

Содержание

Введение	5
1. Потоки событий	6
1.1. Определение потока событий.....	6
1.2. Вероятностные характеристики потока.....	6
1.3. Наиболее важные свойства потока.....	7
1.4. Расчет вероятности появления событий потока на заданном временном интервале для пуассоновского потока.....	8
1.5. Условие стационарности потока	10
1.6. Простейший поток событий.....	10
1.7. Случайная величина - число событий потока на заданном временном интервале	11
1.8. Интенсивность потока событий.....	12
1.9. Закон распределения интервала времени между последовательными событиями потока	12
1.10. Эквивалентное определение простейшего потока событий.....	13
1.11. Объединение (наложение) пуассоновских потоков событий	13
1.12. Разрежение пуассоновского потока событий.....	14
2. Имитационное моделирование (метод Монте-Карло)	17
2.1. Моделирование простейшего потока событий.....	17
2.2. Проверка качества моделирования.....	18
2.3. Построение доверительного интервала.....	19
2.4. Проверка гипотезы об законе распределения по критерию Пирсона	19
2.5. Моделирование объединения (наложения) пуассоновских потоков событий.....	21
2.6. Моделирование разряжения пуассоновского потока событий.....	22
2.7. Моделирования числа событий потока на заданном временном интервале.	22
2.8. Проверка гипотезы о Пуассоновском законе распределения	24
2.9. Задача о характеристиках случайного временного интервала	25
3. Цепи Маркова	27
3.1. Случайный процесс в системе с дискретным числом состояний.....	27
3.2. Марковское свойство	27
3.3. Цепи Маркова	27
3.4. Матрица переходных вероятностей (матрица перехода)	28
3.5. Матрица перехода за n шагов	29
3.6. Распределение вероятностей состояний системы	29
3.7. Изображение цепи Маркова в виде ориентированного графа	30

3.8.	Классификация состояний цепи Маркова	31
3.9.	Пример анализа состояний цепи Маркова	35
3.10.	Эргодическая теорема	36
3.11.	Стационарное распределение вероятностей	38
3.12.	Связь финального и стационарного распределения вероятностей.....	38
4.	Моделирование цепей Маркова	39
4.1.	Пример технической системы	39
4.2.	Изображение системы в виде ориентированного графа	39
4.3.	Моделирование функции перехода	41
4.4.	Моделирования последовательности переходов.....	42
4.5.	Моделирование времени нахождения системы в каждом из состояний	42
4.6.	Сравнение результатов моделирования с теоретическими результатами	43
4.7.	Среднее время непрерывного нахождения в произвольном состоянии и средняя длина цикла от попадания в состояние до следующего попадания в это же состояние	44
4.8.	Моделирование функции перехода с использованием стандартных функций пакета Wolfram Mathematica	45
4.9.	Вычисление заданной функции вдоль траектории.....	46
5.	Марковский случайный процесс с дискретным числом состояний и непрерывным временем.....	49
5.1.	Случайный процесс с дискретным числом состояний и непрерывным временем	49
5.2.	Марковский случайный процесс с непрерывным временем	50
5.3.	Потоки событий, связанные со сменой состояний.....	50
5.4.	Уравнения Колмогорова	51
5.5.	Пример решения системы дифференциальных уравнений.....	54
6.	Моделирование Марковского процесса	56
6.1.	Описание исходной системы.....	56
6.2.	Построение описания системы в виде списка	56
6.3.	Построение графа переходов с указанием интенсивностей переходов	56
6.4.	Генератор временного интервала для событий потока	57
6.5.	Моделирование Марковского процесса.....	58
6.6.	Решение системы дифференциальных уравнений Колмогорова.....	61
	СПИСОК ЛИТЕРАТУРЫ.....	62

Введение

Дисциплина «Математическое моделирование» предназначена для подготовки магистров по специальности 38.04.05 «Бизнес-информатика». Курс знакомит с дополнительными разделами теории вероятности и математической статистики. Эти разделы объединяются общей идеей, связанной с понятиями марковского свойства и марковского процесса. Конкретными примерами являются цепи Маркова и марковские процессы с непрерывным временем. Использование последнего понятия иллюстрируется важной для экономических приложений теорией систем массового обслуживания (СМО).

Учебно-методическое пособие предназначено для самостоятельного изучения теоретических основ дисциплины и освоению методов имитационного моделирования применительно к марковским процессам. Приведены основные теоретические факты и модели марковских процессов. Основной упор делается на имитационном моделировании (метод Монте-Карло) основных математических понятий. Процесс моделирования проводится с использованием пакета символьных и численных вычислений Wolfram Mathematica и языка программирования Wolfram Language. В рамках этого пакета моделируются потоки событий и основные свойства таких потоков, цепи Маркова с основным упором на существовании финальных распределений, марковские процессы с непрерывным временем, определение законов распределения и характеристик состояний, а также системы массового обслуживания (часть 2).

Использование пакета Wolfram Mathematica (версий не ниже 7.0) и языка программирования Wolfram Language позволяет легко осуществлять процесс моделирования, вычислять основные характеристики марковских систем и, оставаясь в рамках пакета, осуществлять визуализацию полученных данных и давать рекомендации по использованию полученных результатов.

1. Потоки событий

1.1. Определение потока событий

Потоком называется последовательность событий, происходящих одно за другим в случайные моменты времени. Различают потоки **однородных** и **неоднородных** событий. Однородные события отличаются только моментом своего появления, во всем остальном эти события считаются неразличимыми. События в неоднородном потоке отличаются еще какими-то дополнительными характеристиками.

Наглядно поток событий, точнее одну из его реализаций (рис. 1), можно изобразить на временной оси последовательностью точек с абсциссами $\theta_1, \theta_2, \dots$

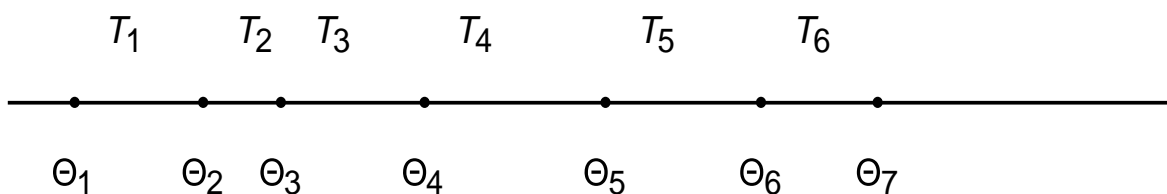


Рис. 1

В дальнейшем будут рассматриваться только потоки однородных событий. Поэтому у каждого из событий потока никаких других характеристик, кроме момента появления θ , нет.

Потоки событий различаются своей внутренней структурой, то есть свойствами случайной последовательности $\theta_1, \theta_2, \dots$. Рассмотрим интервалы времени $T_i = \theta_{i+1} - \theta_i, i = 1, 2, \dots$ между последовательными событиями потока.

Поток событий называется **регулярным потоком**, если эти интервалы постоянны, $T_i = Const$.

Наибольший интерес на практике представляют потоки, у которых временные интервалы T_i являются случайными величинами. Если эти временные интервалы T_i статистически независимы между собой, то поток событий называется **потоком Пальма** или потоком с ограниченным последствием. Если, дополнительно, все эти случайные величины имеют одинаковое вероятностное распределение, то такой поток событий называется **рекуррентным**.

Наиболее важным частным случаем потока Пальма является **пуассоновский поток** событий, который служит хорошей математической моделью для изучения общих свойств событий потока. В дальнейшем основное внимание будет уделено именно этому типу потоков событий.

1.2. Вероятностные характеристики потока

Перейдем к вероятностным характеристикам произвольного потока. Обозначим через $A_k(t, t+\tau)$ случайное событие, состоящее в том, что на временном интервале $[t; t + \tau)$ происходит ровно k событий потока. Вероятность этого события

$$P_k(t, t + \tau) = P(A_k(t, t + \tau)), k = 0, 1, 2, 3, \dots$$

Таким образом:

$P_k(t, t + \tau)$ - вероятность того, что на заданном временном интервале не произошло ни одного события потока,

$P_{\geq k}(t, t + \tau)$ - вероятность того, что на заданном временном интервале произошло не менее k событий потока,

$P_{\leq k}(t, t + \tau)$ - вероятность того, что на заданном временном интервале произошло не более k событий потока.

1.3. Наиболее важные свойства потока.

Ординарный поток. Поток событий называется ординарным потоком, если он удовлетворяет условию

$$\lim_{\Delta t \rightarrow 0} \frac{P_{\geq 2}(t, t + \Delta t)}{\Delta t} = 0$$

для любой точки t . Это условие означает, что вероятность наступления двух или более событий потока на временном отрезке $[t; t + \Delta t)$ стремится к нулю (при $\Delta t \rightarrow 0$) и при этом быстрее, чем Δt . Так как при $\Delta t \rightarrow 0$ временной отрезок стягивается к точке, то осуществление двух или более событий потока предполагается практически невозможным. Следовательно, на этом временном интервале может произойти максимум только одно событие потока, или вообще ни одного.

Условие ординарности можно переписать в виде $P_{\geq 2}(t, t + \Delta t) = o(\Delta t)$, то есть вероятность осуществления двух или более событий потока на временном интервале длительности Δt есть бесконечно малая более высокого порядка малости, чем Δt .

Отсутствие последействия. Рассмотрим два непересекающихся временных интервалов $[t_1; t_1 + \tau_1)$ и $[t_2; t_2 + \tau_2)$ и, соответственно, два события $A_k(t_1, t_1 + \tau_1)$ и $A_m(t_2, t_2 + \tau_2)$, где k и m - произвольные числа событий потока на интервалах $[t_1; t_1 + \tau_1)$ и $[t_2; t_2 + \tau_2)$, соответственно. Отсутствие последействия означает, что эти два случайных события статистически независимы. Таким образом, осуществление любого числа событий потока на одном интервале никак не влияет на количество осуществлений событий потока на другом интервале.

Пуассоновский поток. Поток событий называется пуассоновским, если этот поток ординарный и для него отсутствует последействие, то есть ординарный поток без последействия есть пуассоновский поток.

Параметр потока. Одной из важнейших характеристик потока является его параметр. Рассмотрим предел

$$\lim_{\Delta t \rightarrow 0} \frac{P_{\geq 1}(t, t + \Delta t)}{\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{1 - P_0(t, t + \Delta t)}{\Delta t}$$

Если этот предел существует, он обозначается $\lambda(t)$ и называется параметром потока. Величина $\lambda(t)$ естественно, больше или равна нулю.

Из этого определения вытекает, что

$$P_0(t, t + \Delta t) = 1 - \lambda(t)\Delta t + o(\Delta t)$$

Если поток ординарный, то $P_{\geq 2}(t, t + \Delta t) = o(\Delta t)$ и, следовательно,

$$P_1(t, t + \Delta t) = \lambda(t)\Delta t + o(\Delta t)$$

1.4. Расчет вероятности появления событий потока на заданном временном интервале для пуассоновского потока

Для пуассоновского потока можно получить в явном виде выражения для $P_k(t_0, t)$. Начнем с $P_0(t_0, t)$.

Вероятность $P_0(t_0, t)$ – отсутствие событий потока на заданном интервале. Рассмотрим временной интервал $[t_0, t_0 + t + \Delta t]$ и соответствующее событие $A_0(t_0, t + \Delta t)$. Очевидно следующее равенство

$$A_0(t_0, t + \Delta t) = A_0(t_0, t) * A_0(t, t + \Delta t)$$

В силу отсутствия последействия эти два события независимы, так временные интервалы не пересекаются. Следовательно,

$$P(A_0(t_0, t + \Delta t)) = P(A_0(t_0, t)) * P(A_0(t, t + \Delta t))$$

или

$$P_0(t_0, t + \Delta t) = P_0(t_0, t) * P_0(t, t + \Delta t)$$

В силу условия ординарности второй сомножитель равен

$$P_0(t, t + \Delta t) = (1 - \lambda(t)\Delta t + o(\Delta t))$$

и

$$P_0(t_0, t + \Delta t) = P_0(t_0, t) * (1 - \lambda(t)\Delta t + o(\Delta t))$$

что, эквивалентно равенству

$$P_0(t_0, t + \Delta t) - P_0(t_0, t) = -P_0(t_0, t) * \lambda(t)\Delta t + o(\Delta t)$$

Разделим обе части на Δt и перейдем к пределу при $\Delta t \rightarrow 0$

$$\lim_{\Delta t \rightarrow 0} \frac{P_0(t_0, t + \Delta t) - P_0(t_0, t)}{\Delta t} = -P_0(t_0, t) * \lambda(t)$$

Предел в левой части равен производной, поэтому получаем дифференциальное уравнение для вероятности события $P_0(t_0, t)$

$$\frac{dP_0(t_0, t)}{dt} = -P_0(t_0, t) * \lambda(t)$$

Получили обыкновенное дифференциальное уравнение первого порядка с разделяющимися переменными. Поделив переменные, получим

$$\frac{dP}{P} = -\lambda(t_0 + t)dt$$

или, после интегрирования

$$P_0(t_0, t) = C e^{-\int_{t_0}^t \lambda(t') dt'}$$

Так как при $t = 0$ должны получить 1, то постоянная интегрирования C должна равняться 1. Окончательно

$$P_0(t_0, t) = e^{-\Lambda(t_0, t)}, \Lambda(t_0, t) = \int_{t_0}^t \lambda(t') dt'$$

Вероятность $P_k(t_0, t)$. Аналогичные рассуждения позволяют получить и вероятность для произвольного числа событий потока. Представим событие $A_k(t_0, t + \Delta t)$ в виде следующей суммы произведений попарно-несовместных событий

$$A_k(t_0, t + \Delta t) = \sum_{i=0}^{i=k} A_{k-i}(t_0, t) * A_i(t, t + \Delta t)$$

или для соответствующих вероятностей, учитывая статистическую независимость событий $A_k(t_0, t)$ и $A_k(t, t + \Delta t)$ (свойство отсутствие последствия)

$$P_k(t_0, t + \Delta t) = \sum_{i=0}^{i=k} P_{k-i}(t_0, t) * P_i(t, t + \Delta t)$$

Используя свойство ординарности, заменим вероятности $P_0(t, t + \Delta t) = 1 - \lambda(t)\Delta t + o(\Delta t)$, $P_1(t, t + \Delta t) = \lambda(t)\Delta t + o(\Delta t)$

и

$$P_{i>1}(t, t + \Delta t) = o(\Delta t)$$

на их значения при малых значениях Δt :

$$\begin{aligned} P_k(t_0, t + \Delta t) &= P_k(t_0, t)(1 - \lambda(t)\Delta t + o(\Delta t)) + \\ &+ P_{k-1}(t_0, t)\lambda(t)\Delta t + o(\Delta t) = \\ &= P_k(t_0, t) - P_k(t_0, t)\lambda(t)\Delta t + \\ &+ P_{k-1}(t_0, t)\lambda(t)\Delta t + o(\Delta t) \end{aligned}$$

Перенесем первое слагаемое налево, разделим на Δt и перейдем к пределу при $\Delta t \rightarrow 0$. Получим линейное дифференциальное уравнение первого порядка.

$$\frac{dP_k(t_0, t)}{dt} = -P_k(t_0, t) * \lambda(t) + P_{k-1}(t_0, t) * \lambda(t)$$

Точнее, систему рекуррентных дифференциальных уравнений. Решение уравнения при некотором значении k зависит от решения при значении $(k - 1)$. Так при $k = 1$ получаем уравнение

$$\frac{dP_1(t_0, t)}{dt} = -P_1(t_0, t) * \lambda(t) + P_0(t_0, t) * \lambda(t)$$

Получили линейное неоднородное дифференциальное уравнение, дополнительное начальное условие $P_1(t_0, t)$ при $t = t_0$ равно нулю, вытекает из свойства ординарности.

Решение этого уравнения имеет вид

$$P_1(t_0, t) = \Lambda(t_0, t) e^{-\Lambda(t_0, t)}, \text{ где } \Lambda(t_0, t) = \int_{t_0}^t \lambda(t') dt'$$

А решение уравнения для любого k (с дополнительным условием $P_k(t_0, t)$ при $t = t_0$ равно нулю)

$$P_k(t_0, t) = \frac{(\Lambda(t_0, t))^k}{k!} e^{-\Lambda(t_0, t)}, \text{ где } \Lambda(t_0, t) = \int_{t_0}^t \lambda(t') dt'$$

1.5. Условие стационарности потока

Поток называется **стационарным**, если для любых интервалов времени $[t_1; t_1 + \tau)$ и $[t_2; t_2 + \tau)$ одинаковой длительности и любого k числа событий потока выполнено равенство

$$P_k(t_1, \tau) = P_k(t_2, \tau)$$

то есть соответствующая вероятность зависит только от величины временного интервала τ и не зависит от начальной точки интервала. Для стационарного потока параметр потока $\lambda(t)$ не зависит от времени, $\lambda(t) = \text{Const} = \lambda$.

1.6. Простейший поток событий

Пуассоновский поток, для которого выполняется условие стационарности, называется **простейшим** потоком событий. Следовательно, простейший поток - это стационарный, ординарный поток без последствия.

Для простейшего потока событий основная формула для расчета вероятности упрощается и равна

$$P_k(t_0, t_0 + t) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}$$

где λ - параметр простейшего потока. Видно, что результат не зависит от выбора начальной точки t_0 . В дальнейшем эту вероятность будет обозначаться $P_k(t)$.

График (рис.2) вероятности $P_0(t)$ - отсутствия событий потока в зависимости от интервала времени t (при $\lambda = 1.057$)

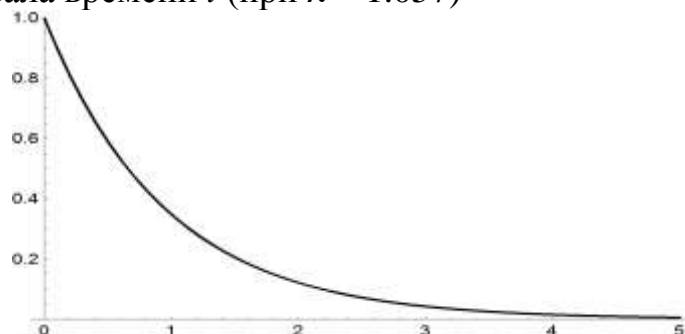


Рис. 2

График (ри.3) вероятности $P_1(t)$ – наличие ровно одного события потока в зависимости от интервала времени t (при $\lambda = 1.057$)

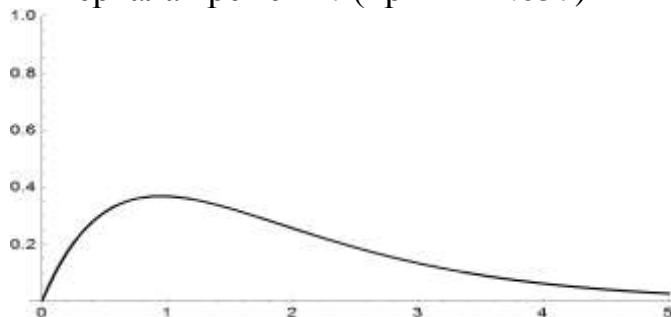


Рис. 3

1.7. Случайная величина - число событий потока на заданном временном интервале

Для пуассоновского потока событий введем случайную величину $X(t_0, t)$ равную числу событий потока на интервале $[t_0, t)$. Это дискретная случайная величина со следующим рядом распределения вероятностей, случайная величина принимает целочисленные значения $\{0, 1, 2, 3, \dots\}$ с вероятностями

$$P_k(t_0, t) = \frac{(\Lambda(t_0, t))^k}{k!} e^{-\Lambda(t_0, t)}, \text{ где } \Lambda(t_0, t) = \int_{t_0}^t \lambda(t') dt', k = 0, 1, 2, \dots$$

или

k	0	1	2	3	...
$P_k(t_0, t)$	$e^{-\Lambda[t_0, t]}$	$e^{-\Lambda[t_0, t]} \Lambda[t_0, t]$	$e^{-\Lambda[t_0, t]} \frac{1}{2!} \Lambda[t_0, t]^2$	$e^{-\Lambda[t_0, t]} \frac{1}{3!} \Lambda[t_0, t]^3$...

Для простейшего потока событий эти формулы существенно упрощаются, случайная величина X зависит только от длины временного интервала $\tau = t - t_0$ и не зависит от выбора начальной точки t_0 : $X(t_0, t) = X(t - t_0) = X(\tau)$ и закон распределения имеет следующий вид

k	0	1	2	3	4	...
$P_k(t_0, t)$	$e^{-\lambda t}$	$e^{-\lambda t} (\lambda t)$	$e^{-\lambda t} \frac{1}{2!} (\lambda t)^2$	$e^{-\lambda t} \frac{1}{3!} (\lambda t)^3$	$e^{-\lambda t} \frac{1}{4!} (\lambda t)^4$...

Во все случаях получается пуассоновский ряд распределения с параметром $\Lambda[t_0, t]$, причем для простейшего потока этот параметр равен $\Lambda[t_0, t] = \lambda(t - t_0) = \lambda * \tau$.

Из свойств пуассоновского закона распределения вытекает, что математическое ожидание случайной величины $X(t_0, t)$ равно $M[X(t_0, t)] = \Lambda[t_0, t]$. Для простейшего потока все результаты упрощаются - пуассоновский ряд

распределения: $P_k(t_0, t) = \frac{(\lambda \tau)^k}{k!} e^{-\lambda \tau}$, где $k \geq 0$, его числовые характеристики: среднее значение $-\lambda * \tau$, дисперсия $-\lambda * \tau$ средне-квадратичное отклонение $\sqrt{\lambda * \tau}$

1.8. Интенсивность потока событий.

Интенсивностью потока событий называется среднее число событий за единицу времени. Для нестационарного потока, соответствующее значение определяется как предел отношения среднего числа событий на временном интервале к величине этого временного интервала

$$\lim_{\Delta t \rightarrow 0} \frac{\Lambda(t_0, t_0 + \Delta t)}{\Delta t} = \mu(t_0)$$

Так как $\Lambda(t_0, t_0 + \Delta t) = \int_{t_0}^{t_0 + \Delta t} \lambda(t') dt' = \lambda(t') \Delta t$, то интенсивность пуассоновского потока событий равна параметру потока для любого значения t_0 : $\mu(t_0) = \lambda(t_0)$ а для простейшего потока - интенсивность равна параметру потока: $\mu(t_0) = \lambda$

1.9. Закон распределения интервала времени между последовательными событиями потока

Интервал времени $T_i = \theta_{i+1} - \theta_i$, между событием с номером i и событием с номером $(i+1)$ является случайной величиной. Получим закон распределения этой случайной величины для простейшего потока событий. Тогда в силу условия стационарности все случайные величины T_i имеют один и тот же закон распределения, не зависящий от номера i , а в силу отсутствия последействия все эти случайные величины статистически независимы между собой.

Рассмотрим вероятность $P(T \geq t)$ события, заключающегося в том, что интервал времени между последовательными событиями потока больше или равен величине t . Это событие эквивалентно событию, связанному с отсутствием событий потока на интервале длительности t , следовательно, эта вероятность равна

$$P(T \geq t) = P_0(t) = e^{-\lambda t}, t \geq 0$$

Интегральная функция распределения вероятностей случайной величины T имеет вид (рис. 4)

$$F_T(t) = P(T < t) = 1 - P(T \geq t) = 1 - e^{-\lambda t}, t \geq 0$$

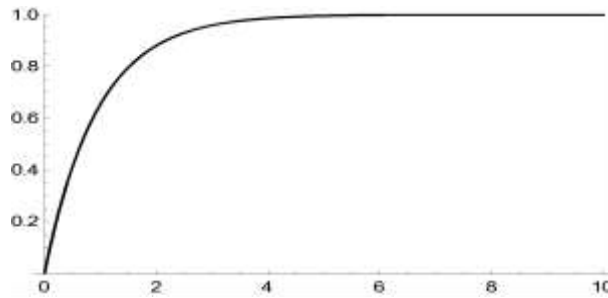


Рис. 4

а плотность вероятности (рис.5)

$$f_T(t) = \frac{d}{dt} F_T(t) = \frac{d}{dt} (1 - e^{-\lambda t}) = \lambda e^{-\lambda t}, t \geq 0$$

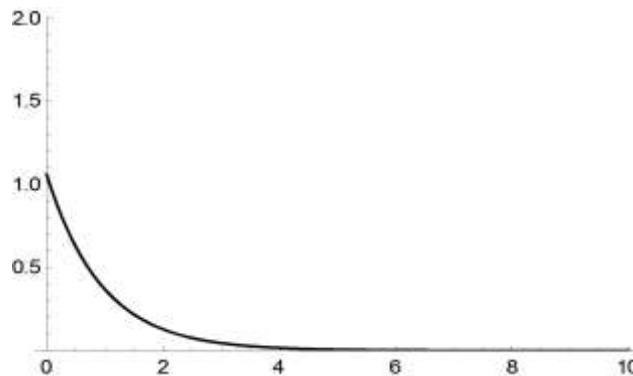


Рис. 5

Получили случайную величину, распределенную по показательному закону с параметром λ .

Числовые характеристики этого закона: среднее значение - $\frac{1}{\lambda}$, дисперсия - $\frac{1}{\lambda^2}$, средне-квадратичное отклонение - $\frac{1}{\lambda}$

1.10. Эквивалентное определение простейшего потока событий

Мы показали, что для стационарного, ординарного потока без последствия закон распределения интервала времени между последовательными событиями потока является показательным. Верно и обратное, если для стационарного, ординарного потока закон распределения интервала есть показательный, то в таком потоке отсутствует последствие, то есть поток будет простейшим.

Это обстоятельство является очень полезным при моделировании потоков событий, вместо трудно проверяемого условия отсутствия последствия можно моделировать независимые случайные величины для интервала времени между событиями с показательным законом распределения.

1.11. Объединение (наложение) пуассоновских потоков событий

Рассмотрим два пуассоновских потока событий интенсивности $\lambda_1(t)$ и

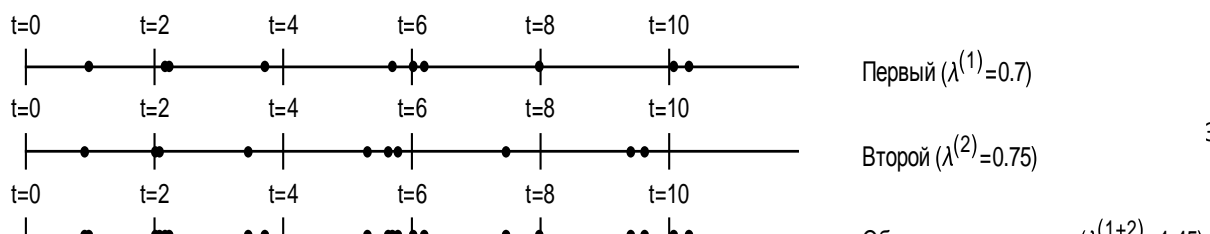


Рис. 6

$\lambda_2(t)$ соответственно. Объединением (наложением) двух потоков называется новый поток (рис.6), в который входят события как первого, так и второго потока.

Два потока называются независимыми потоками событий, если число событий первого потока, попадающих на некоторый временной интервал, статистически не зависит от числа событий второго потока на том же самом временном интервале.

Основной результат: объединение двух независимых пуассоновских потоков есть снова пуассоновский поток, интенсивность которого есть сумма интенсивностей слагаемых потоков. Докажем это для простейших потоков.

Рассмотрим некоторый временной интервал длительности t : $[t_0, t_0+t)$, и событие $A_k^{(1+2)}(t)$ - на данном временном интервале длительности t происходит k событий объединенного потока, которое может быть представлено в виде суммы произведений попарно-несовместных событий

$$A_k^{(1+2)}(t) = \sum_{i=0}^k A_{k-i}^{(1)}(t) * A_i^{(2)}(t)$$

В силу стационарности и независимости потоков событий получаем

$$\begin{aligned} P_k^{(1+2)}(t) &= P\left(A_k^{(1+2)}(t)\right) = \sum_{i=0}^k P\left(A_{k-i}^{(1)}(t)\right) * P\left(A_i^{(2)}(t)\right) = \\ &= \sum_{i=0}^k \frac{\left(\lambda^{(1)}t\right)^{k-i}}{(k-i)!} e^{-\lambda^{(1)}t} \frac{\left(\lambda^{(2)}t\right)^i}{i!} e^{-\lambda^{(2)}t} = \\ &= e^{-\lambda^{(1)}t} e^{-\lambda^{(2)}t} \frac{1}{k!} \sum_{i=0}^k \left(\lambda^{(1)}t\right)^{k-i} \left(\lambda^{(2)}t\right)^i \frac{k!}{(k-i)!i!} = \\ &= e^{-\left(\lambda^{(1)}t + \lambda^{(2)}t\right)} \frac{1}{k!} \left(\lambda^{(1)}t + \lambda^{(2)}t\right)^k = \\ &= e^{-\left(\lambda^{(1)} + \lambda^{(2)}\right)t} \frac{1}{k!} \left(\lambda^{(1)} + \lambda^{(2)}\right)^k t^k = e^{-\lambda^{(1+2)}t} \frac{1}{k!} \left(\lambda^{(1+2)}t\right)^k \end{aligned}$$

где $\lambda^{(1+2)} = \lambda^{(1)} + \lambda^{(2)}$

Использована формула бинома Ньютона. Для числа событий объединенного потока получился ряд Пуассона. Полученный поток является ординарным и закон распределения интервала времени между событиями объединенного потока является показательным. Следовательно, объединенный поток также будет простейшим, а его интенсивность равна сумме интенсивностей слагаемых $\lambda^{(1+2)} = \lambda^{(1)} + \lambda^{(2)}$.

1.12. Разрежение пуассоновского потока событий

Предположим, что исходный поток подвергается следующему преобразованию - относительно каждого из событий потока по определенному

правилу принимается решение, либо это событие остается в потоке, либо это событие удаляется из потока. Таким образом в новый поток входит только часть событий исходного потока. Если правило удаления событий является детерминированным, например, удаляется каждое второе событие или, скажем, третье, то даже если исходный поток является пуассоновским, новый поток пуассоновским не будет. Такие потоки называются потоками Эрланга. Подробности обсудим дальше.

Предположим, что процесс удаления событий является случайным, например, следующим. Каждое событие потока независимо от остальных событий либо остается в потоке с некоторой фиксированной вероятностью p , либо удаляется из потока с вероятностью $q = 1 - p$.

При случайном разрежении пуассоновского потока событий (рис. 7) результирующий поток останется пуассоновским. Покажем это. Очевидно, что разреженный поток будет ординарным. Для проверки отсутствия последствия вычислим вероятности появления k событий разреженного потока на временном интервале $[t_0, t) - \tilde{A}_k(t_0, t)$.

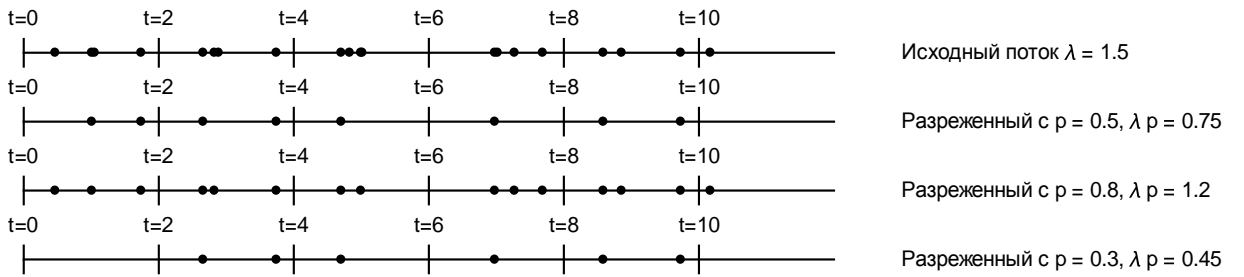


Рис. 7

Можно связать это событие с событиями исходного потока, исходный поток может содержать любое число событий потока $(k + i)$, причем любые i событий должны быть удалены. Следовательно, событие $\tilde{A}_k(t_0, t)$ равно сумме произведения попарно-несовместных событий $\tilde{A}_{k+i}(t_0, t)$ и B_{k+i}^i , событие B_{k+i}^i означает, что из $(k + i)$ событий потока должно быть удалено ровно i произвольных событий

$$\tilde{A}_k(t_0, t) = \sum_{i=0}^{\infty} A_{k+i}(t_0, t) * B_{k+i}^i$$

Если переходить к вероятностям, то получим

$$\begin{aligned} \tilde{P}_k(t_0, t) &= \sum_{i=0}^{\infty} P(A_{k+i}(t_0, t)) * P(B_{k+i}^i) = \\ &= \sum_{i=0}^{\infty} \frac{(\Lambda(t_0, t))^{k+i}}{(k+i)!} e^{-\Lambda(t_0, t)} C_{k+i}^i p^{(k+i)-i} q^i = \\ &= \frac{(\Lambda(t_0, t))^k}{k!} e^{-\Lambda(t_0, t)} p^k \sum_{i=0}^{\infty} \frac{(\Lambda(t_0, t))^i k! (k+i)!}{(k+i)! i! k!} q^i = \end{aligned}$$

$$\begin{aligned}
&= \frac{(\Lambda(t_0, t))^k}{k!} e^{-\Lambda(t_0, t)} p^k \sum_{i=0}^{\infty} \frac{(\Lambda(t_0, t))^i}{i!} q^i = \\
&= \frac{(\Lambda(t_0, t))^k}{k!} e^{-\Lambda(t_0, t)} p^k e^{\Lambda(t_0, t)q} = \frac{(\Lambda(t_0, t) p)^k}{k!} e^{-\Lambda(t_0, t)(1-q)} = \\
&= \frac{(\Lambda(t_0, t) p)^k}{k!} e^{-\Lambda(t_0, t)p}
\end{aligned}$$

Полученная величина означает, что разреженный поток событий представляет собой также пуассоновский поток событий с интенсивностью

$$\tilde{\lambda}(t) = \lambda(t) * p$$

2. Имитационное моделирование (метод Монте-Карло)

При моделировании потока событий и различных его характеристик можно использовать методы имитационного моделирования (метод Монте-Карло). При обычном статистическом моделировании необходимо иметь выборку, связанную с моделированием большого набора реализаций потока событий. Такой подход связан с большими вычислительными трудностями. Поэтому более экономичным является подход, связанный с имитационным моделированием.

При имитационном моделировании используется не выборка из многих потоков, а одна реализация потока событий, рассмотренная на достаточно большом временном интервале. Вместо статистического усреднения по множеству реализаций используется усреднение по времени для одной единственной реализации.

В теории для эргодических Марковских процессов доказано, что усреднение по множеству реализаций и усреднение по времени приводит практически к одним и тем же результатам.

Замечание о методе Монте-Карло. Моделирование производится с использованием датчика псевдослучайных чисел, в Wolfram Mathematica для этого используется функции типа Random. При этом возникает проблема воспроизведения результатов. Для воспроизведения случайных результатов нужно использовать задание некоторого произвольного затравочного значения для датчика псевдослучайных чисел с помощью функции SeedRandom. Во всех примерах в качестве затравочного выбрано значение 1234567.

2.1. Моделирование простейшего потока событий

Моделирование потока из $n = 1000$ событий с параметром $\lambda = 0.2$. Начальный момент времени 0. Среднее длина интервала времени между событиями потока равна $1/\lambda = 5$.

Основная идея моделирования простейшего потока событий заданной интенсивности λ заключается в предварительном моделировании интервалов времени между последовательными событиями потока. Для простейшего потока интервалы времени - независимые случайные величины, распределенные по показательному закону с параметром λ . Для моделирования таких случайных величин можно использовать стандартный датчик псевдослучайных чисел RandomReal, с параметром ExponentialDistribution[λ]. В качестве примера рассмотрим моделирование 1000 интервалов времени с параметром $\lambda = 0.2$

Пример кода

$\lambda = 0.2;$

$n = 1000;$

SeedRandom[1234567]

`intervals = Array[RandomReal[ExponentialDistribution[λ]]&, n];`

`intervals[[1;;10]]`

Результат вычисления для длин интервалов (первые 10 элементов)
{3.4127, 4.1458, 0.2185, 5.2168, 6.9447, 1.1533, 0.5640, 6.2969, 7.3062, 0.8273}

Сами моменты наступления событий получаются суммированием полученных интервалов.

Пример кода

`times = Accumulate[intervals];`

`times[[1;;10]]`

Результат вычисления для моментов осуществления событий потока (первые 10 элементов)

{3.41277, 7.55861, 7.77715, 12.994, 19.9387, 21.0921, 21.6561, 27.9531, 35.2594, 36.0867}

2.2. Проверка качества моделирования

Проверка качества моделирования может осуществляться в простейшем варианте вычислением выборочных значений длины интервала и его дисперсии и сравнением его с теоретическими значениями $1/\lambda$ и $1/\lambda^2$.

Пример кода

`{mean = Mean[intervals], 1 / λ } (* в теории 1 / λ *)`

`{Variance[intervals], (1 / λ) ^ 2} (* в теории (1 / λ)2 *)`

`{std = StandardDeviation[intervals], 1 / λ } (* в теории 1 / λ *)`

Результаты моделирования:

выборочное и теоретическое среднее: {5.03582, 5.}

выборочная и теоретическая дисперсия: {24.0028, 25.}

выборочное и теоретическое средне квадратичное отклонение: {4.89927, 5.}

Сравнивая полученные значения, можно говорить о неплохом соответствии эксперимента и теории.

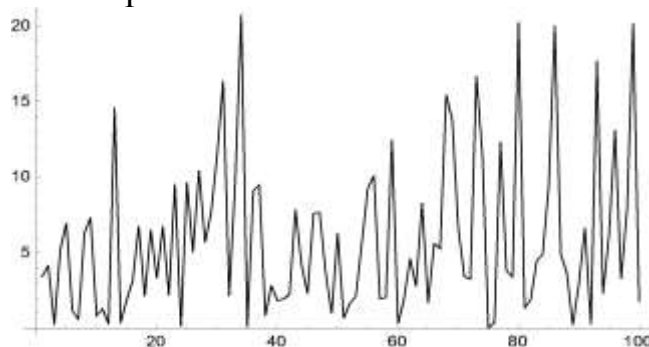


Рис. 8

На рис. 8 представлена наглядная иллюстрация длин интервалов между событиями потока (представлены первые 100 интервалов)

2.3. Построение доверительного интервала

Построение доверительного интервала для оценки среднего значения длины интервала. Для заданной доверительной вероятности $pdov$ строится доверительный интервал $\{dovinterval[[1]], dovinterval[[2]]\}$ и проверяется, попало ли выборочное среднее в построенный интервал.

Пример кода

$pdov = 0.95$

$quantile = \text{Quantile}[\text{StudentTDistribution}[nvyb - 1], (1 + pdov) / 2]$

$dovinterval = \{mean - \frac{std}{\sqrt{nvyb}} quantile, mean + \frac{std}{\sqrt{nvyb}} quantile\}$

$dovinterval[[1]] < mean < dovinterval[[2]]$

Доверительный интервал $\{4.7318, 5.3399\}$, а $mean = 5.0358$, что обеспечивает попадание этого значения в интервал.

2.4. Проверка гипотезы об законе распределения по критерию Пирсона

Проверка гипотезы об экспоненциальном законе распределения для длины временного интервала по критерию Пирсона. Определяется размах выборки: min и max , число разрядов $nbins$ и определяется количество элементов выборки, попавших в каждый из разрядов $experiment$.

Пример кода.

$min = \text{Min}[intervals];$

$max = \text{Max}[intervals];$

$ngran = 65;$

$nbins = 15;$

$del = ngran / nbins$

$experiment = \text{Table}[\text{Length}[\text{Select}[intervals, ((k - 1)del < \# \leq kdel) \&]],$

$\{k, 1, nbins\}] / nvyb / N$

$\text{Total}[experiment] (* \text{ контроль } *)$

$experimentmod = experiment[[1; 9]];$

$experimentmod[[9]] = \text{Total}[experiment[[9; 15]]];$

$experimentmod$

$\text{Total}[experimentmod] (* \text{ контроль } *)$

Результаты расчета: длина разряда 4.3333, частота попадания элементов выборки в каждый из разрядов.

$\{0.558, 0.279, 0.103, 0.03, 0.023, 0.004, 0.001, 0.001, 0., 0., 0., 0., 0., 0., 0.001\}$

Видно, что можно оставить только первые 9 разрядов
 {0.558, 0.279, 0.103, 0.03, 0.023, 0.004, 0.001, 0.001, 0.001}

Необходимо рассчитать теоретические вероятности для этих разрядов, модифицируя последний разряд (разряд 9) для получения суммарной вероятности 1.

Пример кода.

```
teor = Table[expCDF[kdel] - expCDF[(k - 1)del], {k, 1, nbins}];
teoret = teor[[1;;9]];
teoret[[-1]] = 1 - Total[teor[[1;;8]]];
```

teoret

Полученные теоретические значения частот:

{0.57965, 0.243656, 0.102421, 0.0430527, 0.0180972, 0.00760716, 0.00319767, 0.00134414, 0.000974746}

Сравнение на графике (рис. 9) экспериментальных и теоретических значений. Теоретические значения – сплошная линия, результаты моделирования – выделенные точки.

Пример кода.

```
list1 = ListPlot[teor, PlotRange -> {0, 0.6}, Joined -> True]
list2 = ListPlot[{experimentmod}, PlotRange -> {0, 0.6},
  PlotStyle -> {{Black, PointSize[0.03]}]}]
Show[list1, list2]
```

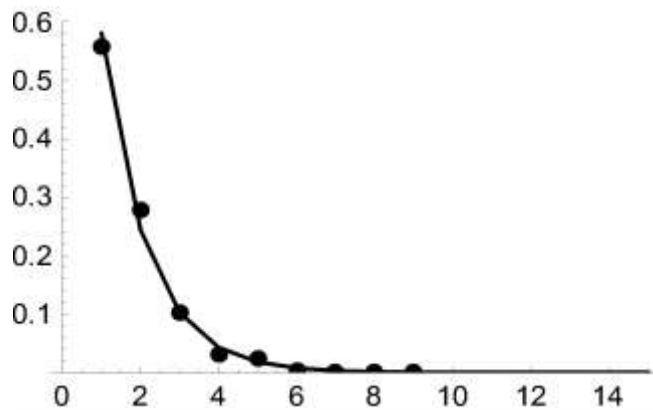


Рис. 9

Вычислим выборочное значение критерия Пирсона χ^2

Пример кода.

```
exphisquare = Total[nvyb(experimentmod - teoret)^2 / teoret]
```

Полученное значение 14.5339

Сравним его с критическим значением, вычисленным через квантиль экспоненциального закона

Пример кода.

```
Quantile[ChiSquareDistribution[9 - 1], pdov]
```

Критическое значение равно 15.5073, что больше выборочного, следовательно, по критерию Пирсона гипотеза об экспоненциальном законе принимается

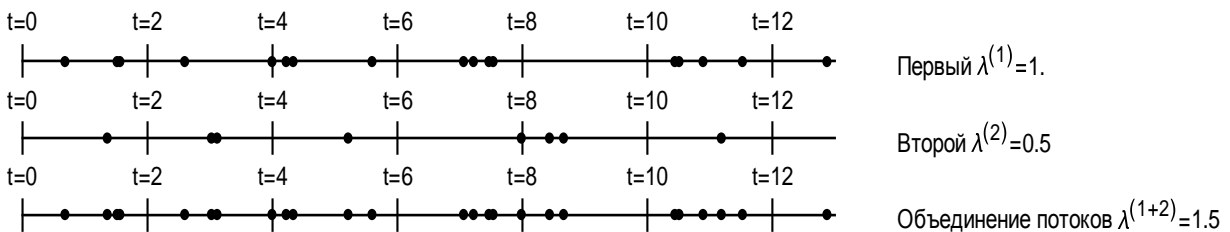
2.5. Моделирование объединения (наложения) пуассоновских потоков событий

Рассмотрим два простейших потока, первый интенсивности $\lambda_1 = 1$ и второй интенсивности $\lambda_2 = 0.5$. Смоделируем $n = 1000$ событий каждого из потоков. Получим два набора времен осуществления событий. Объединим эти два набора (рис. 10) и отсортируем объединенный набор в порядке возрастания, результат и есть последовательность времен событий объединенного потока.

Пример кода

```
t0 = 13; (*интервал для демонстрации*)
n = 1000;
λ1 = 1;
SeedRandom[1234567]
intervals1 = Array[RandomReal[ExponentialDistribution[λ1]]&, n];
times1 = Accumulate[intervals1]; (*первый поток*)
timesMax = times1[[-1]];
λ2 = 0.5;
SeedRandom[1234567]
intervals2 = Array[RandomReal[ExponentialDistribution[λ2]]&, n];
times2 = Accumulate[intervals2];
times2 = Select[times2, # ≤ timesMax &]; (*второй поток*)
timesall = Sort[times1 ~ Join ~ times2]; (*объединенный поток*)
Select[times1, # < t0 &]; (*для демонстрации*)
Select[times2, # < t0 &]; (*для демонстрации*)
Select[timesall, # < t0 &]; (*для демонстрации*)
```

Реализация событий потока.



Оценка длин интервалов по полной выборке.

Рис. 10

Оценка длины интервала для первого потока 1.00716. Теоретическое значение: 1.

Оценка длины интервала для второго потока 2.01433. Теоретическое значение: 2.

Оценка длины интервала для объединения потоков 0.680056. Теоретическое значение: 0.666667.

2.6. Моделирование разряжения пуассоновского потока событий

Рассмотрим простейший поток интенсивности $\lambda = 1.5$. Смоделируем $n = 1000$ событий потоков. Потом этот поток разрядим с вероятностью $p = 0.5$.

Пример кода

```
n = 1000;
```

```
 $\lambda = 1.5$ ;
```

```
SeedRandom[1234567]
```

```
intervals1 = Array[RandomReal[ExponentialDistribution[ $\lambda$ ]]&, n];
```

```
times1 = Accumulate @ intervals1; (*исходный поток*)
```

```
probability = 0.5;
```

```
SeedRandom[1234567]
```

```
BlockRandom[z = If[RandomReal[] < probability, 1, 2]];
```

```
timesres = GatherBy[times1, (RandomReal[] < probability)&][[z]];
```

Иллюстрация к результату расчета (рис. 11), на первой оси изображены события исходного потока $times1$, на второй оси события разреженного потока $timesres$ с вероятностью $p = 0.5$.

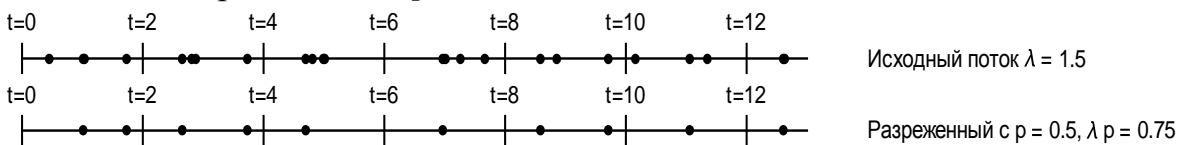


Рис. 11

Оценка длины интервала по полной выборке 1.2639. Теоретическое значение 1.3333.

2.7. Моделирования числа событий потока на заданном временном интервале.

Рассмотрим простейший поток с параметром $\lambda = 0.2$. Выберем временной интервал длины в 10 единиц, $t_0 = 10$. Согласно теории, количество событий потока на этом интервале распределено по закону Пуассона с параметром $\lambda t_0 = 2$. Построим график (рис. 12) ряда распределения для этого закона.

Пример кода.

```

pois = PoissonDistribution[λt0];
DiscretePlot[PDF[pois, k], {k, 0, 10}, PlotRange → {0, 0.5},
PlotStyle → PointSize[Medium]]

```

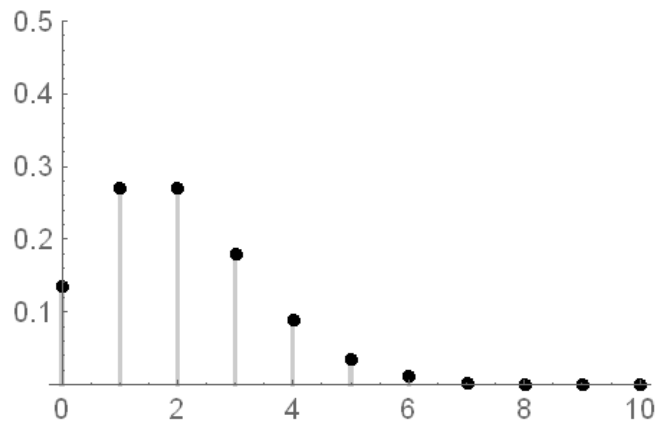


Рис. 12

При моделировании вместо среднего по ансамблю будем использовать среднее по времени. Весь интервал времени от 0 до $\text{Max}[potok]$ делим на равные отрезки длины $t0$. Всего таких отрезков $nmax$.

Вычислим количества событий потока, попавших в каждый из заданных отрезков длины $t0$.

Пример кода.

```

sob = Table[Length @ Select[potok, ((k - 1)t0 < # < t0k) &], {k, 1, nmax}];
sob[[1;;10]]

```

```

Max[sob]

```

Приведем ответ для первых 10 отрезков {3, 2, 3, 4, 0, 4, 2, 2, 2, 2}, максимальное значение числа событий равно 7.

Вычислим частоту каждого из значений от 0 до 7.

Пример кода.

```

coun = Count[sob, #] & / @ Range[0, 7]

```

```

Total @ coun

```

```

counfr = coun / nmax / N

```

Полученный результат: абсолютные значения частот {61, 137, 146, 90, 47, 11, 7, 1}, общее количество событий потока – 500, относительная частота каждого из возможных значений {0.122, 0.274, 0.292, 0.18, 0.094, 0.022, 0.014, 0.002}

Пример кода.

```

d1 = DiscretePlot[PDF[pois, k], {k, 0, 9}, PlotRange → {0, 0.4}, Joined → True];

```

```

d1 = DiscretePlot[counfr[[k + 1]], {k, 0, 9}, PlotRange → {0, 0.4}];

```

```

Show[d1, d2]

```

Сравнение теоретических и экспериментальных значений на графике (рис. 13) (сплошная линия – теория, точки - эксперимент).

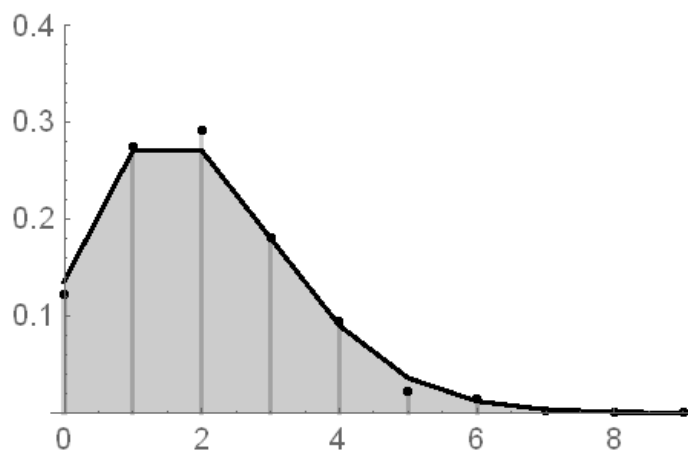


Рис. 13

Оценка среднего числа событий потока, сравнение теоретического и экспериментального значений, а также оценка средне квадратичного отклонения.

Пример кода.

```
{λt0, meansr = Mean[sob]} // N
```

```
stdsr = StandardDeviation[sob] // N
```

Полученные результаты, сравнение средних - {2., 1.982}, средне квадратичное отклонение 1.33913.

Доверительный интервал для среднего значения.

Пример кода.

```
pdov = 0.95;
```

```
quantile = Quantile[StudentTDistribution[nmax - 1], (1 + pdov) / 2]
```

$$dovintsr = \left\{ meansr - \frac{stdsr}{\sqrt{nmax}} quantile, meansr + \frac{stdsr}{\sqrt{nmax}} quantile \right\}$$

Для доверительного интервала с доверительной вероятностью 0.95 получен интервал {1.86434, 2.09966}, этот интервал накрывает полученное ранее значение 1.982.

2.8. Проверка гипотезы о Пуассоновском законе распределения

Проверка гипотезы о Пуассоновском законе распределения для числа событий потока на интервале длительности t0 по критерию Пирсона. Для этого необходимо определить выборочное значение критерия хи-квадрат и сравнить его с критическим значением.

Пример кода.

```
xivyb = nmax * Total @ Table[(counfr[[k]] - PDF[pois, k - 1])^2 / PDF[pois, k - 1],  
{k, 1, 15}]
```

```
Quantile[ChiSquareDistribution[10 - 1], pdov]
```

Выборочное значение хи-квадрат - 5.35786, что меньше критического значения - 16.919. Следовательно, гипотеза о распределении Пуассона принимается.

2.9. Задача о характеристиках случайного временного интервала

На оси времени случайным образом выбирается временная точка. Она попадает на некоторый интервал между двумя последовательными событиями потока. Рассматривает статистические характеристики, связанные с этим случайным интервалом, длина интервала, длина интервала времени после последнего события потока (время опоздания) и длина интервала времени до следующего события потока (время ожидания).

Случайно выбирается момент времени t_1 , и для него определяется время до следующего события потока (время ожидания), время после предыдущего события потока (время опоздания) и длина интервала, на который попадает случайный момент времени. Введем обозначения *rightn* - номер правой границы интервала, на который попадает выбранная точка. Переменные *lgr* и *rgr*, соответственно, левая и правая временная граница отрезка, на который попадает t_1 .

На рис. 14 указаны события потока с номерами 467 и 488, между которыми попала точка с абсциссой $t_1 = 508.9$, сам интервал [507.1; 509.5]. Длина интервала времени после последнего события потока равна 1.84, длина интервала времени до следующего события потока равна 0.52, а длина самого интервала времени 2.36.

Пример кода

```
SeedRandom[1234567]
expon[λ _]:= ExponentialDistribution[λ]
λ1 = 1
intervals4 = RandomReal[expon[λ1], nvyb];
potok4 = {0} ⓓ Accumulate[intervals4];
potok4[[1;;5]]
nsob = Length[potok4]
max = Max[potok4]
SeedRandom[1234567]
t1 = RandomReal[max]
rightn = Nearest[potok4 → Automatic, t1,
DistanceFunction → (If[#1 ≤ #2, #2 - #1, 10000] &)] [[1]]
{lgr = potok4[[rightn - 1]], rgr = potok4[[rightn]]}
{lgr - t1, t1 - lgr, rgr - lgr}
(*Иллюстрация*)
```

Graphics

```
[
  {Text["potok[" <> ToString[rightn - 1] <> "]" = " <> ToString[lgr],{lgr,0.1}],
  Line[{{lgr - 0.1,0},{rgr + 0.1,0}}],
  Point[{{lgr,0},{rgr,0}}],
  Text["potok[" <> ToString[rightn] <> "]" = " <> ToString[rgr],{rgr,0.1}],
  Red,PointSize[Medium],
  Point[{t1,0}],
  Text["t1 = " <> ToString[t1],{t1,0.1}]]
]
```

Результат моделирования

potok[487]=507.1 t1=508.9 potok[488]=509.5

Рис. 14

Вычислим статистические характеристики полученных интервалов. Для этого смоделируем *nvyb* значений случайно выбранных точек. Для каждой из них вычислим характеристики вычисленного интервала.

Пример кода

```
SeedRandom[1234567]
result = Table[
  t1 = RandomReal[max];
  rightn =
  Nearest[potok4 → Automatic,t1,
  DistanceFunction → (If[#1 ≤ #2,#2 - #1,10000]&)] [[1]];
  {potok4[[rightn]] - t1,t1 - potok4[[rightn - 1]], potok4[[rightn]] - potok4[[rightn - 1]]},
  {nvyb}];
{2 / λ1, Mean[result[[All,3]]]}
{1 / λ1, Mean[result[[All,1]]]}
{1 / λ1, Mean[result[[All,2]]]}
```

Сравнение теоретических результатов с результатами эксперимента. Теоретическое математическое ожидание равно, для длины интервала - $2/\lambda$, для длин интервала ожидания и опоздания - $1/\lambda$. Результат моделирования.

	Тео- рия	Экспери- мент
Длина интервала	2	1.961
Время ожидания	1	1.015
Время опоздания	1	0.945

3. Цепи Маркова

3.1. Случайный процесс в системе с дискретным числом состояний

Рассмотрим некоторую физическую или экономическую систему S , имеющую дискретное число возможных состояний: $\{S_1, S_2, S_3, \dots, S_m\}$. Число m таких состояний будем считать конечным или бесконечным, но счетным. В произвольный момент времени состояние системы S может быть полностью описано указанием того состояния, в котором она в данный момент времени находится. С течением времени система S может переходить из одного состояния в другое.

Если такие переходы происходят в случайные моменты времени или переходы происходят в заранее заданные моменты, но новое состояние, в которое система S переходит, выбирается случайно, или осуществляется одновременно переход в случайное время в случайно выбранное состояние, то говорят о **случайном процессе, протекающем в системе S** . Предполагается, что смена состояния происходит мгновенно, никаких переходных состояний нет

3.2. Марковское свойство

Рассмотрим систему S в некоторый момент времени $t = t_0$. По отношению к этому моменту времени, которое условно будем называть “настоящее”, состояния системы S при $t < t_0$ естественно назвать “прошлым”, а состояние системы S при $t > t_0$ - “будущим”. Говорят, что случайный процесс, протекающий в системе обладает Марковским свойством, если состояния системы в “будущем” (при $t > t_0$) не зависят от “прошлого” (при $t < t_0$). Или, по-другому, состояние системы S при $t > t_0$ зависит только от состояния системы при $t = t_0$ и не зависит от того, какие состояния у системы S были при $t < t_0$, то есть от того, как и каким образом система пришла в данное состояние при $t = t_0$. Марковское свойство случайного процесса, протекающего в системе S , есть другое выражение понятия отсутствие последствия для потоков событий. Это определение является неформальным и в каждом конкретном случае подлежит уточнению.

3.3. Цепи Маркова

Пусть переходы системы S из одного состояния в другое происходят в неслучайные моменты времени, которые без ограничения общности можно обозначить $t = k, k = 0, 1, 2, 3, \dots$. Предположим также, что число состояний конечно $\{S_1, S_2, S_3, \dots, S_m\}, m < \infty$. Введем случайный процесс $X(k)$ определяющий состояние системы S в момент времени $t = k$. Случайное событие $\{X(k) = S_i\}$ означает, что в момент времени $t = k$ система находится в состоянии $S_i, i = 1, 2, 3, \dots, m$. Так как время дискретно, то можно рассмотреть состояние системы S в следующий момент времени $t = (k+1)$, событие $\{X(k+1) = S_i\}$. Если состояние системы $X(k+1)$ зависит только от состояния системы S в момент времени $t = k$ и статистически не зависит

от состояний в моменты времени $t = 0, 1, \dots, (k-1)$, то говорят о Марковском свойстве процесса, протекающего в системе. Такие случайные процессы принято называть цепями Маркова.

В силу Марковского свойства имеем следующее равенство

$$P(\{X(k+1) = S_j\} / \{X(k) = S_i, X(k-1) = S_{i_{k-1}}, X(k-2) = S_{i_{k-2}}, \dots, X(0) = S_{i_0}\}) = P(\{X(k+1) = S_j\} / \{X(k) = S_i\})$$

Эту условную вероятность принято называть **переходной вероятностью** и обозначать $p_{i,j}(k)$:

$$p_{i,j}(k) = P(\{X(k+1) = S_j\} / \{X(k) = S_i\})$$

3.4. Матрица переходных вероятностей (матрица перехода)

Для описания всех возможных переходов из одного состояния в другое состояние можно ввести квадратную матрицу размера $m \times m$, **матрицу переходных вероятностей** P или матрицу перехода:

$$P(k) = \begin{pmatrix} p_{1,1}(k) & \dots & p_{1,m}(k) \\ p_{2,1}(k) & \dots & p_{2,m}(k) \\ \dots & \dots & \dots \\ p_{m,1}(k) & \dots & p_{m,m}(k) \end{pmatrix}$$

Каждая строка этой матрицы с номером i описывает условные вероятности перехода из состояния S_i во все другие состояния $\{S_1, S_2, S_3, \dots, S_m\}$. Так первая строка описывает условные вероятности перехода из состояния S_1 . Пусть известно, что система S в момент времени $t = k$ находится в состоянии S_1 . Тогда $p_{1,1}(k)$ - вероятность того, что она и в следующий момент времени $t = (k+1)$ останется в этом состоянии, $p_{1,2}(k)$ - вероятность того, что в следующий момент времени система S перейдет в состояние S_2 и так далее. Так как в каждой из строк перечислены все возможные состояния, а переход в какое-то состояние обязательно происходит (включая, что система останется в том же состоянии), то сумма вероятностей каждой из строк должна равняться 1:

$$\sum_{j=1}^m p_{i,j}(k) = 1$$

для любого момента времени $t = k$ и для любого состояния S_i в этот момент времени. Такие матрицы принято называть **стохастическими матрицами**.

Если условная вероятность перехода из одного состояния в другое $p_{i,j}(k)$ не зависит от времени $t = k$ для любых i и j , то есть $p_{i,j}(k) = p_{i,j}$, то говорят об однородной цепи Маркова. Матрица $P(k)$ становится постоянной, не зависящей от времени $t = k$.

3.5. Матрица перехода за n шагов

Матрица переходных вероятностей P описывает переходы из одного состояния в другое за один шаг, то есть за один такт времени. Рассмотрим теперь переход из состояния S_i в состояние S_j за два шага, то есть условную вероятность $P(\{X(k+2) = s_j\} / \{X(k) = s_i\})$. Так как в промежуточный момент времени $t = (k+1)$ система S должна находиться в одном из возможных состояний, то в силу формулы полной вероятности, получим

$$\begin{aligned} P(\{X(k+2) = s_j\} / \{X(k) = s_i\}) &= \\ \sum_{l=1}^m P(\{X(k+2) = s_j\} / \{X(k+1) = s_l\}) P(\{X(k+1) = s_l\} / \{X(k) = s_i\}) &= \\ \sum_{l=1}^m p_{l,j}(k+1) * p_{i,l}(k) &= \sum_{l=1}^m p_{i,l}(k) * p_{l,j}(k+1) \end{aligned}$$

Эту условную вероятность обозначим $p_{i,j}^{(2)}(k)$:

$$p_{i,j}^{(2)}(k) = P(\{X(k+2) = s_j\} / \{X(k) = s_i\})$$

и соответствующую матрицу переходных вероятностей за два шага

$$P^{(2)}(k) = \begin{pmatrix} p_{1,1}^{(2)}(k) & \dots & p_{1,m}^{(2)}(k) \\ p_{2,1}^{(2)}(k) & \dots & p_{2,m}^{(2)}(k) \\ \dots & \dots & \dots \\ p_{m,1}^{(2)}(k) & \dots & p_{m,m}^{(2)}(k) \end{pmatrix} = P(k) * P(k+1)$$

Для однородной цепи Маркова формула упрощается

$$P^{(2)} = P * P = P^2$$

то есть для однородной цепи Маркова матрица перехода за два шага есть квадрат матрицы перехода за один шаг.

Аналогично можно получить матрицу перехода за n шагов. Для однородной цепи Маркова получается следующая простая формула

$$P^{(n)} = P * P^{(n-1)} = P^n, P^{(1)} = P$$

3.6. Распределение вероятностей состояний системы

Рассмотрим состояние системы S в момент времени $t = k$. Либо состояние системы в этот момент времени точно известно $X(k) = s_{i_0}$, либо точной информации нет, тогда можно указать некоторое распределение вероятностей состояний системы в этот момент

$$\vec{w}(k) = \{P(X(k) = s_1), P(X(k) = s_2), \dots, P(X(k) = s_m)\} = \{w_1(k), w_2(k), \dots, w_m(k)\},$$

$$\sum_{i=1}^m w_i(k) = 1$$

Если состояние точно известно, $X(k) = S_{i_0}$, то такую ситуацию можно описать следующим образом, $w_{i_0}(k) = 1$, остальные значения равны 0.

Если распределение вероятностей состояний в некоторый момент времени $t = k$ задано, то можно получить распределение вероятностей состояний системы и в любой другой момент времени $t = k+n$, $n > 0$. Действительно, используя формулу полной вероятности,

$$w_j(k+1) = P(X(k+1) = s_j) = \sum_{i=1}^m P(X(k+1) = s_j / X(k) = s_i) P(X(k) = s_i) = \sum_{i=1}^m w_i(k) * p_{i,j}(k)$$

или, в матричных обозначениях,

$$\vec{w}(k+1) = \vec{w}(k) * P(k)$$

Аналогично получаем

$$\vec{w}(k+n) = \vec{w}(k) * P^{(n)}(k)$$

Для однородной цепи Маркова получаем

$$\vec{w}(k+n) = \vec{w}(k) * P^n(k)$$

3.7. Изображение цепи Маркова в виде ориентированного графа

Цепь Маркова можно представить в виде ориентированного графа (рис. 15), где вершины графа - состояния системы, а направленные ребра - возможные переходы из одного состояния в другое, причем каждому ребру приписывается вес, равный вероятности перехода для этой пары состояний. Изобразим в виде графа цепь Маркова с тремя состояниями $\{S_1, S_2, S_3\}$ и матрицей переходных вероятностей P :

$$P = \begin{pmatrix} 0.1 & 0.3 & 0.6 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix}$$

Наличие направленной дуги означает возможный переход из состояния в состояние, пометка у дуги равна соответствующей вероятности перехода.

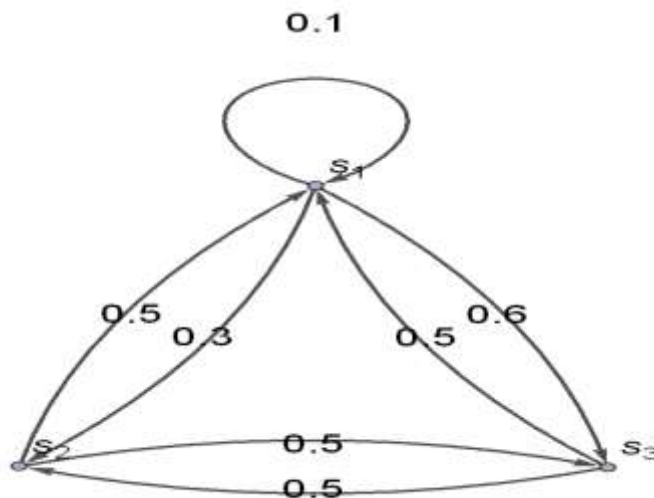


Рис. 15

3.8. Классификация состояний цепи Маркова

Основной вопрос, который представляет интерес для цепи Маркова, как ведут себя вероятности $p_{i,j}^{(n)}$, то есть условные вероятности перехода из состояния S_i в состояние S_j за n шагов. В некоторых условиях происходит стабилизация значений $p_{i,j}^{(n)}$ к некоторым значениям, не зависящим от начального распределения вероятностей w . Необходимо выяснить условия, при которых возникает такая стабилизация и определить сами предельные вероятности.

Оказывается, что это связано со структурой цепи Маркова, точнее зависит от свойств состояний этой цепи.

Первый термин **достижимость** состояний. Будем говорить, что состояние S_j **достижимо** из состояния S_i , если существует такой интервал времени n , что $p_{i,j}^{(n)} > 0$, то есть за n шагов можно с ненулевой вероятностью перейти из S_i в S_j . Для графа цепи Маркова это означает, что существует путь из вершины S_i в вершину S_j . Обозначим эту ситуацию следующим образом $S_i \Rightarrow S_j$. Это отношение является транзитивным, то есть, если $S_i \Rightarrow S_j$ и $S_j \Rightarrow S_l$, то $S_i \Rightarrow S_l$.

Состояния S_i и S_j называются **сообщающимися**, если одновременно $S_i \Rightarrow S_j$ и $S_j \Rightarrow S_i$.

Состояние S_i называется **существенным**, если всякое S_j , достижимое из S_i , является, одновременно, сообщаемым с S_i . То есть если $S_j \Rightarrow S_i$, то $S_i \Rightarrow S_j$. Состояние S_i называется **несущественным**, если существует состояние S_j достижимое из S_i , но не сообщаемое с S_i .

Рассмотрим пример. Матрица переходных вероятностей равна

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

а граф (рис. 16) имеет вид

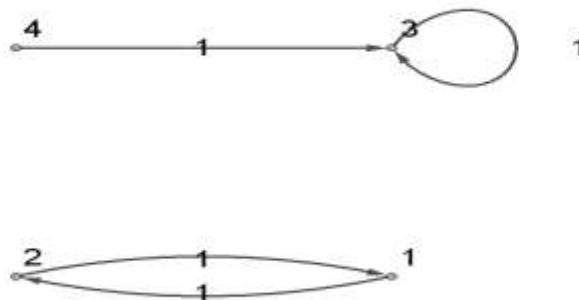


Рис. 16

Состояние 1 и 2 - сообщающиеся, состояния 1 и 2 - существенные, состояние 4 - несущественное, состояние 3 существенное, состояния 1 и 3 не сообщающиеся.

Любое несущественное состояние не достижимо из существенного. Пусть S_i - несущественное состояние, S_j - существенное и существует путь $S_j \Rightarrow S_i$. Так как S_i - несущественное состояние, то существует состояние S_k такое, что $S_i \Rightarrow S_k$, а обратное неверно, то есть пути из S_k в S_i - нет. Так как есть путь $S_j \Rightarrow S_i$ и $S_i \Rightarrow S_k$, то есть и путь $S_j \Rightarrow S_k$. Состояние S_j - существенное, следовательно есть путь $S_k \Rightarrow S_j$. А тогда есть путь $S_k \Rightarrow S_j \Rightarrow S_i$, то есть $S_k \Rightarrow S_i$. Пришли к противоречию. Следовательно, несущественное состояние не достижимо из существенного.

Теперь все множество состояний цепи Маркова можно разбить на два подмножества, одно из них - все несущественные состояния, а второе - все существенные состояния, причем элементы первого множества не достижимы из второго. Второе множество, множество существенных состояний, разобьем на отдельные классы, каждый из классов состоит из сообщающихся между собой состояний. Состояния из разных классов друг с другом не достижимы. Каждый такой класс называется **эргодическим** классом. Всякий эргодический класс представляет собой отдельную цепь Маркова, причем в этой цепи нет несущественных состояний. Такая цепь Маркова называется также **неразложимой**.

Разбиения на указанные множества и классы можно сформулировать в терминах матрицы перехода P . Состояния можно перенумеровать так, что матрица P будет иметь следующую структуру (рис. 17).

Здесь C представляет все несущественные состояния, возможно, это множество пустое. $D1, D2, \dots, Dl, l \geq 1$ - эргодические классы. Белые ячейки заполнены нулями. Серые ячейки представляют собой стохастические подматрицы.

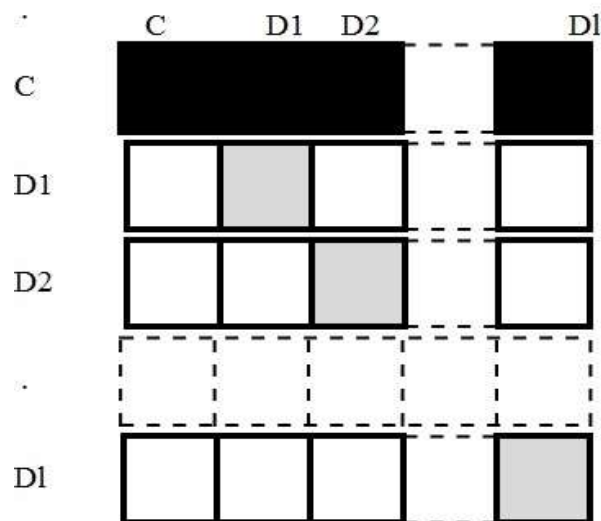


Рис. 17

Состояние S_i называется **периодическим** с периодом $q \geq 2$, если переход сам в себя, $S_i \rightarrow S_i$ возможен только в моменты времени кратные q , то есть при $t = q, 2q, 3q, 4q, \dots$, $p_{i,i}^{(kq)} > 0$, а для остальных времен $p_{i,i}^{(t)} = 0$.

В примере ниже рассмотрим состояние 2 и соответствующие вероятности перехода из 2 \rightarrow 2 за произвольное число шагов. Эта вероятность отлична от нуля только при $t = 3k$, при остальных значениях t вероятность равна 0. Следовательно, это состояние периодически с периодом 3.

$$P = \begin{pmatrix} 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Граф переходов (рис. 18) для этой цепи:

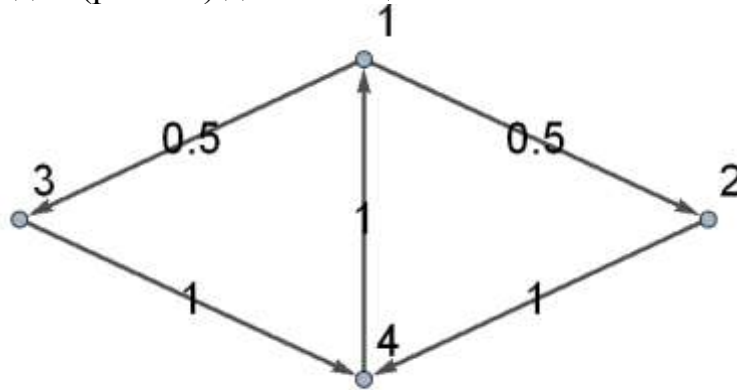


Рис. 18

Нетрудно заметить, что каждое состояние в этой цепи является периодическим с периодом 3.

Рассмотрим теперь конечную неразложимую Марковскую цепь. Все состояния составляют один эргодический класс. Верно следующее утверждение, если в такой цепи есть периодическое состояние, то и все остальные состояния то же периодические с тем же самым периодом. Приведенный выше пример дает иллюстрацию этого утверждения. Цепь Маркова, в которой все состояния периодические называется **периодической**. Все множество состояний S разбивается при этом на непересекающиеся подмножества $H_1, H_2, H_3, \dots, H_h$, называемыми циклическими подклассами. Выйдя из состояния, входящего в состав H_i , цепь за один шаг может перейти лишь в состояние H_{i+1} , если $i = h$, то вместо $(i+1)$ получаем 1. Таким образом, выйдя из состояния, входящего в состав H_i цепь за h шагов вернется в состояние, входящее в состав этого же подкласса H_i . И этот процесс будет периодически повторяться через $h, 2h, 3h, \dots$ шагов. Для рассмотренного выше примера множество состояний $S = \{1, 2, 3, 4\}$ разбивается на 3 циклических подкласса: $H_1 = \{1\}$, $H_2 = \{2, 3\}$, $H_3 = \{4\}$. Переходы при этом имеют следующий вид: $H_1 \Rightarrow H_2 \Rightarrow H_3 \Rightarrow H_1$.

Матрицу переходов циклической цепи (возможно при переобозначении состояний) имеет следующий вид (рис. 19). Белые клетки заполнены нулями, а каждая серая клетка представляет собой стохастическую подматрицу.

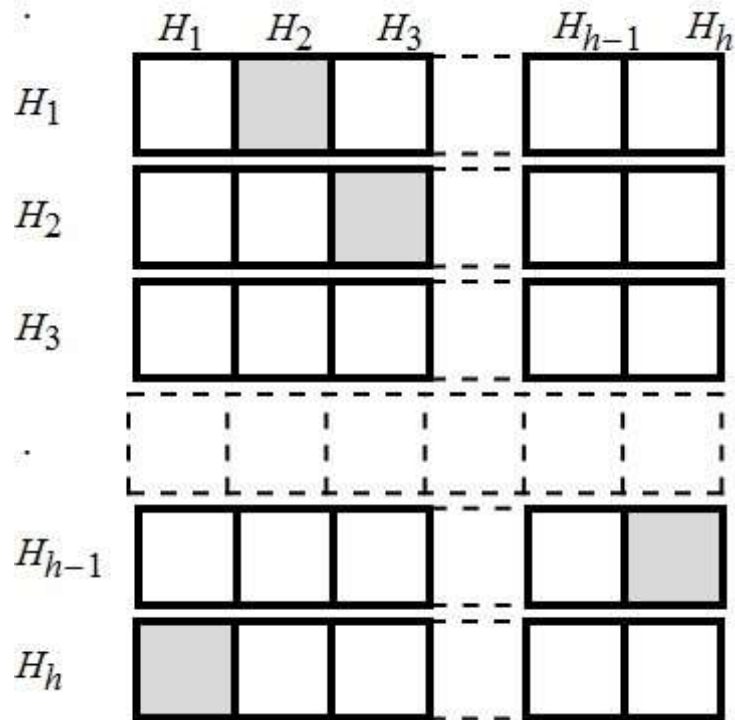


Рис. 19

Для рассмотренного выше примера разбиение матрицы переходов на циклические подклассы содержит следующие блоки для матрицы P (рис. 20)

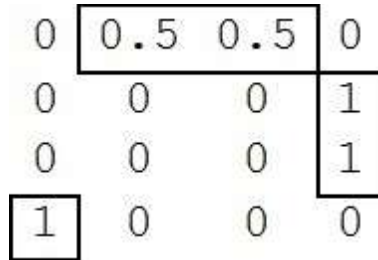


Рис. 20

и для матрицы P^3 (рис. 21)

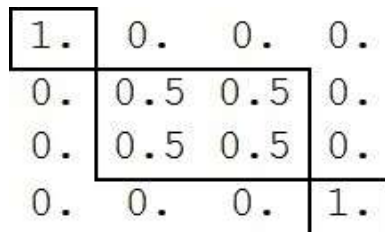


Рис. 21

Так как период равен 3, то представление P^3 имеет стандартный блочный вид, описанный выше.

3.9. Пример анализа состояний цепи Маркова

Рассмотрим для примера цепь Маркова с 10 состояниями $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ и матрицей перехода P . Эта матрица стохастическая, так сумма элементов каждой из строк равна 1.

$$P = \begin{pmatrix} 0.7 & 0.2 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0 & 0 & 0 & 0.4 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0.8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.6 & 0 & 0 & 0 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.8 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Построим граф (рис. 22) полученной цепи Маркова. Из картинке видно, что состояния $C = \{1, 3, 5\}$ - несущественные, наборы состояний $D1 = \{2, 7\}$, $D2 = \{4, 6, 8, 10\}$ и $D3 = \{9\}$ образуют эргодические классы. Переименуем состояния для получения стандартного представления матрицы перехода. Правило переименования $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \rightarrow \{1, 3, 5, 2, 7, 4, 6, 8, 10, 9\}$ и определим матрицу перехода $P1$ в новых обозначениях. Выделим на полученной матрице блоки $C, D1, D2, D3$ (рис. 23 и 24)

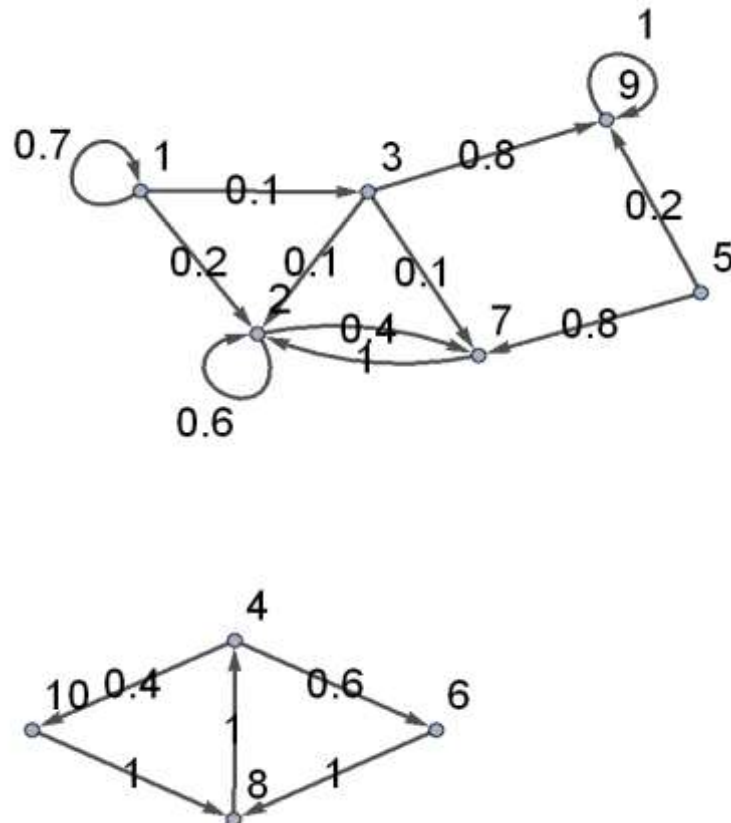


Рис. 22

0.7	0.1	0	0.2	0	0	0	0	0	0
0	0	0	0.1	0.1	0	0	0	0	0.8
0	0	0	0	0.8	0	0	0	0	0.2
0	0	0	0.6	0.4	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0.6	0	0.4	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1

Рис. 23

$$\begin{array}{l}
 \text{Блок C} \quad \begin{pmatrix} 0.7 & 0.1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\
 \text{Блок D2} \quad \begin{pmatrix} 0 & 0.6 & 0 & 0.4 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \\
 \text{Блок D1} \quad \begin{pmatrix} 0.6 & 0.4 \\ 1 & 0 \end{pmatrix} \\
 \text{Блок D3} \quad (1)
 \end{array}$$

Рис. 24

3.10. Эргодическая теорема

Как уже говорилось выше, для цепи Маркова интерес представляет поведение распределения вероятностей системы при больших временах, $t \rightarrow \infty$ и выяснения вопроса - "забывает" ли система свое начальное распределение вероятностей при $t = 0$. Ответ заведомо отрицательный, если в системе есть несколько эргодических классов или есть циклические подклассы.

При нескольких эргодических классах предположим, что начальное состояние принадлежит одному из них, тогда система вечно останется в этом классе. Аналогично, если начальное состояние принадлежит некоторому циклическому подклассу, то с некоторым периодом h система будет возвращаться в этот подкласс и никакого "забывания" начального состояния не будет.

Другое поведение системы будет, если она состоит из одного эргодического класса без циклических подклассов. Есть простой признак такой системы. Если существует некоторый момент времени $t = k_0$ такой, что для любых S_i и S_j переходная вероятность $p_{i,j}^{(k_0)} > 0$. Это означает, что за время $t = k_0$ возможен переход из состояния S_i в состояние S_j . Или, что то же самое, все элементы матрицы $P^{(k_0)}$ отличны от нуля.

Для таких систем справедлива **эргодическая теорема Маркова**. Пусть число состояний цепи Маркова конечно, $m < \infty$, и пусть цепь содержит только один эргодический класс без циклических подклассов. Тогда для любых S_i и S_j существует конечный предел

$$\lim_{n \rightarrow \infty} p_{i,j}^{(n)} = p_j, 1 \leq j \leq m,$$

не зависящий от начального состояния S_i . При этом

$$p_j > 0, \quad 1 \leq j \leq m$$

$$\sum_{j=1}^m p_j = 1,$$

$$p_j = \sum_{k=1}^m p_k * p_{k,j}, \quad 1 \leq j \leq m$$

Основная сложность - доказать существование предела (мы этого делать не будем). Системы уравнений для предельных вероятностей p_j при этом получается просто. Действительно,

$$p_{i,j}^{(n)} = \sum_{k=1}^m p_{i,k}^{(n-1)} * p_{k,j},$$

$$\lim_{n \rightarrow \infty} p_{i,j}^{(n)} = \lim_{n \rightarrow \infty} \sum_{k=1}^m p_{k,k}^{(n-1)} * p_{k,j},$$

$$p_j = \sum_{k=1}^m p_k * p_{k,j}$$

Последняя система уравнений может быть переписана в матричном виде, введем вектор-столбец

$$\vec{p} = \begin{pmatrix} p_1 \\ p_2 \\ \dots \\ p_m \end{pmatrix}$$

и матрицу P^T - транспонированную матрицу перехода P .

$$P^T = \begin{pmatrix} p_{1,1} & p_{2,1} & \dots & p_{m,1} \\ p_{1,2} & p_{2,2} & \dots & p_{m,2} \\ \dots & \dots & \dots & \dots \\ p_{1,m} & p_{2,m} & \dots & p_{m,m} \end{pmatrix}$$

Тогда система уравнений имеет следующий вид

$$\begin{cases} p_1 = p_{1,1} * p_1 + p_{2,1} * p_2 + \dots + p_{m,1} * p_m \\ p_2 = p_{1,2} * p_1 + p_{2,2} * p_2 + \dots + p_{m,2} * p_m \\ \dots \\ p_m = p_{1,m} * p_1 + p_{2,m} * p_2 + \dots + p_{m,m} * p_m \end{cases}$$

ИЛИ

$$\begin{aligned} \vec{p} &= P^T * \vec{p} \\ (P^T - E) * \vec{p} &= 0 \end{aligned}$$

Определитель этой системы равен нулю: $\det(P^T - E) = 0$, что связано со стохастичностью матрицы P , сумма элементов любого столбца матрицы $(P^T - E)$ равна 0.

Поэтому система имеет бесконечное множество решений, но есть дополнительное условие нормировки, $\sum_{j=1}^m p_j = 1$, что позволяет получить единственное решение этой системы.

Полученное распределение вероятностей называется **финальным** распределением вероятностей для цепи Маркова.

3.11. Стационарное распределение вероятностей

Распределение вероятностей состояний системы S называется **стационарным**, если при выборе его в качестве начального оно не меняется со временем. Если $w = \{a_1, a_2, \dots, a_m\}$ начальное состояние при $t = 0$, то условие его неизменяемости во времени $\vec{a} = \vec{a}P$ или $(\vec{a})^T = P^T(\vec{a})^T$. Другими словами вектор-столбец $(\vec{a})^T$ должен быть собственным вектором матрицы P^T , отвечающим собственному значению $\lambda = 1$. Такое собственное значение в силу стохастичности матрицы P всегда есть, так как уравнение для собственных чисел имеет вид $\det(P^T - \lambda E) = 0$ или, при $\lambda = 1$, получаем соотношение $\det(P^T - E) = 0$. А это всегда верно, так как если сложить все столбцы матрицы $(P^T - E)$, то получится 0, следовательно, определитель равен нулю.

3.12. Связь финального и стационарного распределения вероятностей

Если в цепи Маркова есть финальное распределение вероятностей, выполняются условия эргодической теоремы Маркова, то это финальное распределение совпадает со стационарным.

Действительно, уравнение для финального распределения совпадает с уравнением для собственного вектора, отвечающего собственному числу $\lambda = 1$.

Таким образом в предположении эргодической теоремы финальное распределение можно искать как нормированный собственный вектор матрицы P^T для собственного числа $\lambda = 1$ (нормировка – сумма элементов вектора равна 1).

4. Моделирование цепей Маркова

4.1. Пример технической системы

Имеется некоторое техническое устройство, которое в процессе своего функционирования может выходить из строя, процесс определения типа неисправности требует некоторого времени, в зависимости от типа неисправности устройство может подвергаться либо профилактике, либо среднему, либо капитальному ремонту, после окончания ремонта устройство продолжает нормально функционировать. В процессе выполнения профилактики может оказаться, что устройство требует серьезного ремонта, среднего или полного. Для описания функционирования этого устройства можно ввести следующие состояния: s_1 - устройство исправно и нормально функционирует, s_2 - устройство вышло из строя, определяется тип необходимого ремонта, s_3 - устройство требует среднего ремонта, s_4 - устройство не работает, находится на профилактике, s_5 - устройство требует полного ремонта. Из опыта получены следующие данные о вероятностях возможных переходов, переходы из состояния s_1 :

$$p_{1,2} = 0.1, p_{1,4} = 0.1, p_{1,1} = 1 - p_{1,2} - p_{1,4},$$

$$\text{из состояния } s_2: p_{2,3} = 0.5, p_{2,5} = 0.1, p_{2,2} = 1 - p_{2,3} - p_{2,5},$$

$$\text{из состояния } s_3: p_{3,1} = 0.8, p_{3,3} = 1 - p_{3,1},$$

$$\text{из состояния } s_4: p_{4,1} = 0.7, p_{4,3} = 0.1, p_{4,5} = 0.1, p_{4,4} = 1 - p_{4,1} - p_{4,3} - p_{4,5},$$

$$\text{из состояния } s_5: p_{5,1} = 0.1, p_{5,5} = 1 - p_{5,1}$$

В рассматриваемом примере система имеет пять состояний: $\text{vertexes} = \{s_1, s_2, s_3, s_4, s_5\}$, и матрицу перехода

$$P = \begin{pmatrix} 0.8 & 0.1 & 0 & 0.1 & 0 \\ 0 & 0.4 & 0.5 & 0 & 0.1 \\ 0.8 & 0 & 0.2 & 0 & 0 \\ 0.7 & 0 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0 & 0 & 0 & 0.9 \end{pmatrix}$$

Эта матрица является стохастической - сумма элементов каждой из строк равна 1:

$$\text{Total}[P, \{2\}] = \{1, 1, 1, 1, 1\}$$

4.2. Изображение системы в виде ориентированного графа

Изобразим систему в виде ориентированного графа, для этого получим описание ориентированного графа в виде списка ребер с весами, равными вероятностям перехода (ребра определяются условием не отрицательности вероятностей перехода).

Пример кода

defin =

```
Flatten[DeleteCases[MapIndexed[  
{DirectedEdge @@ vertexes[[#2]], #1} &, P, {2}], {_, 0}, {2}], 1]
```

Результат

```
{ {s1 • → s1, 0.8}, {s1 • → s2, 0.1}, {s1 • → s4, 0.1},  
  {s2 • → s2, 0.4}, {s2 • → s3, 0.5}, {s2 • → s5, 0.1},  
  {s3 • → s1, 0.8}, {s3 • → s3, 0.2}, {s4 • → s1, 0.7}, {s4 • → s3, 0.1},  
  {s4 • → s4, 0.1}, {s4 • → s5, 0.1}, {s5 • → s1, 0.1}, {s5 • → s5, 0.9} }
```

Получение меток для ребер в виде вероятностей перехода и списка весов в виде вероятностей перехода.

Пример кода

```
labels = defin /. {e_., w_} → e → w
```

```
weights = defin /. {_, w_} → w
```

Построение графа переходов для цепи Маркова.

Пример кода

```
Graph[vertexes, defin[[All, 1]],
```

```
VertexLabels → "Name", EdgeLabels → labels,
```

```
EdgeWeight → weights, ImagePadding → 15,
```

```
VertexLabelStyle → 20, VertexShapeFunction → "Square"]
```

Результат (рис. 25)

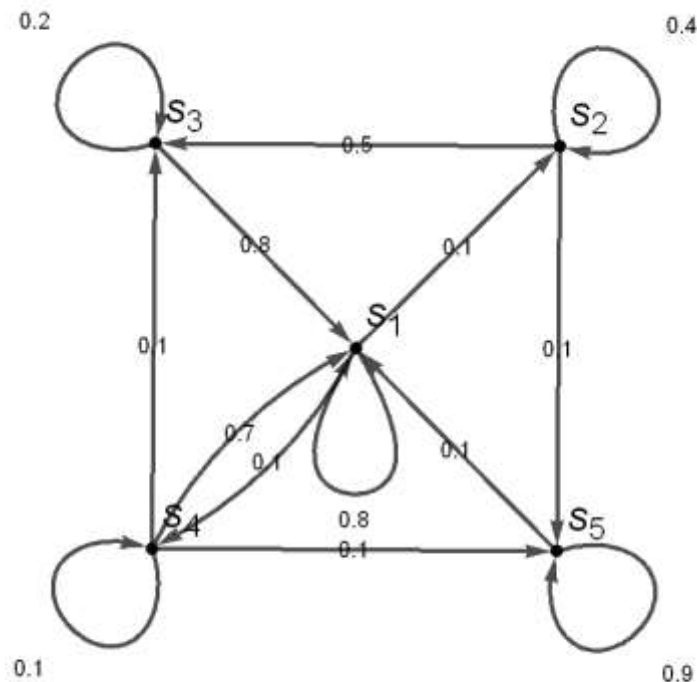


Рис. 25

4.3. Моделирование функции перехода

Основные принципы имитационного моделирования заключаются в следующем. Предположим, что система в данный момент времени находится в некотором состоянии, например, в состоянии S_1 . Тогда в следующий момент система либо останется в состоянии S_1 (с вероятностью 0.8), либо перейдет в состояние S_2 (с вероятностью 0.1) или в состояние S_4 (с вероятностью 0.1). Таким образом нам необходимо смоделировать дискретную случайную величину, принимающую три значения $\{S_1, S_2, S_4\}$, с вероятностями $\{0.8, 0.1, 0.1\}$. Стандартный способ моделирования заключается в разбиении отрезка $[0, 1]$ на три части с длинами, равными этим вероятностям $\{0.8, 0.1, 0.1\}$: первый отрезок $[0, 0.8]$, второй отрезок $[0.8, 0.9]$ и третий $[0.9, 1]$. Далее генерируется случайное (псевдослучайное) число $x, 0 \leq x \leq 1$ и в зависимости от того, в какой отрезок оно попадает, определять новое состояние. Например, если $x=0.92$, то новое состояние будет S_4 .

Таким образом необходимо смоделировать функции перехода из произвольного состояния в любое другое в соответствии с заданной вероятностью перехода. Для этого для произвольного состояния введем интегрированную вероятность перехода $P1 = (\text{Accumulate} / @ P)$. Так, для рассматриваемого примера, $P1[[1]] = \{0.8, 0.9, 0.9, 1, 1\}$, соответствующая кусочная функция перехода из состояния $s=1$ равна

$\text{Piecewise}[\{\{1, x < 0.8\}, \{2, x < 0.9\}, \{4, x < 1.\}\}]$

В зависимости от величины случайного значения $x, 0 \leq x \leq 1$, получаем новое состояние, например, при $x=0.92$ получаем новое состояние S_4 .

Можно определить единую функцию, моделирующую переходы из произвольного состояния, здесь P – исходная матрица перехода и x – псевдослучайное число. Далее в коде состояния обозначаются просто их номерами.

Пример кода

```
randomPerehodFunction[x_, P_] :=
```

```
Piecewise[MapIndexed[{\#2[[1]], x < \#1}&, \#]] & / @(Accumulate / @ P)
```

Пример ее использования, функция перехода из состояния $s = 1$

```
randomPerehodFunction[x, P][[1]]
```

Результат

$$\begin{cases} 1 & x < 0.8 \\ 2 & x < 0.9 \\ 4 & x < 1. \\ 0 & True \end{cases}$$

При фиксированной матрице перехода P можно ввести функцию

```
randomPerehod[x_] := randomPerehodFunction[x, P]
```

зависящую только от псевдослучайного числа x .

4.4. Моделирование последовательности переходов

Смоделируем последовательность 4 переходов, исходя из начального состояния $s=2$.

```
Пример кода
SeedRandom[1234567]
{s = 2,
 s = randomPerehod[RandomReal[]][[s]],
 s = randomPerehod[RandomReal[]][[s]],
 s = randomPerehod[RandomReal[]][[s]],
 s = randomPerehod[RandomReal[]][[s]]
}
```

Полученный результат {2,3,1,4,1}

Можно организовать циклическую процедуру, моделирующую последовательность $nmod$ переходов из начального состояния $s=1$

```
Пример кода
SeedRandom[1234567]
nmod = 10;
s = 1;
NestList[
 randomPerehod[RandomReal[]][[#]]&,
 s, nmod
]
```

Контрольный вывод содержит последовательность смены состояний системы, начиная с первого в момент времени $t=0$ до конечного момента $t=nmod$: {1,1,1,4,1,1,1,2,2,2,3}

4.5. Моделирование времени нахождения системы в каждом из состояний

Введем массив $time$ - суммарное абсолютное время нахождения системы в каждом из состояний и относительное время $time/nmod$ нахождения системы в каждом из состояний. При каждом попадании системы в состояние s величина $time[[s]]$ увеличивается на 1.

Пример кода


```
SeedRandom[1234567]
nmod = 100;
time = ConstantArray[0.,5];
s = 1;
Nest[
(time[[#]] ++;randomPerehod[RandomReal[]][[#]])&,
s,nmod
];
time
time / nmod
```

Пример вывода: массив абсолютных времен {81, 6, 3, 9, 1} и массив относительных времен {0.81, 0.06, 0.03, 0.09, 0.01}.

При моделировании достаточно большого интервала времени, например, $nmod=10\ 000$, получаем результат для относительных времен пребывания системы в каждом из состояний

```
{0.5874, 0.0935, 0.0642, 0.0656, 0.1894}
```

4.6. Сравнение результатов моделирования с теоретическими результатами

Рассматриваемая система является эргодической системой. Следовательно, у нее есть предельное распределение вероятностей, которое можно получить вычисляя достаточно большие степени матрицы перехода P , например, P^{10} или P^{50}

Пример кода

```
P // MatrixForm
```

```
MatrixPower[P,10] // MatrixForm
```

```
MatrixPower[P,50] // MatrixForm
```

Результаты расчета

$$P = \begin{pmatrix} 0.8 & 0.1 & 0 & 0.1 & 0 \\ 0 & 0.4 & 0.5 & 0 & 0.1 \\ 0.8 & 0 & 0.2 & 0 & 0 \\ 0.7 & 0 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0 & 0 & 0 & 0.9 \end{pmatrix}$$

$$P^{10} = \begin{pmatrix} 0.631 & 0.1066 & 0.0764 & 0.0707 & 0.1153 \\ 0.6075 & 0.1017 & 0.0723 & 0.0677 & 0.1508 \\ 0.6371 & 0.1079 & 0.0774 & 0.0715 & 0.1062 \\ 0.6135 & 0.1029 & 0.0733 & 0.0684 & 0.1418 \\ 0.4476 & 0.0682 & 0.0445 & 0.0471 & 0.3926 \end{pmatrix}$$

$$P^{50} = \begin{pmatrix} 0.5977 & 0.0996 & 0.0706 & 0.0664 & 0.1657 \\ 0.5976 & 0.0996 & 0.0705 & 0.0664 & 0.1659 \\ 0.5977 & 0.0996 & 0.0706 & 0.0664 & 0.1657 \\ 0.5976 & 0.0996 & 0.0706 & 0.0664 & 0.1658 \\ 0.5967 & 0.0994 & 0.0704 & 0.0663 & 0.1672 \end{pmatrix}$$

Полученные результаты показывают существование предельного значения для степеней матрицы перехода при $n \rightarrow \infty$. При этом все строчки предельной матрицы будут одинаковыми. Это предельное значение можно получить как собственный вектор матрицы P^T , отвечающий собственному числу $\lambda = 1$.

Eigenvectors[Transpose[P]][[1]] / Total[Eigenvectors[Transpose[P]][[1]]]

Соответствующее нормированное на 1 (сумма элементов равна 1) собственное значение равно

{0.59751, 0.0995851, 0.0705394, 0.06639, 0.165975}

Видно, что элементы матрицы P^{50} практически равны элементам собственного вектора.

Полученное теоретическое значение вероятностей можно сравнить со значениями относительных времен, найденными при имитационном моделировании

{0.5874, 0.0935, 0.0642, 0.0656, 0.1894}

Из теории следует, что для эргодических цепей Маркова финальные вероятности должны равняться среднему значению времени нахождения системы в каждом из состояний. Получаем достаточно хорошее согласие результата эксперимента с теорией.

4.7. Среднее время непрерывного нахождения в произвольном состоянии и средняя длина цикла от попадания в состояние до следующего попадания в это же состояние

Обозначим через t_0 - среднее время нахождения системы в состоянии s_0 , k_0 - количество циклов от попадания в это состояние до следующего попадания в него, h_0 - средняя длина цикла.

Пример кода. Переменная ins_0 определяет, находимся или нет в состоянии s_0 . Если ответ – да, счетчик времени $times_0$ увеличивается на 1 и, если это новое попадание в это состояние, то увеличивается на 1 и счетчик циклов k_0 .

Начальное состояние при моделировании $s = 1$. Состояние, для которого определяются все характеристики, $s_0 = 2$

```
SeedRandom[1234567]
```

```
nmod = 10000;
```

```
s = 1; s0 = 2;
```

```
ins0 = (s != s0);
```

```
times0 = k0 = 0;
```

Основной цикл моделирования

```
Nest[
```

```
(If[# != s0, ins0 = False,
```

```
times0 ++; If[!ins0, k0 ++; ins0 = True]];
```

```
randomPerehod[RandomReal[]][[#]])&,
```

```
s, nmod];
```

```
t0 = times0 / k0 // N
```

```
h0 = nmod / k0 // N
```

Для времени моделирования $nmod=10000$ получены следующие результаты. Среднее время нахождения системы в состоянии $s_0=2$ равно $t_0=1.6386$, а средняя длина цикла от попадания до следующего попадания $k_0=17.5439$

Результаты моделирования можно сравнить с теоретическими значениями

```
p0 = P[[s0, s0]];
```

```
t0teor = (1 - p0) Sum[(i + 1) p0^i, {i, 0, ∞}]
```

```
p0fin = Limit[MatrixPower[P, i][[s0, s0]], i → ∞] // Chop;
```

```
h0teor = t0teor / p0fin
```

$t_{0teor} = 1.6667$, $h_{0teor} = 16.7361$, что дает достаточно хорошее согласие теории и эксперимента.

4.8. Моделирование функции перехода с использованием стандартных функций пакета Wolfram Mathematica

Определим набор состояний (используется кодировка в виде целых чисел)

```
S = Range[1, Length[P]]
```

Построим закон распределения переходов из произвольного состояния

```
distr = EmpiricalDistribution[# → S] & / @ P
```

Для моделирования используется функция RandomVariate. Вызов этой функции для произвольного состояния *sold* определяет новое случайное состояние *snew* с учетом матрицы перехода *P*:

```
snew = RandomVariate[distr[[sold]]]
```

Проведем моделирование четырех моментов времени, исходя из начального состояния $s=2$.

Пример кода.

```
SeedRandom[1234567]
{
  s = 2,
  s = RandomVariate[distr[[s]],
  s = RandomVariate[distr[[s]],
  s = RandomVariate[distr[[s]],
  s = RandomVariate[distr[[s]]
}
```

Эта функция может использоваться вместо применяемой ранее функции `randomPerehod`. Тогда цикл моделирования случайного процесса имеет следующий вид.

Пример кода.

```
(*Моделирование траекторий из состояния s0*)
SeedRandom[1234567]
nmod = 10000;
s0 = 1;
traekt = NestList[RandomVariate[distr[[#]]]&, s0, nmod];
Length @ traekt
traekt[[1;;20]](*первые 20 элементов траектории*)
```

Результат – смена состояний {1,1,1,4,1,1,1,2,2,2,3,1,4,1,4,1,1,1,1}

Нетрудно вычислить и относительные времена нахождения системы в каждом из состояний.

```
Count[traekt, #] / nmod & / @ S // N
```

Результаты совпадают с полученными ранее в 4.4.

4.9. Вычисление заданной функции вдоль траектории

Рассмотрим более общую задачу – вычисление произвольной функции вдоль траектории случайного процесса. Эта функция, `evalpost[old_, new_]`, вычисляется в каждый момент времени, ее аргументы – старое `old` и новое `new` состояние. Один шаг итерации реализует одна из функций `iter`.

Пример кода

```
iter[sold _ Integer, evalpost _] :=
Block[{snew = RandomVariate[distr[[sold]]], evalpost[sold, snew]; snew}
(*Либо по умолчанию не вычисляется ничего*)
iter[sold _ Integer] := iter[sold, Null &]
```

Цикл моделирования для определения времени нахождения в каждом из состояний будет тогда иметь следующий вид.

Пример кода с вычислением функции *eval1*

```
SeedRandom[1234567]
```

```
nmod = 10000;
```

```
s = 1;
```

```
time = ConstantArray[0, Length @ S];
```

```
(*задание функции eval1*)
```

```
eval1 = (time[[#1]]++)&;
```

```
(*основной цикл моделирования*)
```

```
Nest[iter[#, eval1]&, s, nmod];
```

```
(*вычисление относительных времен*)
```

```
N[time / nmod]
```

Полученные результаты совпадают с 4.5.

Аналогично можно повторить и результаты 4.7.

Пример кода с вычислением функции *eva21*

```
SeedRandom[1234567]
```

```
nmod = 10000;
```

```
s = 1;
```

```
s0 = 2;
```

```
ins0 = (s Ё s0);
```

```
times0 = Boole[ins0];
```

```
k0 = Boole[ins0];
```

```
(*задание функции eval2*)
```

```
eval2 = (If[#1 ≠ s0, ins0 = False, times0++; If[! ins0, k0++; ins0 = True]])&;
```

```
(*основной цикл моделирования*)
```

```
Nest[iter[#, eval2]&, s, nmod];
```

```
(*Среднее время пребывания в заданном состоянии*)
```

```
t0 = times0 / k0 // N
```

```
(*средняя длина цикла*)
```

```
h0 = nmod / k0 // N
```

Аналогично можно смоделировать и частоту перехода из состояния i_0 в состояние j_0 , то есть вычислить количество переходов вида $i_0 \rightarrow j_0$.

```
SeedRandom[1234567]
```

```
 $nmod = 10000;$ 
```

```
 $s = 1;$ 
```

```
 $i_0 = 4;$ 
```

```
 $j_0 = 3;$ 
```

```
 $k = 0;$  (*количество переходов вида  $i_0 \rightarrow j_0$ *)
```

```
 $n = 0;$  (*количество попаданий в состояние  $i_0$ *)
```

```
(*задание функции eval5*)
```

```
eval5[ $i_0$ _,  $j_0$ _]:=
```

```
(If[#1  $\text{Lb}$  $i_0$ ,  $n++$ ];  $k++ = \text{Boole}$ [#1  $\text{Lb}$  $i_0 \wedge$  #2  $\text{Lb}$  $j_0$ ]) &
```

```
(*основной цикл моделирования*)
```

```
Nest[iter[#, eval5[ $i_0$ ,  $j_0$ ]]&,  $s$ ,  $nmod$ ];
```

```
(*вычисление частоты попаданий в состояние  $i_0$ *)
```

```
 $N[k / n]$ 
```

```
(*теоретическое значение*)
```

```
P[[ $i_0$ ,  $j_0$ ]]
```

Результаты моделирования. Имитация дает 0.0915, точное значение равно 0.1.

5. Марковский случайный процесс с дискретным числом состояний и непрерывным временем

5.1. Случайный процесс с дискретным числом состояний и непрерывным временем

Предположим, что система S имеет дискретное, обычно конечное, множество состояний $\{S_1, S_2, \dots, S_m\}$ и переход из состояния в состояние происходит мгновенно в случайные моменты времени. Если предполагается, что временная переменная t меняется непрерывно, то говорят о случайном процессе $X(t)$ с непрерывным временем. Для каждого момента времени t определено распределение вероятностей нахождения системы S в некотором состоянии

$$p_i(t) = P(\{X(t) = S_i\}), \quad i = 1, 2, \dots, m$$

$$\sum_{i=1}^m p_i(t) = 1$$

Соответствующую систему можно изображать с помощью ориентированного графа, вершины которого есть возможные состояния системы, а ориентированные дуги отвечают возможным переходам из одного состояния в другое.

Пример. Рассмотрим систему S , для которой задана матрица смежности P , описывающая возможные переходы из состояния в состояние

$$S = \{s_1, s_2, s_3, s_4, s_5\}$$

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

Граф, описывающий эту систему (рис. 26), имеет вид

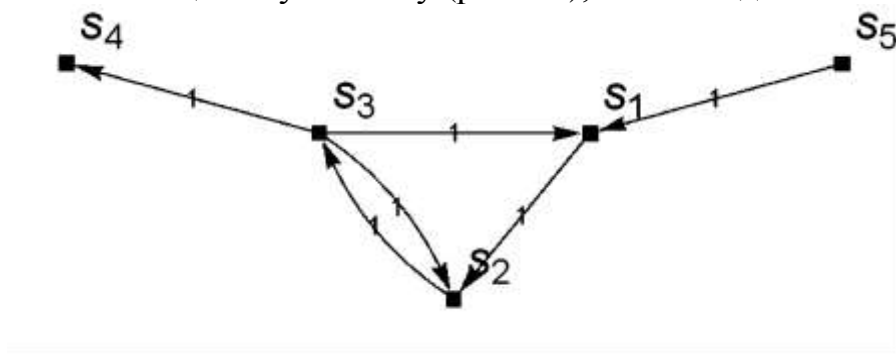


Рис. 26

Особенность этого графа заключается в том, что переходы из любого состояния в это же состояние не изображаются в виде дуги. Причина этого заключается в том, что если система в некоторый момент находится в каком-то состоянии, то переход в это же состояние означает просто, что си-

стема не меняет состояния. А целью графа является описание только возможных смен состояний.

5.2. Марковский случайный процесс с непрерывным временем

Случайный процесс с дискретным числом состояний и непрерывным временем называется **Марковским случайным процессом**, если для любых двух моментов времени t_1 и t_2 , $t_1 < t_2$, условная вероятность нахождения системы S в момент времени t_2 в состоянии S_j зависит только от состояния системы в момент времени t_1 , S_i , и не зависит от того, что было в предшествующие моменту t_1 моменты времени, то есть от состояний в моменты времени при $t < t_1$.

Таким образом можно ввести **условную вероятность перехода** из состояния S_i в момент времени t_1 в состояние S_j в момент времени t_2 .

$$p_{i,j}(t_1, t_2) = P(\{X(t_2) = s_j\} / \{X(t_1) = s_i\}),$$

$$i = 1, 2, \dots, m, j = 1, 2, \dots, m$$

Совокупность всех переходных вероятностей позволяет ввести матрицу перехода

$$P(t_1, t_2) = \begin{pmatrix} p_{1,1}(t_1, t_2) & p_{1,2}(t_1, t_2) & \dots & p_{1,m}(t_1, t_2) \\ p_{2,1}(t_1, t_2) & p_{2,2}(t_1, t_2) & \dots & p_{2,m}(t_1, t_2) \\ \dots & \dots & \dots & \dots \\ p_{m,1}(t_1, t_2) & p_{m,2}(t_1, t_2) & \dots & p_{m,m}(t_1, t_2) \end{pmatrix}$$

Эта матрица является стохастической матрицей, сумма элементов каждой строки равна 1.

Рассмотрим произвольный момент времени t , промежуточный между t_1 и t_2 , $t_1 < t < t_2$. Так же, как и для цепи Маркова, можно показать, что

$$P(t_1, t_2) = P(t_1, t) * P(t, t_2)$$

5.3. Поток событий, связанные со сменой состояний

Введем потоки событий, связанные с переходом из одного состояния в другое. Зафиксируем некоторую пару состояний, S_i и S_j , для которой в графе системы есть дуга перехода. И рассмотрим поток событий, связанный с этим переходом. События этого потока происходят только в момент перехода из состояния S_i в S_j .

Отметим на временной оси точками всевозможные смены состояний системы S и выделим только соседние пары, переход из S_i в S_j . Перенесем их на одну ось, получим необходимый поток событий (рис. 27).

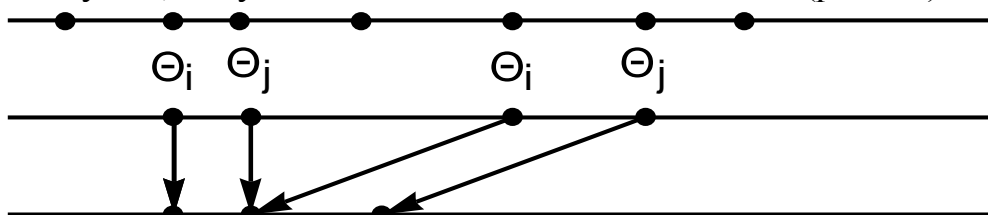


Рис. 27

Основное утверждение. Для того, чтобы случайный процесс был Марковским необходимо и достаточно, чтобы все потоки событий, связанные со сменой состояний, были пуассоновскими.

Каждый пуассоновский поток событий однозначно характеризуется своей интенсивностью. Таким образом, с каждым возможным переходом из состояния S_i в S_j связана интенсивность этого перехода $\alpha_{i,j}(t)$. Эту интенсивность обычно изображают на графе, считая интенсивность весом соответствующей дуги. Если переход из одного состояния в другое невозможен, то полагают интенсивность этого перехода равной нулю.

В силу ординарности пуассоновского потока условные вероятности перехода удовлетворяют следующим условиям

$$p_{i,j}(t, t + \Delta t) = \alpha_{i,j}(t)\Delta t + o(\Delta t), i \neq j,$$

$$p_{i,i}(t, t + \Delta t) = 1 - \sum_{j=1}^m \alpha_{i,j}(t)\Delta t + o(\Delta t)$$

Последнее соотношение получается в силу того, что матрица $P(t_1, t_2)$ должна быть стохастической, сумма элементов каждой строки равна 1.

5.4. Уравнения Колмогорова

Для вероятностей перехода $p_{i,j}(t_1, t_2)$ можно получить систему дифференциальных уравнений. Покажем, как эта система выводится на примере системы S с двумя состояниями $\{S_1, S_2\}$ и матрицей интенсивностей переходов

$$P = \begin{pmatrix} 0 & \alpha_{1,2}(t) \\ \alpha_{2,1}(t) & 0 \end{pmatrix}$$

Соответствующий граф представлен на рисунке 28

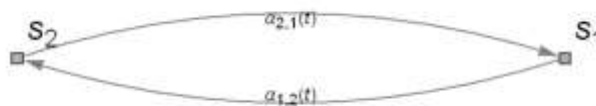


Рис. 28

Рассмотрим, для определенности, функцию $p_1(t) = P(X(t) = S_1)$ и вычислим ее значение в момент времени $t + \Delta t$. По формуле полной вероятности, получаем

$$\begin{aligned} p_1(t + \Delta t) &= \sum_{i=1}^2 P(\{X(t + \Delta t) = s_1\} / \{X(t) = s_i\})P(\{X(t) = s_i\}) = \\ &= p_{1,1}(t, t + \Delta t)p_1(t) + p_{2,1}(t, t + \Delta t)p_2(t) \end{aligned}$$

Потоки событий в этой системе ординарны, поэтому

$$p_{1,1}(t, t + \Delta t) = 1 - \alpha_{1,2}(t)\Delta t + o(\Delta t)$$

$$p_{2,1}(t, t + \Delta t) = \alpha_{2,1}(t)\Delta t + o(\Delta t)$$

Можно рассуждать следующим образом, вероятность $p_{1,1}(t, t+\Delta t)$ означает, что либо в течении интервал времени $[t, t+\Delta t)$ система непрерывно оставалась в состоянии S_1 , то есть не было ни одного перехода в S_2 , либо было два перехода, сначала из S_1 в S_2 , а потом назад в S_1 , либо 4 перехода и так далее. В силу ординарности потоков любой переход осуществляется с вероятностью пропорциональной Δt , следовательно, два перехода имеют вероятность пропорциональную $(\Delta t)^2 = o(\Delta t)$. Поэтому необходимо учитывать только возможность отсутствия переходов, а эта вероятность равна $1 - \alpha_{1,2}(t)\Delta t + o(\Delta t)$. Аналогичные рассуждения для вероятности $p_{2,1}(t, t+\Delta t)$, либо только один переход с вероятностью $\alpha_{2,1}(t)\Delta t + o(\Delta t)$, либо 3 перехода, из S_2 в S_1 , потом назад в S_2 , а потом опять в S_1 . Вероятность такой ситуации пропорциональна $(\Delta t)^3 = o(\Delta t)$. Поэтому справедлива приведенная формула для $p_{2,1}(t, t+\Delta t)$.

Подставляем полученные вероятности в формулу полной вероятности, получаем

$$\begin{aligned} p_1(t + \Delta t) &= (1 - \alpha_{1,2}(t)\Delta t + o(\Delta t))p_1(t) + (\alpha_{2,1}(t)\Delta t + o(\Delta t))p_2(t) = \\ &= p_1(t) - \alpha_{1,2}(t)p_1(t)\Delta t + \alpha_{2,1}(t)p_2(t)\Delta t + o(\Delta t) \end{aligned}$$

или, перенося $p_1(t)$ налево и разделив на Δt

$$\frac{p_1(t + \Delta t) - p_1(t)}{\Delta t} = -\alpha_{1,2}(t)p_1(t) + \alpha_{2,1}(t)p_2(t) + \frac{o(\Delta t)}{\Delta t}$$

Переходя к пределу при $\Delta t \rightarrow 0$

$$\frac{dp_1(t)}{dt} = -\alpha_{1,2}(t)p_1(t) + \alpha_{2,1}(t)p_2(t)$$

Аналогично рассуждая можно вывести и дифференциальное уравнение для $p_2(t)$. Итог - имеем систему дифференциальных уравнений первого порядка

$$\begin{aligned} \frac{dp_1(t)}{dt} &= -\alpha_{1,2}(t)p_1(t) + \alpha_{2,1}(t)p_2(t) \\ \frac{dp_2(t)}{dt} &= \alpha_{1,2}(t)p_1(t) - \alpha_{2,1}(t)p_2(t) \end{aligned}$$

Чтобы система уравнений имела единственное решение, необходимо задать начальные условия при $t = 0$, начальные значения вероятностей при $t = 0$

$$\begin{aligned} p_1(t) |_{t=0} &= p_1^{(0)} \\ p_2(t) |_{t=0} &= p_2^{(0)} \end{aligned}$$

Так как $p_1^{(0)} + p_2^{(0)} = 1$, то условие такого вида должно выполняться при любых t . Это действительно так, если сложить два дифференциальных уравнения, то получим

$$\begin{aligned} \frac{dp_1(t)}{dt} + \frac{dp_2(t)}{dt} &= \\ &= (-\alpha_{1,2}(t)p_1(t) + \alpha_{2,1}(t)p_2(t)) + (\alpha_{1,2}(t)p_1(t) - \alpha_{2,1}(t)p_2(t)) = 0 \\ \frac{d}{dt}(p_1(t) + p_2(t)) &= 0 \end{aligned}$$

Следовательно, $p_1(t) + p_2(t) = const$, если в начальный момент времени эта константа равна 1, то и при всех временах будет $p_1(t) + p_2(t) = 1$.

Покажем теперь, как выглядят эти уравнения для произвольной системы S . Для каждой вершины S_i графа переходов пусть индекс j пробегает все множество входящих в вершину S_i дуг, а индекс k - множество выходящих из вершины S_i дуг.

Тогда уравнение для $p_i(t)$ имеет следующий вид

$$\begin{aligned} \frac{dp_i(t)}{dt} &= -\sum_k \alpha_{i,k}(t)p_i(t) + \sum_j \alpha_{j,i}(t)p_j(t) = \\ &= -p_i(t) \sum_k \alpha_{i,k} + \sum_j \alpha_{j,i}(t)p_j(t), \\ i &= 1, 2, \dots, m \end{aligned}$$

В первое слагаемое входит суммирование по дугам, выходящим из вершины S_i , а во второе слагаемое суммирование по входящим в S_i дугам (рис. 29).

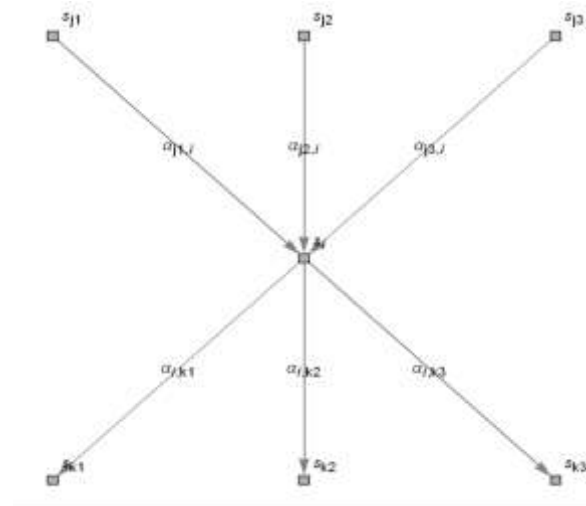


Рис. 29

Полученная система дифференциальных уравнений носит название **системы уравнений Колмогорова**.

Для однозначного решения этой системы необходимо задать набор начальных условий при $t = 0$

$$\begin{cases} p_1(t)|_{t=0} = p_1^{(0)} \\ p_2(t)|_{t=0} = p_2^{(0)} \\ \dots \\ p_{2m}(t)|_{t=0} = p_m^{(0)} \end{cases}$$

$$\sum_{i=1}^m p_i^{(0)} = 1$$

5.5. Пример решения системы дифференциальных уравнений

Рассмотрим систему, состоящую из двух состояний и интенсивностями переходов $\alpha_{1,2}$ и $\alpha_{2,1}$ не зависящими от времени. Соответствующая матрица интенсивностей переходов имеет следующий вид

$$P = \begin{pmatrix} 0 & \alpha_{1,2} \\ \alpha_{2,1} & 0 \end{pmatrix}.$$

Общее решение системы дифференциальных уравнений Колмогорова тогда равно

$$p_1(t) = \frac{\alpha_{2,1}}{\alpha_{1,2} + \alpha_{2,1}} + p_1^0 e^{-t(\alpha_{1,2} + \alpha_{2,1})} - \frac{e^{-t(\alpha_{1,2} + \alpha_{2,1})} \alpha_{2,1}}{\alpha_{1,2} + \alpha_{2,1}}$$

$$p_2(t) = \frac{\alpha_{1,2}}{\alpha_{1,2} + \alpha_{2,1}} - p_1^0 e^{-t(\alpha_{1,2} + \alpha_{2,1})} + \frac{e^{-t(\alpha_{1,2} + \alpha_{2,1})} \alpha_{2,1}}{\alpha_{1,2} + \alpha_{2,1}}$$

где p_1^0 - начальное условие для $p_1(0)$: $p_1(0) = p_1^0$. Начальное условие для $p_2(0)$: $p_2(0) = 1 - p_1^0$.

Положим $\alpha_{1,2} = 1$ и $\alpha_{2,1} = 2$ и рассмотрим графики в зависимости от начальных условий. Первый вариант - начальные условия $\{p_1^0 = 1, p_2^0 = 0\}$, то есть система в начальный момент времени находится в состоянии S_1 , второй вариант - $\{p_1^0 = 0, p_2^0 = 1\}$, то есть система в начальный момент времени находится в состоянии S_2 , третий вариант - $\{p_1^0 = 0.5, p_2^0 = 0.5\}$, то есть равновероятное распределение вероятностей в начальный момент времени.

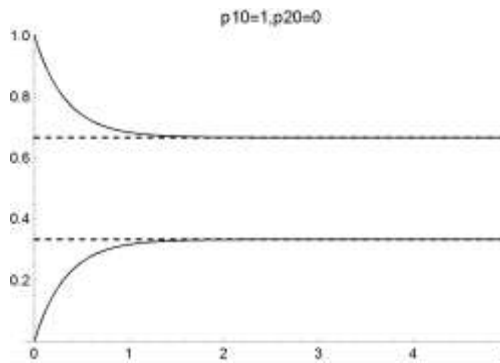


Рис. 30

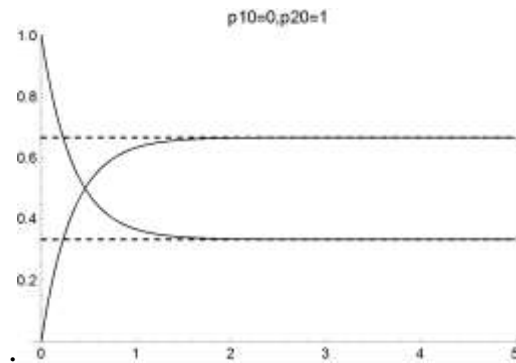


Рис. 31

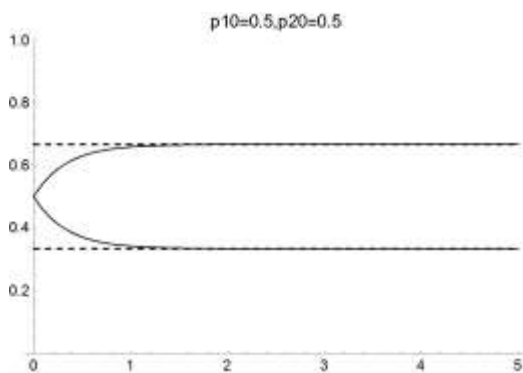


Рис. 32

На графиках (рис. 30, 31, 32) представлены решения системы дифференциальных при различных начальных условиях. Видно, как и для цепей Маркова, что временной интервал можно разделить на две части: переходной процесс, для данной системы при временах приблизительно меньших 2, и стабилизация к некоторым постоянным значениям, не зависящим от начального распределения вероятностей, при временах больших 2

Существование **предельных (финальных) распределений** вероятностей гарантировано для эргодических Марковских процессов. Простым условием эргодичности системы является условие достижимости любого состояния системы из любого другого и условие постоянства интенсивностей переходов или, в более слабой форме, существование предельных значений интенсивностей при $t \rightarrow \infty$.

Эти условия будем всегда считать выполненными.

6. Моделирование Марковского процесса

6.1. Описание исходной системы

Имитационное моделирование будем рассматривать на примере системы S , имеющей пять состояний, $S=\{1, 2, 3, 4, 5\}$, и матрицу интенсивностей переходов между состояниями, переходы между состояниями рассматриваются как элементы простейшего потока событий заданной интенсивностью

$$P = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 5 & 0 & 1 \\ 8 & 0 & 0 & 0 & 0 \\ 7 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Так, например, элемент матрицы $P[[2, 3]]=5$ описывает переход из состояния 2 в состояние 3 и интенсивность потока, связанного с этим переходом, равна 5.

6.2. Построение описания системы в виде списка

Удобно вместо матрицы интенсивностей переходов рассматривать описание системы S в виде списковой структуры. Список $spisok$ задается следующим образом: элемент этого списка $spisok(i)$ сам представляет список, состоящий из элементов вида $\{j, \lambda\}$, что означает наличие перехода из состояния i в состояние j с интенсивностью потока λ .

Так для нашей системы списковое описание имеет следующий вид

$$spisok[P] = \begin{pmatrix} \{\{2,1\},\{4,1\}\} \\ \{\{3,5\},\{5,1\}\} \\ \{\{1,8\}\} \\ \{\{1,7\},\{3,1\},\{5,1\}\} \\ \{\{1,1\}\} \end{pmatrix}$$

и может быть получено с помощью следующего кода

```
spisok[p_] := DeleteCases[
  MapIndexed[
    If[#1 > 0, {#2[[1]], #1}] &, #], Null] & /@ p
```

В этом описании $spisok[P] [[2]] = \{\{3,5\}, \{5,1\}\}$, что означает возможные переходы из состояния 2: в состояние 3 с интенсивностью 5 и в состояние 5 с интенсивностью 1.

6.3. Построение графа переходов с указанием интенсивностей переходов

Построим список ребер и отвечающий ему список весов.

Пример кода

```
edges = Flatten @ MapIndexed[DirectedEdge[#2[[1]],#1[[1]]]&,spisok[P],{2}]
weights = Flatten @ MapIndexed[#1[[2]]&,spisok[P],{2}]
```

Построение графа (рис. 34) интенсивностей переходов. Отличие от цепей Маркова - отсутствие петель, петля для цепи Маркова означала, что система остается в соответствующем состоянии.

```
Пример кода
Graph[S,edges,
VertexLabels -> "Name",
EdgeLabels -> Thread[edges -> weights],
EdgeWeight -> weights,
ImagePadding -> 15,
VertexLabelStyle -> 20,
VertexShapeFunction -> "Square"]
```

Результат

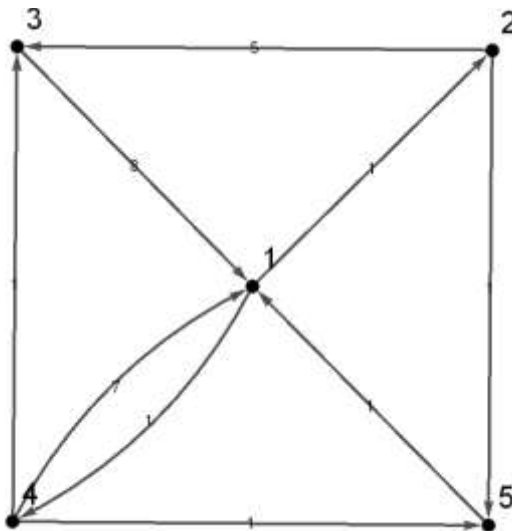


Рис. 33

6.4. Генератор временного интервала для событий потока

Для простейших потоков событий интервал времени между событиями потока распределен по показательному закону.

Пример кода для генерации реализаций показательного закона распределения с параметром λ :

```
trandom[λ_] := RandomReal[ExponentialDistribution[λ]]
```

Проверка работы генератора. Первые 10 чисел, распределенных по показательному закону с параметром $\lambda = 1$

```
SeedRandom[1234567]
```

```
Array[trandom[1]&,10]
```

Результат (с точностью до 4 знаков после точки)
 {0.6826,0.8292,0.0437,1.0434,1.3890,0.2307,0.1128,1.2594,1.4613,0.1655}

6.5. Моделирование Марковского процесса

Основой моделирования является следующее замечание. Пусть система находится в некотором состоянии i и есть список всех возможных переходов из этого состояния. Смоделируем с помощью генератора показательного закона для каждого возможного перехода интервал времени для события соответствующего потока. Из всех полученных интервалов времени выберем наименьший, это значение и определяет реальное время выхода из состояния i . Новое состояние определяется полученным переходом.

Предположим, что система находится в состоянии i и в момент времени t переходит в состояние j . Тогда удобно ввести обобщенное описание этого перехода в виде следующего списка $q = \{j, t\}$, то есть переменная q описывает следующие характеристики $q[[1]] = \{\text{новое состояние}\}$, $q[[2]] = \{\text{общее время от 0 до перехода в это состояние}\}$. Описание всей системы дается в виде списка $spisok[P]$.

Пример кода

```
jumpsp[q_, spisok_] :=
```

```
Block[{min, pos, sp = spisok[[q[[1]]], tp},
```

```
(*генерация интервалов времени для всех переходов  
из состояния q[[1]]*)
```

```
tp = trandom / @ sp[[All, 2]]
```

```
(*определение реализуемого состояния для перехода*)
```

```
min = Min[tp];
```

```
pos = Position[tp, min][[1, 1]];
```

```
(*результат-новое состояние и суммарное время*)
```

```
{sp[[pos, 1]], q[[2]] + min}
```

```
jumpq[q_] := jumpsp[q, spisok[P]]
```

Тестирование.	Результаты тестирования
SeedRandom[1234567]	{1,0}
$q0 = \{1,0\}$	{2,0.68}
$q1 = \text{jumpq}[q0]$	{3,0.69}
$q2 = \text{jumpq}[q1]$	

Пусть $q0 = \{1,0\}$, то есть начальное состояние 1 и начальное время $t=0$. Далее переход в $q1 = \{2, 0.68\}$, то есть переход в состояние 2 и общее время 0.68, далее переход в $q2 = \{3, 0.69\}$, то есть переход в состояние 3 и общее время 0.69.

Удобно ввести функцию *iteration*, выполняемую на каждом шаге моделирования. Эта функция определяет новое состояние и осуществляет дополнительное вычисление некоторой задаваемой дополнительно функции *eval*(*q_old*, *q_new*), зависящей как от нового, *q_new*, так и от старого, *q_old*, состояния.

```

Пример кода
iteration[q_,eval _]:=
With[{qn = jumpq[q]},
eval[q,qn];
qn]

```

Весь процесс моделирования тогда описывается с помощью функции *simulationMarkov* - симуляции процесса смены состояний и вычисление на каждой итерации дополнительной внешней функции *eval*[*q_old*, *q_new*]. В функции *simulationMarkov* аргументы: *s0* – начальное состояние системы, *tmod* – общее время моделирования, *eval* – функция, которая вычисляется на каждом шаге итерации

```

Пример кода
simulationMarkov[eval_,s0_,tmod _]:=
NestWhile[iteration[#,eval]&,{s0,0},(#[[2]] < tmod)&]

```

Пример1. Проверка работоспособности функции *simulationMarkov*-имитационное моделирование и подробный вывод траектории случайного процесса. Дополнительная функция *eval1* выводит на печать отладочную информацию:

```

Пример кода.
SeedRandom[1234567]
eval1[q1_,q2_]:=Print[{q1,q2}," Переход ",q1[[1]]->q2[[1]],
" время задержки ",q2[[2]]-q1[[2]]]
s0 = 1;
tmod = 3;
simulationMarkov[eval1,s0,tmod]

```

Вывод результатов

Старое и новое состояния	Переход	Время задержки
{{1,0},{2,0.682554}}	1->2	0.6825
{{2,0.682554},{3,0.691296}}	2->3	0.0087
{{3,0.691296},{1,0.864915}}	3->1	0.1736
{{1,0.864915},{4,0.9777}}	1->4	0.1128

18}}		
{4,0.977718},{5,1.14319}}	4->5	0.1655
{{5,1.14319},{1,1.39558}}	5->1	0.2524
{{1,1.39558},{2,1.44917}}	1->2	0.0536
{{2,1.44917},{3,1.46339}}	2->3	0.0142
{{3,1.46339},{1,1.54172}}	3->1	0.0783
{{1,1.54172},{4,1.95789}}	1->4	0.4162
{{4,1.95789},{1,2.14341}}	4->1	0.1855
{{1,2.14341},{2,2.57345}}	1->2	0.4300
{{2,2.57345},{3,2.57774}}	2->3	0.0043
{{3,2.57774},{1,2.70275}}	3->1	0.1250
{{1,2.70275},{4,3.8344}}	1->4	1.1317

Результат работы функции – вывод последнего значения {4,3.8344}

Пример 2. Дополнительная внешняя функция *times* - - вычисление суммарного времени *tall* нахождения системы в каждом из состояний, начальное суммарное время равно 0.

Пример кода.

```
times[q1_,q2_] := tall[[q1[[1]]]]+ = (q2[[2]] - q1[[2]])
```

```
SeedRandom[1234567]
```

```
tall = ConstantArray[0,Length @ S];
```

```
tmod = 1000;
```

```
simulationMarkov[times,1,tmod];
```

```
tall
```

```
Total[tall]
```

```
tall / Total[tall]
```

Результаты

Абсолютные времена нахождения в каждом из состояний {609, 104, 76, 66, 146}.

Общее время 1000.33

Относительные времена нахождения в каждом из состояний системы {0.61,0.10,0.08,0.07,0.15}

6.6. Решение системы дифференциальных уравнений Колмогорова

Введем функцию $p[k, t]$ - вероятность нахождения системы в состоянии k в момент времени t и матрицу интенсивности переходов $\lambda p = P$. Каждое из системы уравнений Колмогорова $eq[k]$ равно

$eq[k] :=$

$D[p[k, t], t] \text{ } \mathbb{L} \text{ } - \text{Sum}[\lambda p[[k, i]] p[k, t], \{i, 1, n\}] + \text{Sum}[\lambda p[[i, k]] p[i, t], \{i, 1, n\}]$

Например $eq[1]$ задается, в соответствии с матрицей P , следующим образом

$$p^{(0,1)}[1, t] \text{ } \mathbb{L} \text{ } - 2p[1, t] + 8p[3, t] + 7p[4, t] + p[5, t]$$

Получаем систему дифференциальных уравнений Колмогорова

$$eqv = \text{Table}[eq[k], \{k, 1, n\}]$$

или

$$\{p^{(0,1)}[1, t] \text{ } \mathbb{L} \text{ } - 2p[1, t] + 8p[3, t] + 7p[4, t] + p[5, t],$$

$$p^{(0,1)}[2, t] \text{ } \mathbb{L} \text{ } p[1, t] - 6p[2, t],$$

$$p^{(0,1)}[3, t] \text{ } \mathbb{L} \text{ } 5p[2, t] - 8p[3, t] + p[4, t],$$

$$p^{(0,1)}[4, t] \text{ } \mathbb{L} \text{ } p[1, t] - 9p[4, t],$$

$$p^{(0,1)}[5, t] \text{ } \mathbb{L} \text{ } p[2, t] + p[4, t] - p[5, t]\}$$

Введем обозначения для набора функций решений системы уравнений $pv[t]$ и начальных условий $pv0$. Одновременно зададим некоторый набор начальных условий $p00 = \{1, 0, 0, 0, 0\}$. Соответствующее равенство задаётся с помощью функции $\text{Thread}[pv0 \text{ } \mathbb{L} \text{ } p00]$:

$$pv[t _] = \text{Table}[p[k, t], \{k, 1, n\}]$$

$$pv0 = \text{Table}[p[k, 0], \{k, 1, n\}]$$

$$p00 = \{1, 0, 0, 0, 0\}$$

Численное решение системы дифференциальных уравнений с начальными условиями на интервале времени $[0, t0]$ реализует следующий код:

$t0 = 10;$

$$nds = \text{NDSolve}[eqv \sim \text{Join} \sim \text{Thread}[pv0 \text{ } \mathbb{L} \text{ } p00], pv[t], \{t, 0, t0\}]$$

Результат получается в виде набора подстановок, каждое из решений представлено в виде интерполяционной функции:

```

{{p[1, t] -> InterpolatingFunction[{{0., 10.}}, <>][t],
 p[2, t] -> InterpolatingFunction[{{0., 10.}}, <>][t],
 p[3, t] -> InterpolatingFunction[{{0., 10.}}, <>][t],
 p[4, t] -> InterpolatingFunction[{{0., 10.}}, <>][t],
 p[5, t] -> InterpolatingFunction[{{0., 10.}}, <>][t]}

```

Окончательный результат в виде набора обычных функций

```

result[t_] = Flatten @ Table[p[k, t] /. nds, {k, 1, n}]

```

Можно построить график (рис.35) всех функций решения начальной задачи

```

Plot[result[t], {t, 0, t0}, PlotRange -> {0, 1}].

```

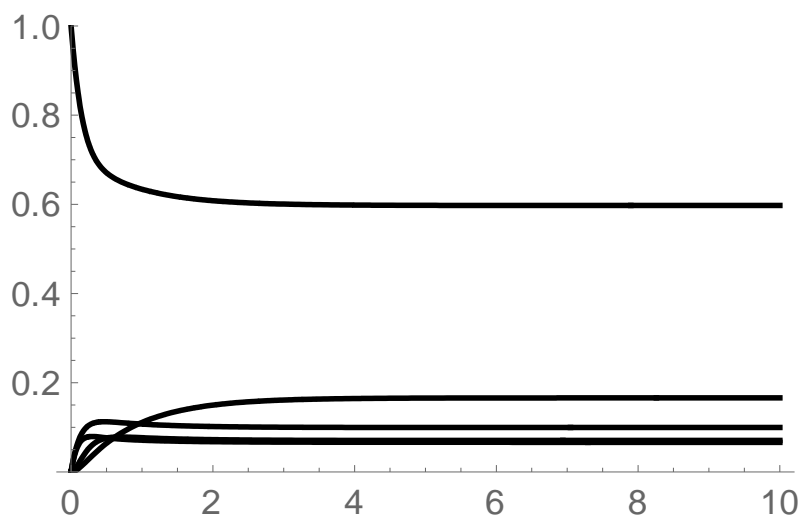


Рис. 34

Видно, что уже при $t > 2$ происходит стабилизация решения к некоторому финальному распределению вероятности. При $t = 10$ получаем

```

result[10] = {0.5975, 0.0996, 0.0705, 0.0664, 0.1660}

```

Полученные результаты можно сравнить с результатами пункта 6.5.

СПИСОК ЛИТЕРАТУРЫ

1. Вентцель, Е.С. Теория случайных процессов и ее инженерные приложения: учебное пособие / Е. С. Вентцель, Л. А. Овчаров. — 5-е изд., стер. — М.: КНОРУС, 2013. — 448 с.
2. Климов, Г. П. Теория массового обслуживания [Электронный ресурс] : учебное пособие / Климов Г. П. - Москва : Московский государственный университет имени М.В. Ломоносова, 2011. - 312 с.
3. Маликов, Р. Ф. Основы математического моделирования [Электронный ресурс]: учебное пособие / Маликов Р. Ф. - Москва: Горячая линия - Телеком, 2010. - 368 с.

4. Баскин, Л. М. Применение пакета "МАТНЕМАТІСА" для выполнения вычислений [Электронный ресурс] : учебно-методическое пособие по выполнению самостоятельной работы / Л. М. Баскин ; Федеральное агентство связи, Федеральное государственное образовательное бюджетное учреждение высшего профессионального образования "СПбГУТ им. проф. М. А. Бонч-Бруевича". - СПб. : СПбГУТ, 2014. - 42 с.