

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”

ФАКУЛЬТЕТ СИСТЕМ УПРАВЛЕНИЯ И РОБОТОТЕХНИКИ

Курсовой проект по теме:  
«Исследование задачи оптимизации траектории мобильного робота с  
дифференциальным приводом»  
по дисциплине «Разработка и оптимизация мехатронных систем»

Студент группы R4234с:

Ассистент ФСУиР Университета ИТМО

Выполнил:  
Ларионов А.Ю.

Научный руководитель :  
Ракшин Е.А.

Санкт-Петербург, 2026

# Содержание

<b>1</b>	<b>Цели и задачи</b>	<b>3</b>
1.1	Цель работы . . . . .	3
1.2	Поставленные задачи . . . . .	3
<b>2</b>	<b>Построение базовой траектории</b>	<b>4</b>
2.1	Построение карты среды . . . . .	4
2.2	Алгоритм $A^*$ . . . . .	4
2.3	Параметризация траектории В-сплайном . . . . .	5
<b>3</b>	<b>Постановка и решение задачи оптимизации</b>	<b>6</b>
3.1	Параметризация траектории . . . . .	6
3.2	Формирование целевой функции . . . . .	6
3.2.1	Длина траектории . . . . .	7
3.2.2	Кривизна . . . . .	7
3.2.3	Расстояние до препятствий . . . . .	7
3.3	Метод решения . . . . .	7
<b>4</b>	<b>Результаты</b>	<b>8</b>
<b>5</b>	<b>Вывод</b>	<b>9</b>
	<b>Список использованных источников</b>	<b>10</b>

# 1 Цели и задачи

## 1.1 Цель работы

Реализовать алгоритм планирования оптимальной траектории движения для мобильного робота: поиск кратчайшего пути ( $A^*$ ) с последующей аппроксимацией и оптимизацией по критериям - плавность, длина и безопасность (расстояние до препятствий).

## 1.2 Поставленные задачи

1) **Настройка среды:** Создать карты для тестирования (один центральный объект, небольшое кол-во объектов, много объектов).

2) **Планирование начального пути и параметризация:** Найти базовый ломаный путь с помощью алгоритма  $A^*$  по свободным ячейкам и интерполировать полученные точки  $A^*$  с помощью В-сплайна для перехода от дискретных координат к непрерывной функции.

3) **Оптимизация:** Определить целевую функцию для оптимизации на основе критериев длины, кривизны и расстояний до препятствий и применить минимизацию методом SLSQP из SciPy.

4) **Анализ и визуализация:** Построить графики сравнения начального и оптимизированного пути, визуализировать функции оптимизации и рассчитать метрики (длина пути, кривизна, расстояние до препятствий).

## 2 Построение базовой траектории

### 2.1 Построение карты среды

Среда представляется квадратной областью размером  $10 \times 10$ . Препятствия моделируются в виде окружностей с центрами  $(x_i, y_i)$  и радиусами  $r_i$ .

Для работы алгоритма поиска пути карта дискретизируется в регулярную сетку с шагом  $\Delta = 0.25$ . Каждая ячейка сетки определяется как:

$$\text{cell}(i, j) = \begin{cases} 1, & \text{если ячейка занята препятствием,} \\ 0, & \text{если свободна.} \end{cases}$$

Ячейка считается занятой, если расстояние от её центра до центра любого препятствия меньше радиуса препятствия:

$$\sqrt{(x - x_i)^2 + (y - y_i)^2} \leq r_i$$

### 2.2 Алгоритм A\*

Для поиска начальной траектории используется алгоритм A\*[1].

Он минимизирует функцию стоимости:

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

где:

- $g(n)$  — стоимость пути от старта до текущего узла,
- $h(n)$  — эвристическая оценка расстояния до цели.

В работе используется евклидова эвристика:

$$h(n) = \sqrt{(x_n - x_{goal})^2 + (y_n - y_{goal})^2}$$

Алгоритм возвращает дискретную ломаную траекторию, соединяющую старт и цель.

## 2.3 Параметризация траектории В-сплайном

Дискретная траектория не обладает достаточной гладкостью для управления мобильным роботом. Поэтому она аппроксимируется кубическим В-сплайном[2].

Параметрическая форма сплайна:

$$\mathbf{S}(u) = \sum_{i=0}^N P_i N_i^k(u)$$

где:

- $P_i$  — контрольные точки,
- $N_i^k(u)$  — базисные функции степени  $k$ ,
- $u \in [0, 1]$  — параметр.

Кубический сплайн соответствует  $k = 3$  и является непрерывной функцией, обладает непрерывностью второй производной, что обеспечивает плавность движения.

## 3 Постановка и решение задачи оптимизации

### 3.1 Параметризация траектории

После построения дискретной траектории алгоритмом  $A^*$  из неё выбирается конечное число точек, которые используются в качестве начальных контрольных точек В-сплайна.

Траектория представляется параметрической кривой:

$$S(u; P) = \sum_{i=0}^N P_i N_i^k(u), \quad u \in [0, 1]$$

где:

- $P_i = (x_i, y_i)$  — контрольные точки,
- $N_i^k(u)$  — базисные функции степени  $k$ ,
- $k = 3$  — кубический сплайн.

Контрольные точки образуют вектор переменных оптимизации:

$$P = (x_1, y_1, \dots, x_N, y_N)$$

Было выбрано 12 ключевых точек, первая и последняя точки фиксируются (старт и цель).

### 3.2 Формирование целевой функции

Задача оптимизации формулируется как:

$$\min_P J(P)$$

где целевая функция включает три критерия:

$$J = w_1 L + w_2 C + w_3 O$$

### 3.2.1 Длина траектории

$$L = \int_0^1 \left\| \frac{dS(u; P)}{du} \right\| du$$

Численно вычисляется по дискретным точкам кривой.

### 3.2.2 Кривизна

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

В качестве критерия используется среднее значение кривизны вдоль траектории.

### 3.2.3 Расстояние до препятствий

Для каждой точки траектории определяется минимальное расстояние до ближайшего препятствия. Если расстояние меньше безопасного порога  $d_{safe}$ , вводится штраф:

$$O = \sum_i \max(0, d_{safe} - d_i)^2$$

## 3.3 Метод решения

Минимизация выполняется методом SLSQP[3]. На каждой итерации:

1. обновляются контрольные точки,
2. строится В-сплайн[4],
3. вычисляется значение целевой функции,
4. выполняется шаг оптимизации.

Результатом является оптимальный набор контрольных точек  $P^*$  и соответствующая гладкая траектория.

## 4 Результаты

Ниже представлены графики сравнения до и после оптимизации таких значений как траектория движения, функция оптимизации и метрики критериев качества

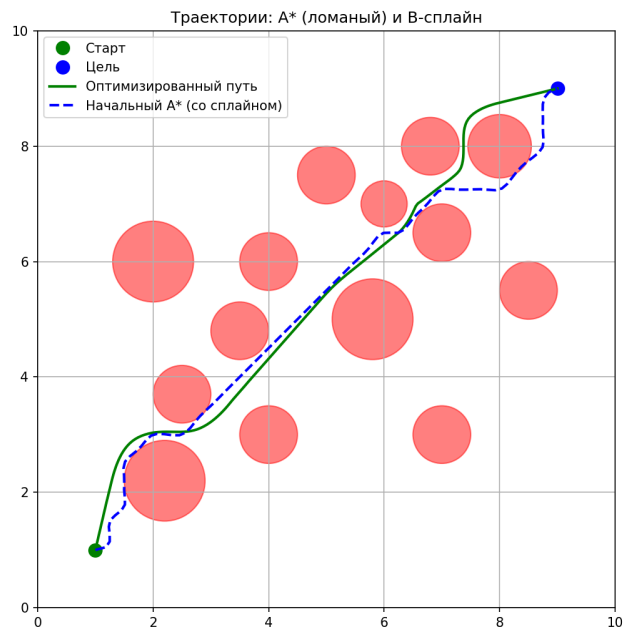


Рисунок 1—Траектория до и после оптимизации

Ниже представлена поверхность целевой функции. Значение функции до оптимизации равнялось 47.35, а после оптимизации 37.15

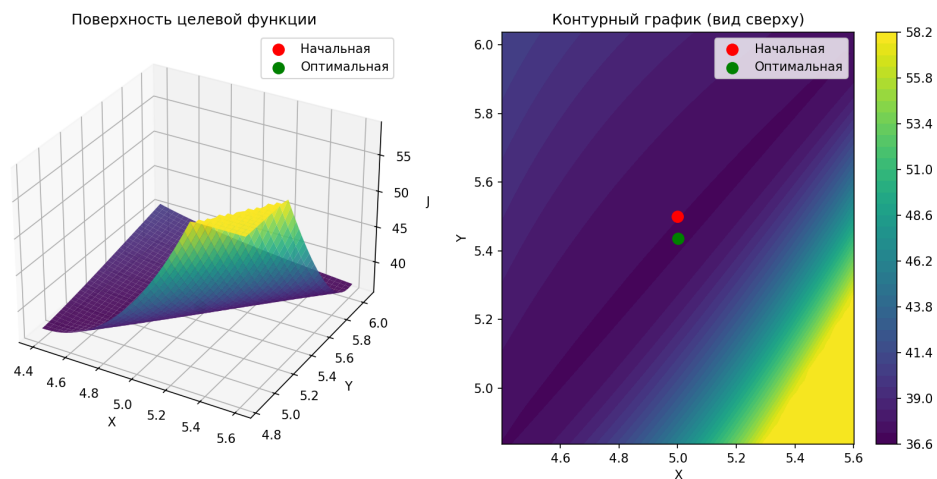


Рисунок 2—График функции оптимизации



Сравнение значений метрик до и после оптимизации:

```
=== МЕТРИКИ ===  
До оптимизации:  
Длина: 12.467  
Кривизна: 0.994  
Obstacle cost: 0.886  
  
После оптимизации:  
Длина: 12.190  
Кривизна: 0.474  
Obstacle cost: 0.236
```

Рисунок 3—Метрики критериев качества

Целевые критерии качества были оптимизированы и уменьшены, что отражается на графике траектории, которая стала сглаженной и оптимальной.

## 5 Вывод

В ходе данной работы была построена 2д карта с препятствиями, найдена базовая траектория движения робота через алгоритм  $A^*$ , а после составлена и оптимизирована функция на основе критериев качества (длина траектории, кривизна и минимальное расстояние до препятствия). Применение В-сплайна позволило получить гладкую непрерывную траекторию, а минимизация целевой функции обеспечила снижение длины пути, уменьшение кривизны и снижения критического расстояния до препятствий. В результате работы мы успешно применили методику оптимизации траектории с учетом критериев качества и получили оптимальную сглаженную траекторию движения робота.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Nilsson N. J. Raphael B. Hart, P. E. *A Formal Basis for the Heuristic Determination of Minimum Cost Paths*. IEEE Transactions on Systems Science and Cybernetics, 1968.
- [2] Ravankar A. A. Kobayashi Y. Emaru T. Hoshino Y. Ravankar, A. *Path Smoothing Techniques in Robot Navigation: State-of-the-Art, Current and Future Challenges*. Sensors, 2018.
- [3] SciPy Developers. `scipy.optimize.minimize` — `slsqp` method. <https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html>, 2024.
- [4] SciPy Developers. *Interpolation (`scipy.interpolate`)*. SciPy Developers, 2024.