

УДК 004.942

Управление виртуальной моделью мобильного робота с адаптацией на основе информации о виртуальной среде

Е. В. Страшнов¹, Л. А. Финагин²¹НИЦ «Курчатовский институт» - НИИСИ, Москва, Россия, strashnov_evg@mail.ru;²НИЦ «Курчатовский институт» - НИИСИ, Москва, Россия, antifin@mail.ru

Аннотация. В работе рассматривается задача разработки методов и алгоритмов управления мобильными роботами с адаптацией на основе информации о виртуальной среде. Реализация такого подхода приводится на примере движения виртуальной модели робота в объезд препятствий и захвата виртуального объекта его манипулятором. Предлагаемое решение включает применение датчиков сенсорного осязания, инверсной кинематики и ПД-регуляторов. Управление движением моделью виртуального робота осуществляется на основе показаний виртуальных датчиков положения и дальномеров, установленных на этой модели робота. В свою очередь, захват виртуального объекта манипулятором робота реализуется с использованием показаний датчиков положения, ориентации и силы, а также с применением метода поординатного спуска для решения задачи инверсной кинематики и ПД-регуляторов для обеспечения требуемого движения рабочего органа манипулятора. Апробация предлагаемых методов и алгоритмов была проведена в системе виртуального окружения VirSim, разработанной в НИЦ «Курчатовский институт» - НИИСИ, на примере управления виртуальной моделью гусеничного робота МФ-4.

Ключевые слова: мобильный робот, датчики, инверсная кинематика, метод поординатного спуска, ПД-регуляторы, системы виртуального окружения

1. Введение

Одно из направлений в робототехнике связано с выполнением работ в экстремальных условиях, опасных или вредных для непосредственного участия человека. Для этих целей могут применяться мобильные роботы колесного или гусеничного типа, оснащенные манипулятором для захвата и переноса объектов. Управление такими роботами может осуществляться человеком-оператором с помощью пульта или автоматически на основе заранее подготовленных программ. Особый интерес представляют системы управления [1, 2], в которых действия робота определяются на основе обратной связи с окружающей средой, которая может быть частично или полностью недетерминированной. Реализация такой обратной связи осуществляется с применением датчиков сенсорного осязания роботов, которые измеряют необходимую информацию о внешней среде. К основным датчикам относятся инерциальные датчики для вычисления пространственного положения роботов (гироскопы, акселерометры и магнитометры), датчики силомоментного осязания для измерения контактного взаимодействия робота с объектами

и дальномеры для определения расстояний до объектов. Создание систем управления роботами с применением обратной связи на основе показаний датчиков является важной и актуальной задачей.

Тестирование разрабатываемых методов и алгоритмов управления роботами актуально проводить на виртуальных моделях роботов в виртуальной среде. Это позволяет облегчить процесс создания и тестирования систем управления и в дальнейшем уменьшить риск поломки реальных роботов. В данной работе предлагаются методы и алгоритмы управления мобильными роботами, основанные на использовании показаний виртуальных датчиков [3, 4]. Моделирование движения мобильных роботов реализовано в системе виртуального окружения VirSim [5], разработанной в НИЦ «Курчатовский институт» - НИИСИ. В этом программном комплексе управление роботами осуществляется путем расчета функциональных схем, которые создаются в специальном редакторе [6]. Эти схемы состоят из набора связанных блоков различного типа (логические, математические и т.д.), включая блоки датчиков и исполнительных устройств. Апробация предлагаемых в статье методов и алгоритмов

проводилась в VirSim на примере движения виртуальной модели гусеничного робота МФ-4 в заданную позицию в объезд препятствий, перемещения рабочего органа его манипулятора в требуемое положение с заданной ориентацией и захвата объекта с его удержанием.

2. Движение виртуальной модели мобильного робота к цели в объезд препятствий

Рассматривается задача движения мобильного робота в горизонтальной плоскости при наличии препятствий. При этом предполагается, что препятствия на пути к цели являются выпуклыми и их можно объехать. Положение робота задается в мировой системе координат с помощью точки $P = (X, Y)$, а ориентация – направлением его продольной оси, задаваемым единичным вектором $\mathbf{n} = (n_x, n_y)^T$. Пусть известно текущее положение робота $P_c = (X_c, Y_c)$ с направлением продольной оси, задаваемым единичным вектором $\mathbf{n}_c = (n_{cx}, n_{cy})^T$. Тогда данная задача заключается в том, что требуется обеспечить движение робота в объезд препятствий в заданную целевую точку $P_d = (X_d, Y_d)$ и развернуть его по направлению $\mathbf{n}_d = (n_{dx}, n_{dy})^T$.

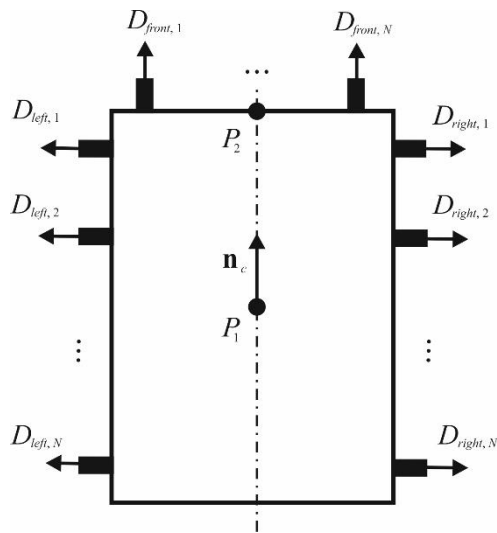


Рис. 1. Схематичное изображение датчиков на роботе

Для решения этой задачи предлагается использовать показания датчиков положения и дальномеров, установленных на роботе (см. рис. 1). Пусть один датчик положения расположен в центре робота и вычисляет координаты точки $P_1 = (X_1, Y_1)$,

а другой – на передней части робота и определяет координаты точки $P_2 = (X_2, Y_2)$. Тогда положение и вектор продольной оси робота вычисляются как

$$P_c = P_1, \quad \mathbf{n}_c = \overrightarrow{P_1 P_2} / \|\overrightarrow{P_1 P_2}\|. \quad (1)$$

Пусть дальномеры располагаются на левой, передней и правой стороне робота и измеряют расстояния до ближайшего объекта вдоль направления, перпендикулярного стороне робота. В этом случае расстояние до объекта выбирается как минимальное значение из показаний дальномеров по каждой стороне робота:

$$\begin{aligned} D_{left} &= \min(D_{left,1}, D_{left,2}, \dots, D_{left,N}); \\ D_{front} &= \min(D_{front,1}, D_{front,2}, \dots, D_{front,N}); \\ D_{right} &= \min(D_{right,1}, D_{right,2}, \dots, D_{right,N}). \end{aligned} \quad (2)$$

Управление движением мобильного робота осуществляется путем подачи напряжений на электродвигатели его ведущих колес. Для простоты будем считать, что у робота всего два ведущих колеса – левое и правое. Тогда на левое колесо подается напряжение U_L , а на правое колесо – напряжение U_R . Для движения робота вперед на оба колеса подаются напряжения $U_L = U_R = U_1$, для поворота по часовой стрелке – $U_L = U_2$ и $U_R = -U_2$, для поворота против часовой стрелки – $U_L = -U_2$ и $U_R = U_2$, где $U_1 > 0$ и $U_2 > 0$ – заданные значения напряжений.

Опишем сначала алгоритм движения мобильного робота в отсутствие препятствий. В этом случае выделяются три стадии движения робота: поворот по направлению к целевой точке P_d , его движение к ней и разворот по заданному направлению \mathbf{n}_d . В предлагаемом алгоритме сначала проверяется, что робот еще не достиг целевой точки, то есть выполнено условие $d = \|\overrightarrow{P_c P_d}\| > \varepsilon_1$, где $\varepsilon_1 > 0$ – заданное значение.

Тогда поворачиваем робот по направлению

$$\mathbf{n}_1 = (n_{1x}, n_{1y})^T = \overrightarrow{P_c P_d} / \|\overrightarrow{P_c P_d}\| \quad (3)$$

Для этого нужно совместить вектор оси робота \mathbf{n}_c с вектором \mathbf{n}_1 . Направление поворота зависит от величины проекции векторного произведения векторов \mathbf{n}_1 и \mathbf{n}_c на ось Z , которая перпендикулярна горизонтальной плоскости движения робота. Эта проекция вычисляется как

$$n_{1c,z} = (\mathbf{n}_1 \times \mathbf{n}_c)_z = n_{1x} n_{cy} - n_{1y} n_{cx}. \quad (4)$$

Тогда, если $n_{1c,z} > \varepsilon_2$, то поворачиваем робот по часовой стрелке, если $n_{1c,z} < -\varepsilon_2$ – против

часовой стрелки, где $\varepsilon_2 > 0$ – заданное значение. В том случае, если векторы \mathbf{n}_c и \mathbf{n}_1 – противоположно направленные, что соответствует условиям $|n_{1c,z}| \leq \varepsilon_2$ и $\mathbf{n}_1 \cdot \mathbf{n}_c \leq -\sqrt{1-\varepsilon_2^2}$, то поворачиваем робот по часовой стрелке. После выполнения поворота осуществляем движение робота вперед к целевой точке P_d до тех пор, пока не будет выполнено условие $d \leq \varepsilon_1$. Затем поворачиваем его так, чтобы совместить векторы \mathbf{n}_c и \mathbf{n}_d .

При наличии препятствий движение робота корректируется так, чтобы их объехать. Для этого используются показания дальномеров (2), для которых зададим два пороговых значения: D_{close} – препятствие находится близко, D_{near} – очень близко. Если $D_{front} \leq D_{near}$, то поворачиваем робот по часовой стрелке, пока слева не обнаружится препятствие, что соответствует условию $D_{left} \leq D_{close}$. Далее робот движется вперед до тех пор, пока слева есть препятствие и не будет выполнено $D_{left} > D_{close}$. Затем робот снова осуществляет движение по направлению к цели. При наличии следующего препятствия алгоритм объезда препятствий повторяется. Данный алгоритм работает при допущении, что препятствия по направлению к цели являются выпуклыми и расположены на таком расстоянии друг от друга, что робот в состоянии их объехать.

Приведем полный алгоритм управления движением мобильного робота.

Задано: целевая точка $P_d = (X_d, Y_d)$ и единичный вектор \mathbf{n}_d .

1. Используем показания датчиков и определяем P_c , \mathbf{n}_c , D_{left} , D_{right} и D_{front} по формулам (1) и (2).

2. Вычисляем $d = \|\overline{P_c P_d}\|$.

3. Цикл, пока $d > \varepsilon_1$:

3.1. Если нет препятствий ($D_{left} > D_{near}$, $D_{front} > D_{near}$ и $D_{right} > D_{near}$), то

3.1.1. Вычисляем вектор \mathbf{n}_1 по формуле (3).

3.1.2. Определяем $n_{1c,z}$ по формуле (4).

3.1.3. Если $n_{1c,z} > \varepsilon_2$, то $U_L = U_2$ и $U_R = -U_2$.

3.1.4. Если $n_{1c,z} < -\varepsilon_2$, то $U_L = -U_2$ и $U_R = U_2$.

3.1.5. Если $|n_{1c,z}| \leq \varepsilon_2$ и $\mathbf{n}_1 \cdot \mathbf{n}_c \leq -\sqrt{1-\varepsilon_2^2}$,

то $U_L = U_2$ и $U_R = -U_2$.

3.1.6. Если $|n_{1c,z}| \leq \varepsilon_2$ и $\mathbf{n}_1 \cdot \mathbf{n}_c \geq \sqrt{1-\varepsilon_2^2}$,

то $U_L = U_1$ и $U_R = U_1$.

3.2. Если обнаружено препятствие слева или справа от робота ($D_{left} \leq D_{near}$ или $D_{right} \leq D_{near}$), то движемся вперед $U_L = U_1$ и $U_R = U_1$.

3.3. Если препятствие находится впереди робота $D_{front} \leq D_{near}$, то поворачиваем робот по часовой стрелке:

3.3.1. Пока не будет выполнено $D_{left} \leq D_{close}$: $U_L = U_2$ и $U_R = -U_2$.

3.3.2. Пока не будет выполнено $D_{left} > D_{close}$, движемся вперед $U_L = U_1$ и $U_R = U_1$.

3.4. Выполняем пункты 1 и 2.

4. Вычисляем $n_{dc,z} = (\mathbf{n}_d \times \mathbf{n}_c)_z$ по формуле (4) подстановкой \mathbf{n}_d вместо \mathbf{n}_1 .

5. Если $n_{dc,z} > \varepsilon_2$, то $U_L = U_2$ и $U_R = -U_2$.

6. Если $n_{dc,z} < -\varepsilon_2$, то $U_L = -U_2$ и $U_R = U_2$.

7. Если $|n_{dc,z}| \leq \varepsilon_2$ и $\mathbf{n}_d \cdot \mathbf{n}_c \leq -\sqrt{1-\varepsilon_2^2}$, то $U_L = U_2$ и $U_R = -U_2$.

8. Выполняем пункт 1.

9. Повторяем пункты 4-8 до тех пор, пока не будут выполнены условия $|n_{dc,z}| \leq \varepsilon_2$ и $\mathbf{n}_d \cdot \mathbf{n}_c \geq \sqrt{1-\varepsilon_2^2}$.

3. Захват объекта манипулятором виртуальной модели робота

В этой задаче требуется захватить и перенести объект с помощью манипулятора мобильного робота. При этом предполагается, что рабочий орган (РО) манипулятора представляет собой механизм, содержащий две губки захватного устройства с плоскопараллельным движением и управляемый посредством одного двигателя [7]. Рассматриваемая задача включает в себя несколько стадий: подведение РО к объекту, захват этого объекта и его перенос в требуемую позицию. Для решения этой задачи задействованы метод расчета инверсной кинематики, ПД-регуляторы, а также показания датчиков положения, ориентации и силы. Для того, чтобы переместить и ориентировать РО в заданную систему координат необходимо решить задачу инверсной кинематики (рис. 2). Пусть целевое положение РО задается точкой

$P_d^{wo} = (X_d, Y_d, Z_d)$, а его ориентация – углами Эйлера φ_d , ψ_d и θ_d последовательных поворотов вокруг осей X , Y и Z относительно мировой системы координат, соответственно.

Управление РО робота осуществляется путем его совмещения с промежуточными системами координат $P_k X_k Y_k Z_k$, $k = \overline{1, M}$. Движение РО рассматривается вдоль прямой, соединяющей его начальное положение в точке $P_c^{wo} = (X_c, Y_c, Z_c)$ с точкой P_d^{wo} . На этой прямой выбираем промежуточные точки следующим образом

$$P_k = P_c^{wo} + k \Delta l \frac{\overline{P_c^{wo} P_d^{wo}}}{\| \overline{P_c^{wo} P_d^{wo}} \|}, \quad k = \overline{1, M}, \quad (5)$$

где M – количество точек, Δl – заданное расстояние между точками.

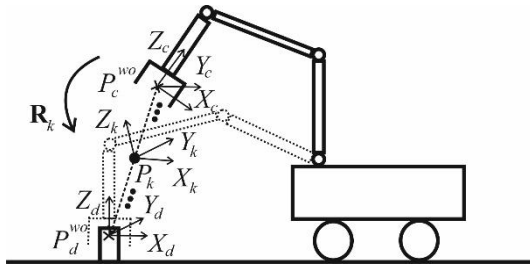


Рис. 2. Задача инверсной кинематики

Текущая ориентация РО задается системой координат $P_c^{wo} X_c Y_c Z_c$ с матрицей \mathbf{R}_c . Перевод РО в целевую систему координат $P_d^{wo} X_d Y_d Z_d$ реализуется посредством вычисления углов Эйлера φ , ψ и θ , задающих последовательные повороты сначала вокруг оси X на угол φ , затем вокруг оси Y на угол ψ и наконец вокруг оси Z на угол θ . Для задания промежуточных ориентаций РО рассмотрим матрицу

$$\mathbf{R}_k = \mathbf{R}_{Z_c}(\theta_k) \cdot \mathbf{R}_{Y_c}(\psi_k) \cdot \mathbf{R}_{X_c}(\varphi_k) \cdot \mathbf{R}_c, \quad (6)$$

$$\text{где } \varphi_k = \begin{cases} k\varphi / k_1, & k < k_1 \\ \varphi, & k_1 \leq k \leq M \end{cases},$$

$$\psi_k = \begin{cases} 0, & k < k_1 \\ (k - k_1)\psi / (k_2 - k_1), & k_1 \leq k < k_2 \\ \psi, & k_2 \leq k \leq M \end{cases}$$

$$\theta_k = \begin{cases} 0, & k < k_2 \\ (k - k_2)\theta / (M - k_2), & k_2 \leq k \leq M \end{cases},$$

$$k = \overline{1, M}, \quad 0 < k_1 < k_2 < M.$$

В задаче инверсной кинематики требуется найти такие изменения углов Δq_i , $i = \overline{1, N}$ поворотов в сочленениях робота, чтобы перевести РО робота в систему координат

с положением P_k и матрицей \mathbf{R}_k , вычисляемые по формулам (5) и (6). Для решения этой задачи нами был задействован метод покоординатного спуска [8, 9], с реализацией которого разработан алгоритм [10, 11], основанный на применении ПД-регуляторов, а также показаний датчиков положения и ориентации. В рамках предложенного подхода с помощью ПД-регуляторов осуществляется расчет напряжений U_i , подаваемых на электродвигатели в сочленениях манипулятора для обеспечения требуемых углов поворотов Δq_i , $i = \overline{1, N}$. Датчики положения и ориентации предназначены для контроля достижения РО своих целевых координат. После того, как РО переведен в промежуточную систему координат $P_k X_k Y_k Z_k$, что соответствует выполнению условий $\| \overline{P_c^{wo} P_k} \| \leq \sigma_1$, $\| \mathbf{R}_c - \mathbf{R}_k \| \leq \sigma_2$, где σ_1 и σ_2 – заданные значения точности, осуществляется переход к следующей системе координат $P_{k+1} X_{k+1} Y_{k+1} Z_{k+1}$. И этот процесс продолжается до тех пор, пока РО не достигнет целевой системы координат $P_d^{wo} X_d Y_d Z_d$.

После этого осуществляется захват объекта с подачей напряжения на двигатель захватного устройства $U_{grasp} = U_* > 0$. Для измерения силы сжатия F применяется датчик силы, установленный непосредственно на губках РО. С помощью датчика силы осуществляется контроль над выполнением операции захвата объекта. Если датчик силы показывает значение $F \geq F_*$, то считается, что захват объекта выполнен, и можно переносить объект в новое положение, применяя алгоритм управления движением мобильного робота.

Приведем полный алгоритм управления манипулятором робота.

Задано: целевое положение РО $P_d^{wo} = (X_d, Y_d, Z_d)$ и его ориентация, задаваемая углами Эйлера φ_d , ψ_d и θ_d .

1. С помощью датчиков положения и ориентации определяем текущее положение РО $P_c^{wo} = (X_c, Y_c, Z_c)$ и углы Эйлера φ , ψ и θ для перевода РО в систему координат $P_d^{wo} X_d Y_d Z_d$

2. Инициализация $k = 1$.

3. Цикл, пока $\| \overline{P_c^{wo} P_d^{wo}} \| > \sigma_1$, $\| \mathbf{R}_c - \mathbf{R}_d \| > \sigma_2$:

3.1. Вычисляем точку P_k по формуле (5) и матрицу \mathbf{R}_k по формуле (6).

3.2. Решаем задачу инверсной кинематики: вычисляем углы Δq_i в сочленениях.

3.3. Применяя ПД-регуляторы, вычисляем напряжения U_i для полученных Δq_i .

3.4. Повторяем пункт 1.

3.5. Если $\|P_c^{wo} P_k\| \leq \sigma_1$, $\|R_c - R_k\| \leq \sigma_2$, то

$k = k + 1$.

4. Осуществляем захват объекта с подачей напряжения на двигатель захватного устройства $U_{grasp} = U_* > 0$.

5. Если $F \geq F_*$, то захват объекта выполнен. Переводим РО робота в новое положение P_d^{wo} и ориентируем его новыми углами Эйлера φ_d , Ψ_d и θ_d .

4. Управление роботом в системе виртуального окружения VirSim

Апробация предложенных в статье решений была проведена в системе виртуального окружения VirSim [5] на примере управления виртуальной моделью гусеничного робота МФ-4. Данный программный комплекс разработан в НИЦ «Курчатовский институт» - НИИСИ и предназначен для моделирования движения сложных динамических систем и тренировки операторов по выполнению операций с помощью компьютерной симуляции. В рамках этого программного комплекса виртуальные модели роботов и окружающей обстановки создаются в системе компьютерного моделирования 3ds Max и затем экспортируются в подсистемы динамики и визуализации. Управление роботами в VirSim осуществляется с

помощью функциональных схем, которые создаются в специальном редакторе из набора соединенных линиями блоков различного типа, включая блоки датчиков и исполнительных устройств (двигатели манипулятора и ведущих колес робота). На каждом шаге моделирования в подсистеме управления программного комплекса VirSim выполняется расчет функциональных схем с целью вычисления управляющих сигналов – напряжений, подаваемых на электродвигатели робота. Эти управляющие сигналы используются в подсистеме динамики для расчета новых координат виртуальных объектов, включая их положение и ориентацию. Затем вычисленные координаты объектов передаются в подсистему визуализации, которая осуществляет высококачественный рендеринг виртуальной сцены в режиме реального времени.

Для мобильного робота МФ-4 была рассмотрена задача захвата и переноса радиоактивного контейнера внутри ангара, содержащем бочки, коробки, паллеты и т.д. В этой задаче робот должен подъехать к контейнеру в объезд препятствий, схватить его, затем перевезти к специальной урне и поместить его туда. В качестве пороговых значений расстояний для объезда препятствий были выбраны значения $D_{close} = 50$ см и $D_{near} = 25$ см.

Рассматриваемые задачи были реализованы с помощью разработанных в статье алгоритмов, которые представлены в виде функциональной схемы автоматического управления виртуальной моделью гусеничного робота МФ-4. На рис. 3 показан фрагмент такой схемы для управления движением робота в объезд препятствий.

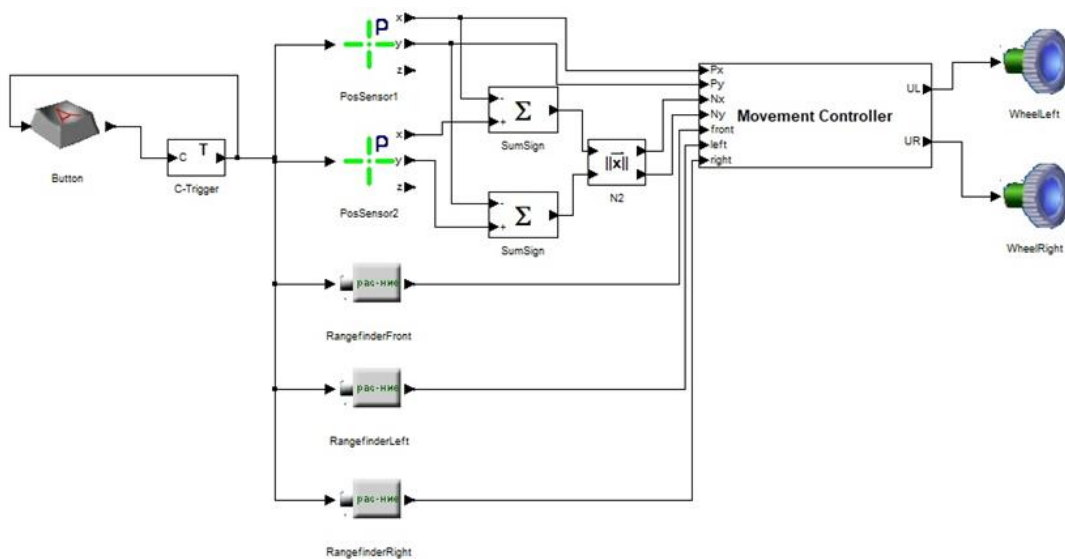


Рис. 3. Функциональная схема управления виртуальной моделью робота МФ-4

В этой схеме нажатием на кнопку виртуального пульта человек-оператор передает роботу задание, выполняемое роботом автоматически. Показания датчиков положений и дальномеров передаются в блок регулятора управления движения роботом, в котором

выбирается режим перемещения робота. Выходами этой схемы являются напряжения, подаваемые на двигатели ведущих колес робота.

На рисунках 4 и 5 показан результат выполнения задач объезда роботом препятствий и опускание радиоактивного контейнера в урну.



Рис. 4. Объезд препятствия виртуальной моделью робота МФ-4



Рис. 5. Опускание радиоактивного контейнера в урну манипулятором виртуальной модели робота МФ-4

5. Заключение

В работе предложены методы и алгоритмы управления виртуальными роботами, основанные на обратной связи с помощью датчиков сенсорного осязания. Рассмотрены задачи движения робота к цели в объезд препятствий и захват объекта с помощью манипулятора робота. В дальнейшем ожидается развитие

предложенных подходов с применением методов нечеткой логики для повышения качества движения роботов.

Работа выполнена в рамках государственного задания НИЦ «Курчатовский институт» - НИИСИ по теме № FNEF-2024-0002 «Математическое моделирование многомасштабных динамических процессов и системы виртуального окружения».

Control of a Virtual Model's Mobile Robot with Adaptation Based on Information About the Virtual Environment

E. V. Strashnov, L. A. Finagin

Abstract. This paper considers the task of methods and algorithms development for control of mobile robots with adaptation based on information about the virtual environment. The implementation of this approach is shown using the example of a virtual robot model moving around obstacles and grasping a virtual object with its manipulator. Proposed solution includes the use of sensors, inverse kinematics and PD controllers. The virtual robot model movement control is based on the virtual position sensors readings and rangefinders mounted on this robot model. In turn, the grasping of an object by a manipulator of robot is implemented using position, orientation and force sensors, as well as using the coordinate descent method to solve the inverse kinematics problem and PD-regulators to provide the desired manipulator motion. The proposed methods and algorithms were tested in the VirSim virtual environment system, developed at the NRC «Kurchatov Institute» - SRISA, using the example for control of a virtual MF-4 tracked robot model.

Keywords: mobile robot, sensors, inverse kinematics, coordinate descent method, PD-regulators, virtual environment systems

Литература

1. Е. И. Юревич Основы робототехники. 3-е изд., пер. и доп. СПб.: БХВ-Петербург, 2010, 360 с.
2. M. Spong and M. Vidyasagar. Robot dynamics and control. Hoboken, NJ, USA: Wiley, 2008.
3. Е. В. Страшнов, И. Н. Мироненко. Имитационное моделирование инерциальных датчиков в системах виртуального окружения. «Успехи кибернетики», Т. 5 (2024), № 1, 24-33.
4. М. В. Михайлюк, Е. В. Страшнов, Д. М. Логинов. Моделирование датчиков в системах виртуального окружения. «Труды НИИСИ РАН», Т. 8 (2018), № 2, 70-76.
5. М. В. Михайлюк, А. В. Мальцев, П. Ю. Тимохин, Е. В. Страшнов, Б. И. Крючков, В. М. Усов. Система виртуального окружения VirSim для имитационно-тренажерных комплексов подготовки космонавтов. «Пилотируемые полеты в космос». Т. 37 (2020), № 4, 72-95.
6. М. В. Михайлюк, М. А. Торгашев. Визуальный редактор и модуль расчета функциональных схем для имитационно-тренажерных комплексов. «Программные продукты и системы», 2014, № 4. 10-15.
7. М. А. Торгашев, Е. В. Страшнов. Моделирование динамики электроприводов виртуальных роботов в имитационно-тренажерных комплексах. «Мехатроника, автоматизация, управление», Т. 17 (2016), № 11, 762 – 768.
8. L. C. Wang, C. C. Chen. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. “IEEE Transactions on Robotics and Automation”, V. 7 (1991), no. 4, 489-499.
9. B. Kenwright. Inverse kinematics – cyclic coordinate descent (ccd). “Journal of Graphics Tools”, V. 16 (2012), no. 4, 177-217.
10. Е. В. Страшнов, М. В. Михайлюк. Моделирование полуавтоматического режима управления манипуляционными роботами в системах виртуального окружения. «Вестник кибернетики», 2017, № 4, 191-198.
11. Е. В. Страшнов, М. А. Торгашев. Супервизорное управление антропоморфными роботами с применением инверсной кинематики. «Виртуальное моделирование, прототипирование и промышленный дизайн. Материалы IV Международной научно-практической конференции», 2017, Выпуск 4, Том 1, 186-190.

Поступила в редакцию / Received: 20.03.2026.

Поступила после рецензирования / Revised: 30.03.2026.

Принята к печати / Accepted: 31.03.2026