МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Пятигорский институт (филиал) СКФУ

Методические указания

по выполнению лабораторных работ
по дисциплине «**ТЕОРИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ**»
для студентов направления подготовки /специальности

09.03.02 Информационные системы и технологии

(ЭЛЕКТРОННЫЙ ДОКУМЕНТ)

СОДЕРЖАНИЕ

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью изучению дисциплины «Теория информационных процессов и систем» является освоение студентами теоретических и практических основ создания информационных систем, а также способов описания информационных систем.

Задачи дисциплины: определить современное состояние развития информационных систем и их классификацию; познакомить с методологиями и технологиями разработки информационных систем; представить этапы создания и проектирования информационных систем.

3. ЛАБОРАТОРНЫЕ РАБОТЫ

Лабораторная работа №1.

Тема 1. Общие понятия информационных процессов. Основные понятия теории систем

Нахождение энтропии дискретного сообщения.

Цель работы. Научиться практически определять количество информации в дискретных сообщениях.

Теоретические основы.

Количество информации, содержащееся в дискретном сообщении (I) можно найти из простого соотношения:

$$I=n\cdot H,$$

где n — число символов в сообщении, H — энтропия источника сообщений, то есть среднее количество информации, приходящееся на один символ.

Энтропия источника сообщения определяется из основного соотношения теории информации, которое для удобства практического использования преобразуется к виду наиболее простому и удобному в зависимости от свойств дискретного источника сообщений. В случае, если символы источника сообщения появляются равновероятно и взаимно независимо, то для подсчета энтропии такого рода сообщений используют формулу Хартли:

$$I = n \cdot \log_2 m(бит), H_1 = \log_2 m(^{бит}/_{символ}),$$

где m — объем алфавита источника дискретных сообщений. Если символы источника сообщения генерируются с различными вероятностями, но взаимно независимы, то используют формулу Шеннона:

$$I = -n \cdot \sum_{i=1}^{m} P_{a1} \cdot \log_2 P_{a1}(бит),$$

$$H_2 = -\sum_{i=1}^{m} P_{a1} \cdot \log_2 P_{a1}(^{бит}/_{символ}),$$

где P_{a1} — вероятность появления символа a_1.

В случае же не равновероятного появления символов источника сообщения и наличия статистических зависимостей между соседними символами энтропию такого источника можно определить с помощью формулы Шеннона с условными вероятностями:

$$H_2 = -\sum_{i=1}^{m} P_{a1} \cdot \sum_{i=1}^{m} P\left(^{a_j}/_{a_i}\right) \cdot \log_2 P\left(^{a_j}/_{a_i}\right)(^{бит}/_{символ}),$$

где $P\left(^{a_j}/_{a_i}\right)$ — условная вероятность появления символа a_i после символа a_i

Содержание работы.

4. Посчитать среднее количество информации, приходящееся на один символ (энтропию) источника дискретных сообщений в случаях:

а) — равновероятного и взаимно независимого появления символов;

б) — не равновероятного и взаимно независимого появления символов;

в) — при не равновероятном появлении символов и наличии статистических связей между соседними символами.

В качестве дискретного источника сообщений взять источник с объемом алфавита m= 34 (аналогичный по объему алфавита тексту на русском языке: 33 буквы и пробел), а его статистические характеристики смоделировать с помощью генератора случайных чисел.

5. Подсчитать количество информации в сообщении, представляющим собой Вашу фамилию, имя и отчество, считая, что символы сообщения появляются не равновероятно и независимо. Закон распределения символов найти путем анализа участка любого текста на русском языке длиной не менее 300 символов.

Выполнение.

Работа выполняется на персональном компьютере в программном средстве «Mathcad». Так как в этом программном продукте в качестве встроенных функций используются только функции натуральных и десятичных логарифмов, то в процессе выполнения работы необходимо выполнить переход к логарифмам по основанию 2 по формуле перехода к иному основанию

$$\log_b N = \frac{\log_a N}{\log_a b},$$

где a — основание известных логарифмов;

б — основание требуемых логарифмов;
N — логарифмируемая величина.

П.1.а. Используя формулу Хартли, найти энтропию указанного источника дискретных сообщений (H_1).
П.1.б. Смоделировать закон распределения символов дискретного источника сообщений, используя оператор rnd (A), который генерирует случайные числа из диапазона [0, A] по следующей программе:
m: = 34 — задание объема алфавита (m);
i: = 1, 2, …, m — i - порядковый номер символа алфавита;
$r(i)$ = rnd (1) — генерирование 34 случайных чисел
в интервале от 0 до 1;
$l := \sum_i r(i)$ — нахождение суммы всех $r(i)$;
$P(i) := \frac{r(i)}{l}$ — $P(i)$ – вероятность появления i-го символа (a_i).
Проверить правильность вычислений, найдя сумму всех $P(i)$ при i= 1,2,…,m.
Построить график закона распределения $P(i)$. Используя формулу Шеннона, определить энтропию смоделированного источника дискретных сообщений (H_2).
П.1.в. Смоделировать матрицу условных вероятностей появления символа a_j после символа a_i по следующей программе:
m: =34 -— задание объема алфавита (m);
$\left.\begin{array}{l} i := 1,2,...m \\ j := 1,2,...m \end{array}\right\}$ — порядковый номер символа алфавита;
$r(i,j)$:= rnd (1) — генерирование матрицы (34X34) случайных чисел в интервале от 0 до1; $W_i := \sum_i r(i,j)$ — нахождение суммы элементов в каждой строке матрицы $r(i,j)$;
$S(i,j) := \frac{r(i,j)}{W_i}$ — нормировка по строкам матрицы $r(i,j)$ с целью получения суммы элементов в каждой строке, равной 1;
$U_j := \sum_i S(i,j)$ — нахождение сумм элементов в каждом столбце матрицы $S(i,j)$;
$PP(i,j) := \frac{S(i,j)}{U_j}$ — нормировка по столбцам матрицы $S(i,j)$ с целью получения суммы элементов в каждом столбце, равной 1.

Полученные значения элементов матрицы $PP(i,j)$ приближенно можно считать условными вероятностями появления символа под номером j после i-го символа. Используя формулу Шеннона с условными вероятностями определить энтропию смоделированного источника дискретных сообщений (H_3).
П.2. Определить вероятность появления каждого символа (буквы) P_i путем деления числа появлений этого символа (a_i) на общее число символов (не менее 300), входящих в сообщение. В случае, если какой-либо символ (из m= 34) в сообщении не встретился, считать, что он встретился 1 раз, иначе может возникнуть неопределенность в формуле Шеннона. Отсутствие какого-либо символа, принадлежащего алфавиту, свидетельствует лишь о том, что анализируемое сообщение не содержит достаточного числа символов (не достаточно длинное).

Построить график закона распределения символов (букв). Проверить правильность полученного закона распределения, для чего найти сумму вероятностей появления каждого символа. Эта сумма должна быть равна 1.

С помощью формулы Шеннона найти энтропию (H_4) дискретного источника (текста на русском языке). Подсчитав число символов в Вашей фамилии, имени и отчестве (включая пробелы), найти количество информации, содержащейся в этом сообщении.

Контрольные вопросы.

1. Какие источники сообщений называют дискретными?
2. Для каких источников дискретных сообщений применимы формулы Хартли, Шеннона?
3. Каким образом описывается статистическая зависимость между соседними символами в дискретных сообщениях?
4. Дайте определение энтропии источника дискретных сообщений.
5. Как проверить правильность нахождения закона распределения символов источника дискретных сообщений?
6. Какой вид дискретных сообщений обладает наибольшей энтропией?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-2	1-2

Оценочные средства: письменный отчет, собеседование.

Лабораторная работа № 2.

Тема 6. Алфавитное кодирование информации. Критерий взаимной однозначности алфавитного кодирования

Определить, является ли код, с заданным кодирующим алфавитом однозначно декодируемым.

Цель работы. Используя критерий взаимной однозначности алфавитного кодирования, определить является ли заданная схема кодирования взаимно-однозначной.

Теоретические основы.

В общем случае задачу кодирования можно представить следующим образом. Пусть заданы два *алфавита A* и *B*, состоящие из конечного числа символов:

$$A = \{a_1, a_2, ..., a_n\}, \quad B = \{b_1, b_2, ..., b_m\}.$$

Элементы алфавита называются *буквами*. Упорядоченный набор в алфавите *A* назовем *словом*

$$\alpha = a_{i_1}, a_{i_2}, ..., a_{i_n}, ..., a_{i_n},$$

где $i \in [1, n]$, число *n* показывает количество букв в слове и называется *длиной* слова α, обозначается: $n = l(\alpha) = |\alpha|$.

Пустое слово обозначается:

$$\Lambda : l(\Lambda) = |\Lambda| = 0$$

Для слова

$$\alpha = a_{i_1}, a_{i_2}, ..., a_{i_i}, ..., a_{i_n}$$

буква a_1, называется *началом*, или *префиксом*, слова α, а буква a_n – *окончанием*, или *постфиксом*, слова α. Слова можно соединять. Для этого префикс второго слова должен следовать сразу за постфиксом первого, при этом в новом слове они, естественно, утрачивают свой статус, если только одно из слов не было пустым. Соединение слов α_1 и α_2 обозначается α_1, α_2, соединение *n* одинаковых слов α обозначается α^n, причем $\alpha^0 = \Lambda$. Множество всех непустых слов алфавита *A* обозначим *A**:

$$A* = \{\alpha \mid l(\alpha) > 0\}$$

Множество *A* называют *алфавитом* сообщений, а множество *B* – *кодирующим алфавитом*. Множество слов, составленных в алфавите *B*, обозначим *B**. Обозначим через *F* отображение слов алфавита *A* в алфавит *B*. Тогда слово β=*F*(α) назовем *кодом* слова α.

Кодированием называют универсальный способ отображения информации при ее хранении, передаче и обработке в виде системы соответствий между элементами сообщений и сигналами, при помощи которых эти элементы можно зафиксировать. Таким образом, *код* – правило однозначного преобразования (т.е. функция) сообщения из одной символической формы представления (исходного алфавита *A*) в другую (объектный алфавит *B*), обычно без каких-либо потерь информации. Процесс преобразования *F*: *A**→*B** слов исходного алфавита *A* в алфавит *B* называется *кодированием* информации. Процесс обратного преобразования слова βЄ*B** в слово αЄ*A** называется *декодированием*. Таким образом, декодирование – функция, обратная *F*, т.е. F^{-1}. Так как для любого кодирования должна выполняться операция декодирования, то отображение должно быть обратимым (*биекция*). Если |*B*|=*m*, то *F* называется *m-ичным кодированием*, наиболее распространенный случай *B* = {0, 1} – *двоичное* кодирование. Именно этот случай и рассматривается в дальнейшем.

Если все кодовые слова имеют одинаковую длину, то код называется *равномерным*, или *блочным*. Алфавитное (или побуквенное), кодирование можно задать *таблицей кодов*. Кодом или кодирующей функцией, будет служить некоторая подстановка σ. Тогда

$$\sigma = \begin{pmatrix} \alpha_1 & \alpha_2 & \dots\alpha\dots\beta & \alpha_n \\ \beta_1 & \beta_2 & & \beta_n \end{pmatrix}$$

Такое побуквенное кодирование обозначается K^{α}_{β}. Множество кодов букв $\{\beta_i\}$ называется *множеством элементарных кодов*. Алфавитное кодирование можно использовать для любого множества сообщений. Таким образом, алфавитное кодирование является самым простым, и его всегда можно ввести на непустых алфавитах.

Содержание работы.

Определить, является ли код, с кодирующим алфавитом {0;1;2} однозначно декодируемым.

Варианты:

1. {01;201;112;122;0112;0102}
2. {001;021;102;201;001121;01012101}
3. {01;110;102;222;0010001001}
4. {20;01202;22;2001;2012010;1112}
5. {01;011;100;2100;101210;001210}
6. {01;011;100;2100;10110;00112}
7. {01;12;021;0102;10112;22221}
8. {01;012;111;0102;10112;01112}
9. {01;02;12;012;0102;020112}
10. {01;10;210;121;0210;0112}

Выполнение.

Для вывода общего критерия взаимной однозначности понадобится следующее понятие. Представление элементарного кода β_i схемы алфавитного кодирования в виде

$$\beta_i = \beta` \beta_{i1} \dots \beta_{in} \beta``$$

где слово β` не может оканчиваться на элементарный код, а слово β`` начинаться с элементарного кода, будем называть нетривиальным разложением элементарного кода. При этом одно из слов β` или β`` может быть пустым (^). Для определения однозначности или неоднозначности схемы алфавитного кодирования

существует эффективный алгоритм, использующий понятие нетривиального разложения элементарных кодов.

1. Для каждого элементарного кода выписываем все нетривиальные разложения.
2. Выписываем множество M_1 состоящее из слов β`, которые входят в качестве начал в нетривиальные разложения элементарных кодов.
3. Выписываем множество M_2, состоящее из всех слов β``, которые являются окончанием нетривиальных разложений элементарных кодов.
4. Составляем множество $M = M_1 \cap M_2 \cup \{ \wedge \}$, т.е. множество слов, встречающихся как в качестве начала, так и в качестве окончания в нетривиальных разложениях элементарных кодов.
5. Выписываем все разложения элементарных кодов, связанных с множеством *M*, т.е. все разложения элементарных кодов вида:

$$\beta_i = \beta` \beta_{i1} \dots \beta_{in} \beta``,$$

где β`, β`` ∈ *M*, а *k* может быть равно 0.

6. По разложениям, полученным в пункте 5, строится ориентированный граф следующим образом. Вершины графа отождествляют с элементами множества *M*. Пара вершин β`, и β`` соединяются ориентированными ребрами в том и только в том случае, если существует разложение $\beta` \beta_{i1} \dots \beta_{in} \beta``$. При этом ребру (β`, β``) приписывается слово $\beta_{i1} \dots \beta_{in}$, соответствующее этому разложению.
7. По полученному графу легко проверить, обладает или нет исходная схема кодирования свойством взаимной однозначности.

Теорема А.А. Маркова: Алфавитное кодирование не обладает свойством взаимной однозначности тогда и только тогда, когда граф содержит ориентированный цикл, проходящий через вершину ^.

Контрольные вопросы.

1. Что такое кодирование информации и для чего оно используется?
2. Охарактеризуйте основные принципы кодирования.
3. Что такое алфавиты и как они используются?
4. Понятие элементарного кода.
5. Нетривиальное разложение элементарного кода.
6. Сформулируйте теорему Маркова.
7. Что определяет неравенство Макмиллана?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-2	1-2

Оценочные средства: письменный отчет, собеседование.

Лабораторная работа № 3.

Тема 8. Метод Хаффмана.

Закодировать заданное выражение методом Хаффмана.

Цель работы.

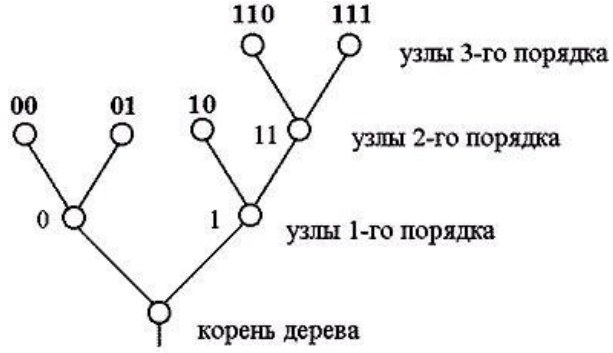
Использование метода эффективного кодирования.

Теоретические основы.

Введем понятие кодового дерева для множества кодовых слов. Наглядное графическое изображение множества кодовых слов можно получить, установив соответствие между сообщениями и концевыми узлами двоичного дерева. Пример двоичного кодового дерева изображен на рисунке 3. Две ветви, идущие от корня дерева к

узлам первого порядка, соответствуют выбору между 0 и 1 в качестве первого символа кодового слова: левая ветвь соответствует 0, а правая – 1. Две ветви, идущие из узлов первого порядка, соответствуют второму символу кодовых слов, левая означает 0, а правая – 1 и т.д. Ясно, что последовательность символов каждого кодового слова определяет необходимые правила продвижения от корня дерева до концевого узла, соответствующего рассматриваемому сообщению.

Формально кодовые слова могут быть приписаны также промежуточным узлам. Например, промежуточному узлу второго порядка на рис. 3 можно приписать кодовое слово 11, которое соответствует первым двум символам кодовых слов, соответствующих концевым узлам, порождаемых этим узлом. Однако кодовые слова, соответствующие промежуточным узлам, не могут быть использованы для представления сообщений, так как в этом случае нарушается требование префиксности кода. Требование, чтобы только концевые узлы сопоставлялись сообщениям, эквивалентно условию, чтобы ни одно из кодовых слов не совпало с началом (префиксом) более длинного кодового слова.



Пример двоичного кодового дерева

Любой код, кодовые слова которого соответствуют различным концевым вершинам некоторого двоичного кодового дерева, является префиксным, т.е. однозначно декодируемым.

Одним из часто используемых методов эффективного кодирования является так называемый код Хаффмана.

Содержание работы.

Осуществить кодирование методом Хаффмана заданной фразы:

1. A man who has not been flogged is not educated
2. Many a good cow hath an evil (bad) calf
3. Many a good father has but a bad son
4. Many a little makes a tickle
5. Many a true word is spoken in jest
6. Many commanders sink the ship
7. Many physicians have killed the king
8. Marriages are made in heaven
9. A man is known by the company he keeps
10. Man proposes, God disposes

Выполнение.

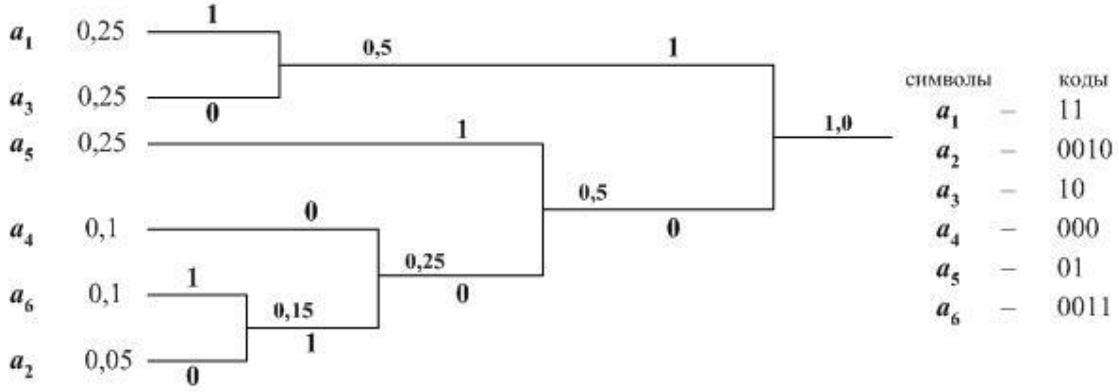
Шаг 1. Упорядочиваем символы исходного алфавита в порядке невозрастания их вероятностей. (Записываем их в столбец.)

Шаг 2. Объединяем два символа с наименьшими вероятностями. Символу с большей вероятностью приписываем "1", символу с меньшей – "0" в качестве элементов их кодов.

Шаг 3. Считаем объединение символов за один символ с вероятностью, равной сумме вероятностей объединенных символов.

Шаг 4. Возвращаемся на Шаг 2 до тех пор, пока все символы не будут объединены в один с вероятностью, равной единице.

Закодируем тот же 6-буквенный алфавит методом Хаффмана, изобразив соответствующее двоичное дере.



Двоичное дерево, соответствующее кодированию по методу Хаффмана

Считывание кода идет от корня двоичного дерева к его вершинам с обозначением символов. Это обеспечивает префиксность кода. Метод Хаффмана также содержит неоднозначность, поскольку в алфавите могут оказаться несколько символов с одинаковой вероятностью, и код будет зависеть от того, какие символы мы будем объединять в первую очередь.

Контрольные вопросы.

1. В чем смысл понятия: избыточность сообщения?
2. Что определяет коэффициент избыточности?
3. Алгоритм построения дерева Хаффмана.
4. Какие задачи решает кодирование методом Хаффмана?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-2	1-2

Оценочные средства: письменный отчет, собеседование.

Лабораторная работа № 4.

Тема 12. Марковские случайные процессы, классификация. Марковские цепи

Определить вероятности состояний системы после заданного количества интервалов времени.

Цель работы.

Освоить понятия: Классификация Марковских процессов. Граф состояний цепи Маркова. Матрица вероятностей перехода, ее свойства. Рекуррентная формула для вычисления вероятностей состояний системы.

Теоретические основы.

Марковский случайный процесс с дискретными состояниями и дискретным временем называют *марковской цепью*. Для такого процесса моменты t_1, t_2, …, t_k, когда система S может менять свое состояние, рассматривают как последовательные шаги процесса, а в качестве аргумента, от которого зависит процесс, выступает не время t, а номер шага 1, 2, …, k. Случайный процесс в этом случае характеризуется последовательностью состояний $S(0)$, $S(1)$, $S(2)$, …, $S(k)$, …, где $S(0)$ – начальное состояние системы (перед первым шагом); $S(1)$ – состояние системы после первого шага; $S(k)$ – состояние системы после k-го шага. Событие $\{S(k) = S_i\}$, состоящее в том, что сразу после k-го шага система находится в состоянии S_i (i = 1, 2, …), является случайным событием. Последовательность состояний $S(0)$, $S(1)$, $S(2)$, …, $S(k)$, … можно рассматривать

как последовательность случайных событий. Начальное состояние $S(0)$ может быть заданным заранее или случайным. Вероятностями состояний цепи Маркова называются вероятности $P_i(k)$ того, что после k-го шага (и до $(k + 1)$-го) система S будет находиться в состоянии S_i ($i = 1, 2,\ldots,n$). Очевидно, что для любого k выполняется условие:

$$\sum_{i=1}^{n} P_i(k) = 1$$

Начальным распределением вероятностей марковской цепи называется распределение вероятностей состояний в начале процесса:

$$P_1(0), P_2(0), \ldots, P_i(0), \ldots, P_n(0).$$

В частном случае, если начальное состояние системы S в точности известно $S(0) = S_i$, то начальная вероятность $P_i(0) = 1$, а все остальные равны нулю. Вероятностью перехода (переходной вероятностью) на k-ом шаге из состояния S_i в состояние S_j называется вероятность того, что система S после k-го шага окажется в состоянии S_j при условии, что непосредственно перед этим (после $(k\text{-}1)$-го шага) она находилась в состоянии S_i. Поскольку система может пребывать в одном из n состояний, то для каждого момента времени t необходимо задать n^2 вероятностей перехода P_{ij}, которые удобно представить в виде следующей матрицы:

$$\left(P_{ij}\right) = \begin{pmatrix} P_{11} & P_{12} & \ldots & P_{1n} \\ P_{21} & P_{22} & \ldots & P_{2n} \\ \ldots\ldots & & \ldots & \ldots \\ P_{i1} & P_{i2} & \ldots & P_{in} \\ & & & \\ & & \ldots\ldots\ldots & \\ P_{n1} & P_{n2} & \ldots & P_{nn} \end{pmatrix}$$

где P_{ij} – вероятность перехода за один шаг из состояния S_i в состояние S_j;
P_{ii} – вероятность задержки системы в состоянии S_i.

Эта матрица называется переходной или матрицей переходных вероятностей. Если переходные вероятности не зависят от номера шага (от времени), а зависят только от того, из какого состояния в какое осуществляется переход, то соответствующая цепь Маркова называется *однородной*. Переходные вероятности однородной марковской цепи P_{ij} образуют квадратную матрицу размера $n \times n$. Отметим некоторые ее особенности:

1. Каждая строка характеризует выбранное состояние системы, а ее элементы представляют собой вероятности всех возможных переходов за один шаг из выбранного (из i-го) состояния, в том числе и переход в самое себя;

2. Элементы столбцов показывают вероятности всех возможных переходов системы за один шаг в заданное (j-ое) состояние (иначе говоря, строка характеризует вероятность перехода системы из состояния, а столбец – в состояние);

3. Сумма вероятностей каждой строки равна единице, так как переходы образуют полную группу несовместных событий:

$$\sum_{j=1}^{n} P_{ij} = 1,\ i = 1, \ldots, n;$$

4. По главной диагонали матрицы переходных вероятностей стоят вероятности P_{ii} того, что система не выйдет из состояния S_i, а останется в нем. Если для однородной марковской цепи заданы начальное распределение вероятностей и матрица переходных вероятностей, то вероятности состояний системы $P_i(k)$, ($i = 1, \ldots, n$; $j = 1, \ldots, n$) определяются по рекуррентной формуле:

$$P_{i}(k) = \sum_{j=1}^{n} P_{ji,(i=1,\ldots,n)} P_{j}(k-1)$$

Содержание работы.

Даны матрицы переходных характеристик. Определить вероятности состояний системы после заданного количества интервалов времени (N). Привести сигнальный граф системы и все вычисления. Начальные условия: $P_1(0)=1$, $P_2(0)=P_3(0)= ... =0$.

Варианты:

1. N=3

0,5	0,3	0,2	0
0	0,4	0,4	0,2
0	0	0,3	0,7
0	0	0	1

2.N=4

0,5	0,3	0,2
0	0,4	0,6
0,7	0	0,3

3. N=3

0,5	0,3	0,2	0
0	0,4	0,3	0,3
0	0	0,4	0,6
0	0	0	1

4. N=4

0,5	0,3	0,2
0	0,4	0,6
0	0	1

5. N=3

0,5	0,3	0,2	0
0	0,4	0,4	0,2
0	0	0,3	0,7
0	0	0	1

Выполнение.

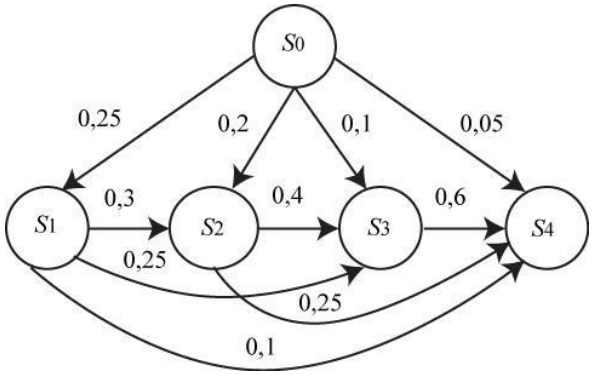
Сформулируем методику моделирования по схеме дискретных марковских процессов (марковских цепей).

1. Зафиксировать исследуемое свойство системы. Определение свойства зависит от цели исследования.
2. Определить конечное число возможных состояний системы и убедиться в правомерности моделирования по схеме дискретных марковских процессов.
3. Составить и разметить граф состояний.
4. Определить начальное состояние.
5. По рекуррентной зависимости определить искомые вероятности.

Пример. По группе из четырех объектов производится три последовательных выстрела. Найти вероятности состояний группы объектов после третьего выстрела. Матрица переходных вероятностей имеет вид:

$$||P_{ij}|| = \begin{pmatrix} 0,4 & 0.25 & 0.2 & 0.1 & 0.05 \\ 0 & 0.35 & 0.3 & 0.25 & 0.1 \\ 0 & 0 & 0.45 & 0.4 & 0.15 \\ 0 & 0 & 0 & 0.4 & 0.6 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Размеченный граф состояний приведен на рисунке.



Размеченный граф состояний четырех объектов

Решение:

Так как до первого выстрела все объекты целы, то $p_1(0) = 1$.

После первого выстрела все значения вероятностей $p_j(1)$ соответствуют первой строке матрицы переходных вероятностей. Рассчитаем вероятности остальных состояний.

$$P_1(2) = P_1(1) \cdot p_{11} + P_2(1) \cdot p_{21} + P_3(1) \cdot p_{31} + P_4(1) \cdot p_{41} + P_5(1) \cdot p_{51} = 0,16;$$
$$P_2(2) = P_1(1) \cdot p_{12} + P_2(1) \cdot p_{22} + P_3(1) \cdot p_{32} + P_4(1) \cdot p_{42} + P_5(1) \cdot p_{52} = 0,19;$$
$$P_3(2) = P_1(1) \cdot p_{13} + P_2(1) \cdot p_{23} + P_3(1) \cdot p_3 3 + P_4(1) \cdot p_{43} + P_5(1) \cdot p_5 3 = 0,245;$$
$$P_4(2) = P_1(1) \cdot p_{14} + P_2(1) \cdot p_{24} + P_3(1) \cdot p_{34} + P_4(1) \cdot p_{44} + P_5(1) \cdot p_{54} = 0,22;$$
$$P_5(2) = P_1(1) \cdot p_{15} + P_2(1) \cdot p_{25} + P_3(1) \cdot p_{35} + P4(1) \cdot p_{45} + P_5(1) \cdot p_{55} = 0,185;$$
$$P = 0,16 + 0,19 + 0,245 + 0,22 + 0,185 = 1.$$
$$P_1(3) = P_1(2) \cdot p_{11} = 0,16 \cdot 0,4 = 0,064;$$
$$P_2(3) = P_1(2) \cdot p_{12} + P_2(2) \cdot p_{22} = 0,16 \cdot 0,25 + 0,19 \cdot 0,35 = 0,11;$$
$$P_3(3) = P_1(2) \cdot p_{13} + P_2(2) \cdot p_{23} + P_3(2) \cdot p_3 3 == 0,16 \cdot 0,2 + 0,19 \cdot 0,3 + 0,245 \cdot 0,45 = 0,2;$$
$$P_4(3) = P_1(2) \cdot p_{14} + P_2(2) \cdot p_{24} + P_3(2) \cdot p_{34} + P_4(2) \cdot p_{44} = 0,25;$$
$$P_5(3) = P_1(2) \cdot p_{15} + P_2(2) \cdot p_{25} + P_3(2) \cdot p_{33} + P_4(2) \cdot p_{45} + P_5(2) \cdot p_{55} = 0,38;$$
$$P = 0,064 + 0,11 + 0,2 + 0,25 + 0,38 = 1.$$

Контрольные вопросы.

1. Какие процессы называются Марковскими?
2. Что называется цепью Маркова?
3. Перечислите свойства матрицы вероятностей перехода.
4. Какой Марковский процесс называется однородным?

Работа с литературой:

Рекомендуемые источники информации (№ источника)			
Основная	Дополнительная	Методическая	Интернет-ресурсы
1-2	1	1-2	1-2

Оценочные средства: письменный отчет, собеседование.

4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

4.1. Рекомендуемая литература

4.1.1. Основная литература:

1.Чернышев, А.Б. Теория информационных процессов и систем : учебное пособие / А.Б. Чернышев, В.Ф. Антонов, Г.Б. Суюнова ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет». – Ставрополь : СКФУ, 2015. – 169 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=457890 – Библиогр. в кн. – Текст : электронный.

2. Шкундин, С.З. Теория информационных процессов и систем : учебное пособие / С.З. Шкундин, В.Ш. Берикашвили. – Москва : Горная книга, 2012. – 475 с. – Режим

доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=229031 – ISBN 978-5-98672-285-6. – Текст : электронный.

4.1.2. Дополнительная литература:

1. Душин, В.К. Теоретические основы информационных процессов и систем : учебник / В.К. Душин. - 5-е изд. - М. : Дашков и Ко, 2014. - 348 с. - ISBN 978-5-394-01748-3 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=221284

4.1.3. Методическая литература:

1. Методические указания по выполнению лабораторных работ по дисциплине «Теория информационных процессов и систем».
2. Методические рекомендации для студентов по организации самостоятельной работы по дисциплине «Теория информационных процессов и систем».

4.1.4. Интернет-ресурсы:

1. http://www.intuit.ru – сайт дистанционного образования в области информационных технологий
2. http://window.edu.ru – образовательные ресурсы ведущих вузов

4.1.5. Программное обеспечение

1. Операционная система Windows версия XP и выше, браузер Internet Explorer или любой другой, интегрированный пакет MathCad.

5.2. Материально-техническое обеспечение дисциплины

Лекционный курс проводится в аудиториях, оснащенных мультимедийным проектором. Лабораторные занятия проводятся в компьютерных классах, в которых установлена программа Microsoft Visual Studio 2010, 2012, а также другие системы для разработки программных приложений.