

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА МЕХАНИКИ УПРАВЛЯЕМОГО ДВИЖЕНИЯ

Бубнов Владислав Владимирович

Выпускная квалификационная работа бакалавра

Построение траекторий в задаче прямохождения

010400

Прикладная математика и информатика

Научный руководитель,
кандидат физ.-мат. наук,
доцент
Латыпов В.Н.

Санкт-Петербург
2016

Содержание

Введение	3
Постановка задачи	4
Обзор литературы	6
Глава 1. Управление	7
1.1. Опорные траектории	7
1.2. Управляющие моменты	8
1.3. Интегрирование уравнений движения	10
1.4. Пример	12
Глава 2. Построение траекторий	14
2.1. Rapidly Exploring Random Tree	14
2.2. Адаптация алгоритма RRT	16
2.2.1. Начальные данные	16
2.2.2. Генерация величин	16
2.2.3. Добавление вершины	17
2.2.4. Аппроксимация	19
Глава 3. Эксперименты	20
3.1. Описание работы расчетной программы	20
3.2. Пример работы расчетной программы	21
Заключение	23
Список литературы	24

Введение

Человечество с давних времен грезило о создании антропоморфного механизма, способного выполнять задачи не хуже человека. Возможность заменить человека в опасных для его здоровья условиях, тестирование защитных костюмов, экзоскелеты, обслуживание клиентов в компаниях и многие другие области применения человекоподобных роботов привлекают умы и деньги со всего мира участвовать в разработках алгоритмов для решения задач, с которыми обычный человек сталкивается ежедневно.

Большинство ученых из разных стран пытаются воплотить эту мечту, участвуя в создании искусственного интеллекта, близким по характеристикам к человеческому. Машинное обучение, моделирование мыслительных процессов, машинное творчество — направления, исследования в которых проводятся последние годы особо активно.

Также актуальной остается задача механического движения и, в частности, перемещения антропоморфных механизмов, методы решения которой стали активно предприниматься еще в 70-ые годы XX века. Некоторые популярные методы основаны на ведении робота по заранее заданным траекториям [1], [2]. В подобных алгоритмах с учетом динамики механизма определяется траектория движения во время одного шага, а ходьба или бег основываются на циклическом повторении этой траектории с соответствующей сменой ног.

В случае наличия на пути робота препятствий задача конструирования функций перемещения значительно усложняется. В таких ситуациях траектории требуется вычислять, основываясь не только на динамике самого механизма, а также и на расположении, форме и состоянии объектов в окружающей среде робота.

В данной работе рассмотрена и реализована модификация вероятностного алгоритма планирования [3] движений робота для построения траекторий пятизвенного шагающего механизма по заданной поверхности. В первой главе рассмотрен метод управления механизмом, основанный на следовании заранее заданным траекториям, а также приведен пример. Во второй главе предложен алгоритм для построения функций перемещения робота с учетом его динамики и карты местности. В третьей главе рассмотрена реализация этого метода и численный эксперимент.

Постановка задачи

Рассматривается пятизвенный двуногий шагающий механизм с безмассовыми стопами в сагиттальной плоскости Рис. 1. Все звенья соединяются между собой идеальными (без трения) шарнирами. О разумности исследования такой модели говорится в работе [1]. Для описания конфигурации робота введем обобщенные координаты $q = [x, y, \theta_0, \theta_1, \theta_2, \theta_3, \theta_4]$, определенные в инерциальной системе координат XU (ось X направлена по горизонтали, ось Y направлена вертикально вверх). Обозначим x, y —

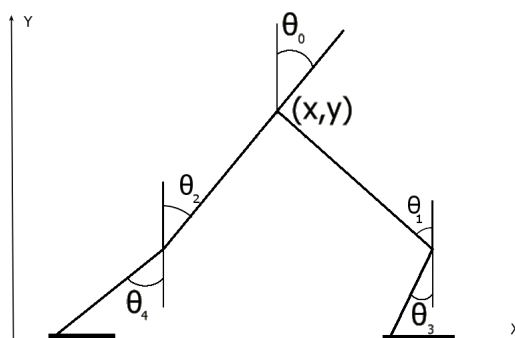


Рис. 1: Пятизвенный двуногий механизм с безмассовыми стопами

координаты тазобедренного сустава, θ_i — углы между вертикалью и звеньями. Угол является положительным, если звено отклоняется от вертикали в направлении, противоположном ходу часовой стрелки.

Ходьба двуногого шагающего механизма состоит из двух основных положений: двухопорная фаза, в которой обе ноги робота контактируют с поверхностью, и одноопорная фаза, в которой лишь одна нога на земле. Каждая из этих фаз накладывает свои ограничения на перемещение механизма.

Перемещение двуногого робота может быть представлено в виде уравнений Лагранжа второго рода

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + A(q) = Bu, \quad (1)$$

где $u = [\tau_0, \tau_1, \tau_2, \tau_3, \tau_4]$, D — матрица масс, C — матрица Кориолисовых сил, A — матрица внешних сил, B — матрица отображения u в пространство обобщенных сил Bu .

Пусть посредством датчиков, камер или других устройств робот получает карту местности, в которой он находится. В случае наличия движущейся

щихся объектов или ограничений дальности измерения приборов информация должна с некоторым шагом по времени обновляться. На карте местности представлены неровности поверхности, а также относительные или абсолютные координаты, размеры и формы препятствий. Кинематические и динамические данные рассматриваемого механизма известны.

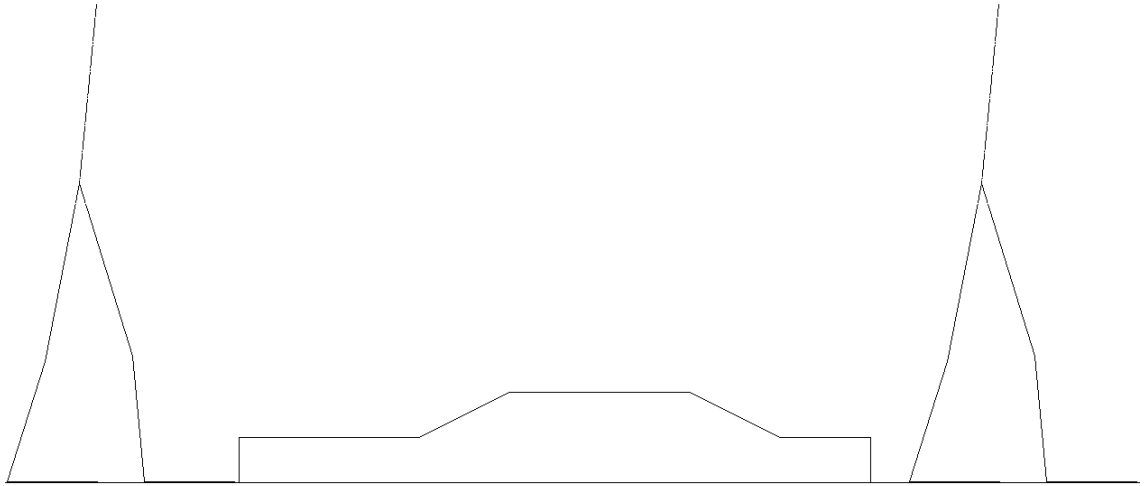


Рис. 2: Графическая формулировка задачи

Задача заключается в построении траектории перемещения робота из начального положения в заданное конечное (или его окрестность), которое может быть представлено как полным описанием конфигурации механизма Рис. 2, так и только положением его стоп или центра масс. Траектория должна быть представлена дискретным набором состояний, в которые робот способен попасть с учетом его динамики и препятствий, представленных картой местности.

Обзор литературы

В работах [1], [5], [6] подробно описана динамика двуногих механизмов и способы получения уравнений движения. Применяются различные подходы к классу управляющих моментов. К примеру, Формальский [1] предлагает использовать импульсное управление, которое прикладывается в начале и конце шага, заставляя механизм перемещаться по баллистической траектории.

Во многих трудах предлагается ходьба робота по заранее заданным траекториям (trajectory tracking). Для уменьшения отклонения механизма от заданного пути предлагается применять стабилизацию с использованием ПИД-регуляторов [2], [7].

В работах [1], [8] предлагается строить траектории механизма, исходя из желаемой траектории точки нулевого момента в поддерживающем многограннике [9]. В других трудах [10], [11] заранее выведен общий вид траекторий с неизвестными параметрами для каждой координаты механизма, и по необходимому положению точки нулевого момента определяются соответствующие компоненты в функциях. Подобные подходы обеспечивают полную управляемость робота на протяжении всего пути, но при этом походка механизма получается неестественной, а также накладываются ограничения на генерацию траекторий.

Поиск пути в трехмерном пространстве рассматривается в трудах [12], [13], [14]. Обход препятствий сводится к построению траекторий положения стоп с помощью вероятностных алгоритмов (A^* , Probabilistic roadmap), potential fields, а также обучения с подкреплением. Плюсы подобных методов в том, что они могут учитывать неровности поверхности, а их скорость работы позволяет использовать поиск пути многократно, уменьшая область, что может способствовать обходу движущихся объектов.

Большой вклад в планирование движения внесли LaValle и Kuffner с публикацией нового вероятностного алгоритма [3], [4], основанном на заполнении пространства быстро расширяющимися деревьями. Их работы уже нашли применение в планировании пути автомобилей и летающих объектов: вертолетов, мультикоптеров, космических летательных аппаратов. Во второй главе будет подробнее рассмотрен этот алгоритм, а также представлена его адаптация для построения траекторий ходьбы робота.

Глава 1. Управление

В качестве управления механизмом будет рассмотрен метод, основанный на следовании заранее заданным траекториям и обратной задаче динамики [15]. Подобные алгоритмы с различными улучшениями подробнее рассматриваются в трудах [1], [2], [6].

1.1. Опорные траектории

Пусть в начальный момент времени $t = 0$ механизм находится в конфигурации $q(0) = [x^{(0)}, y^{(0)}, \theta_0^{(0)}, \theta_1^{(0)}, \theta_2^{(0)}, \theta_3^{(0)}, \theta_4^{(0)}]^T$ в неподвижном состоянии $\dot{q}(0) = 0$ Рис. 3. Зафиксируем время шага T . Сделав шаг, робот окажется в положении $q(T) = [x^{(1)}, y^{(1)}, \theta_0^{(1)}, \theta_1^{(1)}, \theta_2^{(1)}, \theta_3^{(1)}, \theta_4^{(1)}]^T$, подобном начальному со сменой ног и перемещением горизонтальной координаты таза. Скорости при $t = T$ также потребуем нулевые, т. е. $\dot{q}(T) = 0$.

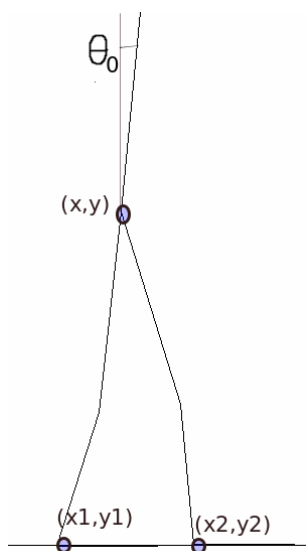


Рис. 3: Начальное положение

Положим $x1, y1$ — координаты голеностопного сустава переносимой ноги, тогда $x1^{(0)}, y1^{(0)}$ и $x1^{(1)}, y1^{(1)}$ — положения при $t = 0$ и $t = T$ соответственно, определяемые из $q(0)$ и $q(T)$. Зададим дважды непрерывно дифференцируемые функции $x1(t), y1(t)$, удовлетворяющие граничным условиям: $x1(0) = x1^{(0)}, y1(0) = y1^{(0)}, x1(T) = x1^{(1)}, y1(T) = y1^{(1)}$, причем $\dot{x}1(0) = \dot{y}1(0) = \dot{x}1(T) = \dot{y}1(T) = 0$. Аналогично проделаем для компонентов координаты таза $x(t), y(t)$ и угла торса $\theta_0(t)$. Так как у опорной ноги положение голеностопного сустава фиксированное (скольжение не допускается), то функции углов в коленях $\theta_1(t), \theta_2(t)$ и углов в голеностопных

суставах $\theta_3(t), \theta_4(t)$ можно вычислить через координаты таза и координаты голеностопных суставов соответствующих ног с учетом ограничений Рис. 4.

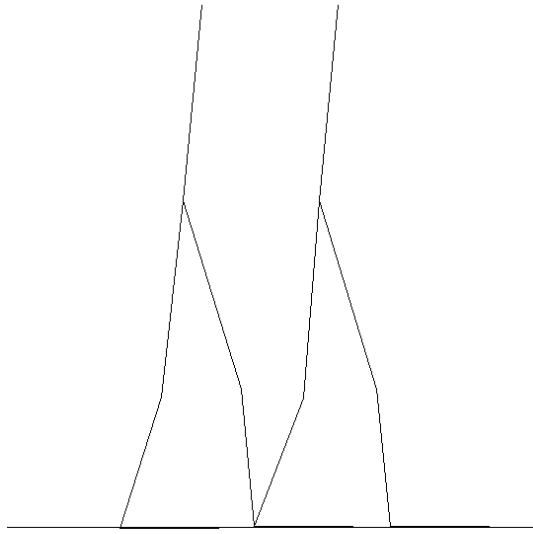


Рис. 4: Шаг

Получаем функцию для всего вектора обобщенных координат $q(t)$.

1.2. Управляющие моменты

В рассматриваемом пятизвеннике с безмассовыми стопами моторы расположены у таза, в коленях, а также в голеностопных суставах Рис 5. Компоненты вектора u представляют собой управляющие моменты, которые соответствующие моторы производят.

Пусть задана опорная траектория $q_{ref}(t)$. Определим постоянный шаг интегрирования уравнений движения Δt . Необходимо вычислить вектор управляющих моментов u , позволяющий роботу перейти из конфигурации $q(t_0)$ в $q(t_0 + \Delta t)$, соответствующее точке на следуемой траектории $q_{ref}(t_0)$. Для удобства назовем механизм с координатами $q_{ref}(t)$ *виртуальным*, а работа с $q(t)$, найденными из уравнений Лагранжа, — *реальным*.

Сначала необходимо вычислить вектора координат, скоростей и ускорений виртуального механизма $q_{ref}(t_0 + \Delta t), \dot{q}_{ref}(t_0 + \Delta t), \ddot{q}_{ref}(t_0 + \Delta t)$. По условию $q_{ref}(t)$ дважды непрерывно дифференцируема, поэтому $\dot{q}_{ref}(t_0 + \Delta t), \ddot{q}_{ref}(t_0 + \Delta t)$ существуют и конечны.

Выражаем координаты центра масс, через которые находится сила

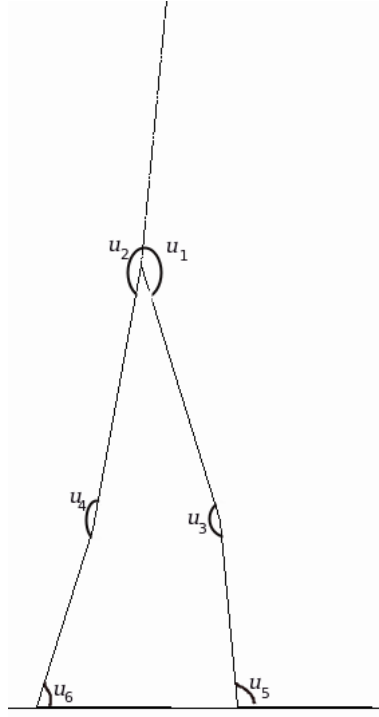


Рис. 5: Расположение моторов

реакции опоры

$$x_c = x - \frac{1}{M} (m_t r_t \sin \theta_0 + (m_h r_h + m_a L_h) (\sin \theta_1 + \sin \theta_2) + m_a r_a (\sin \theta_3 + \sin \theta_4)), \quad (2)$$

$$y_c = y + \frac{1}{M} (m_t r_t \cos \theta_0 - (m_h r_h + m_a L_h) (\cos \theta_1 + \cos \theta_2) - m_a r_a (\cos \theta_3 + \cos \theta_4)). \quad (3)$$

Здесь M — полная масса тела, m_t — масса корпуса, m_h — масса бедра, m_a — масса голени, r_t — расстояние от таза до центра масс корпуса, r_h — расстояние от таза до центра масс бедра, r_a — расстояние от колена до центра масс голени, L_h — длина бедра.

Определяем силу реакции опоры в опорной ноге

$$R_{1x} = M \ddot{x}_c, \quad R_{1y} = M \ddot{y}_c + Mg. \quad (4)$$

Подставим найденные вектора виртуального механизма в уравнения движения (1)

$$D(q_{ref}) \ddot{q}_{ref} + C(q_{ref}, \dot{q}_{ref}) \dot{q}_{ref} + A(q_{ref}) = Bu$$

И выразим вектор управления

$$u = B^+(D(q_{ref})\ddot{q}_{ref} + C(q_{ref}, \dot{q}_{ref})\dot{q}_{ref} + A(q_{ref})), \quad (5)$$

где $\{\}^+$ — псевдообращение.

1.3. Интегрирование уравнений движения

В идеальном случае найденное в предыдущем параграфе управление (5) переведет реальный механизм из состояния $q(t_0)$ в $q(t_0 + \Delta t) = q_{ref}(t_0 + \Delta t)$ с аналогичными скоростями и ускорениями Рис. 6. К сожалению, на практике неизбежно выявляются несоответствия, связанные с погрешностями в вычислении псевдообратной матрицы, производных, положений реального механизма. Поэтому авторы трудов [2], [6] уделяют внимание вопросу интегрирования уравнений движения, а также стабилизации механизма [7].

Осуществим переход в пространство состояний [6]

$$x = \begin{pmatrix} q \\ \dot{q} \end{pmatrix}$$

Тогда система (1) примет вид

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} &= \begin{pmatrix} \dot{q} \\ -D(q)^{-1}C(q, \dot{q})\dot{q} - D(q)^{-1}A(q) + D(q)^{-1}Bu \end{pmatrix} \\ \dot{x} &= f(x, u) \end{aligned} \quad (3)$$

Получили систему дифференциальных уравнений первого рода (3). Подставив вычисленный в (5) вектор управляющих моментов, интегрируем уравнения и находим $q(t_0 + \Delta t)$, $\dot{q}(t_0 + \Delta t)$.

Одна из часто возникающих проблем на этом этапе - перемещение опорной стопы, появляющееся в результате погрешностей интегрирования. Подобную проблему предлагается решать добавлением неголономной связи на опорную ногу [1], [2]. Например, для рассматриваемого механизма они примут вид

$$\begin{aligned} x - x_f + \sin(\theta_5)L_a + \sin(\theta_3)L_h &= 0, \\ y - y_f - \cos(\theta_5)L_a - \cos(\theta_3)L_h &= 0, \end{aligned}$$

$$\dot{x} + \cos(\theta_5)\dot{\theta}_5 L_a + \cos(\theta_3)\dot{\theta}_3 L_h = 0,$$

$$\dot{y} + \sin(\theta_5)\dot{\theta}_5 L_a + \sin(\theta_3)\dot{\theta}_3 L_h = 0,$$

где x, y, \dot{x}, \dot{y} — координаты и скорости таза, x_f, y_f — координаты голеностопа опорной ноги, θ_3, θ_5 — углы в бедре и колени опорной ноги, L_h, L_a — длины бедра и голени.

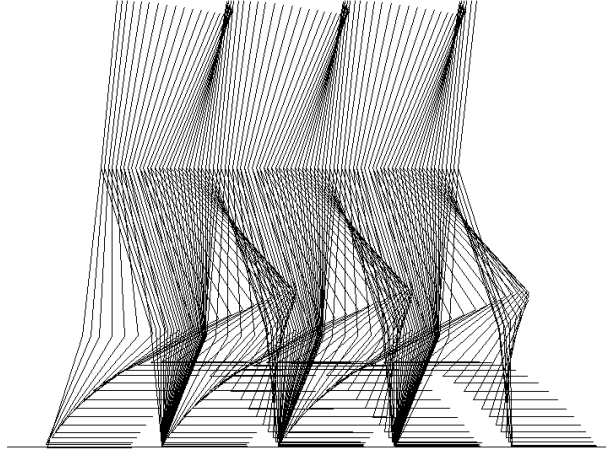


Рис. 6: Походка на основе опорных траекторий

Используя найденные координаты и скорости реального механизма, вычислим ускорения с помощью уравнений движения (1)

$$\ddot{q} = D(q)^{-1}(Bu - C(q, \dot{q})\dot{q} - A(q))$$

1.4. Пример

В качестве примера возьмем простой механизм: центр масс каждого звена расположен в центре звена, стопы не имеют массы. Определим шаг интегрирования Δt и конечное время T . Пусть h, d — высота и длина шага, соответственно.

В качестве траектории для горизонтальной компоненты таза возьмем:

$$x = x(0) + \frac{d}{T}t$$

.

Вертикальная компонента y пусть остается постоянной, т.е. $y = y(0)$.

Голеностоп x_f, y_f предлагается переносить по параболе:

$$x_f = x_f(0) + d(1 - \cos(\frac{\pi}{T}t)),$$

$$y_f = y_f(0) + h(1 - \cos^2(\frac{\pi}{T}t))$$

С помощью угла наклона торса регулируется положение центра масс. При подъеме ноги торс желательно наклонять вперед, а при опускании ноги — возвращать в изначальное положение.

$$\theta_0 = \theta_0(0) - K \sin(\frac{\pi}{T}t),$$

где K — величина отклонения.

На Рис. 7 представлен шаг с использованием подобных опорных траекторий.

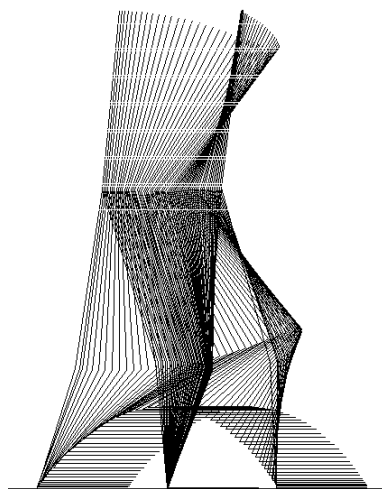


Рис. 7: Шаг реального механизма

Глава 2. Построение траекторий

В прошлой главе был рассмотрен метод управления на основе следования по заранее заданным траекториям. При наличии полной информации о строении и конфигурации робота можно представить в виде непрерывных периодических функций перемещение двуногого механизма по ровной поверхности. Однако, подобный подход не практичен, так как накладывает сильные ограничения на наличие неровностей и препятствий. Поэтому ставится не менее важная задача автоматизации построения траекторий на основе динамики механизма и карты местности. В качестве решения предложен метод, использующий алгоритм построения быстро расширяющихся деревьев в пространстве конфигураций робота (Rapidly exploring random tree, RRT) [3].

2.1. Rapidly Exploring Random Tree

Чтобы понять суть алгоритма RRT, рассмотрим конфигурации q_0 в конфигурационном пространстве Q . Необходимо построить дерево $G = (V, E)$, выходящее из данной начальной точки и заполняющее пространство, огибая все препятствия. На каждом шаге алгоритма случайным образом генерируется новая конфигурация $q_{rand} \in Q$. Далее, находится ближайшая к ней вершина дерева $q_{near} \in V$ и вычисляется конфигурация q_{new} , находящаяся на заранее заданном расстоянии Δq от q_{near} в сторону q_{rand} . Если $q_{new} \in Q$, то она добавляется в граф. K ограничивает число итераций. Чем больше K , тем шире дерево, и больше заполняется данное пространство.

Алгоритм RRT:

```
V.add( $q_0$ )
for k=1:K
     $q_{rand} \leftarrow Rand(Q)$ 
     $q_{near} \leftarrow Nearest(q_{rand}, V)$ 
     $q_{new} \leftarrow New(q_{near}, q_{rand}, \Delta q)$ 
    V.add( $q_{new}$ )
    E.add( $q_{near}, q_{new}$ )
return  $G=(V,E)$ 
```

Пример работы алгоритма на плоскости представлен на Рис. 8-12.

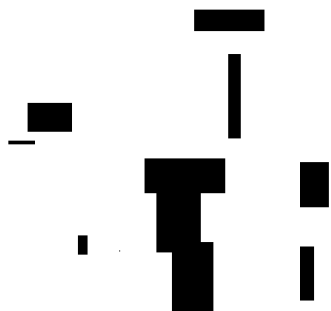


Рис. 8: Шаг 0



Рис. 9: Шаг 1000

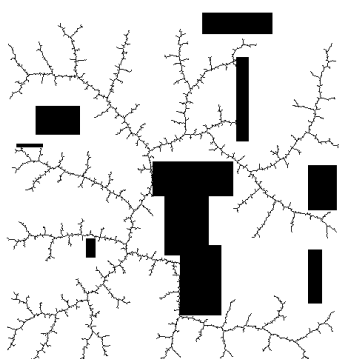


Рис. 10: Шаг 5000

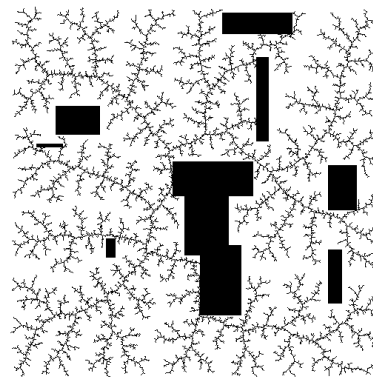


Рис. 11: Шаг 20000

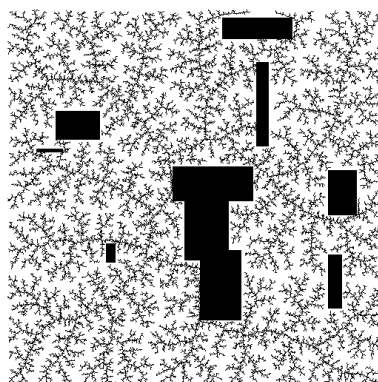


Рис. 12: Шаг 50000

2.2. Адаптация алгоритма RRT

2.2.1. Начальные данные

Входными параметрами являются конфигурация механизма, карта поверхности, конечное положение (также может быть представлено через число шагов или время конца интегрирования), частота дискретизации по времени Δt и по расстоянию Δq . Карта представляет собой набор препятствий и неровностей, которые полностью или частично должны быть геометрически описаны перед использованием алгоритма. По карте определяется примерное положение голеностопа переносимой ноги по окончании шага. Алгоритм RRT используется для построения траектории одного шага. Рис 13. При наличии движущихся препятствий необходимо использовать алгоритм поиска пути чаще.

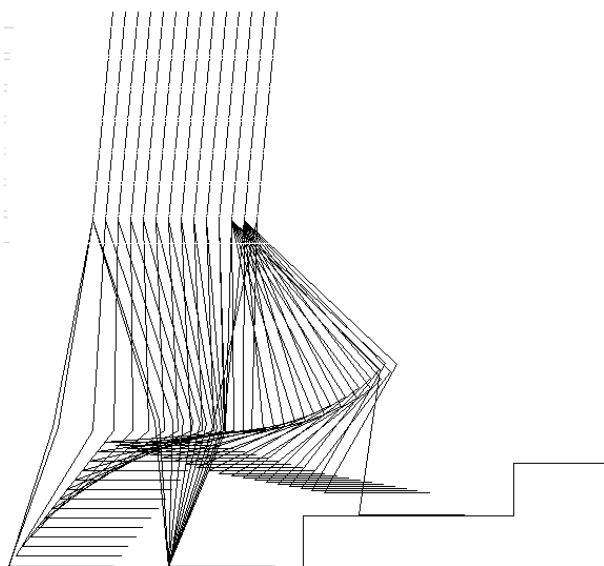


Рис. 13: Ступенчатая поверхность

2.2.2. Генерация величин

Скорость и результат работы алгоритма во многом зависит от того, как будет выбрано конфигурационное пространство. Для рассматриваемой модели плоского пятизвенного механизма с безмассовыми стопами конфигурация представляет собой положение механизма, заданное координатами таза (x, y) , голеностопных суставов (x_{f1}, y_{f1}) , (x_{f2}, y_{f2}) , а также углом наклона торса θ_0 . Так как во время шага ступня опорной ноги остается

неподвижной, генерируются лишь пять величин: x, y, x_f, y_f, θ_0 , где (x_f, y_f) — координаты голеностопа переносимой ноги Рис. 14.

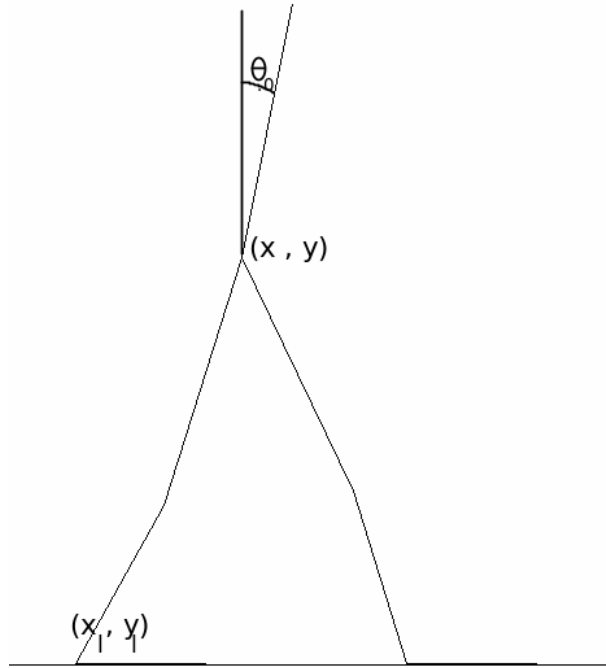


Рис. 14: Генерируемые величины

Во время поднятия ноги желательно более частое выпадание случайных координат голеностопа y_f выше, чем его положение в начальный момент. Аналогичная работа с перемещением таза и с покачиванием торса на траектории. Поэтому шаг разбивается на два этапа: поднятия и опускания ноги, используя RRT для каждого случая отдельно Рис. 15.

Вместо изменения конфигурационного пространства можно определить по какому закону будет генерироваться случайная величина.

2.2.3. Добавление вершины

После того как q_{rand} сгенерировано, необходимо отыскать ближайшую к ней вершину дерева. При определении функции расстояния надо учитывать, что угол наклона торса, а также координаты таза и ног вычисляются в разных величинах.

После нахождения q_{near} , используя Δq , вычисляется новая потенциальная вершина графа. Δq может представлять собой вектор, компоненты которого являются изменением угла торса $\Delta\theta_0$, координат таза Δd и переносимой ноги Δd_f Рис. 16,17.

Проверка новой вершины q_{new} проходит в два этапа: соответствие но-

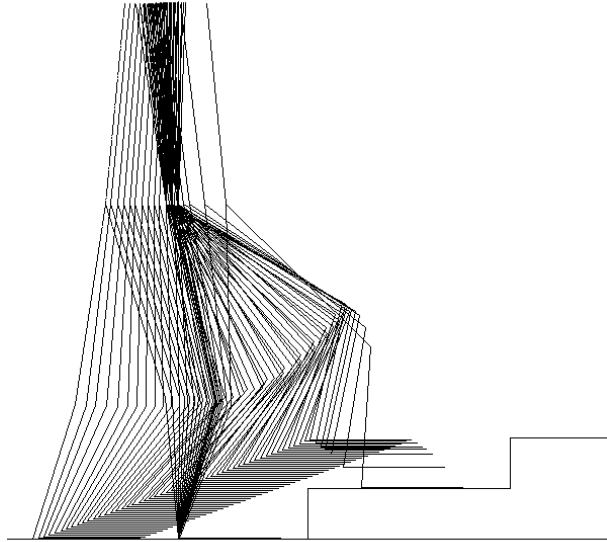


Рис. 15: Двухэтапный шаг

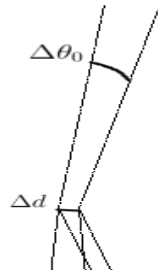


Рис. 16: Изменение положения торса и таза

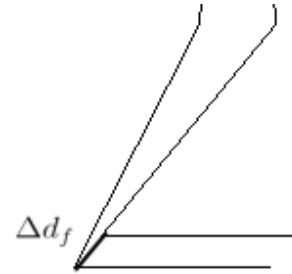


Рис. 17: Изменение положения голенистопа

вой конфигурации параметрам механизма и пересечение препятствий. Если расстояние между тазом и переносимой ногой, а также между тазом и опорной ногой меньше либо равно сумме длин бедра и голени, то есть для пятизвенника

$$(x - x_{f1})^2 + (y - y_{f1})^2 \leq L_a^2 + L_h^2,$$

$$(x - x_{f2})^2 + (y - y_{f2})^2 \leq L_a^2 + L_h^2,$$

то q_{new} можно рассматривать дальше. Также можно определить максимальный угол наклона торса $\tilde{\theta}_0$ и ввести проверку:

$$|\theta_0| < \tilde{\theta}_0$$

Второй этап проверки более затратный. Как говорилось ранее, карта хранит информацию о неровностях поверхности и препятствиях на пути механизма. Препятствия лучше хранить в порядке близости к механизму, дабы исключить лишние проверки. На данном этапе проверяется пересе-

чение звенами робота препятствий и "проваливание" ноги. В качестве проверки на пересечение отрезков или фигур можно использовать алгоритм, основанный на векторном произведении, приведенный в книге [16] .

2.2.4. Аппроксимация

Траектории строятся как последовательность узлов дерева. Они должны представлять собой дважды непрерывно дифференцируемые функции. Поэтому для построения функций, проходящих через найденные конфигурации, подходят кубические сплайны с ограничением на то, что ускорения в начальной и конечной точках нулевые.

$$S_i(t) = a_i + b_i(t - t_i) + \frac{c_i}{2}(t - t_i)^2 + \frac{d_i}{6}(t - t_i)^3, t \in [t_{i-1}, t_i[$$

Кусочно-постоянный рывок на таких траекториях обеспечит плавность ускорений, что уменьшит нагрузку на моторы. По заданному Δt можно варьировать скорости и ускорения каждой генерированной координаты. В результате получаем дважды непрерывно дифференцируемые траектории, которые можно использовать для нахождения управления по формулам (2)-(5) из предыдущей главы.

Глава 3. Эксперименты

Для проверки алгоритма разработана расчетная программа, принимающая на вход параметры механизма (начальное положение, размеры и массы звеньев), описание карты местности в виде координаты препятствий и неровностей, максимальное число итераций RRT, координаты конечной точки или количество шагов, а также шаг интегрирования уравнений движения.

3.1. Описание работы расчетной программы

После ввода данных, если Δq явно не задан, то вычисляется, исходя из длины звеньев механизма. Для плотного заполнения конфигурационного пространства узлами дерева можно взять $\Delta d_f = 0.05\sqrt{(L_a^2 + L_h^2)}$ и аналогично для других дискретизаций. Далее, запускается поиск промежуточных позиций стоп с учетом местности. То есть определяется примерная длина одного шага (расстояние между стопами переносимой ноги в начальный и конечный моменты шага) на основе размеров стоп и длин ног. Основываясь на наличии и форме препятствий в пределах этого шага, определяется высота постановки стопы. Вычисленная точка добавляется в вектор промежуточных позиций.

Далее, для каждого шага запускается алгоритм построения траекторий. Из вектора промежуточных позиций берется конечная точка для текущего шага. Если вершина дерева траекторий стопы попадет в окрестность этой точки, то дискретная траектория построена. При запуске RRT массив вершин V добавляется начальная конфигурация механизма, переведенная из обобщенных координат в генерируемые. В процессе выполнения алгоритма вычисляется q_{rand} относительно начального состояния робота, находится ближайшая к нему вершина $q_{near} \in V$, и, используя Δq определяется q_{new} . Затем новая конфигурация проходит проверки на соответствие параметрам механизма и пересечение препятствий. В случае успешного выполнения этих условий, q_{new} добавляется в V , а расстояние между ней и ближайшей вершиной — в матрицу смежности E .

После добавления q_{new} запускается еще одна проверка на близость стопы в этой конфигурации с конечной точкой шага. Задачей является попадание в окрестность конечного положения стопы размера Δd_f . Ес-

Параметр	Торс	Бедро	Голень	Стопа
Масса, кг	20	15	10	0
Длина, см	50	50	35	25
Расстояние до центра масс, см	25	25	17.25	0

Таблица 1: Характеристики механизма.

ли стопа находится в пределах этой окрестности, то последнюю вершину можно достроить опустив в конечную точку, предварительно проверив эту конфигурацию как q_{new} .

Набор узлов ветви построенного дерева, приводящей из начальной конфигурации в конечную, интерполируется кубическим сплайном. Полученные дважды непрерывно дифференцируемые функции подаются методам, отвечающим за следованию опорным траекториям. В каждый момент времени вычисляются значения обобщенных координат, а также их первые и вторые производные. По ним определяются текущая силы реакции опоры и управляющие моменты, используя формулы (2)-(5) из первой главы. Далее, при подставлении найденного управления проходит интегрирование уравнений движения неявным методом Эйлера с учетом дополнительных связей на координаты голеностопа опорной ноги.

3.2. Пример работы расчетной программы

В расчетную программу подается описание механизма из таблицы 1. Установки метода представлены в таблице 2. Координата y таза остается постоянной. На пути робота встречаются ступенчатые неровности, на которые он должен забраться.

Высота первой ступеньки равна 12.5см, второй — 25см относительно

Параметр	Значение
Максимальное количество итераций	10000
Число шагов	N
Δd , см	1.25
$\Delta \theta_0$, рад	0.005
Δd_f , см	2.5

Таблица 2: Установки алгоритма.

поверхности. Их длина равны 50см. На Рис. 18-22 представлены различные результаты работы расчетной программы для заданной местности при разном числе шагов N . Как видно на кинограммах, при одних и тех же остановках алгоритм выдает весьма отличающиеся траектории.

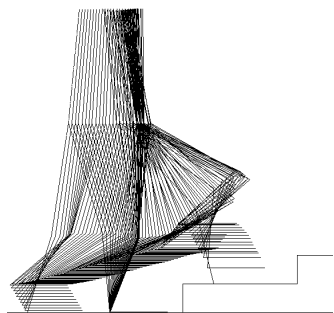


Рис. 18: $N=1$

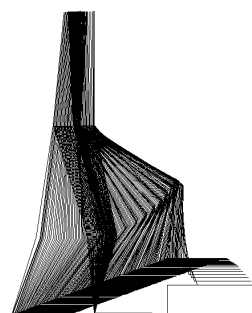


Рис. 19: $N=1$

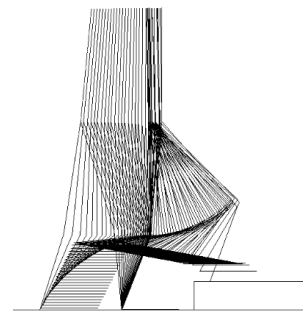


Рис. 20: $N=1$

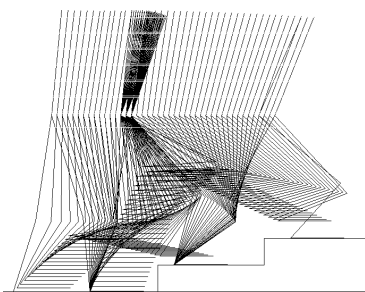


Рис. 21: $N=2$

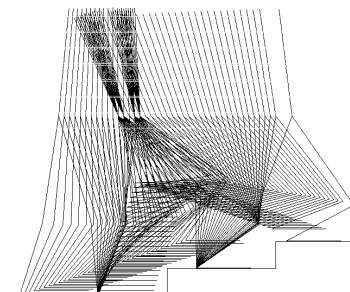


Рис. 22: $N=2$

Заключение

В данной работе была рассмотрена задача построения траекторий плоского пятизвенного механизма для его перемещения по известной горизонтальной поверхности. Для решения этой задачи адаптирован и реализован вероятностный алгоритм RRT (rapidly exploring random trees, быстрорасширяющиеся поисковые деревья). Работоспособность и практическая применимость реализованного метода продемонстрирована численными примерами.

Список литературы

- [1] Формальский А. М. Перемещение антропоморфных механизмов. М.: Наука, 1982. 368 с.
- [2] Braun D. J., Goldfarb M. A control approach for actuated dynamic walking in biped robots // IEEE Transactions on Robotics, DOI: 10.1109/TRO.2009.2028762
- [3] LaValle S. M. Rapidly-exploring random trees: A new tool for path planning // Technical Report (Computer Science Department, Iowa State University), 1998.
- [4] LaValle S. M., Kuffner J. J. Randomized kinodynamic planning // International Journal of Robotics Research. 2001. Vol. 20, No 5. P. 378–400.
- [5] Белецкий В. В. Двуногая ходьба: модельные задачи динамики и управления. М.: Наука, 1984. 288 с.
- [6] Westervelt E. R., Grizzle J. W., Chevallereau C., Choi J. H., Morris B., Feedback Control of Dynamic Bipedal Robot Locomotion. USA: Taylor and Francis Group, LLC, 2007. 505 p.
- [7] Lewis F. L., Dawson D. M., Abdallah C. T. Robot Manipulator Control – Theory and Practice // Marcel Dekker, 2004. P. 185–197.
- [8] Kagami S., Kitagawa T., Nishiwaki K., Sugihara T., Inaba M., Inoue H., A fast dynamically equilibrated walking trajectory generation method of humanoid robot, // Autonomous Robots. 2002. Vol. 12, No 1. P. 71–82.
- [9] Vukobratović M., Borovac B. Zero-Moment Point — thirty five years of life // International Journal of Humanoid Robotics. 2004. Vol. 1, No 1. P. 157–173.
- [10] Huang Q., Yokoi K., Kajita S., Kaneko K., Arai H., Koyachi N., Tanie K. Planning walking pattern for a biped robot, // IEEE Transactions on Robotics, 2001. Vol. 17, No 3, P. 280–289.
- [11] Esfahani E. T., Elahinia M. H. Stable Walking Pattern for an SMA-Actuated Biped // IEEE/ASME Transactions on Mechatronics, DOI: 10.1109/TMECH.2007.905707

- [12] Bourgeot J. M., Cislo N., Espiau B. Path-planning and tracking in a 3D complex environment for an anthropomorphic biped robot // Intelligent Robots and Systems, DOI: 10.1109/IRDS.2002.1041646
- [13] Kulkarni P., Goswami D., Guha P., Dutta A. Path planning for a statically stable biped robot using PRM and Reinforcement learning // Journal of Intelligent and Robotic Systems. 2006. Vol. 47, P. 197–214.
- [14] Kuffner J. J., Nishiwaki K., Kagami S., Inaba M., Inoue H. Footstep planning among obstacles for biped robots // Intelligent Robots and Systems, DOI: 10.1109/IROS.2001.973406
- [15] Бубнов В. В, Белошапко А. Г. Управление в задачах перемещения двуногoго механизма // Процессы управления и устойчивость. 2015. Т. 2. № 1. С. 119-124
- [16] Кормен Т. Х, Лейзерсон Ч. И, Ривест Р. Л, Штайн К. Алгоритмы: построение и анализ, 3-е издание. М.: «Вильямс», 2013, 1328 с.