

УДК 004.94

МОДЕЛИРОВАНИЕ ИЗМЕНЧИВОСТИ В ВИРТУАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ СРЕДЕ С ПОМОЩЬЮ МЕТОДОВ ТЕОРИИ ГРАФОВ И ГИПЕРГРАФОВ

Евсеева Ю.И.

ФГБОУ ВПО «Пензенский государственный университет», Пенза, e-mail: shymoda@mail.ru

Вопросы моделирования изменчивости в виртуальных образовательных средах – достаточно актуальная тема в настоящее время. Разработка соответствующего математического и программного обеспечения открывает возможности для серийного создания обучающих систем, способных быстро адаптироваться к индивидуальным особенностям учащихся, а также к изменениям образовательных стандартов и программно-аппаратных возможностей. В статье предлагается математическая модель, которая позволит наглядно описать, структурировать и алгоритмизировать основные процессы изменчивости обучающих систем в виртуальной образовательной среде. Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 15-07-01553. Для решения поставленной задачи был применен математический аппарат теории графов, в частности, гиперграфовое представление структуры программного обеспечения виртуальной образовательной среды. В результате был предложен способ формализации структурной и параметрической изменчивости типового интерактивного обучающего приложения на уровне теоретико-множественного представления. Разработана математическая модель изменчивости в виртуальной образовательной среде. Предложенная математическая модель позволяет синтезировать адаптивное программное обеспечение (ПО), способное менять свой функционал без перекомпиляции исходного кода в зависимости от текущих требований со стороны обучаемых, работодателей, а также учитывать современное состояние развития науки и техники. Использование такого ПО позволит в целом существенно повысить качество и эффективность работы систем электронного и дистанционного образования.

Ключевые слова: адаптивное обучающее программное обеспечение, теория графов, ориентированный гиперграф, автоматизированное проектирование, моделирование изменчивости, виртуальная образовательная среда

VARIABILITY MODELING IN THE VIRTUAL LEARNING ENVIRONMENT USING METHODS OF GRAPHS AND HYPERGRAPHS THEORY

Evseeva Yu.I.

*Federal state budgetary educational institution of Higher Education Penza State University, Penza,
e-mail: shymoda@mail.ru*

The question of creating mathematical and algorithmic foundation of virtual learning environment adaptive software is quite relevant now. Resolving this issue will allow to develop training programs that can adapt to the individual needs of students and changes in the external (eg, hardware) environment. The aim of this work to create a mathematical model that will describe the basic variability process of learning software in a virtual learning environment in a structured form suitable for further processing. To solve this problem author used the mathematical apparatus of graph theory, in particular, the hypergraph representation of the virtual educational environment software structure. Author developed a mathematical model of interactive learning applications, used as a major component of the virtual educational environment. Developed mathematical model can effectively describe the basic variability process of learning software in a virtual learning environment. This model can be used in the implementation of multi-loop algorithms of educational software adaptation.

Keywords: adaptive educational software, graph theory, oriented hypergraf, computer-aided design, variability modeling, virtual learning environment

Современная виртуальная образовательная среда – это быстроразвивающаяся, многоуровневая и многофункциональная система, объединяющая педагогические, дидактические и методические технологии; информационные ресурсы, включая базы данных и знаний, электронные учебные материалы и т.д.; современные программные средства, включающие программные оболочки и средства электронной коммуникации. В качестве компонента такой среды могут выступать различные тестирующие приложения, программы-тренажеры, в т.ч. использующие различные передовые техно-

логии, такие как трехмерная компьютерная графика, беспроводные коммуникации, мобильные программно-аппаратные платформы. Значительная часть таких приложений используют различные стратегии адаптации к пользователю, проходящему обучение. Однако в ряде случаев может быть необходимо определение многоконтурных обратных связей, обеспечивающих возможность адаптации по ряду различных показателей. Для реализации возможности эффективного и быстрого определения таких обратных связей необходима разработка соответствующей математической модели.

Структура адаптивной обучающей программы в формате модели характеристик

В качестве основного инструмента при создании математической модели предлагается использовать методы морфологического анализа и синтеза. Данные методы широко применяются в инженерной практике для моделирования сложных систем в условиях, когда выбор оптимальной структуры осложняется неопределенностями поведения и состояния окружающей среды [1].

Результатом выполнения морфологического анализа является морфологическое множество, называемое также множеством альтернатив, содержащим в себе описание всех возможных вариантов системы. В процессе морфологического синтеза осуществляется поиск на морфологическом множестве оптимальной структуры синтезируемого объекта.

Наиболее часто для проведения процедур морфологического анализа и синтеза используются И/ИЛИ-деревья. Одной из основных сфер применения описанных подходов является разработка различных технических систем.

Основная идея подхода, предлагаемого в данной работе, заключается в применении методов морфологического анализа и синтеза для разработки математического и алгоритмического обеспечения адаптивных обучающих программ, являющихся компонентами виртуальной образовательной среды. В данном случае предлагается использовать морфологическое множество не для синтеза конечных вариантов системы (как в случае с техническими системами), а для определения отдельных состояний функционирования одной программной системы. Адаптивность такой системы будет заключаться в том числе и в возможности синтеза состояний в процессе выполнения программы.

Основные компоненты программы и взаимосвязи между ними при использовании предлагаемого подхода будут описываться как морфологическое множество. Данное морфологическое множество будем называть общей структурой программы. Для моделирования общей структуры будем использовать не стандартные И/ИЛИ-деревья, а модели характеристик [2, 3]. Модель характеристик (или диаграмма характеристик) представляет собой усовершенствованное И/ИЛИ-дерево, в котором, помимо стандартных отношений между вершинами по типу И/ИЛИ, предусмотрены также отношения множественного ИЛИ, обязательно-

го включения дочерней вершины, произвольного включения дочерней вершины, а также отношения, связывающие между собой вершины не являющиеся дочерними или родительскими по отношению друг к другу (отношение требования посторонней вершины и отношение исключения посторонней вершины).

Для задания морфологического множества разработчик должен определить общую структуру программы в формате диаграммы характеристик. Конкретные же состояния работы программы будут определяться по диаграмме характеристик как вручную пользователем, так и автоматически в процессе выполнения на основе имеющихся обратных связей.

Процедура морфологического анализа будет включать в себя следующие шаги:

- 1) определение классов объектов обучающего программного приложения;
- 2) выделение классификационных признаков объектов;
- 3) определение значений выделенных классификационных признаков;
- 4) объединение выделенных классов и признаков в единую систему для последующего проведения процедуры синтеза.

Результатом выполнения приведенных действий будет морфологическое множество, определенное в формате диаграммы характеристик.

Задача синтеза в таком случае сводится к оптимальному выбору по составленной диаграмме набора вершин, характеризующего отдельное состояние функционирования системы.

Общая структура типовой обучающей программы представлена на рис. 1. Общая структура, описываемая с помощью модели характеристик, будет включать следующие основные категории объектов:

- 1) модели,
- 2) программные функции,
- 3) параметры.

Основной единицей изменчивости в данном случае будет выступать конфигурация диаграммы характеристик. Под конфигурацией диаграммы характеристик следует понимать отдельное подмножество ее элементов, удовлетворяющее условиям, налагаемым связями между вершинами. Так как диаграмма в целом является описанием всего пространства возможных состояний программы, то ее конфигурация будет использоваться в качестве описания отдельного состояния.

Для обеспечения дальнейшей работы с полученной структурой программы требуется ее преобразование в теоретико-множественное представление.

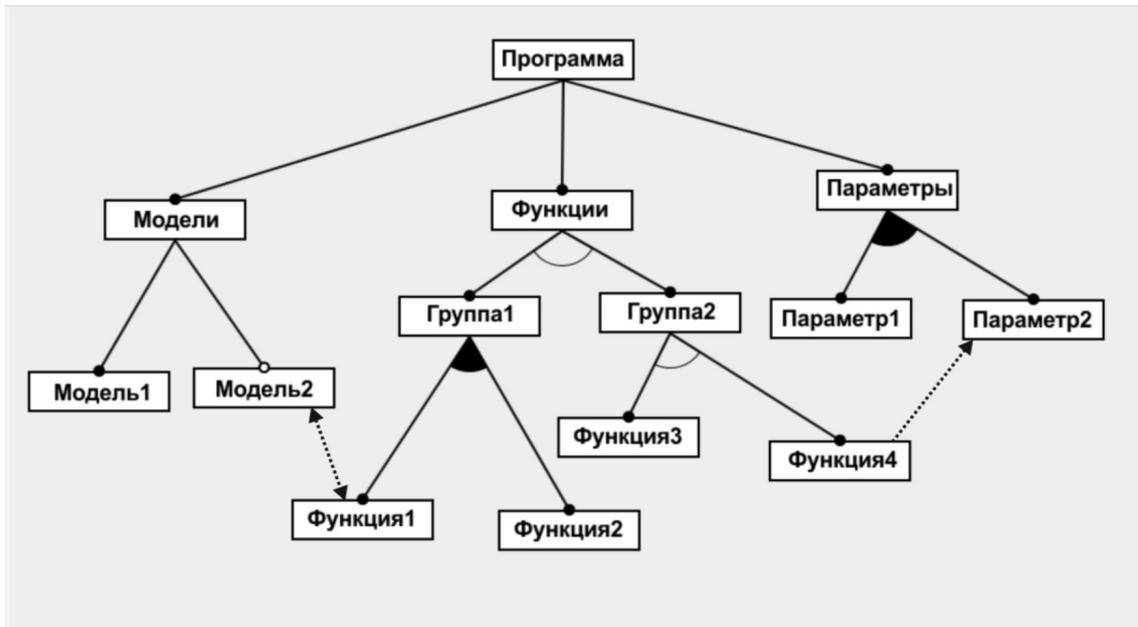


Рис. 1. Общая структура типовой адаптивной обучающей программы

Теоретико-множественное представление общей структуры адаптивной обучающей программы

В качестве альтернативной формы представления диаграммы характеристик был выбран ориентированный гиперграф.

Рассмотрим необходимые определения. Под ориентированным гиперграфом H (или просто гиперграфом) будем понимать упорядоченную пару $H = (V, E)$, которая включает следующие элементы:

$V = \{v_1, v_2, \dots, v_n\}$ – конечное множество вершин (узлов) гиперграфа;

$E = \{E_1, E_2, \dots, E_m\} / E_i \subseteq V \forall i = 1, \dots, m$ – множество упорядоченных гиперребер;

$$E_i = (T(E_i), H(E_i)) /$$

$T(E_i) \subseteq V \wedge H(E_i) \subseteq V - T(E_i)$ – ориентированное гиперребро, для которого $T(E_i)$ – хвостовое множество и $H(E_i)$ – головное множество.

Под хвостовым множеством гиперребра понимается множество вершин, из которых «исходит» гиперребро, под головным множеством – множество вершин, в которые «входит» гиперребро. Именно порядок следования данных множеств определяет ориентацию гиперребра.

Перейдем теперь к преобразованию диаграммы характеристик в форму ориентированного гиперграфа. При данном преобразовании к основным элементам модели характеристик будут применяться следующие правила [4]:

Множество характеристик модели будет отображено на множество вершин соответствующего гиперграфа $Features \rightarrow V$.

Множество взаимоотношений модели будет отображено на множество гиперребер, соединяющих вершины гиперграфа (характеристики модели) $Relations \rightarrow E$.

Основные типы взаимоотношений модели характеристик будут иметь следующие аналогии в гиперграфовой форме:

1. Отношение основной дочерней характеристики является гиперребром $E = (T_E, H_E)$, так что $|H(E)| = 1 \wedge$ маркировка $(E) = [1 \dots 1]$.

2. Отношение опциональной дочерней характеристики является гиперребром $E = (T_E, H_E)$, так что $|H(E)| = 1 \wedge$ маркировка $(E) = [0 \dots 1]$.

3. Взаимоотношение выбора (исключающее ИЛИ, XOR) является гиперребром $E = (T_E, H_E)$, так что $|H(E)| = q / q > 1 \wedge$ маркировка $(E) = [1 \dots 1]$.

4. Взаимоотношение множественного выбора (ИЛИ, OR) является гиперребром $E = (T_E, H_E)$, так что $|H(E)| = q / q > 1 \wedge$ маркировка $(E) = [m \dots n], 1 \leq m \leq n \leq q$.

5. Отношение включения является гиперребром $E = (T_E, H_E)$, так что $|H(E)| = q / q > 1 \wedge$ маркировка $(E) = [q \dots q]$.

6. Отношение исключения является гиперребром $E = (T_E, H_E)$, так что $|T(E)| = \text{Корень} \wedge |H(E)| = q / q > 1 \wedge$ маркировка $(E) = [0 \dots 1]$.

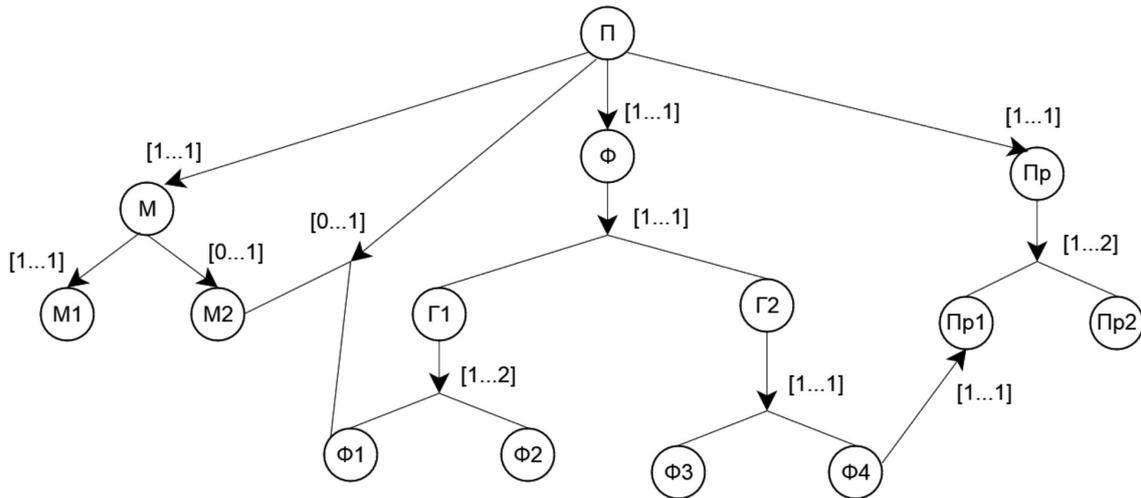


Рис. 2. Гиперграфовое представление диаграммы характеристик

Формально модель характеристик представляет собой ориентированный прямой гиперграф (или F-граф)

$$F = (N, E, \Delta, \Psi), \quad (*)$$

где $N = \{N_1, N_2, \dots, N_n\}$ – конечное множество вершин графа (характеристик исходной модели);

$E = \{E_1, E_2, \dots, E_m\}$ – конечное множество ребер графа (взаимоотношений исходной модели);

$\Delta \in N$ – корневая вершина графа (корневая характеристика модели);

$\Psi : E \rightarrow M, M \subset N \times N$ – функция маркировки, присваивающая значение мощности $mv(E_i) = (\min, \max) \in M$, где M – множество значений мощности диаграммы, N – множество вершин гиперграфа, каждому ребру E_{i_2} так что $\min, \max \in Z \wedge \min \geq 0, \max > 0, \min \leq \max \wedge \max \leq q$, где Z – множество целых чисел.

Под значением мощности понимается минимальное и максимальное количество вершин из головного множества гиперребра, которые могут быть включены в конфигурацию описываемой гиперграфом модели характеристик (подграф исходного гиперграфа).

Корневая вершина $\Delta \in N$ является единственной вершиной гиперграфа, которая не принадлежит ни одному головному множеству ни одного гиперребра. По отношению к корневой вершине и множеству R , содержащему данную вершину, также справедливы следующие правила [5, 6, 7]:

- 1) $R \subseteq N \wedge R = \{\Delta\}$;
- 2) $BS(\Delta) = 0$;
- 3) $BS(n) = 0 / \forall n \in (N - R)$.

На рис. 2 представлен гиперграф, соответствующий диаграмме характеристик, изображенной на рис. 1.

Преобразование общей структуры программы в гиперграфовую форму позволит провести ее дальнейший теоретико-множественный анализ (например, верификацию), а также алгоритмизировать процесс ее адаптивного поведения.

Математическая модель изменчивости адаптивной обучающей программы

Требуемая математическая модель должна обеспечивать возможность алгоритмизации основных процессов изменчивости без перекомпиляции исходного кода. Такая модель должна настраиваться пользователем под нужды предметной области (т.е. из общей абстрактной модели преобразовываться в уточненную модель), и на основе полученной уточненной модели должна производиться настройка алгоритма, реализующего процесс адаптивного поведения системы. Свойство инвариантности к предметной области возможно обеспечить за счет корректного использования модели морфологического множества программы.

С учетом приведенных требований предлагается следующая математическая модель:

$$M = (F, S, X),$$

где F – представление общей структуры приложения в форме ориентированного гиперграфа, полученного на основе диаграммы характеристик и описываемого формулой (*);

Пример матрицы переходов

Строки	Столбцы				
	1	2	3	4	5
1	[0; 0,5)	[0,5; 2)	[2; ∞)	∅	∅
2	∅	[0; 0,5)	[0,5; 2)	[2; 5)	[5; ∞)
3	∅	[0; 0,1)	[0,1; 0,5)	[0,5; 4)	[4; ∞)
4	∅	∅	[0; 0,1)	[0,1; 0,7)	[0,7; ∞)
5	∅	∅	∅	∅	[0; 1)

$S = \{S_1, S_2, \dots, S_k\}$ – конечное множество конфигураций диаграммы характеристик, каждая из которых является описанием определенного состояния адаптивной системы (подграфом исходного гиперграфа); X – матрица переходов между состояниями программы.

Матрица содержит элементы x_{ij} , каждый из которых представляет собой интервал $[a; b)$ либо пустое множество \emptyset , причем $\bigcap x_{ij} = \emptyset$ и $\bigcup x_{ij} = [0; \infty)$, $j = 1 \dots k$, i – фиксировано. Из состояния p осуществляется переход в состояние q , если показатель работы пользователя с программой $C \in x_{pq}$. Если $x_{pq} = \emptyset$, то переход из состояния p в состояние q невозможен.

Пример матрицы переходов представлен в таблице.

Чтобы с помощью описанной модели произвести синтез адаптивной системы, необходимо:

1. Описать общую структуру программы в форме диаграммы характеристик.
2. Определить отдельные состояния работы программы как подмножества характеристик исходной диаграммы.
3. Задать связь между состояниями в формате матрицы переходов.

Как видно, предложенная математическая модель изменчивости является достаточно наглядным и удобным инструментом синтеза структуры адаптивной программной системы. Элементы данной модели служат для описания пространства возможных состояний программы.

Заключение

В данной работе впервые предложена математическая модель изменчивости компьютерных обучающих программ, основанная на теоретико-множественном представлении объектов характеристического моделирования. В основе построения данной модели лежат принципы морфологического анализа и синтеза. Модель позволяет определять многоконтурные обратные связи в адаптивной обучающей программе, а также создавать отдельные состояния работы приложения не только в процессе про-

ектирования, но и в процессе выполнения программы.

Список литературы

1. Кумунжиев К.В. Теория систем и системный анализ: учебное пособие / К.В. Кумунжиев. – Ульяновск: УлГУ, 2003. – 240 с.
2. Финогеев А.А. Анализ информационных рисков в системах обработки данных на основе туманных вычислений / А.А. Финогеев, А.Г. Финогеев, И.С. Неведова // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2016. – № 2. – С. 5–16.
3. Dinkelaker T.A dynamic software product line approach using aspect models at runtime / T. Dinkelaker [et al.] // In 1st Workshop on Composition and Variability. Proceedings of the 1st Workshop on Composition and Variability. – Washington: IEEE Computer Society, 2010. – P. 23–32.
4. Kang K.C. Feature-oriented domain analysis (FODA): feasibility study / K.C. Kang [et al.]. – Pittsburgh: Software Engineering Institute, 1990. – 161 p.
5. Rosenmuller M. Tailoring dynamic software product lines / M. Rosenmuller [et al.] // Proceedings of the 10th ACM international conference on Generative programming and component engineering. – New York: ACM, 2011. – P. 3–12.
6. Schobbens P.E. Feature diagrams: a survey and formal semantics / P.E. Schobbens, P. Heymans, J.C. Trigaux // In 14th IEEE International Requirements Engineering Conference (RE'06). – Washington: IEEE Computer Society, 2011. – P. 139–148.
7. Shen L. Towards feature-oriented variability reconfiguration in dynamic software product lines / L. Shen [et al.] // Top productivity through software reuse / edited by K. Schmid. – Heidelberg: Springer Berlin Heidelberg, 2011. – P. 52–68.

References

1. Kumunzhiev K.V. Teoriya sistem i sistemnyj analiz: uchebnoe posobie / K.V. Kumunzhiev. Uljanovsk: UIGU, 2003. 240 p.
2. Finogeev A.A. Analiz informacionnyh riskov v sistemah obrabotki dannyh na osnove tumannyh vychislenij / A.A. Finogeev, A.G. Finogeev, I.S. Nefedova // Izvestija vysshih uchebnyh zavedenij. Povolzhskij region. Tehnicheskie nauki. 2016. no. 2. pp. 5–16.
3. Dinkelaker T.A dynamic software product line approach using aspect models at runtime / T. Dinkelaker [et al.] // In 1st Workshop on Composition and Variability. Proceedings of the 1st Workshop on Composition and Variability. Washington: IEEE Computer Society, 2010. pp. 23–32.
4. Kang K.C. Feature-oriented domain analysis (FODA): feasibility study / K.C. Kang [et al.]. Pittsburgh: Software Engineering Institute, 1990. 161 p.
5. Rosenmuller M. Tailoring dynamic software product lines / M. Rosenmuller [et al.] // Proceedings of the 10th ACM international conference on Generative programming and component engineering. New York: ACM, 2011. pp. 3–12.
6. Schobbens P.E. Feature diagrams: a survey and formal semantics / P.E. Schobbens, P. Heymans, J.C. Trigaux // In 14th IEEE International Requirements Engineering Conference (RE06). Washington: IEEE Computer Society, 2011. pp. 139–148.
7. Shen L. Towards feature-oriented variability reconfiguration in dynamic software product lines / L. Shen [et al.] // Top productivity through software reuse / edited by K. Schmid. Heidelberg: Springer Berlin Heidelberg, 2011. pp. 52–68.