

Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего
образования

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ

Кафедра систем управления и информатики

Отчет по практической работе

«РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ПРОЦЕССОМ ПАРКОВКИ
РОБОТА-МАШИНКИ»

по дисциплине

«Бесконтактные средства локальной ориентации роботов»

Выполнили: студенты гр. Р4235
Антонов Е. С.,
Артемов К. А.

Преподаватель: Власов С. М.,
ассистент каф. СУиИ

Санкт-Петербург
2017

Содержание

Введение	3
1 Особенности строения робота	5
2 Управление движением робота	7
2.1 Кинематическая модель	7
2.2 Локализация робота	8
2.3 Структура системы управления	9
3 Поиск парковочного места	15
4 Планирование траекторий движения	19
Заключение	21
Приложение А Описание экспериментов по проверке состоятельности используемых методов локализации робота с помощью сторонней системы технического зрения	22
Приложение Б Уравнения используемых элементарных траекторий	26

Введение

Данный документ призван познакомить читателей с результатами работы авторов над задачей создания системы управления для робота-машинки, которая бы давала ему способность автоматически (самостоятельно) выполнять параллельную парковку.

Более конкретно ее можно описать примерно так.

Имеется робот-машинка, ходовая часть которого устроена примерно так же, как у настоящего заднеприводного автомобиля: один из пары его двигателей приводит во вращение задние колеса, второй отвечает за поворот передних, рулевых колес. Данный робот должен проехать вдоль возможного места парковки, обозначенного с помощью посторонних объектов, имитирующих собой другие стоящие неподвижно транспортные средства (см. рисунок 1), оценить его геометрические параметры, необходимые для совершения маневра, характерного для параллельной парковки, и, собственно, проделать последний.

Для ее решения авторам пришлось проработать следующие технические вопросы:

- создание упомянутого робота из конструктора LEGO Mindstorms EV3;
- подбор для него датчиков и программная реализация алгоритмов обработки поступающей с них информации;
- проектирование системы управления движением робота;
- создание алгоритма картирования парковочного места и его окрестностей.

Описанию их ключевых моментов и посвящена основная часть этого документа.



Рисунок 1 – Общий вид зоны проведения экспериментов.

1 Особенности строения робота

Особенности строения использованного в работе робота-машинки (см. рисунок 3) даются следующим перечислением:

- робот собран из конструктора LEGO Mindstorms EV3;
- робот имеет два двигателя со встроенным энкодерами, один из которых (тяговый, 1) приводит в движение задние колеса, а второй (рулевой, 2) поворачивает передние;
- усилие с тягового двигателя на задние колеса передается через дифференциал 3 с передаточным отношением, обеспечивающим равенство угловой скорости вращения вала двигателя с полусуммой угловых скоростей задних колес;
- рулевые колеса связаны друг с другом и с рулевым двигателем через рулевую трапецию, кинематическая схема которой изображена на рисунке 2;

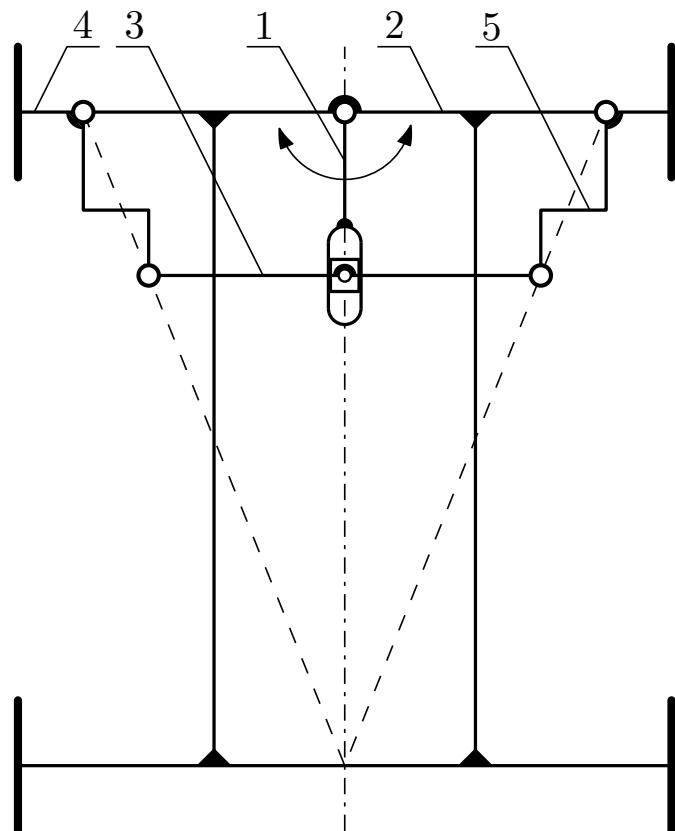
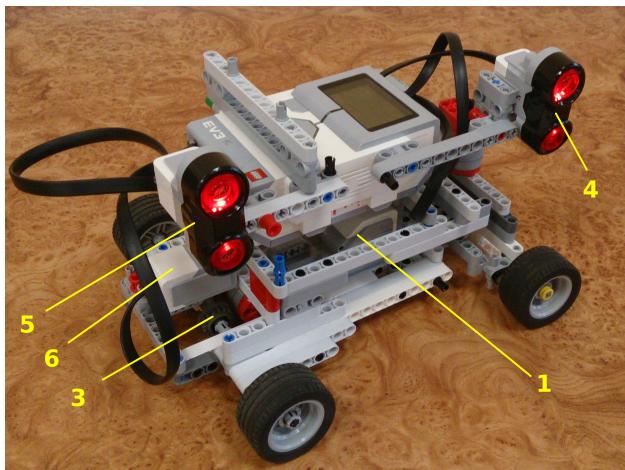
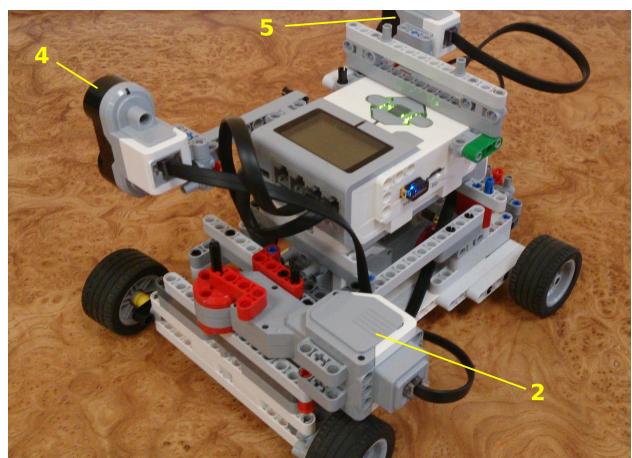


Рисунок 2 – Кинематическая схема рулевой трапеции: 1 — коромысло, приводимое в движение рулевым двигателем, 2 — шасси робота, 3 — шатун, 4, 5 — коромысла, жестко соединенные с осями вращения передних колес.

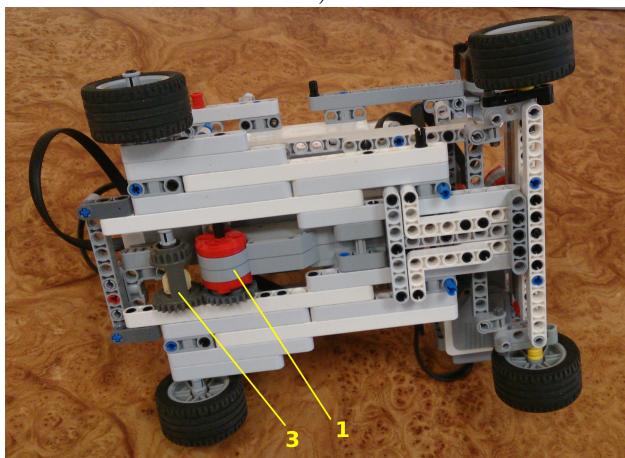
- для измерения расстояний до объектов окружающей среды робот имеет два ультразвуковых дальномера 4, 5;
- для определения собственного угла поворота и угловой скорости робот снабжен возвращающим их датчиком-гироскопом 6.



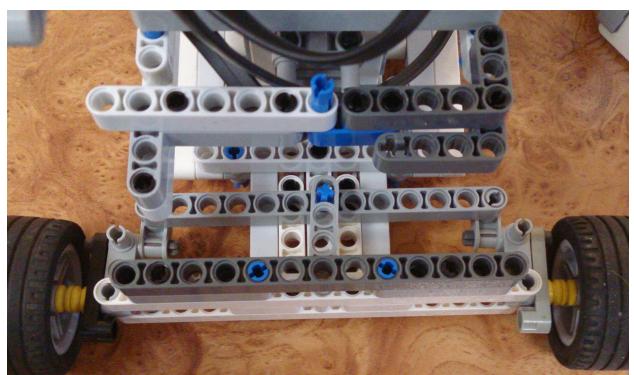
а)



б)



в)



г)

Рисунок 3 – Внешний вид использованного в работе робота-машинки: а — с одного из бортов, б — с другого, в — снизу, г — спереди без рулевого двигателя и некоторых его креплений.

2 Управление движением робота

2.1 Кинематическая модель

Кинематическая модель робота имеет вид [?]:

$$\begin{cases} \dot{x} = v \cos \theta \\ \dot{y} = v \sin \theta \\ \dot{\theta} = \omega \end{cases} \quad (1)$$

где x, y — декартовы координаты точки C , являющейся серединой задней оси (см. рисунок 4); θ — угол поворота робота (угол между осями абсцисс неподвижной системы координат Ox_0y_0 и системы координат Ox_1y_1 , жёстко связанной с роботом); v — проекция скорости \vec{v} точки C на ось абсцисс системы координат $Ox_1y_1^*$; ω — угловая скорость вращения робота.

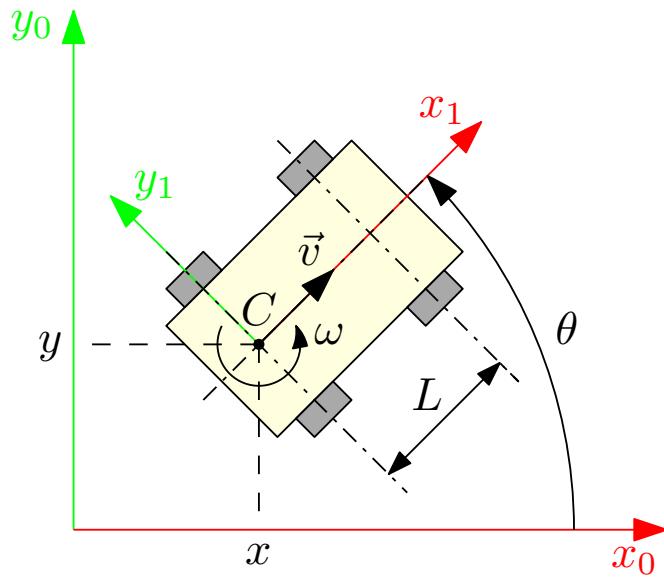


Рисунок 4 – Чертеж-пояснение к кинематической модели робота.

Угловая скорость ω для робота-машинки с одним поворотным колесом оказывается связанный с углом его поворота φ выражением [?]

$$\omega = \frac{v}{L} \operatorname{tg} \varphi. \quad (2)$$

Это равенство получается объединением следующих двух уравнений:

$$\omega = \frac{v}{R}, \quad (3)$$

$$\operatorname{tg} \varphi = \frac{L}{R}, \quad (4)$$

где R — радиус дуги, по которой движется робот, а точнее точка C (см. рисунок 5).

Так как у робота из данной работы колеса два, и примененная в нем рулевая трапеция не обеспечивает их точного поворота на те углы, при которых они не будут проскальзывать в

* В данной работе проскальзывание задних колес робота считается отсутствующим, а, следовательно, вектор \vec{v} — всегда коллинеарным оси абсцисс системы координат Ox_1y_1 .

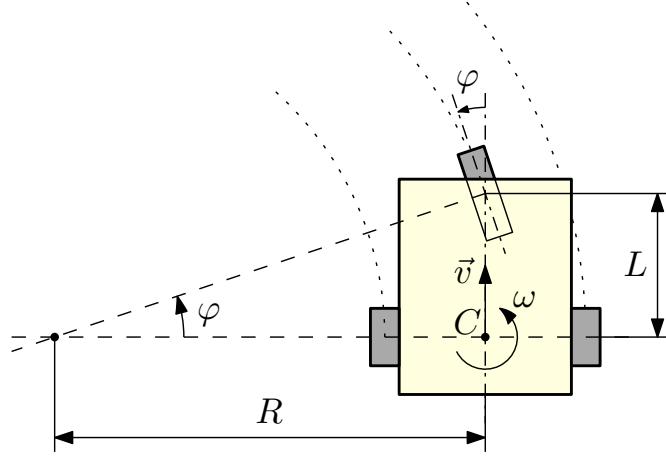


Рисунок 5 – Движение робота с одним рулевым колесом по дуге.

поперечном направлении [?], использование по отношению к нему выражений (2)–(4), строго говоря, невозможно. При этом с ожиданием получения от нее приближенных, но близких к истинным результатов расчетов первое из этих выражений в форме

$$\omega = \frac{v}{L} \operatorname{tg} \bar{\varphi}. \quad (5)$$

где $\bar{\varphi}$ — угол поворота вала рулевого двигателя, все же используется в данной работе. Основанием этого решения являются результаты «моделирования» работы рулевой трапеции в программе GeoGebra (см. рисунок 6), согласно которым перпендикуляры к передним колесам и поворотному коромыслу не пересекаются на прямой, проходящей через ось вращения задних колес, совсем немного.

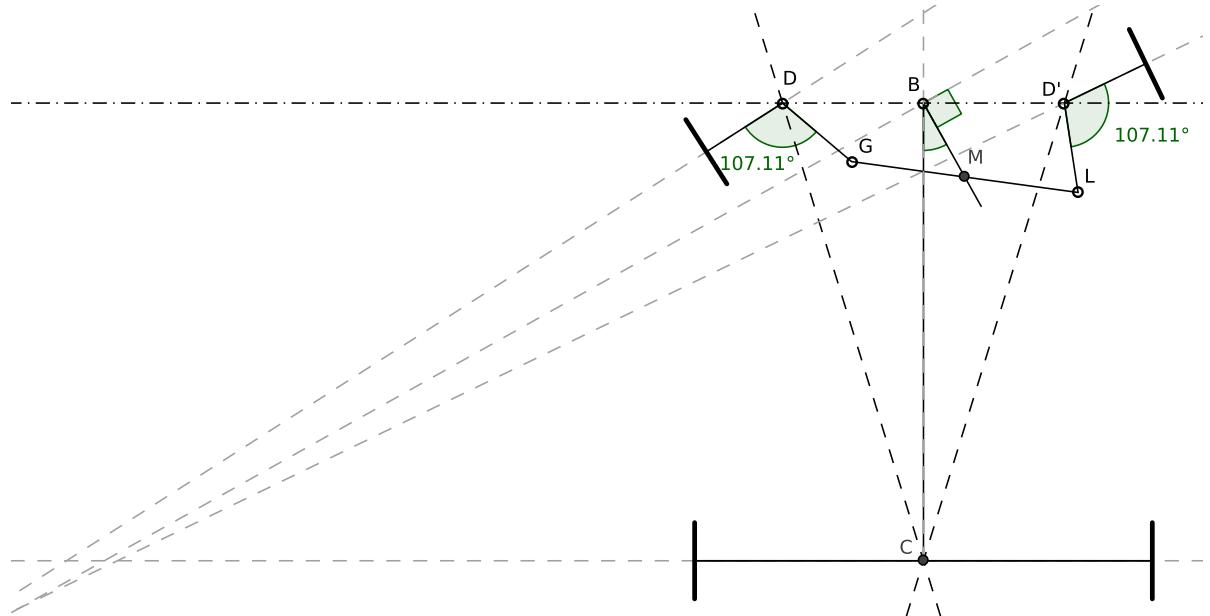


Рисунок 6 – Рулевая трапеция в смещенном относительно центрального положении ($\angle CBM = \bar{\varphi}$, $GM = ML$, $|BM| = var$).

2.2 Локализация робота

В качестве угла θ и угловой скорости ω в работе использовались угол и угловая скорость, возвращаемые установленным на робота датчиком-гироскопом. Координаты x и

y , в свою очередь, непосредственно не измерялись, а рассчитывались с использованием первых двух уравнений модели (1). При этом линейная скорость точки C с учетом третьего пункта перечня, представленного в разделе 1, определялась в соответствии со следующим выражением

$$v = \underline{\omega}r, \quad (6)$$

где $\underline{\omega}$ — угловая скорость вращения вала тягового двигателя, r — радиус задних колес робота.

Состоительность описанного принципа локализации робота была проверена с помощью сторонней системы технического зрения. Подробности соответствующих экспериментов и полученные результаты доступны в Приложении А.

2.3 Структура системы управления

Общая структура системы управления движением робота, позволяющая ему двигаться по желаемой траектории, показана на рисунке 7. Указанные на ней физические величины, ранее не упоминавшиеся в тексте данного отчета, значат следующее:

U_1 (U_2) — напряжение, подаваемое на тяговый (рулевой) двигатель, выраженное в процентах от максимального напряжения (знак определяет направление вращения);

$\bar{\varphi}$ — угол поворота вала рулевого двигателя;

$\bar{\varphi}_{min}$, $\bar{\varphi}_{max}$ — его минимальное и максимальное значения ($\bar{\varphi}_{min} = -\bar{\varphi}_{max}$);

x_r и y_r — координаты, которые должен иметь робот в данный момент времени, чтобы следовать по желаемой траектории;

X_{des} — желаемое значение величины X .

Входящие в блок Velocity Controller ПИД-регуляторы были настроены вручную. Качество их работы можно оценить из переходных функций, показанных на рисунках 8 и 9.

Качество управления угловой скоростью по прямой связи, включающей в себя функцию $\text{arctg}()$, можно оценить из рисунка 10.

Формирование желаемых значений для линейной и угловой скоростей робота блоком Converter, а также расчет «предварительных» управляющих воздействий в блоке «Simple Controller» производятся в соответствии со следующими выражениями [?]:

$$\begin{cases} \dot{\xi} = u_1 \cos \theta + u_2 \sin \theta, \\ v_{des} = \xi, \\ \omega_{des} = \bar{\omega}_{des}, \end{cases} \quad \bar{\omega}_{des} = \begin{cases} \frac{-u_1 \sin \theta + u_2 \cos \theta}{\xi}, & \xi \neq 0 \\ 0, & \xi = 0 \end{cases} \quad (7)$$

$$\begin{cases} u_1 = \ddot{x}_r + k_{p1}(x_r - x) + k_{d1}(\dot{x}_r - \dot{x}), \\ u_2 = \ddot{y}_r + k_{p2}(y_r - y) + k_{d2}(\dot{y}_r - \dot{y}). \end{cases} \quad (8)$$

Результаты запуска схемы моделирования, показанной на рисунке 11*, свидетельствуют о надлежащем функционировании этой системы управления (см. рисунок 12). То же следует и из реальных экспериментов — см. рисунок 13.

* В данной схеме в качестве модели робота-машинки использована модель (1). Другими словами, блок Velocity Controller опущен «за ненадобностью», и $v_{des} = v$, $\omega_{des} = \omega$.

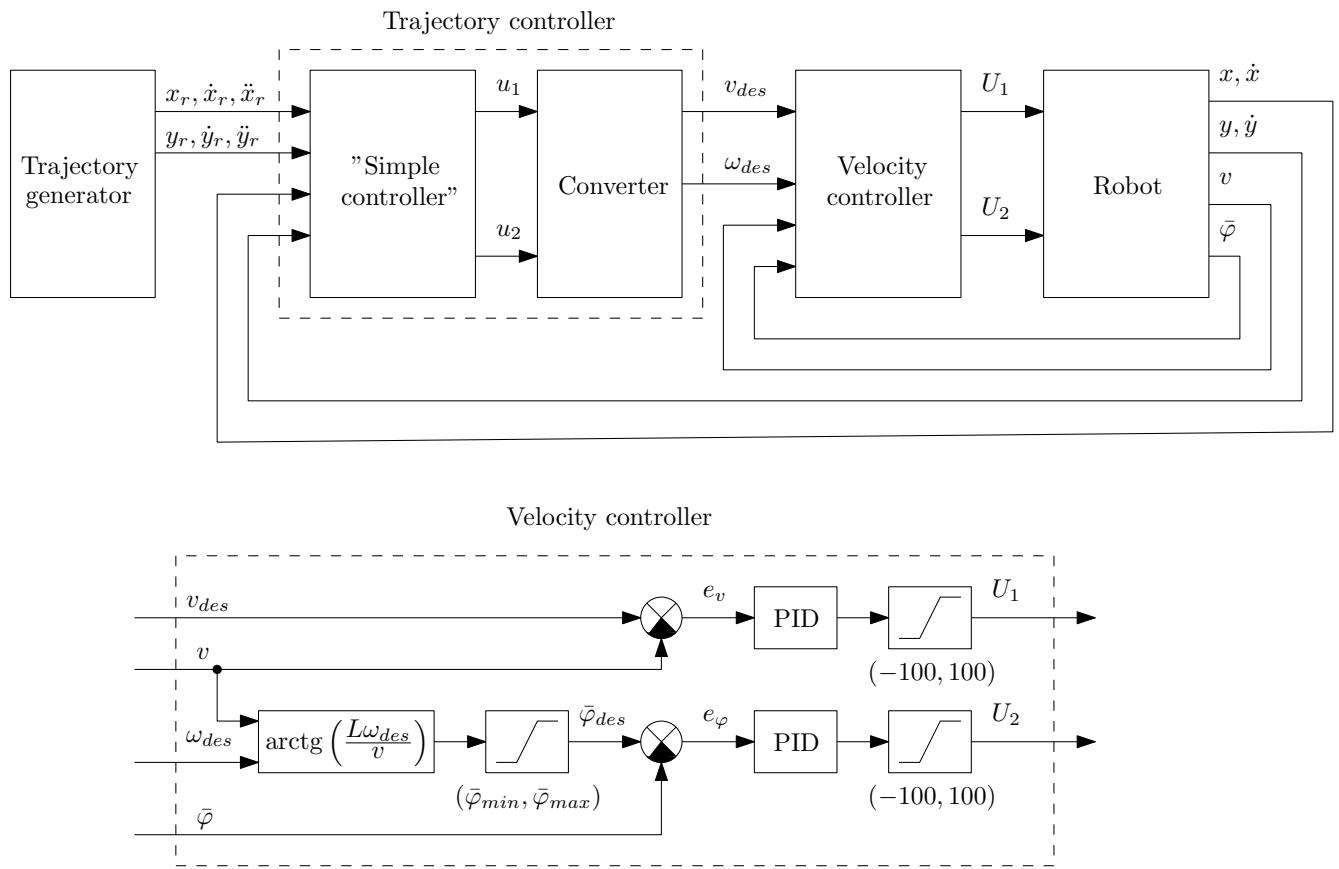
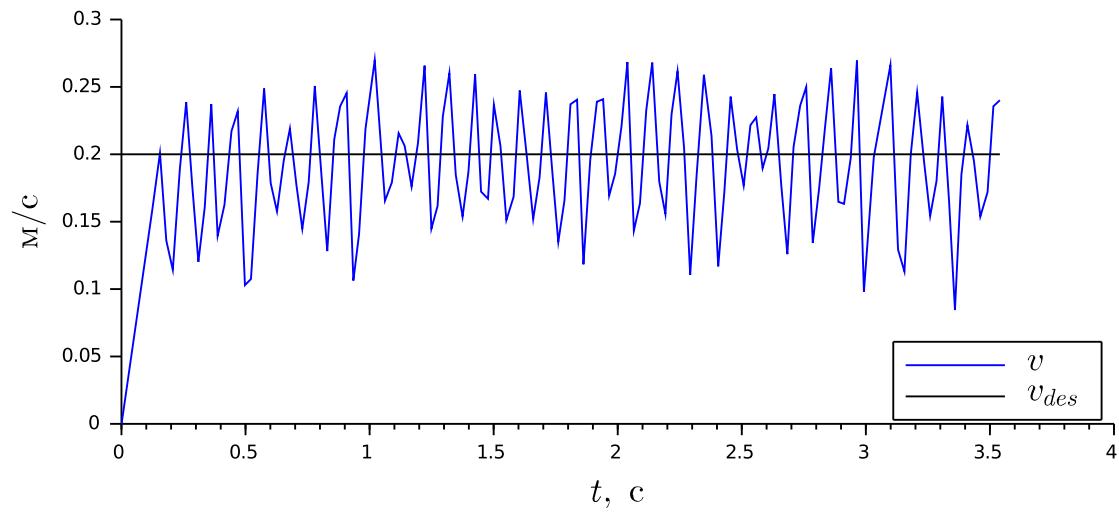
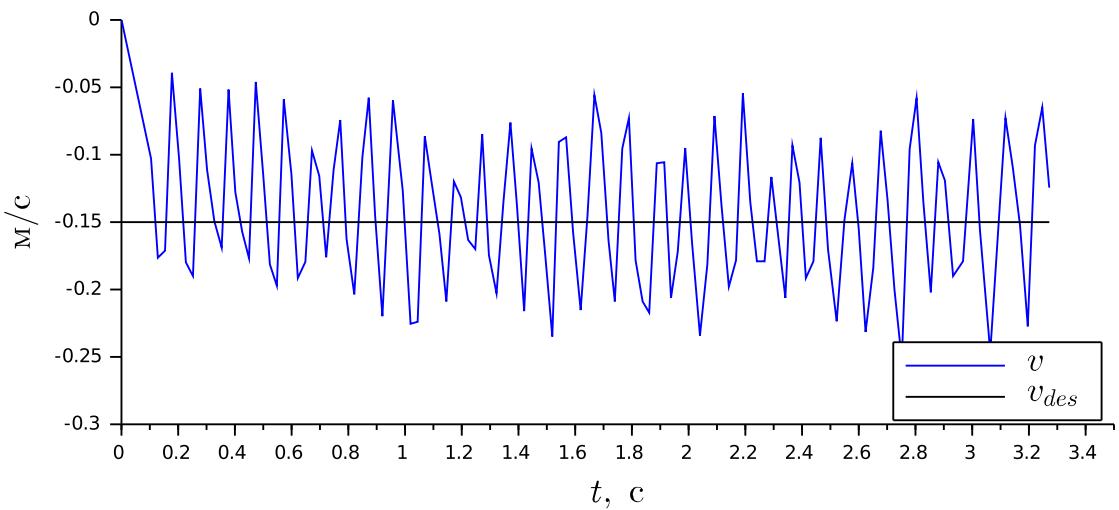


Рисунок 7 – Структура системы управления движением робота.



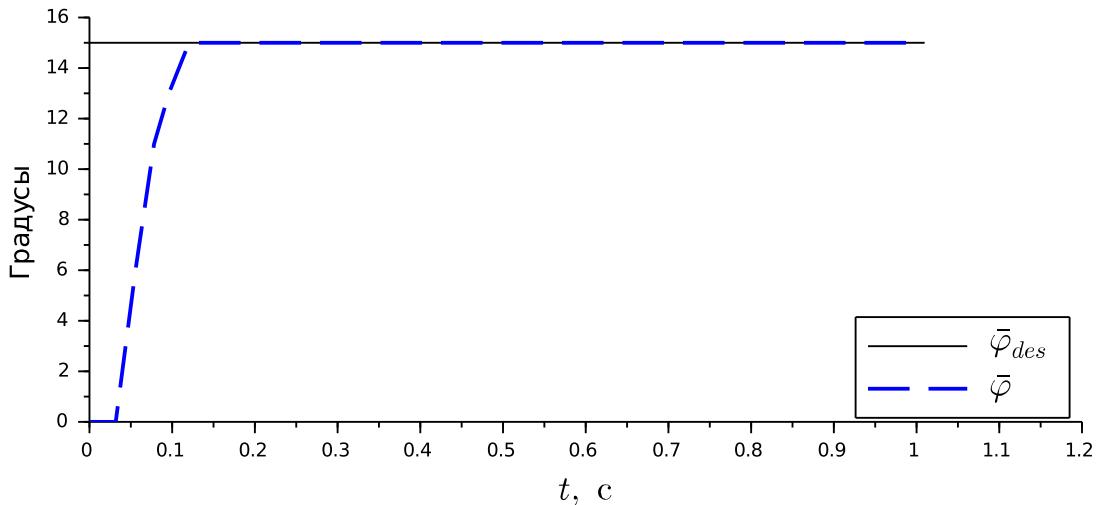
a)



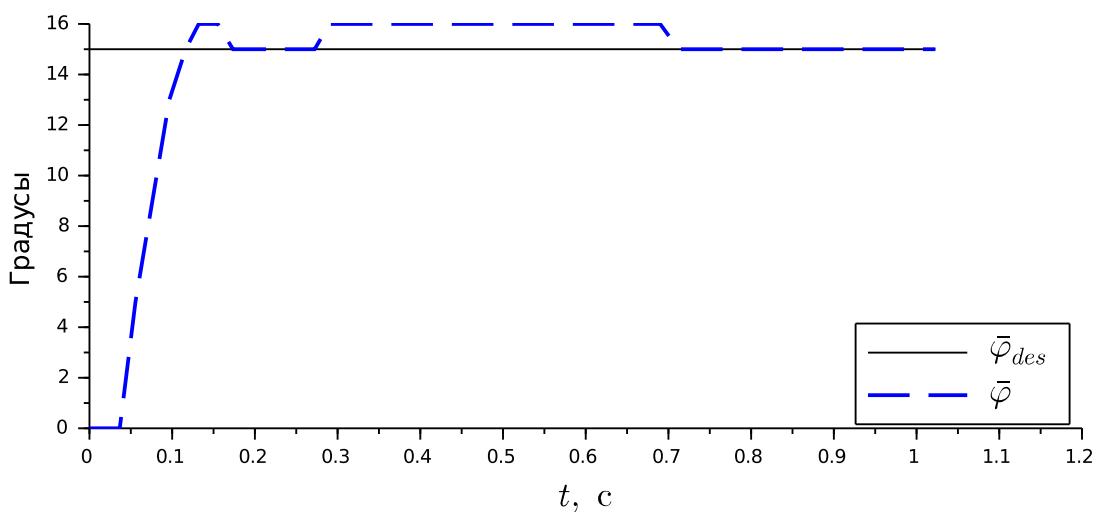
б)

Рисунок 8 – Переходные функции ПИД-регулятора, управляющего значением скорости v :

а — при $v_{des} > 0$, б — при $v_{des} < 0$.



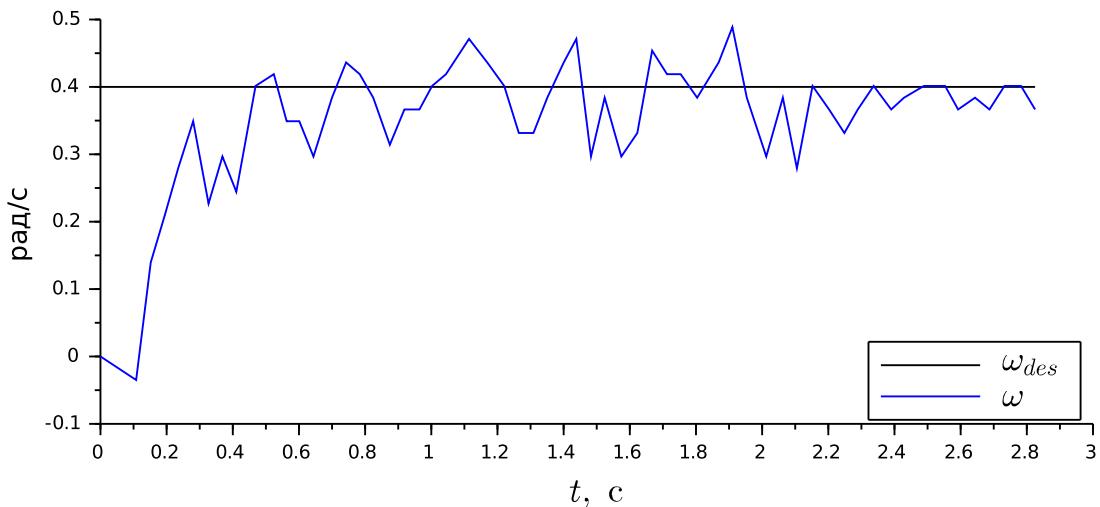
a)



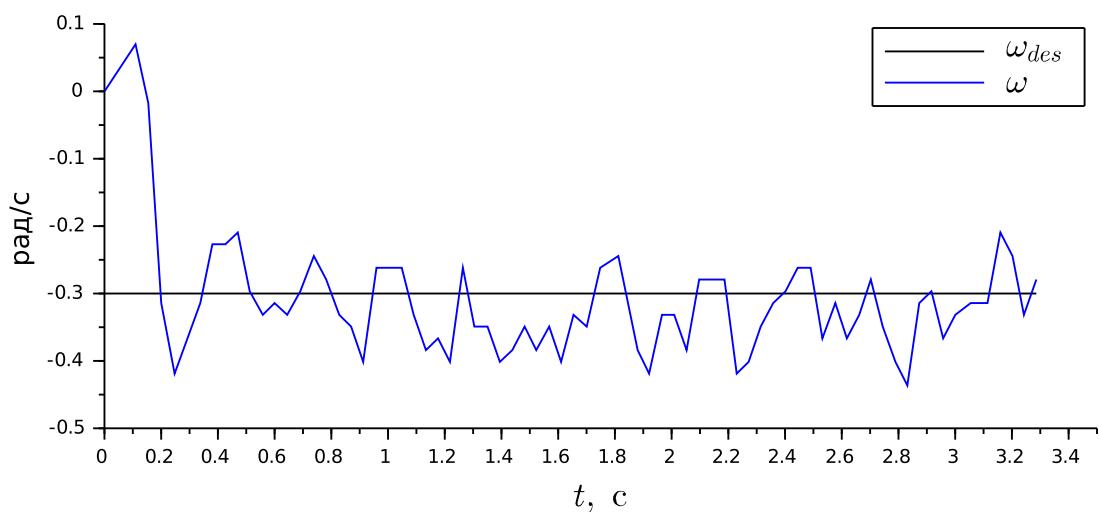
б)

Рисунок 9 – Переходные функции ПИД-регулятора, управляющего значением угла $\bar{\varphi}$:

а — при $v = 0$, б — при $v \neq 0$.



a)



б)

Рисунок 10 – Переходные функции системы управления угловой скоростью.

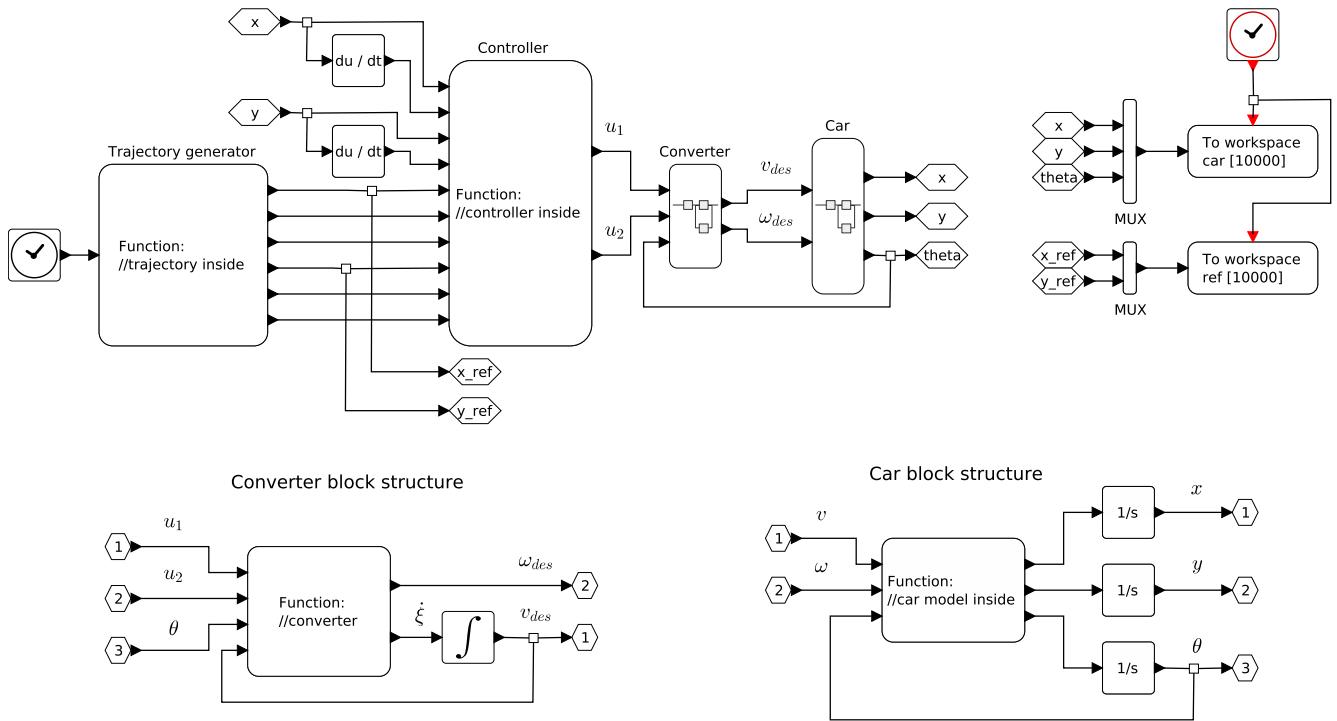


Рисунок 11 – Схема моделирования системы управления движением робота.

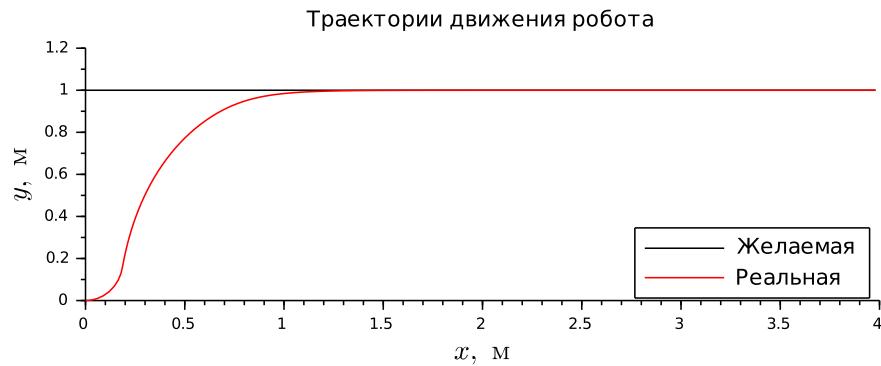


Рисунок 12 – Результаты запуска схемы моделирования.

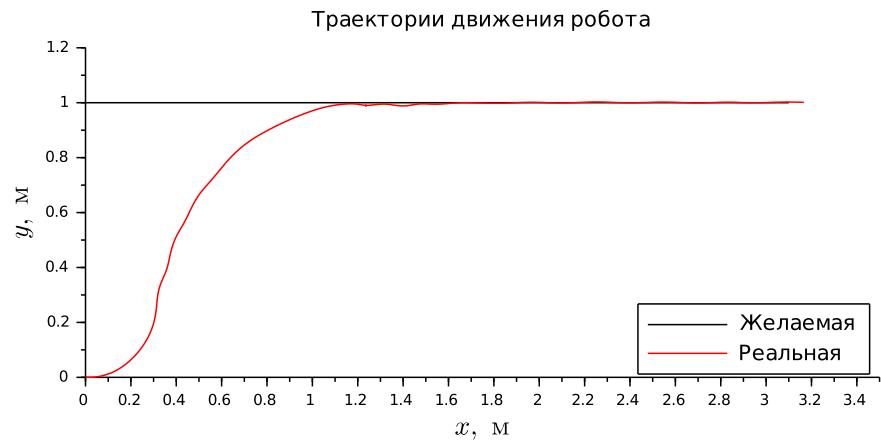
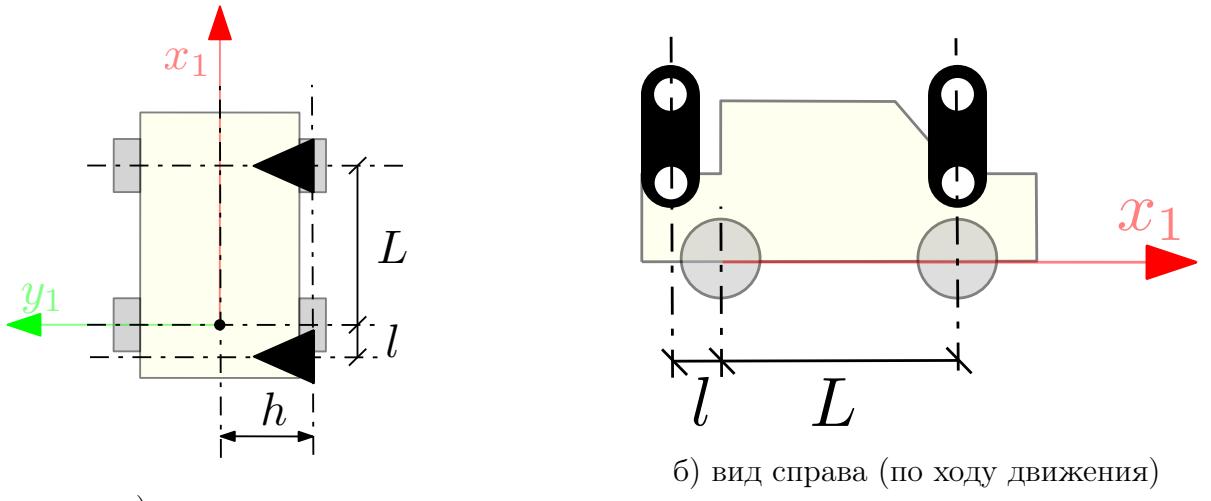


Рисунок 13 – Результат одного из экспериментов, проверяющего работоспособность системы управления движением.

3 Поиск парковочного места

Поиск парковочного места осуществляется по тому же принципу, что и в реальной жизни: робот-машинка движется параллельно припаркованному ряду автомобилей с постоянной скоростью v , одновременно сканируя пространство справа от себя. Сканирование олицетворяет собой измерение расстояния между роботом и объектами справа от него. Измерения (S_{front} и S_{rear}) производятся двумя ультразвуковыми дальномерами, расположение которых на роботе изображено на рисунке 14.



а) вид сверху

Рисунок 14 – Расположение ультразвуковых дальномеров.

Измерения показаний ультразвуковых дальномеров представляются в локальной СК робота точками (p_1 и p_2) (9). После чего преобразуются в глобальную СК путем перемножения матрицы однородных преобразования H и соответствующей измеренной точки (??)–(??)

$$p_1 = \begin{bmatrix} x_{p_1} \\ y_{p_1} \end{bmatrix} = \begin{bmatrix} L \\ -(S_{front} + d) \end{bmatrix}; \quad p_2 = \begin{bmatrix} x_{p_2} \\ y_{p_2} \end{bmatrix} = \begin{bmatrix} -l \\ -(S_{rear} + d) \end{bmatrix}. \quad (9)$$

где l — расстояние между задней осью и ультразвуковым дальномером, закрепленным на задней части робота; d — расстояние от продольной оси робота до сенсоров дальномеров.

$$H = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & x \\ \sin \theta & \cos \theta & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

$$P = H \cdot p = \begin{bmatrix} x_P \\ y_P \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & x \\ \sin \theta & \cos \theta & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_p \\ y_p \\ 0 \\ 1 \end{bmatrix} \quad (11)$$

В результате описанного выше сканирования строится карта. Построение карты происходит в несколько этапов:

- a) робот движется вдоль припаркованного ряда автомобилей на расстоянии от него не более, чем 0.3 метра;
- б) записываются измерения расстояний с дальномеров не более, чем 0.5 метров;
- в) из массива данных с картой удаляются все точки дальше, чем среднее арифметическое расстояний.

В результате этой процедуры на карте остаются проекции препятствий на плоскость дорожного полотна. На рисунке 15 показана зона проведения эксперимента, на рисунке 16 – пример полученной карты.*

После готовности карты, на ней разыскивается парковочное место подходящего размера, который задается постоянным параметром. Причем, линия параллельно которой будет парковаться робот выбирается как среднее от отфильтрованных данных карты. На нашей карте такое место определяется поиском двух соседних точек таких, что расстояние между ними вдоль линии движения робота должно быть больше постоянного параметра, о котором говорилось выше. Пример карты с выделенным парковочным местом показан на рисунке 17.

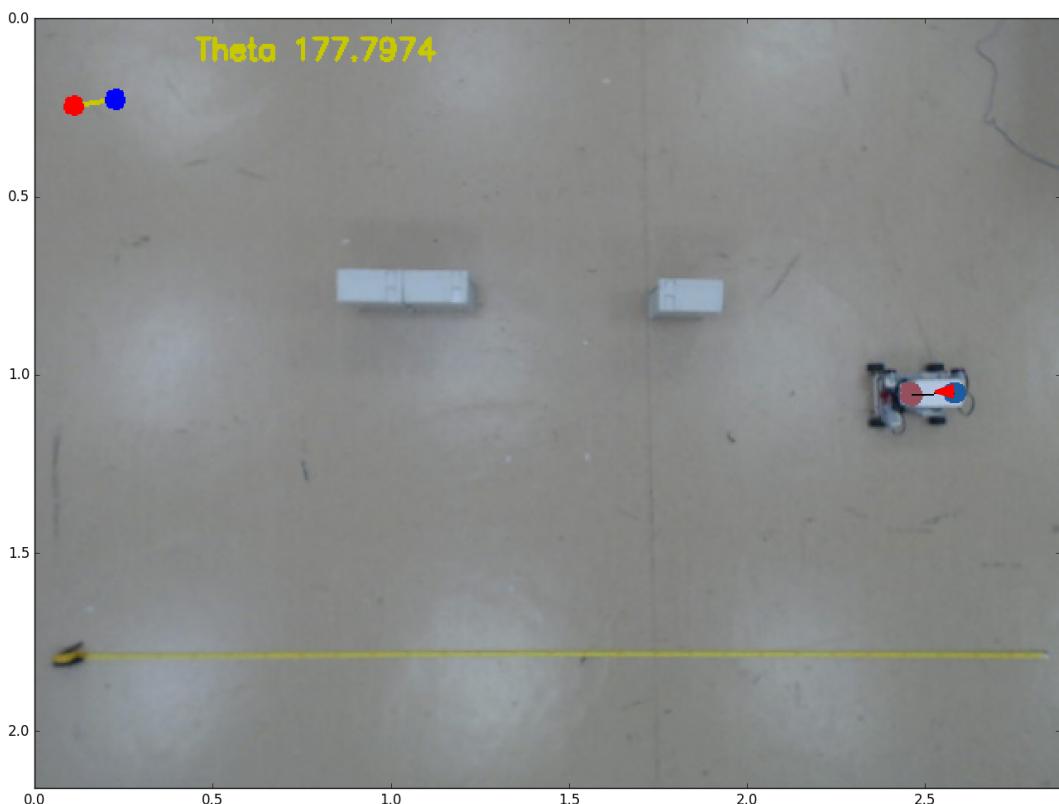
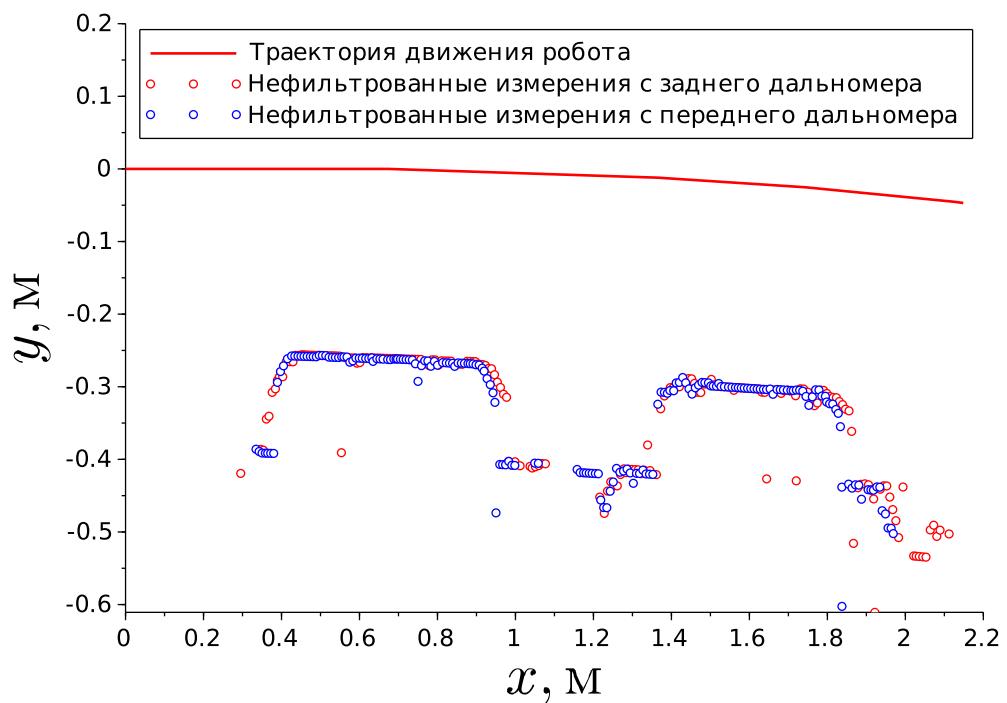
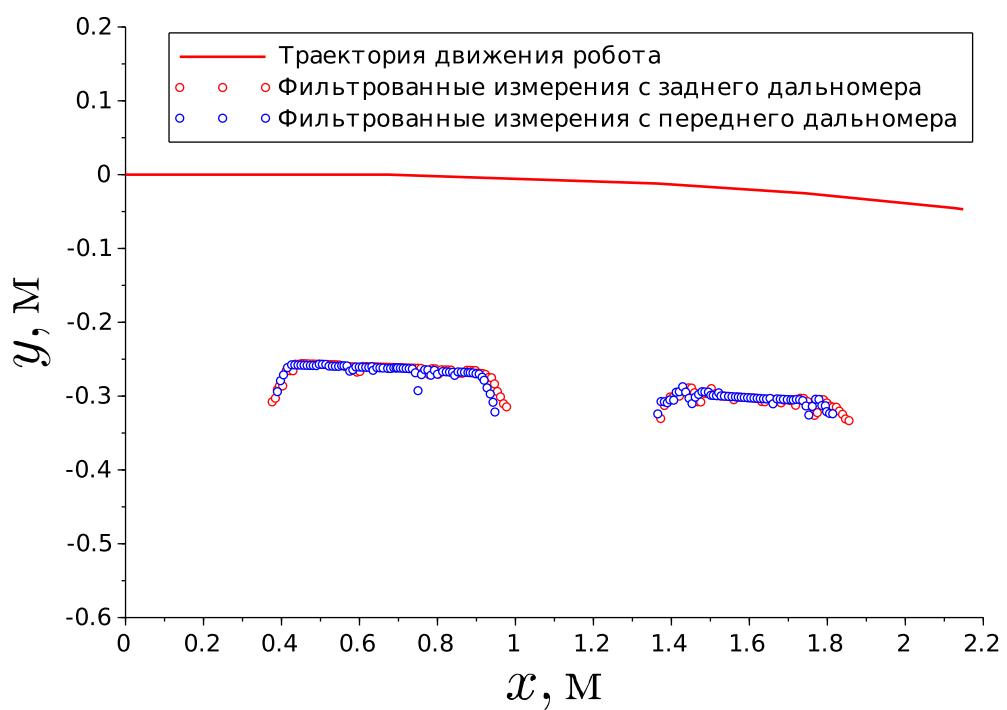


Рисунок 15 – Зона проведения эксперимента в реальности. Вид сверху.

* Представленные на рисунке 16 результаты фильтрации сканированных данных с дальномеров были получены не в том же эксперименте, что на рисунках 15 и 17



a)



б)

Рисунок 16 – Результаты сканирования: а — карта до фильтрации, б — карта после фильтрации.

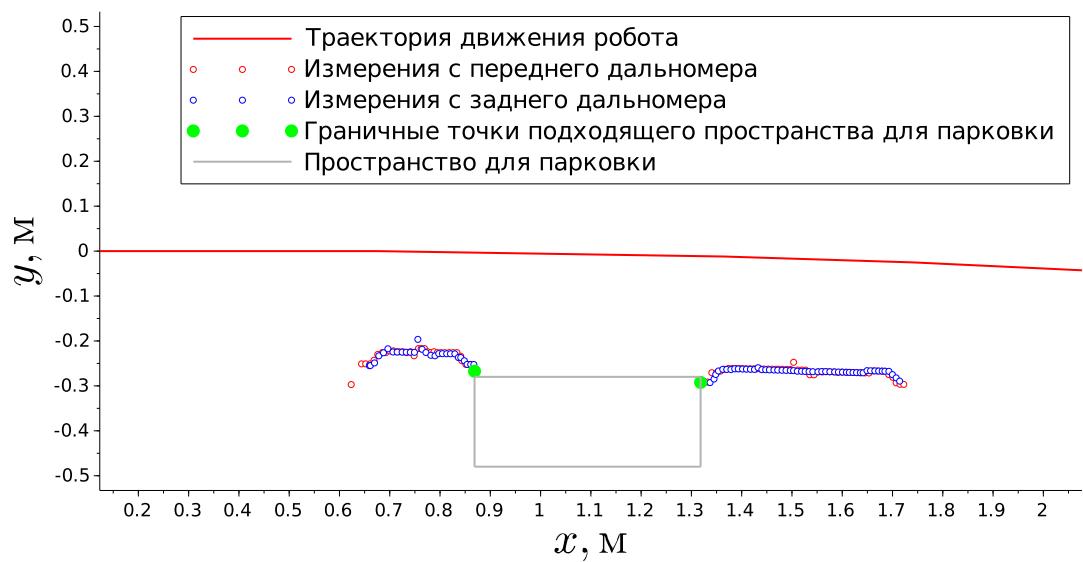


Рисунок 17 – Карта с выделенным парковочным местом.

4 Планирование траекторий движения

Для выполнения парковочного маневра система строит траекторию, показанную на рисунке 18. Координаты зеленых точек на нем становятся известными после выполнения этапа картирования, остальных — рассчитываются после с учетом ширины робота H , параметров δ_1 и δ_2 , задаваемых системе перед началом ее работы вручную, удвоенного минимального радиуса дуги, по которой может пройти робот, R и некоторых математических соотношений. Из последних стоит отметить лишь уравнения, совместное решение которых дает координаты (x_5, y_5) — координаты точки касания нижней из дуг и прямой, на которой лежит отрезок, ограниченный точками с координатами (x_3, y_3) и (x_5, y_5) :

$$\begin{cases} (x_7 - x_5)^2 + (y_7 - y_5)^2 = R^2, \\ \frac{y_4 - y_5}{x_4 - x_5} \cdot \frac{y_5 - y_7}{x_5 - x_7} = -1. \end{cases} \quad (12)$$

Задача же определения координат остальных точек авторам представляется тривиальной.

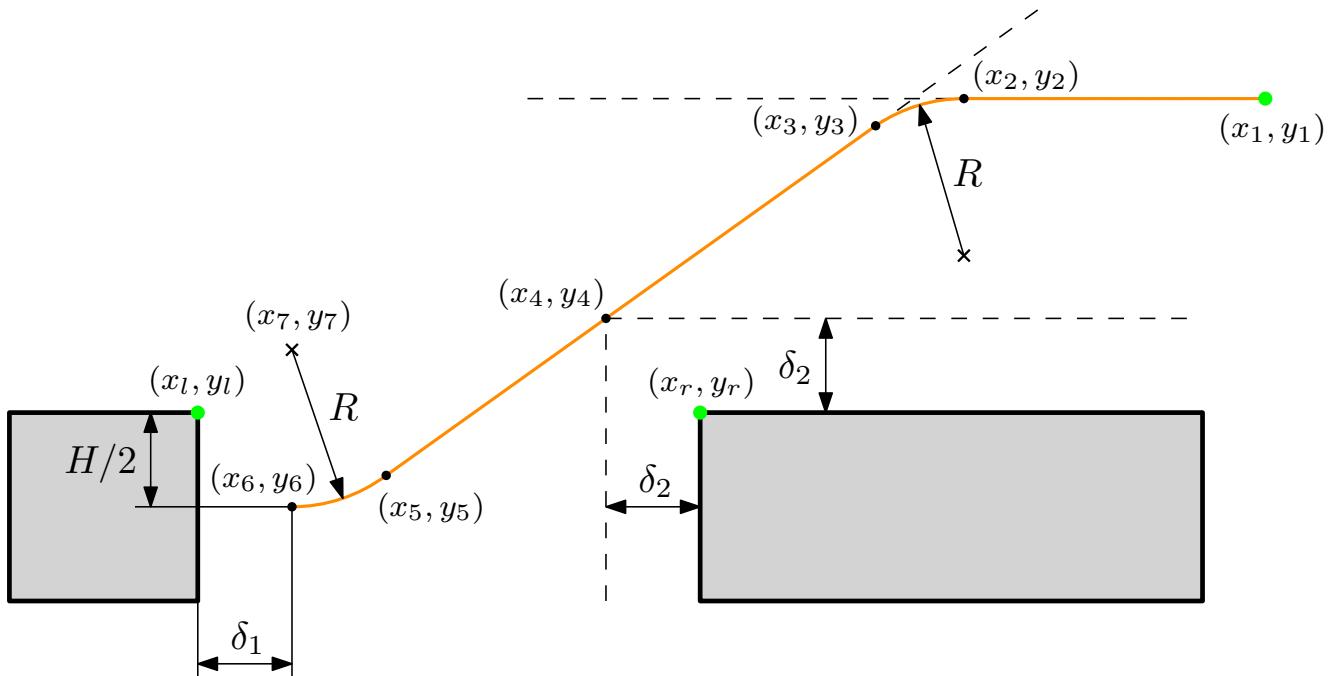


Рисунок 18 – Пояснения к принципам планирования траектории парковочного маневра.

Пример траектории, спланированной благодаря созданному авторами ПО, можно видеть на рисунке 19.

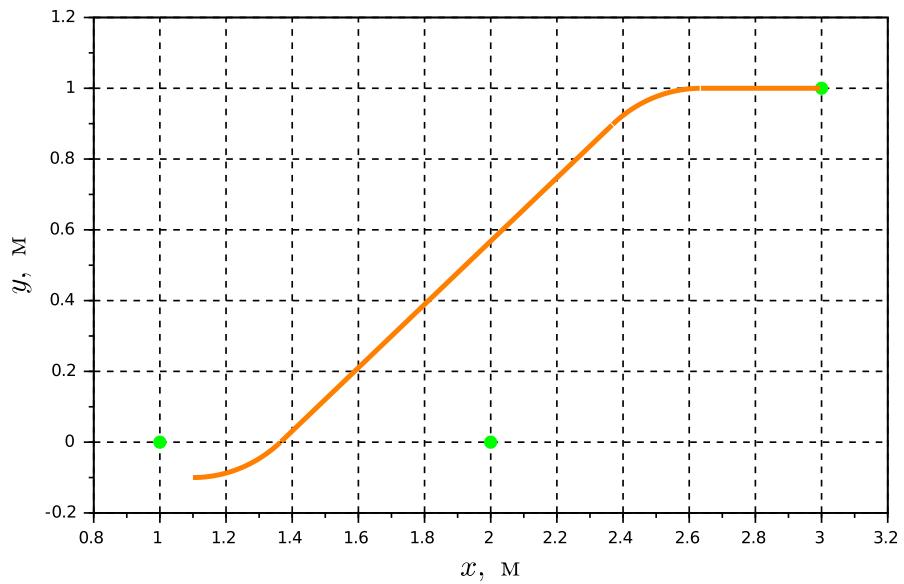


Рисунок 19 – Пример спланированной траектории движения.

Заключение

В результате проделанной работы авторы смогли решить все из подзадач, составляющих проблему автоматической парковки мобильного робота. Ими успешно:

- была собрана машинка с рулевым управлением и составлена ее математическая модель;
- были решены вопросы, касающиеся управления ее движением и локализации (локальной ориентации): в первом из случаев была построена система управления, дающая роботу возможность двигаться по желаемой траектории, а во втором — подобраны датчики и выбраны алгоритмы расчета, позволяющие получать хорошую оценку положения робота в пространстве, что, в свою очередь, было проверено с помощью сторонней системы технического зрения;
- был разработан метод (алгоритм) картирования окрестностей предполагаемого места парковки и поиска последнего;
- был синтезирован алгоритм планирования траектории парковочного движения.

При этом, однако, из-за нехватки (или нерационального использования) времени система целиком протестирована не была.

Исходный код данного отчета и относящихся к проекту программ (расчетных, управляющих и проч.) доступны по адресу <https://github.com/kirillin/parking-lego-car>.

Приложение А
(обязательное)

Описание экспериментов по проверке состоятельности использованных методов локализации робота с помощью сторонней системы технического зрения

Система технического зрения (СТЗ) представляет собой веб-камеру Logitech C920 FullHD закрепленную над зоной проведения экспериментов на высоте 2.85 метра. Камера подключена к стационарному компьютеру, не связанному с роботом, на котором запускается программа на языке программирования Python. Для решения задач СТЗ используется библиотека OpenCV. На роботе закреплена плоская панель с двумя маркерами: синим – над задней осью и красным – в передней части робота. Центр масс синего маркера располагается над точкой C (см. рисунок 4).

СТЗ работает по следующему алгоритму:

- a) каждый из кадров видео-потока представляется в цветовом пространстве HSV (Hue, Saturation, Value);
- б) из кадра выделяются области с красным и синим цветами;
- в) вычисляются координаты маркеров в СК кадра в пикселях;
- г) координаты переводятся из пикселей в метры в соответствии со следующим выражением:

$$d_m = d_{px} \frac{D_m}{D_{px}} \quad (\text{A.1})$$

где d_m и d_{px} – координата или размер объекта в метрах и пикселях, соответственно; D_m и D_{px} – длина диагонали кадра в метрах и пикселях;

- д) то место, где стоит робот при запуске СТЗ, принимается за базовую СК (она же является базовой и для одометрии);
- е) по расположению красного маркера относительно синего рассчитывается ориентация робота;
- ж) все последующие координаты робота и его ориентации выражаются в базовой СК;
- з) значения координат и ориентация робота записываются в файл и выводятся на кадре.

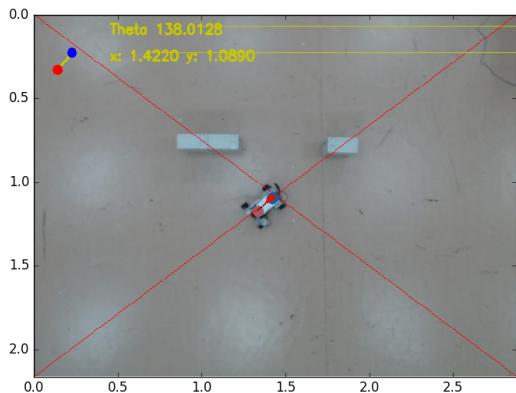
Для проверки СТЗ был проведен эксперимент:

- а) при помощи отвеса и двух человек, камера была установлена так, чтобы оптическая ось была строго перпендикулярна плоскости пола зоны эксперимента;
- б) на полу были отмечены четыре точки и измерены расстояния между ними;
- в) последовательно в каждую точку ставился робот так, чтобы точка C (или что тоже самое – центр масс синего маркера) была над отмеченной на полу точкой;

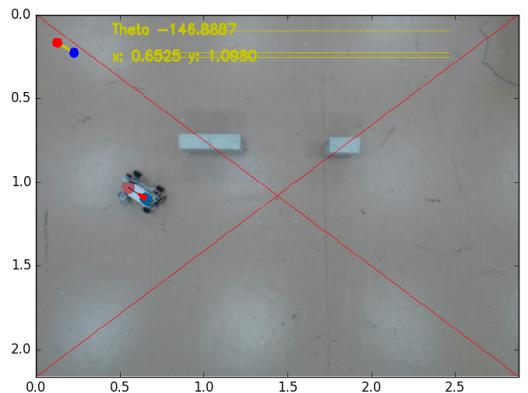
г) фиксировался результат в виде скриншота с отмеченными на нем координатами робота.

Данные, полученные при этом с СТЗ приведены на рисунке А.1, сами же результаты проверки — в таблице А.1.

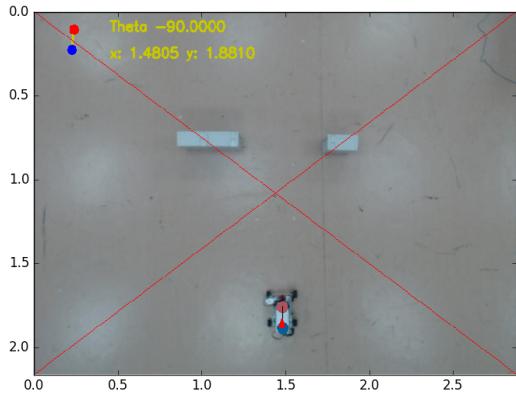
Смотря на результаты проверки разработанной в этой работе СТЗ можно заключить о невозможности достоверного определения траектории движения робота. Так как, во-первых, изображение получаемое с камеры имеет дисторсию, что наглядно видно на рисунке 15 (растянутая в нижней части кадра рулетка имеет видимый изгиб), а также из таблицы А.1 (расстояния между точкой в центре кадра и близкими к нему точками имеют меньшую ошибку, чем расстояние до точки у края) и, во-вторых, проверка СТЗ проводилась «на скорую руку», в связи с чем были допущены, возможно, существенные ошибки при экспериментах.



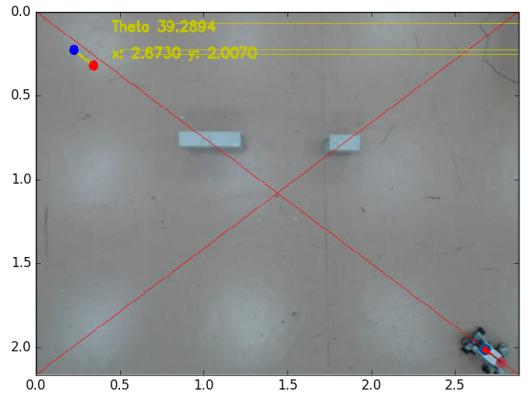
а) (1.422, 1.089)



б) (0.6525, 1.098)



в) (1.4805, 1.881)



г) (2.673, 2.007)

Рисунок А.1 – Проверка СТЗ.

При этом траектории движения робота, полученные с помощью СТЗ и его собственных бортовых средств локализации (одометрии и датчика-гироскопа), при сравнении оказываются достаточно похожими (см. рисунки А.2 и А.3).

Таблица А.1 – Результаты проверки СТЗ

Расстояние между точками	СТЗ, м	Линейка, м	Ошибка, м
а и б	0.7695	0,75	0.0194
а и в	0.7941	0,75	0.0441
а и г	1.5516	1,44	0.1116

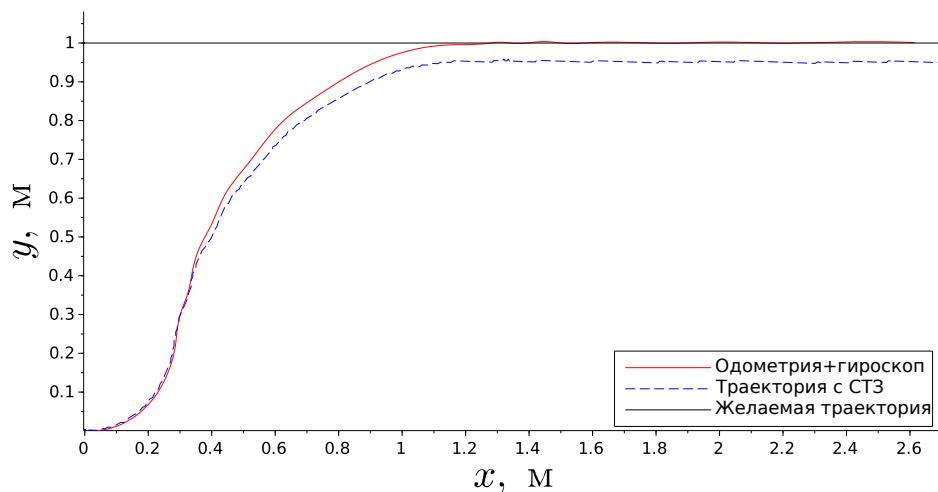


Рисунок А.2 – Траектория движения робота, полученная из СТЗ

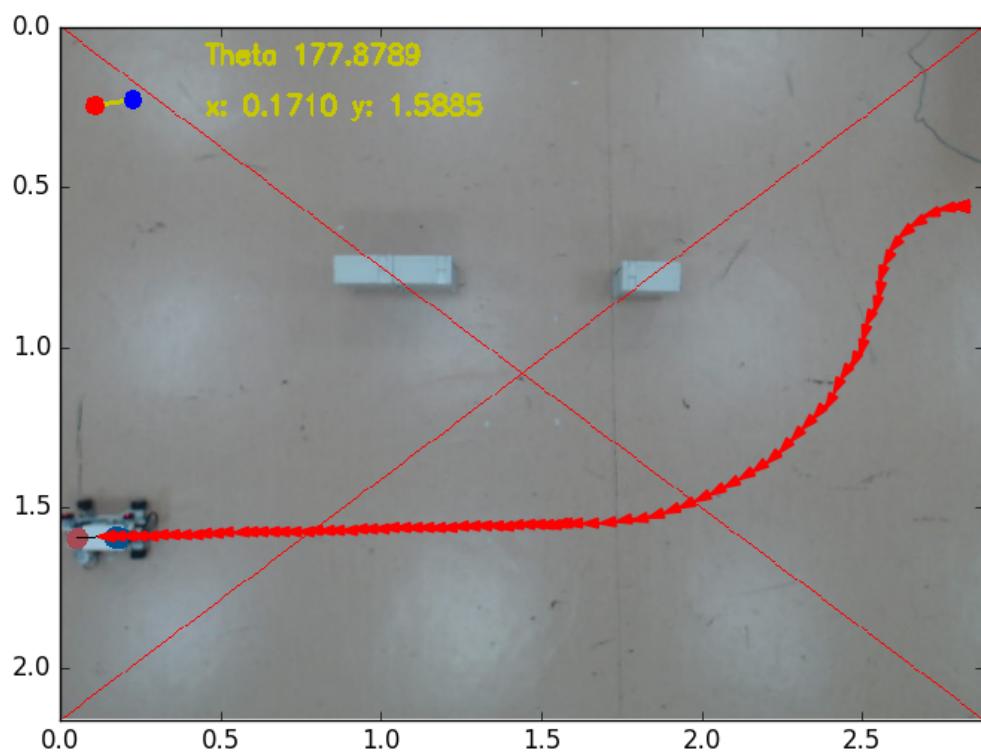


Рисунок А.3 – Траектория движения робота. Вид с камеры

Возможным решением проблем, возникших с СТЗ, авторам работы видится выполнение следующих действий:

- а) обязательная калибровка камеры (устранение дисторсии и проч. средствами OpenCV);
- б) закрепление камеры так, чтобы она не смешалась с течением времени (плоскость пола была строго перпендикулярна оптической оси камеры);
- в) произведение насколько это возможно более точных измерений высоты (от пола до камеры);
- г) для перевода размеров объектов в кадре из пикселей в метры использование угла обзора камеры (сейчас используются вручную измеренные величины);
- д) проведение описанного выше (или еще более точного) эксперимента с проверкой СТЗ, включающего в себя помимо всего прочего исследование возвращаемой СТЗ ориентации робота.

Приложение Б

(обязательное)

Уравнения используемых элементарных траекторий

В работе используются траектории в виде отрезка прямой и дуги окружности с центральным углом, на который она опирается, меньшим π , параметризованные временем, (см. рисунок Б.1) и описываемые следующими выражениями (v — желаемая скорость движения по траектории, t_{end} — время, необходимое для прохождения всей траектории):

- отрезок прямой:

$$\begin{cases} x_r(t) = x_0 + v_x t, \\ y_r(t) = y_0 + v_y t, \end{cases} \quad \begin{cases} \dot{x}_r(t) = v_x, \\ \dot{y}_r(t) = v_y, \end{cases} \quad \begin{cases} \ddot{x}_r(t) = 0, \\ \ddot{y}_r(t) = 0, \end{cases} \quad (\text{Б.1})$$

$$\gamma = \text{atan}2(y_1 - y_0, x_1 - x_0), \quad v_x = v \cos \gamma, \quad v_y = v \sin \gamma, \quad (\text{Б.2})$$

$$t_{end} = \frac{\sqrt{(y_1 - y_0)^2 + (x_1 - x_0)^2}}{v}; \quad (\text{Б.3})$$

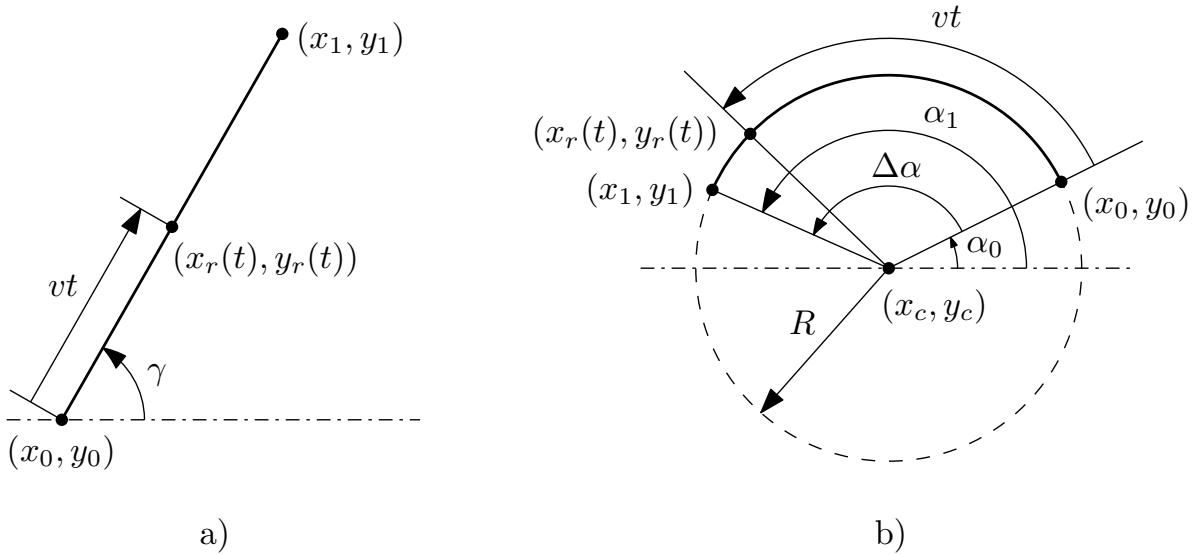


Рисунок Б.1 – Рассматриваемые траектории: а — отрезок прямой, б — дуга окружности.

- окружность:

$$\begin{cases} x_r(t) = x_c + R \cos \alpha, \\ y_r(t) = y_c + R \sin \alpha, \end{cases} \quad \begin{cases} \dot{x}_r(t) = -R\omega \sin \alpha, \\ \dot{y}_r(t) = R\omega \cos \alpha, \end{cases} \quad \begin{cases} \ddot{x}_r(t) = -R\omega^2 \cos \alpha, \\ \ddot{y}_r(t) = -R\omega^2 \sin \alpha, \end{cases} \quad (\text{Б.4})$$

$$R = \sqrt{(x_0 - x_c)^2 + (y_0 - y_c)^2}, \quad \alpha_0 = \text{atan}2(y_0 - y_c, x_0 - x_c), \quad (\text{Б.5})$$

$$\alpha = \alpha_0 + \omega t, \quad \alpha_1 = \text{atan}2(y_1 - y_c, x_1 - x_c), \quad \omega = \frac{v}{R} \text{sign}(\Delta\alpha), \quad (\text{Б.6})$$

$$t_{end} = \frac{|\Delta\alpha|}{\omega}, \quad \Delta\alpha = \begin{cases} \alpha_1 - \alpha_0, & \alpha_1 - \alpha_0 \in (-\pi; \pi), \\ \alpha_1 - \alpha_0 - 2\pi, & \alpha_1 - \alpha_0 > \pi, \\ \alpha_1 - \alpha_0 + 2\pi, & \alpha_1 - \alpha_0 < -\pi. \end{cases} \quad (\text{Б.7})$$