

О ТЕОРИИ МОДЕЛИРОВАНИЯ И ГИПЕРГРАФЕ КЛАССОВ

И.В. Волович, М.Н. Хохлова

Математический институт им. В.А. Стеклова РАН,
119991, Москва, ул. Губкина, 8
e-mail:volovich@mi.ras.ru

Компания "Цефей"
115054, Москва, ул. Щипок, д.11, стр. 1
e-mail:info@cefey.ru

(30.12.2003)

Аннотация

Представлена формулировка новой теории моделирования, предназначенной для эволюционного проектирования моделей широкого класса сложных, открытых, динамических, сетевых, иерархических систем. Эта формулировка основана на идее гиперграфа классов как единой модели, предложенной ранее одним из авторов и эффективно использованной при разработке Эталон-CASE-технологий и корпоративных систем управления. Классы и объекты представляются конечными автоматами. Теория усовершенствует концепции моделирования, разработанные в общей теории систем, в теории автоматов, в объектно-ориентированном проектировании и при построении универсального языка моделирования UML.

1. Введение

Моделирование используется для решения задач исследования, анализа, планирования, управления в самых разнообразных областях деятельности. В последние десятилетия моделирование стало важной составной частью проектирования программного обеспечения, основанного на CASE - технологиях [1,2]. Хорошо известны структурный и объектно-ориентированный подходы к моделированию, в которых используется нотация диаграмм потоков данных и "сущность-связь" для первого подхода и диаграммы вариантов использования, диаграммы классов и др. для второго подхода. Активно продвигаются такие средства моделирования как Silverrun, Oracle Designer, Rational Rose, Microsoft.NET и др. и такие технологии как DATARUN, RUP, AgentBuilder и т.д. Широкую известность получил универсальный язык моделирования (UML), использующий принципы объектно-ориентированного проектирования [3,4]. Однако указанные методы моделирования обладают серьезными недостатками, в частности отсутствуют точные формулировки основных понятий и единая модель системы, см. например, [5].

В настоящей работе предлагается формулировка новой теории моделирования, предназначенной для построения моделей широкого класса сложных, открытых,

динамических, сетевых, иерархических систем. Эта формулировка основана на идее гиперграфа классов как единой модели, предложенной ранее одним из авторов (М.Хохловой) и эффективно использованной при разработке Эталон-CASE-технологий эволюционного проектирования моделей сложных, открытых, динамических, слабо-детерминированных систем и корпоративных систем управления [6,7]. Классы и объекты представляются конечными автоматами. Теория усовершенствует концепции моделирования, разработанные в общей теории систем, в теории автоматов, в объектно-ориентированном проектировании и при построении универсального языка моделирования UML. Авторы надеются, что данная работа будет полезной, в частности при разработке усовершенствованных, более семантически точных версий UML, см. [8,9].

В отличие от UML, где нет единой модели системы, новая теория моделирования строит единую модель. Моделирование в предлагаемой теории осуществляется на основе фиксированного набора базовых классов, замены итерационного проектирования на эволюционный и усовершенствованного диаграммного представления. Моделью системы является гиперграф классов и определяемый им конечный автомат.

Авторы предпринимали попытки использовать методы р-адического анализа [10] в теории управления и моделирования. Эти методы адекватны для описания сложных иерархических систем. Однако в процессе работы стало ясно, что сначала требуется дать надлежащую математическую формулировку методов моделирования, используемых в реальных современных системах управления, широко используемых на практике, таких как Эталон, а затем уже для анализа построенных моделей можно будет использовать более тонкие методы р-адического анализа. В данной работе мы ограничиваемся формулировкой основ новой теории моделирования.

Модель системы, описанная в настоящей работе, может быть связана с понятием управляющей системы в смысле [11] и теорией информационного поиска [12]. В управляющей системе выделяются четыре основных звена: схема, информация, координаты и функция. *Схема* управляющей системы представляет собой некоторое соединение элементов, каждый из которых связан с заданной памятью, образуя в ней элементарные подсхемы. *Информация* управляющей системы задается состояниями памяти, принимаемыми из некоторого конечного множества. *Координаты* элементов схемы характеризуют расположение схемы. *Функция* управляющей системы определяет ее преобразования, включающие возможные изменения схемы, информации, координат и поведения.

В следующих разделах рассматриваются сначала эвристическое описание гиперграфа классов. Затем приводятся сведения из теории конечных автоматов и формальных языков, необходимые для формулировки гиперграфа классов, а также уравнения балансов для аддитивных благ. Далее дается точное определение и поясняется функционирование основного понятия данной работы - гиперграфа классов, лежащего в основе новой общей теории моделирования.

2. Гиперграф классов. Общее описание

В объектно-ориентированном проектировании основными являются понятия класса и объекта [1,2,3]. Класс задается совокупностью свойств членов класса - описаний некоторых сущностей и методов их изменения. Формально - это множество, состоящее из пары подмножеств. Одно из подмножеств называется переменными, другое - методами их изменения. Отношение - это связь между классами, отражающая различные принципы взаимодействия классов в моделях. Приведем здесь эвристическое описание гиперграфа классов системы Эталон. Ниже будет дана точная формулировка. Мы различаем *структуру, функционирование, проектирование и сложность* гиперграфа классов и соответствующего конечного автомата.

Приведем сначала структурное описание гиперграфа классов. Любой гиперграф (эквивалентный термин - сеть) задается множеством V , элементы которого называются вершинами, и семейством E наборов элементов множества V , называемых ребрами гиперграфа. Гиперграф обозначается $\langle V, E \rangle$. Модель системы в ОТМ задается следующим гиперграфом классов $X \langle V, E \rangle$. Вершины, образующие множество V - это классы, соответствующие элементам модели. Базисные классы описаны ниже. Любой класс имеет один и только один из трех указанных ниже типов. Ребра, т.е. наборы вершин, образующие семейство E - это отношения (связи) перечисленных ниже семи видов между элементами модели. Классы имеют один из следующих трех типов:

- 1) содержание (база данных, хранилище, таблица,...),
- 2) форма (формы ввода/корректировки, отчеты, документы,...),
- 3) поведение (процедуры, алгоритмы, ...).

Ребра гиперграфа - это отношения классов в модели, отражающие наследование и другие связи, см. ниже.

Гиперграф содержит следующий важный подграф. Если рассмотреть его вершины и для них только отношения наследования, то будет получено дерево наследования. Это дерево используется для эволюционного проектирования гиперграфа классов.

3. Конечные автоматы, классы, формальные языки

Мы будем представлять класс как конечный автомат.

Конечный автомат A [15,16] задается следующим набором данных: $A = \{X, Q, Y, f, g\}$. Здесь X - множество (алфавит) входных сигналов, Y - множество выходных сигналов, Q - множество состояний автомата, отображение f из $Q \times X$ в Y - функция выхода, отображение g из $Q \times X$ в Q - функция перехода. Предполагается, что дискретная временная динамика конечного автомата задается следующим образом:

$$y(t) = f(q(t), x(t)), \quad q(t) = g(q(t-1), x(t)), \quad t = 0, 1, 2, \dots \quad (1)$$

где $q(t) \in Q$, $x(t) \in X$, $y(t) \in Y$.

Геометрически конечный автомат задается ориентированным графом. Вершины - состояния автомата; около каждого ребра (q_i, q_j) ставится буква входного алфавита x_k , вызывающая переход из состояния q_i в q_j (т.е. $q_j = g(q_i, x_k)$) и буква выходного алфавита y_s , которая появляется на выходе конечного автомата.

При этом принципы ООП такие как инкапсуляция, наследование и

полиморфизм допускают естественную интерпретацию. Пространство состояний Q конечного автомата $A = \{X, Q, Y, f, g\}$, представляющего класс, состоит из трех частей: $Q = Q_1 \cup Q_2 \cup Q_3$, где подмножество Q_1 называется *именем*, Q_2 *переменными* и Q_3 *методами*.

Принцип *инкапсуляции* заключается в том, что определенный уровень проектирования и функционирования гиперграфа классов осуществляется на уровне имен классов Q_1 .

Принцип *наследования* заключается в том, что если два класса, представленные автоматами $A = \{X, Q, Y, f, g\}$ и $A' = \{X', Q', Y', f', g'\}$, принадлежат одному ребру наследования, то пространство состояний Q' получается из Q добавлением некоторых новых элементов: $Q' = Q \cup Q'$.

Наконец, *полиморфизм* означает просто, что при подаче на вход автомата одного и того же сигнала на выходе могут появляться различные сигналы в зависимости от того в каком состоянии находится автомат.

Три типа классов на языке конечных автоматов имеют следующий смысл. Содержание означает абстрактное задание конечного автомата, форма - его конкретное представление, поведение - формальное описание формирования сигнала на выходе при подаче на вход достаточно длинных слов.

Для описания структуры гиперграфа классов понадобятся некоторые сведения из теории формальных языков. Если A - алфавит (конечный набор символов), то обозначим A^* множество всех слов в A . Слово - это последовательность символов из A . *Языком* L называется подмножество $L \subset A^*$. Предложение в языке - это цепочка слов, т.е. тоже слово. Грамматика языка - это набор правил вывода вида $A \rightarrow B$, где A, B - слова. Правило вывода применимо к слову P , если в P имеется хотя бы одно подслово, совпадающее с A ; тогда A в P заменяется на B . Множество всех полученных таким образом слов образует язык, порожденный грамматикой. Если правила вывода имеют вид $O \rightarrow A$, где O - вспомогательное слово, а A - слово, то грамматика называется контекстно-свободной, [16].

Если при подаче на вход конечного автомата любого слова из языка L и только из L автомат переходит в одно из фиксированного множества финальных состояний, то говорят, что язык L *представим* данным автоматом. Большинство языков не представимы конечными автоматами. Всякий конечный язык представим конечным автоматом.

4. Уравнения балансов

В экономике и управлении имеется основное уравнение материального баланса, см. [13,14]. Предполагается, что имеются некие субъекты (они образуют множество S и называются агентами), воздействующие на некоторые блага (goods). Агенты - это не обязательно люди, это могут быть организации, госорганы, компьютеры и т.д. Блага могут включать продукты, услуги (включая услуги по управлению), ресурсы. Блага образуют множество M . В каждый момент времени t в распоряжении агента $s \in S$ находится запас i -го блага $G_i^s(t)$, здесь $i \in M$.

Мы предполагаем здесь, что блага описываются числовыми параметрами, т.е. $G_i^s(t)$ рациональное число. В дальнейшем мы планируем модифицировать это описание и описывать блага также векторными и более общими величинами в соответствии с подходом, изложенным в [10].

Простейшее уравнение баланса имеет вид

$$\Delta G_i^s = J_i^s, \quad i \in M, s \in S \quad (2)$$

где $\Delta G_i^s = G_i^s(t) - G_i^s(t-1)$ - изменение запаса благ за единицу времени, а J_i^s соответствующие потоки благ за единицу времени,

$$J_i^s = D_i^s - C_i^s - Z_i^s - K_i^s - \sum_{r \in S} (R_i^{sr} - R_i^{rs}) \quad (3)$$

Здесь D_i^s - выпуск блага i у агента s , C_i^s - потребление блага i агентом s , Z_i^s - текущие затраты, K_i^s - капитальные затраты, R_i^{sr} - передача блага i от агента s агенту r . Предполагается, что все величины берутся в один и тот же момент времени t , который мы не указываем явно.

Отметим существенную идеализацию в уравнениях баланса (2),(3) – отсутствие взаимодействия по индексу благ i , хотя ясно, что производство и потребление различных благ влияют друг на друга. Мы надеемся вернуться к этому вопросу в дальнейшем.

Уравнения баланса выражают свойство *аддитивности* при передаче благ – сколько один агент отдал, столько другой агент получил. Уравнение баланса позволяет описывать (моделировать) изменение аддитивных благ или услуг, однако оно не дает возможности описывать общественные блага (безопасность, образование, экологические условия,...), а также знания и информацию и не всегда адекватно описывает процессы управления. В частности, уравнение баланса не позволяет адекватным образом описывать некоторые важные характеристики государства и других агентов, производящих неаддитивные блага.

Ниже при построении гиперграфа классов понятие баланса будет использоваться не для всех классов, а только для классов, отвечающих аддитивным благам.

В экономике важным средством оценки благ, приведенной к единой аддитивной единице измерения, являются деньги. В идеализированной теории предполагается, что передачам благ R_i^{sr} отвечает встречный поток денежных платежей H_i^{sr} . Возможны, однако, также обмен одних благ на другие (бартерный обмен) и задержка платежей. Коэффициент пропорциональности между потоком платежей и встречным потоком блага называется ценой. Обычно предполагается [13], что в каждый момент времени существуют единые для всех агентов цены, $H_i^{sr} = p_i R_i^{sr}$, где цена p_i не зависит от индексов s, r . Исходя из основных уравнений материального баланса микроэкономики (2) можно получить уравнения макроэкономического баланса, межотраслевого баланса и финансовых балансов.

5. Гиперграф классов

Дадим теперь математическое определение гиперграфа классов, используемого в предлагаемой теории моделирования. Напомним сначала некоторые стандартные определения. Все рассматриваемые множества предполагаются конечными. Множество $V = \{v_1, v_2, \dots\}$ и семейство $\{E_1, E_2, \dots\}$, в котором каждое E_i есть набор (возможно с повторениями) элементов из V , т.е. $E_i = \{v_{s_1(i)}, v_{s_2(i)}, \dots\}$, называется *гиперграфом* и обозначается $\langle V; E_1, E_2, \dots \rangle$. Элементы множества V называются *вершинами*, а наборы E_1, E_2, \dots – *ребрами* гиперграфа.

Частным случаем гиперграфа является граф. *Графом* называется пара $\langle V; E \rangle$,

где V множество, а набор E состоит из двух элементов множества V . Про ребро (v_i, v_j) говорят, что оно *соединяет* вершины v_i и v_j . Система ребер $\{(v_i, v_j), (v_j, v_k), (v_k, v_l), \dots, (v_m, v_n)\}$ называется *путем*, соединяющим вершины v_i и v_n . Путь, не содержащий дважды одно ребро, называется *циклом*, если $v_i = v_n$. Граф называется *связным*, если любые две его вершины соединены некоторым путем. *Деревом* называется связный граф, не содержащий циклов, с выделенной вершиной, именуемой *корнем*.

Опишем сначала структуру гиперграфа классов, а затем его функционирование.

Определение. *Гиперграфом классов* называется гиперграф, снабженный дополнительными структурами, задаваемыми следующим образом. Пусть задано дерево $\langle V; T_1, T_2, \dots \rangle$, где каждое T_i есть пара элементов из $V = \{v_1, v_2, \dots\}$. Задаются связи (ребра) E_1, E_2, \dots , определяющие гиперграф $\langle V; T_1, T_2, \dots, E_1, E_2, \dots \rangle$. Ребра E_i описаны ниже. На множестве вершин V задается функция F со значениями в $\{1, 2, \dots, k\}$ так что каждая вершина v_i имеет один из k типов $F(v_i)$. (В системе Эталон $k = 3$). Вершинам v_i сопоставляются *классы* различных типов, описанные ниже. Классы описываются конечными автоматами. Гиперграфу классов описанным ниже способом ставится в соответствие конечный автомат.

Дерево $\langle V; T_1, T_2, \dots \rangle$ называется деревом *наследования*.

Моделью системы будем называть гиперграф классов и определяемый им конечный автомат. Структура гиперграфа и набор классов и связей отвечают соответствующей системе.

6. Базисные классы

Вместо понятий агент и благо мы будем пользоваться более общими понятиями *сущностей* и классов, как это принято в ООП. В предлагаемой теории моделирования в соответствии с системой Эталон выделены следующие базисные классы:

- 1) универсальный класс,
- 2) кодификаторы,
- 3) нормативы,
- 4) картотеки,
- 5) регистраторы,
- 6) движения состояний,
- 7) балансы,
- 8) прогнозы,
- 9) показатели.

Напомним, что класс (сущность) описывается конечным автоматом, который имеет состояние и перерабатывает входные сигналы в выходные.

Универсальный класс служит для обобщенного представления в модели непознанных на момент проектирования, неспецифицированных сущностей.

Кодификатор - это глоссарий используемых понятий. Он представляет некоторый конечный язык.

Норматив - набор параметров, фиксированных на определенный период времени.

Картотека - энциклопедия используемых сущностей.

Регистратор - осуществляет регистрацию, учет и хранение информации о входных и выходных сигналах конечных автоматов (сущностей, классов).

Движение состояний - осуществляет регистрацию, учет и хранение информации об изменениях состояний конечных автоматов (сущностей, классов).

Баланс - служит для формирования баланса аддитивных сущностей.

Прогноз - набор математических методов предсказания и прогнозирования будущих изменений сигналов и состояний классов.

Показатели - интегральные характеристики, описывающие изменения сигналов и состояний классов

Приведенные базисные классы могут задаваться как формальные языки. Им сопоставляются конечные автоматы, которые решают задачи распознавания соответствующих языков.

В терминологии конечных автоматов множество Q_1 (имя класса) всегда имеет следующий вид: $A_1/A_2/.../A_i/...$, где A_i одно из слов уровня наследования классов.

В частности, A_1 одно из трех слов (содержание, форма, поведение), обозначающих тип класса, A_2 – название одного из девяти базисных классов.

Опишем кратко как задаются ребра (связи) классов E_i в гиперграфе и как функционирует соответствующая модель. Ребра гиперграфа - отношения (связи) классов в модели следующих видов: 1) Наследование. 2) Связи классов типа содержание. 3) Связи классов типа поведение. 4) Связи классов типа форма. 5) Контейнеры проектирования. 6) Контейнеры использования. 7) Контейнеры управления событиями.

Наследование было описано выше.

Моделирование в предлагаемом подходе заключается в эволюционном проектировании конечных автоматов с использованием принципа наследования и в *синтезе* [15] спроектированных конечных автоматов с использованием связей вида 2)-7).

Функционирование модели как конечного автомата осуществляется при помощи контейнеров использования, проектирования и управления событиями, с которых производятся возможные изменения состояний конечных автоматов и входных и выходных сигналов. Открытость системы отражается в том, что функции, описывающие возможные изменения автоматов, могут зависеть от параметров, соответствующих окружающей среде. Более подробное рассмотрение будет дано в отдельной работе.

В настоящей работе обсуждалось моделирование *классических* систем при помощи детерминированных конечных автоматов и гиперграфов классов. Более общий подход к моделированию классических систем представляют *вероятностные* автоматы, которые описываются не функциями перехода, а вероятностями перехода [17]. Для моделирования *квантовых* систем [18, 19] следует использовать квантовые автоматы, которые описываются амплитудами перехода и соответствующими квантово-механическими правилами сложения амплитуд.

Список литературы:

- [1] А. Элиенс. Принципы объектно-ориентированной разработки программ. М.: Изд. дом "Вильямс", 2002
- [2] Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Питер, 2001

- [3] Джеймс Рамбо, Айвар Якобсон, Грэди Буч. UML. Специальный справочник. Питер, 2002
- [4] Unified Modeling Language Specification. <http://www.uml.org>.
- [5] A. J. H. Simons and I. Graham "37 Things that don't work in object modelling with UML", Proc. 2nd. ECOOP Workshop on Precise Behavioural Semantics, pub. TUM-I9813, eds. H Kilov and B Rumpe (Brussels : TU Munich, 1998), 209-232
- [6] М.Н. Хохлова. Свидетельство об официальной регистрации финансово-экономического комплекса "Эталон". РосАПО, N 950380, 1995 г.
- [7] Эталон. Информационная система управления ресурсами предприятия. Руководство системного аналитика. Компания "Цефей", 1992
- [8] The precise UML group. <http://www.cs.york.ac.uk/puml/maindetails.html>
- [9] S.J. Mellor, J.S. Fontaine, I. Wilkie, and D.E. Statezni, An Approach to Adding Precision and Clarity to UML.
- [10] В.С. Владимиров, И.В. Волович, Е.И. Зеленев. р-Адический анализ и математическая физика. Наука, 1994.
- [11] С.В. Яблонский, Проблемы кибернетики, в.2, с.7-38, 1959
- [12] Э.Э. Гасанов, В.Б. Кудрявцев. Теория хранения и поиска информации. М.:Физматлит, 2002
- [13] И.Г. Поспелов. Моделирование экономических структур. М.:Фазис, 2003
- [14] П.С. Краснощеков, А.А. Петров. Принципы построения моделей. М.:Фазис, 2000
- [15] В.М. Глушков. Синтез цифровых автоматов. М.: Физматгиз, 1962
- [16] Д. Хопкрофт, Р. Мотвани, Д. Ульман. Введение в теорию автоматов, языков и вычислений. М.: Изд. дом "Вильямс", 2002
- [17] Р.Г.Бухараев. Основы теории вероятностных автоматов. М.: Наука, 1985
- [18] P.A.M. Dirac. The Principles of Quantum Mechanics. Oxford Univ.Press, 1930
- [19] I.V. Volovich. Seven Principles of Quantum Mechanics. <http://www.arxiv.org/abs/quant-ph/0212126>.