

Hochschule, 1915. 85 S.

### **Обрезка криволинейных поверхностей в системах автоматизированного проектирования**

к.т.н., доц. Макаров А.И.  
МГТУ «МАМИ»

Математический аппарат, используемый для моделирования криволинейных поверхностей в системах CAD/CAM, не позволяет описывать поверхности со сложной геометрией грани. Трудности возникают уже при построении поверхностей с пятью граничными линиями, не говоря уже о вырезании отверстий внутри поверхности.

Для решения проблемы нерегулярного края и отверстий принят следующий подход: математическое описание поверхностей создается на топологически правильных фигурах (прямоугольниках, квадратах), а созданные на поверхностях ориентированные кривые производят ограничение поверхностей и вырезание в них отверстий.

Плодотворность такого подхода подтверждена тем, что он реализован в известных CAD/CAM системах; связанные с таким подходом структуры предусмотрены в стандарте межсистемного обмена информации VDA FS и IGES.

Остановившись на идее ограничения поверхностей с помощью лежащих на них кривых можно рассмотреть несколько способов построения таких ограничений. Однако безотносительно к способу построения ограничений следует принять два решения. Первое решение состоит в том, что ограничения осуществляются ориентированными кривыми. То есть каждая ограничивающая кривая имеет направление; если двигаться вдоль кривой по этому направлению, то выделяемая часть поверхности, допустим, будет находиться слева, а отсекаемая - справа. Отметим, что направление ограничивающей кривой должно быть связано с направлением нормали поверхности.

Второе решение касается организации ограничений: ограничивающие кривые должны образовать замкнутую фигуру (контур) так, чтобы конечная точка предыдущей кривой совпадала с начальной точкой следующей кривой. Если не принять такое решение, возможна нестыковка ограничивающих кривых, в результате которой в ограничениях могут возникать щели или пересечения. Естественно, что нельзя требовать от пользователя задание кривых, точно состыкованных в конечных точках. Программные средства должны отсекают концевые части пересекающихся кривых или достраивать кривые, имеющие небольшие расхождения.

Под автоматическим построением ограничений понимается процесс, в котором проектировщик создает на поверхности ограничивающие кривые, а программные средства системы проводят анализ на возможность построения из них ограничивающих контуров.

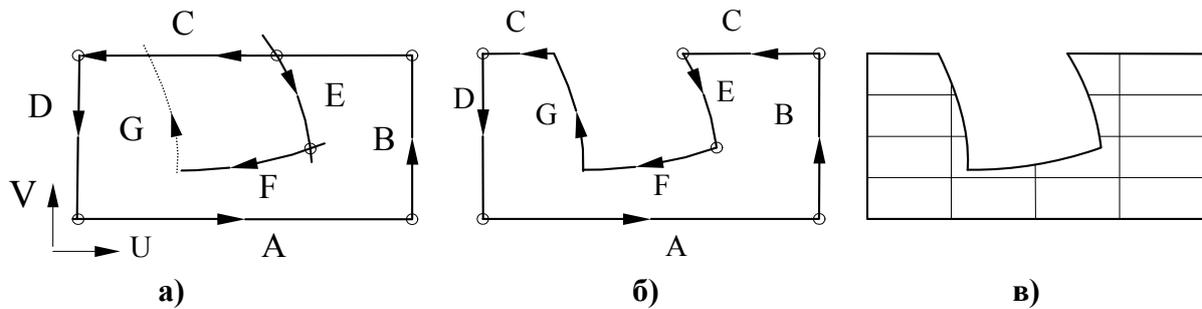
Пусть на поверхности имеются ограничивающие кривые, часть из которых уже вошла в контуры, а другая часть еще не связана в контуры. Кривые, не включенные в контуры, не выполняют функций ограничения, то есть являются пассивными ограничениями.

Пусть на поверхности создана еще одна ограничивающая кривая, такая, что ее взаимодействие с имеющимися кривыми приводит к перестройке топологии ограничений: изменению уже созданных контуров, появлению новых контуров.

Ограничивающие кривые создаются на поверхности в пространстве и преобразуются программными средствами в свои образы на UV-плоскости. Операции по построению ограничивающих контуров осуществляются на UV-плоскости.

На рис. 1а показаны кривые A, B, C, D, из которых связан контур, кривые E и F, не включенные в контур (пассивные ограничения), и вновь созданная кривая G, появление которой приводит к перестройке ограничений. Стрелками показана ориентация кривых. При движении вдоль кривой выделяемая ею поверхность будет находиться слева.

В результате взаимодействия вновь введенной кривой G с другими ограничивающими элементами происходит модификация имеющегося контура: в него включаются новые кривые E, F, G, а кривая C теперь входит в контур двумя кусками (рис. 1б). На рис. 1в линии координатной сетки показывают влияние ограничений.



**Рис. 1. Контур, ограничивающие кривые и перестройка ограничений**

Топологическая структура ограничений в виде контуров и кусков кривых эквивалентна некоторому плоскому ориентированному графу: точки пересечения кривых можно считать вершинами графа, куски кривых - его дугами.

Куски ограничивающих кривых, концевые точки и точки пересечения кривых образуют граф ограничений  $G$ . При этом куски кривых являются дугами, а названные точки - вершинами графа. Граф  $G$  является ориентированным и плоским.

Задача связки контуров интерпретируется в терминологии теории графов следующим образом: на графе ограничений  $G$  определить все циклы (замкнутые пути), являющиеся ограничивающими контурами для поверхности.

Исключив ориентацию дуг в графе  $G$ , получим неориентированный граф  $G$ . Остовное дерево графа  $G$  будет включать все его вершины и содержать между каждой парой вершин в точности один путь. Процесс построения остовного дерева можно начать в любой вершине графа  $G$ , и в процессе прохождения текущей вершины из нее можно выйти по любой из инцидентных ей дуг, кроме той, по которой вошли в вершину [1,2]. В процессе построения остовного дерева определяется список обратных дуг. Включение в остовное дерево каждой обратной дуги ведет к образованию цикла, которые в совокупности образуют фундаментальное множество циклов графа относительно его остовного дерева.

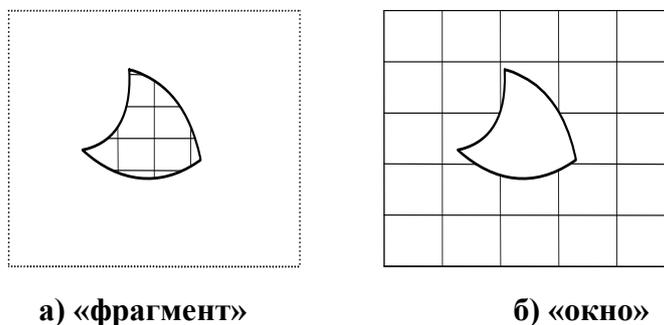
Смысл построения фундаментального множества циклов состоит в том, что на основе фундаментального множества можно получить все циклы графа через теоретико-множественную операцию симметричной разности. В теории графов не просматриваются методы, которые позволили бы разделить полученные циклы на порождающие и не порождающие ограничивающие контуры. Для такой сортировки циклов необходимо привлечение средств вычислительной геометрии.

Метод поиска в глубину осуществляет поиск всех циклов на ориентированном графе ограничений. Отличия процесса поиска в глубину на ориентированном графе от процесса построения остовного дерева заключаются в том, что перемещение по дуге возможно только в определенном направлении. Процесс нацелен на достижение начальной вершины. При этом возможно появление попутных циклов, не содержащих начальную вершину. Применение к вершинам графа правила нулевой полустепени позволяет с помощью простых средств не только избежать повторного прохождения циклов, но и значительно уменьшить объем работы за счет исключения из графа вершин, не порождающих циклы. Правило нулевой полустепени будет гласить: если какая-либо вершина графа ограничений имеет нулевую полустепень захода или нулевую полустепень исхода, то она не принадлежит ни одному циклу. Такая вершина может быть исключена из графа. Если по этому правилу обработать первоначальный граф ограничений, то будут отсечены все пути, не входящие в циклы. Это сокращает поиск в глубину.

Полное множество циклов графа ограничений, полученное одним из двух описанных выше способов, содержит отдельные циклы, которые ограничивают поверхность контурами. Такие циклы нужно отсортировать и зафиксировать как ограничивающие контуры.

Контур может выделять кусок поверхности или вырезать отверстие в ней. Соответственно, контур 1-го типа будем называть «фрагментом», а 2-го типа – «окном». На рис. 2 показаны примеры контуров двух типов. Будем считать, что границы поверхности образуют неявно контур типа фрагмент.

Контурные могут иметь вложенную структуру: «фрагмент»—«окно»—«фрагмент» и т.д.. При правильной структуре контуров на нулевом уровне всегда будет один (возможно, неявный) или несколько контуров типа фрагмент, на первом уровне - контуры типа окно и т.д. То есть четные уровни вершин дерева принадлежат контурам типа «фрагмент», а нечетные - типа «окно». Нарушения правильности построения дерева структур можно использовать для определения циклов, не порождающих контуров.



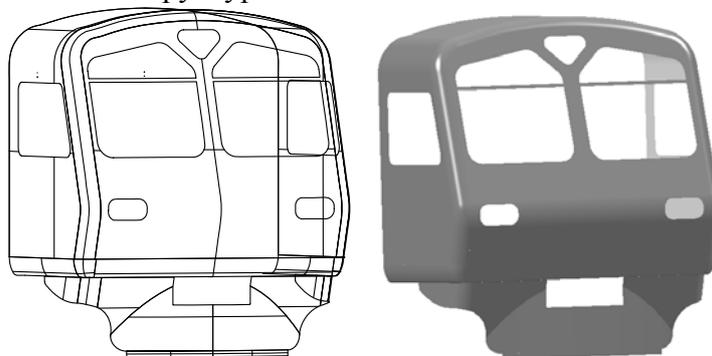
**Рис. 2 Типы контуров**

На нулевом уровне может находиться неявный контур фрагмент только в одном случае - если непосредственно в него вложен один или несколько контуров типа окно. В других случаях на нулевом уровне будут находиться один или несколько контуров типа "фрагмент", созданных на поверхности.

Предположим, что все циклы графа ограничений классифицированы по типам контуров и отображены имеющие общие дуги или связанные вложенностью в дерево их структур. Сначала посмотрим содержание всех уровней дерева на однотипность контуров. Если на каком-либо уровне имеются контуры разных типов, то циклы, их породившие, исключаются из множества.

Далее нужно определить последовательности вершин дерева, на которых не происходит смена типов контура. Если вершины этой последовательности имеют тип "фрагмент", то в качестве ограничивающего контура следует взять цикл, соответствующий последней вершине последовательности, а в случае типа "окно" - начальной вершине последовательности. В результате применения этих двух процедур дерево контуров (или несколько деревьев) должно принять правильную структуру.

Такие же операции по построению правильных деревьев контуров следует выполнить для всех групп циклов, не связанных между собой общими дугами или вложенностью. Если для какого-либо дерева правильная структура не получается, то это значит, что на поверхности нарушена структура контуров. Это нарушение создал пользователь неправильным заданием ограничений. При обработке такой ситуации следует позаботиться о сохранении ранее существовавших правильных структур и выводе сообщения об ошибочных действиях.

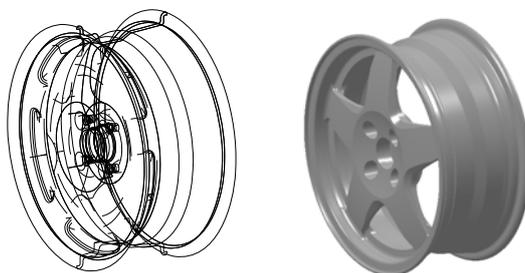


**Рис. 3. Пример обрезки поверхностей при проектировании головной части электропоезда в системе СПОП-3**

Описанные процедуры ограничения поверхностей реализованы в системе проектирования и обработки поверхностей СПОП-3 и показали высокую надежность и быстродействие. Ограничивающие контуры являются необходимым средством при проектировании реальных

объектов со сложными поверхностями. На рис. 3 показана головная часть электропоезда, спроектированная в СПОП-3.

На рис. 4 показана геометрическая модель диска автомобильного колеса. На рис.5 представлен протектор автомобильной шины, элементы которого созданы с помощью контуров на внешней поверхности шины.



**Рис. 4. Геометрическая модель диска колеса**



**Рис.5. Фрагмент протектора шины**

#### **Выводы**

Разработаны алгоритмы автоматического построения ограничивающих контуров с привлечением методов теории графов. Разработан метод преобразования полного множества циклов графа ограничений в ограничивающие контуры. Сформулированы требования к правильности структуры дерева контуров.

#### **Литература**

1. Э. Рейнгольд, Ю. Нивергельт, Н. Део. Комбинаторные алгоритмы. Теория и практика. М.: Мир, 1980. 476 с.
2. Paton K. An Algorithm for finding a Fundamental Set of Cycles of a Graph, Comm. ACM, 12 (1969), 514-518.