Расчетно-графическая работа

по курсу:

«Компьютерное моделирование»

тема: «Моделирование Petri Nets»

Выполнил студент группы ИВ-73

НТУУ «КПИ» ФИВТ

Ашаев Юрий

**Киев-2009**

Оглавление

[Введение 3](#_Toc251224897)

[Теоретические сведения 3](#_Toc251224898)

[Этапы моделирования: 4](#_Toc251224899)

[Сети Петри 5](#_Toc251224900)

[Виды сетей Петри 6](#_Toc251224901)

[Разработка технического задания. 7](#_Toc251224902)

[1.1. Выбор варианта 11](#_Toc251224903)

[2. Разработка программного продукта моделирования сетей Петри 13](#_Toc251224904)

[2.1.Инструкция пользователя 15](#_Toc251224905)

[3.Дерево достижимости сетей Петри 17](#_Toc251224906)

[3.1. Теоретические сведения 17](#_Toc251224907)

[3.2. Реализация режима построения дерева достижимости ССП 18](#_Toc251224908)

[3.3. Алгоритм работы программы построения дерева достижимости сети Петри 19](#_Toc251224909)

[3.4. Инструкция пользователя. Построение дерева достижимости 21](#_Toc251224910)

[4. Марковский граф. 22](#_Toc251224911)

[4.1 Инструкция пользователя. Построение Марковского графа 24](#_Toc251224912)

[5. Исследование моделей информационных потоков 26](#_Toc251224913)

[Теоретические сведения 26](#_Toc251224914)

[5.1 Линейный конгруэнтный генератор: 29](#_Toc251224915)

[5.2 Аддитивный генератор псевдослучайных чисел. 32](#_Toc251224916)

[5.3 Равномерное распределение 37](#_Toc251224917)

[5.4 Распределение Пуассона. 41](#_Toc251224918)

[5.5 Распределение Эрланга 46](#_Toc251224919)

[5.6 Распределение гиперэкспоненциальное 50](#_Toc251224920)

[6.Моделирование процессов стохастических сетей Петри 55](#_Toc251224921)

[6.1. Моделирование процессов в Petri-nets моделях. 56](#_Toc251224922)

[6.2. Инструкция пользователя. Моделирование сетей Петри 61](#_Toc251224923)

[7.Моделирование log-сервера и сервера приложений 62](#_Toc251224924)

[7.1 Статистические результаты эмуляции работы log-сервера 70](#_Toc251224925)

[8. Общая структура подсистемы анализа безопасности ИАС на этапах проектирования 73](#_Toc251224926)

[Вывод 81](#_Toc251224927)

# Введение

**Модели́рование** — исследование объектов познания на их моделях; построение и изучение моделей реально существующих предметов, процессов или явлений с целью получения объяснений этих явлений, а также для предсказания явлений, интересующих исследователя.

Моделирование используется для двух основных целей:

1. расширение наших знаний об окружающем мире;

2. разработка эффективных производственных процессов.

**Модель** - объект произвольной природы, который отражает главные, с точки зрения решаемой задачи, свойства объекта моделирования.

**Моделирование** - создание, применение, использование модели.

**Главные функции модели** - упрощение получения информации о свойствах объекта; передача информации и знаний; управление и оптимизация объектами и процессами; прогнозирование; диагностика.

В силу многозначности понятия «модель» в науке и технике не существует единой классификации видов моделирования: классификацию можно проводить по характеру моделей, по характеру моделируемых объектов, по сферам приложения моделирования (в технике, физических науках, кибернетике и т. д.). Например, можно выделить следующие **виды моделирования**:

* Информационное моделирование
* Компьютерное моделирование
* Математическое моделирование
* Математико-картографическое моделирование
* Молекулярное моделирование
* Цифровое моделирование
* Логическое моделирование
* Педагогическое моделирование
* Психологическое моделирование
* Статистическое моделирование
* Структурное моделирование
* Физическое моделирование
* Экономико-математическое моделирование
* Имитационное моделирование
* Эволюционное моделирование
* Историческое моделирование
* Нечеткое моделирование
* Модельное моделирование

## Теоретические сведения

**Процесс моделирования** включает три элемента:

1. субъект (исследователь);
2. объект исследования;
3. модель, определяющую (отражающую) отношения познающего субъекта и познаваемого объекта.

### Этапы моделирования:

**Первый этап** построения модели предполагает наличие некоторых знаний об объекте-оригинале. Познавательные возможности модели обусловливаются тем, что модель отображает (воспроизводит, имитирует) какие-либо существенные черты объекта-оригинала. Вопрос о необходимой и достаточной мере сходства оригинала и модели требует конкретного анализа. Очевидно, модель утрачивает свой смысл как в случае тождества с оригиналом (тогда она перестает быть моделью), так и в случае чрезмерного во всех существенных отношениях отличия от оригинала. Таким образом, изучение одних сторон моделируемого объекта осуществляется ценой отказа от исследования других сторон. Поэтому любая модель замещает оригинал лишь в строго ограниченном смысле. Из этого следует, что для одного объекта может быть построено несколько «специализированных» моделей, концентрирующих внимание на определенных сторонах исследуемого объекта или же характеризующих объект с разной степенью детализации.

**На втором этапе** модель выступает как самостоятельный объект исследования. Одной из форм такого исследования является проведение «модельных» экспериментов, при которых сознательно изменяются условия функционирования модели и систематизируются данные о ее «поведении». Конечным результатом этого этапа является множество (совокупность) знаний о модели.

**На третьем этапе** осуществляется перенос знаний с модели на оригинал — формирование множества знаний. Одновременно происходит переход с «языка» модели на «язык» оригинала. Процесс переноса знаний проводится по определенным правилам. Знания о модели должны быть скорректированы с учетом тех свойств объекта-оригинала, которые не нашли отражения или были изменены при построении модели.

**Четвертый этап** — практическая проверка получаемых с помощью моделей знаний и их использование для построения обобщающей теории объекта, его преобразования или управления им.

Моделирование— циклический процесс. Это означает, что за первым четырехэтапным циклом может последовать второй, третий и т.д. При этом знания об исследуемом объекте расширяются и уточняются, а исходная модель постепенно совершенствуется. Недостатки, обнаруженные после первого цикла моделирования, обусловленные малым знанием объекта или ошибками в построении модели, можно исправить в последующих циклах.

Сейчас трудно указать область человеческой деятельности, где не применялось бы моделирование. Разработаны, например, модели производства автомобилей, выращивания пшеницы, функционирования отдельных органов человека, жизнедеятельности Азовского моря, последствий атомной войны. В перспективе для каждой системы могут быть созданы свои модели, перед реализацией каждого технического или организационного проекта должно проводиться моделирование.

### Сети Петри

Для исследования в качестве исходной модели используется модель Petri-nets системы.

**Сети Петри** - формализованный язык описания взаимодействующих процессов в дискретных технических системах. С их помощью можно описывать функционирование как элементов и узлов ЭВМ, так и организацию вычислительных процессов в машинах, комплексах и системах.

Сеть Петри представляет собой двудольный ориентированный граф, состоящий из вершин двух типов — позиций и переходов, соединённых между собой дугами, вершины одного типа не могут быть соединены непосредственно. В позициях могут размещаться метки (маркеры), способные перемещаться по сети.

**Событием** называют срабатывание перехода, при котором метки из входных позиций этого перехода перемещаются в выходные позиции. События происходят мгновенно, либо разновременно, при выполнении некоторых условий.

**Позиция** (место) - это состояние, в котором находится система или определенная ее часть.

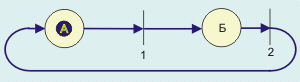


Рисунок 1 Сеть Петри с двумя позициями и двумя переходами. Цифрами 1 и 2 обозначены переходы, а буквами А и Б - позиции.

Состояние системы формируется в результате реализации локальных операций, называемых **условиями** реализации событий. Условие имеет емкость:

* Условие не выполнено - емкость равна нулю
* Условие выполнено - емкость равна 1.
* Условие выполнено с n-кратным запасом - емкость равна n.

Определенная комбинация условий может стимулировать определенное событие, которое вызовет в свою очередь изменение условий. В сетях Петри события и условия отображаются абстрактными символами, называемыми переходами (вертикальными или горизонтальными полосками - "барьерами") и позициями (кружками). Условия-позиции и события-переходы связаны отношениями зависимости, которые отображаются с помощью ориентированных дуг. Позиции, из которых исходят дуги данного перехода, называются входными позициями. Позиции же, к которым ведут дуги данного перехода, называются выходными позициями. Выполнение условий отображается помещением соответствующего числа меток в соответствующую позицию. Если число меток велико (более 2-3), емкость условия может быть отображена числом.

Формально работа сети Петри описывается множеством последовательностей срабатываний и множеством реализуемых разметок позиций.

Для сетей Петри существует удобная алгебраическая нотация. Каждому переходу ставится в соответствие правило грамматики. Каждое правило специфицирует входную и выходную позиции. Текущее состояние сети Петри характеризуется неупорядоченным набором позиций. Каждая позиция присутствует в этом наборе столько раз, сколько меток она имеет.

Сети Петри (СП)- это математические структуры, которые задаются объединением множеств:

СП=<P,T>,

где P - множество позиций , T - множество переходов.

Для стохастических временных сетей Петри множество переходов разбивают на 2 подмножества:

T= Tm+Tv(мгновенные и временные)

Для временных задаются λ и g интенсивность срабатывания и ковариация.

Потоком с такими характеристиками будет сгенерировано **t** задержка между запуском и срабатыванием события.

Активными компонентами в сетях являются переходы. Есть множество версий сетей Петри. Могут вводиться новые типы переходов.

Сеть Петри может быть задана графом или матрицей.

Сеть Петри характеризуется двумя матрицами:

1. Di – матрица входов-переходов
2. Dq – матрица выходов переходов.

Эти матрицы имеют вид:

где I(tj) = αj1, αj2,…, αjn

где Q(tj) = αj1, αj2, ..., αjn

### Виды сетей Петри

1. Временная сеть Петри — переходы обладают весом, определяющим продолжительность срабатывания (задержку).
2. Стохастическая сеть Петри — задержки являются случайными величинами.
3. Функциональная сеть Петри — задержки определяются как функции некоторых аргументов, например, количества меток в каких-либо позициях, состояния некоторых переходов.
4. Цветная сеть Петри — метки могут быть различных типов, обозначаемых цветами, тип метки может быть использован как аргумент в функциональных сетях.
5. Ингибиторная сеть Петри — возможны ингибиторные дуги, запрещающие срабатывания перехода, если во входной позиции, связанной с переходом ингибиторной дугой, находится метка.
6. Иерархическая сеть — содержит не мгновенные переходы, в которые вложены другие, возможно, также иерархические, сети. Срабатывание такого перехода характеризует выполнение полного жизненного цикла вложенной сети.
7. WF-сети

**Основными свойствами сети Петри являются:**

1. ограниченность — число меток в любой позиции сети не может превысить некоторого значения K;
2. безопасность — частный случай ограниченности, K=1;
3. сохраняемость — постоянство загрузки ресурсов, постоянна. Где Ni — число маркеров в i-той позиции, Ai — весовой коэффициент;
4. достижимость — возможность перехода сети из одного заданного состояния (характеризуемого распределением меток) в другое;
5. живость — возможность срабатывания любого перехода при функционировании моделируемого объекта.

В основе исследования перечисленных свойств лежит анализ достижимости.

**Маркировка** или состояние сетей Петри задаются путём размещения фишек в позиции сети. При чём, фишки могут иметь разный смысл:

1. Файл, который требует обслуживание
2. Флажок свободного или занятого ресурса.

Маркировка сетей задаётся вектором маркировки на всех позициях сети.

mi - число фишек в позиции сети:

M(p) = mi(pi)

M0 - начальная маркировка.

Сети Петри как математическая структура обладает некоторыми свойствами, которые могут найти важную интерпретацию в моделях исследуемых прикладных систем. С помощью сетей Петри решается задача анализа развития во времени процессов с целью выявления конфликтных и тупиковых ситуаций, а также проверки механизмов исключения подобных ситуаций.

Анализ свойств в нашей модели будет выполняться в режиме построения дерева достижимости.

Моделирование процессов в ССП осуществлено разработкой имитационной модели для исследования асинхронных процессов. В качестве языка программирования был выбран язык Java.

# Разработка технического задания.

Для исследования в качестве исходной используется обобщенная Petri-nets модель системы, которую необходимо сократить в соответствии с заданной структурой и многократно выполняемыми этапами заданий и используемыми ресурсами (по вариантам из табл.1.1 и табл. 1.2). Варианты структурной S-системы и соответствующей Petri-nets модели выбираются по таблице 1.1: система ИЭ – измерительного эксперимента, система ПЭ – планируемого эксперимента, система АО – аналитической обработки, система ПР – принятия решений, система ПК – проведения конференций. В задании начальной маркировки сети табл. 1.1:

* U – общее количество используемых модулей ввода-вывода информации в связях с объектом;
* R – количество разделов оперативной памяти для реализации многозадачного режима обработки информации центральным процессором;
* V – количество терминалов пользователей, которые участвуют в принятии решений;
* с – количество файлов-заданий, обслуживаемых процессором ввода-вывода в начальный момент времени – T0 моделирования;
* d – количество файлов-заданий, обслуживаемых диспетчером памяти в начальный момент времени – T0 моделирования;
* e - количество файлов-заданий, обслуживаемых процессором терминалов в начальный момент времени – T0 моделирования;
* L – количество (входных и выходных дуг первого перехода) файлов-заданий накапливаемых в буфере вывода для последующей обработке выводными устройствами.

Для формирования случайных значений интервалов между событиями запуска-срабатывания временных переходах Petri-nets модели используются генераторы событий информационных потоков с тем или иным распределением в зависимости от коэффициента вариации g, которые выбираются по вариантам в таблице 1.2 на основе заданных интенсивностей λ1λ2λ3λ4λ5 информационных потоков обслуживания запросов в соответствующих приборах:

* λ1 – процессором ввода-вывода (с коэффициентом вариации в потоке g = 1/5);
* λ2 – процессором диска (с коэффициентом вариации в потоке g = 1);
* λ3 – диспетчером памяти (с коэффициентом вариации в потоке g = 1/3);
* λ4 – центральным процессором (с коэффициентом вариации в потоке g = 8);
* λ5 – процессором терминалов (с коэффициентом вариации в потоке g = 1).

Для завершения моделирования должно быть определено общее время моделирования Tmod , которое необходимо задавать произвольно перед проведением модельного эксперимента, как и времена промежуточных отчетов по результатам моделирования.

Общая структура управляющей компьютерной системы, связанной с объектом и группой терминалов операторов представлены на рис. 1.1.

Центральный процессор

Процессор терминалов

Терминал 1 оператора

Терминал 2 оператора

Терминал M оператора

Модуль 1 ввода-вывода

Модуль 1 ввода-вывода

Модуль 1 ввода-вывода

Буфер ввода

Буфер ввода

Процессор ввода

Процессор ввода

Диски

Процессор ввода

1

2

…

3

Диспетчер оперативной памяти

**Рис. 1.1.** Структура исследуемой системы поддержки принятия решений группой операторов – лиц принимающих решения (ЛПР)

Прикладные программы (файлы) пользователей занимают те или иные ресурсы системы в соответствии с многократно циклически выполняемыми этапами заданий, определяемых по варианту (табл. А.3). Среди вариантов структур, включающих отдельные ресурсы системы для выполнения типовых заданий, выделяются следующие:

**Структура ИЭ – измерительного эксперимента**, включает: средство ввода-вывода, связанные с объектом на основе процессора ввода-вывода (ПВВ) и его буферной памяти: буфера вывода (Бвыв) и буфера ввода (БВв), а также процессора диска (ПД), на который записываются данные измерений (Бвыв-ПВВ-БВв-ПД).

**Структура ПЭ – планируемого эксперимента**, в котором дополнительно используется диспетчер памяти (ДП) и соответствующие разделы оперативной памяти (Бвыв-ПВВ-БВв-ПД-ДП).

**Структура АО (аналитической обработки)**, в которой активно используется процессор диска, диспетчер памяти и выделенные разделы оперативной памяти, а также центральный процессор (ЦП), т.е. структура определяемая как (ПД-ДП-ЦП).

**Структура ПР (принятия решений)**, включающая все средства аналитической обработки и процессор терминалов (ПТ) для связи с операторами (ПД-ДП-ЦП-ПТ).

**Структура ПК (проведения конференций)**, включающая процессор терминалов, центральный процессор и диспетчер памяти, и обеспечивающая интерактивный режим взаимодействия операторов (ДП-ЦП-ПТ).

**Неиспользуемые ресурсы исключаются** из структуры системы в модели. Начальное размещение файлов на устройствах выбирается по варианту.

Одним из этапов формирования имитационной модели технической системы является создание формальной структуры системы, например, как системы массового обслуживания (СМО). Принимая очереди заданий простейшими, составим модель рассматриваемой системы как системы массового обслуживания следующего вида (рис. 1.1).

**Рис. 1.1.** Формальная структура исходной системы в виде системы массового обслуживания потоков заданий ЛПР и их файлов-программ

Буфер ввода

Процессор ввода-вывода

Буфер ввода

Процессор диска

Очередь FIFO

Очередь FIFO

Диспетчер оперативной памяти

Центральный процессор

Процессор терминалов

Очередь FIFO

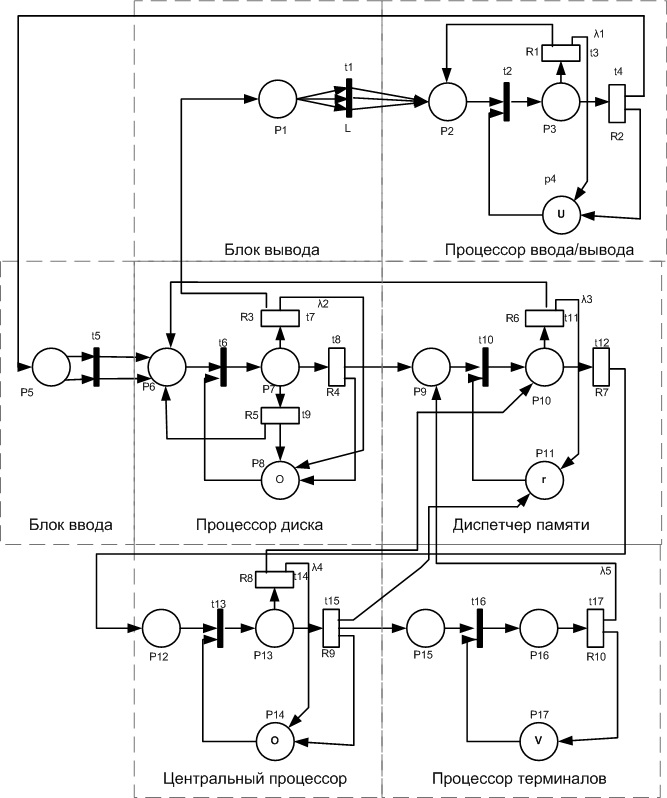
Очередь FIFO

Очередь FIFO

Наиболее эффективным способом исследования такой не простой модели СМО является имитационное моделирование случайного процесса, который порождается в много приборной СМО с обратными связями (сети с очередями) и различными характеристиками потоков входных и обслуженных заданий при различной длительности обслуживания в каждом приборе.

Среди средств описания имитационных моделей формальных дискретных систем выделяется мат. аппарат временных стохастических сетей Петри (ССП), которые относятся к семантическим сетям и отличаются высокой информативностью и простотой выполнения моделей.

На рис. 1.2 представлена модель формальной системы массового обслуживания на языке временных стохастических сетей Петри.



**Рис. 1.2.** Обобщенная структура Petri-nets модели исследуемой системы как системы массового обслуживания

### Выбор варианта

Параметры структуры имитационной модели системы управления

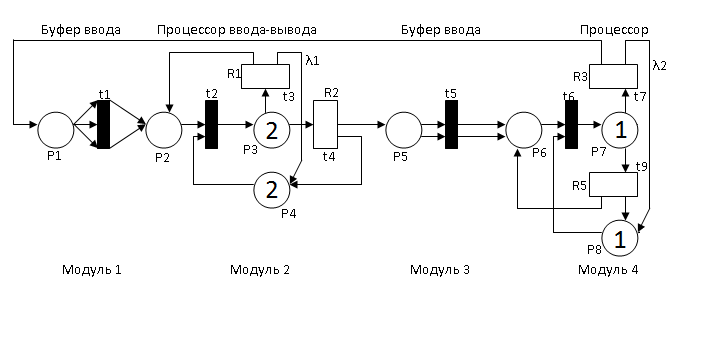
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Вариант по двум | Вариант | Внешних | Разделов | Терминалов | Размещение файлов, обьём буфера вывода | | | |
| последним цифрам зачётки | структуры(S) | устройств(U) | памяти(R) | операторов(V) | с | d | e | L |
| 01 | ИЭ | 4 | "-" | "-" | 2 | "-" | "-" | 2 |

Параметры потоков обслуживания задач управления обьектом

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вариант по первой букве отчества | Процессора ввода вывода λ1(1/с) | Процессора диска λ2(1/с) | Диспетчера памяти λ3(1/мс) | Центрального процессора λ4(1/мс) | Процессора терминалов  λ5(1/с) |
| Н | 0.05 | 60 | 5 | 80 | 1 |

В соответствии с вариантом:

Структура ИЭ – измерительного эксперимента, включает: средство ввода-вывода, связанные с объектом на основе процессора ввода-вывода (ПВВ) и его буферной памяти: буфера вывода (Бвыв) и буфера ввода (БВв), а также процессора диска (ПД), на который записываются данные измерений (Бвыв-ПВВ-БВв-ПД).



Маркировка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 |
| M | 0 | 0 | 2 | 2 | 0 | 0 | 1 | 1 |

|  |  |  |
| --- | --- | --- |
| **Таблица 3** | Назначения фишек в маркировке позиций сети Петри | |
| Позиция сети | Назначение фишек в маркировке сети | Начальное значение |
| **P1** | Файлы-задания в буфере вывода информции на объект | 0 |
| **P2** | Файлы-задания в очереди на обслуживание процессоров ввода-вывода (ПВВ) и соответствующими устройствами | 0 |
| **P3** | Устройства ввода-вывода обслуживают файлы-задания | с |
| **P4** | Устройства ввода-вывода свободны | U-c |
| **P5** | Файлы в буфере ввода информации | 0 |
| **P6** | Файлы в очереди на облуживание процессором диска (ПД) | 0 |
| **P7** | Процессор диска обслуживает файлы-задания | 0 (1) |
| **P8** | Процессор диска свободен | 1 (0) |

**Таблица А.2.** Интерпретации событий и времена задержек между событиями: запуск-срабатывание переходов

|  |  |  |
| --- | --- | --- |
| **Переходы сети** | Содержание событий запуск-срабатывание перехода | **Времена задержек** |
| **T1** | запуск-срабатывани – L файлов поступает из буфера вывода в очередь на обслуживание ПВВ | 0 |
| **T2** | запуск-срабатывание – файл поступает из очереди на обслуживание ПВВ и устройствами ввода-вывода | 0 |
| **T3 R1 = 0.3** | запуск-начало обслуживания файла устройствами вывода срабатывание – конец обслуживания файла уст-м вывода | Случайное  1/ λ1=1/0.05=20 |
| **T4 R2 = 0.7** | запуск-начало обслуживания файла устройствами ввода срабатывания – конец обслуживания файла уст-м ввода | Случайное  1/ λ1=1/0.05=20 |
| **T5** | запуск-срабатывание –S фалов поступает из буфера ввода в очередь для обслуживания процессором диска ПД | 0 |
| **T6** | запуск-срабатывание – файл поступает из очереди на обслуживание процессором диска | 0 |
| **T7 R3 = 0.3** | запуск-начало обслуживания файла процессором диска срабатывание – конец перезаписи диска в буфер вывода | Случайное  1/ λ2=1/60=0,0166 |
| **T9**  **R5=0.1** | Запуск-начало обслуживания файла процессором диска срабатывание – конец перезаписи с диска на диск | Случайное  1/ λ2=1/60=0,0166 |

Матрица входов-переходов.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DI | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 |
| t1 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| t2 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| t3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| t4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| t5 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| t6 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| t7 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| t9 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

Матрица выходов-переходов.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dq | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 |
| t1 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| t2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| t3 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| t4 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| t5 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| t6 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| t7 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| t9 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

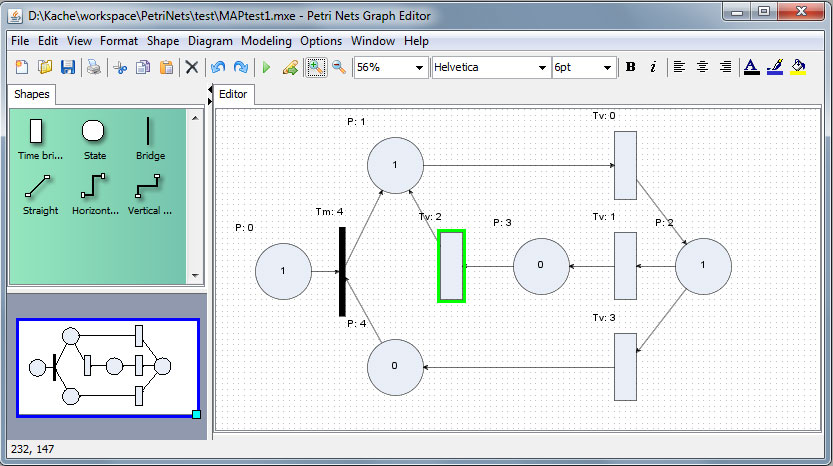
# 2. Разработка программного продукта моделирования сетей Петри

На данном этапе проектирования было разработано программное обеспечение, позволяющие моделировать стохастические сети Петри. Особенностью данного продукта является то, что исходными данными является сама модель стохастической сети Петри, а не матрицы входов и выходов.

**Краткий алгоритм работы с программой:**

В начале пользователь может добавить на рабочую область элементы –«состояния» и элементы «переходы». После этого надо установить связи между элементами. Третьим шагом является задание начальной маркировки стохастической сети Петри. После этого надо задать параметры переходов и описать конфликтные ситуации, если таковые имеются. После этого программа готова к моделированию.

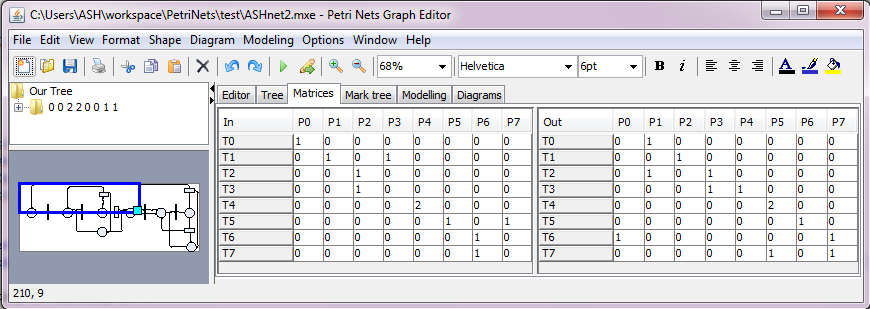
Внешний вид программы при набранной стохастической сети Петри рис. 2.1:



**Рис. 2.1.** Внешний вид программы**Таблицы описания**

Таблицами описания являются матрицы входов-переходов Di, матрицы выходов-переходов Dq, вектора начальной маркировки и вектор интенсивности переходов (λ)

Матрицы достижимости можно посмотреть во вкладке Descriptive Tables

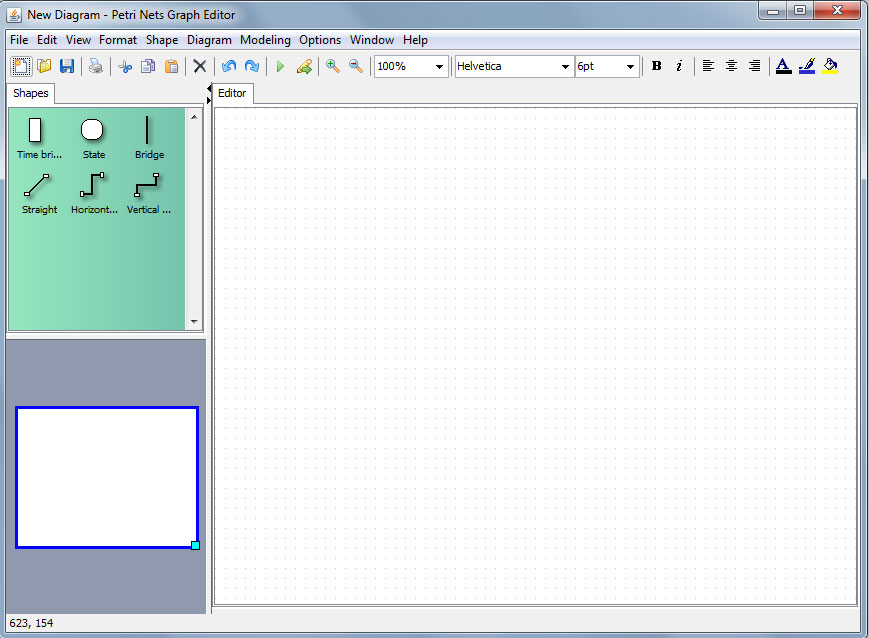


**Рис. 2.2** Матрицы входов переходов и выходов переходов.

На данном этапе проектирования было разработано программное обеспечение, позволяющие моделировать стохастические сети Петри. Особенностью данного продукта является то, что исходными данными является сама модель стохастической сети Петри, а не матрицы входов и выходов.

Разработанное программное обеспечение является свободно распространяемым, на условиях лицензии GNU Public License. Любой желающий может ознакомится с исходными кодами проекта по адресу http://code.assembla.com/trueblo/subversion/nodes

2.1.Инструкция пользователя**:**



**Рис. 2.3** Интерфейс программы

1. **Запуск**
   1. Программа запускается через запуск файла petrinets.jar из каталога Petri-nets\dist.
   2. После запуска открывается пустая рабочая область, в которую можно начать добавлять элементы.
2. **Создание новой модели сети**
   1. Создать новый файл можно такими способами:

* нажать на иконку New Diagram new
* выбрать File - New
* нажать Ctrl + N
  1. Добавить позицию можно перетянув элемент State ellipse из панели форм Shapes на рабочую область

Позиции нумеруются автоматически. По умолчанию количество фишек в позиции равно нулю, поэтому чтобы указать количество фишек нужно дважды нажать на цифру внутри позиции и отредактировать содержимое.

* 1. Добавить переход можно перетянув элемент Bridge vline из панели форм Shapes на рабочую область

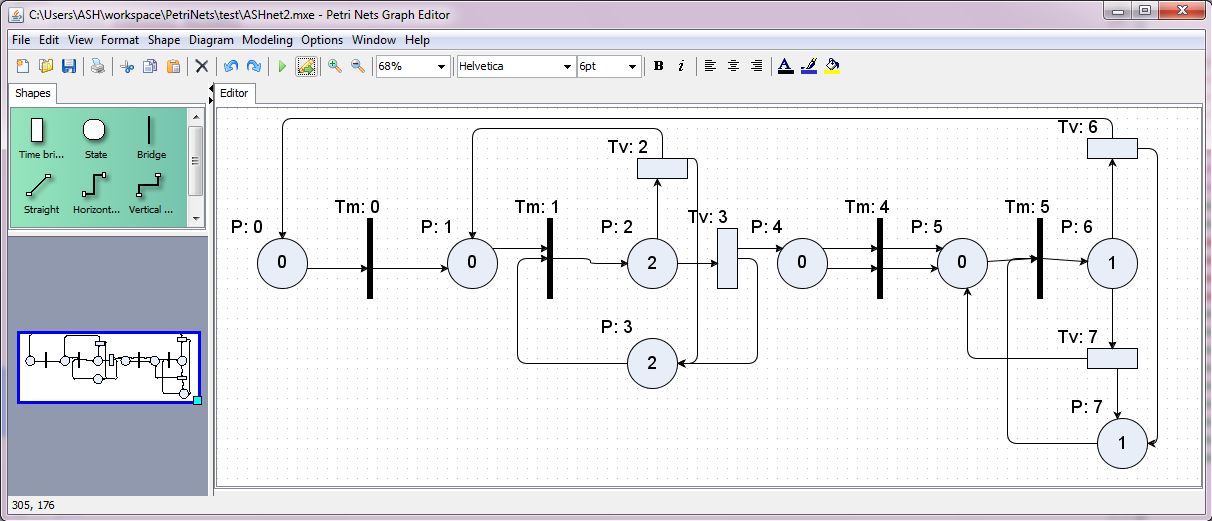
Переходы нумеруются автоматически.

* 1. Добавить временной переход можно перетянув элемент Time Bridge rectangle из панели форм Shapes на рабочую область

После чего нажать левой клавишей мыши на желаемое место в рабочей области. Переходы нумеруются автоматически. Для задания величин интенсивности (λ), вариации (g) и вероятности решения конфликта (r), следует нажать правой клавишей мыши на переходе (меню Modelling > Set Parameters for bridge) и последовательно задать данные

* 1. Добавить линию связи (дугу) можно перетянув один из элементов (в зависимости от того какого вида нужна дуга) Straight straight, Horizontal , Vertical vertical из панели форм Shapes на рабочую область

При помощи drag n drop менять форму (продлять и т.д.) дуги, а также связывать переходы с позициями и т.д.



**Рис.2.4** Программа с набранной моделью сети Петри.

1. **Работа с файлами**
   1. Открыть уже существующий файл можно такими способами:

* нажать на кнопку open
* выбрать File – Open File
* нажать Ctrl + O
  1. Сохранить уже существующий файл можно такими способами:
* нажать на кнопку Save save в меню
* выбрать File – Save as...
* нажать Ctrl + S
  1. Выйти из программы можно такими способами:
* Стандартным выходом из приложения в среде Windows
* выбрать File – Exit...
  1. Установить параметры страницы можно через File -> Page Setup
  2. Вывести на печать можно через File -> Print

1. **Редактирование**
   1. Отменить предыдущее действие можно через Edit – Undo undo
   2. Повторить отмененное действие можно через Edit – Redo redo

# 3.Дерево достижимости сетей Петри

## Теоретические сведения

**Свойства стохастических сетей Петри.**

1. Безопасность
2. Ограниченность
3. Абсолютная (строгая) сохраняемость
4. Сохраняемость по отношению к вектору взвешивания
5. Активность /пассивность/ фрагмента сети или перехода
6. Достижимость той или иной маркировки сети
7. Покрываемость той или иной маркировки сети

Позиция сети Петри называется **безопасной**, если маркировка сети , т.е. mi≤1, т.е.  Сеть Петри называется безопасной, если все ее позиции безопасны (в соответствующей позиции сети не может появиться более одной фишки).

Позиция сети Петри Pi называется **N-ограниченной**, если mi ≤ N любой маркировки (N>1). Сеть Петри называется N-ограниченной, если все позиции сети N-ограничены.

Сеть Петри с начальной маркировкой M0 называется **строгосохраняемой**, если соответствующей M’ сети справедливо:



Сеть Петри называется сохраняемой по отношению к некоторому вектору взвешивания W =w1,w2,...,wn , заданному на множестве позиций P сети, где Wi =1,2,..., если соответственно M’ сети справедливо:



таким образом, wi - весовой коэффициент фишки в i-й позиции.

В сетях Петри с M0 может оказаться такая ситуация, в которой переход tj будет неразрешенным соответственно последовательности запусков. В этом случае, переход tj называется **пассивным**, а соответствующий фрагмент сети называется пассивным фрагментом. С понятием пассивности связано понятие **тупика** сети Петри.

Если в сети Петри с M0 существует некоторая последовательность запуска G, приводящая к M’ , когда все переходы сети оказываются пассивными, то в такой сети Петри могут иметь место тупики, а называется M’ **тупиковым состоянием**. Для исследования сетей Петри необходимо определить, что некоторая M’ достижима из M0.

M’ называется **достижимой** в сети Петри с начальной M0, если существует некоторая последовательность запусков G, превращающая сеть Петри в данную M’. Говорят, что M’ принадлежит дереву достижимости сети Петри с начальной M0При исследовании сетей Петри важно выделять специальное свойство: наличие **конфликта** в сети. Этим свойством могут обладать 2 или более перехода сети.

Если в некоторой маркировке M’ сети Петри существует 2 или более разрешенных перехода, срабатывание одного из которых приводит к снятию условий разрешения для других переходов, то такие переходы называются **конфликтующими**, а сеть Петри - **конфликтной**.

## Реализация режима построения дерева достижимости ССП

**Дерево достижимости ССП** - это граф, вершинами которого являются реальные состояния (маркировки) сети, которые могут быть достигнуты из каждого очередного реального состояния последовательными и независимыми запусками всех разрешенных переходов в соответствующей граничной маркировке *Мх,* начиная с начальной *М0.* Запуски-срабатывания мгновенных переходов, если они разрешены, отмечаются, как единое событие вместе с запуском-срабатыванием предыдущего временного перехода.

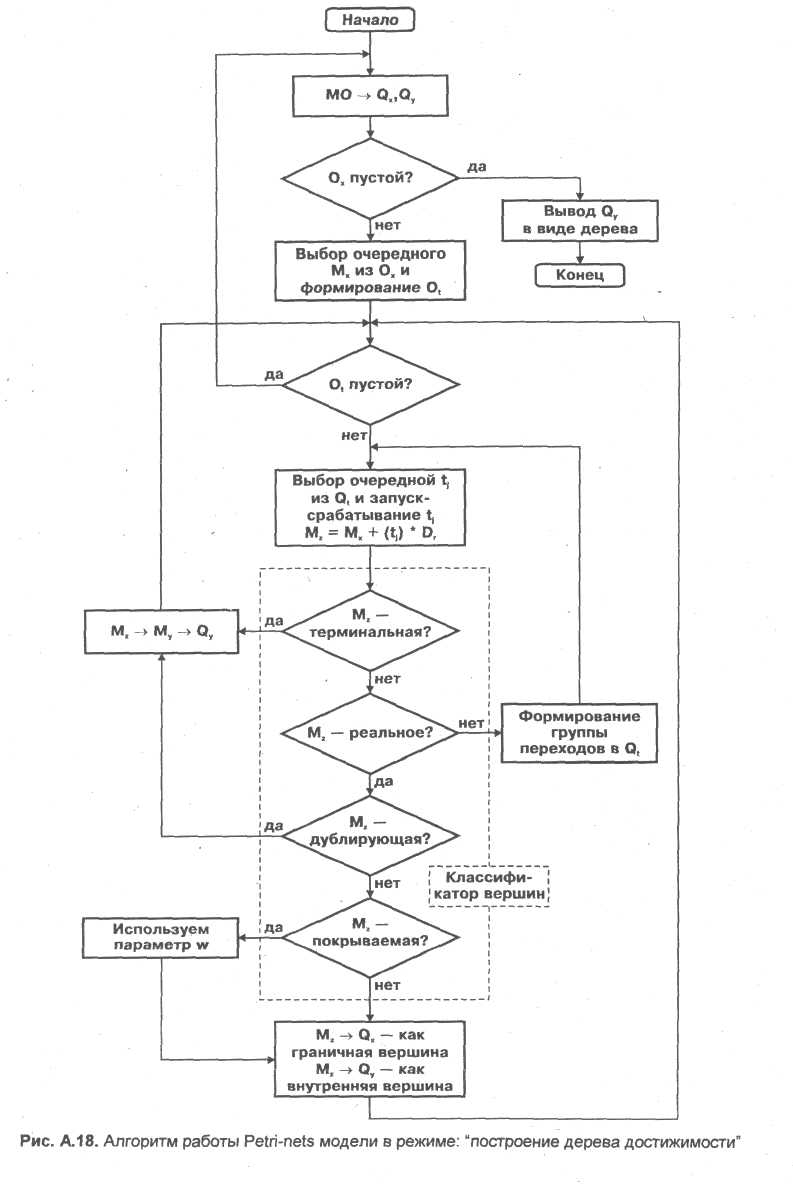
В целом ряде моделей взаимодействующих процессов дерево достижимости оказывается бесконечным. Анализ процедуры построение дерева позволяет выявить несколько факторов, позволяющих сократить (свернуть) дерево достижимости сети Петри.

Во-первых: появляются вершины, в которых нет разрешенных переходов. Это так называемые пассивные маркировки *МТ,* а соответствующие вершины называются **терминальными** или **конечными**.

Во-вторых: некоторые формируемые вершины соответствуют маркировкам *Mz,* полностью совпадающим (эквивалентными) уже имеющимся маркировкам *Му* внутренних вершин, т.е. *Мz = Мy.* Такиевершины *Mz* называются **дублирующими** по отношению к внутренним вершинам.

В-третьих: получаемые новые вершины могут иметь маркировки *Mz,* обладающие свойством покрываемости по отношению к внутренним вершинам *Му,* которое определяется как *Mz > Мy .* Это соответствует случаю неограниченного "размножения" фишек в одной или в нескольких позицияхсети. Тогда вводится специальный параметр *w* (например, *99)* как элемент вектора маркировки для тех позиций *Pi, для* которых *mz(pi) > my(pi).* Тогда *mz(pi) = w.* При последующих запусках-срабатываниях к w не прибавляются фишки и не вычитаются, полагая, что *w* равняется *бесконечности.*

## Алгоритм работы программы построения дерева достижимости сети Петри



Результат работы процедуры по построению дерева достижимости представлен на рисунке.

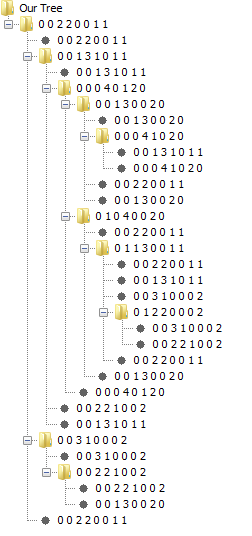


Таблица 5 Дерево достижимости

Алгоритм построения дерева достижимости ССП включает в себя классификатор вершин, *кото*рый переводит все исследуемые вершины *Mz* в четыре вида:

* Граничные — *Мх,* список *Qx,*
* Внутренние — *Му,* список *Qy,*
* Дублирующие — *Mz* = *Мy* список *Qy.*
* Терминальные — *MT*

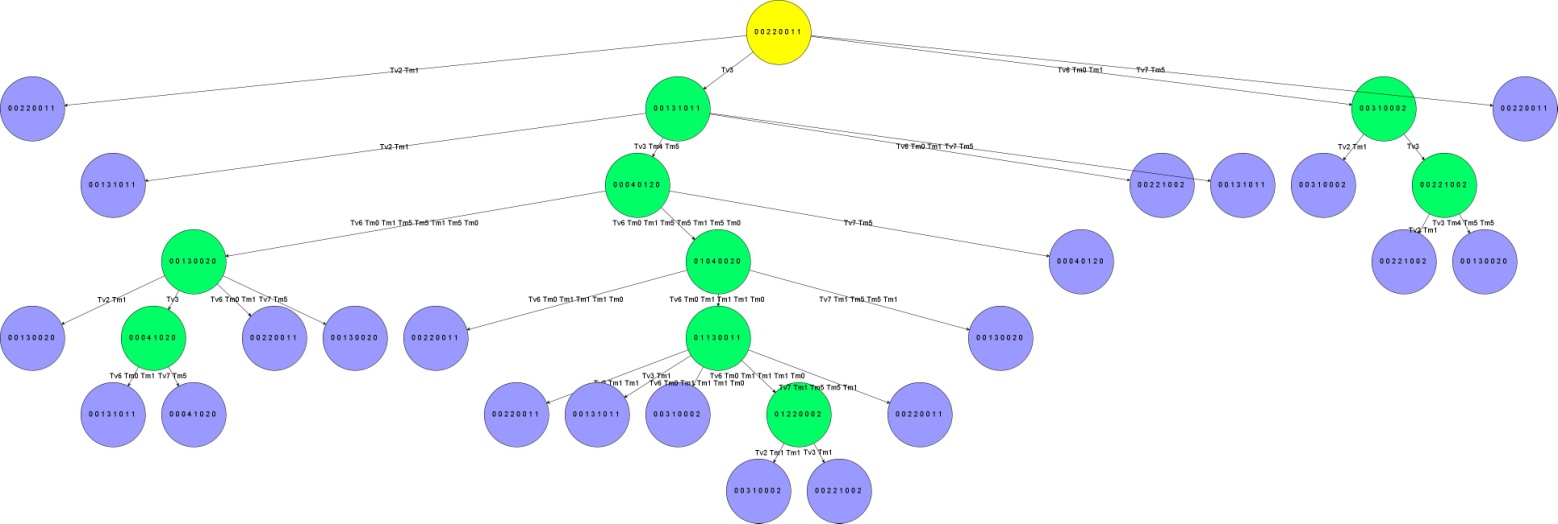
Построение дерева достижимости завершается формированием марковского графа состояний сети Петри, среди которых отмечаются только реальные состояния.

Алгоритм работы Petri-nets модели в режиме построения дерева достижимости представлен на рис. А.18.

Все разрешенные переходы или группы переходов в исследуемой граничной вершине дерева Mx записываются в специальный список *Qt,* который очищается по мере построения всех возможных ветвей *из Мх.*

Результат работы процедуры построения дерева достижимости представлен в таблице А. 19.

Граф достижимости на рис. 3.1



**Рис. 3.1** Граф достижимости.

В данном примере свойства сети Петри:

* сеть небезопасна, т.к. уже в начальной маркировке в позициях сети бывает больше чем одна фишка.
* сеть является ограниченной – максимальное количество фишек в одной вершине - три.
* сеть не строго сохраняема, т.к. не всегда выполняется равенство

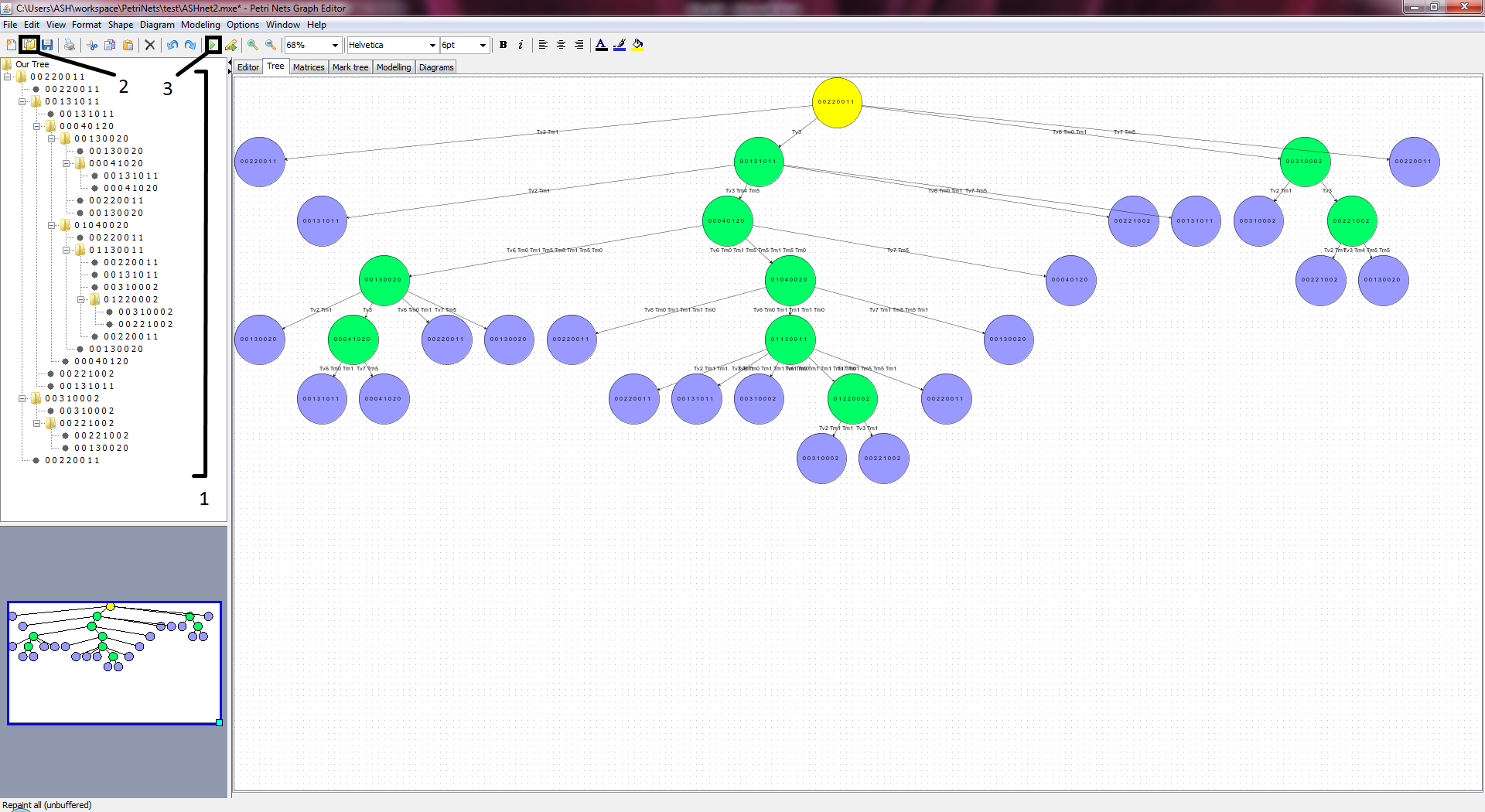


* сеть активна, так как в ней нет тупиковых состояний
* сеть достижима для всех M'
* сеть конфликтная, так как имеют место конфликты временных переходов.
* количество разных достижимых переходов в сети равно 10

## Инструкция пользователя. Построение дерева достижимости

Для того что бы получить дерево достижимости заданной сети Петри, нужно:

1. набрать или открыть (через диалог открытия файлов, в который можно попасть или из меню File -> Open или через иконку на панели инструментов).
2. выставить начальную маркировку (двойной клик на вершине сети инициирует изменение количества фишек в вершине)
3. перейти в режим моделирования (или через меню Modeling -> Modeling Mode. Так же для перехода в режим моделирования есть соответствующая кнопка на панели инструментов)



На иллюстрации показан скриншот программы в которой построено дерево достижимости.

Обозначения:

1. Дерево достижимости в текстовой форме
2. Кнопка диалога открытия диаграммы
3. Кнопка перехода в режим моделирования

В основном поле программы мы видим дерево достижимости в виде графа достижимости, представленном в ярусно-параллельной форме.

# 4. Марковский граф.

Метод моделирования на основе Марковских цепей широко применяют в таких областях, как автоматизация проектирования и организации в автоматизированных системах научных исследований, в системах исследования и проектирования, в системах массового обслуживания, при анализе различных сторон деятельности человека, в автоматизированном управлении производственными и другими процессами. Модели на основе Марковских цепей используется на этапах проектирования, создания, внедрения, эксплуатации систем, а также на различных уровнях их изучения, начиная от анализа работы элементов и кончая исследованием системы в целом при их взаимодействии с окружающей средой.

Марковские процессы являются частным видом случайных процессов.

Различают следующие виды Марковских случайных процессов:

* с дискретными состояниями и дискретным временем (цепь Маркова);
* с непрерывными состояниями и дискретным временем (Марковские последовательности);
* с дискретными состояниями и непрерывным временем (непрерывная цепь Маркова);
* с непрерывным состоянием и непрерывным временем.

Марковские процессы с дискретными состояниями представляют в графа состояний, где кружками обозначены состояния  системы , а стрелками – возможные переходы из состояния в состояние. Возможные задержки в прежнем состоянии изображают «петлей». Число состояний системы может быть как конечным, так и бесконечным (но счетным).

Безопасность – это частный случай более общего свойства ограниченности. Позиция является k-безопасной или k-ограниченной, если количество фишек в ней не может превышать k. Позиция называется ограниченной, если она k-безопасна для некоторого k. Сеть Петри ограниченна, если все ее позиции ограниченны.

Сеть Петри является строго сохраняющей, если общее число фишек в сети остается постоянным.

Переход tj обладает активностью уровня 0, если он никогда не может быть запущен. Переход tj обладает активностью уровня 1, если он потенциально запустим, то есть существует такая маркировка, что tj разрешен в этой маркировке. Переход tj обладает активностью уровня 2, если для всякого целого n существует последовательность запусков, в которой tj присутствует по крайней мере n раз.

Марковский процесс — случайный процесс, эволюция которого после любого заданного значения временного параметра t не зависит от эволюции, предшествовавшей t, при условии, что значение процесса в этот момент фиксировано («будущее» процесса не зависит от «прошлого» при известном «настоящем»; другая трактовка (Вентцель): «будущее» процесса зависит от «прошлого» лишь через «настоящее»).

Однородный марковский процесс называется такой стохастический процесс в котором вероятности переходов из состояния в состояние не зависят от времени t.

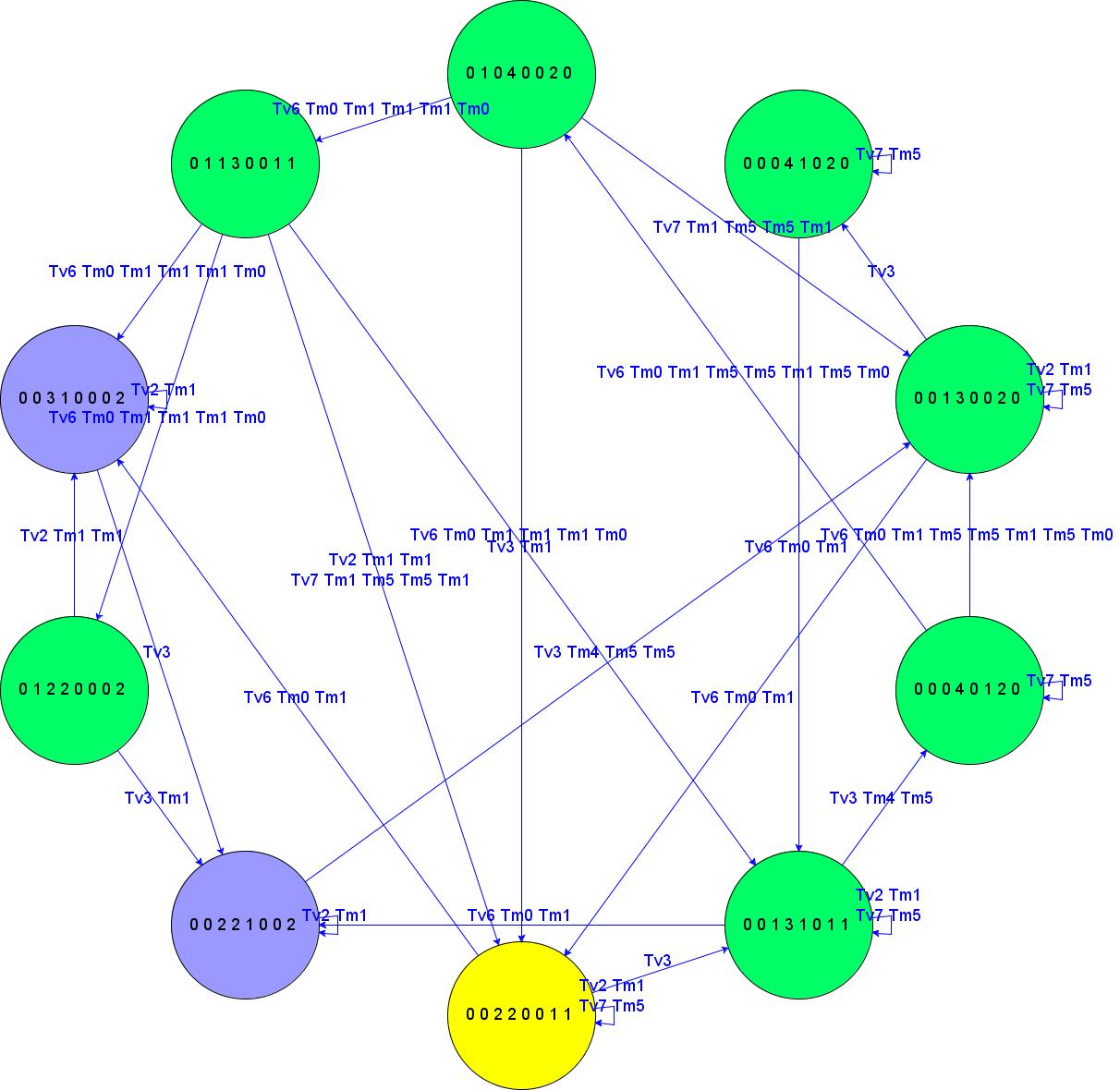
В общем случае и граф переходов G(k) марковской цепи и таблица переходов Т(к) отражают то распределение, которое имело место в к -ом сечении. И хотя множество распределений одинаково для любых пар моментов, но история процесса отражается уже многими графами и многими таблицами.

Идея построения графа на основе дерева достижимости очень проста:

Мы знаем все маркировки, которые возникали в режиме построения дерева, а также знаем из какой в какую переходили. То есть построение графа на основе дерева достижимости сводится к умению работать с графикой. Одинаковые маркировки из графа достижимости склеиваются в одну.

Рассмотрим примеры построения графа:

1)Приведем пример графа на основе дерева достижимости по данному заданию по сетям Петри рис. 4.1:



**Рис 4.1** Марковский граф

Различных маркировок 10 и нет тупиковых маркировок.

Третий этап разработки программного продукта - построение Марковского графа.

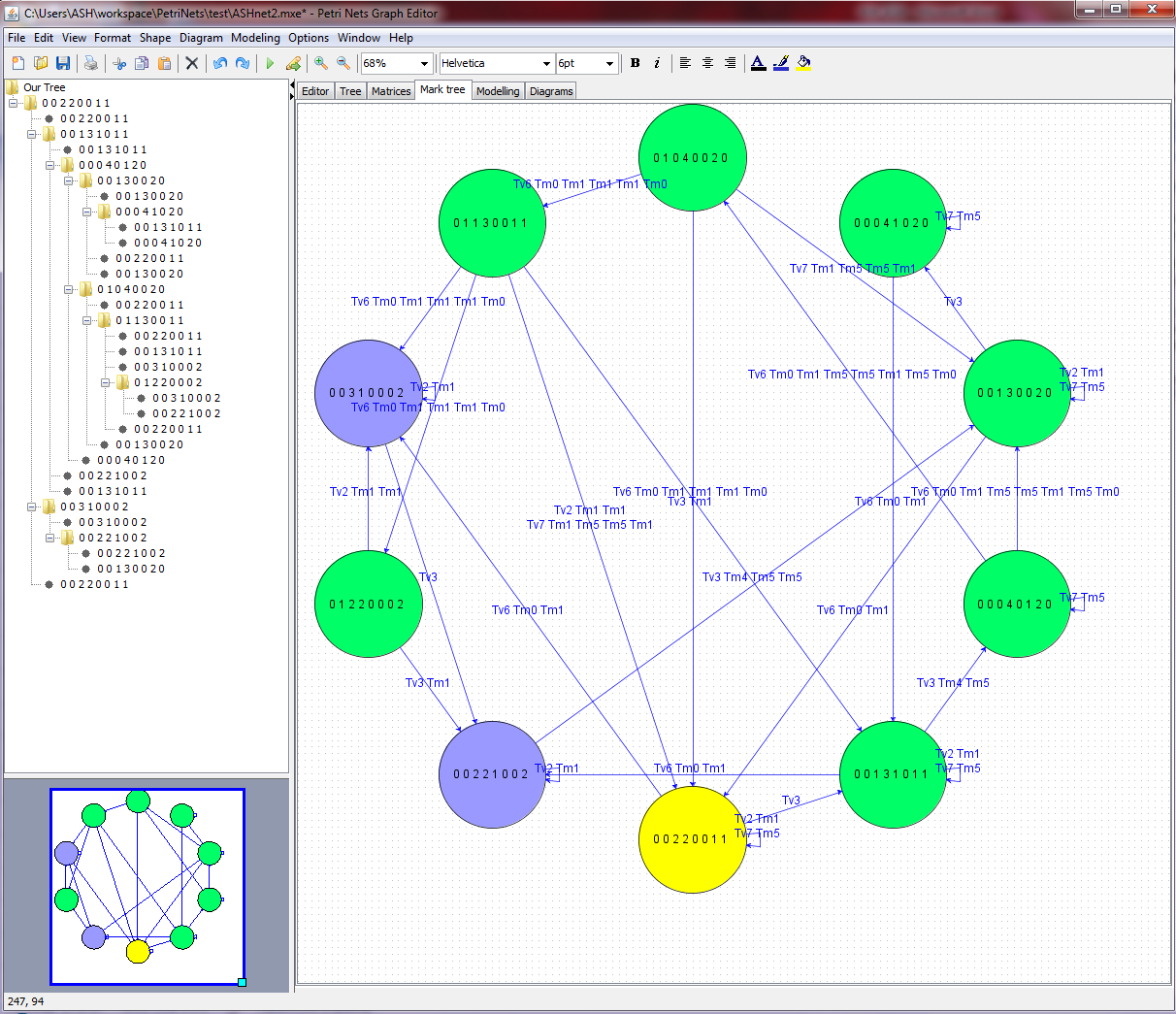
Им занимается метод fillTree() класса ModelConvertor, который, на ряду с преобразованием внутренней модели дерева достижимости во внешнюю, позволяет строить Марковский граф. Особенностью такого построения является обход внутренней модели дерева достижимости и, вместо построения каждой дублирующий вершины, построение связей к первой таковой найденной.

После построения необходимых связей, вызывается layout-менеджер библиотеки JGraph который выполняет расстановку вершин в окружность.

## 4.1 Инструкция пользователя. Построение Марковского графа

Для построения Марковского графа достижимости следует открыть уже готовую или набрать новую модель сети.

* Открыть Марковский граф можно на вкладке “Mark tree”. Переходы между вершинами обозначаются стрелками. Также присутствуют подписи переходов, которые срабатывают при переходе из одной вершины Марковской цепи в другую.



**Рис.4.2** Марковский граф в рабочей области приложения

# 5. Исследование моделей информационных потоков

## Теоретические сведения

Информационный поток – последовательность однородных случайных событий, каждое из которых несёт одну или несколько заявок.

Если в каждом событии 1 заявка, то поток ординарный. В одном и том же потоке события однородны – требуют одинакового характера обслуживания. События потока могут быть просто зарегистрированы.

Поток событий называется **регулярным**, если события следуют одно за другим через строго определенные промежутки времени. Такой поток сравнительно редко встречается в реальных системах, но представляет интерес как предельный случай.

1. Поток событий называется **стационарным**, если вероятность попадания того или иного числа событий на участок времени длиной х зависит только от длины участка и не зависит от того, где именно на оси Ot расположен этот участок.

2. Поток событий называется **потоком без последействия**, если для любых неперекрывающихся участков времени число событий, попадающих на один из них, не зависит от числа событий, попадающих на другие.

3. Поток событий называется **ординарным**, если вероятность попадания на элементарный участок двух или более событий пренебрежимо мала по сравнению с вероятностью попадания одного события.

Если поток событий обладает всеми тремя свойствами (т. е. стационарен, ординарен и не имеет последействия), то он называется **простейшим** (или стационарным пуассоновским) потоком. Название «пуассоновский» связано с тем, что при соблюдении условий 1-т-З число событий, попадающих на любой фиксированный интервал времени, будет распределено по закону Пуассона.

**Характеристики потоков**

**Диспе́рсия случа́йной величины́** — мера разброса данной [случайной величины](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0), т. е. её отклонения от [математического ожидания](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BE%D0%B6%D0%B8%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5). Обозначается *D*[*X*] в русской литературе и \operatorname{var}\,X([англ.](http://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *variance*) в зарубежной. В статистике часто употребляется обозначение \sigma_X^2или \displaystyle \sigma^2. Квадратный корень из дисперсии \displaystyle \sigmaназывается [среднеквадрати́чным отклоне́нием](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D1%87%D0%BD%D0%BE%D0%B5_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%BE%D0%B5_%D0%BE%D1%82%D0%BA%D0%BB%D0%BE%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5), [станда́ртным отклоне́нием](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D1%87%D0%BD%D0%BE%D0%B5_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%BE%D0%B5_%D0%BE%D1%82%D0%BA%D0%BB%D0%BE%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5) или стандартным разбросом. Стандартное отклонение измеряется в тех же [единицах](http://ru.wikipedia.org/wiki/%D0%95%D0%B4%D0%B8%D0%BD%D0%B8%D1%86%D1%8B_%D0%B8%D0%B7%D0%BC%D0%B5%D1%80%D0%B5%D0%BD%D0%B8%D1%8F), что и сама случайная величина, а дисперсия измеряется в квадратах этой единицы измерения.

Из [неравенства Чебышёва](http://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D1%80%D0%B0%D0%B2%D0%B5%D0%BD%D1%81%D1%82%D0%B2%D0%BE_%D0%A7%D0%B5%D0%B1%D1%8B%D1%88%D1%91%D0%B2%D0%B0_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9)) следует, что случайная величина удаляется от её математического ожидания на более чем *k* стандартных отклонений с вероятностью менее 1/*k*2. Так, например, как минимум в 75% случаев случайная величина удалена от её среднего не более чем на два стандартных отклонения, а в примерно 89% — не более чем на три.

**Математи́ческое ожида́ние** — понятие [среднего значения](http://ru.wikipedia.org/wiki/%D0%A1%D1%80%D0%B5%D0%B4%D0%BD%D0%B5%D0%B5_%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) [случайной величины](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0) в [теории вероятностей](http://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9). В зарубежной литературе обозначается через \mathbb{E}[X], в русской *M*[*X*]. В статистике часто используют обозначение μ.

* Если *FX*(*x*) — [функция распределения](http://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) случайной величины, то её математическое ожидание задаётся [интегралом Лебега — Стилтьеса](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B0%D0%BB_%D0%9B%D0%B5%D0%B1%D0%B5%D0%B3%D0%B0_%E2%80%94_%D0%A1%D1%82%D0%B8%D0%BB%D1%82%D1%8C%D0%B5%D1%81%D0%B0):

M[X]=\int\limits_{-\infty}^{\infty}\!x\, dF_X(x).

Математическое ожидание дискретного распределения

* Если *X* — [дискретная случайная величина](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D1%80%D0%B5%D1%82%D0%BD%D0%BE%D0%B5_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5), имеющая [распределение](http://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B8)

\mathbb{P}(X=x_i) = p_i,\; \sum\limits_{i=1}^{\infty} p_i = 1,

Созданы модели информационного потока по заданной интенсивности и коэффициенту вариации:

* Поток с равномерным распределением
* Поток с экспоненциальным распределением (Поток Пуассона)
* Поток с распределением Эрланга
* Поток с гипер экспоненциальным распределением

Вывести гистограмму распределения для различных интенсивностей.

Генератор псевдослучайных чисел— алгоритм, генерирующий последовательность чисел, элементы которой почти независимы друг от друга и подчиняются заданному распределению (обычно равномерному).

Современная информатика широко использует псевдослучайные числа в самых разных приложениях — от метода Монте-Карло и имитационного моделирования до криптографии. При этом от качества используемых ГПСЧ напрямую зависит качество получаемых результатов. Это обстоятельство подчёркивает известный афоризм Роберта Р. Кавью : «генерация случайных чисел слишком важна, чтобы оставлять её на волю случая».

Никакой детерминированный алгоритм не может генерировать полностью случайные числа, он может только аппроксимировать некоторые свойства случайных чисел. Как сказал Джон фон Нейман, «всякий, кто питает слабость к арифметическим методам получения случайных чисел, грешен вне всяких сомнений».

Любой ГПСЧ с ограниченными ресурсами рано или поздно зацикливается — начинает повторять одну и ту же последовательность чисел

Большинство простых арифметических генераторов хотя и обладают большой скоростью, но страдают от многих серьёзных недостатков:

1)Слишком короткий период/периоды.

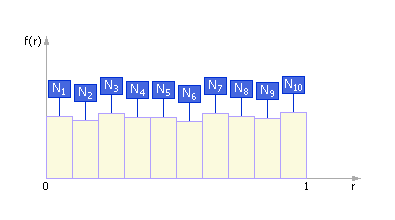
2)Последовательные значения не являются независимыми.

3)Некоторые биты «менее случайны», чем другие.

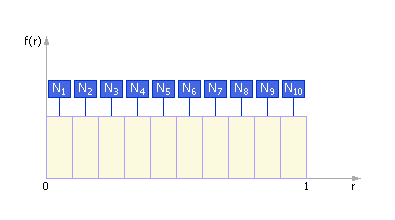
4)Неравномерное одномерное распределение.

5)Обратимость.

За эталон генератора случайных чисел (ГСЧ) принят такой генератор, который порождает последовательность случайных чисел с равномерным законом распределения в интервале (0; 1). За одно обращение данный генератор возвращает одно случайное число. Если наблюдать такой ГСЧ достаточно длительное время, то окажется, что, например, в каждый из десяти интервалов (0; 0.1), (0.1; 0.2), (0.2; 0.3), …, (0.9; 1) попадет практически одинаковое количество случайных чисел — то есть они будут распределены равномерно по всему интервалу (0; 1). Если изобразить на графике k = 10 интервалов и частоты Ni попаданий в них, то получится экспериментальная кривая плотности распределения случайных чисел (см. рис. 1).



Заметим, что в идеале кривая плотности распределения случайных чисел выглядела бы так, как показано на рис. 2. То есть в идеальном случае в каждый интервал попадает одинаковое число точек: Ni = N/k, где N — общее число точек, k — количество интервалов, i = 1, …, k.



Линейный конгруэнтный метод — один из алгоритмов генерации псевдослучайных чисел. Применяется в простых случаях и не обладает криптографической стойкостью. Входит в стандартные библиотеки различных компиляторов.

Этот алгоритм заключается в итеративном применении следующей формулы:

******

где a > 0, c > 0, m > 0 — некоторые целочисленные константы.

Период не может быть больше числа m. Следовательно m должно быть довольно большим.

Множитель a (0 <= a <= m).

Инкрементирующее значение c (0 <= c <= m).

Начальное значение X0 (0 <= X0 < m).

Получаемая последовательность зависит от выбора стартового числа X0 и при разных его значениях получаются различные последовательности случайных чисел. В то же время, многие свойства последовательности Xj определяются выбором коэффициентов в формуле и не зависят от выбора стартового числа. Ясно, что последовательность чисел, генерируемая таким алгоритмом, периодична с периодом, не превышающим m. Метод довольно старый - 1950х годов. Разработал его Деррик Лемер.

Статистические свойства получаемой последовательности случайных чисел полностью определяются выбором констант a и c при заданной разрядности e. Для этих констант выписаны условия, гарантирующие удовлетворительное качество получаемых случайных чисел.

Кроме линейного конгруэнтного генератора существует множество других. Например Вихрь Мерсенна, изобретённый двумя японскими учёными (непомню как из зовут) в 1997. У него очень большой (очень большой это мягко сказано) период. Кстати, вихрь Мерсенна использует линейный конгруэнтный генератор для установления начального значения (seed).

Генераторы случайных чисел применяются при шифровании. Но здесь используются специальные генераторы - криптографически защищённые генераторы псевдослучайных чисел. Например блочный шифр (block cipher), потоковый шифр (stream cipher), который был разработан на основе шифра Вернама.

Проверка качества работы генератора

От качества работы ГСЧ зависит качество работы всей системы и точность результатов. Поэтому случайная последовательность, порождаемая ГСЧ, должна удовлетворять целому ряду критериев.

Осуществляемые проверки бывают двух типов:

1)проверки на равномерность распределения;

2)проверки на статистическую независимость.

## 5.1 Линейный конгруэнтный генератор:

Для N=5000, k=10.

1) a=29^4 c=61^4 d=15^4 Погрешность=0.024

2) a=33^4 c=95^4 d=16^4 Погрешность=0.028

3) a=34^4 c=27^4 d=35^4 Погрешность=0.0020

4) a=36^4 c=67^4 d=37^4 Погрешность=0.0020

5) a=37^4 c=62^4 d=11^4 Погрешность=0.02

6) a=39^4 c=11^4 d=16^4 Погрешность=0.022

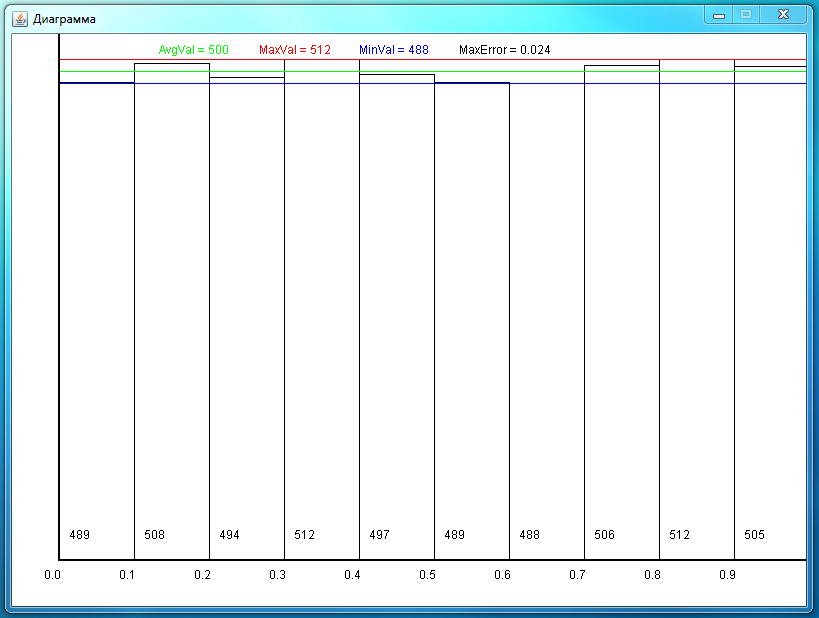
7) a=39^4 c=25^4 d=14^4 Погрешность=0.028

8) a=41^4 c=12^4 d=29^4 Погрешность=0.022

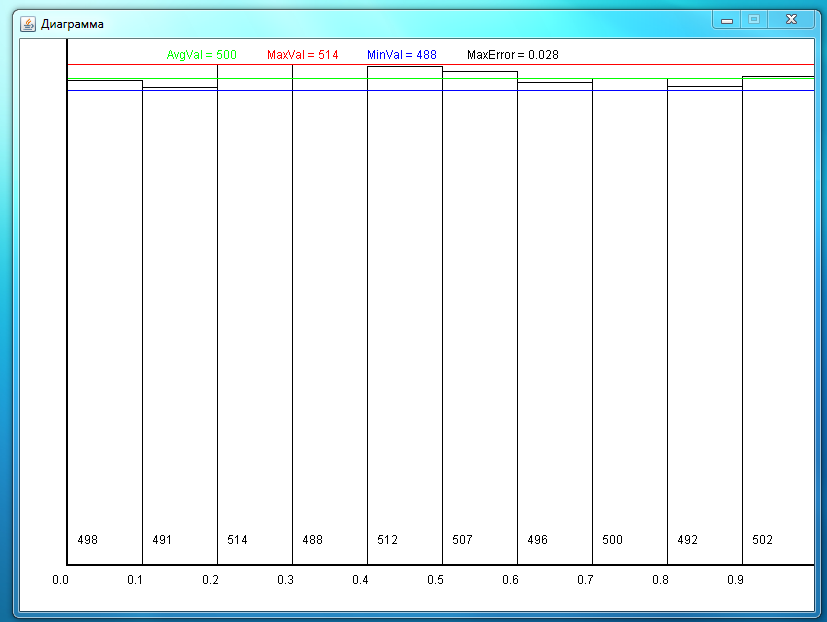
9) a=41^4 c=33^4 d=29^4 Погрешность=0.028

10) a=41^4 c=91^4 d=11^4 Погрешность=0.014

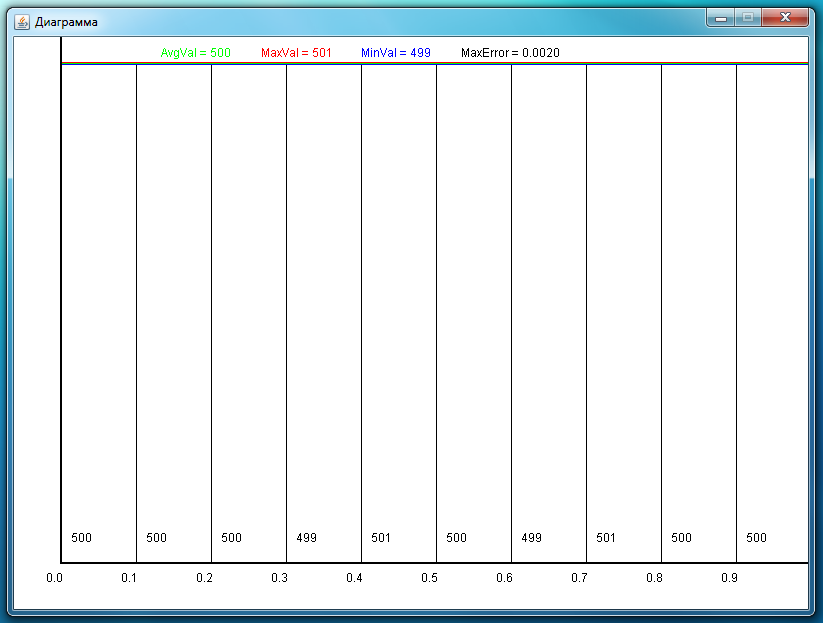
Для генераторов 1,2 и 3 представим гистограммы. (рис. 5.1.1, рис 5.1.2, рис 5.1.3)



**Рис. 5.1.1** Гистограмма №1(Линейный конгруэнтный генератор)



**Рис. 5.1.2**. Гистограмма №2(Линейный конгруэнтный генератор)

******

**Рис. 5.1.3**. Гистограмма №3(Линейный конгруэнтный генератор)

## 5.2 Аддитивный генератор псевдослучайных чисел.

**W[i]= frac(Wi-1+Wi-2+….+Wi-n) n>10.**

Первые n чисел берутся из других генераторов.

В 1958 году Дж.Ж. Митчелл и Д.Ф. Мур предложили несколько необычную последовательность, определенную так:



Где m – четное число, n ≥ 55, а X0, … , X54 – произвольные целые не все четные числа.

Аддитивность — свойство величин по отношению к сложению, состоящее в том, что значение величины, соответствующее целому объекту, равно сумме значений величин, соответствующих его частям, в некотором классе возможных разбиений объекта на части.

Аддитивный (лат. additivus ← additio прибавляю) — относящийся к сложению.

В теории чисел аддитивная функция — функция, удовлетворяющая соотношению



Очень эффективна с точки зрения производительности схема, на-

зываемая *аддитивным генератором*. Самостоятельного значения эти

генераторы в силу своей криптографической слабости не имеют, но

могут использоваться в качестве строительных блоков при создании

стойких генераторов ПСП. Генератор состоит из *n* регистров разряд-

ностью *M* каждый и сумматора по модулю 2*M*.

Примером аддитивного генератора является Fish (Fibonacci shrinking

generator – прореживаемый генератор Фибоначчи).

**Центра́льные преде́льные теоре́мы (Ц.П.Т.)** — класс теорем в теории вероятностей, утверждающих, что сумма большого количества независимых случайных величин имеет распределение, близкое к нормальному. Так как многие случайные величины в приложениях являются суммами нескольких случайных факторов, центральные предельные теоремы обосновывают популярность нормального распределения.

Пусть X_1,\ldots, X_n,\ldotsесть бесконечная последовательность независимых одинаково распределённых случайных величин, имеющих конечное [математическое ожидание](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BE%D0%B6%D0%B8%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5) и [дисперсию](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BF%D0%B5%D1%80%D1%81%D0%B8%D1%8F_%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%BE%D0%B9_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D1%8B). Обозначим последние μ и σ2, соответственно. Пусть S_n = \sum\limits_{i=1}^n X_i. Тогда

\frac{S_n - \mu n}{\sigma \sqrt n} \to N(0,1)[по распределению](http://ru.wikipedia.org/wiki/%D0%A1%D1%85%D0%BE%D0%B4%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C_%D0%BF%D0%BE_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8E) при n \to \infty,

где N(0,1) — [нормальное распределение](http://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5) с нулевым [математическим ожиданием](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BE%D0%B6%D0%B8%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5) и [стандартным отклонением](http://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%BE%D0%B5_%D0%BE%D1%82%D0%BA%D0%BB%D0%BE%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5), равным единице. Обозначив символом \bar{X}[выборочное среднее](http://ru.wikipedia.org/wiki/%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D1%87%D0%BD%D0%BE%D0%B5_%D1%81%D1%80%D0%B5%D0%B4%D0%BD%D0%B5%D0%B5) первых n величин, то есть \bar{X} = \frac{1}{n} \sum\limits_{i=1}^n X_i, мы можем переписать результат центральной предельной теоремы в следующем виде:

\frac{\bar{X} - \mu}{\sigma / \sqrt{n}} \to N(0,1)[по распределению](http://ru.wikipedia.org/wiki/%D0%A1%D1%85%D0%BE%D0%B4%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C_%D0%BF%D0%BE_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8E) при n \to \infty.

***Результаты:***

Для Выборки 5000, m=10.

Отберем 10 лучших значений N для линейного конгруэнтного генератора:

ЛКГ

a=34^4 c=27^4 d=35^4

1) 12 Отклонение: 0.038

2) 20 Отклонение: 0.042

3) 28 Отклонение: 0.044

4) 45 Отклонение: 0.03

5) 69 Отклонение: 0.044

6) 73 Отклонение: 0.036

7) 82 Отклонение: 0.042

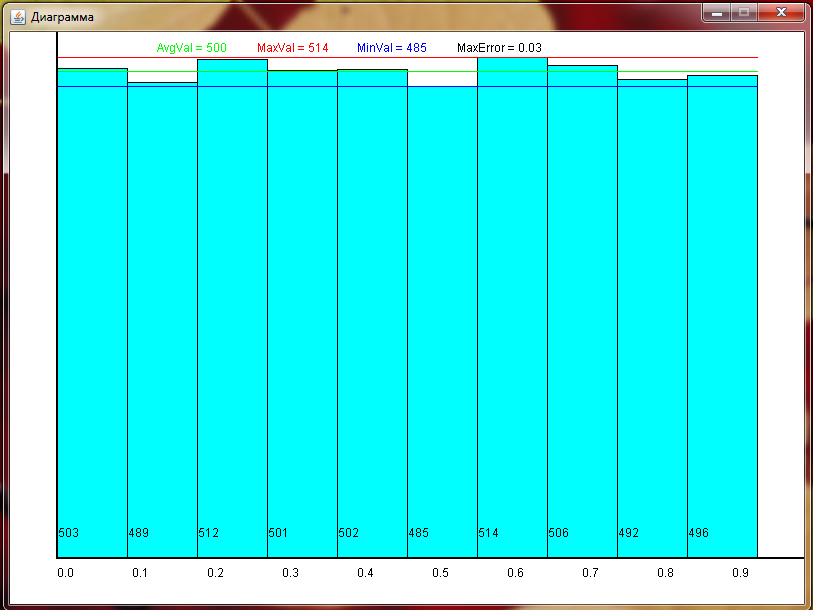
8) 90 Отклонение: 0.042

9) 105 Отклонение: 0.026

10) 116 Отклонение: 0.042

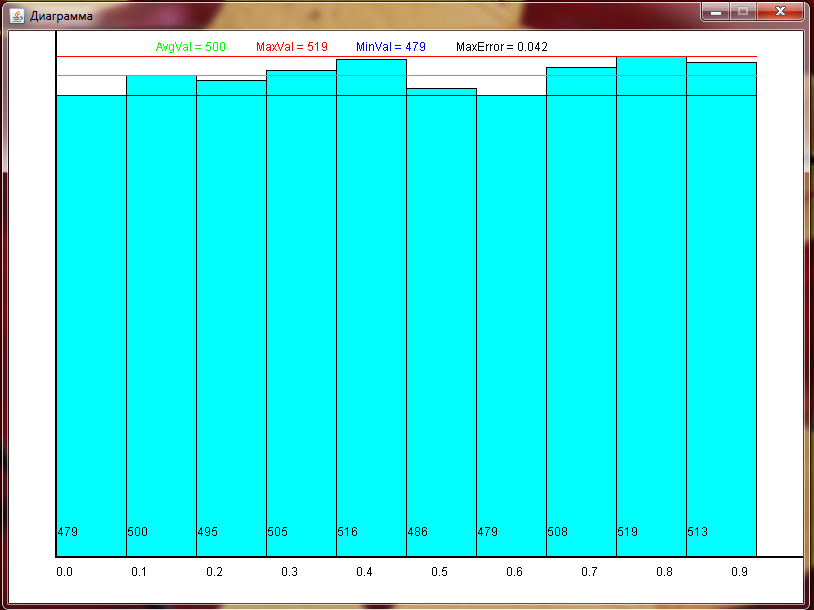
Для генераторов 4,9 и 9 представим гистограммы. (рис. 5.2.1, рис 5.2.2, рис 5.2.3)

№4



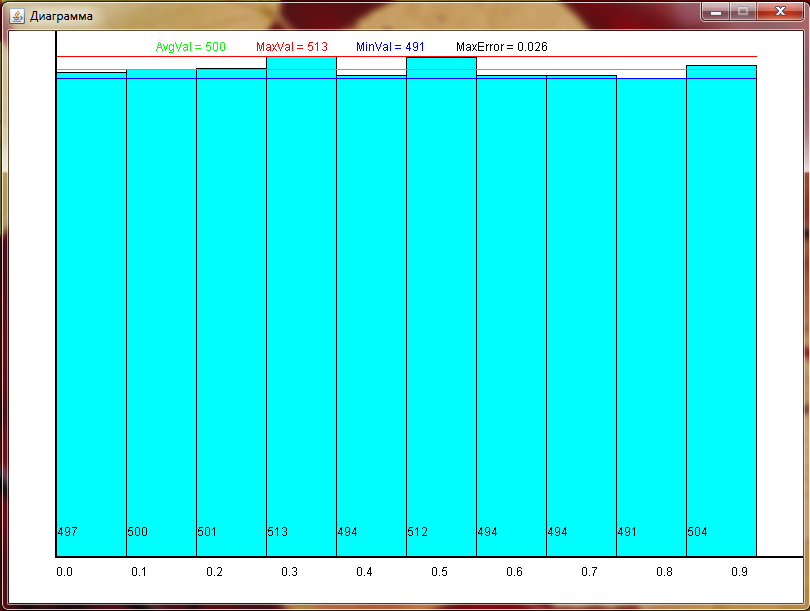
**Рис. 5.2.1** Гистограмма №1 (Аддитивный генератор)

№8



**Рис. 5.2.2** Гистограмма №8(Аддитивный генератор)

№9



**Рис. 5.2.3** Гистограмма №9(Аддитивный генератор)

Для равномерного потока: Матожидание =b/2; b=2/альфа

Дисперсия=b2/12

Вариация=1/3

Для Пуассона: Матожидание =1/альфа

Дисперсия=1/(альфа)2

Вариация=1

**Распределение Пуассона** моделирует [случайную величину](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0), представляющую собой число [событий](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%BE%D0%B5_%D1%81%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D0%B5), произошедших за фиксированное время, при условии, что данные события происходят с некоторой фиксированной средней интенсивностью и [независимо](http://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9)) друг от друга. Распределение Пуассона играет ключевую роль в [Теории массового обслуживания](http://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%BC%D0%B0%D1%81%D1%81%D0%BE%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D1%81%D0%BB%D1%83%D0%B6%D0%B8%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F).

Выберем фиксированное число λ > 0 и определим [дискретное распределение](http://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9), задаваемое следующей [функцией вероятности](http://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B8):

p(k) \equiv \mathbb{P}(Y=k) = \frac{\lambda^k}{k!}\, e^{-\lambda},

где

* *k*! обозначает [факториал](http://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%B0%D0%BB),
* e = 2.718281828\ldots— [основание натурального логарифма](http://ru.wikipedia.org/wiki/E_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BA%D0%BE%D0%BD%D1%81%D1%82%D0%B0%D0%BD%D1%82%D0%B0)).

Тот факт, что случайная величина *Y* имеет распределение Пуассона с параметром λ, записывается: Y \sim~ \mathrm{P}(\lambda).

Свойства распределения Пуассона

* Сумма независимых пуассоновских случайных величин, также имеет распределение Пуассона. Пусть Y_i \sim \mathrm{P}(\lambda_i),\; i=1,\ldots,n. Тогда

Y = \sum\limits_{i=1}^n Y_i \sim \mathrm{P}\left(\sum\limits_{i=1}^n \lambda_i\right).

* Пусть Y_i \sim \mathrm{P}(\lambda_i),\; i=1,2, и *Y* = *Y*1 + *Y*2. Тогда [условное распределение](http://ru.wikipedia.org/wiki/%D0%A3%D1%81%D0%BB%D0%BE%D0%B2%D0%BD%D0%BE%D0%B5_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5) *Y*1 при условии, что *Y* = *y*, биномиально. Более точно:

Y_1\mid Y = y \sim \mathrm{Bin}\left(y, \frac{\lambda_1}{\lambda_1+\lambda_2}\right) .

## 5.3 Равномерное распределение

***Результаты:***

---Линейный конгруэнтный генератор---

Выборка: 10000. Кол-во интервалов: 20.

a=93^4 c=91^4 d=19^4

Максимальное отклонение от матожидания: 0.05390539053905388

0) Лямбда: 0.1 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 10.0 Дисперсия: 33.333333333333336 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 9.977666656741475 Дисперсия: 33.38056348070157 Коефициент вариации: 0.3353016437363842

1) Лямбда: 0.5 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 2.0 Дисперсия: 1.3333333333333333 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 1.9955333313483024 Дисперсия: 1.335222539228069 Коефициент вариации: 0.33530164373638327

2) Лямбда: 0.7 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 1.4285714285714286 Дисперсия: 0.6802721088435374 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 1.4253809509630766 Дисперсия: 0.6812359894020741 Коефициент вариации: 0.3353016437363807

3) Лямбда: 1.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 1.0 Дисперсия: 0.3333333333333333 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.9977666656741512 Дисперсия: 0.33380563480701725 Коефициент вариации: 0.33530164373638327

4) Лямбда: 4.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 0.25 Дисперсия: 0.020833333333333332 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.2494416664185378 Дисперсия: 0.020862852175438578 Коефициент вариации: 0.33530164373638327

5) Лямбда: 8.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.125 Дисперсия: 0.005208333333333333 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.12453236244961434 Дисперсия: 0.005229336341966782 Коефициент вариации: 0.337195773035249

6) Лямбда: 12.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.08333333333333333 Дисперсия: 0.0023148148148148147 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.08302157496640945 Дисперсия: 0.0023241494853185564 Коефициент вариации: 0.3371957730352479

7) Лямбда: 25.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.04 Дисперсия: 5.333333333333334E-4 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.039850355983876457 Дисперсия: 5.354840414173989E-4 Коефициент вариации: 0.3371957730352515

8) Лямбда: 30.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:30

---Теоретические значения---

Математическое ожидание: 0.03333333333333333 Дисперсия: 3.7037037037037035E-4 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.03320862998656372 Дисперсия: 3.7186391765097084E-4 Коефициент вариации: 0.3371957730352507

9) Лямбда: 50.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:30

---Теоретические значения---

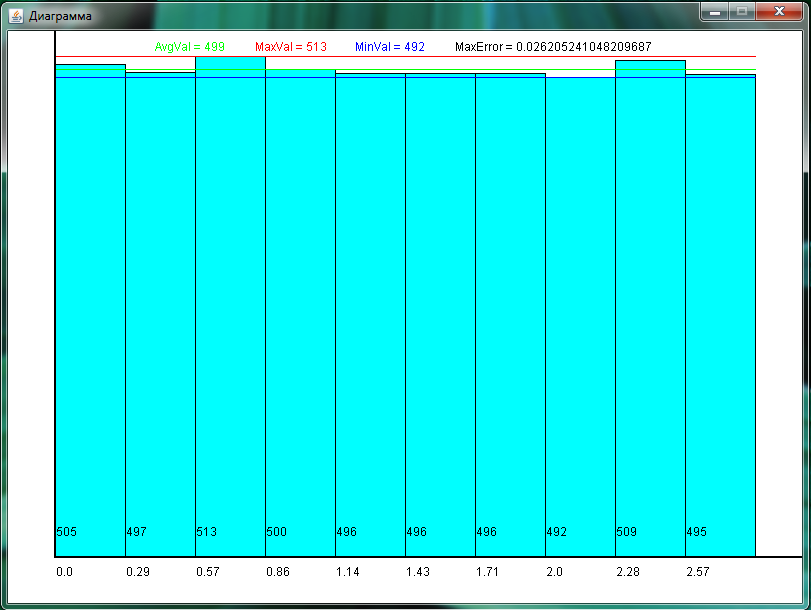
Математическое ожидание: 0.02 Дисперсия: 1.3333333333333334E-4 Коефициент вариации: 0.3333333333333333

---Эксперементальные значения---

Математическое ожидание: 0.019925177991938228 Дисперсия: 1.3387101035434974E-4 Коефициент вариации: 0.3371957730352515

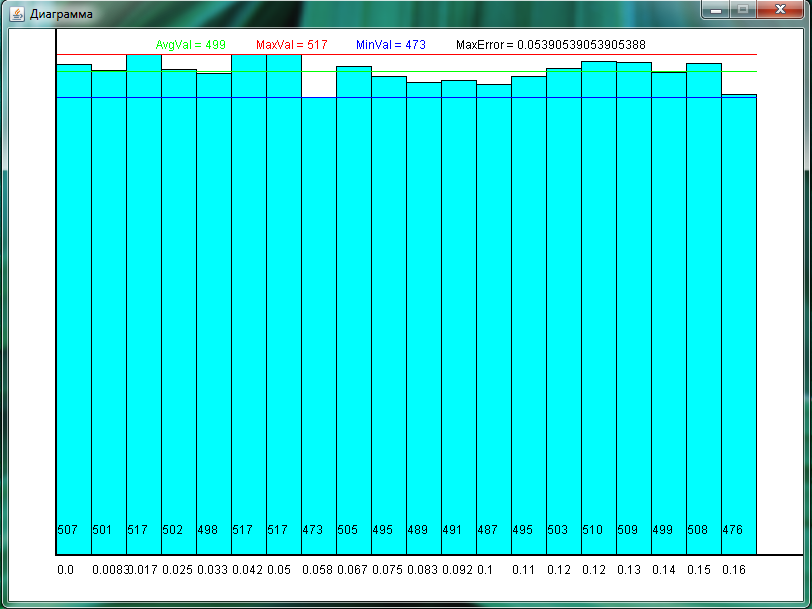
Представим гистограммы (рис. 5.3.1, рис 5.3.2, рис 5.3.3)

№2



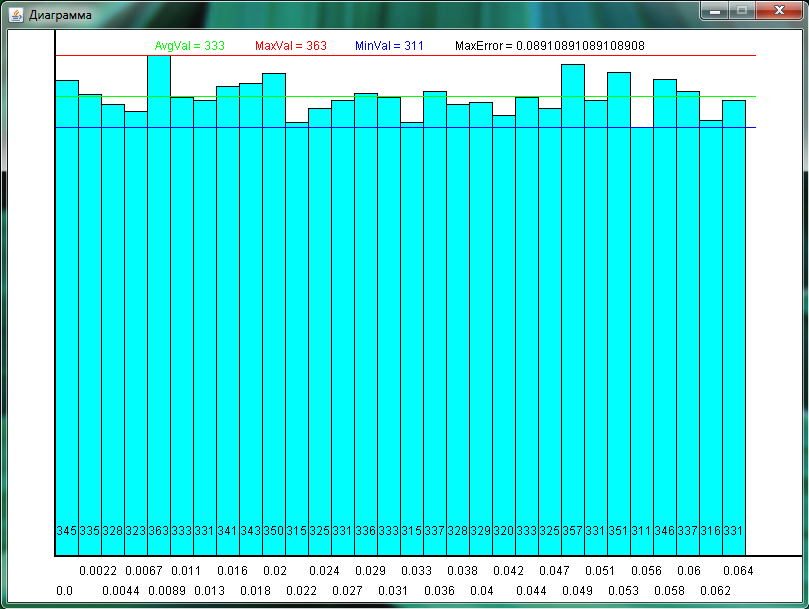
**Рис. 5.3.1** Гистограмма №2 (Равномерное распределение)

№6



**Рис. 5.3.2** Гистограмма №6(Равномерное распределение)

№8



**Рис. 5.3.3** Гистограмма №8(Равномерное распределение)

## 5.4 Распределение Пуассона.

***Результаты:***

---Линейный конгруэнтный генератор---

Выборка: 10000. Кол-во интервалов: 20.

a=93^4 c=91^4 d=19^4

Максимальное отклонение от матожидания: 0.05390539053905388

0) Лямбда: 0.1 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 10.0 Дисперсия: 99.99999999999999 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 9.982936599193875 Дисперсия: 99.8214457897173 Коефициент вариации: 1.001629783646974

1) Лямбда: 0.5 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 2.0 Дисперсия: 4.0 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 1.9965873198387745 Дисперсия: 3.992857831588689 Коефициент вариации: 1.0016297836469739

2) Лямбда: 0.7 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 1.4285714285714286 Дисперсия: 2.0408163265306127 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 1.426133799884838 Дисперсия: 2.03717236305546 Коефициент вариации: 1.0016297836469783

3) Лямбда: 1.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 1.0 Дисперсия: 1.0 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.9982936599193872 Дисперсия: 0.9982144578971722 Коефициент вариации: 1.0016297836469739

4) Лямбда: 4.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10

---Теоретические значения---

Математическое ожидание: 0.25 Дисперсия: 0.0625 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.2495734149798468 Дисперсия: 0.062388403618573264 Коефициент вариации: 1.0016297836469739

5) Лямбда: 8.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.125 Дисперсия: 0.015625 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.12432842471637948 Дисперсия: 0.01547771731021282 Коефициент вариации: 1.0013042240442893

6) Лямбда: 12.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.08333333333333333 Дисперсия: 0.006944444444444444 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.08288561647758644 Дисперсия: 0.0068789854712056776 Коефициент вариации: 1.0013042240442835

7) Лямбда: 25.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:20

---Теоретические значения---

Математическое ожидание: 0.04 Дисперсия: 0.0016 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.03978509590924135 Дисперсия: 0.0015849182525657954 Коефициент вариации: 1.0013042240442953

8) Лямбда: 30.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:30

---Теоретические значения---

Математическое ожидание: 0.03333333333333333 Дисперсия: 0.0011111111111111111 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.03315424659103432 Дисперсия: 0.0011006376753929141 Коефициент вариации: 1.0013042240443044

9) Лямбда: 50.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:30

---Теоретические значения---

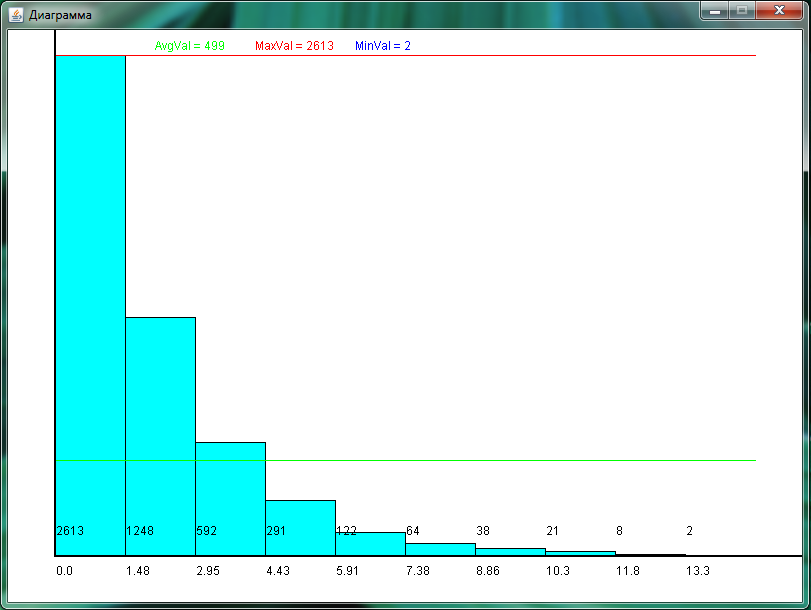
Математическое ожидание: 0.02 Дисперсия: 4.0E-4 Коефициент вариации: 1.0

---Эксперементальные значения---

Математическое ожидание: 0.019892547954620674 Дисперсия: 3.9622956314144884E-4 Коефициент вариации: 1.0013042240442953

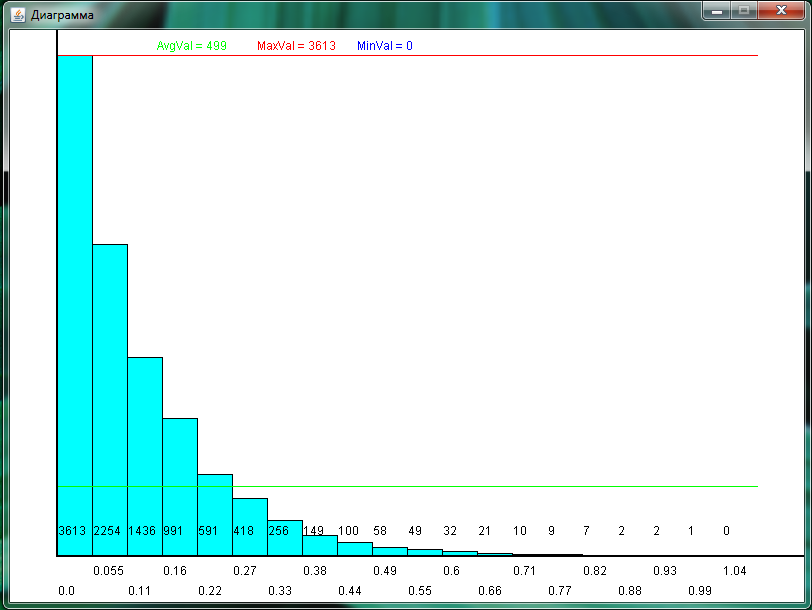
Представим гистограммы для 1, 5, 9 случаев(рис. 5.4.1, рис 5.4.2, рис 5.4.3)

№1



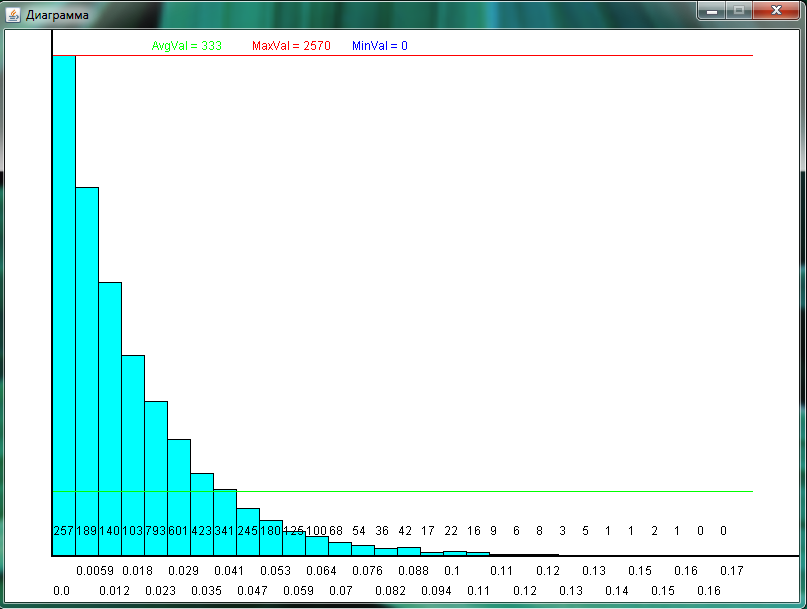
**Рис. 5.4.1** Гистограмма №1(Распределение Пуассона)

№5



**Рис. 5.4.2** Гистограмма №5(Распределение Пуассона

№9

******

**Рис. 5.4.3** Гистограмма №9(Распределение Пуассона)

**Распределение Эрланга** – это гамма-распределение er6с параметромa, принимающим лишь целые значения. Здесь оно приводится лишь из-за того, что часто встречается в инженерных приложениях, особенно телефонии.

При a=1 распределение Эрланга совпадает с экспоненциальным.

Сумма a независимых случайных величин er7, i=1…a, подчиняющихся экспоненциальному распределению с средним b, имеет распределение Эрланга с параметрами a и b.

**Эрланга: ti= -1/(k\*λ) \*∑ln Wi; g<1; λ=α/k; D=1/ λ2/k; M=k/ λ; g=1/k;**

[Cлучайная величина](http://www.wikiznanie.ru/ru-wz/index.php/%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0) ξ имеет **гиперэкспоненциальное распределение** с параметрами (m; \alpha _1, \alpha _2, \ldots, \alpha _m; \lambda _1, \lambda _2, \ldots, \lambda _m), \alpha _k \lambda _k \geq 0, \sum _{k=1}^m \alpha _k =1(смесь экспоненциальных распределений), если

f(x) = \left\{ \begin{matrix} \sum_{k=1}^m \alpha _k \lambda _k e^{-\lambda _k x}, & x \geq 0 \\ 0, & x < 0 \end{matrix} \right..

Характеристическая функция

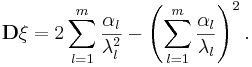
\varphi (t) = \sum_{k=1}^m \frac{\alpha _k \lambda _k}{\lambda _k - it}.

Свойства

* Моменты:

\mathbf{E}\xi ^k = \sum_{l=1}^m \alpha _l \frac{k!}{\lambda _l^k} ;

* Дисперсия:



**Гиперэкспоненциальный: ti=-1/α\*ln Wi; g>1;**

**α1=2 λ\*phi; α2=2(1-phi) λ; phi=1/2-sqrt(1/4+1/(2g+2)**

## 5.5 Распределение Эрланга

***Результаты***

Эрланга.

---Линейный конгруэнтный генератор---

Выборка: 10000. Кол-во интервалов: 20.

a=93^4 c=91^4 d=19^4

Максимальное отклонение от матожидания: 0.05390539053905388

0) Лямбда: 0.1 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 1

---Теоретические значения---

Математическое ожидание: 10.0

Дисперсия: 99.99999999999999

Коэффициент вариации: 0.9999999999999999

---Экспериментальные значения---

Математическое ожидание: 10.053310950260403 Дисперсия: 102.15557333517162 Коэффициент вариации: 1.0107501965598815

Максимальное значение: 108.63849778625764

Макс. знач. / Мат. ожидание: 10.806240682672176

1) Лямбда: 0.5 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 2

---Теоретические значения---

Математическое ожидание: 2.0

Дисперсия: 2.0

Коэффициент вариации: 0.5

---Экспериментальные значения---

Математическое ожидание: 2.003096839987106 Дисперсия: 1.9996541073396121 Коэффициент вариации: 0.4983689630102637

Максимальное значение: 10.032106756310815

Макс. знач. / Мат. ожидание: 5.008298428734675

2) Лямбда: 0.7 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 3

---Теоретические значения---

Математическое ожидание: 1.4285714285714286

Дисперсия: 0.6802721088435375

Коэффициент вариации: 0.33333333333333337

---Экспериментальные значения---

Математическое ожидание: 1.4052809234844517 Дисперсия: 0.6794935887822766 Коэффициент вариации: 0.3440797086306347

Максимальное значение: 4.994166280850213

Макс. знач. / Мат. ожидание: 3.5538561702431517

3) Лямбда: 1.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 4

---Теоретические значения---

Математическое ожидание: 1.0

Дисперсия: 0.25

Коэффициент вариации: 0.25

---Экспериментальные значения---

Математическое ожидание: 1.02204102932731 Дисперсия: 0.23927418761968178 Коэффициент вариации: 0.22906523883513394

Максимальное значение: 2.889442895015665

Макс. знач. / Мат. ожидание: 2.827130038915803

4) Лямбда: 4.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 5

---Теоретические значения---

Математическое ожидание: 0.25

Дисперсия: 0.0125

Коэффициент вариации: 0.2

---Экспериментальные значения---

Математическое ожидание: 0.25546399173815737 Дисперсия: 0.011862124373543654 Коэффициент вариации: 0.18176199691224001

Максимальное значение: 0.6134695850665672

Макс. знач. / Мат. ожидание: 2.4013935619363314

5) Лямбда: 8.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 1

---Теоретические значения---

Математическое ожидание: 0.125

Дисперсия: 0.015625

Коэффициент вариации: 1.0

---Экспериментальные значения---

Математическое ожидание: 0.12599746152994307 Дисперсия: 0.015902921000159453 Коэффициент вариации: 1.0017360669382196

Максимальное значение: 1.3579812223282206

Макс. знач. / Мат. ожидание: 10.777845885454596

6) Лямбда: 12.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 3

---Теоретические значения---

Математическое ожидание: 0.08333333333333333

Дисперсия: 0.0023148148148148147

Коэффициент вариации: 0.3333333333333333

---Экспериментальные значения---

Математическое ожидание: 0.08364738890946061 Дисперсия: 0.002328004338046578 Коэффициент вариации: 0.33272007681534227

Максимальное значение: 0.2941621020154214

Макс. знач. / Мат. ожидание: 3.5166919834619175

7) Лямбда: 25.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 5

---Теоретические значения---

Математическое ожидание: 0.04

Дисперсия: 3.2E-4

Коэффициент вариации: 0.2

---Экспериментальные значения---

Математическое ожидание: 0.03982584146020412 Дисперсия: 3.0467941445027393E-4 Коэффициент вариации: 0.19209373069300242

Максимальное значение: 0.11212389362849769

Макс. знач. / Мат. ожидание: 2.815355294891565

8) Лямбда: 30.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:30 Коэффициент k: 2

---Теоретические значения---

Математическое ожидание: 0.03333333333333333

Дисперсия: 5.555555555555556E-4

Коэффициент вариации: 0.5

---Экспериментальные значения---

Математическое ожидание: 0.03351103650086813 Дисперсия: 5.652590129633172E-4 Коэффициент вариации: 0.5033519717292653

Максимальное значение: 0.2069661881521162

Макс. знач. / Мат. ожидание: 6.176060479261946

9) Лямбда: 50.0 Кол-во значений ЛКГ: 1000000 Кол-во интервалов ЛКГ:30 Коэффициент k: 4

---Теоретические значения---

Математическое ожидание: 0.02

Дисперсия: 1.0E-4

Коэффициент вариации: 0.25

---Экспериментальные значения---

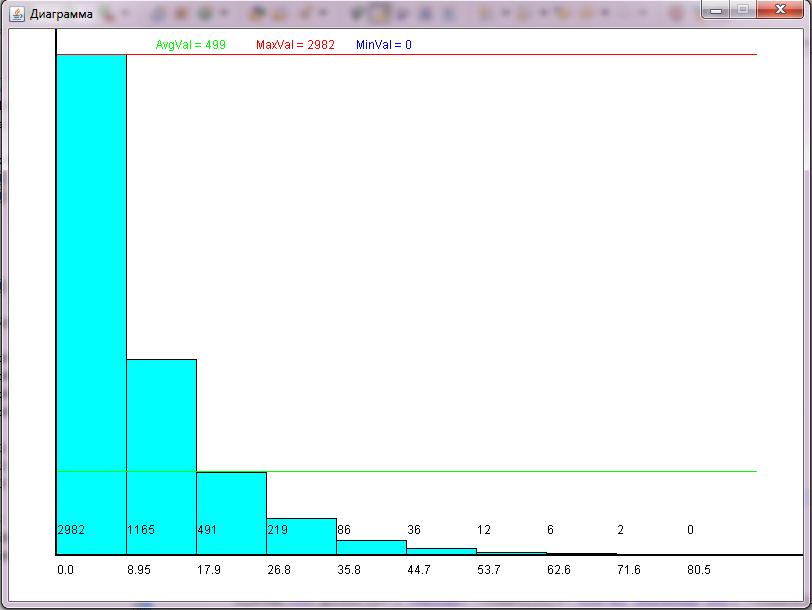
Математическое ожидание: 0.020055328531357212 Дисперсия: 1.0032540693118146E-4 Коэффициент вариации: 0.24943154031944842

Максимальное значение: 0.08802134818394432

Макс. знач. / Мат. ожидание: 4.388925768347292

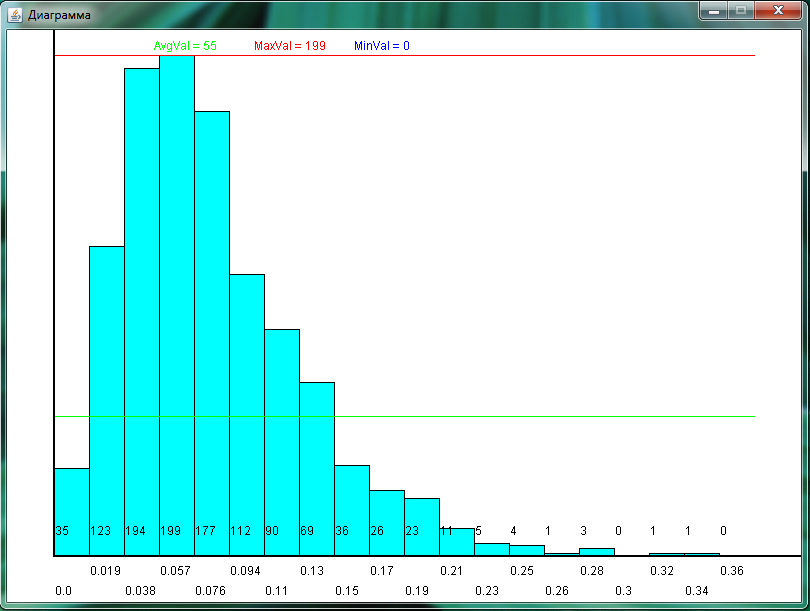
Для генераторов 0,6 и 7 представим гистограммы.(рис. 5.5.1, рис 5.5.2, рис 5.5.3)

№0



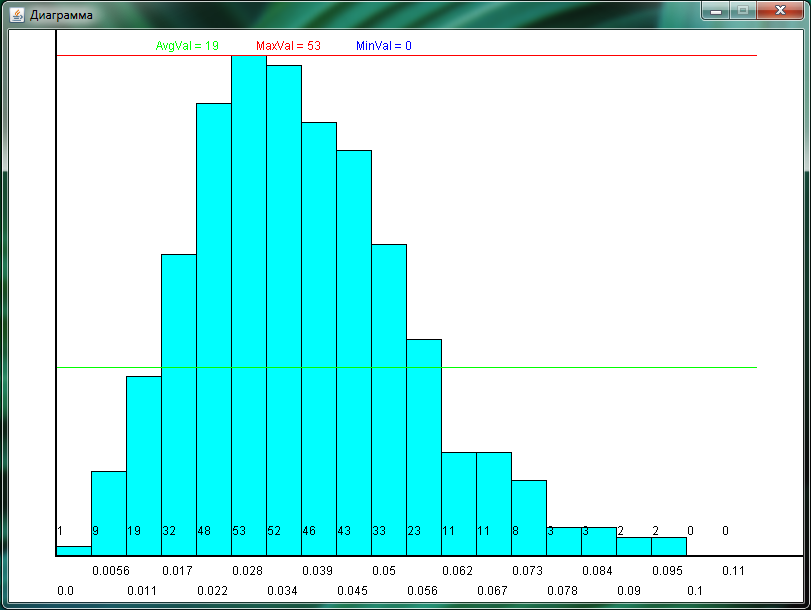
**Рис. 5.5.1** Гистограмма №0 (Распределение Эрланга)

№6



**Рис. 5.5.2** Гистограмма №6(Распределение Эрланга)

№7



**Рис. 5.5.3** Гистограмма №7(Распределение Эрланга)

## 5.6 Распределение гиперэкспоненциальное

***Результаты***

Гиперэкспоненциальный.

---Линейный конгруэнтный генератор---

Выборка: 10000. Кол-во интервалов: 20.

a=93^4 c=91^4 d=19^4

Максимальное отклонение от матожидания: 0.05390539053905388

0) Лямбда: 0.1 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 1

---Теоретические значения---

Математическое ожидание: 10.0

Дисперсия: 99.99999999999999

Коэффициент вариации: 0.9999999999999999

---Экспериментальные значения---

Математическое ожидание: 10.049036941955576 Дисперсия: 102.18158331810388 Коэффициент вариации: 1.0118677218419267

Максимальное значение: 108.63849778625764

Макс. знач. / Мат. ожидание: 10.810836741248583

1) Лямбда: 0.5 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 2

---Теоретические значения---

Математическое ожидание: 2.0

Дисперсия: 8.000000000000002

Коэффициент вариации: 2.0000000000000004

---Экспериментальные значения---

Математическое ожидание: 2.002682898382603 Дисперсия: 8.005151424460347 Коэффициент вариации: 1.9959293887393899

Максимальное значение: 38.31017102244068

Макс. знач. / Мат. ожидание: 19.12942436038204

2) Лямбда: 0.7 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 3

---Теоретические значения---

Математическое ожидание: 1.4285714285714286

Дисперсия: 6.12244897959184

Коэффициент вариации: 3.0000000000000013

---Экспериментальные значения---

Математическое ожидание: 1.4157727072950033 Дисперсия: 5.890243132370914 Коэффициент вариации: 2.9386384027926216

Максимальное значение: 39.48729051575499

Макс. знач. / Мат. ожидание: 27.890981590682028

3) Лямбда: 1.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 4

---Теоретические значения---

Математическое ожидание: 1.0

Дисперсия: 4.000000000000001

Коэффициент вариации: 4.000000000000001

---Экспериментальные значения---

Математическое ожидание: 0.987397547377344 Дисперсия: 3.6209583928910063 Коэффициент вариации: 3.7139790213283725

Максимальное значение: 24.776743536684567

Макс. знач. / Мат. ожидание: 25.092976585261745

4) Лямбда: 4.0 Кол-во значений ЛКГ: 5000 Кол-во интервалов ЛКГ:10 Коэффициент k: 5

---Теоретические значения---

Математическое ожидание: 0.25

Дисперсия: 0.3125

Коэффициент вариации: 5.0

---Экспериментальные значения---

Математическое ожидание: 0.24413757859353108 Дисперсия: 0.2669307339911194 Коэффициент вариации: 4.478466345840957

Максимальное значение: 7.608523790391416

Макс. знач. / Мат. ожидание: 31.16490232361557

5) Лямбда: 8.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 1

---Теоретические значения---

Математическое ожидание: 0.125

Дисперсия: 0.015625

Коэффициент вариации: 1.0

---Экспериментальные значения---

Математическое ожидание: 0.12605606052927934 Дисперсия: 0.015922240550576644 Коэффициент вариации: 1.0020207609135205

Максимальное значение: 1.3579812223282206

Макс. знач. / Мат. ожидание: 10.772835646508238

6) Лямбда: 12.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 3

---Теоретические значения---

Математическое ожидание: 0.08333333333333333

Дисперсия: 0.02083333333333334

Коэффициент вариации: 3.000000000000001

---Экспериментальные значения---

Математическое ожидание: 0.08401020941195231 Дисперсия: 0.020859722667928817 Коэффициент вариации: 2.955591409441238

Максимальное значение: 2.303425280085708

Макс. знач. / Мат. ожидание: 27.418397075891523

7) Лямбда: 25.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:20 Коэффициент k: 5

---Теоретические значения---

Математическое ожидание: 0.04

Дисперсия: 0.0080

Коэффициент вариации: 5.0

---Экспериментальные значения---

Математическое ожидание: 0.03954211643072705 Дисперсия: 0.007368212098662594 Коэффициент вариации: 4.712401632058589

Максимальное значение: 1.7284718409370463

Макс. знач. / Мат. ожидание: 43.71217317022263

8) Лямбда: 30.0 Кол-во значений ЛКГ: 10000 Кол-во интервалов ЛКГ:30 Коэффициент k: 2

---Теоретические значения---

Математическое ожидание: 0.03333333333333333

Дисперсия: 0.0022222222222222227

Коэффициент вариации: 2.0000000000000004

---Экспериментальные значения---

Математическое ожидание: 0.03379503535507337 Дисперсия: 0.002322737946343749 Коэффициент вариации: 2.0337351966669495

Максимальное значение: 0.6385028503740112

Макс. знач. / Мат. ожидание: 18.893391992802222

9) Лямбда: 50.0 Кол-во значений ЛКГ: 1000000 Кол-во интервалов ЛКГ:30 Коэффициент k: 4

---Теоретические значения---

Математическое ожидание: 0.02

Дисперсия: 0.0016000000000000005

Коэффициент вариации: 4.000000000000001

---Экспериментальные значения---

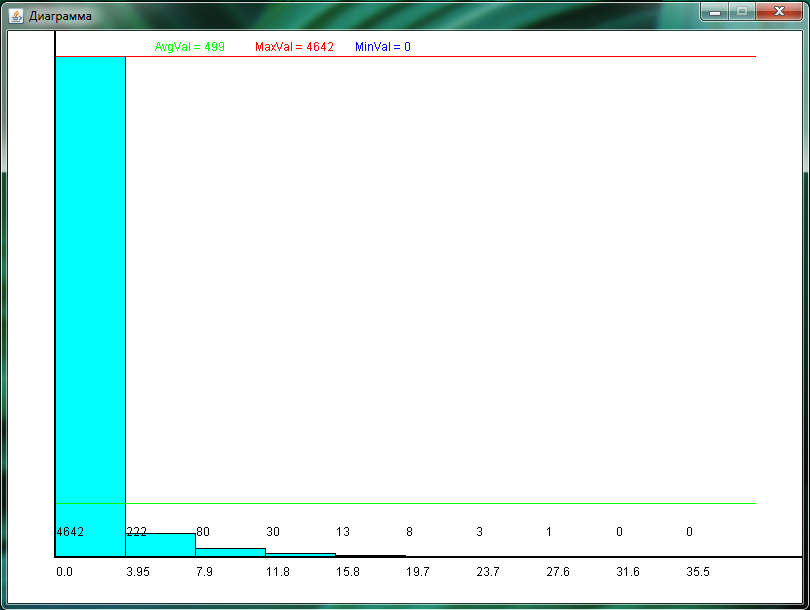
Математическое ожидание: 0.020033037995498763 Дисперсия: 0.0015937594081879708 Коэффициент вариации: 3.9712674122939915

Максимальное значение: 0.9727953789188848

Макс. знач. / Мат. ожидание: 48.559553430561195

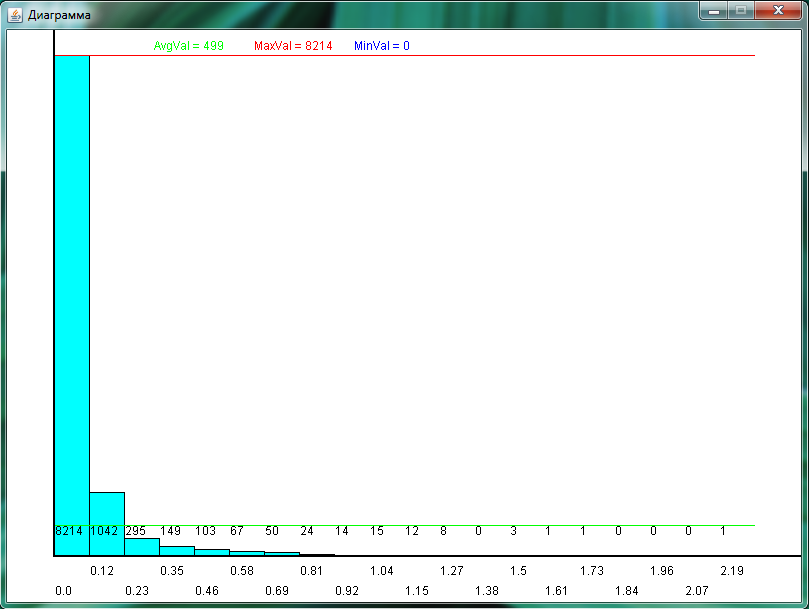
Для генераторов 2,5 и 7 представим гистограммы. (рис. 5.6.1, рис 5.6.2, рис 5.6.3)

№2



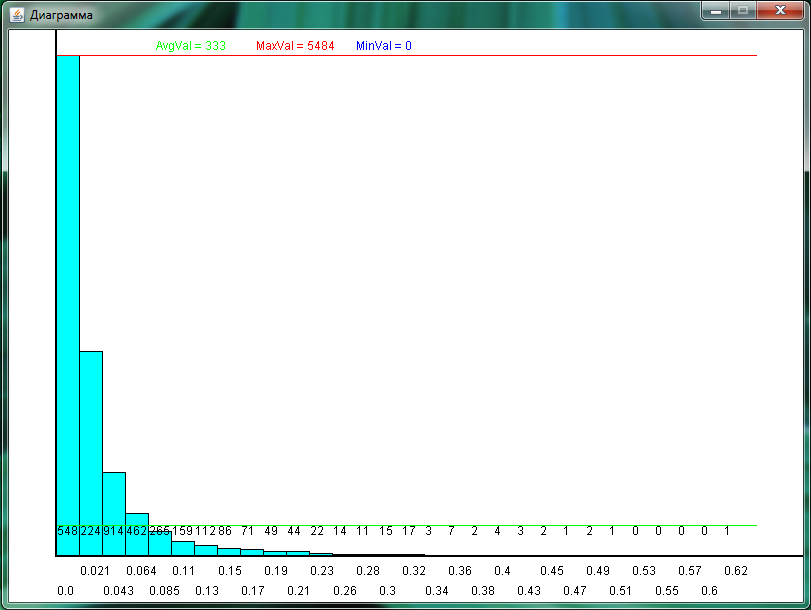
**Рис. 5.6.1** Гистограмма №2(Распределение гиперэкспоненциальное)

№5



**Рис. 5.6.2** Гистограмма №5 (Распределение гиперэкспоненциальное)

№7



**Рис. 5.6.3** Гистограмма №7 (Распределение гиперэкспоненциальное)

# 6.Моделирование процессов стохастических сетей Петри

**Имитационное моделирование** — это метод, позволяющий строить модели, описывающие процессы так, как они проходили бы в действительности. Такую модель можно «проиграть» во времени как для одного испытания, так и заданного их множества. При этом результаты будут определяться случайным характером процессов. По этим данным можно получить достаточно устойчивую статистику.

**Имитационное моделирование** — это метод исследования, при котором изучаемая система заменяется моделью с достаточной точностью описывающей реальную систему и с ней проводятся эксперименты с целью получения информации об этой системе. Экспериментирование с моделью называют имитацией (имитация — это постижение сути явления, не прибегая к экспериментам на реальном объекте).

**Имитационная модель** — логико-математическое описание объекта, которое может быть использовано для экспериментирования на компьютере в целях проектирования, анализа и оценки функционирования объекта.

СМО(Система массового обслуживания) – организованная совокупность приборов для обслуживания заявок.

В режиме выполнения сети обрабатываются два типа событий – запуск, планирование срабатывания переходов с помощью генераторов.

События срабатывания записываются в строгом хронологическом порядке.

Начальное время выполнения принимается равное нулю.

После обработки события срабатывания – маркировка должна быть реальной.

Новое состояние записывается в управляющий список.

На основе обработки событий формируется статистика.

**Статистика при режиме выполнения сети**

1. Частота появления

ω-количество раз попадания в данную маркировку за время выполнения сети.

1. Суммарное время пребывания- T∑ -количество времени пребывания в данной маркировке за всё время моделирования.
2. Суммарное время возвращения -Tr∑ - Суммарное время от начала появления в данной маркировке до последнего момента появления.
3. Вероятность пребывания Pj=T∑/ Tmod Вероятность пребывания в данной маркировке.(∑ Pj=1)
4. Частота испытаний ν= T∑/ ΔT

ΔT- условный шаг проверки (не должен быть слишком малым, но и не должен быть слишком большим)

### 6.1. Моделирование процессов в Petri-nets моделях.

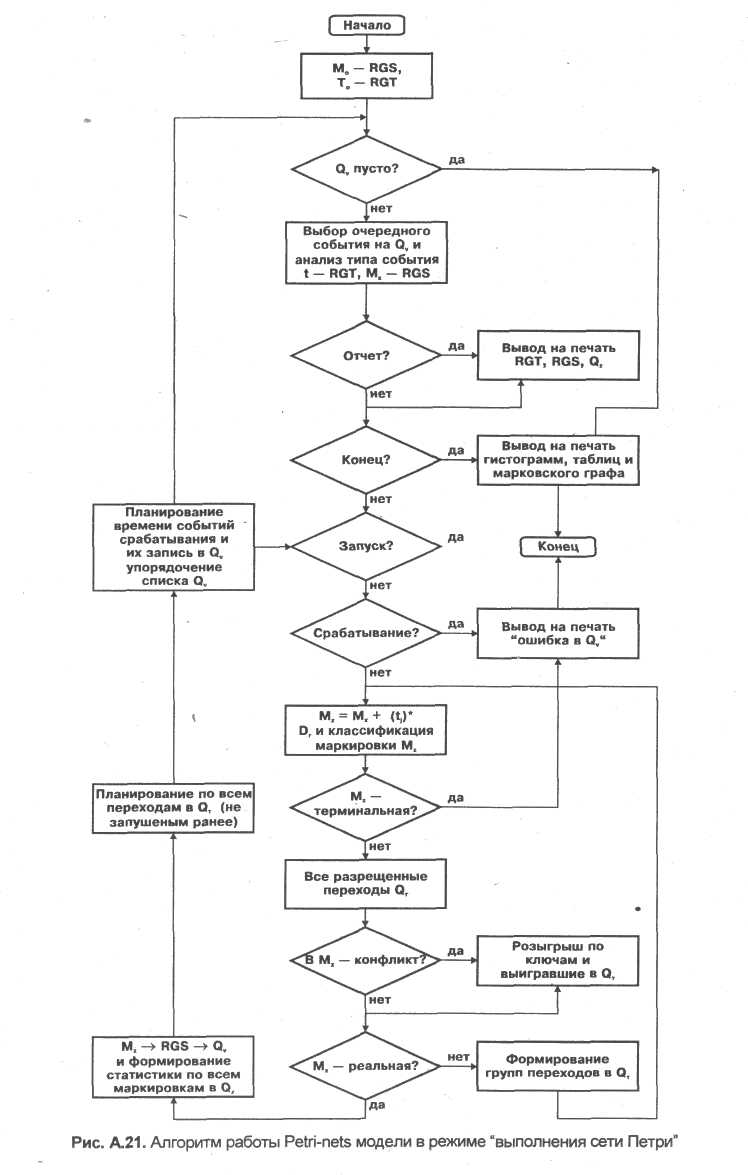
В процессе моделирования реализуется основной режим работы имитационной модели — выполнение сетей Петри (рис. А.21).

Особенностью режима выполнения ССП является то, что при обработке событий запуски временных переходов планируются конкретные времена будущих событий срабатывания этих переходов. Это второй принцип управления имитационными моделями формальных дискретных систем.

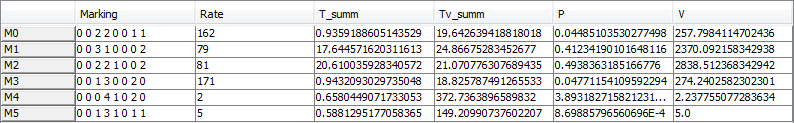
Для планирования интервалов времени между событиями "срабатывания переходов" (обслуживания заявок) используются имитационные модели информационных потоков обслуживания с конкретными заданными по вариантам характеристиками интенсивности потока и коэффициента вариации. Асинхронное развитие процессов реализуется на основе принципа самодостаточности: если переход разрешен, то он запускается и одновременно планируется время его срабатывания.

• Исключение составляют конфликтные переходы. Для разрешения конфликтов в ССП могут использоваться различные средства, например, приоритеты переходов, семафоры и т.п. Однако наиболее удобным механизмом разрешения конфликтов для ССП являются ключи распределения — *Ri,* которые становятся в соответствие конфликтующим переходам и отражают вероятности срабатывания каждого из них в конфликте. В режиме моделирования процессов в случае конфликта производится розыгрыш с учетом вероятностей Ri и запускается только один переход.

Алгоритм работы имитационной модели в режиме выполнения сети Петри представлен на рис. А.21.



Результаты эмуляции. (для 500) рис. 6.1



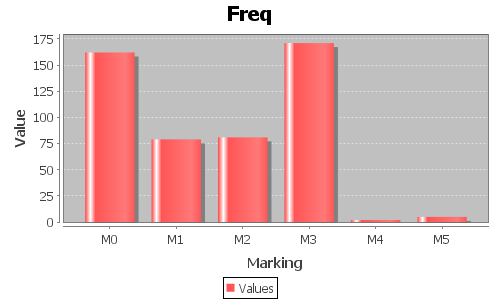
**Рис. 6.1** Статистические данные при эмуляции.

Из рис 6.1 видно, что в маркировках M0 и М5 сеть находилась меньше всего.

**Гистограммы:**

Необходимо построить 4 гистограммы:

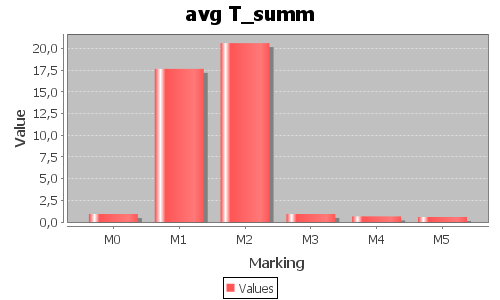
Для ωj- количество раз попадания в данную j-ую маркировку за время выполнения сети



**Рис. 6.2** Гистограмма частоты попадания в маркировку.

Из рис. 6.2 видно, что наиболее редкими были переходы в маркировки М0 и М5.

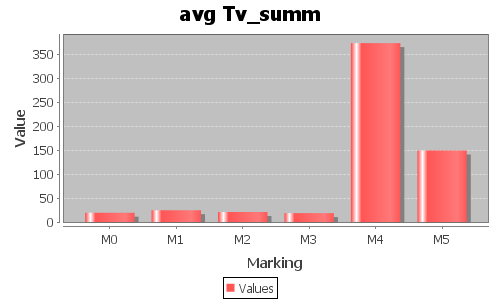
Для Tjср=T∑ j/ ωj – среднее время пребывания в данной j маркировке



**Рис. 6.3** Гистограмма среднего времени пребывания в маркировке.

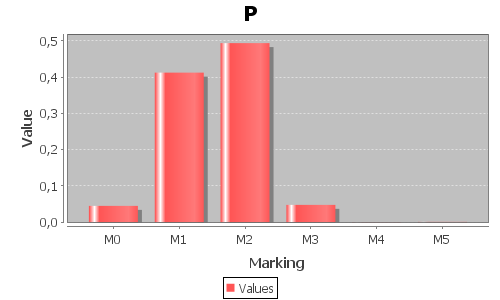
Из рис. 6.3 видно, что наибольшее среднее время пребывания было в маркировке М0.

Для Tj= T∑R/ (ωj-1) – среднее время возвращения в данную j маркировку.

**Рис. 6.4** Гистограмма среднего времени возвращения в маркировку.

Из рис. 6.4 видно, что наибольшее среднее время возвращения было в маркировку М5

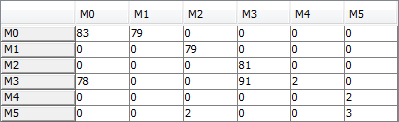
Для pj=T∑ j /Tmod – вероятность пребывания в данной j маркировке.



**Рис. 6.5** Гистограмма вероятности пребывания в маркировке.

Из рис. 6.5 видно, что наибольшая вероятность нахождения в маркировке М6, и достаточно небольшие вероятности нахождения в М0, М2, М3,М7 маркировках

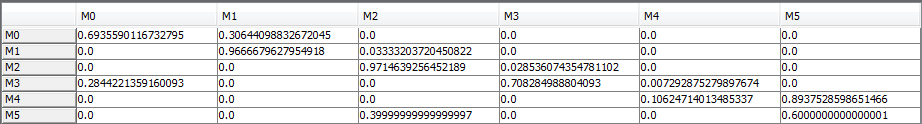
αij – количество раз попадания из i маркировки в j.



**Рис. 6.6** Матрица количества попаданий из одной маркировки в другую.

Из рис. 6.6 видно, что некоторые переходы были более частыми, некоторые очень редки, из-за того что они редки – они представляют особую важность и интерес исследования.

pij= αij/ νi

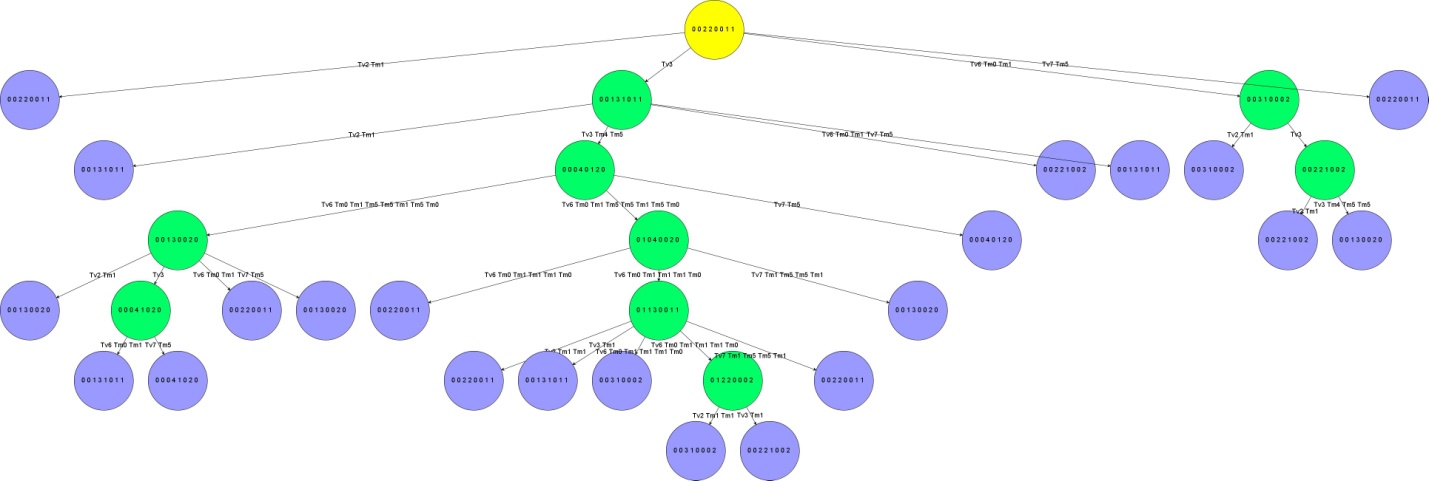


**Рис. 6.7** Матрица вероятности попаданий из одной маркировки в другую.

Из рис. 6.7 видны вероятности перехода из одной маркировки в другую, одни переходы более вероятны другие менее. Суммарная вероятность перехода из маркировки в любую другую равна единице (то есть сумма вероятности по строке равна единице).

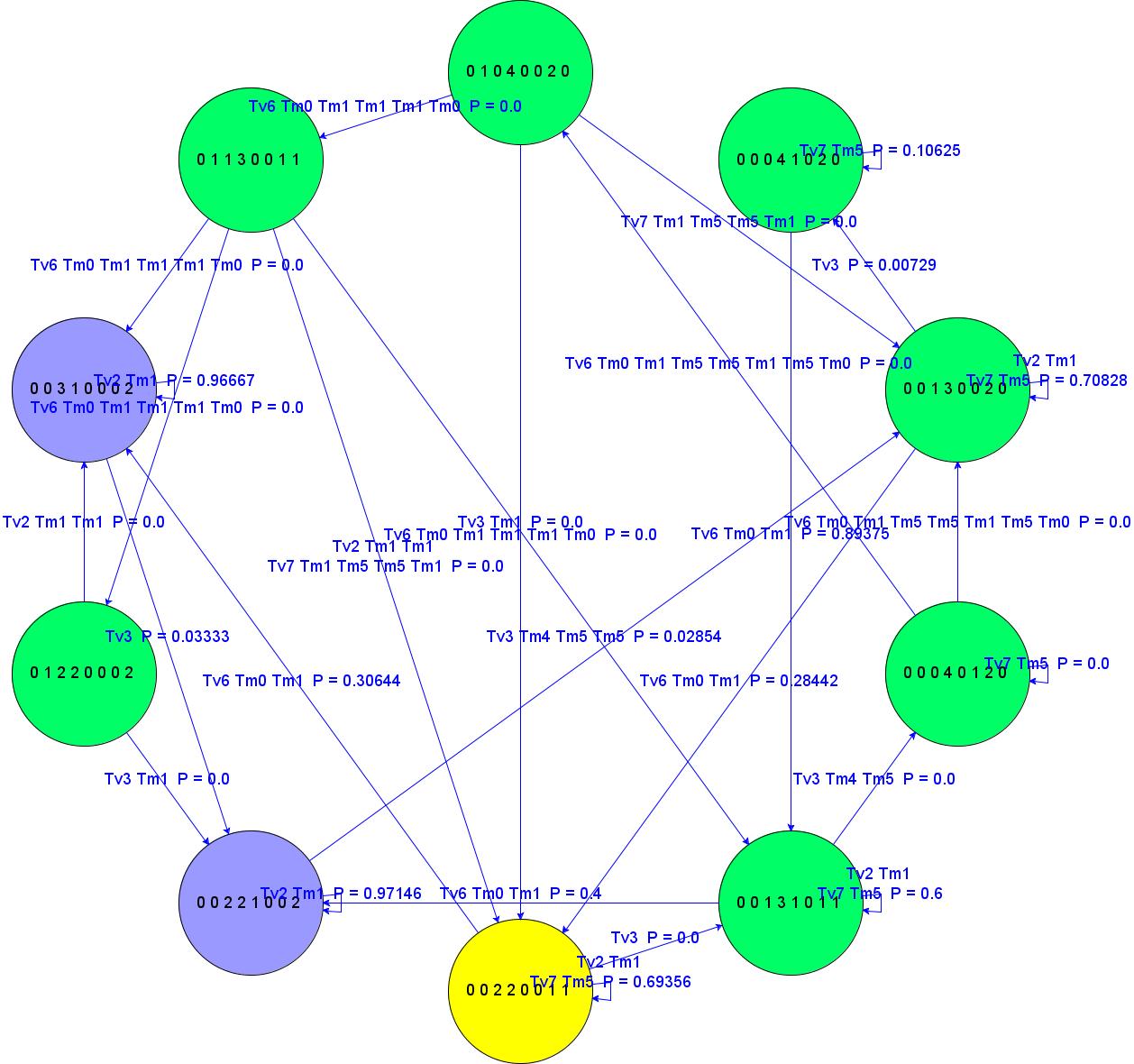
dt- условный шаг проверки. Рассчитывается таким образом, чтоб сумма вероятностей перехода из любой маркировки во все другие не превышала единицу.

Дерево достижимости:



**Рис. 6.9** Дерево достижимости.

Марковский граф с отмеченными вероятностями переходов.



**Рис. 6.9** Марковский граф с расставленными вероятностями переходов по рёбрам.

### 6.2. Инструкция пользователя. Моделирование сетей Петри

Для моделирования необходимо открыть готовую или создать новую модель, выставить её параметры:

* фишки в позициях
* интенсивности, коэффициенты вариации и разрешения конфликтов в переходах

Также можно задать максимальное количество фише в позиции через Modeling - > Enter max count of chips

Чтобы перейти в режим моделирования нужно перейти Modeling -> Modeling mode или или нажав на кнопку resultset_next

Чтобы вернуться обратно в режим редактирования нужно перейти Modeling -> Editing mode или нажав на кнопку pencil_go

После того, как перешли в режим моделирования система автоматически делает базовій анализ построенной модели. Слева в виде позиционного дерева открывает дерево достижимости сети Петри.

Также дерево достижимости можно посмотреть в ярусно-параллельной форме в появившейся вкладке **Tree** на ней отображены возможные маркировки и пути переходов между ними. Маркировка с желтым фоном – стартовая, зеленым – значимая, синим – дублирующая, красным – тупиковая.

Во вкладке **Matrices** находятся матрицы входов-переходов и выходов-переходов

Также сразу формируется Марковский граф, который можно наблюдать во вкладке **Mark tree**. Здесь все маркировки изображены в виде Марковского графа, а на каждой дуге подписана вероятность перехода по ней из маркировки в маркировку.

Моделирование с учетом времени осуществляется во вкладке Modeling. Здесь можно указать либо количество шагов моделирования, либо время моделирования. Нажав кнопку **Start** выполняется моделирование сети Петри в соответствии с заданными параметрами. В поле справа отображается таблица статистики моделирования.

Таблица статистики маркировок содержит поля:

* Маркировка
* Частота появления маркировки
* Суммарное время пребывания в маркировке
* Суммарное время возвращения в маркировку
* Вероятность пребывания в маркировке
* Частота испытаний

Во вкладке Diagrams отображаются гистограммы, которые соответствуют указанной выше статистике

7.Моделирование log-сервера и сервера приложений

**Выбор варианта:**

Параметры структуры

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 буква  фамилии | Вариант структур (А) | Запросов Сессии(U) | Кол-во  путей (S) | Зарег. польз-  ователей (N) | Кол-во  ключей(К) | Время  расчета  (мс) |
| А | 2 | 4 | 1 | 3 | 2 | 0,02 |

**Таблица 7.1** Параметры структуры log сервера

Структура 1 – полная конфигурация.

U-общее число запросов пользователя в сессии принятия решения,

S-количество маршрутов к серверу приложений,

N-количество пользователей, которые имеют право доступа.

K-Количество ключей для реализации сеансов обмена информацией по запросам пользователей.

Tk-время расчёта – генерация сеансовых ключей шифрования для реализации соответствующего алгоритма маскирующих преобразований(DES, RSA).

Первая буква имени – Ю.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | пр |
| 1,6 | 4 | 45 | 120 | 0,3 |

* λ1 – пользователем в диалоговом окне приложения (с коэффициентом вариации в потоке g = 1/5);
* λ2 – маршрутизатором при поиске кратчайшего пути (с коэффициентом вариации в потоке g = 1);
* λ3 – интерфейсными структурами сети(с коэффициентом вариации в потоке g = 1/3);
* λ4 – log- сервером при установлении прав доступа(с коэффициентом вариации в потоке g = 8);
* λ5 – сервером приложений при поиске и обработке информации в соответствии с запросом пользователя(с коэффициентом вариации в потоке g = 1);

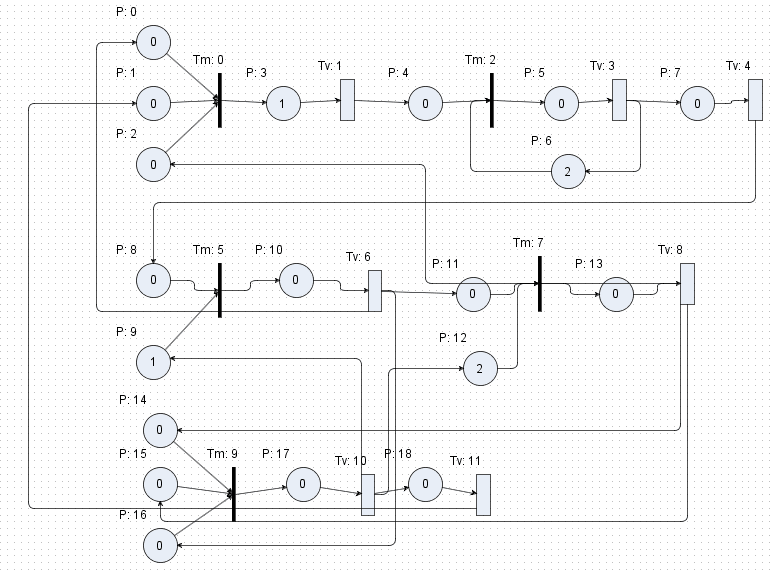
|  |  |  |
| --- | --- | --- |
| **Таблица 3** | Назначения фишек в маркировке позиций сети Петри | |
| Позиция сети | Назначение фишек в маркировке сети | Начальное значение |
| **P1** | Login пользователя, а в дальнейшем право доступа к серверу приложений при каждом сеансе | 1 |
| **P2** | Запросы пользователя к серверу для информационно-аналитической поддержки сессии принятия решения | 1(U=4) |
| **P3** | Пароль пользователя, а в дальнейшем сеансовые ключи шифрования | 1 |
| **P4** | Обработка запроса и предоставление доступа к окну приложения, формирование запроса к серверу | 0 |
| **P5** | Очередь запросов к узлу доступа маршрутизатора сети | 0 |
| **P6** | Обработка запроса на передачу сообщения сети | 0 |
| **P7** | Количество свободных путей передачи сообщений в компьютерной сети | 1 (S=1) |
| **P8** | Транспонировка по сети и сборка пакетов сообщений в процессе удалённого взаимодействия | 0 |
| **P9** | Запрос на предоставление прав доступа к серверу приложений | 0 |
| **P10** | Количество пользователей, которым могут быть предоставлены права доступа в конкретном сеансе | 1(N=3) |
| **P11** | Определение log сервером прав доступа по конкретному запросу. | 0 |
| **P12** | Запрос на предоставление сеансовых ключей шифрования для работы с сервером приложений | 0 |
| **P13** | Количество сеансовых ключей, которые могут быть предоставлены пользователям в конкретном сеансе | 2(K=2) |
| **P14** | Формирование сервером квитанций сеансового ключа в соответствии с правами доступа | 0 |
| **P15** | Право доступа к серверу приложений | 0 |
| **P16** | Удалённый запрос пользователя к серверу приложений | 0 |
| **P17** | Сеансовый ключ шифрования сигнатуры ответа сервера приложений удалённому пользователю. | 0 |
| **P18** | Обработка запроса, поиск и анализ соответствующих данных, формирование сигнатуры сообщения | 0 |
| **P19** | Транспонировка пакетов по сети и сборка сообщений в процессе удалённого взаимодействия | 0 |

**Таблица А.2.** Интерпретации событий и времена задержек между событиями: запуск-срабатывание переходов

|  |  |  |
| --- | --- | --- |
| **Переходы сети** | Содержание событий запуск-срабатывание перехода | **Времена задержек** |
| **T1** | Запуск-срабатывание – открытие для конкретного пользователя диалогового окна приложения. | 0 |
| **T2** | Запуск-начало работы с окном приложения срабатывание – формирование запроса в сеть | Случайное  1/λ1=0.625 |
| **T3** | Запуск-начало приём запроса на передачу его по свободному маршруту | 0 |
| **T4** | Запуск-срабатывание – начало работы по выбору и прокладке маршрута. Передача запроса в сеть | 1/ λ2=0,25 |
| **T5** | Запуск-начало начало передачи запроса в сеть срабатывание – завершение передачи запроса в сеть. | Случайное  1/λ3=0,022 |
| **T6** | Запуск- срабатывание приём запроса для анализа прав доступа к серверу приложений | 0 |
| **T7** | Запуск-срабатывание – начало анализа – рассылка конкретных прав | 1/λ4=  0.00833 |
| **T8** | Срабатывание – приём запроса для формирования сеансового ключа шифрования | 0 |
| **T9** | Запуск-расчёт ключа , срабатывание – рассылка конкретного сеансового ключа | Tk=0.005 |
| **T10** | Запуск-срабатывание приём запроса для поиска информации в БД и последующей обработки её сервером приложений | 0 |
| **T11** | Запуск-начало начало работы информационно-аналитического приложения, срабатывание – передача пользователю сообщения | 1/λпр=3,33 |
| **T12** | Запуск-начало начало передачи запроса в сеть срабатывание – завершение передачи запроса в сеть. | 1/ λ3=0,022 |

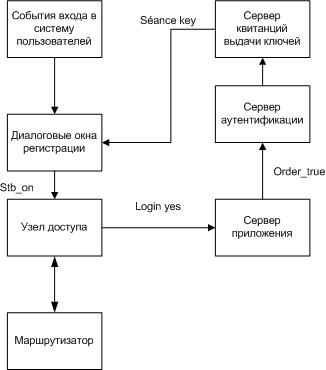
Общая структура информационно – аналитической системы

на база log-сервера и сервера приложений рис. 7.1



**Рис. 7.1** Общая структура заданной Сети Петри.

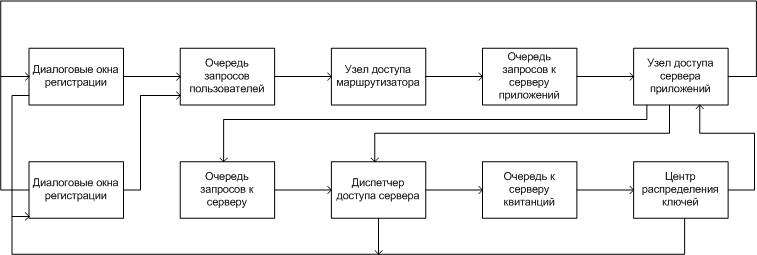
Организация потока запросов в моделирующей системе инициализируется входом в систему. Запросы входов в систему передаются от диалоговых окон верхнего уровня к узлу доступа маршрутизатора. Затем они передаются от узла доступа события к модулю-маршрутизатору, который считает статистику событий маршрутизации. Запросы также передаются защищённому серверу приложения, который инициализирует аутентификацию пользователя, после чего регистрируется вход в систему, а пользователю предоставляется определение права доступа. В зависимости от того, удачно ли произошёл вход в систему, инициализируется сервер квитанций, который выдаёт сеансовые ключи для осуществления передач информации между пользователем и сервером приложений. Рис. 7.1.1



**Рис. 7.1.1** Структура потока login действий в системе Kerberos для управления доступом к сетевым ресурсам.

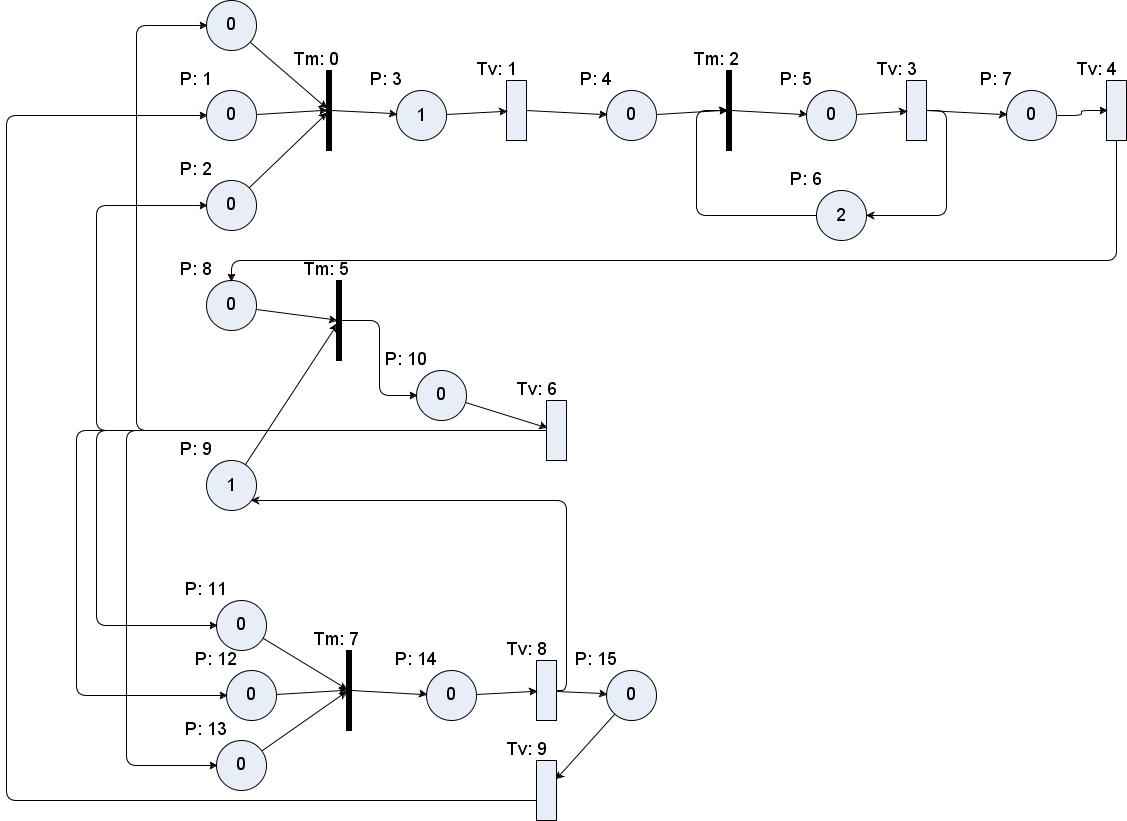
Предоставление услуг пользователям в распределённой информационно-аналитической системе с защищаемым сервером приложений как системе массового обслуживания со многими приборами и очередями. Рис 7.1.2

Данная система массового обслуживания запросов пользователей к серверу приложений через соответствующую систему предоставления прав доступа и ключей шифрования существенно упрощена. Однако она даёт представление о характере удалённого взаимодействия и предоставления прав доступа и ключей к ресурсам по схеме обмена сообщениями, принятой в системе Kerberos, оперирующей с конфиденциальной информацией, в процессе группового проектирования и разработки программ искусственного интеллекта.



**Рис 7.1.2** Формальная структура исходной системы.

**Структура по варианту рис.7.2:**

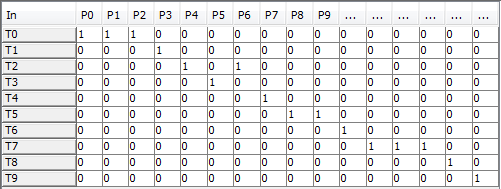
****

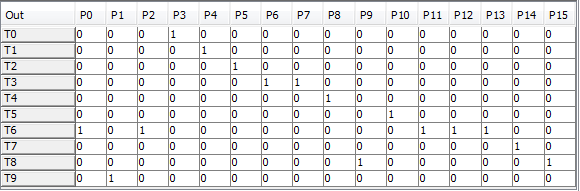
**Рис. 7.2** Структура заданной Сети Петри в соответствии с вариантом.

Login пользователя в сети передаётся в открытом виде, а пароль и вся служебная информация шифруются с помощью ключа, который формируется на основе пароля пользователя с помощью секретной функции. Администратору известна функция, но не известен пароль, пользователю известен пароль но не известна секретная функция, злоумышленнику неизвестны ни пароль, ни функция, серверу приложений известны и то, и другое, в частности, пароль был определён при регистрации пользователя на сервере приложений.

Информация, поступающая к пользователю, шифруется сеансовым ключом, но чаще сама информация не шифруется, а формируется дополнительная вставка – сигнатура или цифровая подпись, которая обеспечивает проверку целостности информации в окне приложения при наличии у пользователя индивидуального сеансового ключа.

На рис. 7.3 приведены матрицы DI и DQ.

****

****

**Рис. 7.3** Параметры сети.

**Граф дерева достижимости**

****

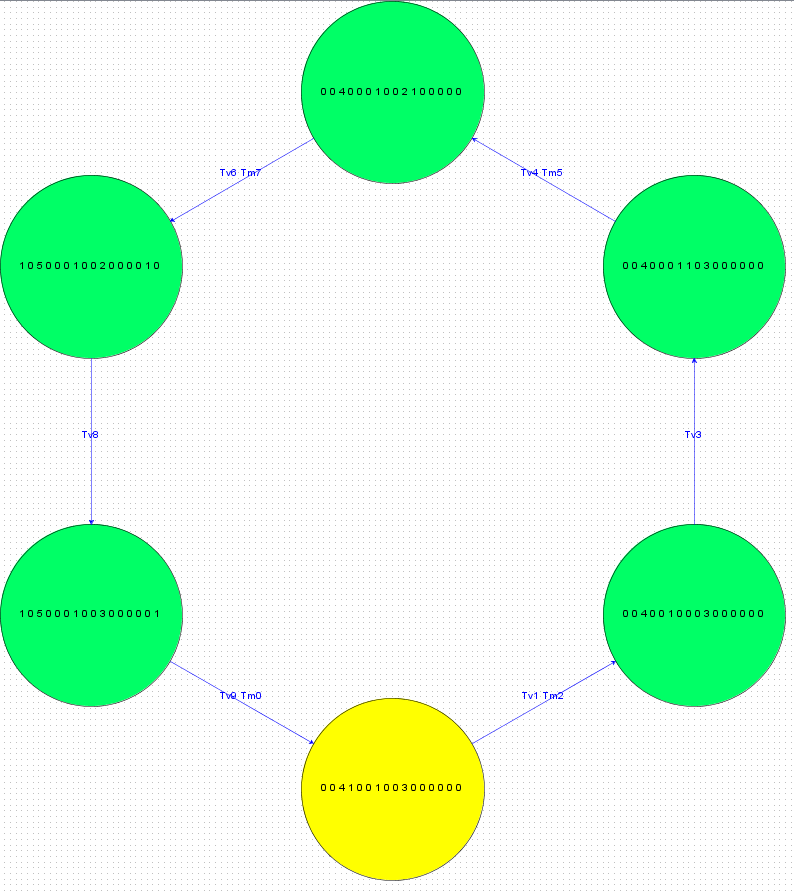
**Рис. 7.5** Граф достижимости.

В данном примере свойства сети Петри:

* сеть небезопасна, т.к. уже в начальной маркировке в позициях сети бывает больше 1 фишки.
* сеть является ограниченной – максимальное количество фишек в одной вершине - 2.
* сеть не строго сохраняема, т.к. не всегда выполняется равенство 
* сеть активна, т.к. нет тупиковых состояний
* сеть достижима для всех M'
* сеть конфликтная, т.к. имеет одновременно разрешенные временные переходы

Следует отметить, что количество различных маркировок - 6.

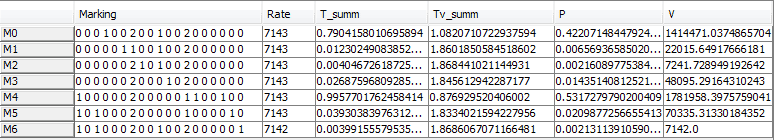
**Марковский граф**

****

**Рис. 7.6** Марковский граф сети.

### 7.1 Статистические результаты эмуляции работы log-сервера

Результаты эмуляции. (для 50К)

****

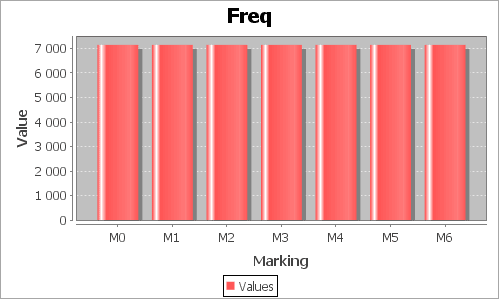
**Рис. 7.1.1** Статистические результаты при эмуляции.

Из рис 7.1.1 видно, что в маркировках М10-М25 сеть не находилась, а также некоторые Маркировки встречались более часто, чем другие при эмуляции.

**Гистограммы:**

Необходимо построить 4 гистограммы:

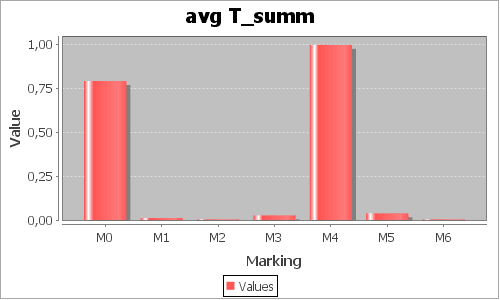
Для ωj- количество раз попадания в данную j-ую маркировку за время выполнения сети



**Рис. 7.1.2** Гистограмма частоты попадания в маркировку.

Из рис 7.1.2 видно, что частота попадания во все маркировки одинаковая.

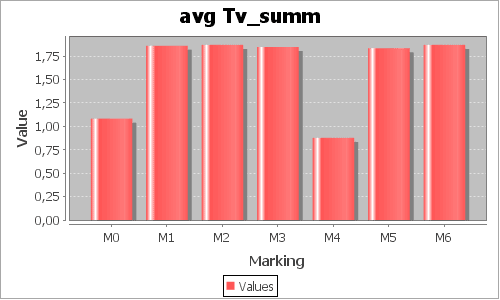
Для Tjср=T∑ j/ ωj – среднее время пребывания в данной j маркировке



**Рис. 7.1.3** Гистограмма среднего времени пребывания в маркировке.

Из рис. 7.1.3 видно, что наибольшее среднее время пребывания было в маркировке М4, так же достаточно долгое среднее время пребывания в М0.

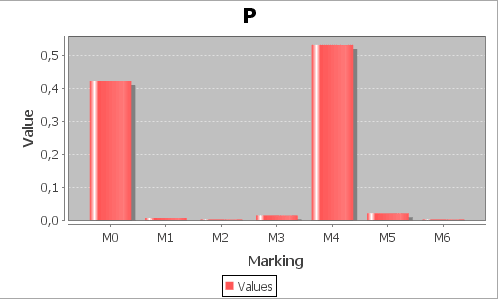
Для Tj= T∑R/ (ωj-1) – среднее время возвращения в данную j маркировку.



**Рис. 7.1.4** Гистограмма среднего времени возвращения в маркировку.

Из рис. 7.1.4 видно, что наименшее среднее время возвращения было в маркировки М0 и М4.

Для pj=T∑ j /Tmod – вероятность пребывания в данной j маркировке.

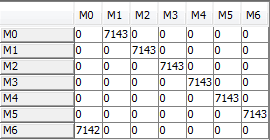


**Рис. 7.1.5** Гистограмма вероятности пребывания в маркировке.

Из рис. 7.1.5 видно, что наибольшая вероятность пребывания в маркировках М4 и М0.

Каждая из гистограмм представляет разные характеристики, и в зависимости характера исследования используется одна или несколько из них.

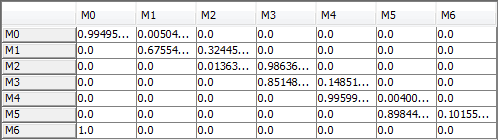
αij – количество раз попадания из i маркировки в j.



**Рис. 7.1.6** Матрица количества попаданий из одной маркировки в другую.

Из рис. 7.1.6 видно, что многие маркировки не достигаются и не проявляются, и происходят лишь некоторые переходы из возможных по графу достижимости, что связано с параметрами схемы.

pij= αij/ νi



**Рис. 7.1.7** Матрица вероятности попаданий из одной маркировки в другую

Из рис. 7.1.7 видны вероятности перехода из одной маркировки в другую, одни переходы более вероятны другие менее. Суммарная вероятность перехода из маркировки в любую другую равна единице (то есть сумма вероятности по строке равна единице).

dt- условный шаг проверки. Рассчитывается таким образом, чтоб сумма вероятностей перехода из любой маркировки во все другие не превышала единицу.

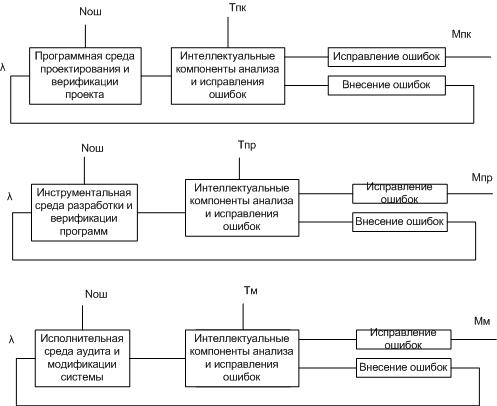
# 8. Общая структура подсистемы анализа безопасности ИАС на этапах проектирования

# зачетной книжки – 7301, Ашаев Юрий Николаевич

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3Буква  фамилии | Модель безопасности | Темп испытаний | Поток исправления ошибок | Поток внесения ошибок | Поток исправления ошибок | Поток внесения ошибок |
| А | ММ | 2 | 2\*10^(-2)  B=0.25 | 6\*10^(-3)  T=300 | 3\*10^(-3)  G=1/2 | 9\*10^(-4)  T=1000 |

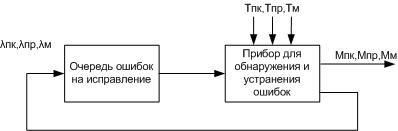
|  |  |
| --- | --- |
| Поток исправления ошибок | Поток внесения ошибок |
| 5\*10^(-4)  G=4 | 4\*10^(-4)  T=5000 |

Исходная структура организации процессов диагностики и исправления ошибок на трёх основных этапах: проектирование, разработка и модификация может быть рассмотрена с позиций теории массового обслуживания как многоканальная СМО с отказами и обратной связью по отказам, выполняющая с помощью диагностического прибора обслуживание заявок на поиск и исправление ошибок рис 9.1



**Рис 8.1** Организация аудита и исправления ошибок.

Наиболее эффективный способ – имитационное моделирование случайного процесса, который порождается в многоканальной СМО с обратной связью и различными характеристиками потоков ошибок при разной длительности обслуживания в каждом канале.



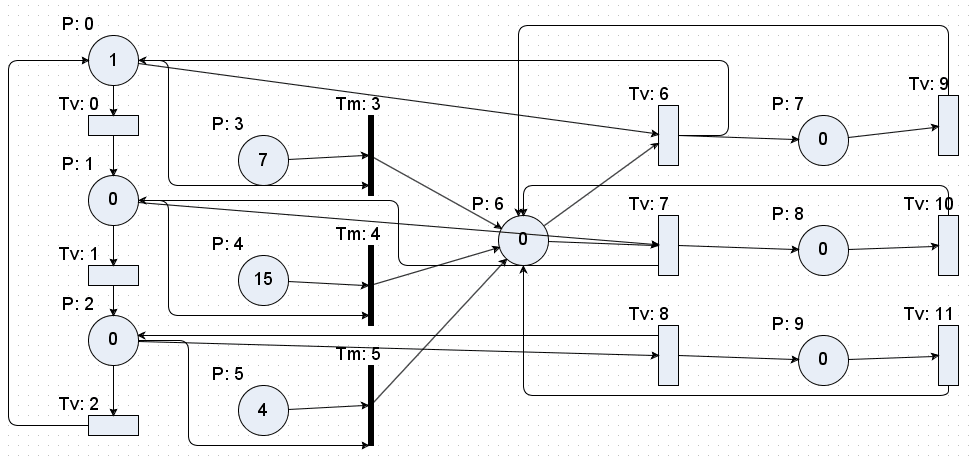
**Рис. 8.2** Структура трёхканальной СМО с отказами для анализа и исправления ошибок при разработке программного обеспечения.

|  |  |  |
| --- | --- | --- |
| **Таблица 3** | Назначения фишек в маркировке позиций сети Петри | |
| Позиция сети | Назначение фишек в маркировке сети | Начальное значение |
| **P1** | Признак этапа проектирования ПО | 1 |
| **P2** | Признак этапа кодирования ПО | 0 |
| **P3** | Признак этапа модификации ПО | 0 |
| **P4** | Начальное количество ошибок в проекте. | 7 |
| **P5** | Дополнительные ошибки в исходном коде | 15 |
| **P6** | Дополнительные ошибки при модификации | 4 |
| **P7** | Динамика количества ошибок | 0 |
| **P8** | Количество исправленных ошибок в проекте | 0 |
| **P9** | Количество исправленных ошибок в программе | 0 |
| **P10** | Количество исправленных ошибок при моделе | 0 |

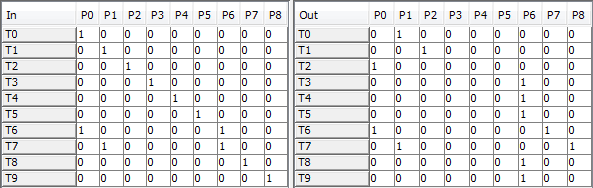
**Таблица А.2.** Интерпретации событий и времена задержек между событиями: запуск-срабатывание переходов

|  |  |  |
| --- | --- | --- |
| **Переходы сети** | Содержание событий запуск-срабатывание перехода | **Времена задержек** |
| **T1** | Запуск-срабатывание – начало моделирования, конец проектирования | 300 |
| **T2** | Запуск-начало начало кодирования, конец программирования | 1000 |
| **T3** | Запуск-начало начало эксплуатации, конец модификации | 5000 |
| **T4** | Запуск-срабатывание – передача планируемой группы ошибок проекта | 0 |
| **T5** | Запуск-начало передача планируемой группы ошибок программы | 0 |
| **T6** | Запуск- срабатывание передача планируемой группы ошибок модификации | 0 |
| **T7** | Запуск-срабатывание –исправление очередной ошибки проекта. | 1/μпк |
| **T8** | Срабатывание – исправление очередной ошибки программ | 1/μпр |
| **T9** | Срабатывание – исправление очередной ошибки модификации | 1/μм |
| **T10** | Запуск-срабатывание –внесение очередной ошибки в проект. | 1/λпк |
| **T11** | Запуск-срабатывание –внесение очередной ошибки в программу. | 1/λпр |
| **T12** | Запуск-срабатывание –исправление очередной ошибки при модификации. | 1/ λм |

Исходная сеть:

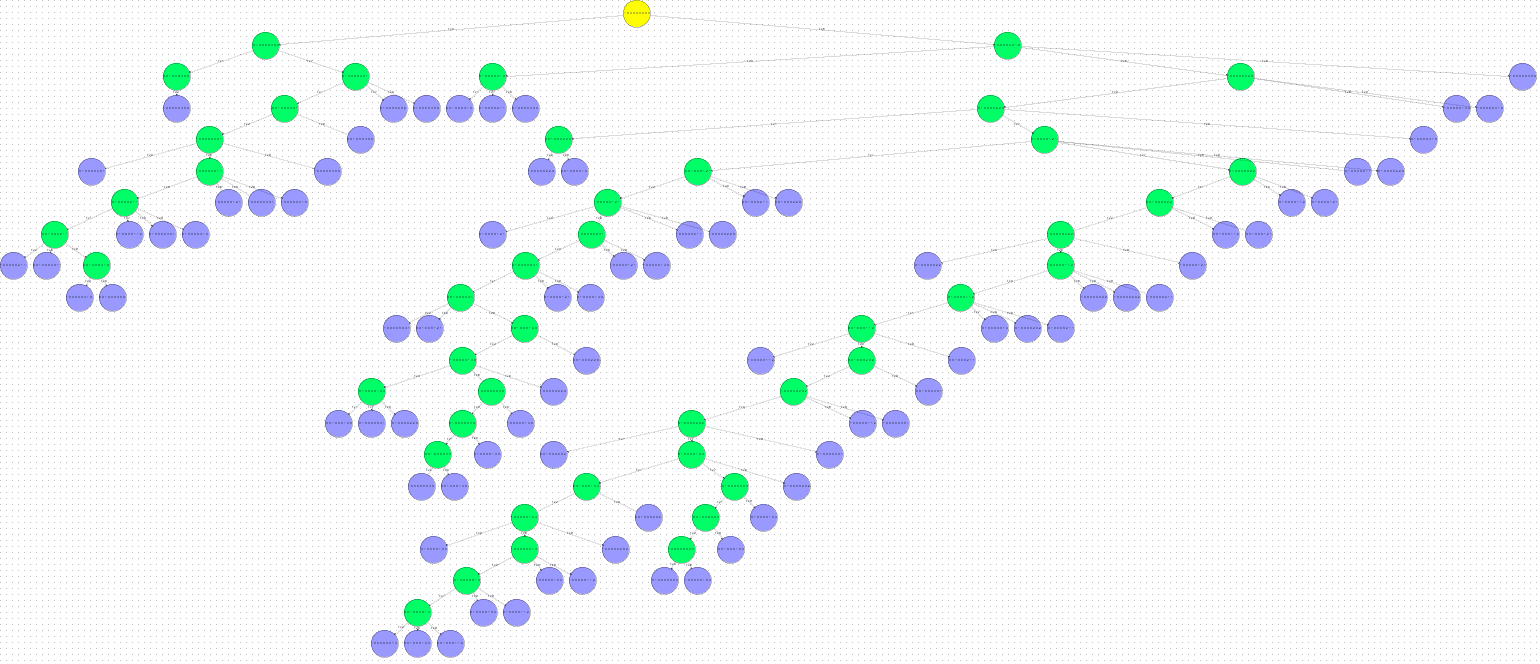


**Рис 8.3** Сеть Петри.



**Рис 8.4** Характеристика Сети(DI, DQ)

**Граф дерева достижимости**

****

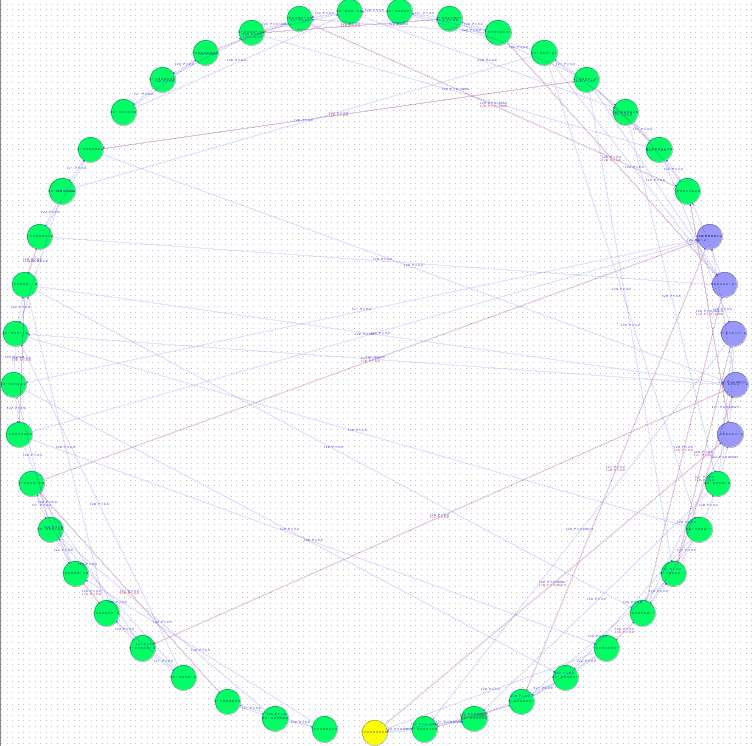
**Рис. 8.6** Граф достижимости.

В данном примере свойства сети Петри:

* сеть небезопасна, т.к. уже в начальной маркировке в позициях сети бывает больше 1 фишки.
* сеть является ограниченной – максимальное количество фишек в одной вершине - 4.
* сеть не строго сохраняема, т.к. не всегда выполняется равенство 
* сеть активна, т.к. нет тупиковых состояний
* сеть достижима для всех M'
* сеть конфликтная, т.к. имеет одновременно разрешенные временные переходы

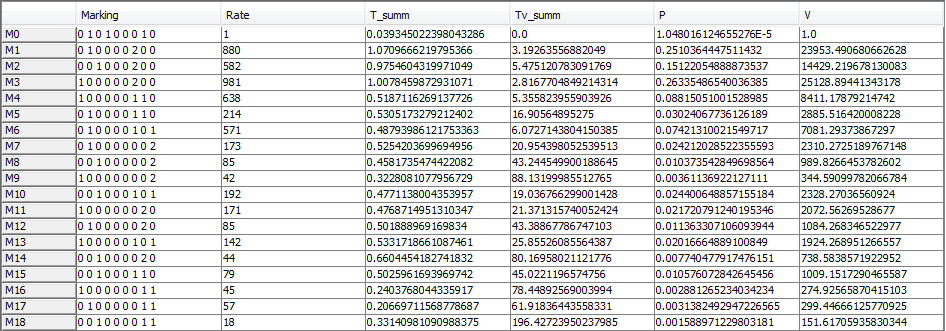
Следует отметить, что количество различных маркировок - 10.

**Марковский граф**

****

**Рис. 8.7** Марковский граф сети.

Результаты эмуляции

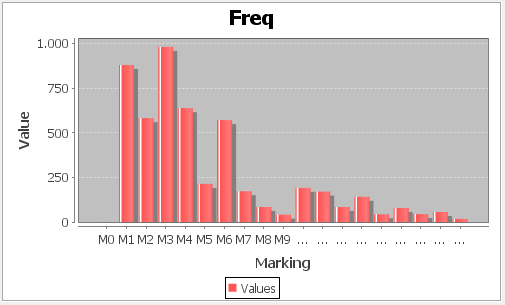
****

**Рис. 8.9** Статистические результаты при эмуляции.

**Гистограммы:**

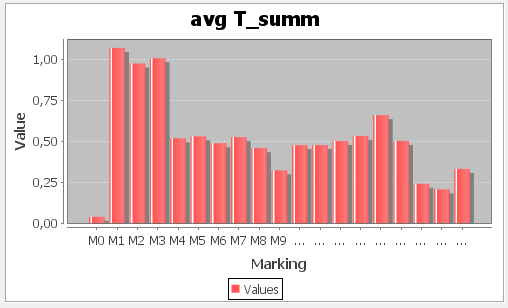
Необходимо построить 4 гистограммы:

Для ωj- количество раз попадания в данную j-ую маркировку за время выполнения сети



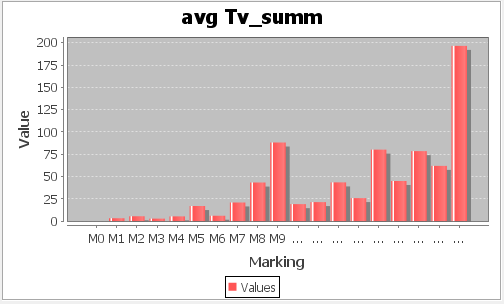
**Рис. 8.10** Гистограмма частоты попадания в маркировку.

Для Tjср=T∑ j/ ωj – среднее время пребывания в данной j маркировке



**Рис. 8.11** Гистограмма среднего времени пребывания в маркировке.

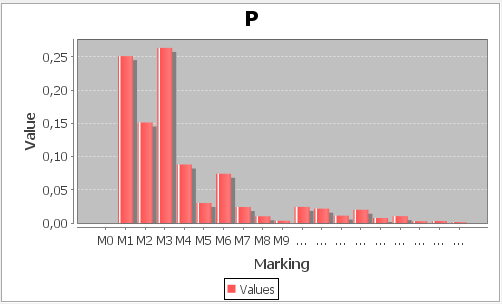
Для Tj= T∑R/ (ωj-1) – среднее время возвращения в данную j маркировку.



**Рис. 8.12** Гистограмма среднего времени возвращения в маркировку.

Из рис. 8.12 видно, что наибольшее время возврата – в М10.

Для pj=T∑ j /Tmod – вероятность пребывания в данной j маркировке.

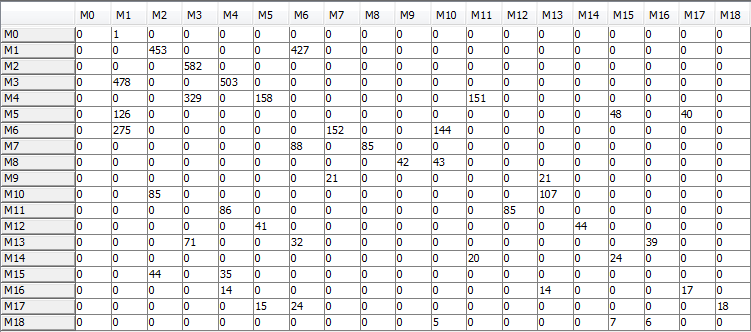


**Рис. 8.13** Гистограмма вероятности пребывания в маркировке.

Из рис. 8.13 видно, что наибольшая вероятность пребывания в маркировке М0.

Каждая из гистограмм представляет разные характеристики, и в зависимости характера исследования используется одна или несколько из них.

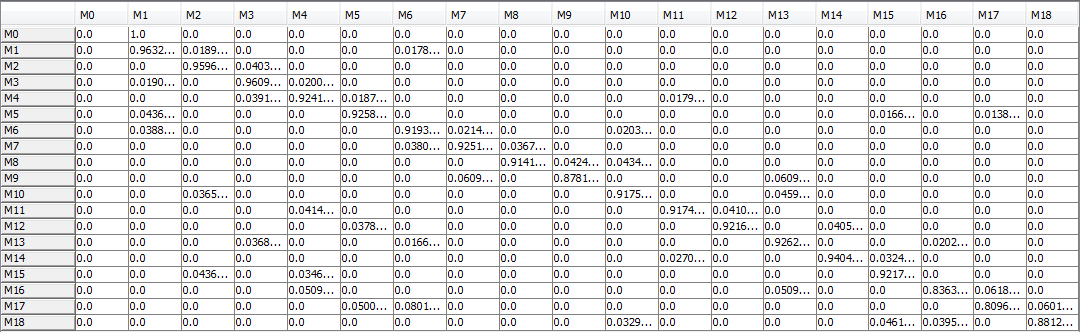
αij – количество раз попадания из i маркировки в j.



**Рис. 8.14** Матрица количества попаданий из одной маркировки в другую.

Из рис. 8.14 видно, что многие маркировки не достигаются и не проявляются, и происходят лишь некоторые переходы из возможных по графу достижимости, что связано с параметрами схемы.

pij= αij/ νi



**Рис. 8.15** Матрица вероятности попаданий из одной маркировки в другую

Из рис. 8.15 видны вероятности перехода из одной маркировки в другую, одни переходы более вероятны другие менее. Суммарная вероятность перехода из маркировки в любую другую равна единице (то есть сумма вероятности по строке равна единице).

dt- условный шаг проверки. Рассчитывается таким образом, чтоб сумма вероятностей перехода из любой маркировки во все другие не превышала единицу.

# Вывод

В результате выполнения данной расчетно-графической работы, мы разработали программное обеспечение для построения, редактирования и моделирования сетей Петри. Разработанная программа объединяет в себе удобный пользовательский интерфейс и богатый потенциал. Среди приемуществ стоит выделить удобный редактор сетей Петри с возможностью сохранения в файл, построение дерев достижимости и марковских графов СМО, описаных сетью, а также пошаговое моделирование с возможностью получения подробной статистики по каждой маркировке с ее последующей визуализацией. Были разработаны и оптимизированы уникальные алгоритмы для работы с сетями. Программа является мультиплатформенной и контекстно-независимой. Разработанное программное обеспечение может быть использовано как в учебных, так и в научных целях. Исходные коды находятся в открытом доступе и распространяются по лицензии GNU.