

СРЕДСТВА СОЗДАНИЯ ГЕОМЕТРИЧЕСКИХ ТЕЛ В ПРЕПРОЦЕССОРЕ ЛОГОС-ПРЕПОСТ ПРИ ПОДГОТОВКЕ К РЕШЕНИЮ ЗАДАЧ ПРОЧНОСТИ

*Турутина Светлана Юрьевна (SYuTurutina@vniief.ru), Зверев Сергей Геннадиевич,
Кузнецов Михаил Геннадиевич, Логинов Дмитрий Викторович, Соловьёв Александр Николаевич*

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

Рассматриваются средства создания геометрических тел, реализованные в препроцессоре ЛОГОС-ПРЕПОСТ. Дается описание их реализации. Описывается подход к формированию интерфейса пользователя, а также возможности по созданию тел в пакетном режиме средствами макроязыка. Рассматриваются возможные направления развития средств создания тел. Приводятся примеры использования реализованных операций.

Ключевые слова: пакет программ ЛОГОС, ЛОГОС-ПРЕПОСТ, геометрическая модель, геометрические тела, макроязык, макрокоманды, c3d.

LOGOS PREPOST PREPROCESSOR: TOOLS FOR CREATION OF GEOMETRIC SOLIDS IN PREPARATION TO SOLVING STRENGTH PROBLEMS

*Turutina Svetlana Yurevna (SYuTurutina@vniief.ru), Zverev Sergey Gennadievich,
Kuznetsov Michail Gennadievich, Loginov Dmitry Victorovich, Solovev Alexander Nikolaevich*

FSUE «RFNC-VNIIEF», Sarov Nizhny Novgorod region

The paper describes geometric solids creating tools implemented in LOGOS-PREPOST preprocessor. Details of these tools implementation are given. An approach to forming the user interface is described, as well as means of creating solids in the batch mode by using macro language tools. Possible lines of the development of solids creating tools are considered. Examples of using implemented operations are given.

Key words: LOGOS program package, LOGOS-PREPOST, geometric model, geometric solids, macro language, macro commands, c3d.

Введение

В современной промышленности большую роль играет компьютерное моделирование и инженерный анализ. Их использование позволяет сократить затраты времени и средств на разработку, создание и модернизацию технических устройств, от простейших конструкций до самолетов и атомных электростанций. Одну из ключевых ролей в обеспечении конкурентных преимуществ любого пакета инженерного анализа играют средства работы с геометрическими моделями и технологии 3D-моделирования, поэтому чрезвычайно важно предоставить пользователю широкий набор средств создания геометрических тел.

В настоящее время в ИТМФ ведутся работы по созданию и внедрению отечественного базового программного обеспечения для комплексного имитационного моделирования на супер-ЭВМ с массовым параллелизмом, предназначенного для использования на предприятиях высокотехнологичных отраслей промышленности. Пакет программ инженерного

анализа и суперкомпьютерного моделирования ЛОГОС [1] создан в рамках программы развития суперкомпьютерных технологий. Одним из программных модулей пакета ЛОГОС является ЛОГОС-ПРЕПОСТ, предназначенный для обеспечения решателей средствами работы с геометрическими моделями, подготовки сеточных моделей, задания математической модели и параметров материалов, запуска и мониторинга счета, постобработки результатов расчета.

Данный доклад фокусируется на одной категории средств работы с геометрическими моделями – средствах создания геометрических тел в препроцессоре ЛОГОС-ПРЕПОСТ. В докладе будет дан обзор возможностей по работе с геометрическими моделями в пакете ЛОГОС-ПРЕПОСТ. Перечислены реализованные возможности по созданию сплошных тел. Описан подход к формированию интерфейса пользователя, а также возможности по созданию тел в пакетном режиме средствами макроязыка. Приведены примеры использования реализованных операций.

Ключевой частью пакетов моделирования и инженерного анализа является геометрическое ядро – набор функций, выполнение которых обеспечивает импорт, экспорт и построение трехмерных моделей. В ЛОГОС-ПРЕПОСТ для редактирования геометрических моделей при подготовке инженерных расчетов задач прочности используется геометрическое ядро C3D. Геометрическое ядро C3D выполняет построение геометрической модели, редактирование модели путем изменения ее внутренних данных, построение триангуляции, вычисление массово-центровочных характеристик модели, построение плоских проекций модели, определение столкновений элементов модели [2].

Для описания геометрии используется граничное представление. Описание трехмерного тела с помощью граничного представления опирается на топологические объекты: вершина, ребро, контур, грань.

В ЛОГОС-ПРЕПОСТ для представления геометрии была реализована собственная модель данных, в которой объектами геометрической модели могут являться как геометрические, так и сеточные элементы. Геометрическая модель может разделяться на компоненты, представляющие собой группы элементов, которые определяются пользователем для ясности или логической организации процесса моделирования. Компоненты, в свою очередь, могут объединяться в сборки, которые формируют иерархическую структуру, позволяющую пользователю по желанию скрывать или показывать отдельные части геометрии.

Подсистема геометрического ядра обеспечивает работоспособность в ЛОГОС-ПРЕПОСТ средств обработки геометрических моделей. В части обработки и трансляции геометрии ЛОГОС-ПРЕПОСТ позволяет:

- Импортировать/экспортировать геометрии (форматов .step, .iges, .x_t, .sat, .c3d, последний – формат геометрического ядра c3d, которое лежит в основе САПР Компас-3д);

- Проводить диагностику и анализ качества геометрий (проверка до 72 типов геометрических и топологических проблем);

- Проводить «лечение» геометрий (исправление осуществляется в рамках точности модели, что обеспечивает исходную целостность представления и исключает какие-либо деформации [3]);

- Редактировать геометрию [1].

Всего в ЛОГОС-ПРЕПОСТ реализовано более 70 операций редактирования геометрических моделей для подготовки геометрий к генерации расчетных сеток. Реализованные операции разделяются на следующие категории: создание/удаление геометрических элементов, редактирование тел и граней, разрезание тел, граней и ребер, трансформации, упрощение.

На текущий момент функциональные возможности по созданию тел в ЛОГОС-ПРЕПОСТ включают в себя следующие операции:

- создание элементарных тел:
 - параллелепипеда;
 - цилиндра;
 - сферы;
 - конуса;
 - тора;
 - пирамиды;
- создание тел выдавливанием;
- создание тел вращением;
- создание тел заметания;
- создание тел по сечениям.

Таким образом, в ЛОГОС-ПРЕПОСТ представлен минимальный типовой набор возможностей по созданию тел, входящий в коммерческие пакеты компьютерного моделирования и инженерного анализа, такие как SpaceClaim, HyperMesh, NX CAE и т. д. Это позволяет сделать вывод, что средствами препроцессора ЛОГОС-ПРЕПОСТ можно создавать геометрические модели для задач прочностного анализа.

На рис. 1 приводится архитектура препроцессора ЛОГОС, в рамках которой реализованы процедуры создания геометрических тел.



Рис. 1. Архитектура ЛОГОС-ПРЕПОСТ

В процедуру входят «Диалог», «Операция» (непосредственно сам алгоритм), «Вспомогательные библиотеки» с базовыми операциями и системные библиотеки. Все функциональные возможности создания геометрических тел используют стандартный механизм одношаговых процедур. Выполнение процедур начинается с задания пользователем набора необходимых параметров алгоритма через графический пользовательский интерфейс (ГПИ), затем параметры передаются соответствующей модельной операции создания геометрического тела. Модельной операцией называют операцию, которая модифицирует модель данных.

Рассмотрим реализацию операции создания геометрического тела на примере операции создания тела выдавливанием.

Диалог операции создания тела выдавливанием реализован в классе *CreateExtrusionSolidDialog* (рис. 2). Все классы диалога являются наследниками класса *uppProcedureStepDialog*, описывающего основную функциональность диалогов процедур. Для обмена сообщениями используется стандартный для библиотеки Qt механизм сигнал/слот.



Рис. 2. Структура классов диалога операции

По вызову метода *Execute()*; заданные в диалоге параметры передаются операции *undo::ExtrusionShell*, как показано на рис. 3.



Рис. 3. Структура классов отмены/повтора операции

Operation – базовый класс операций гибридного генератора, поддерживающий *undo/redo*, *OperationWithComponent* – класс операций гибридного генератора, поддерживающий *undo* и связанный с компонентами, *ExtrusionShell* – операция создания тела выдавливания, поддерживающая *undo/redo*.

По вызову метода *Execute()*; заданные параметры передаются реализованной для построения тела выдавливанием моделирующей операции *hybrid::ops::geometry::GeometryExtrusionShell()*. Описанная структура выполнения операции построения тела выдавливанием схематично представлена на рис. 4.

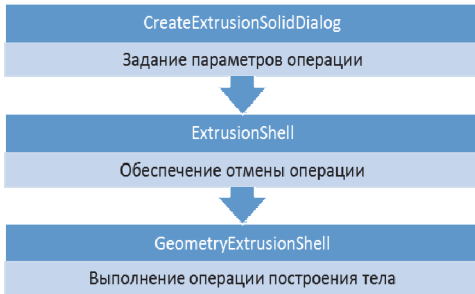


Рис. 4. Структура выполнения операции построения тела выдавливания

Параметры для данной операции: набор ребер или граней, направление, дистанция, тип тела (твердое, оболочка). При варианте построения по граням для каждой грани происходит получение поверхности и задающего ее контура. При варианте построения по ребрам для каждого ребра находится задающая его кривая, по набору данных кривых создается контур. Заполняются параметры операции выдавливания *ExtrusionValues* (при варианте построения по ребрам выставляется отрицательное значение параметра, определяющего замкнутость тела), с помощью метода библиотеки *C3D ExtrusionSolid* строится результирующее тело выдавливания. При варианте

построения по ребрам после проверяется значение заданного пользователем типа тела. Если параметры заданы таким образом, что в результате должно получиться твердое тело, то в торцах строятся две грани-заплатки и сшиваются с гранями, полученными в результате работы метода *ExtrusionSolid*, в единое тело.

Пользовательский интерфейс

Пользовательский интерфейс является одним из основных модулей программного продукта, обуславливающих эффективность и удобство его применения. В процессе разработки пакета ЛОГОС-ПРЕПОСТ была выработана единая концепция оформления пользовательского интерфейса (отступы, цветовая схема, стиль иконок, стандартизированные виджеты).

Как пример ГПИ процедуры создания геометрического тела приведен диалог построения тела выдавливания, реализованный с помощью типовых виджетов задания параметров (рис. 5). Диалог содержит следующие элементы управления: виджет выбора объектов (граней или ребер), по которым проводится построение, виджеты задания направления и расстояния, виджет задания параметров, виджет задания целевого компонента.

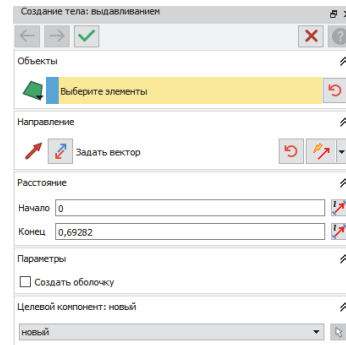


Рис. 5. Диалог операции создания тела выдавливания

В процессе работы процедур создания геометрических тел происходит предварительная визуализация результата построения. Она реализована в методе *UpdatePreview()* класса диалога, соответствующего создаваемому телу.

По рис. 6–8 можно наблюдать, как развивалась функция предварительной визуализации результата построения. Изначально данная возможность отсутствовала, затем было добавлено превью в виде каркаса, затем в виде полупрозрачного тела, также были добавлены инструменты для интерактивного задания параметров.

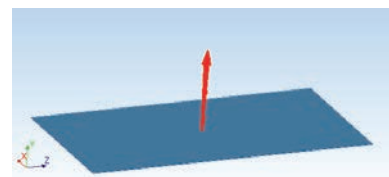


Рис. 6. Построение тела выдавливания без предварительной визуализации результата

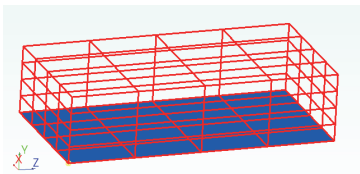


Рис. 7. Построение тела выдавливания с предварительной визуализацией в виде каркаса

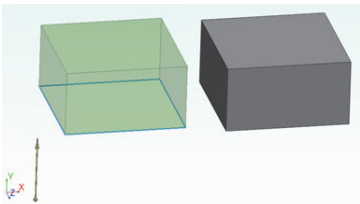


Рис. 8. Построение тела выдавливания с предварительной визуализацией в виде тела

Визуализация в виде каркаса реализована с использованием класса полилинии (линии создаются на каждой грани с определенным шагом: поверхности разбиваются на области, на стыке областей вычисляются точки, по которым рисуется полилиния, все линии собираются в массив для отображения).

Визуализация в виде полупрозрачного тела реализована с использованием библиотеки vtk [5]. В функции превью формируется экземпляр класса vtkActor, с использованием библиотеки c3d строится объемное тело по заданным пользователем параметрам, по нему создается поверхностная сетка MbMesh, по сетке формируются полигональные данные vtkPolyData для визуализации (массив вершин и ячеек сетки), эти данные передаются в vtkPolyDataMapper (содержит ссылку на входные данные и обеспечивает функционирование визуализатора), тот назначается актеру, актер добавляется в визуализатор для отображения данных предварительного просмотра.

На данный момент еще не во всех операциях создания геометрических тел используется предварительная визуализация результата в виде полупрозрачного тела. В будущем для поддержания единообразия ГПИ планируется осуществить полный переход на новую схему предварительной визуализации результата с применением библиотеки vtk.

Примеры использования средств создания геометрических тел

Реализованные функции создания геометрических тел представляют собой базовые функциональные возможности, входящий в программные пакеты для 3d-моделирования. Существует два сценария применения операций создания тел: доработка уже существующей геометрической модели и построение модели с нуля. Функции создания элементарных тел используются для получения объемов простых форм, на основе которых можно моделировать более сложные объекты. Пример применения функции создания

тела заметания в сфере промышленного моделирования – построения части трубы – приведен на рис. 9.

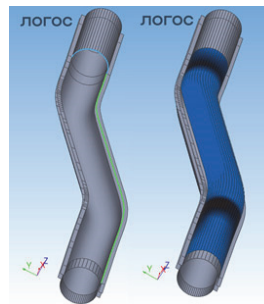


Рис. 9. Создание тела заметания для построения части трубы

Функция построения тел по сечениям может применяться в авиа- или автомобилестроении, например, для моделирования заклепочных или болтовых соединений. На рис. 10 представлен пример применения двух операций – создания тела по сечениям и создания тела вращением – для построения корпуса ракеты.

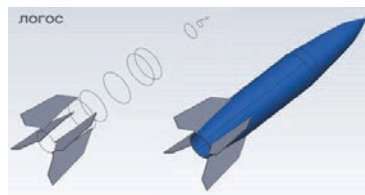


Рис. 10. Построение корпуса ракеты

Макроязык

Одной из ключевых особенностей современных коммерческих САЕ-пакетов является поддержка макроязыка (наборов макрокоманд). Использование макрокоманд позволяет создавать необходимые инженеру сценарии автоматического задания параметров и выполнения цепочек операций без использования их ГПИ.

Место подсистемы макроязыка схематично изображено на рис. 11.



Рис. 11. Многоуровневая схема реализации выполнения операций в ЛОГОС-ПРЕПОСТ

Логика обработки макрокоманд:

- сохранение в журнал операций введенной пользователем макрокоманды на языке Python [4];
- передача введенной команды в процедуру обработки операций для выполнения соответствующей операции;

- при необходимости, последующее воспроизведение всех выполненных действий путем передачи на выполнение полученного скрипта.

Подсистема макроязыка обеспечивает возможность параметризации решаемой задачи, запись взаимодействия пользователя с ГПИ в виде последовательности макрокоманд и возможность последующего воспроизведения этих действий в автоматическом режиме.

Для обработки макрокоманд на языке Python и создания кода вызова операций ЛОГОС-ПРЕПОСТ используется утилита SWIG [6]. На вход утилите SWIG подается интерфейсный файл, содержащий объявления C++ и специальные директивы утилиты SWIG, например, туретар-функции, преобразующие типы данных из Python в C++ и наоборот. Обработка интерфейсного файла, утилита SWIG формирует файл на C++, содержащий специальным образом оформленный связующий код, обеспечивающий командам на языке программирования Python доступ на уровень C++. Полученный файл компилируется вместе с файлами реализации вызова операций.

Достаточно сложно сделать возможной инициализацию сущностей C++ (функций, классов и объектов данных) на уровне Python. Поэтому данный процесс происходит в теле макрокоманд.

Реализации макрокоманд в ЛОГОС-ПРЕПОСТ по области назначения распределены на следующие группы:

- операции редактирования геометрии;
- сеточные операции;
- операции над моделью;
- общие операции.

Внутри групп макрокоманды распределены по типам, например, операции редактирования геометрии, разделены на макрокоманды импорта геометрии, доступа к объектам, создания, разрезания и трансформации тел, а также логических операций над ними.

В качестве примера будет рассмотрена автоматизация алгоритма создания тела экструзией грани с помощью макрокоманд на языке Python. Макрокоманда для операции создания тела экструзией, аналогично диалогу, принимает от пользователя входные данные в качестве аргументов, проверяет их на корректность, производит необходимые преобразования и вызывает соответствующую моделерную операцию. Скрипт создания тела экструзией можно увидеть на рис. 12.

```

1  m = mdb.NewModel( "ModelLogosSA" )           #Создание модели
2  c = CreateComponent( m, "cmpl", 0.5, 0, 0.6 ) #Создание компонента
3
4  pointA = [0,0,0]
5  pointB = [1,0,0]
6  CreateEdge( m, pointA, pointB, c )           #Создание ребра по 2 точкам
7
8  tolerance = 0.1                             #Допуск
9  edgeID = GetEdgeByPoint( m, pointA, tolerance ) #Получение ID ребра по точке
10
11 direction = [0,1,0]                          #Вектор направления
12 isSolid = False                             #Режим построения оболочки
13 distance = 1                                #Расстояние по направлению
14 inversionDistance = 0                       #Несогласие в обратную от направления сторону
15
16 #Протягивание грани из ребра
17 CreateExtrusionSolid( m, [edgeID], [ ], direction, distance, inversionDistance, isSolid, c )
18
19 faceID = GetFaceByPoint( m, pointA, tolerance ) #Получение ID грани по точке
20 direction = [0,0,1]
21 isSolid = True                               #Режим построения твердого тела
22
23 #Протягивание тела из грани
24 CreateExtrusionSolid( m, [ ], [faceID], direction, distance, inversionDistance, isSolid, c )

```

Рис. 12. Скрипт автоматизации алгоритма создания тела экструзией

В отличие от диалога, действия пользователя по интерактивному выбору элементов также заменены макрокомандами и производятся в теле скрипта. Результат выполнения, приведенного выше скрипта, представлен на рис. 13.

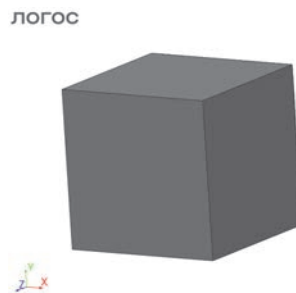


Рис. 13. Результат выполнения скрипта создания тела экструзией

Заключение

В целях обеспечения высокого качества модуля ЛОГОС-ПРЕПОСТ и пакета ЛОГОС в целом коллективом авторов были разработаны средства создания геометрических тел для моделирования задач прочностного анализа. В докладе были описаны возможности данных средств, их реализация, представлен пример интерфейса, приведены примеры использования как через ГПИ, так и с помощью макрокоманд. Реализованные функциональные возможности работают в основной ветке разработки ЛОГОС-ПРЕПОСТ. Уже сформированы основные направления развития средств создания геометрических тел: приведение к единому стилю вида предварительной визуализации результата, расширение функциональных возможностей некоторых операций, добавление недостающих макрокоманд, улучшение интерфейсных возможностей. К концу года планируется включить это в очередной релиз пакета ЛОГОС.

Список литературы

1. ЛОГОС-ПРЕПОСТ. [Электронный ресурс] – Режим доступа: <http://logos.vniief.ru/products/prepost> (дата обращения: 14.05.2021).
2. Геометрическое ядро C3D [Электронный ресурс] – Режим доступа: <https://c3dlabs.com/ru/products/c3d-kernel> (дата обращения: 20.06.2021).
3. Фархутдинов В. Ф., Тарасов В. И., Соловьёв А. Н., Борисенко О. Н., Смолкина Д. Н., Кузьменко М. В. и др. Обзор возможностей по обработке геометрических и построению сеточных моделей в ЛОГОС-ПРЕПОСТ // XIV Международная конференция «Супервычисления и математическое моделирование»: сб. науч. тр. / под ред. Р. М. Шагалиева. Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2013. С. 585–592.
4. Python: About Python. [Электронный ресурс] – Режим доступа: <https://www.python.org/about> (дата обращения: 17.06.2021).
5. The VTK User's Guide 11th edition. Kitware, Inc. 2010. ISBN 978-1-930934-23-8.
6. SWIG: Simplified Wrapper and Interface Generator. [Электронный ресурс] – Режим доступа: <http://swig.org/doc.html> (дата обращения: 14.07.2021).