Решение любой задачи оценивания для f по g может быть выражено через решение подобной задачи для любого случайного элемента h, такого, что $h \overset{\bullet}{\sim}_f g$, например, на основании данных другого преобразователя, функционирующего параллельно этому, или ПИ другой конфигурации. С учетом [4] h и g — семантически f-эквивалентны, следовательно, переход от g к h не приведет к потере качества решения задачи, т. е. для любого ПИ $l \in \mathcal{O}_f(h)$ найдется ПИ $\rho \in \mathcal{O}_f(g)$, такой, что $\rho \circ g \overset{\circ}{\approx}_f l \circ h$, и наоборот. Пусть $g \simeq_f n \circ h$ для $n \perp f$, h и $\rho \in \mathcal{O}_f(h)$. Выбирая $\rho = l \circ n$, мы получаем, что $l \circ h \simeq_f \rho \circ g$ и, следовательно, $l \circ h \overset{\circ}{\approx}_f \rho \circ g$.

Изложенная задача оценивания может быть сведена к задаче оценивания с априорной информацией, а именно, к построению $\mathcal{O}_{n \circ g}(g)$, т. е. к задаче оценивания по случайному элементу g для СЭ $n \circ g$ (где n — заданный ПИ, $n \perp g$). Для рассматриваемого примера — это оценка случайного элемента $q_i \circ n \circ g$ по g, т. е. нахождение ПИ ρ, для которого погрешности оценивания

$$\theta(f * h) = \mathbf{E}||\rho \circ g - q_i \circ n \circ g||^2$$

минимальны.

ЗАКЛЮЧЕНИЕ

Таким образом, используя термины теории категорий, можно рассматривать преобразователи информации, оценивать их информационную устойчивость, т. е. возможность функционирования в условиях отказов, как с учетом, так и безотносительно их топологической структуры. Подход позволяет анализировать сложные информационно-измерительные системы и аппаратно-программные комплексы с единых терминологических позиций к осуществляемым в них информационным процессам в условиях возможных информационных отказов.

СПИСОК ЛИТЕРАТУРЫ

1. Сильвестров С. Д., Васильев В. В. Структура космических измерительных систем.— М.: Сов. радио, 1979.
2. Калман Р., Фалб П., Арбиб М. Очерки по математической теории систем.— М.: Мир, 1971.
3. Месарович М., Тахара Я. Общая теория систем: Математические основы.— М.: Мир, 1978.
4. Голубцов П. В. Относительная информативность и априорная информация в категории линейных преобразователей информации// Пробл. передачи информ.— 1995 .— Т. 31 .— № 3.
5. Бурый А. С., Васильев В. В. Декомпозиция и интеграция отказоустойчивых процедур оценивания состояния динамических систем// Изв. РАН. Техн. кибернетика.— 1993 .— № 3 .— С. 152–159.
6. Брандин В. Н., Васильев А. А., Куницкий А. А. Экспериментальная баллистика космических аппаратов.— М.: Машиностроение, 1984.
7. Пытьев Ю. П. Математические методы интерпретации эксперимента.— М.: Высш. шк., 1989.

Материал поступил в редакцию 10.11.97.

УДК 002:[004:004.7]

Л. В. Шуткин

Новое мышление компьютерного мира

В статье введены в обиход Д-сети Гренандера, позволившие адаптировать общую теорию паттернов к моделированию и инженерному проектированию открытых компьютерных систем. Дано популярное определение специальной теории паттернов, ориентированной на практическое применение к компьютерным системам

ВВЕДЕНИЕ

Интернет и World Wide Web (Всемирная информационная паутина) объединили миллионы ранее изолированных компьютеров и их пользователей в единую глобальную информационно-техническую сеть. В результате стремительной интеграции компьютерного мира образовался значительный разрыв между быстро прогрессирующими способами практического проектирования открытых компьютерных систем и медленно развивающимися методами их математического описания. Об этом свидетельствуют следующие факты. Гипертексты и гипермедиа составляют основу World Wide Web. Однако до сих пор не созданы математические модели, пригодные для их проектирования, классификации и стандартизации. В компьютерных технологиях повсеместно используются разметка, копирование, соединение и разъединение текстовых, графических и иных данных. Вместе с тем, нет моделей, формально описывающих эти массовые операции, составляющие основу любой компьютерной технологии. Для проектирования и стандартизации взаимодействующих комбинированных баз данных необходим единый математический подход к представлению линейных, иерархических, сетевых и табличных структур данных. Специалистам, работающим в области проектирования и стандартизации открытых систем, базирующихся на Интернет, нужна математическая модель сетевой компьютерной архитектуры. Перечень ожидающих решения актуальных компьютерных проблем можно продолжить.

Разрыв между математическими моделями и практическими методами проектирования компьютерных систем тревожит не только математиков. Он беспокоит и компьютерных специалистов, поскольку отрицательно сказывается на инженерном проектировании и стандартизации компьютеров, Интернет и технологии World Wide Web. Результаты выполненных в нашей стране научных исследований и инженерных разработок убедительно показали, что отставание компьютерной науки от практики существенно сокращается с помощью теории паттернов, поскольку ее методы позволяют решать перечисленные выше компьютерные проблемы. Основы общей теории паттернов создал известный американский математик Ульф Гренандер [1, 2]. До последнего времени она не применялась к компьютерам. Мы адаптировали (приспособили) общую теорию паттернов к решению практических компьютерных задач, а затем использовали ее в целях моделирования и проектирования компьютерных объектов, систем и технологий [3, 4, 5]. Адаптированную к практике общую теорию паттернов мы назвали специальной теорией паттернов. Эта идея оказалась исключительно плодотворной в теоретическом и, главное, в практическом плане. Достаточно сказать, что первое же применение специальной теории паттернов к практическому проектированию компьютеров привело к созданию нескольких ноу-хау в области гипертекстов/гипермедиа и технологии World Wide Web. Более того, на основе ноу-хау разработаны оригинальные программные средства и технология разметки и кодирования гипертекстов/гипермедиа. С помощью специальной теории паттернов удалось получить новые теоретические результаты, в частности, показать, что реляционную модель Кодда можно построить путем преобразования паттерновой модели данных [5].

Теория паттернов в виде, приведенном в работах Гренандера, трудна для понимания. Причина трудностей заключается в незавершенности этой многообещающей теории вследствие недостатка примеров ее практического использования. Специальная теория паттернов построена на основе решения практических задач, потому ее можно изложить в доступной форме.

Данная статья — первая из цикла, представляющего собой опыт популярного изложения идей и методов специальной теории паттернов. Чтобы сделать ее удобной для практического применения, мы ввели в обиход понятие "сети Гренандера", благодаря которым удалось "приземлить" общую теорию паттернов до уровня, понятного пользователям компьютеров. Цикл состоит из четырех статей.

В первой рассмотрена адаптация общей теории паттернов к моделированию и проектированию компьютерных систем, и введено понятие "сети Гренандера".

Популярное и, вместе с тем, достаточно строгое изложение специальной теории паттернов стало возможным только после накопления многочисленных примеров практического использования сетей Гренандера в моделировании и проектировании программных средств и компьютерных технологий. Поэтому вторая статья будет посвящена примерам применения сетей Гренандера к компьютерной науке и практике.

В последней статье будут приведены наложенные на общую теорию паттернов ограничения, позволившие построить специальную теорию паттернов. Кроме того, там мы попытались заглянуть в будущее и оценить возможное влияние практического применения сетей Гренандера и специальной теории паттернов в открытом компьютерном мире и в интеграции теории открытых систем. Оценка многочисленных примеров решения компьютерных задач позволяет с большой степенью уверенности утверждать, что широкое практическое использование специальной теории паттернов и сетей Гренандера приведет компьютерный мир к новому методу мышления.

1. ТЕОРИЯ ПАТТЕРНОВ И СЕТИ ГРЕНАНДЕРА

Природа устроена просто

Льюис Томас

Теория паттернов представляет взаимосвязанные объекты реального мира формальными моделями, обладающими связями. Английское слово *паттерн* означает образец, трафарет, шаблон. Можно сказать, что Гренандер создал основы теории логических шаблонов, моделирующих открытые объекты и состоящие из них открытые системы. Основным элементом теории паттернов является *образующая (generator)*. Отдельная образующая обозначается символом g, а множество образующих — символом G. В общей теории паттернов множество образующих G может быть конечным, счетным и даже континуумом. Наша задача заключается в применении теории паттернов к компьютерам, которые являются устройствами, состоящими из конечного числа деталей. Поэтому будем рассматривать только конечные множества образующих. Конечное множество, состоящее из n образующих, обозначается G_n. Каждая образующая g_i, $i=1, 2, \ldots, n$ обладает не отделимыми от нее связями. Они могут быть неориентированными и ориентированными (входными и выходными). Путем попарного соединения связей двух или большего числа образующих конструируется конфигурация теории паттернов. Попарное соединение связей означает, что одна связь всегда соединяется только с одной другой связью. Если образующие представляют компьютерные и иные открытые объекты реального мира, то конфигурации моделируют состоящие из них системы.

Теория паттернов рассматривает связи как неотъемлемые компоненты образующих. В этом ее главная идея. А попарное соединение связей образующих — основной принцип теории паттернов. Тем самым, теория паттернов проводит четкую границу между понятиями "связь" и "соединение двух связей". Принцип попарного соединения связей образующих отражает одно из фундаментальных свойств природы. Дело в том, что основы природы устроены просто, и потому связи двух объектов реального мира зачастую соединяются одна с одной, т. е. самым простым из возможных способов. В частности, синапсы (связи) нейронов мозга соединены попарно. Валентности (связи) атомов химических элементов соединяются одна с одной. Попарно соединяются связи компьютерных объектов. Например, любая ссылка-кнопка (выходная связь) окна главного меню системы Windows соединяется в памяти компьютера с одним другим окном, вызываемым на экран дисплея компьютера щелчком по кнопке.

Всякая образующая g_i представляется вектором признаков, обозначаемым как $a(g)_i$. Компонентами вектора $a(g)_i$ служат символы, характеризующие признаки, присущие образующей g_i и объектам, которые она моделирует. Практика паттернового моделирования компьютеров показала, что компьютерные системы, состоящие из взаимосвязанных объектов, можно представлять формально и изображать схематично на бумаге в виде конфигураций теории паттернов. Конфигурациями моделируются гипертексты и гипермедиа, электронные книги, наборы данных о людях, хранящиеся в сети Интернет документы HTML, информационная паутина World Wide Web и другие компьютерные системы, состоящие из взаимосвязанных объектов. При этом компьютерные объекты (окна Windows, гипертекстовые/гипермедиа темы, страницы электронных книг и т. п.) представляются образующими теории паттернов. Сильная сторона паттернового моделирования заключается в возможности изображать образующие и конфигурации, представляющие открытые компьютерные объекты и системы, наглядными схемами, рисуемыми на бумаге подобно графам. Паттерновые модели и их наглядные схемы позволяют имитировать операции копирования, соединения и разъединения компьютерных объектов и систем, различные способы их кодирования, а также решать другие задачи моделирования и проектирования компьютеров.

Анализ многочисленных практических примеров паттернового проектирования компьютерных систем показал, что множество G_n образующих, моделирующих компьютерные объекты, целесообразно представить обобщенным вектором признаков:

$$a(g_i) = a(i, \alpha, \gamma_{il}, \beta_{im}^{in}, \beta_{ir}^{out}). \qquad (1)$$

Рассмотрим компоненты этого вектора, играющего важную роль в паттерновом моделировании и проектировании компьютеров. Символ i=1, 2, ..., n означает порядковый номер образующей g_i в конечном множестве образующих G_n. Символ α называется индексом класса множества образующих. Обратите внимание, что все компоненты вектора $a(g_i)$ кроме α, помечены индексом i и, следовательно, принадлежат только i-й образующей. Символ α не имеет индекса i и присваивается двум или большему числу образующих. Отмечая несколько образующих, индекс указывает, что они моделируют реальные объекты, обладающие одним или несколькими общими свойствами. Например, если n=5, то первые две образующие множества G_n могут принадлежать классу α_1, а остальные — классу α_2. Индексом α_1, в частности, можно пометить образующие, моделирующие текстовые строки, хранящиеся в одном файле, а индексом α_2 образующие, моделирующие строки, хранящиеся в другом файле. Индексы i, m, in и i, r, out компонент β_{im}^{in}, β_{ir}^{out} обозначают входные (in) и выходные (out) связи образующей g_i, а символы m, r являются параметрами.

Путем изменения параметров m, r в диапазонах m=0, 1, 2, ..., m^{max}, r=0, 1, 2, ..., r^{max}, при условии i=1, 2, ..., n, из вектора признаков (1) получаются образующие с различными числами входных и выходных связей (Рис. 1). Рисунок наглядно показывает, что образующие являются "открытыми" объектами. Компоненты β_{im}^{in}, β_{ir}^{out} называются показателями входных и выходных связей образующей g_i. Если m=2, то β_{i2}^{in} — показатель второй входной связи образующей, а если r=3, то β_{i3}^{out} — показатель третьей выходной связи образующей g_i (Рис. 1,b). Компоненты γ_{il} — это атрибуты (характерные свойства) образующей g_i. Если l=1, то образующая имеет один атрибут γ_{il}, значением которого может служить, например, имя объекта, моделируемого образующей. Вектор признаков (1) определяет лишь небольшое подмножество образующих, рассматриваемых в общей теории паттернов. Но этот относительно узкий класс образующих дает возможность решать многие компьютерные практические задачи и теоретические проблемы.

У образующих и конфигураций формы отделены от их содержания. Это обеспечивает теории паттернов многочисленные практические и теоретические применения, поскольку проблема разделения форм и содержания — центральная в научных и прикладных дисциплинах. Она актуальна также в поэзии, музыке, и других видах искусства. Теория паттернов изучает подобие и общие свойства форм (структурных шаблонов) объектов и составленных из них систем. Покажем на примерах образующих, получаемых путем изменения параметров l, m, r обобщенного вектора признаков (1), каким образом теория паттернов моделирует разделение форм и содержания компьютерных и иных объектов. С этой целью компонентам γ_{il}, β_{im}^{in}, β_{ir}^{out} вектора (1) ставятся в соответствие конечные или счетные множества D_{il}, D_{im}^{in}, D_{ir}^{out}, которые называются *доменами компонент образующей.*

Вектора признаков, получаемые от соотношений (1) изменением значений параметров l, m, r, моделируют формы, а в соответствующих им доменах содержатся данные (информация) об объектах реального мира, обладающих связями. Каждая компонента образующей g_i, определяемой вектором признаков (1), имеет свой домен. Например, компонента γ_{i1} имеет домен D_{i1}, а компонента β_{i2}^{in} — домен D_{i2}^{in}. В специальной теории паттернов домены являются конечными или счетными множествами значений (данных). Данные, которые находятся в доменах при моделировании конкретных компьютерных или иных объектов, присваиваются в качестве значений соответствующим компонентам векторов признаков образующих. Любой домен образующей g_i всегда содержит значение λ=0. В этом смысле домены образующих g_i, i=1, 2, ..., n аналогичны доменам реляционной модели Кодда, которые также содержат значение λ=0. Образующая, всем компонентам которой присвоены значения λ=0, называется *неассоциированной* (пустой). Если компонентам образующей присвоены ненулевые значения из соответствующих доменов, то она называется *ассоциированной* образующей.

Если домены всех компонент образующей g_i содержат значение $\lambda=0$ и не имеют в своем составе других значений, то такая образующая не определена на какой-либо информационной среде. Очевидно, что у любой абстрактной образующей домены всех ее компонент можно объединить в один домен, содержащий единственное значение $\lambda=0$. Если домены образующей g_i содержат, помимо $\lambda=0$, другие значения, то такая образующая называется *конкретной*. Конкретная образующая определена на некоторой "не пустой" информационной среде. Абстрактная образующая "привязывается" к конкретной информационной среде путем "наполнения" ее доменов данными этой среды.

Образующие не только представляются векторами признаков, но и изображаются наглядными схемами, отчасти напоминающими схемы графов. Подобно дугам графов, связи образующих могут быть ориентированными и неориентированными. На рис. 1, *а, б* показаны схемы двух линейных абстрактных образующих — ориентированной и неориентированной. На рис. 1, *а* показана схема линейной ориентированной образующей, ее домены и вектор признаков. Центральной точке схемы приписаны символы i, α и компонента γ_{i1}, которые обозначают порядковый номер, индекс класса и атрибут образующей g_i. Входная связь образующей изображена треугольником и стрелкой, направленной к точке, а выходная связь — стрелкой, направленной от точки, и треугольником. Стрелки и треугольники не отделимы от точки. Связи образующей имеют на рис. 1, *а* показатели связей β_{i1}^{in}, β_{i1}^{out}, которым соответствуют домены D_{i1}^{in}, D_{i1}^{out}. Атрибуту соответствует домен D_{i1}. Показанный на рисунке вектор признаков представляет собой форму абстрактной линейной ориентированной образующей, домены которой содержат единственное значение $\lambda=0$. Домены конкретной образующей содержат также другие значения, присваиваемые показателям связей и атрибутам образующей, и служат как бы хранилищем содержания конкретной образующей. В зависимости от конкретных значений, находящихся в доменах, образующая представляет те или иные объекты реального мира. Например, схемой на рис. 1, *а*, при $i=2$, можно изобразить видимую на экране дисплея компьютера вторую строку текста, состоящего из 15 строк. В этом случае домены D_{21}^{in}, D_{21}^{out} будут содержать значения — "начало строки 2", "конец строки 2", а домен D_{21} — значение "строка 2". Если домены дополнить хранящимися в памяти компьютера кодами этих значений, то схема на рис. 1, *а* изобразит видимую на экране строку и ее кодовое представление в памяти. Форме образующей в данном случае соответствуют два хранящиеся в доменах набора содержания. Если положить $i=1, 2, \ldots, 15$ и поместить в доменах информацию о всех 15 строках текста, то образующая g_i, со схемой на рис. 1, *а* будет служить паттерновой моделью 15 строк текста. Схема на рис. 1, *а* может представлять другие реальные объекты с двумя ориентированными связями, например, пассажирский вагон $N2$, на торцах которого закреплены две автосцепки для его соединения с вагонами $N1$ и $N3$. В этом случае домены образующей будут содержать информацию о вагоне и его автосцепках. Схема на рис. 1, *б* может изображать, например, атом кислорода, имеющий, как известно, валентность равную двум. Две неориентированные связи схемы изобразят двукратную валентность атома кислорода. Эти примеры показывают, что форма образующей остается постоянной, а ее содержание можно изменять.

Схема на рис. 1, *в* изображает образующую g_i с двумя входными и тремя выходными связями (домены не показаны). Она может представлять, например, сведения о человеке (i — номер человека в множестве n людей). В этом случае атрибуты γ_{il}, $l=1, 2, 3$ ассоциируются с хранящимися в доменах индивидуальными признаками человека, такими как "имя и фамилия", "возраст", "пол"; показатели входных связей β_{i1}^{in}, β_{i2}^{in} — с признаками "сотрудник Сбербанка", "член правления Сбербанка", показатели выходных связей β_{i1}^{out}, β_{i2}^{out}, β_{i3}^{out} — с признаками "женат", "отец", "владелец автомобиля". Связи образующей, моделирующей набор сведений о человеке, могут попарно соединяться со связями образующих, моделирующих его окружение (Сбербанк, жена, дети).

Образующая анализа обладает одной входной и двумя или большим числом выходных связей (рис. 1, *г*). Основная входная связь с показателем β_{i1}^{in} может иметь копии с показателями $\beta_{i,1,k}^{in}$, где $k=1, 2, \ldots$ Основной связи и ее копиям соответствует один и тот же домен D_{i1}^{in}. Входная связь образующей анализа ассоциируется с каким-либо общим понятием (например, ЦВЕТ), а выходные — с частными понятиями (например КРАСНЫЙ, ЗЕЛЕНЫЙ, СИНИЙ цвет). Образующая анализа может представлять, в частности, окно главного меню системы Windows. Тогда входная связь образующей анализа ассоциируется с окном в целом, к примеру, с его идентификатором, а выходная связь — со ссылками (кнопками) окна, соединяемыми в памяти компьютера с другими окнами системы Windows. Копиями входной связи можно моделировать несколько входов в окно главного меню системы Windows. Образующими анализа моделируются страницы электронных книг, темы гипертекстов/гипермедиа и многие другие компьютерные объекты.

Образующая синтеза обладает одной выходной и двумя или большим числом входных связей (Рис. 1, *д*). Основная выходная связь с показателем β_{i1}^{out} может иметь копии с показателями $\beta_{i,1,k}^{out}$. Входная связь образующей синтеза ассоциируется с общим понятием, а входные связи — с частными понятиями. Образующая синтеза может моделировать нейрон мозга. В таком случае i — номер нейрона среди n нейронов, α — класс однородных нейронов, входные связи образующей представляют собой дендриды нейрона и их синапсы, выходная связь с копиями — аксон и его синапсы. Образующая синтеза моделирует также операционные усилители нейрокомпьютеров.

На рис. 1, *е, ж* показаны образующие, называемые начальной и конечной (терминальной). Конкретная терминальная образующая может моделировать терминальную тему гипертекста, конкретная начальная образующая — первую его тему. Образующие являются открытыми объектами, поскольку их связи готовы к соединению со связями других родственных образующих.

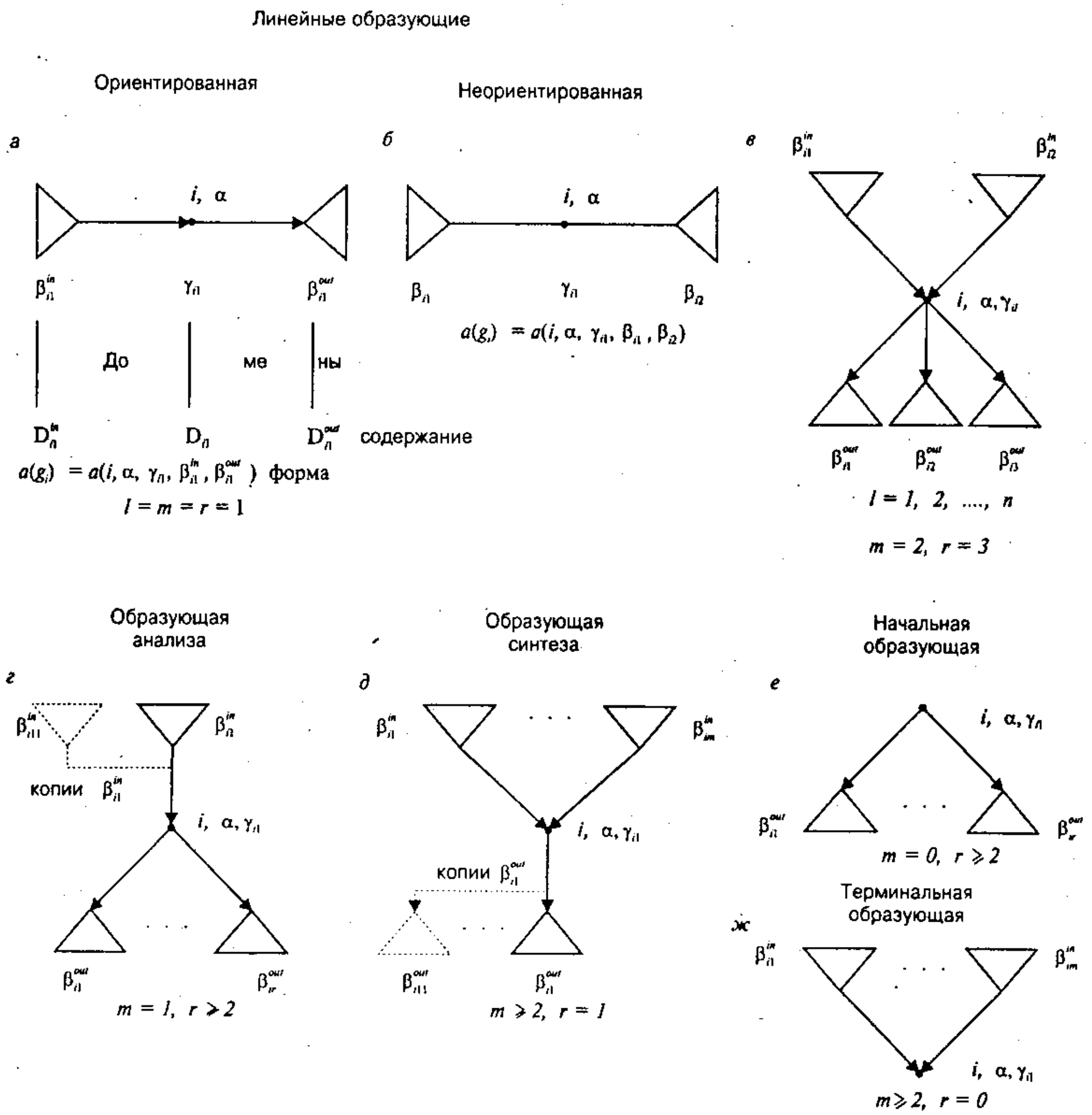


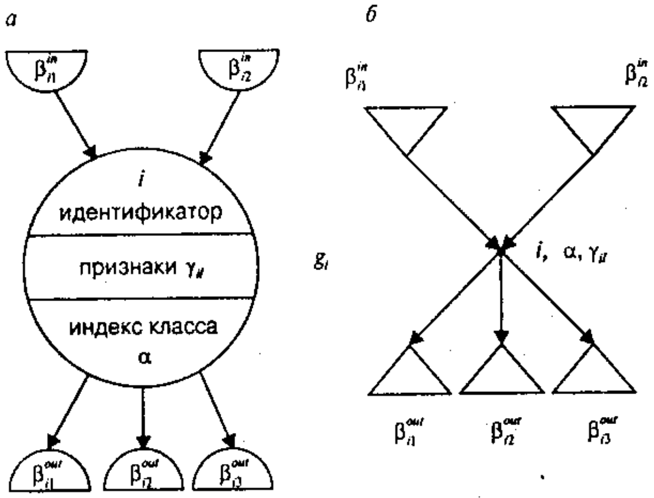
Рис. 1

Теория паттернов рассматривает образующие как условно атомарные элементы (объекты). Это означает, что части, из которых состоит образующая (идентификатор, индекс класса, связи), в определенном контексте или промежутке времени, считаются не отделимыми друг от друга, т. е. составляют единое целое. Принцип условной атомарности образующих отражает свойства объектов реального мира. Действительно, валентности не отделимы от атомов химических элементов, дендриты и аксоны с синапсами и ядро нейрона составляют единое целое. Признаки, характеризующие человека (возраст, пол) и его связи (женат, отец, сотрудник банка), трактуются в базах данных как его неотъемлемые компоненты и потому могут быть представлены образующей. Однако, если человек разведется с женой, то образующая, моделирующая его персональные данные, потеряет связь "женат". Чтобы пояснить свойство неделимости образующих, на рис. 2 показаны две схемы образующей с двумя входными и тремя выходными связями. На первой схеме (рис. 2, a) стрелки и полукруги не отделимы от круга, а показанные внутри окружности идентификатор i, индекс класса и признаки составляют единое целое со связями образующей. На второй схеме (более простой и практичной) стрелки не отделимы от точки, которая изображает образующую "в целом".

Путем попарного соединения связей образующих конструируется конфигурация теории паттернов, обозначаемая символом c. Конфигурацию определяют ее состав, домены и пары соединенных показателей связей образующих. Состав конфигурации — это набор всех входящих в нее образующих, обозначаемый "состав c"$=g_1, g_2, \ldots, g_n$. Очевидно, что "состав c" задается набором векторов признаков образующих. Доменами конфигурации являются все домены образующих из состава c. На всех парах показателей соединенных связей образующих из состава конфигурации c устанавливается одно и то же отношение связей, обозначаемое ρ-соединено и принимающее значения "истина" или "ложь" (условие соединения связей образующих).

Два показателя связей образующих, на которых установлено отношение ρ-соединено, называются *связка*. В общем случае связка двух показателей ориентированных связей обозначается:

$$\beta_i^{out} \rho \beta_j^{in}. \tag{2}$$



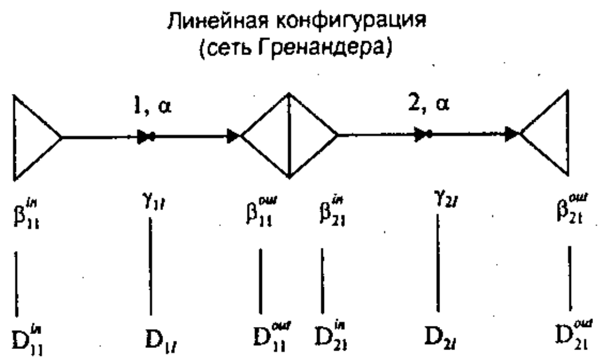
$a(g_i) = a(i, \alpha, \gamma_{ii}, \beta^{in}_{i1}, \beta^{in}_{i2}, \beta^{out}_{i1}, \beta^{out}_{i2}, \beta^{out}_{i3})$

Рис. 2

С помощью термина *связка* приведенное выше условие соединения связей образующих формулируется кратко: "На всех связках конфигурации c установлено отношение связей ρ-соединено". В общей теории паттернов отношение связей, установленное на связках образующих, ассоциированных с данными, может иметь различные значения, например, ρ-равенство, ρ-больше, ρ-меньше. В конфигурациях, моделирующих компьютерные системы широко, используется отношение связей ρ-соединено, установленное на связках неассоциированных образующих. Конфигурации представляются наглядными схемами. Рис. 3, *а* показывает схему простой конфигурации, состоящей на двух линейных ориентированных образующих g_1, g_2. На рисунке указаны домены конфигурации. Конфигурация имеет одну связку, на которой установлено отношение связей ρ-соединено. В зависимости от значений, содержащихся в доменах, конфигурация со схемой на рис. 3*а* представляет различные взаимосвязанные объекты, например, две последовательные текстовые строки, видимые на экране дисплея, или две соединенные темы гипертекста. Конфигурация может иметь много образующих и связок. На рис. 3*б* показана схема конфигурации со структурой типа дерева, состоящая из трех ориентированных образующих (домены не показаны, чтобы не загромождать схему). Она может представлять, например, фрагмент системы оказания помощи пользователям компьютерной программы. В этом случае образующие g_1, g_2, g_3 моделируют темы, а их связи имитируют ссылки на другие темы системы помощи. Конфигурации, имеющие внешние связи, позволяют моделировать реальные открытые системы.

В инженерной практике используются различного рода сети — графовые, массового обслуживания, сети Петри и другие. Термин *сеть* удобен, поскольку он создает понятный всем зрительный образ. Поэтому множество конфигураций теории паттернов, на всех связках которых установлено отношение связей ρ-соединено, назовем *сетями Гренандера.*

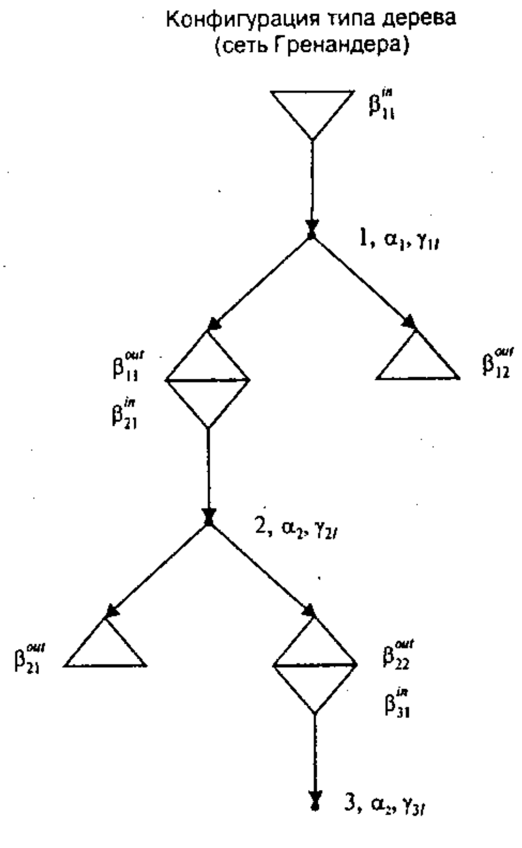
Окончание см. на с. 18



содержание
состав = g_1, g_2
связка: $\beta^{out}_{11}\ \varrho\ \beta^{in}_{21}$, ϱ - соединено

Показатели Внешних связей: β^{in}_{11} , β^{out}_{21}

а



состав = g_1, g_2, g_3
связки: $\beta^{out}_{11}\ \varrho\ \beta^{in}_{21}$, $\beta^{out}_{22}\ \varrho\ \beta^{in}_{31}$
ϱ - соединено

б

Рис. 3