

Лабораторная работа №5 «Подготовка модели для сглаживания»

ЦЕЛЬ РАБОТЫ

Изучить принципы подготовки модели для сглаживания и оптимизации её поверхностной сетки. По референсу создать низко-полигональную модель для сглаживания с правильной топологией.

СОДЕРЖАНИЕ РАБОТЫ

1. Правила подготовки объектов для сглаживания. Сохранение формы при сглаживании.
2. Оптимизация топологии.
3. Модификатор TurboSmooth (Сглаживание).
4. Создание низко-полигональной модели для сглаживания.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Горячие клавиши

1,2,3,4,5 – в модификаторе Edit Poly (Редактирование полигонов) – переключение для работы в режиме подъектов: вершин (Vertex)/ рёбер (Edge)/ граней (Border)/ полигонов (Polygon)/ элемента (Element).

F2 – переключение затенения выбранных полигонов (затенять полигон/затенять рёбра полигона)

Alt+1 – Swift Loop (Быстрая петля)

Клавиша «ж» на клавиатуре (раскладка не имеет значения) – повтор последнего действия

Правила подготовки объектов для сглаживания

Сглаживание обычно необходимо, чтобы придать модели более реалистичный вид, так как в реальной жизни у предметов практически не бывает прямых и острых граней. Выполнение этого действия с простыми объектами довольно легко, однако, чем сложнее полигональный объект, тем более тщательно требуется проработать сетку перед этой операцией. С первого взгляда кажется, что весьма просто назначить, например, модификатор TurboSmooth (Сглаживание), чтобы он автоматически выполнил нужные преобразования объекта. Однако так можно сделать только с простейшими полигональными объектами, если форма модели достаточно примитивна и симметрична.

Существует несколько базовых принципов, которых нужно придерживаться для того, чтобы сглаживание объекта было выполнено правильно, и чтобы получить качественный конечный результат.

Чтобы добиться сглаженности граней, но при этом не слишком утяжелять модель, необходимо поработать с полигональной сеткой объекта. Для этого нужно назначить ему модификатор редактирования полигонов (Edit Poly) и далее, используя инструменты полигонального моделирования, и там, где необходимо, применить описанные ниже принципы.

Алгоритм создания круглых элементов на прямоугольных полигонах представлен на рисунках ниже. Любыми инструментами полигонального моделирования нужно разбить полигон на 4 части, чтобы получился результат как на рисунке 1,б (т.е. на каждом ребре будет по одной дополнительной вершине). Затем необходимо сделать преобразование, показанное на рисунке 1,в.

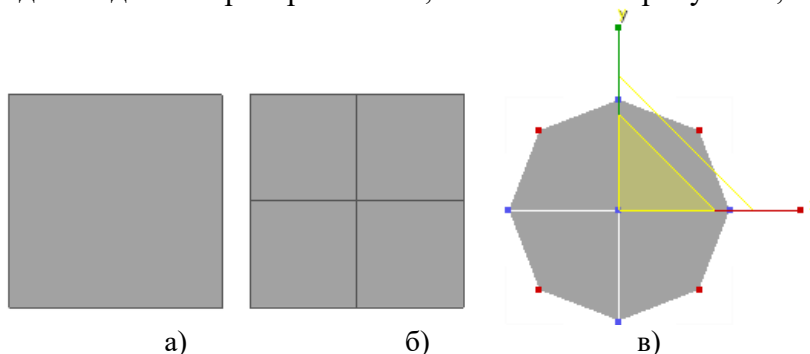


Рисунок 1 – Преобразование полигона для сглаживания: а) исходная плоскость размерами 1х1 сегментов; б) преобразованная плоскость размерами 2х2 сегмента; в) сужение крайних точек инструментом масштабирования (Select and Uniform Scale)

В результате получен объект, похожий на окружность. Однако при назначении этому объекту сглаживания окружность не получится. Получится результат, отраженный на рисунке 3.

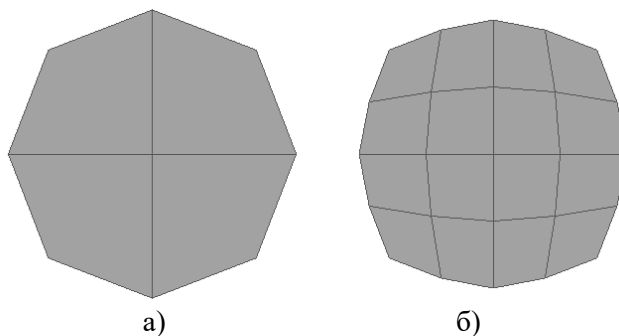


Рисунок 3 – Частично подготовленный к сглаживанию объект: а) исходная плоскость 2х2 сегмента с применением сужения крайних точек; б) результат сглаживания такого частично подготовленного объекта

Чтобы получить окружность, необходимо создать защитный контур рядом с краем по принципу, показанному на рисунке 4. На этом же рисунке показан результат сглаживания объекта с таким контуром. Чем этот контур ближе к краю исходного полигона, тем меньше будет влиять скругление при сглаживании. Таким образом, получена окружность из прямоугольного полигона.

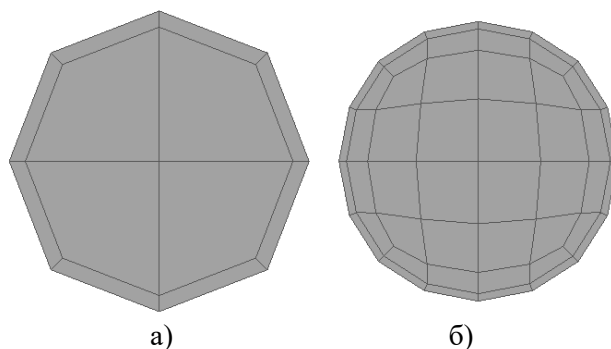


Рисунок 4 – Принцип подготовки объекта к сглаживанию: а) исходная плоскость 2х2 сегмента с защитным контуром по краю; б) качественный результат сглаживания – получена окружность из квадрата

На рисунке 5 приведены примеры, как правильно преобразовать различные плоские полигоны перед сглаживанием. На рисунке 6 показан этап создания защитного контура для каждого из преобразованных полигонов и результат правильного сглаживания.

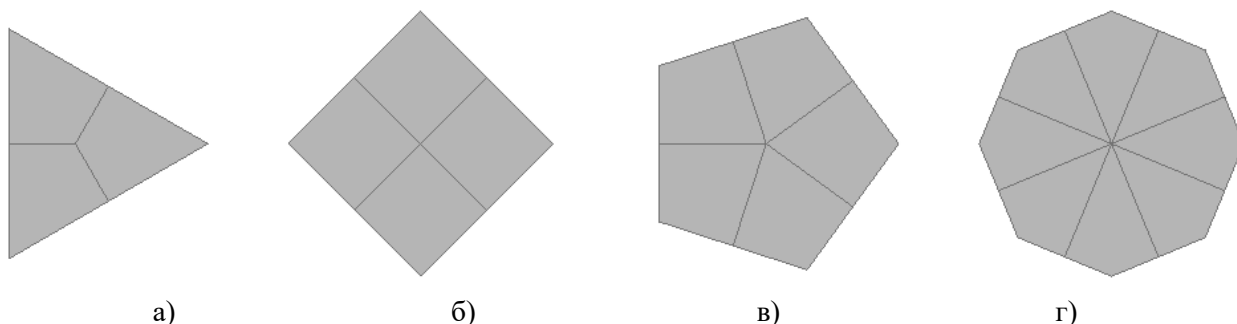


Рисунок 5 – Правильная подготовка различных полигонов к сглаживанию: а) треугольный полигон; б) правильный четырехугольный полигон; в) полигон с пятью вершинами; г) восьмиугольный полигон

Плоские объекты при сглаживании делятся на полигоны, число которых равно количеству вершин исходного полигона. При этом сглаживание по краю не происходит. Для сглаживания по краю необходимо создать защитный контур полигонов.

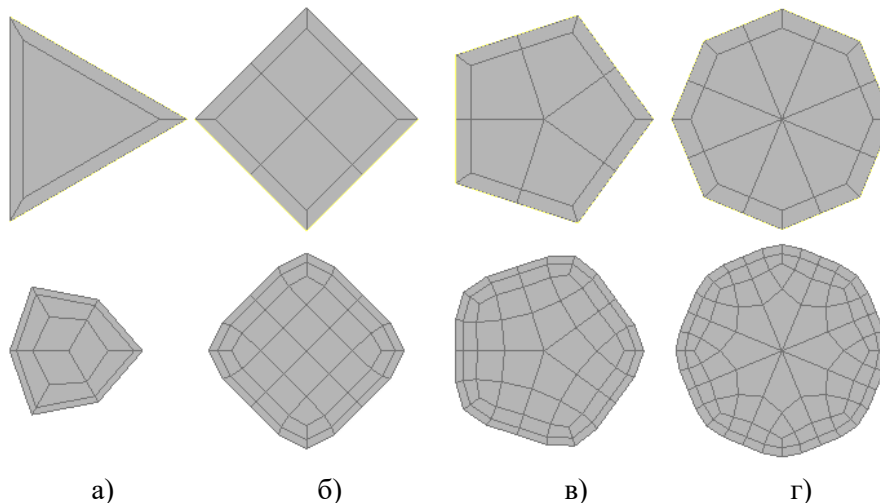


Рисунок 6 – Правильная подготовка различных плоских полигонов к сглаживанию: а) треугольный полигон; б) правильный четырехугольный полигон; в) полигон с пятью вершинами; г) восьмиугольный полигон

При наличии защитной линии процесс сглаживания искажает объект, и создается сильно искаженный рисунок топологии (для всех полигонов, кроме четырехугольника, обозначенного на рисунке буквой б).

Следовательно, необходимо использовать четырехугольники, чтобы топология была правильной. Для того, чтобы получить правильную форму конечного объекта, одной топологии мало, нужно также использовать приемы сглаживания, которые помогут предотвратить нежелательные изменения объекта и будут способствовать желательным.

Сохранение формы при сглаживании. Если необходимо сохранить форму объекта при сглаживании, то в защитном контуре в вершине излома необходимо создать четырехугольник для каждой плоскости, где необходимо сохранить форму при сглаживании. Примеры приведены на рисунках 7 и 8.

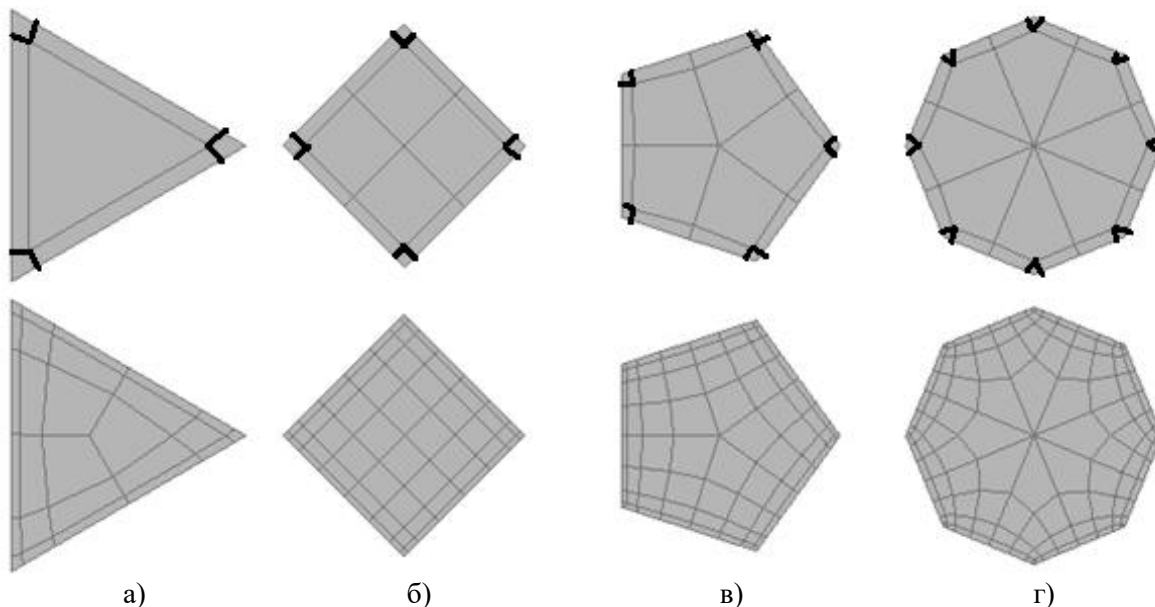


Рисунок 7 – Правильная подготовка различных плоских форм с защитным контуром к сглаживанию: а) треугольный полигон; б) правильный четырехугольный полигон; в) полигон с пятью вершинами; г) восьмиугольный полигон

Для правильного сглаживания объемных угловатых моделей необходимо по такому же принципу в углах защитного контура сделать четырехугольники (рисунок 8).

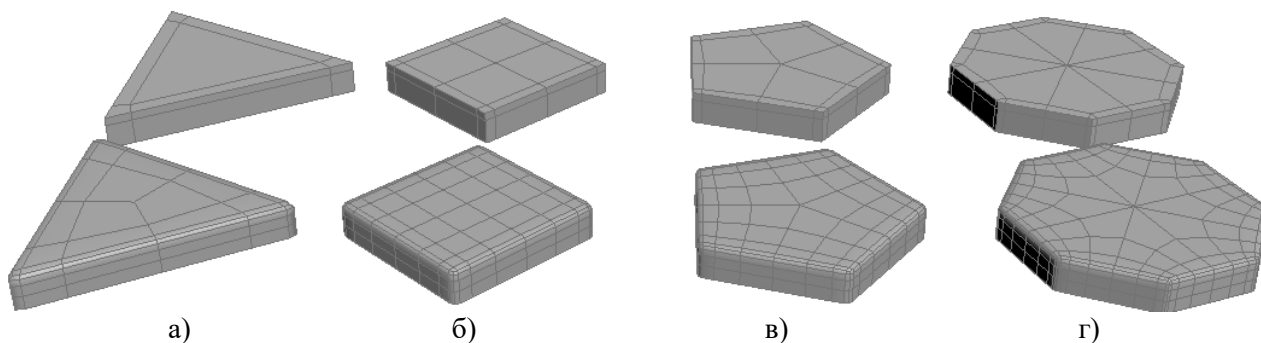


Рисунок 8 – Правильная подготовка объемных угловатых моделей к сглаживанию: а) треугольная форма; б) правильная четырехугольная форма; в) пятиугольная форма; г) восьмиугольная форма

Для объемных объектов, форму которых нужно сохранить, защитные контуры необходимо делать во всех плоскостях, где понадобится это сохранение формы. Если защитных контуров не сделать, сглаживание будет работать по принципу, показанному на рисунке 9.

Для внутренних углов алгоритм работает аналогично.

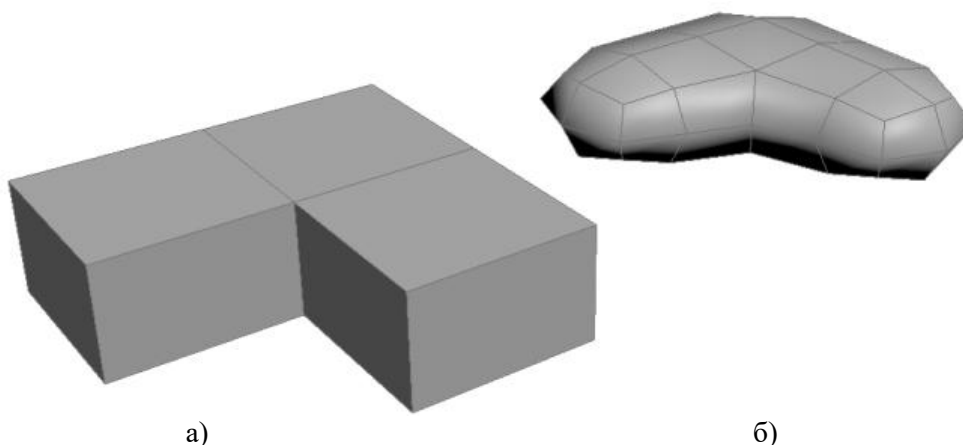


Рисунок 9 – Угловой объемный объект с применением к нему сглаживания: а) исходная фигура с острыми гранями; б) результат сглаживания без подготовки модели к сглаживанию

На примере на рисунке 9 видно, что если применять правильную топологию, но не делать защитных контуров, то объект теряет свои очертания. Необходимо применять прием сглаживания с использованием четырехугольников в местах перелома, пример приведен на рисунках 10 и 11.

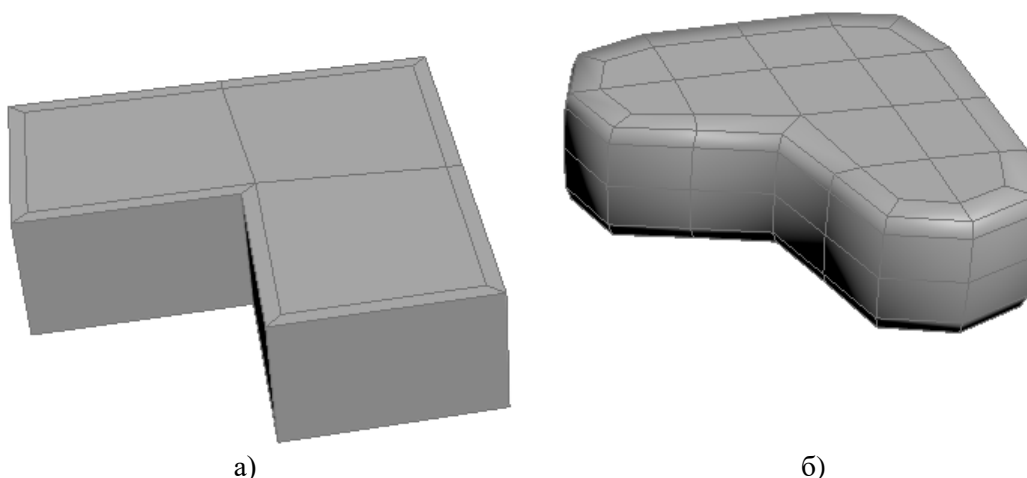


Рисунок 10 – Угловой объемный объект с применением к нему сглаживания: а) исходная фигура с защитными контурами только в одной плоскости; б) результат сглаживания объекта с использованием защитного контура, но без четырехугольников в местах излома

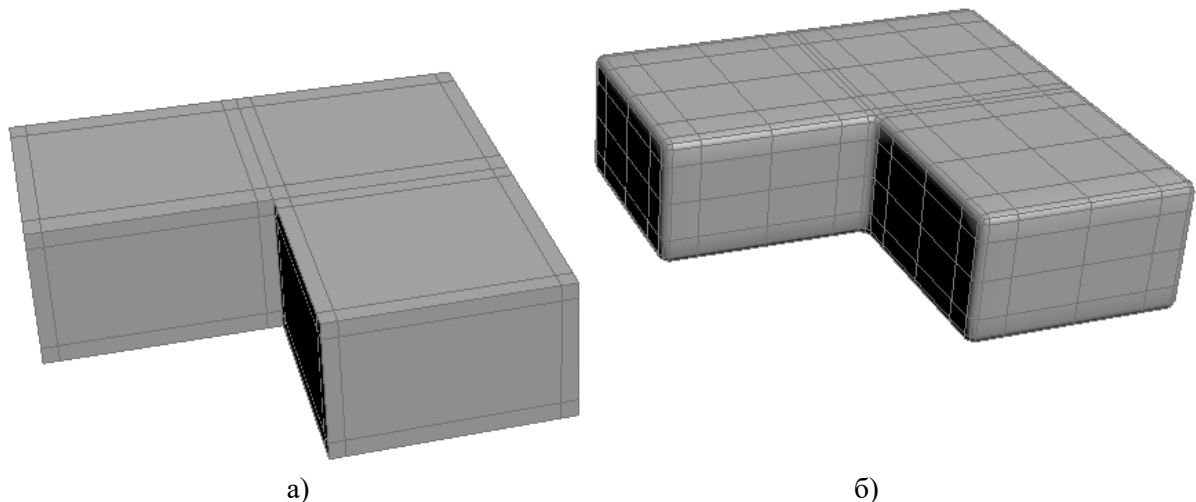


Рисунок 11 – Угловой объемный объект с применением к нему сглаживания: а) исходная фигура с защитными контурами во всех плоскостях (четырёхугольники на углах, которые необходимо сохранить); б) результат сглаживания углового объекта с сохранением формы объекта

Если объект имеет сложную геометрическую форму, часто после создания защитных контуров (чтобы эта форма не искажалась при сглаживании) существенно возрастает число полигонов поверхностной сетки. Потому бывает необходимо выполнить её оптимизацию.

Оптимизация топологии

Оптимизация заключается в том, чтобы при наименьшем количестве полигонов получить необходимый конечный результат. На рисунке 11 приведён пример, где количество полигонов у несглаженной модели 126, при сглаживании оно увеличилось до 1008. Изменив топологию внутреннего угла, можно получить 98 полигонов, а при сглаживании – 784 полигона (пример на рисунке 12).

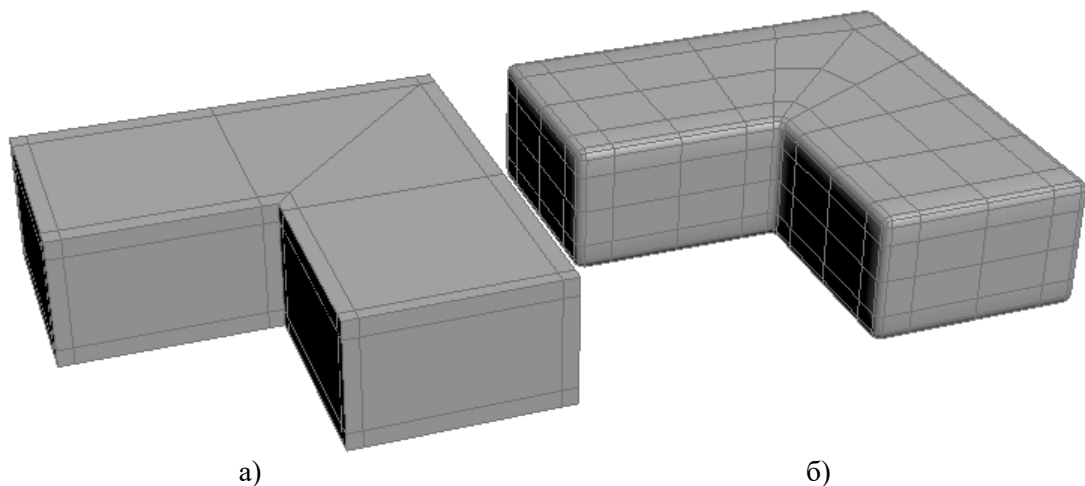


Рисунок 12 – Угловой объемный объект с применением к нему сглаживания: а) исходная фигура с защитными контурами и с оптимизацией внутреннего угла; б) сглаживание объекта с использованием защитных контуров и оптимизацией внутреннего угла

На примере ниже (рисунок 13) показано, как происходит сглаживание объекта с правильной топологией в зависимости от того, как сформирована его поверхностная сетка (т.е. на каких участках происходит скругление, а на каких – сохранение формы).

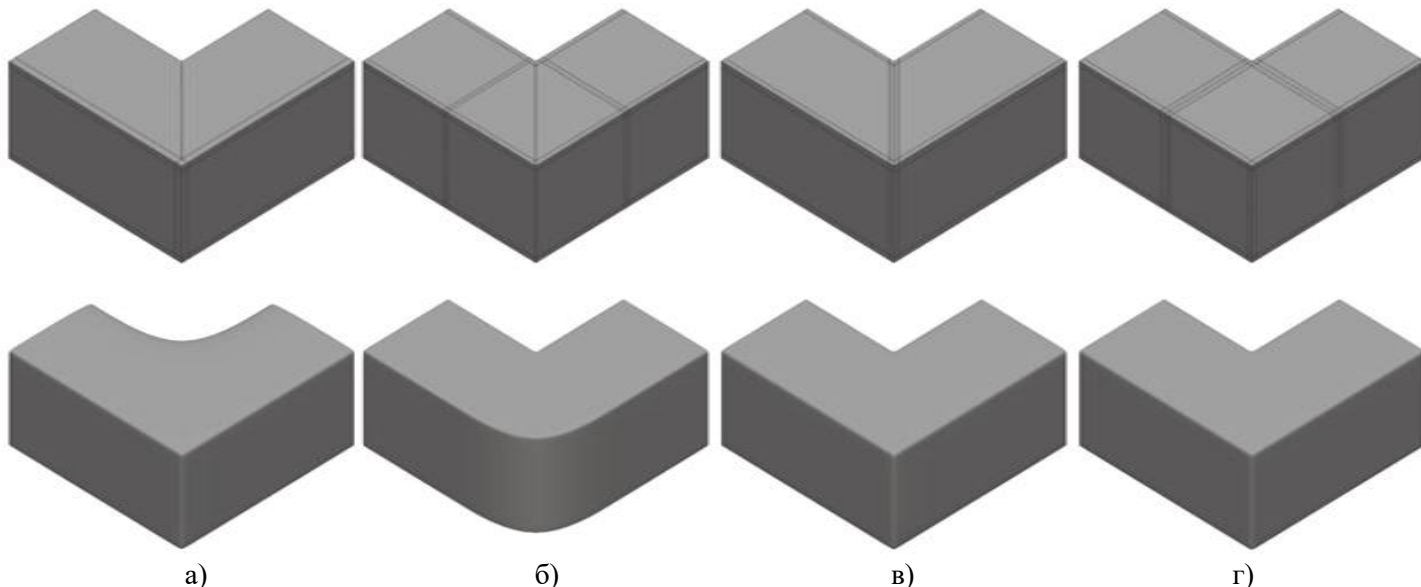


Рисунок 13 – Поверхностная сетка и результат сглаживания с сохранением формы детали: а) скругление только внутреннего угла; б) скругление только внешнего угла; в) сохранение формы внутреннего и внешнего углов с оптимизацией; г) сохранение формы внутреннего и внешнего углов без оптимизации

Применение данных правил позволит получать качественный конечный результат сглаживания при сохранении правильной топологии объекта, а оптимизация позволит уменьшить число полигонов конечных объектов, что очень важно для больших сцен.

Модификатор TurboSmooth (Сглаживание)

Модификатор TurboSmooth (Сглаживание) производит сглаживание геометрии объекта, т.е. закругление углов и краев. Используя параметры модификатора, можно управлять размером и количеством новых граней, а также их влиянием на поверхность объекта. Список параметров указан на рисунке 14.

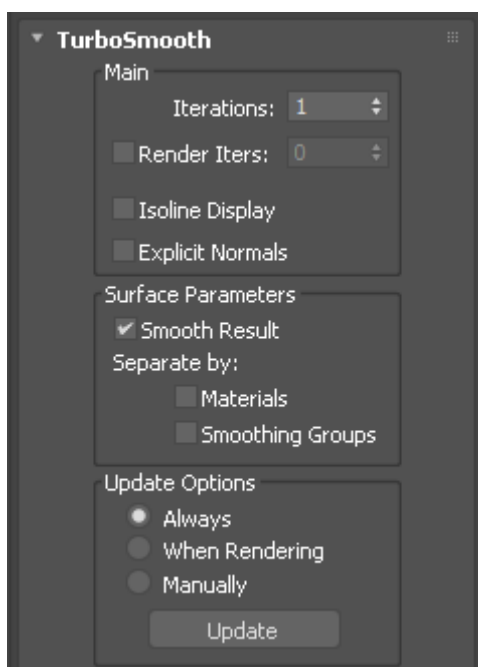


Рисунок 14 – Список параметров модификатора TurboSmooth (Сглаживание)

Iterations (Итерации) – какое количество раз разделяется сетка. При значении 1 данного счётчика каждый полигон будет разделен одним ребром. При значении 2 каждый полигон исходного объекта будет разделен одним ребром, затем каждый вновь созданный полигон будет разделен одним ребром.

Т.е. каждая новая итерация подразделяет сетку, создавая гладко интерполированные вершины для каждой вершины, ребра и грани из предыдущей итерации.

Совет. Не следует задавать этому счётчику большие значения! Ресурса компьютера или ноутбука может не хватить на обработку данной операции, и программа зависнет. Желательно не использовать числовое значение больше 5, а иногда и больше 3.

Render Iterations (Итерации при визуализации) позволяет выбрать другое количество итераций сглаживания, которое будет применено к объекту во время визуализации. Для установки количества итераций визуализации используется поле справа.

Пример применения модификатора *TurboSmooth* к кубическому объекту приведён на рисунке 15. При числе итераций (*Iterations*) = 3 можно из куба получить идеальной формы шар с правильной топологией (т.е. все полигоны будут четырехугольными).

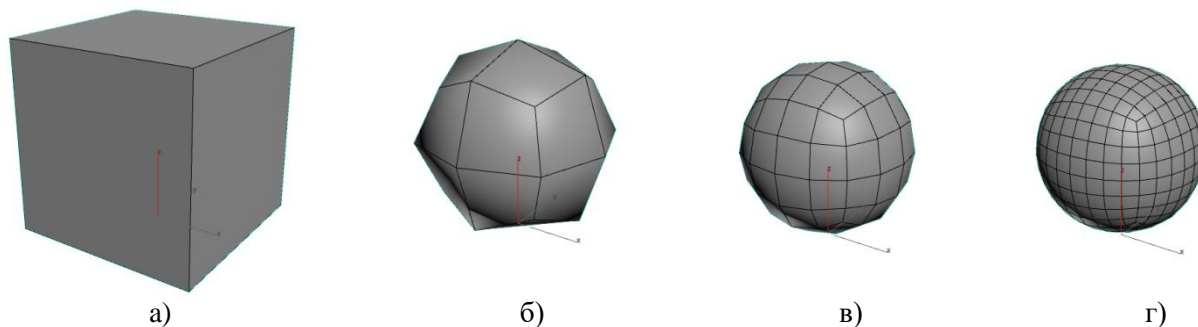


Рисунок 15 – Применение модификатора TurboSmooth с разными параметрами для куба размерами 100x100x100мм: а) исходный объект; б) Iterations = 1; в) Iterations = 2; г) Iterations = 3

Isoline Display (Отображение изолиний). При включении отображаются только изолинии: исходные края объекта до сглаживания. Преимущество использования этой опции – менее загроможденный дисплей.

Explicit Normals (Точные нормали). Включение этого параметра позволяет модификатору вычислять нормали для его выходных данных.

Surface Parameters (Параметры поверхности). Настройки в этом поле позволяют применять группы сглаживания к объекту и ограничивать эффект сглаживания поверхности.

Smooth Result (Гладкий результат). Применение одной и той же группы сглаживания ко всем граням.

Separate by Smoothing Groups (Сглаживание по группам). Предотвращает создание новых граней на ребрах между гранями, которые не имеют хотя бы одной группы сглаживания.

Пример применения модификатора *TurboSmooth* с расширенными настройками к кубическому объекту приведён на рисунке 16.

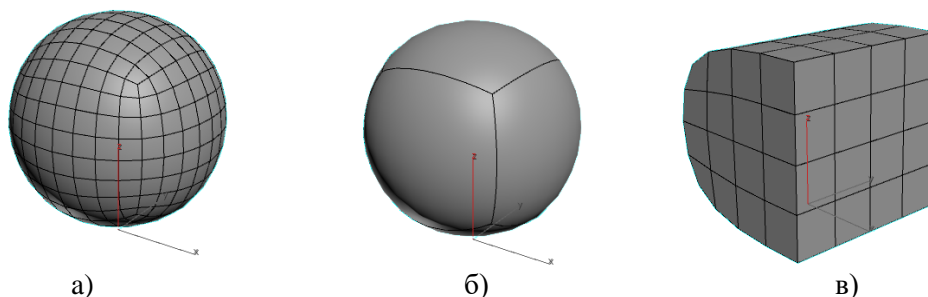


Рисунок 16 – Применение параметров модификатора TurboSmooth (Сглаживание) для куба размерами 100x100x100мм: а) исходный объект, итерации (*Iterations*) = 3; б) итерации (*Iterations*) = 3, отображение изолиний (*Isoline Display*) = on; в) итерации (*Iterations*) = 2, сглаживание по группам (*Separate by Smoothing Groups*) = on

Рассмотренные принципы позволяют создавать сложные и качественные модели с наименьшими трудозатратами, сохраняя при этом правильную топологию модели.

Создание низко-полигональной модели для сглаживания

Сглаживание обычно необходимо, чтобы придать модели более реалистичный вид, так как в реальной жизни у предметов практически не бывает прямых и острых граней. Выполнение этого действия с простыми объектами довольно легко, однако чем сложнее форма и сетка полигонального объекта, тем более тщательно требуется проработать эту сетку перед операцией сглаживания. С первого взгляда кажется, что весьма просто назначить, например, модификатор *TurboSmooth*, чтобы он автоматически выполнил нужные преобразования объекта. Однако так можно сделать только с простейшими полигональными объектами, если форма модели достаточно примитивна и симметрична. Чаще всего необходимо создавать защитные контуры, чтобы сохранить форму объекта при сглаживании, но чтобы при этом сглаживание работало нужным образом на тех участках, где оно необходимо.

Пример 1. Ниже приведен пример подготовки для сглаживания объекта цилиндрической формы. Чтобы создать защитные контуры для такой несложной геометрии, достаточно использовать модификатор Chamfer (Фаска), дальнейшая работа в полигональном моделировании не потребуется. Однако потребуется правильно настроить параметры модификатора, чтобы получить не фаску, а именно защитные контуры для сглаживания. Далее дано разъяснение принципа применения этого модификатора для создания именно защитных контуров. На рисунках 17-18 приведен пример настройки модификатора Chamfer (Фаска) для данного объекта и результат сглаживания.

На рисунке 17,а показан исходный объект цилиндрической формы высотой 50 мм, при сглаживании которого цилиндрические части должны быть круглыми, а плоские стороны должны сохранить свою форму. Если к данному объекту без подготовки его сетки применить модификатор TurboSmooth, результат получится неудовлетворительный (рисунок 17,б). Нужные защитные контуры можно создать с помощью модификатора фаска, но если его не настроить, результат также может оказаться неудовлетворительным. Например, на рисунке 17,в показано применение модификатора Фаска (Chamfer: Mitering = Uniform, Amount Type = Fixed, Amount = 1 mm, Segments = 2, Min.Angle = 20) с текущими настройками, на рисунке 17, г показан результат сглаживания объекта. Число сегментов фаски в данном случае излишне, а также фаска применена в том числе и к тем рёбрам, к которым её применение не нужно.

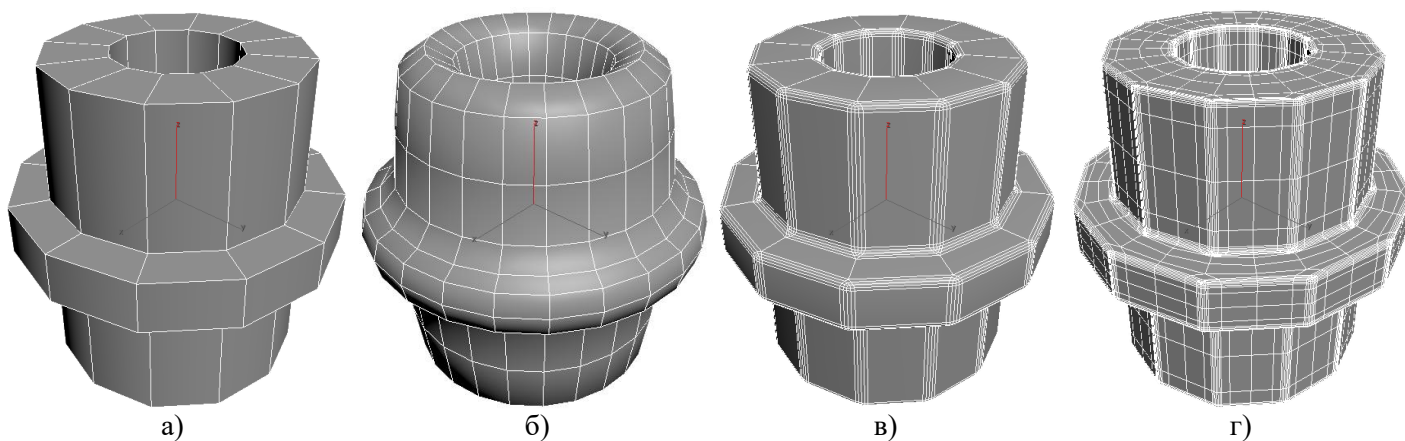


Рисунок 17 – Примеры неудовлетворительного сглаживания цилиндрического объекта: а) исходный объект; б) неудовлетворительный результат сглаживания без подготовки сетки; в) недостаточная настройка модификатора фаска для данного объекта (Chamfer: Segments = 2, Min.Angle = 20); г) неудовлетворительный результат сглаживания

Если настроить модификатор Chamfer должным образом, то можно получить защитный контур именно на тех участках, на которых необходимо сохранить форму. В данном случае нужно уменьшить число сегментов фаски до единицы (Segments=1) и увеличить значение минимального угла (Min.Angle), чтобы избежать применения фаски к рёбрам, составляющим стороны цилиндра. Для минимального угла можно выставить любое значение больше 40 градусов, но не более 90 градусов, иначе фаски (т.е. защитного контура) не будет и на тех участках, где он нужен.

Для создания именно защитного контура, который не искажает форму объекта, важно, чтобы в параметрах фаски был установлен фиксированный тип величины (Chamfer: Mitering = Uniform, Amount Type = Fixed) и значение глубины было равно единице (Depth = 1,0). Если значение глубины оставить по умолчанию (Depth = 0,5), то модификатор будет создавать некоторый скос, а это уже нежелательное изменение формы самого объекта. Размер фаски в мм подбирается в зависимости от размеров объекта, защитный контур будет находиться именно на этом расстоянии от контура рёбер объекта (в данном случае: Amount = 1 мм). На рисунке 18, в показан пример правильного создания защитных контуров для данного объекта с помощью модификатора Chamfer (Фаска).

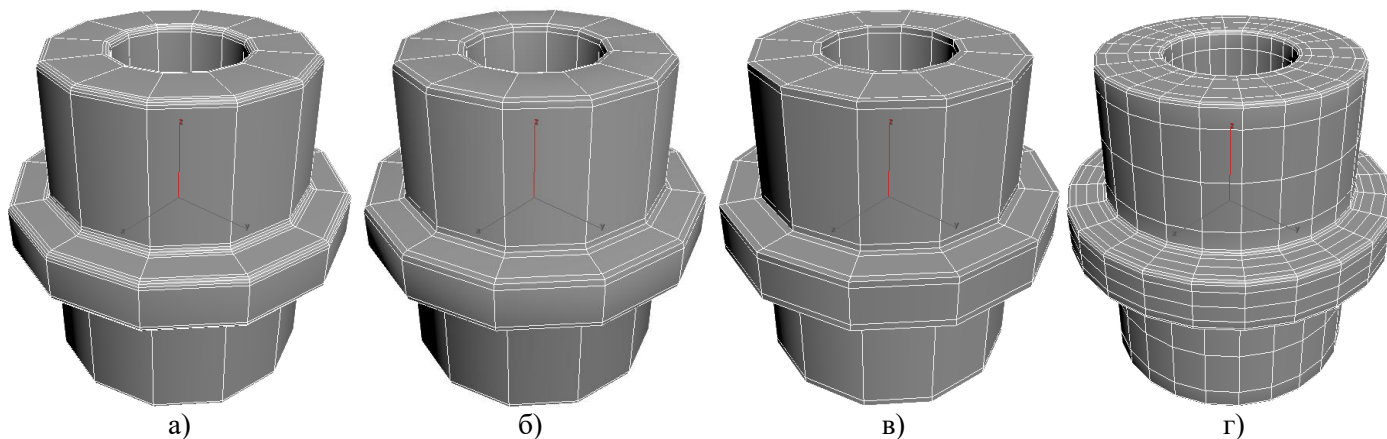


Рисунок 18 – Подготовка цилиндрического объекта для сглаживания: а) фаска применена к нужным рёбрам, но избыточно число сегментов фаски (Chamfer: Segments = 2, Min.Angle = 60, Depth = 0,5); б) нужное число сегментов фаски, но искажение формы объекта фаской (Chamfer: Segments = 1, Min.Angle = 60, Depth = 0,5); в) верная настройка модификатора фаска для данного объекта для создания защитных контуров (Chamfer: Segments = 1, Min.Angle = 60, Depth = 1,0); г) правильный результат сглаживания (TurboSmooth: Iterations = 1)

Иногда после создания защитных контуров с помощью модификатора *Chamfer* может потребоваться доработка сетки в модификаторе *Edit Poly*, чтобы сглаживание работало корректно, либо с целью оптимизации сетки.

Пример 2. Модель более сложной формы может потребовать работы разными инструментами полигонального моделирования при подготовке к сглаживанию. При применении модификатора *TurboSmooth* к неподготовленной модели, можно проверить, есть ли разрывы в сетке. Если их нет, то вся модель будет скругляться с искажением формы, но сетка будет оставаться единой непрерывной поверхностью (рисунок 19).

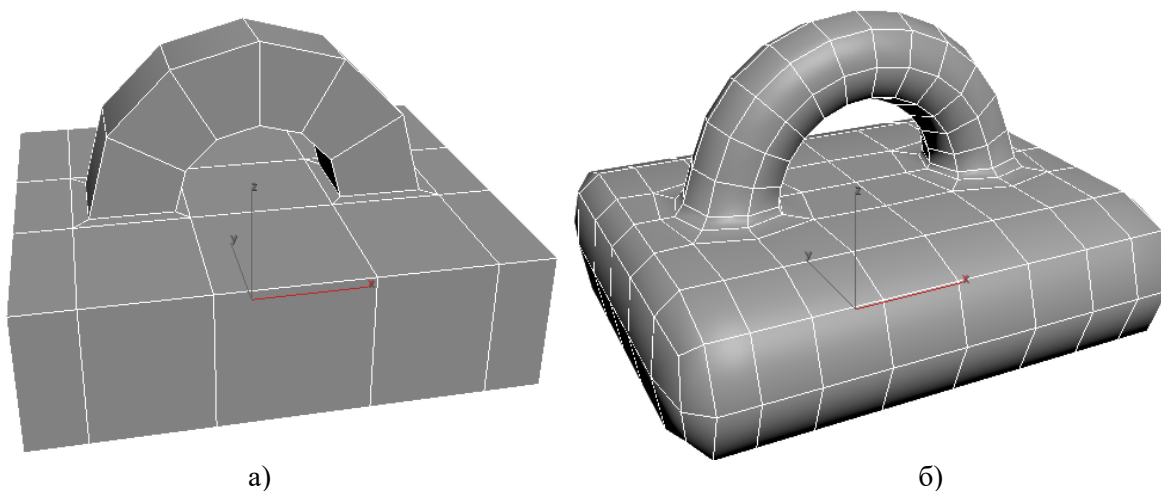
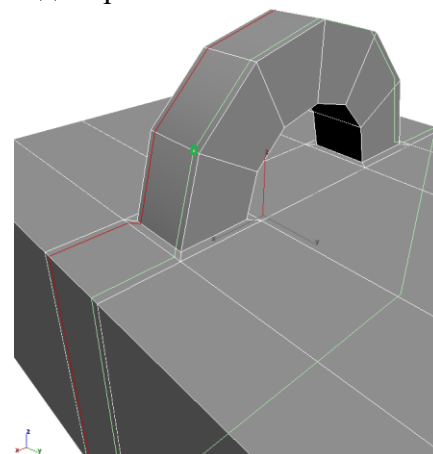
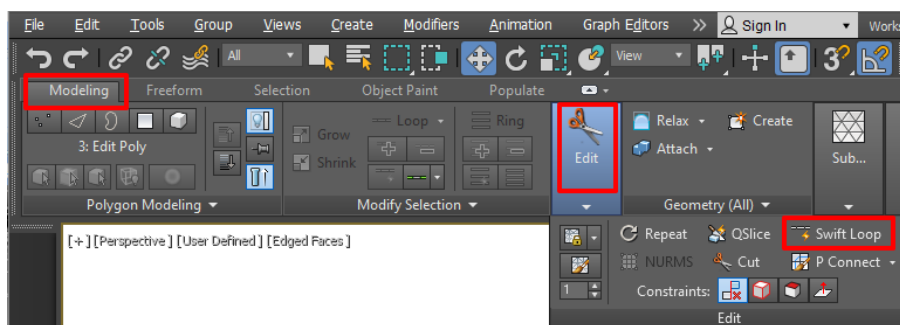


Рисунок 19 – Низко-полигональная модель: а) поверхностная сетка при создании объекта; б) сглаженная модель без подготовки сетки для сглаживания

Для быстрого создания защитных контуров часто бывает удобен инструмент Swift Loop (Быстрая петля, комбинация клавиш <Alt+L>), который находится на панели Риббон (Ribbon → Modeling → Edit → Swift Loop). Если навести курсор на ребро, перпендикулярно которому нужно создать контур, и щелкнуть мышью, будет создана петля рёбер, опоясывающая весь объект. На рисунке 20 приведен пример работы этого инструмента: красным цветом выделена только что созданная петля рёбер, зелёным цветом подсвечен контур, который будет создан в случае щелчка левой кнопкой мыши в указанном месте. Недостатком данного инструмента является то, что если создавать несколько таких контуров, нельзя сделать для них одинаковый отступ от края (никакие привязки с ним не работают, возможно только приблизительное расположение курсора «на глаз», потому приходится стараться соблюдать одинаковый отступ, ориентируясь по другим элементам). Однако благодаря этому инструменту работу по созданию защитных контуров у объектов сложной формы можно выполнить значительно быстрее, чем обычными инструментами полигонального моделирования.



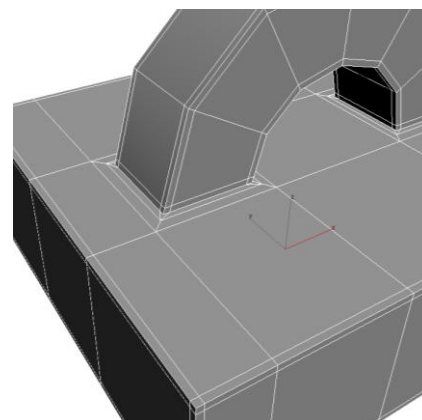
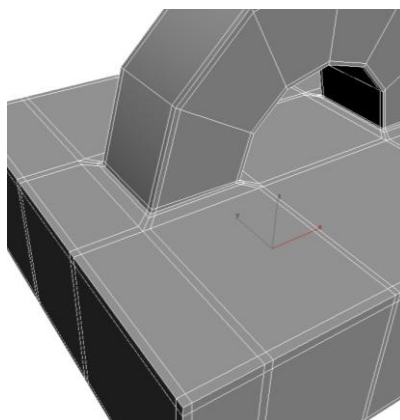
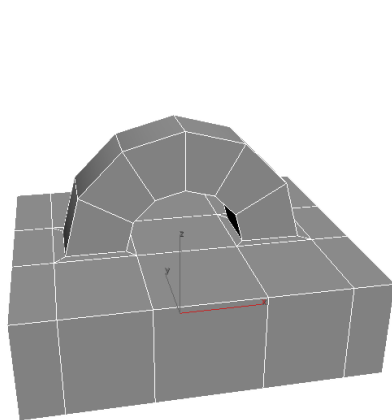
а)

б)

Рисунок 20 – Использование инструмента Swift Loop (Быстрая петля): а) расположение инструмента на панели Риббон (Ribbon); б) применение инструмента: красным цветом выделена только что созданная петля рёбер, зелёным цветом подсвечен контур, который будет создан в случае щелчка левой кнопкой мыши, курсор в данном случае наведён на место на ребре, отмеченное зеленым кружком

Т.к. контур рёбер замкнутый и опоясывает весь объект, а петля может идти неоптимальным образом, возможно после работы быстрых петель (Swift Loop) потребуется оптимизация и доработка сетки на уровне полигонального моделирования (Edit Poly).

При создании защитных контуров возможно сочетание различных инструментов: и модификатора фаски (Chamfer), и быстрой петли (Swift Loop), и различных операций полигонального моделирования. На рисунке 21 показано сравнение результата создания защитных контуров разными инструментами – быстрой петлей (Swift Loop) и модификатором фаска (Chamfer).



а)

б)

в)

Рисунок 21 – Результат создания защитных контуров разными инструментами: а) исходный объект; б) с помощью быстрой петли (Swift Loop); в) с помощью модификатора фаски (Chamfer)

В качестве первого шага подготовки сетки для сглаживания можно использовать любой из этих инструментов – быструю петлю (Swift Loop) или модификатор фаска (Chamfer). Но т.к. с помощью модификатора фаска (Chamfer) можно получить контуры на одинаковом расстоянии, т.е. сетка будет ровной, то в качестве первого шага для данной модели был взят именно модификатор фаска (рисунок 21,в), а не быстрая петля (Swift Loop). После применения фаски произведена доработка в модификаторе редактирования полигонов (Edit Poly): инструментом быстрой нарезки (QuickSlice) при включенной трехмерной привязке только к вершинам (Vertex) на виде сверху (Top) были получены дополнительные контуры (рисунок 22,а). Затем лишние контуры рёбер были выделены двойным щелчком мыши (рисунок 22,б) и удалены (<Ctrl+BackSpace>), а нужные оставлены. Полученный результат на рисунке 22,в.

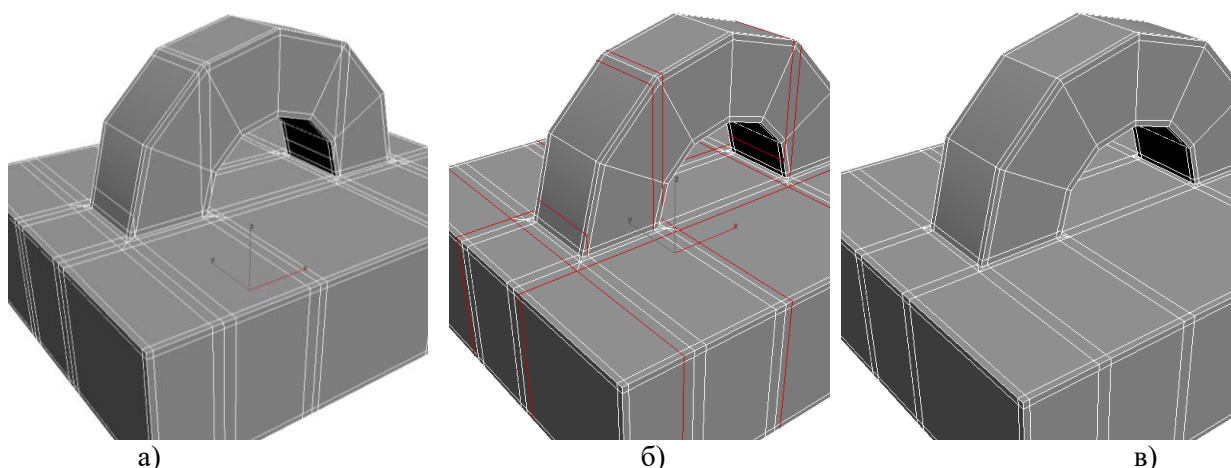


Рисунок 22 – Доработка сетки модели для сглаживания после создания защитных контуров модификатором фаска (Chamfer): а) результат быстрой нарезки (QuickSlice) с привязкой только к вершинам (Vertex) на виде сверху (Top); б) выделение лишних контуров рёбер; в) результат после удаления (<Ctrl+Backspace>) лишних контуров рёбер

Далее нужно сделать оптимизацию углов. В данном случае она довольно простая, алгоритм этой работы приведен на рисунке 23. Полученная модель подготовлена для сглаживания (рисунок 24,а). Последним шагом нужно выполнить оптимизацию сетки всей модели: свести рёбра в углы там, где это возможно, а также некоторые рёбра можно распределить более равномерно по полигону (рисунок 24,б). Результат сглаживания готовой модели показан на рисунке 24,в.

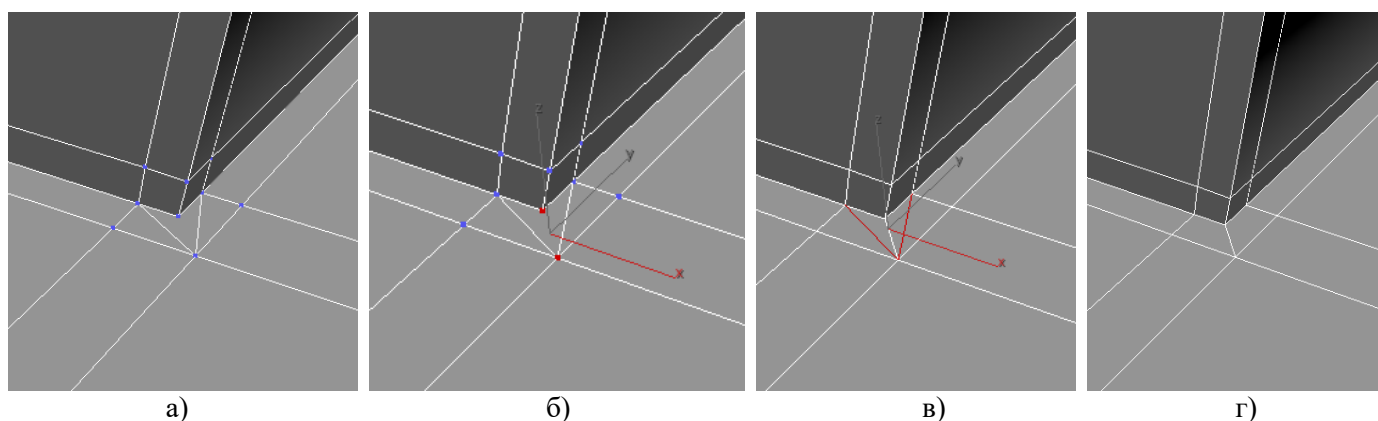


Рисунок 23 – Оптимизация углов: а) исходная сетка угла с треугольными полигонами и четырехугольным полигоном нежелательной формы; б) соединение выделенных вершин ребром (Connect); в) удаление (<Ctrl+Backspace>) двух соседних диагональных рёбер; г) полученная правильная топология угла

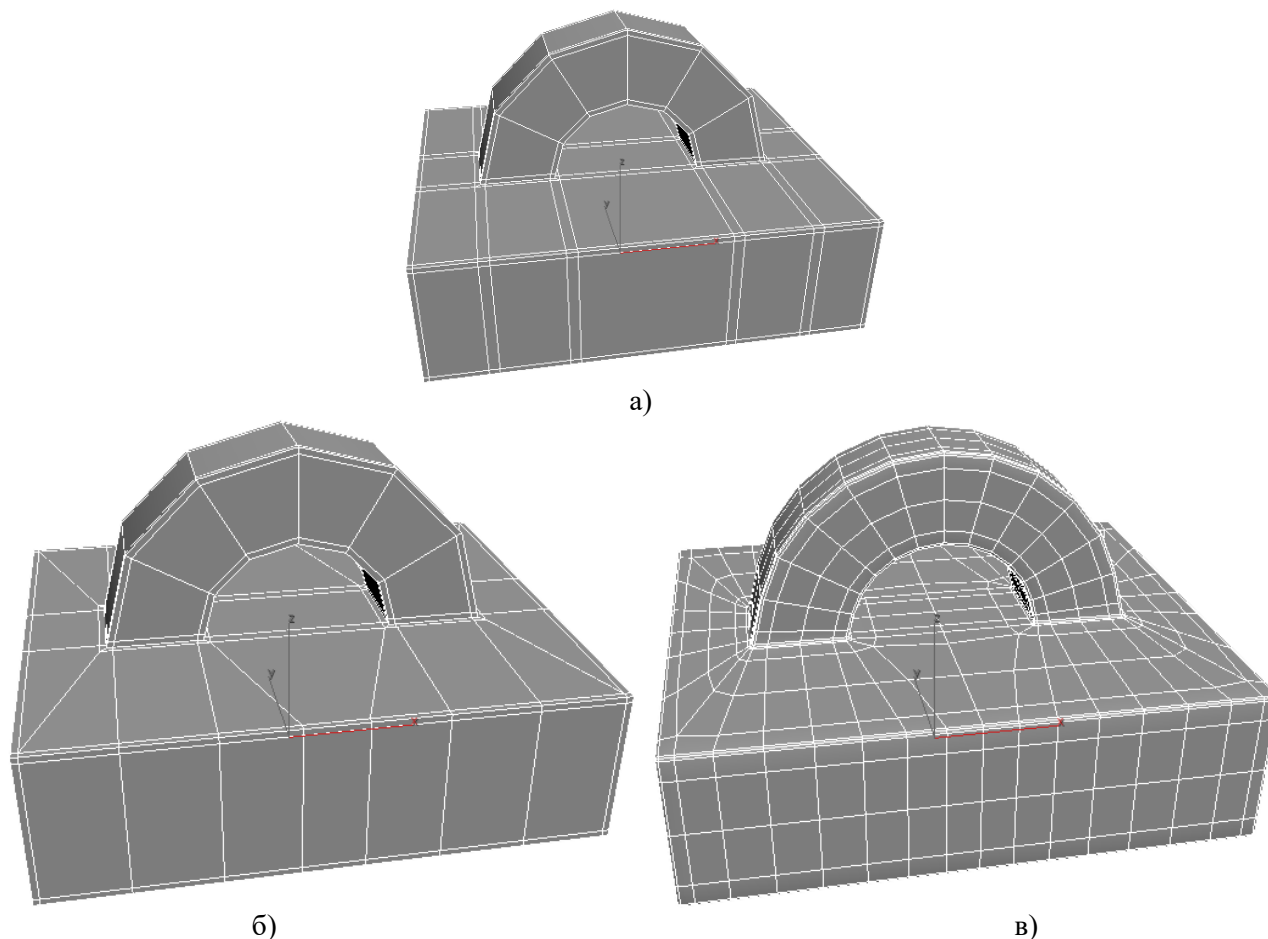


Рисунок 24 – Низко-полигональная модель, подготовленная для сглаживания: а) созданы нужные защитные контуры, все полигоны четырехугольные и имеют правильную форму; б) выполнена оптимизация сетки модели – сведение рёбер в углы; в) сохранена форма модели при сглаживании (TurboSmooth: Iterations = 1)

После оптимизации обязательно нужно ещё раз проверить модель на наличие нечетырехугольных полигонов, и если такие есть, доработать сетку.

При работе над сеткой следует избегать неправильной формы полигонов, т.к. это приводит к искажениям поверхностной сетки модели. На рисунке 25 показан пример правильной и неправильной формы полигонов.

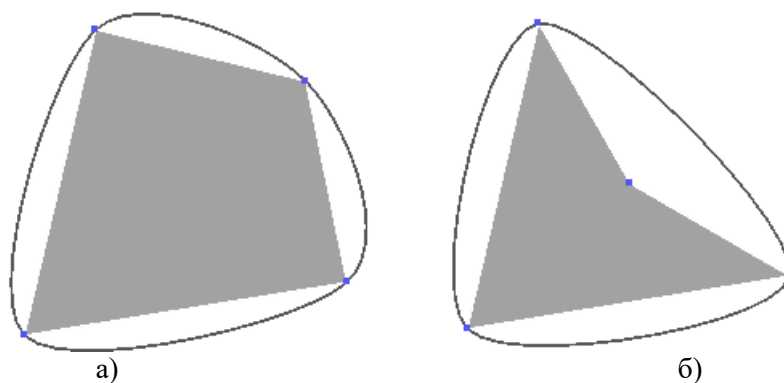


Рисунок 25 – Пример правильного и неправильного 4-угольного полигонов: а) через все вершины правильного полигона можно описать кривую в форме окружности; б) через какую-либо из вершин неправильного полигона нельзя провести кривую в форме окружности (такой полигон похож на стрелку)

Часто полигоны неправильной формы образуются в углах при создании фаски (как на рисунке 23,а). В этом случае необходимо вручную доработать поверхностную сетку объекта в зависимости от его формы.

Если у модели углы имеют чуть более сложную сетку, на рисунке 26 показано, как уменьшить число контуров рёбер модели (т.е. в итоге – число полигонов модели) за счёт оптимизации углов. При этом следует помнить, что прежде чем удалять «лишний» контур рёбер, следует проверить, куда он приходит, и выполнена ли оптимизация того угла на том участке.

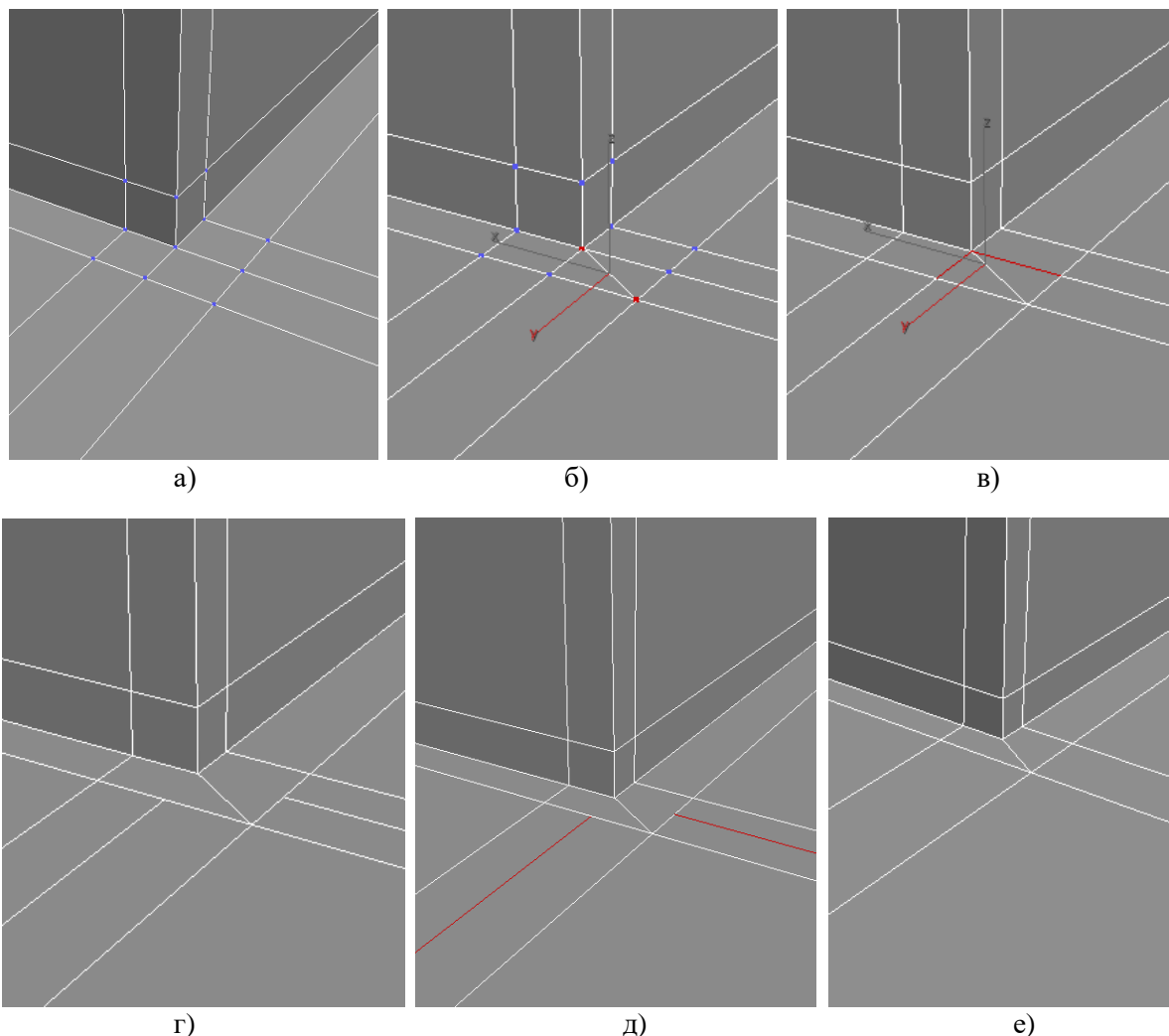


Рисунок 26 – Оптимизация сетки – уменьшение числа контуров рёбер: а) исходная сетка с тремя контурами рёбер для сохранения формы угла; б) соединение выделенных вершин ребром (Connect); в) удаление (<Ctrl+Backspace>) двух соседних диагональных рёбер; г) результат после удаления двух рёбер, на этом этапе нужно выполнить такую же оптимизацию для того угла, куда эти контуры приходят; д) удаление (<Ctrl+Backspace>) выделенных двойным щелчком мыши контуров рёбер; е) результат оптимизации с двумя контурами рёбер для сохранения формы угла

Совет! Подготовку для сглаживания нужно делать в новом модификаторе *Edit Poly*, но не в том, в котором была создана форма модели. Это позволит в случае фатальной ошибки выполнить заново только часть работы, а не всю работу целиком.

Совет! Если форма модели сложная, можно выполнить подготовку для сглаживания в нескольких модификаторах *Edit Poly*. Например, в одном выполнить работу с защитными контурами, в другом проработать углы, в следующем выполнить оптимизацию сетки.

Важно! Группы сглаживания или параметр сгиба (Crease), или жесткость ребра (Hard) можно использовать только при создании заготовки низко-полигональной модели. Если этот инструментарий использован в последнем модификаторе редактирования полигонов (*Edit Poly*), и сразу после этого применено сглаживание (*TurboSmooth*), то такой подход будет неверным. Подготовка объекта для сглаживания должна быть выполнена именно за счёт топологии: правильная сетка и защитные контуры.

Итак, в зависимости от назначения модели – низко-полигональная модель или низко-полигональная для сглаживания – может отличаться её поверхностная сетка. На рисунке 27 приведено сравнение сетки таких моделей для одной и той же формы объекта. Например, создана модель нужной формы (рисунок 27,а). Если требуется именно низко-полигональная модель, то её сетку необходимо оптимизировать (как на рисунке 27,б). Если требуется низко-полигональная модель для сглаживания, то удобнее оставить сетку как на рисунке 27,а, при необходимости разрядить сетку на нужных участках, затем создать защитные контуры, и только потом оптимизировать сетку уже с созданными контурами.

Следует обратить внимание, что иногда скругленные части у низко-полигональной модели (в ЛР4) могут становиться высоко-полигональными у модели для сглаживания (в ЛР5), т.е. при сглаживании их сетка становится слишком плотной. Чтобы этого избежать, иногда бывает необходимо либо их доработать в полигональном моделировании (разрядить сетку), либо изготовить заново с меньшим числом сегментов, учитывая дальнейшее сглаживание. На примере данной модели можно увидеть, что число сегментов «ручки» у низко-полигональной модели (рисунок 27,б) составляет 10, тогда как для низко-полигональной модели для сглаживания (рисунок 27,в) их достаточно всего 6.

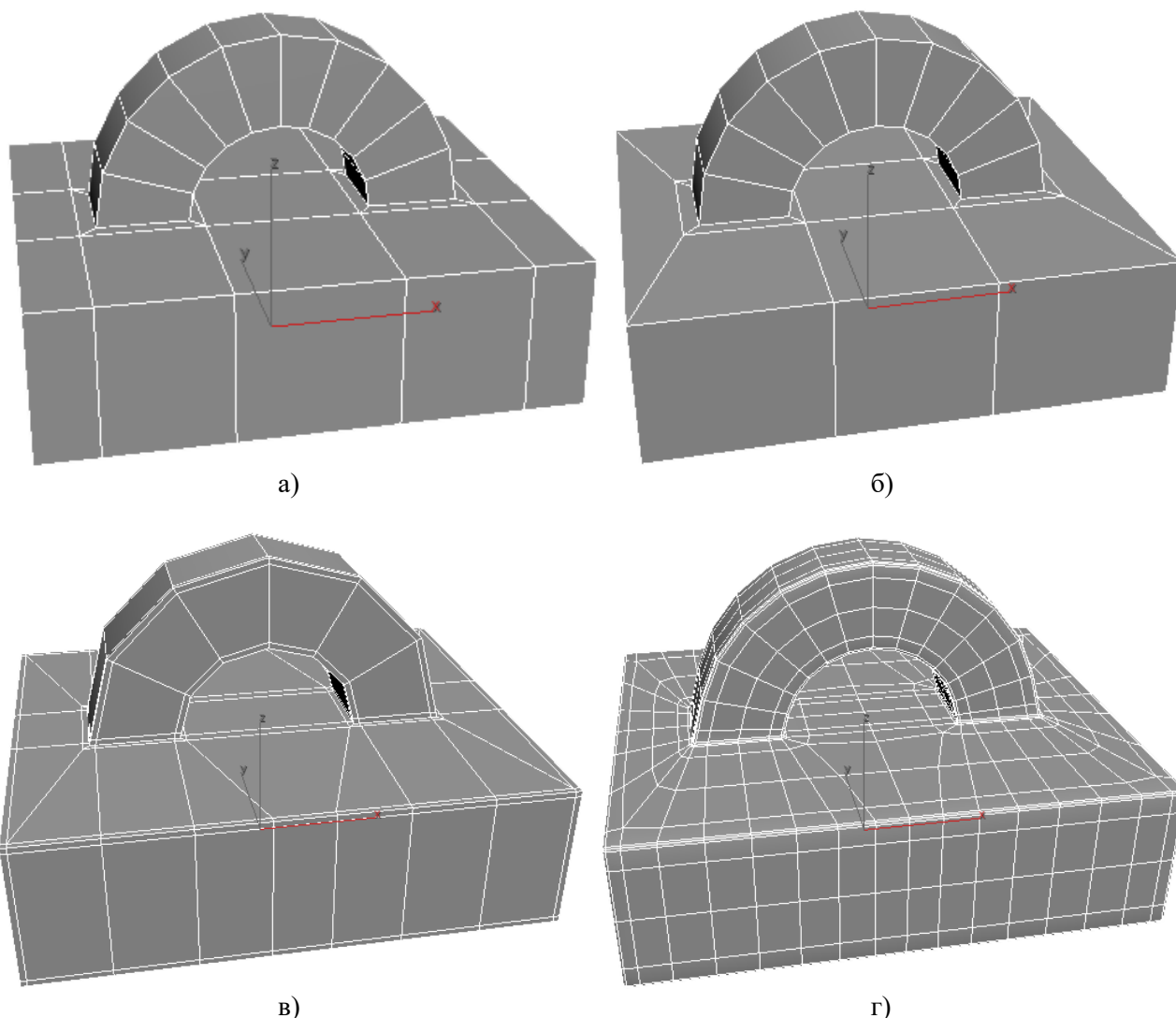


Рисунок 27 – Разная поверхностная сетка в зависимости от назначения модели: а) поверхностная сетка, полученная при изготовлении нужной формы, без оптимизации; б) низко-полигональная модель с оптимизацией сетки; в) низко-полигональная модель для сглаживания с оптимизацией сетки; г) низко-полигональная модель со сглаживанием

При визуализации модели уже с материалами у низко-полигональных моделей, при приближении к ним, становятся заметны неправдоподобно острые углы. Потому именно сглаженные модели при их визуализации позволяют достичь наибольшей правдоподобности их формы.