

1. Введение

В настоящей статье мы рассмотрим вопросы моделирования траекторий между преследующим и преследуемым объектами. Моделируемая траектория должна удовлетворять некоторым условиям. Одно из условий – это то, что минимальный радиус кривизны траектории не должен меньше некоторого порогового значения. Еще одно условие в рассматриваемой задаче – это то, что моделируемая траектория выходит под определенным углом из точки нахождения преследующего объекта и приходит в точку нахождения преследуемого объекта также под заданным углом.

2. Постановка задачи

Рассмотрим случай на горизонтальной плоскости проекций. Пусть преследуемый объект движется по траектории $R_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \end{bmatrix}$, а преследующий объект движется по траектории $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ (Рис. 1).

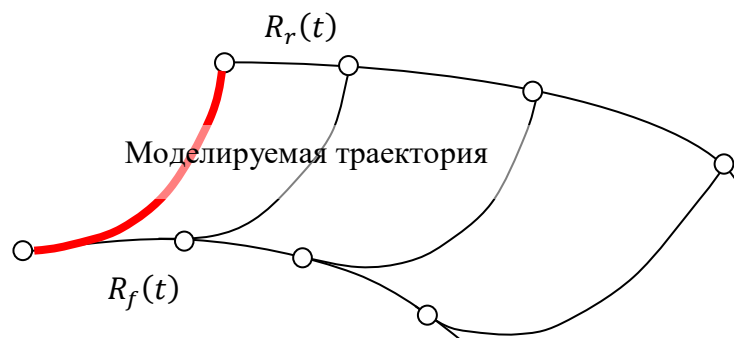


Рис. 1 Моделируемая траектория

В нашей задаче, которая решается в настоящей статье, моделирование траектории в каждый момент времени t . Моделируемая траектория может выходить из точки $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ под любым углом. Но в программном коде, написанном по материалам статьи, мы приняли угол между

касательной к моделируемой траектории и касательной к траектории $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ равным 0, что не имеет принципиального значения в плане моделирования алгоритма. Также угол, между касательной к моделируемой траектории и касательной к траектории $R_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \end{bmatrix}$, в листинге программы принят равным $\frac{\pi}{2}$. При этом моделируемая траектория имеет ограничения по своей кривизне. То есть радиус кривизны моделируемой кривой не может быть меньше некоторой пороговой величины.

3. Моделирование окружности у преследующего объекта

Мы уже говорили, что преследующий объект движется по траектории $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$. Выполнение условия того, что угол между касательными к моделируемой траектории и к траектории $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ равен 0, и условие ограничения по кривизне, приводят построению окружности, изображенной на Рис. 2.

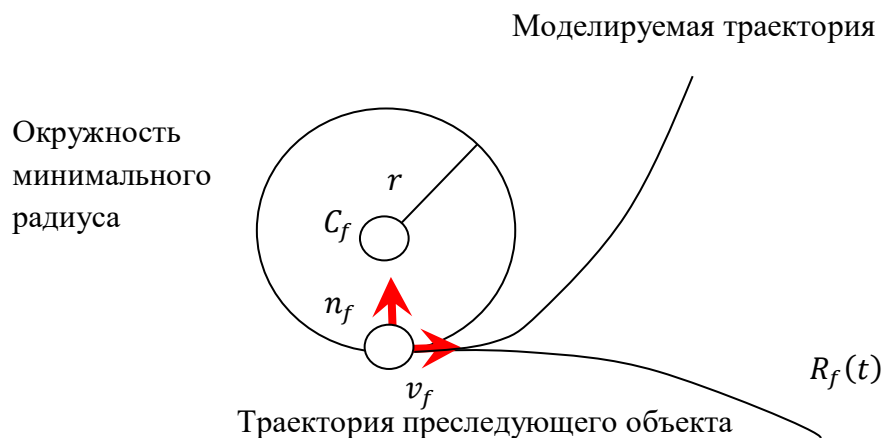


Рис. 2 Окружность к траектории преследующего объекта

На Рис. 2 показано, что вдоль траектории преследующего объекта $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ формируется локальный динамический базис (v_f, n_f) , где v_f

- единичный вектор, сонаправленный с вектором скорости $V_f(t) = \frac{dR_f(t)}{dt}$ преследуемого объекта. Вектор n_f – есть вектор нормали к траектории $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$.

Отсюда следует, центр моделируемой окружности $C_f(t)$ формируется следующим образом: $C_f(t) = R_f(t) + r \cdot n_f(t)$, где r - это минимальный пороговый радиус траектории преследуемого объекта.

Центр окружности может формироваться иначе (Рис. 3): $C'_f(t) = R_f(t) + r \cdot n'_f(t)$. В таком случае (как на рис. 3) моделируемая траектория выходит под углом α к траектории преследуемого объекта $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$.

В коде программы, написанной по материалам статьи угол между траекториями равен 0.

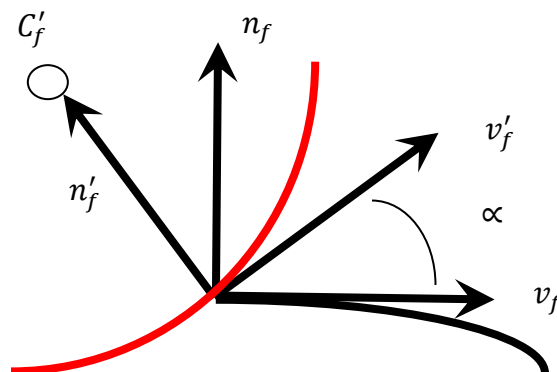


Рис. 3 Формирование окружности на траектории преследуемого объекта

4. Моделирование окружности у преследуемого объекта

Рассмотрим формирование окружности при входе моделируемой траектории в траекторию преследуемого объекта (Рис. 4).

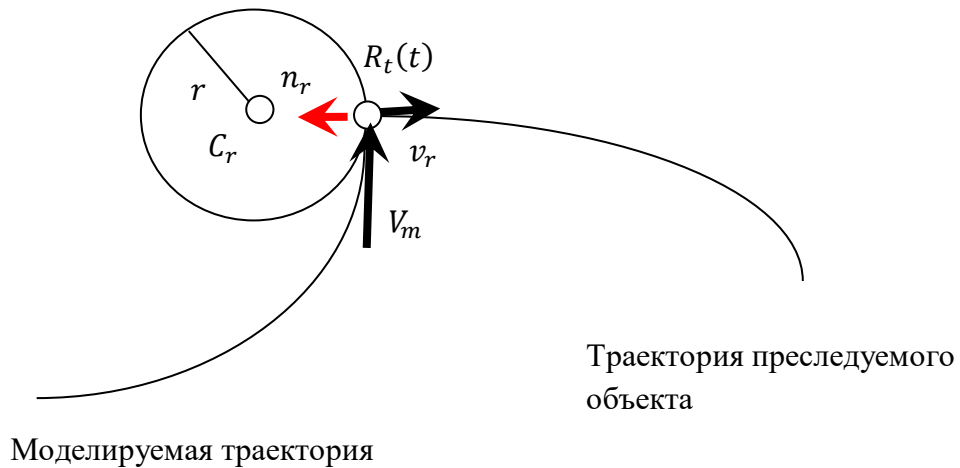


Рис. 4 Окружность к траектории преследуемого объекта

Здесь центр окружности формируется так: $C_r(t) = R_r(t) + r \cdot n_r(t)$, где $R_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \end{bmatrix}$ - это точка траектории преследуемого объекта, а r - минимальное значение радиуса моделируемой траектории. Здесь направление вектора $n_r(t)$ полностью определяется направлением вектора $V_m(t)$ скорости моделируемой траектории в момент пересечения с траекторией преследуемого объекта $R_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \end{bmatrix}$. Векторы $n_r(t)$ и $V_m(t)$ должны быть перпендикулярны. В листинге программы по материалам статьи используется перпендикулярность векторов $V_m(t)$ и v_r .

5. Линии сопряжения между двумя окружностями

Рассмотрим вопрос о сопряжении между двумя окружностями одного радиуса передвигающимися по траекториям $R_f(t) = \begin{bmatrix} x_f(t) \\ y_f(t) \end{bmatrix}$ и $R_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \end{bmatrix}$ (Рис. 5).

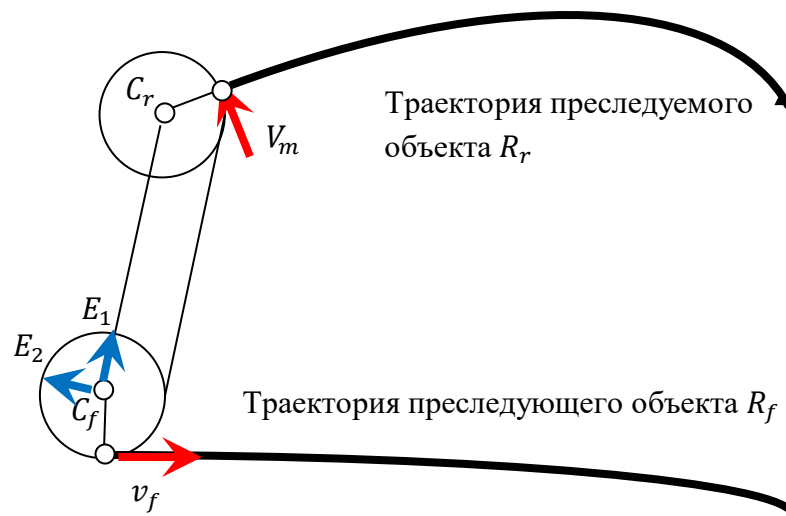


Рис. 5 Линия сопряжения в моделируемой траектории

Исходя из данных, представленных на рис. 5, к построению общей касательной между окружностями (C_f, r) и (C_r, r) . Если мы перейдем в систему координат, образованную точкой $C_f(t)$ и векторами (Рис. 6):

$$E_1(t) = \frac{C_r(t) - C_f(t)}{|C_r(t) - C_f(t)|}, E_2(t) = \begin{bmatrix} -E_{1y}(t) \\ E_{1x}(t) \end{bmatrix}.$$

То задача построения линий сопряжения между двумя окружностями преобразуется к виду, изображенному на Рис. 6, где $L = |C_r - C_f|$. Как видно, на рис. 6 существует четыре варианта линии сопряжения. Выбор варианта сопряжения оставим разработчику. Кроме того, Существуют варианты расположения окружностей, которые изображены на рис. 7. В листинге программы, написанной по материалам статьи, мы использовали один вариант взаимного расположения окружностей и построили для демонстрации четыре линии сопряжения.

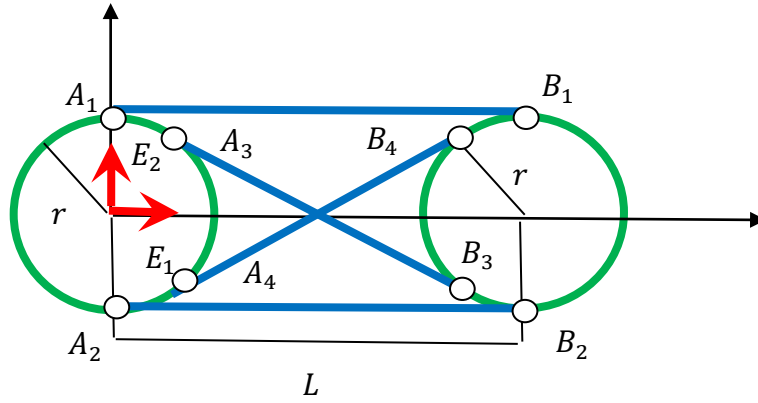


Рис. 6 Сопряжение двух окружностей

Для точек $A_i, B_i, i = 1..4$ на рис. 6 существуют решения:

$$A_1 = \begin{bmatrix} 0 \\ r \end{bmatrix}, A_2 = \begin{bmatrix} 0 \\ -r \end{bmatrix}, A_3 = \begin{bmatrix} 2\frac{r^2}{L} \\ r\sqrt{(L-2r) \cdot (L+2r)} \end{bmatrix}, A_4 =$$

$$\begin{bmatrix} 2\frac{r^2}{L} \\ -\frac{r}{L}\sqrt{(L-2r) \cdot (L+2r)} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} L \\ r \end{bmatrix}, B_2 = \begin{bmatrix} L \\ -r \end{bmatrix}, B_3 = \begin{bmatrix} \frac{L^2-2r^2}{L} \\ -\frac{r}{L}\sqrt{(L-2r) \cdot (L+2r)} \end{bmatrix}, B_4 =$$

$$\begin{bmatrix} \frac{L^2-2r^2}{L} \\ \frac{r}{L}\sqrt{(L-2r) \cdot (L+2r)} \end{bmatrix}.$$

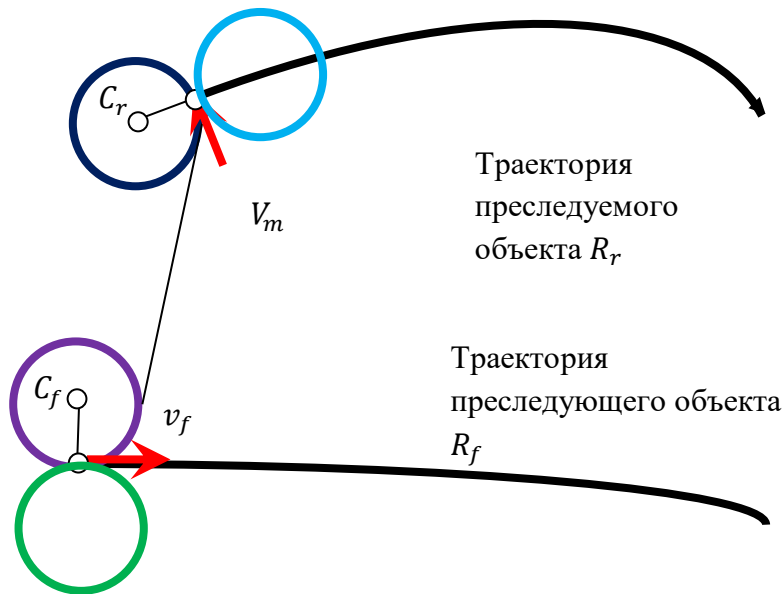


Рис. 7 Варианты расположения окружностей

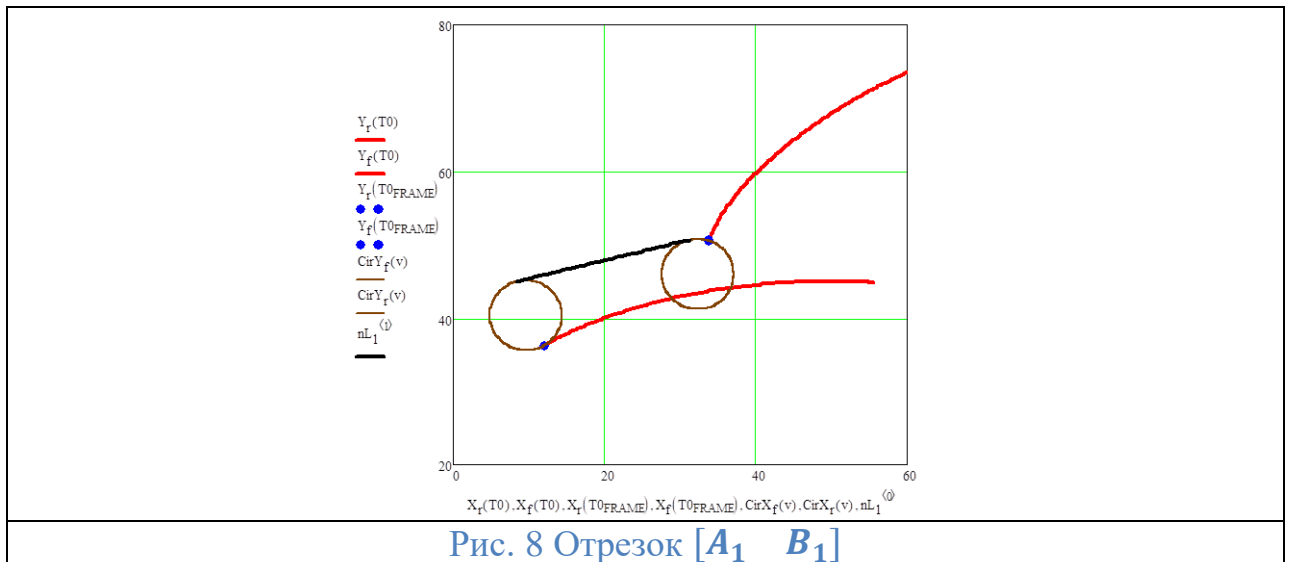
Но эти решения найдены для локальной системы координат (C_f, E_1, E_1) (Рис. 5), чтобы перевести точку P'_i , принадлежащую отрезку $[A_i B_i]$, в мировую систему координат (O, H_1, H_2) необходимо выполнить преобразование по формуле:

$$P_i = \begin{bmatrix} P'_i \cdot h_1 \\ P'_i \cdot h_2 \end{bmatrix} + C_f, \text{ где } h_1, h_2 - \text{это векторы } H_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ и } H_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

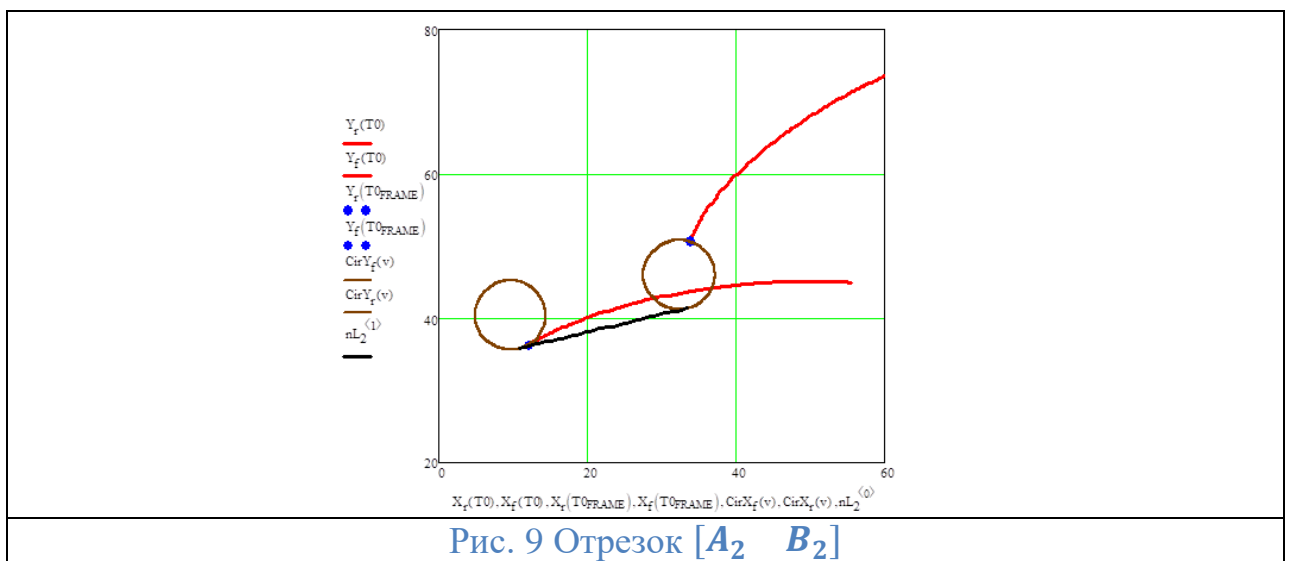
разложенные по векторам базиса (C_f, E_1, E_1) , То есть $h_1 = \begin{bmatrix} H_1 \cdot E_1 \\ H_1 \cdot E_2 \end{bmatrix}$ и $h_2 = \begin{bmatrix} H_2 \cdot E_1 \\ H_2 \cdot E_2 \end{bmatrix}$.

6. Результаты моделирования траекторий

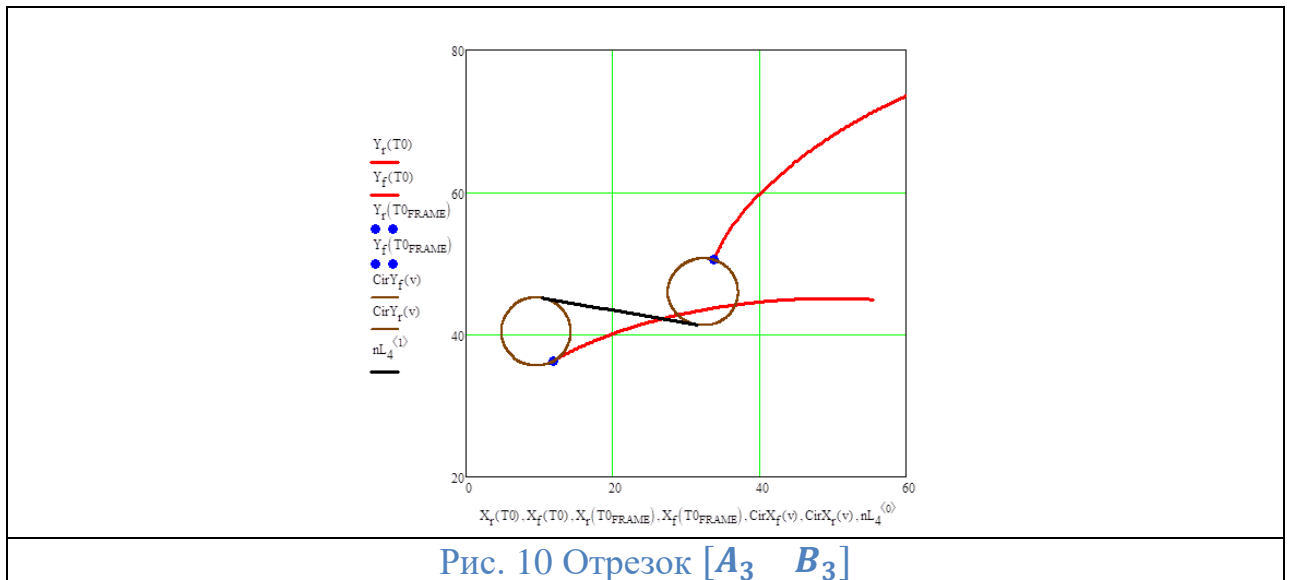
По результатам, изложенным в данной статье, был написан программный код в системе компьютерной математики MathCAD.



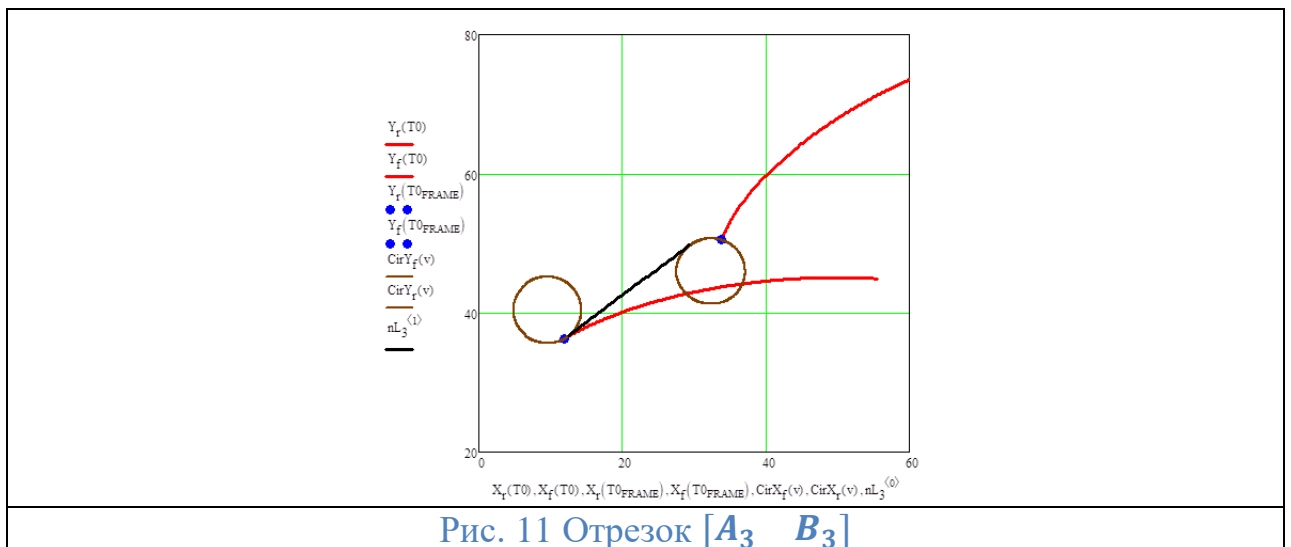
На рис. 8 изображен отрезок соединяющий точки A_1 и B_1 . Рис. 8 получен в результате выполнения программы и дополнен ссылкой на анимированное изображение. Тут следует отметить, что моделируемая траектория перейдет в прямую линию, если в момент отрыва от траектории преследующего объекта точка совершит движение по часовой стрелке, то есть в обратном направлении.



На рис. 9 изображен отрезок, соединяющий точки A_2 и B_2 , он также дополнен ссылкой на анимированное изображение. Здесь же ситуация такая, для перехода на прямолинейный отрезок, точка после отрыва от траектории преследующего объекта должна совершить почти круг против часовой стрелки.



На рис. 10 мы видим, что точка моделируемой траектории сходит с траектории преследующего объекта по часовой стрелке и входит в траекторию преследуемого под прямым углом, но уже против часовой стрелки. Рис. 10 также дополнен ссылкой на анимированное изображение.



На рис. 11 мы видим, что точка моделируемой траектории отрывается от траектории преследующего объекта, переходит на прямолинейный участок и входит в траекторию преследуемого по часовой стрелке. На первом кадре не заметно, но при просмотре полной анимации, видно, что отрыв от траектории преследующего объекта происходит против часовой стрелки. Рис. 11 дополнен ссылкой на анимированное изображение.