

УДК 004.434

**ФОРМАЛИЗАЦИЯ МОДЕЛЕЙ МНОГОСЛОЙНЫХ ТОНКОСТЕННЫХ
КОНСТРУКЦИЙ В САПР****С. В. ЧОПОРОВ*, С. И. ГОМЕНЮК, Б. Г. М. АЛЬ-АТЕМНЕХ, М. А. В. АЛЬ-ОМАРИ,
Х. Х. АЛАТЕМНЕХ**

Кафедра математического моделирования, Запорожский национальный университет, Запорожье, УКРАИНА
*email: s.choporoff@znu.edu.ua

АННОТАЦИЯ В работе рассмотрена проблема формализации задач исследования напряженно-деформированного состояния многослойных тонкостенных конструкций. Предложен соответствующий проблемно-ориентированный язык описания тонких многослойных пластин и их исследования с использованием метода конечных элементов в системах автоматизированного проектирования. В основу проблемно-ориентированного языка заложен стандарт ЕСМА-262, что позволяет использовать относительно несложный в изучении синтаксис и упрощает описание сложных моделей.

Ключевые слова: многослойные конструкции, проблемно-ориентированный язык, метод конечных элементов, САПР, ЕСМА-262

LAYERED THIN PLATES FORMALIZATION IN CAE SYSTEMS**S. CHOPOROV*, S. GOMENYUK, B. AL ATHAMNEH, M. AL-OMARY, H. ALATAMNEH**

Department of Mathematical Modelling, Zaporizhzhya National University, Zaporizhzhya, UKRAINE

ABSTRACT In the herein paper is decried the domain specific language for the finite element analysis of thin layered plates using computer aided engineering. The main objective of the paper is development of domain specific language for thin plates and shells analysis using the Finite element method. The first-order shear deformation theory of thin layered plates is used. The domain specific language is based on ECMA-262. This language allows users to describe geometric model and simulation parameters (nodal and distributed loads, boundary conditions). Implicit functions are used for definition of geometric models. Loads and boundary conditions can be defined using a special class of the language objects. Logical functions are parameters of this class. These functions should be defined in terms of logical expressions that true for a set of points in which condition or load is applied. Developed domain specific language is embedded in computer aided engineering -software. Examples of layered circular plate stress-strain analysis are written. Scientific novelty of the paper is development of new principles of building and application of the domain specific language for analysis of thin layered plates using the Finite element method.

Keywords: layered plate, domain specific language, the Finite element method, computer aided engineering, ECMA-262

Введение

Одним из наиболее сложных этапов проектирования инженерных конструкций и сооружений является анализ их прочностных характеристик. При разработке современных объектов аэрокосмической, судостроительной и др. отраслей возникают задачи определения характеристик многослойных тонкостенных конструкций: параметров их напряженно-деформированного состояния (НДС), критических или разрушающих нагрузок и т.п. Автоматизация их решения приводит к проблеме формализации соответствующих математических моделей в форме, пригодной для компьютерной обработки.

Анализ литературы

Наиболее распространенным способом решения проблемы формализации исходных данных в САПР является разработка лингвистического обеспечения – специализированных входных языков, состоящих из двух частей: языка описания объектов и

языка описания заданий [1]. Язык описания объектов в контексте поставленной задачи должен предоставлять средства описания геометрических моделей и физико-механических свойств объекта. Язык описания заданий должен включать средства выполнения определенных проектных операций и процедур.

На сегодня для описания геометрических моделей представлены визуальные средства и проблемно-ориентированные языки. Визуальные средства описания геометрических моделей (например, AutoCAD, Компас-3D и др.), как правило ориентированы на оформление проектной и конструкторской документации. Проблемно-ориентированные языки для описания геометрических моделей (AML/X [2-3], PlaSM [4], BDRY [5] и другие) [6-7], представляют собой декларативные или функциональные языки, основанные на твердотельном (CSG) или граничном (BRep) представлении объекта. Их основным недостатком является сложность организации прямой и обратной связи с процессом конечно-элементного анализа.

Также активно разрабатываются средства формализации задач, решаемых с помощью метода конечных элементов (МКЭ) на основе проблемно-ориентированных языков [8]. В частности, предложены средства создания встраиваемых проблемно-ориентированных языков (embedded domain specific languages) путем трансляции кода в программы на языке C++ (например, трансляция кода программ языка Scala [9]), основанные на прототипизировании и шаблонах проблемно-ориентированные расширения C++ [10 - 11], которые упрощают разработку программ и модулей на базе МКЭ. Основным недостатком такого подхода является необходимость знания соответствующих подмножеств C++.

Существуют также работы по созданию универсальных языков для описания задач с использованием МКЭ [12 - 13]. Основным недостатком указанных работ является возможность использования только треугольников для плоских задач и тетраэдров для трехмерных, а также отсутствие средств описания многослойных конструкций.

Цель работы

Таким образом, целью работы является разработка способа формализации тонкостенных конструкций в САПР.

Объект исследования – процесс проектирования сложных технических объектов.

Предмет исследования – методы и средства формализации моделей сложных технических объектов.

Изложение основного материала

В общем случае принципиальная схема автоматизированного проектирования многослойных тонкостенных конструкций может быть представлена в виде следующего алгоритма (рис. 1). Обработкой входных параметров занимается блок работы проблемно-ориентированного языка (DSL). Затем в блоке компьютерного анализа (CAE) препроцессор формирует соответствующую дискретную модель объекта, процессор МКЭ получает необходимые для модуля оптимизации данные (параметры НДС, критические или разрушающие нагрузки и т.п.). По результатам обработки в модуле оптимизации вносятся изменения в дискретную и расчетные модели. Следовательно, при таком подходе лингвистическое обеспечение САПР должно позволять описывать геометрическую модель, физико-механические характеристики (ФМХ) слоев и нагрузки.

Одним из возможных путей решения задачи создания проблемно-ориентированного языка может быть расширение синтаксиса, определенного стандартом универсального языка программирования.

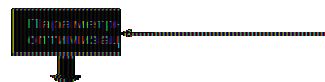


Рис. 1 – Принципиальная схема автоматизированного проектирования

Использование в качестве базы стандарта, на котором основываются популярные языки программирования, позволяет упростить изучение синтаксиса проблемно-ориентированного языка. В настоящей работе предложен язык моделирования дискретных объектов, основанный на стандарте ЕСМА-262 [14]. В основе стандарта ЕСМА лежат несколько исходных технологий, наиболее известными из которых является JavaScript. Среди его преимуществ можно выделить динамическую типизацию, которая позволяет существенно сократить код моделей.

Стандарт ЕСМА определяет спецификации объектно-ориентированного языка для прикладных вычислений и манипулирования вычислительными объектами хост-окружения. Базовый синтаксис такого языка можно представить при помощи расширенной формы Бэкуса-Наура (рис. 2). Программа представляет собой набор предписаний и деклараций функций, с помощью которых определяются геометрические модели и параметры расчета. В качестве алфавита для идентификаторов используются латинские буквы. Строковые константы могут состоять из произвольного набора символов таблицы Unicode, заключенных в двойные или одинарные кавычки.

Все переменные при таком синтаксисе (рис. 2) рассматриваются как объекты, для которых можно определить поля и методы с заданными областями видимости. Объявление переменной начинается с ключевого слова var, за которым после пробельного символа следует список переменных, разделённых запятыми с необязательной инициализацией.

Особенности создания проблемно-ориентированного языка на базе приведенного синтаксиса для определения геометрических моделей путем логического конструирования, используя неявные функции, рассмотрены в предыдущих работах авторов [15-16]. Соответственно к формальному описанию базового синтаксиса языка (рис. 2) необходимо добавить синтаксические конструкции для описания задач исследования тонкостенных конструкций при помощи МКЭ.

Для расчета НДС многослойных пластин и оболочек в современных САПР активно используется теория сдвиговых деформаций первого порядка [17], которая является обобщением теории Тимошенко-

```

программа = { предписание | декларация функции } ;
предписание = блок | пустое выражение | ( объявление переменных, ';' ) | ( выражение, ';' ) |
    условное выражение | циклическое выражение | стоп итерация | стоп цикл | выражение возврата
    | выражение переключения | пробел ;
пустое выражение = ';' ;
блок = '{', { предписание }, '}' ;
объявление переменных = 'var', пробел, идентификатор, [ '=', выражение ], { ',', идентификатор,
    [ '=', выражение ] } ;
выражение = вызов функции | декларация функции | ( идентификатор, присваивание, ( вычисляемое
    выражение | вызов функции ) ) | создание объекта | инкремент | декремент ;
условное выражение = 'if', '(', выражение, ')', предписание, { 'else', предписание } ;
циклическое выражение = ( 'do', предписание, 'while', '(', выражение, ')', ';' ) |
    ( 'while', '(', выражение, ')', предписание ) |
    ( 'for', '(', ( ( выражение, ';', выражение, ';', выражение ) | ( инициализация переменной,
        ';', выражение, ';', выражение ) | ( выражение, 'in', выражение ) | ( инициализация
        переменной, 'in' выражение ), ')', предписание ) ;
стоп итерация = 'continue', ';' ;
стоп цикл = 'break', ';' ;
выражение возврата = 'return', { выражение }, ';' ;
выражение переключения = 'switch', '(', выражение, ')', '{'
    { 'case', выражение, ':', [ предписание ] }, [ 'default', ':' [ предписание ] ], '}' ;
вычисляемое выражение = простое выражение | условное выражение | логическое выражение ;
условное выражение = '(', выражение, ')', '?', выражение, ':', выражение ;
инициализация переменной = 'var', пробел, идентификатор, '=', вычисляемое выражение ;
присваивание = '=' | '+=' | '-=' | '*=' | '/=' ;
создание объекта = 'new', пробел, идентификатор, ( выражение ) ;
декларация функции = 'function', { пробел, идентификатор, } '(' [ идентификатор, [ { ',',
    идентификатор } ], ')', предписание ;
вызов функции = идентификатор, '(', [ выражение, [ { ',', выражение } ], ')' ;
инкремент = ('++', идентификатор) | (идентификатор, '++') ;
декремент = ('--', идентификатор) | (идентификатор, '--') ;
логическое выражение = лог_и [ { '||', лог_и } ] ;
лог_и = простое выражение, [ '&&', простое выражение ] ;
простое выражение = арифметическое выражение, [ оператор отношения, арифметическое выражение ] ;
оператор отношения = '<' | '<=' | '>' | '>=' | '==' | '!=' | '===' | '!==' ;
арифметическое выражение = [ '+' | '-' ] терм { ('+' | '-'), терм } ;
терм = фактор { ('*' | '/' | '%'), фактор } ;
фактор = идентификатор | константа | строка | выражение | ('!', фактор) ;
константа = [ '+' | '-' ] число [ '.' [число] ] [ ('e' | 'E') [ '+' | '-' ] число ] ;
число = цифра { цифра } ;
цифра = '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9' ;
идентификатор = буква { буква | цифра | '_' } ;

```

Рис. 2 – РБНФ базового синтаксиса языка

Миндлина для пластин и базируется на следующих гипотезах: а) слои материала идеально соединены друг с другом; б) материал каждого слоя является упругим и ортотропным; в) каждый слой обладает постоянной толщиной; г) перемещения и деформации малы; д) слои материала несжимаемые; е) нормаль к срединной поверхности не остается перпендикулярной к ней после деформации. Локальная матрица жесткости билинейного четырехугольного элемента в предположении, что плоскость xOy совпадает со срединной плоскостью недеформированной пластинки, может быть представлена в виде

$$\begin{aligned}
 \mathbf{K}_e = & \sum_{k=1}^n \left\{ (z_k - z_{k-1}) \int_{-1}^1 \int_{-1}^1 \mathbf{B}_m^T \mathbf{D}^{(k)} \mathbf{B}_m |\mathbf{J}| d\xi d\eta + \right. \\
 & + \frac{1}{2} (z_k^2 - z_{k-1}^2) \int_{-1}^1 \int_{-1}^1 [\mathbf{B}_m^T \mathbf{D}^{(k)} \mathbf{B}_b + \mathbf{B}_b^T \mathbf{D}^{(k)} \mathbf{B}_m] |\mathbf{J}| d\xi d\eta + \\
 & + \frac{1}{3} (z_k^3 - z_{k-1}^3) \int_{-1}^1 \int_{-1}^1 \mathbf{B}_b^T \mathbf{D}^{(k)} \mathbf{B}_b |\mathbf{J}| d\xi d\eta + \\
 & \left. + (z_k - z_{k-1}) \int_{-1}^1 \int_{-1}^1 \mathbf{B}_s^T \mathbf{D}_s^{(k)} \mathbf{B}_s |\mathbf{J}| d\xi d\eta \right\}, \quad (1)
 \end{aligned}$$

где n – количество слоев;
 $[z_{k-1}; z_k]$ – границы k -го слоя по толщине;
 $|\mathbf{J}|$ – Якобиан перехода в местную систему координат;

\mathbf{B}_m , \mathbf{B}_b и \mathbf{B}_s – матрицы связей между перемещениями и деформациями;
 $\mathbf{D}^{(k)}$ и $\mathbf{D}_s^{(k)}$ – матрицы связей между напряжениями и деформациями для k -го слоя;
 m – количество узлов в конечном элементе ($m=3$ – треугольник, $m=4$ – четырехугольник).

Пусть вектор обобщенных перемещений имеет вид

$$\mathbf{d}_e^T = \{ u_1 \ v_1 \ w_1 \ \theta_{x1} \ \theta_{y1} \ \dots \ u_m \ v_m \ w_m \ \theta_{xm} \ \theta_{ym} \}, \quad (2)$$

где u_i и v_i – перемещения i -го узла вдоль оси x или y , соответственно;

w_i – прогиб i -го узла;

θ_{xi} и θ_{yi} – повороты нормали к срединной плоскости пластинки i -го узла относительно оси x

или y , соответственно.

Тогда матрицы связей между перемещениями и деформациями примут вид

$$\mathbf{B}_m = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & 0 & 0 & 0 & \dots & \frac{\partial N_m}{\partial x} & 0 & 0 & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & 0 & 0 & \dots & 0 & \frac{\partial N_m}{\partial y} & 0 & 0 & 0 \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & 0 & 0 & 0 & \dots & \frac{\partial N_m}{\partial y} & \frac{\partial N_m}{\partial x} & 0 & 0 & 0 \end{bmatrix}, (3)$$

$$\mathbf{B}_b = \begin{bmatrix} 0 & 0 & 0 & \frac{\partial N_1}{\partial x} & 0 & \dots & 0 & 0 & 0 & \frac{\partial N_m}{\partial x} & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial N_1}{\partial y} & \dots & 0 & 0 & 0 & 0 & \frac{\partial N_m}{\partial y} \\ 0 & 0 & 0 & \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \dots & 0 & 0 & 0 & \frac{\partial N_m}{\partial y} & \frac{\partial N_m}{\partial x} \end{bmatrix}, (4)$$

$$\mathbf{B}_s = \begin{bmatrix} 0 & 0 & \frac{\partial N_1}{\partial x} & N_1 & 0 & \dots & 0 & 0 & \frac{\partial N_m}{\partial x} & N_m & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial y} & 0 & N_1 & \dots & 0 & 0 & \frac{\partial N_m}{\partial y} & 0 & N_m \end{bmatrix}, (5)$$

где N_i – функция формы i -го узла конечного элемента.

Матрицы связей между напряжениями и деформациями соответствуют плосконапряженному состоянию и принимают вид

$$\mathbf{D}^{(k)} = \frac{E^{(k)}}{1 - (\mu^{(k)})^2} \begin{bmatrix} 1 & \mu^{(k)} & 0 \\ \mu^{(k)} & 1 & 0 \\ 0 & 0 & \frac{1 - \mu^{(k)}}{2} \end{bmatrix}, (6)$$

$$\mathbf{D}_s^{(k)} = \begin{bmatrix} \kappa_1 G^{(k)} & 0 \\ 0 & \kappa_2 G^{(k)} \end{bmatrix}, (7)$$

где $E^{(k)}$, $\mu^{(k)}$ и $G^{(k)}$ – соответственно, модуль Юнга, коэффициент Пуассона и модуль сдвига k -го слоя;

коэффициенты κ_1 и κ_2 являются поправочными коэффициентами для сдвиговых деформаций, которые могут быть получены при помощи специальных методов [18].

Таким образом, исходными данными приведенной выше конечно-элементной модели являются сетка конечных элементов, толщины и ФМХ слоев, а также условия закрепления и нагрузки. Толщины и ФМХ слоев, в общем случае, являются массивами выражений. Формально лингвистические конструкции, необходимые для описания таких задач, используя РБНФ, можно описать в виде рис. 3.

При этом ввод граничных условий можно формализовать в виде, представленном на рис. 4: при определении граничного условия необходимо указать направление действия, условие отбора и начальное значение (в общем случае оно может быть отличным от нуля).

```
пластинка Миндлина = 'MindlinPlate', '(' ,
сетка2д, толщина, модуль Юнга, коэффициент
Пуассона, (граничное условие | нагрузка),
',', (граничное условие | нагрузка), ')'
;
сетка2д = ('Triangles2D' | 'Quads2D'), '(' ,
n, m, origin, ширина, высота, функция
области, ')' ;
n = выражение ; (* количество узлов вдоль оси
абсцисс *)
m = выражение ; (* количество узлов вдоль оси
ординат *)
origin = 'Point2D', '(' выражение, выражение,
')' ;
ширина = выражение ; высота = выражение ;
функция области = декларация функции |
идентификатор ;
толщина = выражение | массив ;
модуль Юнга = выражение | массив ;
коэффициент Пуассона = выражение | массив ;
```

Рис. 3 – РБНФ определения модели расчета

```
граничное условие = 'var', пробел, '=',
'new', 'BoundaryCondition', '(' ,
направление, ',,' условие отбора, ',,' ,
начальное значение, ')', ',' ;
направление = 'ALL' | 'FIRST' | 'SECOND' |
'THIRD' ;
условие отбора = предписание | декларация
функции ;
начальное значение = выражение ;
```

Рис. 4 – РБНФ определения граничного условия

Аналогично можно определить нагрузки (рис. 5): для каждого типа нагрузки необходимо указать направление действия, необязательную функцию-условие отбора узлов и выражение или функцию для определения интенсивности в точке.

В качестве примера рассмотрим задачу исследования НДС круглой трехслойной пластинки радиуса 0,4 м, находящейся под действием равномерно распределенной поперечной нагрузки. Заполнитель: толщина – 0,016 м, модуль Юнга – 72017,3326914 МПа, коэффициент Пуассона – 0,29995185363. Внешние слоя: толщина – 0,001 м, модуль Юнга – 203200 МПа, коэффициент Пуассона – 0,27. Такая задача может быть представлена кодом, приведенным на рис. 6. В первой строке задан радиус пластинки. В строках 2 - 4 заданы, соответственно, толщина, модуль Юнга и коэффициент Пуассона в виде массивов из трех элементов каждый (значения элементов соответствуют характеристикам слоев материала). Строка 5 содержит значение интенсивности нагрузки. В строках 6 - 7 определяется геометрическая модель (детальное описание в работах [15 - 16]). Далее в строках 8 - 13 в соответствии РБНФ (рис. 4) определено условие закрепления (защемление края). В строке 14 указано, что действует равномерно распределенная нагрузка в третьем направлении (в направлении прогиба). В последней строке все

определенные параметры передаются в функцию расчета на базе модели (1) - (5).

```
нагрузка = 'var', пробел, '=', 'new',
('PlateDistributedForce' | 'NodalForce'),
(' ', направление, ', ' [условие отбора,
', '], начальное значение, ')', ';';
условие отбора = предписание | декларация
функции ;
начальное значение = предписание | декларация
функции ;
```

Рис. 5 – РБНФ определения нагрузки

```
1. var r = 0.4; // радиус
2. var h = new Array(0.001, 0.016, 0.001);
3. var E = new Array(203200, 72017.3326914,
203200); // модули Юнга слоев
4. var nu = new Array(0.27, 0.29995185363,
0.27); // коэффициенты Пуассона слоев
5. var q = 0.05; // интенсивность нагрузки
6. function shape(x, y) {
7.     return circle(x, y, r);
8. }
9. var mesh = new Quads2D(151, 151, new
Point2D(-r, -r), 2*r, 2*r, shape); //
генерация дискретной модели
10. var fixed = new BoundaryCondition( ALL,
11. function(x, y){
12.     if (approx(shape(x, y), 0))
13.         return true;
14.     return false;
15. },
16. 0);
17. var force = new
PlateDistributedForce(THIRD, q);
18. MindlinPlate( mesh, h, E, nu, fixed,
force );
```

Рис. 6 – Пример определения модели расчета

В приведенном на рис. 6 коде формализован случай заземления границы. В строках 9 - 10 определена функция-критерий для отбора узлов, на которых определено граничное условие (в основе описания геометрических моделей лежит функциональный подход на базе неявных функций: функция, описывающая форму области равна нулю на границе). В строке 13 указано, что начальное перемещение в узлах, на которые действует граничное условие, равно 0. На рис. 7 последовательно приведены прогибы, поворот θ_x (θ_y эквивалентно с точность до поворота на 90°) и напряжения Мизеса, полученные в результате интерпретации программы. Для решения этой задачи в случае свободного опирания необходимо в определении граничного условия (строка 8) изменить ALL на FIRST|SECOND|THIRD (запретить перемещения в обоих направлениях и прогибы).

Также необходимо отметить, что в соответствии с РБНФ (рис. 5), определение нагрузок в общем случае состоит из двух пользовательских функций: необязательная функция отбора точек, на которые действует нагрузка и функция интенсивности

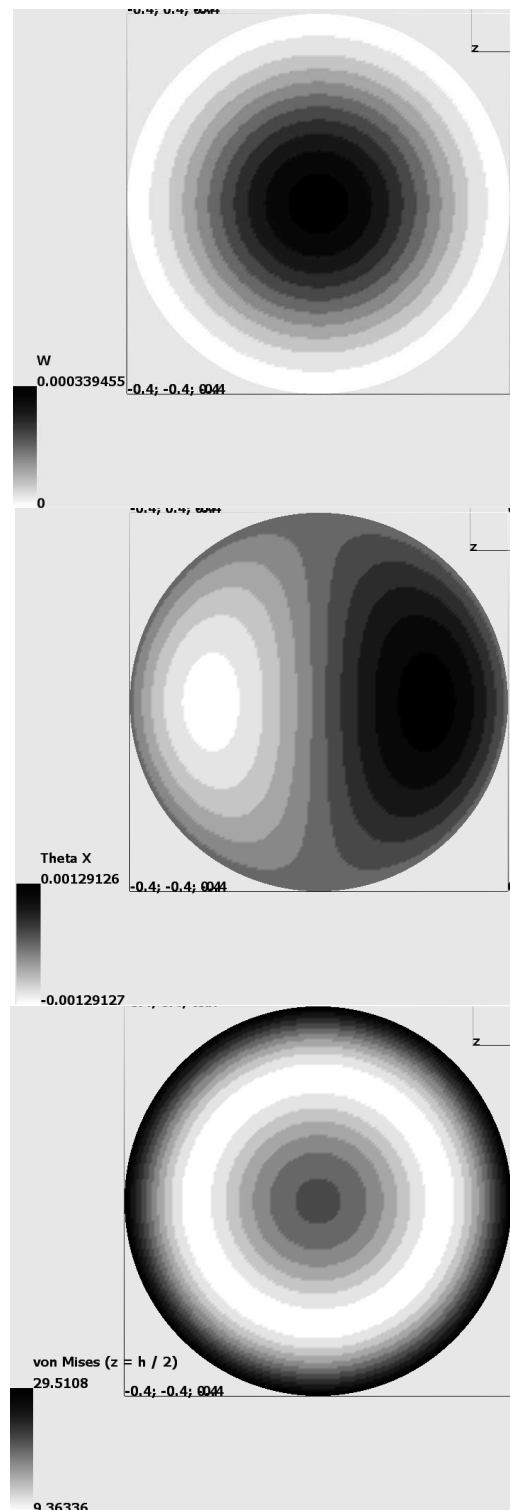


Рис. 7 – Компоненты НДС круглой пластинки

(для решения задач с неравномерной интенсивностью нагрузки). Например, если в рассмотренной задаче равномерная нагрузка должна действовать по кольцу (r_a и r_b – внешний и внутренний радиусы кольца), то, используя логическую конъюнкцию двух круглых областей (операция con), определение такой нагрузки

примет вид, представленный на рис. 8 (нагрузка равна q внутри кольца и 0 – снаружи).

```
var force = new PlateDistributedForce(THIRD,
function(x, y) {
    if (con(circle(x, y, ra), -circle(x, y,
rb)) >= 0)
        return q;
    return 0;
} );
```

Рис. 8 – Пример задания нагрузки, распределенной по кольцу

Аналогично можно исследовать НДС пластинки в форме прямоугольника со скругленными углами. Предположим, что пластинка шарнирно оперта по границе. В этом случае код описания задачи примет вид, представленный на рис. 9. Для описания формы такой пластин использован соответствующий геометрический примитив языка (строка 9). В строках 12-18 определено граничное условие, указывающее, что запрещены перемещения на границе области (а пороты, соответственно, разрешены).

```
1. var a = 1; // ширина
2. var b = 0.7; // высота
3. var r = 0.3; // радиус скругления
4. var h = new Array(0.001, 0.016, 0.001);
5. var E = new Array(203200, 72017.3326914, 203200); // модули Юнга слоев
6. var nu = new Array(0.27, 0.29995185363, 0.27); // коэффициенты Пуассона слоев
7. var q = 0.05; // интенсивность нагрузки
8. function shape(x, y) {
9.     return rectangle(x, y, a, b, r);
10. }
11. var mesh = new Quads2D(151, 151, new Point2D(-a/2, -a/2), a, a, shape);
12. var fixed = new BoundaryCondition(FIRST|SECOND|THIRD, // шарнир
13. function(x, y){
14.     if (approx(shape(x, y), 0))
15.         return true;
16.     return false;
17. },
18. 0);
19. var force = new PlateDistributedForce(FIRST, q);
20. MindlinPlate(mesh, h, E, nu, fixed,
```

Рис. 9 – Пример определения модели расчета

На рис. 10 приведены компоненты НДС, полученные в результате выполнения программы с рис. 9. Сверху вниз последовательно приведены распределения значений прогиба, поворота θ_x и напряжения Мизеса.

Обсуждение результатов

В САПР, разрабатываемой на базе предложенного языка, абсолютно аналогично можно задавать узловые и объемные нагрузки. В основу рассмотренного выше синтаксиса заложен стандарт ЕСМА, который с одной стороны

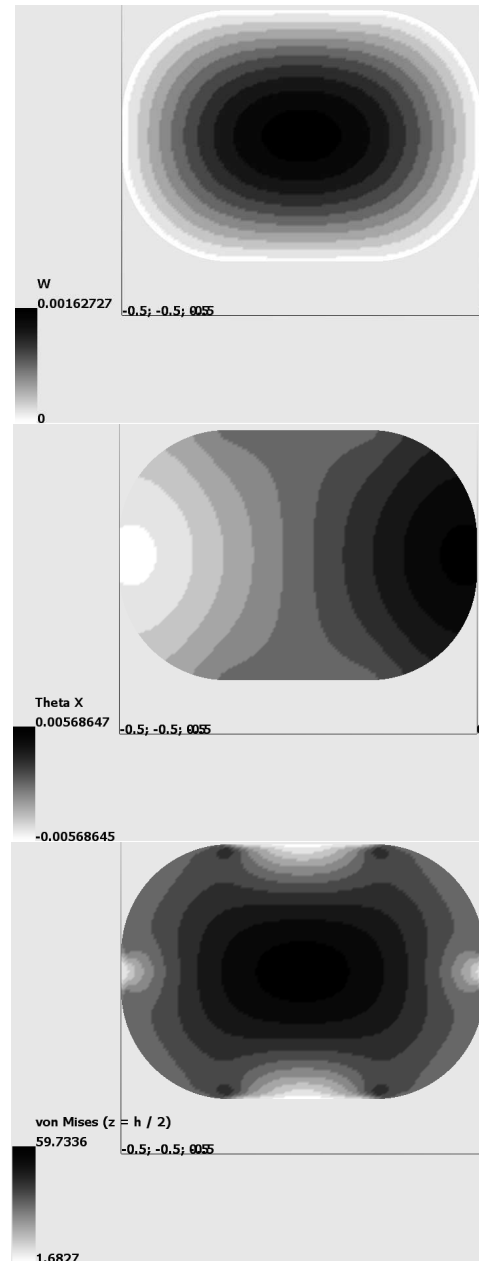


Рис. 10 – Компоненты НДС прямоугольной пластинки

ограничивает возможности в модификации синтаксиса производных языков, с другой стороны – позволяет упростить изучение, т.к. используется в получивших широкое распространение технологиях и языках программирования. Предложенные синтаксические конструкции (в части определения сил и граничных условий) также могут быть использованы при формализации моделей напряженно-деформированного состояния тонких оболочек.

Выводы

Таким образом, предложенный способ формализации моделей тонкостенных конструкций позволяет описывать задачи исследования НДС

пластин и оболочек (добавив незначительные модификации) в САПР. Представленный проблемно-ориентированный язык обладает сравнительно простым в изучении синтаксисом. Научная новизна работы состоит в том, что предложены принципы построения и применения проблемно-ориентированного языка для решения задач исследования многослойных тонкостенных конструкций в САПР.

Список литературы

- 1 **Норенков, И. П.** Основы теории и проектирования САПР / **И. П. Норенков, В. Б. Маничев.** – М.: Высшая школа. – 1990. – 335 с.
- 2 **Nackman, L. R.** AML/X: A Programming Language for Design and Manufacturing / **L. R. Nackman, M. A. Lavin, R. H. Taylor, W. C. Dietrich, D. D. Grossman** // *ACM Fall Joint Computer Conference: National Computer Conference*, Dallas, Texas, November 1986 : proceedings. – Dallas: ACM. – 1986. – P. 145-159.
- 3 **Rossignac, J. R.** Constraints in Constructive Solid Geometry / **J. R. Rossignac** // *Interactive 3D Graphics.* – 1986. – P. 93-110.
- 4 **Paoluzzi, A.** Geometric programming for computer aided design / **A. Paoluzzi.** – Chichester: Wiley. – 2003. – 776 p.
- 5 GRUMMP – Generation and Refinement of Unstructured, Mixed-Element Meshes in Parallel. [*Электронный ресурс*] : – <http://tetra.mech.ubc.ca/GRUMMP>.
- 6 **Croft, H. T.** Unsolved Problems in Geometry / **H. T. Croft, K. J. Falconer, R. K. Guy.** – London: Springer-Verlag. – 1991. – 220 p.
- 7 **Pasko A.** Function-Based Shape Modeling: Mathematical Framework and Specialized Language / **A. Pasko, V. Adzhiev** // *Lecture Notes in Computer Science.* – 2004. – Vol. 2930. – P. 132-160.
- 8 **Japheth, B. R.** Supporting engineering design modeling by domain specific modeling language / **B. R. Japheth, E. E. Ogheneovo** // *American Journal of Software Engineering and Applications.* – 2013. – № 2(3). – P. 86-91.
- 9 **DeVito, Z.** Liszt: A Domain Specific Language for Building Portable / **Z. DeVito, N. Joubert, F. Palacios, S. Oakley, M. Medina, M. Barrientos, E. Elsen, F. Ham, A. Aiken, K. Duraisamy, E. Darve, J. Alonso, P. Hanrahan** // *High Performance Computing, Networking, Storage and Analysis.* – 2011. – P. 1-12.
- 10 **Janssens, B.** Finite Element Assembly Using an Embedded Domain / **B. Janssens, T. Bányai, K. Limam, W. Bosschaerts** // *Scientific Programming.* – 2015. – Vol. 2015. – P. 1-22.
- 11 **Prud'Homme, C.** Advances in Feel++: a domain specific embedded language in C++ for partial differential equations / **C. Prud'Homme, V. Chabannes, V. Doyeux, M. Ismail, A. Samake, G. Pena, C. Daversin, C. Trophime** // *Eccomas'12: European Congress on Computational Methods in Applied Sciences and Engineering.* – Vienna, Austria: proceedings. – Vienna: HAL. – 2012. – 21 p.
- 12 **Гоменюк, С. И.** Вычислительный комплекс для автоматизации проектирования сложных инженерных конструкций и сооружений / **С. И. Гоменюк** // *Вестник Херсонского государственного технического университета.* – 2004. – № 1. – С. 339-342.
- 13 **Гоменюк, С. И.** Инструментальный подход к построению универсальной САПР в машиностроении и строительстве FORTU-FEM / **С. И. Гоменюк,**

- В. В. Мухин, Д. Н. Морозов** // *Вестник Херсонского национального технического университета.* – 2005. – № 2. – С. 107 - 113.
- 14 Standard ECMA-262. ECMAScript Language Specification [*Электронный ресурс*] – 2009. – 252 p. Режим доступа: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-262.pdf>
- 15 **Чопоров, С. В.** Проблемно-ориентированный язык геометрического моделирования на базе теории R-функций / **С. В. Чопоров, С. И. Гоменюк** // *Вісник Національного технічного університету «Харківський політехнічний інститут». Збірник наукових праць. Тематичний випуск: Інформатика і моделювання.* – Харків: НТУ «ХПІ». – 2011. – № 17. – С. 181 - 189.
- 16 **Чопоров, С. В.** Проблемно-орієнтована мова для моделювання геометричних об'єктів при функціональному підході / **С. В. Чопоров** // *Вестник Херсонского национального технического университета.* – 2014. – № 3(50). – С. 532-539.
- 17 **Reddy, J. N.** Mechanics of Laminated Composite Plates and Shells Theory and Analysis / **J. N. Reddy.** – Florida: CRC Press LLC. – 2004. – 854 p.
- 18 **Birman, V.** On the Choice of Shear Correction Factor in Sandwich Structures / **V. Birman, Ch. W. Bert** // *Journal of Sandwich Structures and Materials.* – 2002. – vol. 4. – P. 83-95.
- 19 **Кудин, А. В.** Аналитический и численный анализ изгиба круглой трехслойной пластины под действием локальных нагрузок / **А. В. Кудин, Ю. Н. Тамуров, С. В. Чопоров** // *Вісник Запорізького національного університету: Збірник наукових статей. Фізико-математичні науки.* – 2014. – № 1. – С. 67-81.

Bibliography (transliterated)

- 1 **Norenkov I. P., Manichev, V. B.** Osnovy teorii i proektirovaniya SAPR, Moskov: Vysshaja shkola, 1990, 335 p.
- 2 **Nackman, L. R., Lavin, M. A., Taylor, R. H., Dietrich, W. C., Grossman, D. D.** AML/X: A Programming Language for Design and Manufacturing. *ACM Fall Joint Computer Conference: National Computer Conference*, Dallas, Texas, November 1986 : proceedings. – Dallas: ACM, 1986, 145-159.
- 3 **Rossignac, J. R.** Constraints in Constructive Solid Geometry. *Interactive 3D Graphics*, 1986, 93-110.
- 4 **Paoluzzi, A.** Geometric programming for computer aided design. Chichester: Wiley, 2003, 776 p.
- 5 GRUMMP – Generation and Refinement of Unstructured, Mixed-Element Meshes in Parallel, [Web]: <http://tetra.mech.ubc.ca/GRUMMP>.
- 6 **Croft, H. T., Falconer, K. J., Guy, R. K.** Unsolved Problems in Geometry. London: Springer-Verlag, 1991, 220 p.
- 7 **Pasko, A., Adzhiev, V.** Function-Based Shape Modeling: Mathematical Framework and Specialized Language. *Lecture Notes in Computer Science*, 2004, 2930, 132-160.
- 8 **Japheth, B. R., Ogheneovo, E. E.** Supporting engineering design modeling by domain specific modeling language. *American Journal of Software Engineering and Applications*, 2013, 2(3), 86-91.
- 9 **DeVito, Z., Joubert, N., Palacios, F., Oakley, S., Medina, M., Barrientos, M., Elsen, E., Ham, F., Aiken, A., Duraisamy, K., Darve, E., Alonso, J., Hanrahan, P.** Liszt: A Domain Specific Language for Building Portable. *High Performance Computing, Networking, Storage and Analysis*, 2011, 1-12.

10. **Janssens B., Bányai, T., Limam, K., Bosschaerts, W.** Finite Element Assembly Using an Embedded Domain, *Scientific Programming*, 2015, **2015**, 1-22.
11. **Prud'Homme, C., Chabannes, V., Doyeux, V., Ismail, M., Samake, A., Pena, G., Daversin, C., Trophime, C.** Advances in Feel++: a domain specific embedded language in C++ for partial differential equations. *Eccomas'12 : European Congress on Computational Methods in Applied Sciences and Engineering*, Vienna, Austria: proceedings. – Vienna: HAL, 2012, 21 p.
12. **Gomenjuk, S. I.** Vychislitel'nyj kompleks dlja avtomatizacii proektirovaniya slozhnyh inzhenernyh konstrukcij i sooruzhenij. *Vestn. Herson. gos. tehn. un-ta*, 2004, **1**, 339-342.
13. **Gomenjuk, S. I., Muhin, V. V., Morozov, D. N.** Instrumental'nyj podhod k postroeniju universal'noj SAPR v mashinostroenii i stroitel'stve FORTU-FEM. *Vestn. Herson. nac. tehn. un-ta*, 2005, **2**, 107-113.
14. Standard ECMA-262. ECMAScript Language Specification, 2009, 252 p. [Web] <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-262.pdf>
15. **Choporov, S. V., Gomenjuk, S. I.** Problemno-orientirovan-nyj jazyk geometricheskogo modelirovaniya na baze teorii R-funkcij. *Visnik Nacional'nogo tehničnogo universitetu «Harkivs'kij politehničnij institut». Zbirnik naukovij prac'. Tematičnij vipusk: Informatika i modeljuvannja.* – Kharkiv: NTU «KhPI», 2011, **17**, 181-189.
16. **Choporov, S. V.** Problemno-orientirovana mova dlja modeljuvannja heometryčnykh ob'ektiv pry funktsional'nomu pidkhodi. *Vestnyk Khersonskoho natsional'noho tekhnicheskoho unyversyteta*, 2014, **3(50)**, 532-539.
17. **Reddy, J. N.** Mechanics of Laminated Composite Plates and Shells Theory and Analysis. Florida: CRC Press LLC, 2004, 854 p.
18. **Birman, V., Bert, Ch. W.** On the Choice of Shear Correction Factor in Sandwich Structures. *Journal of Sandwich Structures and Materials*, 2002, **4**, 83-95.
19. **Kudin, A. V., Tamurov, Ju. N., Choporov, S. V.** Analiticheskiy i chislennyj analiz izgiba krugloj trehslojnoj plastiny pod dejstviem lokal'nyh nagruzok. *Visnyk Zaporizkoho natsionalnoho universytetu: Zbirnyk naukovykh statei. Fyzyko-matematyčnij nauky*, 2014, **1**, 67-81.

Сведения об авторах (About authors)

Чопоров Сергей Викторович – кандидат технических наук, Запорожский национальный университет, доцент кафедры математического моделирования; г. Запорожье, Украина; e-mail: s.choporoff@znu.edu.ua.

Sergey Choporove – Candidate of Technical Sciences (Ph. D. in Engineering), Associate Professor, Department of Mathematical Modelling, Zaporizhzhya National University, Zhukovsky str., 66, Zaporizhzhya, Ukraine, 69600, Ukraine; e-mail: s.choporoff@znu.edu.ua.

Гоменюк Сергей Иванович – доктор технических наук, профессор, декан математического факультета, Запорожский национальный университет, Запорожье; e-mail: serega@znu.edu.ua.

Sergey Gomenyuk – Doctor of Technical Sciences (D. Sc. in Engineering), Professor, Dean of mathematical faculty, Zaporizhzhya National University, Zaporizhzhya, Ukraine; e-mail: serega@znu.edu.ua.

Аль-Атамнех Белаль Ганим Мохаммад – аспирант кафедры математического моделирования, Запорожский национальный университет, Запорожье; e-mail: kmm@znu.edu.ua.

Belal Al-Athamneh – postgraduate, Department of Mathematical Modelling, Zaporizhzhya National University, Zaporizhzhya, Ukraine; e-mail: kmm@znu.edu.ua.

Аль-Омари Мохаммад Атеф Вади – аспирант кафедры математического моделирования, Запорожский национальный университет, Запорожье; e-mail: kmm@znu.edu.ua.

Mohammad Al-Omari – postgraduate, Department of Mathematical Modelling, Zaporizhzhya National University, Zaporizhzhya, Ukraine; e-mail: kmm@znu.edu.ua.

Алатамнех Хамзе Халед – аспирант кафедры математического моделирования, Запорожский национальный университет, Запорожье; e-mail: kmm@znu.edu.ua.

Hamze Al-Athamneh – postgraduate, Department of Mathematical Modelling, Zaporizhzhya National University, Zaporizhzhya, Ukraine; e-mail: kmm@znu.edu.ua.

Пожалуйста ссылайтесь на эту статью следующим образом:

Чопоров, С. В. Формализация моделей многослойных тонкостенных конструкций в САПР / **С. В. Чопоров, С. И. Гоменюк, Б. Г. М. Аль-Атамнех, М. А. В. Аль-Омари, Х. Х. Алатамнех** // *Вестник НТУ «ХПИ», Серия: Новые решения в современных технологиях.* – Харьков: НТУ «ХПИ». – 2015. – № 62 (1171). – С. 99 - 106. – ISSN 2079-5459.

Please cite this article as:

Choporov, S., Gomenyuk, S., Al-Athamneh, B., Al-Omari, M., Al-Athamneh, H. Layered thin plates formalization in CAE systems. *Bulletin of NTU "KhPI". Series: New solutions in modern technologies.* – Kharkiv: NTU "KhPI", 2015, **62** (1171), 99 - 106, ISSN 2079-5459.

Будь ласка посилайтесь на цю статтю наступним чином:

Чопоров, С. В. Формалізація моделей багатошарових тонкостінних конструкцій у САПР / **С. В. Чопоров, С. І. Гоменюк, Б. Г. М. Аль-Атамнех, М. А. В. Аль-Омарі, Х. Х. Алатамнех** // *Вісник НТУ «ХПІ», Серія: Нові рішення в сучасних технологіях.* – Харків: НТУ «ХПІ». – 2015. – № 62 (1171). – С. 99 - 106. – ISSN 2079-5459.

АНОТАЦІЯ В роботі розглянута проблема формалізації завдань дослідження напружено-деформованого стану багатошарових тонкостінних конструкцій. Запропоновано відповідну проблемно-орієнтовану мову для опису тонких багатошарових пластин та їх дослідження з використанням методом скінченних елементів у системах автоматизованого проектування. В основі проблемно-орієнтованої мови покладено стандарт ECMA-262, що дозволяє використовувати відносно простий у вивченні синтаксис та спрощує опис складних моделей.

Ключові слова: багатошарові конструкції, проблемно-орієнтована мова, метод скінченних елементів, САПР, ECMA-262

Поступила (received) 15.12.2015