МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»
Факультет прикладной информатики

МЕТОДЫ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ СЛУЧАЙНЫХ ПРОЦЕССОВ

КУРС ЛЕКЦИЙ

Учебно-методическое пособие для аспирантов
по направлению **38.06.01 Экономика**

по профилю **Математические и инструментальные методы экономики**

Краснодар 2015

Лекция 1

Понятие потока случайных событий.

Потоком событий называется последовательность событий, происходящих одно за другим в случайные моменты времени.

Поток событий называется *однородным,* если он характеризуется только моментами поступления этих событий (вызывающими моментами) и задается следующей последовательностью:

$$\{t_n\} = \{0 \le t_1 < t_2 \ldots < t_n < \ldots\},$$

где t_n – момент поступления n-ого события.

Поток событий *неоднородный,* если он задается не только вызывающими моментами, но и функцией $f(n)$ – набор признаков события (наличие приоритета, принадлежащего к какому-либо типу заявки, класса и т.д.).

Если интервалы времени между событиями независимы между собой и являются случайными величинами, то такой поток называется *потоком с ограниченным последействием.*

Поток событий называется *ординарным,* если вероятность того, что на малый интервал времени Δt, примыкающий к моменту времени t, попадает более одного события, пренебрежимо мала по сравнению с вероятностью того, что на этот же интервал времени попадает ровно одно событие.

Поток *стационарный,* если вероятность появления того или иного события на некотором интервале времени зависит лишь от длины этого интервала и не зависит от того, где на оси времени взят этот интервал.

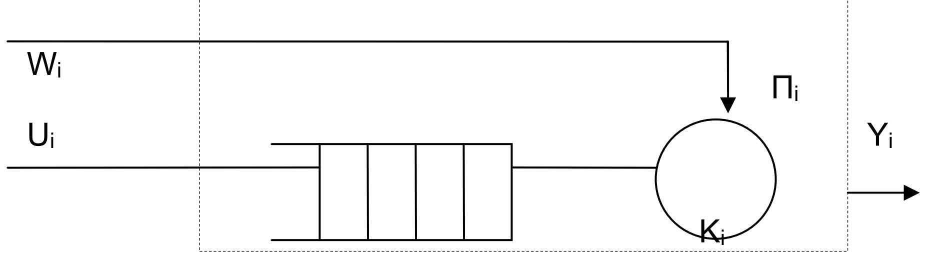
Для ординарного потока среднее число сообщений за интервал времени Δt равно $P_1(t, \Delta t)$, тогда

$$\lim_{\Delta t \to 0} \frac{P_1(t, \Delta t)}{\Delta t} = \lambda$$ - *интенсивность ординарного потока.*

Для стационарного потока *интенсивность* не зависит от времени и представляет собой постоянное значение равное среднему числу событий, наступивших в единицу времени.

В любом элементарном акте обслуживания можно выделить 2 основные составляющие:

1. собственно, обслуживание
2. ожидание обслуживания заявки



К – канал, Н – накопитель, П – прибор обслуживания

В i-ом приборе обслуживания имеем:

- w_i – поток заявок т.е. *интервалы времени* между моментами появления заявок (вызывающие моменты) на входе канала Ki.
- u_i – поток обслуживания – интервалы времени между началом и окончанием обслуживания заявок.

Заявки, обслуженные каналом, и заявки, покинувшие i-ый прибор не обслуженными, образуют *выходной поток Yi,* т.е. *интервалы времени* между моментами выхода заявок.

Процесс функционирования прибора Пi можно представить как процесс изменения состояний его элементов во времени. Переход в новое состояние для прибора означает *изменение количества заявок,* которые в нем находятся.

$$Z^i = (Z^i_H, Z^i_K) \quad Z_H = \{0; L^H_i\} \quad Z_H = \begin{cases} 0 - канал\_свободен \\ 1 - канал\_занят \end{cases}$$

где Z – заявка, L – емкость накопителя

В практике моделирование элементарные Q-схемы обычно объединяются. При этом, если каналы различных приборов обслуживания соединены параллельно, то имеет место *многоканальное* обслуживание, а если последовательно – то *многофазное*. Следовательно, для задания Q-схемы необходимо использовать некоторый оператор - R-оператор сопряжения, отражающий взаимосвязь элементов структуры между собой.

Различают разомкнутые и замкнутые Q-схемы. В разомкнутых схемах выходной поток не может попасть на вход, а в замкнутых схемах количество заявок постоянно.

Собственными внутренними параметрами Q-схемы будут являться:

- Количество фаз
- Количество каналов в каждой фазе
- Количество накопителей в каждой фазе
- Емкость i-ого накопителя

В зависимости от емкости накопителя в теории массового обслуживания принято:

- **Емкость 0** – система с потерями.
- **Емкость** $\to \infty$ - система с ожиданием
- **Емкость конечна** – система смешанного типа

Для задания функционирования Q-схемы также необходимо описать алгоритм её функционирования, который определяет набор правил поведения заявок в различных ситуациях.

Неоднородность заявок, отражающая реальный процесс учитывается введением классов приоритетов. Следовательно, Q-схема, описывающая процесс функционирования системы массового обслуживания любой сложности, однозначно задается:

Q = (W, U, R, H, Z, A)

1. **W** - подмножеством входных потоков
2. **U** - подмножеством потоков обслуживания
3. **H** - подмножеством собственных параметров
4. **R** - оператором сопряжения элементов структуры
5. **Z** - множеством состояний элементов системы
6. **A** - оператором алгоритмов обслуживания заявок

Для получения соотношений, связывающих характеристики описания функционирования Q-схемы вводятся некоторые допущения относительно входных потоков, функций распределения параметров, длительности обслуживания запросов, дисциплин обслуживания.

Для математического описания функционирования устройств, развивающегося в форме случайного процесса, может быть использован математический аппарат, который носит название Марковского случайного процесса.

Лекция 2.

Марковские случайные процессы

Случайный процесс, протекающий в некоторой системе называется *Марковским случайным процессом,* если он обладает следующим свойством: для каждого момента времени t_0 вероятность любого состояния системы в будущем (при $t > t_0$) зависит только от состояния системы в настоящем и не зависит от того, когда и каким образом система пришла в это состояние.

В марковском случайном процессе будущее развитие зависит только от настоящего состояния и не зависит от предыстории. Для марковских случайных процессов определяется вероятности нахождения системы в том или ином состоянии используя уравнения Колмогорова:

$F = (p'(t), p(t), \Lambda)$,

где $p(t)$ – вероятность попадания в какое-либо состояние

λ - множество определенных коэффициентов

Для стационарного потока:

$\Phi = (p(t), \Lambda) = 0$

$Y = Y(P(\Lambda))$ - базисная модель

$Y = Y(V, X, H)$ - интерфейсная модель

Математическая модель некоторой сложной системы строится как совокупность базисной модели и интерфейсной модели, что позволяет использовать одни и те же базисные модели для разных задач проектирования, осуществляя настройку на соответствующую конкретную задачу, посредством изменения только параметров интерфейсной модели.

Математическая модель сложной Q-схемы должна обеспечивать вычисление времени реакции на запрос и производительность системы.

Методика вывода уравнений Колмогорова

$S = \{S_i\}, i = \overline{1,4}$

$\Lambda = \{\lambda_{ij}\}$-интенсивность перехода

$P_i - ?$

1. Найдем вероятность того, что в момент времени t система находится в состоянии S1. Придадим t малое приращение Δt и определим, что система в момент времени $t+\Delta t$ находится в состоянии S1.

$$p_1(t+\Delta t) = p_1(t)(1-\lambda_{12}\Delta t) + p_3(t)\lambda_{31}\Delta t$$

$$\lim_{\Delta t \to 0} \frac{p_1(t+\Delta t) - p_1(t)}{\Delta t} = -p_1(t)\lambda_{12} + p_3(t)\lambda_{31}$$

$$p_1'(t) = -p_1(t)\lambda_{12} + p_3(t)\lambda_{31}$$

2. Найдем вероятность того, что система находится в состоянии S2:

$$p_2(t+\Delta t) = p_2(t)(1-\lambda_{24}\Delta t - \lambda_{23}\Delta t) + p_1(t)\lambda_{12}\Delta t + p_4(t)\lambda_{42}\Delta t$$

$$\lim_{\Delta t \to 0} \frac{p_2(t+\Delta t) - p_2(t)}{\Delta t} = -p_2(t)\lambda_{24} - p_2(t)\lambda_{23} + p_1(t)\lambda_{12} + p_4(t)\lambda_{42}$$

$$p_2'(t) = -p_2(t)\lambda_{24} - p_2(t)\lambda_{23} + p_1(t)\lambda_{12} + p_4(t)\lambda_{42}$$

3. Найдем вероятность того, что система находится в состоянии S3:

$$p_3'(t) = -p_3(t)\lambda_{31} - p_3(t)\lambda_{34} + p_2(t)\lambda_{23}$$

4. Найдем вероятность того, что система находится в состоянии S4:

$$p_4'(t) = -p_4(t)\lambda_{42} + p_2(t)\lambda_{24} + p_3(t)\lambda_{34}$$

В результате получаем систему уравнений Колмогорова:

$$\begin{cases} p_1' = -p_1\lambda_{12} + p_3\lambda_{31} \\ p_2' = -p_2\lambda_{24} - p_2\lambda_{23} + p_1\lambda_{12} + p_4\lambda_{42} \\ p_3' = -p_3\lambda_{31} - p_3\lambda_{34} + p_2\lambda_{23} \\ p_4' = -p_4\lambda_{42} + p_2\lambda_{24} + p_3\lambda_{34} \end{cases}$$

Интегрирование данной системы даст искомые вероятности состояний, как функций времени. Начальные условия берутся в зависимости от того, какого было начальное состояние системы. Если при $t=0$ система находится в состоянии S1, то начальные условия будут p_1=1, p_2= p_3= p_4=0. Кроме этого, к системе добавляются условия нормировки:

$$\sum_{i=1}^{N} p_i = 1$$

Все уравнения строятся по определенному правилу:

1. В левой части каждого уравнения стоит производная вероятности состояния, а в правой части содержится столько членов, сколько стрелок связано с этим состоянием.

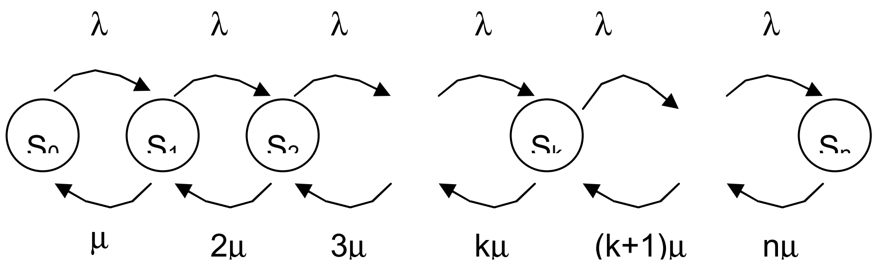
2. Если стрелка направлена «из» состояния, соответствующий член имеет знак “-“, если «в» состояние, то знак “+”.

3. Каждый член равен произведению плотности вероятности перехода (интенсивность), соответствующий данной стрелке, и вероятности того состояния, из которого выходит стрелка.

Пример.

Рассмотрим многоканальную СМО с отказами. Состояние системы характеризуется по числу занятых каналов, т.е. по числу заявок.

S0 – все каналы свободны
S1 – занят один канал, остальные свободны
Sk – занято k каналов, остальные свободны
Sn – заняты все n каналов.



Разметим граф, т.е. проставим у стрелок интенсивности соответствующих потоков событий. Пусть система находится в состоянии S1. Как только закончится обслуживание заявки, занимающей этот канал, система переходит в состояние S0, интенсивность перехода μ. Если занято 2 канала, а не один, то интенсивность перехода составит 2μ.

$$\begin{cases} p_0' = -p_0\lambda + p_1\mu \\ p_1' = -p_1\lambda - p_1\mu + p_0\lambda + p_2 2\mu \\ p_2' = -p_2\lambda - p_2 2\mu + p_1\lambda + p_3 3\mu \\ p_k' = -p_k\lambda - p_k k\mu + p_{k-1}\lambda + p_{k+1}(k+1)\mu \\ p_n' = p_{n-1}\lambda - p_n n\mu \end{cases}$$

Предельные вероятности состояний p_0 и p_n характеризую установившийся режим работы системы массового обслуживания при $t \to \infty$.

$$p_0 = \frac{1}{1 + \frac{\lambda/\mu}{1!} + \frac{(\lambda/\mu)^2}{2!} + \ldots + \frac{(\lambda/\mu)^n}{n!}}$$

$$p_k = \frac{(\lambda/\mu)^k}{k!} p_0$$

$\lambda/\mu = \rho$ - среднее число заявок, приходящих в систему за среднее время обслуживания одной заявки.

$$p_0 = \left[1 + \frac{\rho}{1!} + \frac{\rho^2}{2!} + \ldots + \frac{\rho^n}{n!}\right]^{-1}$$

$$p_k = \frac{\rho^k}{k!} p_0$$

Зная все вероятности состояний p_0 , … , p_n , можно найти характеристики СМО:

- *вероятность отказа* – вероятность того, что все n каналов заняты

$$p_{ОТК} = p_n = \frac{\rho^n}{n!} p^0$$

- *относительная пропускная способность* – вероятность того, что заявка будет принята к обслуживанию

$$q = 1 - p_n$$

- среднее число заявок, обслуженных в единицу времени

$$A = \lambda q$$

Полученные соотношения могут рассматриваться как базисная модель оценки характеристик производительности системы. Входящий в эту модель параметр $\lambda = 1 / t_{ОБРАБОТКИ}$, является усредненной характеристикой пользователя. Параметр μ является функцией технических характеристик компьютера и решаемых задач.

Эта связь может быть установлена с помощью соотношений, называемых интерфейсной моделью. Если время ввода/вывода информации по каждой задачи мало по сравнению со временем решения задачи, то логично принять, что время решения равно 1 / μ и равно отношению среднего числа операций, выполненных процессором при решении одной задачи к среднему быстродействию процессора.

Лекция 3

Метод статистических испытаний

На практике далеко не все случайные процессы являются Марковскими или близкими к ним. В СМО поток заявок не всегда Пуассоновский, ещё реже наблюдается показательное или близкое к нему распределение времени обслуживания.

Для произвольных потоков сообщений, переводящих систему из одного состояния в другое, аналитическое решение получено только для отдельных частных случаев. Когда построение аналитической модели по той или иной причине трудно осуществимо, применяется другой метод моделирования – *метод статистических испытаний (Монте-Карло)*. Широкое распространение метода связано с возможностью его реализации на компьютере.

Идея метода: вместо того, чтобы описывать случайное явление с помощью аналитической зависимости производится «розыгрыш», т.е. происходит моделирование случайного явления с помощью некоторой процедуры, дающей случайный результат. Произведя такой розыгрыш n раз, получаем статистический материал, т.е. множество реализаций случайного явления, которое потом можно обработать обычными методами математической статистики. Метод Монте-Карло предложил Фон-Нейман в 1948 году, как метод численного решения некоторых математических задач.

Суть метода:

1. Вводим в некотором единичном квадрате любую поверхность S.
2. Любым получаем 2 числа x_i, y_i, подчиняющиеся равномерному закону распределения случайной величины на интервале [0, 1].
3. Полагаем, что одно число определяет координату x, второе– координату y
4. Анализируем принадлежность точки (x, y) фигуре. Если принадлежит, то увеличиваем значение счетчика на 1.
5. Повторяем n раз процедуру генерации 2х случайных чисел с заданным законом распределения и проверку принадлежности точки поверхности S.
6. Определяем площадь фигуры как количество попавших точек, к количеству сгенерированных.

Фон-Нейман доказал, что погрешность $\varepsilon \le \sqrt{\dfrac{1}{n}}$.

Преимущества метода статистических испытаний в его универсальности, обуславливающей возможность всестороннего статистического исследования объекта, однако, для реализации этой возможности нужны довольно полные статистические сведения о параметрах элементов.

К *недостаткам* относится большой объем требуемых вычислений, равный количеству обращений к модели. Поэтому вопрос выбора величины n имеет важнейшее значение. Уменьшая n – повышаем экономичность расчетов, но одновременно ухудшаем их точность.

Способы получения последовательностей случайных чисел

На практике используются 3 основных способа генерации случайных чисел:

1. **аппаратный (физический)**
2. **табличный (файловый)**
3. **алгоритмический (программный)**

Аппаратный способ.

Случайные числа вырабатываются специальной электронной приставкой (генератором случайных чисел). Реализация этого способа не требует дополнительных вычислительных операций по выбору случайных чисел. Необходимо только оперативное обращение к ВУ.

В качестве физического эффекта, лежащего в основе генератора, чаще всего используют шуму в электронных приборах.

Простейшие алгоритмы генерации последовательности псевдослучайных чисел

Одним из первых способов получения является выделение значения дробной части у многочлена первой степени:

$$y_n = Ent(an + b)$$

Если n пробегает значения натурального ряда числе, то поведение y_n выглядит весьма хаотично.

К. Якоби доказал, что при рациональном коэффициенте *a* множество y конечно, а при иррациональном – бесконечно и всюду плотно в интервале [0, 1].

Критерий равномерности распределения любой функции: свойство эргамичности – среднее по реализациям псевдослучайное число равно среднему по всему их множеству с вероятностью 1.

1. **1946 год, Фон Нейман.**

Каждое последующее случайное число образуется возведением в квадрат предыдущего и отбрасывание цифр с обоих концов. Способ оказался ненадежным.

2. **Лемер.**

$g_{n+1} = kg_n + (c \bmod m)$. Для подборов коэффициентов k, c, m были потрачены десятки лет. Подбор почти иррациональных чисел ничего не дает.

3. Разумнее ввести вычисления с целыми числами.

при c = 0 и m = 2^n наибольший период достигается при k=3+8i или k=5+8i и при нечетном начальном числе.

Для имитации равномерного распределения на [a, b] используется обратное преобразование функции плотности вероятности.

$$\frac{x-a}{b-a} = R$$

$$x = a + (b-a)R$$

где R – равномерно распределенное псевдослучайное число на [0, 1].

В основе построения программы генерации случайной числа с законом распределения, отличным от равномерного, лежит метод преобразования последовательности случайных чисел с равномерным законом распределения в последовательность случайных чисел с заданным законом распределения.

Метод основан на том, что случайная величина x принимает значения, равные корню уравнения

$$F(t) = \int_{-\infty}^{t} f(x)dx = R$$

, имеет плотность распределения f(x). R – случайная величина, равномерно распределенная на [0, 1].

Значение случайной величины, распределенной по показательному закону может быть вычислено из данного уравнения следующим образом:

$$1 - e^{-\lambda X} = R$$

$$x = (-\frac{1}{\lambda})\ln(1-R)$$

Распределение Пуассона относится к числу дискретных (переменная может принимать лишь целочисленные значения, включая 0, с математическим ожиданием и дисперсией $\lambda > 0$). Для генерации Пуассоновских переменных можно использовать метод точек, в основе которого лежит генерируемое случайное значение R_i , равномерно распределенное на [0, 1], до тех пор, пока не станет справедливым

$$\prod_{i=0}^{x} R_i \ge e^{-\lambda} > \prod_{i=0}^{x+1} R_i$$

При получении случайной величины, функция распределения которой не позволяет найти решение уравнения

$$F(t) = \int_{-\infty}^{t} f(x)dx = R$$

в явной форме, можно произвести кусочно-линейную аппроксимацию, а затем вычислить приближенное значение корня. Кроме того, при получении случайной величины часто используют те или иные свойства распределений.

Воспользуемся этим методом, чтобы сгенерировать случайную величину с законом распределения Эрланга. *Распределение Эрланга* характеризуется параметрами λ и k.

При вычислении случайно величины воспользуемся тем, что поток Эрланга может быть получен путем прорешивания потока Пуассона k раз. Поэтому, достаточно получить k значений случайной величины распределенной по показательному закону и усреднить их.

$$x = \frac{1}{k}\left(\sum_{i=1}^{k}\left(-\frac{1}{\lambda}\right)\ln(1-R_i)\right) = -\frac{1}{k\lambda}\sum_{i=1}^{k}\ln(1-R_i)$$

Нормальное распределение случайной величины может быть получено как сумма большого числа случайных величин, распределенных по одному и тому же закону распределения с одними и теми же параметрами.

Случайная величина X имеющая нормальное распределение с математическим ожиданием M_X и среднеквадратичным отклонением σ_X может быть получена по следующей формуле:

$$x = \sigma_X \cdot \sqrt{\frac{12}{N}} \cdot \left(\sum_{i=1}^{N} R_i - \frac{N}{2}\right) + M_X$$

Для сокращения вычислений на практике принимаю N=12, что дает вполне относительно хорошее приближение к нормальному распределению.

Лекция 4

Немарковские случайные процессы, сводящиеся к марковским

Реальные процессы часто обладают последствием и поэтому не являются марковскими. Иногда при исследовании таких процессов удается воспользоваться методами, разработанными для марковских процессов. Наиболее распространены два способа:

1. **Метод разложения случайного процесса на фазы (метод псевдосостояний)**
2. **Метод вложенных цепей Маркова**

Метод псевдосостояний

Суть метода состоит в том, что состояния системы, потоки переходов из которых являются немарковскими, заменяются эквивалентной группой фиктивных состояний, потоки переходов из которых являются марковскими.

Условие статистической эквивалентности реального состояния и соответствующих ему фиктивных состояний в каждом конкретном случае выбирается по-разному. В качестве одного из критериев эквивалентности можно принять следующее условие:

$$\min \int_{t_2}^{t_1} [\lambda_{i экв}(\tau) - \lambda_i(\tau)] d\tau$$

, где $\lambda_{i\ экв}(\tau)$ – эквивалентная интенсивность перехода в i-той группе переходов, заменяемой реальный переход обладающей интенсивностью $\lambda_i(\tau)$.

За счет расширения числа состояний системы, некоторые процессы удается точно свести к марковским. Созданная таким образом новая система по своим характеристикам статистически эквивалентна или близка реальной системе, но она должна быть обязательно подвергнута обычному исследованию на адекватность, с помощью хорошо проработанного математического аппарата с использованием уравнений Колмогорова.

К числу процессов, которые введением фиктивных состояний можно точно свести к марковским, относятся процессы, происходящие в системе под воздействием потока Эрланга.

В случае потока Эрланга k-го порядка интервал времени между сообщениями представляет собой сумму k независимых случайных интервалов распределенных по показательному закону. Поэтому сведение потока Эрланга k-го порядка к Пуассоновскому осуществляется введением k псевдосостояний. Интенсивности перехода между псевдосостояниями равны соответствующему параметру потока Эрланга.

Полученная таким образом эквивалентный случайный процесс является марковским, т.к. интервалы времени нахождения его в различных состояниях подчинены показательному закону распределения.

Пример.

Некоторое устройство S выходит из строя с интенсивностью λ, причем поток отказов Пуассоновский. После отказа устройство восстанавливается. Время восстановления распределено по закону Эрланга 3-го порядка с функцией плотности $f(t) = 0{,}5\mu(\mu \cdot t)^2 e^{-\mu \cdot t}$. Найти предельные вероятности возможных состояний системы.

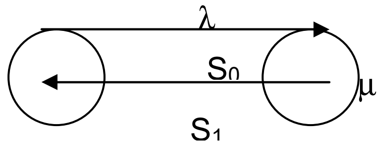
Система S может принимать два состояния:

S0 – устройство исправно

S1 – устройство отказало и восстанавливается

S0 → S1 – Пуассоновский поток

S1 → S0 – поток Эрланга

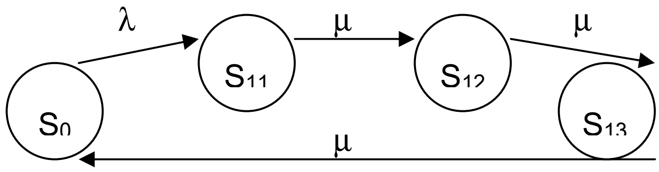


Представим случайное время восстановления в виде суммы 3х интервалов, распределенных по показательному закону с интенсивностью μ:

$t = t_1 + t_2 + t_3$

S1 заменяем эквивалентной цепочкой из трех псевдосостояний.

$$\begin{cases} p_0'(t) = \mu \cdot p_{13}(t) - \lambda \cdot p_0(t) \\ p_{11}'(t) = -\mu \cdot p_{11}(t) + \lambda \cdot p_0(t) \\ p_{12}'(t) = \mu \cdot p_{11}(t) - \mu \cdot p_{12}(t) \\ p_{13}'(t) = \mu \cdot p_{12}(t) - \mu \cdot p_{13}(t) \\ p_0 + p_1 = 1 \\ p_1 = p_{11} + p_{12} + p_{13} \end{cases}$$



Последние 2 уравнения являются условием нормировки и условием эквивалентности замены состояния S1 псевдосостояниями.

Метод вложенных цепей Маркова.

Вложенные марковские цепи образуются следующим образом: в исходном случайном процессе выбираются такие моменты времени tk, в которых значения характеристик процесса образует марковскую цепь. Моменты времени t_k обычно являются случайными и зависят от свойств самого процесса. Исходный процесс исследуется в выбранные моменты времени с помощью стандартных методов теории массового обслуживания.

Частным случаем вложенных цепей Маркова, являются *полумарковские* случайные процессы. Случайный процесс с конечным или счетным множеством состояний называется *полумарковским,* если заданны вероятности перехода системы из одного состояния в другое и распределение времени пребывания процесса в каждом состоянии (в виде функции распределения *F(t)* или в виде плотности распределения *f(t)*)

Лекция 5

Классификация систем массового обслуживания

В общем случае СМО классифицируется по следующим признакам:

- **закону распределения входного потока**
- **числу обслуживающих приборов**
- **закону распределения времени обслуживания в обслуживающих приборах**
- **числу мест в очереди**
- **дисциплине обслуживания**

При определении любой системы обслуживания для сокращенной записи используется следующая система кодировки, в которой на месте букв ставится соответствующая характеристика СМО:

A | B | C | D | E

A	Закон распределения интервалов времени между поступлением заявок.	***M*** – экспоненциальный ***E*** – Эрланга ***H*** – гиперэкспоненциальный ***Г*** – гамма-распределение ***D*** – детерминированное распределение ***G*** – произвольное распределение
B	Закон распределения времени обслуживания в приборах СМО.	
C	Число обслуживающих приборов.	***1*** – для одноканальных систем ***l*** – для многоканальных систем
D	Число мест в очереди. Если число мест не ограничено, то поле можно опустить.	***r*** либо ***n*** – для конечного числа мест
E	Дисциплина обслуживания. По умолчанию LIFO – в этом случае поле может опускаться.	*FIFO, LIFO, RANDOM*

Примеры.

M | M | 1

СМО с одним обслуживающим прибором, бесконечной очередью, экспоненциальным законом распределения интервалов времени между поступлением заявок и временем обслуживания и дисциплиной обслуживания FIFO.

E | H | l | r | LIFO

СМО с несколькими обслуживающими приборами, конечной очередью, законом распределения Эрланга интервалов между поступлением заявок, гиперэкспоненциальным законом распределения времени обслуживания заявок в приборах и дисциплиной обслуживания LIFO.

$G \mid G \mid l$

СМО с несколькими приборами, бесконечной очередью, произвольными законами распределения времени между поступлением заявок и времени обслуживания, FIFO.

Для моделирования вычислительной системы наиболее часто используются следующие типы СМО:

1. **Одноканальная СМО с ожиданием.**

- один обслуживающий прибор с бесконечной очередью
- является наиболее распространенной при моделировании
- может заменить практически любой узел вычислительной системы или ЛВС

2. **Одноканальная СМО с потерями.**

- один обслуживающий прибор с конечным числом мест в очереди
- если число заявок превышает число мест в очереди, то лишние заявки теряются
- используется при моделировании каналов передач данных в ВС и ЛВС

3. **Многоканальная СМО с ожиданием.**

- несколько параллельно работающих обслуживающих приборов с общей бесконечной очередью
- используется при моделировании групп абонентских терминалов, работающих в диалоговом режиме

4. **Многоканальная СМО с потерями.**

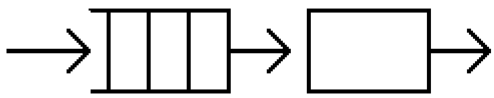
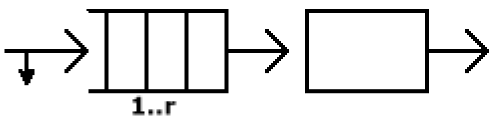
- несколько параллельно работающих обслуживающих приборов с общей очередью, число мест в которой ограничено
- часто используется, как и (2), при исследовании каналов связи

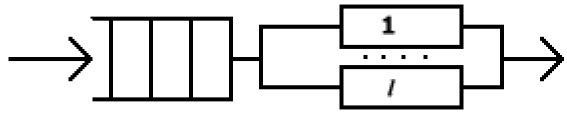
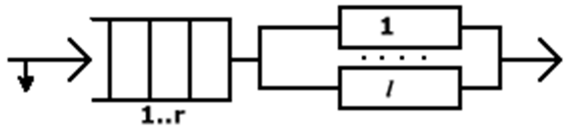
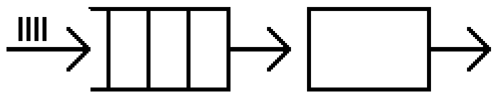
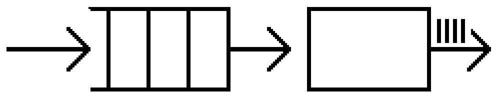
5. **Одноканальная СМО с групповым поступлением заявок**

- один обслуживающий прибор с бесконечной очередью
- перед обслуживанием заявки группируются в пакеты по определенном признаку или правилам
- используется для моделирования центров коммутации

6. **Одноканальная СМО с групповым обслуживанием заявок**

- один обслуживающий прибор с бесконечной очередью
- заявки обслуживаются пакетами, составленными по определенному правилу
- используется для моделирования центров коммутации

Наименование	Обозначение	Схема
Одноканальная с ожиданием	**G | G | 1**	
Одноканальная с потерями	***G | G | 1 | r***	

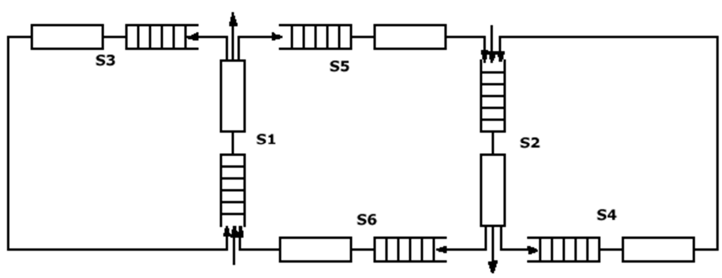
Многоканальная с ожиданием	***G* | *G* | *l***	
Многоканальная с потерями	***G* | *G* | *l* | *r***	
Одноканальная с групповым поступлением заявок	***Gr* | *G* | 1** ***r***	
Одноканальная с групповым обслуживанием заявок	***G* | *Gr* | 1** ***r***	

Вычислительные сети в целом могут быть представлены в виде СМО. Различают следующие типы сетей:

1. **Открытые**
2. **Замкнутые**
3. **Смешанные**

Открытой называется СМО, состоящая из m узлов, причем хотя бы в один из узлов сети поступают извне входной поток заявок и обязательное имеется хотя бы один сток заявок из сети.

Для открытой системы характерно то, что интенсивность поступления заявок в сеть не зависит от состояния сети, т.е. от числа заявок уже поступивших в сеть. Такие сети как правило используются для моделирования ВС, работающих в неоперативном режиме.



S1, S2 – моделируют работу узлов коммутации

S3, S4 – моделируют работу серверов

S5, S6 – моделируют работу межузловых каналов

В сети циркулируют 2 потока заявок. Каждая заявка поступает на вход соответствующего узла коммутатора, где определяется место её обработки. Затем заявка передается на свой сервер или по каналу связи на соседний сервер, где обрабатывается. После чего, возвращается к источнику и покидает сеть.

Замкнутой называется СМО с множеством узлов m без источника и стока, в которой циркулирует постоянное число заявок.

Замкнутые СМО используются для моделирования таких ВС, источником информации для которых служат абонентские терминалы, работающие в диалоговом режиме. В этом случае каждая группа абонентских терминалов представляется в виде многоканальной СМО с ожиданием и включается в состав устройств сети.

Простой режим работы диалоговых абонентов: абонент не производит никаких действий, кроме посылки заданий в ВС и обслуживания полученного ответа.

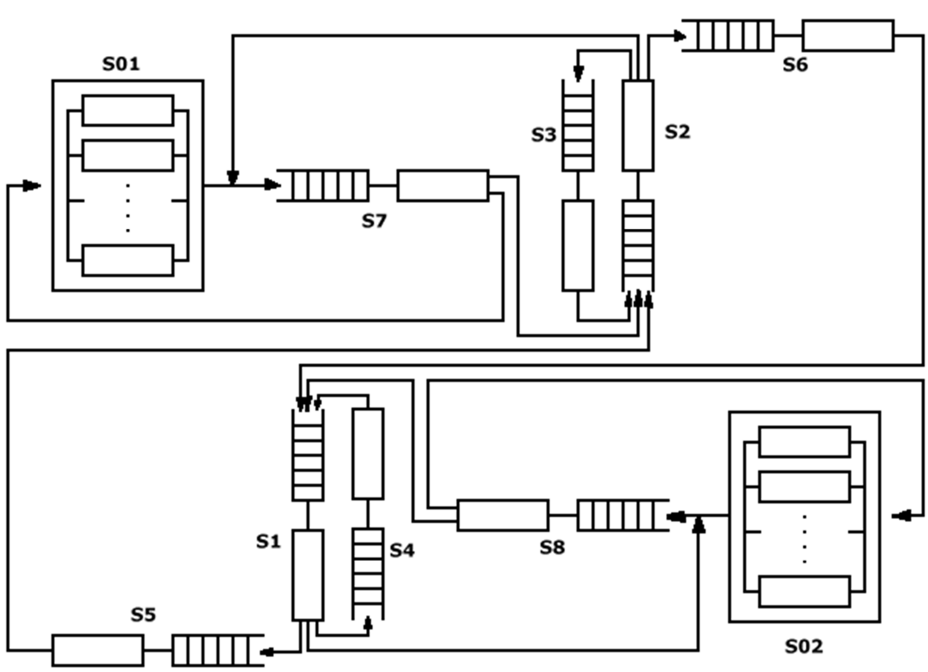
S01, S02 – группа абонентских терминалов

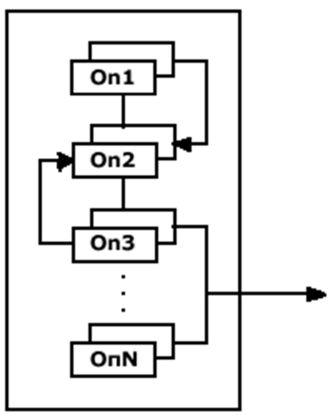
S7, S8 – каналы связи с абонентами

S1, S2 – узлы коммутации

S3, S4 – серверы

S5, S6 – каналы межузловой связи





Абоненты с терминалов посылают запросы, которые по каналам связи поступают на узлы коммутатора, а оттуда на обработку на свой или соседний сервер.

При сложном режиме диалога работа абонентов представляется в виде совокупности операций некоторого процесса, называющимся *технологическим*. Каждая операция технологического процесса моделируется соответствующей

СМО. Причем, часть операций может предусматривать обращение к ВС, а другая часть может замыкаться сама на себя.

Схема представленная на рисунке находится на месте S01 и S02.

Смешанной называется СМО, в которой циркулируют несколько различных типов заявок (трафика). Причем относительно одних типов заявок сеть замкнута, а относительно других – открыта.

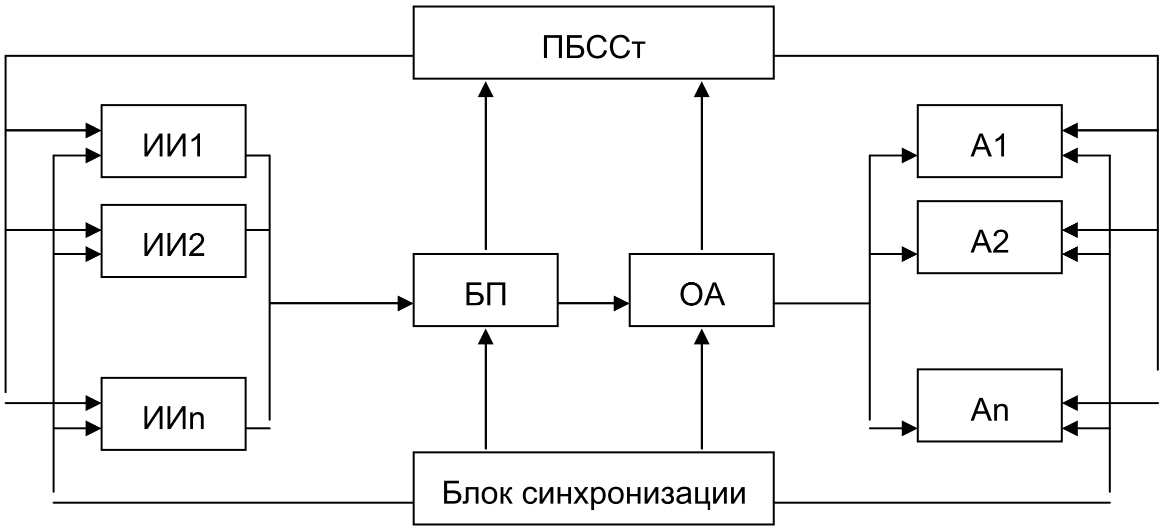
С помощью смешанных сетей моделируются такие ВС, часть абонентов которых работает в диалоговом режиме, а часть в неоперативном. Для диалоговых абонентов различают также простой и сложный режимы работы. Часто смешанные СМО используются в сетях, в которых сервер дополнительно загружается задачами, решаемыми на фоне работы самой сети.

Лекция 6

Методика построения программной модели

Для разработки программной модели исходная система должна быть представлена как стохастическая СМО. Это объясняется следующим: информация от внешней среды поступает в случайные моменты времени, длительность обработки различных типов информации в общем случае также является различной. Таким образом, внешняя среда может быть отображена как генератор сообщений, а комплекс ВС – обслуживающими устройствами.

Обобщенная структурная схема ВС.



ИИ – источники информации – выдают на вход буферной памяти независимые друг от друга сообщения. Закон появления сообщений - произвольный.

В **БП** (буферной памяти) сообщения записываются «в навал» и выбираются по одному в обслуживающий аппарат по принципу FIFO/LIFO. Длительность обработки одного сообщения в **ОА** (обслуживающий аппарат) в общем случае может быть случайной, но закон обработки всегда должен быть задан. Т.к. быстродействие ОА ограничено, то на входе системы в БП возможно сложение данных, ожидающих обработку.

ПБССт - программный блок сбора статистики.

Блок синхронизации необходим, чтобы система заработала.

Моделирование потока сообщений.

Поток сообщений обычно моделируется моментами появления очередного сообщения в потоке. Текущий момент времени появления очередного сообщения:

$$t_i = \sum_{k=1}^{i-1} T_k + T_i$$

где T_i – интервал времени между появлением i-го и (i-1)-го сообщения.

Процедура.
обращение к процедуре выражения случайного числа Rnd

$$T_i = -\frac{1}{\lambda}\ln(1-R_i)$$

$$T = T + T_i$$

Вид распределения	**Выражение**
равномерное на [a,b]	$T_i = a + (b-a)R$
нормальное	$T_i = \sigma_X \sqrt{\frac{12}{n}}(\sum_{i=1}^{n} R_i - \frac{n}{2}) + M_X$
экспоненциальное	$T_i = -\frac{1}{\lambda}\ln(1-R)$
Эрланга	$T_i = \frac{1}{k\lambda}\sum_{i=1}^{k}\ln(1-R_i)$

Моделирование работы обслуживающего аппарата (ОА).

Программа – имитатор работы ОА представляет собой комплекс, вырабатывающий случайные отрезки времени, соответствующие длительностям обслуживания требований. Например, если требования от источника обрабатываются в ОА по нормальному закону с параметрами *Mx* и σx, то длительность обработки i-ого требования:

$$t_{ОБР} = M_X + (\sum_{i=1}^{12} R_i - 6)\cdot\sigma_X$$

Схема алгоритма имитатора.

R_i – случайное число с равномерным законом распределения
$T_{ОБР}$ – время обработки очередного сообщения
T – время освобождения ОА
XM – Мат ожидание для заданного закона обратки
DX – СКО для заданного закона обратоки

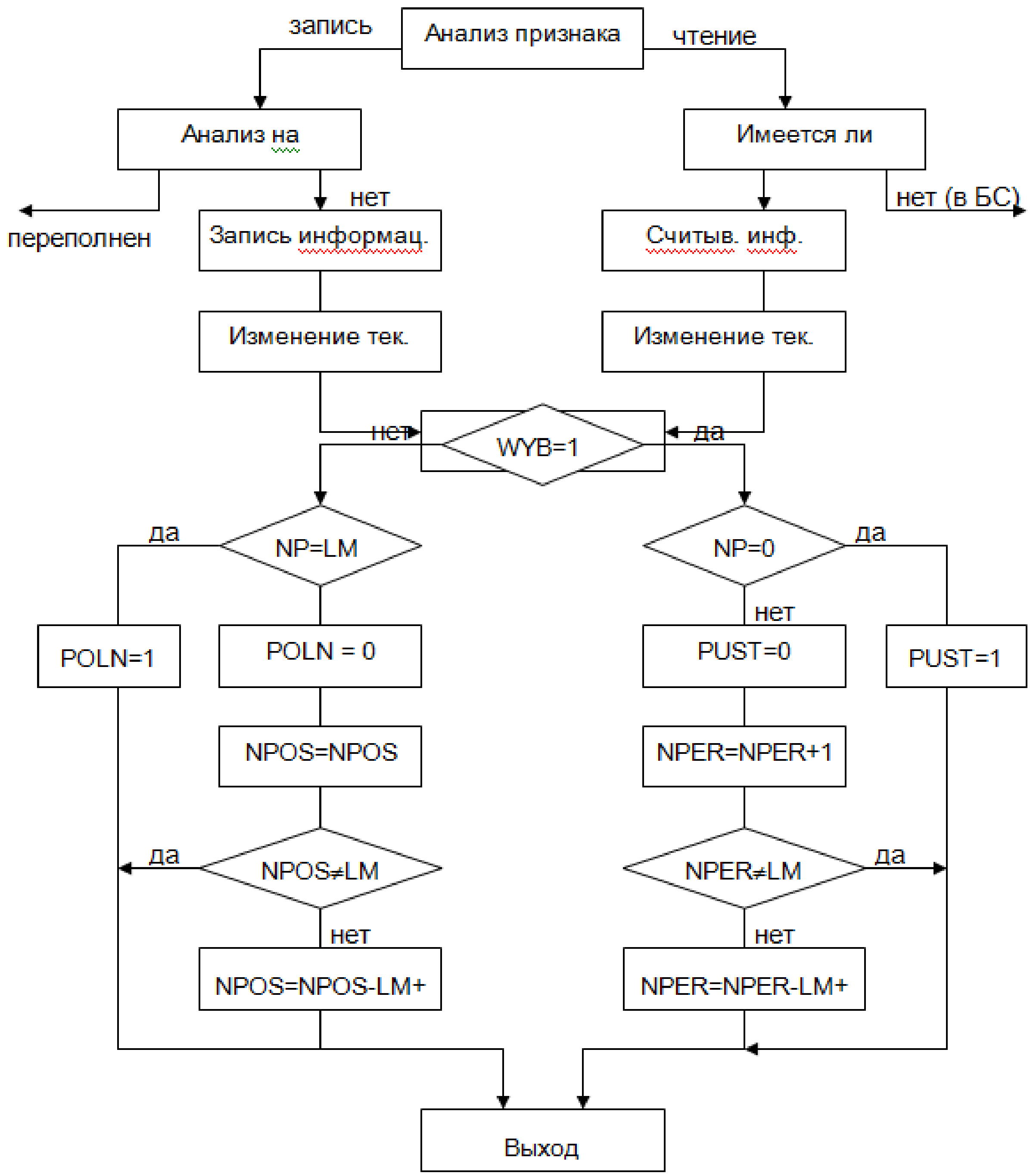
$$T_{ОБР} = XM + (S-6) * DX$$

Моделирование работы абонента

Абонента можно рассматривать как ОА, поток информации на который поступает от процессора. Для моделирования работы абонентов необходимо вырабатывать длительности обслуживания требований. Кроме того, абонент сам может быть источником заявок, претендуя на те или иные ресурсы вычислительной системы. Эти заявки могут имитироваться с помощью генератора сообщений по наперед заданному закону. Таким образом, абонент либо имитируется как ОА, либо как генератор.

Моделирование работы буферной памяти

Блок буферной памяти должен производить запись и считывания числа, выдавать сигналы переполнения и отсутствия данных. В любой момент времени располагать сведениями о количестве требований в блоке. Сама запоминающая среда имитируется некоторым одномерным массивом, размер которого определяет размер БП. Каждый элемент этого массива может быть либо свободен, либо занят.



P	массив сообщений	*LM*	объем буферной памяти
WYB	признак обращения к буф. памяти = 1 – режим выборки сообщений = 0 – режим записи	*NPOS*	номер последнего сообщения, поступившего в память
NP	число сообщений в памяти	*NPER*	номер первого сообщения в памяти
POLN	признак переполнения памяти = 1 – нет свободных ячеек	*PUST*	признак отсутствия сообщений = 1 – в памяти нет сообщений
NPOS	= NPOS + 1, если NPOS < LM = NPOS – LM + 1, иначе	*NPER*	= NPER – 1, если NPER < 1 = NPER – LM + 1, иначе
X	ячейка для сообщения		