

# Видимость в пространстве среди многогранников

Интерактивное приложение, в котором видна работа описанного алгоритма, размещено по адресу <https://apinges.github.io/v2/index-demo2.html?lang=ru>.

Для работы приложения необходим компьютер с графическим процессором (GPU).

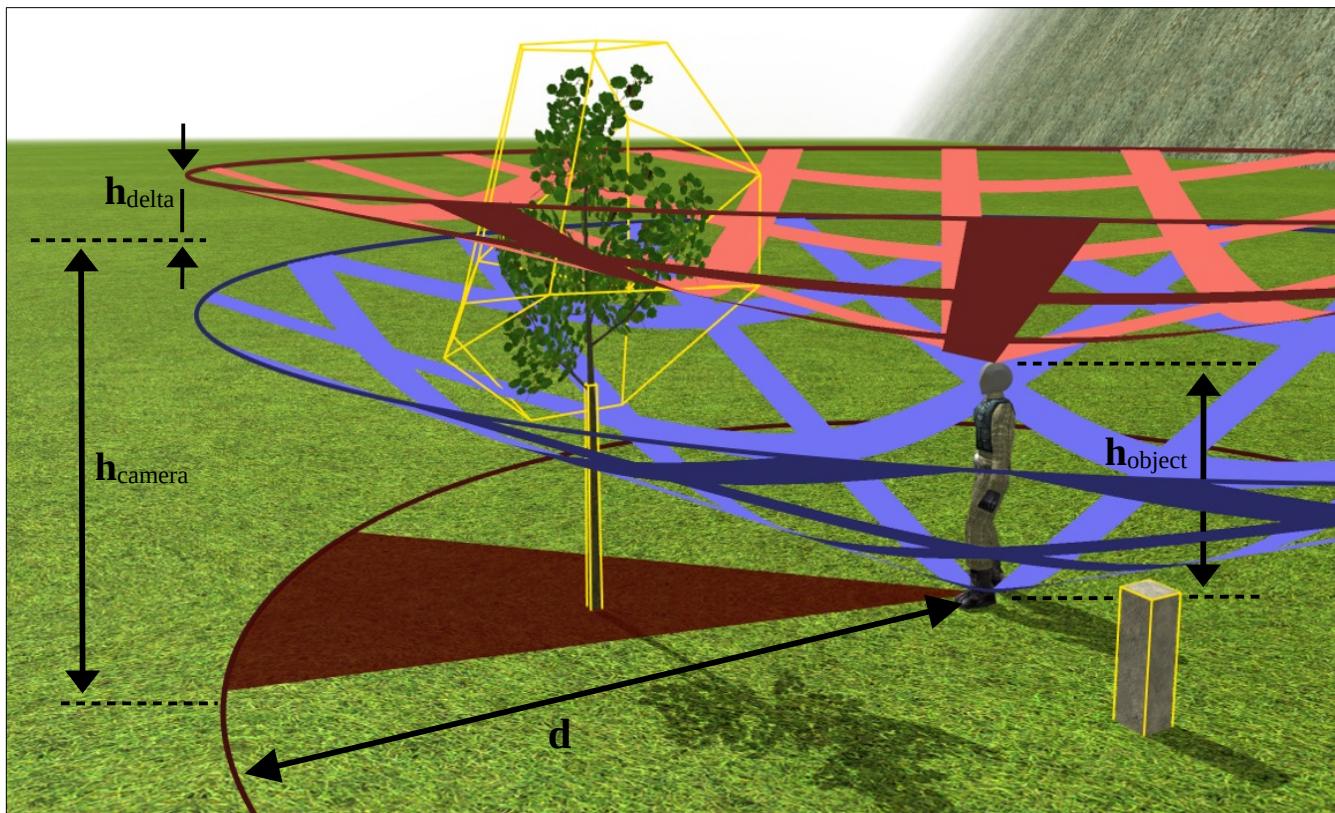
## Что нового по сравнению с версией 2024.06

- Переработано построение системы координат при обработке грани многогранника, что существенно сократило дальнейшие вычисления;
- Переработано “[Добавление интервала](#)”;
- Добавлены описания “[Улучшение производительности](#)”, “[Сопровождающие алгоритмы](#)”.

## Условия задачи

При разработке приложения с трехмерной компьютерной графикой возникла задача расположения “камеры” (откуда формируется изображение на экран) относительно заданного объекта так, чтобы были выполнены условия:

- Объект должен находиться в центре экрана;
- Расстояние до объекта по горизонтали  $d$ , высота расположения камеры  $h_{camera}$ ;
- Объект и камера имеют определенные размеры. Заданы высота объекта  $h_{object}$  и половина высоты камеры  $h_{delta}$ ;
- Какие-либо препятствия не должны затруднять видимость объекта.

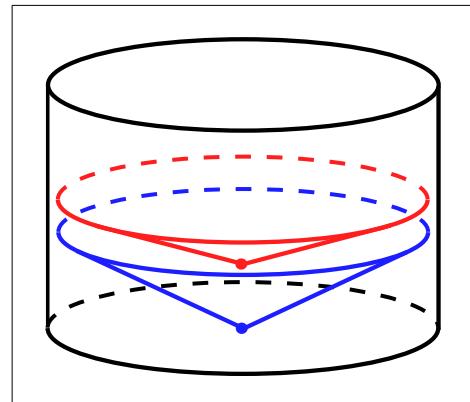


Необходимо найти множество точек, удовлетворяющих условиям задачи. Это отрезки окружности, расположенной горизонтально на высоте  $h_{camera}$ , с радиусом  $d$  и центром, по вертикали совпадающим с центром объекта.

## Геометрические построения

- Создается система координат с началом координат в центре объекта. Ось Y направлена вверх, что типично для приложений компьютерной графики;
- Строится вертикальный цилиндр с радиусом  $d$ , ось которого совпадает с осью Y;
- Строятся два конуса с осями, совпадающими с осью цилиндра, и вершинами, находящимися в самой нижней и самой верхней частях объекта.

Если в объем, ограниченный 3 фигурами, попадает препятствие, оно затрудняет видимость.



## Уравнения конуса и цилиндра

Уравнение конуса с осью, совпадающей с осью Y:

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} + \frac{z^2}{c^2} = 0.$$

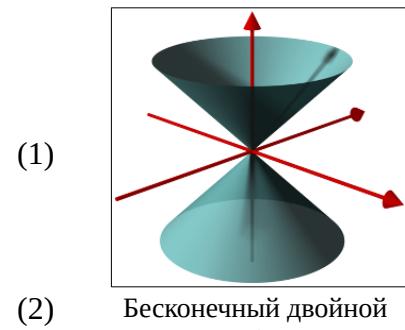
В данном случае у нас конус вращения,  $a=c$ . Используем упрощенную формулу  $a=c=1$ :

$$x^2 + z^2 - b^{-2} y^2 = 0.$$

Это бесконечный двойной конус. В программный объект типа конус добавлено свойство `halfSgn`, определяющее, какая часть используется: верхняя(1), нижняя(-1) или обе(0).

Для создания конуса по заданной вершине  $\vec{A}$  и точке на поверхности  $\vec{P}$ , необходимо найти параметр  $b$ :

$$\vec{V} = \vec{P} - \vec{A}; \quad b = \sqrt{V_y^2 / (V_x^2 + V_z^2)}, \quad V_y \neq 0, V_x^2 + V_z^2 > 0. \quad (3)$$



(2)

Бесконечный двойной конус (изображение из Wikipedia).

Уравнение бесконечного цилиндра:

$$x^2 + z^2 - r^2 = 0. \quad (4)$$

## Объекты, препятствующие видимости

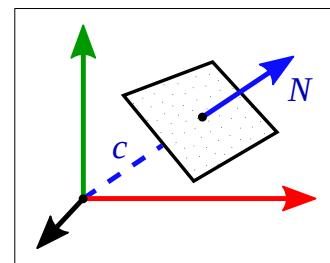
В приложении объекты заданы многогранниками. Структура данных - пространственный индекс для протяженных (ненулевого размера) объектов - позволяет эффективно найти объекты, имеющие общие точки с заданным прямоугольным параллелепипедом, грани которого параллельны осям координат (встречается название Axis Aligned Box или AA Box).

## Обзор алгоритма

- Проводится выборка из пространственного индекса. Каждый многогранник переводится в созданную ранее систему координат с началом в центре объекта;
- Проверяется, как соотносятся 3 фигуры (2 конуса и цилиндр) с ограничивающей сферой многогранника. Если сфера полностью находится за пределами объема, ограниченного 3 фигурами, то многогранник не влияет на видимость;
- Проводится обработка каждой грани многогранника. При этом добавляются интервалы на окружности, закрывающие видимость. Добавляемые интервалы могут расширять уже имеющиеся или быть абсорбированы.

## Обработка грани многогранника

- Грани многогранника соответствует плоскость. На плоскости создается система координат (начало и 2 оси находятся на плоскости);
- В эту систему координат переводятся 3 фигуры: 2 конуса и цилиндр. Это функции трех переменных. Переменная по оси, не лежащей на плоскости, приравнивается к нулю и получаются кривые - функции двух переменных;
- В систему координат на плоскости переводятся ребра многогранника, получается многоугольник;
- Находятся точки пересечения сторон многоугольника и кривых, а также двух кривых. По спискам точек пересечения определяются интервалы, закрывающие видимость.

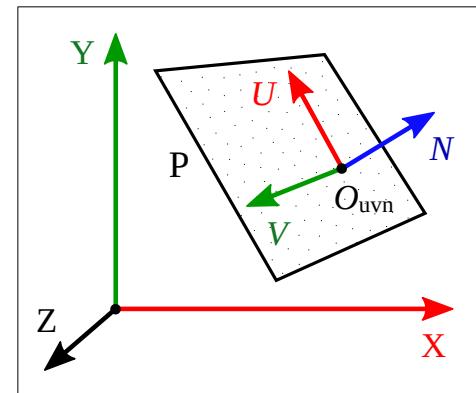


Hesse Normal Form – плоскость задается единичным вектором-перпендикуляром  $N$  и расстоянием  $c$  до начала координат вдоль  $N$ .  
На изображении  $c < 0$ .

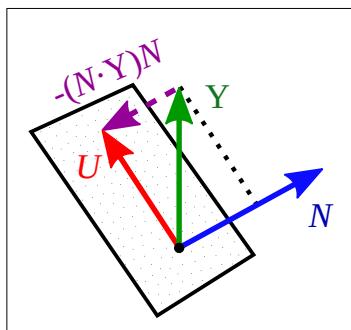
## Координаты на плоскости

Обозначим плоскость  $P$ . Для создания системы координат необходимы 3 ортогональных единичных вектора (обозначим их  $U, V, N$ ) и точка начала координат  $O_{uvn}$ . Нам нужно, чтобы оси  $Y, U, N$  находились в одной плоскости, соответственно  $V$  перпендикулярен им, а также чтобы угол между векторами  $Y$  и  $U$  был не более  $90^\circ$ . Такой выбор координат позволит существенные сокращения, в том числе кривые будут не повернуты относительно осей.

- Вектор  $N$ , перпендикулярный  $P$ , берется из свойств  $P$ .
- $O_{uvn}$  берется как ближайшая точка на  $P$  к началу координат  $O_{xyz}$ , вычисляется по формуле  $O_{uvn} = -cN$ , где  $c$  – расстояние от  $P$  до  $O_{xyz}$  в направлении  $N$ . В результате оси  $N$  и  $Y$  находятся в одной плоскости.



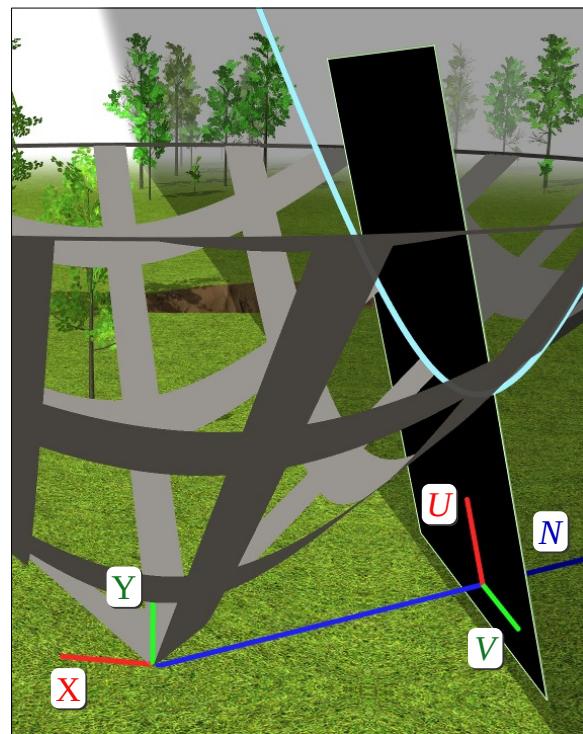
Система координат на плоскости.



- Вектор  $U$  создается проекцией  $Y$  на плоскость:  $U = Y - (N \cdot Y)N$  и приводится к единичному. Если окажется  $\|U\|^2 = 0$ , плоскость горизонтальна, берется  $U = X = (1, 0, 0)$ .

При таком построении угол между  $U$  и  $Y$  получится не более  $90^\circ$  (скалярное произведение  $U \cdot Y \geq 0$ ), даже если угол между  $N$  и  $Y$  более  $90^\circ$ .

- $V$  равен векторному произведению  $U \times N$ .



Координаты на плоскости, визуализированные в приложении, конус и получившаяся кривая.

## Преобразование координат

Для преобразования из одной системы координат в трехмерном пространстве в другую используют матрицы 4x4. Для составления преобразования из координат  $uvn$  в  $xxyz$ , базисные векторы  $U, V, N$  и точка  $O_{uvn}$  записываются в столбцы матрицы  $M$ :

$$M = \begin{bmatrix} U_x & V_x & N_x & -cN_x \\ U_y & V_y & N_y & -cN_y \\ U_z & V_z & N_z & -cN_z \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad M^{-1} = \begin{bmatrix} U_x & U_y & U_z & 0 \\ V_x & V_y & V_z & 0 \\ N_x & N_y & N_z & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Матрица обратного преобразования  $M^{-1}$  может быть получена инвертированием  $M$ . В данном случае применим упрощенный подход: ведущая субматрица 3x3 ортонормальная, ее инверсия равна транспонированию. Точка  $O_{xxyz}$  в координатах  $uvn$ , по построению, равна  $(0, 0, c)$ .

Для преобразования координат векторов, точек выполняется умножение матрицы на вектор. Это обычное действие в 3D компьютерной графике. Выполним преобразование конуса - функции трех переменных.

Сначала запишем преобразование точки из  $uvn$  в  $xxyz$  в развернутом виде:

$$\begin{cases} x = U_x u + V_x v + N_x n - c N_x \\ y = U_y u + V_y v + N_y n - c N_y \\ z = U_z u + V_z v + N_z n - c N_z \end{cases} \quad (6)$$

## Преобразование координат конуса

Для преобразования конуса из  $xxyz$  в  $uvn$  возьмем уравнения (6), приравняем  $n$  к нулю и подставим в уравнение конуса (2):

$$(U_x u + V_x v - c N_x)^2 + (U_z u + V_z v - c N_z)^2 - b^{-2} (U_y u + V_y v - c N_y)^2 = 0; \quad (7)$$

Раскроем скобки и сгруппируем по степеням переменных  $u, v$ :

$$\begin{aligned} & u^2 (U_x^2 + U_z^2 - b^{-2} U_y^2) + 2uv (U_x V_x + U_z V_z - b^{-2} U_y V_y) + v^2 (V_x^2 + V_z^2 - b^{-2} V_y^2) \\ & + 2u (-U_x c N_x - U_z c N_z + b^{-2} U_y c N_y) + 2v (-V_x c N_x - V_z c N_z + b^{-2} V_y c N_y) \\ & + c^2 (N_x^2 + N_z^2 - b^{-2} N_y^2) = 0; \end{aligned} \quad (8)$$

Выполним дальнейшие действия для упрощения:

- $V$  перпендикулярен оси  $Y$ ,  $V_y = 0$ ;
- $U$  и  $V$  перпендикулярны, скалярное произведение  $U \cdot V$  равно нулю. Развернем скалярное произведение:  $U \cdot V = U_x V_x + U_y V_y + U_z V_z$ . Так как  $V_y = 0$ , то  $U_x V_x + U_z V_z = 0$ .
- $V$  и  $N$  перпендикулярны, и если один из них умножить на число (масштабировать), то скалярное произведение останется равным нулю, итого:  $V_x c N_x + V_z c N_z = 0$ . Это свойство скалярного произведения – ассоциативность при умножении вектора на число:  $V \cdot (c N) = c(V \cdot N)$ .
- $\|V\|^2 = V_x^2 + V_y^2 + V_z^2 = V_x^2 + V_z^2 = 1$ .

Итого получается:

$$\begin{aligned} & u^2 (U_x^2 + U_z^2 - b^{-2} U_y^2) + v^2 \\ & + 2u c (-U_x N_x - U_z N_z + b^{-2} U_y N_y) + c^2 (N_x^2 + N_z^2 - b^{-2} N_y^2) = 0. \end{aligned} \quad (9)$$

Это алгебраическая функция двух переменных 2 степени (одно из названий – коническое сечение).

После замены  $x=u$ ,  $y=v$  получается:

$$\begin{aligned}
 Ax^2 + Bxy + Cy^2 + Dx + Ey + F &= 0, \quad \text{где} \\
 A &= U_x^2 + U_z^2 - b^{-2}U_y^2, \\
 B &= 0, \\
 C &= 1, \\
 D &= 2c(-U_x N_x - U_z N_z + b^{-2}U_y N_y), \\
 E &= 0, \\
 F &= c^2(N_x^2 + N_z^2 - b^{-2}N_y^2).
 \end{aligned} \tag{10}$$

$B = 0$  означает, что кривая не повернута относительно осей,  $E = 0$  не смещена вдоль оси Y.

Для цилиндра коэффициенты выводятся аналогично.

## Подготовка к преобразованию координат

Перед преобразованием координат решаются следующие задачи:

- Определить тип кривой (эллипс, гипербола и т.п.) по взаимному расположению конуса и плоскости. Это можно также сделать после получения коэффициентов, вычислив дискриминант и детерминант, но предложенный метод проще.
- Минимизировать число типов кривых, которые могут получиться в результате, не допустить дегенеративных случаев и численных проблем.

Выполняются действия:

1. Вычисляется разница между наклонами (к оси Y) конуса и плоскости  $\delta = 1/b - |N_y| / \sqrt{N_x^2 + N_z^2}$ :
 
$$\begin{cases} \delta < 0 : \text{ эллипс} \\ \delta = 0 : \text{ парабола} \\ \delta > 0 : \text{ гипербола} \end{cases}$$

При  $|\delta| < \epsilon$  возникают численные проблемы, такие как сокращение точности и переполнение экспоненты (например, получаются эллипсы с длиной оси  $10^{15}$  и более). Парабола  $\delta = 0$  (точное совпадение наклонов) возникает очень редко, она не обрабатывается для минимизации числа типов кривых. При  $|\delta| < \epsilon$  (в приложении  $\epsilon = 3 \cdot 10^{-5}$ ) плоскость слегка поворачивается.

2. Если расстояние между вершиной конуса и плоскостью менее  $\epsilon$ , плоскость слегка передвигается в направлении  $N$ . Это не допускает получения пересекающихся прямых, а также точки.
3. В случае цилиндра проверяется  $|N_y| < \epsilon$  (ось почти параллельна плоскости). Например, у объектов с вертикальными плоскостями при манипулировании ими появляются погрешности при округлениях, плоскости отклоняются от вертикальных и получаются эллипсы с большой длиной оси. В таких случаях плоскость ставится строго вертикально  $N_y = 0$ .
4. Если ось цилиндра параллельна плоскости, проверяется расстояние между цилиндром и плоскостью  $c - r$ . Не допускается расстояние между параллельными прямыми менее  $\epsilon$ .

После подготовки типы возникающих кривых ограничены эллипсом, гиперболой и параллельными прямыми, и кривые не имеют численных проблем.

В результате вносимых изменений, результат будет не соответствовать входным данным. При этом:

- Изменения небольшие - в диапазонах длины, которые встречаются в приложении, разница не превышает миллиметра. Это считено допустимым.
- Перемещения плоскостей происходят вдоль вектора  $N$ , который у многогранников в приложении направлен наружу, то есть при вносимых изменениях сокращаются интервалы видимости.

## Составление списков пересечений

После получения многоугольника и 3 кривых на плоскости, составляются списки пересечений:

- Выполняется пересечение каждой стороны многоугольника с каждой из 3 кривых.  
См. “[Приложение В. Пересечение отрезка и квадратичной кривой](#)”. Если точка пересечения находится за пределами объема, ограниченного 3 фигурами, то она не добавляется в список.
- Выполняется пересечение кривой, полученной от цилиндра, с 2 кривыми, полученными от конусов. См. “[Приложение С. Пересечение двух квадратичных кривых](#)”.

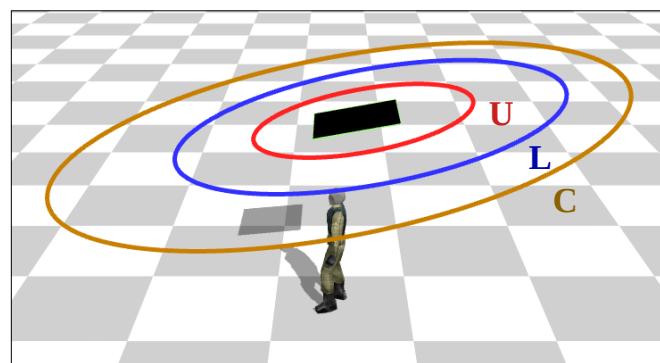
## Отсутствие пересечений

Берется любая вершина многоугольника.

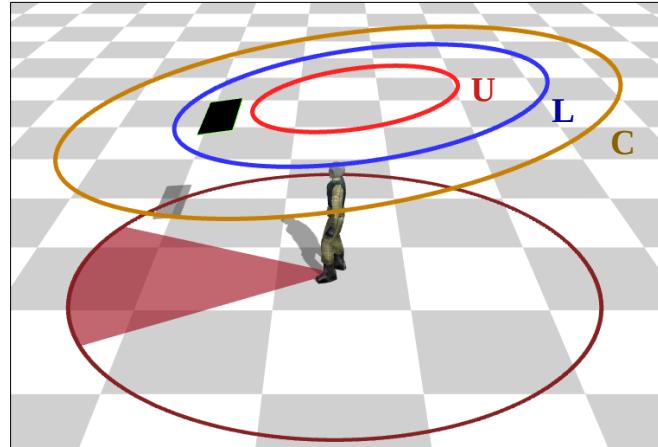
Проводится проверка взаимного расположения этой точки и кривой, полученной от верхнего конуса (см. раздел “[Точка внутри кривой](#)”).

Если точка находится внутри, то многоугольник не влияет на видимость (пример на изображении справа).

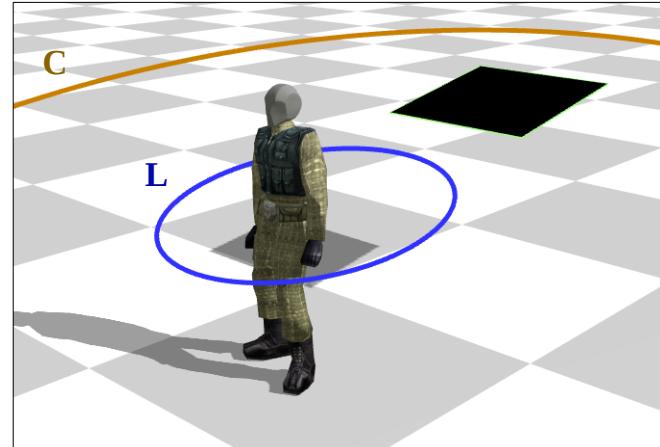
Далее проверяется - находится ли точка внутри кривой от нижнего конуса. Если да, то возможны варианты.



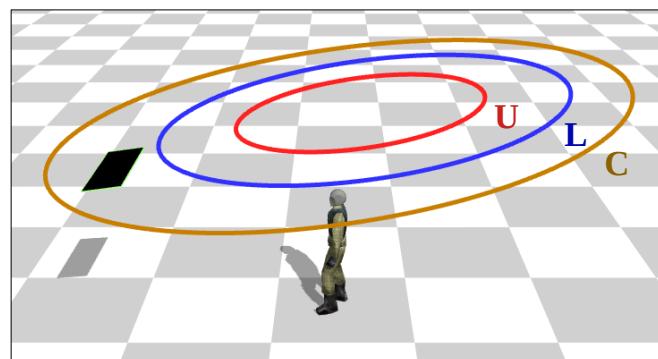
Обозначения: **U** (Upper) – кривая от верхнего конуса;  
**L** (Lower) – кривая от нижнего конуса;  
**C** (Cylinder) – кривая от цилиндра.



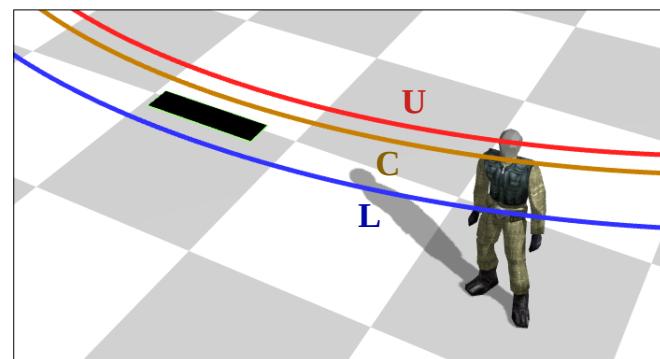
Многоугольник внутри кривой от нижнего конуса **L**, формируются закрывающие обзор интервалы.



Отсутствует кривая от пересечения верхнего конуса и плоскости **U**.

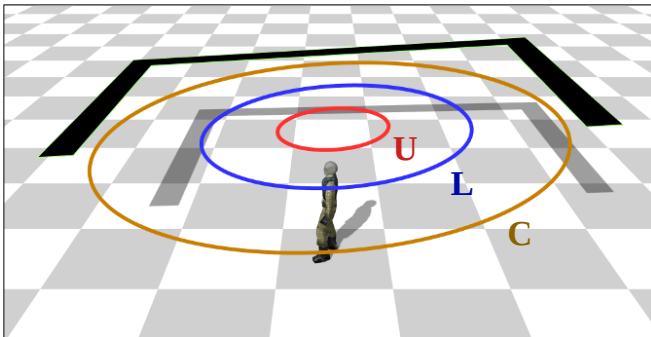


Многоугольник снаружи кривой от нижнего конуса **L**, не влияет на видимость.

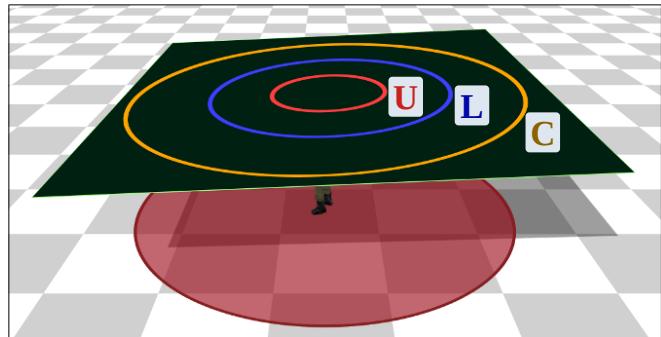


Многоугольник снаружи кривой от цилиндра **C**, не влияет на видимость.

После проведенных проверок остается вариант – все вершины многоугольника находятся снаружи кривой от нижнего конуса.



Кривая от нижнего конуса **L** снаружи многоугольника, нет влияния на видимость.



Кривая от нижнего конуса **L** внутри многоугольника, закрывает обзор на 360°.

Для проверки нахождения кривой внутри многоугольника, берется любая точка на кривой или внутри нее (в приложении используется [центр кривой](#)) и решается задача о принадлежности точки многоугольнику.

Бесконечные кривые находятся внутри многоугольника не могут.

## Обработка списков пересечений

Было составлено 4 списка (3 кривых и многоугольник). Каждому пересечению соответствует точка, параметр  $t$  (подробнее см. “[Параметр вдоль кривой](#)”), идентификатор второго объекта пересечения, тип – внутрь или наружу (см. “[Типы пересечения IN и OUT](#)”).

Алгоритм корректно обрабатывает все возможные случаи, в том числе невыпуклые многоугольники. В текущей версии камера всегда выше верхней точки объекта, используются только верхние половины конусов, в случае изменений будет необходим пересмотр / переформулировка.

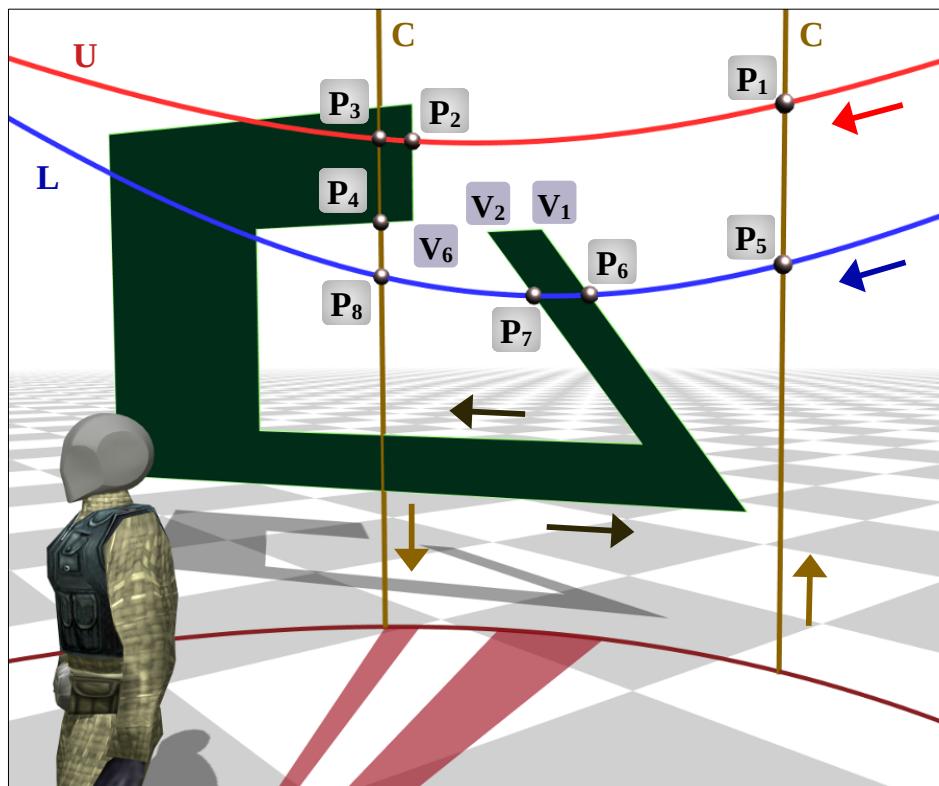
1. Каждый список сортируется по возрастанию параметра вдоль кривой  $t$ .
2. Сначала обрабатываются списки пересечений каждой из 3 кривых:
  1. Обходится список, пересечения пропускаются, пока не будет найдено пересечение кривой внутрь закрывающего обзора интервала, это:
    - (a) Если кривая от цилиндра пересекается с кривой от верхнего конуса: тип OUT и точка пересечения находится внутри многоугольника;
    - (b) Пересечение со стороной многоугольника типа IN;
    - (c) Остальные случаи: тип IN и точка пересечения находится внутри многоугольника.
  2. Для этого и следующего пересечения, интервал из 2 точек добавляется в интервалы, закрывающие видимость;
  3. С пересечения после следующего продолжается выполнение с шага 2.1.
3. Обрабатываются списки пересечений многоугольника:
  1. Обходится список, пока не будет пересечение внутрь объема, ограниченного 3 фигурами, это:
    - (a) Пересечение типа OUT с кривой от верхнего конуса;
    - (b) Пересечение типа IN с другой кривой.
  2. Обходятся стороны многоугольника до следующего пересечения. Каждая сторона частично или полностью добавляется в интервалы.
  3. С пересечения после следующего продолжается выполнение с шага 3.1.

## Пример обработки списков пересечений

Даны списки пересечений, соответствующие изображению справа, упорядоченные по возрастанию  $t$ .

### Пересечения **U**

1	P <sub>1</sub>	IN	<b>C</b>
2	P <sub>2</sub>	IN	сторона
3	P <sub>3</sub>	OUT	<b>C</b>



### Пересечения **L**

1	P <sub>5</sub>	IN	<b>C</b>
2	P <sub>6</sub>	IN	сторона
3	P <sub>7</sub>	OUT	сторона
4	P <sub>8</sub>	OUT	<b>C</b>

### Пересечения **C**

1	P <sub>5</sub>	IN	<b>L</b>
2	P <sub>1</sub>	IN	<b>U</b>
3	P <sub>3</sub>	OUT	<b>U</b>
4	P <sub>4</sub>	OUT	сторона
5	P <sub>8</sub>	OUT	<b>L</b>

### Пересечения сторон многоугольника

1	P <sub>7</sub>	OUT	<b>L</b>
2	P <sub>4</sub>	IN	<b>C</b>
3	P <sub>2</sub>	IN	<b>U</b>
4	P <sub>6</sub>	IN	<b>L</b>

Обработать списки пересечений.

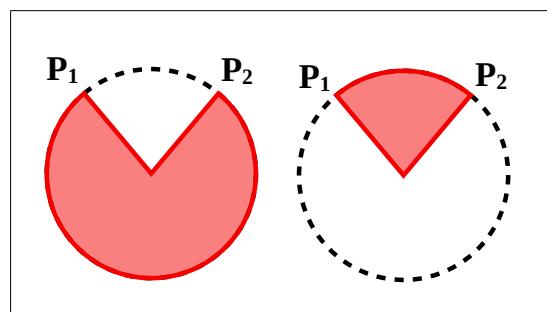
- Начинаем обрабатывать пересечения **U**. Точка P<sub>1</sub> - пересечение IN с другой кривой, точка пересечения не находится внутри многоугольника, пропускаем.
- Точка P<sub>2</sub> - пересечение IN со стороной многоугольника. Добавляем интервал P<sub>2</sub> – P<sub>3</sub>.
- Начинаем пересечения **L**. Точка P<sub>5</sub> - пересечение IN с другой кривой, точка не внутри многоугольника, пропускаем.
- Точка P<sub>6</sub> - пересечение IN со стороной многоугольника. Добавляем интервал P<sub>6</sub> – P<sub>7</sub>.
- Точка P<sub>8</sub> - пересечение OUT, пропускаем.
- Кривая **C**. В данном случае это параллельные прямые. В приложении они получаются только вертикальные (в координатах xyz). Интервалы от них могут быть только нулевой ширины и только на границах других интервалов. Пропускаем.
- Начинаем стороны многоугольника. P<sub>7</sub> - пересечение OUT с **L**, пропускаем.
- P<sub>4</sub> - пересечение IN с **C**. Добавляем интервалы P<sub>4</sub> – V<sub>6</sub> и V<sub>6</sub> – P<sub>2</sub>.
- P<sub>6</sub> - пересечение IN с **L**. Добавляем интервалы P<sub>6</sub> – V<sub>1</sub>, V<sub>1</sub> – V<sub>2</sub> и V<sub>2</sub> – P<sub>7</sub>.

## Добавление интервала

Даны точки начала и конца интервала -  $P_1$  и  $P_2$  (в координатах  $uvn$ ), добавить интервал на окружности.

Точки переводятся в координаты  $xuz$ , используя умножение матрицы  $M$  на вектор (5). Далее точки проецируются на плоскость  $XZ$  и с использованием функции  $\text{atan2}()$  находятся углы к оси  $X$ .

Если две точки принадлежат стороне многоугольника, то ширина интервала на окружности (в радианах) будет не более  $\pi$ , создается короткий интервал.



Точки начала и конца интервала -  $P_1$  и  $P_2$ .  
Длинный и короткий интервалы.

## Пример с длинным интервалом

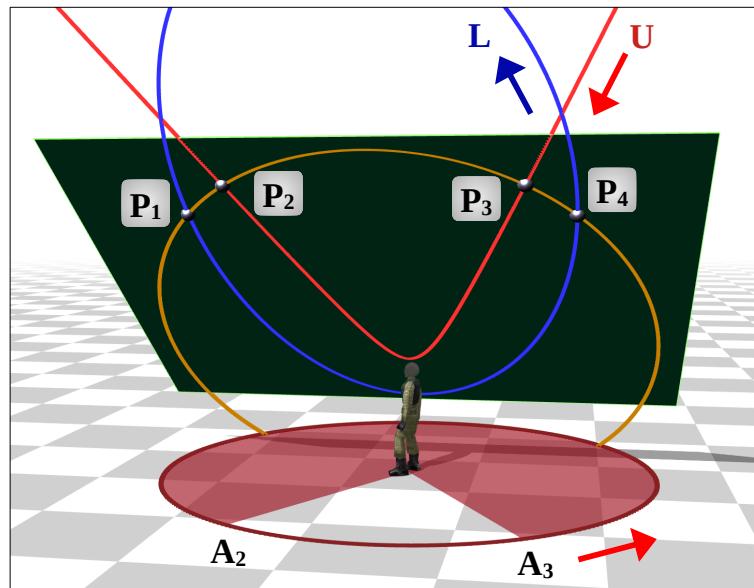
Обработать соответствующий изображению справа список пересечений кривой  $U$ .

1	$P_3$	IN	C
2	$P_2$	OUT	C

Точка  $P_3$  - пересечение IN с кривой, точка пересечения внутри многоугольника.  
Добавляем интервал  $P_3 - P_2$ .

После перевода в  $xuz$  и проекции на плоскость  $XZ$  получаем интервал  $A_3 - A_2$ .

Возникает задача определить – в каком направлении заполнять интервал.



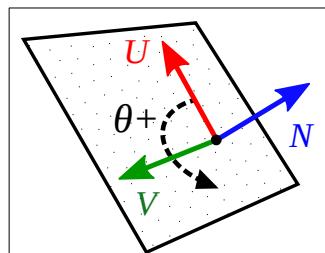
## Начало рассуждений

Во-первых, если плоскость  $P$  расположена вертикально (угол между  $Y$  и  $N$  равен  $90^\circ$ ,  $N_y = 0$ ), то интервалов шириной более  $\pi$  не будет.

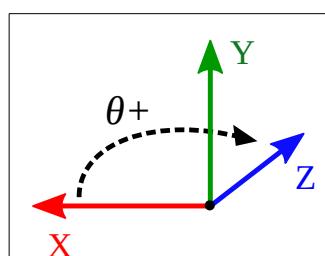
Далее, на плоскости  $P$  точки пересечений упорядочены по возрастанию  $t$  (см. “[Параметр вдоль кривой](#)”), и интервал заполняется в этом же направлении, это направление увеличения угла  $\theta$  между точкой и положительным направлением оси  $U$  ( $U^+$ ), при том что  $V^+$  находится через четверть оборота ( $\theta = \pi / 2$ ). Если смотреть из полупространства, в которое направлен  $N$ , то это направление против часовой стрелки.

На плоскости  $XZ$  увеличение угла  $\theta +$  направлено от  $X^+$  к  $Z^+$ . Если смотреть “сверху” (из полупространства, в которое направлен  $Y$ ), то это направление по часовой стрелке.

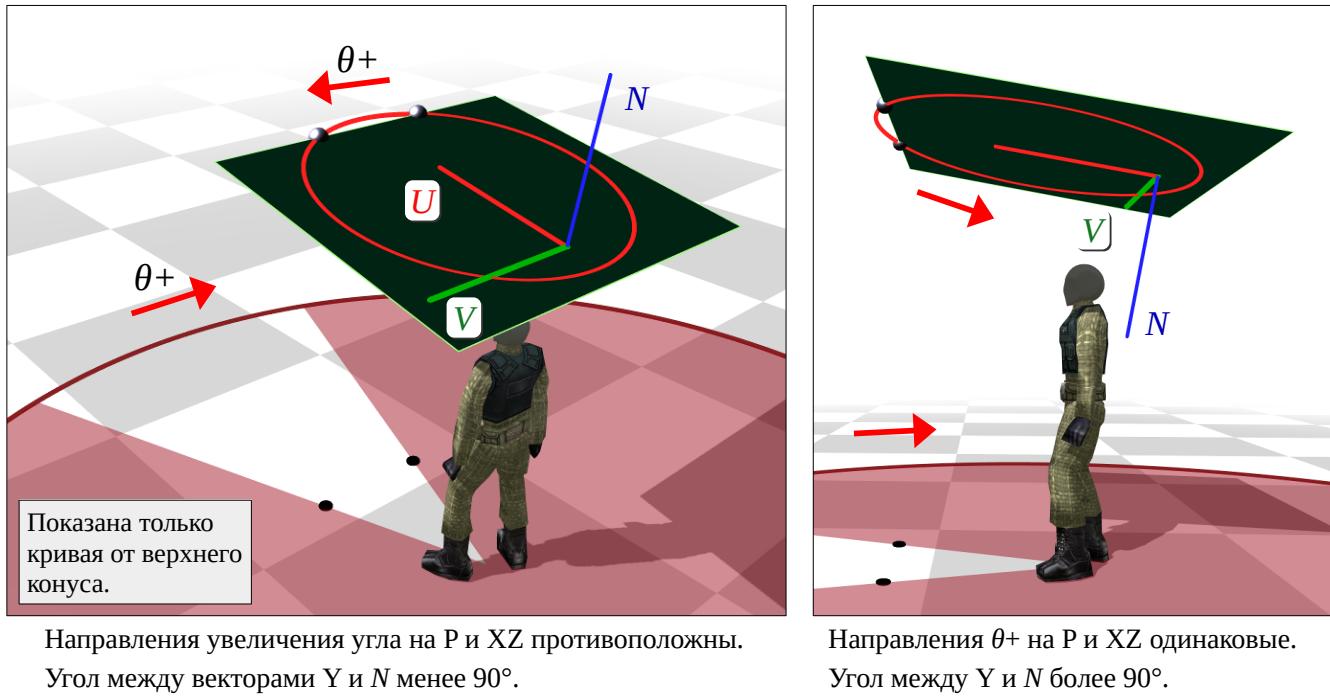
Возникает идея – плоскости  $P$  и  $XZ$  рассмотреть с одного и того же направления, и установить – направления  $\theta +$  одинаковые или противоположные. Будем рассматривать со стороны  $Y^+$ .



На плоскости  $P$ .

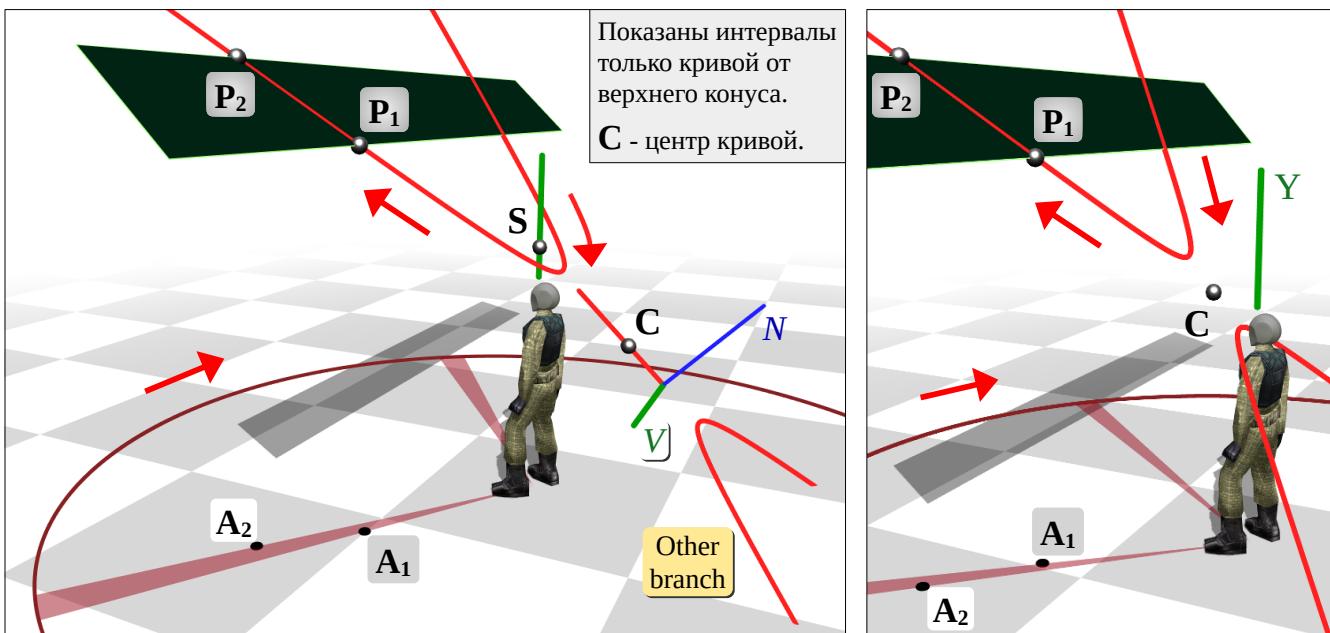


На плоскости  $XZ$ .



## Точка S

Необходимо принять во внимание – относительно какой точки вычисляется угол. На XZ это начало координат, на Р - центр кривой. Направление  $\theta+$  на Р необходимо определить из такой точки на Р, которая после перевода в xyz и проекции на XZ оказывается в начале координат. Обозначим эту точку S.



На изображениях добавляется интервал  $P_1 - P_2$ . После проекции на XZ получается интервал  $A_1 - A_2$ .

## Нахождение точки S и ее свойства

После проекции на XZ, в начале координат ( $x = 0, z = 0$ ) оказываются точки, лежащие на оси Y. Также  $S \in P$ . Выше в “начале рассуждений” был отдельно рассмотрен случай, если P и ось Y параллельны или совпадают. Соответственно они пересекаются в одной точке. Ось  $U \in P$  и ось Y также лежат в одной плоскости, пересекаются в одной точке, это и есть искомая точка. Точка нужна в координатах  $uvn$ .

Возьмем уравнения (6). S находится на оси Y, следовательно  $x = 0, z = 0$ .  $S \in U \Rightarrow v = 0, n = 0$ .

После алгебраических манипуляций получаем  $u$ -координату точки:

$$S_u = \begin{cases} \frac{c N_x}{U_x}, & \text{если } |U_x| > |U_z|, \\ \frac{c N_z}{U_z}, & \text{если иначе.} \end{cases} \quad (11)$$

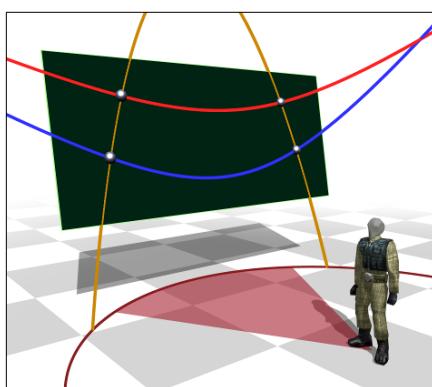
### Свойства точки

- S не лежит на кривой. Кривая пересекается с осью Y, только когда плоскость проходит через вершину конуса, что исключается на этапе “Подготовка к преобразованию координат”.
- Если кривая – эллипс, то S всегда находится внутри кривой, потому что ось Y по построению всегда находится внутри эллипса.

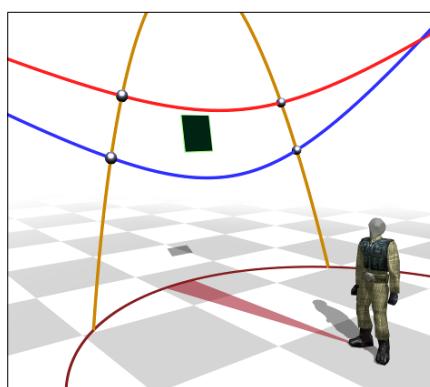
Чтобы определить – находятся ли точка S и центр кривой по одну сторону от кривой, решается задача “Точка внутри кривой”.

## Отсутствие интервалов

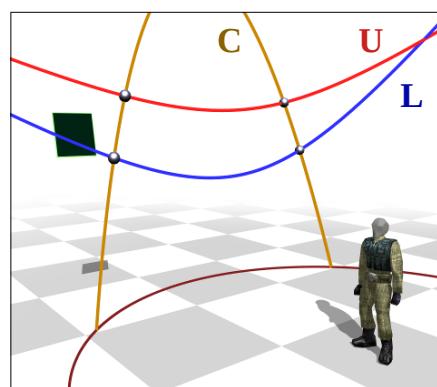
Итак, были пересечения и их списки были обработаны. Если пересечений сторон многоугольника не было (соответственно не было добавлений интервалов при обработке пересечений сторон), то возможно одно из трех:



Есть интервалы от кривых.



Не было интервалов. После проверки добавлены интервалы от сторон многоугольника.



Нет интервалов. После проверки интервалы не добавлены.

В случае отсутствия интервалов от кривых, проверяется – влияет ли многоугольник на видимость. Для этого берется любая вершина, и если она находится снаружи кривой от верхнего конуса и при этом внутри кривых от нижнего конуса и цилиндра, то добавляются интервалы от сторон многоугольника.

Обработка грани (в соответствии с “Обзор алгоритма”) завершена.

## Приложение А. Работа с квадратичными кривыми

### Центр кривой

Формула центра кривой:

$$(x, y) = (2CD - BE, 2AE - BD) / (B^2 - 4AC). \quad (12)$$

Если дискриминант  $B^2 - 4AC$  равен нулю (то есть такая кривая, что если ее повернуть так, чтобы она не была повернута относительно осей координат, это  $B = 0$ , у нее оказывается только один квадратичный коэффициент  $A$  или  $C$  ненулевой), эта формула непригодна. Это парабола, параллельные прямые.

В этой работе параллельные прямые появляются только вида  $Cy^2 + F = 0$  (горизонтальные, симметричные относительно оси  $X$ ). В качестве их центра используется  $(0, 0)$ .

### Ветви гиперболы

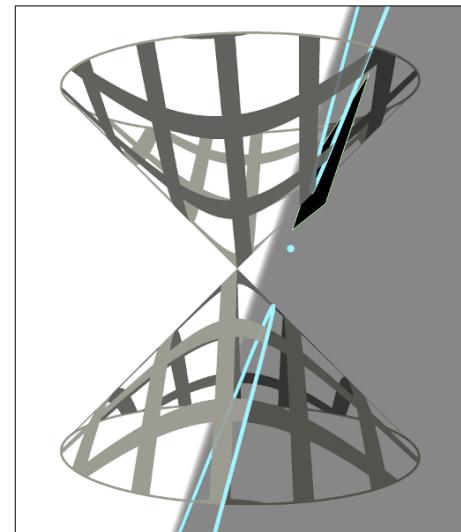
У гиперболы 2 ветви, так же как у конуса 2 половины.

В программный объект типа кривая добавлено свойство `halfSgn`, определяющее, какая ветвь используется.

Задача - определить, принадлежит ли точка выбранной ветви.

В этой работе преобразование координат проведено так, что ось кривой совпадает с осью  $U$ . В связи с тем, что угол между векторами  $U$  и  $Y$  не более  $90^\circ$  ( $U \cdot Y \geq 0$ ), точки, полученные из верхней половины конуса, будут иметь  $u$ -координату больше  $u$ -координаты центра гиперболы.

Если оси  $U$  и  $Y$  перпендикулярны ( $U \cdot Y = 0$ ), то гиперболы не получится.



Ветви и центр гиперболы.

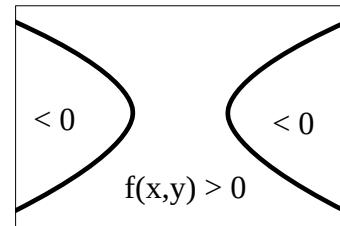
### Точка внутри кривой

Для замкнутой кривой, такой как эллипс, понятие “внутри” интуитивно понятно. Параллельные прямые могут быть рассмотрены как эллипс с бесконечной одной из осей.

Определим, что точка находится внутри кривой в случае, если значения функции  $Ax^2 + Bxy + Cy^2 + Dx + Ey + F$  в точке и в центре кривой имеют разные знаки в случае гиперболы и одинаковые знаки в случае эллипса и параллельных прямых. Это применимо для функций, у которых значение функции в центре не равно нулю (центр не находится на кривой).

При таком определении, точка находится внутри кривой тогда и только тогда, когда точка находится внутри соответствующего 3D объекта – конуса или цилиндра.

В этой работе получаются кривые с коэффициентом  $C = 1$ . Из этого следует, что у эллипсов и параллельных прямых значение функции в центре будет отрицательным. Гиперболы получаются только горизонтальные (по построению), у них значение функции в центре будет положительным.



$$f(x, y) = -0.3x^2 + y^2 + 0.3$$

В приложении гиперболы получаются только горизонтальные.

Для определения принадлежности точки многоугольнику используется алгоритм *even-odd rule*.

В приложении ребра многогранников и соответственно стороны многоугольников не пересекающиеся, что необходимо для корректной работы этого алгоритма.

## Параметр вдоль кривой

При обработке точек на кривых возникает следующая задача. Допустим, даны 3 точки на кривой:  $P_1$ ,  $P_2$  и  $P_3$ . Если двигаться в заданном направлении от  $P_1$  (еще нужен способ задания направления), какая точка встретится первой -  $P_2$  или  $P_3$ . Кривые, состоящие из нескольких несоединенных множеств считаются как соединенные в бесконечности.

В рамках данной работы используется параметр  $t \in [-\pi, \pi]$  в точке на кривой, равный углу между  $X+$  (положительным направлением оси  $X$ ) и лучем из центра кривой через точку. Для этого должно быть взаимно-однозначное соответствие между углом и точкой. Используемые эллипс, гипербола и несовпадающие параллельные прямые этому условию удовлетворяют.

## Нахождение касательного вектора в точке в направлении увеличения параметра

Вычисляется градиент в точке:

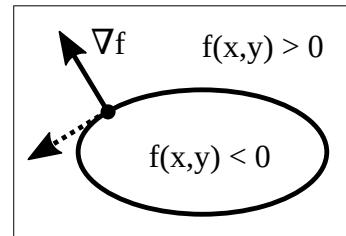
$$\nabla f = \left( \frac{\partial x}{\partial f}, \frac{\partial y}{\partial f} \right) = (2Ax + By + D, 2Cy + Bx + E). \quad (13)$$

Одна кривая может быть выражена бесконечным количеством наборов коэффициентов  $A - F$ , отличающихся мультипликативной константой (например,  $x^2 + y^2 - 1 = 0$  и  $-2x^2 - 2y^2 + 2 = 0$ ).

У функций, отличающихся мультипликативной константой, ряд свойств одинаковы (множества нулевых значений, центр), но некоторые другие свойства отличаются, такие как значение функции в точках, в которых значение не равно нулю, и соответственно направления градиента могут быть противоположными.

Градиент направлен в сторону увеличения значения функции. Если функция в центре больше нуля, градиент умножается на  $-1$ , в результате вектор будет направлен в сторону, противоположную центру кривой. Далее вектор поворачивается на  $90^\circ$  против часовой стрелки, для этого берется отрицательный перпендикуляр:  $-(x, y)^\perp = -(y, -x) = (-y, x)$ . Получается касательный вектор в направлении увеличения параметра  $t$ .

Это верно для создаваемой на плоскости  $P$  системы координат  $uvn$ .



Функция  $2x^2 + y^2 - 1$ , градиент и касательный вектор в точке.

## Параметр для многоугольников

Для многоугольников,  $t$  в точке на стороне многоугольника вычисляется как сумма целочисленного номера стороны и параметра вдоль стороны (где 0 – начальная вершина и 1 – конечная вершина).

## Типы пересечения IN и OUT

Определим, что пересечение между `curve1` и `curve2` относится к типу IN, если точки на `curve1`, если смотреть в направлении увеличения параметра вдоль кривой, непосредственно перед пересечением находятся снаружи `curve2`, а после пересечения – внутри `curve2`.

## Приложение В. Пересечение отрезка и квадратичной кривой

Отрезок прямой задается концами  $P_1$  и  $P_2$ . Он направлен от  $P_1$  к  $P_2$ . Переведем в параметрическую форму:

$$\vec{V} = P_2 - P_1 ;$$

$$\begin{cases} x = P_{1x} + V_x t \\ y = P_{1y} + V_y t . \end{cases} \quad (14)$$

Квадратичная кривая задана следующим уравнением:

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 . \quad (15)$$

Из уравнений (14) возьмем  $x, y$ , подставим в (15):

$$\begin{aligned} A(P_{1x} + V_x t)^2 + B(P_{1x} + V_x t)(P_{1y} + V_y t) + C(P_{1y} + V_y t)^2 \\ + D(P_{1x} + V_x t) + E(P_{1y} + V_y t) + F = 0 . \end{aligned} \quad (16)$$

В результате проведения алгебраических преобразований с целью получить в левой части многочлен, сгруппированный по степеням  $t$ , получается алгебраическое уравнение 2 степени:

$$\begin{aligned} at^2 + bt + c = 0 , \quad \text{где} \\ a = AV_x^2 + BV_x V_y + CV_y^2 , \\ b = 2(AV_x P_{1x} + CV_y P_{1y}) + B(V_x P_{1y} + V_y P_{1x}) + DV_x + EV_y , \\ c = AP_{1x}^2 + BP_{1x} P_{1y} + CP_{1y}^2 + DP_{1x} + EP_{1y} + F . \end{aligned} \quad (17)$$

Если у уравнения есть корни  $t_{1,2}$ , расположим их в порядке возрастания  $t$ , это 1-е и 2-е пересечения в направлении прямой. Координаты точек пересечения:

$$x_n = P_{1x} + t_n V_x , \quad y_n = P_{1y} + t_n V_y , \quad n = 1, 2 . \quad (18)$$

Если  $t_n \notin [0, 1]$ , то пересечение находится за пределами границ отрезка.

Если обрабатывается последовательность сторон многоугольника, то пересечение точно в вершине не должно быть обработано дважды. В таком случае пропускается  $t_n = 1$ .

Для гиперболы происходит проверка нахождения точки в выбранной ветви (см. “[Ветви гиперболы](#)”).

### Определение типа пересечения IN или OUT

См. “[Типы пересечения IN и OUT](#)”

Для эллипса и параллельных прямых 1-е пересечение всегда IN и 2-е всегда OUT.

В остальных случаях вычисляется знак функции (15) в точке  $P_1 + (t_1 - 1)V$  (перед 1-м пересечением), и сравнивается со знаком функции в центре кривой. См. “[Точка внутри кривой](#)”.

У 2-го пересечения тип всегда противоположный 1-му.

### Обработка пересечения с мультиплексивностью 2 ( $t_1=t_2$ )

Вычисляются знаки функции (15) в точках  $P_1 + (t_1 \mp 1)V$  (до и после пересечения). Примеры случаев с одинаковыми знаками: пересечение по касательной, пересечение с совпадающими параллельными прямыми (напр.  $x^2 = 0$ ), пересечение с точкой (напр.  $x^2 + 2x + y^2 + 2y + 2 = 0$ ). В приложении возможно только пересечение по касательной. В результате будет интервал нулевой ширины. Это пропускается.

Если знаки функции (15) в точках до и после пересечения разные, то это считается одним пересечением. Пример – гипербола и прямая, параллельная асимптоте.

## Приложение С. Пересечение двух квадратичных кривых

По теореме Безу, у двух кривых на плоскости степеней  $n$  и  $m$  не более  $nm$  точек пересечения.

Следовательно, у двух кривых 2 степени не более 4 точек пересечения.

Уравнения квадратичных кривых:

$$\begin{cases} A_1x^2 + B_1xy + C_1y^2 + D_1x + E_1y + F_1 = 0 \\ A_2x^2 + B_2xy + C_2y^2 + D_2x + E_2y + F_2 = 0 \end{cases} \quad (19)$$

В этой работе преобразование координат проведено так, что коэффициенты  $B, E$  равны нулю (ось кривой совпадает с осью X), а также  $C = 1$ . Пересечение таких кривых намного проще, чем в общем случае.

### Случай $B, E = 0, C = 1$

Из уравнений (19) убраны слагаемые  $B, E$ , подставлено  $C = 1$ :

$$\begin{cases} A_1x^2 + y^2 + D_1x + F_1 = 0 \\ A_2x^2 + y^2 + D_2x + F_2 = 0 \end{cases} \quad (20)$$

Скомбинируем уравнения (20), чтобы убрать  $y^2$ . Для этого из 1-го уравнения вычтем 2-е уравнение:

$$x^2(A_1 - A_2) + x(D_1 - D_2) + F_1 - F_2 = 0. \quad (21)$$

Получено алгебраическое уравнение 2 степени.

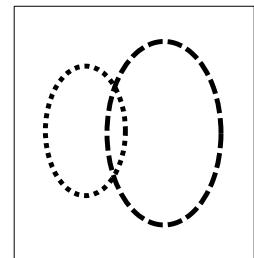
### Если коэффициент при $x^2$ равен нулю

Рассмотрим уравнение (10). Однаковые коэффициенты  $A$  получатся в случае, если  $U_y = 0$  (горизонтальная плоскость) или у конусов одинаковые  $b$  ( $h_{\text{delta}} = h_{\text{object}}$ ).

Если коэффициент при первой степени переменной  $x$  в уравнении (21) не ноль, то это линейное уравнение (см. изображение справа). В тех случаях, в которых получатся одинаковые  $A$ , в соответствии с (10) получатся одинаковые  $D$ .

Далее при  $F_1 \neq F_2$  уравнение не имеет решений (концентрические кривые).

При  $F_1 = F_2$  кривые совпадают, для данного приложения в этом случае будет логичен (для последующей обработки) результат – отсутствие пересечений.



Отношения квадратичных коэффициентов равны (1:2), остальные не равны.

### Вычисление $y$ – координат точек

Для каждого  $x_{1,2}$  найдем соответствующие  $y$ -координаты точек. Выразим любое из уравнений (20) как функцию  $y$ :

$$\begin{aligned} Ax^2 + Dx + F &= -y^2; \\ y^2 &= -(Ax^2 + Dx + F); \\ y_{1,2} &= \pm\sqrt{-(Ax^2 + Dx + F)}. \end{aligned} \quad (22)$$

Итого получено 0, 2 или 4 точки, расположенные попарно симметрично относительно оси X.

Для гипербол происходит проверка нахождения точки в выбранной ветви (см. “[Ветви гиперболы](#)”).

## Определение типа пересечения IN или OUT

В точке пересечения вычисляются касательные векторы в направлении увеличения параметра, обозначим их  $V_1$  и  $V_2$ . Вычисляется знак перпендикулярного скалярного произведения (скалярного произведения перпендикуляра к  $V_1$  и вектора  $V_2$ ):  $s = \text{sgn}(V_1 \perp V_2) = \text{sgn}(V_1^\perp \cdot V_2)$ .

По таблицам определяется тип IN или OUT для `curve1` (`type1`) и для `curve2` (`type2`). В этой работе `curve1` всегда происходит от цилиндра и не может быть гиперболой.

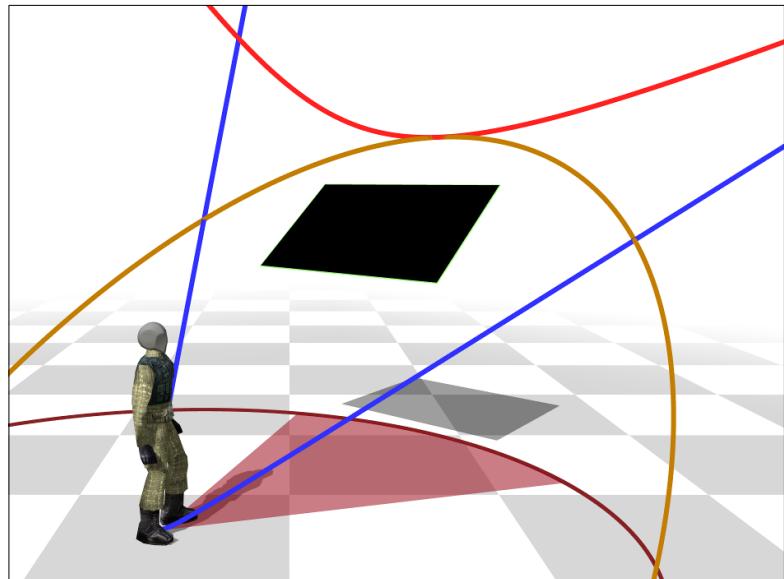
Таблица для $s = 1$		curve2	
		гипербола	другая
<code>curve1</code>	гипербола	-	-
	другая	<code>type1: OUT, type2: OUT</code>	<code>type1: IN, type2: OUT</code>

Таблица для $s = -1$		curve2	
		гипербола	другая
<code>curve1</code>	гипербола	-	-
	другая	<code>type1: IN, type2: IN</code>	<code>type1: OUT, type2: IN</code>

## Особые случаи

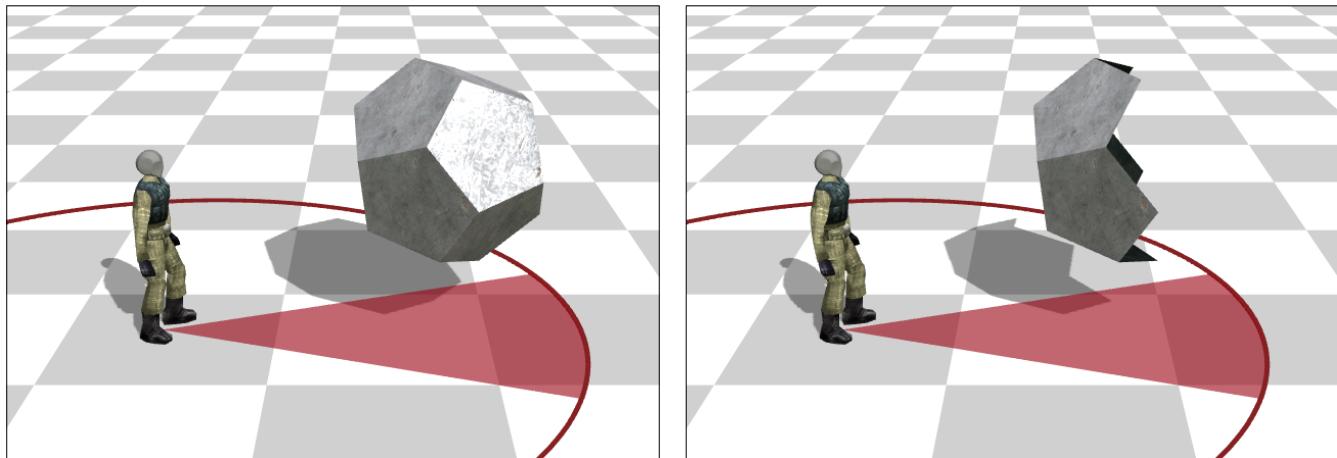
На изображении справа касающиеся кривые.

- Точки пересечения не показаны. Они находятся на небольшом расстоянии, могут совпадать;
- Перпендикулярное скалярное произведение в точках пересечения близко к нулю или равно нулю (касательные векторы совпадают или противоположны по направлению).

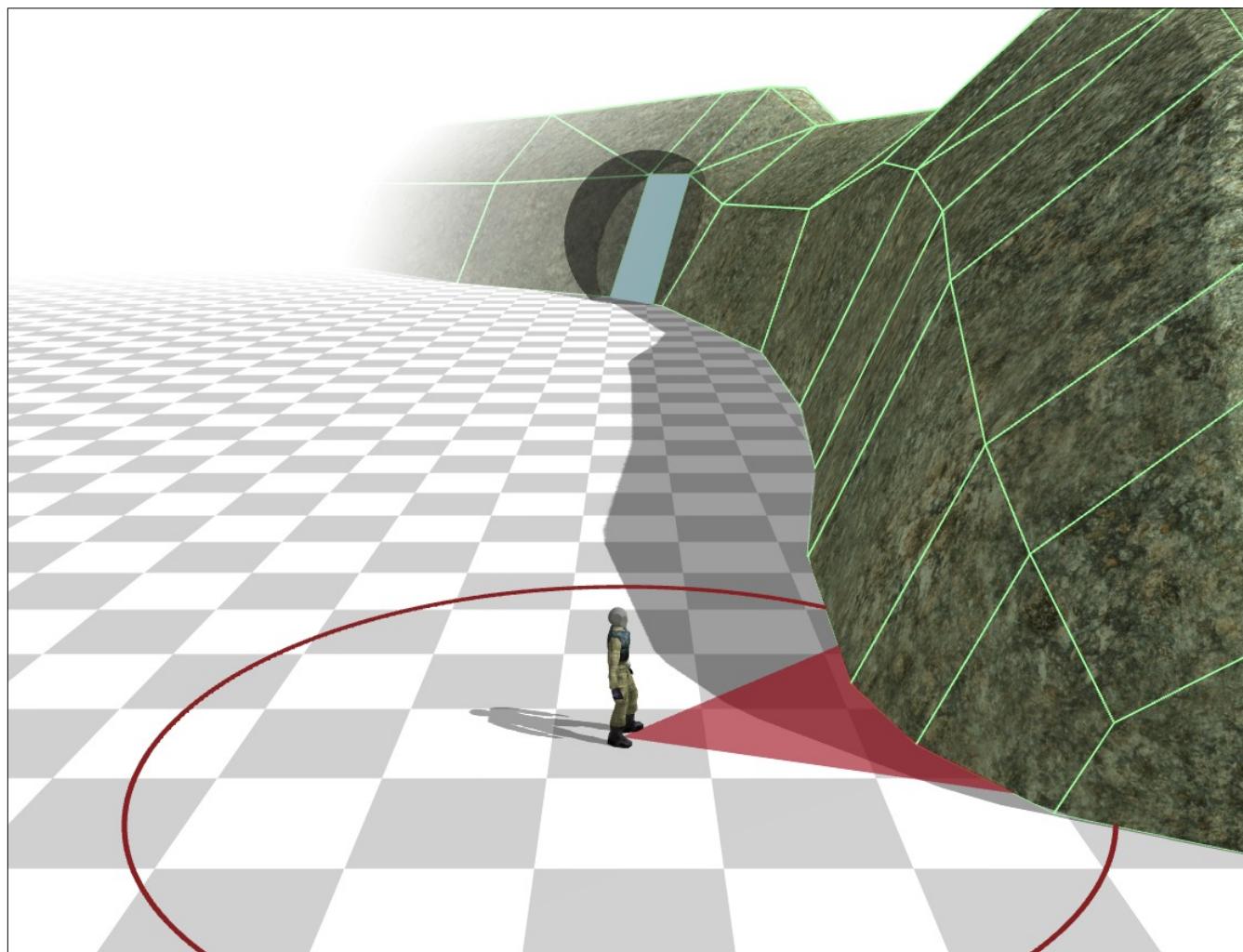


## Приложение D. Улучшение производительности

Пропускание грани, если объект не в фронтальном полупространстве (хотя бы частично)

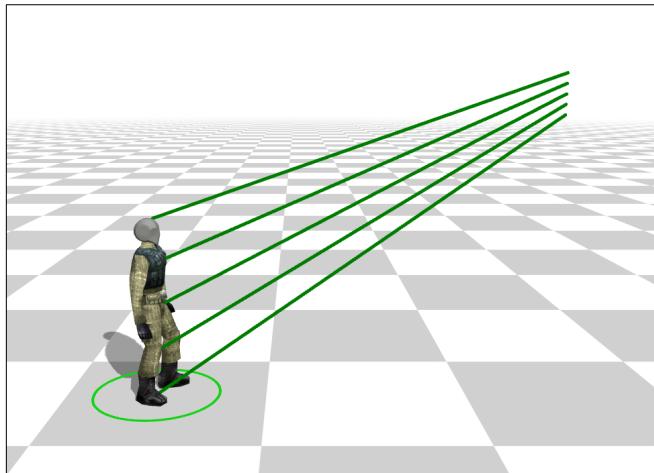


Использование ограничивающей сферы грани

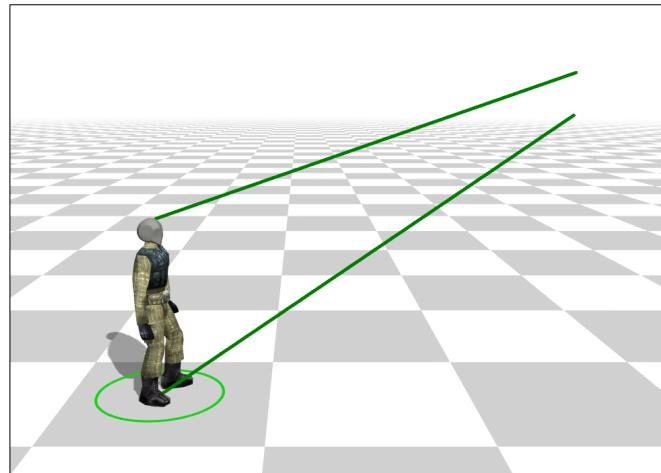


## Приложение E. Сопровождающие алгоритмы

### Проверка условий расположения камеры



Вариант 1



Вариант 2

После того, как камера занимает положение в пространстве, соответствующее поставленным условиям, в случае, если включен режим “следовать за объектом”, регулярно происходит проверка – сохраняются ли условия расположения камеры, в том числе отсутствие препятствий для видимости.

Потребуется, как минимум:

- Преобразование координат в трехмерном пространстве;
- Пересечение отрезка прямой и плоскости;
- Работа с многоугольниками на плоскости.