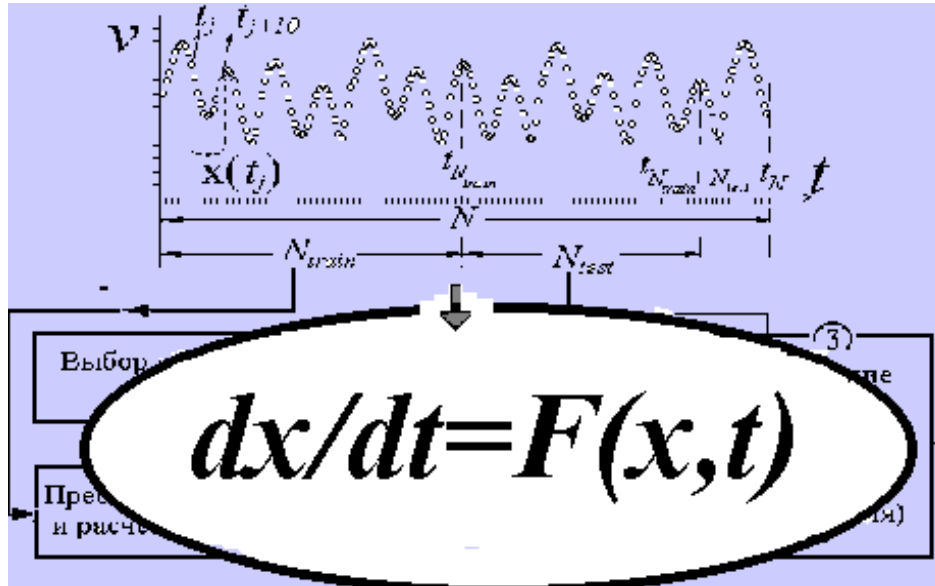


Б.П. БЕЗРУЧКО, Д.А. СМИРНОВ

РЕКОНСТРУКЦИЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ПО ВРЕМЕННЫМ РЯДАМ



Саратов 2000

САРАТОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет нелинейных процессов

Кафедра электроники, колебаний и волн

САРАТОВСКОЕ ОТДЕЛЕНИЕ ИНСТИТУТА РАДИОТЕХНИКИ
И ЭЛЕКТРОНИКИ РАН

Учебно-научная лаборатория

«Нелинейная динамика (физический эксперимент)»

Поддержано ФЦП «Интеграция» и грантом

РФФИ «Ведущие научные школы»

Б.П. БЕЗРУЧКО, Д.А. СМИРНОВ

РЕКОНСТРУКЦИЯ ОБЫКНОВЕННЫХ
ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ
ПО ВРЕМЕННЫМ РЯДАМ

Учебно-методическое пособие

Государственный учебно-научный центр «Колледж»

Саратов, 2000

УДК 530.18

Б 39

Безручко Б.П., Смирнов Д.А.

Б 39 Реконструкция обыкновенных дифференциальных уравнений по временным рядам. Учебно-методическое пособие, – Саратов: Издательство ГосУНЦ “Колледж”, 2000 – 46 с.

Рассматриваются подходы к построению динамических моделей с непрерывным временем по временным рядам. Демонстрируются методы глобальной реконструкции обыкновенных дифференциальных уравнений по векторным и скалярным рядам. Описываются приемы уменьшения влияния шумов. В практической части предлагается, пользуясь готовыми программами, получить временные реализации хаотических колебаний эталонных систем дифференциальных уравнений, реконструировать по ним модельные уравнения и оценить качество реконструкции.

Работы предназначены для практических занятий по курсу «Математическое моделирование» для студентов факультета нелинейных процессов и физического факультета Саратовского госуниверситета.

Рецензент: старший научный сотрудник Саратовского отделения института радиотехники и электроники РАН, к.ф.-м.н. Селезнев Е.П.

© Б.П. Безручко,
Д.А. Смирнов,
2000

© Изд-во ГосУНЦ
«Колледж»,
2000

Содержание

1. Введение.....	4
1.1. Дифференциальные уравнения и моделирование с позиций детерминизма.....	4
1.2. Временные ряды.....	5
1.3. Конструирование, реконструкция, восстановление	6
2. Реконструкция дифференциальных уравнений по временному ряду	7
2.1. Общая схема реконструкции математической модели по временному ряду	7
2.2. Основные элементы процедуры для дифференциальных уравнений, стандартный подход.....	8
2.3. Критерии эффективности модели	12
3. Реконструкция уравнений по векторному ряду (работа № 1)	16
3.1. Процедура получения временного ряда производных, борьба с шумами	16
3.2. Практическое задание	18
3.3. Контрольные вопросы	19
4. Реконструкция уравнений по скалярному ряду (работа № 2)	20
4.1. Метод Брумхеда-Кинга.....	21
4.2. Вычисление нескольких производных наблюдаемой	24
4.3. Практическое задание	25
4.4. Контрольные вопросы	27
Приложение 1. Описание программы FlowSimulator	28
Приложение 2. Описание программы VectorODE.....	31
Приложение 3. Описание программы ScalarODE.....	38
Литература	44

1.1. Дифференциальные уравнения и моделирование с позиций детерминизма

Классическим «материалом» для создания динамических (детерминистических¹) математических моделей являются дифференциальные уравнения. Появление этого математического аппарата в 17-м веке, почти одновременно с дифференциальным и интегральным исчислениями, было вызвано потребностями механики и связано с именами И. Ньютона и Г. Лейбница. Уравнения механики Ньютона (уравнения динамики) являются признанным эталоном «*динамической системы*» — математической модели, позволяющей однозначно предсказать эволюцию объекта по заданному исходному состоянию. Под «состоянием» понимают совокупность значений величин x_j ($j = 1, 2, \dots, D$), характеризующих объект, т.е. это D -мерный вектор \mathbf{x} (x_1, x_2, \dots, x_D), называемый еще *вектором состояния*. Далее мы ограничимся «обыкновенными» дифференциальными уравнениями (ОДУ), содержащими производные лишь по одной независимой переменной:

$$\frac{d\mathbf{x}}{dt} = \mathbf{F}(\mathbf{x}), \quad (1)$$

где \mathbf{x} – зависимая (*динамическая*) переменная, t – независимая переменная. К уравнению первого порядка (1) или эквивалентной ему системе уравнений:

$$\begin{aligned} \dot{x}_1 &= F_1(x_1, x_2, \dots, x_D), \\ \dot{x}_2 &= F_2(x_1, x_2, \dots, x_D), \\ &\dots \\ \dot{x}_D &= F_D(x_1, x_2, \dots, x_D), \end{aligned} \quad (2)$$

заменой переменных можно свести ОДУ, содержащие производные $\frac{d^n x_j}{dt^n}$ любого порядка n .

¹ Детерминизм – мировоззренческое направление, признающее причинную обусловленность, а отсюда - и закономерность, предсказуемость явлений природы. Слово «динамический» в нашем случае используется в смысле «детерминированный», «определенный», «как в уравнениях динамики Ньютона», а не в смысле «подвижный, динамичный» или «силовой».

Одним из достоинств дифференциальных уравнений является возможность их наглядной геометрической интерпретации. Уравнения задают векторное поле – определяют значение *вектора скорости изменения состояния* в точках фазового пространства (пространства состояний). Координатам точек в этом пространстве соответствуют значения x_j , проекции вектора скорости изменения состояния равны \dot{x}_j , а геометрической интерпретацией решения является линия, касательная к векторам скорости и проходящая через начальную точку. При выполнении условий теоремы существования и единственности решения через начальную точку $x(0)$ проходит единственная линия, соответствующая эволюции состояния при изменении независимой переменной. Это означает, что будущее однозначно определено, что является основным требованием детерминированного описания.

1.2. Временные ряды

«Ряды» — дискретные последовательности значений величины, характеризующей объект, измеренные или вычисленные при некоторых значениях независимой переменной. Если независимой переменной является время, ряды называют *временными*². Их непосредственно или после преобразований используют при моделировании в качестве реализаций динамических переменных. В том случае, когда наблюдению и измерению в каждый момент времени t_i доступны несколько величин: x_1, x_2, \dots, x_D , ряд называют *векторным* (так как переменные x_j можно считать компонентами D -мерного вектора x), если исходная величина одна – *скалярным*.

Далее мы будем рассматривать лишь ряды, которые содержат значения наблюдаемой величины v , измеренные через равные интервалы времени Δt (*интервалы выборки*). Члены ряда v_i – значения наблюдаемой величины в дис-

² Хотя описанная в работе техника конструирования уравнений может быть приложена и к моделированию по рядам, соответствующим дискретным значениям любой другой независимой переменной, в частности, координаты.

кретные моменты t_i – будем называть *точками*, i – порядковым номером точки или *дискретным временем*, количество точек в ряде N – *длиной ряда*. Для обозначения самого ряда будем использовать фигурные скобки:

$$\{v_i\}_{i=1}^N = \{v_1, v_2, \dots, v_i, \dots, v_N\}.$$

1.3. Конструирование, реконструкция, восстановление

Наиболее часто при построении модельных дифференциальных уравнений, следуя принципу «от общего к частному», исходят из фундаментальных законов, которым подчиняются объекты рассматриваемой природы, вводя ограничения, соответствующие моделируемому случаю. Например, при решении задач по механике записывают законы сохранения или уравнения Ньютона с силами и допущениями, определяемыми условиями задачи. В таких «асимптотических» моделях обычно ясен физический смысл параметров и переменных; они универсальны и хорошо зарекомендовали себя в различных областях знаний. Однако можно привести много примеров, когда информации об объекте недостаточно или записать базовую систему уравнений не представляется возможным из-за сложности объекта. Например, о ходе процессов на далекой звезде можно судить лишь по зависимости ее светимости от времени. Одним из путей получения математических моделей в таких ситуациях является конструирование уравнений по экспериментальным временным рядам [1-10]. Получающиеся эмпирические модели могут претендовать только на описание наблюдаемого процесса, а не поведения объекта в целом. Если искать аналогии в механике, то эмпирическая модель, полученная из временной реализации движения объекта аналогична «кинематической», которая не несет информации о причинах изменения скорости движения, имеющейся в уравнениях динамики.

Процедуру получения модельных уравнений по временным рядам, которая достаточно часто обсуждается в научной литературе последнего десятилетия, называют *реконструкцией* уравнений. В русскоязычной литературе это слово обычно несет смысл «восстановление» или «реорганизация». При вос-

становлении математических уравнений по временным рядам, полученным при их численном решении, действительно производится реконструкция. Но при моделировании реальных систем не приходится говорить о некоей исходной, истинной модели, поэтому «восстанавливать» нечего и следовало бы пользоваться словами «конструирование» или «создание». Мы будем использовать устойчивый термин, не обращая внимание на отмеченные особенности³; тем более, что в практической части данной работы в качестве исходного ряда используются результаты численного решения некоторого известного дифференциального уравнения или решение со случайной добавкой, моделирующей обязательные для реальных объектов шумы.

2.1. Общая схема реконструкции математической модели по временному ряду

Реконструкция дифференциальных уравнений проводится по схеме, с которой вы уже сталкивались при моделировании отображениями (рис.1). На первом этапе (1) на основании априорных данных, по оценке размерности множе-

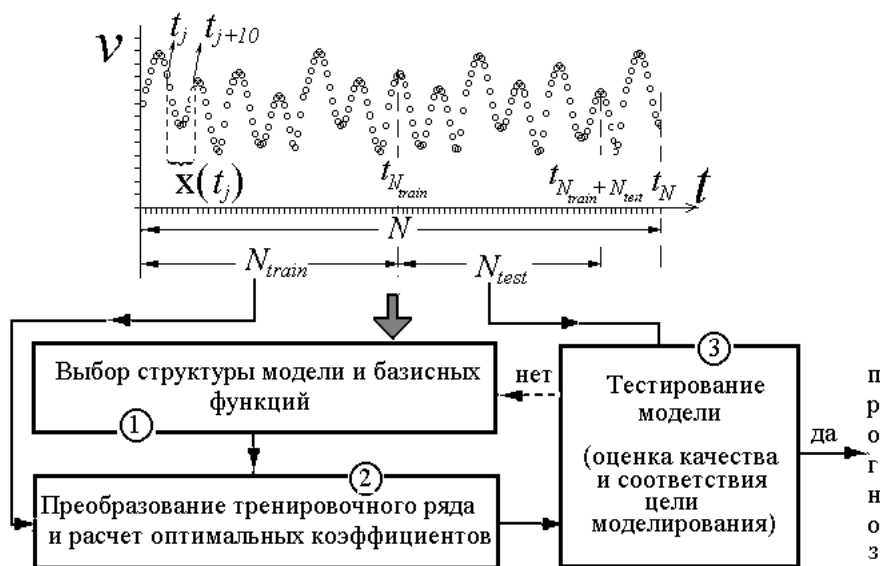


Рис.1. Схема процесса моделирования по временному ряду.

ства восстановленного в фазовом пространстве по временному ряду $\{v_i\}$ или интуитивно выбирается размерность D модели, а также вид функций, с помощью которых будет осуществляться аппроксимация. На следующем этапе (2) тренировочная часть временного ряда преобразуется в соответствии с выбран-

³ Смена сложившейся терминологии, даже если она не очень удачна, оправдана лишь в слу-

ной структурой модели: ослабляются шумы, с необходимой частотой выбираются точки, восстанавливаются дополнительные переменные и т.п. Далее, искомую функцию F представляют с помощью выбранных на первом этапе *базисных* функций с неизвестными коэффициентами и находят значения коэффициентов, которые обеспечивают наилучшее соответствие модели и экспериментальных данных. На третьем этапе результаты предсказания с помощью полученной модели сравниваются с данными тестовой части ряда. При этом оценивают качество получившейся модели, для чего используются критерии, определяемые целью моделирования.

Специфика восстановления дифференциальных уравнений в основном проявляется на втором этапе, при обработке исходного ряда. Так как операции численного дифференцирования критичны к выбору величины временного интервала между дискретными точками и приводят к усилению влияния высокочастотных шумов, здесь требуется проводить ряд специальных «мероприятий». На них мы обратим в данной работе особое внимание. Что касается выбора структуры модели, то здесь мы воспользуемся стандартной аппроксимацией зависимостей алгебраическими полиномами. Учитывая, что в практической части работы в качестве объекта моделирования используются эталонные уравнения, качество реконструкции будем оценивать, сравнивая полученную модель непосредственно с объектом.

2.2. Основные элементы процедуры для дифференциальных уравнений, стандартный подход

Оставим пока трудные вопросы, встречающиеся при моделировании конкретных объектов, тем более реальных. Расширим представления о моделировании по рядам на анализе наиболее *простой ситуации*: имеется векторный временной ряд $\{\mathbf{x}_i\}_{i=1}^N$, т.е. в каждый момент времени t_i наблюдению доступны D величин $x_1(t_i), \dots, x_D(t_i)$, однозначно определяющих состояние объекта; вре-

чаях большого выигрыша в решении поставленных задач, а это – не тот случай.

менные реализации достаточно гладкие, а интервал Δt , с которым они оцифровываются, мал по сравнению с характерными масштабами их изменения. То есть имеются все основания искать модель в виде (2).

Для построения модели используем временной ряд из первых N_{train} значений каждой из наблюдаемых (назовем его «тренировочным»). Продолжение тренировочного временного ряда («тестовый» ряд длиной N_{test}) используем *позже* для проверки эффективности построенной модели (рис.1).

В соответствии со схемой рис.1 процедура построения модели сводится к тому, чтобы

- по временному ряду $\{x_i\}$ рассчитать временной ряд $\left\{ \frac{dx(t_i)}{dt} \right\}$,
- подобрать функции F_j , аппроксимирующие зависимости dx_j/dt от x ,
- проверить эффективность построенной модели.

Первую задачу решают путем численного дифференцирования. В рассматриваемом случае можно использовать любой стандартный метод, при наличии шумов требуются специальные методы (см. разделы 3, 4)⁴. *Стандартным* путем решения второй задачи является представление функций F_j полиномами порядков K_j :

$$F_j(x_1, x_2, \dots, x_D) = \sum_{l_1, l_2, \dots, l_D=0}^{K_j} c_{j, l_1, l_2, \dots, l_D} \prod_{k=1}^D x_k^{l_k}, \sum_{k=1}^D l_k \leq K_j. \quad (3)$$

Затем вычисляют значения коэффициентов, которые обеспечивают наилучшее соответствие модели и экспериментальных данных (*подгонка* модели к временному ряду). Обычно для этого используют метод наименьших квадратов, т.е. вычисляются такие значения коэффициентов $c_{j, l_1, l_2, \dots, l_D}$, которые минимизируют среднеквадратичные ошибки аппроксимации ε_j ($j = 1, \dots, D$):

⁴ Отметим, что временной ряд производных содержит меньше значений, чем исходный ряд, т.к. для расчета производной в данный момент времени, используется m соседних точек. Та-

$$\varepsilon_j^2 = \frac{1}{N_{train} - m + 1} \sum_i \left[\frac{dx_j(t_i)}{dt} - F_j(x_1(t_i), x_2(t_i), \dots, x_D(t_i)) \right]^2 = \min. \quad (4)$$

В более трудном случае, когда наблюдению доступен лишь скалярный временной ряд $\{v_i\}_{i=1}^N$, процедура построения модели (2) включает в себя еще один предварительный этап (по сравнению с моделированием по векторному ряду). Необходимо выяснить, как изменялся во времени вектор состояния системы, т.е. по скалярному временному ряду $\{v_i\}$ построить («восстановить») временной ряд векторов состояния $\{\mathbf{x}_i\}$. Эту процедуру называют *реконструкцией фазовой траектории* [11,12]. Один из часто используемых методов — метод последовательного дифференцирования, согласно которому координатами вектора состояния в момент времени t_i являются последовательные временные производные наблюдаемой в этот момент:

$$\begin{aligned} x_1(t_i) &= v(t_i), \\ x_2(t_i) &= \frac{dv(t_i)}{dt}, \\ &\dots \\ x_D(t_i) &= \frac{d^{D-1}v(t_i)}{dt^{D-1}}. \end{aligned} \quad (5)$$

Выбор размерности модели D можно осуществить на основе предварительного анализа временного ряда⁵. Модель имеет вид

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_3, \\ &\dots \\ \dot{x}_D &= F_D(\mathbf{x}), \end{aligned} \quad (6)$$

где $x_1 = v$, а F_D — искомая функция, аппроксимирующая зависимость $\dot{x}_D(\mathbf{x})$ (отметим, что в данном случае нужно аппроксимировать только одну зависи-

ким образом, если исходный ряд содержал N_{train} значений, то ряд производных будет содержать $N_{train} - m + 1$ значений (см. ниже, раздел 3.1).

⁵ Ниже (в разделе 4.1) будет рассмотрен метод Брумхеда-Кинга [13], однако используются и другие методы: оценка корреляционной размерности методом Грассбергера-Прокаччия (см., например, [10]), метод ложных ближайших соседей [14].

мость), она может быть представлена, например, полиномом (3) некоторого порядка K , значения коэффициентов которого также определяются методом наименьших квадратов.

Уравнения (6) представляют собой частный случай системы $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})$, но какова степень их общности? Приведем сначала несколько примеров систем такого «стандартного» вида. В качестве первого примера можно указать осциллятор Ван-дер-Поля:

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= \lambda(1 - x^2)y.\end{aligned}\tag{7}$$

Уравнения (7) являются частным случаем уравнений (6) если $D = 2$, функция F — полином порядка 3 и в качестве наблюдаемой v выбрана координата x .

К виду (6) можно привести и уравнения системы Ресслера:

$$\begin{aligned}\dot{x} &= -y - z, \\ \dot{y} &= x + a \cdot y, \\ \dot{z} &= b - cz + xz.\end{aligned}\tag{8}$$

Если перейти к другим переменным — последовательным производным координаты y , то уравнения (8) можно переписать [7] в виде:

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_3, \\ \dot{x}_3 &= -b - cx_1 + (ac - 1)x_2 + (a - c)x_3 - ax_1^2 + (a^2 + 1)x_1x_2 - ax_1x_3 - ax_2^2 + x_2x_3,\end{aligned}\tag{9}$$

где $x_1 = y$. Уравнения (9) — частный случай уравнений (6), если $D = 3$, а функция F — полином порядка 2.

Для последовательных производных координаты x уравнения Ресслера переписываются как

$$\begin{aligned}\dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_3, \\ \dot{x}_3 &= ab - cx_1 + (ac - 1)x_2 + (a - c)x_3 + x_1^2 - ax_1x_2 + x_1x_3 - \\ &\quad - \frac{x_2(x_1 + b - ax_2 + x_3)}{a + c - x_1},\end{aligned}\tag{10}$$

где $x_1 = x$. Уравнения (10) — частный случай уравнений (6), если $D = 3$, а функция F — дробно-рациональная функция с полиномом порядка 3 в числителе и полиномом порядка 1 в знаменателе (это легко увидеть, если правую часть последнего уравнения (10) привести к общему знаменателю).

Для производных переменной z системы Ресслера также можно записать уравнения (8) в виде, аналогичном (9) и (10) с $D = 3$ и дробно-рациональной функцией в правой части.

Таким образом, уравнения (6) обладают достаточно большой степенью общности. К такому виду можно привести любую систему уравнений $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})$, но в общем случае размерность D системы (6) *должна быть больше*, чем размерность d исходной системы (согласно теореме Такенса достаточно потребовать $D \geq 2d + 1$, но иногда это оказывается слишком строгим требованием)⁶.

2.3. Критерии эффективности модели

После того, как модель построена (значения всех коэффициентов вычислены), необходимо проверить ее работоспособность. Как уже говорилось, критерии эффективности модели определяются целями моделирования. Далее мы обсудим несколько возможных критериев (на примере модели (2),(3), построенной по векторному ряду⁷):

- погрешности аппроксимации ε_j ($j = 1, \dots, D$),
- погрешности аппроксимации $\varepsilon_{test,j}$, рассчитанные по тестовому ряду,⁸
- дальность прогноза, обеспечиваемого моделью,
- качественное соответствие поведения модели и объекта;

⁶ Последнее утверждение очень существенно, так как из него следует, что исходная система и система (6) могут быть не эквивалентны (при $D > d$). Их поведение идентично только в том случае, когда начальные условия системы (6), заданные в пространстве \mathbb{R}^D , соответствуют каким-либо начальным условиям исходной системы (т.е. принадлежат некоторому d -мерному многообразию).

⁷ Отличия в их использовании для модели (6) не существенны и будут указываться в сносках.

⁸ Для модели (6) есть только величины ε_D и $\varepsilon_{test,D}$.

- в некоторых примерах (где это возможно) будем также сравнивать полученные значения коэффициентов с их истинными значениями.

Обсудим подробнее каждый из перечисленных критериев.

1) Величина погрешности аппроксимации по исходному, тренировочному ряду ε_j может служить показателем качества модели. Удобно использовать нормированную величину ε_j/S_j (где

$$S_j = \sqrt{\frac{1}{N_{train}} \sum_{i=1}^{N_{train}} (\dot{x}_j(t_i) - \bar{\dot{x}}_j)^2}$$

— стандартное отклонение производной \dot{x}_j от ее среднего значения) и выразить ее в процентах. Если нормированная погрешность аппроксимации превышает, скажем, 50 %, то трудно рассчитывать на эффективность модели.

2) Казалось бы, чем меньше величины ε_j , тем лучше модель. Однако обратим внимание, что ε_j может обратиться и в ноль, если количество коэффициентов модели очень велико (равно $N_{train} - m + 1$). Но такая громоздкая модель наверняка не будет сколько-нибудь эффективной для описания исходной системы. Используя значения коэффициентов c_{j,l_1,l_2,\dots,l_D} , рассчитанные по тренировочному ряду, можно вычислить погрешность аппроксимации зависимости dx_j/dt от \mathbf{x} ($\varepsilon_{test,j}$) по тестовому временному ряду:

$$\varepsilon_{test,j}^2 = \frac{1}{N_{test}} \sum_{i=N_{train}+1}^{N_{train}+N_{test}} [\dot{x}_j(t_i) - F_j(\mathbf{x}(t_i))]^2. \quad (11)$$

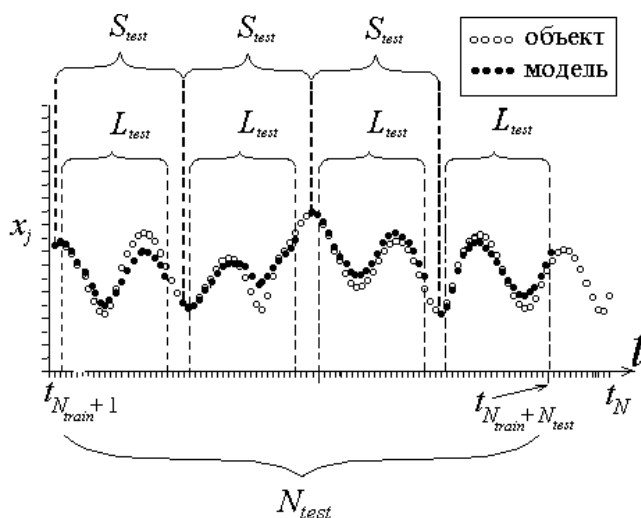


Рис.2. Иллюстрация к вычислению дальности прогноза по тестовому временному ряду. Здесь светлыми кружками показана временная реализация объекта (одной из наблюдаемых величин), черными — прогноз для отдельных участков. В данном случае длина одного тестового участка $L_{test} = 20$, общее количество тестовых участков $K_{test} = 4$, сдвиг между соседними участками $S_{test} = 23$, длина тестового ряда $N_{test} = 90$.

Это более надежная характеристика качества модели, т.к. тестовый ряд не используется для подгонки значений коэффициентов.

3) Для расчета дальности прогноза используем следующий подход [3]. Рассчитаем среднеквадратичную ошибку прогноза, обеспечиваемого моделью, на определенное число выборочных интервалов вперед. Причем для проверки будем использовать различные участки тестового временного ряда.

Зададим начальные условия модели, используя первые значения наблюдаемых из тестового ряда⁹: $x_1 = x_1(t_{N_{train}+1})$, $x_2 = x_2(t_{N_{train}+1})$, ..., $x_D = x_D(t_{N_{train}+1})$. Численно интегрируя модельные уравнения, получим временную реализацию модели длиной L_{test} интервалов выборки. Эта реализация содержит *предсказанные* значения наблюдаемых в моменты времени $t_{N_{train}+2}$, ..., $t_{N_{train}+1+L_{test}}$. Сравним ее с соответствующим участком тестового ряда (рис.2). Обозначим $\hat{x}_j(t_{N_{train}+1}, T)$ — предсказанное с помощью модели значение наблюдаемой x_j на T интервалов выборки вперед (т.е. *прогноз с упреждением T*), начиная с момента времени $t_{N_{train}+1}$. Квадрат ошибок прогноза с упреждением T ($1 \leq T \leq L_{test}$) определяется выражением

$$\sigma_j^2(t_{N_{train}+1}, T) = (\hat{x}_j(t_{N_{train}+1}, T) - x_j(t_{N_{train}+1+T}))^2, j = 1, \dots, D.$$

Определенная таким образом ошибка прогноза зависит от выбранного начального момента времени, с которого мы начинаем прогнозировать процесс.

Выберем теперь другие начальные условия, сместившись по тестовому ряду на S_{test} интервалов Δt (см. рис.2): $x_1 = x_1(t_{N_{train}+1+S_{test}})$, ..., $x_D = x_D(t_{N_{train}+1+S_{test}})$.

Вновь рассчитаем квадрат ошибки прогноза:

$$\sigma_j^2(t_{N_{train}+1+S_{test}}, T) = (\hat{x}_j(t_{N_{train}+1+S_{test}}, T) - x_j(t_{N_{train}+1+S_{test}+T}))^2.$$

И так далее рассчитаем ошибки прогноза σ_j^2 для K_{test} различных начальных условий (различных участков тестового ряда, каждый из которых смещен

⁹ Для задания начальных условий модели (6) нужно взять первые m значений наблюдаемой y и рассчитать по ним $D-1$ временных производных.

на S_{test} шагов относительно предыдущего)¹⁰. В итоге рассчитаем нормированную *среднеквадратичную* ошибку прогноза в зависимости от упреждения T :

$$\sigma_j(T) = \sqrt{\frac{1}{K_{test}} \sum_{i=1}^{K_{test}} \sigma_j^2(t_{N_{train}+1+S_{test}(i-1)}, T)} / S_j^2}, \quad (12)$$

Дальность прогноза $\tau_{pred,j}$ наблюдаемой x_j определяется как время возрастания величины $\sigma_j(T)$ от нуля (при $T = 0$) до определенного критического значения σ_c : $\sigma_j(\tau_{pred,j}) = \sigma_c$. Это критическое значение определяется тем, какая погрешность прогноза допустима. Мы примем значение $\sigma_c = 0.05$ (т.е. допустима погрешность 5 % от общего размаха колебаний).

Обращаем ваше внимание на то, что время прогноза хаотических процессов даже с использованием «идеальной» модели принципиально ограничено и это надо учитывать при оценке качества модели. Для оценки предельно возможной дальности прогноза можно использовать выражение

$$\tau_{pred,j} = \frac{1}{2\lambda_+} \ln \frac{\sigma_c^2 \cdot S_j^2}{\sigma_v^2 + \sigma_\mu^2 + \sigma_{\Delta M}^2}, \quad (13)$$

где λ_+ — старший ляпуновский показатель, σ_v^2 определяет флуктуации в системе, σ_μ^2 — погрешности измерительных приборов, $\sigma_{\Delta M}^2$ — погрешность модели («шумы незнания») [15].

4) Для качественного сравнения режимов поведения модели и объекта нужно задать начальные условия для модели (из тестового ряда) и получить ее фазовую траекторию. Затем сравнить проекции фазовых портретов объекта и модели на плоскость (x_1, x_2) .

5) Если реконструируется известная математическая конструкция, критерием качества модели может служить относительная погрешность определения значения некоторого коэффициента c_i , которая определяется как

¹⁰ Таким образом, общая длина ряда, используемого для тестирования модели, составляет $N_{test} = L_{test} + 1 + (K_{test} - 1) \cdot S_{test}$ значений.

$$\varepsilon_{coeff}^{(i)} = \frac{|c_i - c_i^{(0)}|}{|c_i^{(0)}|} \quad (14)$$

в случае $c_i^{(0)} \neq 0$, где $c_i^{(0)}$ — истинное значение данного коэффициента. Если $c_i^{(0)} = 0$, то ошибку восстановления значения коэффициента определяют как

$$\varepsilon_{coeff}^{(i)} = |c_i|. \quad (15)$$

Общую ошибку восстановления значений коэффициентов по временному ряду ε_{coeff} можно определить, например, как среднее значение индивидуальных ошибок или как максимальную из всех индивидуальных ошибок:

$$\varepsilon_{coeff} = \max_{i=1, \dots, M} \{ \varepsilon_{coeff}^{(i)} \} \quad (16)$$

Приведенный перечень критериев не является исчерпывающим. Заметим, что введенные выше обозначения используются далее в описаниях программ для конструирования модельных уравнений (приложения).

3.1. Процедура получения временного ряда производных, борьба с шумами

В отсутствие шумов для численного дифференцирования исходного ряда можно использовать, например, формулу

$$\dot{x}_1(t_i) = \frac{x_1(t_{i+1}) - x_1(t_i)}{\Delta t}. \quad (17)$$

Погрешность вычисления производной для данной схемы составляет $O(\Delta t)$.¹¹

Точность можно увеличить, если от (17) перейти к схеме

$$\dot{x}_1(t_i) = \frac{x_1(t_{i+1}) - x_1(t_{i-1}))}{2 \cdot \Delta t}, \quad (18)$$

для нее погрешность составляет $O(\Delta t^2)$. Однако в обоих упомянутых случаях предполагается, что шумы в системе отсутствуют.

¹¹ Т.е. погрешность по порядку величины равна Δt при $\Delta t \rightarrow 0$.

При наличии шумов погрешности дифференцирования с помощью формул (16) или (17) будут очень велики (см. рис.3а-в). Поэтому на практике используют другие методы расчета производных, которые связаны с усреднением шума.

В данной работе будет использован следующий подход. Для расчета производной в момент времени t_i используется m значений наблюдаемой величины в окрестности t_i . Предпочтительнее использовать нечетное значение m , так как тогда используется равное количество значений слева и справа от рассматриваемой точки: $x_1(t_{i-(m-1)/2}), \dots, x_1(t_{i-1}), x_1(t_i), x_1(t_{i+1}), \dots, x_1(t_{i+(m-1)/2})$ (центрированная схема). Этот «короткий ряд» из m значений (наблюдаемая реализация в окрестности точки t_i) на основании разложения в ряд Тейлора аппроксимируется функцией (рис.4, с. 47)

$$x_1(t) = a_0 + a_1(t - t_i). \quad (19)$$

Коэффициенты a_0 и a_1 рассчитываются методом наименьших квадратов (должно быть $m \geq 2$). Значение производной определяется как $\dot{x}_1(t_i) = a_1$. Чем выше уровень шума, тем большее значение m следует выбирать. Однако оно не должно быть слишком большим, иначе не будет эффективной линейная аппроксимация (19). Отметим, что точность вычисления производной зависит также от такого параметра, как количество измеренных значений на характерном периоде колебаний. Опыт показывает, что если на один период приходится менее 30-40 точек, то точность вычисления производной не велика; желательно, чтобы было не менее 100 точек на периоде.

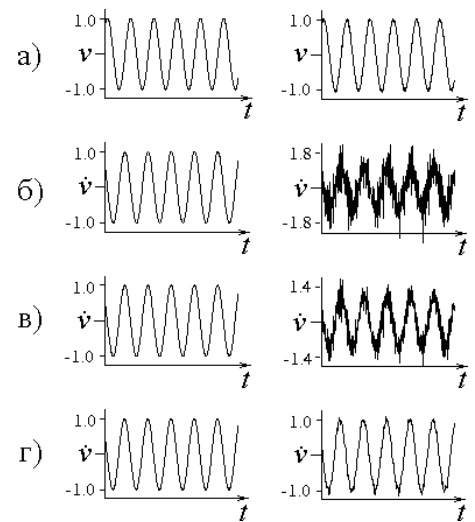


Рис.3. В левом столбце (а) представлен график наблюдаемого сигнала $v(t) = \sin(t)$, справа показан тот же сигнал с добавленным к нему нормальным белым шумом, стандартное отклонение которого составляет примерно 3 % от стандартного отклонения сигнала (отношение сигнал/шум примерно 30 дБ). Ниже (б,в,г) представлены графики первых производных этих сигналов, вычисленных с использованием формул (17), (18) и (19) при $m = 9$.

3.2. Практическое задание

Цель работы. Применить на практике изложенный алгоритм:

- потренироваться в выборе оптимальных значений количества точек для численного дифференцирования и порядков полиномов,
- убедиться в трудности численного дифференцирования при наличии шумов,
- отследить зависимость эффективности восстановленной модели от количества данных,
- выяснить зависимость необходимого количества данных от уровня шума.

Задача. Восстановить дифференциальные уравнения с полиномиальными нелинейностями по хаотической временной реализации эталонной динамической системы, заданной преподавателем (например, системы Ресслера).

- 1) С помощью программы **FlowSimulator** (приложение 1) получите и запишите векторный временной ряд, содержащий 10000 векторов: не зашумленные хаотические реализации эталонной системы, заданной преподавателем. Например, для системы Ресслера при $a = 0.398$, $b = 2.0$, $c = 4.0$, шаг интегрирования 0.01, интервал выборки 0.1 (при этих параметрах временной ряд содержит примерно 60 точек на характерном периоде колебаний).
- 2) Восстановите дифференциальные уравнения с помощью программы **VectorODE** (приложение 2). Используйте тренировочный векторный временной ряд длиной $N_{train} = 600$ векторов. Примите «верное» численное значение интервала выборки (для указанного примера $\Delta t = 0.1$). Используйте для дифференцирования минимальное количество точек $m = 2$. Постройте модель, выбирая для каждого уравнения «верные» значения порядка полинома (например, для системы Ресслера: $K_1 = K_2 = 1, K_3 = 2$).
- 3) Теперь постройте модель, используя для дифференцирования $m = 3$ (центри-

рованная схема). Убедитесь, что это позволяет более точно восстановить значения коэффициентов по сравнению с $m = 2$.

- 4) При фиксированном $N_{train} = 600$ постройте график зависимости дальности прогноза τ_{pred} и погрешности аппроксимации ε для какой-либо переменной от порядка одного из полиномов (например, $\tau_{pred,3}(K_3)$ и $\varepsilon_3(K_3)$ при $K_1 = K_2 = 1$). Определите по этим графикам оптимальное значение порядка.
- 5) Запишите хаотическую временную реализацию той же системы (не зашумленную) длиной 30000 векторов (шаг интегрирования 0.01 и интервал выборки 0.01 — примерно 600 точек на характерном периоде колебаний).
- 6) Используйте для построения модели $N_{train} = 6000$ векторов, $m = 3$ и $K_1 = K_2 = 1, K_3 = 2$. Сравните достигнутую дальность прогноза τ_{pred} и погрешность восстановления коэффициентов ε_{coeff} с полученными в п.4.
- 7) Запишите временной ряд п.5 с добавленными шумами (для отношения амплитуды сигнала к стандартному отклонению шума равного $10^4, 10^3$).
- 8) По зашумленным рядам восстановите уравнения при фиксированных оптимальных значениях порядков полиномов и различных значениях N_{train} (для $N_{train} = 100, 300, 600, 1500, 3000, 6000, 10000$). Помните, что для эффективного дифференцирования нужно подобрать оптимальное значение m . Постройте графики зависимости $\varepsilon(N_{train})$, $\tau_{pred}(N_{train})$ и $\varepsilon_{coeff}(N_{train})$ для какой-либо переменной. Как меняется величина τ_{pred} с ростом уровня шума в системе?

3.3. Контрольные вопросы

1. Что такое временной ряд? скалярный и векторный ряды?
2. Изложите принцип конструирования модельных дифференциальных уравнений по векторному временному ряду. Из каких основных этапов состоит процедура построения модели?

3. Какие критерии эффективности модели могут использоваться?
4. Как рассчитывается дальность прогноза, обеспечиваемого моделью?
5. Какие методы численного дифференцирования существуют?
6. Какие меры предпринимаются для «борьбы» с шумами при численном дифференцировании?
7. Почему для проверки эффективности модели не следует использовать тренировочный временной ряд?

Пусть единственной информацией об объекте является скалярный временной ряд $\{v_i\}_{i=1}^N$. Для реконструкции недостающих координат вектора состояния используется метод последовательного дифференцирования (5), а модель строится в виде (6). Обратим внимание, что первые $D - 1$ уравнений в (6) имеют заданный простейший вид, но за это надо «платить» необходимостью вычисления по исходному ряду производных высокого порядка – до $\dot{x}_D = \frac{d^D v}{dt^D}$

При наличии шума это может оказаться весьма непростой задачей, так как требует принять меры для уменьшения влияния шума (см. ниже).

Рассмотрим метод вычисления нескольких производных наблюдаемой v при наличии шума [9], который включает в себя метод Брумхеда-Кинга для оценки размерности движения [13].

4.1. Метод Брумхеда-Кинга

Идея метода Брумхеда-Кинга состоит в следующем. Сначала по скалярному ряду восстанавливаются методом задержек векторы $\mathbf{w}(t_j)$ в пространстве вложения достаточно большой размерности (на рис.5 для наглядности представлен случай гармонических колебаний при отсутствии шума и при размерности вложения $m = 3$). Затем в этом пространстве производится преобразование поворота, причем направления новых осей координат (на рис. 5б, в это

$\{s_1, s_2, s_3\}$) выбираются в соответствии с теми направлениями, вдоль которых наиболее интенсивно развивается движение. Оценкой сверху для размерности служит количество направлений, вдоль которых имеется существенная протяженность траектории (на рис.5в движение развито только вдоль направлений s_1 и s_2). Составляющие векторов вдоль тех направлений, которые слабо «осваиваются траекторией» (на рис.5в это направление s_3) списываются на влияние шума и ими пренебрегают.

Для вычисления нескольких производных наблюдаемой v в момент времени t_j используются несколько первых (главных) компонент вектора $w(t_j)$ в системе координат $\{s_i\}$. Далее будет более подробно изложен данный подход.

1) По исходной реализации восстанавливаются векторы в пространстве достаточно большой размерности m методом временных задержек. Время задержки полагают равным интервалу выборки Δt . Получают временной ряд m -мерных векторов $\{w(t_i)\}$ (рис.5а,б). Восстановленные векторы выражаются при нечетном m как

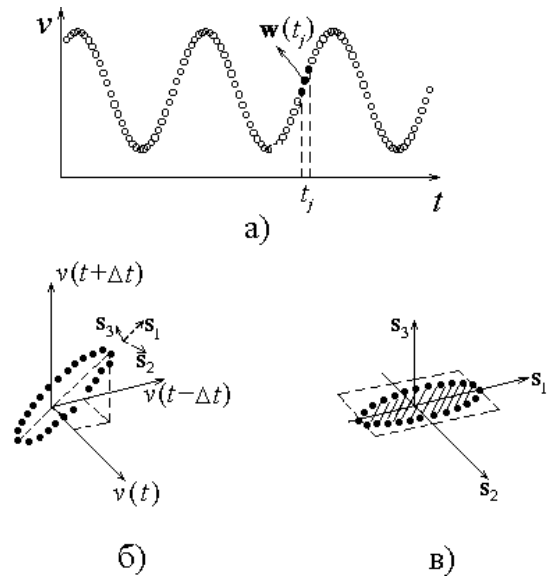


Рис.5. Гармонические колебания при отсутствии шума: а) восстановление векторов w методом задержек по скалярному ряду ($m = 3$), б) восстановленная фазовая траектория представляет собой плоский эллипс, вытянутый вдоль главной диагонали пространства R^m , в) положение фазовой траектории в новой системе координат — после преобразования поворота (составляющая векторов вдоль направления s_3 равна нулю).

$$\mathbf{w}(t_i) = \begin{pmatrix} v(t_{i-(m-1)/2}) \\ \dots \\ v(t_{i-1}) \\ v(t_i) \\ v(t_{i+1}) \\ \dots \\ v(t_{i+(m-1)/2}) \end{pmatrix}, \quad (20)$$

а при четном m :

$$\mathbf{w}(t_i) = \begin{pmatrix} v(t_{i-m/2+1}) \\ \dots \\ v(t_{i-1}) \\ v(t_i) \\ v(t_{i+1}) \\ \dots \\ v(t_{i+m/2}) \end{pmatrix}. \quad (21)$$

2) Оценкой сверху для размерности наблюдаемого движения может служить размерность подпространства, которому принадлежит полученная траектория. Для того чтобы найти размерность этого подпространства, определяется максимальное количество линейно независимых векторов траектории $\{\mathbf{w}(t_i)\}$. Можно показать, что при отсутствии шума это число равно рангу матрицы ковариаций Θ наблюдаемого процесса:

$$\Theta = \begin{pmatrix} \sum_{i=1}^{N_{train}-m+1} v_i v_i & \sum_{i=1}^{N_{train}-m+1} v_i v_{i+1} & \dots & \sum_{i=1}^{N_{train}-m+1} v_i v_{i+m-1} \\ \sum_{i=1}^{N_{train}-m+1} v_{i+1} v_i & \sum_{i=1}^{N_{train}-m+1} v_{i+1} v_{i+1} & \dots & \sum_{i=1}^{N_{train}-m+1} v_{i+1} v_{i+m-1} \\ \dots & \dots & \dots & \dots \\ \sum_{i=1}^{N_{train}-m+1} v_{i+m-1} v_i & \sum_{i=1}^{N_{train}-m+1} v_{i+m-1} v_{i+1} & \dots & \sum_{i=1}^{N_{train}-m+1} v_{i+m-1} v_{i+m-1} \end{pmatrix}. \quad (22)$$

3) Матрица Θ (размерности $m \times m$) — симметричная, вещественная, положительно определенная. Следовательно, ее собственные векторы образуют полный ортонормированный базис пространства \mathbb{R}^m , а ее собственные значе-

ния являются неотрицательными величинами. Обозначим $\sigma_1^2, \sigma_2^2, \dots, \sigma_m^2$ — собственные значения Θ (в порядке убывания), $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_m$ — соответствующие собственные векторы:

$$\mathbf{s}_i = \begin{pmatrix} \mu_{i1} \\ \mu_{i2} \\ \dots \\ \mu_{im} \end{pmatrix}. \quad (23)$$

При переходе к базису $\{\mathbf{s}_i\}$ матрица Θ примет диагональный вид:

$$\Theta = \begin{pmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \sigma_m^2 \end{pmatrix}. \quad (24)$$

Диагональные элементы матрицы Θ в новом базисе σ_i^2 — это средние квадраты проекций восстановленной фазовой траектории на координатные оси $\{\mathbf{s}_i\}$. Они определяют протяженность траектории вдоль соответствующего направления. Ранг матрицы Θ равен числу ненулевых собственных значений (для ситуации, представленной на рис.5, ненулевыми будут только σ_1^2 и σ_2^2) и равен размерности подпространства, в котором содержится траектория.

4) При наличии шума все σ_i^2 отличны от нуля, так как в направлениях, которые не “осваиваются” детерминированной составляющей траектории, представлена шумовая компонента. В этом случае размерность движения оценивают

как число \tilde{D} существенных собственных значений матрицы Θ (иллюстрацию для произвольного случая см. на рис.6). Проекции вектора $\mathbf{w}(t_i)$ на соответ-

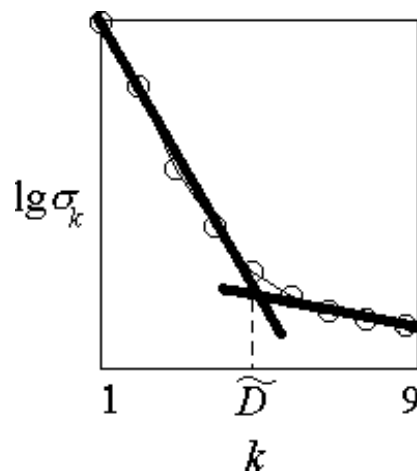


Рис.6. Возможный качественный характер зависимости значений собственных чисел матрицы ковариаций (расположенных в порядке убывания) от их номера. «Точка излома» \tilde{D} на графике служит оценкой размерности наблюдаемого движения.

вующие направления называют его главными компонентами. Остальные собственные значения составляют так называемый “шумовой пьедестал”.

5) Задают размерность модели D (если нет других соображений, то можно, например, положить ее равной \tilde{D} — числу главных компонент). Производят переход от стандартного ортонормированного базиса пространства \mathbf{R}^m к базису $\{\mathbf{s}_i\}$ (преобразование поворота, см. рис.5б,в) и ограничиваются только первыми $D+1$ координатами. Получают новый векторный временной ряд $\{\mathbf{z}(t_i)\}$ из $(D+1)$ -мерных векторов:

$$\mathbf{z}_k(t_i) = \mathbf{w}(t_i)\mathbf{s}_k, k = 1, \dots, D + 1. \quad (25)$$

Описанная процедура позволяет получить более гладкую траекторию. Выражение (25) аналогично операции усреднения с некоторыми весами¹².

4.2. Вычисление нескольких производных наблюдаемой

Для вычисления всех D производных в момент времени t_i используют разложение наблюдаемой временной зависимости в ряд Тейлора (при условии достаточной малости выборочного интервала Δt):

$$v(t_i + j\Delta t) = \sum_{k=0}^D \frac{(j\Delta t)^k}{k!} \frac{d^k v(t_i)}{dt^k}, \quad (26)$$

где отброшены члены порядка $(j\Delta t)^{D+1}$ и выше (см. рис.7). Подставляя (26) в (25) с использованием (20) или (21), получают линейную связь между компонентами вектора $\mathbf{z}(t_i)$ и компонентами вектора $\mathbf{x}(t_i)$ (вектора последовательных производных наблюдаемой величины v):

$$\mathbf{z}_k(t_i) = \sum_{j=1}^{D+1} \mathbf{u}_{kj} \frac{d^{j-1} v(t_i)}{dt^{j-1}}, k = 1, \dots, D + 1, \quad (27)$$

где u_{kj} — элементы некоторой матрицы \mathbf{U} , равные (при нечетном m)

¹² Такое преобразование известно в теории информации как преобразование Карунена-Лозва [14].

$$u_{kj} = \sum_{l=1}^m \mu_{kl} \frac{((l-1-(m-1)/2)\Delta t)^{j-1}}{(j-1)!}. \quad (28)$$

Можно записать обратное к (28) соотношение, и таким образом перейти от ряда $\{\mathbf{z}(t_i)\}$ к ряду из векторов последовательных производных $\{\mathbf{x}(t_i)\}$:

$$\mathbf{x}(t_i) = \mathbf{U}^{-1} \mathbf{z}(t_i), \quad (29)$$

где \mathbf{U}^{-1} — матрица, обратная к \mathbf{U} .

Варьируя первоначальную размерность вложения m , тем самым меняют количество точек в окрестности t_i , используемых для вычисления производных, что позволяет существенно снизить влияние шумов.

Теперь построение модели сводится к вычислению коэффициентов функции F_D методом наименьших квадратов (подгонке). Для проверки эффективности модели используются критерии, изложенные в разделе 2.3.

4.3. Практическое задание

Цель работы. Применить изложенный алгоритм:

- потренироваться в выборе оптимальных значений параметров алгоритма построения модели,
- познакомиться с методом Брумхеда-Кинга для оценки размерности,
- убедиться в трудности вычисления нескольких производных при наличии шума даже малой интенсивности,
- отследить зависимость качества модели от длины тренировочного временного ряда,
- отследить зависимость необходимой длины тренировочного ряда от уровня шума.

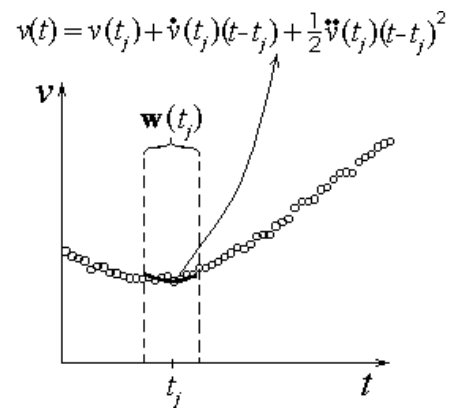


Рис.7. Иллюстрация к вычислению сразу двух производных наблюдаемой при наличии шума. Для расчета производных в момент времени t_j используется три главных компоненты вектора $\mathbf{w}(t_j)$, а для восстановления этого вектора нужны m соседних значений величины v (в данном случае $m = 9$).

Задача. Применить описанный алгоритм для реконструкции дифференциальных уравнений по временным реализациям¹³

- координаты y системы Ресслера,
- координаты x системы Ресслера (здесь можно использовать при восстановлении уравнений дробно-рациональную функцию, чтобы сравнить коэффициенты восстановленных и исходных уравнений, а можно выбрать и полином, который будет аппроксимацией этой функции).

1. С помощью программы **FlowSimulator** (приложение 1) получите и запишите хаотическую скалярную временную реализацию координаты y системы Ресслера (не зашумленную) длиной 30000 значений при значениях параметров, заданных преподавателем (например, $a = 0.398$, $b = 2.0$, $c = 4.0$, шаг интегрирования 0.01, интервал выборки 0.01).
2. С помощью программы **ScalarODE** (приложение 3) реконструируйте дифференциальные уравнения. Используйте для построения модели 6000 точек ($D = 3$ и $m = 5$). Постройте модели с полиномом различных порядков K (от 1 до 5). Постройте графики $\tau_{pred}(K)$ и $\varepsilon(K)$. Определите оптимальное значение порядка полинома.
3. Запишите тот же временной ряд с добавленным шумом (при отношении амплитуды сигнала к стандартному отклонению шума, равном 10^5 , 10^4 , 10^3). Постройте по зашумленным рядам модели с $D = 3$ и найденным оптимальным значением K . (Убедитесь в том, что для вычисления производных теперь требуется большее число m .) Обратите внимание на собственные значения матрицы ковариаций (файл `s_values.txt`) и сделайте оценку размерности. Для наиболее зашумленного ряда постройте график, аналогичный приведенному на рис.6. Постройте графики $\tau_{pred}(\sigma_{noise})$ и $\varepsilon_{coeff}(\sigma_{noise})$.

¹³ Для анализа подобраны временные реализации такой системы, для которой можно было бы сравнить восстановленные по временному ряду значения коэффициентов с исходными

4. Запишите при тех же условиях, что и в п.1, хаотическую временную реализацию координаты x системы Ресслера.
5. Используйте для построения модели 6000 точек ($D = 3$ и $m = 5$). Постройте модель с рациональной функцией в правой части, выберите нужные значения порядков полиномов. Сравните восстановленные значения коэффициентов с истинными.
6. По ряду п.4 постройте модель, используя стандартную полиномиальную аппроксимацию ($D = 3$ и $m = 5$). Выберите оптимальное значение порядка полинома K_{opt} .
7. К временному ряду п.4 добавьте шумы различной интенсивности. Для каждого из зашумленных рядов отследите зависимость качества полиномиальной модели (при $K = K_{opt}$) от длины тренировочного ряда: постройте графики зависимости $\varepsilon(N_{train})$ и $\tau_{pred}(N_{train})$. Определите оптимальную длину ряда. Постройте график максимального времени предсказания от уровня шума $\tau_{pred}(\sigma_{noise})$.

4.4. Контрольные вопросы

1. Что такое временной ряд? Скалярный и векторный ряды?
2. В чем состоит принцип конструирования дифференциальных уравнений по временному ряду?
3. Перечислите основные этапы процедуры моделирования.
4. Что понимают под реконструкцией фазовой траектории?
5. В чем состоит метод Брумхеда-Кинга для оценки размерности наблюдаемого движения?
6. Изложите метод вычисления нескольких временных производных наблюдаемой?
7. Какие основные критерии эффективности динамической модели используются на практике?

значениями. То есть уравнения этой системы должны иметь вид (6) с полиномом или рацио-

Программа FlowSimulator позволяет записать скалярный или векторный временной ряд, генерируемый одной из эталонных систем обыкновенных дифференциальных уравнений. Здесь можно

- выбрать нужную систему,
- значения ее параметров и начальные условия, при которых будет получен временной ряд,
- добавить нормальный белый шум нужной интенсивности к записываемому ряду,
- выбрать файл для записи данных (или создать новый),
- задать *установки* для записи ряда (длительность переходного процесса N_{trans} — первые N_{trans} значений наблюдаемых не записываются, длину записываемого ряда N , метод и шаг интегрирования дифференциальных уравнений, интервал выборки — интервал, с которым записываются значения переменных в файл данных).

Имеется возможность также изменить все указанные величины и записать другой временной ряд в новый файл. Пример внешнего вида окна программы представлен на рис. П.1.1.

Запустите файл FlowSimulator.exe. Откроется окно программы и в левой его части – окно ввода (Input Window).

1) В окне ввода появится пронумерованный список систем, реализации которых можно записать: “You can select one of the following systems: 1 — Rossler system, 2 — Lorenz system, 3 — van der Pol oscillator, ...” («Вы можете выбрать одну из следующих систем: 1 – система Ресслера, 2 — система Лоренца, 3 — осциллятор ван-дер-Поля, ...»): “Select a system: ” («Выберите систему: »). Введите номер нужной системы.

2) Появится явный вид этой системы уравнений, и программа запросит поочередно значения ее параметров: “Enter values of parameters: a = , b = ”

нальной функцией F или приводиться к нему.

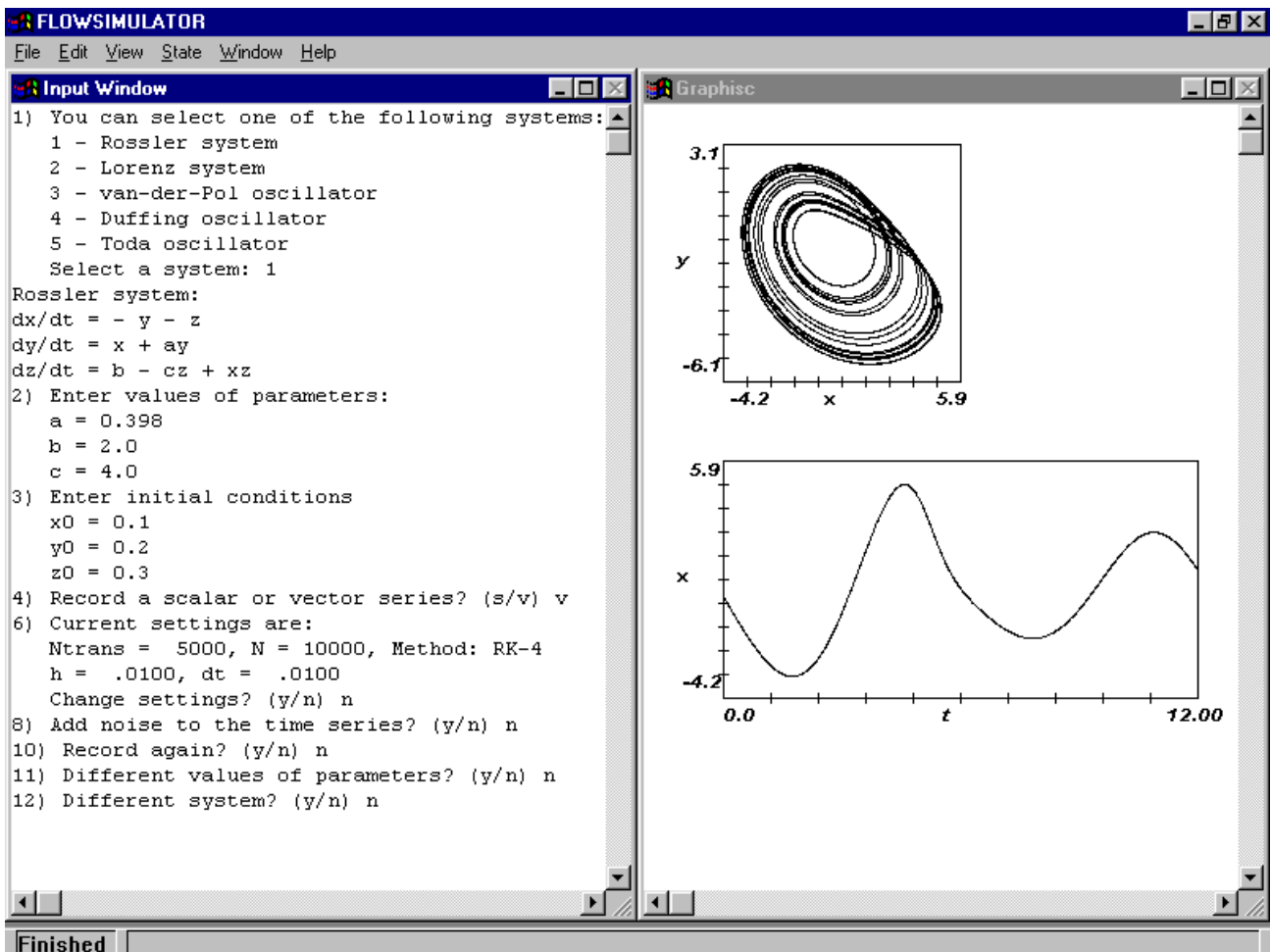


Рис.П.1.1. Внешний вид окна программы FlowSimulator (данный сеанс работы с программой включал в себя интегрирование системы Ресслера методом Рунге-Кутты 4-го порядка с шагом 0.01 при значениях параметров $a = 0.398$, $b = 2.0$, $c = 4.0$ и запись векторного временного ряда длиной 10000 векторов с интервалом выборки, равным шагу интегрирования).

(«Задайте значения параметров: $a =$, $b =$ ») и т.д. Введите нужные значения (обязательно вещественные числа – с разделяющей десятичной точкой).

3) Аналогично потребуется ввести начальные условия: “Enter initial conditions: $x_0 =$, $y_0 =$ ” («Введите начальные условия: $x_0 =$, $y_0 =$ »).

4) Следующий вопрос: “Record a scalar or vector series? (s/v)” («Записать скалярный или векторный временной ряд? (с/в)»). Если нужен скалярный ряд, то введите символ “s” и см. пункт 5. Для записи векторного ряда введите символ “v”, при этом будут записываться реализации всех динамических переменных системы, см. далее пункт 6.

5) Программа запросит номер динамической переменной, временная реализация которой будет записана: “Number of the variable to record: ” («Номер

переменной для записи: »). Введите соответствующее целое число: переменные нумеруются сверху вниз — по левой части системы уравнений.

6) Появится сообщение: “Current settings are: Ntrans = ..., N = ..., Method: ..., h = ..., dt = ...” («Текущие установки: Ntrans = ..., N = ..., Method: ..., h = ..., dt = ...») и вопрос: “Change settings? (y/n)” «Изменить установки? (да/нет)». Если установки Вас не устраивают, то отвечайте положительно (введите символ “y”) и см. п. 7. Иначе отвечайте “n” и далее см. пункт 8.

7) Потребуется ввести длину переходного процесса N_{trans} : “Length of a transient process: Ntrain = ” («Длительность переходного процесса: Ntrans = »), длину записываемого временного ряда N : “Length of the series to record: N = ” («Длительность временного ряда для записи: N = »). Введите нужные целые числа.

“Methods of integration: 1 — Euler, 2 — 4-th order Runge-Kutta routine, 3 — 5-th and 6-th order Runge-Kutta. Select a method: ” («Методы интегрирования: 1 — метод Эйлера, 2 — метод Рунге-Кутты 4-го порядка, 3 — метод Рунге-Кутты 5-го и 6-го порядков с автоматической подстройкой шага. Выберите метод »). “Step of integration: h = ” («Шаг интегрирования: h = »), “Sampling interval: ” («Интервал выборки: »). Введите номер, соответствующий нужному методу интегрирования модельных уравнений. Задайте величину шага интегрирования (для 3-го метода имеется в виду начальное значение шага) и интервала выборки — вещественные числа с разделяющей десятичной точкой.

8) Следующий вопрос: “Add noise to the time series? (y/n)” («Добавить шум к временному ряду? (да/нет)»). Если хотите записать зашумленный ряд, ответьте положительно и см. пункт 9, иначе отвечайте “n” и см. пункт 10.

9) Потребуется ввести величину стандартного отклонения случайной величины: “Noise standard deviation: std = ” («Стандартное отклонение шума: std = »). При этом к значениям «чистых» временных реализаций $x_j(t_i)$ при записи в файл прибавляются независимые случайные величины ξ_i , распределенные по нормальному закону с нулевым средним и заданным стандартным от-

клонением. В сами дифференциальные уравнения случайная составляющая не вводится, т.е. в данном случае добавленный шум моделирует шумы измерительной аппаратуры, а не флуктуации в системе.

10) Далее откроется окно диалога для выбора файла, в который будет записан временной ряд. Выберите нужный файл (или создайте новый). Временной ряд будет записываться в заданный файл, а в правой части окна программы появится окно “Graphics” («Графика»). В этом окне представлены фазовая траектория системы и участок записанной временной реализации (или одной из реализаций).

Появится вопрос: “Record again? (y/n)” («Записать снова? (да/нет)»). В случае положительного ответа — переходите к пункту 4: можно изменить настройки для записи временного ряда, выбрать шум другой интенсивности, другую динамическую переменную для записи, другой файл. Иначе см. пункт 11.

11) Следующий вопрос: “Different values of parameters? (y/n)” («Другие значения параметров? (да/нет)»). Отвечайте “y”, если нужно повторить запись временного ряда при других значениях параметров и/или начальных условий — возвращайтесь к пункту 2. Иначе см. пункт 12.

12) Последний вопрос: “Different system? (y/n)” («Другая система? (да/нет)»). Отвечайте положительно, если хотите выбрать другое отображение и см. пункт 1.

Иначе выполнение программы завершится и появится окно диалога с вопросом: “Exit Window?” «Закреть окно?». Можно ответить «нет», если нужно еще раз просмотреть последние графики. Окно программы можно закрыть позже. В случае положительного ответа окно программы закроется.

Программа VectorODE позволяет построить по векторному временному ряду модельные дифференциальные уравнения. Можно выбрать необходимые значения *параметров алгоритма реконструкции*:

- количество векторов в тренировочном временном ряде,

- вид аппроксимирующих функций,
- порядок полинома, если выбрана полиномиальная аппроксимация, порядки двух полиномов, если выбрана рациональная аппроксимация, период воздействия, если выбрана аппроксимация с внешним гармоническим воздействием,

и *установок* для проверки эффективности модели:

- длину участка тестового временного ряда L_{test} ,
- количество этих участков K_{test} ,
- величину сдвига между соседними участками S_{test} ,
- метод интегрирования модельных дифференциальных уравнений,
- шаг интегрирования модельных дифференциальных уравнений,

Значения всех этих параметров можно менять в течение сеанса работы с программой. Значения всех параметров и установок алгоритма и основные результаты построения моделей выводятся в файл отчета report.txt, который создается в текущем каталоге (если файл с таким именем уже существовал до запуска программы, то новые результаты добавляются к ранее записанной информации). Примеры внешнего вида окна программы и содержимого файла отчета представлены на рис.П.2.1 и рис.П.2.2.

Запустите файл VectorODE.exe. Откроется окно программы и окно диалога для выбора файла данных. Откройте файл, который содержит скалярный временной ряд. Это должен быть текстовый файл (в ASCII-формате), в котором значения переменных записаны в столбец в следующем порядке: $x_1(t_1), x_2(t_1), \dots, x_D(t_1), x_1(t_2), \dots, x_D(t_2), \dots$. Эти значения должны представлять собой вещественные числа (обязательно с разделяющей десятичной точкой).

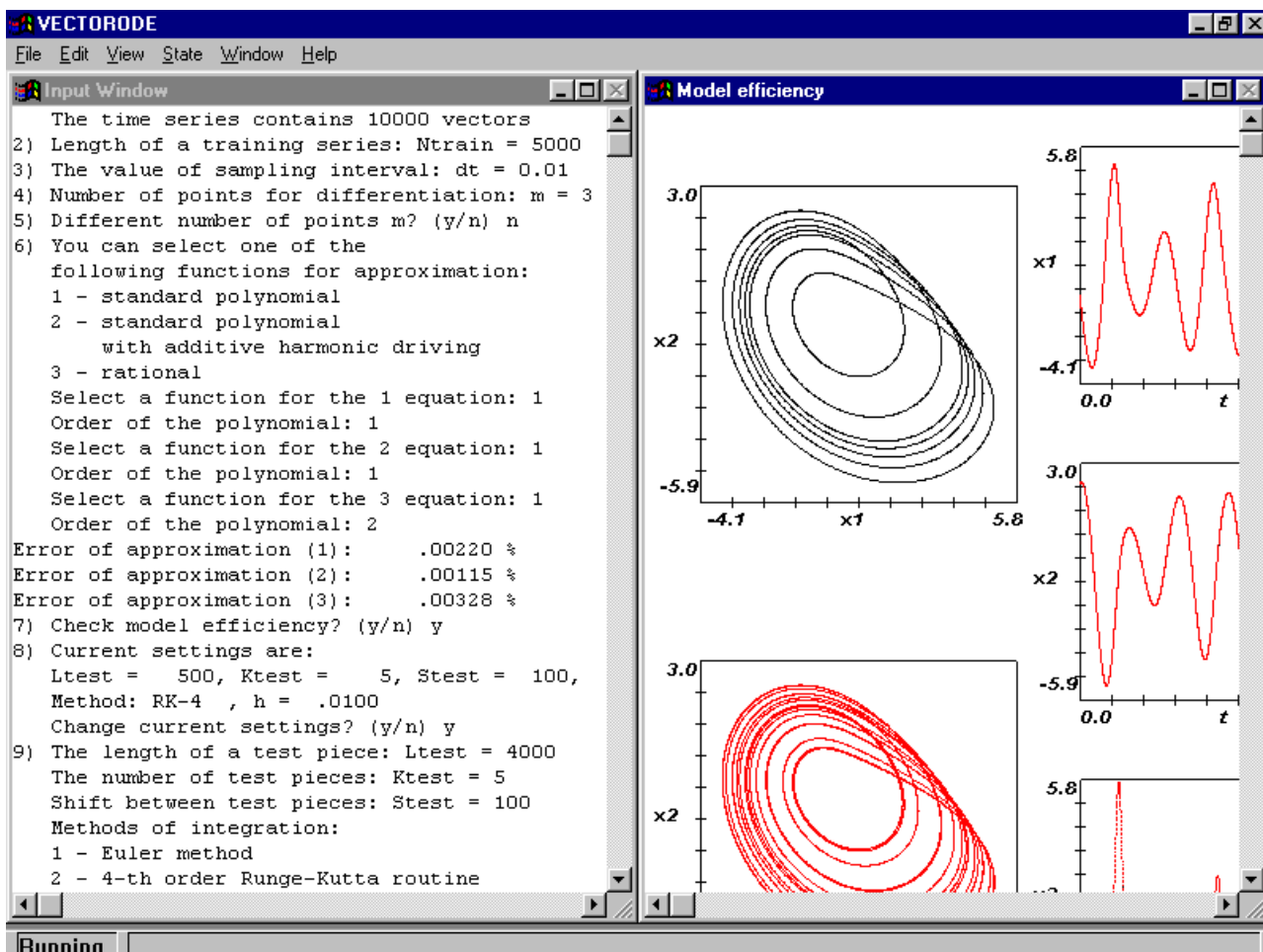


Рис.П.2.1. Внешний вид окна программы VectorODE (данный сеанс работы с программой включал в себя восстановление модельных уравнений по векторному временному ряду, полученного интегрированием системы Ресслера методом Рунге-Кутты 4-го порядка с шагом 0.01 при значениях параметров $a = 0.398$, $b = 2.0$, $c = 4.0$).

1) Программа запросит значение размерности векторов временного ряда: “Dimension of vectors: $D =$ ” («Размерность векторов $D =$ »). Введите нужное число $D > 1$.

2) Программа сообщит общее количество векторов во временном ряду “The time series contains ... vectors” («Временной ряд содержит ... векторов») и запросит количество векторов тренировочного ряда: “Length of a training series: $N_{train} =$ ” («Длина тренировочного ряда $N_{train} =$ »). Если нужно будет проверить эффективность построенных моделей, то задайте длину тренировочного ряда меньше длины всего ряда, чтобы осталось необходимое количество данных и для тестирования.

```
report.txt - Блокнот
Файл  Правка  Поиск  ?

Dimension of vectors D = 3
The number of vectors in a training time series Ntrain = 5000
Sampling interval dt = .010
Number of points for differentiation m = 3

The type of function for the 1 equation:
standard polynomial
Polynomial order nu = 1
Coefficients of the 1 equation:
.0000620934
.0000235032
-1.0000097188
-1.0000779873
Error of approximation: .0022048764 %

The type of function for the 2 equation:
standard polynomial
Polynomial order nu = 1
Coefficients of the 2 equation:
-.0000001186
1.0000187337
.3980074457
.0000000616
Error of approximation: .0011500720 %

The type of function for the 3 equation:
standard polynomial
Polynomial order nu = 2
Coefficients of the 3 equation:
1.9996495255
```

Рис.П.2.2. Содержимое файла отчета для сеанса работы, проиллюстрированного на рис.П.2.1.

3) Будет запрошено значение интервала выборки: “The value of sampling interval: dt = ” «Величина интервала выборки dt = ». Введите нужное вещественное число (обязательно с разделяющей десятичной точкой).

4) Появится приглашение задать количество точек для вычисления производных m : “Number of points for differentiation: m = ” «Количество точек для дифференцирования $m =$ ». Введите целое число $m > 1$.

Производные наблюдаемых будут рассчитаны и выведены графически в окне “Differentiation” («Дифференцирование»), которое откроется в правой части окна программы.

5) Появится вопрос: “Different number of points m ? (y/n)” («Другое количество точек m ? (да/нет)»). Если графики производных недостаточно гладкие, то отвечайте положительно (введите символ “y”) и используйте большее коли-

чество точек для дифференцирования — возврат к пункту 4, иначе отвечайте “n” и см. пункт 6.

б) Появится нумерованный список функций, которые можно использовать для аппроксимации: “You can select one of the following functions for approximation: 1 — standard polynomial, 2 — standard polynomial with additive harmonic driving, 3 — rational” («Вы можете выбрать один из следующих видов функций для аппроксимации: 1 — стандартный полином, 2 — стандартный полином с аддитивным гармоническим воздействием, 3 — дробно-рациональная функция»). И последует запрос: “Select a function for the 1 equation: ” («Выберите вид функции для первого уравнения»). Введите соответствующий номер.

Затем потребуется задать порядок полинома (если выбрана полиномиальная аппроксимация): “Order of the polynomial: ” («Порядок полинома: »). Задайте целое число $K \geq 0$.

Если выбран 2-ой вид функции, то потребуется задать и период воздействия: “Period of driving (in dt units): ” («Период воздействия (в единицах dt)»). Введите вещественное число (обязательно с разделяющей десятичной точкой).

Если выбрана рациональная аппроксимация, то потребуется задать поочередно порядок каждого полинома (в числителе и в знаменателе): “Order of the numerator polynomial: . Order of the denominator polynomial: ” («Порядок полинома в числителе: . Порядок полинома в знаменателе: »).

Далее необходимо сделать выбор для каждого из D уравнений модели.

После этого рассчитываются коэффициенты модельных уравнений. В окне ввода появляется сообщение: “Error of approximation (1): ... %”, “Error of approximation (2): ... %”, ... («Погрешность аппроксимации (1-го уравнения): ... %», «Погрешность аппроксимации (2-го уравнения): ... %», ...). Значения коэффициентов выводятся в файл отчета¹⁴.

¹⁴ Для каждого уравнения значения коэффициентов записываются в столбец в следующем порядке. Сначала записывается свободный член полинома. Затем — коэффициенты при слагаемых в первой степени: x_1, \dots, x_D . Затем — при слагаемых во второй степени: $x_1^2, x_1x_2, \dots, x_1x_D, x_2^2, x_2x_3, \dots, x_D^2$. И так далее с повышением степени слагаемых. Если исполь-

7) Появится вопрос: “Check model efficiency? (y/n)” («Проверить эффективность модели? (да/нет)»). Введите символ “y”, если Вы хотите протестировать модель, и далее см. пункт 8. Иначе отвечайте “n” и далее см. пункт 11.

8) Далее появится сообщение о текущих установках для проверки эффективности модели: “Current settings are: Ltest = ..., Ktest = ..., Stest = ..., Method: ..., h = ...” («Текущие установки: Ltest = ..., Ktest = ..., Stest = ..., Method: ..., h = ...»), и вопрос: “Change settings? (y/n)” («Изменить установки? (да/нет)»). Дайте положительный ответ, если установки Вас не устраивают, и см. пункт 9. В противном случае ответьте “n” и см. пункт 10.

9) Программа предложит задать установки для тестирования модели. “The length of a test piece: Ltest = ” («Длина тестового участка: Ltest = »), “The number of test pieces: Ktest = ” («Количество тестовых участков: Ktest = »), “Shift between test pieces: Stest = ” («Сдвиг между тестовыми участками: Stest = »). Введите нужные значения (положительные целые числа), причем нужно следить, чтобы для тестового ряда хватило данных, т.е. $N_{train} + N_{test} \leq N$.

“Methods of integration: 1 — Euler, 2 — 4-th order Runge-Kutta routine, 3 — 5-th and 6-th order Runge-Kutta. Select a method: ” («Методы интегрирования: 1 — метод Эйлера, 2 — метод Рунге-Кутты 4-го порядка, 3 — метод Рунге-Кутты 5-го и 6-го порядков с автоматической подстройкой шага. Выберите метод»). “Step of integration: h = ” («Шаг интегрирования: h = »). Введите номер, соответствующий нужному методу интегрирования модельных уравнений. Задайте величину шага интегрирования (для 3-го метода имеется в виду начальное значение шага) — вещественное число с разделяющей десятичной точкой.

10) Будет проведено тестирование модели. В правой части окна программы появится окно “Model efficiency” («Эффективность модели»). В нем иллюстрируются результаты тестирования.

зовалась дробно-рациональная функция, то сначала выводятся коэффициенты полинома в числителе, затем — полинома в знаменателе, причем свободный член полинома в знаменателе, равный 1, не выводится. Если использовалась аппроксимация с внешней силой, то после коэффициентов полинома выводятся коэффициенты при косинусе и синусе.

Слева: сверху показана фазовая траектория объекта, восстановленная по тестовому временному ряду, ниже — рассчитанная фазовая траектория модели.

В центре представлены временные реализации переменных объекта (черным цветом) и модели (красным) при одинаковых начальных условиях.

Справа приведены графики зависимости ошибки прогноза каждой из переменных от времени.

В окне ввода распечатываются результаты тестирования: “Error of approximation (test) (1): ... %”, “Error of approximation (test) (2): ... %”, ..., “Prediction time (1): ...”, “Prediction time (2): ...”, ..., “Number of infinite solutions: ...” («Погрешность аппроксимации (по тестовому ряду) (1-го уравнения): ... %», «Погрешность аппроксимации (по тестовому ряду) (2-го уравнения): ... %», «Дальность прогноза (1-ой переменной): ...», «Дальность прогноза (2-ой переменной): ...», ..., «Количество неограниченных решений: ...»). Все результаты записываются также в файл отчета.

Появится вопрос: “Check again? (y/n)” («Проверить снова? (да/нет)»). Если Вы хотите проверить модель с другими установками, то отвечайте “y” и возвращайтесь к п. 8. Иначе отвечайте отрицательно и см. пункт 11.

11) Последует вопрос: “Different model? (y/n)” («Другую модель? (да/нет)»). Отвечайте положительно, если нужно построить модель, например, с другими порядками полинома, и см. п. 6. Иначе отвечайте “n” и см. пункт 12.

12) Появится вопрос: “Different length Ntrain? (y/n)” («Другую длину тренировочного ряда Ntrain? (да/нет)»). Отвечайте “y”, если нужно провести моделирование по тренировочному ряду большей или меньшей длины — см. пункт 2. Иначе отвечайте “n” переходите к пункту 13.

13) Последний вопрос: “Different time series? (y/n)” («Другой временной ряд? (да/нет)»). В случае положительного ответа Вам будет предложено выбрать новый файл данных для анализа (см. начало описания программы).

Иначе выполнение программы завершится и появится окно диалога с вопросом: “Exit Window?” («Закреть окна?»). Можно ответить «нет», если нужно

еще раз просмотреть последние графики (окно программы можно закрыть позже). В случае положительного ответа окно программы закроется.

Программа ScalarODE позволяет построить по скалярному временному ряду модельные дифференциальные уравнения. Можно выбрать необходимые значения *параметров алгоритма реконструкции*:

- количество векторов в тренировочном временном ряду,
- вид аппроксимирующих функций,
- порядок полинома, если выбрана полиномиальная аппроксимация, порядки двух полиномов, если выбрана рациональная аппроксимация, период воздействия, если выбрана аппроксимация с внешним гармоническим воздействием,

и *установок* для проверки эффективности модели:

- длину участка тестового временного ряда L_{test} ,
- количество этих участков K_{test} ,
- величину сдвига между соседними участками S_{test} ,
- метод интегрирования модельных дифференциальных уравнений,
- шаг интегрирования модельных дифференциальных уравнений,

Значения всех этих параметров можно менять в течение сеанса работы с программой. Значения всех параметров и установок алгоритма и основные результаты построения моделей выводятся в файл отчета report.txt, который создается в текущем каталоге (если файл с таким именем уже существовал до запуска программы, то новые результаты добавляются к ранее записанной информации). Примеры внешнего вида окна программы и содержимого файла отчета представлены на рис.П.3.1 и рис.П.3.2.

Запустите файл ScalarODE.exe. Откроется окно программы и окно диалога для выбора файла данных. Откройте файл, который содержит скалярный временной ряд. Это должен быть текстовый файл (в ASCII-формате), в котором

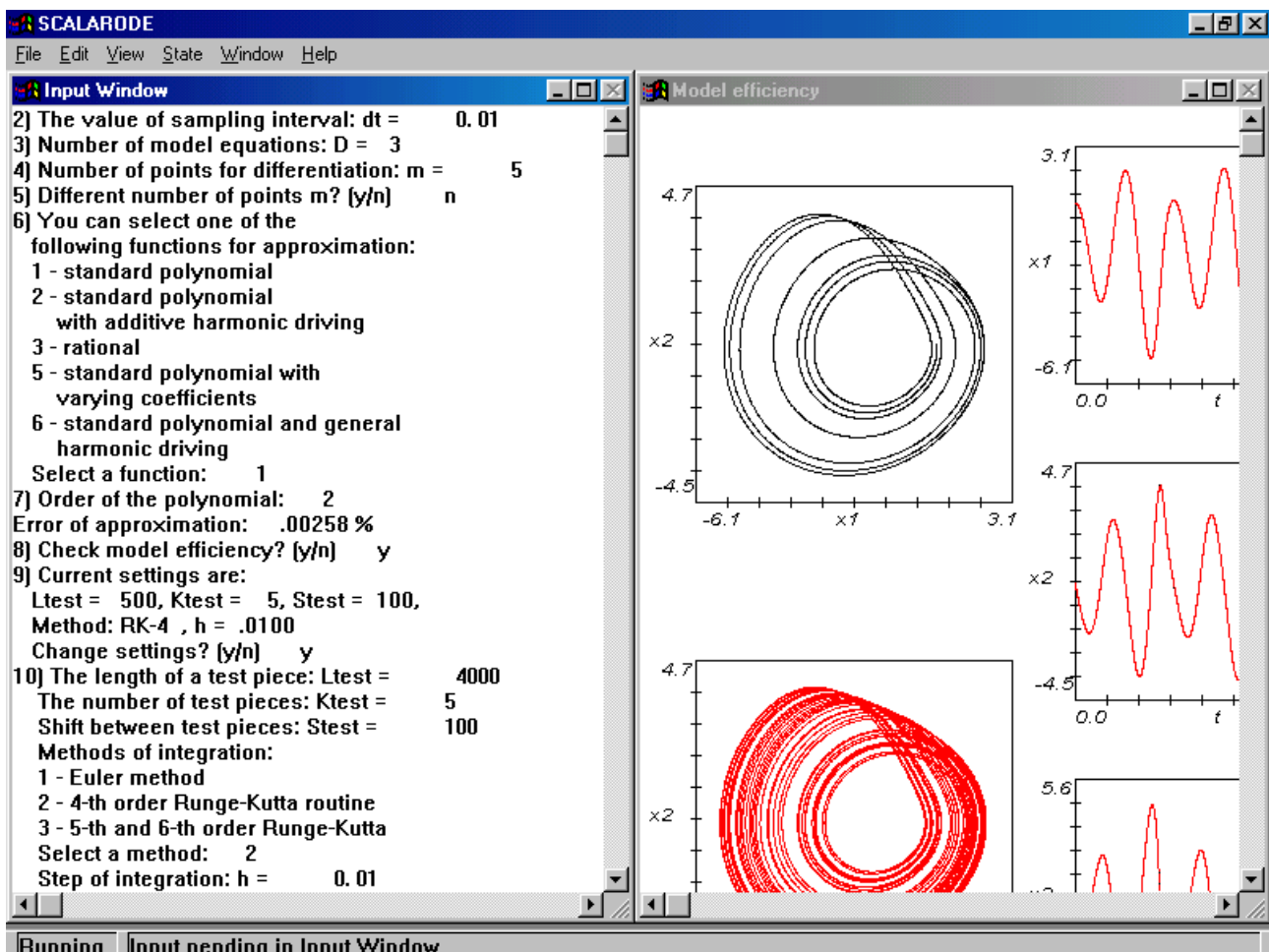


Рис.П.3.1. Внешний вид окна программы ScalarODE (данный сеанс работы с программой включал в себя восстановление модельных уравнений по временной реализации координаты y системы Ресслера, полученной интегрированием этой системы ОДУ методом Рунге-Кутты 4-го порядка с шагом 0.01 при значениях параметров $a = 0.398$, $b = 2.0$, $c = 4.0$).

в столбец записаны последовательные значения наблюдаемой — вещественные числа (обязательно с разделяющей десятичной точкой).

В левой части обрамляющего окна открывается окно ввода (Input Window), которое остается открытым в течение всего сеанса работы с программой — это окно диалога пользователя с программой. Сначала программа считает количество значений наблюдаемой N в данном временном ряде и выдает сообщение: “The time series contains ... values” («Временной ряд содержит ... значений»).

1) Появится приглашение задать длину тренировочного временного ряда (количество значений): “Length of a training time series: Ntrain = ” («Длина тренировочного временного ряда Ntrain = »). Если нужно будет проверять эффек-


```

report.txt - Блокнот
Файл  Правка  Поиск  ?

The number of values in a training time series Ntrain = 5000
Sampling interval dt = .010
The number of model equations D = 3
The number of points for differentiation m = 5

The type of function for the model equation: standard polynomial
Polynomial order nu = 2
Coefficients of the model equation:
-1.9993766351
-3.9988749461
.5914249749
-3.6009749120
-.3980461923
1.1580004519
-.3980961902
-.3978429686
.9996227508
-.0000467970
Error of approximation: .00277 %

Error of approximation (test series): .00337 %
Length of a test piece: Ltest = 4000
Number of test pieces: Ktest = 5
Shift between test pieces: Stest = 100
Method of integration: RK-4
Step of integration: h = .01000
Prediction time for the 1 variable is equal to 4000
Prediction time for the 2 variable is equal to 4000
Prediction time for the 3 variable is equal to 4000
Number of infinite solutions: 0

```

Рис.П.3.2. Содержимое файла отчета для сеанса работы, проиллюстрированного на рис.П.3.1.

тивность построенных моделей, то задайте длину тренировочного ряда меньше длины всего ряда, чтобы осталось необходимое количество данных и для тестирования. Значение N_{train} — целое число, для которого справедливо.

2) Будет запрошено значение интервала выборки: “The value of sampling interval: dt = ” «Величина интервала выборки dt = ». Введите нужное вещественное число (обязательно с разделяющей десятичной точкой).

3) Программа запросит количество уравнений модели: “Number of model equations: D = ” («Количество уравнений модели D = »). Задайте целое число $D > 1$.

4) Появится приглашение задать количество точек для вычисления производных m : “Number of points for differentiation: m = ” «Количество точек для дифференцирования m = ». Введите целое число $m > D$.

Производные наблюдаемой будут рассчитаны и выведены графически в окне “Differentiation” («Дифференцирование»), которое откроется в правой части окна программы. Собственные значения и координаты собственных век-

торов матрицы ковариаций будут выведены соответственно в файлы `s_values.txt` и `s_vectors.txt` в текущем каталоге.

5) Появится вопрос: “Different number of points m ? (y/n)” («Другое количество точек m ? (да/нет)»). Если графики производных недостаточно гладкие, то отвечайте положительно (введите символ “y”) и используйте большее количество точек для дифференцирования — возврат к пункту 4, иначе отвечайте “n” и см. пункт 6.

6) Появится нумерованный список функций, которые можно использовать для аппроксимации: “You can select one of the following functions for approximation: 1 — standard polynomial, 2 — standard polynomial with additive harmonic driving, 3 — rational, 4 — function for a harmonically driven oscillator with linear damping” («Вы можете выбрать один из следующих видов функций для аппроксимации: 1 — стандартный полином, 2 — стандартный полином с аддитивным гармоническим воздействием, 3 — дробно-рациональная функция, 4 — функция для гармонически возбуждаемого осциллятора с линейным затуханием¹⁵»). И последует запрос: “Select a function: ” («Выберите вид функции»). Введите соответствующий номер.

7) Затем потребуется задать порядок полинома (если выбрана полиномиальная аппроксимация): “Order of the polynomial: ” («Порядок полинома: »). Задайте целое число $K \geq 0$.

Если выбран 2-ой или 4-ый вид функции, то потребуется задать и период воздействия: “Period of driving (in dt units): ” («Период воздействия (в единицах dt)»). Введите вещественное число (обязательно с разделяющей десятичной точкой).

Если выбрана рациональная аппроксимация, то потребуется задать поочередно порядок каждого полинома (в числителе и в знаменателе): “Order of the numerator polynomial: . Order of the denominator polynomial: ” («Порядок полинома в числителе: . Порядок полинома в знаменателе: »).

¹⁵ 4-ый вид функции может использоваться только при $D = 2$.

После этого рассчитываются коэффициенты модельных уравнений. В окне ввода появляется сообщение: “Error of approximation: ... %” («Погрешность аппроксимации: ... %»). Значения коэффициентов выводятся в файл отчета¹⁶.

8) Появится вопрос: “Check model efficiency? (y/n)” («Проверить эффективность модели? (да/нет)»). Введите символ “y”, если Вы хотите протестировать модель, и далее см. пункт 9. Иначе отвечайте “n” и далее см. пункт 12.

9) Далее появится сообщение о текущих установках для проверки эффективности модели: “Current settings are: Ltest = ..., Ktest = ..., Stest = ..., Method: ..., h = ...” («Текущие установки: Ltest = ..., Ktest = ..., Stest = ..., Method: ..., h = ...»), и вопрос: “Change settings? (y/n)” («Изменить установки? (да/нет)»). Дайте положительный ответ, если установки Вас не устраивают, и см. пункт 10. В противном случае ответьте “n” и см. пункт 11.

10) Программа предложит задать установки для тестирования модели. “The length of a test piece: Ltest = ” («Длина тестового участка: Ltest = »), “The number of test pieces: Ktest = ” («Количество тестовых участков: Ktest = »), “Shift between test pieces: Stest = ” («Сдвиг между тестовыми участками: Stest = »). Введите нужные значения (положительные целые числа), причем нужно следить, чтобы для тестового ряда хватило данных, т.е. $N_{train} + N_{test} \leq N$.

“Methods of integration: 1 — Euler, 2 — 4-th order Runge-Kutta routine, 3 — 5-th and 6-th order Runge-Kutta. Select a method: ” («Методы интегрирования: 1 — метод Эйлера, 2 — метод Рунге-Кутты 4-го порядка, 3 — метод Рунге-Кутты 5-го и 6-го порядков с автоматической подстройкой шага. Выберите метод »). “Step of

¹⁶ Для каждого уравнения значения коэффициентов записываются в столбец в следующем порядке. Сначала записывается свободный член полинома. Затем — коэффициенты при слагаемых в первой степени: x_1, \dots, x_D . Затем — при слагаемых во второй степени: $x_1^2, x_1x_2, \dots, x_1x_D, x_2^2, x_2x_3, \dots, x_D^2$. И так далее с повышением степени слагаемых. Если использовалась дробно-рациональная функция, то сначала выводятся коэффициенты полинома в числителе, затем — полинома в знаменателе, причем свободный член полинома в знаменателе, равный 1, не выводится. Если использовалась аппроксимация с внешней силой, то после коэффициентов полинома выводятся коэффициенты при косинусе и синусе. Если использовалась аппроксимация № 4, то будут коэффициенты будут выведены в следующем порядке: $c_{0,0}, c_{1,0}, c_{0,1}, c_{2,0}, c_{3,0}, c_{4,0}, \dots, c_{K,0}, A, B$.

integration: $h =$ ” («Шаг интегрирования: $h =$ »). Введите номер, соответствующий нужному методу интегрирования модельных уравнений. Задайте величину шага интегрирования (для 3-го метода имеется в виду начальное значение шага) — вещественное число с разделяющей десятичной точкой.

11) Будет проведено тестирование модели. В правой части окна программы появится окно “Model efficiency” («Эффективность модели»). В нем иллюстрируются результаты тестирования.

Слева: сверху показана фазовая траектория объекта, восстановленная по тестовому временному ряду, ниже — рассчитанная фазовая траектория модели.

В центре представлены временные реализации переменных объекта (черным цветом) и модели (красным) при одинаковых начальных условиях.

Справа приведены графики зависимости ошибки прогноза каждой из переменных от времени.

В окне ввода распечатываются результаты тестирования: “Error of approximation (test): ... %”, “Prediction time for the ... variable: ...”, “Number of infinite solutions: ...” («Погрешность аппроксимации (по тестовому ряду): ... %», «Дальность прогноза для ... переменной: ...», «Количество неограниченных решений: ...»). Все результаты записываются также в файл отчета.

Появится вопрос: “Check again? (y/n)” («Проверить снова? (да/нет)»). Если Вы хотите проверить модель с другими установками, то отвечайте “y” и возвращайтесь к п. 8. Иначе отвечайте отрицательно и см. пункт 12.

12) Последует вопрос: “Different model? (y/n) ” («Другую модель? (да/нет)»). Отвечайте положительно, если нужно построить модель, например, с другим порядком полинома, и см. п. 6. Иначе отвечайте “n” и см. пункт 13.

13) Очередной вопрос: “Different number of equations? (y/n) ” («Другое количество уравнений модели? (да/нет) »). Если требуется построить модель большей размерности и/или выбрать другое количество точек для дифференцирования, то нужно ответить «да» — и возвратиться к пункту 4. Иначе — см. пункт 14.

14) Появится вопрос: “ Different length Ntrain? (y/n)” («Другую длину тренировочного ряда Ntrain? (да/нет)»). Отвечайте “y”, если нужно провести моделирование по тренировочному ряду большей или меньшей длины — см. пункт 1. Иначе отвечайте “n” переходите к пункту 15.

15) Последний вопрос: “ Different time series? (y/n) ” («Другой временной ряд? (да/нет)»). В случае положительного ответа Вам будет предложено выбрать новый файл данных для анализа (см. начало описания программы).

Иначе выполнение программы завершится и появится окно диалога с вопросом: “Exit Window?” («Закреть окна?»). Можно ответить «нет», если нужно еще раз просмотреть последние графики (окно программы можно закрыть позже). В случае положительного ответа окно программы закроется.

1. Crutchfield J.P., McNamara B.S. “Equations of motion from a data series”, Complex Systems, 1987, Vol. 1, P. 417-452.
2. Cremers J., Hubler A. “Construction of differential equations from experimental data”, Z.Naturforschung A, 1987, Vol. 42, P. 797-802.
3. Farmer J.D., Sidorowich J.J. “Predicting chaotic time series”, Phys.Rev.Lett., 1987, Vol. 59, P. 845-848.
4. Breeden J.L., Hubler A. “Reconstructing equations of motion from experimental data with unobserved variables”, Phys.Rev. A, 1990, Vol. 42, № 10, P. 5817-5826.
5. Baake E., Baake M., Bock H.J., Briggs K.M. “Fitting ordinary differential equations to chaotic data”, Phys.Rev. A, 1992, Vol. 45, № 8, P. 5524-5529.
6. Gouesbet G., Maquet J. “Construction of phenomenological models from numerical scalar time series”, Physica D, 1992, Vol. 58, P. 202-215.
7. Gouesbet G., Letellier C. “Global vector-field approximation by using a multivariate polynomial L_2 approximation on nets”, Phys.Rev. E, 1994, Vol. 49, P. 4955-4972.
8. Brown R., Rulkov N.F., Tracy E.R. “Modeling and synchronizing chaotic systems

- from time-series data”, Phys.Rev. E, 1994, Vol. 49, № 5, P. 3784-3800.
9. Грибков Д.А., Грибкова В.В., Кравцов Ю.А., Кузнецов Ю.И., Ржанов А.Г. “Восстановление структуры динамической системы по временным рядам”, Радиотехника и электроника, 1994, Т. 39, В. 2, С. 269-277.
 10. Янсон Н.Б., Анищенко В.С. “Моделирование динамических систем по экспериментальным данным”, Изв. ВУЗов. Прикладная нелинейная динамика, 1995, Т. 3, № 3, С.112-121.
 11. Packard N.H., Crutchfield J.P., Farmer J.D., Shaw R.S. “Geometry from a time series”, Phys.Rev.Lett., 1980, Vol. 45, № 9, P. 712-716.
 12. Takens F. “Detecting strange attractors in turbulence”, in Dynamical Systems and Turbulence, Warwick, 1980, eds. D.Rang and L.S.Young, Lecture Notes in Mathematics, Vol. 898, P. 366-381.
 13. Broomhead D.S., King G.P. “Extracting qualitative dynamics from experimental data”, Physica D, 1986, Vol. 20, P. 217-236.
 14. Kennel M.B., Brown R., Abarbanel H.D.I. “Determining embedding dimension for phase-space reconstruction using a geometrical construction”, Phys.Rev. A, 1992, Vol. 45, № 6, P. 3403-3411.
 15. Кравцов Ю.А. “Случайность, детерминированность, предсказуемость”, Успехи физ. наук, 1989, Т. 158, № 1, С. 93-115.

Учебно-методическое пособие

БЕЗРУЧКО Борис Петрович
СМИРНОВ Дмитрий Алексеевич

**Реконструкция обыкновенных дифференциальных
уравнений по временным рядам**

ГосУНЦ «Колледж», Лицензия ЛР №020773 от 15.05.98

Подписано к печати 10.05.00. Формат 60x84 1/16.
Бумага Papirus Slim. Гарнитура Times.
Усл. печ. л. 2,79. Уч.-изд. л. 2,6. Тираж 100 экз. Заказ 166.

Издательство ГосУНЦ «Колледж»
410026, Саратов, ул. Астраханская, 83.
Тел. (845-2) 523864



Отпечатано на ризографе издательства ГосУНЦ «Колледж»

Замеченные недостатки: не напечатан следующий рис.4, на который имеется ссылка на стр.17.

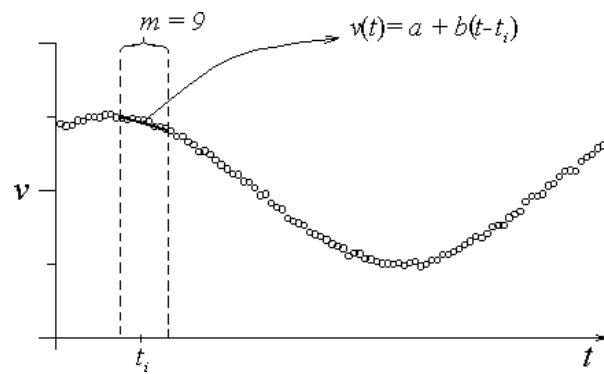


Рис.4. Расчет производной при наличии шума. Производная в момент времени t_i вычисляется здесь с использованием 9 соседних точек.