

## РЕФЕРАТ

Дипломная работа содержит 61 страниц, 28 рисунков, 1 таблиц, 13 источников.

МОБИЛЬНЫЙ РОБОТ, КОЛЕСО ИЛОНА, ОМНИ КОЛЕСО,  
ВИРТУАЛЬНОЕ ТЕСТИРОВАНИЕ РОБОТОВ, ROS, GAZEBO

Объектом исследования являются мобильные роботы, использующие роликонесущие колеса и методы управления ими.

Цель курсовой работы — разработка конфигурации и алгоритма движения голономной роботизированной тележки.

В результате работы были разработаны кинематическая и динамическая модели мобильных роботов, использующих роликонесущие колеса для движения. Разработана компьютерная модель трехколесного мобильного робота, на основе которой протестирована кинематическая модель поведения голономного робота.

## СОДЕРЖАНИЕ

Введение . . . . .	4
1. Классификация мобильных роботов . . . . .	5
1.1. Роботы, использующие ноги для движения . . . . .	6
1.2. Роботы, использующие колеса для движения . . . . .	8
1.2.1. Обыкновенное колесо . . . . .	9
1.2.2. Колесо castor wheel . . . . .	11
1.2.3. Роликонесущее колесо . . . . .	12
1.3. Роботы, использующие гусеницы для движения . . . . .	16
2. Кинематическая модель робота с N всенаправленными колесами .	18
3. Динамическая модель робота с N всенаправленными колесами . .	23
4. Кинематическая модель роликонесущего колеса . . . . .	28
5. Кинематическая модель тележки на четырех роликонесущих колесах . . . . .	32
6. Технологии моделирования робототехнических систем . . . . .	36
6.1. Фреймворк для программирования робототехнических систем ROS . . . . .	36
6.1.1. Уровень файловой системы . . . . .	37
6.1.2. Уровень графа вычислений системы . . . . .	39
6.1.3. Формат описания робототехнических систем URDF . .	41
6.2. Среда моделирование робототехнических систем Gazebo . . .	43
6.3. Системы автоматического управления. ПИД регуляторы . . .	45
7. Программная реализация . . . . .	48
7.1. SDF модель . . . . .	48
7.2. Плагин модели Gazebo . . . . .	49
7.3. Алгоритм движения тележки по аналитически заданному пути	51

7.4. Клиент управления . . . . .	55
7.5. Тестирование алгоритма движения по аналитически заданному пути . . . . .	57
Заключение . . . . .	59
Список использованных источников . . . . .	60

## ВВЕДЕНИЕ

В рамках данной работы рассматривается задача — исследование видов движущихся роботизированных систем и методов управления ими. Особенный интерес изучения представляют роботизированные системы, использующие для передвижения роликонесущие колеса. Этот тип колес широко используется при создании роботизированных систем и позволяет роботам двигаться в любом заданном направлении в плоскости движения. Это увеличивает область применения роботизированных систем: в малых помещениях, в которых роботизированным системам на обычных колесах не хватает места для передвижения, роботы на роликонесущих колесах способны передвигаться без каких-либо ограничений.

В рамках работы ставится целью разработка модели роликонесущего колеса, механической и кинематической модели тележек, опирающихся на  $N$  роликонесущих колес,  $N > 2$ , изучение и обзрение методов управления роботизированными системами. Полученные сведения следует протестировать на виртуальной модели робота с учетом всех физических сил, воздействующих на систему.

## **1 Классификация мобильных роботов**

Мобильным роботом называется робот, способный менять свое местоположение в пространстве. Мобильные роботы могут быть автономными или управляемыми вручную. Автономные мобильные роботы способны без участия человека, основываясь на показаниях установленных на нем сенсоров и датчиков, определять свое местоположение и окружение, в котором они находятся [1]. Управляемый вручную робот не имеет такую возможность и способен передвигаться только по заранее заданной траектории.

Мобильные роботизированные системы подразделяются на голономные и неголономные. Мобильная роботизированная система называется голономной, если количество степеней свободы, доступных для управления, равно общему количеству степеней свободы системы. Иначе система называется неголономной [5]. Характеристика голономности робота напрямую зависит от конфигурации механизма, приводящего его в движение.

Для того, чтобы передвигаться в пространстве, мобильный робот должен иметь в своем устройстве механизм, приводящий его в движение. Мобильные роботы способны передвигаться используя следующие техники:

- ходьба;
- прыжки;
- скольжение;
- качение;
- плавание;
- полет;

- кувырки.

Естественно, техники могут комбинироваться. В рамках работы исследуются механизмы, позволяющие роботу двигаться по твердым горизонтальным поверхностям в земной среде. В рамках работы рассматриваются возможные техники движения ходьбой и качением. Существующие роботы, способные двигаться по горизонтальной плоскости, делятся на следующие категории:

- роботы, использующие ноги для движения;
- роботы, использующие колеса для движения;
- роботы, использующие гусеницы для движения.

### **1.1 Роботы, использующие ноги для движения**

Способ движения существующих роботов, использующих ноги, во многом повторяет способы передвижения биологических существ. Роботы этого типа имеют больше степеней свободы в сравнении с колесными роботами, что делает их устройство гораздо сложнее.

По своей конфигурации роботы, использующие ноги, отличаются количеством используемых ног. Существуют одноногие роботы, распространенными являются шестиногие роботы. Вообще, количество используемых для построения робота ног не ограничено.

Для того, чтобы робот был способен стоять на поверхности, при этом не балансируя посредством приложения сил, ему необходимо как минимум три ноги. Кроме того, центр тяжести такого робота должен находиться внутри области, образуемой многоугольником, каждая из вершин которой совпадает с координатой ноги. Но для движения такого типа роботов, он должен быть способен передвигать конечностями.

Имея в арсенале три ноги, роботу невозможно добиться статического

равновесия во время ходьбы. Минимальное количество ног, необходимое для движения и постоянного поддержания статического равновесия - шесть. В такой конфигурации существует способ движения конечностями, при котором робот всегда имеет три точки опоры.

На практике, количество степеней свободы каждой ноги зависит от их общего количества. Так, каждой ноге шестиногого робота достаточно двух степеней свободы для движения. В то же время, двуногие роботы используют гораздо большее степеней свободы. Так, например, двуногий робот Atlas имеет 27 степеней свободы для перемещения. На рисунке 1.1 изображены существующие решения роботов такого типа.

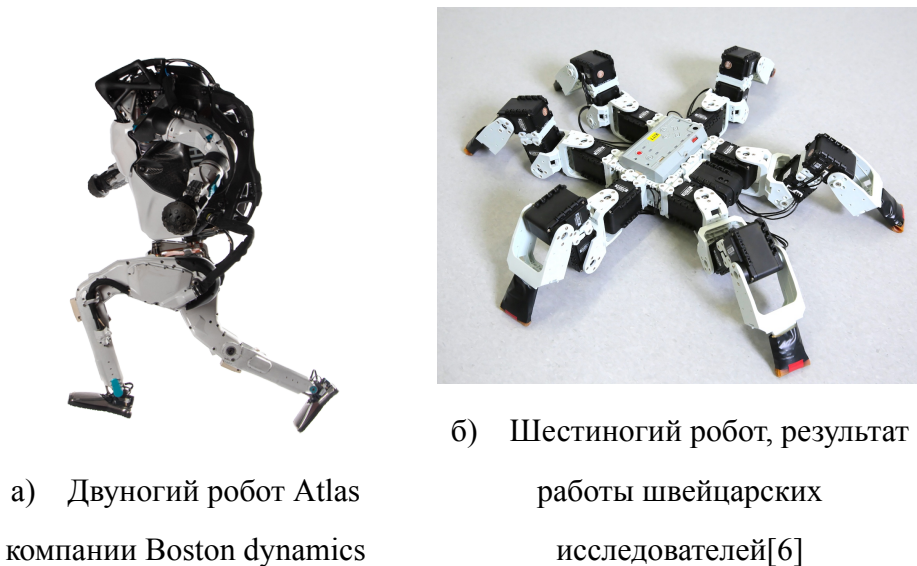


Рисунок 1.1 – Современные роботы, использующие ноги для передвижения

Основная характеристика роботов, использующих ноги, — последовательность пятен контакта ног робота с поверхностью. Так как для движения такому роботу нужно всего несколько малых областей контакта, качество поверхности не столь важно для его передвижения.

Роботы, использующие ноги, используются в условиях, когда поверхность движения не является плоской или материал поверхности

мягкий. Во время качения по плоской твердой поверхности колесо имеет малую площадь соприкосновения с поверхностью, поэтому при качении колесо испытывает малое количество сопротивления. Неровности и мягкий материал поверхности увеличивает площадь поверхности колеса и уменьшает его эффективность. Для создания условий движения колеса требуется большое количество ограничений. Роботы, использующие ноги для движения, в отличие от колесных, имеют большую площадь соприкосновения с поверхностью, что дает им преимущество в сложных условиях.

Роботы, использующие ноги для движения, способны передвигаться в сложных условиях, когда поверхность не является ровной, имеет подъемы или спуски или состоит из мягкого материала. Это позволяет широко использовать таких роботов в сложной среде. Кроме того, такие роботы способны преодолевать препятствия просто перешагивая их. К недостаткам роботов с ногами можно отнести высокую сложность механизма и сравнительно низкую скорость передвижения.

## **1.2 Роботы, использующие колеса для движения**

Использование колес для передвижения - самый распространенный подход к построению мобильных роботов. Во время проектирования роботов, использующих ноги, много внимания уделяется проблеме их устойчивости, в то время как разработка колесных роботов практически лишена этой части проектирования. Несмотря на то, что для устойчивого положения робота в пространстве необходимо всего два колеса, большинство колесных мобильных роботов используют три и более, что решает проблему устойчивости.

Основная задача, стоящая перед проектировщиками мобильных



колесных роботов - маневренность и управляемость модели.

Как было описано выше, всякий мобильный робот может быть или голономным, или неголономным. Для случая колесных роботов это значит, что в любой момент времени робот может передвинуться в любом заданном направлении в плоскости. Так например стандартная конфигурация автомобиля не является голономной, в то время как тележка на роликонесущих колесах, описанная ниже, является.

На текущий момент существует множество типов конфигураций колес:

- обычное колесо;
- роликовое колесо;
- роликонесущее колесо.

### **1.2.1 Обыкновенное колесо**

Обыкновенное колесо по своей сути — диск или обод, вращающийся на оси или укрепленный на валу и служащий для приведения механизма в движение [3].

Обычное колесо - самый распространенный тип колес, прародитель всех остальных типов, древнейшее изобретение человечества. Такие колеса повсеместно используются в автомобилях, поездах, самолетах и так далее.

Для того, чтобы вычислить скорость центра обыкновенного колеса в условиях механики сплошных сред, необходимо знать его угловую скорость  $\omega$ , радиус вектор некоторой точки границы диска  $r$  и скорость точки касания с поверхностью  $V_0$ .

Если считать, что скорость поверхности движения равна нулю, то в случае, если вектор скорости  $V_0$  отличен от нуля, колесо скользит по

поверхности.

Тогда скорость центра колеса  $V_c$  будет равна

$$V_c = V_0 + \omega \times r \quad (1.1)$$

Обычное колесо широко используется и в робототехнике. На рисунке 1.2 изображен робот-шпион, использующий обыкновенные колеса.

По причине того, что колесо способно катить колесную тележку только в направлении, перпендикулярном оси его вращения, построение голономного робота, использующего обыкновенные колеса, затруднительно. Для того, чтобы колесный робот на обычных колесах был способен менять направление движения, необходимо менять угол оси его вращения. Так, например, в стандартной конфигурации автомобиля передние оси вращения колес механически соединены с рулевой рейкой, что позволяет маневрировать машиной во время движения.



Рисунок 1.2 – Робот-шпион, использующий обычные колеса

Главное достоинство обычного колеса - простота конструкции и минимальное трение качения в сравнении другими типами колес [5]. К недостаткам относится тот факт, что обычное колесо имеет всего одну степень свободы.

### 1.2.2 Колесо castor wheel

Castor wheel - распространенный тип колес, повсеместно используемый в каталках, продуктовых тележках, мебели и так далее. У этого типа колес нет определенного русского наименования; будем называть такие колеса роликовыми. Примеры роликовых колес изображены на рисунке 1.3.

Главное отличие роликового колеса от обычного - дополнительная степень свободы. Ось вращения такого колеса может вращаться на 360 градусов. Также, к таким колесам относят закрепленные шарики, способные катиться в любом направлении.



а) Роликовое колесо тележки



б) Шаровое колесо

Рисунок 1.3 – Роликовые колеса

Благодаря двум степеням свободы роликовые колеса способны катиться в любом направлении. Однако из-за своей конструкции, привести само роликовое колесо в движение затруднительно. Роликовые колеса обычно используются для того, чтобы облегчить движение тяжелых

объектов по поверхности, причем объекты приводятся в движения за счет внешних сил. В случае продуктовой тележки, силу для её движения прикладывает покупатель.

В робототехнике роликовые колеса широко используются для увеличения маневренности. Распространенная конфигурация робота, использующего два обычных колеса и одно роликовое, изображена на рисунке 1.4.

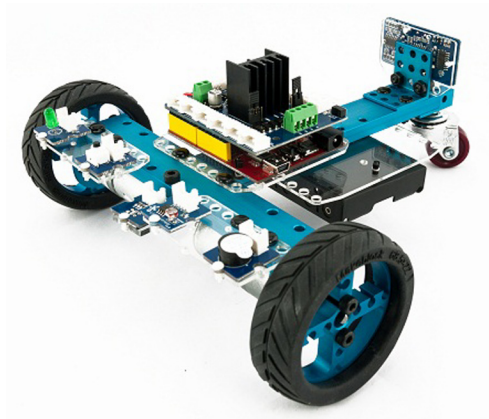


Рисунок 1.4 – Робот, использующий в своей конфигурации два обычных колеса и одно роликовое

Такая конфигурация позволяет значительно увеличить маневренность робота. Приводя в движение одно из двух колес, тележка вращается на месте, описывая круги вокруг точки касания статичного колеса. Однако такая модель все-еще не является голономной. Движение строго перпендикулярно оси обычных колес все-еще невозможно.

### 1.2.3 Роликонесущее колесо

Роликонесущее колесо - колесо, имеющее на своем ободе ролики, каждый из которых вращается вокруг собственной оси. Для примера распространенные виды роликонесущих колес изображены на рисунке 1.5.



а) колесо Илона



б) омни колесо

Рисунок 1.5 – Роликонесущие колеса

Главная характеристика роликонесущего колеса - угол между осью вращения колеса и осью ролика, касающегося поверхности вращения,  $\alpha$ . Благодаря вращению роликов, скорость точки касания с поверхностью  $V_0$  из выражения 1.1 не равна нулю. Роликонесущее колесо, во время движения, как-бы скользит на установленном на нем ролике в направлении, перпендикулярном оси этого ролика. Это позволяет роботам, использующим такие колеса для передвижения, двигаться в любом заданном направлении. Подробное математическое исследование этого явления описывается в главе 4.

Роликонесущее колесо относительно молодое изобретение. Первое упоминание о нем относится к 1919 году[8], однако в том виде, в котором оно используется сегодня, такое колесо, названное омни колесом, появилось только в 1974 году[9]. Особенность омни колеса заключается в том, что угол  $\alpha = 90^\circ$ , благодаря чему оно может свободно скользить в любом направлении.

8 апреля 1975 года шведский изобретатель Бенгт Ирланд Илон запатентовал свое изобретение - колесо Илона или, по названию компании,

в которой он работал, Mecanum wheel[10]. Инновация заключается в том, что угол  $\alpha$  лежит в пределах  $30^\circ - 60^\circ$  градусов, что позволяет устанавливать такие колеса параллельно друг-другу, и при этом не теряя свойство голономности. На рисунке 1.7 изображены роботы, использующие для движения роликонесущие колеса типа омни колесо и колесо Илона.

На рисунке 1.6 изображена возможная конфигурация робота на колесах Илона. Так как угол  $\alpha$  не прямой, то устанавливая колеса так, чтобы прямые, проходящие через оси касающихся поверхности роликов, попарно пересекались ( $l_2, l_3$  и  $l_1, l_4$ ), приходим к выводу: вектора скорости точек касания колес с поверхностью не равны нулю и не параллельны. Следовательно, изменяя длину векторов скоростей, возможно задать любое направление движения тележки. В случае омни колеса, колеса не могут находиться параллельно друг-другу: для голономности необходимо, чтобы прямые, проходящие через оси роликов, пересекались.

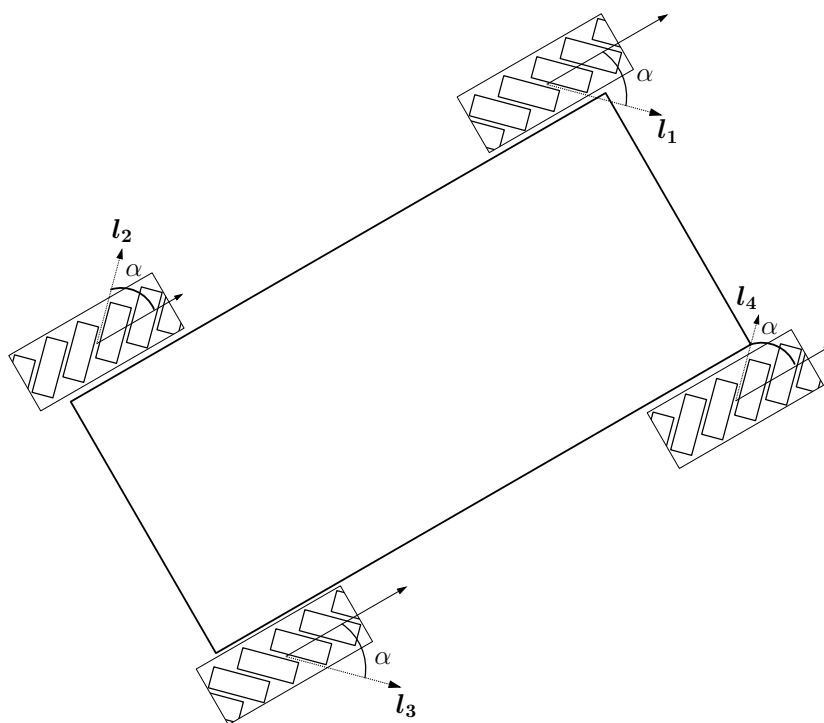
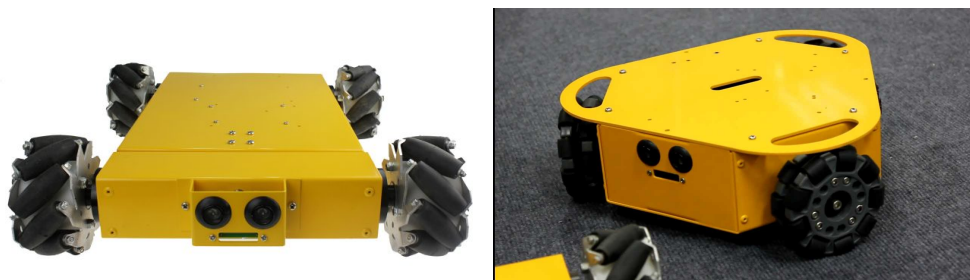


Рисунок 1.6 – модель тележки на четырех колесах Илона



а) мобильный робот, движимый четырьмя колесами Илона      б) мобильный робот, движимый тремя омни колесами

Рисунок 1.7 – Голономные мобильные роботы компании Nexus robot

Выбор между углом наклона роликов  $\alpha$  зависит от поставленной перед мобильным роботом задачей, однако механизм колеса Илона сложнее механизма омни колеса; омни колесо проще в производстве и надежнее. Однако невозможность одновременного расположения омни колес параллельно друг-другу и сохранение голономности делает колесо Илона предпочтительней в случае, когда иные конфигурации невозможны.

Кроме того, скорость робота при движении параллельно дискам колес, благодаря параллельному расположению, выше: каждое колесо дает положительный вклад в вектор скорости. В то время тележка на омни колесах, ввиду невозможности такой конфигурации, не может двигаться в некотором направлении так, чтобы каждое колесо делало положительный вклад в общее движение.

Главный недостаток роликонесущих колес - большой вес и высокое сопротивление поверхности качения в сравнении с обычным колесом. Кроме того, устройство роликонесущих колес значительно сложнее устройства обычных, что негативно сказывается на их надежности.

### 1.3 Роботы, использующие гусеницы для движения

Гусеничный ход — движитель самоходных машин, обеспечивающий повышенную проходимость. Принцип работы гусеничного хода — непрерывное подкладывание гусениц под колёса машины, то есть создание для колёс бесконечного пути, на котором сопротивление движению значительно ниже, чем на мягком грунте[4]. Гусеницей, в свою очередь, называется замкнутая сплошная лента или цепь из шарнирно-соединённых звеньев, применяемая в гусеничном ходу. На внутренней поверхности гусеницы имеются впадины или выступы, с которыми взаимодействуют ведущие колёса машины. Внешняя поверхность гусениц снабжена выступами, которые обеспечивают сцепление с грунтом. гусеницы могут быть металлическими, резино-металлическими и резиновыми [4].

Этот тип механизма передвижения распространен среди тяжелой техники и вездеходов. За счет большой площади пятна касания с поверхностью, давление на поверхность движения гораздо меньше, чем в случае других типов колес, благодаря чему транспортные средства не вязнут в рыхлой почве, песке, болотах и так далее. Гусеницы также распространены и среди роботов: такой механизм передвижения позволяет им преодолевать ступени и различные препятствия.

На практике самая распространенная конфигурация гусеничной машины — два гусеничных хода, расположенных параллельно друг-другу. Пример такой конфигурации изображен на рисунке 1.8. Степень свободы гусеничного хода равна единице, однако самоходные гусеничные машины достаточно манёвренные и способны делать разворот на месте, направляя пару гусеничных ходов в противоположных направлениях с равной скоростью.





Рисунок 1.8 – Гусеничный робот Nasa grover, разработанный для использования в условиях ледяной пустыни

Главный недостаток гусеничных роботов — большая вариация возможных позиций робота после выполнения маневров. Маневрируя, гусеничный робот устанавливает скорость одного из гусеничных ходов отличной от другой, заставляя медленную гусеницу скользить по поверхности. В зависимости от типа поверхности и её состояния, положение робота после маневра может сильно варьироваться [1], поэтому точное определение положения гусеничного робота затруднительно. Этот факт затрудняет построение автономного гусеничного робота. Кроме того, большая площадь пятна касания дает большее сопротивление качению, что замедляет робота. Большое количество элементов, содержащееся в гусеничном ходу, подвержено износу и имеет меньший запас прочности, чем обычное колесо.

## 2 Кинематическая модель робота с $N$ всенаправленными колесами

Описание движения робота требует моделирования его поведения. Простейшая модель движения робота в пространстве - кинематическая. Эта модель описывает движения исключительно через зависимость координат от времени. То есть в кинематической модели рассматривается движение тела, но не рассматриваются причины, его создающие.

Рассматривается движение тележки с  $N$  всенаправленными колесами ( $N > 3$ ) по гладкой двумерной поверхности без учета действующих сил, причем плоскости колес тележки вертикальны и неподвижны относительно платформы тележки. В рамках модели всенаправленные колеса способны скользить в любом направлении с пренебрежимо малой силой трения. В рамках модели задается глобальная система координат, связанная с поверхностью  $\{o, x, y, z\}$  и локальная, инерциальная относительно глобальной, жестко связанная с тележкой  $\{c, x_l, y_l, z_l\}$ , причем плоскость  $x_l y_l$  параллельна плоскости  $xy$ . Не теряя общности начало локальных координат полагается в точке центра масс тележки. Положение тележки определяется вектором координат  $(x, y, \varphi)$  где  $x, y$  - координаты на плоскости, и  $\varphi$  - угол между осью  $ox$  и  $cx_l$ . Скорость тележки определяется вектором  $(\dot{x}, \dot{y}, \omega)$  где  $\omega = \dot{\varphi}$  - угловая скорость тележки.

$\{c_i, x_{w,i}, y_{w,i}, z_{w,i}\}$  - обозначение локальной системы координат  $i$ -го колеса, изображенное на рисунке 2.1, где  $c_i$  - ось вращения,  $x_{w,i}$  - ось, направленная из  $c_i$  в сторону точки касания с поверхностью,  $y_{w,i}$  - ось, параллельная поверхности качения, направленная вправо,  $z_{w,i} = x_{w,i} \times y_{w,i}$ .

Для примера на рисунке 2.2 схематически изображается кинематическая модель робота с тремя всенаправленными колесами.

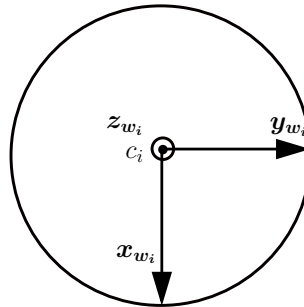


Рисунок 2.1 – координатные оси  $i$  - го колеса.

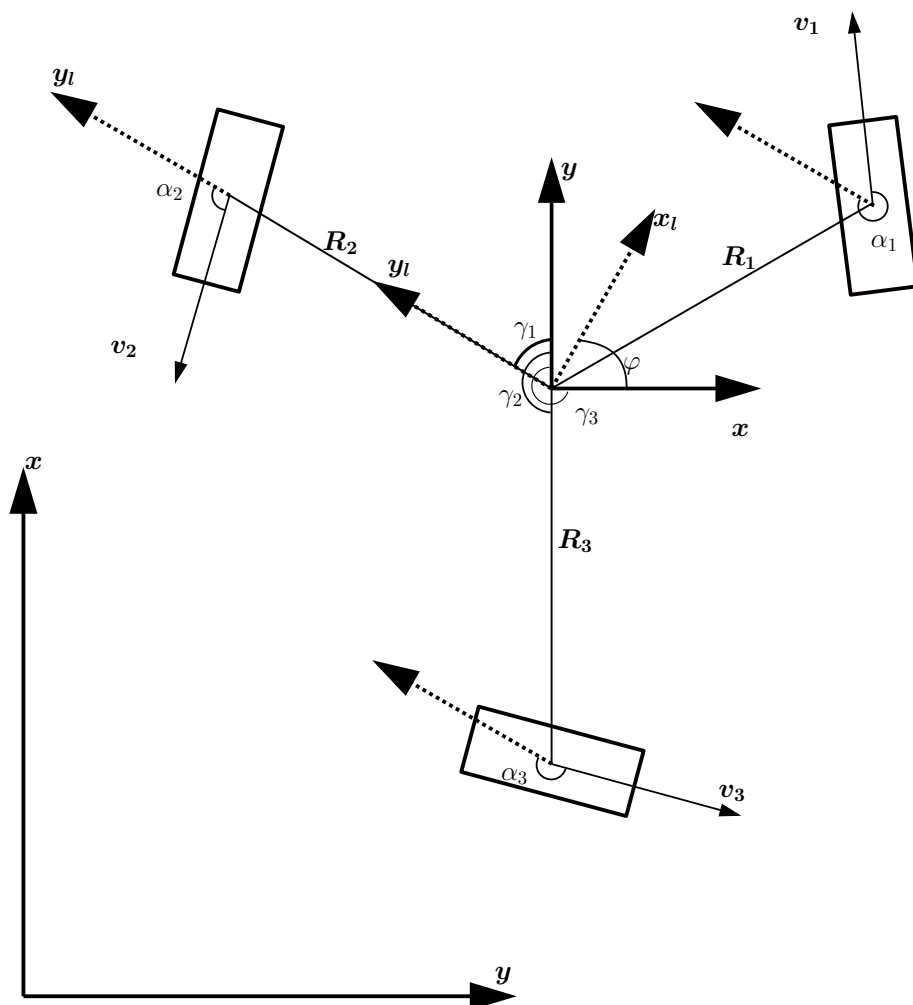


Рисунок 2.2 – кинематическая модель тележки с тремя всенаправленными колесами.

Вектор скорости  $i$ -го колеса обозначается как  $v_i$ ,  $i = \overline{1, N}$ . Он направлен по касательной к диску  $i$ -го колеса  $y_{w,i}$ . Его разложение на поступательную и вращательную составляющие имеет вид:

$$v_i = v_{i,tr} + v_{i,rot} \quad (2.1)$$

Обозначения:  $\alpha_i$  - угол между  $y_{w,i}$  и осью  $y_l$ . Векторы  $v_i$  и  $y_{w,i}$  сонаправлены, и поэтому  $v_i$  составляет с вектором  $y_l$  угол  $\alpha_i$ , что иллюстрируется на рисунке 2.2.

Рассматривается вектор поступательной скорости тележки  $v_{tr} = [\dot{x}, \dot{y}, 0]$ . Для того, чтобы тележка имела скорость  $v_{tr}$ , величина вектора скорости  $i$ -го колеса должна быть равна проекции  $v_{tr}$  на направление  $y_{w,i}$ . Вектор поступательной скорости  $i$ -го колеса составляет с осью  $y$  угол  $\varphi + \alpha_i$ , что иллюстрируется на рисунке 2.3. Тогда  $v_{tr,i}$  представляется в виде

$$v_{tr,i} = [-\sin(\varphi + \alpha_i)\dot{x} + \cos(\varphi + \alpha_i)\dot{y}]y_{w,i} \quad (2.2)$$

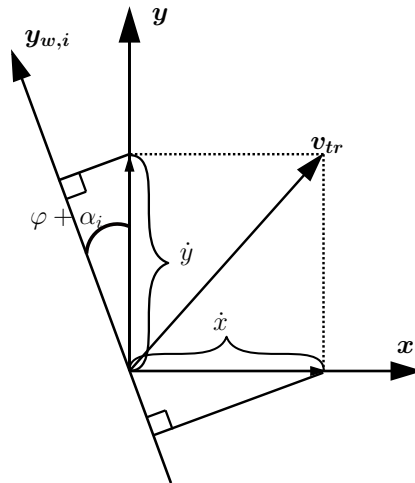


Рисунок 2.3 – проекция вектора поступательной скорости на направление движения  $i$ -го колеса

Вводятся обозначения:  $R_i$  - радиус вектор оси  $i$ -го колеса от оси вращения (точки центра масс) длины  $R_i$ . Для того, чтобы заставить

тележку вращаться вокруг оси вращения  $Cz_l$  с угловой скоростью  $\omega$ , необходимо задать вектор скоростей каждого колеса перпендикулярно  $R_i$ . При этом длина вектора должна составлять  $R_i\omega$  [7]. Таким образом скорости вращательного движения  $i$  - го задается следующим уравнением

$$\mathbf{v}_{rot,i} = \mathbf{R}_i \times \omega \mathbf{z} \quad (2.3)$$

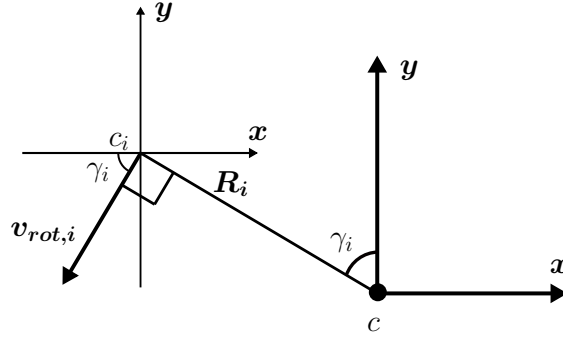


Рисунок 2.4 – проекция вектора вращательной скорости  $i$  - го колеса на глобальную систему координат

Вводятся обозначения:  $\gamma_i$  - угол между осью  $y$  и  $R_i$ . Тогда результатом проекции  $\mathbf{v}_{rot,i}$  на  $xy$ , что иллюстрируется на рисунке 2.4 является уравнение

$$\mathbf{v}_{rot,i} = R_i\omega[-\cos(\gamma_i)\mathbf{x} - \sin(\gamma_i)\mathbf{y}] \quad (2.4)$$

Результатом проекции  $\mathbf{v}_{rot,i}$  на направление  $\mathbf{y}_{w,i}$ , аналогично уравнению 2.2, является

$$\begin{aligned} \mathbf{v}_{rot,i} &= R_i\omega[\sin(\varphi + \alpha_i)\cos(\gamma_i) - \cos(\varphi + \alpha_i)\sin(\gamma_i)]\mathbf{y}_{w,i} \\ \mathbf{v}_{rot,i} &= R_i\omega\sin(\varphi + \alpha_i - \gamma_i)\mathbf{y}_{w,i} \end{aligned} \quad (2.5)$$

Угол  $\theta_i = \gamma_i - \varphi = const$  есть угол между  $R_i$  и  $\mathbf{y}_l$ , следовательно коэффициент вектора угловой скорости не зависит от  $\varphi$  в общем случае. Очевидно  $\alpha_i - \theta_i$  - угол между  $\mathbf{y}_{w,i}$  и  $R_i$ . Если касательная диска  $i$  - го колеса  $\mathbf{y}_{w,i}$  перпендикулярна  $R_i$ , то  $\alpha_i - \theta_i = \frac{\pi}{2}$ ,  $\mathbf{v}_{rot,i} = R_i\omega$ .

Таким образом вектор скорости  $i$  - го колеса записывается в виде:

$$\mathbf{v}_i = [-\sin(\varphi + \alpha_i)\dot{x} + \cos(\varphi + \alpha_i)\dot{y} + R_i\omega\sin(\alpha_i - \theta_i)]\mathbf{y}_{w,i} \quad (2.6)$$

Вводятся обозначения:  $\omega_i$  - искомая угловая скорость  $i$  - го колеса. Результатом соотношения вектора скорости  $i$  - го колеса с его угловой скоростью является

$$\mathbf{v}_i = \mathbf{r}_i \times \omega_i \mathbf{z}_{w,i} \quad (2.7)$$

Где  $\mathbf{r}_i$  - вектор, имеющий координаты  $[-r_i, 0, 0]$  в системе координат  $i$ -го колеса тележки,  $r_i$  - радиус  $i$ -го колеса тележки,  $\omega_i$  - угловая скорость этого колеса. Результатом вычисления векторного произведения в 2.7 является

$$\mathbf{v}_i = \mathbf{r}_i \times \omega_i \mathbf{z}_{w,i} = \omega_i r_i \mathbf{y}_{w,i} \quad (2.8)$$

Таким образом, результатом выражения  $w_i$  из 2.8 и подстановки  $\mathbf{v}_i$  из 2.6 является уравнение:

$$\omega_i = \frac{1}{r_i}(-\sin(\varphi + \alpha_i)\dot{x} + \cos(\varphi + \alpha_i)\dot{y} + R_i\omega\sin(\alpha_i - \theta_i)) \quad (2.9)$$

или в матричной форме

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \dots \\ \omega_N \end{bmatrix} = \begin{bmatrix} -\frac{1}{r_1}\sin(\varphi + \alpha_1) & \frac{1}{r_1}\cos(\varphi + \alpha_1) & \frac{1}{r_1}R_1\sin(\alpha_1 - \theta_1) \\ -\frac{1}{r_2}\sin(\varphi + \alpha_2) & \frac{1}{r_2}\cos(\varphi + \alpha_2) & \frac{1}{r_2}R_2\sin(\alpha_2 - \theta_2) \\ \dots & \dots & \dots \\ -\frac{1}{r_N}\sin(\varphi + \alpha_N) & \frac{1}{r_N}\cos(\varphi + \alpha_N) & \frac{1}{r_N}R_N\sin(\alpha_N - \theta_N) \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \omega \end{bmatrix} \quad (2.10)$$

Таким образом, координаты траектории пути  $(x, y, \varphi)$  однозначно определяют величину угловой скорости каждого колеса. Для нахождения угловой скорости каждого колеса необходимо вычислить  $(\dot{x}, \dot{y}, \omega)$  и подставить в формулу 2.10.

### 3 Динамическая модель робота с $N$ всенаправленными колесами

Динамическая модель, в отличие от кинематической, рассматривает движение твердых тел с учетом сил, приводящих это тело в движение. Тележка приводится в движения за счет крутящего момента  $T$ .

В рамках этой главы рассматривается движение тележки с  $N$  всенаправленными колесами ( $N > 3$ ) по негладкой двумерной поверхности с учетом действующих сил. В описании модели используются те же обозначения, что и в кинематической модели за исключением того, что вместо векторов скорости рассматриваются векторы сил  $f_i$ . Сила  $f_i$  приложена к верхней точке  $i$  - го колеса в направлении качения. Для примера на рисунке 3.1 схематически изображена динамическая модель робота с тремя всенаправленными колесами. Равнодействующая всех сил, действующих на тележку  $F$  имеет вид:

$$F = F_x x + F_y y + M_t z \quad (3.1)$$

Сила как мера воздействия на тело характеризует поступательное движение модели, в то время как момент сил  $M_t z$  характеризует вращательное движение. Равнодействующую всех сил можно разложить на поступательную и вращательную составляющие, аналогично кинематической модели, следующим образом:

$$F = \begin{bmatrix} F_X \\ F_Y \\ M_t \end{bmatrix} = \begin{bmatrix} M & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & J_r \end{bmatrix} \cdot \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \dot{\omega} \end{bmatrix} = B \cdot \ddot{u} \quad (3.2)$$

Где  $M$  - масса тележки,  $J_r$  - её момент инерции.

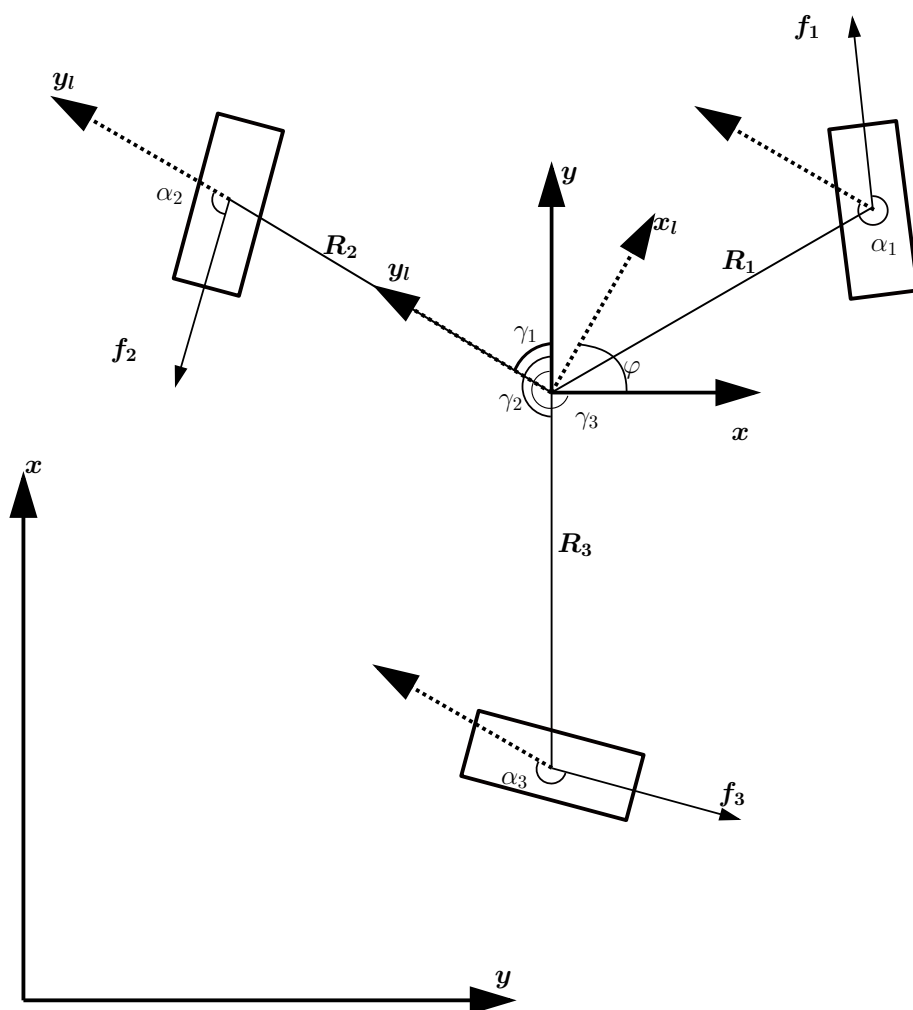


Рисунок 3.1 – динамическая модель тележки с тремя всенаправленными колесами

Момент инерции тележки можно считать приближенно равным моменту инерции однородного цилиндра радиуса  $R = \max_i(R_i)$

$$J_r = \frac{1}{2}MR^2 \quad (3.3)$$

Кроме того, вектор  $\mathbf{F}$  можно представить в виде:

$$\mathbf{F} = \sum_{i=1}^N \mathbf{f}_i \quad (3.4)$$

где  $\mathbf{f}_i$  - вектор силы, приложенной верхней точке  $i$  - го колеса. Имеет направление  $\mathbf{y}_{w,i}$ , величину  $f_i$  и составляет с осью  $\mathbf{y}$  угол  $\varphi + \alpha_i$ . Тогда,



результатом разложения  $F$  по компонентам и проекции вектора  $f_i$  на координатные оси является

$$F_X = -x \sum_{i=1}^N f_i \sin(\varphi + \alpha_i) \quad (3.5)$$

$$F_Y = y \sum_{i=1}^N f_i \cos(\varphi + \alpha_i) \quad (3.6)$$

$$M_t = \sum_{i=1}^N f_i \times R_i = z \sum_{i=1}^N f_i R_i \sin(\alpha_i - \theta_i) \quad (3.7)$$

Величины компонент вектора сил связываются следующим образом:

$$\begin{bmatrix} F_X \\ F_Y \\ M_t \end{bmatrix} = \begin{bmatrix} -\sin(\varphi + \alpha_1) & -\sin(\varphi + \alpha_2) & \dots & -\sin(\varphi + \alpha_N) \\ \cos(\varphi + \alpha_1) & \cos(\varphi + \alpha_2) & \dots & \cos(\varphi + \alpha_N) \\ R_1 \sin(\alpha_1 - \theta_1) & R_2 \sin(\alpha_2 - \theta_2) & \dots & R_N \sin(\alpha_N - \theta_N) \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \dots \\ f_n \end{bmatrix} = A f \quad (3.8)$$

Результатом является уравнение

$$F = A f \quad (3.9)$$

Результатом разрешения уравнение 3.9 относительно  $f$  является

$$f = A^{-1} \cdot F = A^{-1} \cdot B \cdot \ddot{u} \quad (3.10)$$

Зная, что вращающий момент колеса равен силе, приложенной в его верхней точке на его радиус и обозначая

$$r = \begin{bmatrix} r_1 \\ r_2 \\ \dots \\ r_N \end{bmatrix} \quad (3.11)$$

возможно описать зависимость координат пути  $(x, y, \varphi)$  от крутящих моментов каждого колеса:

$$\begin{bmatrix} T_1 \\ T_2 \\ \dots \\ T_N \end{bmatrix} = \mathbf{r} \mathbf{f}^T = \bar{\mathbf{r}} (\mathbf{A}^{-1} \cdot \mathbf{B} \cdot \ddot{\mathbf{u}})^T \quad (3.12)$$

В свою очередь величину момента сил каждого колеса можно выразить через его угловое ускорение:

$$T_i = J_{w,i} \dot{\omega}_i \quad (3.13)$$

Где  $J_{w,i}$  - момент инерции  $i$ -го колеса,  $\dot{\omega}_i$  - угловое ускорение  $i$  - го колеса.

Учтем силу трения качения модели по некоторой поверхности, предполагая равномерное распределение веса на каждое колесо. Известно, что сила трения качения противоположна по направлению силе, приводящей колесо в действие и для  $i$ -го колеса может быть вычислена по формуле:

$$f_{t_i} = \frac{Mg\eta}{Nr_i} \quad (3.14)$$

Где  $Mg$  - сила реакции опоры, равная силе тяжести робота,  $\eta$  - коэффициент трения качения, зависящий от характеристик поверхности.  $\eta = 0$  в случае, когда трения между колесом и поверхностью не происходит и  $\eta = \infty$  когда трение поверхности непреодолимо сильно. Сила трения качения направлена в противоположную сторону от направления движения колеса. Поэтому для того, чтобы её компенсировать, необходимо увеличить силу, приложенную в верхней точке колеса на величину силы

трения.

$$\tilde{f}_i = f_i + f_{t_i} \quad (3.15)$$

Вводятся обозначения

$$\tilde{\mathbf{f}} = \mathbf{f} + \begin{bmatrix} f_{t_1} \\ f_{t_2} \\ \dots \\ f_{t_N} \end{bmatrix} = \mathbf{f} + \frac{Mg\eta}{N} \cdot \begin{bmatrix} \frac{1}{r_1} \\ \frac{1}{r_2} \\ \dots \\ \frac{1}{r_N} \end{bmatrix} \quad (3.16)$$

Тогда заменяя в формуле 3.12  $f$  на  $\tilde{f}$ , определяется зависимость момента сил каждого колеса от координат пути, учитывающая силу трения:

$$\begin{bmatrix} T_1 \\ T_2 \\ \dots \\ T_N \end{bmatrix} = \mathbf{r} \left( A^{-1} \cdot B \cdot \ddot{\mathbf{u}} + \frac{Mg\eta}{N} \cdot \begin{bmatrix} \frac{1}{r_1} \\ \frac{1}{r_2} \\ \dots \\ \frac{1}{r_N} \end{bmatrix} \right)^T \quad (3.17)$$

Таким образом уравнение 3.17 определяет динамическая модель тележки, учитывающую её массу и силу трения качения. Зная координаты пути  $\mathbf{u} = (x, y, \varphi)$  и вычисляя ускорение для каждой координаты  $\ddot{\mathbf{u}} = (\ddot{x}, \ddot{y}, \ddot{\omega})$ , из уравнения 3.17 определяются соответствующие заданному пути моменты сил каждого колеса.

#### **4 Кинематическая модель роликонесущего колеса**

Для нероликонесущего колеса, которое вращается по поверхности без скольжения, мгновенная скорость низшей точки касания равна нулю [2]. На практике соприкасающиеся тела из-за физических ограничений всегда соприкасаются множеством точек, называемым пятном контакта. В качестве иллюстрации можно привести гусеничную тележку,двигающуюся по поверхности без просткальзываний. Пятно касания гусеницы достаточно большое чтобы заметить, что в определенное мгновение оно не двигается относительно поверхности. При достаточно низкой скорости тележки можно заметить, что некоторая подобласть пятна касания гусеницы не двигается относительно поверхности некоторый промежуток времени.

Особенность роликонесущих колес заключается в том, что мгновенная точка касания имеет ненулевую скорость относительно поверхности качения, так как укрепленные на колесе ролики под силой тяжести тележки вращаются относительно своей оси.

Простейшей неголономной моделью роликонесущего колеса является плоский диск, для которого скорость точки соприкосновения с несущей поверхностью направленная вдоль прямой, составляющий постоянный угол с плоскостью колеса.

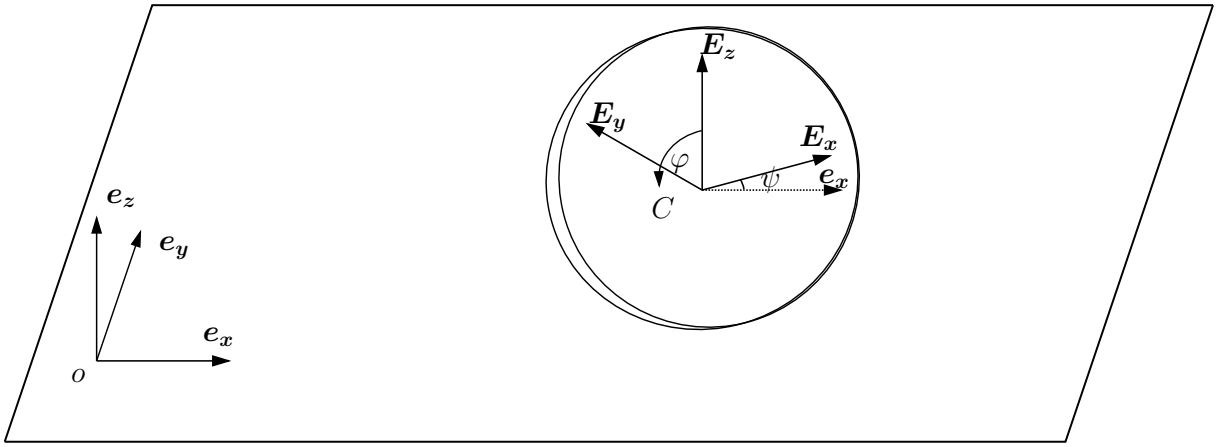


Рисунок 4.1 – модель роликнесущего колеса: вид сбоку

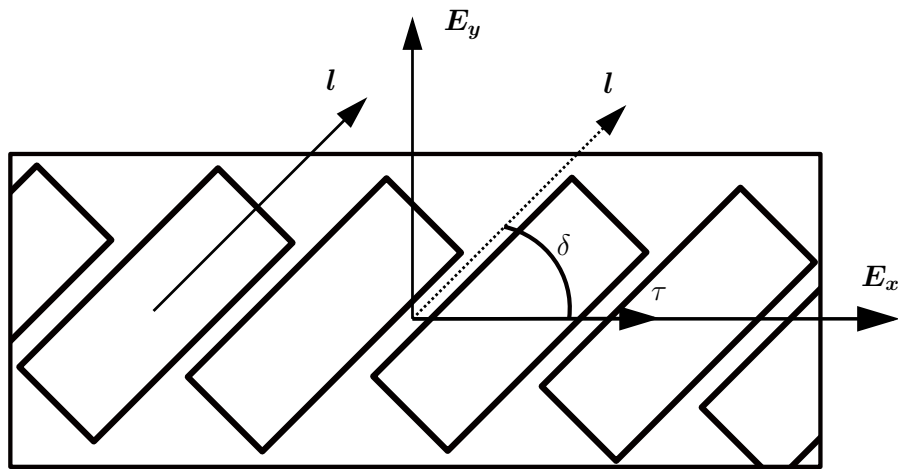


Рисунок 4.2 – модель роликнесущего колеса: вид сверху

В качестве модели роликнесущего колеса рассматривается диск радиуса  $R$  с центром в точке  $C$ , причем плоскости колес тележки вертикальны и неподвижны относительно платформы тележки[2]. Обозначения:  $l$  - единичный вектор оси вращения ролика,  $\tau$  - касательная к плоскости диска,  $\delta = \widehat{ll\tau}$ . Тогда уравнение связи имеет вид:

$$(v_M, l) = 0 \quad (4.1)$$

где  $M$  - точка касания,  $v_M$  - скорость точки касания. Вектор  $v_M$  определяется следующим образом:

$$v_M = v_C + \omega \times CM \quad (4.2)$$

где  $v_C$  - скорость центра колеса,  $\omega$  - вектор угловой скорости колеса. Обозначения:  $\{O, e_x, e_y, e_z\}$  - глобальная инерциальная система координат,  $\{C, E_x, E_y, E_z\}$  - система координат, жестко связанная с диском. Плоскость  $\{C, E_x, E_y\}$  параллельна плоскости поверхности.  $\varphi$  - угол поворота диска вокруг перпендикулярной плоскости диска оси, проходящей через точку  $C$  против часовой стрелки,  $\psi = \widehat{\tau e_x}$ . Обозначения иллюстрируются на рисунках 4.1 и 4.2. Тогда вектора  $\omega$  и  $CM$  можно представить в виде:

$$\omega = \dot{\varphi} E_y + \dot{\psi} E_z \quad (4.3)$$

$$CM = -R E_z \quad (4.4)$$

Отсюда справедливо

$$\omega \times CM = -R \dot{\varphi} E_x \quad (4.5)$$

Обозначая за  $x_C$   $y_C$  координаты точки  $C$  в глобальных координатах, проектируя вектор скорости точки  $C$  на локальную систему координат, что иллюстрируется на рисунке 4.3, верно уравнение:

$$v_C = (\dot{x}_C \cos \psi + \dot{y}_C \sin \psi) E_x + (-\dot{x}_C \sin \psi + \dot{y}_C \cos \psi) E_y \quad (4.6)$$

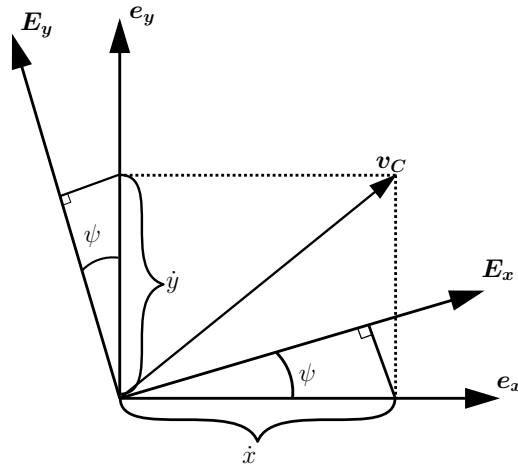


Рисунок 4.3 – иллюстрация проектирования вектора скорости на локальную систему координат

Проецируя единичный вектор  $l$  на локальную систему координат:

$$l = \cos\delta \mathbf{E}_x + \sin\delta \mathbf{E}_y \quad (4.7)$$

определяем уравнение связи в виде

$$\begin{aligned} (\mathbf{v}_m, l) &= (\mathbf{v}_c + \boldsymbol{\omega} \times \mathbf{CM}, l) = \\ &= ((\dot{x}_C \cos\psi + \dot{y}_C \sin\psi - R\dot{\varphi}) \mathbf{E}_x + (-\dot{x}_C \sin\psi + \dot{y}_C \cos\psi) \mathbf{E}_y, \cos\delta \mathbf{E}_x + \sin\delta \mathbf{E}_y) = \\ &= (\dot{x}_C \cos\psi + \dot{y}_C \sin\psi - R\dot{\varphi}) \cos\delta + (-\dot{x}_C \sin\psi + \dot{y}_C \cos\psi) \sin\delta = 0 \\ &\quad \dot{x}_C \cos(\psi + \delta) + \dot{y}_C \sin(\psi + \delta) = R\dot{\varphi} \cos\delta \end{aligned} \quad (4.8)$$

Уравнение 4.8 связывает между собой тройку координат  $(x_C, y_C, \psi)$  с угловой скоростью колеса  $\varphi$ . В работе [11] изучена динамика тележки с  $N$  роликонесущими колесами и сформулирован критерий управляемости: если  $N \geq 3$ , хотя бы одна пара векторов  $\mathbf{l}_i, \mathbf{l}_j$   $i, j = \overline{1, N}$ , которые соответствуют осям роликов, не параллельна и точки контакта колес не лежат на одной прямой, то для любой траектории всегда возможно найти такие управляющие моменты (то есть функции  $\varphi_i(t)$ ), что тележка, управляемая этими моментами, переместится по заданной траектории.

## 5 Кинематическая модель тележки на четырех роликонесущих колесах

В рамках данной главы рассматривается кинематическую модель тележки на четырех роликонесущих колесах, проиллюстрированная на рисунке 5.1. Обозначения:  $C$  - точка вращения тележки,  $\{O, e_x, e_y, e_z\}$  - глобальная инерциальная система координат,  $\{C, E_x, E_y, E_z\}$  - система координат, жестко связанная с тележкой. Плоскость  $\{C, E_x, E_y\}$  параллельна плоскости поверхности.  $r_i$  - радиус  $i$ -го колеса,  $\delta_i$  - угол между касательной к плоскости колеса  $p_i$  и осью ролика  $l_i$ ,  $\psi$  - угол между осью  $e_x$  и  $E_x$ ,  $\alpha_i$  - угол между осью  $E_x$  и радиус вектором  $R_i$  длины  $R_i$ , направленным из точки  $C$  в точку касания с поверхностью  $i$ -го колеса  $C_i$ ,  $i = \overline{1, 4}$ .

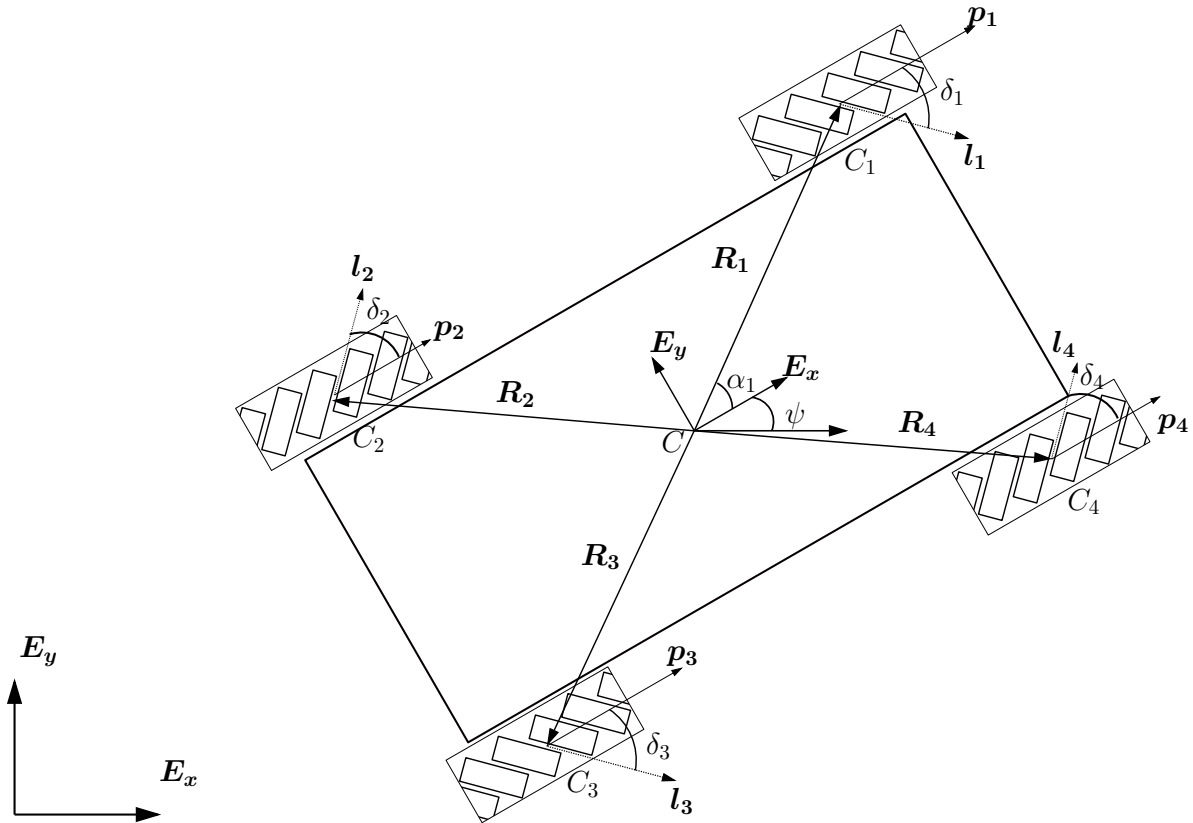


Рисунок 5.1 – модель тележки на четырех роликонесущих колесах



Рассматривается вектор скорости центра  $i$ -го колеса  $v_{c,i}$ . Результатом его разложения на поступательную и вращательную составляющие является:

$$v_{c,i} = v_{tr,i} + v_{rot,i} \quad (5.1)$$

Вектор поступательной скорости  $v_{tr,i}$  имеет координаты  $[\dot{x}, \dot{y}, 0]$ . Уравнение связи с угловой скоростью  $i$  - го колеса  $\dot{\varphi}_i$ , основываясь на уравнении 4.8 имеет вид:

$$\dot{x}\cos(\psi + \delta) + \dot{y}\sin(\psi + \delta) = r_i\dot{\varphi}_i\cos\delta_i \quad (5.2)$$

Где  $[\dot{x}, \dot{y}, 0]$  - координаты вектора поступательной скорости тележки.

Для того, чтобы заставить тележку вращаться вокруг оси вращения  $CE_z$ , необходимо задать вектор скоростей каждого колеса перпендикулярно радиус вектору  $R_i$ . При этом длина вектора должна составлять  $R_i[\dot{\varphi}_i]$ . Расположение векторов иллюстрируется на рисунке 5.2.

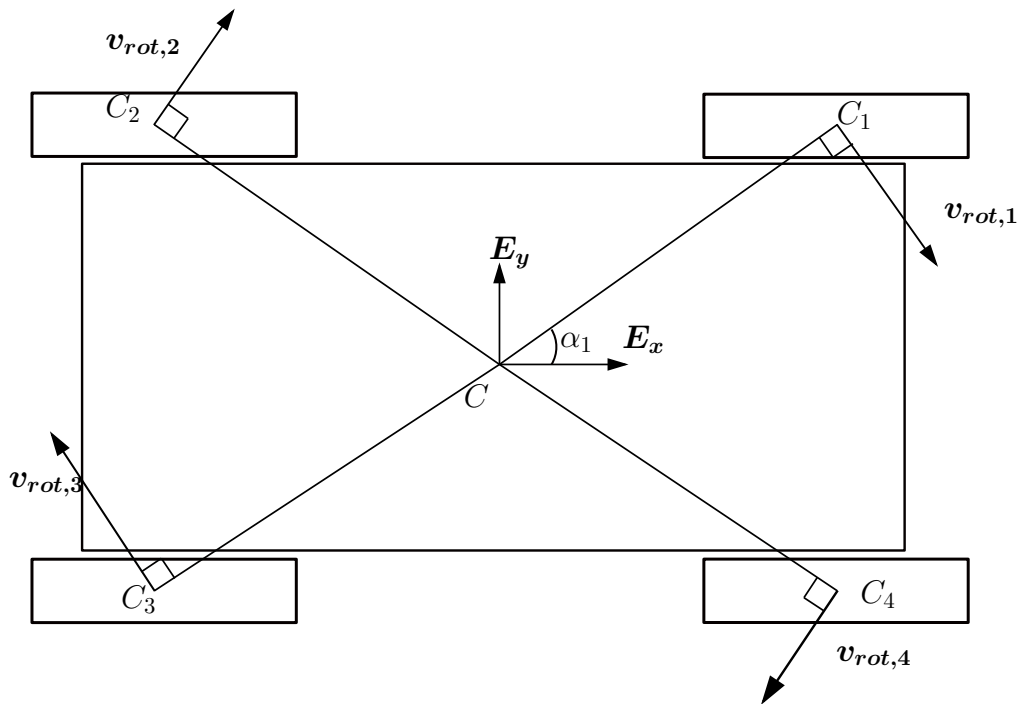


Рисунок 5.2 – вращение тележки на четырех роликонесущих колесах  
вокруг оси вращения

Рассматривается вектор вращательной скорости  $i$ -го колеса  $\mathbf{v}_{rot,i}$ .  
Результатом его проекции на оси  $\mathbf{E}_x$   $\mathbf{E}_y$ , как показано на рисунке 5.3, является

$$\mathbf{v}_{rot,i} = R_i \dot{\psi} \sin(\alpha_i) \mathbf{E}_x - R_i \dot{\psi} \cos(\alpha_i) \mathbf{E}_y \quad (5.3)$$

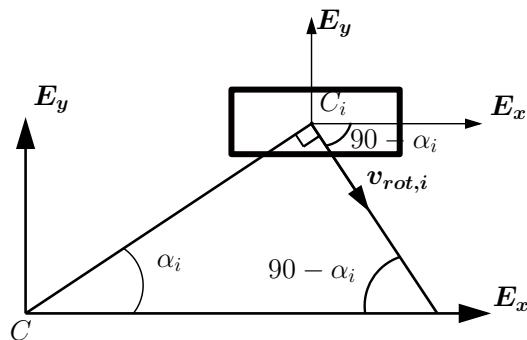


Рисунок 5.3 – проекция вектора вращательной скорости на локальную систему координат

Результатом проекции вектора  $\mathbf{v}_{rot,i}$  на оси  $\mathbf{e}_x$   $\mathbf{e}_y$ , как изображается на рисунке 5.4, является

$$\mathbf{v}_{rot,i} = R_i \dot{\psi} \sin \alpha_i (\cos \psi \mathbf{e}_x + \sin \psi \mathbf{e}_y) - R_i \dot{\psi} \cos \alpha_i (-\sin \psi \mathbf{e}_x + \cos \psi \mathbf{e}_y)$$

$$\mathbf{v}_{rot,i} = R_i \dot{\psi} \sin(\alpha_i + \psi) \mathbf{e}_x - R_i \dot{\psi} \cos(\alpha_i + \psi) \mathbf{e}_y \quad (5.4)$$

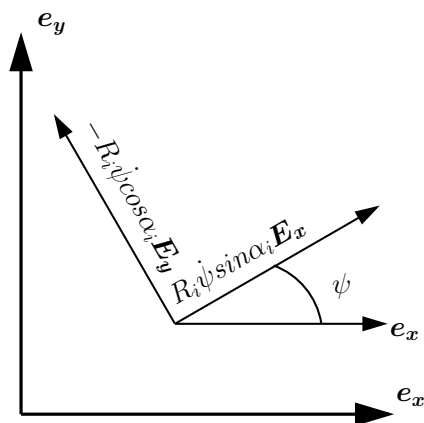


Рисунок 5.4 – проекция вектора вращательной скорости на глобальную систему координат

Тогда результатом подстановки 5.4 в уравнение связи 4.8 является выражение:

$$R_i \dot{\psi} \sin(\alpha_i + \psi) \cos(\psi + \delta_i) - R_i \dot{\psi} \cos(\alpha_i + \psi) \sin(\psi + \delta_i) = r_i \dot{\varphi}_i \cos \delta_i$$

$$R_i \dot{\psi} \sin(\alpha_i - \delta_i) = r_i \dot{\varphi}_i \cos \delta_i \quad (5.5)$$

Тогда уравнение связи для  $v_{c,i}$ , исходя из 5.1, принимает вид

$$\dot{x} \cos(\psi + \delta_i) + \dot{y} \sin(\psi + \delta_i) + R_i \dot{\psi} \sin(\alpha_i - \delta_i) = r_i \dot{\varphi}_i \cos \delta_i \quad (5.6)$$

Результатом выражения  $\dot{\varphi}_i$  из 5.1 является

$$\dot{\varphi}_i = \frac{\dot{x} \cos(\psi + \delta_i) + \dot{y} \sin(\psi + \delta_i) + R_i \dot{\psi} \sin(\alpha_i - \delta_i)}{r_i \cos \delta_i} \quad (5.7)$$

При чисто поступательном ( $\dot{\psi}_i = 0$ ) или чисто вращательном ( $\dot{x} = \dot{y} = 0$ ) движении связь 5.7 является голономной, то есть угловое ускорение каждого колеса не зависит от предыдущих движений робота. В противном случае, для вычисления 5.7 необходимо учитывать текущий угол поворота следующим образом:

$$\psi = \int_{t_0}^{t_{cur}} \dot{\psi}(t) dt \quad (5.8)$$

Где  $t_0, t_{cur}$  - начальный и текущий момент времени соответственно.

Таким образом представляется зависимость величины угловой скорости от координат пути. Для того, чтобы найти угловую скорость  $i$ -го колеса,  $i = \overline{1, 4}$ , соответствующую заданному пути  $(x, y, \psi)$ , необходимо вычислить  $(\dot{x}, \dot{y}, \dot{\psi})$  и подставить в уравнение 5.7.

## **6 Технологии моделирования робототехнических систем**

### **6.1 Фреймворк для программирования робототехнических систем ROS**

ROS (Robot Operating System - операционная система для роботов) - фреймворк с открытым исходным кодом, нацеленный на реализацию программного обеспечения для роботизированных систем[12]. ROS включает в себя коллекцию инструментов, библиотек и соглашений, нацеленных на упрощение разработки сложных роботизированных систем разных конфигураций.

Другими словами ROS - уровень абстракции над аппаратными средствами робототехнических систем. Однако несмотря на то, что в названии присутствует связка операционная система, ROS операционной системой не является. Для функционирования ROS необходима операционная система в привычном смысле; ROS поддерживает множество linux дистрибутивов, а также windows и android. В некоторых источниках такую модель абстракции называют мета-операционной системой.

После установки, ROS предоставляет следующие функциональные возможности [15]:

- Распределенный процесс: робототехническая система представляется как множество связанных друг с другом узлов, каждый из которых работает независимо от остальных;
- Использование пакетов: несколько программных компонентов или узлов, непосредственно связанных друг с другом можно объединить в пакет, который легко интегрируется в другую робототехническую

систему;

- API: поставляемые вместе с системой ROS библиотеки позволяют управлять внутренними процессами системы посредством распространенных языков программирования: python, C++, lisp и другими.

В открытом доступе доступно множество пакетов, которые реализуют управление кинематикой, методы компьютерного зрения совместно с библиотекой openCV, измерение параметров окружения с помощью физических датчиков и сенсоров, построение карт окружения робота, планирование маршрута в двумерном и трехмерном пространстве и другие.

Система ROS нацелена на ускорение развития робототехники в целом, понижая порог входа в область построения роботов. Используя доступные пакеты и существующие системы, разработка роботизированных систем значительно ускоряется.

Архитектура системы ROS делится на следующие уровни:

- уровень файловой системы;
- уровень графа вычислений системы;
- уровень глобального распространения пакетов.

### **6.1.1 Уровень файловой системы**

На этом уровне архитектуры системы ROS описываются элементы, хранящиеся в памяти операционной системы

- пакет;
- мета-пакет;
- пакетный манифест;
- репозиторий;

- определение типов сообщений
- определение сервисных типов

Здесь и далее термины определяются в контексте системы ROS.

Пакет — главная единица организации программного обеспечения. Пакет может содержать узлы, связанные библиотеки, файлы данных, конфигурацию и любую другую информацию, необходимую для описания пакета. Пакет - единица компиляции или сборки программного обеспечения робота, содержащая в себе конфигурацию, необходимую для компиляции.

Мета-пакет — специализированный пакет, включающий в себя исключительно информацию для правильной компиляции пакета инструментом catkin. Catkin - официальная система сборки пакетов ROS, являющаяся по сути множеством макросов распространенной системы сборки CMake.

Пакетный манифест — файл формата XML, имеющий фиксированное имя - package.xml и хранящий такие метаданные пакета, как его наименование, список разработчиков, сторонние зависимости и другую мета-информацию. Структура файла строго определена в специализированном документе.

Репозиторий — коллекция пакетов, совместимых с одной из версий системы ROS.

Определение типов сообщений — файлы, хранящиеся обычно в подпапке /msg и имеющие расширение .msg . Эти файлы хранят в себе описание определенных разработчиком типов сообщений. Такое сообщение состоит из одного или нескольких объектов стандартного типа std\_msgs или производных типов.

Определение сервисных типов — файлы, хранящиеся обычно в

подпапке /srv, определяющие структуру запросов и ответов определенных разработчиком сервисов ROS.

### 6.1.2 Уровень графа вычислений системы

Графом вычислений в контексте системы ROS называется сеть связанных друг с другом узлов, работающих независимо друг от друга и общающихся посредством передачи сообщений. На рисунке 6.1 можно видеть пример такого графа, изображенного с помощью инструмента rqt\_graph.

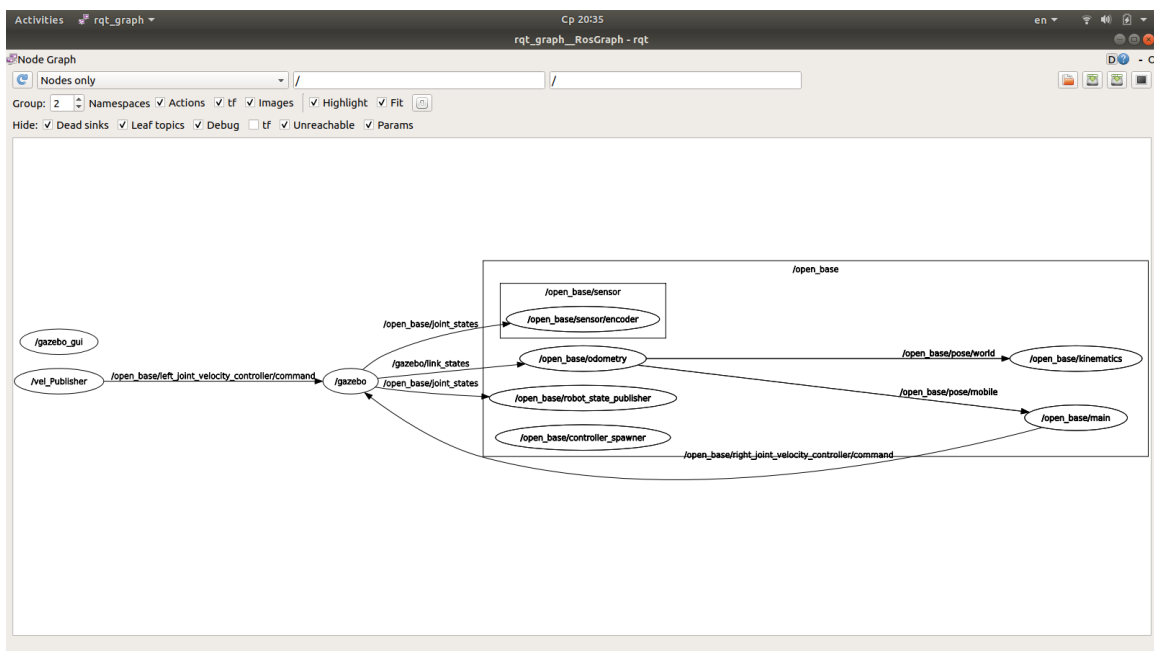


Рисунок 6.1 – граф вычислений во время тестирования роботизированной тележки

Граф вычислений состоит из следующих элементов:

- узел (node);
- главный сервис управления - мастер (master);
- сообщение (message);
- тема (topic);

- сервис (service);
- рюкзак (bag).

Узел — процесс, производящий некоторые вычисления, играющий роль подписчика и издателя в топологии вычислительного графа.

Мастер, или главный сервис управления — программа, регистрирующая узлы и сервисы в графе вычислений и отвечающая за связь узлов между собой. Для связи элементов графа мастер использует tcp/ip протокол, что позволяет связывать узлы графа по сети.

Сообщение — структура данных, передаваемая между узлами. Состоит из стандартных `std_msgs` или производных типов данных.

Тема — промежуточное звено между узлами. Коммуникация между узлами вычислительного графа подчиняется шаблону проектирования издатель-подписчик. В контексте ROS, каждый узел является как издателем, так и подписчиком. Для того, чтобы связать два узла, необходимо определить тему, через который будет происходить коммуникация. Узел-издатель публикует сообщение по адресу темы, в то время как узел-подписчик подписывается на эту тему и получает уведомление каждый раз, когда по её адресу передается сообщение. Такой подход к построению топологии позволяет увеличить независимость вычислительных узлов.

Сервис — процесс, производящий некоторые вычисления, подчиняющийся шаблону проектирования запрос-ответ. Используется в тех случаях, когда необходимо установить двустороннюю связь между процессами.

Рюкзак — формат хранения последовательно передаваемых данных. Позволяет сохранять и воспроизводить последовательности сообщений во времени.



### 6.1.3 Формат описания робототехнических систем URDF

Архитектура ROS описывает способы ведения вычислений связанных с робототехнической системой, однако не позволяет описать таковую. Для описания роботов в системе ROS используется формат URDF.

URDF (Universal Robotic Description Format) – формат описания робототехнических систем в трехмерном пространстве, основанный на XML. Документ представляет из себя XML файл с предопределенным набором тегов. Робот представляется как совокупность звеньев (link) и сочленений (joint). Кроме того, в URDF файле возможно описание сенсоров и других вспомогательных параметров.

Корневым элементом модели является тег <robot> с атрибутом name, в который вкладываются теги звеньев <link> и сочленений <joint>.

Звено (link) описывает твердое тело с инерцией, визуальными особенностями и физическими ограничениями. Звено содержит в себе следующие элементы, описывающие:

- момент инерции <inertial>
- внешний вид твердого тела <visual>
- границы коллизии <collision>

Описание момента инерции, кроме непосредственно тензора инерции, включает в себя такие параметры, как относительный сдвиг, поворот инерциальных координат и массу.

Внешний вид определяется цветом, текстурой и формой объекта. В качестве формы могут использовать как стандартные типы (параллелепипед, цилиндр, сфера), так и трехмерные модели.

Границы коллизии описывают границы твердого тела. Используется

для обнаружения пересечений между собой двух или более объектов. Разделение внешнего вида от границ коллизии позволяет значительно ускорить вычисления в случае сложной визуальной модели, вычисления коллизии с которой не является принципиальным элементом симуляции системы.

Описывая робототехническую систему, состоящую из нескольких звеньев, необходимо задать информации о их сочленении. В описании URDF используются следующие типы сочленений:

- фиксированное сочленение (fixed);
- шарнирное сочленение (revolute) задает ось и границы вращения звеньев относительно друг друга;
- непрерывное сочленение (continuous) является шарнирным соединением без ограничений на угол поворота
- призматическое сочленение (prismatic) позволяет одному звену двигаться относительно другого вдоль некоторой прямой;
- планарное сочленение (planar) позволяет одному звену двигаться относительно другого в некоторой плоскости;
- плавающее сочленение (floating) позволяет одному звену двигаться относительно другого в трех измерениях.

Каждый тег сочленения содержит в себе ссылку на опорное звено, звено-родитель <parent>, и, соответственно, движимое звено, звено-наследник <child>. В зависимости от типа тег сочленения включает в себя описание оси и границ вращения, ограничения на скорость вращения, силу трения между сочленением и другие параметры[12].

## 6.2 Среда моделирование робототехнических систем Gazebo

Gazebo — среда моделирования робототехнических систем в трехмерном пространстве с открытым исходным кодом. Имеет поддержку нескольких пакетов моделирования физических процессов, таких как ODE, Bullet, DART и других. Gazebo позволяет симулировать реальное окружение и сценарии среды, для которой разрабатывается робототехническая система. Кроме того, среда моделирования Gazebo позволяет симулировать реалистичные сигналы для виртуальных датчиков, таких как стерео камеры, системы лазерного измерения расстояния между объектами LIDAR и других сенсоров. На рисунке 6.2 изображается процесс моделирования трехколесной тележки.

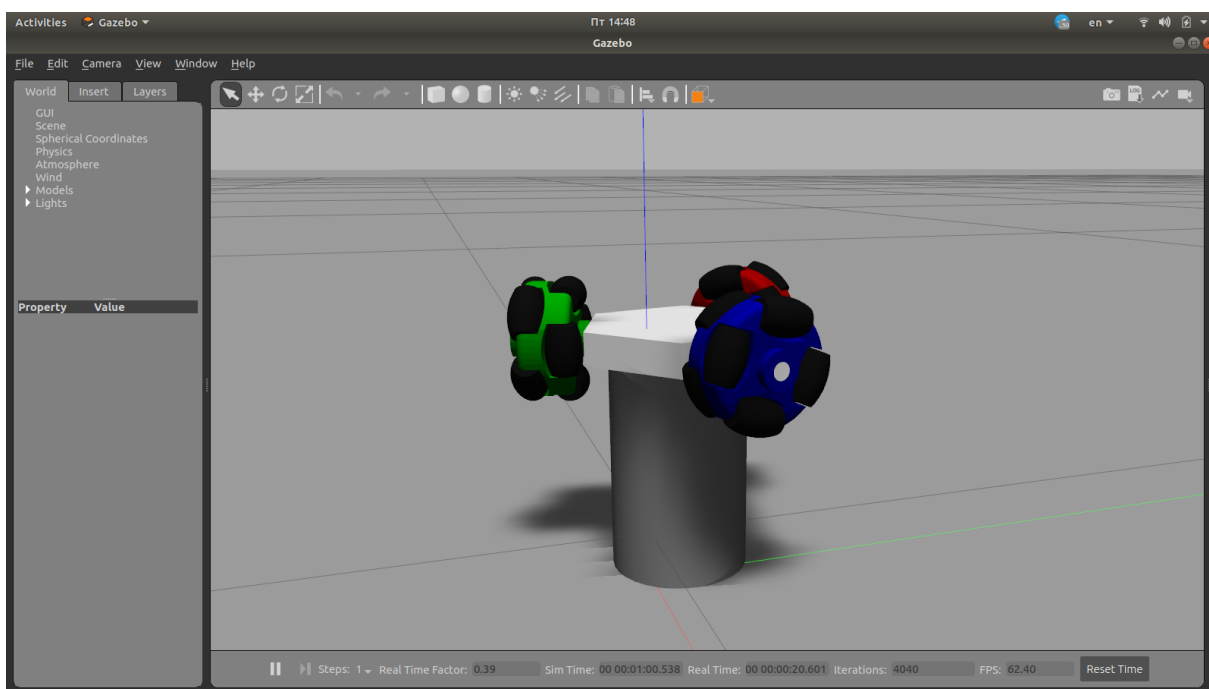


Рисунок 6.2 – Симуляция трехколесной тележки в среде симуляции Gazebo

Среда Gazebo повсеместно используется в соревнованиях, работе и исследованиях таких компаний, как Nasa, Toyota, DARPA и других

компаниях [13].

Для моделирование робототехнических систем Gazebo предоставляет формат SDF, альтернативный формату URDF. Формат SDF во многом схож с URDF: оба являются диалектом языка XML и описывают модель робота с помощью совокупности звеньев и сочленений. Однако, кроме этого, формат SDF позволяет изменять параметры симуляции и содержит описание сенсоров и плагинов управления роботом. С помощью SDF файлов описывается расположение и конфигурация объектов, в том числе других SDF или URDF моделей, на сцене.

Для того, чтобы смоделировать сцену, среде моделирования Gazebo необходимо передать стартовый SDF файл, содержащий описание сцены, которую необходимо смоделировать. Роботизированная система, попадая в симуляцию, подчиняется конфигурации окружающей среды, заданной в стартовом SDF файле. По умолчанию на систему действует сила тяжести, сила трения, и другие силы.

Управление симуляцией возможно с помощью плагинов Gazebo. Плагин представляет из себя скомпилированную библиотеку, присоединяющуюся к объекту симуляции через SDF файл. Имеется множество встроенных плагинов управления кинематикой, камерой и другими сенсорами. Кроме того, с помощью Gazebo API - набора библиотек взаимодействия с программой Gazebo, возможно описать собственный плагин управления симуляцией.

Плагины Gazebo делятся на следующие категории:

- плагин окружения (world);
- плагин модели (model);
- плагин системы (system);
- плагин визуальных параметров (visual);

- плагин графического интерфейса (gui);
- плагин сенсора (sensor).

Каждый тип плагина управляется отдельным компонентом Gazebo. Например, плагин модели присоединяется к конкретной модели и позволяет управлять ею. Плагин окружения подключается к среде моделирования, а плагин сенсора - к определенному сенсору.

Функциональность плагина зависит от его типа. Плагин окружения используется для управления свойствами симуляции, такими как физические характеристики симуляции, окружающее освещение и так далее. Плагин модели используется для управления сочленениями модели. Плагин сенсора используется для получения информации о сенсоре и управления его характеристиками.

### **6.3 Системы автоматического управления. ПИД регуляторы**

Как известно из механики твердых тел для того, чтобы в инерциальной системе исчисления заставить тело двигаться, необходимо приложить к нему силу, придавая тем самым ускорение этому телу. В реальных условиях невозможно придать телу бесконечное ускорение, поэтому тело достигает определённой скорости только через определенный момент времени [7].

Теория систем автоматического управления изучает способы своевременного установления определенного состояния некоторой системы. В частности, в качестве системы можно рассматривать ведущее колесо тележки, а в качестве состояния — определенная скорость вращения этого колеса [14].

Распространенный способ управления состоянием системы — использование пропорционально-интегрально-дифференцирующего

регулятора (ПИД). ПИД регулятор – устройство в управляющем контуре с обратной связью. Его устройство проиллюстрировано на рисунке 6.3.

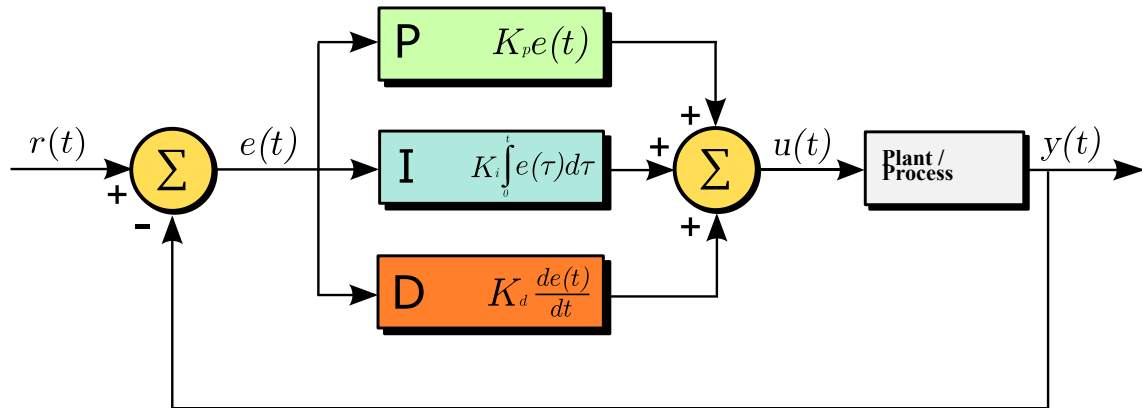


Рисунок 6.3 – Схема устройства ПИД регулятора

На вход ПИД регулятора поступает два сигнала: требуемая величина некоторой характеристики  $r(t)$  и реальное значение этой характеристики  $y(t)$ . В реальных системах сигнал  $y(t)$  снимается с помощью сенсоров или датчиков. В случае ПИД регуляции температуры этот показатель может быть показателем датчика температуры.

На следующем этапе вычисляется значение ошибки  $e(t)$  как

$$e(t) = r(t) - y(t) \quad (6.1)$$

после чего вычисляется величина корректировки  $u(t)$  :

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (6.2)$$

Где  $K_p$ ,  $K_i$ ,  $K_d$  – характеристики ПИД регулятора, коэффициенты пропорциональной, интегральной и дифференциальной составляющих соответственно.

Величина  $u(t)$  добавляется к управляющему сигналу  $r(t)$ , корректируя его поведение.

В случае дискретной функции управления  $r(n), n \in N$  величина ошибки и корректировки вычисляется следующим образом:

$$e(n) = r(n) - y(n) \quad (6.3)$$

$$u(n) = K_p e(n) + K_i^{discr} \sum_{k=0}^n e(k) + K_d^{discr} (e(n) - e(n-1)) \quad (6.4)$$

где  $K_p, K_i^{discr}, K_d^{discr}$  — коэффициенты пропорциональной, интегральной и дифференциальной дискретных составляющих соответственно.

Подбирая коэффициенты элементов ПИД регулятора, можно добиться устойчивой работы системы. Правильная настройка регулятора позволяет достаточно быстро компенсировать отклонения от требуемого значения некоторых параметров системы, приближая теоретические расчеты параметров к реальным.

## 7 Программная реализация

### 7.1 SDF модель

В качестве моделируемой роботизированной системы в работе используется тележка, приводимая в действие тремя омни-колесами. Модель имеет следующие характеристики, проиллюстрированные на рисунке 7.1

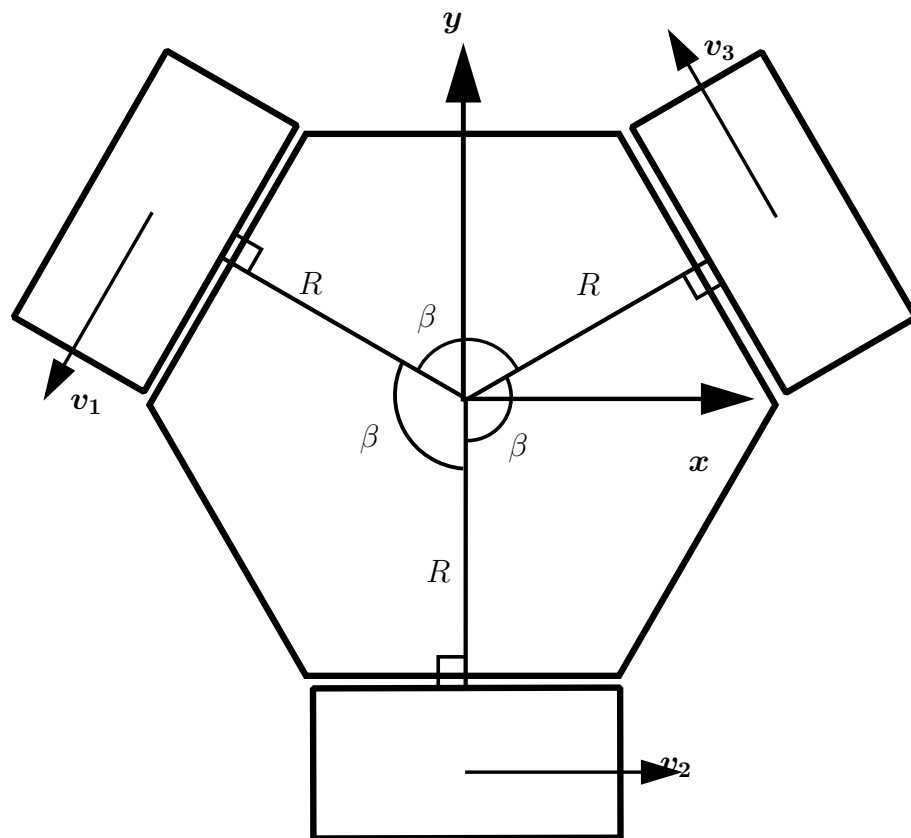
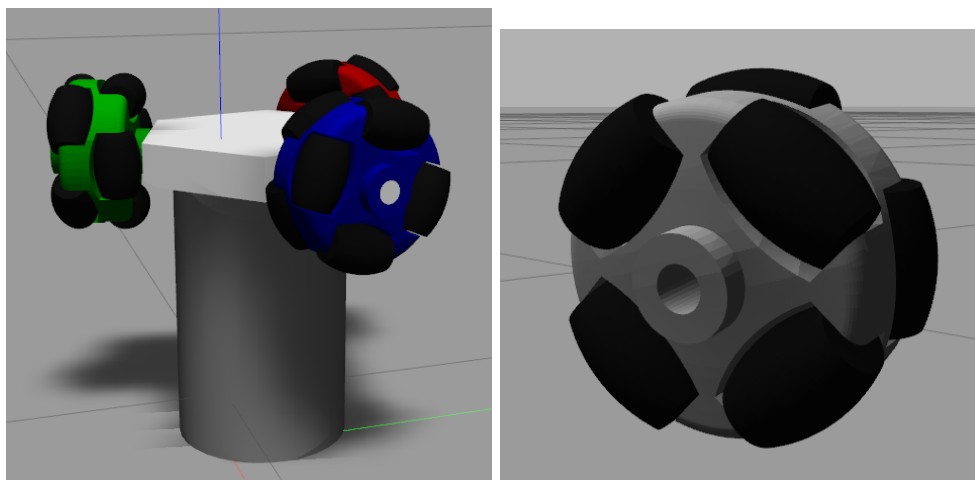


Рисунок 7.1 – Схема моделируемой трехколесной тележки

Расстояние до центра масс до каждого колеса  $R = 0.04$  единицы, радиус колеса  $r = 0.01905$  единиц. Угол  $\beta$ , проиллюстрированный на рисунке 7.1 равен  $120^\circ$ , углы  $\gamma_i$  между осью  $x$  и  $v_i$  равны  $150, 270, 30$  градусов соответственно.





а) трехколесная тележка

б) омни-колесо

Рисунок 7.2 – SDF модели в симуляции Gazebo

На рисунке 7.2 иллюстрируются смоделированные модели: модель омни-колеса и трехколесная тележка, опирающаяся на три таких омни-колеса. Каждое омниколесо имеет на своем ободе восемь роликов, каждый из которых вращается вокруг собственной оси. Все соединения имеют шарнирный тип revolute, описывающийся в главе 6.1.3.

## 7.2 Плагин модели Gazebo

Для управления моделью тележки в данной работе используется Gazebo API, а именно плагин уровня модели. Плагин позволяет двигать тележку в любом заданном направлении голономно, то есть без вращения вокруг собственной оси. Плагин имеет два режима работы: управление непосредственно с помощью команд, поступающих извне и следование по заданному маршруту.

После подключения к симуляции, плагин начинает публиковать тему, в которой сообщается положение робота на плоскости. Эта информация используется клиентским приложением, описанным в следующей главе. Кроме того, плагин подписывается на тему `"/имя робота"/velocity`

типа VelocityWheels. Описание типа сообщения содержится в файле /msg/VelocityWheels.msg.

Клиентское приложение, это может быть любой узел, подписавшийся на эту тему, публикует величину скорости в тему /"имя робота"/velocity. Плагин получает эти сообщения и изменяет скорости колес с помощью функции setTargetVelocity.

Функция

setTargetVelocity(double left, double right, double back)

задает скорости каждого колеса. Для устойчивого управления используется четыре ПИД регулятора, реализация которых включается в Gazebo API. Три ПИД регулятора подключены в цепь управления скоростью вращения колес. Gazebo API позволяет получать реальную скорость вращения колес, что соответствует дискретной функции  $y(n)$  из выражения 6.3.

Требуется, чтобы движения тележки были голономными. Для этого используется четвертый ПИД регулятор, использующий в качестве  $e(t)$  величину отклонения поворота тележки от оси абсцисс глобальной системы координат.

Для регулирования параметров ПИД регуляторов используется стандартный пакет динамической настройки параметров dynamic\_reconfigure. В функции, вызываемой при загрузке плагина, создается поток чтения запросов о изменении параметров, описанных в файле PIDconf.cfg. Посредством утилиты rqt\_reconfigure в поток посылаются запросы на изменение конфигурации, обрабатываемые плагином.

Сообщение /"имя робота"/velocity имеет поле stop. Передавая по адресу темы скорость, с полем stop = true, плагин переконфигурируется в

режим следования по аналитически заданному пути. Плагин отписывается от текущей темы и подписывается на тему /"имя робота"/path типа PathMsg. Описание типа сообщения содержится в файле /msg/PathMsg.msg.

Далее плагин ожидает первого уведомления по этой теме. Первое сообщение содержит в себе вектор точек — аналитический маршрут. Первый элемент вектора — стартовая точка маршрута, последний элемент — финальная. Тележка из своего текущего местоположения отправляется в начальную точку и после, проходя через все промежуточные точки маршрута, хранящиеся последовательно в полученном векторе, достигает финальной точки и останавливается. По окончании следования аналитическому пути плагин переконфигурируется и заново подписывается на тему /"имя робота"/velocity.

### 7.3 Алгоритм движения тележки по аналитически заданному пути

Для движения в заданном направлении с заданной скоростью используется вывод главы 2. Согласно характеристикам, описанным в главе 7.1, уравнение 2.10 принимает вид:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin(\varphi + \alpha_1) & \cos(\varphi + \alpha_1) & R \\ -\sin(\varphi + \alpha_2) & \cos(\varphi + \alpha_2) & R \\ -\sin(\varphi + \alpha_3) & \cos(\varphi + \alpha_3) & R \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \omega \end{bmatrix} \quad (7.1)$$

где углы  $\alpha_1, \alpha_2, \alpha_3$  равны соответственно 150, 270, 30 градусов.

фиксируя величину линейной скорости тележки  $v = const$ , и зная текущее и требуемое положение тележки на плоскости  $p_0, p_1$ , при этом

$$p_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (7.2)$$

определяются величины скоростей вращения колес тележки из 7.1; вектор скорости тележки заменяется на следующее выражение:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \omega \end{bmatrix} = v \begin{bmatrix} \frac{x_1 - x_0}{norm} \\ \frac{y_1 - y_0}{norm} \\ 0 \end{bmatrix} = \mathbf{v}_{desired} \quad (7.3)$$

$$norm = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \quad (7.4)$$

Для того, чтобы тележка преодолела расстояние  $r$  единиц по прямой со скоростью  $v$  единиц в секунду, необходимо поддерживать скорость  $v$  в течение  $\frac{r}{v}$  секунд. Однако на практике движение тележки из-за физических помех является нелинейным. Зная реальную скорость вращения колес, используя преобразование, обратное 7.1, определяется реальная скорость тележки:

$$\begin{bmatrix} \dot{x}_{real} \\ \dot{y}_{real} \\ \omega_{real} \end{bmatrix} = \frac{2r}{3\sqrt{3}R} \begin{bmatrix} -\frac{\sqrt{3}}{2}R & \sqrt{3}R & -\frac{\sqrt{3}}{2}R \\ -\frac{3}{2}R & 0 & \frac{3}{2}R \\ \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \omega_{1_{real}} \\ \omega_{2_{real}} \\ \omega_{3_{real}} \end{bmatrix} = \mathbf{v}_{real} \quad (7.5)$$

где  $\omega_{i_{real}}$  — реальная угловая скорость  $i$ -го колеса,  $\dot{x}_{real}, \dot{y}_{real}, \omega_{real}$  — компоненты вектора реальной скорости  $\mathbf{v}_{real}$ . Раскладывая реальную скорость на желаемое направление  $\mathbf{v}_{desired}$  и перпендикулярное желаемому  $\mathbf{v}_{error}$ , что иллюстрируется на рисунке 7.3, величина пройденного пути  $S$  выражается следующим образом:

$$S = \sum_{k=0}^N v_{steering}^k \cdot \Delta t \quad (7.6)$$

где  $N$  — номер текущего дискретного момента времени,  $\Delta t$  — длительность одно дискретного момента времени,  $v_{steering}^i$  — проекция вектора реальной скорости на желаемое направление движения в момент

времени  $i$ . Индекс в обозначении векторов  $v_{real}^i$  и  $v_{desired}^i$  значит, что данные вектора получены в момент времени  $i$ .

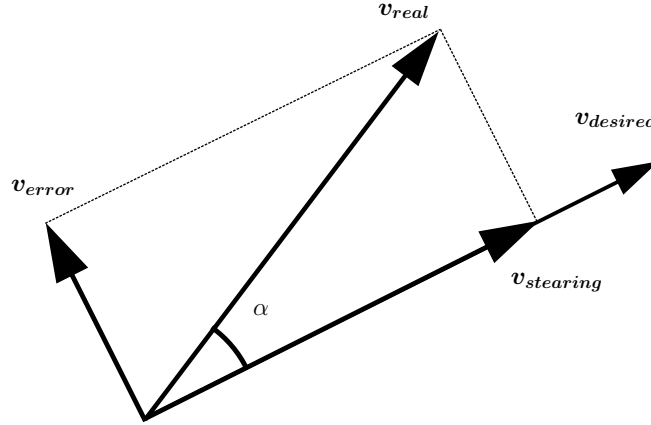


Рисунок 7.3 – Проекция реального вектора скорости на желаемое направление и перпендикулярное ему

Значение  $v_{stearing}^i$  вычисляется следующим образом:

$$v_{stearing}^i = \cos(\alpha^i) \cdot \|v_{real}^i\| \quad (7.7)$$

$$\cos(\alpha^i) = \frac{(v_{desired}^i, v_{real}^i)}{\|v_{desired}^i\| \cdot \|v_{real}^i\|} \quad (7.8)$$

В качестве нормы вектора используется Евклидова норма:

$$\|v_{real}^i\| = \sqrt{(\dot{x}_{real}^i)^2 + (\dot{y}_{real}^i)^2 + (\omega_{real}^i)^2} \quad (7.9)$$

$$\|v_{desired}^i\| = \sqrt{(\dot{x}^i)^2 + (\dot{y}^i)^2 + (\omega^i)^2} \quad (7.10)$$

Для коррекции сдвига робота относительно пути используется ПИД регулятор, использующий в качестве величины ошибки в момент времени  $i$  расстояние между текущим положением робота и текущим отрезком пути  $v_{error}^i$ :

$$v_{error}^i = \pm \frac{|(y_1 - y_0)x_c^i - (x_1 - x_0)y_c^i + x_1y_0 - y_1x_0|}{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}} \quad (7.11)$$

где  $x_c^i, y_c^i$  - положение робота в момент времени  $i$ :

$$x_c^i = x_c^{i-1} + \dot{x}_{real}^i \Delta t \quad (7.12)$$

$$y_c^i = y_c^{i-1} + \dot{y}_{real}^i \Delta t \quad (7.13)$$

где  $x_c^0 = x_0$ ,  $y_c^0 = y_0$ ,  $\dot{x}_{real}^i$ ,  $\dot{y}_{real}^i$  – компоненты вектора реальной скорости тележки в момент времени  $i$ .

Для корректировки движения робот должен двигаться в направлении, перпендикулярном отрезку  $[p_0, p_1]$ . Вектор, перпендикулярный этому отрезку, с помощью уравнения 7.1 преобразуется в скорости колес  $v_{wheels}$ . ПИД регулятор получает на вход значение  $v_{error}^i$ , вектор  $v_{wheels}$  растягивается на величину, полученную на выходе ПИД регулятора, и добавляется к общему вектору движения колес. Знак  $v_{error}^i$  выбирается с учетом того, чтобы вектор  $v_{wheels}$  направлял тележку в сторону отрезка  $[p_0, p_1]$ .

После того, как величина  $S$  совпадет с длиной пути  $r$  допускается, что робот находится в точке  $p_1$ . Если путь состоит из более чем двух точек и точка  $p_1$  не финальная —  $p_0$  заменяется на  $p_1$ , а за  $p_1$  принимается следующая промежуточная точка пути. Если же это не так — задается нулевая скорость движения и движение по аналитическому пути считается завершенным.

Для коррекции заноса тележки в поворотах можно сократить требуемую длину отрезка  $S$  на величину  $S_{skid}$ :

$$S_{skid} = C\theta \quad (7.14)$$

где  $C$  - коэффициент управления заносом,  $\theta$  - угол следующего поворота тележки:

$$\theta = \arccos\left(\frac{b^2 + c^2 - a^2}{2bc}\right) \quad (7.15)$$

где

$$a = \|p_0, p_2\| \quad (7.16)$$

$$b = \|p_1, p_2\| \quad (7.17)$$

$$c = ||p_0, p_1|| \quad (7.18)$$

считая  $p_2$  - следующей за  $p_1$  точкой пути. В качестве нормы используется Евклидова метрика.

## 7.4 Клиент управления

Клиент управления реализует два способа управления тележкой: непрерывная передача величин скорости тележки и передача аналитического пути. Графический интерфейс реализуется средствами библиотек QT. Для взаимодействия с плагином модели используются механизмы передачи сообщений и запросов ROS.



Рисунок 7.4 – Графический интерфейс клиента управления тележкой

Программа состоит из трех виджетов, изображенных на рисунке 7.4. Слева-направо: виджет задания аналитического пути, виджет отображения положения робота и последнего пройденного аналитического пути, джойстик непрерывного управления скоростью тележки.

Виджет задания аналитического пути описывается в файле `carpathsetter.h`. Задание аналитического пути производится путем

рисования ломаной линии на холсте. Границы координат, которые способен отобразить этот виджет - от  $-5$  до  $5$  единиц по каждой из осей, поэтому путь задается только в этих пределах стимулируемой сцены. По нажатию кнопки, расположенной под виджетом, аналитический путь преобразуется в вектор точек и передается плагину управления путем передачи запроса плагину через реализованный в плагине сервис "pathSrv". Кроме того, путь передается напрямую виджету отображения пути для его отрисовки. Клик правой кнопкой мыши позволяет отчистить холст виджета без отправки пути в плагин.

Виджет отображения пути описывается в файле `cartpathgetter.h`. Во время работы плагин публикует собственную позицию на плоскости в тему `"/имя робота"/pos`. Эти данные читаются клиентским приложением посредством класса `RosSubscriber`, описанного в файле `rospublisher.h`. Средствами связи, встроенными в QT (технология сигналов и слотов), эти данные передаются виджету отображения положения робота. После публикации нового аналитического пути виджет удаляет уже отрисованные элементы и рисует аналитический путь. Границы координат, которые способен отобразить этот виджет - от  $-5$  до  $5$  единиц по каждой из осей.

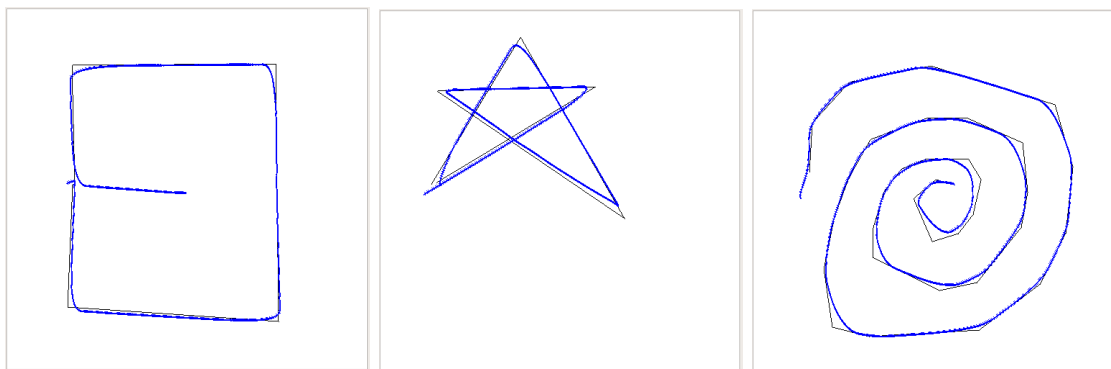
Виджет джойстика описывается в файле `cartcontrollerwidget.h`. Виджет получает координаты клика, вычисляет направление вектора скорости как разность между текущей координатой и серединой окружности джойстика и вычисляет соответствующие скорости колес. Функции преобразования вектора скорости тележки в скорости колес реализованы в файле `cartkinematic.h` и основываются на выводе главы 7.3. Класс этого виджета содержит в себе экземпляр издателя, класса `RosPublisher`, описанного в файле `rospublisher.h`.



Класс `RosPublisher` с частотой 1000 герц публикует значение скорости, хранимое в поле `velocity` типа сообщения `VelocityWheel`, описанного в файле `/msg/VelocityWheel.msg`. Для изменения публикуемой скорости в классе `RosPublisher` реализован метод `setVelocity`. Виджет джойстика вызывает метод `setVelocity` на последнем этапе передачи сообщения плагину, передавая преобразованную в угловые скорости колес глобальный вектор скорости.

### 7.5 Тестирование алгоритма движения по аналитически заданному пути

Моделируемая тележка демонстрирует результаты, изображающиеся на рисунке 7.5. Черная ломаная линия - аналитически заданный путь, синяя кривая - маршрут робота.



а) испытание 1

б) Испытание 2

в) Испытание 3

Рисунок 7.5 – карта следования робота по маршруту

Результаты тестирования отображаются в таблице 7.1.

Таблица 7.1 – сравнение результатов сравнения изображений по  
гистограмме

Номер испытания	Среднее отклонение от пути, единицы	Максимальное отклонение от пути, единицы
1	0.0105794	0.0818119
2	0.0135202	0.0786881
3	0.019622	0.138033

При условии, что модель робота имеет радиус 0.04 единицы и использует алгоритм сглаживания, описанный в конце главы 7.3, результаты тестирования можно считать удовлетворительными.

## ЗАКЛЮЧЕНИЕ

В результате работы были исследованы возможные типы передвижения робототехнических систем, в том числе распространенные виды колес. Изучена кинематика и динамика тележек, использующих  $N$  омни-колес для передвижения, а также представлена кинематическая модель колеса меканума и четырехколесной тележки, использующей эти колеса.

В системе ROS и среде симуляции Gazebo разработана виртуальная модель трехколесного робота, использующего омни-колеса. Средствами ROS API и Gazebo API разработан плагин и клиент управления этим роботом. Разработан и успешно протестирован алгоритм передвижения робота по аналитически заданному пути, основанный на кинематической модели, разработанной в процессе дипломной работы.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Roland Siegwart Introduction to Autonomous Mobile Robots /Roland Siegwart, Illah R. Nourbakhsh. — London, England : The MIT Press, 2004. — 321с.
- 2 Klaus Zimmermann: Dynamics of Mechanical Systems with Mecanum Wheels / Klaus Zimmermann, Igor Zeidis, Mohamed Abdelrahman — Springer International Publishing Switzerland, 2014. — 11с.
- 3 С. И. Ожегов. Словарь русского языка — Мир и Образование, 2008. — 1200с.
- 4 Большая Советская Энциклопедия — Советская энциклопедия, 1970.
- 5 Ksenia Shabalina. Comparative Analysis of Mobile Robot Wheels Design / Ksenia Shabalina, Artur Sagitov, Evgeni Magid. — Kazan, Russian Federation, Higher School of Information Technology and Information Systems 2018. — 5с.
- 6 Pavan Ramdya. Climbing favours the tripod gait over alternative faster insect gaits / Pavan Ramdya, Robin Thandiackal, Raphael Cherney, Thibault Asselborn<sup>1</sup>, Richard Benton , Auke Jan Ijspeert, Dario Floreano. — Nat. Commun 2017. — 11 с.
- 7 И.В.Савельев. Курс общей физики, том I. Механика, колебания и волны, молекулярная физика. — Наука, 1970. — 517с.
- 8 US patent 1305535
- 9 US patent 3789947
- 10 US patent 3876255
- 11 А.А. Зобова. Динамика экипажа с роликонесущими колесами / А.А. Зобова, Я.В. Татаринов. — Прикладная математика и механика Том 73. Выпуск 1, 2009. — 10 с.

- 12 Официальный сайт ROS. — Текст: электронный. URL:  
<https://www.ros.org/> (дата обращения: 25.10.2019 )
- 13 Официальный сайт Gazebo. — Текст: электронный. URL:  
<http://gazebo-sim.org/> (дата обращения: 1.05.2020 )
- 14 Никулин Е.А. Основы теории автоматического управления.  
Частотные методы анализа и синтеза систем / учебное пособие —  
БХВ-Петербург, 2011. — 632с.
- 15 YoonSeok Pyo. ROS Robot Programming - First Edition / YoonSeok Pyo,  
HanCheol Cho, RyuWoon Jung, TaeHoon Lim — ROBOTIS Co.,Ltd, 2017.  
ISBN 979-11-962307-1-5 — 460с.