

УДК 62.52

СИНТЕЗ МОДЕЛИРУЮЩИХ АЛГОРИТМОВ УПРАВЛЕНИЯ МНОГОУРОВНЕВЫХ ДИНАМИЧЕСКИХ ОБЪЕКТОВ

Сиддиков И.Х., Ядгарова Д.Б., Бахриева Х.А.

В работе рассматриваются вопросы синтеза моделирующих алгоритмов управления многоуровневых динамических объектов. Для представления знаний о возможностях вычислений в работе используется семантическая сеть, объектами которых являются переменные, связанные с частными отношениями, такие сети называют вычислительными моделями. Приведены формальные алгоритмы моделирования на базе этих моделей. Компоненты исходной модели описываются через компоненты элементарных моделей. В процессе работы с моделями могут встречаться входные переменные двух типов. С использованием этих переменных определяются непересекающиеся подмножества. Центральное место при генерации моделирующих программ занимает база знаний, где представляются сведения о лингвистических средствах моделирующей системы и знания о предметной области. На базе предложенного подхода реализован экспериментальный вариант программной системы моделирования структурно-сложных динамических систем.

Ключевые слова: управления, синтез, модель, динамический объект, компонент, алгоритм, многоплановость, система, идентификатор, семантическая память, моделирования, программа, семантическая сет.

Ушбу ишда динамик объектларнинг кўпсатҳли бошқариш алгоритмларини моделлаштириш синтези кўриб чиқилади. Билимларни тасвирлашда ҳисоблашлар имкониятлари учун ишда ўзгарувчилар объектлари ҳисобланган семантик тармоқдан фойдаланилади, хусусий муносабатлар билан алоқадор бўлган, бундай тармоқларни ҳисоблаш моделлари деб аташади. Ушбу базалар асосида шаклланган моделлаштириш алгоритмлари келтирилган. Ушбу дастлабки моделлар элементар моделлар компонентлари орқали ёритилади. Ушбу моделлар билан ишлашда икки турдаги кириш ўзгарувчиларини учратишимиз мумкин, ушбу ўзгарувчилардан фойдаланган ҳолда кесишмайдиган кўпчилик бўлимлар аниқланади. Моделлаштирилувчи дастурлар генерациясида билимлар омбори моделлаштирилувчи тизимни ва фан доирасидаги билимларни лингвистик воситалари маълумотлари тасвирланадиган марказий жойни эгаллайди. Таклиф этилаётган ёндошув асосида динамик тизимларнинг мураккаб-структурали моделлаштиришнинг тизимли дастурий тажрибавий варианты амалга оширилган.

Таянч иборалар: бошқарув, синтез, модель, компонент, динамик объект, алгоритм, кўпрежалилик, тизим, идентификатор, семантик хотира, моделлаштириш, дастур, семантик тармоқ.

In this paper, the synthesis of simulation algorithms for managing multi-level dynamic objects are considered. To represent knowledge about the possibilities of calculations in the work, a semantic network is used where objects are variables related to private relations, such networks are called computational models. Formal modeling algorithms based on these models are carried out, and components of the initial model are described through the components of elementary models. In process of working with models, input variables of two types can occur and using these variables, disjoint subsets are defined. Central to the generation of modeling programs are the knowledge base, which provides information on the linguistic tools of the modeling system and knowledge of the subject area. Based on the proposed approach, an experimental version of the software system for modeling structurally complex dynamical systems are implemented. The software here is a set of applied problems that arise when designing and exploring complex technical systems. When considering the structure of the software system that uses the proposed methodology, three main subsystems are identified: the interface subsystem; semantic memory management, semantic memory. They store all the information that determines the functions of the system. Algorithms and specialized knowledge of the permissible properties of classes of algorithms are represented in the semantic memory in the form of semantic models, rules are presented that allow to generate a modeling algorithm, its processing and execution.

Key words: control, synthesis, model, dynamic object, component, algorithm, multidimensionality, system, identifier, semantic memory, modeling, program, semantic set.

I. ВВЕДЕНИЕ

Сложность технических систем как объектов проектирования и сложность процессов управления технологическими процессами делает необходимым выделение ряда общих характеристик задач и моделей проектирования, встречающихся как при разработке новых, так и при модернизации существующих систем. К таким общим характеристикам относятся большая размерность системы, наличие большого числа неопределенных и плохо формализованных факторов, а также факторов изменения внешней среды и внутри модельных связей и др.[1]. Указанные особенности не позволяют подходить к моделированию как однократному акту и порождают значительные трудности при использовании традиционных методов моделирования в процессах проектирования.

В общем случае процесс оптимального проектирования предполагает разработку комплекса моделей, образующих многоуровневую систему и отражающих итеративный, динамический и иерархический характеры рассматриваемых процессов.

Комплексный характер и многоплановость процесса проектирования сложных технических и других систем приводят к существенным трудностям в построении и реализации многоуровневой иерархической системы моделей и моделирующих алгоритмов, в достаточной степени адекватной рассматриваемому процессу. Одним из возможных перспективных путей решения этой проблемы является создание проблемно-ориентированных моделирующих систем [2], в которых модели извлекаются из элементарных операций, заданных в явном виде, т.е. в виде соотношений.

Весь алгоритм складывается из таких соотношений, которые могут пополняться по мере необходимости. При этом предполагается, что соотношения представляются в некотором формальном виде, над которыми работают стандартные механизмы. В результате становится возможной организация активного диалога за счет использования стандартных средств. Пользователь может изменять алгоритм по своему усмотрению, что осуществляется через входной язык системы.

II. ОСНОВНАЯ ЧАСТЬ

Центральное место при генерации моделирующих программ занимает база знаний (БЗ), где представляются сведения о лингвистических средствах моделирующей системы и знания о предметной области (ПО). Лингвистические знания используются для реализации взаимодействия пользователей с системой, а знания о предметной области - при интерпретации запросов (задания на моделирование) и синтезе моделирующих программ.

Для представления знаний о возможностях вычислений в работе используется семантическая сеть, объектами которых являются переменные, связанные с частными отношениями. Такие сети называют вычислительными моделями.

Формально алгоритмы моделирования на базе этих моделей могут быть представлены в следующем виде:

$$M = \langle I, P, \Phi, X, Y, \Omega \rangle, \quad (1)$$

где I -идентификатор модели; $P = P(x_1, x_2, \dots, a_n)$ - одномерный предикат, определенный на множестве X . Смысл этого предиката состоит в формальном определении возможности использования данной модели; $\Phi: X \rightarrow Y (XUY = Z)$ - отображения, описывающие некоторую совокупность свойств моделирующего алгоритма; $X = \{x_1, x_2, \dots, a_n\}$ - входные переменные вычислительной модели; $Y = \{y_1, y_2, \dots, i_n\}$ - выходные переменные вычислительной модели; $Z = \{z_1, z_2, \dots, a_n\}$ - совокупность переменных; Ω - область применения модели, которая задается парой области определения X и значения Y ,

т.е. $\Omega = \{X, Y\}$.

В процессе работы с моделями могут встречаться входные переменные двух типов: определенные и неопределенные. В соответствии с этим множества X будем делить на два непересекающихся подмножества X^H и X^O , т.е. $X^H \cup X^O = X$ ($X^H \cap X^O = \emptyset$).

Элементы множества X^H могут быть определены с введением того или иного отношения между определенными переменными X^O . Такая связь приводит к образованию более сложных многоуровневых вычислительных моделей или иначе, модель M , имеющая в качестве выхода вектор y и неопределенные вектора переменных X^H , может быть представлена в виде совокупности $N_m = N_{ix} \cup N_Y$ моделей-отображений со скалярными $Y_{ak} \in Y, k=1, 2, \dots, N_m$.

Используемые подмодели, являющиеся блоками различной размерности, также n могут быть представлены в виде, аналогичном выражению (1):

$$M_i = \langle I, P_i, f_i, x_i, y_{ak}, \Omega \rangle (2)$$

Тогда компоненты исходной модели описываются через компоненты элементарных моделей следующим образом:

$$M = \{m_i\}, \Phi = \{f_i\}, i \in [1, N_m]$$

$$(\forall x)P(X) \leftrightarrow [P_1(X_1) \wedge P_2(X_2) \wedge \dots \wedge P_{N_m}(X_{N_m})]$$

$$X = \bigcup_{i=1}^{N_m} X_i; \quad Y = \left(\bigcup_{i=1}^{N_m} Y_i \right) / X$$

$$Z = \bigcup_{i=1}^{N_m} Z_i;$$

$$\Omega_i = P_z Z_i(\Omega) : i = 1, 2, \dots, N_m$$

$P_z Z_i(\Omega)$ – проекция множества на гиперплоскость, координатами которой являются компоненты вектора Z_i .

Такие формы представления вычислительных моделей позволяют обобщить процедуры, связанные с формированием моделирующих алгоритмов и представить их как формальные задачи, решаемые на базе этих моделей.

Пусть имеется некоторый технический объект, соответствующий условиям задачи, множества элементарных вычислительных моделей $\{m_i\}_{i \in I}$ и модели декларативных знаний ПО. Тогда задачи формирования моделирующих алгоритмов сводятся к реализации такого вида функционального отображения, которая формирует вычислительную модель для последовательного расширяемого набора данных в процессе эволюционного поиска. Аналитическое представление этой задачи имеет следующий вид:

$$\mu_k : (M_1 \times M_2 \rightarrow M_3),$$

где μ_k - специальный механизм процедуры, работающий над двумя объектами (G_1, G_2) и реализующий отображение $(G_1, G_2) \rightarrow G_3, G_1$ — условия задачи; G_2 -

используемые модели; G_3 — полученная модель, M_1, M_2, M_3 - классы моделей, представляющих как конкретную, так и обобщенную модели ПО.

Задача синтеза моделирующих алгоритмов в такой постановке решается следующим образом.

1. Определяется какому классу формальных постановок относится решаемая задача G_1 .

2. Выбор из множества допустимых методик некоторой конкретной методики $G_A \in M_1$.

3. Нахождение из графы G_{main} (семантические модели), являющейся образом вершин графа G_a .

4. Выделение пары векторов входных и выходных переменных (X, Y) из G_a и G_1 .

5. Нахождение образа каждой неопределенной вершины графа G_a , с помощью которого могут быть выражены через переменные с известными их значениями. Если хотя бы для одного $\{m_i\}_{i \in I}$ не существует такой образ, то формируется сообщение «решение не существует» и выполнение процедуры прекращается.

6. Объединение графов и в единый граф G_3 .

7. Установление на множестве вершин графа G_3 (граф вычислительной модели задачи) порядок их выполнения. Порядок выполнения элементарных моделей m_i устанавливается по следующему правилу: модель m_{in} , может выполняться только тогда, существующий ей предикат $P(X_i)$ принимает истинное значение.

На базе предложенного подхода реализован экспериментальный вариант программной системы моделирования структурно-сложных динамических систем. В качестве ПО здесь выступает совокупность прикладных задач, возникающих при проектировании и исследовании сложных технических систем. При рассмотрении структуры программной системы, использующей предложенную методику, можно выделить три основные подсистемы: интерфейсную (подсистема общения); управления семантической памятью, семантическую память.

Семантическая память (СП) является образом предметной области ПО. В ней хранится вся информация (знания), определяющая функции системы. Алгоритмы и специальные знания о допустимых свойствах классов алгоритмов представляются в СП виде семантических моделей, представляются правила, позволяющие генерировать моделирующий алгоритм, обработку и его выполнение.

Математические модели объектов моделирования хранятся в СП виде набора отношений, которые также допускают изменение и дополнение. В результате пользователь получает возможность корректировать семантику алгоритмов, задавать различные критерии допустимости, вводить новые алгоритмы и модели, корректировать модели объекта все это обеспечивается с

помощью подсистемы общения на удобном пользователю ограниченном естественном языке.

Интерфейсная подсистема для реализации перечисленных функций в своем составе имеет: транслятор с входного языка; монитор диалога; блок расширения и корректировки; блок формирования сообщения. Задачи по поддержанию базы моделей (поиск, запись, хранения и дублирования моделей), синтезу моделей объектов моделирования и синтезу моделирующих программ выполняются подсистемой управления семантической памяти (ПУСП). Она включает следующие блоки: распознавания модели; синтеза моделей объектов управления; синтеза структуры алгоритмов моделирования; генератор моделирующих программ.

Первоначально задание, записанное на входном языке, преобразуется во внутреннее представление, т.е. выявляется смысл задания и строится соответствующая сеть. Это осуществляется с помощью транслятора. В составе транслятора работают также средства преобразования модели, которые преобразуют исходное описание моделей во внутреннюю единую форму представления. Этим достигается универсализация алгоритмов обработки моделей. Далее полученная модель входной информации обрабатывается средствами ПУСП. Подсистема синтеза алгоритмов моделирования, опираясь на знания о предметной области и модели входной информации, синтезирует вычислительную модель задачи, а генератор с помощью этой модели генерирует моделирующие программы. Выходной информацией программной системы является модель системы, алгоритм моделирования и расчетов.

Данная программная система характеризуется быстродействием и минимальностью используемой памяти за счет выполнения двух задач: задачи построения математической модели, объекта моделирования и задачи синтеза моделирующих программ.

III. ВЫВОДЫ

1. Для целей имитационного моделирования предлагаются многоуровневые моделирующие алгоритмы.
2. Для формализации процесса построения моделирующих алгоритмов предлагается семантическая сеть.
3. Разработана методика синтеза моделирующих алгоритмов технических объектов, обеспечивающих удобство для пользователей имитационной системы.

ЛИТЕРАТУРА

- [1] Современные системы управления / Р.Дорф, Р.Бишоп – М: Лаборатория базовых знаний, 2002. - 832 с.
- [2] Джексон П. Введение в экспериментальные системы, - М: Издательский дом «Вильямс», 2001. – 624 с.