

Санкт-Петербургский государственный университет

Кафедра системного программирования

Группа 22М.07-мм

*Лень Юлия Александровна*

Разработка прототипа системы  
моделирования навигации роботов в  
условиях неопределенности

Отчёт по производственной практике

Научный руководитель:  
д.ф.-м.н., проф., О.Н. Граничин

Санкт-Петербург  
2024

# Оглавление

<b>1. Введение</b>	<b>3</b>
<b>2. Постановка задачи</b>	<b>4</b>
<b>3. Обзор</b>	<b>5</b>
3.1. Навигационные системы . . . . .	5
3.2. Поиск положения на карте . . . . .	6
<b>4. Реализация</b>	<b>10</b>
4.1. Инструменты для реализации . . . . .	10
<b>5. Эксперименты</b>	<b>11</b>
5.1. Описание эксперимента . . . . .	11
5.2. Качество работы алгоритма при наличии разных шумов	11
5.3. Скорость работы алгоритмов . . . . .	11
<b>6. Заключение</b>	<b>14</b>
<b>Список литературы</b>	<b>15</b>

# 1. Введение

Роботизация в разных сферах жизни человека значительно выросла в последние годы. Активное развитие в науке и технике активно поспособствовало этому явлению. Сегодня роботы выполняют самый широкий круг задач: от сложных хирургических операций до доставки посылок в труднопроходимые местности. Однако, вместе с развитием технологий в области робототехники выявился ряд проблем, которые необходимо решать. Одной из таких является навигация роботов в условиях неопределенности.

Неопределенность может возникнуть из-за различных факторов, таких как изменение окружающей среды, неточность измерений в датчиках, наличие препятствий. В таких условиях традиционные методы навигации могут оказаться неэффективными, поэтому необходимо разрабатывать новые методы, которые смогут адаптироваться к изменяющимся условиям.

Одним из таких подходов является использование методов стохастической аппроксимации и рандомизированные алгоритмы [5,9]. Такие алгоритмы позволяют случайным образом выбирать некоторые значения параметров, чтобы получить удовлетворяющий потребностям результат с некоторой вероятностью для задач из класса NP-hard. Также рандомизация наблюдений позволяет минимизировать влияние нерегулярных (почти произвольных) шумов (систематических ошибок) [1] на результат.

Прежде чем использовать алгоритмы на практике и реализовывать их в железе, необходимо проверить, как разработанный или существующий алгоритм будет вести себя. В связи с этим встает вопрос о разработке системы, где будет возможность промоделировать поведение робота с разными алгоритмами и выбрать тот, который будет подходить в рамках поставленной перед роботом и человеком задачей.

## 2. Постановка задачи

Целью данной работы является разработка прототипа системы моделирования навигации роботов в условиях неопределенности. Для достижения поставленной цели необходимо выполнить следующие задачи:

- выполнить обзор методов в области моделирования навигации роботов;
- разработать прототип системы моделирования для алгоритмов навигации в условиях неопределенности;
- провести эксперименты с разными модификациями данных.

## 3. Обзор

### 3.1. Навигационные системы

Навигация при разработке роботов является одной из наиболее важных задач. Широкий спектр задач, решаемых роботами, привел к широкой классификации внутри навигационных систем (НС), которые можно разделить на несколько видов категорий, в зависимости от разных критериев разбиения. По типу работы с координатами [2] НС разделяют на:

- глобальные НС;
- локальные НС;
- персональные НС.

В глобальных НС позиция робота рассчитывается в абсолютных координатах. Например, значения ширины и долготы при использовании GPS являются ярким примером такой системы. В локальных же наоборот, система выбирает некую условную точку начала координат и запоминает позиционирование относительно этой точки. В качестве примера можно вспомнить робота-пылесоса, у которого заранее известна ограниченная область, где он будет передвигаться, а точкой отсчета считает свою станцию. Персональная НС чаще всего используется для роботов-манипуляторов, так как положение объектов считается относительно частей робота.

Также в навигации системы подразделяют по принципу получения информации на активные и пассивные. Активные НС рассчитывают свое положение самостоятельно, в то время как пассивные получают эту информацию извне. Исходя из этого описания можно сделать вывод, что в большинстве случаев глобальные НС получают информацию пассивно, персональные — активно, а локальные могут быть как активными, так и пассивными.

Среди задач в области навигации стоит отметить задачу корректировки НС по данным датчика и карты [3], так как в последнее время ее

актуальность растет. Интерес к ней вызван проблемой поиска альтернативы спутниковым системам и работой с зашумленными данными. В рамках корректировки важным шагом является поиск своего текущего положения объекта, если известны только последние несколько точек маршрута. В рамках текущей работы будут рассмотрены варианты решения этой подзадачи корректировки НС.

### 3.2. Поиск положения на карте

Пусть известен маршрут передвижения и его карта высот этого маршрута. Такую карту строит, например, Яндекс Карта для пеших маршрутов (рис. 1).

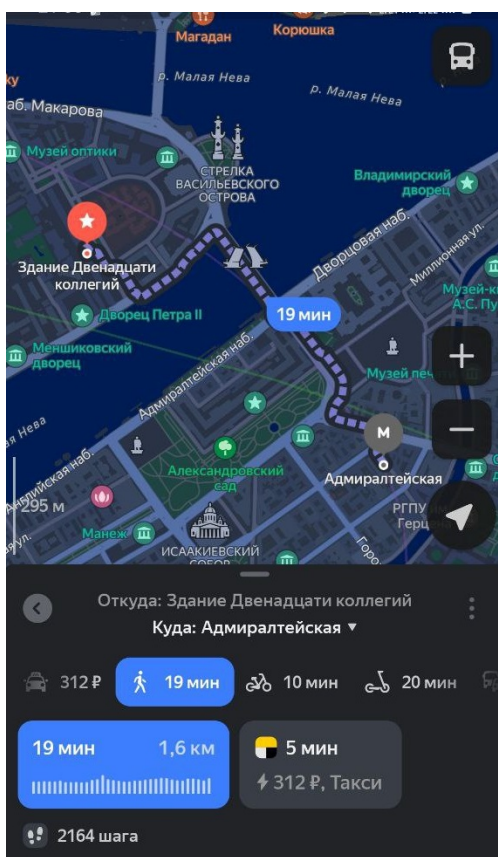


Рис. 1: Скриншот мобильного приложения Яндекс Карты с отображением маршрута на карте (фиолетовая линия) и маршрута в терминах карты высот.

Пусть дан фрагмент, который был пройден и записан в ограниченный буфер с помощью некоего измерителя. Задача — найти положение

записанного в буфер фрагмента. Поиск положения может осложняться тем, что измеритель может получать данные с погрешностью и помехами. поэтому в рамках работы будет рассмотрено три метода для поиска фрагментов: метод перебора, метод корреляции и метод на основе быстрого преобразования Фурье.

### 3.2.1. Метод перебора

Метод перебора является самым простым методом поиска данных в массиве. В рамках текущей задачи искомый фрагмент будет последовательно сравниваться пооконно с оригинальной картой. Главным преимуществом этого метода является его простота и скорость вычислений, однако данный подход перестает работать в случае зашумленных данных.

### 3.2.2. Метод корреляции

Метод корреляции также основан на сравнении фрагмента с каждым доступным участком всего маршрута, только теперь для каждого участка дороги рассчитывается значение корреляции. Пусть  $M$  — это маршрут в терминах карты высот длины  $m$  (карта),  $P$  — измеренный датчиком пройденный маршрут длины  $p$ ,  $M^j \in M$  — это фрагмент маршрута равный по длине пройденному маршруту. Теперь для всех фрагментов  $j = 1, 2..M - P$  рассчитаем значение корреляции по следующей формуле:

$$r_j = \frac{\sum_{i=1}^p (M_i^j - \bar{M}^j)(P_i - \bar{P})}{\sqrt{\sum_{i=1}^p (M_i^j - \bar{M}^j)^2 \sum_{i=1}^p (P_i - \bar{P})^2}}, \quad (1)$$

где  $\bar{M}^j, \bar{P}$  - средние значения фрагмента карты и пройденного маршрута соответственно. Так как значение коэффициента корреляции находится в диапазоне от  $-1$  до  $1$ , то для того, чтобы найти фрагмент, где значение коэффициента максимально приближено к единице.

Преимущество такого подхода заключается в том, что он позволяет найти фрагменты, максимально похожие на измеренный путь, даже ес-

ли измерения были произведены с помехами. Однако вычислительная скорость такого подхода больше, чем у перебора, так как для каждого фрагмента необходимо рассчитать коэффициент, а потом найти максимум среди них.

### 3.2.3. Метод на основе преобразования Фурье

Преобразование Фурье активно применяется в областях, связанных с анализом временных рядов. В частности можно использовать преобразование Фурье для расчета взаимной корреляции двух функций. Пусть этими функциями будут карта и искомый фрагмент, тогда алгоритм поиска пройденного пути на карте с использованием быстрого преобразования Фурье (БПФ, FFT), будет выглядеть следующим образом:

1.  $M$  — карта,  $P$  — пройденный маршрут (фрагмент).
2. Вычитаем среднее значение из каждого элемента фрагмента:  $P_{-cp} = P - \bar{P}$ .
3. Расширяем фрагмент до размера карты (заполняем нулями), затем переворачиваем последовательность элемента таким образом, что первый элемент становится последним, а последний первым. Обозначим полученный массив  $P_{inv}$ .
4. Применяем преобразование Фурье к карте и расширенному фрагменту:  $M_{fft} = FFT(M)$ ,  $P_{fft} = FFT(P_{inv})$ .
5. Перемножаем поэлементно преобразованные карту и фрагмент:  $PM = M \cdot P$ .
6. Применяем обратное преобразование Фурье к произведению:  $PM_{rfft} = RFFT(PM)$ .
7. Ищем индекс, где последовательность имеет максимальное значение. Это и будет искомым положением фрагмента.



Данный подход позволяет выполнить метод, похожий на корреляцию, только более быстрым способом.

## 4. Реализация

### 4.1. Инструменты для реализации

Для программной реализации был взят язык программирования Python 3 [7], обладающий большим количеством математических библиотек. Графический интерфейс разработан при помощи QT Designer [8].

В качестве источника данных для карты высот используется открытый Open Topo Data [6].

## 5. Эксперименты

### 5.1. Описание эксперимента

Для эксперимента были взяты данные из ASTER (The Advanced Spaceborne Thermal Emission and Reflection Radiometer) [4]. Для экспериментов Было скачено 4 700 точек вдоль маршрута, так как в бесплатной версии API стоит ограничение на один запрос в секунду.

Было проведено два типа экспериментов: качество работы алгоритмов при наличии шумов на фрагменте и скорость работы алгоритмов при разном объеме имеющихся данных.

### 5.2. Качество работы алгоритма при наличии разных шумов

В рамках эксперимента проверялось качество работы алгоритма при следующих параметрах:

- на фрагмент не накладывался шум;
- на фрагмент накладывался гауссовский шум;
- искомый фрагмент масштабировался (значение умножалось на константу)

В ходе экспериментов подтвердилось, что метод перебора перестает работать при наложении шумов и масштабировании фрагмента. Метод корреляции и метод на основе преобразования Фурье показали правильные результаты при наложении гауссовских шумов и масштабировании фрагмента.

### 5.3. Скорость работы алгоритмов

Для сравнения скорости работы были проведены эксперименты со следующими параметрами:

- Искомый фрагмент карты может находиться в начале, середине или в конце карты.
- Для сравнения брали длины карты равны 100, 1000, искомый фрагмент был длины 10, 100.
- Для каждого метода поиск проводился 1000 раз и рассчитывалось среднее время выполнения работы.

	Фрагмент в начале, с	Фрагмент в середине, с	Фрагмент в конце, с
Метод перебора	0.00003	0.00003	0.00003
Метод корреляции	0.00200	0.00198	0.00197
Метод Фурье	0.00008	0.00008	0.00007

Таблица 1: Эксперимент:  $m = 100, p = 10$ .

	Фрагмент в начале, с	Фрагмент в середине, с	Фрагмент в конце, с
Метод перебора	0.00036	0.00034	0.00036
Метод корреляции	0.02064	0.02037	0.02070
Метод Фурье	0.00033	0.00032	0.00033

Таблица 2: Эксперимент:  $m = 1000, p = 10$ .

	Фрагмент в начале, с	Фрагмент в середине, с	Фрагмент в конце, с
Метод перебора	0.00074	0.00074	0.00076
Метод корреляции	0.12830	0.12982	0.13000
Метод Фурье	0.00033	0.00034	0.00033

Таблица 3: Эксперимент:  $m = 1000, p = 100$ .

Из таблиц представленных выше видно, что положение искомого фрагмента в последовательности практически не влияет на скорость вычислений. Метод перебора является самым быстрым методом при малом количестве измерений, однако перестает работать при зашумленных данных. Метод корреляции работает дольше всех алгоритмов,

однако позволяет работать с зашумленными данными. Метод Фурье дает вычислительное преимущество при увеличении размера фрагмента и размера карты для поиска, а также показывает хорошие результаты при зашумленных данных фрагмента.

## 6. Заключение

В ходе работы в текущем семестре были выполнены следующие задачи:

- выполнен обзор методов в области моделирования в области навигации;
- разработан прототипа системы моделирования навигации роботов, в который входит программная реализация алгоритмов;
- выполнено сравнение ряда методов в навигационной задаче.

## Список литературы

- [1] Граничин О.Н., Поляк Б.Т. Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах. — М.: Наука, 2003. — С. 291.
- [2] Кремповский Павел Романович, Луцков Юрий Иванович. Навигационные системы автоматизированных робототехнических комплексов // IEEE Transactions on Automatic Control. — 2016. — Т. 61, № 5. — С. 58–61.
- [3] Степанов О.А. Методы обработки навигационной измерительной информации. — СПб: Университет ИТМО, 2017. — С. 196.
- [4] ASTER - OPEN TOPO DATA. — URL: <https://www.opentopodata.org/datasets/aster/>.
- [5] Calafiore G., Polyak B. Stochastic algorithms for exact and approximate feasibility of robust LMIs // IEEE Trans. Autom. Control. — 2001. — Vol. 46. — P. 1755–1759.
- [6] Open Topo Data. — URL: <https://www.opentopodata.org/>.
- [7] Python. — URL: <https://www.python.org/>.
- [8] Qt | Tools for Each Stage of Software Development Lifecycle. — URL: <https://www.qt.io/>.
- [9] Tempo R., Calafiore G., Dabbene F. [Randomized algorithms for analysis and control of uncertain systems. With applications.](#) 2nd revised ed. — New York: Springer-Verlag., 2013. — ISBN: 978-1-4471-4609-4.