; 'вейбулловское'};
dparname={{'mu' 'sigma'};{'mu'};{'a' 'b'};{'mu' 'sigma'};...
  {'mu' 'sigma'};{'b'};{'a' 'b'};{'a' 'b'}};
ndist=length(dlist); % количество распределений
for idist=1:ndist, % подбираем параметры для всех распределений
  phatone=mle(x,'distribution',dlist{idist});
  phat{idist}=phatone; % запомнили
```

```
end
qq=[]; % критические уровни значимости
for idist=1:ndist, % критерий Колмогорова
  phatone=phat{idist}; % значения параметров
  com=['cdf(''' dlist{idist} ''',x']; % команда
  for ipar=1:length(phatone), % добавляем параметры
    com=[com ',' sprintf('%d',phatone(ipar))];
  end
  com=[com ')']; % сформировали команду
  Fx=eval(com); % выполнили команду - вычислили F(x)
  [hkolm,pkolm,kskolm,cvkolm]=kstest(x,[x Fx],0.1,0);
  qq=[qq pkolm]; % критические уровни значимости
end
[maxqq,bdist]=max(qq); % выбрали лучшее распределение
fprintf(['Критерий согласия Колмогорова:\n',...
  'Лучше всего подходит %s распределение;\n'...
  'критический уровень значимости для него = %8.5f\n'], ...
  dlistr{bdist},maxqq);
figure % создаем новую фигуру
cdfplot(x); % эмпирическая функция распределения
phatone=phat{bdist}; % параметры наилучшего распределения
com=['cdf(''' dlist{bdist} ''',xpl']; % команда
for ipar=1:length(phatone), % добавляем параметры
  com=[com ',' sprintf('%d',phatone(ipar))];
end
com=[com ')']; % сформировали команду
del=(xmax-xmin)/20; % добавки влево и вправо
xl=xmin-del;
xr=xmax+del; % границы интервала для построения графиков
xpl=linspace(xl,xr,1000); % абсциссы для графиков
Fxpl=eval(com); % вычислили F(x) для наилучшего распределения
hold on % для рисования на этом же графике
plot(xpl,Fxpl,'r'); % дорисовали F(x)
hold off
set(get(gcf,'CurrentAxes'),...
  'FontName','Times New Roman Cyr','FontSize',10)
title(['\bfПодобрано ' dlistr{bdist} ' распределение'])
xlabel('\itx') % метка оси x
ylabel('\itF\rm(\itx\rm)') % метка оси y
```




```
Критерий согласия Колмогорова:
Лучше всего подходит рэлеевское распределение;
критический уровень значимости для него =  0.91482
```

Контрольные вопросы к лабораторной работе 3

1. В чём состоит простейший критерий проверки 0-гипотезы?
2. Как выглядит критерий Жарка-Бера?
3. Проверка по простейшему критерию показала, что 0-гипотезу нельзя отвергнуть. Значит ли, что распределение X - нормальное? Почему?
4. Как строится гистограмма?
5. Найдите в литературе или Интернете доказательство формулы Стэрджесса.
6. Сформулируйте условия теоремы Колмогорова.
7. Приведите доказательство критерия согласия Пирсона.

Л.4. Модели генерации отказов невосстанавливаемых и восстанавливаемых систем

Задание 2. Найти параметры распределения для заданной выборки.

1. Сгенерировать параметры распределения Вейбулла для своего варианта

```
clear all;
N=10;                        % Номер варианта
rand('state',10);            % Настройка датчика на номер варианта
p1=0.5+2*rand;               % Первый параметр Распределения Вебулла
p2=1+3*rand;                 % Второй параметр
```

(p1,p2) – параметры распределения Вейбулла

2. Исследовать форму функции и плотности распределения вероятности.

3. Сгенерировать случайную выборку из 300 элементов с распределением Вебулла с параметрами (p1,p2).

4. Подобрать параметры распределения для сгенерированной выборки. Задача заключается в поиске подходящей модели надежности по статистическим данным отказов, а также оценки параметров этого распределения.

Сгенерировать параметры распределения Вейбулла для своего варианта


```
clear all;
N=10;                        % Номер варианта
rand('state',10);            % Настройка датчика на номер варианта
p1=0.5+2*rand;               % Первый параметр Распределения Вебулла
p2=1+3*rand;                 % Второй параметр
```

(p1,p2) – параметры распределения Вейбулла

Задание 1. Генерация выборочных значений наработок на отказ и расчет параметров надежности невосстанавливаемых систем

Параметры – файл Vari2.xlsx (строка = номер в группе + смещение по группам)

Группа 1 – с 1-го

Группа 2 – с 21-го

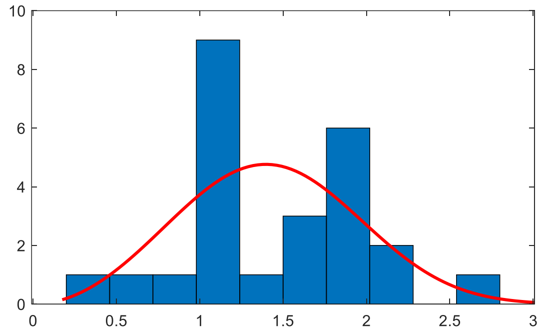
Группа 3 – с 41-го

Группа 4 – с 61-го

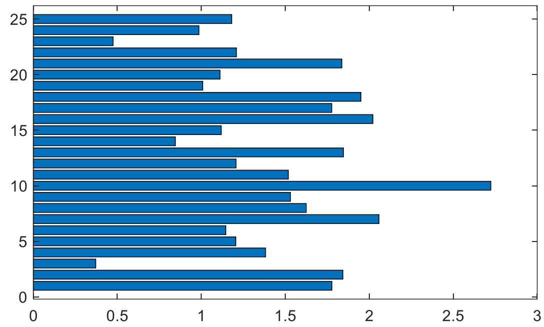
Группа 5 – с 81-го

Иностранцы – с 101

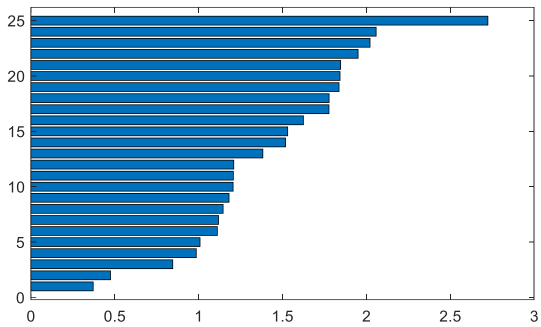
1. Сгенерировать выборку объемом 50 значений времен наработки на отказ по распределению Вейбулла с параметрами (p1,p2). Построить набор диаграмм для интерпретации вероятности отказа и вероятности безотказной работы.



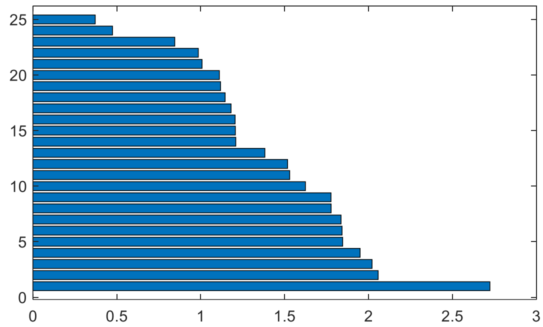
а) гистограмма наработки на отказ



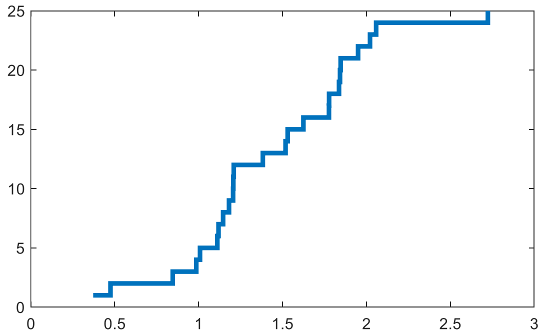
б) неупорядоченные времена наработки



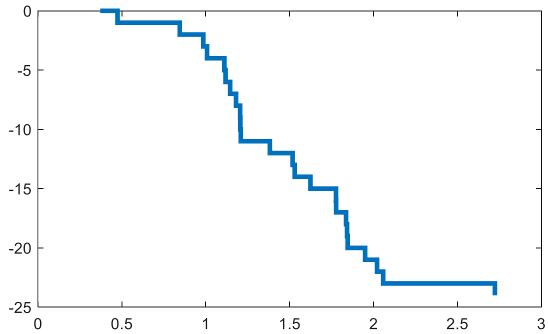
в) упорядоченные по убыванию



г) упорядоченные по возрастанию



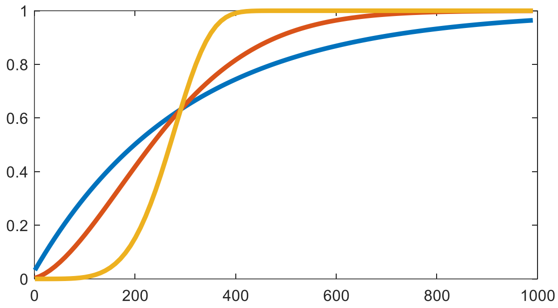
д) вероятность отказа $Q(t)$

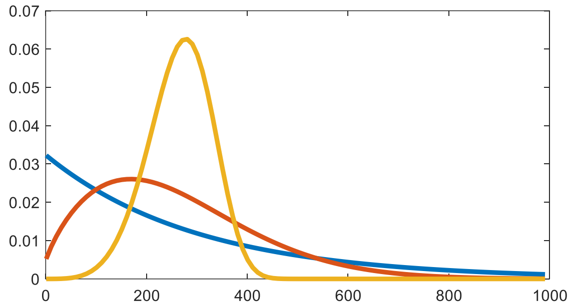


е) ВБР – P(t)

Рисунок Л.6. - Статистическая интерпретация ВБР и вероятности отказа

2. Сгенерировать и построить графики функции и плотности распределения Вейбулла для своего варианта (p1,p2)±50%.





а) $Q(t)$ б) $f(t)$

Рисунок Л.7. - Функция и плотность распределения с параметрами (p1,p2)±50%

3. Выполнить расчет показателей надежности для систем без восстановления с отказами по распределению Вебулла с параметрами своего варианта, а именно:

- вероятности безотказной работы;
- вероятности отказа;
- плотности вероятности отказа;
- интенсивности отказов;
- средней наработки на отказ.

Построить графики указанных показателей как функций времени.

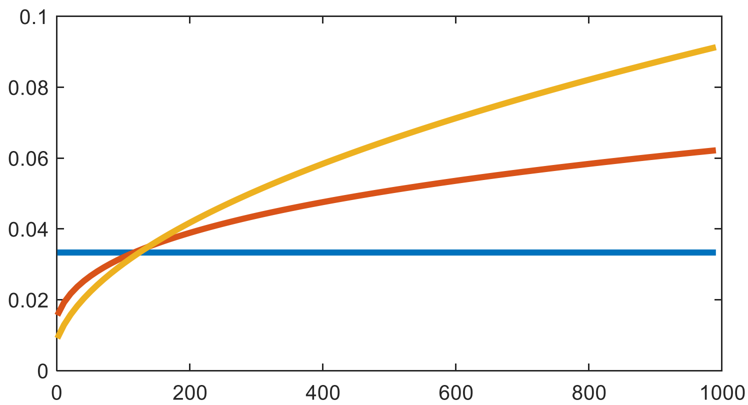
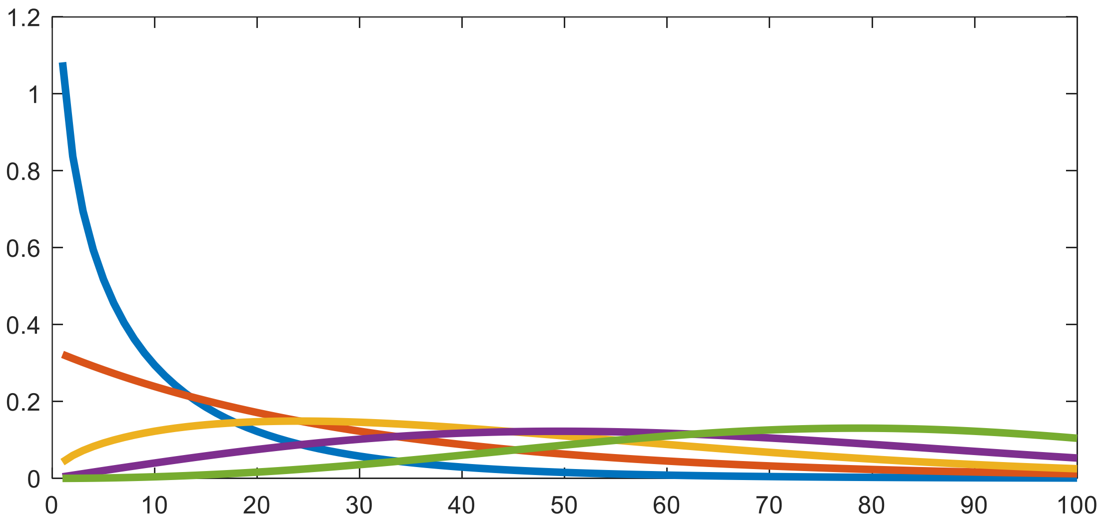
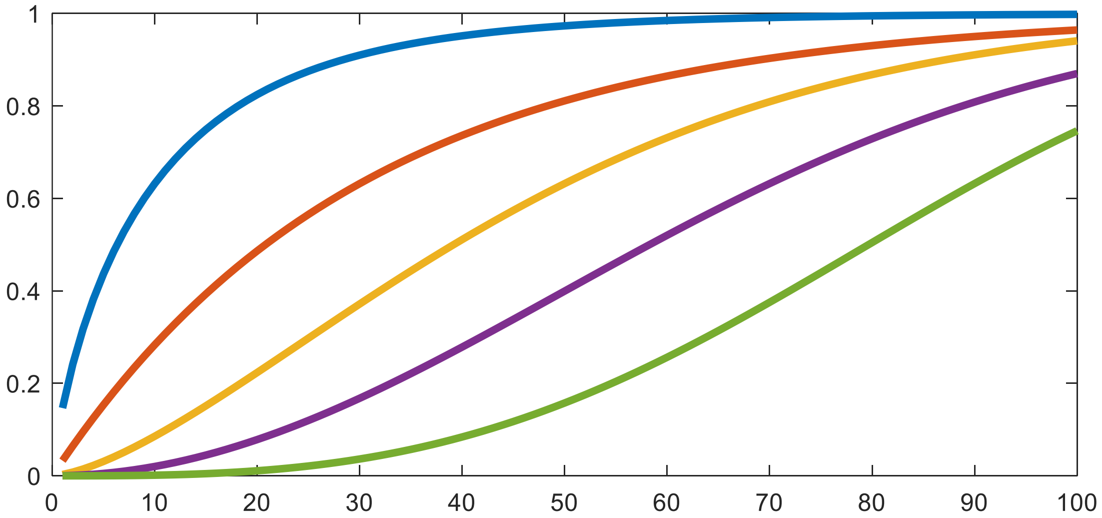


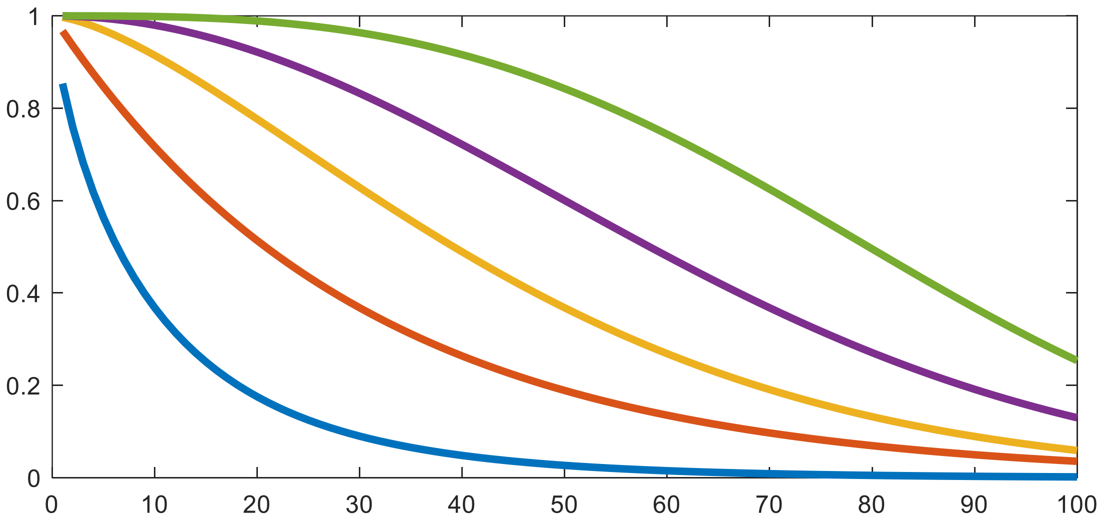
Рисунок Л.8. - Интенсивность отказов для модели Вебулла с параметрами (p1,p2)±50%

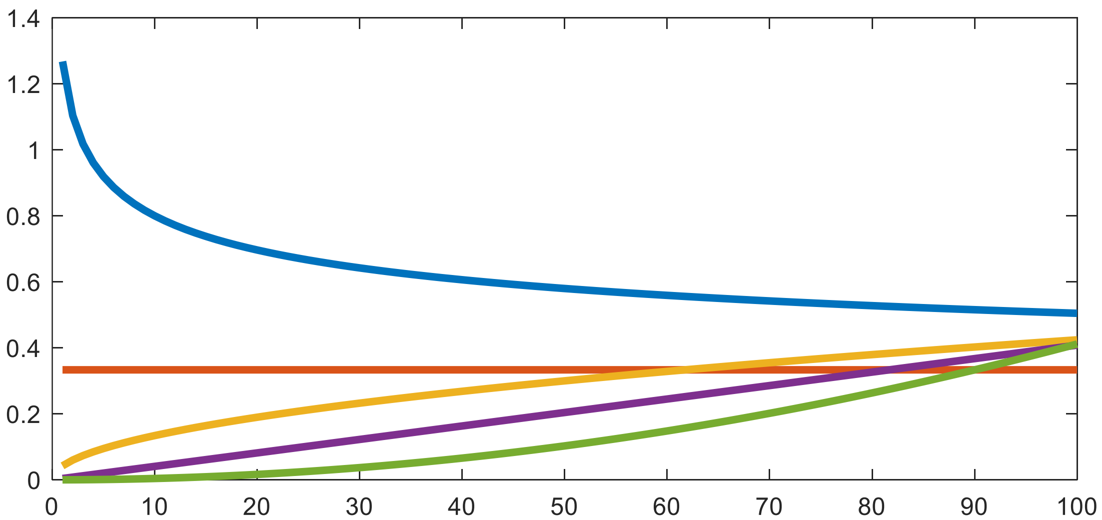
Построение показателей надежности (Pok_fQPL.m)

```
% Показатели надежности - pdf,  ФР - cdf
clear all;   N=5;  Raspr='weibull';
ParmR=[1      3      5    7    9;
              0.8   1   1.5   2    3];
for n=1:1:N
  for t=1:1:100      % восстановление - эквивалент отказа
  tr=t/10;
  f(n,t)=pdf(Raspr,tr,ParmR(1,n), ParmR(2,n));      % Плотность вероятности отказа -
  Q(n,t)=cdf(Raspr,tr,ParmR(1,n), ParmR(2,n));     % Вероятность отказа
  P(n,t)=1-Q(n,t);                                  % Вероятность безотказной работы
  L(n,t)=f(n,t)/P(n,t);                             % Интенсивность отказов
  end
end
figure('Position',[100 200 700 300]);   t=1:1:100;  plot(t, L, 'LineWidth',3);
```



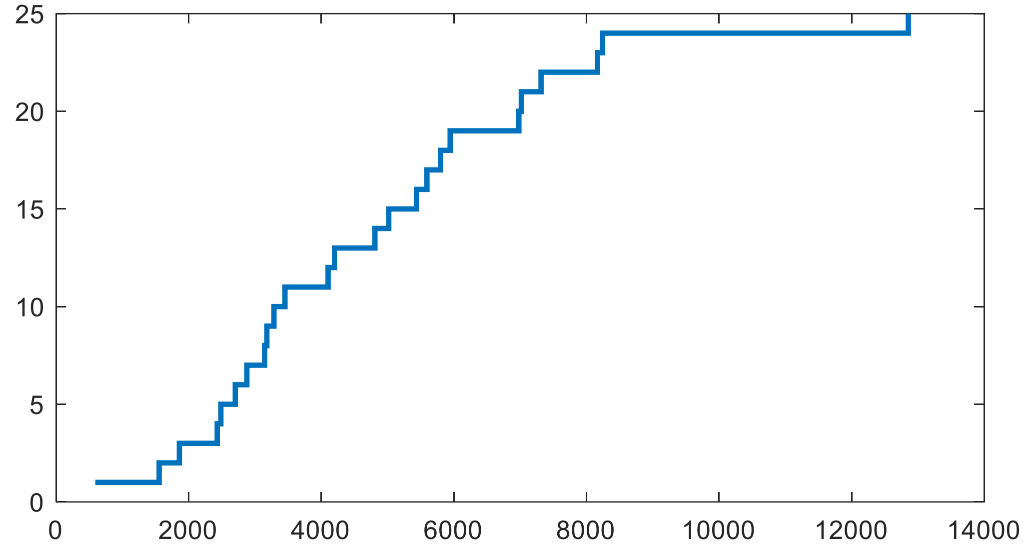

1
0.8
0.6
0.4
0.2
0
0
10
20
30
40
50
60
70
80
90
100

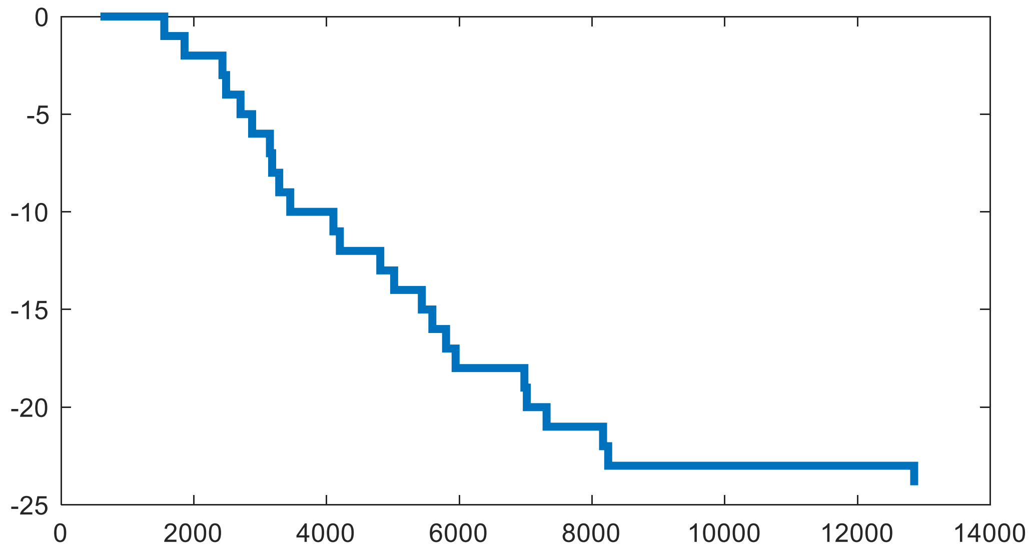

1
0.8
0.6
0.4
0.2
0
0
10
20
30
40
50
60
70
80
90
100

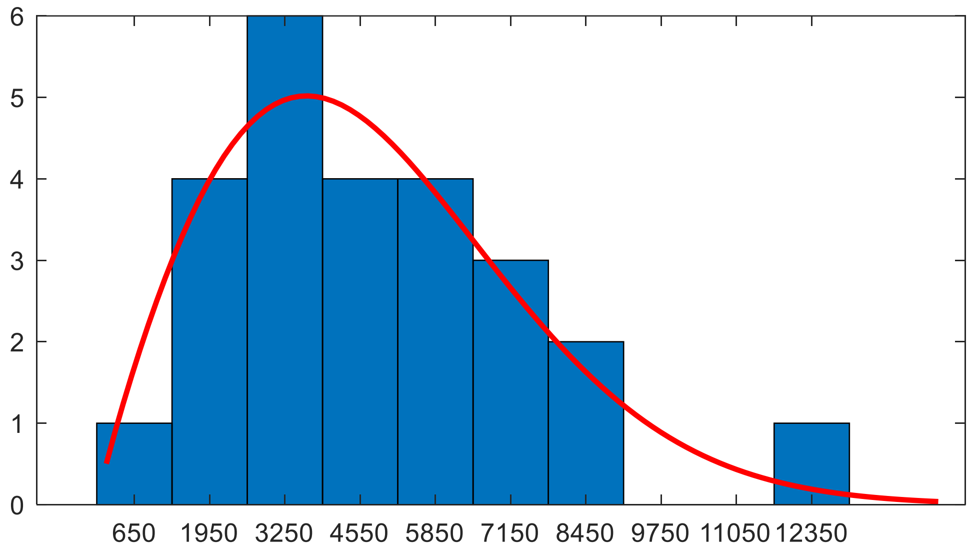

1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
10
20
30
40
50
60
70
80
90
100

Статистические показатели (Pok_PQT.m)

```
% Статистика наработки на отказ
clear all;    n=25;    mm='weibull';
Th=random('weibull',5000, 2,1,n);                  % Генерация времен наработки на отказ
Th_sortP=sort(Th,'descend');                       % Сортировка времен
Th_sortQ=sort(Th);
P=1:1:n;                                            % Доля каждого элемента
figure('Position',[100 200 600 300]);              % figure(1);
stairs(Th_sortQ, P,'LineWidth', 2 );                % График функции безотказной работы
figure(2); stairs(Th_sortQ, 1-P,'LineWidth',3 );  %  График функции вероятности отказа
figure(3); histfit(Th,10,'weibull');                 %  Наработка на отказ
```







Задание 2. Расчет параметров надежности восстанавливаемых систем

Параметры – файл Vari2.xlsx (строка = номер в группе + смещение по группам)

Группа 1 – с 1-го

Группа 2 – с 21-го

Группа 3 – с 41-го

Группа 4 – с 61-го

Группа 5 – с 81-го

Иностранцы – с 101

Столбец2 – интенсивность восстановления - *mu*

Столбец3 – интенсивность отказов- *lam*

1. Написать программу генерации выборочных траекторий с параметрами отказов и восстановления *lam*, *mu*. Построить графики работоспособности и простоя системы.

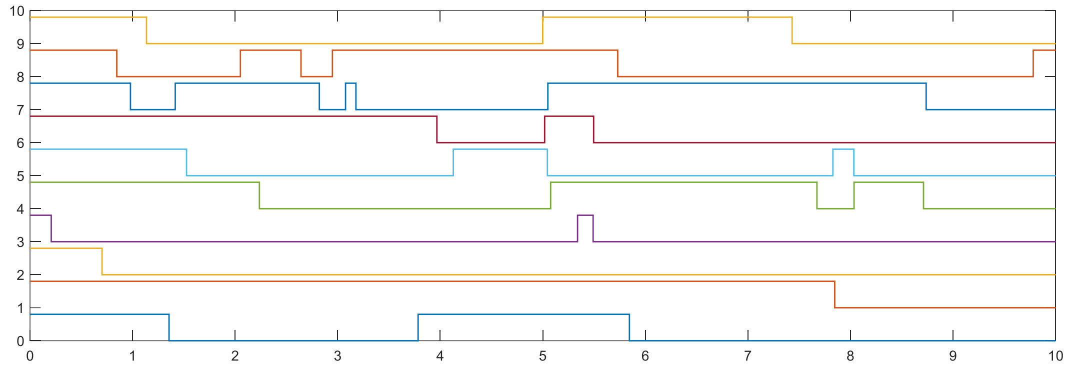


Рисунок Л.9. - Выборочные траектории восстанавливаемой системы

```
Наброски программы генерации
% Статистика наработки на отказ восстанавливаемых систем
clear all;    Nobj=10;     mm='exp';  Tmod=1000;
figure('position', [100 100 900 300]);
for N=1:1:Nobj
t=0;  n=1; Mstat(n,1)=t; Mstat(n,2)=1;
while t<Tmod
    Twk=random(mm, 2);    n=n+1;   t=t+Twk;  Mstat(n,1)=t; Mstat(n,2)=0;
    Tvos=random(mm, 5);   n=n+1;  t=t+Tvos; Mstat(n,1)=t; Mstat(n,2)=1;
end
%figure(1);
stairs((Mstat(:,1))', (Mstat(:,2)*0.8)'+N-1,'LineWidth',1 );
if N==1 hold('on'); end;
end
xlim([0 10]);  %ylim([0 5]);
hold('off');
```

2. Выполнить расчет параметров надежности для систем с восстановлением

- коэффициент готовности;
- коэффициент оперативной готовности;
- коэффициент технического использования.

3. По методы Монте-Карло показать, что среднее по выборок равно среднему по реализациям.

Контрольные вопросы к лабораторной работе 4

1. Как определяется среднее время безотказной работы?
2. Какова связь между вероятностью безотказной работы и интенсивностью отказов?
3. Какова связь между плотностью вероятности и интенсивностью отказов?

4. Какова связь между вероятностью безотказной работы и средним временем безотказной работы?
5. В чем состоит различие между восстанавливаемыми и невосстанавливаемыми системами?
6. Что представляет собой кривая изменения интенсивности отказов во времени и кривая изменения эксплуатационных затрат от наработки изделия во времени?
7. Дайте определения показателей для оценки безотказности – вероятности безотказной работы и вероятности отказа, параметра потока отказов, средней наработки на отказ, средней наработки до отказа, гамма-процентной наработки до отказа, интенсивности отказов. Каковы единицы их измерения?
8. В чем заключается метод статистических испытаний? Расскажите его сущность.

Л.5. Расчет параллельно-последовательной схемы надежности для различных моделей

Задание

Расчет схемы надежности для моделей надежности:

Экспоненциальной, Релея, Вейбулла

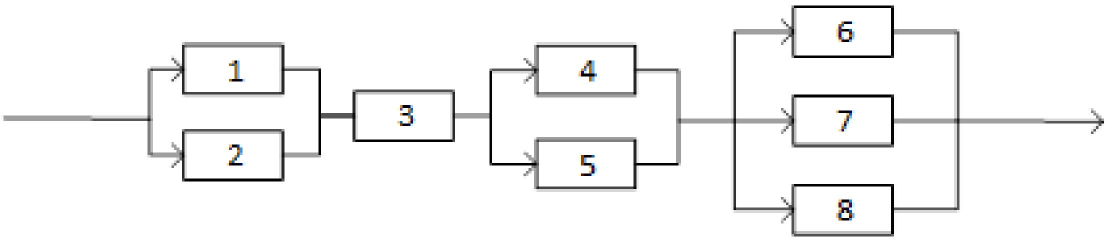


Таблица вариантов – **DLPP.xlsx**

Вариант	Сх		λ - Эксп.	σ - Релея	a, b - Вейбулла	
Nвар	№ сх	1	λ_1	σ_1	a_1	b_1
		2	λ_2	σ_2	a_2	b_2
		9	λ_9	σ_9	a_9	b_9

Параметры с номерами, которые превышают размерность схемы – не учитывать.

Рассчитать вероятность безотказной работы и другие параметры надежности для моментов времени t={100, 300, 500, 1000, 50000}

Теоретические положения

Для расчета надежности объекта с **последовательно-параллельной схемой надежности** применяют метод «свертки», который состоит из этапов:

- на первом этапе рассматриваются все параллельные соединения, которые заменяются эквивалентными элементами с соответствующими показателями надежности;
- -на втором этапе рассматриваются все последовательные соединения, которые заменяются эквивалентными элементами;
- на третьем этапе вновь рассматриваются все параллельные соединения, которые заменяются эквивалентными элементами;
- преобразования продолжаются до тех пор, пока исходная структурная схема надежности не будет преобразована к схеме последовательного соединения элементов.

Пример расчета

1. Придумать любую последовательно-параллельную схему надежности с количеством элементов не менее 9, например:

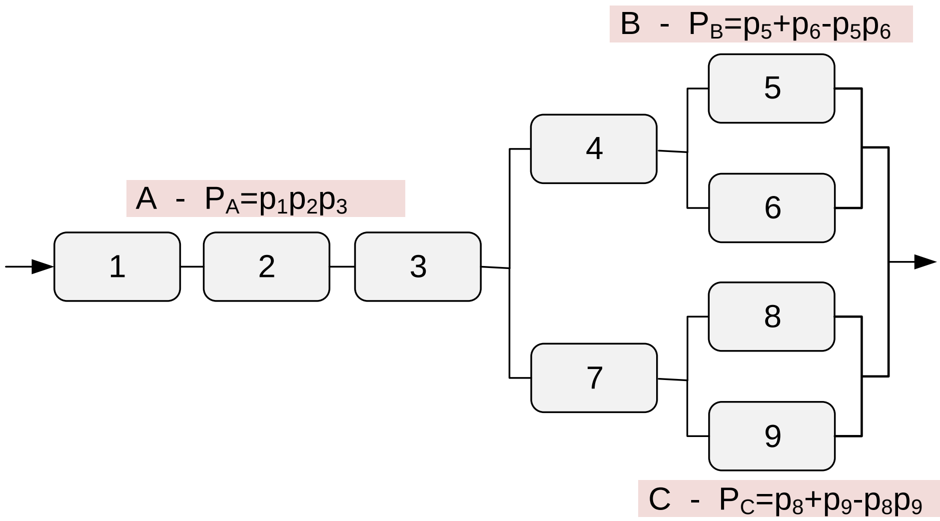


Рисунок Л.10. - Последовательно-параллельная схема надежности

2. Для сформированной схемы сгенерировать параметры экспоненциальных моделей для своего варианта

```
clear all;              % Генерация данных  для параллельно-последовательной
                        %  схемы надежности
N=5;                    % Номер варианта
M=12;                   % Количество элементов
rand('state',N);        % Настройка датчика на номер варианта
P=5+2*rand(M,1)         % Генерация вектора параметров ЭМН
```

P – вектор параметров моделей надежности, например:

```
6.7277
6.8124
5.4003
6.7188
6.5919
5.6764
6.4586
6.0855
5.8604
5.7927
5.4275
5.2675
```

3. Написать программу расчета надежности своей схемы

4. Написать программу расчета надежности для произвольной последовательно-параллельной схемы.

5. Написать программу генерации случайной структуры графа параллельно-последовательной схемы надежности – повышенная сложность

На первом этапе расчета (Рисунок Л.10.) выделим звено последовательного соединения (А) и два дублированных звена (В и С). Для них по известным формулам рассчитаем вероятности безотказной работы (P_A, P_B, P_C) и заменим эти звенья на эквивалентные элементы А, В, С.

На втором этапе (Рисунок Л.11.) последовательное соединение элементов 4 и В, 7 и С заменим на эквивалентные элементы D и E с соответствующими вероятностями безотказной работы P_D и P_E.

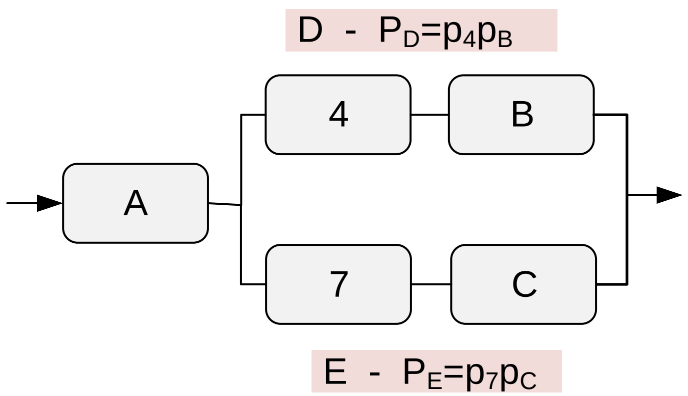
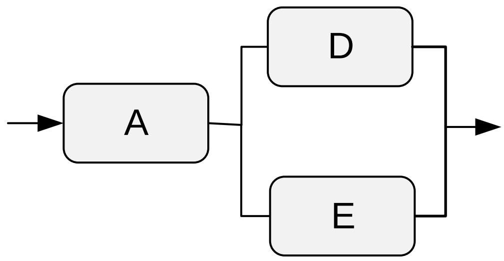


Рисунок Л.11. - Этап 2 расчета последовательно-параллельной схемы надежности

Для полученной простой последовательно-параллельной схемы на третьем этапе (Рисунок Л.12.) записываем выражение для системной вероятности безотказной работы.



$$P_{системы}=p_A(p_D+p_E-p_Dp_E)$$

Рисунок Л.12. - Этап 3 расчета последовательно-параллельной схемы надежности

Типовые вопросы:

1. Дайте определение последовательного и параллельного соединения элементов в смысле надежности.
2. Приведите формулы вычисления надежности системы при последовательном и параллельном соединении элементов.
3. Чему равно среднее время наработки и интенсивность отказов системы при последовательном соединении элементов ?
4. Чему разно среднее время наработки системы при параллельном соединении одинаковых элементов с экспоненциальным законом надежности?
5. Приведите формулы функции надежности системы при последовательном и параллельном соединении элементов с экспоненциальным законом надежности.
6. Приведите приближенные формулы для расчета высоконадежных систем.
7. Как определить функцию надежности элемента, если он может отказать как во время работы, так и во время хранения, если принять закон изменения надежности экспоненциальным
8. Запишите формулы эквивалентных преобразований структурной схемы надежности при последовательном, параллельном и смешанном соединении элементов.

9. Каким образом преобразуются структурные схемы, содержащие поперечные связи?
10. Запишите формулы эквивалентных преобразований структурной схемы из «треугольника» в «звезду» и обратно.
11. Опишите алгоритм логико-вероятностного расчета надежности. Поясните порядок составления дерева отказов.
12. Сформулируйте основные законы алгебры логики, используемые при анализе надежности технических систем.
13. В чем особенность таблично-логического метода расчета надежности? Поясните порядок составления таблицы состояний и переходов.

1. Какое соединение элементов называется основным?
2. Как определяется вероятность безотказной работы при основном соединении элементов в период нормальной эксплуатации и период износов.
3. Какое соединение элементов называется резервным?
4. Как определяется вероятность безотказной работы при нагруженном соединении элементов?
5. Как определяется вероятность безотказной работы при частном резервировании элементов?
6. Как определяется вероятность безотказной работы при общем резервировании элементов?
7. Как определяется вероятность безотказной работы при смешанном соединении элементов?
8. Поясните понятие сложной системы и ее особенности с позиций надежности.
9. Перечислите четыре группы элементов сложных систем.
10. Поясните отличия основных типов структур сложных систем – расчлененных, связанных и комбинированных.
11. Поясните расчет схемной надежности сложных систем при последовательном соединении элементов.
12. Поясните расчет схемной надежности сложных систем при параллельном соединении элементов.
13. Поясните термин структурного резервирования.
14. Перечислите виды резервирования в зависимости от схемы включения резерва.
15. Перечислите виды резервирования в зависимости от способа включения резерва.
16. Перечислите виды резервирования в зависимости от состояния резерва.

Л.6. Аналитические модели Марковских процессов

Задание 1

Рассчитать стационарные вероятности для Марковской цепи, структуру которой определили в лаб.1.

Теоретические положения 1

Пример расчета 1

Типовые вопросы 1

1. Какой процесс называют случайным? Приведите примеры.
2. В чем состоит Марковское свойство случайного процесса?
3. Какие матрицы называются стохастическими?
4. Сформулируйте определение собственных чисел и собственных векторов матрицы.
5. Какие разновидности Марковского процесса вы знаете?
6. Какие процессы носят название Марковских цепей?
7. Алгебраическая и геометрическая интерпретация Марковских цепей.
8. Какие Марковские процессы называют поглощающими?
9. Как задать поглощающие состояния системы в Марковском процессе?
10. Как вычисляется среднее время функционирования системы с поглощающими состояниями?
11. Дайте определение спектра матрицы.
12. Приведите общий вид характеристического уравнения.
13. В чем смысл элементов переходной матрицы?
14. Как определить переходную матрицу спустя m шагов?
15. Для чего задается вектор начального состояния системы?
16. Как определить вектор состояния системы спустя m шагов?
17. Какие компоненты включает размеченный граф стохастической системы?
18. В чем различие размеченного графа дискретных и непрерывных Марковских цепей?
19. Что понимают под предельными вероятностями состояний?
20. Какие Марковские процессы называют эргодическими?

21. Какое состояние системы называется возвратным?
22. В чем смысл координат стационарного вектора?
23. Как определяется среднее относительное пребывание системы в данном состоянии?

Задание 2.

Расчет переходного режима непрерывного Марковского процесса на основе численного решения системы линейных дифференциальных уравнений.

Составить СДУ для своего варианта. Построить графики переходных режимов. Найти стационарные вероятности.

Вариант задания включает матрицу интенсивности переходов (**MPr.xlsx)** и соответствующий граф переходов (**Схемы ОТН.docx**).

Таблица Л.3. - Вариант задания для непрерывного МП

Вариант	Схема	Матрица				
Nвар	Nсх		1	2		9
		1	λ_{11}	λ_{12}		λ_{19}
		2	λ_{21}	λ_{22}		λ_{29}
		9	λ_{91}	λ_{92}		λ_{99}

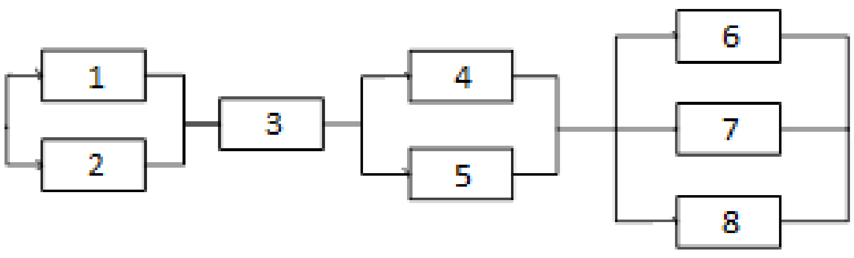


Рисунок Л.13. - Пример графа переходов

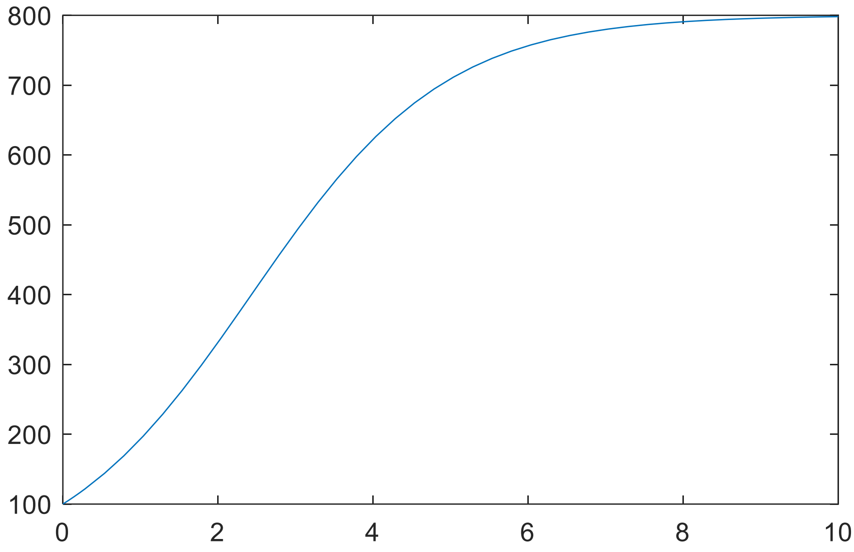
В качестве начального состояния выбирается первое (1). Интенсивности переходов, которых нет на диаграмме переходов не использовать или сделать нулевыми.

Теоретические положения 2

Решение ДУ и СДУ

Программа решения ДУ (Ex_DU_ode45.m)

```
clear all % очистить память
[t y]=ode45(@(t,y) 0.8*y*(1-(y/800)),[0 10], 100);
figure('Position',[100 200 500 300]);
plot(t,y)
```



Программа решения СДУ (Ex_SysDU_ode45.m)

```
clear all % очистить память
t=[0 5];  y0=[ 2 4];
[ T Y] = ode45(@myfun, t, y0);
T=T(1:15);  Y=Y(1:15,:);  plot(Y(:,1),Y(:,2));
 function  dy=myfun(t,y)
dy=zeros(2,1);
dy(1)=y(1)+2*y(2)+t;
dy(2)=2*y(1)+y(2)+t;
end
```

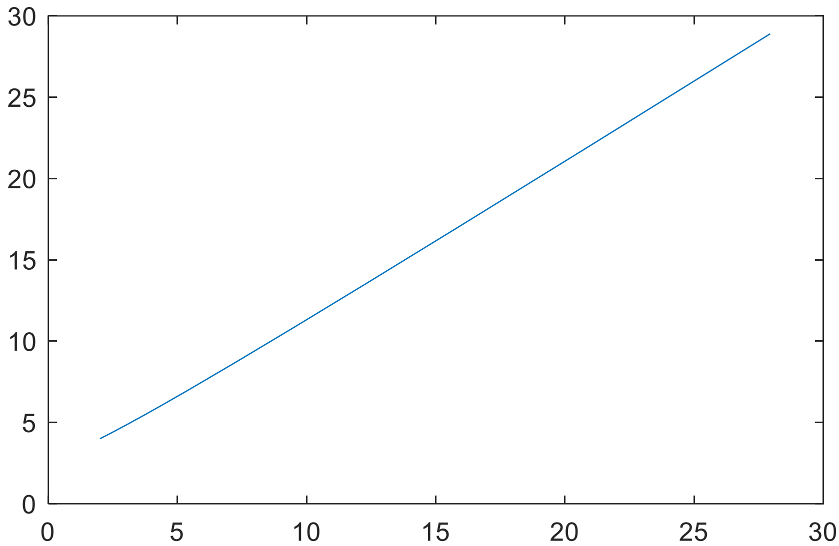


Рисунок Л.14. - Выборочные траектории частот состояний МЦ

Пример 2. Решение системы ДУ

Расчет нестационарного процесса изменения вероятности состояний (MarP_SDU.m)

```
% Моделирование надежности - дифуры Matlab
clear all
t=[0 5];  y0=[1 0 0 0 0];  [ T Y] = ode45(@myfun, t, y0);
figure('Position',[100 200 500 300]);
plot(T,Y(:,1:5),'LineWidth',3);  %plot(Y(:,2),Y(:,3));
function  dy=myfun(t,y)  % фнкция расчета состояний системы
dy=zeros(5,1);
dy(1)=-0.5*y(1)+0.8*y(4)+2*y(5);
dy(2)=0.5*y(1)-2*y(2);
dy(3)=2*y(2)-3*y(3);
dy(4)=1.5*y(3)-0.8*y(4);
dy(5)=1.5*y(3)-2*y(5);
end
```

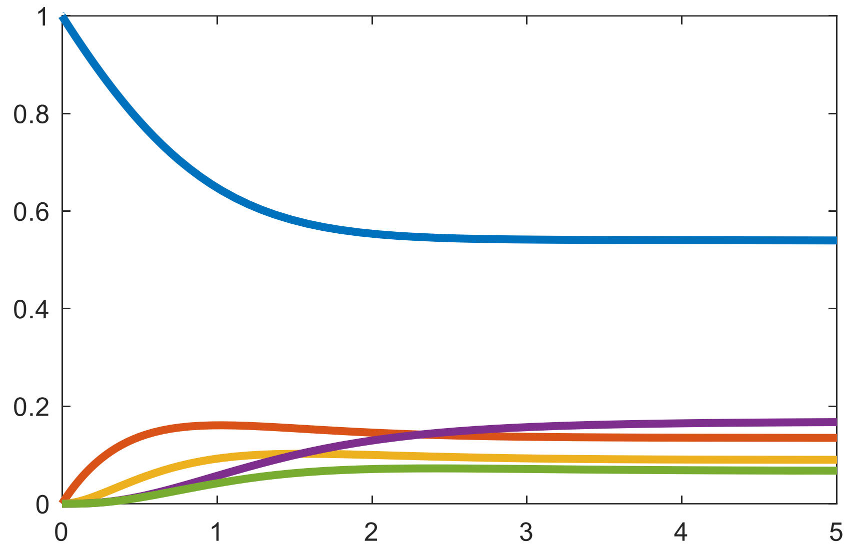


Рисунок Л.15. - Выборочные траектории частот состояний МЦ

Пример 2. Имитационная модель расчета стационарных вероятностей

```
% Непрерывная Марковская цепь
clear all;  Ns=5; N=500000;
MP=[0   0.5 0  0   0 ;
    0   0   2  0   0;
    0   0   0  1.5 1.5;
    0.8 0   0  0   0;
    2   0   0  0   0];
SS=zeros(N+1,3); % Время        Состояние   Пребывание
Tsys=0;           SS(1,1)=Tsys;  SS(1,2)=1;   SS(1,3)=1;
for t=1:1:N
  Smin=0; Tmin=10^20; Sw=SS(t,2);
  for s=1:1:Ns
    if MP(Sw,s)==0 T=10^20;   else T=random('exp',(1/MP(Sw,s)),1,1); end;
    if T<Tmin Smin=s; Tmin=T;  end;
  end
   SS(t,3)=Tmin; Tsys=Tsys+Tmin; SS(t+1,1)=Tsys; SS(t+1,2)=Smin;
end
TTT=sum(SS(:,3)); Tstat=zeros(Ns,1); % Обработка статистики
for t=1:1:N
 Sw=SS(t,2); Tstat(Sw)=Tstat(Sw)+SS(t,3);
end
Tstat=Tstat/Tsys;
```

Стационарные вероятности
0.539091893538387
0.134897227183320
0.0905503890379036
0.167995060676895
0.0674654295634809

Типовые вопросы:

1. Дайте определение основных свойств потока событий.
2. Понятие простейшего потока событий. Какими свойствами он обладает?
3. Как составить систему дифференциальных уравнений Колмогорова для конечной Марковской цепи с непрерывным временем?
4. Как вычислить предельные вероятности состояний цепи с непрерывным временем?

Л.7. Расчет надежности программного обеспечения по модели Джелински–Моранды

Задание

Вариант исходных данных для расчета по модели Джелински–Моранды представляют таблицу длительностей интервалов между отказами ПО (**DM_1.xlsx**).

Таблица Л.4. - Вариант задания для непрерывного МП

Интервалы	t, ч.
1	t_1
…	…
k-1	t_k

Для заданной таблицы интервалов отказов дать оценку среднего времени до следующего отказа.

Теоретические положения модели Джелински–Моранды

Алгоритм модели Джелински–Моранды предполагает выполнение следующих шагов:

Шаг 1. Тестирование программы. В таблицу «Результаты тестирования» записываем интервалы времени (часы) безотказной работы программы. При этом предполагается, что после каждого отказа ошибка должна исправляться, и при исправлении не вносятся новые ошибки (k – номер прогнозируемого отказа, t_i– длительность i-го интервала).

Результаты тестирования

Интервалы	t, ч.
1	t_1
…	…
k-1	t_k

Шаг 2. Нахождение первоначального количества ошибок в программе (N). Данный параметр определяется методом подбора по формуле $(k-1)\frac{\sum_{i=1}^{k-1} t_i}{\sum_{i=1}^{k-1} \frac{1}{N-i+1}} = \sum_{i=1}^{k-1}\left((N-i+1)\cdot t_i\right)$. Так как N может принимать только целые значения, то необходимо добиться минимальной разницы между правой и левой частями формулы.

Шаг 3. Определить интенсивность отказов: $\lambda_k = C_D(N-(k-1))$, где $C_D = \frac{\sum_{i=1}^{k-1} \frac{1}{N-i+1}}{\sum_{i=1}^{k-1} t_i}$ - коэффициент пропорциональности.

Шаг 4. Найти среднее время до следующего отказа: $t_k = \frac{1}{\lambda_k}$.

Шаг 5. Найти вероятность отсутствия k-го отказа $P(t_k) = \boldsymbol{e}^{-\lambda_k t_k}$.

Конец алгоритма.

Пример

Расчет надежности программного обеспечения по модели Джелински – Моранды.

Пусть в ходе отладки зафиксированы интервалы времени, представленные в таблице (шаг 1 модели):

Результаты тестирования

Интервалы	t_i, ч.
1	10
2	20
3	25

Необходимо определить вероятность отсутствия следующего (четвертого) отказа.

Решение. Находим первоначальное количество ошибок N (шаг 2).

Если N=3, то в левой части формулы имеем $3\frac{10+20+25}{1/3+1/2+1}=90$,

а в правой части: 3·10+2·20+1·25=95. При N=4 левая и правая части соответственно равны 152 и 150, если N=5, то – 210 и 205. Следовательно, наименьшую ошибку при решении уравнения обеспечит N=4.

Находим интенсивность отказов (шаг 3). Коэффициент пропорциональности $C_D=\frac{1/4+1/3+1/2}{10+20+25}=0.02$. Интенсивность отказов λ_4=0,02·(4-(4-1))=0,02.

Среднее время до следующего отказа (шаг 4) $t_4=\frac{1}{0.02}=50$.

Вероятность отсутствия четвертого отказа $P(t_4)=\boldsymbol{exp}(-0.02\cdot 50)=e^{-1}$.

Типовые вопросы

1. Выявите характер изменения показателя надежности по модели Джелински – Моранды в ходе устаревания исследуемого программного обеспечения.
2. Предложите формальное описание оптимизационной задачи по повышению надежности программного обеспечения
3. Оцените сложность построения бинарного дерева прецедентов для логико-вероятностного подхода оценки надежности.
4. Приведите разумные на ваш взгляд требования к надежности произвольного общеупотребительного программного средства, выраженные в допустимых значениях показателей, приведенных в данной книге.
5. Проведите экспериментальное исследование степени влияния протоколов обмена данными на надежность сервисов. Обращайте особое внимание на время обработки тестовых данных.
6. Проведите экспериментальное исследование функции распределения отказов произвольного программного средства. Укажите, всегда ли данная функция представляет собой нормальную.

7. Проанализируйте правомерность экстраполяции результатов исследования надежности программного обеспечения на длительные промежутки времени.

Л.8. Расчет надежности программного обеспечения по модели Ла Падула

Задание

Исходные данные расчета по модели представляют таблицу количества отказов на каждом этапе тестирования.

Интервалы	Отказы
1	m_1
…	…
k	m_k

Каждая строка файла **LP_1.xlsx** представляет таблицу для соответствующего варианта. Полагается, что количество тестов на каждом этапе неизменно и S_i=100

Для заданного количества отказов ПО в результате последовательности тестов дать оценку среднего интервала до следующего отказа.

Теоретические положения

По этой модели выполнение последовательности тестов производится в *m* этапов [2]. Каждый этап заканчивается внесением изменений (исправлений) в ПО. Возрастающая функция надежности базируется на числе ошибок, обнаруженных в ходе каждого тестового прогона.

Надежность ПО в течение i-го этапа

$$P(t) = P(\infty) - \frac{A}{i}, \; i = 1,2,... \tag{Л.4}$$

где *A* – параметр роста; $P(\infty) = \lim\limits_{i \to \infty} P(i)$ – предельная надежность ПО. Эти неизвестные величины в данной модели предлагается найти из решения системы уравнений

$$\sum_{i=1}^{m} \left(\frac{S_i - m_i}{S_i} - P(\infty) + \frac{A}{i} \right) = 0, \tag{Л.5}$$

$$\sum_{i=1}^{m}\left(\frac{S_i - m_i}{S_i} - P(\infty) + \frac{A}{i}\right)\cdot\frac{1}{i} = 0, \tag{Л.6}$$

где S_i – число тестов на i-м этапе; m_i – число отказов во время i-го этапа, i = 1,2,…

Модель является прогнозной и на основании данных тестирования позволяет предсказать вероятность безотказной работы программы на последующих этапах ее выполнения.

$$F\left(A, P(\infty)\right) = \sum_{i=1}^{m}\left(\frac{S_i - m_i}{S_i} - P(\infty) + \frac{A}{i}\right)^2 \to \boldsymbol{min}$$

$$\frac{\partial F\left(A, P(\infty)\right)}{\partial A} = 0 \qquad \frac{\partial F\left(A, P(\infty)\right)}{\partial P(\infty)} = 0$$

Пример расчета

Типовые вопросы

Список источников

1. Викторова, В.С., Степанянц, А.С. Модели и методы расчета надежности технических систем / В.С.Викторова, А.С.Степанянц – Москва, 2013. – 219 с.

2. Ефремов А.А. Теория надежности: конспект лекций / А.А.Ефремов - Издательство Томского политехнического университета. – Томск, 2015 – 79 с.

3. Федотов, А.В., Скабкин, Н.Г., Основы теории надежности и технической диагностики: конспект лекций / А. В. Федотов, Н. Г. Скабкин. – Омск: Изд-во ОмГТУ, 2010. – 64 с.

4. Неизвестный

Приложение. Программы генерации вариантов заданий

Л.1. Генерация матрицы переходных вероятностей марковской цпи МП (Mar_GVar.m)

Л.2. Генерация данных для непрерывного МП (Mar_GVar.m)

```
clear all;          % Генерация данных  для непрерывного МП
k=1;
MPr=ones(1200,12)*99999;
for v=1:1:120
    for sv=1:1:10
     if sv==1
         MPr(k,1)=v; MPr(k,2)=ceil(rand*60);
         NN=1:1:9;  MPr(k,4:12)=NN; k=k+1;
     else
       MPr(k,3)=sv-1;   %  Номер строки матрицы
       MPr(k,4:12)=ceil(rand(1,9)*1000)/10000;
       k=k+1;
     end
    end
end
writematrix(MPr,'MarP_Dat_1.xlsx');
```

Результат (MarP_Dat.xlsx)

Вар	Схема		Матрица								
1	9		1	2	3	4	5	6	7	8	9
		1	0,094	0,088	0,033	0,046	0,072	0,043	0,054	0,095	0,05
		2	0,082	0,074	0,016	0,079	0,001	0,072	0,049	0,034	0,076
		3	0,071	0,044	0,082	0,082	0,082	0,05	0,016	0,059	0,091
		4	0,029	0,036	0,033	0,073	0,07	0,028	0,087	0,009	0,007
		5	0,097	0,063	0,077	0,015	0,031	1E-04	0,056	0,092	0,045
		6	0,035	0,051	0,009	0,076	0,039	0,015	0,039	0,027	0,03
		7	0,001	0,024	0,046	0,014	0,038	0,045	0,053	0,055	0,033
		8	0,022	0,015	0,056	0,033	0,065	0,079	0,081	0,068	0,028
		9	0,052	0,055	0,048	0,014	0,034	0,087	0,025	0,019	0,053
2	23		1	2	3	4	5	6	7	8	9
		1	0,085	0,071	0,069	0,031	0,002	0,032	0,075	0,022	0,096

Л3. Генерация данных для оценки распределений и их параметров (Mod_GVar_Weib.m)

```
% Статистика наработки на отказ
clear all;   mm='weibull'; n=500; V=120;
%  Генерация параметров распределения для вариантов
for i=1:1:V
   Parm_Weib(i,1)=i;Parm_Weib(i,2)=1+20*rand; Parm_Weib(i,3)=0.5+2.5*rand;
end
% Генерация данных
Y=zeros(n+1,V);  Y(1,1:V)=1:1:V;
for i=1:1:V
   Y(2:n+1,i)=random(mm,Parm_Weib(i,1),Parm_Weib(i,2),n,1);
end
figure(1); histfit(Y(2:n+1,5),20,'weibull');
writematrix(Parm_Weib,'Mod_Parm_Weib.xlsx');
writematrix(Y,'Mod_Dat_Weib.xlsx');
```

Результат – параметры распределения Вейбулла (Mod_Parm_Weib.xlsx)

1	9,51608346	1,331123778
2	5,4495975	2,989573966
3	17,4431052	1,443272984
4	10,2492955	0,548686901
5	7,31289529	0,616441634
6	17,3233458	1,297997202
7	12,1651199	1,796222037
8	20,2951105	1,682050427
9	2,64893258	1,211150962
10	4,39488598	1,982144974

Результат – сгенерированные данные по распределению Вейбулла (Mod_Dat_Weib.xlsx)

1	2	3	4	5	6	7
0,947682395	2,206798833	2,96384407	2,73396067	5,49501758	5,92478651	6,64362024
0,970724266	2,334632672	3,01505804	4,00182586	3,90936115	6,22312688	6,81472287
1,06593474	1,633389017	3,07631273	3,82791962	5,02521303	5,62848281	7,1438216
0,974756578	1,630351177	2,97327766	3,82920847	4,24294597	5,64903305	6,94089053
1,030756029	2,323639399	3,17022794	4,15317587	5,26587975	5,71843288	5,98272161
0,792756865	2,218379086	2,77008227	3,14527225	5,45640493	6,42904049	6,8684637
0,912936785	2,122682419	2,86821124	4,17182856	5,75078849	5,35248849	6,32044541
1,065103446	1,582540313	2,95307223	3,99012089	4,23672209	5,91538217	6,62492714
0,957053926	2,271238733	3,08181486	3,51581995	3,44732777	5,53494992	7,06296509
0,957584504	2,311450554	3,19950323	3,70263676	4,91743632	5,55456101	6,9807601

Л4. Генерация данных для систем без восстановления и с восстановлением (ParPos_calc.m)

Л5.Генерация данных для параллельно-последовательной схемы надежности (ParPos_calc.m)

```
% Генерация данных  для параллельно-последовательной
clear all;
k=1;  DLPP=ones(120*9,7)*99999;
for v=1:1:120
   for sv=1:1:9
    if sv==1
       DLPP(k,1)=v; DLPP(k,2)=ceil(rand*60);
    end
     DLPP(k,3)=sv;
     DLPP(k,4)=rand*0.01;  % Экспоненциальное
     DLPP(k,5)=rand*0.02;  % Релея
     DLPP(k,6)=rand*0.015; % a - Вейбулла
     DLPP(k,7)=1+rand*2;   % b - Вейбулла
     k=k+1;
   end
end
writematrix(DLPP,'Dat_ParPos.xlsx');
```

Результат генерации (Dat_ParPos.xlsx)

Вар	Схема		Экспоненциальное	Релея	a - Вейбулла	b - Вейбулла
1	51	1	0,00816527	0,016924567	0,005552803	1,766459063
		2	0,008613347	0,009278184	0,008558223	2,390613617
		3	0,009609173	0,01092626	0,009548649	2,141784978
		4	0,009271121	0,017275311	0,002547589	1,357398058
		5	0,002435044	0,015035587	0,002987016	2,965881023
		6	0,007096387	0,003508723	0,012874461	2,818822966
		7	0,009616631	0,011411994	0,008443188	1,353322409
		8	0,005136791	0,010969438	0,002479162	1,987786077
		9	0,005351173	0,003976143	0,009347534	1,052630787
2	20	1	0,005329995	0,006535475	0,009032857	1,723865455
		2	0,001349206	0,01827627	0,009608383	2,317548285
		3	0,006753303	0,014891154	0,012632664	2,033314335

Пример программы расчета надежности

```
% Расчет показателей надежности параллельно-последовательной схемы
clear all;
T=[50 100 500 1000];   % Список времен наработки
% Параметры надежности элементов
L=[0.006728552 0.012293569 0.007594606 2.469596063
0.006492977 0.01502936 0.01079888 1.132955961
0.009379852 0.004512878 0.010795919 2.739985648
0.00202783 0.006352012 0.004464648 1.079832856
0.008475591 0.001502414 0.004324431 1.08244536
0.008971157 0.003974554 0.013892789 2.800490058
0.00997014 0.017084086 0.010665934 1.825712911
0.000138318 0.011155775 0.012577868 1.191457192];
% Корректировка
L(1:8,2)=L(1:8,2)*10000;  L(1:8,3)=L(1:8,3)*100000;
for t=1:1:4  % Расчет показателя надежности общей модели
   for r=1:1:3
      if  r==1
         for i=1:1:8   P(i)=exp(  -L(i,1)*T(t) );  end;
      elseif r==2
          for i=1:1:8   P(i)=exp( -(T(t)^2)/(2*L(i,2)^2)    );  end;
      else
          for i=1:1:8   P(i)=exp(   -(T(t)/ L(i,3))^L(i,4)   );  end;
      end
      % схема 4
        p23=P(2)*P(3);
        p56=P(5)*P(6);
        p23456=1-(1-p23)*(1-p56)*(1-P(4));
        p78=1-(1-P(7))*(1-P(8));
        pall=P(1)*p23456*p78;
        Rez(t,r)= pall;
     end
end
```

Результат расчета

0.690484464914308	0.797975725064037	0.998460910462497
0.444872338182504	0.230480679033888	0.990489102116918
0.0117259939613929	1.24289765835272e-19	0.525391859324792
0.000137114797731264	0	0.0281774017320085

Л7. Генерация данных для модели Джелински-Моранды (GVar_PO_DgMor.m)

```
clear all;         % Генерация данных  для Джелински-Моранда
DM=ones(120,12)*99999;
for v=1:1:120
     DM(v,1)=v;
     for n=3:1:12
          DM(v,n)=20+n*2+ceil(5*rand);
```

```
    end
end
writematrix(DM,'DM.xlsx');
```

Результат (Dat_PO_DgMor.xlsx)

Вар		t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
1		27	32	32	35	35	40	40	44	46	48
2		29	29	32	37	35	41	41	45	43	47
3		27	33	31	36	39	41	39	42	44	49
4		29	33	31	34	35	37	43	43	45	45
5		31	32	32	35	37	37	40	41	43	46
6		29	29	35	37	37	39	40	45	44	45
7		30	30	32	35	35	37	43	45	45	45
8		28	30	35	33	35	37	42	44	46	47
9		29	30	34	33	38	37	40	44	46	45
10		31	32	33	35	37	38	41	43	47	48
11		30	30	35	35	36	41	43	43	46	47

Л8. Генерация данных для модели Ла Падула (GVar_PO_LaPod.m)

```
clear all;          % Генерация данных  для Ла Падула
LP=ones(120,12)*99999;
for v=1:1:120
    LP(v,1)=v;
    for n=3:1:12
        m=40+5*(n-2);
        LP(v,n)=100-(m+ceil((98-m)*rand));
    end
end
writematrix(LP,'LP_3.xlsx');
```

Результат (Dat_PO_LaPod.xlsx)

Вар		m1	m2	m3	m4	m5	m6	m7	m8	m9	m10
1		9	17	22	28	11	19	11	4	4	2
2		5	10	44	39	32	22	24	12	10	5
3		5	35	30	7	23	26	13	4	10	4
4		21	28	24	3	32	22	14	9	3	6
5		31	14	24	7	19	16	13	15	13	9
6		7	38	7	12	6	3	21	12	2	4
7		7	26	44	16	27	15	8	9	7	6
8		42	29	44	16	3	27	24	4	11	9
9		11	43	7	36	23	13	11	7	6	6
10		47	13	34	15	6	27	2	19	4	3