

Автоматическая проверка решений задач инженерной геометрии*

А.А. Бойков, А.М. Федотов
albophx@mail.ru

Россия, Ивановский государственный энергетический университет им. В. И. Ленина

В работе предложен метод автоматической проверки решений конструктивных задач инженерной геометрии, основанный на установлении соответствия между решением как множеством геометрических объектов и семантической структурой решения, задаваемой эталоном. Рассмотрен порядок создания эталона для проверки решения. Приведен пример проверки решения задачи.

Ключевые слова: автоматическая проверка решений, инженерная геометрия, структурный анализ, формальные грамматики, Интернет, дистанционное обучение

Automated verification of engineering geometry task solution*

A.A. Boykov, A.M. Fedotov
Russia, Ivanovo State Power University

The method of automated verification of engineering geometry constructive task solution is proposed. The method is based on the establishment of the correspondence between the solution as the set of geometric objects and the semantic structure of the solution given by the pattern. The technology of creating a pattern is described. Verification example is shown. The system of automated verification of the solution presented in DXF file format was developed. It can be embedded into Internet system or distance learning system.

Keywords: Automated solution verification, engineering geometry, structural analysis, formal grammar, Internet, distance learning

1. Введение

Основу задач инженерной геометрии составляют различные комбинации геометрических элементов, формирующих изображения проектируемого объекта. Эта геометро-графическая информация имеет сложную структуру вне зависимости от того, на каком носителе она находится – бумажном или электронном. Широкое внедрение вычислительной техники в инженерную и научную деятельность создает противоречие между точностью построений при решении задач и способностью человека осуществлять проверку решений, полученных с применением современных систем. Поэтому становится актуальной разработка методов и систем автоматической проверки решений конструктивных задач инженерной геометрии.

2. Анализ проблемы и постановка задачи

Изображения проектируемых объектов представляют собой конструктивные геометрические модели в том смысле, что может быть установлена их эквивалентность проектируемым объектам по заранее выбранным свойствам и признакам [4, 5]. При этом часто моделью некоторого объекта является группа элементов. Полное описание моделируемого объекта A включает в себя набор параметров формы и положения ($p_1 \sim p_n$) и уточняющих атрибутов ($u_1 \sim u_m$). Моделью A' объекта

A является группа элементов $a_1 \sim a_k$, общее число параметров –

$$n'_A = \sum_{i=1}^k n_i \geq n_A$$

, атрибутов –

$$m'_A = \sum_{i=1}^k m_i \geq m_A$$

, где n_i и m_i – число параметров и атрибутов i -го элемента модели (a_i), n_A и m_A – число параметров и атрибутов моделируемого объекта A , соответственно. Для эквивалентности модели объекту необходимо, чтобы число параметров и атрибутов модели было равно числу параметров и атрибутов объекта. Если $n'_A > n_A$ и $m'_A > m_A$, то на параметры и атрибуты элементов модели накладываются условия либо значения игнорируются.

В качестве примера модели рассмотрим комплексный чертеж точки. Точка в пространстве задается тремя параметрами (X, Y, Z) и одним атрибутом (имя), и моделируется двумя или более проекциями, каждая из которых имеет два параметра (X', Y') и четыре атрибута (имя, индекс, имя_Х', имя_У'). На атрибуты проекций накладываются условия: имена должны совпадать, индексы должны иметь определенные значения (например, 1 и 2). На параметры проекций накладывается условие зависимости (например, равенство координат X'), значения имя_Х' и имя_У' игнорируются.

Если объект A , в свою очередь, состоит из подобъектов $F_1 \sim F_r$, тогда каждый i -й подобъект модели-

Работа опубликована по гранту РФФИ №16-07-20482

руется группой $f_{i,1} - f_{i,r_i}$, а всякая связь между парой $F_s F_t$ выражается в виде одной или нескольких связей между $f_{s,i} f_{t,j}$ ($1 \leq i \leq r_s, 1 \leq j \leq r_t$).

Решением конструктивной задачи инженерной геометрии в соответствии с [1, 6] является геометрический объект A , связанный с исходными объектами $B_1 - B_q$ набором условий, причем объект A должен быть представлен моделью A' , описанной выше. Объект A появляется в результате последовательных построений, которые выполняются в соответствии с [7] путем выбора вида фигуры и указания значений параметров или условий, связывающих фигуру с имеющимися.

Анализ работ, посвященных проблеме автоматической проверки решений задач инженерной геометрии [4], показал, что она исследована недостаточно, в частности, имеющиеся методы не учитывают возможность представления объектов группами фигур и не применимы для проверки задач, имеющих множества правильных решений.

Представим модель решения конструктивной задачи инженерной геометрии в виде кортежа $S = \langle O, R, G, A, E \rangle$, где O – множество объектов, составляющих решение, R – множество связей между объектами, G – множество видов связей, A – множество видов объектов (алфавит), E – совокупность свойств пространства, в котором рассматриваются объекты O и связи R . Свойства пространства определяют алфавит $A=A(E)$ и вместе с алфавитом – состав возможных связей $G=G(A, E)$. Вместе G, A и E определяют формальную грамматику, которая порождает язык моделирования $L=L(G, A, E)$ как множество всех геометрических объектов, которые могут быть описаны способом, показанным выше. Правильные решения составляют фрагмент языка $L_R \subset L$. Грамматика G_R , порождающая L_R , контекстно-зависима, поэтому задача распознавания решения S , в общем случае, не формальная.

3. Метод проверки

Анализ действий эксперта, проверяющего решение задачи, показывает, что проверка осуществляется путем сравнения решения с некоторым эталоном (условие задачи). По аналогии автоматическая проверка решения может быть осуществлена путем сравнения модели решения S с эталоном T . При этом необходимо установить соответствие S и T , определить их качественные и количественные различия.

Проверка решения любой задачи включает следующее:

- 1) выбор эталона;
- 2) установление соответствия между S и T ;
- 3) определение меры различия S и T .

Будем автоматизировать шаги 2-3 следующим образом:

1. На основе состава объектов и структуры эталона

строится контекстно-свободная (КС) грамматика.

2. Осуществляется разбор S , при этом каждому элементу T будет поставлен в соответствие какой-нибудь элемент S . Полученное в результате дерево разбора в целом изоморфно эталону и является своего рода гипотезой о семантике модели S .

3. Вычисляется мера различия.

4. Находятся все возможные гипотезы и соответствующие меры.

5. Гипотеза с минимальным значением меры различия – качественный и количественный результат сравнения S и T .

КС-грамматика строится следующим образом:

1. Перечисляются все элементы решения – множество терминальных символов грамматики.

2. Обязательные элементы объединяются в группы при помощи символа `and`.

3. Альтернативные варианты элементов и групп объединяются при помощи символа `or`.

4. Подвижные и изменяемые элементы и группы заключаются в символы `transform`.

5. Элементы, требующие проверки условий, заменяются на символ `instance` с перечислением условий.

Рассмотрим проверку задачи о пересечении прямой с плоскостью общего положения (рис. 1, а). Грамматика в традиционной записи выглядит следующим образом (проекция треугольника считаются заданными и проверки не требуют, терминальные символы для наглядности заключены в фигурные скобки):

$T = \text{transform}_1$

$\text{transform}_1 = \text{and}_1$

$\text{and}_1 = \{F_2\} \{F_1\} \{F_2M_2\} \{F_2Z_2\} \{2_2N_2\} \{3_1M_1\} \{3_1F_1\} \{F_1N_1\}$ `or`

`or` = and_2 | and_3

$\text{and}_2 = \{\beta_2\} \{1_2\} \{2_2\} \{1_1\} \{2_1\} \{1_2, 1_1\}$

$\text{and}_3 = \{\alpha_1\} \{3_1\} \{4_1\} \{3_2\} \{4_2\} \{3_2, 4_2\}$

После этого каждый терминальный символ необходимо заменить символом объекта (проекция точки, луч, отрезок и др.), перечислить значения параметров и атрибутов и указать опции проверки. Этот этап наиболее трудоемкий, и далее будет показано, как его автоматизировать.

Терминальными символами этой грамматики являются геометрические объекты, которые, в отличие от символов традиционных грамматик [2], распределены в пространстве E . Построенная грамматика, как и задача, довольно проста. Нетерминальный символ `and` при разборе обеспечивает обязательное присутствие в модели перечисленных объектов, не ограничивая их взаимного расположения. Строгие символы (выше, ниже, внутри и др.) обеспечивают распознавание более сложных конструкций.

В результате разбора решения задачи (рис. 1, а)

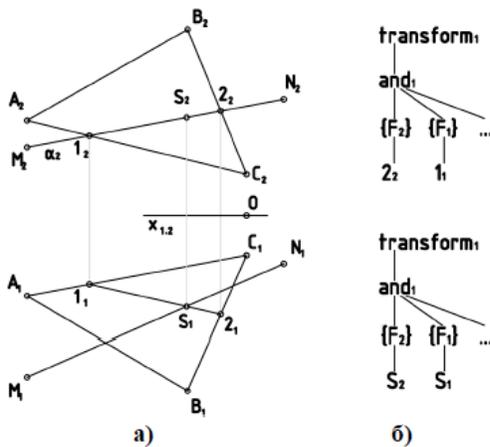


Рис. 1: Решение задачи (а) и варианты дерева разбора (б).

будут сформированы варианты формального соответствия (фрагменты приведены на рис. 1, б), каждый из которых затем подвергается геометрической проверке на соответствие эталону. Обходя дерево разбора, система в каждом узле рассчитывает меру различия с соответствующим узлом дерева эталона. Мера различия для терминальных символов (фигур) определяется следующим выражением:

$$d(A, A') = \sum_{i=1}^n w_i^p \cdot (p_i - p'_i)^2 + \sum_{i=1}^m w_i^m \cdot \text{cmp}_i(u_i, u'_i)$$

где w_i – «вес» параметра или атрибута, cmp_i – оператор сравнения для i -го атрибута. В узлах нетерминальных символов мера различия определяется рекурсивно на основе мер различия дочерних узлов:

$$d_{\text{and}}(A, A') = \sum_{i=1}^r d(a_i, a'_i),$$

$$d_{\text{or}}(A, A') = \min_{i=1}^r d(a_i, a'_i),$$

$$d_{\text{transform}}(A, A') = \min_{i=1}^t d(\text{TR}_i \langle A \rangle, A')$$

где $\text{TR}_i \langle A \rangle$ – объект, полученный в результате преобразования объекта A , с использованием i -го набора параметров преобразования, t – число таких наборов.

Установление соответствия и проверка совмещены. Так, получив при разборе список вакантных соответствий для объекта $\{F_2\}$, система для каждого рассчитывает текущее значение меры различия и, если значение не превышает порогового, переходит к объекту $\{F_1\}$ и т. д.

4. Реализация метода проверки

Самым трудоемким этапом создания эталона является перечисление значений параметров и настройка опций проверки. В практической реализации

использован следующий алгоритм формирования эталона:

1. Создание геометрических объектов в подходящем редакторе (одновременно происходит настройка параметров и атрибутов). Файл представляет собой базу данных параметров терминальных символов.
2. Обработка файла, формирование списка терминальных символов, присвоение идентификаторов. Настройка опций проверки осуществляется вручную для каждого элемента списка.
3. Создание текстовой части эталона, в которой терминальными символами являются идентификаторы объектов (ссылки на объекты).

Эталон к рассмотренной ранее задаче включает геометрическую часть (рис. 2, а) и текстовую (рис. 2, б). В текстовом описании выполнена подстановка нетерминальных символов, в фигурные скобки заключены ссылки на объекты, извлеченные из геометрической части эталона. В язык описания структуры эталона, кроме того, включены операторы присваивания и переменные (знак \$), локальные опции проверки и символы связей для всех объектов, кроме `and` и `or`.

На вход системы проверки подается файл в формате геометрического редактора. Это создает ряд проблем, связанных с отсутствием среди стандартных инструментов редактора объектов предметной области.

Во-первых, структура эталона тесно связана с семантикой решения, которая может быть выражена только в терминах предметной области.

Современные геометрические редакторы имеют возможность расширения набора инструментов за счет применения блоков, фрагментов и др. Мы будем использовать такие объекты для замещения объектов предметной области, отсутствующих среди инструментов геометрического редактора. В примере такими объектами являются проекция точки, плоскости и др.

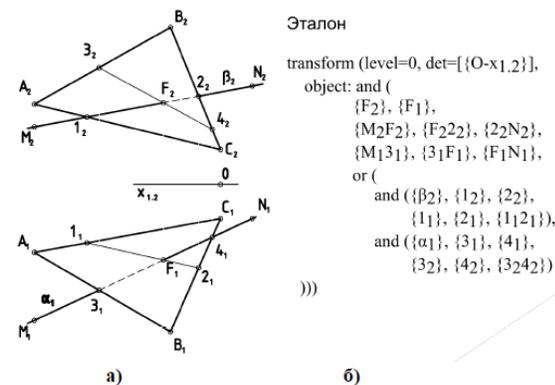


Рис. 2: Эталон к проверке задачи из примера.

Во-вторых, в проверяемом решении объекты предметной области также представлены группами фигур.

Для распознавания структуры, представленной эталоном, мы будем использовать две стратегии:

- преобразование всех терминальных символов в нетерминальные с раскрытием их собственного состава – в примере выше проекция точки может быть представлена как `pointObject = point caption`, встречая в тексте эталона символ `pointObject`, система будет отыскивать изображение точки (`point`) и текстовое поле (`caption`), содержащее имя и индекс проекции; каждый новый нетерминальный символ требует программной реализации;
- предварительная обработка проверяемого файла, поиск комбинаций стандартных элементов (обозначение точки и текстовое поле \mapsto `pointObject`) и объединение их в объекты предметной области; предварительная обработка требует программной реализации. Наличие в системе нескольких блоков предварительной обработки позволяет работать с разными форматами файлов, основная проверка выполняется в терминах предметной области.

На рис. 3 приведен эталон для проверки решения задачи о построении прямой параллельно плоскости треугольника. В текстовой части эталона использованы дополнительные нетерминальные символы `dg3d:point` и `dg3d:plane`, осуществляющие распознавание моделей точки трехмерного пространства, заданной парой проекций, и плоскости трехмерного пространства, заданной проекциями трех точек.

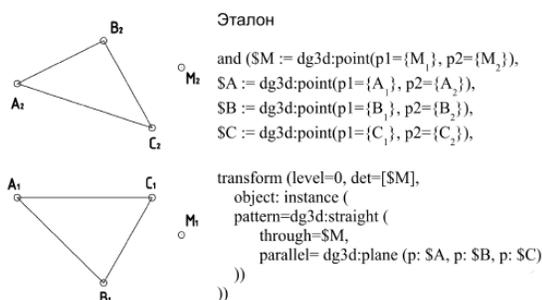


Рис. 3: Эталон с дополнительными символами

На рис. 4 показан результат работы блока предварительной обработки. Результат проверки решения задачи о построении перпендикуляра к плоскости треугольника показан на рис. 5 (изображение эталона и решения совмещены в точке начала координат).

5. Заключение

В данной работе предложен более общий, по сравнению с предыдущими подходами, метод проверки решений конструктивных задач инженерной геометрии. Он учитывает возможность представления объектов группами фигур и позволяет выполнять проверку задач, имеющих множества решений. Метод был опробован на решениях широкого круга задач начертательной геометрии и инженерной графики и показал свою эффективность.

Была создана система, осуществляющая проверку решений задач, представленных в виде файла в формате DXF. Программный код написан на языке РНР и естественным образом может быть внедрен в состав Интернет-систем (в том числе дистанционного обучения). Расширение возможностей системы достигается введением в грамматику новых нетерминальных символов и реализацией связанных с ними подпрограмм.

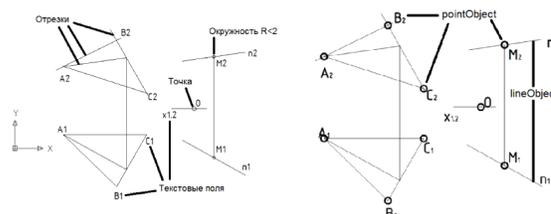


Рис. 4: Модель до (слева) и после (справа) предварительной обработки

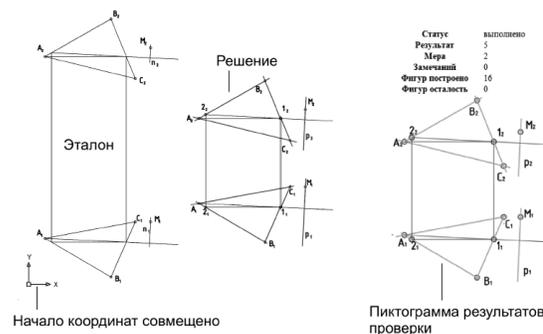


Рис. 5: Пример проверки решения задачи разработанной системой

Литература

- [1] Аргунов Б.И., Балк М.Б. Геометрические построения на плоскости. – М., 1957. – 268 с.
- [2] Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. – М.: Мир, 1978. – Т. 1. – 612 с.
- [3] Бойков А. А. Автоматизация проверки инженернографических заданий. – URL: <http://dgng.pstu.ru/conf2016/papers/97/> (дата обращения: 15.04.2016)

- [4] Вальков К. И. Введение в теорию моделирования. – Л.: ЛИСИ, 1973. – 152 с.
- [5] Волошинов Д. В. Конструктивное геометрическое моделирование. – Saarbrücken: Lambert Academic Publishing, 2010. – 355 с.
- [6] Глоговский В.В. Элементарные конструктивные задачи по начертательной геометрии. – Львов: Вища школа, 1981. – 152 с.
- [7] Полозов В. С. Автоматизированное проектирование / В. С. Полозов [и др.]. – М.: Машиностроение, 1983. – С. 28–54.