

УДК 681.3 + 519.6

## Технология визуального объектно-ориентированного моделирования сложных систем

К. С. КУРОЧКА

### Введение

Интенсивное развитие средств вычислительной техники позволило исследователям рассматривать всё более и более сложные системы с большим спектром свойств их элементов. Однако увеличение уровня детализации рассматриваемых систем приводит к необходимости ввода большого количества исходных данных. Это приводит к тому, что на практике процесс создания модели исследуемой системы с достаточной степенью точности оказывается очень сложной задачей, разрешимой только высококвалифицированными специалистами, как в области моделирования, так и в предметной области рассматриваемой системы [1]. Поэтому является актуальной задача разработки средств, упрощающих построение и исследование математических моделей сложных неоднородных систем.

### Проектирование программного обеспечения компьютерного моделирования сложных систем

Всякая система существует в некоторой окружающей среде, обуславливается ею и имеет свою границу. Для конкретной системы окружающая среда есть совокупность всех объектов, изменение свойств которых влияет на систему, а также тех объектов, свойства которых меняются в результате действия системы. Предполагается, что система может иметь структурное представление, т. е. может быть расчленена на группы элементов с указанием связей между ними. Это расчленение называется декомпозицией и сохраняется на время действия системы. Указанные группы элементов называются модулями, в связи с этим, принято говорить о модульной структуре системы. Модули образуются по принципу общих свойств, характеру связей или других признаков [1]. Рассмотрение систем будем осуществлять с позиций микроподхода, заключающемся в исследовании отдельных модулей системы, их свойств и связей.

Таким образом, на первом этапе проектирования программного обеспечения предлагается осуществить общую декомпозицию рассматриваемого класса систем с целью определения неделимых в рамках принятой дискретизации однородных модулей. Для каждого такого модуля необходимо чётко определить исходные данные (входные воздействия), данные, получаемые в результате функционирования (результат), и установить связи с другими модулями. Кроме того, для динамических систем задаётся последовательность функционирования модулей системы, определяется порядок и способ передачи данных между ними. На основе этого получим информацию о функционировании системы в целом, на основании которой осуществим построение концептуальной модели, заключающееся в выполнении следующих шагов:

- 1) выявление исследуемых элементов системы;
- 2) изучение структуры и функций выделенных элементов системы;
- 3) выявление связей между элементами;
- 4) формирование гипотезы о функционировании процесса в целом.

На втором этапе проектирования программного обеспечения осуществляется построение функциональной модели, в результате которого необходимо установить иерархическую связь между модулями системы. При этом нужно структурировать рассматриваемый класс систем таким образом, чтобы можно было выделить несколько однотипных наборов исход-

ных данных для модулей системы и несколько наборов результирующих данных. Такое описание системы удобно осуществлять средствами функционального моделирования с помощью специализированных языков описаний IDEF0 [2] или UML [3].

Учитывая изложенное выше, можно выделить следующие этапы разработки программного обеспечения по визуальному объектно-ориентированному моделированию сложным систем:

- 1) декомпозиция рассматриваемого класса систем;
- 2) построение концептуальной модели рассматриваемого класса систем;
- 3) построение функциональной модели рассматриваемого класса систем;
- 4) построение математической модели для каждого модуля системы;
- 5) выполнение проектирования интерфейса пользователя;
- 6) разработка алгоритмов и программ для исследования математической модели;
- 7) разработка алгоритмов и программ для визуализации результатов.

Тогда алгоритм компьютерного моделирования сложных систем можно представить в виде:

- 1) создание виртуальной модели системы;
- 2) формирование математической модели системы;
- 3) численное решение математической модели системы;
- 4) формирование виртуальной модели решения системы;
- 5) изменение геометрических, физических параметров и связей между элементами виртуальной модели системы.

### **Построение виртуальной модели системы**

Это создание компьютерного представления реальной системы, описывающего её геометрические, физические свойства и процесс функционирования. Как правило, в сложных неоднородных системах при заданном уровне детализации (точности) можно выделить несколько групп модулей, которые будут иметь одинаковые входные и результирующие данные, однотипное функционирование и связи с другими модулями системы, отличающиеся только набором значений своих свойств и параметров. Причём, с каждым модулем будет отождествляться элемент функциональной модели, а вся система будет определяться совокупностью таких элементов. В компьютерном представлении каждому уникальному элементу функциональной модели поставим в соответствие конструктивный элемент, из которых на экране монитора будет формироваться виртуальная модель системы. Совокупность всех конструктивных элементов определяет предметную область рассматриваемых систем.

Каждый конструктивный элемент в памяти компьютера представляет собой сложную структуру данных, содержащую экземпляры различных пяти классов объектов.

Первый класс объектов определяет физические и геометрические характеристики конструктивного элемента, его описание, параметры и свойства, т. е. определяет конкретный экземпляр конструктивного элемента.

Второй класс – определяет функциональное описание соответствующего модуля системы, формат исходных данных и результата. Таким образом, данная группа свойств и методов определяет уникальность каждого типа конструктивных элементов.

Третий класс – задаёт положение данного элемента относительно системы, определяет связи данного элемента с другими элементами системы, указывает порядок получения и передачи данных, содержит ссылки на множество конструктивных элементов, которые могут являться источниками данных, и множество конструктивных элементов, которые могут являться получателями результатов функционирования элементов рассматриваемого типа. Т. е. объект данного класса определяет граничные и начальные условия для модуля системы.

Четвёртый класс – отвечает за визуализацию данного элемента на экране монитора с учётом отображения других элементов системы, определяет внешний вид отображаемого элемента. При моделировании физических систем отображение конструктивного элемента дополнительно будет определяться объектом первого класса. Для визуализации конструктивных элементов целесообразно применять средства, предоставляемые Microsoft DirectX –

мощным пакетом графических процедур и функций, использующих аппаратные возможности видеоадаптеров, что позволяет с максимальной скоростью обрабатывать сложные графические объекты. Следовательно, данный класс объектов должен дополнительно содержать свойства и методы, позволяющие для визуализации конструктивного элемента использовать DirectX. Одними из наиболее важных свойств этой группы, задаваемые для каждого треугольника, определяющего поверхность конструктивного элемента, являются цвет, материал – определяет реакцию цвета на используемое освещение, текстура – представляет собой цифровое изображение, заполняющее треугольник. Текстуры позволяют сделать изображение на экране монитора наиболее реалистичным.

Ключевым моментом визуализации трехмерного изображения является трансформация объектов трехмерной сцены в пространство просмотра как часть цепочки геометрических преобразований (the geometry pipeline). С математической точки зрения этот процесс представляет собой перерасчет координат объектов из одной локальной системы координат (пространство объекта) в другую (пространство просмотра). Обычно для решения этой задачи применяется матрица перехода четвертого порядка. Эта матрица представляет собой произведение координатной матрицы объекта на обращенную матрицу пространства просмотра (виртуальной камеры).

Пятый класс объектов определяет математическую модель элемента системы и методы её решения. Причём методы построения и решения математической модели могут быть наследованы от некоторого глобального класса, определяющего всё математическое обеспечение приложения.

Таким образом, каждый тип конструктивных элементов будет представлять собой класс, являющийся наследником описанных выше пяти классов, что позволит в памяти компьютера хранить в единственном экземпляре базовые параметры каждого класса, на которые и будет ссылаться потомок.

### **Визуальное объектно-ориентированное моделирование систем**

Из конструктивных элементов виртуальных моделей формируется база данных, элементы которой группируются и размещаются на палитре компонентов. Для удобства поиска в палитре конструктивные элементы размещены по группам. Группы поддерживают несколько уровней вложенности. Конструктивные элементы можно добавлять в группы либо перемещать между группами. Отдельно формируется редактор конструктивных элементов, позволяющий создавать новые конструктивные элементы, либо редактировать существующие. Также имеется возможность создавать новые конструктивные элементы на основе комбинации существующих.

Кроме палитры компонентов, прикладная программа должна содержать окно (или окна) проектирования. Построение виртуальной модели системы осуществляется следующим образом: сначала разработчик выбирает на палитре компонентов необходимый конструктивный элемент (в случае его отсутствия – создаёт) и с помощью мыши в окне проектирования перетягивает его на место в системе согласно разработанной функциональной модели системы или проектной документации. В случае ошибочного размещения конструктивного элемента его можно либо переместить на своё место в системе, либо удалить. Затем необходимо указать связи данного элемента с другими, ранее размещёнными. При этом автоматически контролируются согласованность данных элементов, т. е. выполняется проверка форматов исходных и результирующих данных для связываемых элементов.

### **Использование технологии визуального объектно-ориентированного моделирования для исследования систем механики деформируемого твёрдого тела**

Для рассматриваемых систем указанные выше базовые классы будут содержать следующие свойства и методы:

1) определение геометрических, физических и информационных свойств реального конструктивного элемента: размеры, границы, законы деформирования, модуль упругости, коэффициент Пуассона и т. д., уникальный идентификатор, наименование, марка, характеристика элемента и т. п.;

2) описание математической модели конструктивного элемента;

3) определение граничных и начальных условий, связей с другими элементами;

4) осуществление визуализации виртуального конструктивного элемента, определение цвета, текстур и отображения средствами DirectX;

5) формирование математической модели всей системы, исследование математической модели и формирование виртуальной модели решения.

Проектируемая система на экране монитора отображается в четырёх окнах. В первых трёх отображаются соответственно проекции на координатные плоскости либо сечения плоскостями, параллельными плоскостям  $XY$ ,  $XZ$ ,  $YZ$ . В четвёртом окне отображается вид из виртуальной камеры – показ конструкции в ракурсе, определяемом проектировщиком. Ракурс определяется точкой размещения наблюдателя, расстоянием от наблюдателя до переднего плана и размерами переднего, заднего планов. Таким образом, ракурс представляет собой пирамиду, в вершине которой располагается наблюдатель, а вид из камеры – усечённую пирамиду, где верхняя плоскость – это передний план, а нижняя плоскость – это задний план. В каждом окне могут задаваться три режима визуализации, когда отображаются: только узловые точки, проволочный каркас, вся конструкция. Изображение в каждом окне может масштабироваться.

Геометрия конструктивного элемента определяется его внутренними и внешними границами, которые представляют собой совокупность треугольников и нормалей к плоскости, содержащей данный треугольник. Такое представление однозначно определяет геометрические свойства конструктивного элемента, его границы.

Для описания физических свойств конструктивного элемента целесообразно ввести несколько методов, позволяющих учитывать различное поведение элемента после приложения нагрузки. Например, законы упругопластического и вязкоупругого деформирования конструктивного элемента.

Для решения задач механики деформируемого твёрдого тела предлагается использовать метод конечных элементов, который позволяет с достаточной для практики точностью учитывать неоднородные свойства системы, посредством задания различных свойств конечным элементам. Поэтому на этапе формирования виртуальной физической модели системы параллельно проводится её дискретизация на конечные элементы. После размещения первого конструктивного элемента его верхний левый угол считается узлом сетки. Шаг разбиения может быть неравномерным, однако узлы на границах между конструктивными элементами должны быть строго общими. Если какой-нибудь вновь размещаемый конструктивный элемент не будет удовлетворять данному условию, то будет выдано сообщение об ошибке и будет предложено автоматически изменить дискретизацию области определения системы или элемента.

Для формирования математической модели одним из основных является метод, позволяющий по глобальным координатам узла определить его локальный индекс. Среди свойств можно выделить свойство, хранящее начальный глобальный индекс для данного конструктивного элемента, благодаря которому можно всегда легко получить глобальный индекс любого узла конструктивного элемента. Построение математической модели системы начинается с формирования глобальной матрицы жёсткости. Для этого применяется подход, основанный на методе декомпозиции, используемой на стадии формирования виртуальной физической модели:  $n$ -связанная область определения системы разбивается на  $k$  подобластей ( $k$  конструктивных элементов)  $\Omega_i$  с границей  $\Gamma_i$ ,  $i = \overline{1, k}$ . Для каждой подобласти  $\Omega_i \cup \Gamma_i$  введём функцию:

$$f_i(x, y, z) = \begin{cases} 1, & \text{если } (x, y, z) \in \Gamma_i \\ 0, & \text{если } (x, y, z) \notin \Gamma_i \end{cases}.$$

Пусть  $\Gamma^*$  – это множество всех общих рассмотренных участков границ между подобластями  $\Omega_i$ , т.е., после просмотра всех подобластей,  $\Gamma^*$  будет иметь вид  $\Gamma^* = \Gamma_1 \cap \Gamma_2 \cup \Gamma_1 \cap \Gamma_3 \cup \dots \cup \Gamma_1 \cap \Gamma_k \cup \Gamma_2 \cap \Gamma_3 \cup \dots \cup \Gamma_{k-1} \cap \Gamma_k$ .

Определим на множестве  $\Gamma^*$  функцию

$$f_{\Gamma^*}(x, y, z) = \begin{cases} N_{\Gamma^*}, & \text{если } (x, y, z) \in \Gamma^* \\ 0, & \text{если } (x, y, z) \notin \Gamma^* \end{cases},$$

где  $N_{\Gamma^*}$  – глобальный индекс узла с координатами  $(x, y, z)$ .

Будем последовательно дискретизировать подобласти  $\Omega_i \cup \Gamma_i$  конечными элементами. Положим  $N_{\text{узел}} = 0$  и  $\Gamma^* = \emptyset$ , где  $N_{\text{узел}}$  – глобальный счётчик узлов.

Для каждого узла  $M_j = (x_j, y_j, z_j)$  конечного элемента  $A_l$  подобласти  $\Omega_i \cup \Gamma_i$ , где  $l = \overline{1, N_{\text{КЭ}}^i}$ ,  $j = \overline{1, n_i}$ ,  $n_i$  – количество узлов в подобласти  $\Omega_i \cup \Gamma_i$  выполняем следующие действия:

1. Если  $f_i(x_j, y_j, z_j) = 0$ , то  $N_{\text{узел}} = N_{\text{узел}} + 1$ , а глобальный индекс узла  $M_j$  будет равен  $N_{\text{узел}}$ .
2. Если  $f_i(x_j, y_j, z_j) = 1$  и  $f_{\Gamma^*}(x_j, y_j, z_j) = 0$ , тогда  $N_{\text{узел}} = N_{\text{узел}} + 1$  и доопределим функцию  $f_{\Gamma^*}$  так, чтобы  $f_{\Gamma^*}(x_j, y_j, z_j) = N_{\text{узел}}$ . Глобальный индекс узла  $M_j$  будет равен  $N_{\text{узел}}$ .
3. Если  $f_i(x_j, y_j, z_j) = 1$  и  $f_{\Gamma^*}(x_j, y_j, z_j) < 0$ , то глобальный индекс узла  $M_j$  будет равен  $f_{\Gamma^*}(x_j, y_j, z_j)$ .

Процесс нахождения глобальных индексов узлов можно совместить с вычислением соответствующих коэффициентов в матрице жёсткости. Т. о. после дискретизации конечными элементами всех подобластей  $\Omega_i \cup \Gamma_i$ , будет сформирована глобальная матрица жёсткости для  $n$ -связанной области.

Каждому конструктивному элементу присваивается уникальный идентификатор длиной 64 бита (Int64), где старшие 8 бит кодируют номер первой группы, следующие 8 бит – номер второй группы, 40-47 биты кодируют номер третьей группы, 0-39 – задают уникальный номер элемента. Таким образом, в памяти компьютера для представления системы организуется динамический список. Элементом списка является запись со следующими полями: уникальный идентификатор конструктивного элемента, глобальные координаты левого верхнего угла элемента, масштаб, указатель на следующий элемент в списке. Таким образом, ещё на стадии формирования виртуальной физической модели происходит декомпозиция исследуемой системы.

После построения виртуальной модели системы необходимо задать граничные и начальные условия. Для каждого узла может быть задано либо известное перемещение, либо известная узловая сила. Для этого с помощью мыши выделяется узел (или группа узлов), вызывается контекстное меню, где выбирается, что в этом узле является известным (сила или перемещение) и вводится значение. На этом формирование виртуальной физической модели завершается.

Учитывая граничные и начальные условия, решаем основное уравнение метода конечных элементов и находим искомые деформации. Далее из всех конструктивных элементов системы формируется новый конструктивный элемент, геометрические параметры которого корректируются согласно найденному решению. В итоге получаем изображение системы после приложения нагрузок.

### Выводы

1. В данной работе предлагается подход на основе компьютерного моделирования, позволяющий на экране монитора строить различные системы. Главным преимуществом

предлагаемой технологии является быстрота проектирования системы из созданных ранее конструктивных элементов, не требующая глубоких познаний в области математического моделирования.

2. С помощью предлагаемой технологии на экране монитора компьютера формируется концептуальная модель системы, затем, на её основе, автоматически создаётся математическая модель, которая в дальнейшем может быть исследована.

3. Виртуальная модель системы в памяти компьютера представляется списком, каждая запись которого содержит уникальный идентификатор конструктивного элемента и позицию элемента в модели, что позволяет наиболее эффективно использовать оперативную память ЭВМ и рассматривать системы, состоящие из большого числа элементов.

4. Разработанная технология компьютерного визуального объектно-ориентированного моделирования позволяет решать разнообразные задачи по исследованию сложных систем.

**Abstract.** The paper presents the general methods and the technology of computer visual object-oriented modeling of complicated systems.

### Литература

1. Максимей, И. В. Математическое моделирование больших систем: Учебное пособие для спец. «Прикл. матем.» / И. В. Максимей // Минск: Высшая школа, 1985.

2. Методология функционального моделирования. Рекомендации по стандартизации / Москва: ИПК Издательство стандартов, 2001.

3. Гома, Х. UML. Проектирование систем реального времени, параллельных и распределенных приложений / Х. Гома // Москва: ДМК Пресс, 2002.

Гомельский государственный технический университет имени П. О. Сухого

Поступило 03.04.07