

## МЕТОД ПОЗИЦІОНУВАННЯ РУКИ РОБОТА-МАНІПУЛЯТОРА АВТОМАТИЗОВАНОГО СКЛАДУ В ТРИВИМІРНОМУ ПРОСТОРИ

### *Анотація*

*Запропонований метод позиціонування руки робота-маніпулятора автоматизованого складу в тривимірному просторі з врахуванням технічних особливостей обчислювальної платформи Raspberry Pi 3 та виконавчих механізмів (серводвигунів), що дозволяє підвищити якість управління.*

**Ключові слова:** рука робота-маніпулятора, позиціонування в тривимірному просторі, Raspberry Pi 3, інтелектуальна система.

### *Abstract*

*The proposed method for positioning of robotic arm of automated storage in the three dimensional space with including Raspberry Pi 3 circuit features and servomotors, created positioning method allows to improve the executing quality of robotic arm.*

**Keywords:** robotic arm, positioning in the three-dimensional space, Raspberry Pi 3, intelligence system.

### **Вступ**

В умовах сьогодення розвиток промисловості в Україні пов'язаний з впровадженням сучасних систем управління на базі інтелектуальних технологій, які дозволяють суттєво підвищувати ефективність виробництва за рахунок забезпечення високої швидкодії і точності виконання технологічних операцій. Так, на підприємствах в різних галузях сучасної промисловості широко використовується рука робота-маніпулятора. Тому актуальною є задача розробки інтелектуальних методів управління рукою робота-маніпулятора.

### **Аналіз задачі**

З точки зору кінематики рука робота-маніпулятора – це послідовність з'єднаних ланок сервоприводів, які послідовно переміщуються одна відносно іншої. Однією з задач управління рукою є позиціонування. В роботі розглядається рука робота-маніпулятора, яка складається з послідовності чотирьох ланок. Задача позиціонування руки по заданим координатам зводиться до визначення початкових та кінцевих координат і розрахунку траєкторії руху, при цьому важним є вибор системи координат руки. Найчастіше використовують декартову, циліндричну та ортогональну систему координат. Кожна має певні переваги та недоліки.

Наприклад, декартова система координат, передбачає рух по лінійним, взаємо перпендикулярних осям, тобто, для руху по складних траєкторіях потрібна багатоланкова конструкція. Перевагою є простота обрахунків траєкторії за рахунок осей  $x$ ,  $y$ ,  $z$  та тільки прямого кута. Циліндрична система координат передбачає більш переміщення по більш складним траєкторіям, але координати задаються, двома відстанями та азимутом (кутом), який постійно змінюється, що значно ускладнює обрахунки траєкторії. Ортогональна система координат передбачає, що координатні поверхні перпендикулярні, а також можливість використовувати декартові чи циліндричні в якості ортогональних, що спрощує розрахунки траєкторії і дозволяє рух по траєкторіям різної складності.

Згідно з розглянутою моделлю необхідно забезпечити рух по прямій і криволінійній траєкторіям, отже вибір декартової системи координат призведе до ускладнення конструкції руки, тобто збільшення ланок. У виборі між циліндричною і ортогональною системами координати обрано – ортогональну, завдяки простоті обрахунків.

Розглянемо графічну модель руки робота-маніпулятора, що зображена на рисунку 1.

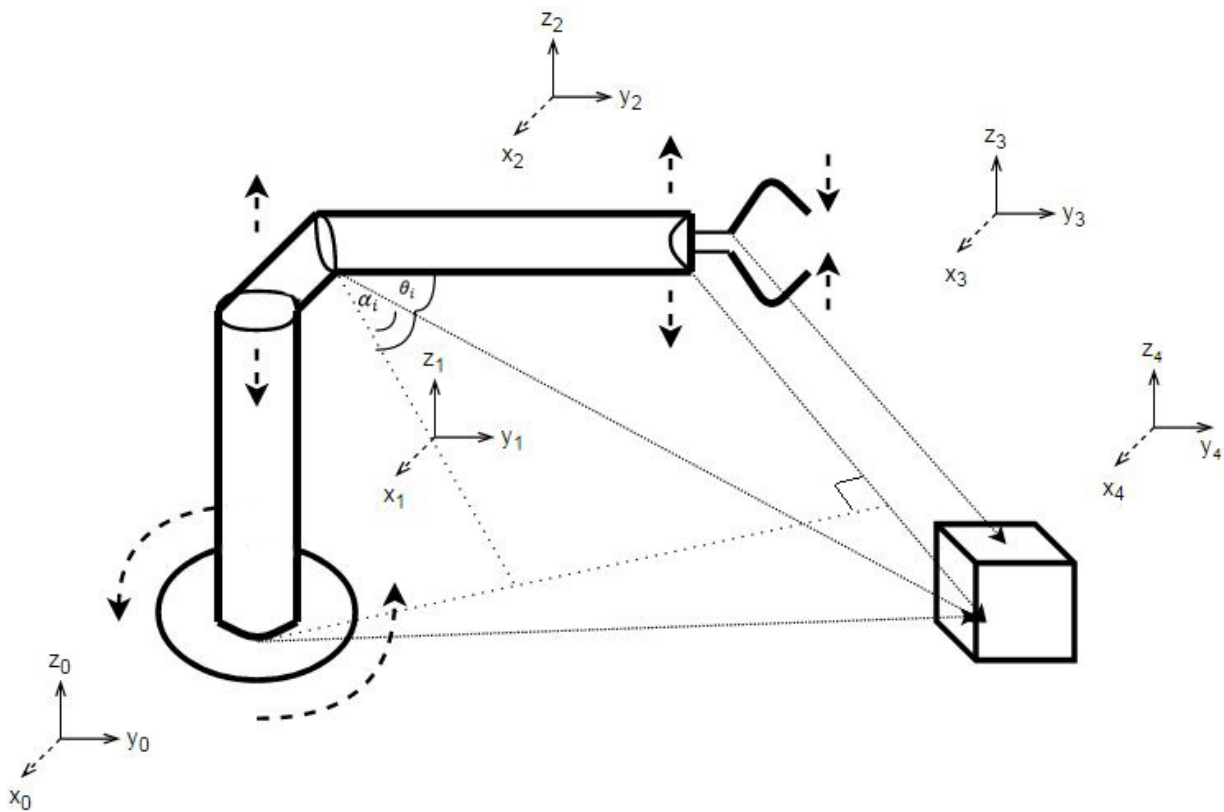


Рис. 1. Схематична зображена модель руки робота-маніпулятора

### Результати дослідження

Руку робота-маніпулятора використовують в великих галузях виробництва наприклад: гірничій справі; в металургії; в ядерній техніці, але виготовлення подібних рук робота-маніпулятора коштує багато грошей та займає великий проміжок на виготовлення та доставки. Більш економний варіант – це використання більш дешевих обчислювальних плат з вбудованими мікропроцесорами та допоміжними модулями у вигляді камери, датчиків чи серводвигунів. Одним із прикладів такої плати є Raspberry Pi, вартість якої близько 40\$, тому такий вибір є більш раціональним для невеликих підприємств.

Першим кроком в алгоритмі позиціонування є ініціалізація початкових та кінцевих координат у загальній системі координат. Перший крок розбивається на визначення координат кожної ланки – початкових координат, визначення координат об'єкта переміщення – кінцевих координат та зведення всіх координат у єдину загальну систему. В роботі використовується формули запропоновані Річардом Полом. Математично кожна ланка руки робота-маніпулятора описується матрицею  $A$  розміру  $4 \times 4$ , яка зображена у формулі (1).

$$A_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \times \sin \theta_i & \sin \alpha_i \times \sin \theta_i & \alpha_i \times \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \times \cos \theta_i & -\sin \alpha_i \times \cos \theta_i & \alpha_i \times \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & s_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

де  $i$  – індекс, який вказує на номер ланки;  $\theta$  – кут приєднання (кут на який треба повернути вісь  $x_i$  відносно осі  $z_i$  щоб вісь  $x_{i+1}$  стала перпендикулярною до  $x_i$ );  $s_i$  – довжина ланки в см;  $\alpha$  – кут зміщення (приймає значення 0 або 90).

Умова поставленої задачі передбачає, що місце об'єкта переміщення відоме, але постає питання до

знаходження і задання його координат в одну систему з рукою робота-маніпулятора. Вирішенням є приведення координат всіх ланок та об'єкта (куба) до однієї системи координат відносно однієї ланки, найчастіше нульової, тобто стійка руки робота-маніпулятора [2]. Приведення відбувається наступним шляхом за допомогою формул (2 - 3).

$$T_i = A_0 \times A_1 \times \dots \times A_i \quad (2)$$

$$R_0 = T_i \times R_i \quad (3)$$

де  $i$  – індекс, який вказує номер ланки;  $A$  – матриця координат окремих ланок;  $R_i$  – вектор координат об'єкту;  $T$  – приведена матриця координат.

Наступна кроком алгоритму є задача пошуку траєкторії. Задача пошуку траєкторії вирішується після розв'язання зворотної задачі кінематики, яка розв'язується координатним, векторним або натуральним способами. Пропонується використовувати координатний спосіб.

Підведення ланки захоплення до об'єкту переміщення відбувається за допомогою управління сервоприводами. Управління сервоприводами виконує система використовуючи файловий тип інтерфейсу та стандартні функції з відповідних бібліотек для Raspberry Pi. Серводвигуни приводяться у рух по заданим траєкторіям завдяки стандартним функціям: `active`, `delayed`, `mode`, `servo_max`, `servo`.

Задача вибору вказаного об'єкту переміщення зводиться до розпізнавання об'єкту використовуючи штучний інтелект. Вирішується дана задача за допомогою камери на апаратному рівні та за допомогою підключення штучного інтелекту на програмному рівні. Використовуючи штучний інтелект у вигляді нечіткої логіки, ми навчаємо нашу інтелектуальну систему розпізнавати два кольори об'єкту – білий та чорний. Коли об'єкт розпізнано, підключена камера фокусується на обраному об'єкті та за допомогою ультразвукового далекоміру отримує відстань. Знаючи початкові координати та відстань до обраного об'єкту переміщення розв'язуючи зворотну задачу кінематики система отримує кінцеві координати в єдиній системі.

Захоплення об'єкту відбувається по заданим координатам, остання ланка приводить серводвигуни у рух під час захоплення. Зазначимо, що остання ланка має два положення перше – вільне, коли серводвигуни у початковому положенні, та друге – схват, коли серводвигуни фіксуються у положенні здатному утримати об'єкт переміщення. За допомогою датчиків тиску на останній ланці система регулює серводвигуни, щоб пальці руки робота-маніпулятора втримали об'єкт та не розчавили його.

Переміщення об'єкту починається з розрахунку траєкторії від поточних до кінцевих координат, а серводвигуни зафіксовані в положенні «схват». Коли об'єкт доставлено до кінцевих координат, серводвигуни переходять з положення «схват» у положення «вільне», тобто пальці руки робота-маніпулятора відпускають його.

Повернення руки робота-маніпулятора в початкове положення відбувається внаслідок розв'язку зворотної задачі кінематики. Коли з даного положення рука повертається в початкове положення, тобто розраховується траєкторія по поточним та кінцевим координатам, а системи приводить у рух серводвигуни.

Таким чином запропонований метод позиціонування має наступний алгоритм: ініціалізація початкових та кінцевих координат у загальній системі координат; розрахунок траєкторії; підведення ланки захоплення до об'єкту переміщення; визначення об'єкту переміщення; захоплення об'єкту; переміщення об'єкту; повернення руки робота-маніпулятора в початкове положення.

### **Висновки**

Запропоновано метод позиціонування руки робота-маніпулятора автоматизованого складу в тривимірному просторі. Представлено покроковий алгоритм даного методу з врахуванням особливостей технологічної обчислювальної платформи Raspberry Pi 3.

### **СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Основы робототехники / Никитин К. Д., Василенко Н. В., Пономарёв В. П., Смолин А. Ю. — ТОМСК МГП «РАСКО» 2013. — 238 с.
2. Заводим Raspberry Pi / Ричардсон М., Уоллес Ш. — пер. на рус. Амперка, 2013. — 230 с.

**Гришин Павло Андрійович** — студент групи 2АВ-13б, факультет комп'ютерних систем та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: pashagrishyn@gmail.com

**Москвіна Світлана Михайлівна** — кандидат технічних наук, професор кафедри комп'ютерних систем управління, Вінницький національний технічний університет, м. Вінниця

**Grishyn Pavlo A.** — student of group 2AV-13b, department of computer systems and automation, Vinnytsia National Technical University, Vinnytsia, e-mail: pashagrishyn@gmail.com

**Moskvina Svitlana M.** — candidate of technical science, professor of computer systems and automation chair, Vinnytsia National Technical University, Vinnytsia