

УДК 621:004.896

Метод генерации 3D моделей в производственных базах знаний

Г.Б. Евгеньев¹, А.А. Кокорев¹, М.В. Пиримьяшкин²

¹ МГТУ им. Н.Э. Баумана, 105005, Москва, Российская Федерация, 2-я Бауманская ул., д. 5, стр. 1

² Московский центр технологической модернизации образования ГБОУ «ТемоЦентр», 107076, Москва, Российская Федерация, 1-й Зборовский пер., д. 3

Method of 3D model generation in knowledge bases

G.B. Evgenyev¹, A.A. Kokorev¹, M.V. Pirimyashkin²

¹ BMSTU, 105005, Moscow, Russian Federation, 2nd Baumanskaya St., Bldg. 5, Block 1

² Centre of Technological Modernization of Education «TemoCenter», 107076, Moscow, Russian Federation, 1st Zbrovsky Per., Bldg. 3



e-mail: info@rkg.bmstu.ru



Создание интеллектуальных систем конструирования изделий — важное направление совершенствования инженерной подготовки машиностроительного производства. Применение таких систем обеспечивает повышение производительности и качества работы конструкторов за счет полуавтоматической генерации 3D моделей изделий в условиях типового вариантного проектирования, например, создание объектов электромашиностроения, электромеханические приводы и т.п. Высший уровень автоматизации в этой области можно достичь путем использования систем полуавтоматического проектирования, позволяющих на основе введенного технического задания выполнить все необходимые расчеты и генерировать 3D модели и полный комплект конструкторской документации. Однако для решения данной проблемы необходимо разработать метод и средство генерации 3D модели с помощью производственных баз знаний. Предложен новый метод 3D моделей в производственных базах знаний. В этом методе процесс разделен на две относительно самостоятельные фазы: собственно проектирование и конструирование. На первой фазе проектировщики и расчетчики определяют основные параметры и схемы объекта. На второй фазе на основе данных, полученных ранее, выполняется конструирование. Для разработки интегрированной интеллектуальной системы, включающей обе фазы, потребовалось создать технологию экспертного программирования, которая обеспечивает автоматическую генерацию программных средств на основе экспертных знаний. Базой экспертных знаний служат модули — правила-продукции. Для автоматической генерации 3D моделей изделий разработан модуль, основанный на параметризованных трехмерных моделях. Созданные средства позволяют генерировать системы полуавтоматического проектирования и конструирования изделий машиностроения.

Ключевые слова: интеллектуальная система, системы автоматизированного проектирования, 3D модели, базы знаний.



Intelligent computer aided design systems are an important way to improve the preparation of engineering production. The use of such systems improves efficiency and quality of designers' work through semi-automatic generation of 3D models of products in typical variant design. This type of design includes, for example, the creation of objects for electrical engineering, electromechanical drives, etc. The highest level of automation in this area can be achieved through semi-automatic design systems enabling the designer to perform all the necessary calculations, and generate 3D models and a full set of design documentation

based on the chosen specifications. However, to attain this goal it is necessary to develop a method and means to generate 3D models using product knowledge bases. A new method of 3D model generation is described in this article. In this method, the process is divided into two distinct phases: the actual design and model generation. The first phase is associated with key parameters and the object schema. The second phase involves model generation based on the previously obtained data. To create an integrated intelligent system that includes the two phases, it was necessary to develop a programming technology that provides automatic generation of software based on expert knowledge. Expert knowledge is based on knowledge modules, which are rules-products. To automatically generate 3D models of products, a module based on parameterized three-dimensional models has been developed. The created means make it possible to develop semi-automatic design systems for engineering products.

Keywords: artificial intelligence, computer-aided design, 3D model, knowledge base.

Автоматизированное проектирование изделий в общем случае включает две фазы: собственно проектирование и конструирование [1]. При проектировании изделие представляется в виде формальной системы с соответствующими схемами и эскизной проектной документацией. На этой фазе выполняются разнообразные инженерные расчеты для анализа геометрии, моделирования и изучения поведения изделия, для совершенствования и оптимизации его конструкции.

На основе проектных расчетов, определяющих основные характеристики изделия, производится конструирование, основной задачей которого является построение модели изделия как объекта производства с получением необходимого комплекта чертежей и спецификаций. В качестве средства конструирования используют САД-системы. Современные стандартные требования к этим системам включают твердотельное моделирование, наличие параметризации и ассоциативность [2].

Противники параметризации основным недостатком считают большое количество параметров, об управлении которыми должен помнить пользователь. На основании этого они утверждают о невозможности создания больших сборок и модернизации сложных деталей. Однако даже в неинтеллектуальных САД-системах, таких как SolidWorks, конструктору не нужно запоминать параметры, поскольку реальное построение изделия фактически повторяет ход мысли автора. Это означает лишь то, что заданная идея построения будет отрабатываться при любых модификациях проекта в целом. Причем SolidWorks «помнит» способ построения элемента, что особенно важно при установлении идеологической зависимости между элементами различных деталей (стыковой узел) [2].

В процессе работы над проектом, а точнее при проведении корректировок размеров или в

случае более глобального изменения в конструкции, возникают противоречия с ранее заданными взаимосвязями или назначенными размерами. Такая коллизия диагностируется системой как ошибка в модели и в этой ситуации без инструмента поиска места и типа ошибки реальная работа (исправление) над сложной сборкой или деталью практически невозможна. В системе SolidWorks такой инструмент есть.

Современные системы проектирования должны не только быстро создавать твердотельные модели, но и позволять вносить изменения, не учтенные заранее, и отслеживать их на всех этапах проекта [2]. В этой связи авторы статьи [2] с полным основанием называют систему SolidWorks именно системой проектирования, а не твердотельного моделирования. Однако следует отметить, что авторы [2] анализировали описанные выше ситуации с точки зрения конструктора, хотя в создании изделия задействованы проектировщики и расчетчики, определяющие основные параметры и схему объекта, которыми должен руководствоваться конструктор.

Цель работы — разработка интегрированных интеллектуальных систем, обеспечивающих автоматизацию всех этапов проектирования — от ввода технического задания до получения всего комплекта конструкторских документов. Для решения этой проблемы необходимы базы знаний, способные, в частности, генерировать 3D модели изделий, а также исключать ошибки при модифицировании этих моделей.

Технология экспертного программирования.

Одно из наилучших определений искусственного интеллекта (ИИ) приведено в работе [3]: «Искусственный интеллект — это одно из направлений информатики, целью которого

является разработка аппаратно-программных средств, позволяющих пользователю-непрограммисту ставить и решать свои, традиционно считающиеся интеллектуальными задачи, общаясь с ЭВМ на ограниченном подмножестве естественного языка». К сожалению, на практике технологии ИИ не обеспечивают решения проблемы в полном соответствии с приведенным определением [3].

Искусственный интеллект изначально использует декларативные, т. е. непроцедурные, неалгоритмические методы представления знаний. Поэтому объектно-ориентированное программирование (ООП) способствует сближению технологий программирования и ИИ. Экспертное программирование сочетает в себе лучшие качества обеих технологий.

Появление алгоритмических языков, поддерживающих технологию процедурного программирования, вывело индустрию информатики на более высокий уровень. В своем развитии процедурное программирование прошло ряд этапов, к числу важнейших из которых следует отнести технологии структурного и модульного программирования. Некоторые фундаментальные положения этих технологий были учтены в экспертном программировании,

которое, однако, не принадлежит процедурному классу.

В *структурном программировании* фундаментальной является теорема о структурировании, доказывающая, что любая простая программа функционально эквивалентна структурной программе, составленной из элементов базисного множества {последовательность, ifthenelse, whiledo}. Это означает, что любая программа, представляющая собой один функциональный узел, может быть составлена из структур трех базовых типов (рис. 1). Традиционные экспертные системы, обеспечивая выполнение первых двух элементов базисного множества, не обладают возможностью организации циклов типа whiledo и поэтому они не генерируют прикладные программы.

На практике полезной рекомендацией структурного программирования является целесообразность декомпозиции сложных систем на части, содержащие не более 40–50 компонент, что и учитывается в экспертном программировании при определении максимального количества модулей знаний, входящих в один метод.

В *модульном программировании* (рис. 2) рекомендуется выделение групп подпрограмм,

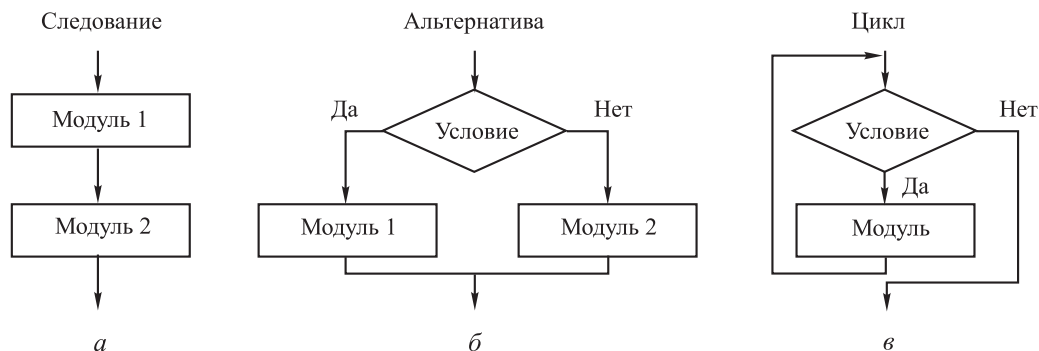


Рис. 1. Основные структуры процедурной декомпозиции: а — последовательная; б — параллельная; в — циклическая

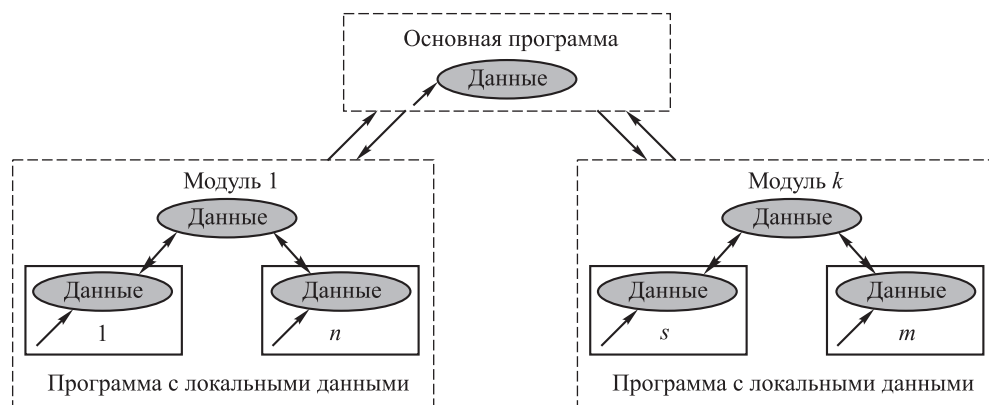


Рис. 2. Архитектура программы, состоящей из модулей

использующих одни и те же глобальные данные, в отдельно компилируемые модули. Связь между модулями осуществляется через специальный интерфейс, а доступ к реализации модуля запрещен.

Подобная иерархическая архитектура использована в экспертном программировании.

Технологии ИИ начали развиваться во второй половине 1950-х годов одновременно с интенсивными поисками моделей и алгоритмов человеческого мышления и разработкой первых программ на их основе. В середине 1970-х годов появились первые системы, основанные на знаниях, или *экспертные системы* (ЭС) [4]. При построении ЭС были учтены результаты исследований в области ИИ, что и определило успех их практического применения. Важнейшим из этих достижений является положение, согласно которому мощность ЭС обусловлена, в первую очередь, мощностью базы знаний, содержащей правила принятия решений, и только во вторую очередь, методами логического вывода, основанными на этих правилах. Как показал опыт развития ИИ важнее иметь разнообразные специальные знания, а не общие процедуры вывода. Это положение использовано в экспертном программировании, в котором программы, реализующие функции вывода, генерируются применительно к каждому конкретному набору правил, составляющих базу знаний.

Чтобы преобразовать продукционную систему ИИ в систему программирования, необходимо разработать методику компиляции как правил вывода, так и экземпляра решателя, осуществляющего вывод на конкретном наборе правил. Такая методика реализована в экспертном программировании.

Понятие «объект» имеет две взаимосвязанные и в то же время относительно самостоятельные компоненты: набор свойств, значения которых определяют состояние объекта, и набор правил поведения, из которых формируется метод объекта. Первая компонента является чисто декларативной и методы ООПр содержат соответствующие механизмы разделения данных. Вторая компонента состоит в объектно-ориентированном подходе при разработке методов объектов, определяющих их поведение. Для решения этой проблемы и используется экспертное программирование.

Разработка любой сложной, в том числе программной системы, должна начинаться с функционального анализа и моделирования системы в целом и всех ее подсистем до неде-

лимых элементов. Для этой цели разработана методология IDEF0, представляющая собой совокупность методов, правил и процедур, предназначенных для построения функциональной структуры сложных иерархических систем [5].

Основной принцип IDEF0 заключается в представлении системы в виде набора взаимодействующих и взаимосвязанных блоков, отображающих процессы, операции и действия, происходящие в системе. Процесс — совокупность последовательно выполняемых операций, преобразующих информацию. Операция — совокупность действий, преобразующих информационные объекты в объекты с другими свойствами. Действие — преобразование какого-либо свойства информационного объекта в другое свойство.

В IDEF0 используются функциональные блоки, принадлежащие одному и тому же классу, который можно назвать объект-функция. В экспертном программировании в качестве суперкласса используется объект-функция IDEF0. Графически такая объект-функция представляется в форме прямоугольника (рис. 3), каждая из сторон которого имеет определенное назначение: левая — входы, правая — выходы, верхняя — управление, нижняя — механизмы. На входы подается вся информация, необходимая для выполнения функции. Результаты выполнения функции преобразуются в выходную информацию. Управление описывает условие, оказывающее влияние на выполнение функции, но само не подвергается переработке.

К нижней части изображения объект-функции присоединены стрелки механизмов, обозначающие программное средство, обеспечивающее выполнение функции. Входы и выходы показывают, что делает функция, управление — почему это делается, а механизмы — с помощью чего делается.

В искусственном интеллекте в качестве эле-

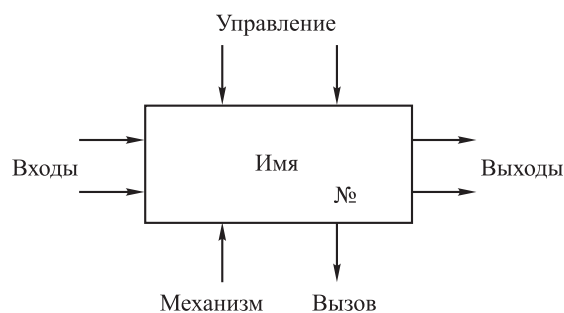


Рис. 3. Функциональный элемент IDEF0

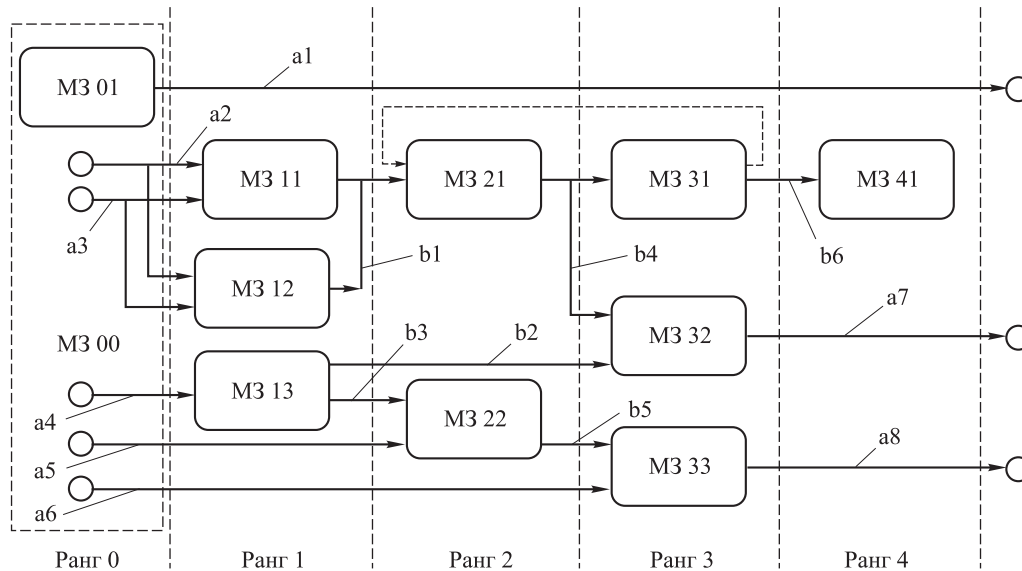


Рис. 4. Ранжированная семантическая сеть МЗ

мента представления знаний используется продукционное правило: если УСЛОВИЕ, то ДЕЙСТВИЕ. Продукционное правило может применяться только в том случае, когда текущее состояние рабочей памяти продукционной системы удовлетворяет УСЛОВИЮ.

Выполнение продукционного правила заключается в изменении информационной структуры продукционной системы в соответствии с заключением ДЕЙСТВИЕ.

Функциональный элемент IDEF0 может рассматриваться как эквивалент продукционного правила: управление определяет УСЛОВИЕ, а действие заключается в преобразовании входов в выходы с использованием механизма или с помощью вызова соответствующего программного средства.

В экспертном программировании продукционное правило называется модуль знаний (МЗ). Механизмы МЗ должны обеспечивать реализацию всех функций, которые могут потребоваться при формировании баз знаний. В число таких функций входят следующие основные: вычисление по формулам (в том числе присвоение значений переменным); определение значений по таблицам; выбор значений из баз данных; обновление значений в базах данных; занесение значений в базы данных, вычисление значений с использованием подпрограмм с помощью методов, сгенерированных из МЗ в форме исполняемых exe-модулей или dll-библиотек из других систем.

Для генерации 3D моделей необходима разработка соответствующего механизма, чему посвящена настоящая статья.

Модули знаний, представляющие собой элементарные порождающие системы, объединяют в структурированные системы, носителем моделей которых они являются. Моделью структурированных порождающих систем с точки зрения ИИ выступают семантические сети. Семантическая сеть МЗ представляет собой ациклический ориентированный граф (рис. 4). Ацикличность необходима для выполнения семантической сетью ее функционального назначения — определение значений выходных переменных по заданным входным.

В экспертном программировании ранжированная семантическая сеть МЗ генерируется автоматически. Таким образом реализуется первый элемент базисного множества структурного программирования «последовательность» (см. рис. 1, а). Второй элемент обеспечивается наличием предусловий МЗ. Третий элемент, связанный с генерацией циклов, реализуется с помощью выделенной переменной FinCalc. По переменной FinCalc формируется цикл, обеспечивающий повторное выполнение множества МЗ до изменения значения этой переменной.

С точки зрения стандарта IDEF0 ранжированная семантическая сеть МЗ реализует процесс, состоящий из операций, выполняемых МЗ. Действия по преобразованию свойств информационной модели выполняются механизмами МЗ.

Модуль знаний генерации 3D моделей. В опубликованных работах, посвященных созданию интеллектуальных CAD-систем [6–8],

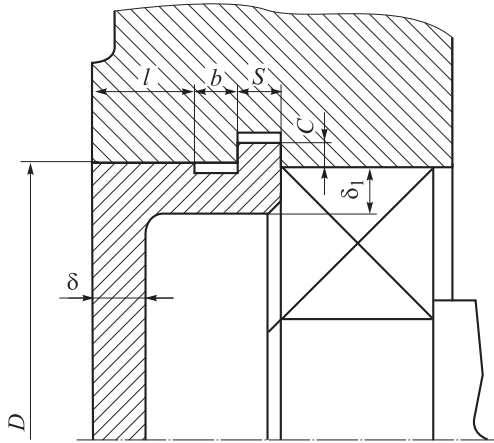


Рис. 5. Бланк-чертеж закладной крышки подшипника

не рассматривается задача разработки полуавтоматических систем проектирования с генерацией 3D моделей.

Чтобы обеспечить возможность генерации 3D моделей посредством баз знаний необходимо создать МЗ, имеющий механизм, с помощью которого на основе параметризованной в САД-системе модели детали и (или) сборочной единицы можно генерировать 3D модель изделия с рассчитанными в других модулях значениями размеров.

Продемонстрируем работу такого МЗ на примере генерации 3D модели простой детали — закладной крышки подшипника цилиндрического редуктора (рис. 5) [9]. В качестве САД-системы в примере использован SolidWorks [10].

Вначале необходимо начертить шаблон па-

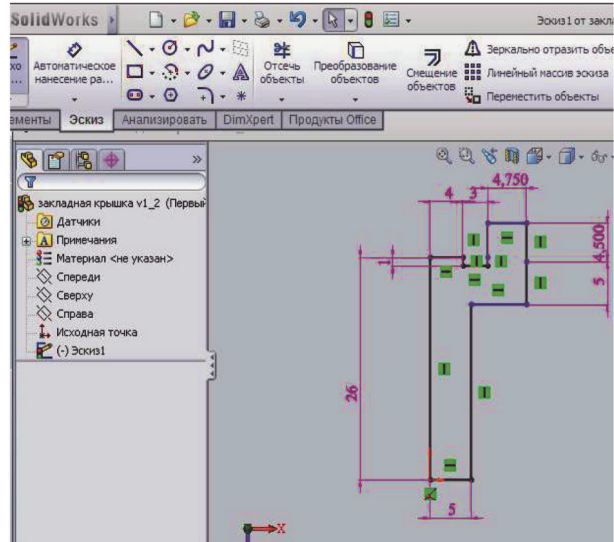


Рис. 6. Интерфейс контура образующей (полноцветную версию см. <http://www.izvuzmash.bmstu.ru>)

аметризованной 3D модели и проставить базовые размеры. Поскольку деталь, изображенная на рис. 5 является телом вращения, то шаблон представляет собой контур образующей (рис. 6).

Для параметризации модели нужно ввести зависимости размеров в таблицу уравнений SolidWorks. Вызвать диалоговое окно **Добавить уравнение** можно дважды кликнув на размер, который хотим параметризовать, и выбрать в выпадающем списке **Добавить уравнение**. В этом окне первым указывается зависимый размер, после знака «=» — управляющий размер и арифметические действия над ним, если они требуются (рис. 7).

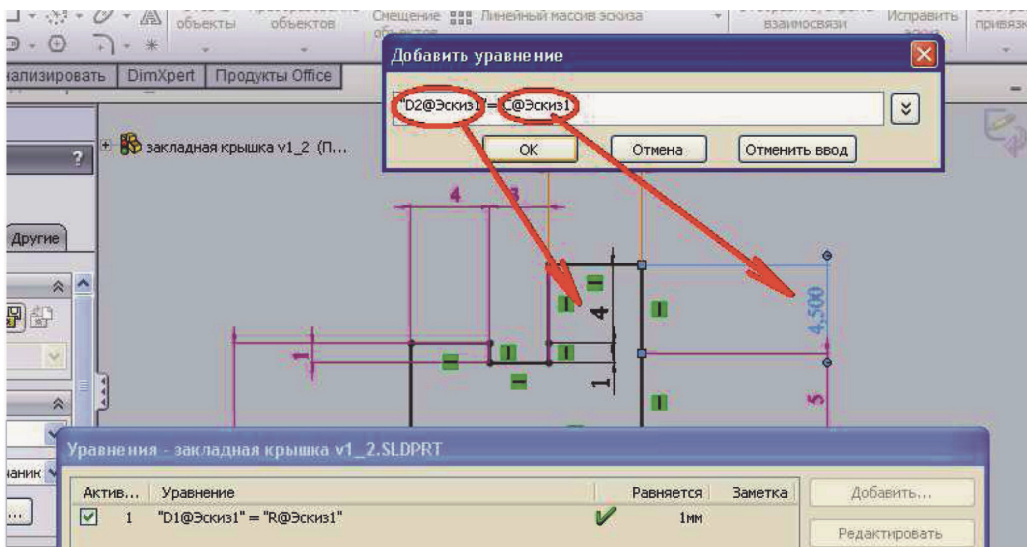


Рис. 7. Интерфейс добавления уравнений (полноцветную версию см. <http://www.izvuzmash.bmstu.ru>)

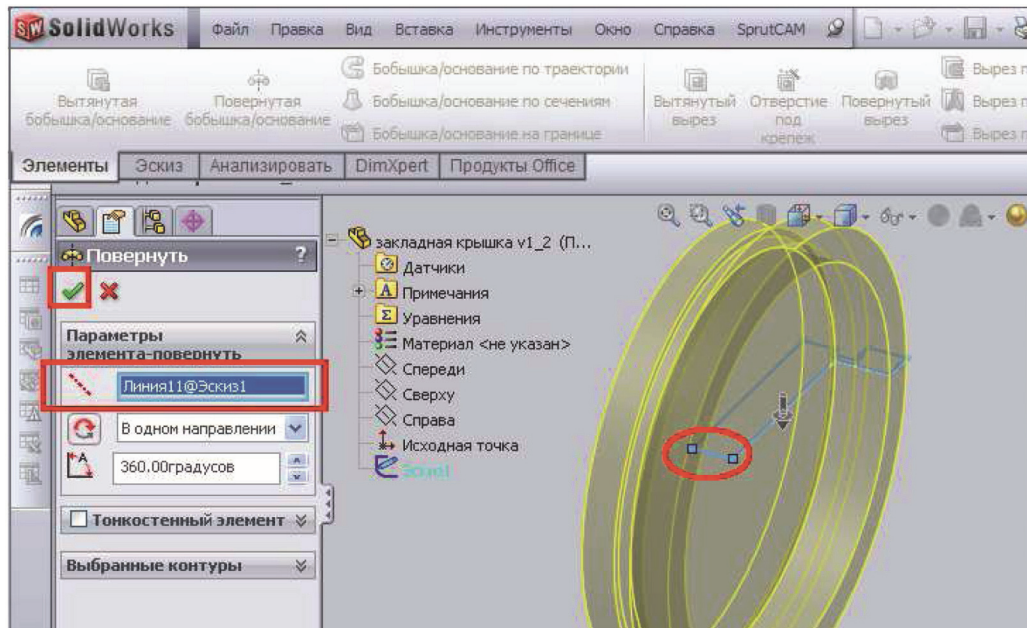


Рис. 8. Интерфейс создания 3D модели
(полноцветную версию см. <http://www.izvuzmash.bmstu.ru>)

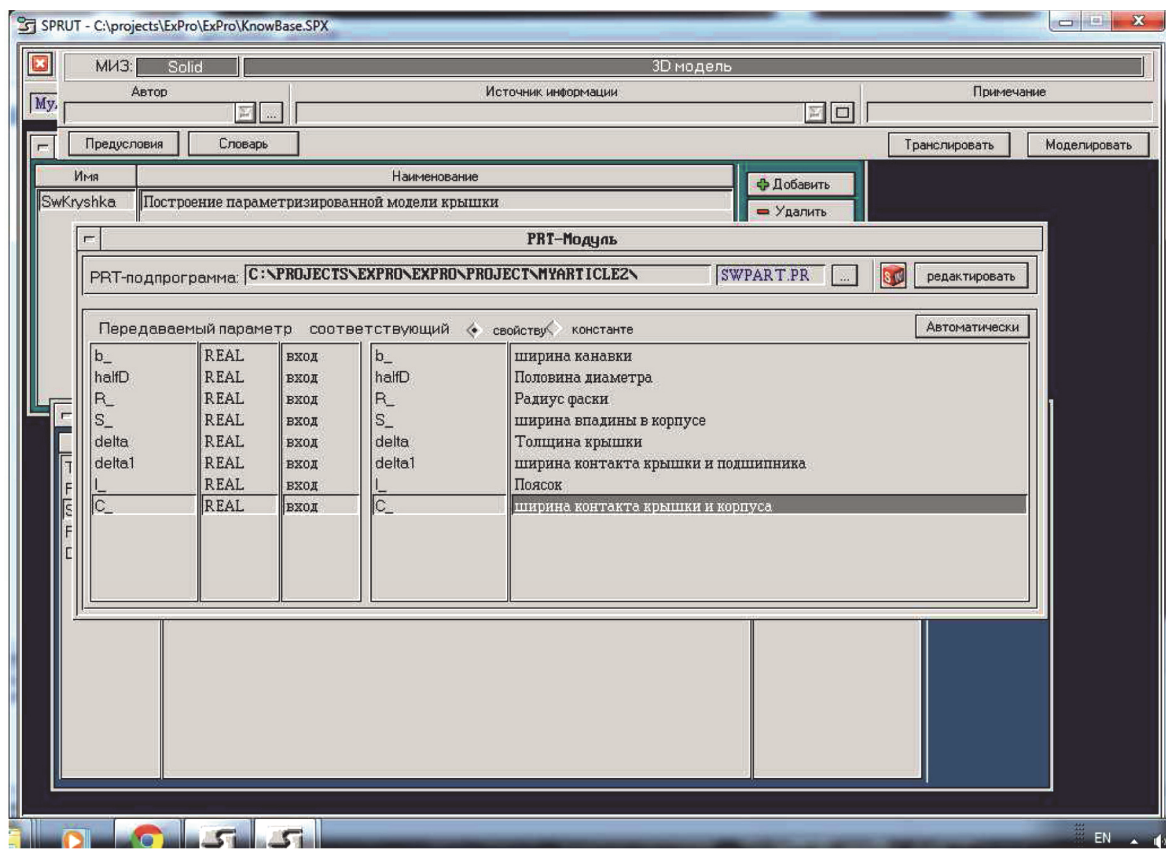


Рис. 9. Интерфейс модуля генерации 3D модели в системе Sprut ExPro
(полноцветную версию см. <http://www.izvuzmash.bmstu.ru>)

Таблица 1

Словарь метода

имя	наименование	тип	вход	выход	вид
D_	Диаметр крышки, мм	REAL	*		вход
halfD	Половина диаметра крышки, мм	REAL			локал.
S_	Ширина впадины в корпусе, мм	REAL			локал.
b_	Ширина канавки, мм	REAL			локал.
l_	Ширина пояска, мм	REAL			локал.
C_	Ширина контакта крышки и корпуса, мм	REAL			локал.
delta1	Ширина контакта крышки и подшипника, мм	REAL			локал.
delta	Толщина крышки, мм	REAL			локал.
R_	Размер фаски, мм	REAL			локал.

В диалоговое окно вводится имя размера, которое можно узнать, кликнув на интересующий размер (вкладка **Значение**, раздел **Первичное значение**). Ввести в диалоговое окно значение размера можно также кликнув на сам размер.

После завершения редактирования эскиза можно перейти к созданию 3D модели. Для этого необходимо вращать полученный контур вокруг оси (рис. 8).

Построенная 3D модель должна быть подключена к базе знаний посредством специализированного МЗ, обеспечивающего выполнение следующих функций:

1) загрузка и открытие 3D модели SolidWorks (формат файлов *.SLDASM, *.SLDPRT);

2) отображение из этой модели размеров и их значений (размеры отображаются в формате <имя размера>@<объект дерева конструирования>@<имя 3D модели>);

3) сопоставление размеров 3D модели переменным словаря Sprut ExPro (с помощью выпадающего списка);

4) формирование текста подпрограммы PRT-модуля, которая изменяет размеры 3D модели в соответствии со значением переменных из базы знаний Sprut ExPro;

5) загрузка этой подпрограммы в интерфейс «Механизм» модуля.

Интерфейс модуля генерации 3D модели в системе Sprut ExPro представлен на рис. 9.

База знаний проектирования и генерации 3D моделей. Разработка метода базы знаний начинается с создания словаря. Словарь метода проектирования закладной крышки подшипника, соответствующий рис. 5, представлен в табл. 1.

Ранжированная семантическая сеть модулей метода проектирования закладной крышки подшипника представлена в табл. 2. В данном случае используется одна входная переменная — диаметр крышки.

Таблица 2

МЗ метода

ранг	имя	наименование
1	Faska	Подбор размера фаски
1	Table	Подбор размеров по диаметру
2	Formula	Расчет размеров
3	Solid	Генерация 3D модели

Входные свойства

имя	наименование	тип	значение
D_	Диаметр крышки, мм	REAL	63

В первый ранг входят модули, имеющие в качестве входной переменной диаметр крышки. К ним относятся представленные в табл. 3 модули подбора размера фаски и ширины канавки, а также толщины крышки, механизмом которых являются таблицы.

Таблица 3

Набор табличных модулей

МЗ: "Faska" — Подбор размера фаски

Входные свойства

имя	наименование	тип	значение
D_	Диаметр крышки, мм	REAL	112

Механизм — Таблица

Конфигурация свойств в таблице

D_
R_
b_

Таблица

[10,50]	(50,100]	(100,)
1	1.6	2
3	5	8

Выходные свойства

имя	наименование	тип	значение
R_	Радиус фаски	REAL	2
b_	Ширина канавки	REAL	8

M3: "Table" — Подбор размеров по диаметру

Входные свойства

имя	наименование	тип	значение
D_	Диаметр контакта с корпусом	REAL	50

Механизм — Таблица

Конфигурация свойств в таблице

D_
delta

Таблица

[50,62]	[63,95]	[100,145]	[150,220]
5	6	7	8

Выходные свойства

имя	наименование	тип	значение
delta	Толщина крышки	REAL	5

На основе данных, приведенных в табл. 3, с помощью модуля расчета по формулам вычис-

ляются все остальные размеры, необходимые для генерации 3D модели детали (табл. 4).

Таблица 4

Модуль расчета по формулам

M3: "Formula" Расчет размеров

Входные свойства

имя	наименование	тип	значение
delta	Толщина крышки	REAL	8
b_	Ширина канавки	REAL	1

Механизм — Формула

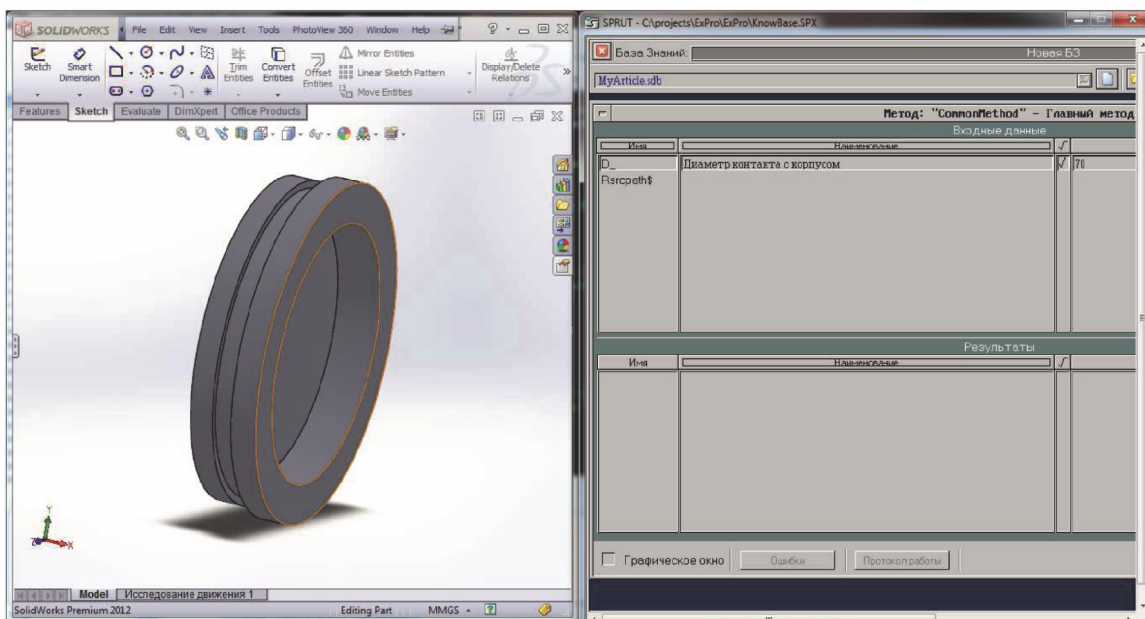
$$\begin{aligned} \text{delta1} &= 0.95 * \text{delta} \\ S_ &= 0.95 * \text{delta} \\ C_ &= 0.5 * S_ \\ l_ &= b_ \end{aligned}$$

Выходные свойства

имя	наименование	тип	значение
delta1	Ширина контакта крышки и подшипника	REAL	7.6
l_	Поясок	REAL	1
S_	Ширина впадины в корпусе	REAL	7.6
C_	Ширина контакта крышки и корпуса	REAL	3.8

Интерфейс базы знаний проектирования и генерации 3D модели закладной крышки в системе Sprut ExPro приведены на рис. 10.

Этот интерфейс (см. рис. 10) представляет собой комбинацию экрана Solid Works с изоб-



а

б

Рис. 10. Интерфейс базы знаний проектирования (а) и генерации (б) 3D модели закладной крышки в системе Sprut ExPro (полноцветную версию см. <http://www.izvuzmash.bmstu.ru>)

ражением 3D модели спроектированной детали и экрана Sprut ExPro с входным параметром метода.

Выводы

1. Технология экспертного программирования позволяет пользователю-непрограммисту ставить и решать свои, традиционно считающиеся интеллектуальными задачи, общаясь с ЭВМ на ограниченном подмножестве естественного языка.

2. Предложенная технология является дальнейшим развитием методов структурного, модульного и объектного программирования и основывается на понятии объект-функция.

3. Автоматическая генерация семантической сети МЗ обеспечивает преобразование экспертной системы в систему программирования.

4. Разработанный МЗ генерации 3D моделей позволяет создавать полуавтоматические системы автоматизированного проектирования и быстрого прототипирования изделий машиностроения.

Литература

- [1] Евгеньев Г.Б. *Интеллектуальные системы проектирования*. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2012. 420 с.
- [2] Антонов А.А., Мурованная Е.Ю. Твердотельное моделирование? Нет! — Проектирование. *САПР и графика*, 2000, № 5.
- [3] Гаврилова Т.А., Хорошевский В.Ф. *Базы знаний интеллектуальных систем*. Санкт-Петербург, Питер, 2000. 384 с.
- [4] Рыбина Г.В. *Основы построения интеллектуальных систем*. Москва, Финансы и статистика, ИНФРА-М, 2010. 432 с.
- [5] *Integration definition for function modeling (IDEF0)*. Draft Federal Information Processing Standards Publication 183, 1993. 116 p.
- [6] Balic J. Intelligent CAD/CAM system for CNC programming-An Overview. *Advances in Production Engineering & Management*, 2006, no. 1, pp. 13–22.
- [7] Marichal G.N., Hernandez A., Gonzalez E.J., Acosta L., Saorin J.L. *3D Modelling and Artificial Intelligence: A Descriptive Overview*, 2010, 11 p. URL: <http://www.irma-international.org/viewtitle/43145/> (дата обращения 15 января 2015).
- [8] *Parametric Technology Corp*. URL: <http://ru.ptc.com/> (дата обращения 15 января 2015).
- [9] Дунаев П.Ф., Леликов О.П. *Детали машин. Курсовое проектирование*. Москва, Машиностроение, 2004. 560 с.
- [10] *SolidWorks Russia*. URL: <http://www.solidworks.ru/> (дата обращения 15 января 2015).

References

- [1] Evgenev G.B. *Intellektual'nye sistemy proektirovaniia* [Intelligent systems design]. Moscow, Bauman Press, 2012. 420 p.
- [2] Antonov A.A., Murovannaia E.Iu. Tverdotel'noe modelirovanie? Net! — Proektirovanie [Solid modeling? No! — Design]. *SAPR i grafika* [CAD and Graphics]. 2000, no. 5.
- [3] Gavrilova T.A., Khoroshevskii V.F. *Bazy znaniia intellektual'nykh sistem* [Knowledge Base Intelligent Systems]. Sankt-Peterburg, Piter publ., 2000. 384 p.
- [4] Rybina G.V. *Osnovy postroeniia intellektual'nykh sistem* [Fundamentals of building intelligent systems]. Moscow, Finansy i statistika, INFRA-M publ., 2010. 432 p.
- [5] *Integration definition for function modeling (IDEF0)*. Draft Federal Information Processing Standards Publication 183, 1993. 116 p.
- [6] Balic J. Intelligent CAD/CAM system for CNC programming-An Overview. *Advances in Production Engineering & Management*, 2006, no. 1, pp. 13–22.
- [7] Marichal G.N., Hernandez A., Gonzalez E.J., Acosta L., Saorin J.L. *3D Modelling and Artificial Intelligence: A Descriptive Overview*, 2010, 11 p. Available at: <http://www.irma-international.org/viewtitle/43145/> (accessed 15 January 2015).
- [8] *Parametric Technology Corp*. Available at: <http://ru.ptc.com/> (accessed 15 January 2015).
- [9] Dunaev P.F., Lelikov O.P. *Detali mashin. Kursovoe proektirovanie* [Machine parts. Course design]. Moscow, Mashinostroenie publ., 2004. 560 p.
- [10] *SolidWorks Russia*. Available at: <http://www.solidworks.ru/> (accessed 15 January 2015).

Статья поступила в редакцию 24.02.2015

Информация об авторах

ЕВГЕНЕВ Георгий Борисович (Москва) — доктор технических наук, профессор кафедры «Компьютерные системы автоматизации производства». МГТУ им. Н.Э. Баумана (105005, Москва, Российская Федерация, 2-я Бауманская ул., д. 5, стр. 1, e-mail: info@rkg.bmstu.ru).

КОКОРЕВ Александр Александрович (Москва) — ассистент кафедры «Компьютерные системы автоматизации производства». МГТУ им. Н.Э. Баумана (105005, Москва, Российская Федерация, 2-я Бауманская ул., д. 5, стр. 1).

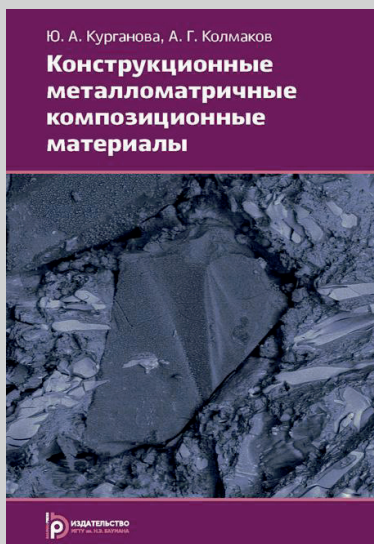
ПИРИМЯШКИН Михаил Владимирович (Москва) — программист. Московский центр технологической модернизации образования ГБОУ «ТемоЦентр» (107076, Москва, Российская Федерация, 1-й Зборовский пер., д. 3).

Information about the authors

EVGENEV Georgy Borisovich (Moscow) — Doctor of Science (Eng.), Professor, Department of Computer Systems of Manufacture Automation. BMSTU (105005, Moscow, Russian Federation, 2nd Baumanskaya St., Bldg. 5, Block 1, e-mail: info@rkg.bmstu.ru).

KOKOREV Alexander Alexandrovich (Moscow) — Assistant, Department of Computer Systems of Manufacture Automation. BMSTU (105005, Moscow, Russian Federation, 2nd Baumanskaya St., Bldg. 5, Block 1).

PIRIMYASHKIN Mikhail Vladimirovich (Moscow) — Programmer. Moscow Centre of Technological Modernization of Education «TemoCenter» (107076, Moscow, Russian Federation, 1st Zbrovsky Per., Bldg. 3).



В Издательстве МГТУ им. Н.Э. Баумана
вышло в свет учебное пособие
Ю.А. Кургановой, А.Г. Колмакова

«Конструкционные металломатричные композиционные материалы»

Изложены основные понятия, относящиеся к науке о композиционных материалах. Рассмотрены классификация, основные способы получения, особенности соединения компонентов металломатричных композиционных материалов, методы исследования их структуры, механические, технологические и эксплуатационные свойства. Теоретически и экспериментально обоснована целесообразность использования дисперсно-упрочненных и волокнистых композиционных материалов на основе металлических сплавов в области авиа-, ракетно- и автомобилестроения.

По вопросам приобретения обращайтесь:

105005, Москва, 2-я Бауманская ул., д. 5, стр. 1.
Тел.: +7 499 263-60-45, факс: +7 499 261-45-97;
press@bmstu.ru; www.baumanpress.ru