

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**РОБОТОТЕХНИЧЕСКАЯ СИСТЕМА С ТЕХНИЧЕСКИМ ЗРЕНИЕМ
ДЛЯ МАНИПУЛЯЦИИ ПОДВИЖНЫМИ ОБЪЕКТАМИ**

Автор Артемов Кирилл _____
(Фамилия, Имя, Отчество) _____
(Подпись) _____

Направление подготовки (специальность) 15.04.06 _____
(код, наименование)
Мехатроника и робототехника _____

Квалификация магистр _____
(бакалавр, магистр)* _____

Руководитель Капитонов А.А., к.т.н. _____
(Фамилия, И., О., ученое звание, степень) _____
(Подпись) _____

К защите допустить

Зав. кафедрой Бобцов А.А., профессор, д.т.н. _____
(Фамилия, И., О., ученое звание, степень) _____
(Подпись) _____

“ ____ ” 20 18 г.

Санкт-Петербург, 20 18 г.

Содержание

Введение	5
1 Обзор существующих решений	7
2 Разработка системы управления манипулятором KUKA YouBot	13
2.1 Описание манипулятора	13
2.2 Кинематическая модель манипулятора	15
2.2.1 Решение задач о положении манипулятора	15
2.2.2 Решение задач о скорости звеньев манипулятора	21
2.3 Система кинематического управления	28
3 Планирование траекторий движения манипулятора	31
3.1 Постановка и описание задачи	31
3.2 Распределение допустимых скоростей в рабочем пространстве . .	34
3.3 Планирование траектории в пространстве обобщенных координат	35
3.3.1 Движение из точки в точку	35
3.3.2 Движение через последовательность точек	38
3.4 Планирование траектории в пространстве координат схватов . . .	42
4 Разработка системы технического зрения	46
4.1 Постановка и описание задачи	47
5 Результаты математического моделирования	54
5.1 Результаты планирования траектории	54
5.2 Моделирование системы управления манипулятором	58
5.3 Результаты работы системы технического зрения	59
Заключение	62

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

КСУИ.106.4235.001 ПЗ

Изм.	Лист	№ докум.	Подп.	Дата
Разраб.	Артемов К.			
Пров.	Капитонов А.А.			
Н. контр.	Быстров С.В.			
Утв.				

Робототехническая система с
техническим зрением для
манипуляции подвижными
объектами
Пояснительная записка

Лит. Лист Листов
3 69
Университет ИТМО
Кафедра СУиИ
гр. Р4235

Список использованных источников

63

Приложение А Программный код в Scilab

65

Инв. № подп.	Подп. и дата	Бзак. инв. №	Инв. № документа	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

КСУИ.106.4235.001 ПЗ

Лист

4

Введение

Основная цель работы — схватывание объектов с вращающегося стола при помощи пятизивенного манипулятора KUKA YouBot с использованием системы технического зрения в условиях соревнований RoboCup@Work.

Мотивация

Одной из задач, которая стоит перед студенческим конструкторским бюро кафедры Систем управлений и информатики это превзойти результаты других команд на ежегодных международных соревнованиях RoboCup.

Цель этих соревнований состоит в продвижении робототехники и исследовании искусственного интеллекта на популярных и идеально простых, но нетривиальных задачах. В 2017 году на RoboCup проводились соревнования в шести основных категориях: футбол роботов (Soccer), роботы-спасатели (Rescue Leagues Homepage), домашние роботы (@Home), промышленные роботы (@Work), инфраструктура снабжения (RoboCup Logistics League) и младшая лига (RoboCupJunior) для школьников.

Суть соревнований в категории AtWork — создать симуляцию реального производственного цикла [1].

На производстве есть детали, которые необходимо доставить из пункта А в пункт Б. Классический способ это сделать — использовать конвейер.

Конвейеры хорошо зарекомендовали себя в горнодобывающей и перерабатывающей промышленности (подъем и транспортирование горной массы), в сельском хозяйстве (сортировка, транспортирование, упаковка сельскохозяйственной продукции, конвейерная разделка туш скота), в производстве продовольственных и непродовольственных товаров (от газировки и пачек с макаронами, до хозяйственных товаров и электроприборов), и в других областях человеческой жизни.

Однако у конвейеров есть свои недостатки, поэтому многие коллективы исследователей работают над разработкой иных методов. Один из главных минусов конвейера — нельзя быстро и дёшево изменить его конфигурацию. Сегодня аналог классическому конвейеру — мобильные роботы, с помощью которых

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

можно решать те же самые задачи, более целенаправленно: робота можно перевести в любую точку и быстро переконфигурировать, задав ему новую цель.

Если мобильную платформу оснастить манипулятором с системой технического зрения, то полученная робототехническая система сможет выполнять более сложные задачи, например, собирать с конвейеров, столов или полок определенные детали и перемещать их в указанный пункт назначения.

Разрабатываемая робототехническая система в повседневной жизни может быть применена, например, в аэропортах, для погрузки и разгрузки багажа, в библиотеках для автоматизации манипуляций с книгами, на железнодорожных станциях для транспортировки грузов между грузовыми вагонами и колесными грузовиками, в портах для эффективной организации грузов на палубах и в трюмах кораблей и многие другие.

Задачи

Одно из испытаний, которое предстоит роботу на соревнованиях в лиге AtWork — заключается в оценке способности робота манипулировать движущимися объектами, которые перемещаются на конвейерной ленте или вращающемся столе (Conveyor Belt Test / Rotating Table Test). Это испытание требует хорошей точности управления манипулятором и качественной системы технического зрения.

В этой работе ставятся следующие задачи:

- а) Построить математическую модель манипулятора KUKA Youbot;
- б) Синтезировать систему управления манипулятором;
- в) Спланировать необходимые для захвата движущихся объектов траектории;
- г) Разработать систему технического зрения для оценки положения и скорости движущегося объекта;
- д) Провести моделирование разработанной системы.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

1 Обзор существующих решений

Для решения поставленных задач нужен системный подход. Множество компаний и исследовательских групп предлагают различные решения как аппаратной части, так и программной.

Компания *Rethink Robotics*, известная своими манипуляторами Sawyer и Baxter [2] для работы на заводах совместно с людьми. Робот Baxter представлен на рисунке 1.1.



Рисунок 1.1 – Робот Baxter

Отличительной особенностью этих роботов является изменение их конфигурации под нужные технологические процессы происходит в виде «обучения» в специальном графическом приложении Intera Studio 1.2, а не классического программирования промышленных роботов. Особенностью технологий *Rethink Robotics* являются роботы, которые экономически эффективны и могут работать совместно с коллегами-людьми, выполняя, например, опасные для людей работы.

По периметру «головы» робота располагаются датчики, которые дают возможность ему адаптироваться к окружающей среде, определять наличие людей рядом. В «руках» робота установлены инфракрасные датчики. Доступно

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

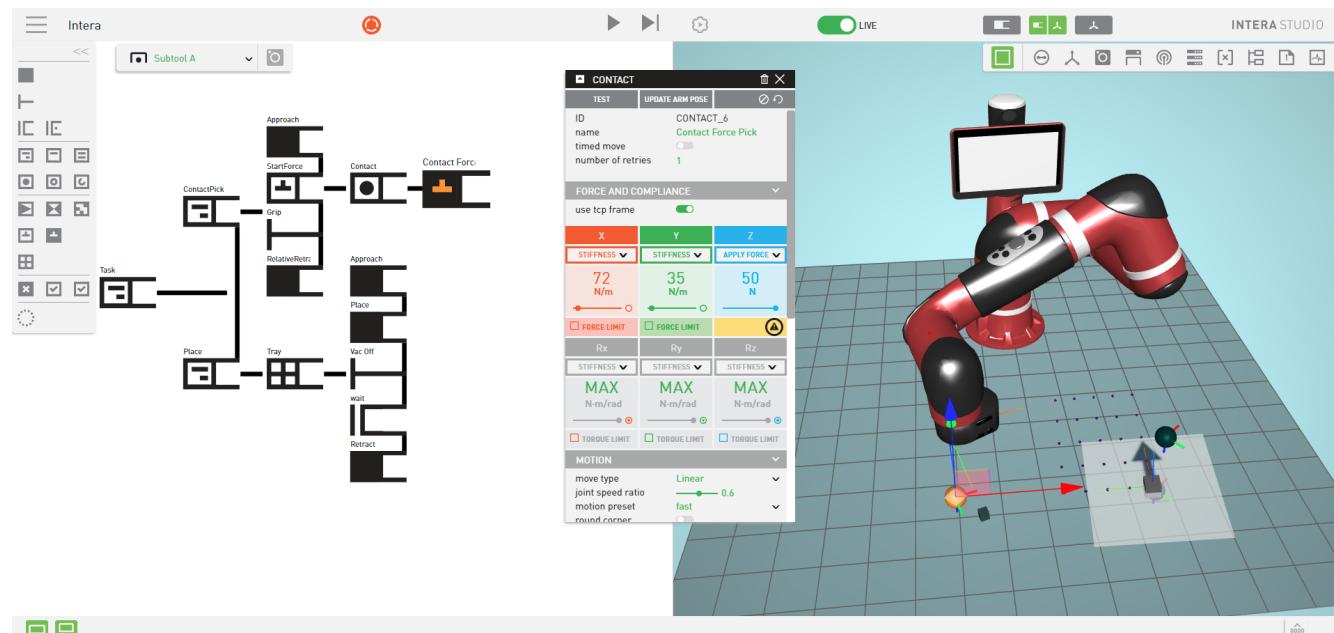


Рисунок 1.2 – Окно приложения Intera Studio для программирования роботов компании Rethink Robotics

большое разнообразие схватов.

Многие университеты используют робота Baxter в рамках своих курсов по робототехнике [3], чтобы дать студентам возможность использования современной технологии робототехники для практического применения в реальном мире. Baxter, в отличие от традиционных роботов-манипуляторов, не требует установки вокруг себя заграждений, поэтому студенты могут работать с ним в непосредственной близости, не беспокоясь о безопасности.

Robotnik — испанская компания, специализирующаяся на разработке роботов и R&D в области робототехники. Один из их роботов RB-1, это модульный мобильный манипулятор, разработанный с возможностью расширения. Применяется для R&D, AAL (Ambient Assisted Living) и удаленного управления. Изображение робота показано на рисунке 1.3.

Манипулятор имеет антропоморфную конфигурацию с семью степенями свободы и двух или трехпальцевый захват. Что касается датчиков, в мобильной платформе RB-1 установлен лидар Hokuyo URG-04LX-UG01, для задач навигации и система компьютерного зрения с одной из RGBD камер Microsoft Kinect или ASUS Xtion PRO Live.

Программное обеспечение с открытыми исходными кодами, управление

Инв. № подл.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата



Рисунок 1.3 – Робот испанской компании Robotnik RB-1

реализуется посредством ROS (Robotic Operation System). Компания поставляет роботов в нескольких модификациях [4].

PAL Robotics — испанская команда вдохновленных инженеров, которые разрабатывают, изготавливают и модифицируют роботов. Их робот TIA Go (Take It And Go) — мобильная исследовательская платформа, предназначенная для навигации, манипуляции и взаимодействия с окружающим миром. Оснащена дифференциальной платформой.

Робот включает в себя систему технического зрения, подъемный торс и манипулятор, обеспечивающие большое рабочее пространство. Полностью совместим с ROS и, кроме того, поставляется с множеством готовых функциональных возможностей, таких как: мультисенсорная навигация, планирование движения без конфликтов, обнаружение людей, лиц и объектов, распознавание речи и синтез. Общий вид робота представлен на рисунке 1.4 [4].

Fraunhofer IPA — немецкая компания, выпускающая целую линейку роботов, для исследований. Один из них Care-O-bot 4, изображение которого представлено на рисунке 1.5.

Самое интересное в Care-O-bot 4 (кроме двух рук и изысканных движе-

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



Рисунок 1.4 – Робот испанской компании PAL Robotics TIAGO



Рисунок 1.5 – Робот немецкой компании Fraunhofer IPA Care-O-bot 4

ний) — его модульность. Робот состоит из пяти модулей: основание, туловище, руки, кольцо датчиков и голова. Каждый из модулей может быть отсоединен, в зависимости от задач.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Робот может найти применение как дома, в качестве повседневного помощника, так и в муниципальных учреждениях для оказания различных услуг [4].

Moley Robotics — это робототехническая компания, основанная Марком Олейник в 2015 году для создания сервисных роботов для использования на кухне. Прототип робота Moley Robotic Kitchen — это роботизированная кухня. Изображение прототипа показано на рисунке 1.6

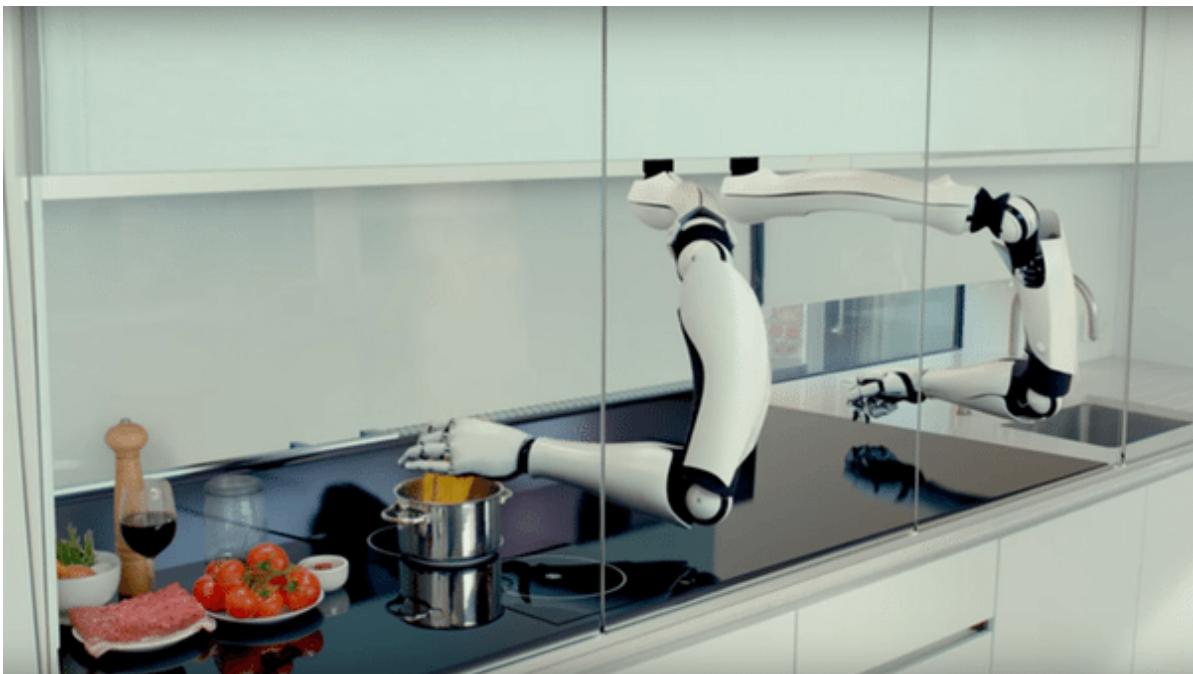


Рисунок 1.6 – Робот компании Moley Robotic Kitchen

Moley Robotic Kitchen включает в себя два манипулятора со схватами в виде кистей рук, оснащенные тактильными датчиками, кухонный стол с духовкой, электрической плитой, посудомоечной машиной и сенсорным экраном. Манипуляторы могут захватывать и взаимодействовать с большинством кухонных принадлежностей [5].

Робот умеет запоминать движения человека, когда тот готовит некоторое блюдо и затем, воспроизводить их при приготовлении того же блюда самостоятельно.

В текущем прототипе пользователь управляет установкой с помощью встроенного сенсорного экрана или приложения для смартфона с готовыми ингредиентами, приготовленными заранее и помещенными в определенные места. Целью Moley Robotic в будущем является предоставление пользователю воз-

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

можности выбирать из библиотеки более 2000 записанных рецептов.

Исследовательский робот немецкой компании *KUKA YouBot* — образовательный робот, специально разработанный для разработки и обучения в области мобильных манипуляторов. Он состоит из омни-платформы, пятиступенчатого манипулятора и двухпальцевого схвата. *KUKA YouBot* — это платформа с открытым исходным кодом, представлена на рисунке 1.7.



Рисунок 1.7 – Робот компании *KUKA YouBot*

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № документа	Подп. и дата

2 Разработка системы управления манипулятором KUKA YOUTBOT

Раздел посвящён теоретическим выкладкам, позволяющим схвату манипулятора перемещаться по заданной траектории. Сначала определяются основные кинематические соотношения, описывающие манипулятор KUKA YouBot (далее манипулятор) без учета действующих на него сил. Они позволяют определить положение манипуляционного механизма в пространстве, а также скорости и ускорения всех звеньев. Синтезируется система управления с использованием кинематических соотношений.

2.1 Описание манипулятора

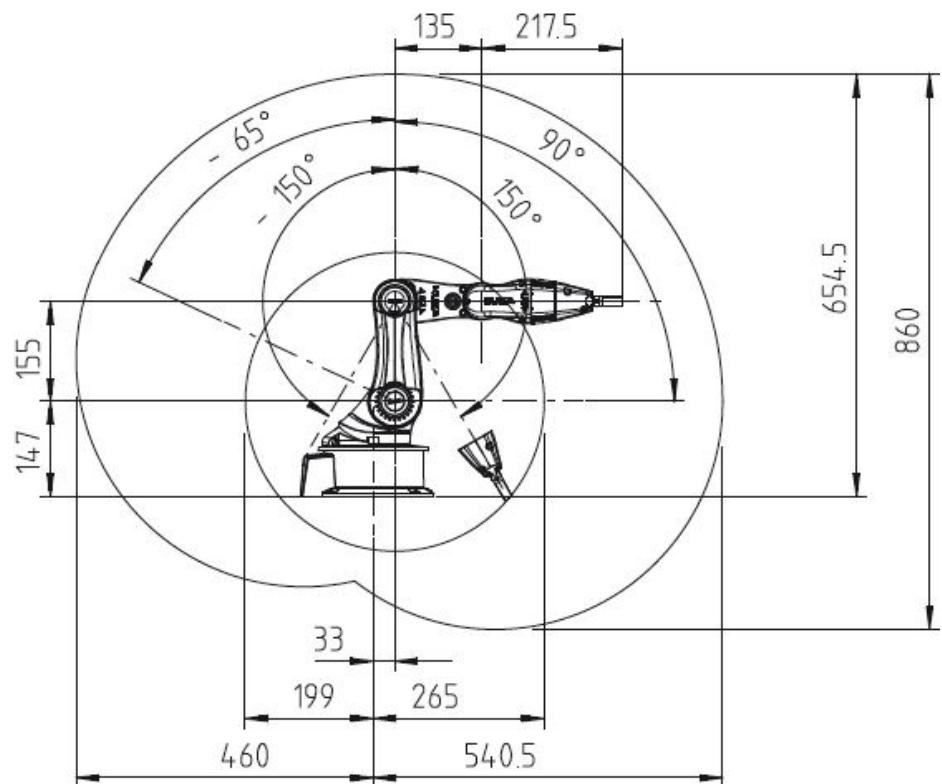
Основной элемент практически любого робота это — манипулятор, в этой работе представленный манипулятором с робота KUKA Youbot и изображённый на рисунке 2.1в. Его механизм имеет пять степеней подвижности. Многозвенная конструкция манипулятора заканчивается двухпальцевым схватом — инструментом, предназначенным для захвата объектов определённой формы.

Описание технических характеристик дается таблицей 2.1 и рисунком 2.1.

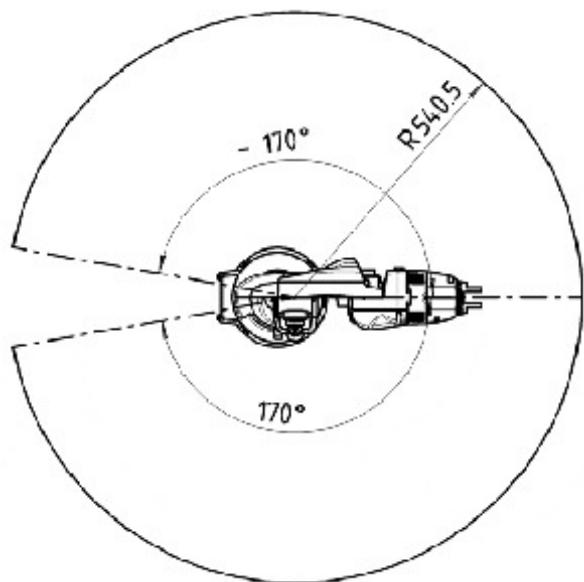
Таблица 2.1 – Общая информация о манипуляторе робота Kuka Youbot.

Параметр	Значение
Количество сочленений	5
Объем рабочей области, м ²	0.513
Масса, кг	5.3
Допустимая нагрузка, кг	0.5
Точность повторного воспроизведения позиции, мм	1
Максимальная скорость в сочленении, ° с ⁻¹	90
Интерфейс	EtherCAT
Напряжение питания, В	24

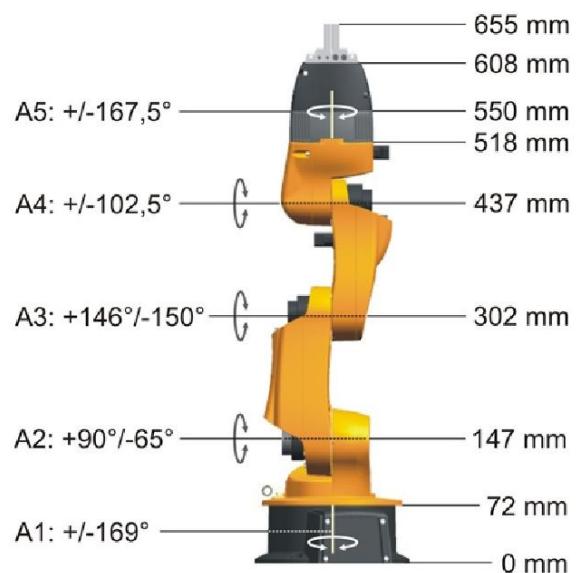
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



а)



б)



в)

Рисунок 2.1 – Некоторые параметры манипулятора Kuka Youbot: а – размеры рабочей области (вид сбоку); б – размеры рабочей области (вид сверху); в – длины звеньев и предельные значения для углов вращения по каждому из сочленений [6].

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2.2 Кинематическая модель манипулятора

2.2.1 Решение задач о положении манипулятора

Планирование траектории движения схвата манипулятора предполагает определение соотношений между координатами схвата, заданными в различной форме: обобщенными координатами самого манипулятора и шестью числами, задающими положение и ориентацию схвата в связанной с ним системе координат.

Последовательная кинематическая цепь рассматриваемого манипулятора, включающая только вращательные кинематические пары (КП) V-класса (цилиндрические шарниры), изображена на рисунке 2.2а.

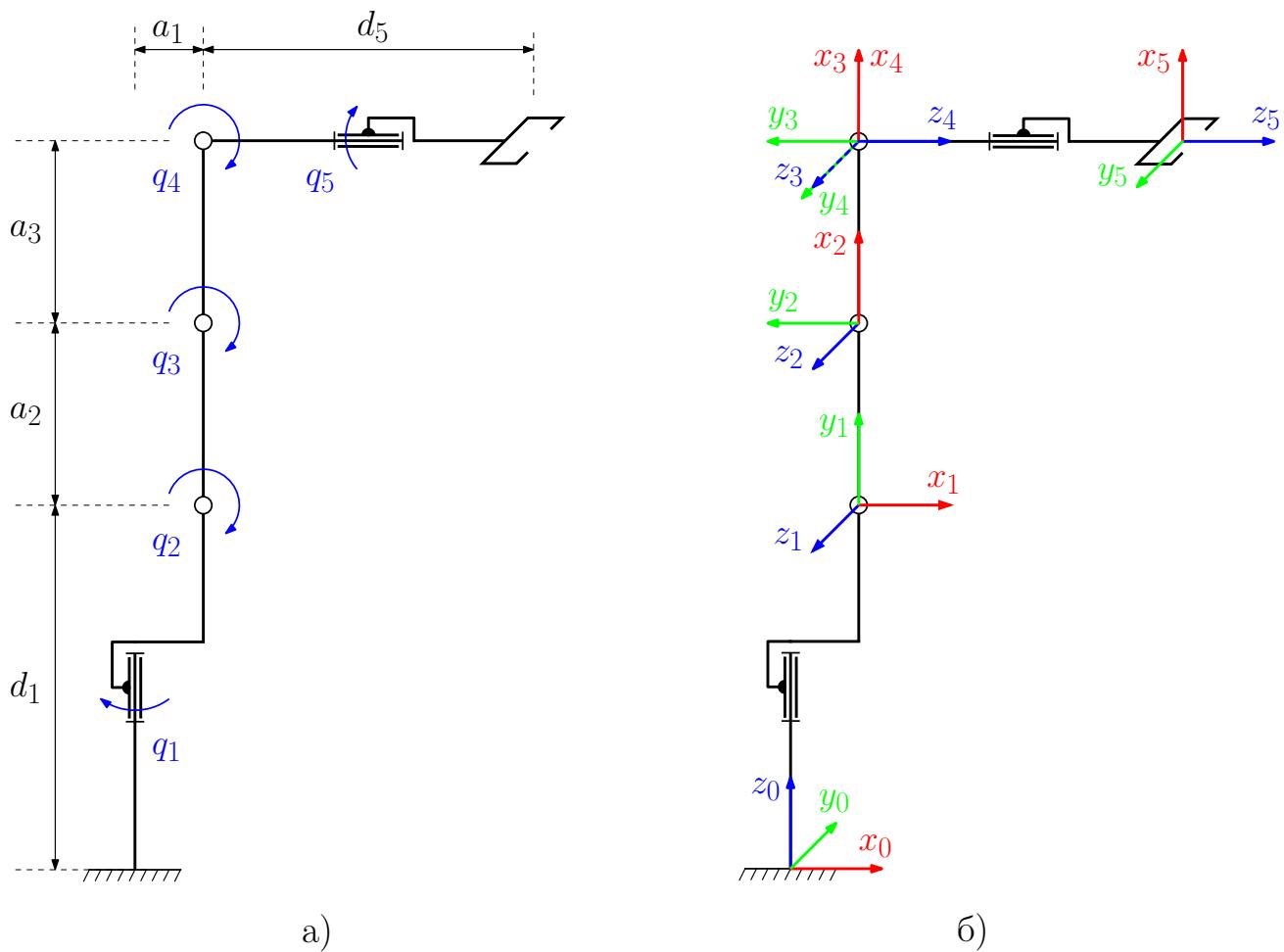


Рисунок 2.2 – Схемы рассматриваемого манипулятора: а – кинематическая при $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T = [0, \pi/2, 0, 0, 0]^T$; б – расположения СК КП.

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Процедура описания положений звеньев манипулятора друг относительно друга в соответствии с методом Денавита-Хартенберга [7], представляется тремя шагами:

а) «привязка» к каждому звену СК, чьи оси удовлетворяют следующим условиям:

- 1) ось z_{i-1} направлена вдоль оси i -ой КП;
- 2) ось x_i перпендикулярна оси z_{i-1} и пересекает ее;
- 3) ось y_i дополняет оси z_i и x_i до правой декартовой СК.

б) определение параметров ДХ:

- 1) a_i — расстояния от z_{i-1} до z_i вдоль x_i ;
- 2) α_i — угла от z_{i-1} до z_i вокруг x_i ;
- 3) d_i — расстояния от x_{i-1} до x_i вдоль z_{i-1} ;
- 4) θ_i — угла от x_{i-1} до x_i вокруг z_{i-1} .

в) расчет матриц однородного преобразования в соответствии со следующими формулами:

$${}^{i-1}A_i = R_{z,\theta_i} \cdot T_{z,d_i} \cdot T_{x,a_i} \cdot R_{x,\alpha_i} \quad (2.1)$$

где R_{z,θ_i} — матрица поворота вокруг оси z на угол θ_i , T_{z,d_i} — матрица смещения вдоль оси z на расстояние d , T_{x,a_i} — матрица смещения вдоль оси x на расстояние a_i , R_{x,α_i} — матрица поворота вокруг оси x на угол α_i , равные

$$R_{z,\theta_i} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T_{z,d_i} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.2)$$

$$(2.3)$$

Изв. № подп.	Подп. и дата	Бзм. изв. № дубл.	Изв. № дубл.	Подп. и дата

$$T_{x,a_i} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R_{x,\alpha_i} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad (2.4)$$

ИТОГО

$${}^{i-1}A_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

Результаты выполнения двух первых шагов для исследуемого манипулятора представлены на рисунке 2.2б и в таблице 2.2, а третьего — в лице следующих выражений:

$$\begin{aligned} {}^0A_1 &= \begin{bmatrix} c_1 & 0 & s_1 & a_1 c_1 \\ s_1 & 0 & -c_1 & a_1 s_1 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^1A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^2A_3 = \begin{bmatrix} c_3 & -s_3 & 0 & a_3 c_3 \\ s_3 & c_3 & 0 & a_3 s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \\ {}^3A_4 &= \begin{bmatrix} c_4 & 0 & s_4 & 0 \\ s_4 & 0 & -c_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; {}^4A_5 = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (2.6)$$

Прямая задача кинематики решается из соотношений, содержащихся в информации о смещении и повороте СК $Ox_5y_5z_5$ относительно СК $Ox_0y_0z_0$, содержащейся в матрице 0A_5 . Следовательно, чтобы решить ПЗК, необходимо найти эту матрицу в соответствии с выражением:

$${}^0A_5 = \prod_{i=1}^5 {}^{i-1}A_i(q_i). \quad (2.7)$$

Решая **обратную задачу кинематики**, найдём соотношения для положений сочленений при известном положении и ориентации схвата. Заданные смещение и поворот СК $Ox_5y_5z_5$ относительно СК $Ox_0y_0z_0$ можно описать с

Извм.	Подп. и дата	Бзм. подп.	Извв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

Таблица 2.2 – Параметры Денавита-Хартенберга

Звено, i	a_i , мм	α_i , рад	d_i , мм	θ_i , рад
1	33	$\pi/2$	147	$\pi \cdot \frac{169^\circ}{180^\circ} - q_1$
2	155	0	0	$\pi \cdot \frac{65^\circ}{180^\circ} + \frac{\pi}{2} - q_2$
3	135	0	0	$-\pi \cdot \frac{146^\circ}{180^\circ} - q_3$
4	0	$\pi/2$	0	$\pi \cdot \frac{102.5^\circ}{180^\circ} + \frac{\pi}{2} - q_4$
5	0	0	218	$\pi \cdot \frac{167.5^\circ}{180^\circ} - q_5$

помощью матрицы 0A_5 . Используя ее и матрицы из (2.2.1), найти расчетные формулы для углов q_i ($i = \overline{1, 5}$) можно из следующих соображений [8].

Для неизвестной матрицы 0A_5 вводятся следующие обозначения ее элементов:

$${}^0A_5 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.8)$$

Далее, составляется равенство из матрицы 0A_5 и правой части выражения (2.7), затем полученное равенство с обеих сторон умножается на ${}^0A_1^{-1}$:

$${}^0A_1^{-1} \cdot {}^0A_5 = {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5, \quad (2.9)$$

где левая часть с учетом (2.2.1) равна

$${}^0A_1^{-1} \cdot {}^0A_5 = \begin{bmatrix} r_{11}c_1 + r_{21}s_1 & r_{12}c_1 + r_{22}s_1 & r_{13}c_1 + r_{23}s_1 & p_x c_1 + p_y s_1 - a_1 \\ r_{31} & r_{32} & r_{33} & p_z - d_1 \\ r_{11}s_1 - r_{21}c_1 & r_{12}s_1 - r_{22}c_1 & r_{13}s_1 - r_{23}c_1 & p_x s_1 - p_y c_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.10)$$

Изв. № подп.	Подп. и дата	Бзм. Изв. № дубл.	Изв. № дубл.	Подп. и дата

а правая

$${}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4 \cdot {}^4A_5 = \begin{bmatrix} c_5c_{234} & -s_5c_{234} & s_{234} & a_2c_2 + a_3c_{23} + d_5s_{234} \\ c_5s_{234} & -s_5s_{234} & -c_{234} & a_2s_2 + a_3s_{23} - d_5c_{234} \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.11)$$

где, в свою очередь,

$$\theta_{23} = \theta_2 + \theta_3, \quad \theta_{234} = \theta_2 + \theta_3 + \theta_4. \quad (2.12)$$

Теперь, решая уравнения, полученные из равенства матриц (2.10) и (2.11), выводятся выражения для углов θ_1 , θ_5 и θ_{234}

– равенство элементов (3, 4):

$$p_x s_1 - p_y c_1 = 0 \Rightarrow \tan \theta_1 = \frac{p_y}{p_x} \Rightarrow \begin{cases} \theta_1^I = \text{atan2}(p_y, p_x) \\ \theta_1^{II} = \text{atan2}(-p_y, -p_x) \end{cases} \quad (2.13)$$

– равенство элементов (3, 1) и (3, 2):

$$\begin{cases} s_5 = r_{11}s_1 - r_{21}c_1 \\ c_5 = r_{12}s_1 - r_{22}c_1 \end{cases} \Rightarrow \begin{cases} \theta_5^I = \text{atan2}(r_{11} \sin \theta_1^I - r_{21} \cos \theta_1^I, r_{12} \sin \theta_1^I - r_{22} \cos \theta_1^I) \\ \theta_5^{II} = \text{atan2}(r_{11} \sin \theta_1^{II} - r_{21} \cos \theta_1^{II}, r_{12} \sin \theta_1^{II} - r_{22} \cos \theta_1^{II}) \end{cases} \quad (2.14)$$

– равенство элементов (2, 3) и (1, 3):

$$\begin{cases} c_{234} = -r_{33} \\ s_{234} = r_{13}c_1 + r_{23}s_1 \end{cases} \Rightarrow \begin{cases} \theta_{234}^I = \text{atan2}(r_{13} \cos \theta_1^I + r_{23} \sin \theta_1^I, -r_{33}) \\ \theta_{234}^{II} = \text{atan2}(r_{13} \cos \theta_1^{II} + r_{23} \sin \theta_1^{II}, -r_{33}) \end{cases} \quad (2.15)$$

Домножая выражение (2.10) на ${}^4A_5^{-1}$ справа – получается матрица 1A_4 :

$${}^1A_4 = \begin{bmatrix} \dots & \dots & \dots & (p_y - d_5r_{23})s_1 + (p_x - d_5r_{13})c_1 - a_1 \\ \dots & \dots & \dots & p_z - d_1 - d_5r_{33} \\ \dots & \dots & \dots & p_x s_1 - p_y c_1 - d_5(r_{13}s_1 - r_{23}c_1) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.16)$$

Изв.	Лист	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

где \dots — неинтересные элементы матрицы.

Нужно заметить, что с учетом (2.13) и равенства элементов (3, 3) в (2.10) и (2.11) справедливо

$$p_x s_1 - p_y c_1 - d_5(r_{13}s_1 - r_{23}c_1) = 0. \quad (2.17)$$

С учетом этого и (2.16):

$$r_{1,4}^1 = \begin{bmatrix} (p_y - d_5 r_{23})s_1 + (p_x - d_5 r_{13})c_1 - a_1 \\ p_z - d_1 - d_5 r_{33} \\ 0 \end{bmatrix}. \quad (2.18)$$

Далее нужно заметить, что одно и то же положение 4-го звена может достигаться при двух разных способах расположения звеньев 2 и 3 (см. рисунок 2.3). Следовательно, углы θ_2 , θ_3 и θ_4 при одних и тех же значениях углов θ_1 и θ_5 имеют по два возможных значения. Ниже выводятся выражения для последних.

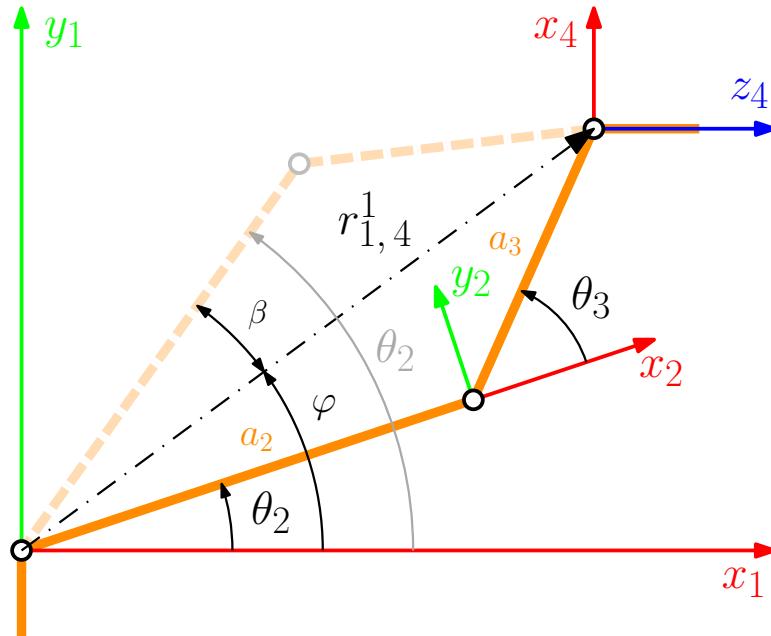


Рисунок 2.3 – Плоская часть манипулятора

По теореме косинусов, выражение для $\cos \theta_3$ (его зависимость от θ_1 обуславливается зависимостью от этого угла вектора $r_{1,4}^1$) примет вид:

$$c_3(\theta_1) = \frac{(r_{1,4}^1)^T \cdot r_{1,4}^1 - a_2^2 - a_3^2}{2a_2a_3} \quad (2.19)$$

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

С учетом этого для θ_3 можно получить следующие выражения:

$$\theta_3^{I,II} = \mp \operatorname{atan2}\left(\sqrt{1 - c_3^2(\theta_1^I)}, c_3(\theta_1^I)\right) \quad (2.20)$$

$$\theta_3^{III,IV} = \mp \operatorname{atan2}\left(\sqrt{1 - c_3^2(\theta_1^{II})}, c_3(\theta_1^{II})\right) \quad (2.21)$$

Как видно из рисунка 2.3, $\theta_2 = \varphi + \beta$ при $\theta_3^{I,II} < 0$ и $\theta_2 = \varphi - \beta$ при $\theta_3^{III,IV} > 0$. Следовательно, принимая во внимание то, что

$$\varphi(\theta_1) = \operatorname{atan2}(y_r, x_r), \quad \beta(\theta_3) = \operatorname{atan2}(a_3 \sin |\theta_3|, a_2 + a_3 \cos |\theta_3|), \quad (2.22)$$

где x_r и y_r — проекции вектора $r_{1,4}^1$ на оси абсцисс и ординат (их значения см. в (2.18)), для возможных значений угла θ_2 можно записать:

$$\theta_2^I = \varphi(\theta_1^I) + \beta(\theta_3^I), \quad \theta_2^{II} = \varphi(\theta_1^I) - \beta(\theta_3^{II}), \quad (2.23)$$

$$\theta_2^{III} = \varphi(\theta_1^{II}) + \beta(\theta_3^{III}), \quad \theta_2^{IV} = \varphi(\theta_1^{II}) - \beta(\theta_3^{IV}). \quad (2.24)$$

Выражения для значений угла θ_4 после этого с учетом (2.12) приобретают вид:

$$\theta_4^{I,II} = \theta_{234}^I - \theta_2^{I,II} - \theta_3^{I,II}, \quad \theta_4^{III,IV} = \theta_{234}^{II} - \theta_2^{III,IV} - \theta_3^{III,IV}. \quad (2.25)$$

Таким образом, любые положение и ориентацию схвата относительно основания манипулятора может обеспечить 4-мя собственными конфигурациями, которым соответствуют следующие наборы значений для его обобщенных координат $q = [q_1, q_2, q_3, q_4, q_5]^T$ (с учетом таблицы 2.2):

$$\theta^I = \begin{bmatrix} \theta_1^I & \theta_2^I & \theta_3^I & \theta_4^I & \theta_5^I \end{bmatrix}^T, \quad \theta^{II} = \begin{bmatrix} \theta_1^I & \theta_2^{II} & \theta_3^{II} & \theta_4^{II} & \theta_5^I \end{bmatrix}^T, \quad (2.26)$$

$$\theta^{III} = \begin{bmatrix} \theta_1^{II} & \theta_2^{III} & \theta_3^{III} & \theta_4^{III} & \theta_5^{II} \end{bmatrix}^T, \quad \theta^{IV} = \begin{bmatrix} \theta_1^{II} & \theta_2^{IV} & \theta_3^{IV} & \theta_4^{IV} & \theta_5^{II} \end{bmatrix}^T, \quad (2.27)$$

$$q^{I,II,III,IV} = \pi \cdot \left[\frac{169^\circ}{180^\circ} \frac{65^\circ}{180^\circ} + \frac{1}{2} - \frac{146^\circ}{180^\circ} \frac{102.5^\circ}{180^\circ} + \frac{1}{2} \frac{167.5^\circ}{180^\circ} \right]^T - \theta^{I,II,III,IV}. \quad (2.28)$$

2.2.2 Решение задач о скорости звеньев манипулятора

Как видно из рассуждений в разделе 2.2.1, i -ое звено связано с СК, начало которой закреплено в $(i+1)$ сочленении. При этом кинематические параметры этого звена задаются в связанной с ним СК $Ox_{i-1}y_{i-1}z_{i-1}$.

Изв.	Лист	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата
Изв. № подп.	Лист	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Получим выражения, которые позволяют последовательно получать скорости и ускорения звеньев манипулятора, начиная от базы в направлении схвата, выраженные в абсолютной СК $Ox_0y_0z_0$.

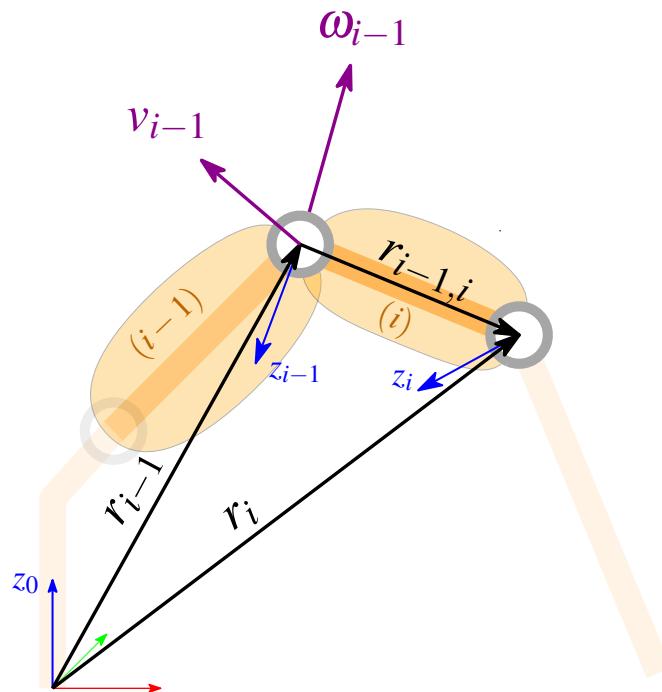


Рисунок 2.4 – Рисунок, поясняющий приведённые в разделе 2.2.2 выражения

Заметим, что в соответствии с рисунком 2.4, на котором изображена упрощенная кинематическая схема для двух соседних звеньев, радиус вектор

$$r_{i-1,i} = r_i - r_{i-1} \quad (2.29)$$

характеризует расположение СК $Ox_iy_iz_i$ относительно СК $Ox_{i-1}y_{i-1}z_{i-1}$.

Согласно (2.29) и теореме механики о сложении скоростей [9], соотношения для скоростей в подвижной и неподвижной системах координат записываются:

$$v_i = v_{i-1} + \omega_{i-1} \times r_{i-1,i} + \frac{d^* r_{i-1,i}}{dt}, \quad (2.30)$$

где $\frac{d^* r_{i-1,i}}{dt}$ — скорость i -ого звена относительно СК $Ox_{i-1}y_{i-1}z_{i-1}$.

Для угловой скорости:

$$\omega_i = \omega_{i-1} + \omega_i^{i-1}, \quad \omega_i^{i-1} = z_{i-1} \cdot \dot{q}_i, \quad (2.31)$$

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

где ω_i^{i-1} — угловая скорость вращения СК $Ox_iy_iz_i$ относительно $Ox_{i-1}y_{i-1}z_{i-1}$, \dot{q}_i — угловая скорость вращения i -ого звена.

С учетом равенств (2.29)–(2.31), окончательно запишем рекуррентные соотношения для вычисления линейных и угловых скоростей манипулятора:

$$v_i = v_{i-1} + \omega_i \times r_{i-1,i}, \quad (2.32)$$

$$\omega_i = \omega_{i-1} + z_{i-1} \cdot \dot{q}_i. \quad (2.33)$$

Далее, согласно теореме о сложении ускорений, запишем соотношения для линейных и угловых ускорений:

$$\ddot{v}_i = \ddot{v}_{i-1} + \omega_{i-1} \times (\omega_{i-1} \times r_{i-1,i}) + 2 \cdot \omega_{i-1} \times \frac{d^* r_{i-1,i}}{dt} + \frac{d^{*2} r_{i-1,i}}{dt^2}, \quad (2.34)$$

$$\dot{\omega}_i = \dot{\omega}_{i-1} + \dot{\omega}_i^{i-1}, \quad (2.35)$$

где

$$\dot{\omega}_i^{i-1} = \frac{d^* \omega_i^{i-1}}{dt} + \omega_{i-1} \times \omega_i^{i-1}, \quad \frac{d^* \omega_i^{i-1}}{dt} = z_{i-1} \cdot \ddot{q}_i. \quad (2.36)$$

После элементарных преобразований, рекуррентные соотношения для линейных и угловых ускорений записутся, как:

$$\ddot{v}_i = \ddot{v}_{i-1} + \omega_i \times (\omega_i \times r_{i-1,i}) + \dot{\omega}_i \times r_{i-1,i}, \quad (2.37)$$

$$\dot{\omega}_i = \dot{\omega}_{i-1} + z_{i-1} \cdot \ddot{q}_i + \omega_{i-1} \times (z_{i-1} \cdot \dot{q}_i). \quad (2.38)$$

Таким образом, соотношения (2.32)–(2.33) и (2.37)–(2.38) позволяют найти линейные и угловые скорости и ускорения звеньев, при $i = \overline{1, n}$ и начальных условиях $\omega = \dot{\omega} = v_0 = 0$, $\dot{v}_0 = g$, где g — вектор ускорения свободного падения.

Отсюда можно без труда получить конечные зависимости линейных и угловых скоростей. Для линейной скорости k -ого звена:

$$v_k = \sum_{i=1}^k z_{i-1} \times r_{i-1,k} \cdot \dot{q}_i. \quad (2.39)$$

Для угловой скорости k -ого звена:

$$\omega_k = \sum_{i=1}^k z_{i-1} \cdot \dot{q}_i. \quad (2.40)$$

Изв.	Лист	№ докум.	Подп.	Дата	Подп. и дата	Изв. № дубл.	Бзм. изв. №	Подп. и дата

Все векторы в рассмотренных соотношениях, описывающие движения манипулятора, заданы в базовой СК $Ox_0y_0z_0$. Руководствуясь соображениями вычислительной эффективности, представим векторы скоростей в СК их собственного звена. Тогда, если матрица перехода из СК i -ого звена $Ox_iy_iz_i$ в СК $Ox_{i-1}y_{i-1}z_{i-1}$, то:

$${}^{i-1}A_i^{-1} = \begin{bmatrix} {}^{i-1}R_i^T & p_{i-1,i} \\ 000 & 1 \end{bmatrix}, \quad (2.41)$$

где $p_{i-1,i} = {}^0R_{i-1}^T \cdot r_{i-1,i}$.

Можно переписать выражения (2.32)–(2.33) и (2.37)–(2.38) в виде:

$$v_i^i = {}^{i-1}R_i \cdot v_{i-1}^{i-1} + \omega_i^i \times p_{i-1,i}, \quad (2.42)$$

$$\omega_i^i = {}^{i-1}R_i \cdot \omega_{i-1}^{i-1} + {}^{i-1}R_i \cdot z_0 \dot{q}_i, \quad (2.43)$$

$$\dot{v}_i^i = {}^{i-1}R_i \cdot \dot{v}_{i-1}^{i-1} + \omega_i^i \times (\omega_i^i \times p_{i-1,i}) + \dot{\omega}_i^i \times p_{i-1,i}, \quad (2.44)$$

$$\dot{\omega}_i^i = {}^{i-1}R_i \cdot \dot{\omega}_i^i + {}^{i-1}R_i \cdot z_0 \ddot{q}_i + {}^{i-1}R_i \cdot \omega_{i-1}^{i-1} \times (z_0 \dot{q}_i). \quad (2.45)$$

В задачах о скорости движения схвата манипулятора описывается вектором $\mathbf{s} = [x, y, z, \varphi, \psi, \theta]$, линейной \mathbf{v} и угловой $\boldsymbol{\omega}$ скоростями, где со схватом связана СК $Ox_ny_nz_n$. Вектор \mathbf{s} определяется конфигурацией манипулятора, задаваемой вектором обобщенных координат $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$, что можно записать, как:

$$\mathbf{s} = f(\mathbf{q}). \quad (2.46)$$

Дифференцируя (2.46) по времени, получаем:

$$\dot{\mathbf{s}} = J(\mathbf{q}) \cdot \dot{\mathbf{q}}, \quad (2.47)$$

где $\dot{\mathbf{s}}$ вектор скорости схвата:

$$\dot{\mathbf{s}} = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix}, \quad (2.48)$$

$J(\mathbf{q})$ — матрица Якоби размерности $6 \times n$, $\dot{\mathbf{q}} = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T$ — вектор скоростей обобщенных координат.

Прямая задача о скорости формулируется, как нахождение линейной \mathbf{v} и угловой $\boldsymbol{\omega}$ скоростей схвата по известным скоростям обобщенных координат $\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n$. Заметим, что из (2.32)–(2.33) следует, что если записать

Изв. № подп.	Подп. и дата	Бзм. Изв. № дубл.	Изв. № дубл.	Подп. и дата

матрицу Якоби в виде:

$$J(\mathbf{q}) = \begin{bmatrix} \mathbf{j}_1 & \mathbf{j}_2 & \cdots & \mathbf{j}_n \end{bmatrix}, \quad (2.49)$$

то вектор \mathbf{j}_k будет рассчитываться из выражения:

$$\mathbf{j}_k = \begin{bmatrix} z_{k-1} \\ z_{z-1} \times p_{k-1,n} \end{bmatrix}, \quad (2.50)$$

Следовательно, выражение (2.47) является решением прямой задачи о скорости, где матрица $J(\mathbf{q})$ находится из (2.49).

Матрицу Якоби получим, используя дифференциальные преобразования. Обозначим матрицу Якоби, как:

$$J = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix}, \quad (2.51)$$

тогда перепишем соотношения для скоростей в виде

$$\mathbf{v} = J_v \dot{\mathbf{q}}, \quad (2.52)$$

$$\boldsymbol{\omega} = J_\omega \dot{\mathbf{q}}. \quad (2.53)$$

Если представить малое изменение матрицы трансформации T (рисунок 2.5а), как

$$\tilde{T} = T + dT, \quad (2.54)$$

откуда $dT = \tilde{T} - T = \tilde{T}(d\mathbf{r}, \delta\varphi) - T$, где $d\mathbf{r}$ и $\delta\varphi$ — малые перемещение и поворот соответственно, то для k-ого звена манипулятора, применяя правило дифференцирования матриц однородных преобразований $\frac{\partial A_k}{\partial q_l} = \theta_k \cdot A_k$, можно получить:

$$d_k T = A_1 A_2 \dots A_{k-1} \theta_k dq_k A_{k+1} \dots A_{n-1} A_n, \quad (2.55)$$

где

$$\theta_k = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.56)$$

Изв.	№ подп.	Подп. и дата	Бзм. № дубл.	Изв. № дубл.	Подп. и дата

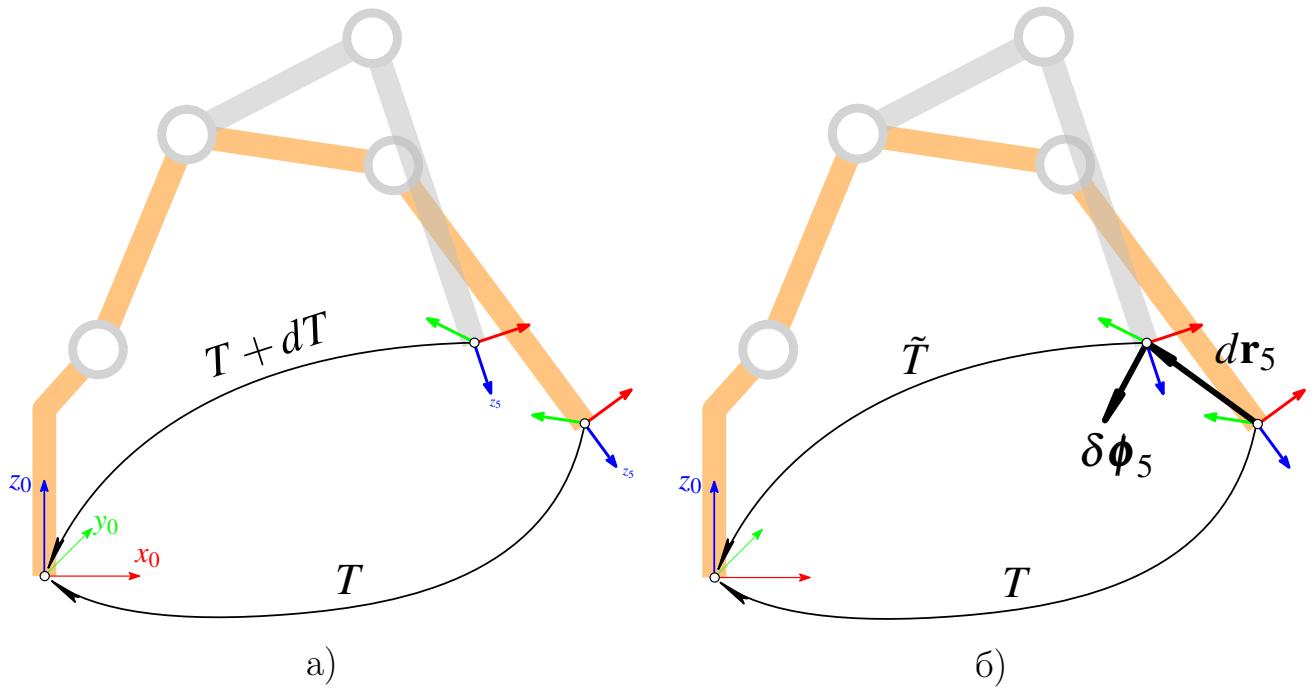


Рисунок 2.5 – а — дифференциальное перемещение, б — изменение матрицы трансформации при дифференциальном перемещении

Выражение (2.55) можно переписать в виде:

$$d_k T = \Delta_{\theta_k} \cdot T, \quad (2.57)$$

где Δ_{θ_k} — матрица дифференциальных преобразований:

$$\Delta_{\theta_k} = T_{k-1} \theta_k T_{k-1}^{-1} d q_k. \quad (2.58)$$

Обозначим разыскиваемые дифференциальные перемещения схвата манипулятора, как $d\mathbf{r}_n$ и $\delta\boldsymbol{\varphi}_n$, изображённые на рисунке 2.5б, тогда матрица дифференциальных преобразований:

$$\Delta_n = \begin{bmatrix} 0 & -\delta z_n & \delta y_n & dx_n \\ \delta z_n & 0 & -\delta x_n & dy_n \\ -\delta y_n & \delta x_n & 0 & dz_n \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \Omega_{\delta\boldsymbol{\varphi}_n} & d\mathbf{r}_n \\ 000 & 0 \end{bmatrix}. \quad (2.59)$$

Раскрыв (2.58), получим:

$$\begin{bmatrix} \Omega_{\delta\boldsymbol{\varphi}_n} & d\mathbf{r}_n - \Omega_{\delta\boldsymbol{\varphi}_n} \cdot p_n \\ 000 & 0 \end{bmatrix} = \begin{bmatrix} R_{k-1} \Omega_{001} R_{k-1}^T & -R_{k-1} \Omega_{001} R_{k-1}^T p_{k-1} \\ 000 & 0 \end{bmatrix} d q_k, \quad (2.60)$$

Изв.	Лист	Подп. и дата	Бзм. подп. и дата	Изв. № дубл.	Подп. и дата

где

$$d\mathbf{r}_n = \Omega_{\delta\varphi_n} \cdot p_n - R_{k-1} \cdot \Omega_{001} \cdot R_{k-1}^T \cdot p_{k-1} \cdot dq_k, \quad (2.61)$$

$$\Omega_{\delta\varphi_n} = R_{k-1} \cdot \Omega_{001} \cdot R_{k-1}^T \cdot dq_k, \quad (2.62)$$

Ω_{001} — матрица, задающая вращение вокруг вектора $[0, 0, 1]^T$.

Вращение вокруг оси z_{k-1} определяется выражением:

$$z_{k-1} = R_{k-1} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (2.63)$$

тогда, в силу (2.62) и (2.61), имеем

$$d\mathbf{r}_n = z_{k-1} \times (p_n - p_{k-1}) \cdot dq_k, \quad (2.64)$$

$$\delta\varphi_n = z_{k-1} \cdot dq_k. \quad (2.65)$$

Из выражения (2.65) видим, что расчет матрицы Якоби J_ω из выражения (2.51) не представляет труда, так как векторы z_i легко извлекаются из матриц трансформации T_i (третий столбец).

Из выражений (2.65)–(2.64) и (2.51) получим все матрицы Якоби для рассматриваемого манипулятора:

$$\begin{aligned} J_{\omega 1} &= \begin{bmatrix} z_0^0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, & J_{\omega 2} &= \begin{bmatrix} z_0^0 & z_1^0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \\ J_{\omega 3} &= \begin{bmatrix} z_0^0 & z_1^0 & z_2^0 & \mathbf{0} & \mathbf{0} \end{bmatrix}, & J_{\omega 4} &= \begin{bmatrix} z_0^0 & z_1^0 & z_2^0 & z_3^0 & \mathbf{0} \end{bmatrix}, \\ J_{\omega 5} &= \begin{bmatrix} z_0^0 & z_1^0 & z_2^0 & z_3^0 & z_4^0 \end{bmatrix}, \end{aligned}$$

где $\mathbf{0} = [0 \ 0 \ 0]^T$ — нулевой вектор.

$$\begin{aligned} J_{v1} &= \begin{bmatrix} z_0^0 \times (r_{0,1}^0 - r_{0,0}^0) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \\ J_{v2} &= \begin{bmatrix} z_0^0 \times (r_{0,2}^0 - r_{0,0}^0) & z_1^0 \times (r_{0,2}^0 - r_{0,1}^0) & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \\ J_{v3} &= \begin{bmatrix} z_0^0 \times (r_{0,3}^0 - r_{0,0}^0) & z_1^0 \times (r_{0,3}^0 - r_{0,1}^0) & z_2^0 \times (r_{0,3}^0 - r_{0,2}^0) & \mathbf{0} & \mathbf{0} \end{bmatrix}, \end{aligned}$$

Изв. № подп.	Подп. и дата	Бзм. изв. №	Изв. № дубл.	Подп. и дата

$$J_{v4} = \begin{bmatrix} z_0^0 \times (r_{0,4}^0 - r_{0,0}^0) \\ z_1^0 \times (r_{0,4}^0 - r_{0,1}^0) \\ z_2^0 \times (r_{0,4}^0 - r_{0,2}^0) \\ z_3^0 \times (r_{0,4}^0 - r_{0,3}^0) \\ \mathbf{0} \end{bmatrix}^T, \quad J_{v5} = \begin{bmatrix} z_0^0 \times (r_{0,5}^0 - r_{0,0}^0) \\ z_1^0 \times (r_{0,5}^0 - r_{0,1}^0) \\ z_2^0 \times (r_{0,5}^0 - r_{0,2}^0) \\ z_3^0 \times (r_{0,5}^0 - r_{0,3}^0) \\ z_4^0 \times (r_{0,5}^0 - r_{0,4}^0) \end{bmatrix}^T.$$

Обратная задача о скорости формулируется как поиск неизвестных скоростей обобщенных координат при известных линейной и угловой скоростях схвата. Соотношения записывается как система шести линейных уравнений с n неизвестными $\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n$:

$$\dot{\mathbf{q}} = J^{-1}(\mathbf{q})\dot{\mathbf{s}}. \quad (2.66)$$

Решение (2.66) существует тогда и только тогда, когда

$$\text{rank}(J(\mathbf{q})) = \text{rank}([J(\mathbf{q}), \dot{\mathbf{s}}]). \quad (2.67)$$

Для пятиступенчатого манипулятора KUKA Youbot возможны три случая:

- a) $\text{rank}(J) \neq \text{rank}([J, \dot{\mathbf{s}}])$ — не имеет решений;
- б) $\text{rank}(J) = \text{rank}([J, \dot{\mathbf{s}}])$ — имеет единственное решение;
- в) $\text{rank}(J) = \text{rank}([J, \dot{\mathbf{s}}])$ — бесконечное множество решений.

Таким образом, в сингулярных конфигурациях манипулятора матрица Якоби не позволяет установить однозначную зависимость в (2.66). Поиск решений уравнения (2.66) является частью алгоритма управления манипулятором.

2.3 Система кинематического управления

Кинематическое управление реализуется в два этапа: первый включает в себя получение траектории движения сочленений манипулятора $\mathbf{q}(t)$ или схвата $\mathbf{s}(t)$ на интервале времени $t \in [t_s, t_f]$; второй этап подразумевает синтез системы управления, способной отработать спланированную траекторию.

Планированию траекторий посвящён следующий раздел. Здесь реализуем систему управления для следования по заданной траектории.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

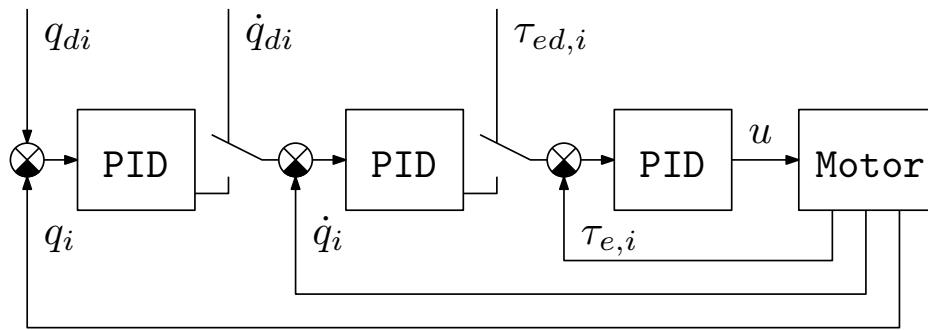


Рисунок 2.6 – Структурная схема системы управления приводами, встроенная в контроллеры манипулятора

Каждый из приводов манипулятора робота KUKA Youbot имеет собственную систему управления, структура которой иллюстрируется схемой, представленной на рисунке 2.6. Из нее видно, что каждый из приводов робота может управляться заданием значения для угла q_{di} , или скорости \dot{q}_{di} , или момента силы $\tau_{ed,i}$, который должен быть на нем обеспечен. Это значение подается на вход соответствующего ПИД-регулятора, коэффициенты которого доступны настройке, и далее (уже в виде сигнала напряжения u) на контролируемый двигатель.

Далее в тексте будет рассмотрена система управления, в которой в качестве управляющего сигнала рассматриваются векторы \mathbf{q} и $\dot{\mathbf{q}}$. Из величин, описывающих состояние робота в данный момент времени, в используемом ПО доступны векторы $\mathbf{q}(t)$, $\dot{\mathbf{q}}(t)$ и $\boldsymbol{\tau}_e(t)$.

Цель управления в минимизации ошибки между заданной траекторией и положением схвата в каждый момент времени. Введём обозначения: траекторию обозначим, как:

$$\mathbf{s}_d = \begin{bmatrix} \mathbf{p}_d \\ \boldsymbol{\varphi}_d \end{bmatrix} = f(\mathbf{q}_d), \quad \dot{\mathbf{s}}_d = \begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = f(\dot{\mathbf{q}}_d), \quad (2.68)$$

а текущее положение схвата:

$$\mathbf{s} = \begin{bmatrix} \mathbf{p} \\ \boldsymbol{\varphi} \end{bmatrix} = f(\mathbf{q}), \quad \dot{\mathbf{s}} = \begin{bmatrix} v \\ \omega \end{bmatrix} = f(\dot{\mathbf{q}}), \quad (2.69)$$

Для следования по траектории, заданной функциями $\mathbf{s}_d(t)$ и $\dot{\mathbf{s}}_d(t)$ реализуем управление по вектору скорости, которое формулируется как минимизация

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

ошибки скорости следования по траектории. Воспользуемся соотношением для обратной задачи о скорости:

$$\dot{\mathbf{q}} = J^+(\mathbf{q})\dot{\mathbf{s}}, \quad (2.70)$$

где \mathbf{q} — вектор обобщенных координат, \mathbf{s} — вектор, определяющий положение схвата, J^+ — псевдообратная матрица (так как размерность матрицы Якоби для рассматриваемого манипулятора $[6 \times 5]$).

Введем ошибку по положению схвата:

$$\mathbf{e}(t) = \mathbf{s}_d(t) - \mathbf{s}(t). \quad (2.71)$$

Затем, дифференцируя (2.71) по времени, получим:

$$\dot{\mathbf{e}}(t) = \dot{\mathbf{s}}_d(t) - \dot{\mathbf{s}}(t) = \dot{\mathbf{s}}_d(t) - J^+(\mathbf{q})\dot{\mathbf{q}}. \quad (2.72)$$

Необходимо выбрать вектор $\dot{\mathbf{q}}(t)$ таким, чтобы выполнялось условие

$$\lim_{t \rightarrow \infty} \dot{\mathbf{e}}(t) = 0. \quad (2.73)$$

Тогда, если

$$\dot{\mathbf{q}}(t) = J^+(\mathbf{q}) \cdot (\dot{\mathbf{s}}_d(t) - \dot{\mathbf{e}}(t)), \quad (2.74)$$

то, выбрав

$$\dot{\mathbf{e}}(t) = -K\mathbf{e}(t), \quad (2.75)$$

можно обеспечить асимптотическую устойчивость системы управления.

Схема системы управления изображена на рисунке 2.7.

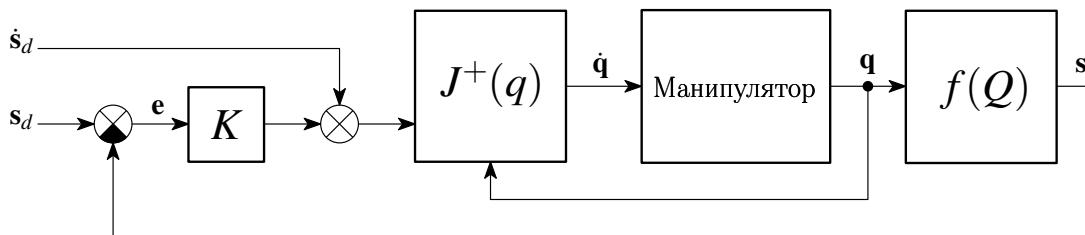


Рисунок 2.7 – Система управления по вектору скорости

В следующем разделе рассмотрим способы задания траектории, которую в последствии будем подавать на вход синтезированной системы управления.

Извм.	Лист	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

3 Планирование траекторий движения манипулятора

Рассматриваются задачи планирования траекторий движения манипулятора в решении комплексных задач: перемещению в рабочем пространстве и взаимодействию с целевыми объектами.

3.1 Постановка и описание задачи

Успешность захвата манипулятором, перемещающегося по конвейеру объекта заключается, как в качественной работе системы управления манипулятором, так и в планировании траектории движения его схвата, которая в точности должна повторять движение объекта.

Планирование траектории подразумевает получение программной зависимости перемещения звеньев манипулятора $\mathbf{q}(t)$ или схвата $\mathbf{s}(t)$ на интервале $t \in [t_s, t_f]$. Тут $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$ — вектор обобщенных координат, $\mathbf{s} = [x, y, z, \varphi, \psi, \theta]^T$ — вектор положения схвата относительно базовой СК $Ox_0y_0z_0$.

Необходимо спланировать траектории в пространстве обобщенных координат: для задания необходимой конфигурации робота — удобная ориентации камеры, закрепленной на последнем звене манипулятора (перевод из точки в точку); для обхода последовательности точек — осознанное перемещения объекта в рабочем пространстве от конвейера в зону погрузки. Недостатком такого подхода является то, что такие траектории нельзя использовать в операциях, где задача движения формулируется в пространстве координат схвата.

Также, необходимо спланировать траекторию в пространстве координат схвата для выполнения захватывания движущегося по конвейеру объекта.

Очевидно, что манипулятор с количеством степеней свободы $N < 6$ ограничен в своих движениях и, как видно из рассмотренной ранее кинематики манипулятора KUKA YouBot, он не может занимать любые необходимые ориентации, из чего следует, что в задаче решаемой в этой работе конвейер необходимо располагать параллельно плоскости OXY СК $Ox_0y_0z_0$.

Также заметим, что рабочая область манипулятора при схвате сориенти-

Изв.	Лист	№ докум.	Подп.	Дата	Подп. и дата	Изв. № дубл.	Взам. изв. №	Подп. и дата

рованном перпендикулярно плоскости OXY СК $Ox_0y_0z_0$ (или, что тоже самое, рабочей плоскости конвейера) существенно ограничена. На рисунке 3.1 зеленым выделена рабочая область, в которой возможен захват объектов с конвейера или любой другой плоскости, удовлетворяющих условию параллельности описанному выше.

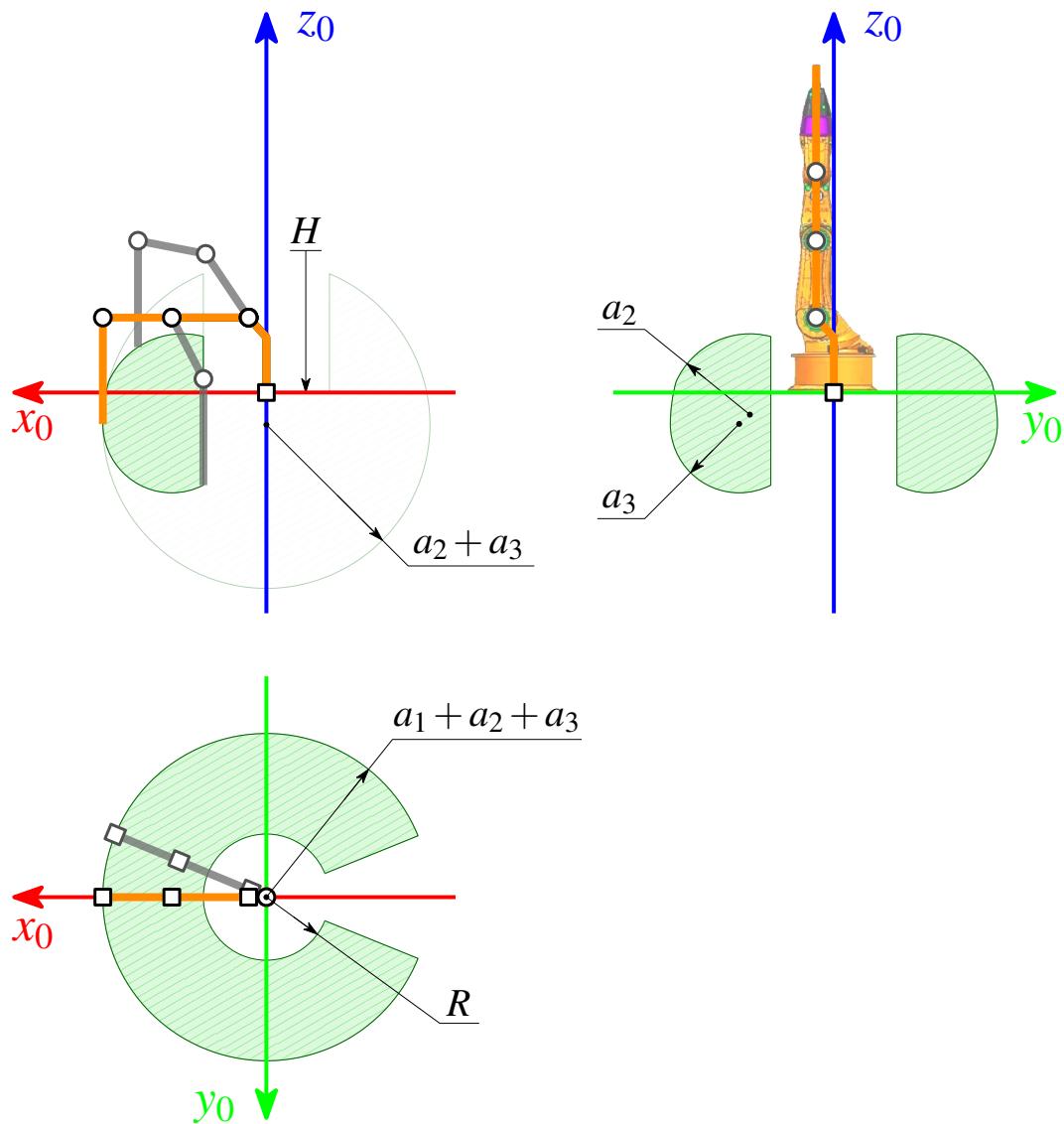


Рисунок 3.1 – Рабочая область манипулятора робота KUKA Youbot при ориентации схвата перпендикулярной плоскости OXY СК $Ox_0y_0z_0$ с учетом ограничений по допустимым углам поворотов сочленений и зоной безопасности в виде цилиндра вокруг оси z_0 с радиусом R и высотой H

Запишем последовательность действий, необходимую для захвата объ-

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

екта с конвейера:

- привести манипулятор в положения для работы СТЗ;
- перевести манипулятор в точку пересечения траектории движения объекта по конвейеру с нормальной рабочей областью манипулятора (точка p_0 на рисунке 3.2);
- проводить манипулятор по траектории, совпадающей с траекторией движения объекта по конвейеру *;
- захватить объект;
- перенести объект в зону для складывания объектов с конвейера.

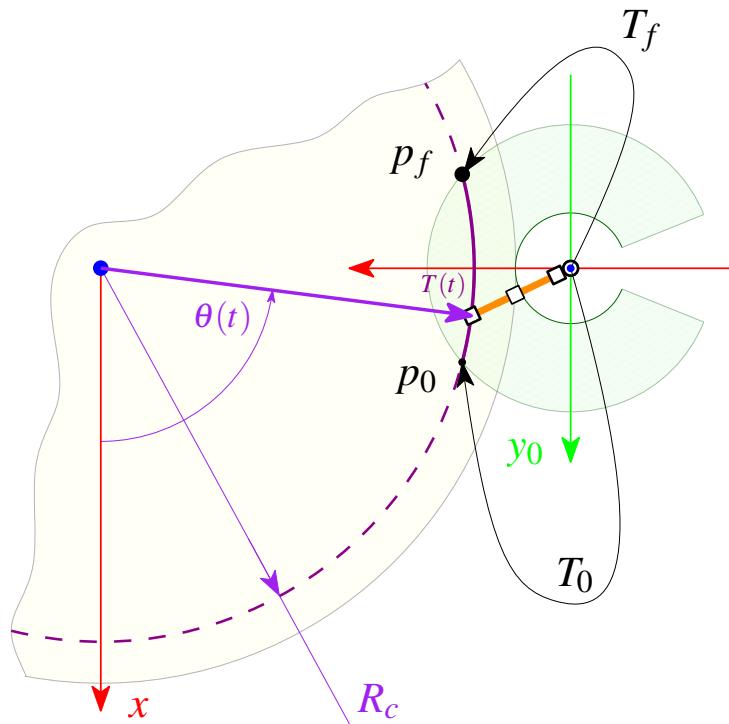


Рисунок 3.2 – Пересечение плоскости конвейера и рабочей области манипулятора

* Эта траектория представляет из себя кривую расположенную над движущимся объектом. Т.е. манипулятор следует за объектом на некотором расстоянии над ним, пока не выровняются их скорости, после чего схват опускается на высоту объекта, захватывает объект и поднимается на прежнее расстояние.

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

3.2 Распределение допустимых скоростей в рабочем пространстве

Необходимо выбрать траекторию движения схвата манипулятора такую, чтобы законы изменения положения, скоростей и ускорения, с одной стороны, соответствовали параметрам движения конвейера, а с другой — возможностям манипулятора.

Ограничения по скоростям в сочленениях заданы константами C_i :

$$|\dot{q}| \leq C_i, \quad i = \overline{1, n}. \quad (3.1)$$

Соотношения (3.1) задают область допустимых скоростей в пространстве обобщенных координат. Скорости схвата рассчитывается из соотношений (2.52)–(2.53) для решения прямой задачи о скорости и определяют отображение области допустимых скоростей в сочленениях в трехмерную область допустимых скоростей \mathbf{v}_n , которую можно найти для каждой точки рабочего пространства. Определяя такие отображения для заданных всех возможных вариантов вектора \mathbf{q} можно найти распределение допустимых скоростей схвата в рабочем пространстве.

В практической реализации ограничимся получением оценки снизу и сверху значений скорости \mathbf{v}_n в рабочей зоне. Для этого определим евклидову норму вектора скорости \mathbf{v}_n :

$$\|\mathbf{v}_n\| = (\mathbf{v}^T \cdot \mathbf{v})^{\frac{1}{2}} = (\dot{\mathbf{q}}^T \cdot J_v^T(\mathbf{q}) \cdot J_v(\mathbf{q}) \cdot \dot{\mathbf{q}})^{\frac{1}{2}} \quad (3.2)$$

Тогда, обозначив за минимальное $\underline{\lambda}(q)$ и максимальное $\bar{\lambda}(q)$ характеристические числа матрицы $J_v^T(\mathbf{q}) \cdot J_v(\mathbf{q})$ и на основании свойств квадратичных форм, запишем:

$$\underline{\lambda}(q) \cdot \|\dot{\mathbf{q}}\|^2 \leq \|\mathbf{v}_n\|^2 \leq \bar{\lambda}(q) \cdot \|\dot{\mathbf{q}}\|^2, \quad (3.3)$$

Окончательно, для ограничения скоростей движения схвата сверху, оценка имеет вид:

$$\|\mathbf{v}_n(q)\| \leq \sqrt{\bar{\lambda}(q)} \cdot \|\dot{\mathbf{q}}\| < \sqrt{\bar{\lambda}(q)} \cdot C, \quad (3.4)$$

где $C = \sqrt{\sum_{i=1}^n C_i^2(q)}$.

Полученная оценка не может быть найдена в вырожденных конфигурациях манипулятора, а также на границах рабочей области.

Инв. № подл.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

3.3 Планирование траектории в пространстве обобщенных координат

Рассмотрим движение манипулятора из точки в точку, затем расширим это решение на движение через заданную последовательность точек.

3.3.1 Движение из точки в точку

Движение манипулятора в пространстве обобщенных координат из точки \mathbf{q}_s в точку \mathbf{q}_f формулируется, как:

$$\mathbf{q} = \mathbf{q}(t), \quad t \in [t_s, t_f], \quad (3.5)$$

при граничных условиях:

$$\mathbf{q}(t_s) = \mathbf{q}_s, \quad \mathbf{q}(t_f) = \mathbf{q}_f. \quad (3.6)$$

Такое движение можно обеспечить используя кривую, составленную из парабол на участках разгона и торможения и прямой на участке с постоянной скоростью. Пример такой траектории изображен на рисунке 3.3.

Введем ограничение максимальной скорости $v = v_{des}$. Тогда время отработки траектории из точки \mathbf{q}_s в точку \mathbf{q}_f можно записать:

$$t_f = \frac{3}{2} \cdot \frac{\mathbf{q}_f - \mathbf{q}_s}{v}. \quad (3.7)$$

Отсюда, время разгона (торможения) и границы ускорений:

$$t_b = \frac{\mathbf{q}_s - \mathbf{q}_f + v * t_f}{v}, \quad a = \frac{v}{t_b}. \quad (3.8)$$

Траектория рассчитывается из следующих соотношений:

$$\mathbf{q}(t) = \begin{cases} \mathbf{q}_s + \frac{a}{2} \cdot t^2, & t_s \leq t \leq t_b, \\ \mathbf{q}_s + \frac{a}{2} \cdot t^2, & t_b < t \leq t_f - t_b, \\ \mathbf{q}_s + \frac{a}{2} \cdot t^2, & t_f - t_b < t \leq t_f. \end{cases} \quad (3.9)$$

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

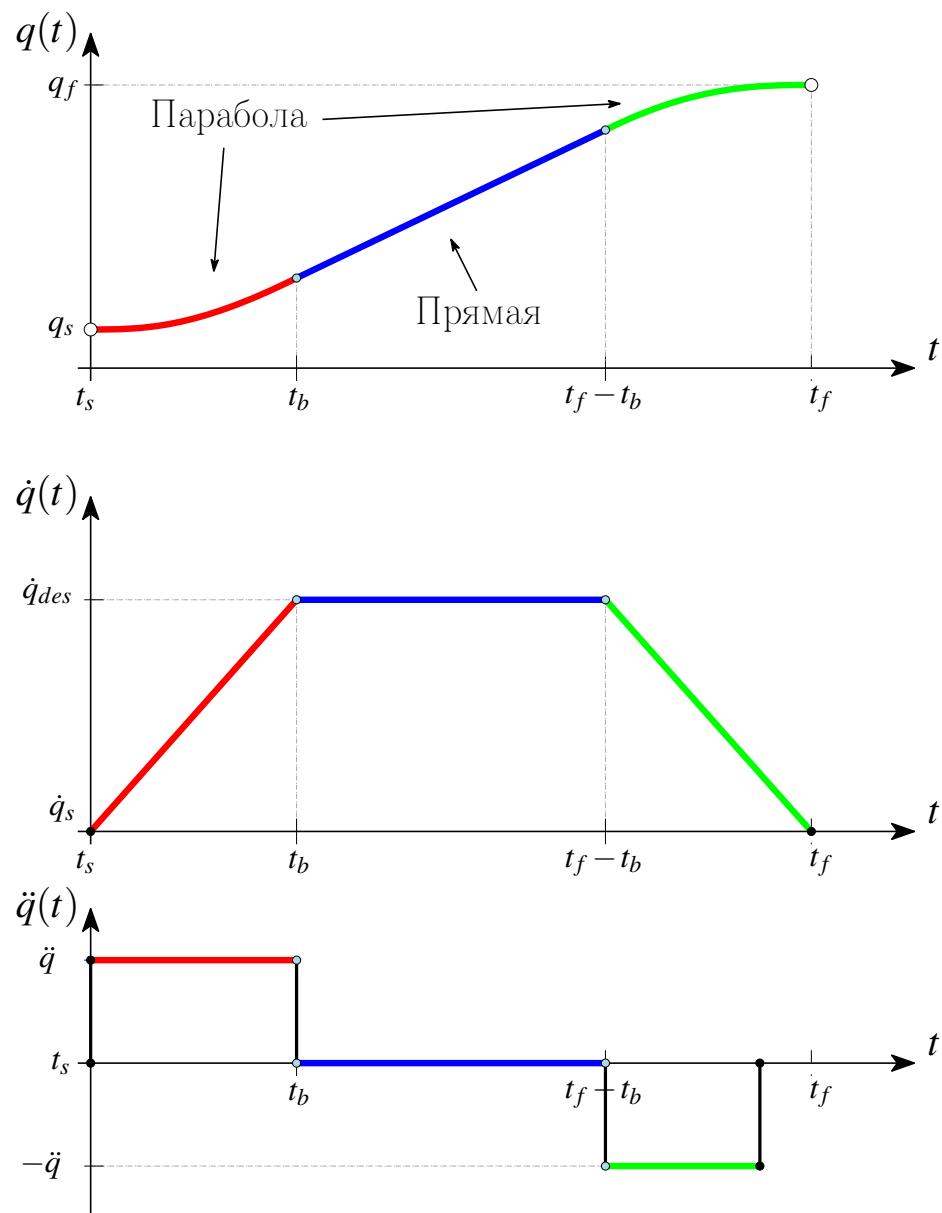


Рисунок 3.3 – Траектория с постоянной скоростью на среднем участке

Скорости получаем из:

$$\dot{q}(t) = \begin{cases} at, & t_s \leq t \leq t_b, \\ v, & t_b < t \leq t_f - t_b, \\ at_f - a * t, & t_f - t_b < t \leq t_f. \end{cases} \quad (3.10)$$

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

Ускорения:

$$\dot{q}(t) = \begin{cases} a, & t_s \leq t \leq t_b, \\ 0, & t_b < t \leq t_f - t_b, \\ -at_f - at, & t_f - t_b < t \leq t_f \end{cases} \quad (3.11)$$

Для сравнения, реализуем менее примитивный способ планирования траектории с использованием полинома пятой степени. Расчёты будем производить для нормализованного времени $t \in [0, 1]$.

Запишем полином пятой степени:

$$q_i(t) = a_i + b_i t + c_i t^2 + d_i t^3 + e_i t^4 + f_i t^5. \quad (3.12)$$

Дважды продифференцируем (3.12) по t :

$$\dot{q}_i t = b_i + 2c_i t + 3d_i t^2 + 4e_i t^3 + 5f_i t^4, \quad (3.13)$$

$$\ddot{q}_i t = 2c_i + 6d_i t + 12e_i t^3 + 20f_i t^3. \quad (3.14)$$

Принудим полином (3.12) и его производные подчиняться следующим условиям:

$$q(0) = q_s, \quad q(1) = q_f, \quad (3.15)$$

$$\dot{q}(0) = q'_s, \quad \dot{q}(1) = q'_f, \quad (3.16)$$

$$\ddot{q}(0) = 0, \quad \ddot{q}(q) = 0, \quad (3.17)$$

где q_s, q_f — точки начала и конца траектории, q'_s, q'_f — скорости в этих точках, а ускорения полагаем равными нулю.

В соответствии с условиями (3.15)-(3.17) легко можно получить шесть уравнений с тремя неизвестными. Опустим элементарные преобразования и сразу запишем выражения для расчета коэффициентов полинома:

$$a = q_s, \quad b = q'_s, \quad c = 0, \quad (3.18)$$

$$d = 10(q_f - q_s) - (6q'_s + 4q'_f), \quad (3.19)$$

$$e = -15(q_f - q_s) + (8q'_s + 7q'_f), \quad (3.20)$$

$$f = 6(q_f - q_s) - (3q'_f + 3q'_s). \quad (3.21)$$

Изв.	Лист	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата
Изв. № подп.	Подп. и дата				

Для расчёта траектории запишем уравнения в матричном виде:

$$Q = TC, \quad Q' = TC', \quad Q'' = TC'', \quad (3.22)$$

где, для $t = [t_1, t_2, \dots, t_f]$:

$$T = \begin{bmatrix} 1 & t & t^2 & t^3 & t^4 & t^5 \end{bmatrix}, \quad (3.23)$$

Q, Q', Q'' — матрицы, размерности $[m \times 6]$, $m = \frac{t_f - t_s}{dt}$,

$$C = \begin{bmatrix} a & b & c & d & e & f \end{bmatrix}^T, \quad C' = \begin{bmatrix} b & 0 & 3d & 4e & 5f & 0 \end{bmatrix}^T, \quad (3.24)$$

$$C'' = \begin{bmatrix} 0 & 6d & 12e & 20f & 0 & 0 \end{bmatrix}^T, \quad (3.25)$$

где все коэффициенты, представлены векторами.

3.3.2 Движение через последовательность точек

Предсказуемое движение всего манипуляционного механизма в рабочем пространстве обеспечим планированием гладкой траектории через заданные узловые точки q_i в пространстве обобщенных координат.

Введем обозначения:

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & \dots & q_n \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} t_1 & t_2 & \dots & t_n \end{bmatrix}, \quad (3.26)$$

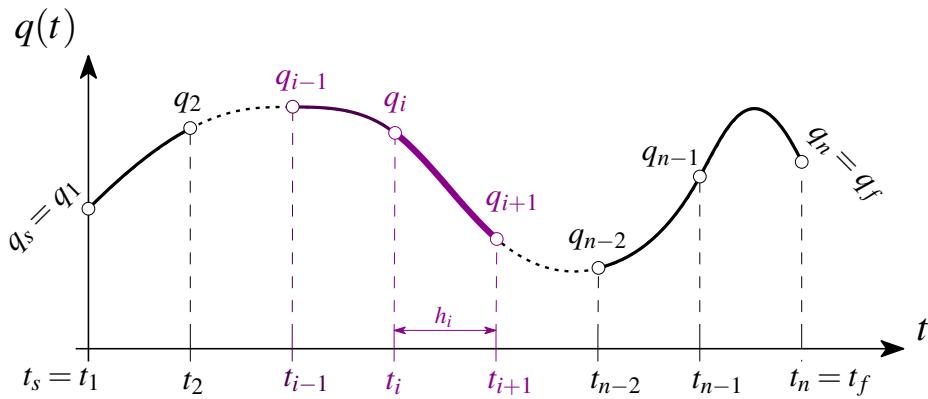
где \mathbf{q} — вектор, составленный из узловых точек q_i , \mathbf{t} — вектор временных отмечок t_i , в которые манипулятор должен пройти через q_i узел, причем

$$t_s = t_1 < t_2 < \dots < t_n = t_f. \quad (3.27)$$

На рисунке 3.4 показана последовательность точек в пространстве обобщенных координат.

Решим поставленную задачу, воспользовавшись одним из наиболее распространенных вариантов аппроксимации неизвестной функции — интерполяцией кубическими сплайнами. Разыскиваемая функция составляется из $n - 1$ кубических сплайнов, сшитых между собой в точках q_i , $i = \overline{2, n - 1}$ по первой и второй производной сплайна. Дополнительно устанавливаются ограничения на скорости или ускорения в начальной и конечной точках функции.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Рисунок 3.4 – Точки q_0, q_1, \dots, q_n , через которые должен пройти манипулятор

Запишем кубический сплайн на интервале времени $t \in [t_i, t_{i+1}]$:

$$q_i(t) = a_i + b_i(t - t_i) + c_i(t - t_i)^2 + d_i(t - t_i)^3, \quad i = \overline{1, n-1}, \quad (3.28)$$

где $q_i(t)$ — сплайн описывающий кривую между точками q_i и q_{i+1} , a_i, b_i, c_i, d_i — подлежащие расчёту коэффициенты сплайна, определяемые из дополнительных условий, i — номер сплайна.

Отсюда видно, что для описания функции проходящей через n точек, имеем $n - 1$ уравнений и $4n$ неизвестных.

Примем за шаг сплайна следующее соотношение:

$$h_i = t_{i+1} - t_i. \quad (3.29)$$

Продифференцируем дважды сплайн (3.28) по t :

$$q'_i(t) = b_i + 2c_i(t - t_i) + 3d_i(t - t_i)^2, \quad (3.30)$$

$$q''_i(t) = 2c_i + 6d_i(t - t_i). \quad (3.31)$$

Используя соотношение (3.29), запишем условия сшивания сплайнов i и $(i+1)$ для $i = \overline{1, n-1}$:

а) конец i должен совпадать с началом $(i+1)$:

$$q_i(t_{i+1}) = q_{i+1}(t_i) \Rightarrow a_{i+1} = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3. \quad (3.32)$$

Имеем: $3n$ — неизвестных, n — уравнений,

Изв.	Подп.	Порядк. и дата	Взам. Изв. № дубл.	Изв. № дубл.	Подп. и дата

б) первые производные сплайнов в точке сшивания должны быть равны:

$$q'_i(t_{i+1}) = q'_i + 1(t_i) \Rightarrow b_{i+1} = b_i + 2c_i h_i + 3d_i h_i^2. \quad (3.33)$$

Имеем: $3n$ — неизвестных, $2n$ — уравнений,

в) вторые производные сплайнов в точке сшивания должны быть равны:

$$q''_i(t_{i+1}) = q''_i + 1(t_i) \Rightarrow c_{i+1} = c_i h_i + 3d_i h_i, \quad (3.34)$$

Имеем: $3n$ — неизвестных, $3n$ — уравнений.

Для расчёта коэффициентов сплайна проведём следующие преобразования над (3.32)–(3.34). Выразим d_i из (3.34):

$$d_i = \frac{c_{i+1} - c_i}{3h_i}. \quad (3.35)$$

Подставив (3.35) в (3.32) и (3.33) и упростили, получим:

$$a_{i+1} = a_i + b_i h_i + \frac{h_i^2}{3} \cdot (2c_i + c_{i+1}), \quad (3.36)$$

$$b_{i+1} = b_i + h_i(c_i + c_{i+1}). \quad (3.37)$$

Выразим b_i из (3.36):

$$b_i = \frac{1}{h_i} \cdot (a_{i+1} - a_i) - \frac{h_i}{3} \cdot (2c_i + c_{i+1}). \quad (3.38)$$

Уменьшив значение индекса i на единицу в (3.37) и (3.38), получим:

$$b_{i-1} = \frac{1}{h_{i-1}} \cdot (a_i - a_{i-1}) - \frac{h_{i-1}}{3} \cdot (2c_{i-1} + c_i), \quad (3.39)$$

$$b_i = b_{i-1} + h_{i-1}(c_{i-1} + c_i). \quad (3.40)$$

После подстановки (3.38) и (3.39) в (3.40) и некоторых элементарных преобразований, запишем:

$$\underbrace{h_{i-1} c_{i-1} + 2(h_{i-1} + h_i) c_i + h_i c_{i+1}}_{\alpha_i} = \underbrace{\frac{3}{h_i} \cdot (a_{i+1} - a_i)}_{\beta_i} - \underbrace{\frac{3}{h_{i-1}} \cdot (a_i - a_{i-1})}_{\gamma_i}, \quad (3.41)$$

Извм.	Лист	№ докум.	Подп.	Дата	Подп. и дата	Извм. № дубл.	Бзм. извм. №	Подп. и дата

Отсюда видно: $n - 1$ — уравнений, $n + 1$ — неизвестных.

Добавим граничные условия, зафиксировав концы крайних сплайнов, обеспечивающие заданные первые производные q'_s и q'_f в точках t_s и t_f соответственно. Используя (3.38), получим:

$$q'(t_s) = q'_i(t_s) = b_1 = q'_s \Rightarrow h_1(2c_1 + c_2) = \frac{3}{h_1} \cdot (a_2 - a_1) - 3q'_s, \quad (3.42)$$

$$q'(t_f) = q'_n(t_f) = b_n = q'_f \Rightarrow \quad (3.43)$$

$$\Rightarrow h_{n-1}(2c_n + c_{n-1}) = 3q'_f - \frac{3}{h_{n-1}} \cdot (a_n - a_{n-1}). \quad (3.44)$$

Таким образом, $n - 1$ уравнений (3.41) при $i = \overline{1, n - 1}$ вместе с условиями (3.32)-(3.34) и (3.42) образуют систему линейных алгебраических уравнений для определения коэффициентов c_i . Коэффициенты b_i и d_i рассчитываются из выражений (3.35) и (3.38), коэффициенты a_i равны значениям аппроксимируемой функции в узлах (3.26).

Запишем уравнения (3.41) с учетом введенных там обозначений $\alpha_i, \beta_i, \gamma_i$ и δ_i в матричном виде:

$$Ac = \delta, \quad (3.45)$$

где A — трехдиагональная матрица, содержащая коэффициенты $\alpha_i, \beta_i, \gamma_i, c$ — вектор разыскиваемых коэффициентов c_i , δ — вектор, составленный из соотношений правой части уравнения (3.41).

Воспользуемся алгоритмом Томаса [10] (метод прогонки) для решения уравнения (3.45), содержащего трехдиагональную матрицу.

Перепишем выражение (3.41) в виде:

$$c_i = z_i - \mu_i \cdot c_{i+1}, \quad (3.46)$$

где прогоночные коэффициенты:

$$z_i = \frac{\delta_i - \alpha_i z_{i-1}}{\beta_i + \alpha_i \mu_{i-1}}, \quad \mu_i = \frac{\gamma_i}{\beta_i + \alpha_i \mu_{i-1}}. \quad (3.47)$$

Алгоритм решает уравнение за два прохода:

Изв. № подп.	Подп. и дата	Бзм. изв. №	Изв. № дубл.	Подп. и дата

а) Прямой ход. Рассчитываются начальные условия (3.47) при $\alpha_1 = 0$:

$$\mu_1 = \frac{\gamma_1}{\beta_1} = \frac{h_1}{2h_1} = 0.5, \quad z_1 = \frac{\delta_1}{\beta_1}. \quad (3.48)$$

Вычисляются прогоночные коэффициенты z_i, μ_i из выражения (3.47) для $i = \overline{2, n-1}$,

б) Обратный ход. Вычисляется $z_n = c_n$. Вычисляются коэффициенты c_i для $i = \overline{n-1, 1}$.

Результат работы на примере интерполяции синуса, показан на рисунке 3.5.

Планирование траектории должно осуществляться перед началом работы. При реализации описаного метода может случиться так, что скорости и ускорения некоторых из компонентов векторов обобщенных координат будут превышать допустимые значения скоростей и ускорений, которые могут развить приводы манипулятора. Один из способов борьбы с этим — увеличить время прохождения траектории на участке, на котором ограничение нарушено.

3.4 Планирование траектории в пространстве координат схвата

Планирование траектории в пространстве координат схвата включает в себя два этапа: на первом этапе планируется траектория относительно базовой СК робота, на втором этапе — полученная траектория переводится в пространство обобщенных координат робота и трансформируется в рассмотренную выше задачу.

Для планирования траектории захвата движущегося по конвейеру объекта, получим аналитические выражения для геометрического пути следования этого объекта. Дальнейшие рассуждения и обозначения наглядно представлены на рисунке 3.6.

Необходимо представить траекторию, как зависимость

$$T = T(t), \quad (3.49)$$

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

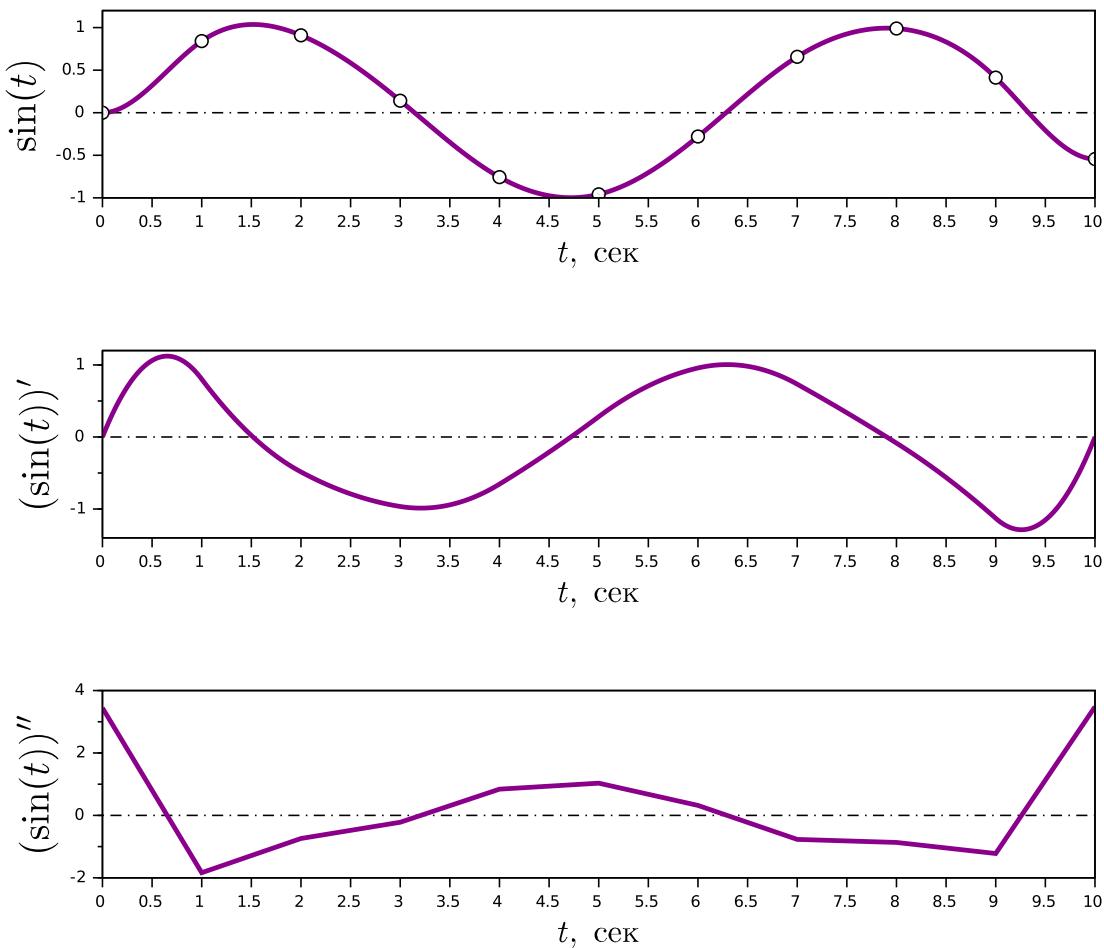


Рисунок 3.5 – Пример результата интерполяции набора из одиннадцати точек функции $\sin(t)$

которая удовлетворяет требованиям плавного разгона и торможения, тогда

$$T(t) = T_0 \cdot A(\vec{n}, \delta(t), \rho(t)), \quad (3.50)$$

где $A(t)$ матрица однородных преобразований между точкой траектории и вектором обобщенных координат:

$$A(t) = \begin{bmatrix} R(t) & \rho(t) \\ 000 & 1 \end{bmatrix}. \quad (3.51)$$

Заметим, что в соответствии с рисунком 3.6 и (3.50):

$$T(t_0) = T_0, \quad T(t_0 + \tau) = T_0 \cdot T_f^{-1}, \quad (3.52)$$

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Полп. и дата

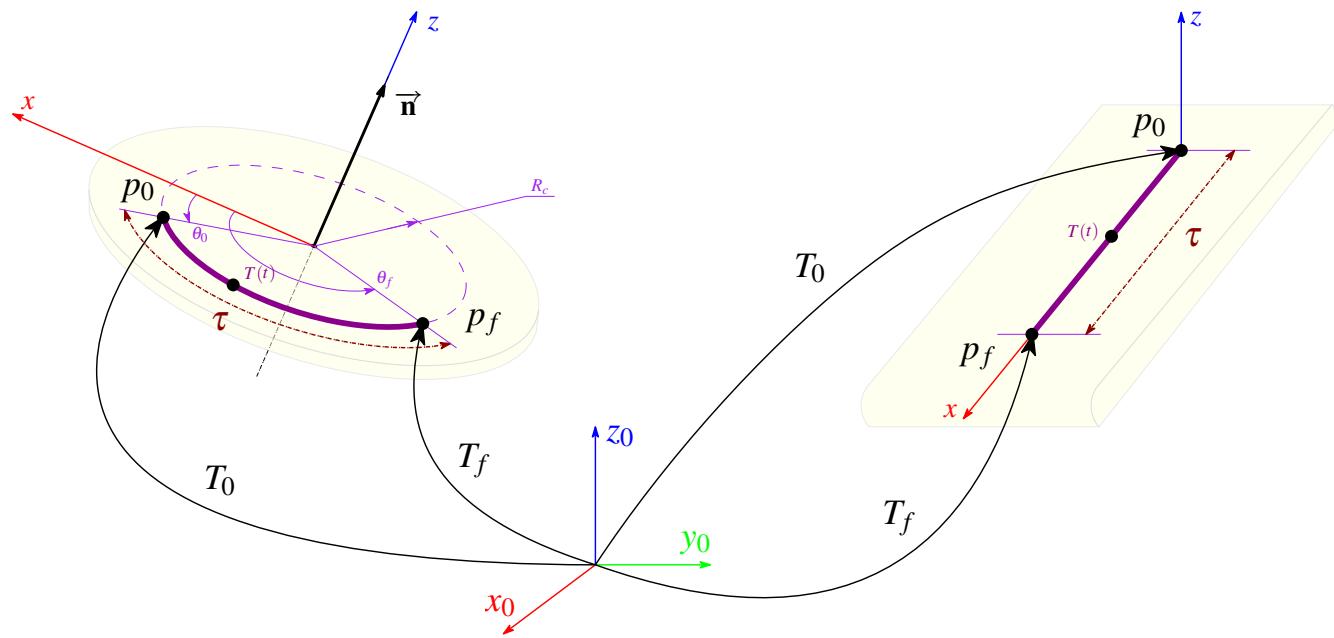


Рисунок 3.6 – Планирование траекторий на конвейерах: слева — вращающийся стол; справа — линейный конвейер

$$R(t_0) = E, \quad R(t_0 + \tau) = R_0 \cdot R_f. \quad (3.53)$$

$$\rho(t_0) = 0, \quad \rho(t_0 + \tau) = R_0^T(r_f - r_0), \quad (3.54)$$

где r_0 и r_f — векторы переноса составленные из первых трех компонент векторов \mathbf{p}_0 и \mathbf{p}_f .

Найдем вектор \vec{n} , вокруг которого осуществляется поворот из следующих выражений:

$$\delta(t_0 + \tau) = \arccos \frac{r_{11} + r_{22} + r_{33} - 1}{2}, \quad (3.55)$$

$$\mathbf{n} = \frac{1}{2 \sin \delta} \cdot \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}, \quad (3.56)$$

где r_{ij} — элементы матрицы $R(t_0 + \tau)$.

Окончательно, матрица поворота рассчитывается из выражения:

$$R(t) = \cos \delta(t) \cdot E + (1 - \cos \delta(t)) \cdot \mathbf{n} \cdot \mathbf{n}^T + \sin \delta(t) \cdot \Omega_n, \quad (3.57)$$

где Ω_n — кососимметрическая матрица для вектора \mathbf{n} .

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Функции $\rho(r)$ и $\delta(t)$ выбираются исходя из условия обеспечения плавности разгона и торможения.

Для случая конвейера, представленного вращающимся столом, объект покоящийся на столе описывает окружность, закон движения по которой в плоскости параллельной плоскости OXY:

$$\begin{cases} x(t) = R_c \cos \theta(t) \\ y(t) = R_c \sin \theta(t), \\ z(t) = \text{const} \end{cases} \quad (3.58)$$

где R_c — радиус окружности, описываемой объектом при вращении стола, $\theta(t)$ — поворот стола.

Тогда, учитывая, что закон движения схвата манипулятора должен в точности повторять движение подвижного объекта, параметризуем закон движения схвата параметром поворота вращающегося стола $\theta = \theta(t)$, получим:

$$\mathbf{p} = \mathbf{p}(\theta) = \begin{bmatrix} R_c \cos \theta \\ R_c \sin \theta \\ z \\ 1 \end{bmatrix}, \quad \theta \in [\theta_0, \theta_f]. \quad (3.59)$$

При этом нужно, чтобы выполнялось условие непрерывности траектории

$$\dot{\mathbf{p}} = \frac{d\mathbf{p}}{d\theta} \neq 0, \quad \forall \theta \in [\theta_0, \theta_f]. \quad (3.60)$$

Траектория ориентации схвата определяется выражением (3.57)

В этом разделе были проанализированы конфигурация и необходимая ориентация схвата в рабочем пространстве манипулятора робота KUKA Yobot. После этого были решены задачи планирования траекторий для перевода манипулятора из точки в точку, для обхода последовательности точек и для задания закона движения в пространстве координат схвата.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

4 Разработка системы технического зрения

Использование системы технического зрения (СТЗ) в разрабатываемой робототехнической системе обусловлено необходимостью наделить СУ манипулятора способностью самостоятельно выбирать свое поведение в соответствии с изменениями окружающей среды.

СТЗ состоит из RGBD видеокамеры Intel RealSense SR300, компьютера и программного обеспечения. Камера закреплена на пятом звене манипулятора и имеет матрицу трансформации из внутренней СК камеры в СК $Ox_5y_5z_5$, причем, оптическая ось камеры и Oz_5 сонаправлены.

Технические характеристики сенсоров видеокамеры Intel RealSense SR300 приведены в таблицах 4.1–4.2, а внешний вид представлен на рисунке 4.1. Принцип получения карты глубины схематично изображен на рисунке 4.2.



Рисунок 4.1 – Внешний вид 3D-камеры Intel RealSense SR300

Таблица 4.1 – Параметры инфракрасной и цветной камер RealSense SR300

Параметр	ИК-камера	RGB-камера
Разрешение, пикс	640×480	1920×1080
Вертикальный угол обзора, град	55 ± 2	41.5 ± 2
Горизонтальный угол обзора, град	71.5 ± 2	68 ± 2
Диагональный угол обзора, град	88 ± 3	75.2 ± 4

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

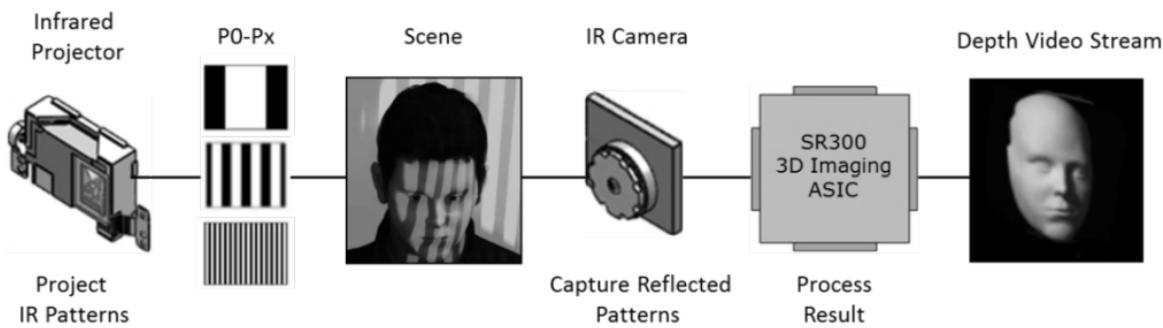


Рисунок 4.2 – Принцип получения карты глубины

Таблица 4.2 – Параметры лазерного проектора камеры Intel RealSense SR300

Параметр	Описание
Проектор	Структурированный свет
Длина волны лазера, нм	860
Безопасность лазера	Class 1
Вертикальный угол проекции, град	60 ± 4
Горизонтальный угол проекции, град	72.5 ± 2

4.1 Постановка и описание задачи

Требуется оценить параметры траектории движения объекта покоящегося на вращающемся столе, пример которого представлен на рисунках 3.2–3.6. Траектория движения — окружность. Подлежащие оценке параметры: R_c — радиус окружности, которую описывает объект при вращении стола, $Oxyz$ — СК вращающегося стола, где Oz — ось вращения, \mathbf{p}_0 и \mathbf{p}_f — точки входа в нормальную рабочую область манипулятора и выхода из нее соответственно, а также векторы \mathbf{s}_o и $\dot{\mathbf{s}}_o$ — положение объекта в пространстве и скорость, выраженные в СК $Ox_0y_0z_0$:

$$\mathbf{s}_o = \begin{bmatrix} \mathbf{p}_o \\ \mathbf{r}_o \end{bmatrix}, \quad \dot{\mathbf{s}}_o = \begin{bmatrix} \mathbf{v}_o \\ \boldsymbol{\omega}_o \end{bmatrix}. \quad (4.1)$$

Объект представлен в лице прямоугольного параллелепипеда.

Выделим на вращающемся столе две зоны: приближение объекта и захвата. Тогда, сценарий работы СТЗ следующий:

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

- а) Манипулятор принимает конфигурацию, при которой в кадр камеры попадает зона приближения объекта;
- б) Проводится процедура выделения объекта из окружающей среды и определение его геометрии, с последующим оцениванием векторов его положения \mathbf{s}_o и скорости $\dot{\mathbf{s}}_o$;
- в) По собранной статистике о положении и ориентации объекта на столе за время Δt , производится оценка оси вращения стола \vec{n} и радиуса кривизны траектории движения объекта R_c .
- г) Перед тем, как закончить свою работу, СТЗ определяет точки входа \mathbf{p}_0 в зону захвата и выхода \mathbf{p}_f из нее объекта. После этого вся накопленная информация пересыпается в генератор траекторий для манипулятора.

Первый пункт рассматривается в разделе 2.3, последний в 3, а второй и третий реализуем в этом разделе.

Прежде, чем приступить к решению поставленных задач, сделаем два замечания:

- а) Использование камеры Intel RealSense SR300 позволяет упростить решение поставленных задач, так как позволяет сразу работать с *облачом точек* — набором вершин в трехмерном пространстве, координаты которых выражены в СК камеры; также, каждая точка может иметь цвет, например, в цветовой модели RGB;
- б) Работа с глубинными картами предполагает большой объем вычислений, поэтому необходим достаточно производительный компьютер.

Непрерывно изменяющееся состояние окружающего мира — вращение стола, — обязывает робототехническую систему работать в реальном времени, так как нет априорной информации о расположении объектов на нем. В связи с этим, необходимо найти компромисс между точностью работы системы и скоростью вычислений обеспечивающих достаточную точность.

Как видно из таблицы 4.1, разрешение получаемой карты глубины и, соответственно, облака точек равно 640×480 , что соответствует 307200 точкам,

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № глубы	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

представляющих из себя, в простейшем случае, структуру из трех координат положения точки и трех векторов, задающих ориентацию, т.е. в сумме, один кадр облака может содержать от четырехсот байт. При работе с частотой 30 кадров в секунду возрастает до двенадцати мегабайт. Дополнительным аргументом становится тот факт, что при уменьшении количества точек в облаке до некоторого порога, возможно добиться не худшей точности, чем при использовании всех точек. Из этих соображений имеет смысл понизить разрешение.

Среди огромного количества фильтров разной степени сложности, выберем наиболее простой, суть которого заключается в представлении облака точек в виде восьмиричного дерева — октодерева. Октодерево делит облако точек на октанты (воксели), каждый из которых рекурсивно делится до тех пор, пока каждая из точек не будет в отдельном октанте. Такое представление позволяет задавать разрешение облака точек простым изменением глубины октодерева [11]. Пример деления представлен на рисунке 4.3.

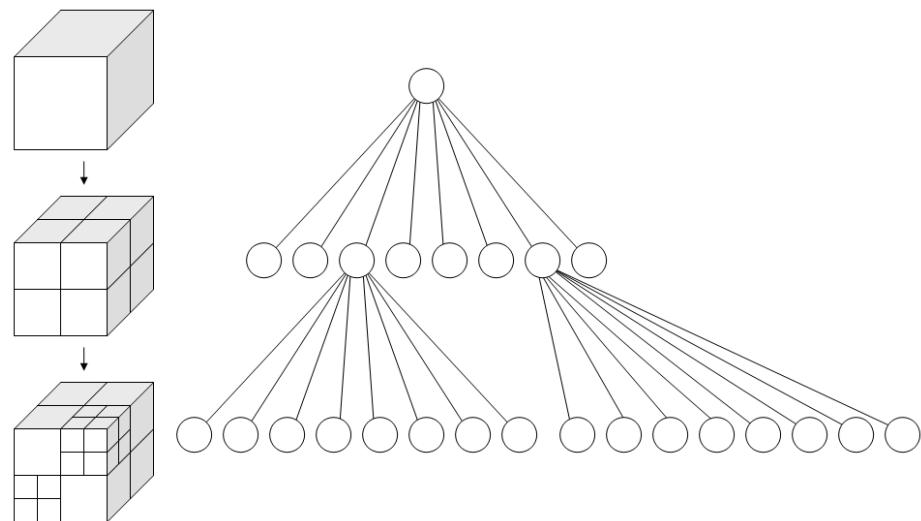


Рисунок 4.3 – Изображение рекурсивного разделения куба на октанты и, соответствующее этому разделению, октодерево

Следующим этапом нужно определить плоскость вращающегося стола, на котором располагаются интересные нам объекты. Для этого воспользуемся методом оценки параметром плоскости на основе случайных выборок точек — RANSAC.

На вход алгоритма подается все облако точек, модель плоскости, кото-

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № глубы	Подп. и дата

рую нужно вписать в это болако, минимальное и максимальное число точек, которые могут входить в плоскость и максимальный разброс. На выходе получим параметры модели плоскости и точки, который входят в эту плоскость. На рисунке 4.4 изображен пример результата работы.

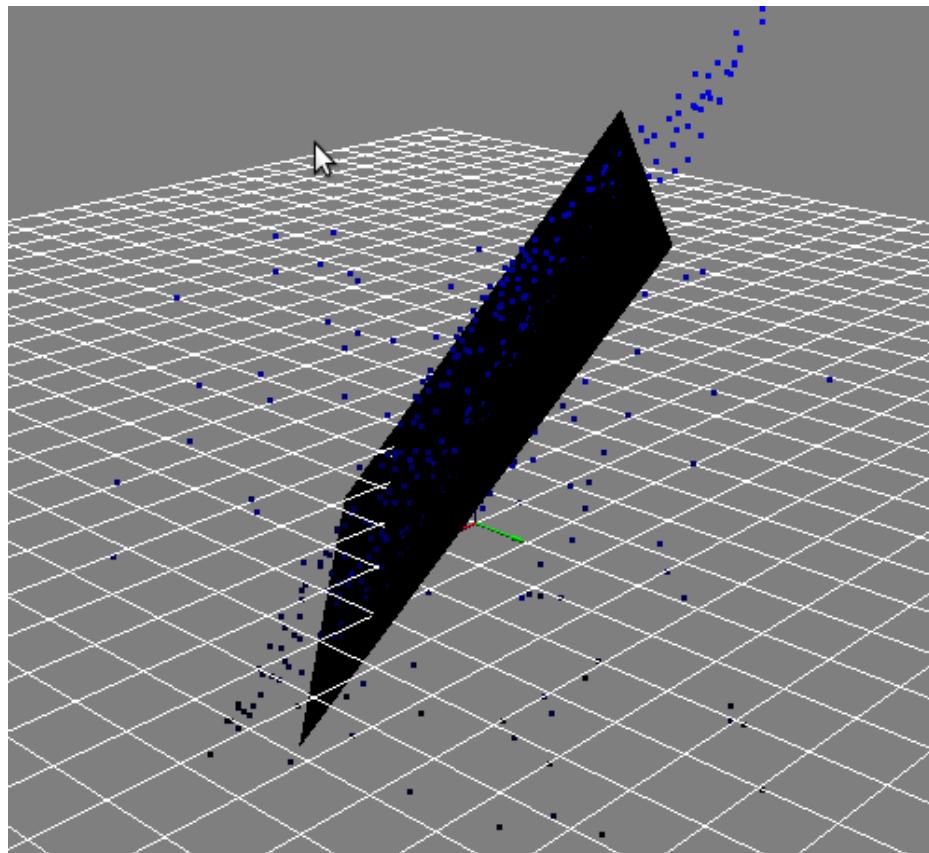


Рисунок 4.4 – Пример вписывания плоскости в облако точек алгоритмом RANSAC

После определения кластера облака точек, в который входят все точки самой большой плоскости в кадре, то есть плоскости конвейера, необходимо найти объекта на этой плоскости. Для этого, построим выпуклую оболочку кластера облака точек входящих в плоскость, получив таким образом некоторый многоугольник. Далее, отбросив все точки, которые не входят в множество, включающее только внутреннюю часть призмы заданной высоты построенной на этом многоугольнике, получим все объекты расположенные на столе. В этой работе рассматривается только один объект, но это решение легко можно расширить на большее их число путем деление кластера с объектами на еще более

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

мелкие кластеры, содержание каждый свой объект [12].

Теперь, имея облако точек, включающее только целевой объект, необходимо найти его центр масс и ориентацию. Для этого воспользуемся методом определения момента инерции облака точек. Идея метода заключается в следующем. Вычисляется ковариационная матрица точечного облака и извлекаются ее собственные значения и векторы, которые представляют собой оси x , y и z , образующие ортогональный правый базис. На каждой итерации выбирается один из полученных векторов, причем вращение всегда одинаковое, что обеспечивает инвариантность к вращению всего рассматриваемого облака точек. После этого, для каждой оси рассчитывается момент инерции.

Полученные значения поворота облака точек и его центр масс будем использовать для оценки траектории движения объекта. Причем, ориентацию объекта достаточно оценить один раз, а после задавать ее дополнительное слагаемое к углу поворота вращающегося стола.

Последним этапом работы технического зрения является отслеживание траектории движения объекта и одновременная идентификация параметров этой траектории заданной в параметрической форме окружности:

$$\begin{cases} x(t) = R_c \cos \theta(t) \\ y(t) = R_c \sin \theta(t), \\ z(t) = \text{const} \end{cases} \quad (4.2)$$

где R_c — радиус окружности, описываемой объектом при вращении стола, $\theta(t)$ — поворот стола.

Таким образом, нужно найти радиус дуги, которую описывает объект при вращении стола. Рассмотрим случай нахождения радиуса по трем известным координатам точек окружности $A(x_1, y_1)$, $B(x_2, y_2)$ и $C(x_3, y_4)$. Возможный пример такого случая изображен на рисунке 4.5

Запишем уравнения для прямых a и b :

$$y_a = k_a(x - x_1) + y_1, \quad y_b = k_b(x - x_2) + y_2, \quad (4.3)$$

где k_a — коэффициент наклона прямой a , k_b — коэффициент наклона прямой b .

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

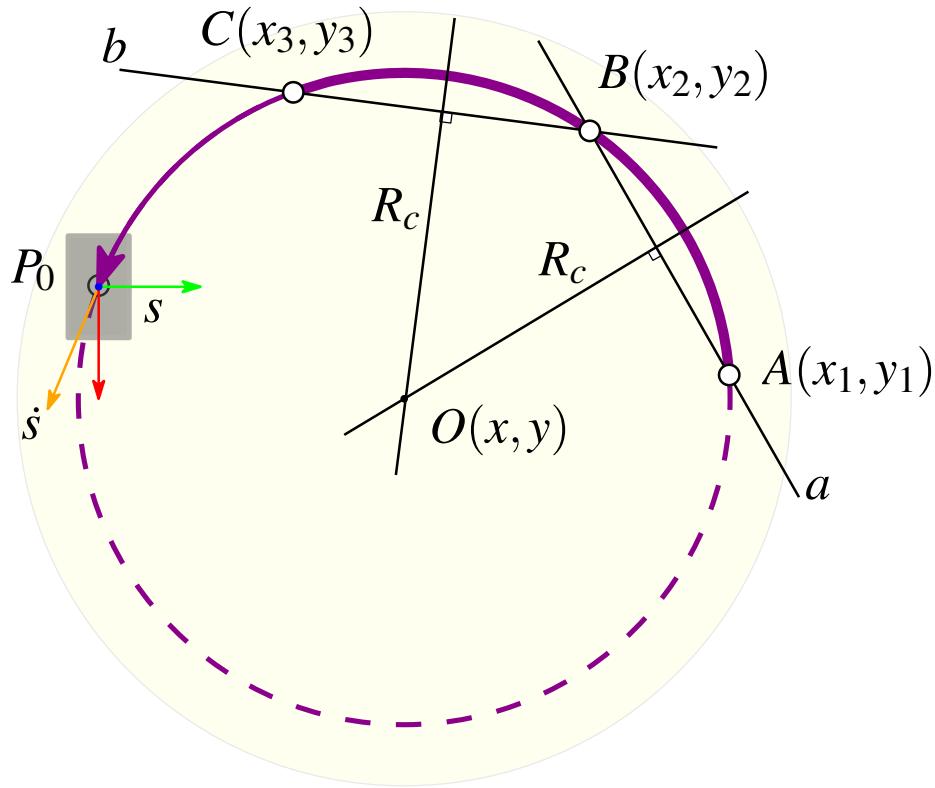


Рисунок 4.5 – Нахождение радиуса окружности по известным точкам, принадлежащим окружности

Коэффициенты рассчитываются из следующих соотношений:

$$k_a = \frac{y_2 - y_1}{x_2 - x_1}, \quad k_b = \frac{y_3 - y_2}{x_3 - x_2}. \quad (4.4)$$

Далее, найдем центр окружности, который находится на пересечении прямых проведенных через середины отрезков AB и BC . Учитывая, что коэффициент наклона перпендикуляра выражается как $k_{a,b} = -\frac{1}{m_{a,b}}$, запишем уравнения перпендикуляров:

$$y_{\perp a} = -\frac{1}{k_a} \cdot \left(x - \frac{x_1 + x_2}{2} \right) + \frac{y_1 + y_2}{2}, \quad y_{\perp b} = -\frac{1}{k_b} \cdot \left(x - \frac{x_2 + x_3}{2} \right) + \frac{y_2 + y_3}{2}. \quad (4.5)$$

И, так как полученные перпендикуляры пересекаются ровно в центре окружности, запишем для координат точки O :

$$x = \frac{k_a k_b (y_1 - y_3) + k_b (x_1 + x_2) - k_a (x_2 + x_3)}{2(k_b - k_a)}, \quad (4.6)$$

Изв.	Подп. и дата	Бзм. подп.	Изв. № дубл.	Подп. и дата

$$y = -\frac{1}{k_a} \cdot \left(\frac{k_a k_b (y_1 - y_3) + k_b (x_1 + x_2) - k_a (x_2 + x_3)}{2(k_b - k_a)} - \frac{x_1 + x_2}{2} \right) + \frac{y_1 + y_2}{2}.$$

Имея оценку координат центра окружности, по которой движется объект, легко найти радиус:

$$R_c = \sqrt{(x - x_3)^2 + (y - y_3)^2} \quad (4.7)$$

Таким образом, при слежении за объектом в области перед точкой P_0 для каждой полученной последовательности трех точек рассчитывается оценка радиусу. Со временем накапливаются возможные значения радиусов и координат его центра. Координаты центра следует отфильтровать, найдя центр минимальной окружности, которая охватывает большинство точек оценок координат центра траектории движения объекта.

Описанная последовательность действий позволяет определить положение и ориентацию объекта на вращающемся столе, а также момент входа объекта в область доступную для захватывания его манипулятором.

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

5 Результаты математического моделирования

В этом разделе приводятся результаты моделирования трех главных частей робототехнической системы: планирование траекторий, управления манипулятором и системы технического зрения.

Коммуникация между всеми частями робототехнической системы реализована на базе фреймворка Robot Operation System (ROS).

Алгоритмы планирования траекторий были реализованы и отлажены в пакете прикладных математических программ Scilab. Затем реализованы в виде модуля на языке программирования C++.

Моделирование системы управления манипулятором также проводилось в Scilab. При этом использовался Robotics Toolbox — набор блоков для моделирования манипуляторов [13], являющийся аналогом Robotics Toolbox Питера Корка для Matlab.

Система технического зрения разрабатывалась с использованием алгоритмов обработки облаков точек Point Cloud Library (PCL). Разработка велась на языке программирования C++, результатом которой стал ros-пакет.

5.1 Результаты планирования траектории

Построим траектории в пространстве обобщенных координат. На рисунке 5.1 изображены четыре выбранные конфигурации, соответствующие значениям углов поворота сочленений манипулятора, заданные векторами в (5.1).

$$q_0 = \begin{bmatrix} 2.95 \\ 1.35 \\ -2.59 \\ 1.58 \\ 0 \end{bmatrix}, \quad q_1 = \begin{bmatrix} 4.06 \\ 2.34 \\ -2.2 \\ 3.32 \\ 1.01 \end{bmatrix}, \quad q_2 = \begin{bmatrix} 3.07 \\ 1.75 \\ -0.93 \\ 2.54 \\ 1.73 \end{bmatrix}, \quad q_3 = \begin{bmatrix} 2.57 \\ 2.33 \\ -2.35 \\ 3.36 \\ 2.97 \end{bmatrix}. \quad (5.1)$$

На рисунках 5.2 и 5.3 показаны графики переходных процессов для переходов из точки в точку при планировании траектории сегментами и исполь-

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. №	Подп. и дата

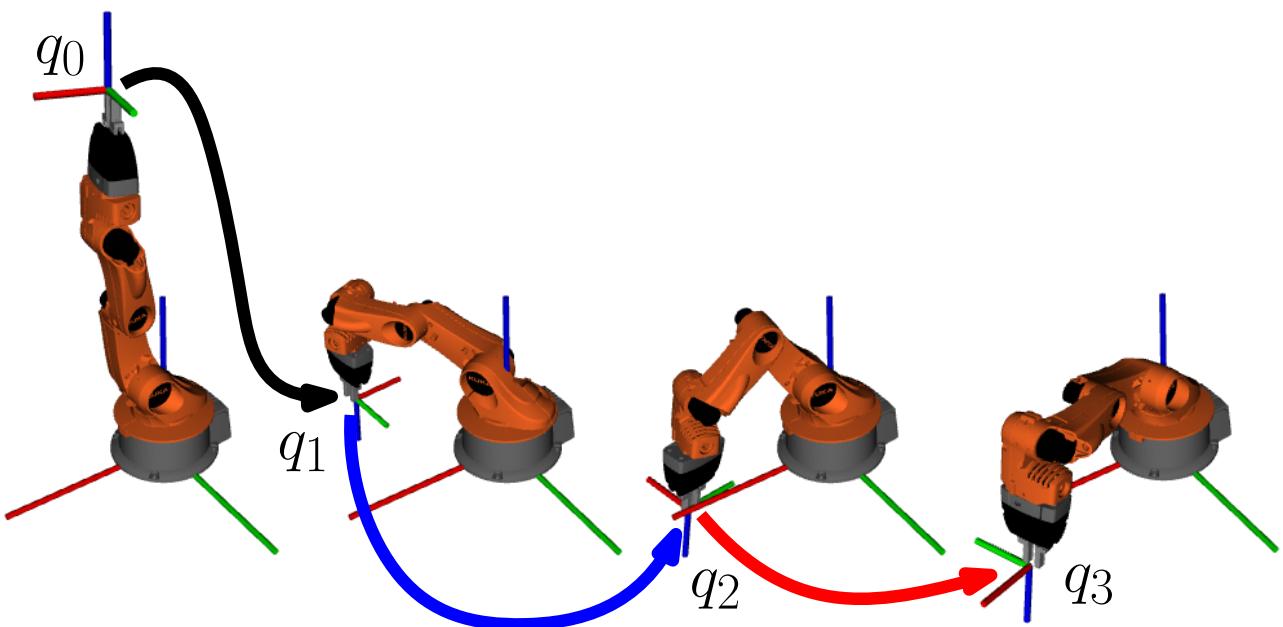
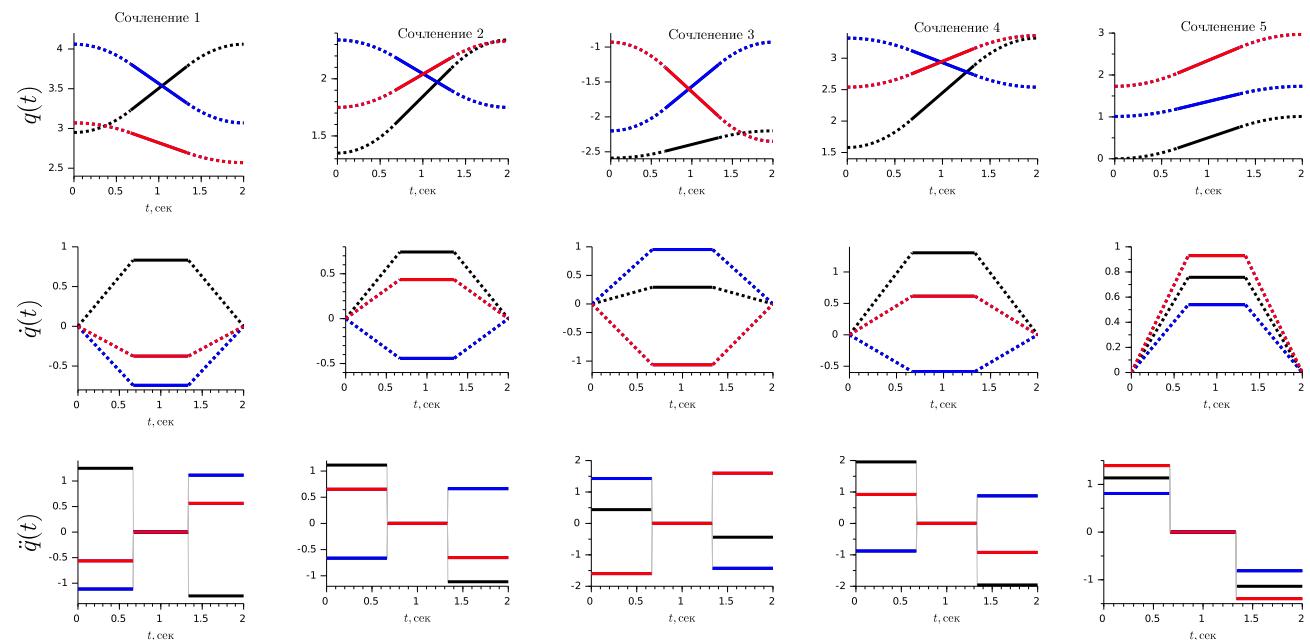


Рисунок 5.1 – Последовательность принятия положений заданных в 5.1

зования полинома пятой степени соответственно.

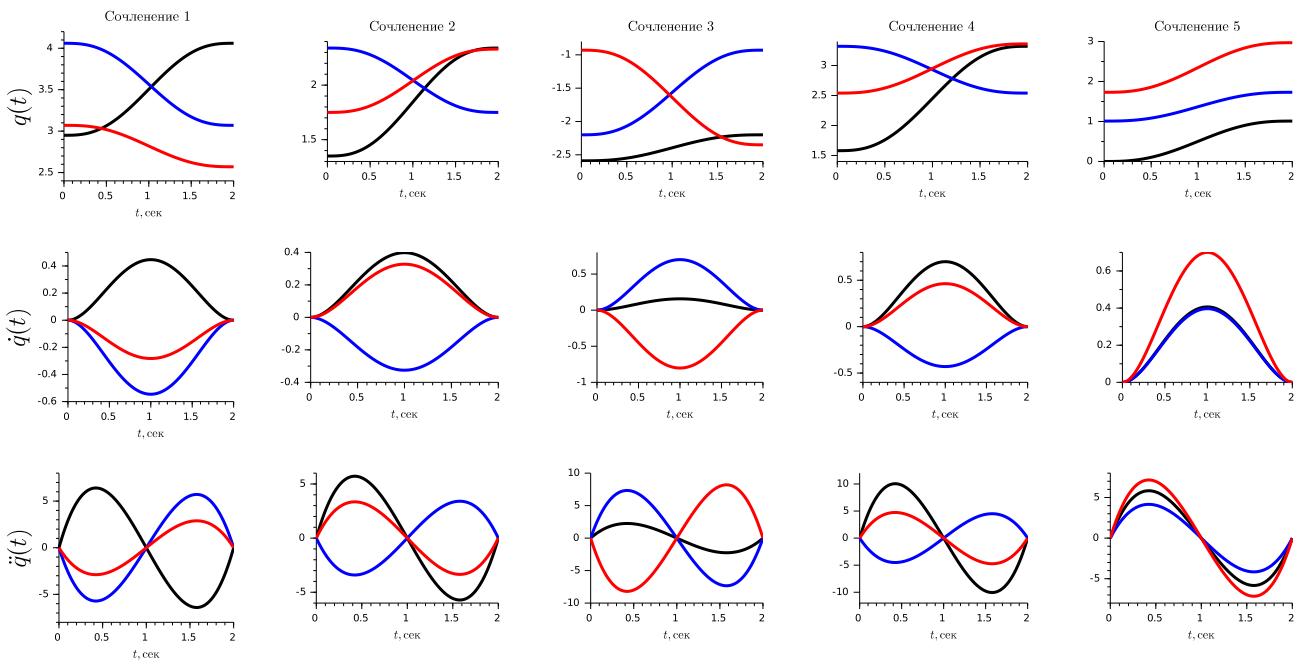


черный: для перехода $q_0 \rightarrow q_1$, синий: для перехода $q_1 \rightarrow q_2$, красный: для перехода $q_2 \rightarrow q_3$

Рисунок 5.2 – Траектория с трапециoidalным профилем скорости

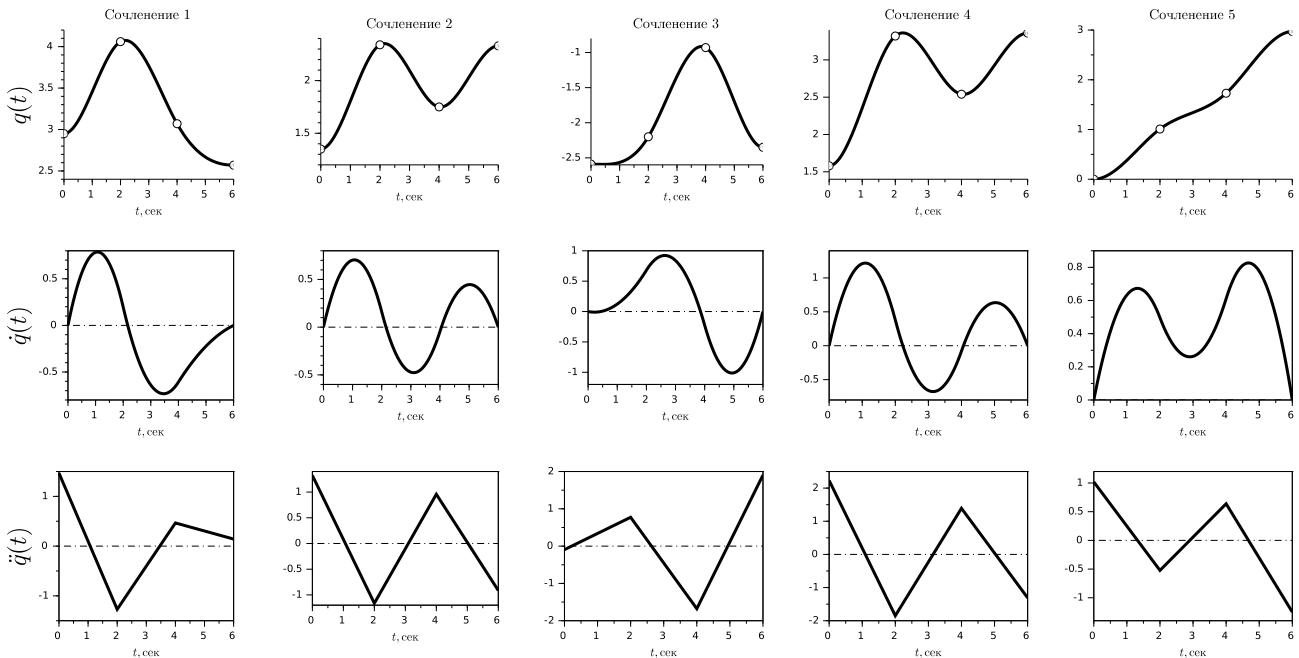
Траектория (дуга) в операционном пространстве и соответствующие ей траектории каждого из сочленений в конфигурационном пространстве пред-

Изм.	Лист	№ докум.	Подп.	Дата	Копировал	Формат А4	Лист
							55



черный: для перехода $q_0 \rightarrow q_1$, синий: для перехода $q_1 \rightarrow q_2$, красный: для перехода $q_2 \rightarrow q_3$

Рисунок 5.3 – Траектория при полиномиальной интерполяции



узлы на верхних графиках представляют из себя точки $q_0(0)$, $q_1(2)$, $q_2(4)$, $q_3(6)$

Рисунок 5.4 – Непрерывная траектория через набор точек

ставлены на рисунке 5.5.

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Полп. и дата

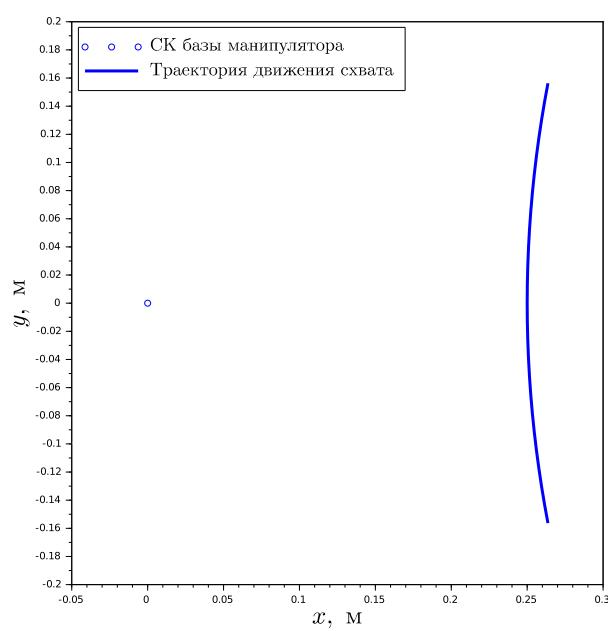


Рисунок 5.5 – Траектория движения схвата манипулятора и соответствующие ей траектории сочленений в конфигурационном пространстве

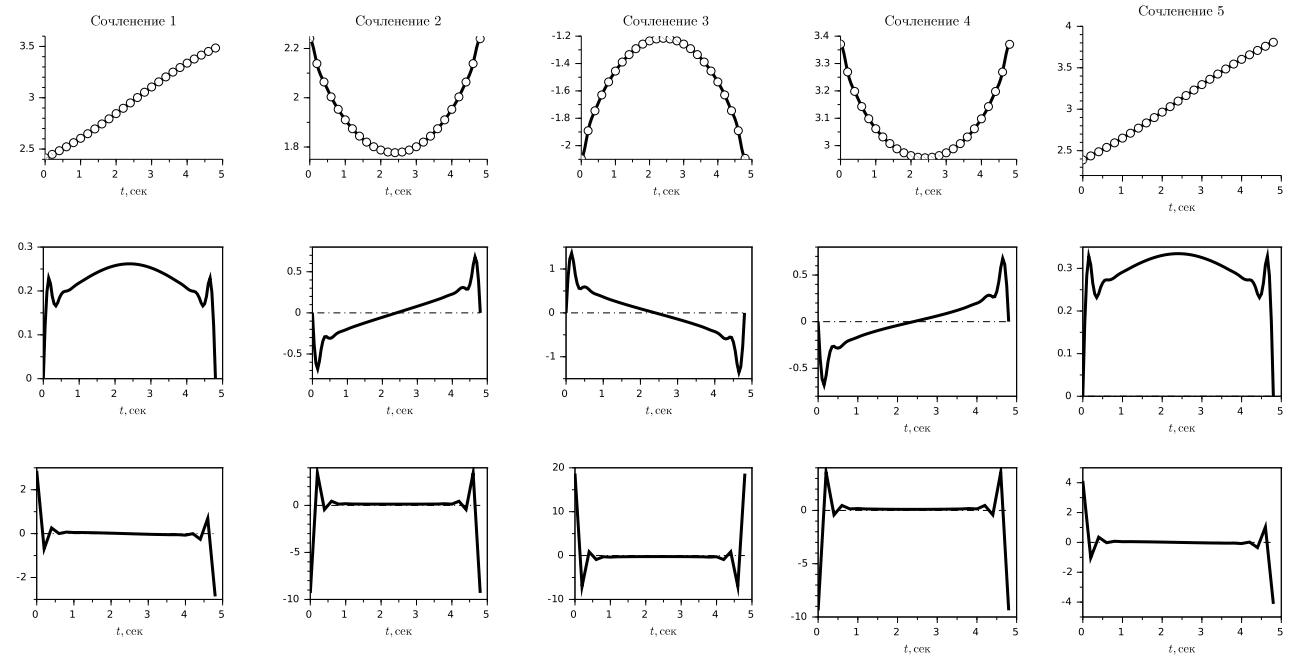


Рисунок 5.6 – Интерполяция траекторий кубическими сплайнами

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Полп. и дата
Изм.	Лист	№ докум.	Подп.	Дата

5.2 Моделирование системы управления манипулятором

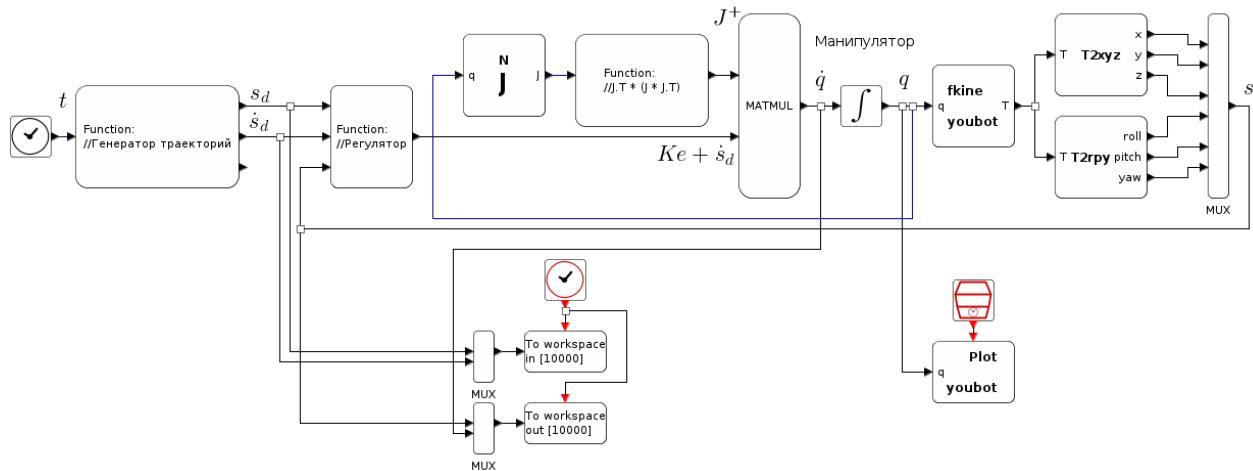
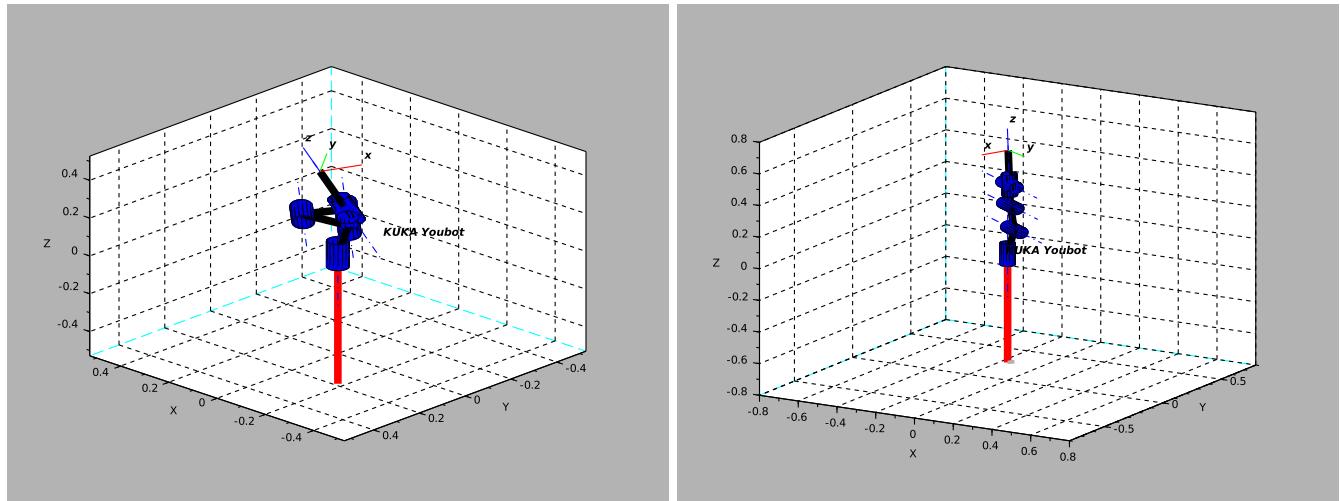


Рисунок 5.7 – Модель системы управления манипуляторов в Scilab

Рисунок 5.8 – а) Манипулятор KUKA YouBot в "домашней" конфигурации; б)
В свече

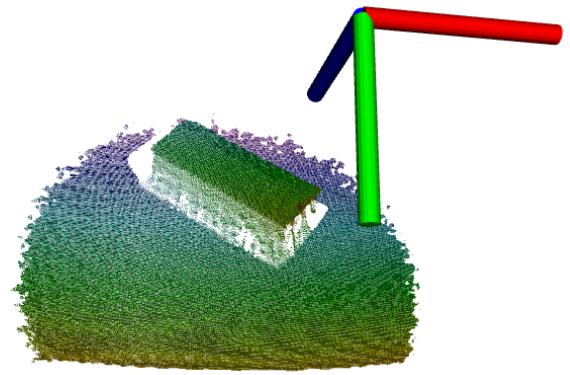
Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

5.3 Результаты работы системы технического зрения

Путь получения координат и ориентации объекта покажем на нижеследующей последовательности рисунков. На рисунке 5.9 изображено облако точек, полученное с камеры Intel RealSense SR300. После фильтрации этого облака, получается облако с пониженным качеством и показывается на рисунке 5.10. Далее, на рисунке 5.10б, изображена плоскость — результат работы алгоритма сегментации. Следующий рисунок 5.11а содержит результат нахождения выпуклой оболочки вокруг плоскости. И, облако точек целевого объекта, показано на рисунке 5.11б. Рисунок 5.12 включает плоскость, объект и СК прикрепленную к объекту.



а)



б)

Рисунок 5.9 – а) Цветное изображение целевого объекта; б) Необработанное облако точек

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

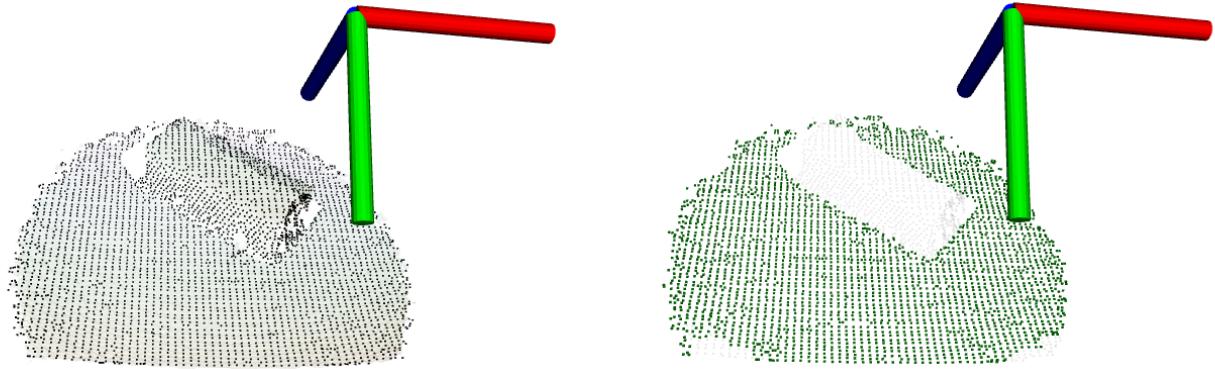


Рисунок 5.10 – а) Облако точек пропущенное через фильтр VoxelGrid; б) Часть облака точек принадлежащая плоскости в кадре самой большой площади

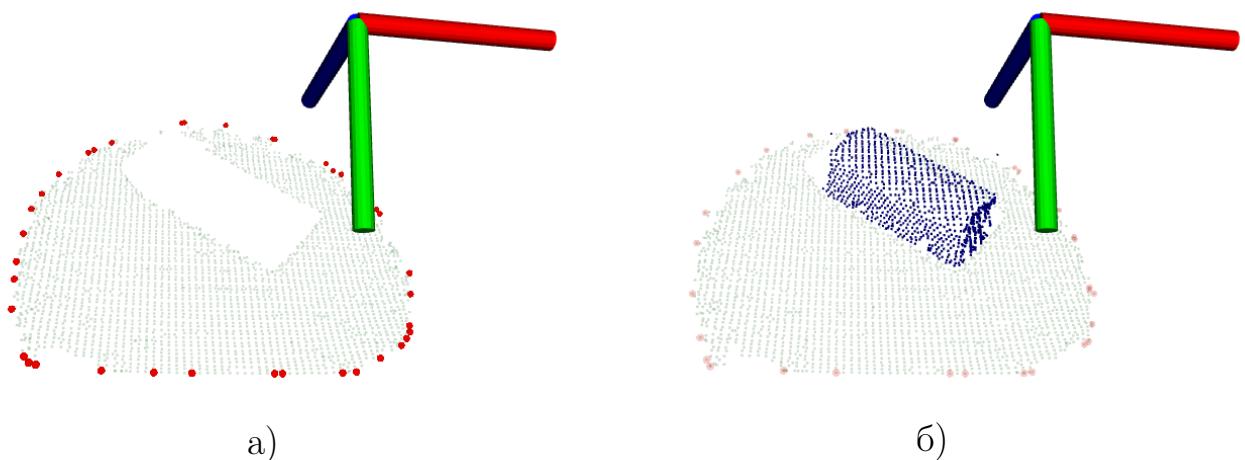


Рисунок 5.11 – а) Точки принадлежащие выпуклой оболочке, вокруг плоскости; б) Кластер облака точек включающих только целевой объект

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

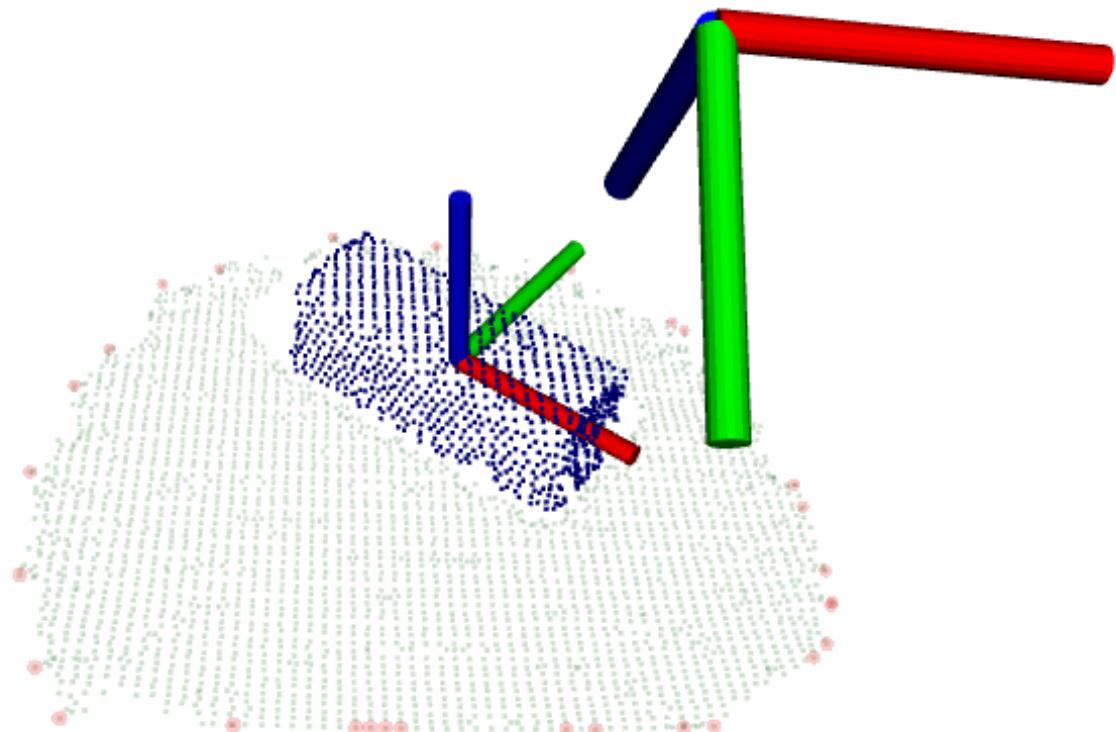


Рисунок 5.12 – Прикрепленная к объекту СК

Инв. № подп.	Подп. и дата	Взам. инв. №	Инв. № документа	Подп. и дата

Заключение

В рамках данной выпускной квалификационной работы было произведено моделирование синтезированной системы управления манипулятором на базе манипулятора робота KUKA YouBot для следования по заданной траектории, планирования траекторий и системы технического зрения для обнаружения целевого объекта на вращающемся столе и оценка траектории его движения.

В ходе работы были достигнуты следующие результаты:

- Построена кинематическая модель манипулятора KUKA Youbot, в рамках которой были решены прямая и обратная задачи по положению и скорости;
- Реализованы алгоритмы планирования траектории в пространстве обобщенных координат для перехода из точки в точку, для обхода заданного набора точек и, в операционном пространстве — для следования по заданной траектории;
- Реализована система технического зрения, способная обнаруживать объекта элементарной геометрической формы на плоских поверхностях, оценивать положение и ориентацию объекта относительно системы координат камеры и оценивать радиус кривизны траектории движения обнаруженного предмета;
- были реализованы два ros-пакета: для системы технического зрения и системы управления и планирования траекторий.

Дальнейшую разработку планируется посвятить моделированию полученной робототехнической системы в симуляторе V-Rep, а затем и на реальном роботе KUKA YouBot.

Хочется выразить слова благодарности моему коллеге Антонову Евгению Сергеевичу, а также всему коллективу студенческого конструкторского бюро кафедры Систем управления и информатики.

Инв. № подп.	Подп. и дата	Бзлм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

Список использованных источников

- 1 RoboCup@ Work: competing for the factory of the future / Gerhard K Kraetzschmar, Nico Hochgeschwender, Walter Nowak et al. // Robot Soccer World Cup / Springer. — 2014. — P. 171–182.
- 2 Rethink Robotics, Baxter [Электронный ресурс]. — URL: <http://www.rethinkrobotics.com/>. Дата обращения: 11.03.2018.
- 3 Baxter for education in europe [Электронный ресурс]. — URL: <https://www.rethinkrobotics.com/blog/baxter-for-education-in-europe-graz-university-of-technology-austria/>. Дата обращения: 11.03.2018.
- 4 ROS (Robot Operating System) [Электронный ресурс]. — URL: <http://www.ros.org/news/robots/mobile-manipulators/>. Дата обращения: 12.03.2018.
- 5 Moley Robotic Kitchen [Электронный ресурс]. — URL: <http://www.moley.com/>. Дата обращения: 11.03.2018.
- 6 YouBot Detailed Specifications [Электронный ресурс]. — URL: http://www.youbot-store.com/wiki/index.php/YouBot_Detailed_Specifications. Дата обращения: 10.03.2018.
- 7 Spong M.W., Hutchinson S., Vidyasagar M. Robot Modeling and Control. — Wiley, 2005. — P. 495.
- 8 Craig J.J. Introduction to Robotics: Mechanics and Control. — Pearson, 2018. — P. 303.
- 9 Яблонский А.А., Никифорова В.М. Курс теоретической механики. Статика. Кинематика. Динамика: [учебник для вузов по техническим специальностям]. — КноРус, 2011. — P. 603.
- 10 Datta B.N. Numerical Linear Algebra and Applications, Second Edition. Other Titles in Applied Mathematics. — Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2010. — P. 530.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

КСУИ.106.4235.001 ПЗ

Лист

63

- 11 Moreno Carlos, Li Ming. A comparative study of filtering methods for point clouds in real-time video streaming // Proceedings of the World Congress on Engineering and Computer Science. — 2016. — P. 5.
- 12 Ferran Roure Garcia, Tools for 3D Point Cloud Registration [Электронный ресурс]. — URL: <http://eia.udg.es/qsalvi/FerranRoureThesis.pdf>. Дата обращения: 16.04.2018.
- 13 Robotics_Toolbox for Scilab [Электронный ресурс]. — URL: https://atoms.scilab.org/toolboxes/Robotics_Toolbox. Дата обращения: 12.05.2018.
- 14 Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation / Aitor Aldoma, Zoltan-Csaba Marton, Federico Tombari et al. // IEEE Robotics & Automation Magazine. — 2012. — P. 80–91.
- 15 Voxel cloud connectivity segmentation-supervoxels for point clouds / Jeremie Papon, Alexey Abramov, Markus Schoeler, Florentin Wörgötter // Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on / IEEE. — 2013. — P. 2027–2034.

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

Приложение А
(обязательное)

Программный код в Scilab

В этом приложении представлен программный код в Scilab, используемый в этой работе.

Программный код функции для расчета коэффициентов кубических сплайнов с фиксированными концами для интерполяции траектории, заданной набором точек q_i в моменты времени t_i :

```
function [a,b,c,d]=splin_v(t, q, dq0, dqf)
    // Computes a cubic spline interpolation function
    // in Hermit form.
    //
    // Params:
    //         t      vector of the time for knots (n >= 3)
    //         q      vector of knots
    //         dq0   initial velocity
    //         dqf   final velocity
    //
    // Returns:
    //         a,b,c,d coeffitions for cubic spline
n = length(q); // quantity knots
a = [] ; b = [] ; c = [] ; d = [] ;
delta = [] ;
z = [] ; mu = [] ; l = [] ;
h = [] ;
a = q;
for i = 1:n-1 do
    h(i) = t(i+1) - t(i);
end
delta(1) = 3 *(a(2) - a(1)) / h(1) - 3 * dq0;
delta(n) = 3 * dqf - 3 * (a(n) - a(n-1)) / h(n-1);
for i = 2:n-1 do
    delta(i) = 3 * (a(i+1) - a(i)) / h(i) - 3 * (a(i)
        - a(i-1)) / h(i-1);
end
```

Изв.	№ подп.	Подп. и дата	Бзм. и нв. №	Инв. №	Подп. и дата

```

// forward
mu(1) = 0.5;
z(1) = delta(1) / (2*h(1));
for i = 2:n-1 do
    l = 2 * (t(i+1) - t(i-1)) - h(i-1) * mu(i-1);
    mu(i) = h(i) / l;
    z(i) = (delta(i) - h(i-1) * z(i-1)) / l;
end
// inverse
l = h(n-1) * (2 - mu(n-1));
z(n) = (delta(n) - h(n-1) * z(n-1)) / l;
c(n) = z(n);
for i = n-1:-1:1 do
    c(i) = z(i) - mu(i) * c(i+1);
    b(i) = (a(i+1) - a(i)) / h(i) - h(i) * (c(i+1)
                                                + 2 * c(i)) / 3;
    d(i) = (c(i+1) - c(i)) / (3 * h(i));
end
endfunction

```

Программный код функции для расчета коэффициентов кубических сплайнов со свободными концами:

```

function [a,b,c,d]=splin_a(t, q)
    n = length(q);
    a = []; b = []; d = [];
    K = zeros(1,n+1); K(2) = 0;
    c = zeros(1,n+1); c(2) = 0;
    for i = 3:n do
        j = i - 1;
        m = j - 1;
        A = t(i) - t(j);
        B = t(j) - t(m);
        R = 2 * (A + B) - B * c(j);
        c(i) = A / R;
        K(i) = (3 * ((q(i)-q(j))/A - (q(j)-q(m))/B)
                  - B*K(j)) / R;
    end
    c(n+1) = K(n+1);
    for i = n:-1:3 do

```

Изв.	№ подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

```

        c(i) = K(i) - c(i) * c(i+1);
    end
    for i = 2:n do
        hi = t(i) - t(i-1)
        a(i-1) = q(i-1);
        b(i-1) = (q(i) - q(i-1)) / hi - (c(i+1)
                                         + 2 * c(i))*hi/3;
        d(i-1) = (c(i+1) - c(i)) / 3 / hi;
    end
    c(1:n) = c(2:n+1);
endfunction

```

Функция для расчета коэффициентов сплайна пятой степени:

```

function [qt,qdt,qddt]=polynomial_trajectory_5(qs,
                                                qf, tv, dqs, dqf, v_max)
// Compute trajectory for 5-th polynom
// q(t) = a + bt + ct^2 + dt^3 + et^4 + ft^5
n = length(qs);
if length(tv)>1 then
    tscal = max(tv);
    t = tv(:)/tscal;
else
    tscal = 1;
    // normalized time from 0 -> 1
    t = (0:(tv-1))'/(tv-1);
end;
qt = [] ; qdt = [] ; qddt = [] ;
a = qs;
b = dqs;
c = zeros(n,1);
d = 10 *(qf - qs) - (6*dqs + 4*dqf);
e = -15 *(qf - qs) + (8*dqs + 7*dqf);
f = 6 *(qf - qs) - (3*dqf + 3*dqs);
times = [ones(length(t),1) t t.^2 t.^3 t.^4 t.^5];
coefs = [a b c d e f]';
qt = times * coefs;
coefs = [ b zeros(n,1) 3*d 4*e 5*f zeros(n,1)]';
//      qdt = times * coefs;
qdt = times * coefs / max(times * coefs)

```

Изв.	№ подп.	Подп. и дата	Бзм. Изв. №	Изв. № дубл.	Подп. и дата

```

    qdt = qdt * v_max
    coefs = [ zeros(n,1) 6*d 12*e 20*f zeros(n,1) zeros(n,1) ];
    qddt = times * coefs;
endfunction

```

Функция рассчитывающая траекторию составленную из трех частей: параболы, прямой, параболы:

```

function [qt,qdt,qddt, t, tb, tf]=plp(qs,
                                         qf, step, v_des)
    // Compute trajectory with velocity
    // profile -- parabolic-linear-parabolic
    v = v_des;
    tf = 1.5*(qf - qs) / v;
    tf = 2
    v = 1.5*(qf - qs) / tf;
    t = 0:step:tf;
    qt = zeros(length(t), 1); qdt = qt; qddt = qt;
    tb = (qs - qf + v*tf) / v;
    a = v/tb;
    for i = 1:length(t)
        tt = t(i);
        if tt <= tb then
            // initial blend
            qt(i) = qs + a/2*tt^2;
            qdt(i) = a*tt;
            qddt(i) = a;
        elseif tt <= (tf-tb)
            // linear motion
            qt(i) = (qf + qs - v*tf) / 2 + v * tt;
            qdt(i) = v;
            qddt(i) = 0;
        else
            // final blend
            qt(i) = qf - a/2*tf^2 + a*tf*tt - a/2*tt^2;
            qdt(i) = a*tf - a*tt;
            qddt(i) = -a;
        end
    end
endfunction

```

Извм.	Лист	№ докум.	Подп.	Дата

Функция возвращающая точку дуги в операционном пространстве в заданный момент времени:

```
function [x,y,alpha]=get_circle_point(t, v, theta_s, alpha0)
    phi = v * t / (%pi * Rc);
    x = Rc * cos(s + phi);
    y = Rc * sin(s + phi);
    alpha = alpha0 + phi;
endfunction
```

Инв. № подп.	Подп. и дата	Бзм. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата