

НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ФАКУЛЬТЕТ СИСТЕМ УПРАВЛЕНИЯ И РОБОТОТЕХНИКИ

Отчет по лабораторной работе №1
Алгоритмы планирования траекторий
Вариант 1

Выполнили студенты

Мовчан Игорь Евгеньевич
Гридусов Денис Дмитриевич
Краснов Александр Юрьевич

Преподаватель

Санкт-Петербург
2025

Содержание

1	Бинарная карта и алгоритм A^*	2
2	C^0-гладкая траектория	4
3	C^1-гладкая траектория	6
4	C^2-гладкая траектория	8
5	Сглаживание с помощью B-сплайна	13
6	Выводы	15

1 Бинарная карта и алгоритм A^*

Сформируем бинарную карту размером 10 на 10 ячеек, содержащую не менее трети недоступных к посещению клеток, обозначенных серым цветом на рисунке ниже. В качестве начальной ячейки выберем верхний левый угол с координатами $(x, y) = (1, 1)$, в качестве конечной - нижний правый угол с координатами $(x, y) = (10, 10)$. Обозначим их соответственно зеленым и красным цветом.

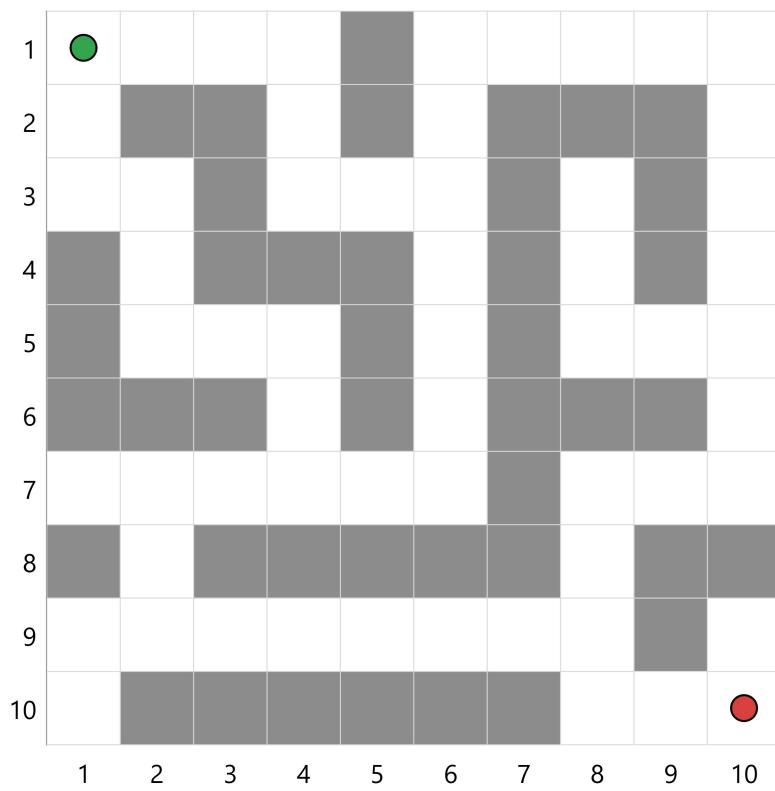


Рис. 1: Сформированная карта движения

Применим алгоритм A^* к этой карте для нахождения оптимального пути от начальной ячейки до конечной, используя функцию

$$f(n) = g(n) + h(n)$$

Здесь $h(n)$ - эвристическая функция для построения пути, а $g(n)$ -

функция подсчета расстояния. Примем их равными соответственно:

$$h(n) = \sqrt{(n_x - \text{goal}_x)^2 + (n_y - \text{goal}_y)^2}$$

$$g(n_{\text{neigh}}) = g(n_{\text{cur}}) + \text{cost}_{\text{neigh}} \cdot d(n_{\text{cur}}, n_{\text{neigh}})$$

В рассматриваемом случае доступными к посещению из отдельно взятой клетки примем все 8 её соседей, поэтому $d = \sqrt{2}$, если сосед находится на диагонале, и $d = 1$ иначе. Отметим также, что «стоимость» всех клеток будем считать равной 1.

На рисунке 2 синими точками помечены ячейки, через которые проходит найденный эффективный путь, состоящий из 18 клеток и 9 поворотов. Декартовы координаты точек (здесь и в дальнейшем) примем равными номеру ячейки карты по горизонтали и вертикали соответственно.

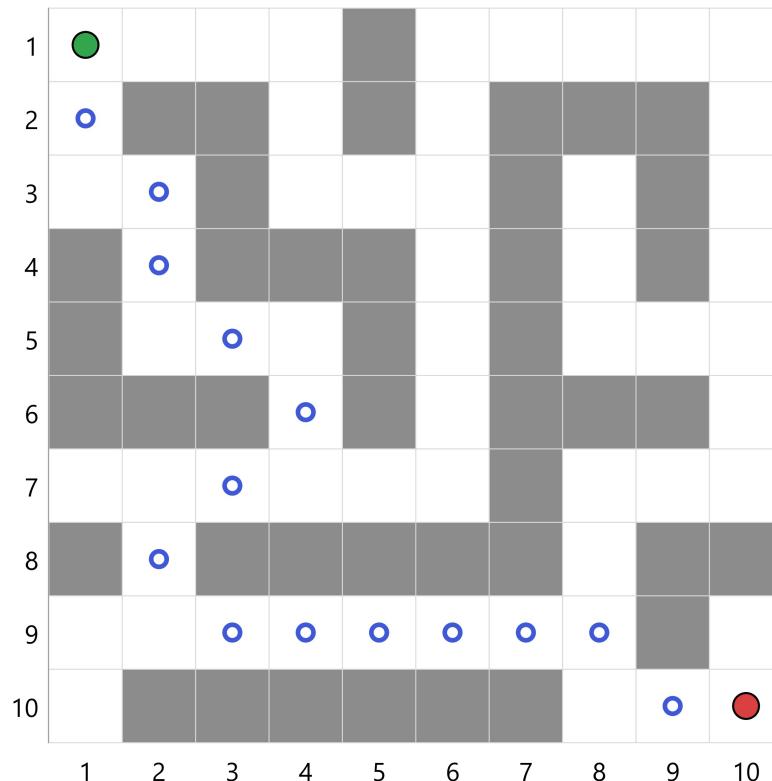


Рис. 2: Бинарная карта с полученной траекторией

2 C^0 -гладкая траектория

Рассмотрим самый очевидный способ соединения полученных точек (x_i, y_i) - прямыми отрезками. Получающаяся при этом траектория аналитически задаётся как:

$$S : \bigcup_{i=1}^{n-1} \begin{cases} -\sin \psi_i(x - x_i) + \cos \psi_i(y - y_i) = 0, \\ \psi_i = \arctan \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right), \\ r_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \\ Q_i(x, y) = \cos \psi_i(x - x_i) + \sin \psi_i(y - y_i) - r_i, \\ \text{Если } Q_i(x, y) > 0, \text{ то } i = i + 1. \end{cases}$$

Её производные не являются непрерывными на всем отрезке движения, поэтому она имеет нулевую гладкость. В целом, для C^n -гладкой кривой все её n производных являются непрерывными.

В описании траектории Q_i задают неявные прямые, разделяющие плоскость на две части, таким образом, возможно произвести переключение направления движения и рассчитать относительное положение объекта управления.

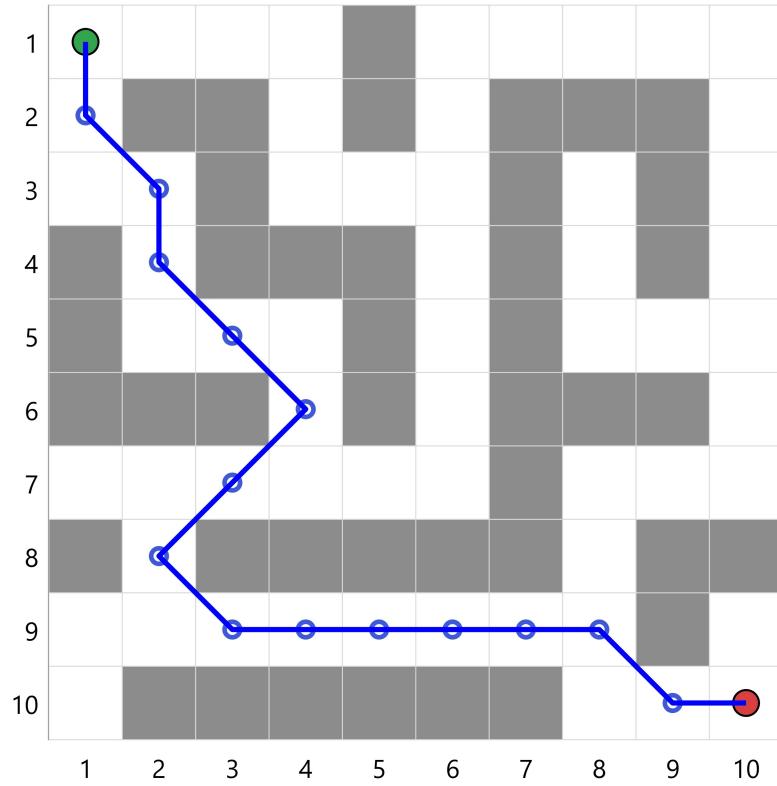
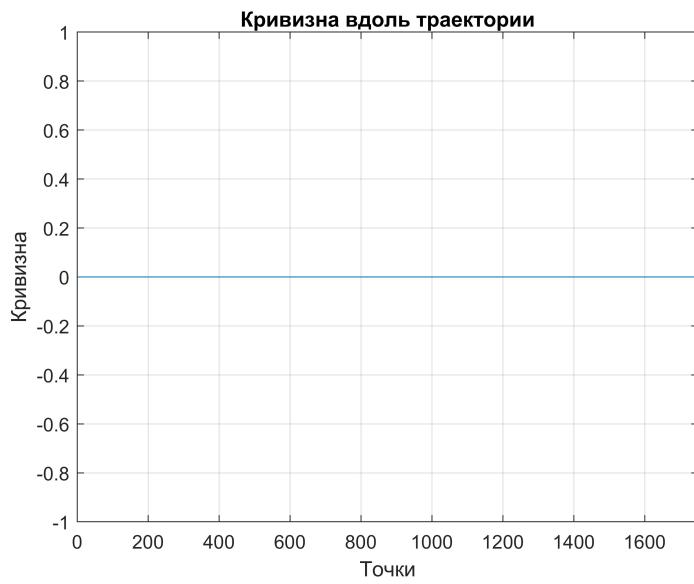
Применение алгоритма для полученных ранее точек приведено на рисунке 3. Расчёт кривизны кривой производится по формуле

$$k = \frac{x'y'' - y'x''}{((x')^2 + (y')^2)^{3/2}}$$

Где производные аппроксимируются разностями:

$$x'_i \approx \frac{x_{i+1} - x_{i-1}}{2\Delta t}, \quad x''_i \approx \frac{x_{i+1} - 2x_i + x_{i-1}}{2\Delta t^2}$$

Итоговый график приведен на рисунке 4. Траектория состоит только из прямых линий, для которых кривизна нулевая, поэтому и вся функция кривизны пути выходит константным нулем.

Рис. 3: C^0 -гладкая траекторияРис. 4: График кривизны C^0 -гладкой траектории

3 C^1 -гладкая траектория

Повысим гладкость получаемой в предыдущем пункте траектории с помощью добавления участков, состоящих из дуг окружностей с центрами в точках (x_{ci}, y_{ci}) и радиусом окружности $R = 0.8$, на которые будем входить из движения по прямому отрезку. Аналитически алгоритм описывается следующим выражением:

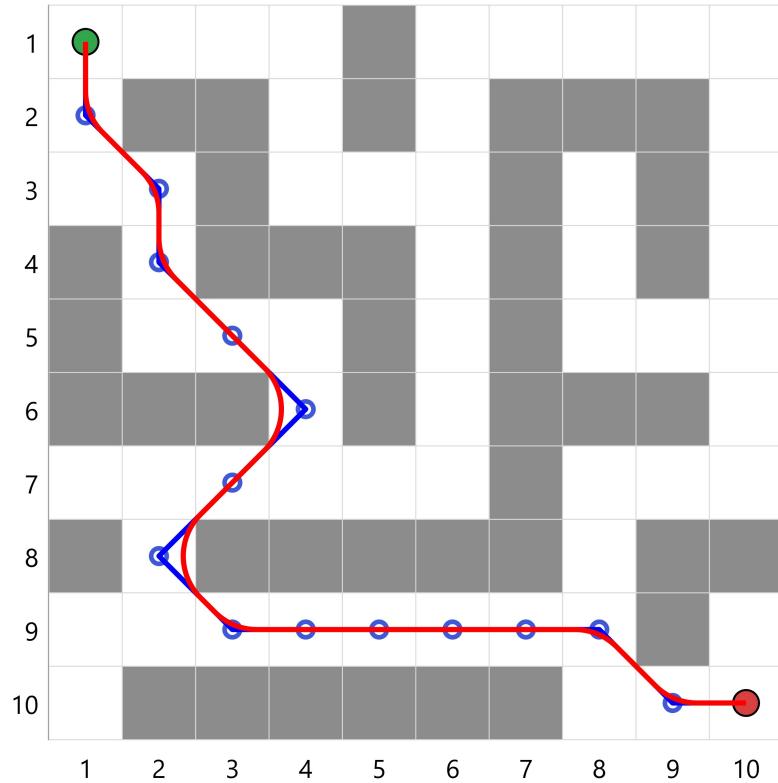
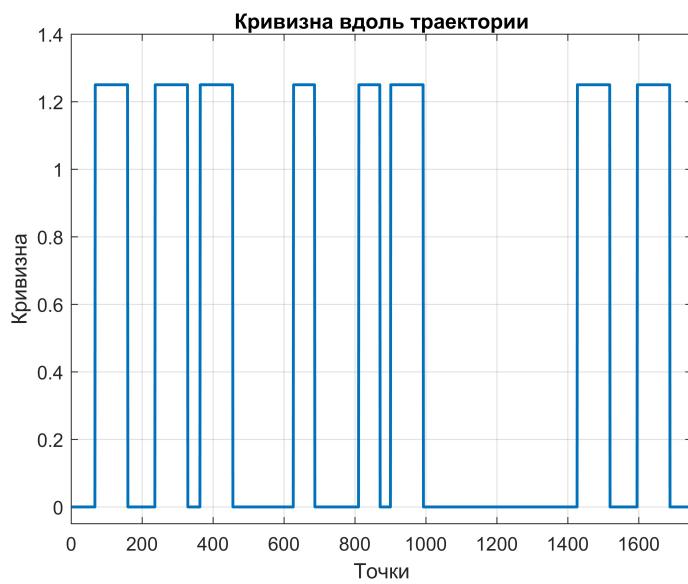
$$S : \bigcup_{i=1}^{n-1} \left\{ \begin{array}{l} -\sin \psi_i(x - x_i) + \cos \psi_i(y - y_i) = 0, \text{ если } Q_{1i} \leq 0 \\ (x - x_{ci})^2 + (y - y_{ci})^2 - R^2 = 0, \text{ если } Q_{2i} \leq 0 \\ \begin{bmatrix} x_{ci} \\ y_{ci} \end{bmatrix} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} + R_I^P(\psi_i)R_I^P(\delta_i) \begin{bmatrix} d_{ci} \\ 0 \end{bmatrix}, \\ \psi_i = \arctan 2 \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right), \\ \sigma_i = \begin{cases} \pi - (\psi_{i+1} - \psi_i), & \text{если } (\psi_{i+1} - \psi_i) > 0, \\ -\pi - (\psi_{i+1} - \psi_i), & \text{если } (\psi_{i+1} - \psi_i) \leq 0, \end{cases} \\ \delta_i = \pi - \frac{\sigma_i}{2}, \quad d_{ci} = \left| \frac{R}{\sin \frac{\sigma_i}{2}} \right|, \quad d_i = \left| \frac{R}{\tan \frac{\sigma_i}{2}} \right|, \\ r_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \\ Q_{1i} = \cos \psi_i(x - x_i) + \sin \psi_i(y - y_i) - r_i + d_i, \\ Q_{2i} = \cos \psi_{i+1}(x - x_i) + \sin \psi_{i+1}(y - y_i) - d_i, \\ \text{Если } Q_{2i} > 0, \text{ то } i = i + 1. \end{array} \right.$$

В описании использовали матрицу поворота

$$R_I^P(\varphi) = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{bmatrix}$$

Переключение участков движения по прямой и окружности происходит с помощью Q_{1i} и Q_{2i} соответственно.

Применение алгоритма для полученных ранее точек приведено на рисунке 5, а график кривизны найденной траектории - на рисунке 6. На последнем можем видеть резкие скачки на величину $1/R = 1.25$, характеризующие движение по окружности.

Рис. 5: C^1 -гладкая траекторияРис. 6: График кривизны C^1 -гладкой траектории

4 C^2 -гладкая траектория

У предыдущего метода есть серьезный недостаток в виде мгновенного изменения скорости движения объекта, что в реальном мире неосуществимо. В связи с чем актуальной становится задача повышения гладкости траектории до второго порядка с помощью дополнительного соединения прямых и окружностей параболами с \approx линейной производной. Описание кривой представлено ниже.

Движение по прямой:

$$S_1 : \begin{cases} -\sin \psi_i(x - x_i) + \cos \psi_i(y - y_i) = 0, \\ \psi_i = \arctan 2 \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right). \end{cases}$$

Движение по окружности:

$$S_3 : \begin{cases} (x - x_{ci})^2 + (y - y_{ci})^2 - R^2 = 0, \\ \begin{bmatrix} x_{ci} \\ y_{ci} \end{bmatrix} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} + R_P^I(\psi_i)R_I^P(\delta_i) \begin{bmatrix} d_{ci} \\ 0 \end{bmatrix}, \\ d_{ci} = d_{1i} + d_{2i}, \\ d_{1i} = \sqrt{R^2 - (h_i/2)^2}, \\ d_{2i} = \sqrt{h_{i2}^2 - (h_i/2)^2} \\ h_i = \sqrt{(x_{S_2e} - x_{S_1e})^2 + (y_{S_2e} - y_{S_1e})^2}, \\ h_{i2} = \sqrt{(x_{S_2e} - x_{i+1})^2 + (y_{S_2e} - y_{i+1})^2}, \\ \delta_i = \pi - \frac{\sigma_i}{2}, \\ \sigma_i = \begin{cases} \pi - (\psi_{i+1} - \psi_i), & \text{если } (\psi_{i+1} - \psi_i) > 0, \\ -\pi - (\psi_{i+1} - \psi_i), & \text{если } (\psi_{i+1} - \psi_i) \leq 0, \end{cases} \\ \psi_i = \arctan 2 \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right). \end{cases}$$

Движение по «открывающей» параболе:

$$S_2 : \begin{cases} \begin{bmatrix} x_L \\ y_L \end{bmatrix} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} + R_I^P(\psi_i) \begin{bmatrix} x - d_i \\ y \end{bmatrix}, \\ k(x_L)^3 - y_L = 0, \\ d_i = \left| \frac{R}{\tan \frac{\theta}{2}} \right|, \\ \psi_i = \arctan 2 \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right). \end{cases}$$

Движение по «закрывающей» параболе:

$$S_4 : \begin{cases} \begin{bmatrix} x_L \\ y_L \end{bmatrix} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} + R_I^P(\psi_{i+1}) \begin{bmatrix} x + d_i \\ y \end{bmatrix}, \\ -k(x_L)^3 - y_L = 0, \\ d_i = \left| \frac{R}{\tan \frac{\pi}{2}} \right|, \\ \psi_i = \arctan 2 \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right). \end{cases}$$

Переключение всё также будет происходить с помощью прямых Q_i . Итоговая кривая S определяется как объединение:

$$S = \bigcup_{i=1}^{n-1} \begin{cases} S_1, \text{ если } Q_{1i} \leq 0, \\ S_2, \text{ если } Q_{2i} \leq 0, \\ S_3, \text{ если } Q_{3i} \leq 0, \\ S_4, \text{ если } Q_{4i} \leq 0, \\ Q_{1i} = \cos \psi_i (x - x_i) + \sin \psi_i (y - y_i) - r_i + d_i, \\ Q_{2i} = \cos \psi_{i+1} (x - x_{i+1}) + \sin \psi_{i+1} (y - y_{i+1}) - d_i, \\ Q_{3i} = \cos \alpha_1 (x - x_i) + \sin \alpha_1 (y - y_i) + t_1, \\ Q_{4i} = \cos \alpha_2 (x - x_i) + \sin \alpha_2 (y - y_i) - t_2, \\ \text{Если } Q_{2i} > 0, \text{ то } i = i + 1. \end{cases}$$

Введенные вспомогательные параметры:

$$\begin{cases} t_1 &= [x_{S_1e} \quad y_{S_1e}]^T, \\ t_2 &= [x_{S_2e} \quad y_{S_2e}]^T, \\ \alpha_1 &= \frac{\psi_{i+1}-\beta}{2}, \\ \alpha_2 &= \psi_{i+1} - \frac{\psi_{i+1}-\beta}{2}, \\ \beta &= \arccos\left(1 - \frac{h_i^2}{2R^2}\right). \end{cases}$$

Геометрические характеристики:

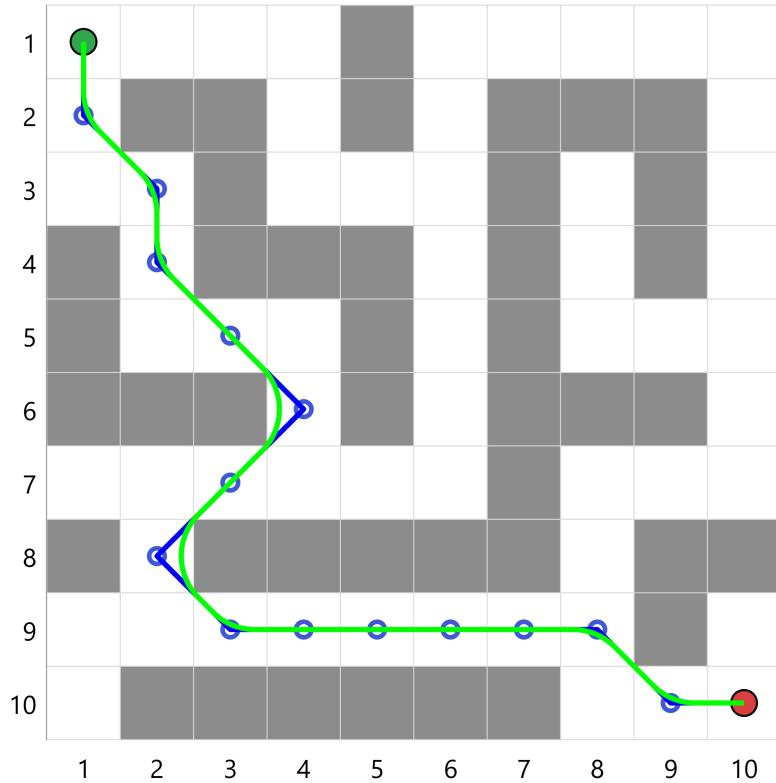
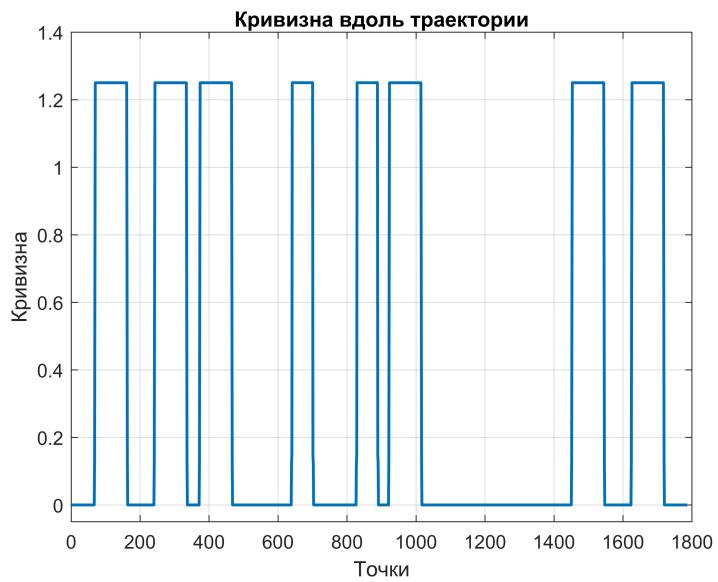
$$\begin{cases} h_i &= \sqrt{(x_{S_2e} - x_{S_1e})^2 + (y_{S_2e} - y_{S_1e})^2}, \\ \psi_i &= \arctan 2\left(\frac{y_{i+1}-y_i}{x_{i+1}-x_i}\right), \\ r_i &= \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \\ d_i &= \left|\frac{R}{\tan \frac{\pi}{2}}\right|. \end{cases}$$

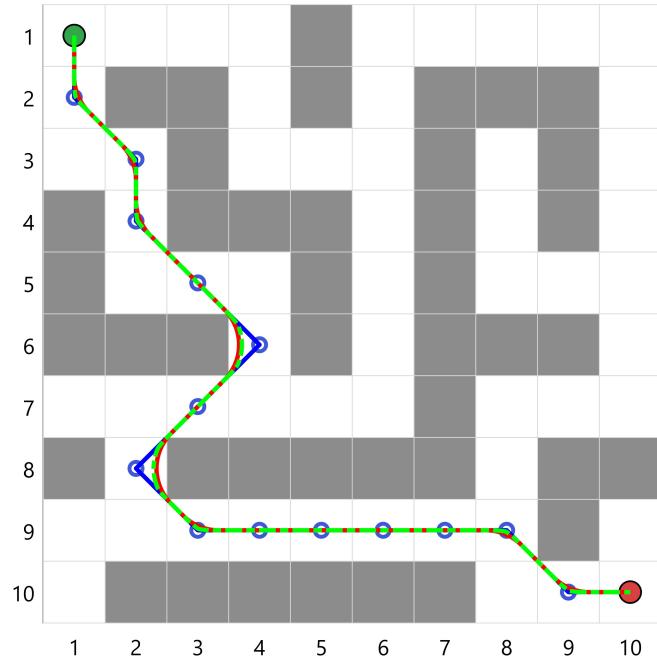
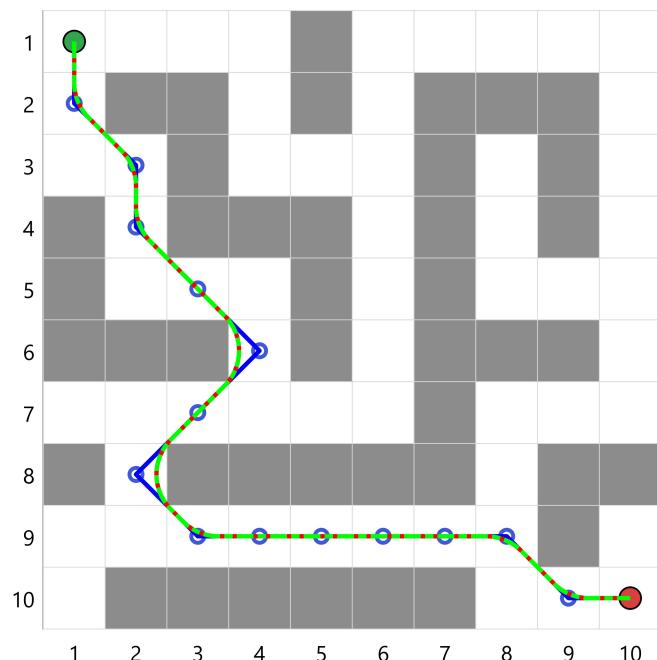
где x_{S_1e} , x_{S_2e} , y_{S_1e} и y_{S_2e} — конечные точки парабол S_2 и S_4 .

Параболы примечательны тем, что их производные можно в определенном порядке представить в виде линейно-нарастающей функции. Они создают дополнительный линейный разгон, предотвращая резкие изменения скорости движения.

Применение алгоритма для полученных ранее через A^* точек для $k = 20$ приведено на рисунке 7, а график кривизны полученной траектории - на рисунке 8. На последнем можем видеть плавный переход на величину $1/R = 1.25$. Резкий скачок уничтожен, движение полностью физически реализуемо!

Стоит отметить, что при увеличении k траектория второй гладкости всё больше приближается к C^1 -гладкой. Пожалуй, наиболее заметен данный эффект при сравнении рисунков 9 и 10 при $k_1 = 0.65$ и $k_2 = 100$, где красным цветом обозначен путь первой гладкости.

Рис. 7: C^2 -гладкая траекторияРис. 8: График кривизны C^2 -гладкой траектории

Рис. 9: C^2 -гладкая траектория при $k = 0.65$ Рис. 10: C^2 -гладкая траектория при $k = 100$

5 Сглаживание с помощью B -сплайна

Наконец, осуществим сглаживание C^0 -гладкой траектории при помощи B -сплайна. Сперва рекурсивно зададим базисные функции $N_{i,k}(t)$. Для $k = 0$:

$$N_{i,1}(t) = \begin{cases} 1, & u_i \leq t < u_{i+1} \\ 0, & \text{иначе} \end{cases}$$

Для $k \neq 0$:

$$N_{i,k}(t) = \frac{t - u_i}{u_{i+k} - u_i} N_{i,k-1}(t) + \frac{u_{i+k+1} - t}{u_{i+k+1} - u_{i+1}} N_{i+1,k-1}(t),$$

Здесь u_i - узлы, равномерно распределенные от 1 до $n - k$ с единичным шагом, а t — параметр в пределах той же области.

Точка кривой вычисляется тогда по формуле:

$$C(t) = \sum_{i=0}^n N_{i,k}(t) \cdot CP_i, \quad \text{где } CP_i \text{ — контрольные точки}$$

Контрольными же точками определим ранее вычисленные с помощью A^* алгоритма нахождения оптимального пути. Они зададут опору, но алгоритм, вообще говоря, не обязан через них проходить.

Применение сглаживания приведено на рисунке 11, а график кривизны получаемой траектории - на рисунке 12. При изучении последнего можем сказать, что необходимая для физической реализуемости гладкость достигается - нет резких скачков или иных аномалий. Однако в целом график сильно отличается от тех, что изучались нами до этого. Как минимум, не сходятся пики, а движение между участками не вполне предсказуемо.

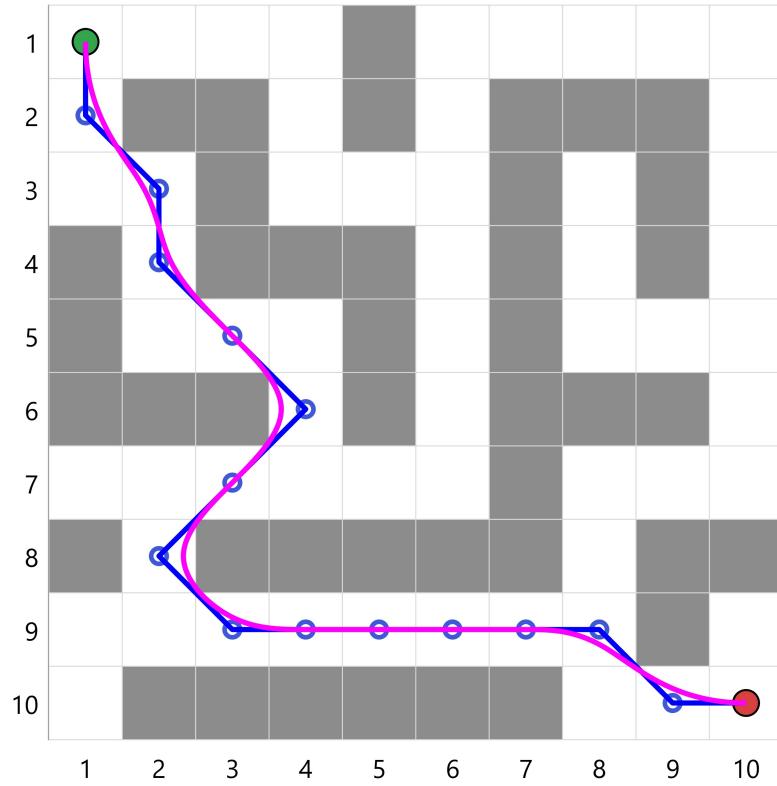


Рис. 11: Сглаживание с помощью B-сплайна

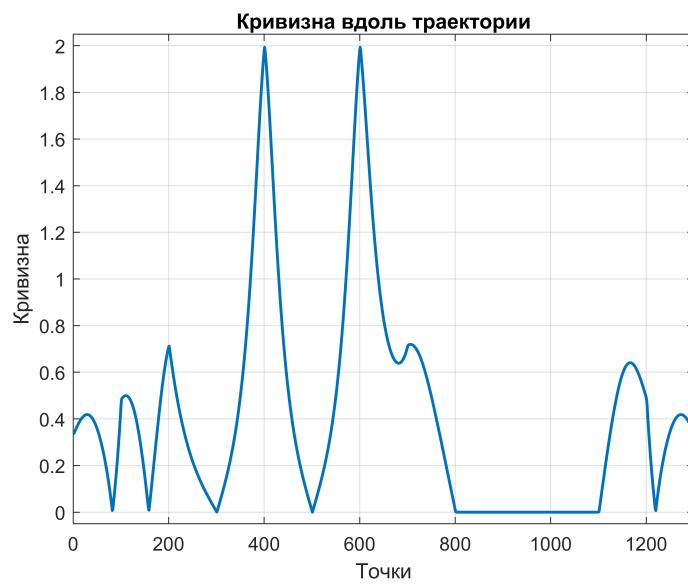


Рис. 12: График кривизны при сглаживании с помощью B-сплайна

6 Выводы

В результате выполнения лабораторной работы были изучены алгоритмы планирования траекторий движения: C^n -гладкие кривые, где $n \leq 2$, и сглаживание B-сплайнами. Было получено, что нулевая гладкость физически не является реализуемой - необходимо резко менять и направление движения, и скорость. Повышение n до 1 помогло решить проблему с видом траектории - она лишилась резких поворотов, а до $n = 2$ - проблему с плавным резким скорости. Достигалось всё это за счёт добавления дополнительных участков движения по окружности и кубической параболе соответственно.

Результаты для сглаживания сплайнами также оказались удовлетворительными - итоговая кривизна не имела резких скачков, а движение являлось плавным.