

НАПРАСНИКОВ В.В., ЖУЙ ВАН ЦЗЫ, КОВАЛЕВА И.Л., НОВИКОВ С.Н.

СЦЕНАРИИ SPACECLAIM В СОЗДАНИИ УЧЕБНЫХ МОДЕЛЕЙ

Белорусский национальный технический университет
г. Минск, Республика Беларусь

Одним из возможных подходов при формировании геометрической части конечно-элементной модели в процессе обучения студентов является использование языка IronPython в среде SpaceClaim с целью последующего моделирования на платформе ANSYS WorkBench. Описываются сравнительные преимущества Design Modeler и SpaceClaim для решения такой задачи. Приведен пример использования этого подхода в реальном учебном процессе.

Ключевые слова: геометрическая модель, SpaceClaim, язык IronPython, обучение студентов, ANSYS WorkBench

Введение

Современный уровень развития общества предполагает ориентацию на сокращение потребляемых ресурсов, экологическую безопасность и энергоэффективность. В то же время особенностью производства в настоящее время является его функционирование в условиях нарастающей конкурентной борьбы, что требует снижения сроков проектирования изделий. Поэтому современный инженер, занимающийся проектированием новых изделий, должен обладать навыками разработки параметрических моделей, которые, с одной стороны, позволяют заменить дорогостоящие натурные испытания виртуальным моделированием, а с другой стороны, обеспечивают возможность сравнения различных вариантов проекта для оптимизационного моделирования [1-9].

Последовательность этапов проектирования описана ниже:

- создание параметрической геометрической модели объекта с использованием современной САД-системы;
- формирование граничных условий, изменяющихся как в пространстве, так и во времени;
- выполнение вариантных расчетов и отладка конечно-элементной модели с использованием современной САЕ-системы;
- проведение оптимизационных вычислений с целью получения рациональных параметров проекта по заданным критериям.

Для того, чтобы студенты и магистранты в рамках обучения получили необходимые компетенции в учебные планы специальностей целесообразно включать дисциплины по конечно-элементному моделированию. При этом для отработки практических навыков следует использовать современные версии промышленных пакетов конечно-элементного моделирования, например, таких как ANSYS [8-9].

Подход, основанный на изучении промышленных пакетов в период обучения в вузах, наилучшим образом обеспечит подготовку выпускника к работе на современном предприятии без длительного периода адаптации и дополнительного обучения. В частности, Белорусский национальный технический университет в настоящее время располагает лицензией на ANSYS, позволяющей реализовать этот подход.

При знакомстве с ANSYS на платформе WorkBench возникает вопрос о том, какую среду следует выбрать для разработки геометрической составляющей конечно-элементной модели. Здесь существует две возможности. Первая – использование встроенного геометрического моделиера Design Modeler, вторая – применение среды SpaceClaim.

Каждое из этих средств геометрического моделирования имеет свои преимущества и недостатки по сравнению с альтернативным.

Так, например, Design Modeler сохраняет информацию о последовательности операций, выполненных при построении модели, в виде дерева, что дает возможность пользователю быстро включать или выключать нужные ветви. Еще одним преимуществом этой среды является простота параметризации модели и отсутствие проблем с передачей параметров в нужный блок решения. Наконец, построение варианта модели при изменении параметров выполняется существенно быстрее, чем в SpaceClaim. Последнее обстоятельство является существенным в случае проведения оптимизационных вычислений, так как существуют объективные временные ограничения, связанные со временем, отведенным на выполнение лабораторной работы.

С другой стороны, SpaceClaim позволяет быстрее освоить основные операции и быстрее пройти базовое обучение, поскольку это система прямого моделирования. Существенным преимуществом SpaceClaim является универсальная

возможность создания программ в виде сценариев на языке IronPython. Также в SpaceClaim реализована возможность включения записи в виде сценария операций, выполняемых инженером. В этом случае соответствующие блоки сценария автоматически снабжаются комментариями, что позволяет обучающимся легче воспринимать особенности применяемых геометрических операций.

При соответствующих настройках среды SpaceClaim комментарии отображаются русскоязычными, что важно для слушателей, не изучавших английский язык. Кроме того, при этих настройках и сам интерфейс программы в целом также становится русскоязычным.

Отметим, что в сети доступна русскоязычный справочный материал, в котором достаточно подробно описываются особенности работы в SpaceClaim [10].

Такая возможность позволяет наглядно представить последовательность действий программиста при непосредственном написании сценария в виде текстового файла. При использовании этого инструмента можно выполнять отладку сценария привычным для программистов образом.

Таким образом, использование SpaceClaim следует рекомендовать для подготовки слушателей, обучающихся на специальностях инженерного профиля, связанных с углубленным изучением программирования.

Пример создания параметрической геометрической модели объекта на основе сценариев на языке IronPython в среде SpaceClaim

Рассмотрим построение учебной модели для вагона-цистерны. На рисунке 1 представлены реальная конструкция (вверху) вагона и его геометрическая модель (внизу), разработанная для выполнения конечно-элементного анализа.

На рисунке 2 приведены фрагменты сценария на языке IronPython с комментариями, запуск которого обеспечивает выполнение нужных операций по построению геометрии объекта.

В предложенном сценарии реализована возможность изменения геометрии объекта в зависимости от его реальных параметров. Таким образом, этот сценарий может быть использован для подготовки модели конструкции вагона любого типоразмера.

После задания граничных условий и нагрузок выполняется конечно-элементное моделирование. На рисунке 3 вверху представлены суммарные перемещения, а внизу напряжения по Мизесу при рабочих нагрузках, полученные для исходного варианта проекта вагон-цистерны.

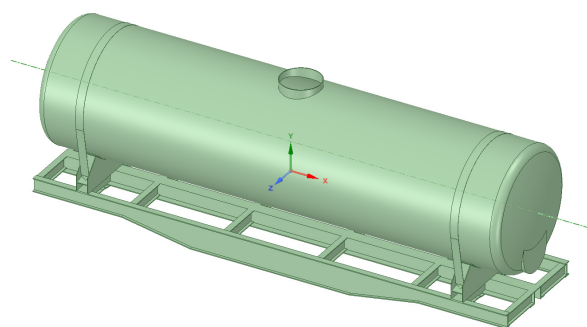
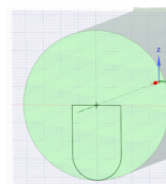


Рисунок 1. Вагон-цистерна и его модель в SpaceClaim

```

112 LU = Point2D.Create(-plugR, 0); #left - up
113 RU = Point2D.Create(plugR, 0); #right - up
114 RD = Point2D.Create(plugR, -D/2 + plugR);
115 LD = Point2D.Create(-plugR, -D/2 + plugR);
116 pmid = Point2D.Create(0, -D/2);
117
118 SketchLine.Create(LD, LU);
119 SketchLine.Create(LU, RU);
120 SketchLine.Create(RU, RD);
121 SketchArc.Create3PointArc(LD, RD, pmid);
    
```



Эскиз для места под сливное отверстие

```

126 selection = Selection.Create(sketch[0])
127 options = ExtrudeFaceOptions()
128 options.ExtrudeType = ExtrudeType.Add
129 result = ExtrudeFaces.Execute(selection,
130     delta/1.5 + CisterneRadius, options, None)
    
```



Результат вытягивания эскиза

Рисунок 2. Фрагменты сценария на языке IronPython и результаты выполнения

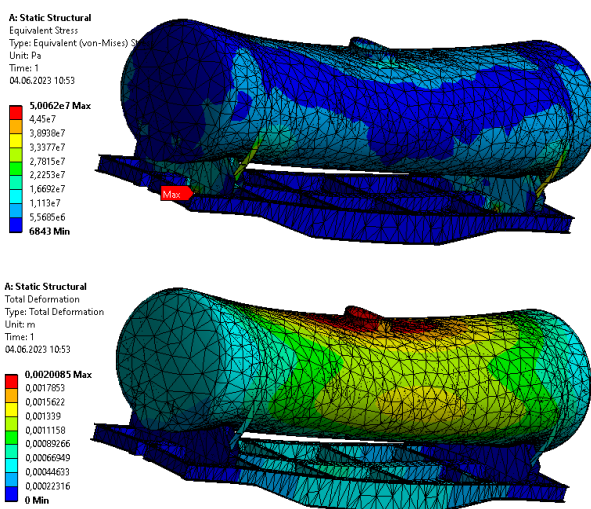


Рисунок 3. Результаты однократного моделирования

На следующем этапе проектирования, как правило, решается задача оптимизации. Однако параметры сценария не могут быть использованы для ее решения непосредственно. Поэтому в среде Space Claim задаются дополнительные параметры для оптимизации.

Рассмотрим этот процесс более подробно. На первом этапе определяются Ruled Dimensions (Управляющие размеры). На панели Edit выбирается инструмент Pull и выделяется поверхность, для которой необходимо задать параметр. После этого, нажав на инструмент Ruler, требуется нажать на кнопку «P», которая находится возле задания размера.

Параметры оптимизации определяются в блоке Response Surface Optimization. Для моде-

ли вагон-цистерны в качестве этих параметров заданы:

– толщина бака цистерны, *cis_th*. Параметр будет изменяться от 5 мм до 20 мм, радиус внешней окружности $R = 500$ мм, радиус внутренней будет варьироваться от 480 мм до 495 мм;

– толщина рамы, *beam_width*. Параметр будет изменяться от 5 мм до 20 мм, высота балки равна 280 мм, диапазон значений от 260 мм до 275 мм.

На рисунке 4 слева представлена поверхность, аппроксимирующая экспериментальные данные по напряжениям в зависимости от двух описанных выше геометрических параметров, выбранных для варьирования, а справа – столбчатая диаграмма чувствительности трех выходных параметров по отношению к варьируемым параметрам.

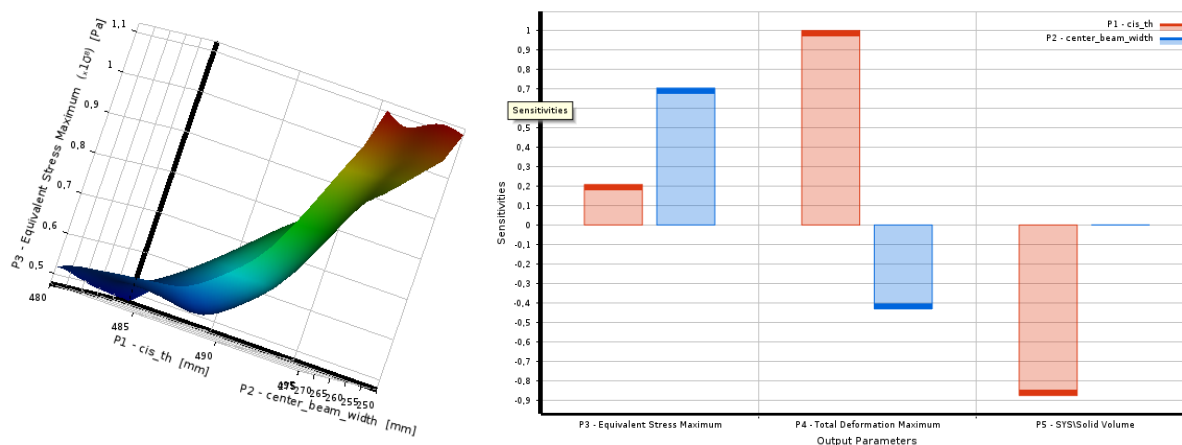


Рисунок 4. Аппроксимирующая поверхность и диаграмма чувствительности

Для оптимизация использовался генетический алгоритм MOGA, встроенный в пакет. Параметры

оптимизационной модели и результаты работы генетического алгоритма приведены на рисунке 5.

	A	B	C	D
1	Optimization Study			
2	Minimize P5	Goal, Minimize P5 (Default importance)		
3	P4 <= 0,005 m	Strict Constraint, P4 values less than or equals to 0,005 m (Default importance)		
4	P3 <= 1,33E+08 Pa	Strict Constraint, P3 values less than or equals to 1,33E+08 Pa (Default importance)		
5	Optimization Method			
6	MOGA	The MOGA method (Multi-Objective Genetic Algorithm) is a variant of the popular NSGA-II (Non-dominated Sorted Genetic Algorithm-II) based on controlled elitism concepts. It supports multiple objectives and constraints and aims at finding the global optimum.		
7	Configuration	Generate 2000 samples initially, 400 samples per iteration and find 3 candidates in a maximum of 20 iterations.		
8	Status	Converged after 4039 evaluations.		
9	Candidate Points			
10		Candidate Point 1	Candidate Point 2	Candidate Point 3
11	P1 - cis_th (mm)	493,51	493,51	493,51
12	P2 - center_beam_width (mm)	274,93	274,75	274,57
13	P3 - Equivalent Stress Maximum (Pa)	★ ★ 9,4056E+07	★ ★ 9,3881E+07	★ ★ 9,3708E+07
14	P4 - Total Deformation Maximum (m)	★ ★ 0,0049995	★ ★ 0,0049995	★ ★ 0,0049995
15	P5 - SYS\Solid Volume (m ³)	★ ★ 1,784	★ ★ 1,7847	★ ★ 1,7853

Рисунок 5. Постановка задачи и результаты оптимизации

Заключение

При подготовке геометрической компоненты конечно-элементных моделей предпочтительно для описания этих компонент использовать встроенные языки пакетов. Это позволяет полностью управлять моделью при вариантных расчетах и оптимизационных вычислениях.

Установлены некоторые преимущества использования геометрического моделиера SpaceClaim и языка IronPython при разработке конечно-элементных моделей:

- удобство использования: благодаря простому и понятному синтаксису Python, разработка скриптов на IronPython является удобной и легкой задачей;
- быстрое выполнение: IronPython работает на платформе .NET, что обеспечивает быстрое выполнение скриптов и обеспечивает высокую производительность;

– возможность параметризации: IronPython позволяет создавать параметризованные скрипты, что упрощает модификацию конечно-элементной модели и повышает ее гибкость. Это особенно полезно при выполнении оптимизации;

– доступность инструментов: IronPython обладает обширной библиотекой инструментов, которые могут быть использованы для работы с конечно-элементными моделями, такими как NumPy, SciPy и Pandas. Эти инструменты позволяют обрабатывать данные, выполнять математические расчеты и генерировать графики.

Описанную методику целесообразно использовать в учебном процессе для студентов и магистрантов, обучающихся на специальностях инженерного профиля, связанных с углубленным изучением программирования.

ЛИТЕРАТУРА

1. **Напрасников В.В., Полозков Ю.В., Кункевич Д.П., Ван Цзы Жуи, Соловьев А.Н.** Подготовка данных о внешних нагрузках в задачах с изменяющимися граничными условиями для деталей с ячеистыми структурами. Математические методы в технологиях и технике. 2022;12(1):22-28. DOI: 10.52348/2712-8873_MMTT_2022_12_22
2. **Ковалева И.Л., Кункевич Д.П., Напрасников В.В., Полозков Ю.В., Чваньков А.А.** Топологическая оптимизация конструктивной геометрии легковесных деталей. «Системный анализ и прикладная информатика». 2022;(3):55-61.
3. **Напрасников В.В., Полозков Ю.В., Бородуля А.В., Кункевич Д.П.** Построение геометрической части конечно-элементной модели одного вида пористых структур. «Системный анализ и прикладная информатика». 2019;(4):55-61. DOI: 10.21122/2309-4923-2019-4-55-61
4. **Красновская С.В., Напрасников В.В.** Исследование возможности идентификации прижимных усилий креплений конструкции с использованием нейронных сетей на основе конечно-элементной модели компрессорно-конденсаторной установки. Журнал "Информатика", 2017;(4):92-99.
5. **Красновская С.В., Напрасников В.В.** Обзор возможностей оптимизационных алгоритмов при моделировании конструкций компрессорно-конденсаторных агрегатов методом конечных элементов. Весці нацыянальнай акадэмп навук беларусі. 2016;(2):92-99.
6. **Напрасников В.В., Красновская С.В.** Влияние упрощающих предположений в конечно-элементных моделях компрессорно-конденсаторных агрегатов на спектр собственных частот. Журнал "Системный анализ и прикладная математика". 2014;(1-3):51-55.
7. **Huei-Huang, L.** Finite elements simulations with ANSYS Workbench 14. Kansas : SDC, 2012. – 602 p.
8. **Free Engineering Simulation Software for Students** [Electronic resource] // ANSYS Inc., 2016. URL: <http://www.ansys.com/products/academic> (accessed: 30.10.2016).
9. **Python 2.7.13 documentation** // PythonSoftwareFoundation., 2016. URL: <https://docs.python.org/2/> (accessed: 03.2023).
10. **SpaceClaim Online Help** [Электронный ресурс] / SpaceClaim Corporation, 2023. – Режим доступа : <https://help.spaceclaim.com/2023.0.0/en/Content/Scripting.htm>. Accessed: 25.03.2023

REFERENCES

1. **Naprasnikov V.V., Polozkov J.V., Kunkevich D.P. Van Czy ZHuj, Solovev A.N.** Preparation of data on external loads in problems with changing boundary conditions for parts with cellular structures. Mathematical methods in technology and engineering. 2022;12(1):22-28. DOI: 10.52348/2712-8873_MMTT_2022_12_22
2. **Kovaleva I.L., Kunkevich D.P., Naprasnikov V.V., Polozkov Yu.B., Chvankov A.A.** Topological optimization of the constructive geometry of lightweight parts. "System analysis and applied informatics." 2022;(3):55-61.
3. **Naprasnikov V.V., Polozkov J.V., Borodulya A.V., Kunkevich D.P.** Building a geometric part finite element model of one kind of porous structures. «System analysis and applied information science». 2019;(4):55-61. (In Russ.) DOI: 10.21122/2309-4923-2019-4-55-61
4. **Krasnovskaya S.V., Naprasnikov V.V.** Investigation of the possibility of identifying the clamping forces of structural fasteners using neural networks based on a finite element model of a compressor-condenser installation. Informatics. 2017;(4):92-99.

5. Krasnovskaya S.V, Naprasnikov V.V. An overview of the possibilities of optimization algorithms for modeling the designs of compressor-condenser units by the finite element method. News of National Academy of Sciences Belarus, series of physical and technical sciences. 2016;(2):92-99.

6. **Naprasnikov V.V., Krasnovskaya S.V.** Simplifying assumptions of condensing unit finite element model and their influence on natural frequency spectrum. «System analysis and applied information science». 2014;(1-3):51-55.

7. **Huei-Huang, L.** Finite elements simulations with ANSYS Workbench 14. Kansas : SDC, 2012, 602 p.

8. **Free Engineering Simulation Software for Students** [Electronic resource] // ANSYS Inc., 2016. URL: <http://www.ansys.com/products/academic> (accessed: 30.10.2016).

9. **Python 2.7.13 documentation** // Python Software Foundation, 2016. URL: <https://docs.python.org/2/> (accessed: 30.10.2016).

10. **SpaceClaim Online Help** [Electronic resource] / SpaceClaim Corporation, 2023. Access mode : <https://help.spaceclaim.com/2023.0.0/en/Content/Scripting.htm>. Accessed: 25.03.2023

NAPRASNIKOV V.V., ZIRUI WANG, KOVALEVA I.L., NOVIKOV S.N.

USING SPACECLAIM SCRIPTS TO CREATE TRAINING MODELS

*Belarusian National Technical University
Minsk, Republic of Belarus*

One of the possible approaches in the formation of the geometric part of the finite-element model in the process of teaching students is the use of the language IronPython in SpaceClaim environment with the purpose of subsequent modeling on the platform ANSYS WorkBench. The comparative advantages of Design Modeler and SpaceClaim for such a problem are described. An example of using this approach in a real learning process is given.

Keywords: *geometric model, SpaceClaim, IronPython language, student learning, ANSYS WorkBench*



Напрасников Владимир Владимирович, кандидат технических наук, доцент кафедры «Программное обеспечение информационных систем и технологий» БНТУ. Научные интересы: конечно-элементное моделирование, компьютерные средства инженерного анализа технических систем.

Naprasnikov Vladimir Vladimirovich, PhD, associate Professor of the Software Department of the Belarusian National Technical University. His research interest focus on finit-element computer aided engineering.

E-mail: vnaprasnikov@bntu.by



Ковалева Ирина Львовна, кандидат технических наук, доцент кафедры «Программное обеспечение информационных систем и технологий» БНТУ. Научные интересы: методы и алгоритмы оптимизации технических систем, машинное обучение, распознавание образов.

Irina L.Kovaleva, PhD, associate Professor of the Department of Software for Information technologies and systems of the Belarusian National Technical University. Her research interests focus on methods and algorithms of optimization and decision-making, pattern recognition and machine learning.

E-mail: ilkovaleva@bntu.by



Ван Цзы Жуй, аспирант кафедры «Программное обеспечение информационных систем и технологий» БНТУ. Научные интересы: конечно-элементное моделирование, компьютерные средства инженерного анализа технических систем.

Wang Zirui, postgraduate student of the department "Software for information systems and technologies" at BNTU. Scientific interests: finite element modeling, computer aids for engineering analysis of technical systems.

E-mail: 610958034@qq.com



Новиков Сергей Николаевич, магистрант кафедры «Программное обеспечение информационных систем и технологий» БНТУ. Научные интересы: программные средства обучения проектированию строительных конструкций.

Novikov Sergey Nikolaevich, graduate student of the department "Software for information systems and technologies" at BNTU. Scientific interests: software tools for teaching the design of building structures.

E-mail: novikov.sergey3214@mail.ru