**Лабораторная работа №1**

**«Разработка модели дискретно-стохастической СМО»**

**Задание**

Для СМО заданной конфигурации построить имитационную модель.

Распределение интервалов времени между заявками во входном потоке и интервалов времени обслуживания – геометрическое с соответствующим параметром (ρ, π1, π2). Если ρ не задано, то входной поток – регулярный (с указанным в обозначении источника числом тактов между заявками).

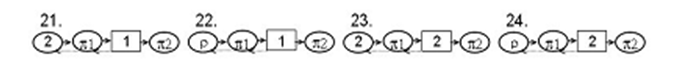
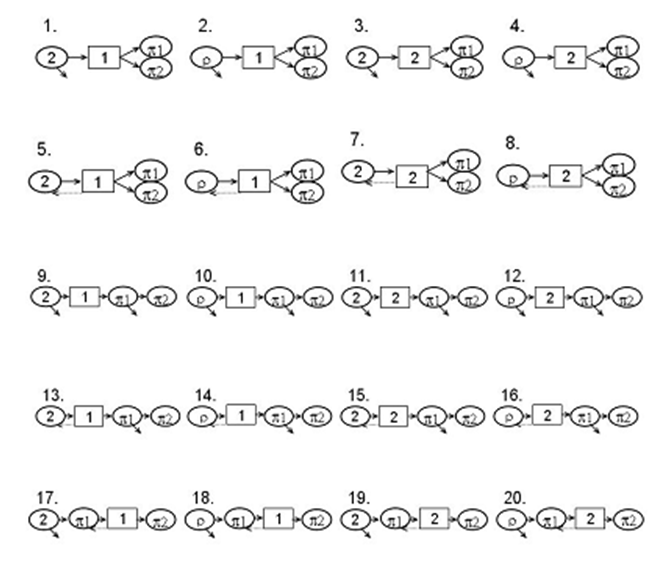
Ротк – вероятность отказа;

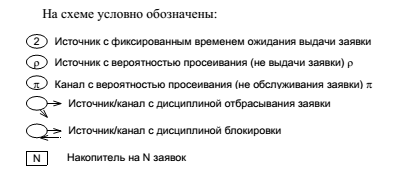
Рбл – вероятность блокировки;

Lоч – средняя длина очереди; Q – относительная пропускная способность;

А – абсолютная пропускная способность.

**Варианты заданий**





|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | ρ | π 1 | π 2 | Цель исследования |
| 1 | - | 0,8 | 0,6 | Ротк, Wоч |
| 2 | 0,5 | 0,6 | 0,8 | Ротк, Wс |
| 3 | - | 0,6 | 0,5 | Lоч, А |
| 4 | 0,5 | 0,5 | 0,6 | Lоч, Wоч |
| 5 | - | 0,75 | 0,6 | Рбл, Wс |
| 6 | 0,75 | 0,85 | 0,9 | Рбл, Wоч |
| 7 | - | 0,75 | 0,6 | Lоч, А |
| 8 | 0,75 | 0,8 | 0,5 | Lоч, А |
| 9 | - | 0,4 | 0,4 | Q, А |
| 10 | 0,5 | 0,45 | 0,35 | Q, Wс |
| 11 | - | 0,4 | 0,4 | А, Lоч |
| 12 | 0,5 | 0,4 | 0,4 | А, Wоч |
| 13 | - | 0,55 | 0,5 | Рбл, А |
| 14 | 0,7 | 0,65 | 0,5 | Рбл, Q |
| 15 | - | 0,48 | 0,5 | Lоч, Ротк, |
| 16 | 0,5 | 0,48 | 0,5 | Lоч, Q |
| 17 | - | VAR | 0,5 | Зависимость Ротк от π1, π1=0,2(0,2)1 |
| 18 | 0,5 | VAR | 0,4 | Зависимость Ротк от π1, π1=0,2(0,2)1 |
| 19 | - | 0,3 | VAR | Зависимость Lоч от π2, π2=0,2(0,2)1 |
| 20 | 0,5 | 0,35 | VAR | Зависимость Lоч от π2, π2=0,2(0,2)1 |
| 21 | - | 0,4 | VAR | Зависимость Рбл источника от π2, π2=0,2(0,2)1 |
| 22 | 0,75 | 0,7 | VAR | Зависимость Рбл источника от π2, π2=0,2(0,2)0,8 |
| 23 | - | 0,4 | 0,5 | А, Lоч |
| 24 | 0,75 | 0,7 | 0,65 | Рбл, Wс |

**Теоретические сведения**

В общем виде вероятностный автомат (англ. Probabilistic automat) можно определить как дискретный потактный преобразователь информации с памятью, функционирование которого в каждом такте зависит только от состояния памяти в нем и может быть описано статистически. *Состояние преобразователя считывается каждый такт.*

Следовательно, при выполнении моделирования работы схемы будем задавать количество анализируемых тактов (≈10000).

Распределение интервалов времени между интервалами времени обслуживания – **геометрическое** с соответствующим параметром (π1, π2).

***Просеянный поток*** – регулярный поток, в котором удалены события с вероятностью q и оставлены с вероятностью 1-q.

***Геометрическому распределению*** соответствует выражение:

Pi = qi-1(1- q) – вероятность того, что величина интервала между событиями ***в просеянном потоке*** окажется равным i тактам:

Следовательно, **вероятность того что заявка будет обслужена за 1 такт** составит P1 = (π)1-1·(1-π) = 1-π, где π – вероятность необслуживания заявки.

**Пример выполнения**

для Варианта 7

1) Исходные данные

Р-схема

2

π

π

2

|  |  |  |  |
| --- | --- | --- | --- |
| *ρ* | *π*1 | *π*2 | Цель исследования |
| - | 0,75 | 0,6 | Lоч, А |

2) Анализ задания

Схема содержит источник с блокировкой и фиксированным временем ожидания выдачи заявки (2 такта до выдачи), накопитель на 2 заявки два канала с вероятностями просеивания (не обслуживания заявки) π1 и π2.

Граф состояний кодируется четырехкомпонентным вектором *TNК1К2,* где

*T* – время до выдачи очередной заявки источником, *T*={2,1,0}

2 – два такта до выдачи заявки

1 – один такт до выдачи заявки (по окончании такта заявка поступит в канал обслуживания)

0 – означает, что источник заблокирован (заявка заблокирована в источнике);

*N* – количество заявок, находящихся в накопителе (длина очереди), *N*={0,1,2}

0 – заявок в очереди на обслуживание нет

1 – одна заявка в очереди

2 – заявки в очереди;

*К1* и *К2* – состояние каналов обслуживания, *К1* (*К2*)={0,1}

0 – канал свободен

1 – канал занят обслуживанием заявки.

3) Рассмотрев все возможные состояния системы, строим матрицу переходов, в виде таблицы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| zk | zk | | | | |
| z1 | z2 | … | zk-1 | zk |
| z1 | р1,1 | р1,2 | … | р1,k-1 | р1,k |
| z2 | р2,1 | р2,2 | … | р2,k-1 | р2,k |
| … | … | … | … |  | … |
| zk | рk,1 | рk,2 | … | рk,k-1 | рk,k |

Где 

Переход из состояния S1 в состояние S1 не возможен, следовательно вероятность перехода равна 0

Таблица 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 2000 | 1000 | 2010 | 1010 | 2011 | 1011 | 1001 | 2111 | 1111 | 2211 | 1211 | 0211 |
| 1 | 2000 | 0 | р1,2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1000 | 0 | 0 | р2,3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 2010 | 0 | р3,2 | 0 | р3,4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1010 | 0 | 0 | р4,3 | 0 | р4,5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 2011 | 0 | р5,2 | 0 | р5,4 | 0 | р5,6 | р5,7 | 0 | 0 | 0 | 0 | 0 |
| 6 | 1011 | 0 | 0 | р6,3 | 0 | р6,5 | 0 | 0 | р6,8 | 0 | 0 | 0 | 0 |
| 7 | 1001 | 0 | 0 | р7,3 | 0 | р7,5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 2111 | 0 | 0 | 0 | р8,4 | 0 | р8,6 | 0 | 0 | р8,9 | 0 | 0 | 0 |
| 9 | 1111 | 0 | 0 | 0 | 0 | р9,5 | 0 | 0 | р9,8 | 0 | р9,10 | 0 | 0 |
| 10 | 2211 | 0 | 0 | 0 | 0 | 0 | р10,6 | 0 | 0 | р10,9 | 0 | р10,11 | 0 |
| 11 | 1211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | р11,8 | 0 | р11,10 | 0 | р11,12 |
| 12 | 0211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | р12,8 | 0 | р12,10 | 0 | р12,12 |

Граф состояний представлен на рисунке 1.

4) Определим вероятности переходов рi,j и заполним таблицу переходов числовыми значениями.

1. Полагаем, что после запуска генератора заявки будут гарантированно формироваться и поступать в канал обслуживания после прохождения двух тактов. Следовательно, вероятности перехода из состояния “2000” в состояние “1000” и состояния “1000” в состояние “2010” будут равны единице.
2. Наступление состояния “1010” зависит только от того, обслужит первый канал заявку или нет. Следовательно, переход из состояния “2010” в состояние “1010” определяется вероятностью необслуживания заявки π1 (р3,4 = π1).

Сумма вероятностей полной группы S несовместных событий составляет равна единице . Следовательно, вероятность того, заявка будет обслужена первым каналом, и система вернется из состояния “2010” в состояние “1000”, будет равна 1- π1 (р3,2 = 1-π1).

Аналогично р4,5 = π1, р4,3 = 1-π1.

1. Обслуживание заявки каждым каналом в данной системе представляет собой **независимое событие**. Для определения вероятностей переходов из состояния “2011” в состояние “1011” и состояние “1001” воспользуемся следующими правилами

***Теорема умножения вероятностей.***

Произведением двух событий р(А·В) называют событие АВ, состоящее в **совместном** появлении этих событий:

р(АВ) = р(А)·р(В|А) = р(В)·р(А|В),

где р(В|А) (р(А|В)) – условная вероятность события В (А), вычисленное в предположении того, что событие А (В) наступило.

Если события **независимы**, то р(В|А) = р(В), р(А|В) = р(А)

Тогда **вероятность наступления двух независимых событий**

р(АВ) = р(А)·р(В)

Значит, вероятность перехода состояния “2011” в “1011” можно определить как произведение вероятностей того, что заявки в обоих каналах не будут обслужены, т.е. р5,6 = π1·π2, а состояния “2011” в “1001” - как произведение вероятностей того, что первый канал завершит обслуживание заявки, а второй – еще нет, т.е. р5,7 = (1-π1)·π2.

1. Переход из состояния “1011” в состояние “2011” возможен в двух случаях:

- первый канал продолжит обслуживание заявки, в то время как второй, завершив обслуживание, перейдет к обслуживанию второй заявки;

- второй канал продолжит обслуживание заявки, в то время как первый, завершив обслуживание, перейдет к обслуживанию второй заявки.

Причем первый вариант развития событий исключает возможность появления другой.

События называются **несовместными**, если появление одного из них исключает появление другого в одном и том же испытании.

**Вероятность появления одного из двух несовместных событий** равна сумме вероятностей этих событий:

р(А+В) = р(А) + р(В)

Следовательно, вероятность перехода состояния “1011” в “2011” р6,5 можно определить следующим образом:

р6,5 = (1-π1)·π2 + π1·(1-π2) = π1 + π2 - 2π1π2.

1. Руководствуясь приведенными рассуждениями, определяем вероятности остальных переходов и заполняем таблицу переходов (см. Таблица 2.а)

Таблица 2.а

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | **∑** |
| 2000 | 1000 | 2010 | 1010 | 2011 | 1011 | 1001 | 2111 | 1111 | 2211 | 1211 | 0211 |
| 1 | 2000 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 1000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 2010 | 0 | 1-π1 | 0 | π1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 1010 | 0 | 0 | 1-π1 | 0 | π1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 2011 | 0 | (1-π1)(1-π2) | 0 | π1(1-π2) | 0 | π1π2 | (1-π1)π2 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6 | 1011 | 0 | 0 | (1-π1)(1-π2) | 0 | π1+π2-2π1π2 | 0 | 0 | π1π2 | 0 | 0 | 0 | 0 | 1 |
| 7 | 1001 | 0 | 0 | 1-π2 | 0 | π2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8 | 2111 | 0 | 0 | 0 | (1-π1)(1-π2) | 0 | π1+π2-2π1π2 | 0 | 0 | π1π2 | 0 | 0 | 0 | 1 |
| 9 | 1111 | 0 | 0 | 0 | 0 | (1-π1)(1-π2) | 0 | 0 | π1+π2-2π1π2 | 0 | π1π2 | 0 | 0 | 1 |
| 10 | 2211 | 0 | 0 | 0 | 0 | 0 | (1-π1)(1-π2) | 0 | 0 | π1+π2-2π1π2 | 0 | π1π2 | 0 | 1 |
| 11 | 1211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | (1-π1)(1-π2) | 0 | π1+π2-2π1π2 | 0 | π1π2 | 1 |
| 12 | 0211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | (1-π1)(1-π2) | 0 | π1+π2-2π1π2 | 0 | π1π2 | 1 |

В последнем столбце для самопроверки просуммируем построчно вероятности переходов.

Подставим числовые значения:

Таблица 2.б

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | **∑** |
| 2000 | 1000 | 2010 | 1010 | 2011 | 1011 | 1001 | 2111 | 1111 | 2211 | 1211 | 0211 |
| 1 | 2000 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 1000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 2010 | 0 | 0,25 | 0 | 0,75 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 1010 | 0 | 0 | 0,25 | 0 | 0,75 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 2011 | 0 | 0,1 | 0 | 0,3 | 0 | 0,45 | 0,15 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6 | 1011 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0 | 0,45 | 0 | 0 | 0 | 0 | 1 |
| 7 | 1001 | 0 | 0 | 0,4 | 0 | 0,6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8 | 2111 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0 | 0,45 | 0 | 0 | 0 | 1 |
| 9 | 1111 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0 | 0,45 | 0 | 0,45 | 0 | 0 | 1 |
| 10 | 2211 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0 | 0,45 | 0 | 0,45 | 0 | 1 |
| 11 | 1211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0,45 | 1 |
| 12 | 0211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0,45 | 1 |

1

1

1-π1

π1(1-π2)

π2

π1

1-π1

π1

π2(1-π1)

π1π2

π1+π2-2π1π2

(1-π1)(1-π2)

1-π2

(1-π1)(1-π2)

π1π2

π1π2

π1π2

π1π2

π1π2

π1+π2-2π1π2

π1+π2-2π1π2

π1+π2-2π1π2

π1+π2-2π1π2

π1π2

(1-π1)(1-π2)

(1-π1)(1-π2)

(1-π1)(1-π2)

(1-π1)(1-π2)

π1+π2-2π1π2

(1-π1)(1-π2)

Рисунок 1 - Граф состояний

6) Расчет параметров данной системы с использованием построенной модели.

**а.** Исходя из графа состояний, построим систему уравнений для нахождения **вероятностей состояний**, воспользовавшись выражением:



дополнив систему нормировочным уравнением 



**б.** Исключим из системы уравнение , подставим значения *π*1=0,75 и *π*2=0,6 и приведем к каноническому виду:



**в.** Решив данную систему, получим значения вероятностей состояний:



Просуммировав для самопроверки вероятности состояний получим единицу.

**г.** Определим значения величин, являющихся целью исследования:

– средняя длина очереди *Lоч*:

,

где *i* – номер состояния,  *j* - число заявок в очереди в *i*-том состоянии, s – число состояний системы



– абсолютная пропускная способность (интенсивность потока обработанных заявок) – среднее число заявок, обслуживаемое СМО в единицу времени:



где *Ро* – вероятность того, что канал обрабатывал заявку(и) ,  – сумма вероятностей состояний, при которых не происходит обслуживание заявок; *Рз* – вероятность того, что обработка закончилась, Т – единица времени, за которую источник выдает 1 заявку.







**Лабораторная работа №2**

**Построение и исследование программного датчика равномерно-распределенных случайных чисел и последовательностей чисел с заданными законами распределения**

**Задание**

1) Разработать программный датчик равномерно-распределенных случайных чисел из интервала от 0 до 1 с использованием алгоритма Лемера.

2) По полученной выборке построить гистограмму, определить значения математического ожидания, дисперсии и среднего квадратического отклонения.

3) Оценить равномерность по косвенным признакам.

4) Найти величину периода и длину участка апериодичности.

5) Варьируя параметрами алгоритма, добиться максимальной длины периода.

6) Разработать программный датчик последовательности случайных чисел с заданным законом распределения:

− равномерное распределение в заданном интервале;

− нормальное распределение;

− показательное распределение;

− гамма-распределение;

− треугольное-распределение;

− распределение Симпсона и т.д.

7) По полученной выборке случайных чисел с заданным законом распределения построить гистограмму, определить значения математического ожидания, дисперсии и среднего квадратического отклонения.

**Теоретические сведения**

**I. Псевдослучайные числа**

При статистическом моделировании систем одним из основных вопросов является учёт стохастических воздействий. Для их формирования обычно используют последовательности случайных чисел с заданными вероятностными характеристиками.

Количество случайных чисел, требуемых для получения статистически устойчивых оценок параметров процессов функционирования системы S при реализации моделирующего алгоритма на ЭВМ, колеблется в достаточно широких пределах в зависимости от

- класса объекта моделирования,

- вида оцениваемых параметров,

- необходимой точности,

- достоверности результатов моделирования.

Метод статистического моделирования на ЭВМ

- большое число операций => большая доля машинного времени расходуется на действия со случайными числами.

- результаты статистического моделирования существенно зависят от качества исходных последовательностей случайных чисел.

* Простые и экономичные способы формирования последовательностей случайных чисел требуемого качества во многом определяет возможности практического использования машинного моделирования систем.

Алгоритмический способ получения последовательностей случайных чисел в настоящее время считается наиболее эффективным - каждое случайное число вычисляется с помощью соответствующей программы по мере возникновения необходимости.

Программная имитация случайных воздействий любой сложности сводится к генерированию некоторых стандартных (базовых) случайных процессов и к их последующему функциональному преобразованию. В качестве базового может быть принят любой удобный процесс. Как показывает практика, оптимальным базовым процессом является последовательность чисел {X}=x1,x2,…,xn, представляющих собой реализацию равномерно распределенной на интервале (0,1) случайной величины ξ, или - в статистических терминах - повторную выборку из равномерно распределенной на (0,1) генеральной совокупности значений величины ξ.

*Непрерывная случайная величина ξ имеет равномерное распределение в интервале (a,b), если ее функции плотности fξ(x) и функция распределения Fξ(x) соответственно имеют вид:*



*Числовые характеристики случайной величины ξ, принимающей значения x - математическое ожидание, дисперсию и среднее квадратическое отклонение соответственно*:



Для случайной величины равномерно распределенной на интервале (0,1), когда границы интервала a=0 и b=1, функция плотности и функция распределения соответственно имеют вид:



Такое распределение имеет математическое ожидание M[ξ]=0.5   
и дисперсию D[ξ]= (≈0,083), а σξ= (≈0,3).

Это распределение получить на цифровой ЭВМ невозможно, так как машина оперирует с n-разрядными числами. Поэтому на ЭВМ вместо непрерывной совокупности равномерных случайных чисел интервала (0,1) используют дискретную последовательность из 2n случайных чисел того же интервала. Закон распределения такой дискретной последовательности называют квазиравномерным распределением.

СВ ξ, имеющая квазиравномерное распределение в интервале (0,1) принимает значения  с вероятностями pi=i/(2n-1), i=0,…,2n-1.

Математическое ожидание и дисперсия квазиравномерной случайной величины соответственно имеют вид:



Таким образом, математическое ожидание квазиравномерной случайной величины совпадает с математическим ожиданием равномерной случайной последовательности интервала (0,1), а дисперсия отличается только множителем , который при достаточно больших n близок к единице.

На ЭВМ невозможно получить идеальную последовательность случайных чисел хотя бы потому, что на ней можно оперировать только с конечным множеством чисел. Кроме того, для получения значений x случайной величины ξ используются формулы (алгоритмы). Поэтому такие последовательности, являющиеся по своей сути детерминированными, называются псевдослучайными.

Набор требований, которым должен удовлетворять идеальный генератор:

- полученные с помощью идеального генератора псевдослучайные последовательности чисел должны состоять из квазиравномерно распределенных чисел,

- содержать статистически независимые числа,

- быть воспроизводимыми,

- иметь неповторяющиеся числа,

- получаться с минимальными затратами машинного времени,

- занимать минимальный объем машинной памяти.

Наибольшее применение в практике моделирования на ЭВМ для генерации последовательностей псевдослучайных чисел находят алгоритмы вида:

xi+1=Ф(xi),

представляющие собой рекуррентные соотношения первого порядка, для которых начальное число x0 и постоянные параметры заданы.

Рассмотрим некоторые процедуры получения последовательностей псевдослучайных квазиравномерно распределенных чисел, которые нашли применение в практике статистического моделирования систем на ЭВМ.

1. **Метод серединных квадратов.**

Имеется 2n - разрядное число, меньшее 1: xi=0,a1a2…a2n. Возведем его в квадрат: xi2=0,b1b2…b4n, а затем выделим средние 2n разрядов xi+1=bn+1bn+2…b3n, которые и будут являться очередным числом псевдослучайной последовательности.

Недостаток этого метода - наличие корреляции между числами последовательности, а в ряде случаев случайность вообще может отсутствовать. Например, если x0=0.4500, то (x0)2=0.20250000, x1=0.2500, (x1)2=0.06250000, , (x2)2=0.06250000, x3=0.2500 и т.д. Кроме того, при достижении некоторых  вообще может наблюдаться вырождение последовательности, т.е. xi=0 при . Это существенно ограничивает возможности использования метода серединных квадратов.

1. **Мультипликативный метод**

Задает последовательность неотрицательных целых чисел , не превосходящих , по формуле:



т.е. частный случай соотношения при .

В силу детерминированности метода получаются воспроизводимые последовательности. Требуемый объем машинной памяти при этом минимален, а с вычислительной точки зрения необходим последовательный подсчет произведения двух целых чисел, т.е. выполнение операции, которая быстро реализуется современными ЭВМ.

Для машинной реализации наиболее удобна версия

,

где  - число цифр в системе счисления, принятой в ЭВМ ( для двоичной и  для десятичной машины);

 - число бит в машинном слове.

Тогда вычисление остатка от деления на  - выделение  младших разрядов делимого,

преобразование целого числа  в рациональную дробь из интервала (0, 1) - подстановка слева от  двоичной или десятичной запятой.

Алгоритм построения последовательности для двоичной машины  сводится к выполнению следующих операций:

1) выбрать в качестве  произвольное нечетное число;

2) вычислить коэффициент , где  - любое целое положительное число;

3) найти произведение , содержащее не более  значащих разрядов;

4) взять  младших разрядов в качестве первого члена последовательности , а остальные отбросить;

5) определить дробь  из интервала (0,1);

6) присвоить ;

7) вернуться к п.З.

1. **Смешанный метод**

Позволяет вычислить последовательность неотрицательных чисел , не превосходящих , по формуле

,

т.е. в отличие от мультипликативного метода .

С вычислительной точки зрения смешанный метод генерации сложнее мультипликативного на одну операцию сложения, но при этом возможность выбора дополнительного параметра позволяет уменьшить возможную корреляцию получаемых чисел.

Качество конкретной версии такого генератора можно оценить только с помощью соответствующего машинного эксперимента.

В настоящее время почти все библиотеки стандартных, программ универсальных ЭВМ для вычисления последовательностей равномерно распределенных случайных чисел основаны на конгруэнтной процедуре.

Вычислительный алгоритм Д.Лемера - линейно-конгруэнтный метод генерации псевдослучайных чисел.

*Xi+1 = (aXi + c) mod m*

*a*,*c,m* — некоторые положительные целые числа 

Качество РРСЧ, весьма существенно зависит от выбора *a*, *c, m*, *X0* где параметры  и *X0* влияют на статистические свойства получаемых чисел, а параметр - на период их повторения.

Значение переменной *X0* должно быть:

а) меньше;

в) достаточно большим;

г) желательно простым числом;

г) содержать в двоичном представлена сравнительно большее число единиц.

В таблице ниже приведены наиболее часто используемые параметры линейных конгруэнтных генераторов, в частности, в стандартных библиотеках различных компиляторов (функция rand()).

При реализации выгодно выбирать M = 2n, где n — число битов в машинном слове, поскольку это позволяет избавиться от относительно медленной операции приведения по модулю.

|  |  |  |  |
| --- | --- | --- | --- |
| **Название библиотеки** | ***m*** | ***a*** | ***c*** |
| Numerical Recipes | 232 | 1664525 | 1013904223 |
| MMIX by Donald Knuth | 264 | 6364136223846793005 | 1442695040888963407 |
| Borland C/C++ | 232 | 22695477 | 1 |
| GNU Compiler Collection | 232 | 69069 | 5 |
| ANSI C: Open Watcom, Digital Mars, Metrowerks, IBM VisualAge C/C++ | 232 | 1103515245 | 12345 |
| Borland Delphi, Virtual Pascal | 232 | 134775813 | 1 |
| Microsoft Visual/Quick C/C++ | 232 | 214013 | 2531011 |
| Apple CarbonLib | 231 -1 | 16807 | 0 |

Приведенное соотношение имеет следующий смысл: *Xi+1* равно остатку от деления *(aXi + c)* на *m*.

Запишем алгоритм в виде пошаговой процедуры.

Ш а г 1. Коэффициент *a* умножается на число *Xi*

Ш а г 2. К результату умножения (*aXi*) прибавляется числ *с*

Ш а г 3. Результат сложения *(aXi + c)*  делится на *m*

*aXi + c = q·m+ Xi+1*

где *qm* - целая часть (*q* = 0,1,2,...), *Xi+1* - остаток от деления .

Ш а г 4. Остаток от деления *Xi+1* делится на *m*, чтобы получить искомое случайное число между нулем и единицей:

Следует отметить, что данный метод не обладает криптографической стойкостью, однако входит в большинство современных стандартных библиотек различных компиляторов.

Пример.

х0 = 7(0111)

а = 5 (0101)

с = 3 (0011)

m = 2n

n – число бит в машинном слове

Пусть n = 4, тогда m = 24 = 16

*х1* = (*a·х0 + c) mod m* = (5·7 + 3) *mod* 16 = 38 (00100110) *mod* 16 = 6 (0110)

*х1\** = 6 /16 = 0,375

*х2* = *a·х1 + c* = 5·6 + 3 = 33

*х2* = (*a·х1 + c) mod m* = (5·6 + 3) *mod* 16 = 33 (00100001) *mod* 16 = 1 (0001)

*х2\** = 1 /16 = 0,063

**……………….**

***Математическое ожидание*** квазиравномерной случайной величины:

N – количество элементов массива, – элемент нашей последовательности.

***Дисперсия квазиравномерной случайной величины -*** мера разброса случайной величины, т.е. её отклонение от математического ожидания.

N – количество элементов массива, – математическое ожидание, – элемент нашей последовательности.

СКО квазиравномерной случайной величины - показатель рассеивания значений случайной величины относительно её математического ожидания.

Эффективность статистического моделирования систем на ЭВМ и достоверность получаемых результатов существенным образом зависят от качества исходных (базовых) последовательностей псевдослучайных чисел, которые являются основой для получения стохастических воздействий на элементы моделируемой системы.

***Проверка качества последовательностей РРСЧ***

1) Проверка равномерности

2) Проверка стохастичности

3) Проверка независимости

Проверка равномерности последовательностей псевдослучайных квазиравномерно распределенных чисел {хi} может быть выполнена по гистограмме или с использованием косвенных признаков.

а) Проверка по гистограмме.

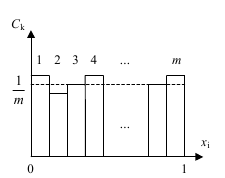
Суть проверки по гистограмме сводится к следующему. Выдвигается гипотеза о равномерности распределения чиел в интервале (0 1). Затем интервал (0 1) разбивается на m равных частей. При генерации последовательности РРСЧ подсчитывается количество попаданий Nk в каждый из m подинтервалов. Вычисляется относительная частота попадания случайных чисел последовательности {хi} в каждый из подинтервалов

Ck= Nk/N,

где N − общее количество чисел в последовательности {хi}.

Очевидно, что при равномерности последовательности чисел, частоты должны быть близкими при достаточно больших N к теоретической вероятности попадания в подинтервалы, равной 1/m.

Оценка степени приближения, т. е. равномерности последовательности{хi}, может быть проведена с использованием критериев согласия. На практике обычно принимается m = 20÷50, N = (102÷103)*m*.



б) Проверка по косвенным признакам.

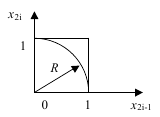
Вся последовательность {хi} разбивается на пары чисел:

(x1, x2), (x3, x4), ... , (x2i-1, x2i), ... , (xN-1, xN).

Затем подсчитывают число пар K, для которых выполняется условие:

<1

Геометрически это означает, что точка с координатами (x2i-1, x2i) расположена внутри четверти круга радиуса R=1, вписанного в единичный квадрат.



В общем случае точка (x2i-1, x2i) всегда попадет внутрь единичного квадрата. Тогда теоретическая вероятность попадания этой точки в четверть круга равна отношению площади четверти круга к площади единичного квадрата:

P = S1/4 круга/Sквадрата = π/4.

Если числа последовательности {хi} равномерны, то в силу закона больших чисел теории вероятностей при больших N относительная частота попадания точки в единичный квадрат, равная отношению числа K пар (x2i-1, x2i), для которых проверочное условие выполнилось к общему числу N/2 пар последовательности должна сходиться к Р:



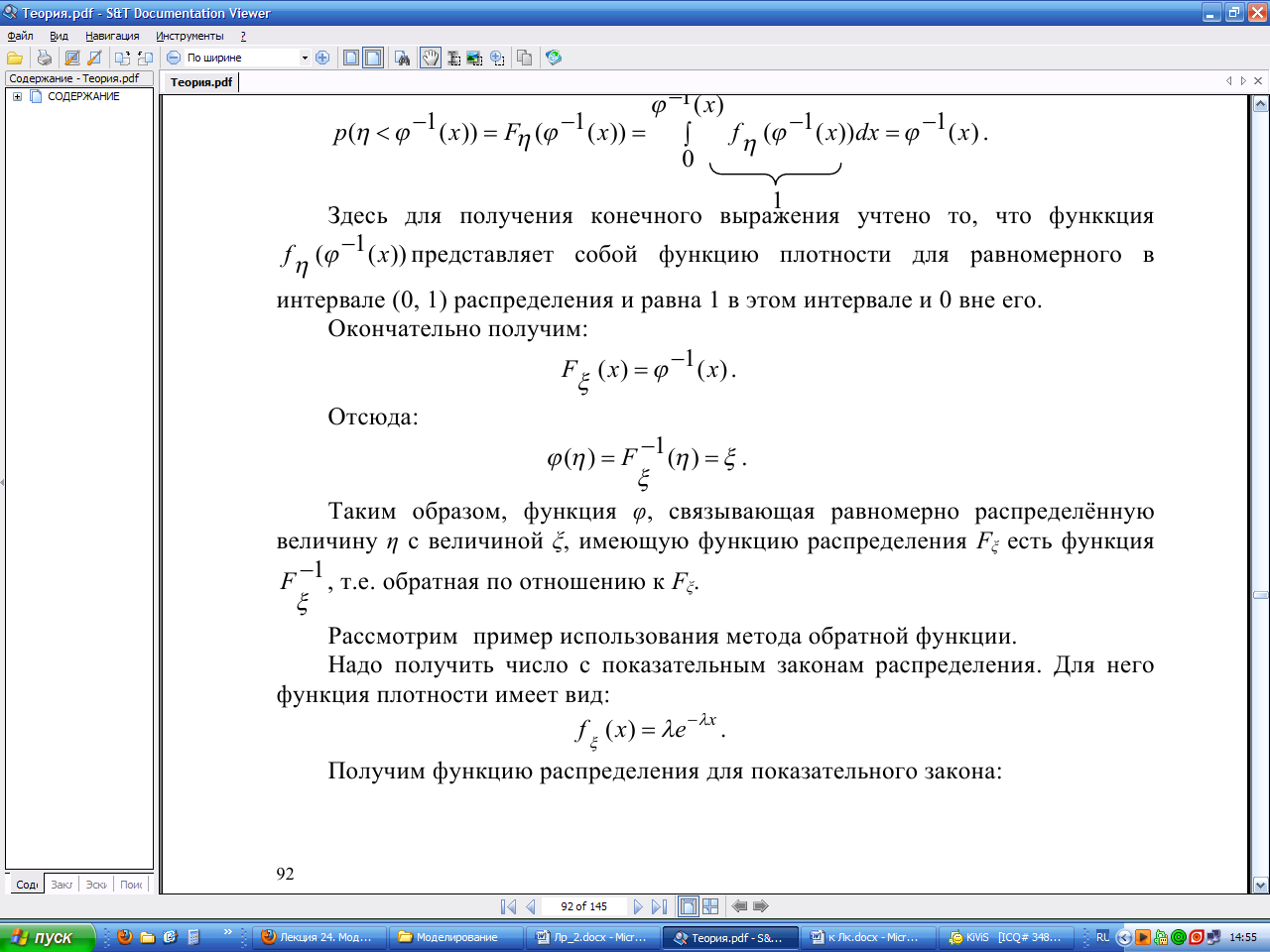
**II. Формирование случайных чисел с заданным распределением**

Основные процедуры формирования

Случайные числа с заданным распределением, как правило, формируются в результате преобразования последовательность РРСЧ, принимающих значения из диапазона от 0 до 1.

Методы, позволяющие имитировать непрерывные и дискретные вероятностные распределения:

– метод обратных функций,

**

– метод исключения,

– метод композиции

– и т.д.

Рассмотрим содержание двух наиболее распространенных на практике процедур.

***1. Для непрерывных распределений***

***Реализуется метод обратных функций***.

Имитации подлежит непрерывная случайная величина *x*, которая описывается плотностью распределения. Плотность распределения *f*(*x*) связана с функцией распределения *F*(*x*) соотношением



Требуется разработать машинный алгоритм. Для этого последовательно выполняются следующие операции.

1. Осуществляется переход от плотности распределения *f*(*x*) к функции распределения *F*(*x*) на основе соотношения:



2. Составляется исходное уравнение.

F (*x*) = *r*

*r* — число, генерируемое ГСЧрр в интервале от 0 до 1

3. Данное уравнение решается относительно *x*:

*x* = F–1(*r*) - искомый машинный алгоритм

*x* – генерируемая в итоге случайная величина с заданным распределением

F–1() - функция, обратная по отношению к функции F().

***2. Для дискретных распределений***

***Реализуется метод обратных функций.***

Имитации подлежит дискретная случайная величина , которая описывается рядом распределения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *xi* | *x1* | *x2* | *…* | *xn* |
| *pi* | *p1* | *p2* | *…* | *pn* |

где 

Для имитации значения дискретной случайной величины *x* используется СЧрр *r* на интервале [0,1].

Очевидно, что в этом случае

P(0 ≤ *r* < p1) = p1;

P(p1≤ *r* < p1 + p2 ) = p2;

P(p1+ p2 ≤ *r* < p1 + p2 + p3) = p3;

. . .

P(p1+ p2 + ...+ pn-1 ≤ *r* < p1+ p2 + ...+ pn ) = pn;

Машинный алгоритм, имитирующий значение дискретной случайной величины *x*:

1. Берется случайное СЧрр *r*.

2. Проверяется логическое условие:



где k принимает целочисленные значения, возрастающие от 1 до n.

3. При некотором k условие начинает выполняться. Это определяет имитируемое значение xk - дискретной случайной величины X.

**1. Имитация равномерного распределения**

Равномерное распределение непрерывной случайной величины  описывается плотностью распределения



Математическое ожидание и дисперсия случайной величины  определяется соотношениями

 и 

*Используем метод обратных функций* для имитации равномерного распределения:



 - искомый машинный алгоритм.

**2. Имитация распределения Симпсона (треугльное распределение)**

Распределение Симпсона непрерывной случайной величины  описывается плотностью распределения

Распределение Симпсона имеет случайная величина , которая представляет собой следующую сумму:

X = y+z ,

где  и  - независимые случайные величины, распределенные равномерно на интервале . Следовательно, распределение Симпмсона можно рассматривать как композицию двух одинаковых законов равномерного распределения.

Машинный алгоритм для имитации распределения Симпсона базируется на применении формулы X = y+z. Согласно этой формуле необходимо получить два случайных числа y и z, распределенных равномерно на интервале , и просуммировать их. Найденное таким образом число  будет иметь распределение Симпсона.

**3. Имитация экспоненциального распределения**

Экспоненциальное распределение непрерывной случайной величины *x* описывается плотностью распределения



где  - параметр экспоненциального распределения.

Математическое ожидание и дисперсия случайной величины  определяются соотношениями

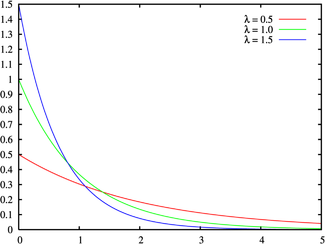
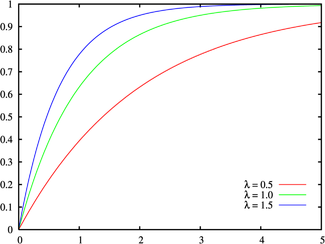
 и .

*Используем метод обратных функций* для имитации равномерного распределения:



 - **искомый машинный алгоритм**.

Плотность вероятности Функция распределения

[](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Exponential_distribution_pdf.p) [](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Exponential_distribution_cdf.p)

**Параметры** \lambda > 0 \,- интенсивность или обратный коэффициент масштаба

**4. Имитация гамма-распределения**

Гамма-распределение непрерывной случайной величины  описывается плотностью распределения:



где  и  - параметры гамма-распределения (η>0, λ>0)).

При η, принимающем целочисленные значения, гамма-распределение называется распределением Эрланга.

Гамма-распределение сводится к экспоненциальному распределению, если положить .

Математическое ожидание и дисперсия случайной величины *x* определяются соотношениями

 и .

Случайная величина *x* может быть представлена в виде суммы независимых случайных величин *x*i

.

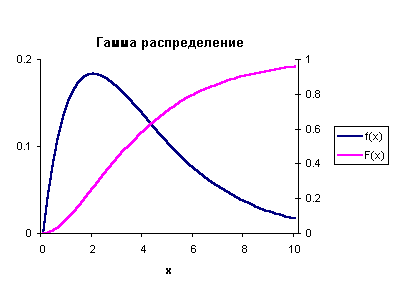
Если гамма-распределение случайной величины свести к экспоненциальному распределению, получим машинный алгоритм для имитации гамма-распределения.



или

 - **искомый машинный алгоритм,**

где  - СЧрр *r*.

**5. Имитация нормального (гауссовского) распределения**

Гауссовское распределение является одним из наиболее распространенных непрерывных распределений.

Гауссовское распределение непрерывной случайной величины *x* описывается плотностью распределения:



где  и  - соответственно математическое ожидание и среднее квадратическое отклонение гаусовского распределения.

Машинный алгоритм для имитации гауссовского распределения можно получить, *базируясь на центральной предельной теореме* (сумма независимых, случайных величин с произвольными распределениями имеет асимптотически гауссовское распределение). Сходимость к гауссовскому распределению осуществляется наиболее быстро, если суммируются величины с одинаковым распределением. В этом случае даже небольшое число слагаемых приводит к гауссовскому распределению.

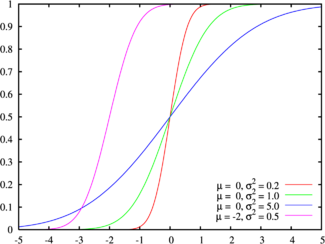
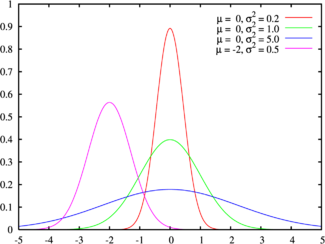
В основе машинного алгоритма для имитации гауссовского распределения лежит суммирование СЧрр *r*.

- искомый машинный алгоритм.

С возрастанием , т.е. числа суммируемых случайных СЧрр *r*, повышается точность имитации гауссовского распределения. Обычно  выбирают в пределах от 6 до 12. При этом достаточная для многих приложений точность обеспечивается при использовании всего шести СЧрр *r*. Для случая, когда  = 6:



Плотность вероятности Функция распределения

[](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Normal_distribution_cdf.p)[](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Normal_distribution_pdf.p)

Зеленая линия соответствует стандартному нормальному распределению

* - коэффициент масштаба (вещественный, строго положительный)

**6. Имитация треугольного распределения**

Может быть использован **метод исключения И.Неймана**.

Треугольное распределение непрерывной случайной величины *x* описывается плотностями распределения:

а) 

или

б) 

**Машинный алгоритм**

1. Формируются два СЧрр *r1* и *r2.*

2. Проверяется условие *r2 < r1*. Если условие выполняется, то находится искомое число

.

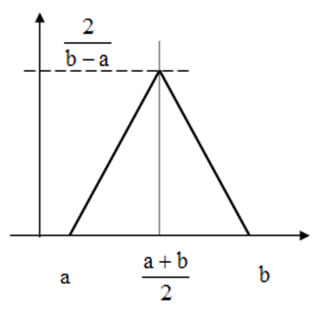
В противном случае пара чисел  отбрасывается и осуществляется переход к шагу 1.

* Приведенный алгоритм имеет существенный недостаток: часть пар чисел , приходится отбрасывать. Принимая во внимание независимость СЧрр *r1* и *r2*, можно предложить более экономичные алгоритмы, основанные на использовании следующих формул:

|  |  |
| --- | --- |
|  | **-** машинные алгоритмы  для имитации треугольного распределения |

где  - взятие максимального числа из совокупности двух СЧрр *r1* и *r2*;

 - взятие минимального числа на совокупности двух СЧрр *r1* и *r2*.

 **7. Построение гистограммы**

Построение гистограммы распределения состоит в последовательном выполнении следующих этапов.

1. Находится минимальное  и максимальное  значения массива реализаций.

2. Определяется размах варьирования



3. Определяется длина интервала



где  - число интервалов.

4. Определяются граничные значения для каждого -го интервала 

5. Фиксируется количество попаданий  в каждый -й интервал 

6. Вычисляются ординаты гистограммы распределения



где  - число выполненных испытаний (объем массива реализаций).

**Лабораторная работа №3**

**Исследование дискретно-стохастической СМО (из Лр №1).**

**Задание**

Исследовать работу дискретно-стохастической СМО:

1) разработать алгоритм работы дискретно-стохастической СМО

2) написать имитирующую программу, предусматривающую сбор и статистическую обработку данных для получения оценок заданных характеристик СМО).

**Пример выполнения**

для Варианта 7

1) Реализация цепи Маркова*.*

Рассмотрим СМО варианта 7.

События Р2000 – Р0211 представляют собой полную группу S несовместных событий. Реализация таких событий производится следующим образом:

1. Для каждой j-ой строки интервал [0,1] разбивается на S частей с длинами pj1, pj2,...,pjs. Точки деления интервала имеют координаты:

,

2. Генерируется случайная величина *ξ* – случайная величина, равномерно распределенная в интервале [0,1]. Если  то считают, что произошло событие Ak, то есть был осуществлен переход из j-го в k-е состояние.

3. Далее генерируется случайная величина в интервале [0,1] и процедура повторяется k-ой строки.

Состояние системы считывается через каждый такт.

Из состояния ‘2000’ в состояние ‘1000’ система по окончании такта переходит с вероятностью равной 1. Из состояния ‘1000’ в состояние ‘2010’ система также переходит с вероятностью равной 1.

Для дальнейшего определения состояния системы в последующие такты воспользуемся методом реализация цепи Маркова таблицей 2.б.

Таблица 2.б

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | **∑** |
| 2000 | 1000 | 2010 | 1010 | 2011 | 1011 | 1001 | 2111 | 1111 | 2211 | 1211 | 0211 |
| 1 | 2000 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 1000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 2010 | 0 | 0,25 | 0 | 0,75 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 1010 | 0 | 0 | 0,25 | 0 | 0,75 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 2011 | 0 | 0,1 | 0 | 0,3 | 0 | 0,45 | 0,15 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6 | 1011 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0 | 0,45 | 0 | 0 | 0 | 0 | 1 |
| 7 | 1001 | 0 | 0 | 0,4 | 0 | 0,6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8 | 2111 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0 | 0,45 | 0 | 0 | 0 | 1 |
| 9 | 1111 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0 | 0,45 | 0 | 0,45 | 0 | 0 | 1 |
| 10 | 2211 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0 | 0,45 | 0 | 0,45 | 0 | 1 |
| 11 | 1211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0,45 | 1 |
| 12 | 0211 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0,1 | 0 | 0,45 | 0 | 0,45 | 1 |

Для состояния ‘2010’ интервал [0,1] разобьем на 2 части с длинами 0,25 и 1,0 и единственной точкой деления 0,25. Пусть ξ = 0,37, ξ < 1,0, значит, система перешла из состояния ‘2010’ в состояние ‘1010’.

Теперь для 4-ой строки (состояние ‘1010’) интервал [0,1] разобьем на 2 части с длинами 0,25 и 1,0 и единственной точкой деления 0,25. Пусть ξ = 0,81, ξ < 1,0, значит, система перешла из состояния ‘1010’ в состояние ‘2011’.

Для 5-ой строки (состояние ‘2011’) интервал [0,1] разобьем на 4 части с длинами 0,1; 0,3; 0,45; 0,15 и точками деления 0,1; 0,4; 0,75. Пусть ξ = 0,63, ξ < 0,75, значит, система перешла из состояния ‘2011’ в состояние ‘1011’.

……

2) Определение искомых параметров

Параметры системы (*Lоч* и А) будем определять:

– средняя длина очереди : ,

где *i* – номер состояния,  *j* - число заявок в очереди в *i*-том наступившем состоянии.

– абсолютная пропускная способность (интенсивность потока обработанных заявок) – среднее число заявок, обслуживаемое СМО в единицу времени:

,

где *Ро* – вероятность того, что канал обрабатывал заявку(и) ,  – сумма вероятностей наступивших состояний, при которых не происходит обслуживание заявок; *Рз* – вероятность того, что обработка закончилась, Т – общее количество тактов.

***Для справки:***

А – абсолютная пропускная способность СМО – среднее число заявок, обслуживаемое СМО в единицу времени;

Q – вероятность обслуживания поступившей заявки Q или относительная пропускная способность СМО: Q = А/λ;

Pобсл – сумма вероятностей всех состояний, в которых происходит обслуживание заявки;

Ротк – сумма вероятностей всех состояний, в которых происходит отказ в обслуживании заявки;

Рбл– сумма вероятностей всех состояний, в которых происходит блокировка заявки;

Lоч – средняя длина очереди – сумма вероятностей всех состояний, в которых заявка находится в очереди, умноженных на количество заявок в данных состояниях.

**Лабораторная работа №4**

**Разработка, отладка и исследование программной модели непрерывно-стохастической СМО.**

**Задание**

Построить имитационную модель непрерывно-стохастической СМО и исследовать ее (разработать алгоритм и написать имитирующую программу, предусматривающую сбор и статистическую обработку данных для получения оценок заданных характеристик СМО)

**Варианты заданий**

***Вариант 1.***

Одноканальная СМО с неограниченной очередью.

Интенсивность источника λ =2.5, интенсивность обработки заявок каждого каналом µ =3



а) Исследовать значения средней длины очереди , среднего времени ожидания в очереди при показательном и равномерном (a = 0.05, b =0,75) распределении интервалов входного потока.

Поток обслуживания - простейший.

б) Исследовать значения средней длины очереди, среднего времени ожидания в очереди, среднего времени пребывания заявки в системе при следующих дисциплинах обслуживания заявок из очереди:

- FIFO;

- первыми обслуживаются заявки, требующие меньшего времени обслуживания.

Потоки - простейшие

***Вариант 2.***

СМО с отказами состоит из двух последовательных фаз.

Количество мест ожидания первой фазы n1, второй – n2. интенсивность обработки заявок каждого канала µ1 = µ2 = 5.



Варианты:

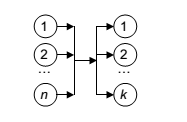
а) Построить зависимости Ротк, Ротк1, Ротк2 при изменении λ от 1 до 6 с шагом 0.5. Входной поток и потоки обслуживаний - простейшие, n1=2, n2=2.

б) Построить зависимости Ротк, Ротк1, Ротк2 при изменении n1 от 1 до 6. Входной поток и потоки обслуживаний - простейшие, n2 = 2, λ = 4.5

***Вариант 3.***

СМО с ожиданием ответа

Каждый источник может выдать заявку с интенсивностью λ, после чего ожидает окончания ее обработки одним из каналов.



Определить среднее число ожидающих ответа источников, среднее время ожидания ответа и абсолютную пропускную способность (интенсивность на выходе системы). Входные потоки и потоки обслуживаний – простейшие, интенсивность каждого источника λ =2.5, интенсивность обработки заявок каждого канала µ, n = 6.

Варианты:

а) k=3, µ=6

б) k =2, µ=9

в) k =1, µ=18

***Вариант 4.***

Одноканальная СМО с неограниченной очередью и дообслуживанием.



Если время обслуживания превышает значение Т, то заявка возвращается в очередь и затем проходит дообслуживание в течение оставшегося времени. Исследовать значения средней длины очереди и среднего времени ожидания в очереди. Входной поток и потоки обслуживаний – простейшие с интенсивностями λ =2.0, µ =2.5, Т=0.4.

***Вариант 5.***

СМО с отказами



На вход поступает простейший поток заявок с интенсивностью λ. Время обслуживания – показательное с параметром μ. Работающий (обслуживающий заявки канал) может выходить из строя (отказывать). Поток отказов – простейший с параметром n. Ремонт начинается мгновенно после отказа. Время ремонта – показательное с параметром g. Количество мест ожидания n = 3.

Найти: вероятности состояний канала (канал свободен, занят, ремонтируется) абсолютную и относительную пропускную способность системы.

λ= 0.5, μ = 1. n =0.01, g =0.1, n = 5.

Варианты:

а) Канал может выходить из строя только при обслуживании заявок;

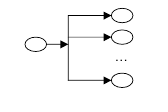
б) Канал может выходить из строя и в неработающем состоянии;

в) Заявка, находившаяся в канале в момент его выхода из строя теряется;

г) Заявка, находившаяся в канале в момент его выхода из строя становится в очередь, если в ней есть места и обслуживается заново.

***Вариант 6.***

Многоканальная СМО со “взаимопомощью.”



Если в свободную систему поступает заявка, то ее обслуживают совместно все каналы. Если во время обслуживания заявки поступает еще одна, то часть каналов переключается на ее обслуживание и т.д., пока все каналы не окажутся занятыми. Интенсивность совместного обслуживания заявки n каналами nμ. Каналы распределяются равномерно между заявками. На вход поступает простейший поток заявок с интенсивностью λ= 7. Время обслуживания – показательное с параметром μ=1, n = 8. Найти абсолютную и относительную пропускную способность системы.

***Вариант 7.***

На вход n-канальной СМО с отказами поступает поток заявок с интенсивностью λ = 6 заявок в час. Среднее время обслуживания одной заявки 0.8 часа. Каждая обслуженная заявка приносит доход 4у.е. Содержание одного канала обходится 2 у.е./час. Определить экономически целесообразное количество каналов.

***Вариант 8.***

На вход 2х-канальной СМО с отказами поступает поток заявок с нтенсивностью λ = 4 заявки в час. Среднее время обслуживания одной заявки 0.8 часа. Каждая обслуженная заявка приносит доход 4у.е. Содержание одного канала обходится 2 у.е./час. Определить,что экономически целесообразнее – увеличение числа каналов до 3, или введение мест ожидания, если содержание одного места обходится в 0.3 у.е./час.

***Вариант 9.***

Одноканальная СМО с неограниченной очередью и “разогревом.”

На вход СМО поступает простейший поток заявок с интенсивностью λ=1.5. Время обслуживания – показательное с параметром μ = 2. Перед обслуживанием заявки свободный до того канал должен “разогреться”. Определить средние времена пребывания заявок в системе и в очереди, среднее количество заявок в системе и в очереди. Если обслуживание начинается сразу после окончания обслуживания предыдущей заявки, то “разогрев” не нужен.

Варианты:

а) время “разогрева ” показательное со средним значением Тр = 0.4

б) время “разогрева ” фиксированное Тр = 0.4

г) время “разогрева ” Тр = 0.2 Ток, если Ток < 6, иначе Тр = 0.3, где Ток – время, прошедшее после окончания обслуживания последней заявки.

***Вариант 10.***

Одноканальная СМО с неограниченной очередью и повторным обслуживанием



Заявка, прошедшая обслуживание, может быть возвращена в очередь на повторное обслуживание с вероятностью р = 0.1. Исследовать значения средней длины очереди, среднего времени ожидания в очереди, среднего времени пребывания заявки в системе. Входной поток и потоки обслуживаний – простейшие с интенсивностями λ =2.0, μ =2.5 .

Варианты:

а) время повторного обслуживания показательное со средним значением Тобсл = 0.2;

б) заявка повторно обслуживается как впервые поступившая;

в) заявка может повторно обслуживаться неограниченное число раз; определить среднее число прохождений заявкой обслуживания;

г) заявка может повторно обслуживаться только 1 раз. В случае второй “выбраковки” она покидает систему необслуженной. Определить вероятность необслуживания заявки.

***Вариант 11.***

Система массового обслуживания представляет собой стоянку такси, на которую поступает поток пассажиров с интенсивностью λ и поток машин с интенсивностью μ . Пассажиры образуют очередь, которая уменьшается на 1, когда к стоянке подходит машина. В случае, когда на стоянке нет пассажиров, в очередь становятся машины. Число мест n для машин на стоянке ограничено.

Варианты:

а) Все потоки простейшие, λ =12.0, μ =15.0 (заявок в час), n = 10. Очередь пассажиров не ограничена, посадка производится мгновенно. Определить средние значения очередей пассажиров и машин и средние значения времени пребывания на стоянке пассажиров и машин.

б) Все потоки простейшие, λ =12.0, μ =12.0 (заявок в час), n = 10. Очередь пассажиров ограничена (l =20), посадка производится мгновенно. Определить средние значения очередей пассажиров и машин и средние значения времени пребывания на стоянке пассажиров и машин, вероятность для пассажира уехать на такси.

***Вариант 12.***

Система массового обслуживания представляет собой стоянку такси, на которую поступает поток пассажиров с интенсивностью λ и поток машин с интенсивностью μ . Пассажиры образуют очередь, которая уменьшается на 1, когда к стоянке подходит машина. В случае, когда на стоянке нет пассажиров, в очередь становятся машины. Число мест n для машин на стоянке ограничено.

Все потоки простейшие, λ =15.0, μ =10.0 (заявок в час), n = 10. Очередь пассажиров ограничена следующими факторами: пассажир покидает очередь, если через 20 минут после начала ожидания количество пассажиров перед ним больше L. Посадка производится мгновенно. Определить средние значения очередей пассажиров и машин и средние значения времени пребывания на стоянке пассажиров и машин, вероятность для пассажира уехать на такси для L = 5, L = 10, L = 15.

***Вариант 13.***

Система массового обслуживания представляет собой стоянку такси, на которую поступает поток пассажиров с интенсивностью λ и поток машин с интенсивностью μ . Пассажиры образуют очередь, которая уменьшается на 1, когда к стоянке подходит машина. В случае, когда на стоянке нет пассажиров, в очередь становятся машины. Число мест n для машин на стоянке ограничено.

Все потоки простейшие, λ =15.0, μ =10.0 (заявок в час), n = 10. Очередь пассажиров ограничена следующими факторами: пассажир покидает очередь, если через S минут после начала ожидания количество пассажиров перед ним больше 3 человек. Посадка производится в течении 2 минут. Определить средние значения очередей пассажиров и машин и средние значения времени пребывания на стоянке пассажиров и машин, вероятность для пассажира уехать на такси, если S равномерно распределено в диапазоне от 5 до 20.

***Вариант 14***.

Автозаправочная станция (АЗС) имеет n колонок; площадка возле нее допускает одновременное ожидание не более μ автомашин. Поток автомашин, пребывающих на АЗС простейший с интенсивностью λ. Время обслуживания показательное со средним значением tобсл.

Варианты:

а) λ=1 маш/мин, tобсл = 2 мин.

Найти вероятности отказа и среднее значение длины очереди при n =2 для значений μ от 3 до 7;

б) λ=1 маш/мин, tобсл = 3 мин, n =3. Определить экономически обоснованное число мест ожидания, если заправка одной машины приносит доход 5 у.е, а аренда одного места ожидания стоит 20 у.е./час.

***Вариант 15.***

На железнодорожную сортировочную горку подается простейший поток составов с интенсивностью λ= 2 состава в час. Время обслуживания состава на горке имеет показательное распределение со средним значением tобсл.=20 мин. В парке прибытия горки могут находиться не более трех составов (включая обслуживаемый). Остальные вынуждены ожидать на внешних путях. За один час ожидания на внешних путях станция платит штраф 1000 руб. Определить срок окупаемости увеличения количества мест в парке прибытия горки до 4, если это увеличение потребует затрат в размере 1 млн. рублей.

***Вариант 16.***

Одноканальная СМО – ЭВМ, на которую поступает поток заявок (требований на расчеты) со средним интервалом между заявками tвх Время обслуживания распределено по закону Эрланга 3-го порядка с математическим ожиданием tобсл.. Определить:

среднее число заявок в очереди;

среднее число заявок в СМО;

-среднее время пребывания заявок в очереди;

-среднее время пребывания заявок в СМО;

Варианты:

а) входной поток простейший, tвх=10 минут; tобсл=8 минут;

б) входной поток – поток Эрланга 4-го порядка, tвх=10 минут; tобсл=8 минут;

***Вариант 17.***

Имеется n-канальная СМО с неограниченной очередью. Входной поток и поток обслуживаний - простейшие с интенсивностями λ и μ соответственно.

Время пребывания в очереди ограничено случайным сроком, распределенным по показательному закону с математическим ожиданием tож.

Определить:

среднее число заявок в очереди;

-среднее время пребывания заявок в очереди (отдельно – для получивших обслуживание и ушедших из очереди до обслуживания);

среднее число занятых каналов.

Варианты:

а) n=2, λ = 3 заявки/час , μ=1 заявка/час , tож=0,5 часа.

б) n=3, λ = 4 заявки/час , время обслуживания заявки постоянно и равно 1 часу , tож=0,5 часа

***Вариант 18.***

Два наладчика обслуживают 6 станков. Станок требует наладки в среднем через каждые 0,5 часа. Наладка занимает у рабочего в среднем 10 минут. Все потоки событий – простейшие. Определить, как изменятся следующие показатели:

среднее число занятых рабочих;

абсолютная пропускная способность;

среднее число неисправных станков,

если рабочие будут налаживать станки совместно, затрачивая при этом на наладку одного станка в среднем 5 минут.

***Вариант 19.***

Одноканальная СМО с приоритетным обслуживанием и неограниченной очередью.

На вход СМО поступает простейший поток заявок с интенсивностью λ =10. Поток обслуживаний показательный с интенсивностью μ= 12. Вероятность появления во входном потоке заявки, требующей приоритетного обслуживания Рпр=0,1.

Определить среднюю длину очереди Lоч, среднее время ожидания в очереди для бесприоритетных Wоч и приоритетных Wоч.пр заявок.

Варианты:

а) заявка, требующая приоритетного обслуживания становится в начало очереди ;

б) заявка, требующая приоритетного обслуживания прерывает обслуживание заявки в канале (если обслуживаемая заявка бесприоритетна) и сразу попадает на обслуживание; если обслуживаемая заявка имеет приоритет, то новая заявка, требующая приоритетного обслуживания становится в начало очереди ; заявка, выполнение которой было прервано, ставится в начало очереди и после освобождения канала дообслуживается в течении оставшегося времени.

***Вариант 20.***

Многоканальная СМО (n=3) с приоритетным обслуживанием и неограниченной очередью.

На вход СМО поступает поток бесприоритетных заявок (приоритет 0) с интенсивностью λ0=7 и два потока заявок с приоритетами 1 (λ1=1,5) и 2 (λ2=1). Интенсивность обслуживания заявок каналами - μ1=2, μ2= 4, μ3=5.

Все потоки простейшие.

Заявки с приоритетом 2 должны обслуживаться только каналом 3, заявки с приоритетом 1 могут обслуживаться каналами 2 и 3, если 3-й канал не занят обслуживанием заявок с приоритетом 2, заявки с приоритетом 0 – каналом 1 и каналами 2 и 3, если они не заняты обслуживанием заявок с более высоким приоритетом.

Поступающие заявки занимают место в очереди в соответствии с их приоритетами. В случае, если заявка застает «свой» канал занятым заявкой с более низким приоритетом, то обслуживаемая заявка перемещается либо в соответствующий ей канал с прерыванием обслуживания в нем менее приоритетной заявки, либо в очередь как вновь поступившая.

Определить среднее время пребывания в системе tc, среднее время пребывания в очереди tоч для заявок с разными приоритетами, среднюю длину очереди Lоч.

***Вариант 21.***

Одноканальная СМО с неограниченной очередью. Входной поток простейший с интенсивностью 2 заяв/час. Найти среднюю длину очереди и среднее время ожидания в очереди, если поток обслуживания имеет

а) показательное распределение времени обслуживания, μ=2,5 заяв/час;

б) равномерное распределение времени обслуживания, а=0,2час, b=0,6 час;

в) нормальное распределение времени обслуживания, μ=0.4час, σ=0,1 час;

г) время обслуживания фиксировано, tобсл=0,4 час;

д) время обслуживания распределено по закону Эрланга



***Вариант 22.***

Многоканальная СМО с отказами Μ/M/n. Входной поток простейший с интенсивностью 2 заяв/час, n=3.

Найти:

абсолютную пропускную способность;

среднее число занятых каналов;

вероятность отказа, если поток обслуживания имеет:

а) показательное распределение времени обслуживания, μ=0,7 заяв/час;

б) равномерное распределение времени обслуживания, а=1,5час, b=2,0 час;

в) нормальное распределение времени обслуживания, μ=1,7час, σ=0,5 час;

г) время обслуживания фиксировано, tобсл=1,75 час;

д) время обслуживания распределено по закону Эрланга



***Вариант 23.***

На промышленном предприятии для контроля за качеством готовой продукции разрабатывается новая система, которая будет включать некоторое количество испытательных стендов и помещения для хранения поступающих на контроль изделий. Вследствие ограниченной площади помещения одновременно в очереди может ожидать не более чем m изделий. Если поступающее на контроль изделие застает ситуацию, что все места для ожидания заняты, то оно отгружается, не проходя контроль. Исследование моментов поступления изделий на контроль показали, что они случайны и распределены по закону Пуассона с параметром λ изд/ч. Время, затрачиваемое на контроль одного изделия, также случайное со средним значением μ изд/ч. Определить при заданных значениях m = 3 изд., λ = 2 изд/ч, μ=1 изд/ч минимальное число испытательных стендов, чтобы было проконтролировано не менее 95% всей выпускаемой продукции.

**Теоретические сведения**



Характеристики эффективности работы СМО

**А** – ***абсолютная пропускная способность*** СМО или среднее число заявок, обслуживаемое СМО в единицу времени;

**Q** – ***относительная пропускная способность*** СМО или вероятность обслуживания поступившей заявки:

Q=A/λ;

**Ротк** – ***вероятность отказа***, т.е вероятность того, что поступившая заявка не будет обслужена, получит отказ:

Ротк = 1 - Q;

**Построение и реализация алгоритмов моделирования Q-Схем.**

**Структура Q-Схемы.**

Для детального ознакомления с технологией машинной имитации рассмотрим Q-Схему достаточно общего вида. Q-Схема содержит три фазы обслуживания и источник заявок.

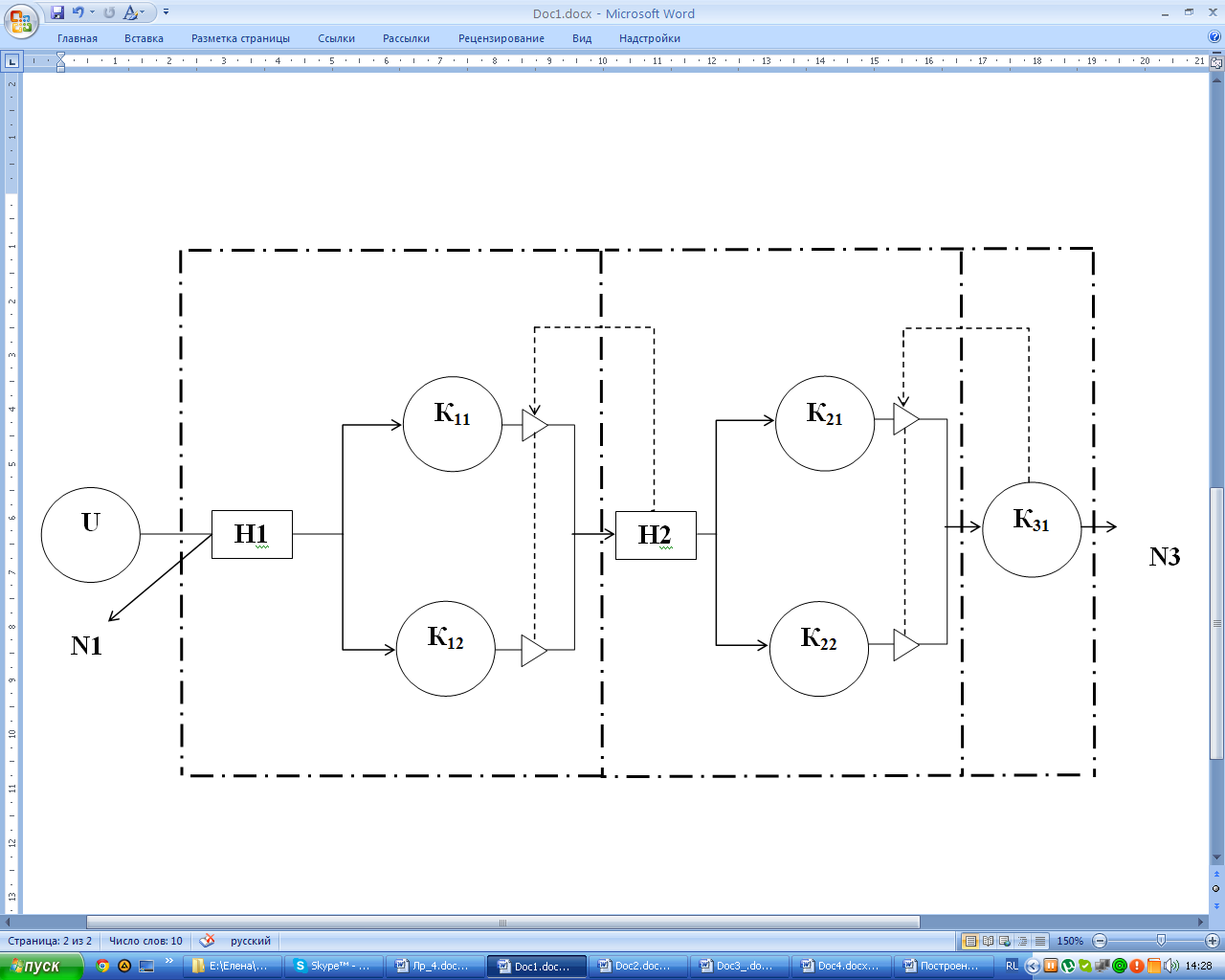


Рисунок 3.1 – Структура Q-схемы.

Первая фаза содержит 2 однотипных канала К11 и К12 и общий входной накопитель заявок Н1. В случае заполнения накопителя Н1 заявки источника получают отказ (дисциплина отказа заявкам на входе фазы I).

Вторая фаза также содержит два однотипных канала К21 и К22 и общий входной накопитель Н2. В случае заполнения накопителя Н2 заявки блокируются в первой фазе. Это означает, что если какой-либо канал К21 или К22 в некоторый момент модельного времени завершил обслуживание заявки и в этот момент каналы второй фазы заняты и накопитель заполнен, то обслуженная заявка не покидает систему, что имеет место в случае отказа, а блокируется в канале первой фазы. Заявка сохраняется каналом первой фазы до тех пор, пока в накопителе Н2 не освободится по крайней мере одна позиция.

Третья фаза содержит только один канал К31 и накопитель Н3 ёмкостью, равной нулю. При занятом канале К31 заявки блокируются во второй фазу.

Для описания имитационной модели Q-Схемы введём следующие переменные:

tn – текущее значение модельного времени;

tm – время появления очередной заявки на выходе источника;

tkj – время окончания обслуживания каналом j k-й фазы K очередной заявки;

Zkj (tn) – состояние канала j фазы k в момент t;

Li – ёмкость накопителя i-й фазы;

Zi – состояние накопителя i-й фазы;

Ni – количество потерянных заявок;

N3 - количество обслуженных системой заявок;

P – вероятность отказа (потери) заявки системой;

Δt – интервал продвижения модельного времени в сплошном моделировании.

Каждый из каналов Q-Схемы может находиться в следующих состояниях:

1. Канал свободен (0);
2. Канал занят обслуживанием (1);
3. Канал заблокирован (хранит уже обслуженную заявку) (2);

Текущее состояние Z накопителя Н равно количеству заявок, хранящемуся в накопителе в текущий момент модельного времени t.

Процедура моделирования начала обслуживания заявки каждым элементарным каналом Kij сводится к следующему.

Выполняется обращение к генератору случайных чисел. Генератор формирует интервал обслуживания заявки каналом Kij, закон распределения, длительности которого должен соответствовать закону F распределения времени обслуживания заявок каналом Kij. Вычисляется время окончания обслуживания tij=tn+tij, где tn – текущий момент модельного времени. Канал Kij переходит в состояние «занят обслуживанием».

Когда модельное время достигает значения tij, соответствующему моменту завершения обслуживания каналом Kij, моделируются процесс передачи заявки с выхода канала Kij в накопитель Hi+1 следующей фазы или в каналы фазы i+1, если ёмкость Li+1 накопителя Нi+1 равна нулю. Если фаза i+1 может принять заявку, то канал Kij переводится в состояние «свободен». В этом случае количество заявок в накопителе Hi+1 фазы i+1 увеличивается на I, а канал Kij может принять заявку из накопителя H1 своей фазы. Канал Kij переходит в состояние «занят обслуживанием», а количество заявок в накопителе H1 уменьшается на единицу. Если фаза i+1 заявку принять не может (накопитель и каналы заняты обслуживанием заявок), канал Kij переводится в состояние «заблокирован».

Укрупнённая схема алгоритма моделирования Q-Схемы, построенного по принципу последовательного просмотра состояний модели через фиксированный временной интервал Δt, представена на рисунке 3.2. Такой метод управления модельным временем называется сплошным моделированием и состоит в том, что после каждого просмотра состояния модели, модельное время tn увеличивается на интервал Δt. Наращивание модельного времени tn=tn+Δt выполняется блоком 10. Момент завершения моделирования Q-Схемы может быть зафиксирован: по числу просмотров N, по длине интервала моделирования T или по количеству обслуженных заявок N3. Проверка соответствующих условий выполняется блоком 3.

Работа вспомогательных блоков – ввода исходных данных 1, установки начальных условий 2, обработки II и вывода результатов моделирования I2 – не отличается по своей сути от аналогичных блоков, используемых в алгоритмах вычислений на ЭВМ. Поэтому остановимся более детально на работе той части моделирующего алгоритма, которая отражает специфику моделирования подхода (блоки 4-9). Детализованные схемы алгоритмов этих блоков приведены на рисунке. На этих и последующих схемах моделирующих алгоритмов Q-Схем приняты следующие обозначения: ZN(1) = z, Z(I,J)= zij, TM=tm, TN=tn, T(I,J)=tij, LO(1)=Li, PO=P.

Процедура формирования времени завершения обслуживания заявок каналами Kij оформлена в виде подпрограммы WORK(T(K,J)). Процедура генерирует tkj – длительность интервала обслуживания очередной заявки и формирует время завершения обслуживания t(k,j) = tn+tkj.

Окончание обслуживания заявки в некотором канале Kij в момент времени может вызвать процесс распространения изменений состояний элементов («особых состояний») системы в направлении противоположном движению заявок в системе, поэтому Н и К системы должны просматриваться при моделировании, начиная с обслуживающего канала последней фазы по направлению к накопителю I-й фазы.

Алгоритм формирования очередного состояния Q-Схемы в дискретный фиксированный момент модельного времени.

Рассмотрим реализацию основных блоков моделирующего алгоритма. Это блоки 9,8, … 4, которые имитируют формирование заявок источником и их обслуживание в каналах 1-й, 2-й и 3-й фаз модели.

Рассмотрим состояние модели Q-схемы на стационарном участке моделирования. Пусть после очередного выполнения блока 10 модельное время приняло значение tn.

Блок 4 имитирует завершение обслуживания заявок каналом K31 третьей фазы. Блок 4.1 проверяет состояние канала K31 и, если канал находится в состоянии «занят обслуживанием» («I»), то в блоке 4.2 проверяется время T31 завершения обслуживания каналом K31. Если это время меньше или совпадает с текущим модельным временем tn, то это означает, что в момент tn на выходе K31 появляется очередная заявка. В этом случае в блоке 4.3 увеличивается на I количество обслуженных заявок N3, а в блоке 4.4 канал K31 переводится в состояние «свободен» («0»).

Блок Б, имитирует завершение обслуживания заявок каналами 2-й фазы и передачу обслуженных заявок на 3-ю фазу. Блоки 5.1, 5.9 и 5.10 составляют цикл просмотра каналов 2-й фазы. Блоки 5.2 и 5.3 проверить состояние и время завершения обслуживания заявки каждым из каналов. Если для некоторого канала j его состояние z2j=0, т.е. он находится в состоянии «занят обслуживанием» или «заблокирован», T2j<=Tn, то это означает, что канал K2j хранит ранее заблокированную заявку (Z2j=2 и T2j<Tn) или именно в момент Tn он завершил обслуживание (T2j=Tn, Z2j=1). В этих случаях блок 5.4 проверяет состояние канала 3.1 3-й фазы. Если этот канал не свободен (Z31<>0), то блок 5.5 переводит канал kij в состояние «заблокирован» (или подтверждает ранее установленное состояние «заблокирован»). Если Z31=0, то блок 5.6 формирует новое время завершения заявки каналом K31, блок 5.7 переводит канал K31 в состояние «занят обслуживанием», а блок 5.8 освобождает канал K2j.

Блок 6 (рис 3.5) имитирует процесс передачи заявок из накопителя Н2 второй фазы в каналы K21, К22. Блоки 6.1, 6.7, 6.8 составляют цикл просмотра состояний каналов второй фазы. Блок 6.2 проверяет состояние накопителя Н2. Если накопитель Н2 содержит хотя бы одну заявку (ZN(2)<>0), выполняется переход к блоку 6.3, который проверяет состояние очередного канала 2-й фазы. Если j-й канал свободен (Z(2,j)=0), то в блоке 6.4 вычисляется время завершения обслуживания заявки каналом К2j, блок 6.5 переводит канал K2j в состояние «занят обслуживанием», а блок 6.6 уменьшает на единицу количество заявок в накопителе Н2. Если при выполнении блока 6.2 оказывается, что накопитель Н2 заявок не содержит (ZN(2)=0), то выполняется переход к блоку 7.

Блок 7 (рис 3.6) воспроизводит процесс передачи заявок из каналов 1-й фазы в накопитель и каналы 2-й фазы. Блоки 7.1, 7.15, 7.16 составляют цикл просмотра состояния каналов 1-й фазы. Если при выполнении блоков 7.2, 7.3 оказывается что некоторый канал К1j хранит заявку в состоянии «заблокирован» или выработал заявку в момент tn, выполняется переход к блокам 7.4, 7.5, 7.6, 7.7, составляющим цикл просмотра состояния каналов 2-й фазы. Если в результате выполнения в цикле блока 7.5 находится некоторый канал К2i в состоянии «свободен» (Я(2,i)=0), то выполняются блоки 7.8, 7.9, 7.10. Эти блоки формируют время завершения обслуживания заявки каналом К2i, канал К2i переводится в состояние «занят обслуживанием», а канал К1j переводится в состояние «свободен».

Если в результате просмотра каналов 2-й фазы все каналы оказываются занятыми, в блоке 7.11 проверяется состояние накопителя Н2. Если накопитель содержит свободные позиции (ZN(2)<L(2)), выполняются блоки 7.12, 7.14, увеличивающие на 1 количество заявок в накопителе Н2 и переводящие канал К1j в состояние «свободен». Если накопитель Н2 полностью заполнен, выполняется блок 7.18, переводящий канал Кij в состояние «заблокирован». Детальный алгоритм блока 8 приведён на рис. 2.7. Блок имитирует процесс передачи заявок из накопителя Н 1-й фазы в каналы 1-й фазы. Структура алгоритма полностью аналогична блоку 6.

Блок 9 (рис 3.8) воспроизводит поступление заявок из источника U на вход 1-й фазы. Если при выполнении 9.1 удовлетворяется Tm<=Tn, то это означает, что в момент времени tn на выходе источников сформирована очередная заявка. Блоки 9.2, 9.6, 9.7 составляют цикл просмотра состояний каналов 1-й фазы. Если в результате просмотра блок 9.3 обнаружит свободный канал К1j, выполняются блоки 9.4, 9.5. Они формируют время завершения обслуживания заявки T1j каналом Kij и переводят канал Kij в состояние «занят обслуживанием».

Если свободных каналов в 1-й фазе нет, то анализируется состояние накопителя Н (9.7). Если накопитель содержит свободную позицию (ZN(1)<L(1)), блок 9.9 увеличивает на 1 количество заявок в накопителе. Если накопитель заполнен, блок 9.10 увеличивает на 1 количество заявок получивших отказ. Во всех случаях в блоке 9.11 вычисляется момент времени t поступления очередной заявки источника на вход системы.

1

2

3

11

4

12

5

6

7

8

9

10

Вход

Ввод исходных данных

Установка начальных условий

Обработка результатов

Выход

Переход к следующему моменту времени

Вывод результатов

Обслуживание заявки  
 каналом 3-й фазы

Переход заявки   
из 2-й фазы в 3-ю

Обслуживание заявки   
каналом 2-й фазы

Переход заявки   
из 1-й фазы в накопитель 2-й фазы

Обслуживание заявки   
каналом 1-й фазы

Поступление заявки на вход   
Q-схемы

Проверка окончания моделирования

да

Рисунок 3.2

4

Z(3,I)=1

T(3,1)=TN

N3=N3+1

Z(3,1)=0

5

4.1

нет

да

да

нет

4.2

4.3

4.4

Рис.3.3. Алгоритм блока 4

Z(2,J)0

5

5.1

J = 1

T(2,J)TN

5.2

Z(3,1) = 0

WORK(T(3,1))

Z(3,1)=1

Z(2,J)=0

5.3

5.4

J2

Z(2,J)=2

5.6

5.7

5.8

J=J+1

да

да

да

нет

нет

нет

5.9

5.5

6

5.10

Рис.3.4. Алгоритм блока 5

Z(2)>0

6

6.1

J = 1

6.2

Z(2,J) = 0

WORK(K(2,J))

Z(2,J)=1

ZN(2)=ZN(2)-1

6.3

6.4

J2

6.5

6.6

J=J+1

да

нет

нет

нет

6.7

7

6.8

Рис.3.5. Алгоритм блока 6

7.6

нет

7

J=1

Z(1,J)≠0

T(1,J)≤TN

I=1

Z(2,I)≠0

I≥2

ZN(2)<L(2)

ZN(2)=ZN(2)+1

Z(1,J)=0

Z(1,J)=0

J≥2

8

J=J+1

I=I+1

Z(1,J)=2

7.1

7.2

7.3

7.4

7.5

нет

нет

нет

7.8

7.9

7.10

7.7

7.11

да

да

да

7.12

7.14

нет

7.16

7.15

нет

7.13

да

WORK(K(2,I))

Z(2,I)=1

Рис.3.6. Алгоритм блока 7

Z(1,J)=0

WORK(K(1,J))

Z(1,J)=1

J=J+1

9

ZN(1)=ZN(1)-1

8.1

8

8.2

8.3

8.4

8.5

8.6

8.8

8.7

да

нет

нет

да

нет

ZN(1)≠0

J=1

J≥2

Рис.3.7. Алгоритм блока 8

10

9

TM TN

J=1

Z(1,J) = 0

Z(1,J)=1

J2

ZN(1)L(1)

ZN(1)=ZN(1)+1

D(TM)

J=J+1

N1=N1+1

9.11

WORK(K(1,J))

9.1

9.2

9.3

9.4

9.5

9.6

9.8

9.7

9.10

9.9

да

нет

да

нет

нет

нет

да

Рис.3.8. Алгоритм блока 9

**Пример выполнения**

для Варианта 7

1) Исходные данные

Р-схема

2

π

π

2

На вход n-канальной СМО с отказами поступает поток заявок с интенсивностью λ = 6 заявок в час. Среднее время обслуживания одной заявки 0,8 часа. Каждая обслуженная заявка приносит доход 4у.е. Содержание одного канала обходится 2 у.е./час. Определить экономически целесообразное количество каналов.

2) Анализ задания

Система содержит n-каналов и источник с отказами.

Данная система является многоканальной СМО с отказами вида M/M/n (так называемая, задача Эрланга).

Схема системы представлена на рисунке 1.



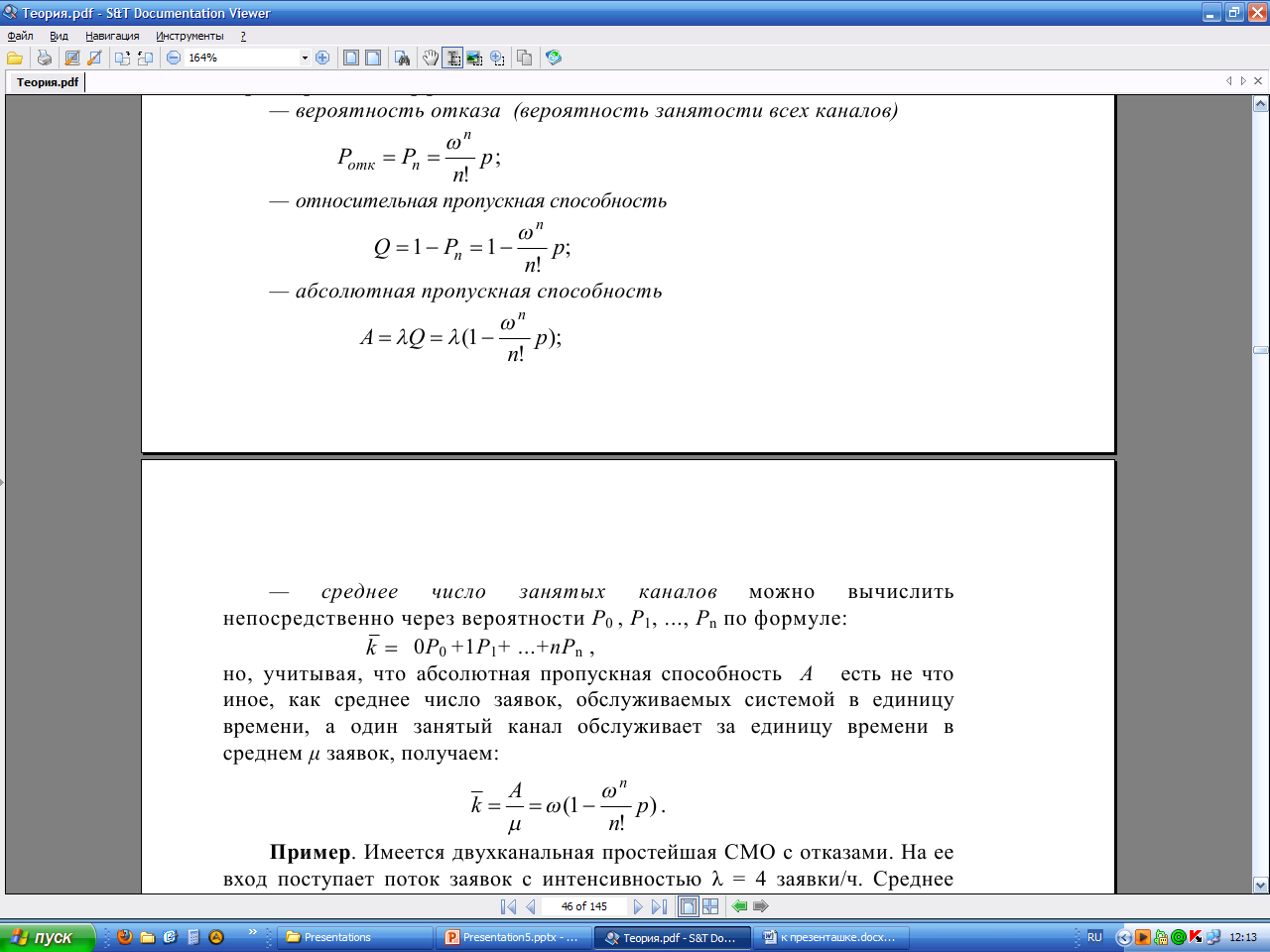
Рисунок 1 – Q-схема

3) Построим диаграмму интенсивностей переходов (ДИП), показанную на рисунке 2.



Рисунок 2 – ДИП Q-схемы

4) Для систем массового обслуживания вида M/M/n



Исходя из того, что сумма вероятностей всех состояний системы равна 1 (нормировочное уравнение), получим:откуда.

Тогда  *n* – число каналов

5) Прибыль от работы системы за один час составит 4*А*, или  у.е.

Затраты на функционирование системы будут равны 2*n* у.е.

Таким образом, чтобы определить экономически целесообразное количество каналов, необходимо найти максимум функции 

Учитывая заданные значения  и , формула примет вид .

6) Построив имитационную модель непрерывно-стохастической СМО и исследовав ее при соответствующих исходных данных получим:

а) наиболее эффективное количество каналов – 6;

б) прибыль, полученная при этом – 850-900 у.е..