

Автоматизированное упрощение машиностроительных CAD-моделей и сборок без использования истории построения

С.Е. Сляднев, А.С. Малышев, В.Е. Турлапов

sergey.slyadnev@gmail.com | al.s.malyshhev@gmail.com | vadim.turlapov@itmm.unn.ru

Национальный исследовательский Нижегородский государственный университет им. Н. И. Лобачевского, Нижний Новгород, Россия

Дается обзор методов автоматизированного упрощения CAD-моделей, включая одиночные детали и сборки. Модели предполагаются заданными в открытом формате без истории построения, что делает рассматриваемые методы независимыми от конкретных систем проектирования. Упрощение геометрии сопровождается приведением в непротиворечивое состояние ассоциированных метаданных, таких как имена, структура сборки, допуски, графические атрибуты и проч. Предложенные алгоритмы могут использоваться для эффективного обмена данными между инженерными коллективами, преобразования конструкторских моделей в расчетные, подготовки сцен виртуальной и дополненной реальности, а также в других приложениях, где требуется упрощение существующих геометрических объектов.

Ключевые слова: упрощение CAD-моделей, оптимизация цифровых двойников, геометрическое моделирование, прямое редактирование, Analysis Situs.

Automated history-free simplification of mechanical CAD models and assemblies

S.E. Slyadnev, A.S. Malyshev, V.E. Turlapov

sergey.slyadnev@gmail.com | al.s.malyshhev@gmail.com | vadim.turlapov@itmm.unn.ru

Lobachevsky State University of Nizhni Novgorod, Nizhni Novgorod, Russian Federation

We present a review of methods for automatic and semiautomatic simplification of CAD models and assemblies. The digital representation of a product comes from an open format such as STEP. Therefore, the proposed approaches are independent of specific MCAD systems. CAD geometry is usually accompanied with engineering metadata, such as product names, assembly structure, visualization attributes, dimensions and tolerances, etc. The considered methods are implemented in a way to keep data consistent during the simplification process. The proposed algorithms are used in a commercial application for AV/VR scenes preparation, optimizing models for mechanical simulations, IP protection during collaboration, and in other engineering workflows which require simplification of existing digital mock-ups.

Keywords: simplification of CAD models and assemblies, digital mock-up, geometric modeling, direct editing, Analysis Situs.

1. Введение

Конструкторские модели машиностроительных изделий, как правило, максимально детализированы. Само их назначение состоит в том, чтобы дать исчерпывающее цифровое описание всех узлов и агрегатов проектируемого объекта. В то же время, многие инженерные работы не требуют высокой детализации, а, напротив, оказываются затруднены, либо попросту невозможны на оригинальных конструкторских моделях. Возникает потребность в упрощении геометрии (Рис. 1) при сохранении необходимой для дальнейших операций инженерной семантики: имен деталей, значений допусков, графических атрибутов и проч.

Практика показывает, что инженерные подразделения редко используют единственную систему автоматизированного проектирования, предпочитая инструментальное разнообразие для решения проектных задач. При этом известно, что обмен данными между разнородным программным обеспечением является существенной проблемой, которая в настоящее время далека от решения. Перенос модели из одной системы в другую нередко сопровождается потерей или искажением данных. Более того, широко используемые открытые форматы (STEP, IGES, JT и проч.), предпочтительные с точки зрения долговременного хранения информации, не содержат истории построения модели и ее конструктивных элементов в явном виде. Тем не менее, использование именно открытых форматов позволяет уйти из-под «опеки»

тех или иных компаний-поставщиков коммерческого ПО. Следовательно, программные решения на базе открытых форматов можно считать наиболее общими, а в ряде случаев – единственно приемлемыми.

Мы предполагаем, что исходная модель задана в нейтральном формате STEP (ISO 10303). Отсутствие истории построения и конструктивных элементов (в явном виде) диктует руководящую идею для реализации операторов упрощения, выносимых в настоящий обзор. Этой идеей является так называемое «прямое редактирование», т.е. редактирование без использования истории.

1.1 Цель исследования

Упрощение модели задействует два уровня абстракции: уровень сборки и одиночной детали. Кроме того, в зависимости от цели упрощения, модификация геометрии может осуществляться либо на точном (граничном) представлении детали, либо на ее полигональном приближении. Достаточно богатая палитра инструментов делает возможным их комбинирование в «рецепты» упрощения произвольной сложности.

Мы даем обзор наиболее эффективных и востребованных операторов упрощения CAD-моделей. В качестве геометрического ядра, определяющего формат представления инженерных данных, нами выбрана библиотека OpenCascade [1]. Операторы упрощения, описываемые в настоящей статье, дополняют набор доступных операторов этой открытой библиотеки (Рис. 2).

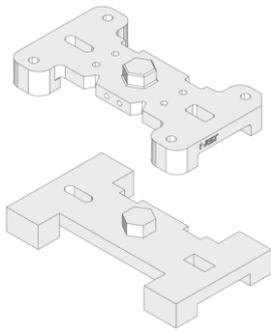


Рис. 1. Упрощение тестовой детали.

1.2 Другие обзоры

Авторы [2] дают подробную компиляцию техник упрощения CAD-моделей для инженерного анализа, причем обсуждаются различные варианты представления геометрической формы. В отличие от [2], мы рассматриваем исключительно граничное представление (B-Rep, boundary representation).

Работа [3] посвящена упрощению сборок, представляющих оборудование для комплектации сложных промышленных объектов, таких как корабли или нефтеплатформы. Ценность этой работы, на наш взгляд, состоит в том, что авторы предъявляют критерии упрощения моделей, полученные «из первых рук» по результатам интервьюирования глав инженерных подразделений. Развитие заявленных идей упрощения содержит статья тех же авторов [4]. Укажем, однако, что работы [3] и [4] основаны на использовании геометрического ядра ACIS [5], которое, хотя и содержит богатую алгоритмическую палитру, является коммерческим продуктом с ограниченным доступом для инженерных и академических сообществ.

Актуальность задачи упрощения сборок возрастает с проникновением в индустрию концепции «цифрового двойника» (Digital Mock-Up, DMU). Мы говорим о механических сборках, получаемых как «временной срез» из PLM-системы предприятия. DMU заменяет физический макет изделия, выполняя роль «справочной» модели, пригодной для выполнения некоторых видов инженерного анализа. В частности, речь идет о проверке сборочных допусков или анализе самой возможности собрать изделие из помещенных в DMU деталей. Подробный обзор, касающийся адаптации цифровых макетов для численного моделирования, содержит диссертация Ф. Буссюж [6]. Поскольку наша работа имеет непосредственное отношение к упрощению больших машиностроительных сборок, мы считаем работу [6] наиболее релевантной нашему исследованию. Дополнительная литература будет разобрана далее в соответствующих специальных разделах.

2. Основные определения

Конструктивный элемент (КЭ, «фичер») – набор граней, имеющих некоторый инженерный смысл. Часто под КЭ понимают грани, образованные одной операцией механообработки на станке с ЧПУ. Наиболее полное описание принципов проектирования в терминах конструктивных элементов можно найти в классической монографии [7].

«Немая» геометрия – граничное представление детали с утерянной историей построения. В немой геометрической модели отсутствует информация о конструктивных элементах («конструкторском замысле»). Как правило, немая модель есть результат преобразования данных из параметрической САПР в нейтральный формат.

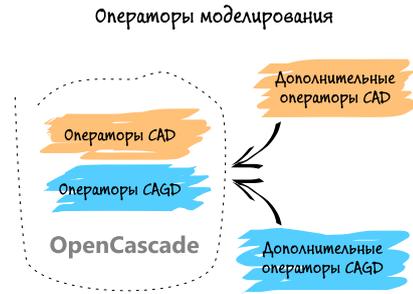


Рис. 2. Операторы упрощения дополняют функциональность выбранного геометрического ядра.

Метаданные -- инженерная семантика, связанная с геометрической моделью. Под метаданными понимают структуру сборки, имена деталей, графические атрибуты, технологические допуски и т.п.

Топологический граф -- ациклический ориентированный граф, представляющий вложенные друг в друга абстрактные топологические элементы, такие как вершины, ребра, грани, оболочки и т.п. Объектная модель геометрического ядра является топологическим графом сама по себе, будь то OpenCascade, ACIS, Parasolid или другая библиотека моделирования, реализующая традиционное разделение структур данных на чисто геометрические и топологические.

Атрибутированный граф смежности – вспомогательный неориентированный граф, вершины которого отвечают граням модели, а дуги представляют отношение смежности. С вершинами и дугами связываются произвольные пользовательские атрибуты. В нашей работе атрибутированный граф смежности является «носителем» конструктивных элементов геометрической модели.

Иерархический граф сборки – ациклический ориентированный граф, вершинами которого являются под сборки и одиночные детали, а дуги представляют отношение «вхождения» с некоторой локальной трансформацией.

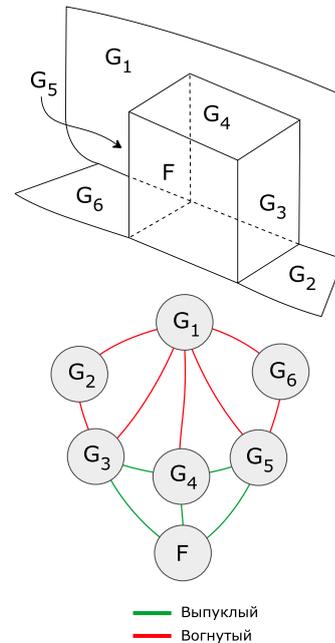


Рис. 3. Грани твердотельной модели и соответствующий атрибутированный граф смежности.

3. Упрощение в контексте детали

Упрощение одиночной детали может заключаться в подавлении ее конструктивных элементов, либо в *идеализации* путем снижения размерности соответствующего топологического многообразия. Операторы идеализации в настоящем обзоре не фигурируют.

Подавление конструктивных элементов мы рассматриваем преимущественно как геометрическую задачу. Однако в общем случае нельзя упускать из внимания влияние операторов упрощения на результаты инженерного анализа. Во избежание потери точности моделирования, некоторые конструктивные элементы подавляться не должны. Работы [8] и [9], на наш взгляд, дают хорошее представление о том, как учесть вклад КЭ в результирующую точность.

Операторы упрощения детали реализуются в парадигме *распознавания конструктивных элементов*, то есть сопровождаются извлечением того или иного типа КЭ из немоной геометрии. Исчерпывающий обзор основных подходов к распознаванию КЭ содержится в работах [10] и [11]. Наш подход состоит в абстрагировании вспомогательных структур данных из граничного представления модели и применении набора *правил* для поиска типовых конструктивных элементов по этим структурам. В классификации [10] такой подход относится к «графовым» с той оговоркой, что в нашей работе правила зафиксированы и не могут задаваться извне. Помимо подходов, основанных на графах, интерес представляют также методы объемной декомпозиции [12], поскольку они позволяют распознавать и подавлять сложные неизолированные КЭ.

Операторы упрощения имеют различную сложность в зависимости от типа конструктивного элемента и топологической реализации его вхождения в деталь (изолированное или с затрагиванием границ). Мы рассматриваем здесь два оператора упрощения: подавление изолированных конструктивных элементов и подавление цепочек скруглений. Математическим языком, адекватным в задачах распознавания КЭ, является язык теории графов. Все методы, представленные в данном разделе, оперируют с *атрибутированным графом смежности* (AAG, Attributed Adjacency Graph) граней [14], представляющим исходную деталь (Рис. 3). Ключевым атрибутом, связанным с дугами графа, является тип соответствующего двугранного угла: «выпуклый», «вогнутый» или «гладкое сопряжение». Мы рассматриваем также дополнительные атрибуты, связанные с вершинами графа. Эти атрибуты содержат признаки принадлежности той или иной грани некоторому КЭ. Таким образом, в результате процедуры распознавания исходный граф оказывается «обогащен» информацией о найденных в модели конструктивных элементах.

3.1 Изолированные КЭ

«Изолированность» означает, что конструктивный элемент реализован только через внутренние циклы «базовых граней» [15]. Удаление базовых граней изолированных КЭ приводит к увеличению числа компонент связности атрибутированного графа смежности.

3.1.1 Распознавание

Рассмотрим атрибутированный граф смежности граней G исходной детали. Требуется получить граф G^* как результат «обогащения» графа G атрибутами, выражающими принадлежность граней некоторым конструктивным элементам. На вход алгоритма распознавания поступает граф G и оператор P . Оператор P

реализует проверку некоторого правила для окрестности вершины f в атрибутированном графе смежности. Некоторые эвристики для распознавания наиболее распространенных типов КЭ могут быть найдены в нашей предыдущей работе [16]. Правило P применяется последовательно ко всем вершинам f , выбранным в качестве кандидатов по некоторому признаку (например, могут быть выбраны все грани, имеющие внутренние циклы). В результате применения правила P происходит «обогащение» графа G новыми атрибутами. Кроме того, те грани, что были «посещены» оператором, исключаются из дальнейшего рассмотрения.

3.1.2 Подавление

Мы рассматриваем только *изолированные* конструктивные элементы поскольку их подавление может быть осуществлено на уровне топологии модели, то есть без применения вычислительно дорогостоящих и ненадежных операторов редактирования геометрии. Подавление изолированных конструктивных элементов осуществляется путем «иссечения» найденных граничных элементов из топологического графа модели (Рис. 4).

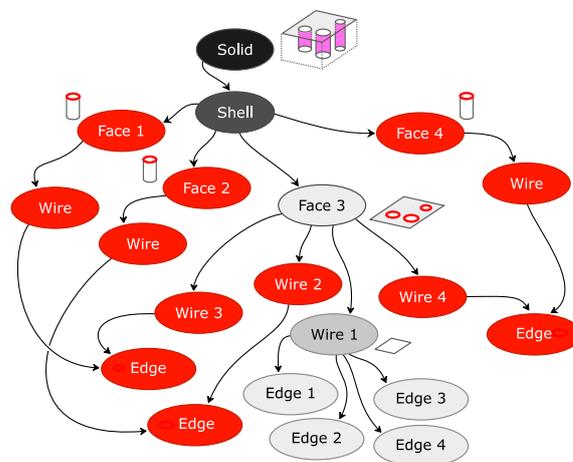


Рис. 4. Подавление изолированных конструктивных элементов путем «иссечения» вершин топологического графа. Красным отмечены вершины графа, удаление которых приводит к подавлению трех сквозных отверстий предполагаемой детали.

3.2 Цепочки скруглений

Подавление скруглений может существенно сказаться на сложности итоговой модели, особенно, если в качестве метрики сложности используется количество фасетов (см. Раздел 5). В нашей работе распознавание и подавление скруглений реализуется согласно принципам, изложенным в докладах [17] и [18]. Важной частью алгоритма подавления являются эйлеровы операторы [19], гарантирующие топологическую целостность итоговой геометрической модели. Применение эйлеровых операторов происходит сообразно результатам локального топологического анализа вблизи граничных элементов скруглений. Привлечение топологического анализа и аппарата эйлеровых операторов позволяет реализовать подавление скруглений как *локальную операцию* упрощения (см. главу 4 монографии [20] о сути «локальных операторов»). Дополнительные ценные сведения об упрощении граничных моделей путем подавления скруглений могут быть найдены в работах [21].

Пример подавления двух цепочек разного радиуса с вершинным скруглением показан на Рис. 5. Ниже приведен листинг соответствующей процедуры для выполнения в среде «Анализ Положения» [24].

```

init-naming

kfmv -name face_8
kev -edge edge_2 edge_18 edge_12
kef -face face_3 -kedge edge_8 -sedge edge_9
kef -face face_10 -kedge edge_22 -sedge edge_16
kef -face face_7 -kedge edge_19 -sedge edge_7

rebuild-edge -name edge_1 edge_4 edge_8
edge_14 edge_16 edge_7
edge_6 edge_11 edge_17
    
```

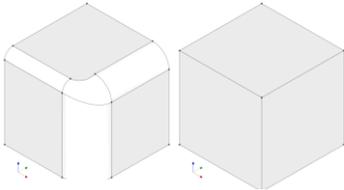


Рис. 5. Подавление цепочки скруглений разных радиусов.

4. Упрощение в контексте сборки

Для представления сборок мы используем реализацию структуры данных типа *иерархический граф сборки* (HAG, Hierarchical Assembly Graph). Описание этой структуры данных было предложено А. Раппопортом в 1993 году [25]. HAG – направленный ациклический граф, вершинами которого являются детали или под сборки, а дуги выражают отношение вхождения (instancing). Другие отношения между компонентами сборки, такие как сопряжения, мы не рассматриваем в силу известной ограниченности нейтральных форматов, используемых для передачи инженерных данных. Для выражения прочих отношений предлагается использовать расширенные графы сборок [26], либо дополнительные структуры данных.

А. Раппопорт отмечает важную специфику HAG-представления сборок, состоящую в неоднозначности привязки метаданных к ее элементам. Действительно, различные варианты вхождения одной детали в сборку представлены не вершинами и дугами, а путями в графе. На Рис. 6 показана тестовая сборка **l-bracket-assembly**, содержащая три вхождения под сборки **nut-bolt-assembly**, состоящей из болта **bolt** и гайки **nut**. Соответствующий иерархический граф сборки показан на Рис. 7.

В отличие от работы [25], где вхождения соответствуют дугам, для представления вхождений мы используем вершины графа (узлы нашей вариации HAG отвечают объектам **next_assembly_usage_occurrence** формата STEP). Из Рис. 7 видно, что привязка метаданных к конкретной вершине или дуге графа не позволяет дифференцировать отдельные вхождения болтов и гаек. Для однозначной адресации вхождений детали или под сборки следует использовать полные пути графа, набранные из уникальных идентификаторов вершин.

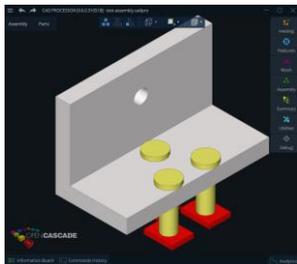


Рис. 6. Тестовая сборка **l-bracket-assembly**, содержащая три вхождения под сборки **nut-bolt-assembly**, представленной болтом **bolt** и гайкой **nut**.

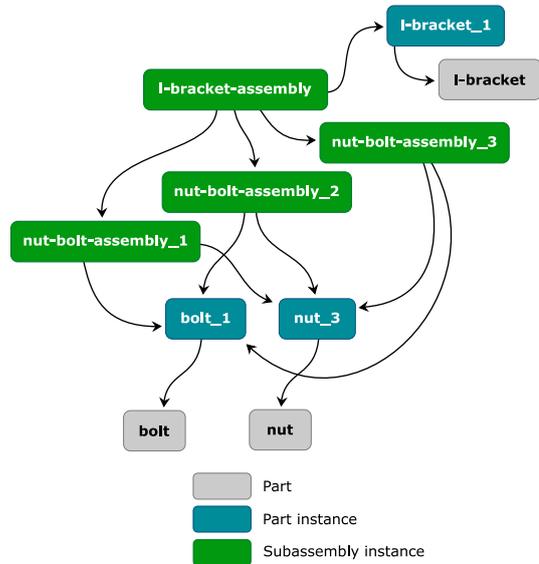


Рис. 7. Граф тестовой сборки **l-bracket-assembly**.

4.1 Удаление невидимых частей

Эффективным средством упрощения сборок является удаление ее невидимых частей. Эта операция востребована в ситуациях, когда интерес представляет только внешняя форма продукта, например, при обмене данными между инженерными подразделениями, при проверке коллизий, а также для упрощения трехмерной модели, погружаемой в контекст более сложного продукта для проектирования сопряженных узлов и агрегатов. Реализация данного оператора основана на испускании лучей с поверхности каждой детали и подсчете количества пересечений с другими компонентами сборки.

4.2 Замена детали примитивом

Проектирование сборки «сверху-вниз» предполагает, что точная геометрическая форма изделия определяется не сразу, но на более поздних этапах работы конструктора [27]. Действительно, первичными являются функциональные требования к продукту, позволяющие начать проектирование еще без детальной геометрической проработки. Оригинальные мысли относительно роли геометрии в машиностроительном проектировании изложены в замечательном эссе [29]. Упрощение сборки является процессом, в некотором смысле обратным проектированию «сверху-вниз». От точного описания формы совершается переход к грубому геометрическому представлению «абстрактного узла» в контексте сборки, которое часто оказывается достаточным для решения прикладных задач.

Простейшим способом перейти от точной формы детали к абстрактной является замена ее исходной геометрической модели примитивом, таким как параллелепипед, цилиндр, шар и т.п. При этом можно сохранить ориентацию детали в пространстве, используя, например, ее главные оси инерции.

4.3 Слияние вхождений

Граф сборки – эффективная структура данных, поскольку идентичная геометрия деталей в ней не дублируется, а реализуется посредством *вхождений* (instances). На практике, однако, встречаются сборки, представленные неоптимальным образом. В худшем случае, иерархический граф может оказаться вырожденным

в дерево, где каждое вхождение детали реплицирует уже существующую геометрию.

Возникает задача, обратная задаче *выделения вхождения* (instance singling), рассмотренной А. Раппопортом [25]. Имея две независимые детали, требуется проверить их идентичность и, в случае таковой идентичности, осуществить *слияние вхождений* с заново рассчитанными матрицами трансформаций. Сравнение моделей осуществляется при помощи простых геометрических эвристик, таких как кардинальное число топологического графа, типы граничных элементов и т.п. Для расчета матрицы трансформации используется сингулярное разложение [30].

5. Переход к сеточным моделям

Мы выделяем два класса поверхностной триангуляции: *фасеты* для визуализации и *сетки* расчетного качества. Фасеты распределены по граням геометрической модели детали, тогда как расчетная сетка теряет эту ассоциативность и помещается в сборке на том же уровне, что и точное представление (Рис. 8).

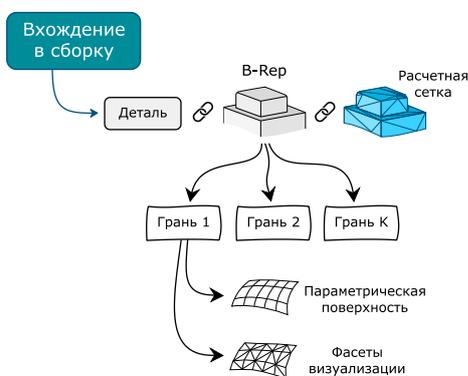


Рис. 8. Принцип организации сеточных данных.

Переход от граничных моделей к сеточным открывает дополнительные возможности для упрощения. Например, на граничных моделях можно осуществить подавление конструктивных элементов, после чего сконструировать расчетную сетку и продолжить упрощение с использованием полигональных операторов, таких как децимация [31] или сглаживание по Лапласу [34].

5.1 Создание расчетной сетки

Конструирование сетки по граничному представлению CAD-модели – стандартная функция геометрического ядра. Теория триангуляции описана в таких монографиях как [35] и [36]. Также существуют открытые программные компоненты, позволяющие создавать расчетные сетки из граничных моделей ядра OpenCascade [37].

5.2 Оборачивание

Оборачивание – специфический способ триангуляции граничного или фасетного представления CAD-модели (сборки). Оборачивание сглаживает дефекты и особенности геометрической модели, требуя лишь, чтобы она описывала некоторый замкнутый объем в пространстве. Этот оператор часто используется для подготовки упрощенных моделей к аэродинамическому анализу. На Рис. 9 представлен результат оборачивания сборки, состоящей из 140 уникальных деталей. Подробно оператор оборачивания описан в работах [39] и [40].

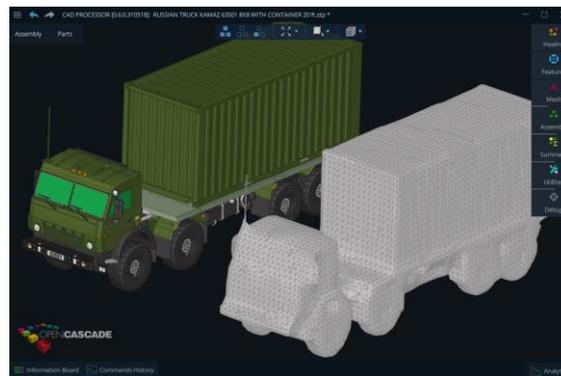


Рис. 9. Результат оборачивания модели «КАМАЗ». Сборка предоставлена Антонино Раписадра (Rapisarda Antonino) в библиотеке GrabCAD (www.grabcad.com).

5.3 Сеточное представление проводов

Некоторые типы деталей не допускают упрощения путем подавления конструктивных элементов или замены примитивом. К такому классу можно отнести кинематические поверхности, моделирующие, например, жгуты с поперечным сечением в виде окружности. Упрощение подобных моделей, как правило, реализуется понижением количества фасетов [41]. С другой стороны, «наивные» методы, состоящие в перестроении сетки с заведомо грубыми параметрами точности, как правило, дают неадекватные результаты (Рис. 10).

В работе [42], авторы решают задачу оптимизации сетки с привлечением таких топологических операторов, как **edge-collapse**, **edge-split** и **edge-swap**. Применение оператора полагается разрешенным, если результат его действия не изменяет топологический тип данной сетки. В частности, не допускается возникновение самопересечений. В отличие от традиционных техник децимации [31], алгоритм Хоппа [42] конструирует сетку, максимально близкую к исходной, имеющую притом существенно меньшую сложность и тот же топологический тип. Метод Хоппа применим для любых геометрических моделей. Однако для кинематических форм, моделирующих электрические провода или гибкие трубки, можно рассмотреть специальные операторы упрощения, дающие больше контроля над результатом.

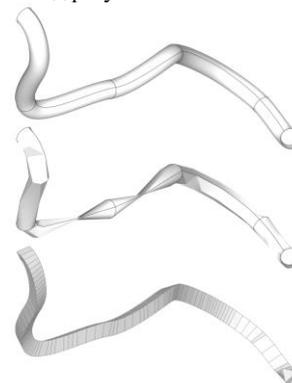


Рис. 10. Варианты геометрического представления провода. Сверху-вниз: оригинальное граничное представление; «грубая» фасетизация с самопересечениями; полиэдральная модель, построенная нашим алгоритмом.

Суть метода состоит в переходе от исходных поверхностей свободной формы к полиэдральному представлению (Рис. 10), фасетная модель которого создается путем разбиения каждой «косой плоскости» по диагонали.

6. Заключение и план исследований

Описанные подходы для упрощения деталей и сборок были реализованы в коммерческом программном обеспечении [43]. В качестве среды для прототипирования большинства алгоритмов применена свободно распространяемая программа «Анализ Положения» [24], показавшая свою эффективность.

Укажем основные направления наших дальнейших исследований в области упрощения геометрических моделей.

6.1 Расширенное распознавание

Планируется расширить модуль распознавания КЭ, перейдя от ограниченного набора правил к формальным грамматикам, как описано в докладе Веткатарамана [15]. Расширенный модуль принимает в качестве исходных данных символьное описание конструктивного элемента, используемое универсальным алгоритмом поиска подграфа в атрибутированном графе смежности. Следующий этап состоит в проверке изолированности КЭ с целью применения оператора топологической редукции, как описано в Разделе 3.1.2.

Во многих сценариях автоматического упрощения требуется распознавание не конструктивных элементов, но специфических типов самих деталей. Так, для применения оператора упрощения проводов (см. Раздел 5.3) требуется предварительно их отыскать среди всех элементов сборки.

6.2 Прямое редактирование

Особый интерес представляет расширение функциональности упрощения механизмами *прямого редактирования* [45] для обеспечения гибкой интерактивной модификации моделей.

6.3 Упрощение сеток

Некоторые геометрические модели изначально лишены точного граничного представления. Для их упрощения требуется разработка дополнительных операторов, таких, например, как оператор оптимизации сеток [42].

6.4 Подготовка данных

Использование операторов упрощения предполагает, что исходная модель лишена геометрических и структурных патологий. Однако опыт показывает, что передача данных между различными инженерными системами нередко сопровождается нарушением целостности модели. Возникает задача разработки специальных операторов для «исправления» модели с целью ее дальнейшего редактирования. Данной проблеме посвящены такие работы, как [46], а также теоретические исследования [50]. Отправной точкой для этих работ является фундаментальный труд А. Рекуича [53].

6.5 Метрики процесса упрощения

Подход, описанный в Разделе 3, может применяться для подготовки САД-моделей к инженерному анализу. В этом случае подавление КЭ должно сопровождаться оценкой ошибки, которую вносит оператор упрощения в результат расчета. Те КЭ, удаление которых приводит к существенному понижению точности моделирования, должны специальным образом помечаться. Трудность состоит в том, чтобы дифференцировать все КЭ, имея в виду их возможные взаимопроникновения. В этом отношении перспективной представляется работа [54], расширяющая «графовый» метод для нахождения неизоллированных фичеров.

7. Благодарности

Авторы признательны руководству компании OPEN CASCADE за предоставленную возможность публикации результатов внутренних исследований.

8. Литература

- [1] S. Slyadnev, Обзор Open CASCADE Technology (2014). URL http://isicad.ru/ru/articles.php?article_num=17367
- [2] A. Thakur, A. G. Banerjee, S. K. Gupta, A survey of CAD model simplification techniques for physics-based simulation applications, *Computer-Aided Design* 41 (2) (2009) 65-80.
- [3] S. Kwon, B. C. Kim, D. Mun, S. Han, Simplification of feature-based 3D CAD assembly data of ship and offshore equipment using quantitative evaluation metrics, *Computer-Aided Design* 59 (2015) 140-154.
- [4] S. Kwon, B. C. Kim, D. Mun, S. Han, User-assisted integrated method for controlling level of detail of large-scale B-Rep assembly models, *International Journal of Computer Integrated Manufacturing* (2018) 1-12.
- [5] J. Corney, T. Lim, 3D modeling with the ACIS Toolkit, Saxe-Coburg Publications, 2001.
- [6] F. Bousuge, Idealization of CAD assemblies for FE structural analyses (2006).
- [7] J. J. Shah, M. Mantyla, Parametric and feature-based CAD/CAM: concepts, techniques, and applications, 1st Edition, John Wiley & Sons, Inc., New York, NY, USA, 1995.
- [8] M. Li, S. Gao, Estimating defeaturing-induced engineering analysis errors for arbitrary 3D features, *Computer-Aided Design* 43 (12) (2011) 1587-1597.
- [9] R. Ferrandes, P. Marin, J.-C. Leon, F. Giannini, A posteriori evaluation of simplification details for finite element model preparation, *Computers & Structures* 87 (1-2) (2009) 73-80.
- [10] J. H. J. Han, M. Pratt, W. Regli, Manufacturing feature recognition from solid models: a status report, *IEEE Transactions on Robotics and Automation* 16 (6) (2000) 1-31.
- [11] J. J. Shah, D. Anderson, Y. S. Kim, S. Joshi, A discourse on geometric feature recognition from CAD models, *Journal of Computing and Information Science in Engineering* 1 (1) (2001) 41.
- [12] С. Сляднев, В. Турлапов, Метод декомпозиции машиностроительных твердотельных моделей на элементы объема изъятия: GraphiCon 2016, Нижний Новгород, 2016, pp. 58-63.
- [13] B. C. Kim, D. Mun, Feature-based simplification of boundary representation models using sequential iterative volume decomposition, *Computers and Graphics (Pergamon)* 38 (1) (2014) 97-107.
- [14] S. Joshi, T. C. Chang, Graph-based heuristics for recognition of machined features from a 3D solid model, *Computer-Aided Design* 20 (2) (1988) 58-66.
- [15] S. Venkataraman, M. Sohoni, V. Kulkarni, A graph-based framework for feature recognition, in: *Proceedings of the sixth ACM symposium on Solid modeling and applications - SMA '01*, ACM Press, New York, New York, SA, 2001, pp. 194-205.
- [16] A. Malyshev, S. Slyadnev, V. Turlapov, Graph-based feature recognition and suppression on solid models, in: *GraphiCon'17 proceedings*, 2017, pp. 319-322.
- [17] S. Venkataraman, M. Sohoni, G. Elber, Blend recognition algorithm and applications, in: *Proceedings of the sixth ACM symposium on Solid modeling and applications - SMA '01*, ACM Press, New York, New York, USA, 2001, pp. 99-108.

- [18] S. Venkataraman, M. Sohoni, R. Rajadhyaksha, Removal of blends from boundary representation models, in: Proceedings of the seventh ACM symposium on Solid modeling and applications - SMA '02, ACM Press, New York, New York, USA, 2002.
- [19] Mantyla, Sulonen, *GWB: A solid modeler with Euler operators*, IEEE Computer Graphics and Applications 2 (7) (1982) 17-31.
- [20] I. Stroud, H. Nagy, *Solid modelling and CAD systems*, Springer London, London, 2011.
- [21] H. Zhu, C. Menq, B-Rep model simplification by automatic fillet/round suppressing for efficient automatic feature recognition, *Computer-Aided Design* 34 (2) (2002) 109-123.
- [22] X. Cui, S. Gao, G. Zhou, An efficient algorithm for recognizing and suppressing blend features, *Computer-Aided Design and Applications* 1 (1-4) (2004) 421-428.
- [23] J.-Y. Lai, C. Wong, T. T. Huynh, M.-H. Wang, C.-H. Hsu, Y.-C. Tsai, C.-Y. Huang, Small blend suppression from B-Rep models in computer-aided engineering analysis, *Journal of the Chinese Institute of Engineers* 39 (6) (2016) 735-745.
- [24] S. Slyadnev, A. Malyshev, V. Turlapov, CAD model inspection utility and prototyping framework based on OpenCascade, in: *GraphiCon'17 proceedings*, 2017, pp. 323-327.
- [25] A. Rappoport, A scheme for single instance representation in hierarchical assembly graphs, in: B. Falcidieno, T. L. Kunii (Eds.), *Modeling in Computer Graphics*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1993, pp. 213-223.
- [26] K. Lee, D. C. Gossard, A hierarchical data structure for representing assemblies: part 1, *Computer-Aided Design* 17 (1) (1985) 15-19.
- [27] M. Mantyla, A modeling system for top-down design of assembled products, *IBM Journal of Research and Development* 34 (5) (1990) 636-659.
- [28] E. C. Libardi, J. R. Dixon, M. K. Simmons, Computer environments for the design of mechanical assemblies: A research review, *Engineering with Computers* 3 (3) (1988) 121-136.
- [29] V. Shapiro, H. Voelcker, On the role of geometry in mechanical design, *Research in Engineering Design* 1 (1) (1989) 69-73.
- [30] O. Sorkine-Hornung, M. Rabinovich, Least-squares rigid motion using SVD.
- [31] W. J. Schroeder, J. A. Zarge, W. E. Lorensen, Decimation of triangle meshes, *SIGGRAPH Comput. Graph.* 26 (2) (1992) 65-70.
- [32] M. Knapp, Mesh decimation using VTK (2002) 1-8.
- [33] L. Kobbelt, S. Campagna, H.-P. Seidel, A general framework for mesh decimation, *Graphics Interface* (1998) 43-50.
- [34] A. Nealen, T. Igarashi, O. Sorkine, M. Alexa, Laplacian mesh optimization, *SIGGRAPH* (2006) 381.
- [35] P. J. Frey, P.-L. George, *Mesh generation: application to finite elements*, ISTE, 2007.
- [36] S. Cheng, T. Dey, J. Shewchuk, *Delaunay mesh generation*, Chapman & Hall/CRC Computer and Information Science Series, CRC Press, 2016.
- [37] J. Schöberl, NETGEN: an advancing front 2D/3D-mesh generator based on abstract rules, *Computing and Visualization in Science* 1 (1) (1997) 41-52.
- [38] C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre-and post-processing facilities, *International Journal for Numerical Methods in Engineering* 79(11) 0 (2009) 1309-1331. arXiv:1010.1724.
- [39] Y. K. Lee, C. K. Lim, H. Ghazialam, H. Vardhan, E. Eklund, Surface mesh generation for dirty geometries by the Cartesian shrink-wrapping technique, *Engineering with Computers* 26 (4) (2010) 377-390.
- [40] A. Malyshev, A. Zhidkov, V. Turlapov, Adaptive mesh generation using shrink-wrapping approach, in: *GraphiCon'18 proceedings*, 2018, preprint.
- [41] Z. Qiu, Y. Wong, J. Fuh, Y. Chen, Z. Zhou, W. Li, Y. Lu, Geometric model simplification for distributed CAD, *Computer-Aided Design* 36 (9) (2004) 809 - 819.
- [42] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, W. Stuetzle, Mesh optimization, in: *Proceedings of the 20th annual conference on Computer graphics and interactive techniques - SIGGRAPH '93*, Vol. d, ACM Press, New York, New York, USA, 1993, pp. 19-26.
- [43] OPEN CASCADE, Digital mock-up simplification. URL <https://www.opencascade.com/content/digital-mockup-simplification>
- [44] OPEN CASCADE, CAD Processor. URL <https://www.opencascade.com/content/cad-processor>
- [45] K.-P. Fahlbusch, T. D. Roser, HP PE/SolidDesigner: dynamic modeling for three-dimensional Computer-Aided Design 46 (1995) 6-13.
- [46] F. Frischmann, Topological and geometric healing on solid models, Master's thesis (2011).
- [47] J. Qi, V. Shapiro, epsilon-Solidity in Geometric Data Translation, Tech. rep., Spatial Automation Laboratory, University of Wisconsin-Madison (2004).
- [48] C. Hoffmann, V. Shapiro, V. Srinivasan, Geometric interoperability via queries, *Computer-Aided Design* 46 (1) (2014) 148-159.
- [49] J. Qi, V. Shapiro, Geometric interoperability with epsilon solidity, *Journal of Computing and Information Science in Engineering* 6 (3) (2006) 213.
- [50] J. Qi, V. Shapiro, Epsilon-regular sets and intervals, in: *International Conference on Shape Modeling and Applications 2005 (SMI' 05)*, Spatial Automation Laboratory, University of Wisconsin-Madison, IEEE Comput. Soc, 2004, pp. 308-317.
- [51] J. Qi, V. Shapiro, epsilon-Topological formulation of tolerant solid modeling, *Computer-Aided Design* 38 (4) (2006) 367-377.
- [52] V. Shapiro, *Reliable implementation of real number algorithms: theory and practice*, Springer-Verlag, Berlin, Heidelberg, 2008, Ch. Homotopy conditions for tolerant geometric queries, pp. 162-180.
- [53] A. G. Requicha, Representations for rigid solids: theory, methods, and systems, *ACM Computing Surveys* 12 (4) (1980) 437-464.
- [54] S. Gao, J. Shah, Automatic recognition of interacting machining features based on minimal condition subgraph, *Computer-Aided Design* 30 (9) (1998) 727-739.

Об авторах

Турлапов Вадим Евгеньевич, д.т.н., профессор кафедры математического обеспечения и суперкомпьютерных технологий ННГУ им.Н.И.Лобачевского, vadim.turlapov@itmm.unn.ru

Мальшев Александр Сергеевич, аспирант 2 года, ИИТММ ННГУ им.Н.И.Лобачевского, email: al.s.malyshev@gmail.com

Сляднев Сергей Евгеньевич, аспирант 3 года, ИИТММ ННГУ им.Н.И.Лобачевского, email: sergey.slyadnev@gmail.com