

**РОСЖЕЛДОР**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Ростовский государственный университет путей сообщения»**  
**(ФГБОУ ВО РГУПС)**

---

О.И. Соколова

**ТЕОРИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ**

Учебно-методическое пособие

Ростов-на-Дону  
2017

УДК 681.5 (07) + 06

**Соколова, О.И.**

Теория информационных процессов и систем: [Электронный ресурс] учебно-методическое пособие / О.И. Соколова; ФГБОУ ВО РГУПС. – Ростов н/Д, 2017. – 47 с. : ил. – Библиогр. : 10 назв.

Приведены задания для выполнения лабораторных работ по дисциплине «Теория информационных процессов и систем». Также даны необходимые сведения об информации, информационных процессах и системах, отражающие историю развития этих понятий. Представлены методические указания по исследованию количественных мер информации, качественных и количественных свойств сложных информационных процессов и систем. Показаны способы описания и исследования фрактальной структуры информационных процессов.

Учебно-методическое пособие предназначено для студентов направления подготовки 09.03.02 Информационные системы и технологии и 09.03.01 Информатика и вычислительная техника, а также для студентов всех специальностей, изучающих дисциплину «Теория информационных процессов и систем».

Одобрено к внесению в «Электронный университет» кафедрой «Вычислительная техника и автоматизированные системы управления».

© Соколова О.И., 2017

© «Электронный университет» ФГБОУ ВО РГУПС, 2017

## ОГЛАВЛЕНИЕ

Задание к лабораторной работе № 1 «Исследование свойств информационных мер» .....	4
Задание к лабораторной работе № 2 «Изучение и исследование качественных характеристик сложных информационных систем» .....	15
Задание к лабораторной работе № 3 «Исследование количественных характеристик устойчивости динамических систем» .....	23
Задание к лабораторной работе № 4 «Исследование фрактальных свойств информационных систем».....	31
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	46

## Задание к лабораторной работе № 1 «Исследование свойств информационных мер»

**Цель занятия:** Получить навыки исследования информационных объектов на основе классической теории информации.

### **Задачи на работу:**

1 Провести анализ основных существующих мер информации, применяемых в информационных системах.

2 Разработать сводную таблицу сравнительных характеристик информационных мер на основе качественного подхода к информации.

3 Разработать программу на (любом) языке высокого уровня, реализующую в интерактивном режиме выбор допустимых информационных мер для заданных условий оценки информационных объектов (ИО) и вычисление количественных значений этих мер.

4 Сравнить чувствительность информационных мер на конкретных примерах.

5 Оформить отчет (индивидуально для каждого студента) исследований и защитить его.

### **Содержание отчета:**

1 Тема, цель и задачи лабораторного занятия, индивидуальное задание на исследование.

2 Результаты изучения понятий «энтропия» и «информационные меры», в том числе и сводная таблица (см. задачу № 2).

3 Программа в электронном варианте (отлаженная).

4 Результаты тестирования полученной программы на заданном наборе исходных данных (индивидуальных заданий).

5 Выводы.

### **Краткая теория. Метрики в информационных системах**

**1 Комбинаторно-вероятностная мера разнообразия заданного множества объектов** [1, 2, 7] (мера неопределенности опытов с  $N$  различными исходами или возможности выбора) Р. Хартли:

$$H = k \ln N, \quad (1.1)$$

где  $k$  – коэффициент пропорциональности (при  $k = 1$  используются натуральные единицы измерения, наты; при  $k = 1/\ln 2$  – двоичные единицы, деды или биты; при  $k = 1/\ln 10$  – десятичные, диты).

*Характеризуется* тем, что:

– математическая строгость определения обеспечена за счет абстрагирования от семантических и прагматических свойств (качества) информации;

– проста и удобна в ряде практических задач связи и управления;

– не учитывает различие между характером имеющихся исходов (почти невероятному исходу придается такое же значение, как и достаточно правдоподобному);

– не отражает случайного характера формирования информационных

массивов (ИМ) в задачах анализа, синтеза и управления информационных системами.

*Предложения по использованию меры:*

– уменьшение меры можно интерпретировать как выигрыш, состоящий в уменьшении разнообразия состояний (степени неожиданного состояния) в управляемом сложном информационном объекте (СИО);

– применение в реальной эргатической системе целесообразно в подсистеме ее централизованной координации (см. Рисунок 1.1) при анализе множества состояний управляемого СИО и множества управляющих воздействий, при определении информационных условий наблюдаемости и управляемости СИО и устойчивости работы информационной системы, а также при синтезе адекватных рациональных мер количества структурной и содержательной информации в информационной системе.

**2 Вероятностно-статистическая усредненная мера неопределенности заданного множества объектов** (статистической «редкости» символов в сообщениях, информационных массивах (ИМ)) Н. Винера и К. Шеннона [1, 7] (селективная энтропия):

$$H = - \sum_i p_{mi} \ln p_{mi}, \quad i = \overline{1, N}, \quad (1.2)$$

где  $p_{mi}$  – вероятность  $i$ -го из возможных  $N$  ИМ  $m_i$ .

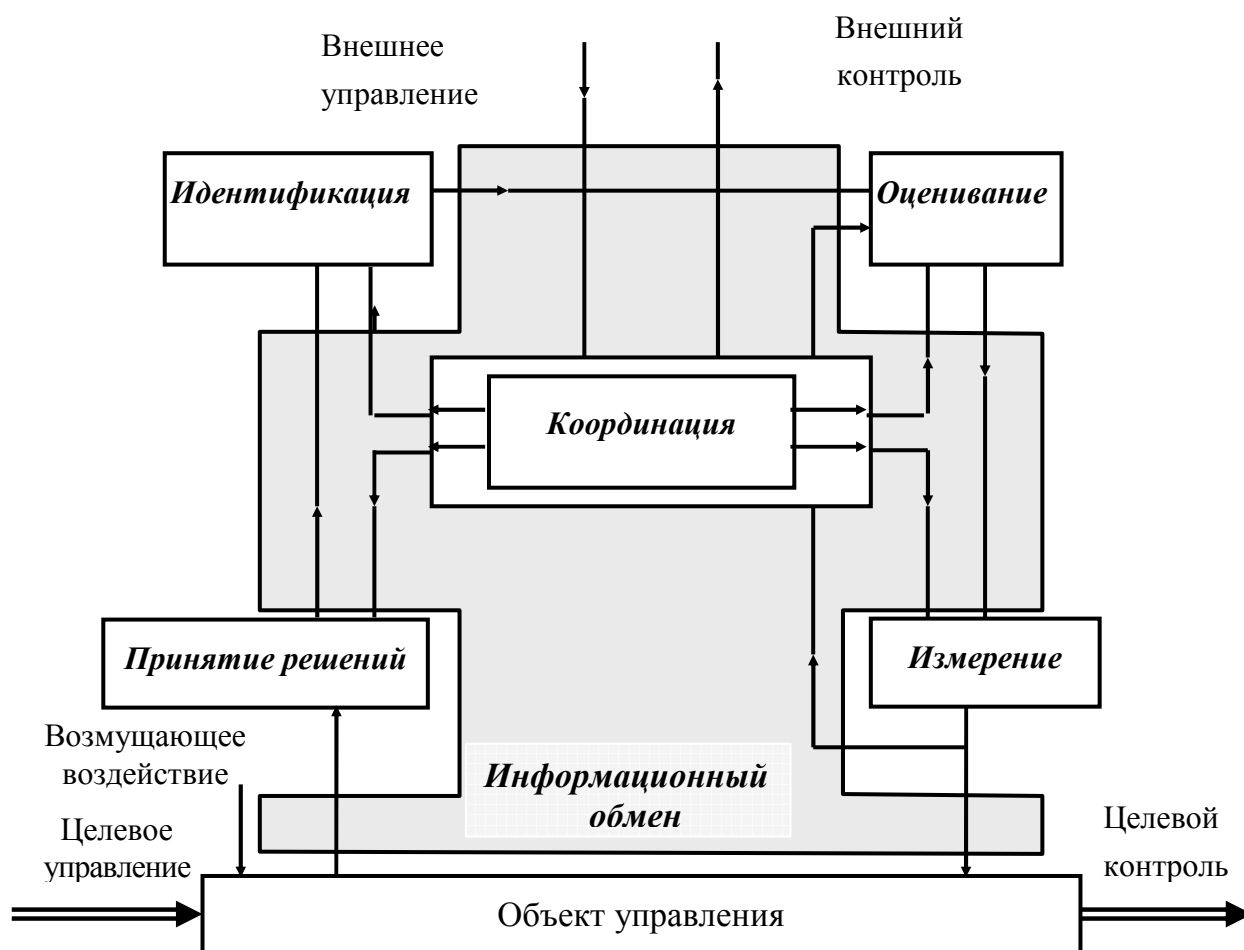


Рисунок 1.1– Координация и обмен информацией в эргатической системе

*Характеризуется* тем, что:

- математическая строгость определения обеспечена за счет абстрагирования от семантических и прагматических свойств (качества) информации;
- удобна в практических задачах передачи информации по физическим каналам связи и некоторых задачах управления;
- учитывает различие между характером имеющихся исходов с помощью использования понятия вероятности (появления символов в ИМ);
- учитывает статистическую структуру ансамбля компонентов сообщений;
- требует получения надежных статистических характеристик ИО;
- не может описать такой важный случай информационного обмена, как поступление дезинформации, так как  $H \geq 0$  и «количество информации»

$$I = H_{apr} - H_{aps} \geq 0;$$

- требует рассмотрения исходного ограниченного множества объектов (сообщений), т.е. относится к закрытым системам;
- не учитывает целей источника и получателя сообщений;
- не учитывает различимость исходов (значений случайной величины);
- не применима для оценки состояния сложной (с большим числом элементов и связей) системы вследствие практической невозможности оценить распределение вероятностей состояний сложной системы.

*Предложения по использованию меры:*

- уменьшение меры можно интерпретировать как особого вида выигрыш, состоящий в уменьшении неопределенности состояния источника сообщений;
- применение в реальной информационных системе целесообразно в подсистеме информационного обмена (Рисунок 1.1) при описании процессов передачи информации в сети информационной системы, т.е. при измерении коммуникационной (связной) информации  $Q_{zc}$ , так как при функционировании информационных каналов связи основную роль играют статистические закономерности вследствие передачи большого количества разнообразных ИМ (сообщений);
- в подсистеме измерения при описании технологических процессов измерений в информационной системе и контроля СИО, особенно, когда имеет место совместная работа измерителя с линией связи и ЭВМ, а также при необходимости согласования измерительного устройства с человеком-оператором информационной системы как биологической системой преобразования информации

При этом мера может характеризовать лишь процесс взаимодействия подсистем информационной системы, но не количество информации, получаемое одной подсистемой от другой в результате такого взаимодействия.

### 3 Алгоритмическая мера сложности индивидуального объекта [2]

(сложности восстановления двоичного слова  $s$  по фиксированной функции или методу программирования  $f: z \rightarrow s$ , который ставит в соответствие программам-описаниям  $z \in Z_f(s)$  различной длины слово  $s$ ) А. Колмогорова:

$$K_f(s) = \begin{cases} \min_{z \in Z_f(s), f(z)=s} |z| & \text{при } Z_f(s) \neq \emptyset, \\ \infty & \text{при } Z_f(s) = \emptyset \end{cases} \quad (1.3)$$

где  $s \in S$ ,  $z \in Z_f(s)$  – двоичные слова;

$f(z)$  – фиксированная оптимальная вычислимая функция восстановления (метод программирования).

Характеризуется тем, что:

- математическая строгость определения обеспечена за счет абстрагирования от семантических и прагматических свойств (качества) информации;
- применима к индивидуальным конструктивным объектам, т.е. позволяет определить количество информации в одном индивидуальном объекте относительно другого;
- не учитывает сложности процесса «развертывания» краткого описания ( $z$ ) объекта в полное описание ( $s$ ), а для многих конструктивных объектов, заданных очень короткими описаниями (по Колмогорову их сложность мала), восстановление осуществляется в результате вычислений нереальной сложности;
- не является эффективно (алгоритмически, регулярно) вычислимой, так как не существует восстанавливающего алгоритма, определяющего по любому описанию конечной длины его сложность.

Предложения по использованию меры:

- уменьшение меры можно интерпретировать как выигрыш, состоящий в уменьшении сложности объекта;
- применение в реальной информационных системах целесообразно в подсистемах оценивания и идентификации (Рисунок 1.1) при описании процессов преобразования информации, при измерении структурной информации  $Q_v$ , а также при синтезе адекватных рациональных мер количества структурной и содержательной информации в АСУ.

**4 Мера неопределенности распределения вероятностей  $R_1(s)$  относительно распределения  $R_2(s)$  [3, 4] (среднее количество информации) С. Кульбака :**

$$I(R_1 | R_2) = \sum_s R_1(s) \ln\{R_1(s) / R_2(s)\}. \quad (1.4)$$

Характеризуется тем, что:

- математическая строгость определения обеспечена за счет абстрагирования от семантических и прагматических свойств (качества) информации;
- проста и понятна;

– значение количества информации по Кульбаку, в частном случае, двух близких гипотез о значении параметра, т.е. в случае, когда в определении количества информации по Кульбаку:

$$R_1(s) = R(s, V), R_2(s) = R(s, V + \delta V), \quad (1.5)$$

где  $V$  – многомерный параметр, представляет собой количество информации по Р. Фишеру.

*Предложения по использованию меры:*

– уменьшение меры можно интерпретировать как выигрыш, состоящий в увеличении определенности распределения вероятностей;

– применение в реальной информационной системе целесообразно в подсистеме измерения (Рисунок 1.1) в частных случаях, например, при проверке измерительных каналов информационной системы.

**5 Мера сложности описания объекта** [4] (его алгебраической структурно-функциональной модели) А. Шилейко и В. Кочнева.

$$I(\Theta) = m_e \ln(n_e) + (\sum_i \ln\{\Lambda_i/\epsilon_{ci}\}) \vee [m_c \ln(2)] + m_a \ln(n_a), \quad (1.6)$$

$$m_e + m_c + m_a = \min, i = 1, n_c,$$

где  $m_e, m_c, m_a$  – число подстрингов (символов), отражающих элементы  $e \in E, c \in C, a \in K$  соответственно в минимальном стринге (упорядоченная последовательность символов, представляющем собой описание объекта (системы)  $\Theta$ ;  $E, C, K$  – множества элементов, информационных связей и структур объекта (системы) соответственно;  $n_e, n_c, n_a$  – количество элементов множеств  $E, C, K$  соответственно;  $\Lambda$  – максимально возможное значение интенсивности информационной связи;  $\epsilon_c$  – погрешность измерения интенсивности связи; 2 – число возможностей при простом наличии-отсутствии информационной связи;  $\vee$  – знак логического «ИЛИ» (здесь означает, что при вычислении  $I(\Theta)$  второе слагаемое может иметь какое-либо одно из двух приведенных выражений в зависимости от характера информационных связей в системе).

*Характеризуется* следующими свойствами:

– аддитивности (так как использован логарифм разнообразия Р. Хартли);  
 – универсальности (поскольку класс рассматриваемых объектов достаточно широк вследствие применения конструктивного подхода А. Колмогорова, а если, в частности, само разнообразие можно определить через вероятности, следуя подходу К. Шеннона, – никаких дополнительных свойств данная мера не приобретает);

– детерминированности (не требует статистических данных);

– конструктивности (не требует ансамбля объектов);

– независимости от потребителя (согласно подходу А. Шилейко и В. Кочнева  $I(\Theta)$  без учета задач переработки информации (ЗПИ) представляет собой пассивный структурно-информационный ресурс информационной системы).



*Предложения по использованию меры:*

– увеличение меры можно интерпретировать как увеличение информационной насыщенности (информационно-структурного ресурса) объекта (системы), т.е. количества и разнообразия взаимодействующих друг с другом неоднородных элементов объекта;

– применение в реальной информационных системе целесообразно в подсистемах оценивания и идентификации (Рисунок 1.1) при описании динамики преобразования информации, в частности, в процессах детерминированных алгоритмических вычислений, а также в подсистеме координации для оценивания информационно-структурного ресурса системы, т.е. для измерения структурной информации  $Q$ , что позволит обеспечить рациональное его использование в функционирующей системе для получения различных технологических эффектов. Оценивание *последних* возможно на основе соответствующих информационных показателей, использующих данную информационную меру в качестве компонента;

$$A = \{a_i\}, i = \overline{1, N} \quad (1.7)$$

для определения меры количества полной структурной информации в системе (пассивной и активной, т.е. информации о жизненном цикле системы) следует учитывать и элементы множества задач (целей), стоящих перед информационной системой. Данные элементы описывают изменения информационной структуры  $\alpha_j(a_i) \in K, j = \overline{1, J}$  информационной системы, совершающиеся для достижения стоящих перед ней целей и задач, непосредственно или в виде последовательности соответствующих задачам  $a_i \in A, i = \overline{1, N}$  структур  $\alpha_k \in K, k = \overline{1, J}$  функционирующей информационной системы. Множество задач формально можно описать с помощью ориентированного графа  $G = [A, B]$  без петель и контуров с множеством вершин  $A = \{a_1, a_2, \dots, a_N\}$ , отображающих заданное множество, и множеством дуг  $B \subset A \times A$ , отображающих частичный порядок. В этом случае, учитывая также, что  $E = \langle S, R \rangle$ , где  $S, R$  – множества средств и ресурсов информационной системы соответственно, получим:

$$I_v(\Theta) = m_s \ln(n_s) + m_r \ln(n_r) + \{(\sum_i \ln\{\Lambda_i/\varepsilon_{ci}\}) \vee [m_c \ln(2)] + [m_a \ln(n_a) + m_b \ln(2)]\} \vee [\sum_k m_{ck} \ln(2)] + m_{\alpha} \ln(n_{\alpha}), \quad (1.8)$$
$$m_s + m_r + m_c + m_{\alpha} + m_a + m_b = \min; i = \overline{1, n_c}; k = \overline{1, J}'$$

где  $m_s, m_r, m_c, m_{\alpha}, m_a, m_b$  – число подстрингов (символов), отражающих элементы  $s, r, c, \alpha, a, b$  соответственно в минимальном стринге, представляющем собой описание информационной системы;  $\vee$  – знак логического «ИЛИ» (применяется при вычислении  $I(\Theta)$  аналогично применению его в мере А. Шилейко и В. Кочнева).

**Пример.** Для информационного узла информационной системы, содержащего  $n = |S| = 4$  однородных информационных преобразователей, использующих

$k = |R| = 3$  видов общесистемных ресурсов и функционирующих совместно в сети с переменной структурой, имеющей  $J = 2$  вариантов (радиальный и полносвязный),

$$m_s = n, m_r = k, m_\alpha = J, m_{c1} = 2(n-1), m_{c2} = n(n-1) \text{ и}$$

$$I_v(\Theta) = n \ln(n) + k \ln(k) + 2(n-1) \ln(2) + n(n-1) \ln(2) + J \ln(J) = 22,86 \text{ нат.}$$

**6 Комбинаторная мера разнообразия общесистемного тезауруса** (запаса знаний) Ю.Шрейдера [ 1, 7 ]:

$$H(m, T) = \ln[T(O_m)] = \ln [\sum_i n_i(O_m)], \quad (1.9)$$

где  $O_m \in O$  – оператор преобразования тезауруса  $T = \langle X, Y, Z \rangle$ , соответствующий единичному ИМ  $m \in M$ ;  $X, Y, Z$  – множества *объектов-троек* <имя-смысл-значение>, предикатов и отношений, событий-высказываний соответственно;  $n_i, i = x, y, z$  – кардинальное число  $i$ -го множества.

*Характеризуется* тем, что:

- математическая строгость определения обеспечена за счет учета семантических и прагматических свойств (качества) информации в виде разнообразия соответствующих подмножеств тезауруса;

- проста и понятна;

- содержит формальную модель (описание) предметной области, что позволяет получателю осмысливать информацию.

*Предложения по использованию меры:*

- изменение меры можно интерпретировать как получение содержательной информации  $Q_z$  или дезинформации;

- применение в реальной информационных системе целесообразно в подсистемах принятия решений и координации (Рисунок 1.1) при описании процесса содержательного (смыслового) анализа и интерпретации ИМ  $m \in M$  как изменения тезауруса  $T$  под влиянием данного ИМ  $m$ , т.е. для измерения информации  $Q_z$ . При этом определение количества  $I_z$  содержательной информации по существу отражает прагматику ИМ (воздействие на тезаурус получателя).

**Пример.** Пусть в информационном узле информационной системы получен текстовый ИМ:  $m_j = \langle \text{Характеристики СИО считаются отработанными, если они определены и приемлемы} \rangle$ . С использованием предикатов:  $y_1$  – быть характеристиками,  $y_2$  – СИО (принадлежать СИО),  $y_3$  – считаются (двуместный предикат),  $y_4$  – определены,  $y_5$  – приемлемы,  $y_6$  – отработанные, можно представить соответствующую смысловую запись в виде:

$$S_j = \langle ( \mathcal{Y}x )(y_1x) ; y_2( \mathcal{D}x )(y_1x) ; ( y_4( \mathcal{D}x )y_1x ) = z_1 ; \\ ( y_5( \mathcal{D}x )y_1x ) = z_2 ; z_1 \wedge z_2 = z_3 ; y_3 [ ( \Phi x )z_3 \cdot x, ( \Phi x )y_6x ] \rangle,$$

где  $\mathcal{Y}, \mathcal{D}, \Phi$  – кванторы рождения (введения в рассмотрение), дескрипции и множественной дескрипции соответственно;  $\wedge$  – знак логического «И».

Согласно определяется  $O_{mj}$ , по которому в  $T$  вводятся предикаты  $y_6$  (если он там отсутствовал ранее) и  $z_3^*$  (участвует в событии  $z_3$ ), отношения включения

$y_6 \leftarrow y_4, y_6 \leftarrow y_5$ , отношение применимости  $y_6 \rightarrow (y_1, y_2)$ , а также события-высказывания  $z_1, z_2, z_3$ .

Тогда  $I_z(m_j, T) = \ln [(n_{yj} + n_{zj})/n_y] = \ln[(10+3)/5] = +0,96$  нат.

**7 Вероятностная мера целесообразности управления** А. Харкевича [1, 3, 7]:

$$I = \ln p_1/p_0, \quad (1.10)$$

где  $p_0, p_1$  – вероятности достижения цели управления до получения и после получения информации соответственно.

Характеризуется тем, что:

- математическая строгость определения обеспечена за счет абстрагирования от природы информации («на основании некоторой информации система принимает решение, изменяющее вероятность достижения цели»). Кроме того, физическая природа сигналов, логическая структура сообщений и их длина полностью игнорируются;

- проста и понятна;

- учитывает семантические и прагматические свойства (качество) содержательной информации;

- может описать случай поступления дезинформации, так как возможно  $I \leq 0$ .

*Предложения по использованию меры:*

- увеличение меры можно рассматривать как выигрыш, состоящий в увеличении вероятности достижения цели управления;

- применение в реальной информационных системе целесообразно в подсистемах принятия решений и координации (Рисунок 1.1) в тех случаях, когда эффективность применения информационной системы не может быть описана только в виде материальных эффектов.

**8 Мера неопределенности результата принятия решения** (наших знаний о возможных результатах принятых нами решений) Н.Н. Моисеева [1] :

$$\Delta = \left| \frac{J^+ - J^-}{J^+ + J^-} \right|, \quad (1.11)$$

$$J^+ = \max_u J(u^*, \xi^+), \quad J^- = \max_u J(u^*, \xi^-)$$

$$\xi^- \leq \xi(t) \leq \xi^+ \quad \forall t \in [t_0, T],$$

где  $x(t)$  – управляемый процесс, о котором отсутствует полная информация, а известны только нижняя ( $\xi^-$ ) и верхняя ( $\xi^+$ ) оценки его значений;  $J[u(t), \xi(t)]$  – целевая максимизируемая функция (функционал) управления;  $u^*(t)$  – оптимальное управление.

Характеризуется тем, что:

- математическая строгость определения обеспечена за счет отвлечения от природы информации («на основании некоторой информации система принимает решение, уменьшающее степень неопределенности достижения цели»). Кроме того, физическая природа сигналов, логическая структура сообщений и их длина полностью игнорируются;

- проста и понятна: является оценкой количества (достаточности) и *качества* (прагматических свойств) содержательной информации о процессе  $\xi(t)$ ;
- учитывает значения показателя эффективности управления, т.е. позволяет судить о ценности имеющейся в системе информации;
- не может описать важный случай поступления дезинформации;
- близка, по существу, к мере Н. Винера и К. Шеннона, которая является именно мерой неопределенности той информации, которая содержится в передаваемом сообщении;
- в частном случае, если  $\xi(t)$  – случайный процесс с математическим ожиданием  $\bar{\xi}(t)$  и не известна его конкретная реализация, то в качестве варианта данной меры можно принять дисперсию случайной величины  $J[u^*(\xi), \xi]$ , т.е.

$$\Delta = D\{ J[u^*(\xi), \xi] \}, \quad (1.12)$$

$$u^* = \arg \max_u J[u(t), \xi(t)].$$

*Предложения по использованию меры:*

- уменьшение меры можно рассматривать как выигрыш, состоящий в повышении ценности информации об объекте (процессе) управления;
- применение в реальной информационных системах возможно в подсистемах принятия решений и координации (Рисунок 1.1) для интегральной оценки количества (достаточности) и качества (содержательности) осведомляющей информации  $Q_o$ , получаемой от СИО, и информации  $Q_o'$  (целеуказаний), получаемой из надсистемы перед началом реализации процедуры управления лицом, принимающим решения (ЛПР). То есть  $\Delta$  выступает как мера информированности ЛПР, определяющая конкретные выражения критерия (например, максиминного критерия) и вид оператора управления.

**9 Негэнтропия** [1, 6](отрицательная энтропия):

$$\bar{H} = H^* - H, \quad (1.13)$$

где  $H$ ,  $H^*$  – термодинамические энтропии системы, находящейся в начальном и конечном состояниях соответственно.

*Характеризуется* тем, что:

- является чисто термодинамической величиной, которую можно интерпретировать как меру того, насколько далеко определенная физическая система находится от доступного ей при данной энергии состояния ( $H^*$ ) полного статистического равновесия;
- ничего общего не имеет с определенностью, упорядоченностью, организованностью и др., поскольку эталон порядка или беспорядка, определенности или неопределенности в природе не существует, а выбирается условно.

*Предложения по использованию меры:*

- применение в реальной информационных системах нецелесообразно.

**10 Теоретико-множественная мера неопределенности сведения об**

элементе (точке) множества [ 1 ] :

$$H\{ (p)\delta(x) \} = \text{mes}\{ \delta(x) \}, \quad (1.14)$$

где  $(p)\delta(x)$  – сведение об элементе  $x$  множества  $X$  по А.В. Чечкину;

$p$  – вероятность  $x \in \delta(x)$ ;

$X$  – множество возможных значений  $x$ :  $x \in X, \forall \delta(x) \subset C$ ;

$\text{mes}\{ \delta(x) \}$  – мера А. Лебега множества  $\delta(x)$ .

Характеризуется тем, что:

- математическая строгость определения обеспечена за счет использования формального аппарата теории множеств;
- аналитически проста и понятна;

$$0 \leq H \leq H_{\max} = H(C);$$

- учитывает семантические свойства содержательной информации.

*Предложения по использованию меры:*

- изменение (уменьшение) меры можно интерпретировать как выигрыш, состоящий в уменьшении неопределенности знания о значении элемента на множестве его возможных значений;

– применение в реальной информационных системе целесообразно в подсистемах оценивания и идентификации (Рисунок 1.1) при описании технологического процесса защищенных функционально-технологических преобразованиях (ФТП) численных ИМ как преобразований совокупности элементарных сведений  $\delta_i(x_i), i = \overline{1, I}$ , являющихся аргументами алгоритмического, аналитического или логического преобразования совокупности  $\delta_i(x_i), i = \overline{1, I}$ , в элементарное сведение  $\delta(x), p(x) = 1$ ;

- целесообразно использование имеющей место «существенной неопределенности» сведения  $\delta(x)$ -результата ФТП как нижней грани значений неопределенности сведения  $\delta(x)$  на множестве сведений  $\delta_i(x_i) \subset X_i$  аргументов ФТП при

$$H(\delta_i(x_i)) = C_i; C_i = \text{const}, i = \overline{1, I}; ( \delta(x) ) = \inf\{ H(\delta_i(x_i)) \}, \quad (1.15)$$

$$\delta_i(x_i) \subset X_i, H(\delta_i(x_i)) = C_i = \text{const}.$$

При этом существенная неопределенность  $H'(\delta(x))$  – инъективное отображение неопределенностей сведений-аргументов ФТП.

#### **Индивидуальные задания**

Разработать и исследовать программы по оценке количества информации:

- для всех: меры Хартли, Шеннона–Винера, Харкевича;
- отдельно для каждого студента: для языков, представленных латиницей (не менее 3).

Справочный материал

Частотность букв русского языка						Частотность букв английского языка					
Символ	Частота	Символ	Частота	Символ	Частота	Символ	Частота	Символ	Частота	Символ	Частота
пробел	0.175	Л	0.035	Б	0.014	А	0.081	Ј	0.002	Т	0.10.5
О	0.090	К	0.028	Г	0.012	В	0.014	К	0.004	U	0.024
Е	0.072	М	0.026	Ч	0.012	С	0.027	L	0.034	V	0.009
Ё	0.072	Д	0.025	Й	0.010	Д	0.039	М	0.025	W	0.015
А	0.062	П	0.023	Х	0.009	Е	0.130	Н	0.072	X	0.002
И	0.062	У	0.021	Ж	0.007	F	0.029	О	0.079	Y	0.019
Т	0.053	Я	0.018	Ю	0.006	G	0.020	P	0.020	Z	0.001
Н	0.053	Ы	0.016	Ш	0.006	Н	0.052	R	0.069		
С	0.045	З	0.016	Ц	0.004	I	0.065	S	0.061		
Р	0.040	Ь	0.014	Щ	0.003						
В	0.038	Ъ	0.014	Э	0.003						
				Ф	0.002						

Наиболее часто встречаются следующие буквы латинского алфавита в художественной литературе (в %):

Английский		Французский		Немецкий		Испанский		Итальянский	
Е	12.86	Е	17.76	Е	19.18	Е	14.15	І	12.04
Т	9.72	S	8.23	N	10.20	A	12.90	E	11.63
A	7.96	A	7.68	I	8.21	O	8.84	A	11.12
I	7.77	N	7.61	S	7.07	S	7.64	O	8.92
N	7.51	T	7.30	R	7.01	I	7.01	N	7.68
R	7.03	I	7.23	T	5.68	R	6.95	T	7.07

### Пример результатов выполнения разработанной программы

Мера	Хартли	Шенон	Координации	Кинформационного обмена	Измерений	Оценивания	Идентификации	Принятия реше
Комби	Колмогоров		+	-	-	-	-	-
Вероя	Кульбак		-	+	-	-	-	-
Алгори	Шилейко, Кочнев		-	-	-	-	-	-
Неопри	Шрейдер		-	-	-	-	-	-
Сложн	Харкевич		+	-	-	-	-	-
	Моисеев		+	-	-	+	+	-
	Негантропия		-	-	-	-	-	-
Комбинаторная разнообразия			-	-	-	-	-	-
Вероятностная			+	-	-	-	-	+
Неопределенности результата			+	-	-	+	+	+
Негантропия			-	-	-	-	-	-

**Вероятностно-статистическая усредненная мера неопределенности заданного множества объек...**

Выберите язык:  
 Русский  
 Английский  
 Другой

Введите сообщение:  
 Срочно грузите апельсины бочк

**H = - СУММА( pmi \* ln ( pmi )**

Количество Информации по Шеннону = 3,9898764311866

## Задание к лабораторной работе № 2 «Изучение и исследование качественных характеристик сложных информационных систем»

**Цель занятия:** Получить навыки исследования поведения информационных объектов и систем на основе динамической теории информации.

### **Задачи на работу:**

1 Провести анализ основных существующих способов исследования поведения сложных систем.

2 Разработать систему дифференциальных уравнений, описывающих поведение эргатической системы на основе качественного подхода к информации.

3 Ввести разработанные дифференциальные уравнения в программу *WInSet* и исследовать СДУ при различных начальных условиях.

4 Сравнить полученные фазовые портреты и дать характеристику имеющимся в системе аттракторам.

5 Оформить отчет (индивидуально для каждого студента) исследований и защитить его.

### **Содержание отчета:**

1 Тема, цель, исходные данные и задачи лабораторного занятия.

2 Модель функционирования ИС, с обоснованием значений всех элементов.

3 Результаты тестирования полученной модели на заданном наборе исходных данных (индивидуальных заданий).

4 Выводы.

## Краткая справка об инструменте *WInSet* для качественного исследования сложных информационных систем

*WInSet* расшифровывается как «*Invariant Sets*» for Windows.

Программа *WInSet* предназначена для исследования и демонстрации инвариантных множеств динамических систем. Программа работает с набором «систем» – объектов, для которых выполняются графические построения. Системы – это:

- двумерные отображения,
- схемы построения фракталов на основе отображений,
- системы дифференциальных уравнений.

Для каждой системы определен свой набор построений из трех возможных:

- 1 Отображение (*Map*).
- 2 Фазовые траектории/решение (*Traject*).
- 3 Фрактал (*Fractal*).

Для двумерных отображений Пуанкаре, заданных явными формулами, (категория систем «*Map*») определено построение *Map*.

Для фрактальных схем (категория систем «*Fractal*») определено построение *Fractal* и, если возможно, построение *Map* – вывод на экран точек исходного двумерного отображения.

Для систем дифференциальных уравнений (категория систем «*Diff Eq*») всегда доступно построение *Traject*.

### **Меню *WinSet***

*File* (Файл)

*Open slide...* (Открыть слайд) – открыть файл слайда и выполнить сохраненное в этом файле построение. Вызывается стандартный диалог открытия файла.

*Save as slide...* (Сохранить слайд...) – сохранить текущее построение в файле слайда *WinSet*. Вызывается стандартный диалог сохранения файла.

*Save picture...* (Сохранить в графическом формате...) – сохранить содержимое рабочей области в файле формата BMP. Вызывается диалог сохранения файла, позволяющий просматривать графические файлы.

*Print...* (Печать...) – вызов стандартного диалога печати на принтере. Изображение печатается в натуральную величину. Если это невозможно (не помещается на бумаге при текущих установках принтера), то изображение масштабируется так, чтобы оно поместилось на листе выбранного формата.

*Exit* (Выход) – выход из программы.

*Edit* (Правка).

*Undo draw* (Отмена построения) – отмена последнего запуска построения, т.е. возврат на один этап назад.

*Initial point...* (Начальная точка...) – вызов диалога ввода начальной точки.

*System parameters...* (Параметры системы...) – вызов диалогового окна просмотра записи системы и ввода значений параметров, если такие имеются.

*Color* (Цвет) – этот пункт показывает подменю для изменения цветов построения: текущего цвета, цвета фона и цвета осей координат. Цвета фона и осей изменяются немедленно, а текущий цвет построения действует со следующего запуска.

*Refresh* (Перерисовать) – перерисовка области построения.

*Refresh axes* (Перерисовать оси) – перерисовать оси поверх имеющегося построения.

*Clear* (Очистить) – очистить рабочую область и «забыть» все выполненные шаги построения.

*View* (Вид).

*Toolbar* (Панель инструментов) – показывать панель инструментов под строкой меню.

*Point coordinates* (Координаты точки) – показывать панель координат текущей точки.

*Color cycling* (Цикл цвета) – этот пункт меню активизирует/останавливает циклическое изменение цветов при построении фракталов. Цикл цвета работает только в видеорежиме с цветовым разрешением 256 цветов, поскольку он основан на циклическом изменении палитры. Другой способ запуска/останова – клавишей «пробел».



*System* (Система).

*Choose...* (Выбор систем...) – вызов диалогового окна выбора системы.

*New user system...* (Добавить новую...) – вызов диалога ввода новой пользовательской системы.

*User system properties...* (Свойства систем пользователя...) – вызов диалогового окна «Свойства систем пользователя».

*Draw* (Запуск) – запуск построения (из текущей точки).

*Options* (Установки).

*Counting...* (Счет...) – вызов диалогового окна для редактирования установок, используемых при счете.

*Graph...* (Построение...) – вызов диалогового окна для изменения типа построения и граничных значений по осям координат.

*Draw rev. time* (В обратную сторону по времени) – действует для систем дифференциальных уравнений, позволяет изменить шаг по времени на противоположный без вызова окна изменения установок счета.

*Draw separatrixes* (Построение сепаратрис) – этот пункт меню действует только в режиме построения отображений Пуанкаре. Когда он отмечен, появляются дополнительные параметры счета, связанные с построением сепаратрис.

*Save full data in slides* (Сохранение полных данных в слайде) – этот пункт меню активен только для систем дифференциальных уравнений в режиме построения отображения Пуанкаре. Он включает/выключает сохранение в слайде данных о всех вычисленных точках.

*Language* (Язык) – выбор языка для меню и большинства текстовых сообщений. Некоторые сообщения будут всегда выдаваться по-английски, например сообщения компилятора выражений, вызываемого при вводе систем пользователя.

*Help* (Справка).

*Contents* (Содержание) – вызов этого файла справки.

*About...* (О программе...) – показ диалога с данными о программе *WInSet*.

*Панель инструментов* – это ряд кнопок, дублирующих некоторые пункты меню. Используйте панель инструментов для оперативного доступа к пунктам меню. Если на некоторое время задержать курсор мыши над одной из кнопок, то рядом с курсором появится всплывающая подсказка, поясняющая назначение кнопки. Для того чтобы скрыть панель инструментов, снимите отметку с пункта меню *View* (Вид) | *Toolbar* (Панель инструментов).

*Рабочая область*

Основное пространство окна приложения занято рабочей областью, внутри которой выполняются построения.

*Панель координат точки* находится справа от рабочей области. В нижней части панели отображаются значения координат точки на плоскости, соответствующей текущему положению курсора мыши. В верхней части «неактивным» цветом отображаются координаты последней вычисленной точки (либо начальной точки, заданной по умолчанию). Именно эти значения будут использованы в качестве начальных условий для построения, если оно запущено без указания начальной точки. Для того чтобы скрыть панель координат точки,

снимите отметку с пункта меню *View* (Вид) | *Point coordinates* (Координаты точки).

### *Строка состояния*

В строке состояния отображается информация о работе программы. Когда курсор мыши над областью построения принимает форму песочных часов, программа выполняет какую-то длительную операцию. Пока эта операция не закончится, в строке состояния находится текст пояснения.

### *Построение отображений*

На плоскости двух переменных курсор мыши имеет форму перекрестия. Переместите его в интересующую Вас область плоскости и нажмите левую кнопку мыши. В качестве начальной точки будет выбрана точка, соответствующая положению курсора мыши, и из нее будет выполнено построение отображения Пуанкаре. Число шагов отображения, выполняемых за один запуск, задается в окне параметров счета. Для того чтобы следующее построение продолжить из последней вычисленной точки (ее координаты выводятся в верхней части панели координат), выберите пункт меню *Draw* (Запуск) или нажмите *Enter*. Для того чтобы точнее задать координаты начальной точки, выберите пункт меню *Edit* (Правка) | *Initial point ...* (Начальная точка ...) и введите значения для координат начальной точки в появившемся окне ввода начальной точки. Для построения сепаратрис отображения Пуанкаре выберите пункт меню *Options* (Установки) | *Draw separatrices* (Построение сепаратрис). Теперь при запуске построения из какой-либо точки будет предполагаться, что начальная точка задана в окрестности гиперболической неподвижной точки отображения, и из начальной точки начнется построение сепаратрисы. Среди параметров счета в режиме построения сепаратрис появляются дополнительные параметры:

*Number of interval divisions* (Число элементов разбиения отрезка).

*Number of Interval iterations* (Число итераций отрезка).

*Order of subharmonic* (Номер субгармоники).

Для того чтобы прервать выполнение построения, нажмите клавишу *Esc*.

### *Построение фазовых траекторий и решений систем дифференциальных уравнений*

На плоскости двух переменных курсор мыши имеет форму перекрестия. Переместите его в интересующую Вас область плоскости и нажмите левую кнопку мыши. В качестве начальной точки будет выбрана точка, соответствующая положению курсора мыши, и из нее будет выполнено численное интегрирование системы диф. уравнений. Интервал по времени, на который выполняется интегрирование за один запуск, задается в окне параметров счета. Там же задаются и метод интегрирования, и начальный шаг, и максимальная погрешность метода (для простейших методов она не используется) Для того чтобы следующее построение продолжить из последней вычисленной точки (ее координаты выводятся в верхней части панели координат), выберите пункт меню *Draw* (Запуск) или нажмите *Enter*. Для того чтобы точнее задать координаты начальной точки, выберите пункт меню *Edit* (Правка) | *Initial point ...* (Началь-

ная точка ...) и введите значения для координат начальной точки в появившемся окне ввода начальной точки.

Когда рабочая область представляет собой проекцию трехмерного пространства, курсор мыши имеет форму стрелки, и начальную точку можно задать только с помощью диалогового окна ввода начальной точки.

Для того чтобы прервать выполнение построения, нажмите клавишу *Esc*.

### **Построение фракталов**

Для запуска построения фрактала выберите пункт меню *Draw* (Запуск) или нажмите *Enter*. (Чтобы прервать построение, нажмите клавишу *Esc*.)

Далее входим в пункт меню *Options* (Установки) | *Language...* (Язык...) главного меню программы. Здесь задается язык меню и сообщений программы. Отметьте пункт *Russian* (Русский), если хотите работать с русским вариантом меню и сообщений.

Рассмотрим основные приемы работы с программой, для этого выберите пункт меню *System* (Система) | *Choose...* (Выбор систем...). Появится Окно выбора системы. Выберите в категории систем «*Map*» отображение Чирикова («*Chirikov*») и нажмите кнопку «ОК».

После того как система – объект исследования выбрана, в рабочей области главного окна программы будут нарисованы оси координат, и появится окно ввода параметров системы. Нажмите в нем на кнопку «ОК», оставив тем самым значения параметров, заданные по умолчанию.

Можно приступить к построению отображения. Курсор мыши над рабочей областью имеет форму перекрестия. Если щелкнуть левой кнопкой мыши в какой-либо точке рабочей области, *эта точка будет принята за начальную*: из нее начнет строиться отображение. Будет построено несколько точек отображения. Для того чтобы продолжить построение из последней точки, нажмите клавишу *Enter*. Число точек, выводимое за один запуск построения, задается вместе с другими параметрами счета с помощью окна ввода установок счета.

Для того чтобы изменить цвет фона, осей или выбрать другой цвет для следующих элементов построения, воспользуйтесь пунктом меню *Edit* (Правка) | *Color* (Цвет). Для каждого из перечисленных цветов в нем есть соответствующий подпункт, который вызывает стандартный диалог выбора цвета.

Для того чтобы изменить граничные значения по осям координат, выберите пункт меню *Options* (Установки) | *Graph...* (Построение...). Появится окно ввода установок построения, в котором граничные значения по осям координат можно ввести с клавиатуры. Кроме этого, какой-либо участок на плоскости можно с помощью мыши выделить и увеличить, «растянув» его на все окно. Для этого нажмите левую кнопку мыши в верхнем левом углу предполагаемого прямоугольника и, удерживая ее в нажатом состоянии, двигайте курсор мыши вправо и вниз, пока пунктирный прямоугольник не охватит интересующую Вас область. После этого отпустите кнопку мыши, и выделенный Вами участок построения будет увеличен на все окно.

Слайды *WinSet* – это текстовые файлы, в которых содержится данные, необходимые для воспроизведения построения:

- название системы,

- параметры системы,
- характеристики окна построения: данные об осях координат, цвете фона, осей и построения,
- установки для проведения вычислений ( для каждого типа систем),
- набор начальных условий.

Загрузка слайда – это повторение программой ранее выполненных шагов построения. После загрузки слайда можно продолжить построение, отменить его часть, изменить характеристики окна построения и т.д. (см. Изменение установок построения, Изменение установок счета), а главное – можно рассмотреть детали построения в увеличенном виде, выделив какой-либо фрагмент (см. Выделение фрагментов построения).

Сохранение построения в виде слайда выгодно отличается от сохранения графического образа в формате BMP, поскольку в BMP-файле сохраняется только статическая картинка, находившаяся в момент сохранения на экране.

Для отображений, порожденных системами дифференциальных уравнений, загрузка слайда может длиться довольно долго из-за большого объема вычислений. Чтобы уменьшить время загрузки таких слайдов, предусмотрена возможность сохранения в слайде полных данных обо всех вычисленных точках. В результате файл слайда получается довольно большим, но его загрузка происходит в десятки раз быстрее. Для того чтобы включить в файл слайда полные данные о построении, отметьте пункт меню *Options (Установки) | Save full data in slides (Сохранять полные данные в слайде)*.

*WinSet* предоставляет пользователю возможность вводить свои системы. Пользовательские системы могут быть только трех типов:

*Map* – Двумерное отображение Пуанкаре, заданное рекуррентной формулой.

*Diff. eq.* – Система дифференциальных уравнений (размерность  $\leq 10$ ).

*Diff. eq. with 3/2 deg. of freedom* – Система «с 3/2 степенями свободы».

Для систем первого типа строится только отображение на плоскости переменных, для систем второго типа – только траектории, для третьего типа – траектории и отображение Пуанкаре.

Для того чтобы добавить собственную систему, выберите пункт меню *System (Система) | New user system... (Добавить новую...)*.

### ***Диалог редактирования свойств пользовательской системы***

*Страница General (Общее)*.

*System name (Название)*.

Редактор для ввода названия. Название системы не должно совпадать с названиями уже имеющихся систем. Редактор активен только при вводе новой системы.

*System type (Тип системы)* Выпадающий список для указания типа системы. Активен только при вводе новой системы.

*System dimension (Размерность)*

Поле ввода размерности системы диф. уравнений. Для отображения и системы с 3/2 степенями свободы всегда 2.

*Default draw type (Построение по умолчанию)*

Для системы с 3/2 степенями свободы указывается вид построения, который будет установлен автоматически, когда Вы начнете работать с системой.

### *Category (Категория)*

Редактор с выпадающим списком для указания, к какой категории отнести систему. Либо выберите уже имеющуюся категорию, либо введите название новой категории. По умолчанию все пользовательские системы заносятся в категорию «*User systems*». Используйте различные категории своих систем для того чтобы быстрее добраться до нужной системы во время выбора объекта исследования.

### *Запись уравнений*

Левые части уравнений фиксированы. Они определяются типом системы. Вам остается ввести выражения для правых частей. Кроме четырех арифметических действий и возведения в степень (обозначаемого знаком  $^$ ), в выражениях могут встречаться вызовы некоторых элементарных функций.

Для обозначения переменных используются символы  $x_1$ ,  $x_2$  и т.д. В правых частях систем могут встречаться параметры, которые обозначаются  $p_1$ ,  $p_2$  и т.д. Максимально возможное число параметров – 20. Для включения в правые части стандартных констант и элементарных функций пользуйтесь их символическими именами. Никакие другие символьные имена, кроме перечисленных, в записи правых частей не допускаются. Строчные и прописные буквы не различаются.

### *Compile text (Компилировать текст)*

Кнопка для вызова компилятора формул. В процессе компиляции могут выявиться ошибки в записи выражений. Сообщения об ошибках выдаются в отдельном диалоговом окне, которое появляется поверх диалога ввода и редактирования систем пользователя. Систему с неправильной записью правых частей сохранить нельзя, поскольку при нажатии на кнопку «ОК» компиляция формул запускается автоматически.

### *Frequency of perturbation (Частота возмущения)*

Эти элементы высвечиваются только для систем с полутора степенями свободы и предназначены для того чтобы указать либо величину частоты возмущения, либо параметр, который ее определяет. Данные элементы управления становятся активными только после компиляции правых частей системы.

### *Список имен констант и встроенных функций:*

(строчные и прописные буквы не различаются)

$\pi$  – число  $\pi$ ;

$e$  – основание натурального логарифма;

<i>Тригонометрические функции</i>	<i>Гиперболическая тригонометрия</i>	<i>Другие функции</i>
sin – синус	Sinh – синус	abs – модуль числа
cos – косинус	cosh – косинус	int – целая часть числа
tan – тангенс	tanh – тангенс	sqrt – корень квадратный
asin – арксинус	asinh – арксинус	exp – экспонента
acos – арккосинус	acosh – арккосинус	log – натуральный логарифм
atan – арктангенс	atanh – арктангенс	log10 – десятичный логарифм

### **Индивидуальные задания на проведение исследований**

Построить фазовые портреты динамических информационных систем и указать типы аттракторов для заданных областей параметров и начальных условий.

Номер варианта	Дифференциальные уравнения модели	Область изменения параметров	Область начальных условий
1	$x_1' = p_1 \cdot (x_2 - p_4 \cdot x_1 - (p_3 - p_4) \cdot [ x_1 + 1  -  x_1 - 1 ]);$ $x_2' = x_1 - x_2 + x_3; x_3' = -p_2 \cdot x_2$	$p_1 = 6-19; p_2 = 5-15;$ $p_3 = -0.1-2.5;$ $p_4 = 0.2-3$	$x_1(0), x_2(0), x_3(0) \in (-2...12)$
2	$x_1' = -(x_2 + x_3);$ $x_2' = x_1 + 0.2 \cdot x_2;$ $x_3' = 0.2 + x_3 \cdot (x_1 - p_1),$	$p_1 = 1-15.$	
3	$x' = y$ $y' = -x - x^3 + (p_1 + p_2 \cdot x^2 + p_3 \cdot x \cdot \sin(p_5 \cdot t)) \cdot y + p_4 \cdot \sin(p_5 \cdot t)$	$p_1 = 0.04 - 7.2;$ $p_2 = -0.005 - 1;$ $p_3 = 0.01 - 2.8;$ $p_4 = 1-3; p_5 = 2-5$	
4	$x' = y$ $y' = -p_1 \cdot x - p_2 \cdot x^3 + p_3 \cdot \sin(p_5 \cdot t) + p_4 \cdot \sin(p_6 \cdot t)$	$p_1 = 0.5-1.5; p_2 = 1-3;$ $p_3 = 0.01-1;$ $p_4 = -0.1-1;$ $p_5 = 0-0.14; p_6 = 1-5.$	$x_1(0), x_2(0), x_3(0) \in (-7...22)$
5	$x' = y$ $y' = -p_1 \cdot x - p_2 \cdot x^3 - p_3 \cdot x^5 + p_4 \cdot \sin(p_5 \cdot t)$	$p_1 = 0.1-1; p_2 = -3-1;$ $p_3 = 0.1-0.8;$ $p_4 = 0.01-0.1;$ $p_5 = 0.1-0.5.$	
6	$x_1' = -p_1 \cdot (x_1 - x_2)$ $x_2' = -x_1 \cdot x_3 - x_2 + p_2 \cdot x_1$ $x_3' = -p_3 \cdot x_3 + x_2 \cdot x_1$	$p_1 = 10; p_2 = 28;$ $p_3 = 2.6.$	
7	$x_1' = p_1 \cdot (x_2 - p_4 \cdot x_1 - (p_3 - p_4) \cdot [ x_1 + 1  -  x_1 - 1 ]);$ $x_2' = x_1 - x_2 + x_3; x_3' = -p_2 \cdot x_2.$	$p_1 = 6-19; p_2 = 5-15;$ $p_3 = -0.1-2.5;$ $p_4 = 0.2-3$	$x_1(0), x_2(0), x_3(0) \in (-2...12)$
8	$x_1' = -(x_2 + x_3);$ $x_2' = x_1 + 0.2 \cdot x_2;$ $x_3' = 0.2 + x_3 \cdot (x_1 - p_1),$	$p_1 = 1-15.$	
9	$x' = y$ $y' = -x - x^3 + (p_1 + p_2 \cdot x^2 + p_3 \cdot x \cdot \sin(p_5 \cdot t)) \cdot y + p_4 \cdot \sin(p_5 \cdot t)$	$p_1 = 0.04-7.2;$ $p_2 = -0.005-1;$ $p_3 = 0.01-2.8;$ $p_4 = 1-3; p_5 = 2-5$	
10	$x' = y$ $y' = -p_1 \cdot x - p_2 \cdot x^3 + p_3 \cdot \sin(p_5 \cdot t) + p_4 \cdot \sin(p_6 \cdot t)$	$p_1 = 0.5-1.5; p_2 = 1-3;$ $p_3 = 0.01-1; p_4 = -0.1-1;$ $p_5 = 0-0.14; p_6 = 1-5.$	$x_1(0), x_2(0), x_3(0) \in (-7...22)$

### **Задание к лабораторной работе № 3 «Исследование количественных характеристик устойчивости динамических систем»**

**Цель занятия:** Получить навыки исследования информационных процессов и систем на основе классической теории устойчивости Ляпунова.

#### **Задачи на занятие**

- 1 Провести анализ основных существующих подходов к исследованию устойчивости, применяемых в информационных системах.
- 2 Разработать программу на (любом) языке высокого уровня, реализующую в интерактивном режиме оценку устойчивости информационных систем на основе показателей Ляпунова.
- 3 Сравнить устойчивость предлагаемого набора динамических систем.
- 4 Оформить отчет (индивидуально для каждого студента) исследований и защитить его.

#### **Содержание отчета:**

- 1 Тема, цель и задачи лабораторного занятия.
- 2 Результаты изучения понятий «устойчивость» и «показатели и критерии устойчивости».
- 3 Программа в электронном варианте (отлаженная).
- 4 Результаты тестирования полученной программы на заданном наборе исходных данных (индивидуальных заданий).
- 5 Выводы.

#### **Краткие теоретические сведения. Показатели Ляпунова**

Поведение траекторий в окрестности состояния равновесия для уравнения:

$$\dot{x} = f(x), x \in R^n \quad (3.1)$$

описываем с помощью собственных чисел матрицы линеаризации. Для описания поведения траекторий в окрестности замкнутой траектории служат ее мультипликаторы. Для исследования поведения траекторий в окрестности произвольной траектории удобно использовать показатели Ляпунова, которые представляют собой аналоги мультипликаторов периодических траекторий.

#### **Основные свойства показателей Ляпунова [5]**

Пусть  $\Gamma(x_0)$  – траектория уравнения (3.1), соответствующая решению  $p(t) = \varphi_{x_0}(t)$ . Асимптотическое поведение траекторий, лежащих вблизи  $\gamma(x_0)$ , определяется поведением фундаментальной матрицы  $Ux_0(t)$  уравнения в вариациях для решения  $p(t)$ . Показатели Ляпунова являются инструментом, служащим для описания асимптотического поведения матрицы  $Ux_0(t)$ .

Рассмотрим  $n$  линейно независимых векторов  $b_1, b_2, \dots, b_n$ , выходящих из точки  $X_0$ . Выберем из них  $k$  векторов ( $1 \leq k \leq n$ ), которые обозначим как  $e_1 = b_{i_1}, e_2 = b_{i_2}, \dots, e_k = b_{i_k}$ . Векторы  $e_1, \dots, e_k$  задают в фазовом пространстве  $R^n$  некоторый  $k$ -мерный параллелепипед  $P^k$  и определяют натянутое на них  $k$ -мерное подпространство  $e^{(k)}$ .

Фазовый поток перемещает точку  $X_0$  за время  $t$  в точку  $\varphi_{x_0}(t)$ ; при этом векторы  $e_1, e_2, \dots, e_k$  отображаются на векторы  $U_{x_0}(t)e_1, \dots, U_{x_0}(t)e_k$ , которые в свою очередь образуют параллелепипед  $\varphi^t(P^k)$ . Нас будет интересовать изменение объема параллелепипеда  $P^k$ , точнее, отношение объемов  $\varphi^t(P^k)$  и  $P^k$ . Обозначим объем параллелепипеда  $P^k$  через  $\|e_1 \wedge e_2 \wedge \dots \wedge e_k\|$ . Этот объем вычисляется следующим образом.

Обозначим через  $(e_i, e_j)$  скалярное произведение векторов  $e_i$  и  $e_j$ ,  $i, j = \overline{1, k}$ . Тогда  $\|e_1 \wedge \dots \wedge e_k\| = [\det A]^{1/2}$ ,

где матрица  $A = (a_{ij})$ ,  $a_{ij} = (e_i, e_j)$ ,  $i, j = \overline{1, k}$ .

Выражение  $\|U_{x_0}(t)e_1 \wedge \dots \wedge U_{x_0}(t)e_k\|$  обозначает объем параллелепипеда  $\varphi^t(P^k)$ .

*Определение.* Предел (если он существует)

$$\lambda(x_0, e^{(k)}) = \lim_{t \rightarrow +\infty} \frac{1}{t} \ln \frac{\|U_{x_0}(t)e_1 \wedge \dots \wedge U_{x_0}(t)e_k\|}{\|e_1 \wedge \dots \wedge e_k\|}, \quad (3.2)$$

называется  $k$ -мерным показателем Ляпунова траектории  $\Gamma(x_0)$ . Этот показатель представляет собой «меру» скорости изменения объема параллелепипеда  $P^k$  при его перемещении вдоль траектории  $\Gamma(x_0)$ .

Нетрудно показать, что  $\lambda(x_0, e^{(k)})$  зависит лишь от подпространства  $e^{(k)}$ , а не от конкретного выбора векторов базиса. Наиболее важная информация, касающаяся показателей Ляпунова, содержится в следующей теореме.

**Теорема.**

1 Одномерные показатели Ляпунова могут иметь не более  $n$  различных значений  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ .

2  $k$ -мерные показатели Ляпунова могут иметь  $C_n^k$  различных значений, причем каждый из них представляет собой сумму  $k$  одномерных показателей Ляпунова.

3 Если линейно независимые векторы  $b_1, \dots, b_n$  выбраны случайным образом, то выражение в правой части формулы (3.2) с вероятностью 1 сходится к максимальному  $k$ -мерному показателю Ляпунова  $\lambda_{\max}^{(k)}$ .

Показатель Ляпунова  $\lambda(x_0, e^{(k)})$  описывает поведение траекторий, проходящих через точки  $x_0 + \varepsilon e$ ,  $|\varepsilon| < 1$ , по отношению к траектории  $\Gamma(x_0)$  (рисунок 3.1).

Если  $\lambda(x_0, e^{(k)}) < 0$ , то это означает, что указанные траектории приближаются к  $\Gamma(x_0)$  при  $t \rightarrow +\infty$ , а если  $\lambda(x_0, e^{(k)}) > 0$ , то удаляются.

Если в этом случае изменить начальное состояние  $x_0$  на  $x_0 + \varepsilon e$ ,  $\varepsilon \leq 1$ , то разность  $\varphi_{x_0 + \varepsilon e}(t) - \varphi_{x_0}(t)$  будет расти со временем с экспоненциальной скоростью: поведение динамической системы очень чувствительно к изменению начальных условий.



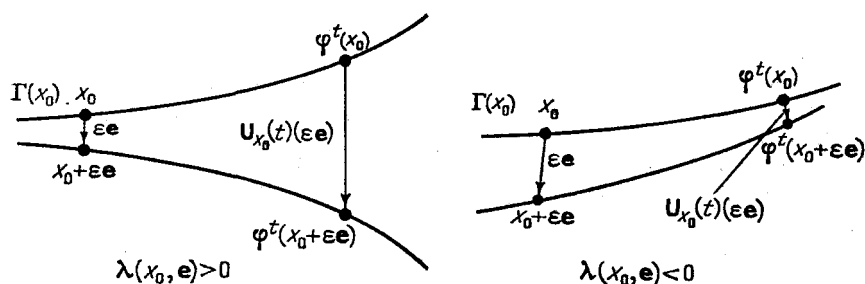


Рисунок 3.1 – Поведение близких траекторий

Если в формуле (3.2) положить  $e=f(x_0)$ , т. е. выбрать вектор, касательный к  $\Gamma(x_0)$ , то вектор  $Ux_0(t)e$  будет касательным все время: имеет место соотношение  $Ux_0(t)f(x_0)=f(\varphi_{x_0}(t))$ . Если векторное поле  $f(x)$  ограничено ( $\|f(x)\|<k$  для всех  $x \in R^n$ ), то из (3.2) вытекает, что  $\lambda(x_0, f(x_0)) = 0$ .

Для того чтобы один одномерный показатель Ляпунова траектории  $\Gamma(x_0)$  был равен нулю, достаточно, чтобы функция  $f(x)$  была ограничена в некоторой окрестности ее  $\omega$ -предельного множества.

Второе утверждение теоремы мы проиллюстрируем на примере траектории в  $R^3$ . Согласно п. 1 этой теоремы, существует три одномерных показателя Ляпунова. Согласно п. 2 этой же теоремы, число двумерных показателей Ляпунова равно  $C_n^k = C_3^2 = 3$  и их можно найти с помощью одномерных показателей Ляпунова:

$$\lambda_1^{(2)} = \lambda_1 + \lambda_2, \lambda_2^{(2)} = \lambda_1 + \lambda_3, \lambda_3^{(2)} = \lambda_3 + \lambda_2 \quad (3.3)$$

трехмерный показатель Ляпунова будет всего один:

$$\lambda_1^{(3)} = \lambda_1 + \lambda_2 + \lambda_3 \quad (3.4)$$

Третий пункт теоремы указывает на возможность вычисления всех одномерных показателей Ляпунова следующим способом. Выберем случайным образом базис  $b_1, b_2, \dots, b_n$  и затем с помощью ЭВМ найдем максимальные  $k$ -мерные показатели Ляпунова,  $k = \overline{1, n}$ . Обозначим их  $\lambda_{\max}^{(1)}, \dots, \lambda_{\max}^{(n)}$ . Тогда все одномерные показатели Ляпунова можно определить с помощью следующих соотношений:

$$\left\{ \begin{array}{l} \lambda_1 = \lambda_{\max}^{(1)}, \\ \lambda_2 = \lambda_{\max}^{(2)} - \lambda_{\max}^{(1)}, \\ \lambda_3 = \lambda_{\max}^{(3)} - \lambda_{\max}^{(2)}, \\ \dots \\ \lambda_n = \lambda_{\max}^{(n)} - \lambda_{\max}^{(n-1)}. \end{array} \right. \quad (3.5)$$

Одномерные показатели Ляпунова  $\lambda_j^{(1)}$  для стационарного решения ( $p(t) = x_0$ ) связаны с собственными значениями  $\mu_j$  матрицы Якоби  $J=f'(x_0)$  соотно-

шениями:  $\lambda_j = \operatorname{Re} \mu_j$ . Показатели  $\lambda_j^{(1)}$  для периодического решения  $p(t)$  периода  $T$  выражаются через его мультипликаторы  $\mu_j$ .

$$\lambda_j^{(1)} = \frac{1}{T} \ln |\mu_j|. \quad (3.6)$$

Рассмотрим проекцию траектории на трехмерный объем  $(p_1, q_1, q_2)$ .

Если движение ограничено, то траектория будет все время пересекать определенную плоскость  $q_2 = \text{const}$  внутри этого объема. Эту плоскость, заданную координатой  $q_1$  и сопряженным импульсом  $p_1$ , удобно выбрать в качестве поверхности сечения  $\Sigma_R$ . В общем случае последовательные пересечения траектории с поверхностью  $\Sigma_R$  будут произвольно распределены.

Хаотическое движение, наблюдаемое в гамильтоновых системах, является следствием их динамики, а не конечной точности счета. Численные эксперименты, в которых точность счета изменялась, приводит к периодичности движения.

Пусть функция дифференцируема,  $A$  – аттрактор отображения  $\mathbf{x}_{k+1} = F(\mathbf{x}_k)$ . Обозначим  $F^k$  суперпозицию  $F$  с собой  $k$  раз;  $DF^k(\mathbf{x})$  – матрица Якоби, т.е. матрица  $n \times n$  частных производных  $F^k$ , вычисленных в точке  $\mathbf{x}$ . Пусть  $a_i(k, \mathbf{x})$  – модуль  $i$ -го собственного значения  $DF^k(\mathbf{x})$ , причем. Характеристические показатели Ляпунова аттрактора определяются соотношениями (3.6).

При подходящих предположениях этот предел существует и одинаков для типичных точек  $\mathbf{x}$  на аттракторе  $A$ . Подобное определение может быть дано и для аттрактора автономной системы обыкновенных дифференциальных уравнений. При этом индекс  $k$  заменяется временем  $t$ . Упорядоченная по убыванию последовательность чисел образует спектр показателей Ляпунова и дает полезную классификацию аттракторов. В таблице 3.1 приведена классификация аттракторов динамических систем, заданных системой дифференциальных уравнений.

Таблица 3.1 – Спектр характеристических показателей Ляпунова и устойчивость динамических информационных систем

Тип аттрактора	Размерность фазового пространства	Знаки показателей Ляпунова
Неподвижная точка	1	(–)
Неподвижная точка	2	(–, –)
Предельный цикл	2	(0, –)
Неподвижная точка	3	(–, –, –)
Предельный цикл	3	(0, –, –)
Двумерный тор	3	(0, 0, –)
Странный аттрактор	3	(+, 0, –)

Сумма показателей Ляпунова траектории  $\mathbf{x}^0(t)$  характеризует скорость изменения фазового объема в ее окрестности. Режим странного аттрактора реализуется только в диссипативных системах и характеризуется наличием в спектре положительных показателей.

Сумма показателей Ляпунова для диссипативных систем отрицательна. Если сумма показателей Ляпунова равна нулю, то фазовый объем системы во времени не изменяется – система консервативна и аттракторов не содержит. В случае положительной суммы показателей Ляпунова фазовый объем во времени нарастает. С физической точки зрения такой режим как стационарный не реален.

Показатели Ляпунова, являясь усредненными характеристиками аттрактора, описывают его свойства независимо от начальных условий из области притяжения. Исключение представляют лишь начальные условия, соответствующие нетипичным траекториям, имеющим меру нуль. Это подтверждается численными экспериментами.

Установлена количественная взаимосвязь показателей Ляпунова с энтропией Колмогорова. Доказано, что энтропия положительна в том и только в том случае, когда фазовая траектория в среднем экспоненциально неустойчива на аттракторе. Значит, спектр показателей Ляпунова такой траектории обязан содержать положительный показатель.

### ***Практические алгоритмы вычисления характеристических показателей Ляпунова***

Пусть динамическая система задана дискретным дифференцируемым отображением:

$$C(r) = \frac{1}{m(m-1)/2} \sum_{i=0}^{m-2} \sum_{j=i+1}^{m-1} \theta(r - \rho(x_i, x_j)). \quad (3.7)$$

$\mathbf{x}_{k+1} = F(\mathbf{x}_k)$ ,  $DF(\mathbf{x})$  – матрица Якоби.

Бенеттин придумал численный алгоритм вычисления показателей Ляпунова аттрактора такого отображения. Он начинается с ортонормированного базиса:

$$\left\{ \mathbf{u}_1^{(0)}, \dots, \mathbf{u}_n^{(0)} \right\}. \quad (3.8)$$

На  $k$ -м шаге вычисляют:

$$\bar{\mathbf{u}}_j^{(k+1)} = DF(\mathbf{x}_k) \mathbf{u}_j^{(k)}, \quad j = \overline{1, n}. \quad (3.9)$$

Затем применяют процедуру ортогонализации Грама-Шмидта:

$$\mathbf{u}_1^{(k+1)} = \bar{\mathbf{u}}_1^{(k+1)} / \delta_1^{(k+1)}, \quad (3.10)$$

где  $\delta_1^{(k+1)} = \|\bar{\mathbf{u}}_1^{(k+1)}\|$

На последующих шагах полагают:

$$\delta_m = \left\| \bar{u}_m - \sum_{i=1}^{m-1} (u_i, \bar{u}_m) u_i \right\| \quad (3.11)$$

где  $(u, v)$  – обычное скалярное произведение (индексы  $(k+1)$  опущены для простоты).

Здесь –  $\delta_m$  длина  $m$ -го вектора после ортогонализации, но перед нормализацией.

Показатели Ляпунова численно выражаются как

$$\lambda_m = \frac{1}{k} \sum_{i=1}^k \ln \delta_m^{(i)} \quad (3.12)$$

для больших  $k$ . Подобный алгоритм применим и к потоку, за исключением того, что вариационные уравнения

$$\dot{u}_i = DF(x)u_i \quad (3.13)$$

интегрируют от  $t$  до  $t + \Delta t$ , начиная с ортонормированного базиса

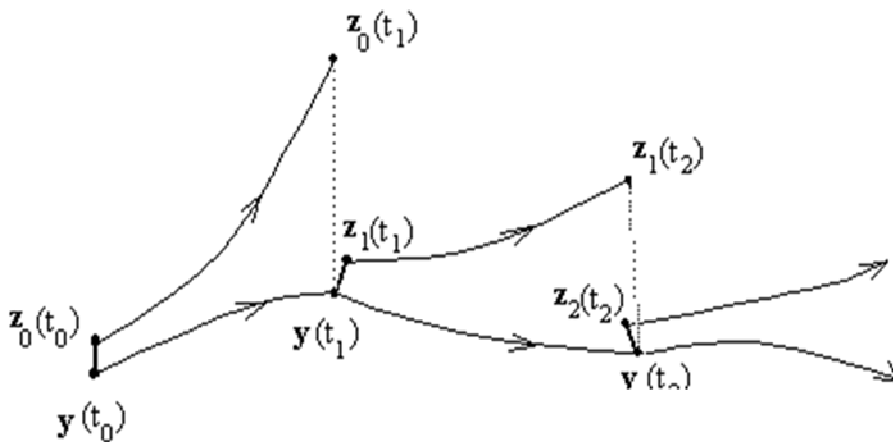


Рисунок 3.2 – К вычислению вариаций (3.12-3.13)

Ортонормализационная процедура применяется как и выше, но множитель  $1/k$  в (3.12) заменяется на  $1/k \Delta t$ .

Если математическая модель динамической системы неизвестна, то требуются другие алгоритмы нахождения показателей Ляпунова, поскольку в этом случае мы не знаем ни матрицы Якоби, ни размерности фазового пространства системы.

В 1985 году Вольф описал алгоритм нахождения наибольшего показателя Ляпунова по временному ряду данных. Он следит за парой точек на аттракторе для оценки на каждом шаге по времени, по которому наибольший показатель Ляпунова может быть вычислен используя (3.12). Схематическая иллюстрация

алгоритма приведена на рисунке 3.2. Он начинается с первой точки данных  $y(t_0)$  и ее ближайшего соседа  $z_0(t_0)$ , которые отделены на расстояние  $r_0$ .

Эти две точки эволюционируют во времени с шагом пока расстояние  $r'_0$  между ними не превысит некоторую величину .

Эволюционирующая первая точка данных  $y(t_1)$  сохраняется, а новый сосед  $z_1(t_1)$  ищется такой, что расстояние

$$r_1 = \|y(t_1) - z_1(t_1)\| \quad (3.14)$$

снова меньше, чем  $\varepsilon$  и такой, что  $z_1(t_1)$  лежит так близко, насколько это возможно в одинаковом направлении, что и направление от  $y(t_1)$  к  $z_0(t_1)$ .

Процедура продолжается до тех пор, пока принятая за основу сравнения траектория  $y$  не дойдет до конца временного ряда. Наибольший показатель Ляпунова аттрактора оценивается как

$$\lambda_1 = \frac{1}{m * \Delta t} \sum_{k=0}^{L-1} \ln \frac{r'_k}{r_k}, \quad (3.15)$$

где  $L$  – число шагов замены и  $m$  – общее число шагов по времени, в течение которых описывается траектория  $y$ .

Каждая заменяющая точка должна лежать в том же направлении, что и старая, но из-за ограниченного числа отсчетов данных необходимы компромиссы. Первоначально поиск точки для замены ограничен в конусе угловой ширины  $\Theta$  и высоты  $\varepsilon$  около  $y(t)$ .

Если  $\lambda_1$  не очень маленькое, то алгоритм Вольфа может быть полезен для определения того, хаотичный ли наблюдаемый временной ряд. Часто, однако, интересуются более чем одним первым показателем Ляпунова. Алгоритм может расширяться для измерения  $\lambda_1 + \lambda_2$ , но становится при этом существенно сложнее. Он должен следить за треугольником точек и сохранять ориентацию относительно треугольника, которому выбирают замену. Этот подход становится плохим для более чем двух положительных показателей Ляпунова.

В том же 1985 году Экманн и Рюэль предложили метод оценки вариационных уравнений, который в принципе может дать все показатели Ляпунова аттрактора. Они следят, по крайней мере, за  $n+1$  точкой, по которым методом наименьших квадратов оценивают матрицу Якоби  $DF(x)$ . Показатели Ляпунова затем можно получить, проинтегрировав вариационные уравнения, используя алгоритм Бенеттина.

Основной недостаток такого подхода – в зависимости результатов от размерности вложения аттрактора.

Прямое вычисление наибольшего показателя Ляпунова основано на слежении за экспоненциально расходящимися соседними траекториями. Однако в присутствии шума детерминированная расходимость на малых масштабах скрыта шумом. Расстояние между траекториями изменяется подобно случайному блужданию. В результате временная зависимость расстояния  $r$  между первоначально близкими траекториями задается формулой:

$$r(k) \approx \alpha\sigma\sqrt{k} + \beta\exp(\lambda_1 k) . \quad (3.16)$$

Следовательно, для малых  $k$  находят  $\lambda_1$ , который существенно больше истинного. Чистое экспоненциальное разбегание возможно только для траекторий, начальное разделение которых больше уровня шума. Однако как только расстояния становятся порядка размера аттрактора, дальнейшего увеличения расстояний по экспоненциальному закону не происходит.

**Индивидуальные задания** те же, что и в лабораторной работе 2. Кроме того, необходимо провести сравнительный анализ качественных и количественных характеристик устойчивости динамических информационных систем для всех диапазонов исходных данных.

## **Задание к лабораторной работе № 4 «Исследование фрактальных свойств информационных систем»**

**Цель занятия:** Получить навыки исследования информационных объектов на основе их фрактального описания

### **Задачи на работу:**

1 Провести анализ основных существующих типов фрактальных структур, применяемых в информационных системах.

2 Разработать сводную таблицу сравнительных характеристик свойств типовых фракталов на основе количественного подхода к информации (энтропия Колмогорова и размерность Хаусдорфа).

3 Разработать программу на (любом) языке высокого уровня реализующую основные известные фракталы (по вариантам).

4 Оформить отчет (индивидуально для каждого студента) исследований и защитить его.

### **Содержание отчета:**

1 Тема, цель и задачи лабораторного занятия.

2 Результаты изучения понятий «энтропия Колмогорова» и «размерность Хаусдорфа».

3 Программа в электронном варианте (отлаженная).

4 Результаты тестирования полученной программы на заданном наборе исходных данных (индивидуальных заданий).

5 Выводы.

### **Краткая теория. Фрактальное описание информационных систем**

Понятие фрактала не имеет строгого определения. Мандельброт поясняет понятие фрактала как некоего образования, самоподобного или самоаффинного в том или ином смысле. Только такое пояснение позволяет охватить без видимых досадных пробелов широкое множество объектов, достойных называться фракталами. Любая попытка дать более строгое определение отсекает какой-то достаточно емкий класс объектов, непозволительно сужая мир фракталов.

Простейшие фракталы, такие, как канторовская пыль, снежинки и ломаные фон Коха, ковер и губка Серпинского, кривые дракона, кривые Пеано и Гильберта и многие другие, обладают регулярной геометрически правильной структурой. Каждый фрагмент такого геометрически правильного фрактала в точности повторяет всю конструкцию в целом. При менее точном следовании самоаффинности или самоподобию возникают другие, например, случайные фракталы, в которых самоаффинность заключается в сохранении нормальности случайного распределения на разных масштабах, возможно, с различными дисперсиями и средними. Примерами случайных фракталов могут служить пограничные и береговые линии, поры в хлебе, дырки в некоторых сортах сыра, частицы в порошках и т.д. Фрактальные примеры могут быть как континуальными, так и дискретными. Например, удивительными свойствами самоподобия обладает последовательность М. Морса–А. Туэ, возникающая в самых различ-

ных нелинейных динамических ситуациях от символической динамики до чисел Фибоначчи и треугольника Паскаля.

Фракталы встречаются везде, где заканчиваются правильные формы евклидовой геометрии. Все, что создано человеком, ограничено плоскостями. Если встречается природный объект, то с первого взгляда видно, что осознать, описать его форму со всеми шероховатостями можно только приблизительно. Здесь на помощь приходят фракталы.

Однако особое место занимают фракталы в исследовании сложных динамических информационных системах. Известно, что *странные аттракторы систем со сложным поведением представляют из себя именно фракталы*. А с учетом особенностей их построения (по любой, даже сравнительно малой части фрактала, можно построить весь фрактал) восстановление аттрактора по ограниченному числу данных (измерений), а, следовательно, и характеристик фазового портрета исследуемой системы, представляется весьма интересным.

Термин «фрактал» (от английского слова «*fraction*» – дробь) введен бельгийским математиком Бенуа Мандельбротом и обозначает множество, имеющее дробную фрактальную размерность и обладающее свойством самоподобия. Для пояснения фрактальной размерности необходимо ввести понятие топологической размерности. Под топологической размерностью  $D_t$  множества в линейном пространстве понимают число линейно независимых координат в пространстве. Например, окружность и линия имеют топологическую размерность 1; круг и квадрат – 2; шар и куб – 3. Фрактальная размерность множества  $D$  – размерность того пространства, которое полностью заполняется множеством. Для связи фрактальной и топологической размерностей используют показатель Херста  $H$ , вычисляемый по формуле:  $H = D - D_t$ .

Фрактальная размерность множества обычно не совпадает с топологической. Например, для кривых Пеано (кривые, заполняющие плоскость)  $D_t = 1$ ,  $D = 2$ .

Рассмотрим классический пример фрактального множества – триадную ломаную Коха (рисунок 4.1).



Рисунок 4.1 – Построение триадной кривой Коха

Построение кривой начинается с единичного отрезка, который называется инициатором и является предфракталом 0-го порядка. Далее инициатор заменяется на образующий элемент – кривую из четырех прямолинейных звеньев, каждое из которых имеет длину  $1/3$ . Так образуется предфрактал 1-го порядка длиной  $4/3$ .

Для построения предфрактала следующего порядка каждое звено заменяется на уменьшенный образующий элемент. В результате получаем кривую, состоящую из  $4 \times 4 = 16$  звеньев, каждое из которых имеет длину  $(1/3) / 3 = 1/9$ , общая длина равна  $16/9$ . Длина предфрактала  $n$ -го порядка равна  $(4/3)^n$ . Очевидно, что предел длины кривой при  $n$ , стремящемся к бесконечности, равен бесконечности. В итоге получили кривую бесконечной длины, заполняющую ограниченное множество на плоскости, что само по себе очень любопытно. Если построение кривой начинать не с отрезка, а с треугольника, и применить вышеперечисленные построения к каждой его стороне, то получим «снежинку» Коха (рисунок 4.2).



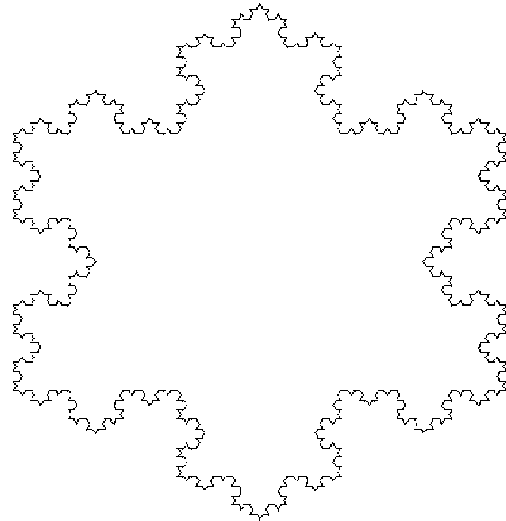


Рисунок 4.2 – «Снежинка» Коха (Предфрактал 4-го порядка)

Она была так названа в честь шведского математика *Helge von Koch*, который впервые описал ее в 1904. Отличительной ее особенностью является то, что она, будучи замкнутой, тем не менее, нигде себя не пересекает, поскольку достраиваемые треугольники каждый раз достаточно малы и никогда не «сталкиваются» друг с другом.

Подсчитаем ее фрактальную размерность. Возьмем в качестве длины стороны исходного треугольника  $l = 1$ , тогда число отрезков такой длины, которые покрывают снежинку Коха на этом (нулевом) шаге, равно  $N(l) = 3$ . Затем при переходе к следующему шагу мы имеем  $l' = 1/3$ , а число отрезков  $N(l') = 12$ . Поэтому фрактальная размерность снежинки Коха равна

$$D = -\frac{\ln(3/12)}{\ln(3)} = \frac{\ln 4}{\ln 3} = 1.2618 \quad (4.1)$$

Эта величина больше единицы (топологической размерности линии), но меньше Евклидовой размерности плоскости,  $d = 2$ , на которой расположена кривая. Таким образом, снежинка Коха представляет собой линию бесконечной длины, ограничивающую конечную площадь.

### ***Системы итерирующих функций (IFS). Математическая генерация хаоса***

Как видно во всех рассмотренных выше случаях, каждое линейное преобразование включало в себя сжатие в  $t$  раз (одинаковое для обеих осей  $X$  и  $Y$ ) и параллельный перенос на некоторый комплексный вектор.

Наши возможности существенно расширятся, если в систему преобразований мы включим еще и повороты. Найдем систему итерирующих функций (СИФ) для кривой Коха.

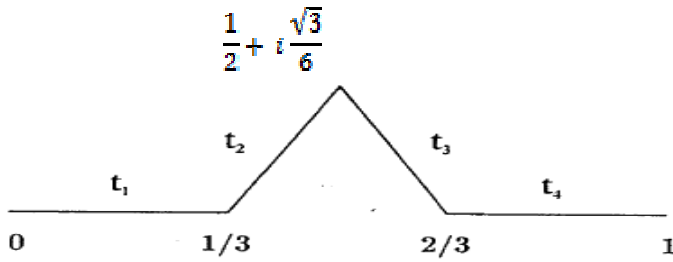


Рисунок 4.3 – Генератор для кривой Коха.

Инициатором для кривой Коха является отрезок единичной длины, который на следующем шаге преобразуется в генератор, изображенный на рисунке 4.1. Числа показывают координаты соответствующих вершин на комплексной плоскости. Система итерируемых функций, осуществляющих данное преобразование, выглядит следующим образом

$$\begin{aligned}
 f_1(z) &= \frac{1}{3}z, \\
 f_2(z) &= \frac{1}{3}e^{i\pi/3}z + \frac{1}{3}, \\
 f_3(z) &= \frac{1}{3}e^{-i\pi/3}z + \frac{3+i\sqrt{3}}{6}, \\
 f_4(z) &= \frac{1}{3}z + \frac{2}{3}.
 \end{aligned} \tag{4.2}$$

Первое преобразование  $t_i$ , осуществляемое функцией  $f_i(z)$ , есть просто сжатие в 3 раза исходного отрезка  $(0,1)$ . Второе преобразование (включает в себя такое же сжатие, поворот против часовой стрелки вокруг начала координат на угол в  $60^\circ$  и последующий сдвиг по горизонтали вправо на  $1/3$ ). Третье преобразование  $t_3$  сжимает в три раза исходный отрезок, поворачивает его на угол  $60^\circ$  по часовой стрелке и затем смещает параллельно самому себе на комплексный вектор  $\frac{1}{2} + i\frac{\sqrt{3}}{6}$ . Наконец, четвертое преобразование сжимает отрезок в 3 раза и смещает его по горизонтали вправо на  $2/3$ . Заметим, что порядок выполняемых операций здесь важен.

Если теперь этим четырём преобразованиям подвергнуть сам генератор, то получим конструкцию, изображённую на рисунке 4.4. Она состоит из 16 отрезков длиной  $1/9$ , и для каждого из них показана последовательность операций, с помощью которых он был получен из исходного единичного отрезка  $K$ .

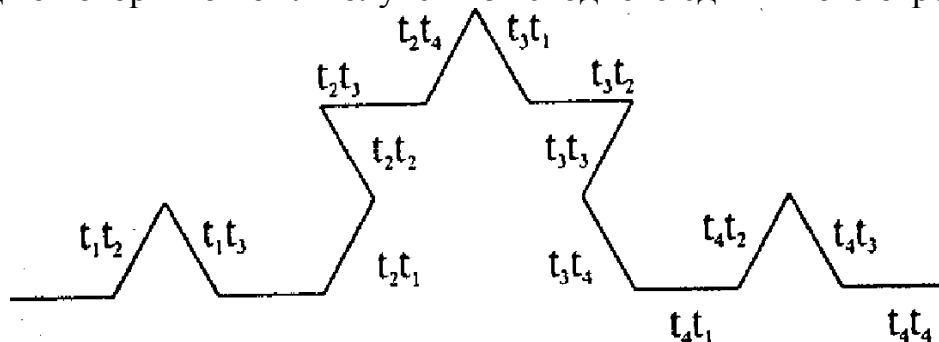


Рисунок 4.4 – Положения элементов  $t_i$   $t_i(K)$

Саму кривую Коха можно получить, повторяя этот процесс до бесконечности. Для этого удобно, как и ранее, воспользоваться методом случайных итераций (игрой в хаос). Каждому из 4 преобразований (4.2) припишем одинаковую вероятность  $p_i = 0.25$ . После этого, начав с некоторой точки  $z_0$  в комплексной плоскости и, выбирая случайным образом, последовательность преобразований, будем получать все новые и новые точки –  $z_1 = f_i(z_0)$ ,  $z_2 = f_j(z_1)$ ,  $z_3 = f_k(z_2)$  и т.д.

В итоге после примерно 150000 итераций придем к множеству точек.

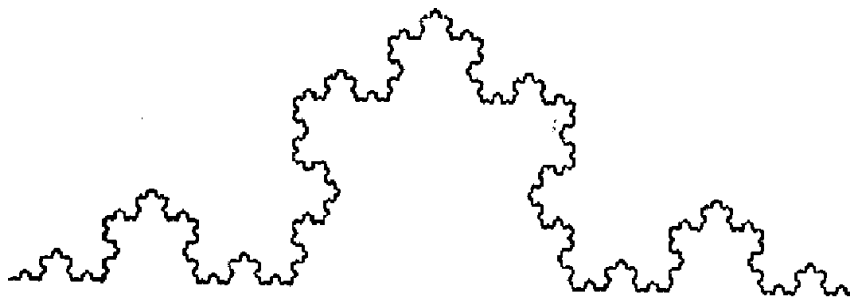


Рисунок 4.5 – Кривая Коха, полученная методом случайных итераций системы функций

Использование такого типа генераторов используется достаточно широко (смотри, например, [9, 10]).

Многие регулярные фракталы любой размерности строятся путем бесконечного повторения нескольких простых операций, скажем, замены одного элемента некоторой комбинацией других, ему подобных. Так, например, салфетка Серпинского получается при замене исходного большого треугольника тремя треугольниками в два раза меньшего размера. Затем эта же операция повторяется с каждым из этих трех маленьких треугольников, и так далее до бесконечности. Ситуация примерно та же, что и для кривой Коха.

Поместим исходный равносторонний треугольник с длиной стороны, для определенности равной единице, на комплексную плоскость. Определим, каким линейным преобразованием  $t_i$  на комплексной плоскости он переводится в равносторонний треугольник в два раза меньшего размера, показанный на рис. 1.20 слева? Ответ достаточно прост. Поскольку левое основание обоих треугольников лежит в начале координат  $z = 0$ , то функция  $f_{10}(z)$ , осуществляющая это преобразование, определяется выражением, показанном на рисунке 4.6:

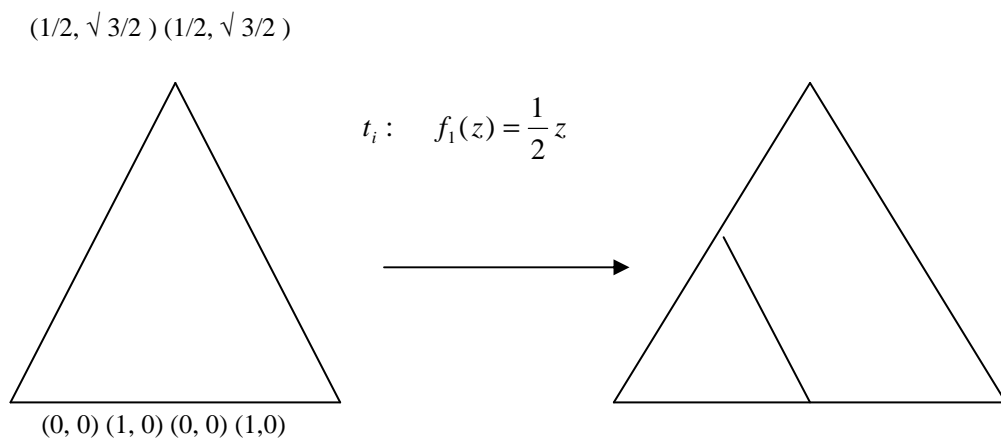


Рисунок 4.6 – Преобразование  $t_1$ . В скобках – декартовы координаты вершин.

Если теперь сместить этот маленький треугольник по горизонтали вправо на величину, равную  $1/2$ , то получим преобразование  $t_2$ , переводящее исходный треугольник в треугольник, изображенный на рисунке 4.7.

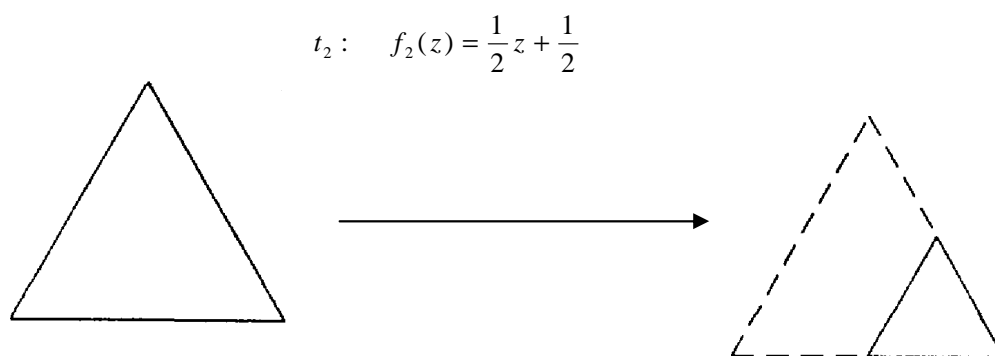


Рисунок 4.7 – Преобразование  $t_2$

Наконец, последний, третий, маленький треугольник получается с помощью преобразования  $t_3$  показанного на рисунке 4.8.

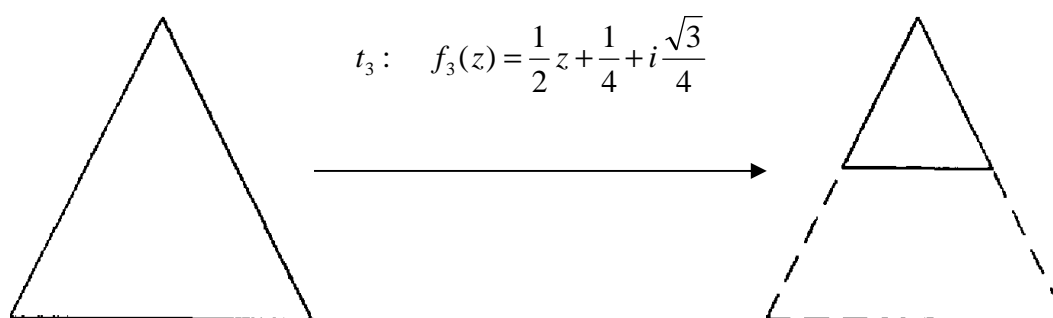


Рисунок 4.8 – Преобразование  $t_3$

В итоге три вышеназванные линейные функции  $f_1(z)$ ,  $f_2(z)$  и  $f_3(z)$  осуществляют искомое преобразование исходного треугольника в три треугольни-

ка в два раза меньшего размера. Возникает вопрос, а что будет, если теперь каждый из этих трех маленьких треугольников свою очередь подвергнуть этим трем преобразованиям. Тогда возникнет уже 9 треугольников с размером в 4 раза меньше исходного. Непосредственной проверкой можно убедиться, что это приводит к картинке, изображенной на рисунке 4.10 справа.

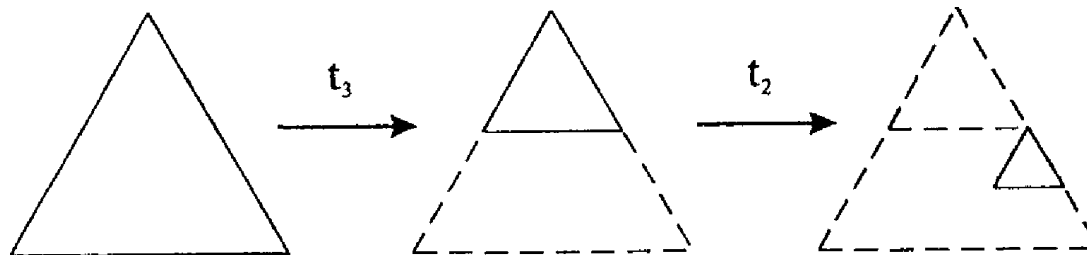


Рисунок 4.9 – Преобразование  $t_2 t_3$

Общий случай показан на рисунке 4.11, где изображены все эти треугольники с обозначением результирующего преобразования – *генеалогического кода*, при помощи которого они были получены из исходного треугольника.

Слева показан первый шаг итерационной процедуры. Большой треугольник, в который «вписаны» подобным образом три маленьких треугольника в два раза меньшего размера, мы будем ниже называть ячейкой.

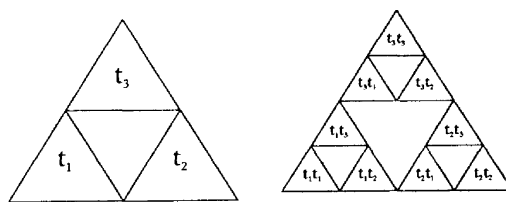


Рисунок 4.10 – Два первых поколения итераций системы

Комбинация  $t_j t_i$ , стоящая в каждом из девяти маленьких треугольников, означает, что этот треугольник был получен из исходного сначала применением преобразования  $t_i$ , а затем к полученному треугольнику было применено преобразование  $t_j$ . Правило построения этой последовательности легко угадывается. На первом месте справа стоит первое преобразование. Оно соответствует позиции данного треугольника в его ячейке в соответствии с обозначениями на рисунке 4.11 (слева). На втором месте стоит второе по счету преобразование, которое соответствует позиции уже этого большого треугольника в его ячейке и т.д. Отметим очевидную некоммутативность двух (разных) преобразований, т.е. генеалогические коды  $(t_1 t_2)$  и  $(t_2 t_1)$  соответствуют разным треугольникам.

Ясно, что, действуя подобным образом, мы в точности воспроизводим алгоритм построения салфетки Серпинского. Поэтому после бесконечного числа шагов мы приходим ко множеству точек, образующих этот фрактал.

В результате получается «дырявая» фигура (см. рис. 1.5), состоящая из бесконечного числа изолированных точек. Фрактальная размерность салфетки Серпинского равна

$$D = \frac{\ln 3}{\ln 2} = 1.5849. \quad (4.4)$$

Сам по себе этот факт безусловно интересен. Но, с другой стороны, что мы при этом узнали нового? Принципиально новое заключается в том, что для получения точно такого же предельного результата мы могли бы стартовать с любой фигуры, необязательно имеющей форму равностороннего треугольника. Это, например, мог быть круг или квадрат или любая другая замысловатая (и даже несвязная) фигура, произвольным образом расположенная на плоскости. На каждом шаге уменьшаясь в размерах в два раза и утраиваясь в количестве, эти фигуры, в конце концов, превратились бы в неразличимые глазом бесформенные точки, образующие фрактал – салфетку Серпинского.

Причина такого поведения заключается в том, что салфетка является своеобразным аттрактором для этой системы из трех линейных преобразований  $f_1(z)$ ,  $f_2(z)$  и  $f_3(z)$  называемых в литературе Системой Итерируемых Функций или сокращенно СИФ. Поскольку салфетка – аттрактор, то процесс его построения можно было начать даже с одной единственной точки.

**Метод случайных итераций, или игра в хаос**

Рассмотрим следующую незамысловатую игру, которую М. Барнсли назвал *игрой в хаос (chaos game)*. Возьмем уже знакомый нам равносторонний треугольник с вершинами в точках  $A$ ,  $B$  и  $C$ . Выберем внутри этого треугольника произвольным образом начальную точку. Бросим теперь игральную кость, представляющую собой кубик, на 6 гранях которого проставлены буквы  $A$ ,  $B$  и  $C$ . Пусть каждая буква присутствует на двух из них, тогда вероятность выпадения любой буквы одинакова и равна  $1/3$ .

Допустим, что в результате первого броска выпала буква  $A$ . Соединим мысленно нашу начальную точку с вершиной треугольника  $A$  отрезком прямой и на его середине поставим точку (см. рисунок 4.11).

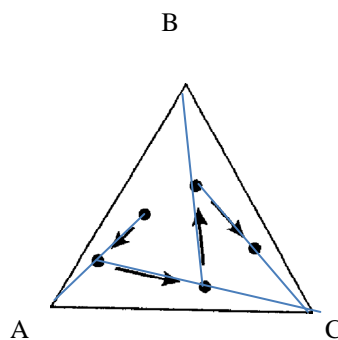


Рисунок 4.11 – Игра в хаос. Первые 4 шага.

Пусть теперь она будет играть роль начальной. После чего повторим вышеописанную процедуру с бросанием кубика и проставлением точки в середине соответствующего отрезка. Допустим, на втором шаге выпала буква  $C$ , потом  $B$ , затем опять  $C$  и т. д. В результате на каждом шаге мы будем получать все новые и новые точки. Спрашивается, как распределятся внутри треугольника эти точки после достаточно большого числа шагов.

Ниже, на рисунке 4.12 (слева направо), показаны результаты этой игры соответственно с 5000, 10000 и 50000 точек. По мере увеличения числа точек все яснее проступает структура треугольника Серпинского.

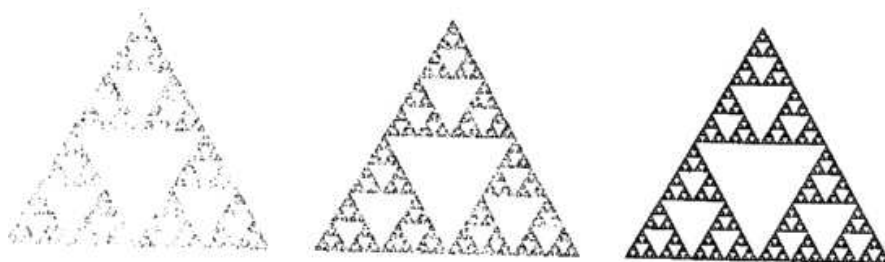


Рисунок 4.12 – Игра в хаос. Результат

Видно, что, хотя каждый раз выбор вершины треугольника происходит чисто случайным образом, возникающее множество точек на плоскости отнюдь не случайно и обладает ярко выраженной фрактальной структурой.

Таким образом, треугольник Серпинского, являясь аттрактором для этой системы итерируемых функций, возникает и при чисто случайном выборе последовательности преобразований  $t_i t_j t_k \dots$ . Можно показать, что изображающая точка в результате бесконечной цепочки случайных итераций сколь угодно близко подойдет к каждой точке этого фрактального множества.

Разумеется, для этой игры было совершенно несущественно, что исходный треугольник являлся равносторонним.

С равным успехом ее можно было провести в треугольнике любой формы (см. рисунок 4.13).

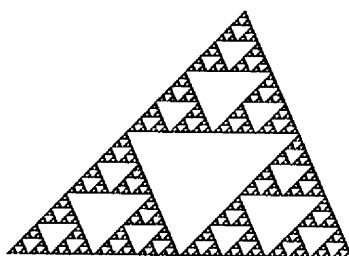


Рисунок 4.13 – Скошенный треугольник Серпинского

А что будет, если мы теперь несколько изменим правила игры? Например, будем проставлять точку не на середине отрезка, а на расстоянии в  $1/3$  от соответствующей вершины. Результат показан на рис. 1.30. Получившееся множество точек можно назвать двумерным аналогом канторовского множества исключенных средних третей. Нетрудно подсчитать, что фрактальная размерность соответствующего аттрактора равна единице.

В качестве исходной фигуры можно выбрать и любой другой многоугольник.

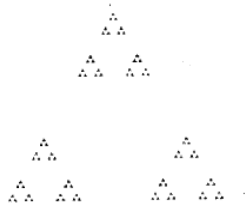


Рисунок 4.14 – Фрактальная пыль,  $D = 1$

Еще одним интересным направлением исследований «Игры в хаос» будет задание различных (неравных) вероятностей выбора направлений к соответствующим вершинам. Студентам предлагается самостоятельно исследовать этот вопрос в теоретическом и практическом плане.

### ***Сжимающие аффинные преобразования***

У фрактальной математики возникают все новые и новые сферы применения. Коснемся лишь одного перспективного направления – создания алгоритма фрактального сжатия графической информации. В 1991 году такой алгоритм был найден. Он имеет ряд уникальных возможностей. Фрактальный архиватор позволяет, например, при распаковке произвольно менять разрешение изображения без появления эффекта зернистости. Более того, он распаковывает гораздо быстрее, чем ближайший конкурент, JPEG, и не только статическую графику, но и видео. В 1992 году компания Microsoft использовала фрактальный архиватор и выпустила компакт-диск Microsoft Encarta мультимедиа-энциклопедия, содержащий информацию о животных, цветах, деревьях и живописных местах. На диск было записано 7 часов звука, 100 анимационных роликов, примерно 800 масштабируемых карт, а также 7000 качественных фотографий. И все это – на одном диске! Напомним, что обычный компакт-диск в 650 Мбайт без использования компрессии может содержать либо 56 минут качественного звука, либо 1 час видео с разрешением  $320 \times 200$  в формате MPEG-1, либо 700 полноцветных изображений размером  $640 \times 480$ .

В настоящее время алгоритмы, используемые для генерации изображений фрактальной графики, находят применение и в традиционных видах компьютерной графики: растровой и векторной. Например, в CorelDRAW эти алгоритмы используются для создания текстурных заливок. В растровой программе PhotoDraw фирмы Microsoft кроме стандартных градиентных заливок контуров можно сгенерировать фрактальный узор и воспользоваться им в качестве заливки.

В самом общем случае для восстановления странного аттрактора не известно никакой определенной модели. Но чаще всего по одной наблюдаемой динамической переменной необходимо восстановить систему дифференциальных уравнений (СДУ) или дискретное отображение, которые управляют поведением данного временного ряда. Обычно модель задается системой обыкновенных дифференциальных уравнений

$$\frac{dx}{dt} = F(x)$$

где  $x$  – точка в  $n$ -мерном фазовом пространстве. Затем  $F(x)$  строится с помощью полиномов от фазовых переменных. В этот способ могут быть добав-



лены различные усовершенствования, включающие использование разложения временного ряда по некоторой системе базисных функций для облегчения эффективного выбора полиномов для СДУ и фильтрации шума в данных.

Стохастические характеристики «подогнанного» аттрактора могут быть затем сравнены с характеристиками «сырого» аттрактора с целью убедиться в адекватности предложенной модели. В простейшем случае модельные параметры входят линейно в СДУ. Типичные примеры – системы Лоренца и Ресслера. В более сложных ситуациях, таких как физический маятник, некоторые модельные параметры входят линейно, в то время как остальные – нелинейно. Доказано, что применение метода наименьших квадратов для поиска параметров дает хорошие результаты в обоих случаях. Допускается даже присутствие умеренного количества аддитивного шума.

### **Модельные данные**

Самый простой способ восстановления дифференциальных уравнений может быть реализован, если есть информация о типе странного аттрактора. Существующие алгоритмы вычисления стохастических характеристик аттракторов протестированы на различных видах модельных данных  $\{S_k\}_{k=0}^{M=1}$ .

В качестве примера диссипативной динамической системы с предельным циклом рассматривался классический нелинейный осциллятор Ван дер Поля, уравнения колебаний которого имеют вид

$$\ddot{x} - a(1 - bx^2)\dot{x} + x = 0. \quad (4.5)$$

Параметр  $a$  характеризует подкачку энергии в систему от внешнего источника и называется параметром возбуждения. Диссипация в осцилляторе нелинейным образом зависит от колеблющейся переменной  $x$ .

В фазовых координатах уравнение колебаний осциллятора записывается как

$$\begin{cases} \dot{x} = y \\ \dot{y} = a(1 - bx^2)y - x \end{cases} \quad (4.6)$$

со знакопеременной дивергенцией, тождественно не равной нулю:

$$a(1 - bx^2) \neq 0 \quad (4.7)$$

В общем случае (4.6) не интегрируются и исследования проводятся с использованием численных методов. В практически важном случае ( $a > 0, b > 0$ ) уравнения (4.6) имеют единственное устойчивое решение в виде предельного цикла  $\Gamma$ , изображенного на рисунке 4.15. В качестве временного ряда данных были выбраны отсчеты координаты  $x$ :

$$s_k = x(k \cdot \Delta t), \quad M = 8192, \quad \Delta t = 0.02, \quad k = 0 \dots M-1.$$

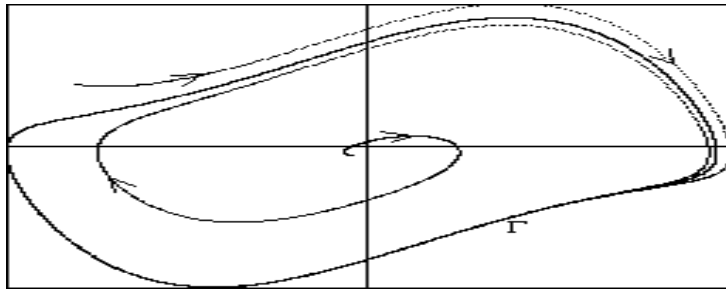


Рисунок 4.15 – Предельный цикл системы (4.6)

Численный счет проведен для значений параметров  $a=1$ ,  $b=0.32$ . Если добавить в уравнение (4.5) внешнюю силу амплитуды  $B$  и частоты  $p$ , несоизмеримой с частотой периодических колебаний автономного осциллятора, то оно перепишется в виде

$$\ddot{x} - a(1 - bx^2)\dot{x} + x = B \sin(pt). \quad (4.8)$$

Периодическая модуляция предельного цикла автономной системы приводит к тому, что фазовая траектория с заданной частотой  $p$  вращается вокруг предельного цикла и лежит на тороидальной поверхности. Аналогично случаю предельного цикла эта поверхность будет устойчивым предельным множеством, к которому стягиваются со временем все траектории из некоторой окрестности тора (как изнутри него, так и снаружи!). Минимальная размерность фазового пространства, в которое можно вложить двумерный тор, равна трем. В результате получается динамическая система вида:

$$\begin{cases} \dot{x} = y, \\ \dot{y} = a(1 - bx^2)y - x + z, \\ \dot{z} = B \sin(pt). \end{cases} \quad (4.9)$$

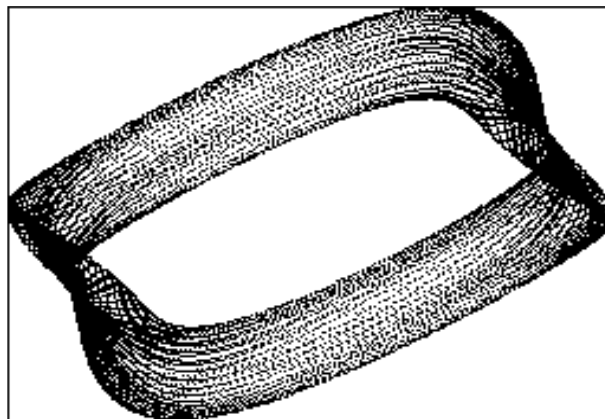


Рисунок 4.16 – Двумерный тор. Численный счет проводился для значений параметров  $a = 1$ ,  $b = 0.3$ ,  $B = 1$ ,  $p = 1.5$ .

Восстановленная в трехмерном пространстве реализация временного ряда данных вида  $M = 16384$ ,  $\Delta t = 0.02$  представлена на рисунке 4.16.

## Система Лоренца

$$\begin{cases} dx/dt = -\sigma x + \sigma y, \\ dy/dt = -xz + rx - y, \\ dz/dt = xy - bz. \end{cases}$$

Временной ряд данных

$$s_k = x(k * \Delta t) \quad k = \overline{0, M-1}, \quad M = 8192, \quad \Delta t = 0,02.$$

Параметрам  $\sigma = 10$ ,  $b = 8/3$ ,  $r = 28$  соответствует странный аттрактор (рисунок 4.17).

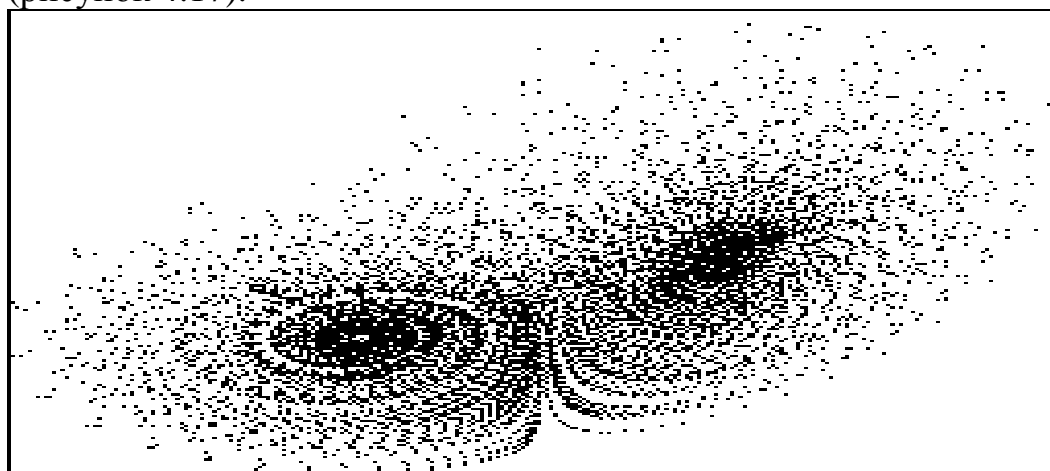


Рисунок 4.17 – «Фрактальная пыль» для модели Лоренца.

## Индивидуальные задания

1 Разработать и исследовать модели типовых фракталов в соответствии с номером варианта студента (номером по списку в журналу группы). При этом программы для салфетки Серпинского (по методам «игры в хаос» и СИФ) должны предусматривать ввод с клавиатуры данных о координатах вершин и вероятностях движения к этим вершинам (для «игры в хаос»).

№ варианта	Игра в хаос	Системы итерирующих функций						Дерево	Папоротник	Дополнительно		
		Салфетка Серпинского	Ковер Серпинского	Губка Менгера	Кривая Коха	Снежинка Коха	Двойной дракон			Мандельброта	Жулия	
1	Обязательные для всех		+				+				+	
2				+				+			+	
3						+			+		+	
4							+		+		+	
5				+				+			+	
6					+				+			+
7						+			+			+
8							+					+
9				+				+				+
10					+				+			+
11						+				+	+	
12				+			+				+	
13					+			+			+	
14						+			+		+	
15							+		+		+	
16				+				+				+
17					+				+			+
18						+			+			+
19				+			+					+
20					+			+				+
21						+			+		+	
22							+		+		+	
23				+				+			+	

2 Сравнить стохастические характеристики для «фрактальной пыли», соответствующей странным аттракторам и построить таблицу соответствия по следующей форме:

Система ДУ	Основные параметры	$s$	$k$	$M$	$\Delta t$	Стохастические характеристики

Сравнения производить по системам дифференциальных уравнений, приведенным в задании к лабораторной работе 2

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 **Ловцов, Д.А.** Введение в информационную теорию АСУ / Д.А. Ловцов . – М.: Изд. МО, 1996.
- 2 **Колмогоров, А.Н.** Теория информации и теория алгоритмов / А.Н. Колмогоров. – М.: Наука, 1987.
- 3 **Мазур, М.** Качественная теория информации / М. Мазур. – М.: Мир, 1974.
- 4 **Шилейко, А.В.** Введение в информационную теорию систем / А.В. Шилейко, В.Ф. Кочнев, Ф.Ф. Химушин . – М.: Радио и связь, 1985.
- 5 **Матросов, В.М.** Метод векторных функций Ляпунова: анализ динамических свойств нелинейных систем / В.М. Матросов . – М.: Физматлит, 2001.
- 6 **Дружинин, В.В.** Проблемы системологии / В.В. Дружинин, Д.С. Конторов. – М.: Советское радио, 1976.
- 7 **Храмов, В.В.** Основы информационного подхода к управлению подготовкой специалистов в сфере военного образования / В.В. Храмов. – Пушкино: ОНТИ ПНЦ РАН, 2001.
- 8 **Роджерс, Д.** Алгоритмические основы машинной графики: пер с англ. / Д. Роджерс. – М.: Мир, 1989.
- 9 **Петров, М.Н.** Компьютерная графика: учебник для ВУЗов. / М.Н. Петров, В.П. Молочков. – : СПб.: Питер, 2002.
- 10 **Голубенко, Е.В.** Компьютерная геометрия и графика: учебное пособие / Е.В. Голубенко. – Ростов н/Д: Изд. РГУПС, 2009.

*Учебное издание*

**Соколова Ольга Ивановна**

**ТЕОРИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ**

Учебно-методическое пособие

«Электронный университет» ФГБОУ ВО РГУПС.

---

Адрес университета:  
344038, Ростов н/Д, пл. Ростовского Стрелкового Полка  
Народного Ополчения, 2.