УДК 681.5:004.94 DOI https://doi.org/10.35546/kntu2078-4481.2025.3.1.9

А. О. ДЕРКАЧЕНКО
аспірант кафедри автоматизації, робототехніки і мехатроніки
Херсонський національний технічний університет
ORCID: 0000-0002-0443-0329

О. В. ПОЛИВОДА
кандидат технічних наук,
доцент кафедри автоматизації, робототехніки і мехатроніки
Херсонський національний технічний університет
ORCID: 0000-0002-6323-3739

РОЗРОБКА АЛГОРИТМУ ВИЗНАЧЕННЯ ПОЛОЖЕННЯ ПЛАТФОРМИ СФЕРИЧНОГО ПАРАЛЕЛЬНОГО МЕХАНІЗМУ В 3D-ПРОСТОРІ ТА ЙОГО ПРОГРАМНА РЕАЛІЗАЦІЯ

У статті розглянуто питання математичного моделювання та програмної реалізації алгоритмів для сферичних паралельних механізмів (СПМ), які завдяки високій точності та жорсткості знаходять широке застосування у сучасній інженерії. Проаналізовано можливості використання СПМ у тренажерних системах, робототехніці, медичному обладнанні та системах стабілізації. Визначено, що ключовим завданням є не лише відтворення орієнтації платформи у просторі, але й побудова математичного апарату для визначення відносних кутів переміщення та безпосередніх керуючих сигналів для серво двигунів.

Запропоновано метод розрахунку координат рухомої платформи на основі матричних перетворень, який дозволяє точно враховувати повороти відносно осей X, Y, Z. Окрему увагу приділено задачі визначення координат проміжних опорних точок механізму, що лежать на колі в площині XY. Сформульовано систему геометричних та кінематичних рівнянь, розв'язання якої породжує множину можливих координат положень ланок механізму. Для відбору єдиного фізично коректного рішення розроблено багатокроковий алгоритм фільтрації, що включає перевірку належності точок колу, дотримання мінімальних відстаней між ланками, відсутність перетину та вибір найближчої до попереднього стану комбінації координат.

У роботі запропоновано метод визначення відносних кутів переміщення середніх точок, яка дає змогу оцінювати як напрямок, так і величину їх зміщення. Це дозволяє перейти від умовних поворотів платформи до обчислення кутів повороту серво двигунів, які виступають безпосередніми керуючими впливами. На основі розробленого алгоритму реалізовано програмний модуль мовою Python із використанням бібліотек NumPy та Matplotlib, що забезпечує як чисельні розрахунки, так і дво- та тривимірну візуалізацію роботи механізму.

Отримані результати підтвердили адекватність математичної моделі та її придатність для інтеграції у системи автоматизованого керування. Запропонований підхід забезпечує точне відстеження положення платформи, обчислення відносних кутів та визначення сигналів для серво двигунів. Це відкриває перспективи практичного застосування у робототехнічних комплексах, системах стабілізації, тренажерах та медичних пристроях, де особливо важливі висока точність, надійність та безперервність керування.

Ключові слова: *робототехнічні системи, сферичний паралельний механізм, алгоритм керування рухом, програмна реалізація, планування траєкторії, зворотна задача кінематики, комп'ютерне моделювання.*

A. O. DERKACHENKO
Postgraduate Student at the Automation, Robotics
and Mechatronics Department
Kherson National Technical University
ORCID: 0000-0002-0443-0329

O. V. POLYVODA
Candidate of Sciences in Technology,
Associate Professor at the Automation, Robotics
and Mechatronics Department
Kherson National Technical University
ORCID: 0000-0002-6323-3739

© Деркаченко А. О., Поливода О. В., 2025
Стаття поширюється на умовах ліцензії CC BY 4.0

DEVELOPMENT OF AN ALGORITHM FOR DETERMINING THE POSITION OF A SPHERICAL PARALLEL MECHANISM PLATFORM IN 3D SPACE AND ITS SOFTWARE IMPLEMENTATION

The paper addresses the problem of mathematical modeling and software implementation of algorithms for spherical parallel mechanisms (SPM), which are widely used in modern engineering due to their high rigidity and accuracy. Potential applications of SPM include training simulators, robotic manipulators, medical equipment, and stabilization systems. It is shown that the key task is not only to reproduce the spatial orientation of the moving platform but also to develop a mathematical framework for determining relative angular displacements and direct control signals for servo drives.

The method is proposed for calculating the coordinates of the moving platform based on rotation matrices, which allows precise consideration of rotations relative to the X, Y, and Z axes. Special attention is given to determining the coordinates of the intermediate reference of the mechanism located at the circle in the XY-plane. The system of geometric and kinematic equations is formulated, the solution of which yields multiple possible coordinates of the mechanism links positions. To select the only physically feasible solution, a multi-step filtering algorithm has been developed, including circle-membership verification, minimum distance checks, link intersection analysis, and the choice of the coordinates configuration closest to the previous state.

The study introduces a method for determining the relative angular displacements of the intermediate points, enabling the estimation of both direction and magnitude of their shifts. This provides a transition from the platform's conditional rotations to the computation of servo drive rotation angles, which act as direct control signals. Based on the developed algorithm, a software module was implemented in Python using NumPy and Matplotlib libraries, enabling both numerical calculations and two- and three-dimensional visualization of the mechanism's operation.

The results obtained confirm the adequacy of the mathematical model and its suitability for integration into automated control systems. The proposed approach ensures accurate tracking of platform position, determination of relative angles, and calculation of control signals for servo drive. This makes the method applicable in robotic systems, stabilization systems, training simulators, and medical devices, where high precision, reliability, and continuous control are crucial.

Key words: *robotic systems, spherical parallel mechanism, motion control algorithm, software implementation, trajectory planning, inverse kinematics problem, computer modeling.*

Постановка проблеми

Просторові платформи (механізми) побудовані за принципом сферичних паралельних механізмів (СПМ) належать до класу паралельних кінематичних структур, що завдяки високій жорсткості та точності широко застосовуються у сучасній інженерії. Їх застосування охоплює надзвичайно широкий спектр завдань: від авіаційних і космічних тренажерів, які потребують реалістичного відтворення рухів у просторі, до робототехнічних маніпуляторів, що виконують високоточні операції у промисловості, зокрема при автоматизованому складанні, зварюванні та роботі з крихкими матеріалами. Завдяки особливостям геометричної структури такі маніпулятори поєднують компактність із підвищеною вантажопідйомністю та забезпечують узгоджені рухи з кількома ступенями свободи – критично важливу властивість для сучасних виробничих процесів. Важливе місце СПМ посідають і в системах стабілізації, де необхідно компенсувати зовнішні коливання та збурення: у стабілізаторах для камер, у платформах віртуальної реальності, а також у морських та авіаційних комплексах. У таких умовах вони забезпечують високу швидкодію, точність і плавність рухів при мінімальних затримках. Окремо варто відзначити медичні системи, де такі механізми дозволяють реалізувати складні, але контрольовані рухи з високою точністю, що є критичним фактором у хірургії та діагностиці.

Для повного вирішення задач ідентифікації параметрів руху елементів каркасних конструкцій необхідно визначати параметри, що безпосередньо можуть бути передані на виконавчі органи системи, а саме на серво двигуни. Окрім визначення координат точок платформи та середньої ланки потрібно також, обчислити відносні кути переміщення та розрахувати кути повороту серво двигунів що і є основним завданням цього дослідження. Розв'язок цієї задачі відкриває шлях до створення замкненої системи керування механізмом, де кожен етап – від вхідних даних до керуючих сигналів – має чітке математичне та алгоритмічне обґрунтування.

Формулювання мети дослідження

Метою дослідження є розробка алгоритму визначення координат точок і кутів повороту платформи та його реалізація у вигляді програмного продукту мовою програмування Python з використанням засобів візуалізації руху платформи у 2D- та 3D-просторі.

Аналіз останніх досліджень і публікацій

Для опису положення рухомої платформи прийнято використовувати три кути повороту відносно координатних осей: α (навколо осі X), β (навколо осі Y), γ (навколо осі Z), які є параметрами математичної моделі зворотної динаміки руху елементів СПМ. На основі них обчислюються координати трьох характерних точок платформи – C, $C2$, $C3$, що утворюють жорсткий трикутник у просторі. Визначення цих координат реалізовано через матричні перетворення та представлено у роботі [1].

Рішення задачі визначення координат точок середньої ланки – B, $B2$, $B3$, що лежать на колі у площині XY ($z = 0$), виконується із застосуванням методу аналітичної геометрії [2–4] і було представлене в роботі [5, 6]. Методи представлені в роботах [1, 5, 6] дають повний набір просторових координат як для рухомої платформи, так і проміжних опорних точок. На рисунку 1 наведено схему розташування опорних точок кола основи (B, $B2$, $B3$) та координат платформи (трикутник з вершинами C, $C2$, $C3$).

Рухома платформа розглядуваного механізму визначається трьома характерними точками C, $C2$, $C3$, які задають її положення у тривимірному просторі відносно декартової системи координат (x, y, z). У початковому положенні координати цих точок мають вигляд: $C(x_C, y_C, z_C)$, $C2(x_{C2}, y_{C2}, z_{C2})$, $C3(x_{C3}, y_{C3}, z_{C3})$.

Фактично трикутник, утворений цими точками, визначає площину платформи, а також дозволяє однозначно відстежувати її орієнтацію у просторі.

Опорні точки основи механізму – B, $B2$, $B3$ – розташовані у площині XY. Вони утворюють коло з радіусом R та центром у початку координат, що відображається у рівняннях: $B(x_B, y_B, 0)$, $B2(x_{B2}, y_{B2}, 0)$, $B3(x_{B3}, y_{B3}, 0)$.

Таким чином, основа задає геометричні обмеження для платформи, а її положення у просторі визначається взаємодією точок C і B, $C2$ і $B2$, $C3$ і $B3$.

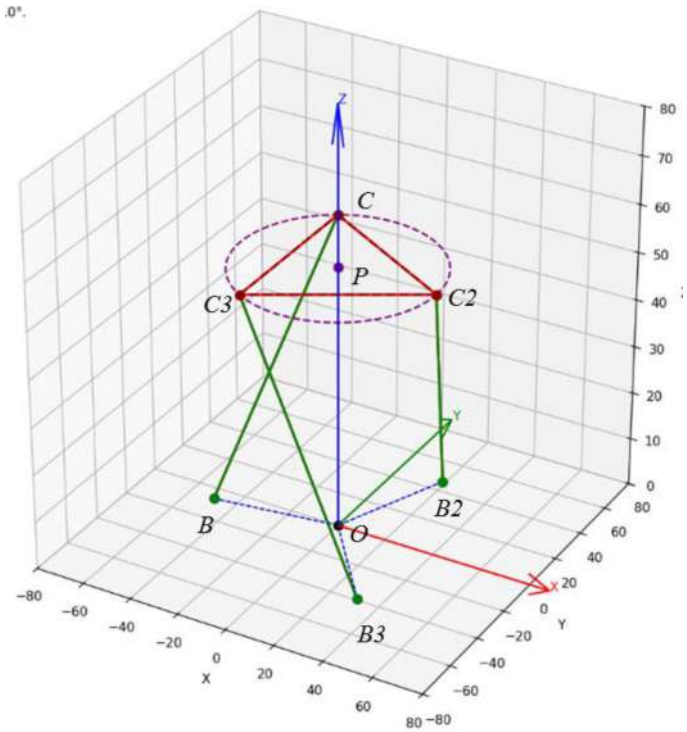

Рис. 1. Схема платформи у початковому положенні

Поточне положення рухомої платформи відносно початкового визначається трьома кутами повороту навколо координатних осей: α – поворот навколо осі X, β – поворот навколо осі Y, γ – поворот навколо осі Z.

Ці кути належать до вхідних параметрів системи і визначають, як саме початкові координати точок C, $C2$, $C3$ будуть перетворені у нове положення.

Перехід від початкових координат до поточних здійснюється за допомогою матриць повороту. Для кожної з осей вони мають вигляд:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}, \quad R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}, \quad R_z(\gamma) = \begin{bmatrix} \cos\gamma & 0 & \sin\gamma \\ 0 & 1 & 0 \\ -\sin\gamma & 0 & \cos\gamma \end{bmatrix}, \tag{1}$$

де $R_x(\alpha)$ – це матриця повороту навколо осі X, $R_y(\beta)$ – це матриця повороту навколо осі Y, $R_z(\gamma)$ – це матриця повороту навколо осі Z.

Загальна матриця повороту R визначається як добуток трьох матриць:

$$R = R_x(\alpha) \cdot R_y(\beta) \cdot R_z(\gamma). \tag{2}$$

Тоді нові координати будь-якої точки платформи C_i^{new} можна обчислити за формулою:

$$C_i^{new} = R \cdot C_i^{old}, \tag{3}$$

де C_i^{old} – це матриця значень останніх попередніх координат відповідних точок платформи, i – номер відповідної точки платформи.

На рисунку 2 наведено приклад послідовного застосування матричного перетворення до координат точок платформи.

Цей підхід забезпечує коректне врахування орієнтації платформи у просторі та дозволяє однозначно відтворити її положення після довільних обертів.

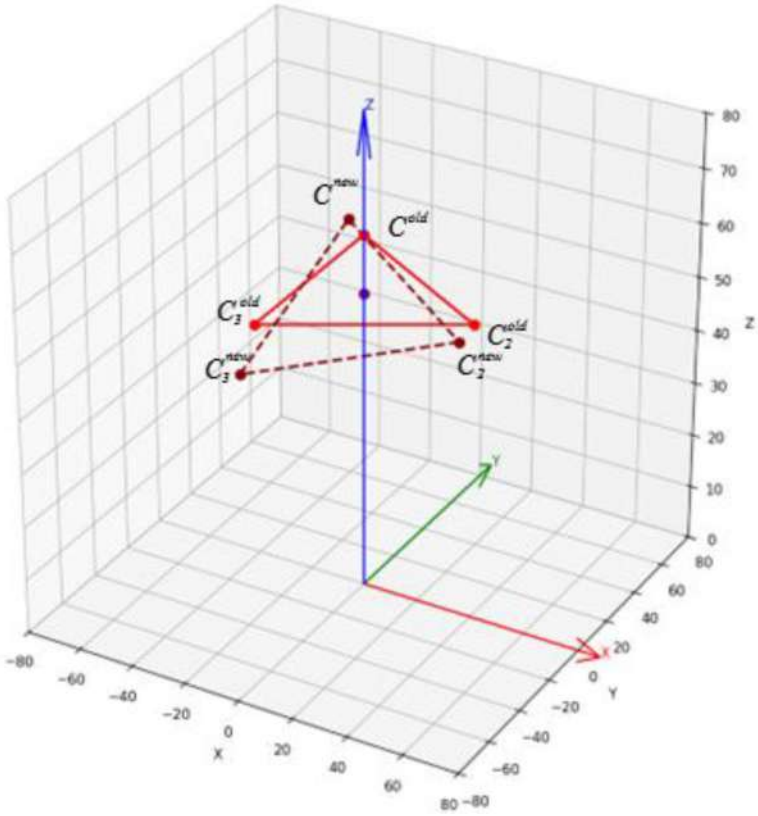

Рис. 2. Результат послідовного застосування матриць R_x, R_y, R_z до точок платформи

Завдяки використанню апарату матриць обертання задача зводиться до послідовного застосування лінійних перетворень, що зручно як для аналітичних досліджень, так і для програмної реалізації. Проте для остаточного формулювання моделі динаміки руху СПМ необхідно враховувати обмеження на координати точок основи та розробити алгоритм визначення відносних кутів переміщення, що безпосередньо пов'язані з роботою серво двигунів.

Викладення основного матеріалу дослідження

Розглянемо математичну модель визначення координат середніх точок механізму типу СПМ.

Робоча платформа механізму описується трьома точками C, $C2$, $C3$ в системі координат XYZ, тоді як шарніри середньої ланки (точки B, $B2$, $B3$) обмежені площиною XY. Це означає, що їх координати мають вигляд (x, y), а значення z завжди дорівнює нулю.

Необхідно знайти таке положення точок B, $B2$, $B3$, яке б одночасно задовольняло двом ключовим обмеженням.

Геометричне обмеження: усі точки B лежать на колі з радіусом R_1 та центром у початку координат $O(0,0)$, що можна записати у вигляді рівняння:

$$x_B^2 + y_B^2 = R_1^2. \tag{4}$$

Кінематичне обмеження: відстань між відповідними точками C та B дорівнює довжині ланки механізму R_c, що в координатній формі виражається як:

$$(x_C - x_B)^2 + (y_C - y_B)^2 + (z_C)^2 = R_c^2. \tag{5}$$

Аналогічні системи рівнянь формуються і для пар: $C2$ і $B2$ та $C3$ і $B3$.

Таким чином, задача зводиться до розв'язання трьох систем нелінійних рівнянь, кожна з яких може мати до чотирьох можливих розв'язків, що вимагає розгляду двох додаткових задач:

1. Визначення відносних кутів переміщення точок середньої ланки (B, $B2$, $B3$), які характеризують зміну положення точок при переході від попереднього стану до прогнозованого, наступного та дають можливість відстежувати не лише абсолютні координати, але й напрямок та величину зміщення кожної точки по колу основи. Таким чином, відносні кути дають можливість за геометричними параметрами механізму досліджувати його реальну динаміку.

2. Обчислення кутів повороту серво двигунів (a, $a2$, $a3$), що є ключовою задачею, оскільки саме ці кути є безпосередніми керуючими впливами, які подаються на виконавчі пристрої. Правильність їх визначення істотно впливає на точність руху платформи, стабільність роботи системи та її відповідність заданій траєкторії. Від обчислення кутів повороту (a, $a2$, $a3$) залежить можливість коректного перетворення умовних поворотів платформи (α, β, γ) у фізично реалізовані положення серво двигунів.

Через надлишковість отриманих рішень виникає потреба у відборі єдиної фізично допустимої конфігурації механізму, що реалізується у багатокроковому алгоритмі фільтрації.

На першому етапі здійснюється відбір точок, що належать колу з радіусом R_1 (з урахуванням допустимої похибки ε). Для цього покроково аналізуються всі можливі комбінації координат (x, y) точок B, $B2$, $B3$. Обчислюється евклідова норма відповідного вектора, після чого визначається різниця між цією величиною та радіусом R_1. Якщо модуль різниці не перевищує ε, точка вважається такою, що задовольняє умові належності колу. На цьому етапі для кожної з точок B, $B2$, $B3$ може існувати кілька допустимих варіантів. З отриманих варіантів формується множина всіх можливих комбінацій, що надалі підлягають подальшій фільтрації. Результати виконання першого етапу, а саме можливі положення точок, які задовольняють геометричному обмеженню $x_B^2 + y_B^2 = R_1^2$ наведено на рисунку 3.

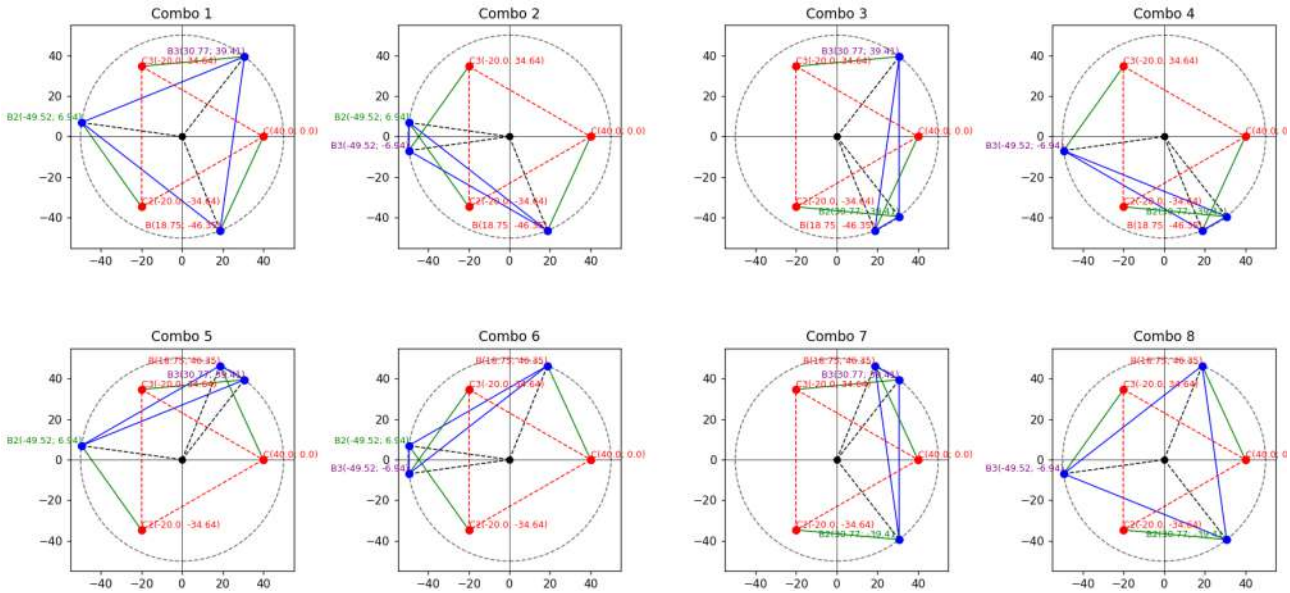

Рис. 3. Визначення множини кандидатів точок *B*, *B*2, *B*3 шляхом перевірки належності колу радіуса R_1

Другий етап включає два послідовні кроки перевірки, спрямовані на забезпечення виконання фізичних обмежень механізму: перевірка мінімальної відстані між можливими положеннями точок та перевірка на відсутність перетину ланок.

Для уникнення вироджених конфігурацій виконується аналіз попарних відстаней між усіма кандидатами точок B, $B2$, $B3$. Якщо будь-яка з відстаней виявляється меншою за встановлену граничну величину $d_{\min}$, така комбінація відкидається як некоректна. Ця умова відображає фізичне обмеження механізму, пов'язане з мінімально допустимою шириною з'єднувальної ланки. Результати процедури відсіювання комбінацій, у яких відстань між точками B, $B2$, $B3$ є меншою за допустиме значення, що відповідає фізичним обмеженням механізму наведено на рисунку 4.

Аналіз орієнтації векторів, що описують з'єднання між точками B і C дозволяє виявити потенційні випадки перетину ланок, що є фізично неможливими у реальній роботі механізму. Всі конфігурації, для яких зафіксовано перетин, відсіюються з подальшого розгляду. Результати цієї перевірки, а саме відсіювання конфігурацій, у яких зафіксовано перетин відрізків B і C, $B2$ і $C2$, $B3$ і $C3$, як таких, що є фізично неможливими наведено на рисунку 5.

Остаточний етап здійснюється за критерієм найбільшої близькості до попереднього положення системи, що забезпечує плавність і безперервність руху.

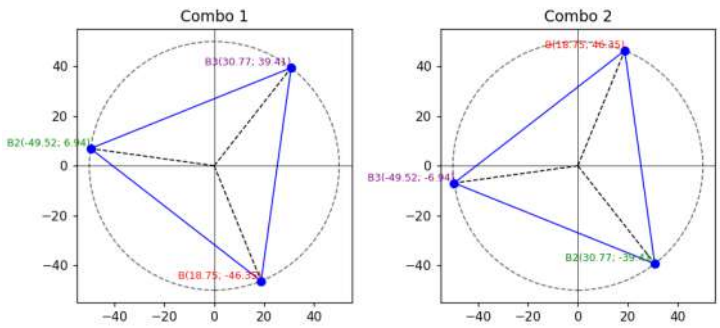

Рис. 4. Результат фільтрації множини кандидатів на основі критерію мінімальної відстані $d_{\min}$

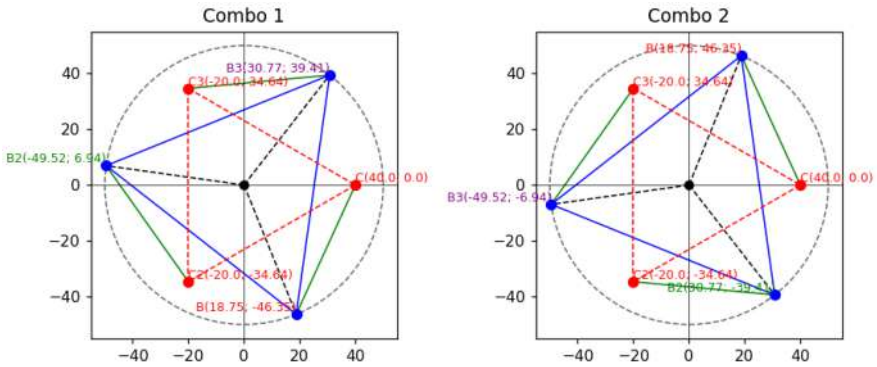

Рис. 5. Результат перевірки на відсутність перетину ланок за допомогою аналізу орієнтації векторів

Таке послідовне виконання процедур дало можливість зменшити кількість комбінацій до одного варіанту, що забезпечує коректне та стабільне керування виконавчими органами системи.

Програмна реалізація алгоритму побудована на мові Python та бібліотек NumPy [7] і matplotlib [8]. Архітектура передбачає створення класу GetAngles, що інкапсулює основну функціональність.

Основні кроки реалізації алгоритму:

1. Завантаження та оновлення конфігураційних даних включає в себе зберігання координат точок платформи (C, $C2$, $C3$), середніх ланок (B, $B2$, $B3$), а також кути (α, β, γ та a, $a2$, $a3$). Дані поділені на «*zero*» (початкові значення), «*last*» (останній стан) та «*all*» (історія змін).
2. Обчислення нових координат платформи відбувається за допомогою матриць повороту навколо осей X, Y, Z згідно рівнянь (1, 2). Таким чином, для будь-якої точки C платформи застосовується перетворення (3).
3. Генерація та відбір точок B, $B2$, $B3$ відбувається з використанням функції *get_m_cords*, що реалізує розв'язання системи (4, 5).На виході маємо до чотирьох кандидатів для кожної точки. Для подальшого відсіювання вироджених точок використовуються процедури:

```
filter_points_on_circle – перевіряє належність точок колу;
   def filter_points_on_circle(self, points, r, eps=1e-6):
      valid_points = []
      for x, y in points:
         if abs(x**2 + y**2 - r**2) < eps:
            rounded = (round(float(x), 8), round(float(y), 8))
            if rounded not in valid_points:
               valid_points.append(rounded)
      return valid_points
```

```
filter_valid_combinations – відкидає дублікати та точки, що надто близько одна до одної;
   def filter_valid_combinations(self, B, B2, B3, min_dist=10.0):
      valid_combinations = []
      for combo in itertools.product(B, B2, B3):
         if (x2 - x1)**2 + (y2 - y1)**2 <= min_dist**2:
            valid_combinations.append(combo)
      return valid_combinations
```

```
is_valid_combo – перевіряє, щоб відрізки B–C, B2–C2, B3–C3 не перетиналися.
   def segments_intersect(self, p1, p2, q1, q2):
      def orient(a, b, c): return (b[0]-a[0])*(c[1]-a[1]) - (b[1]-a[1])*(c[0]-a[0])
      return orient(p1, p2, q1) * orient(p1, p2, q2) < 0
   def is_valid_combo(self, B, B2, B3, C, C2, C3):
      segments = [(B, C), (B2, C2), (B3, C3)]
         if self.segments_intersect(...): return False
      return True
```

4. Вибір оптимальної комбінації виконується з використанням функції *choose_closest_combo*, яка виконує пошук комбінації, що мінімізує сумарну відстань до попередніх *last* координат. Що може бути аналітично виражено як:

$$combo^{*} = \arg \min_{combo} \sum_{i=1}^{3} \left\| B_i - B_i^{last} \right\|. \tag{6}$$

Або у вигляді програмного коду:

```
def choose_closest_combo(self, combos, last_coords):
   def dist(a, b): return np.linalg.norm(np.array(a) - np.array(b))
   best_combo = min(combos, key=lambda combo: sum(dist(c, l) for c, l in zip(combo, last_
coords)), default=None)
   return best_combo
```

5. Для кожної пари (B^{last}, B^{new}) формується вектор зміщення та обчислюється кут повороту за формулою:

$$\theta = \arcsin \sin \frac{(|v|)}{R1}, \tag{7}$$

де v – вектор зміщення.

В результаті отримуємо кути a, $a2$, $a3$.

6. Візуалізація результатів для 2D та 3D варіантів відбувається на допомогою таких функцій:

– *plot_combinations* – відображення можливих варіантів точок у 2D. Результат 2D візуалізації наведені на рисунку 6, де схематично зображено точки C та варіанти точок B на колі;

– *plot_mechanism_3d* – побудова повної 3D-моделі з урахуванням платформи, точок B, $B2$, $B3$, центру O та кола навколо точки P наведена на рисунку 7, де платформа зображена у вигляді трикутника C, $C2$, $C3$, точки B, $B2$, $B3$ у площині XY.

Висновки

У ході дослідження було вдосконалено математичну модель динаміки руху елементів механізму та розроблено програмний алгоритм для визначення координат і кутів повороту для механізмів типу СПМ. Отримані результати демонструють, що запропонований підхід дозволяє ефективно поєднати математичний апарат з практичними методами комп'ютерного моделювання.

По-перше, забезпечено коректне відстеження положення рухомої платформи, що дає змогу не лише визначати її просторову орієнтацію, але й аналізувати динаміку змін у процесі руху. Це відкриває можливості для точного моделювання сценаріїв роботи механізму у різних умовах.

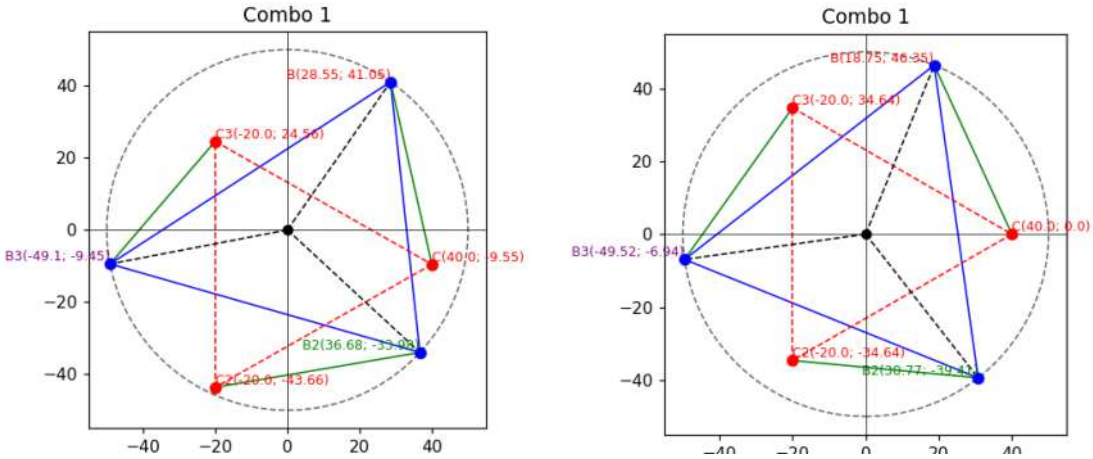

Рис. 6. Приклад 2D візуалізації можливих точок *B*, *B*2, *B*3

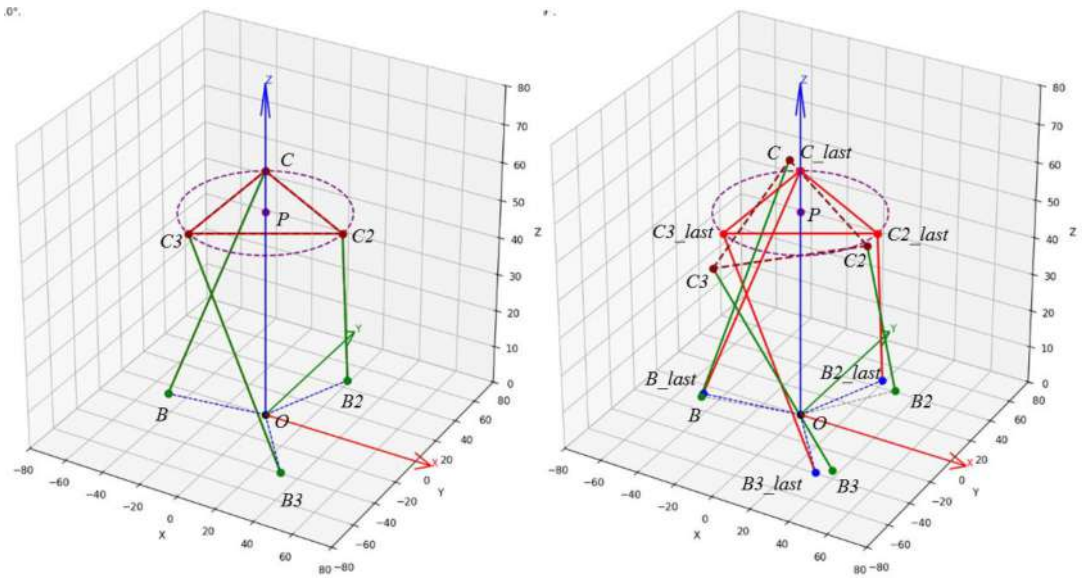

Рис. 7. Приклад 3D візуалізації механізму

По-друге, розроблений алгоритм дозволяє визначати переміщення середніх точок, які відіграють ключову роль у передачі руху між платформою та основою. Завдяки цьому стає можливим більш детально досліджувати кінематичні обмеження та геометричну узгодженість руху всієї конструкції

По-третє, реалізовано обчислення кутів серво двигунів, що виступають безпосередніми керуючими впливами. Це є важливим кроком у напрямку практичної реалізації, оскільки саме ці значення використовуються у системах керування для точного задання положення виконавчих механізмів.

Запропонована система забезпечує візуалізацію руху у дво- та тривимірному просторі, що не лише полегшує налагодження алгоритмів, а й дає інтуїтивне розуміння роботи механізму. Візуальне представлення змін положення платформи робить дослідження більш наочним і дозволяє швидко оцінювати коректність обчислень.

Особливе значення має можливість одночасного відстеження як умовних кутів повороту, так і абсолютних кутів платформи. Це поєднання є критично важливим для систем керування, де необхідно враховувати не лише локальні відносні зміщення, але й глобальне положення механізму у просторі. Такий підхід робить розроблений метод придатним для практичного застосування у робототехнічних комплексах, тренажерах та системах стабілізації, де точність і надійність контролю визначають ефективність всієї системи.

Проведене дослідження доводить, що розроблена математична модель і програмна реалізація дозволяють комплексно вирішувати задачу відстеження координат і кутів у СПМ-подібному механізмі.

Список використаної літератури

1. Derkachenko A., Polyvoda O., Lebedenko Y., Kalinina K. Research of Control Methods of a Spherical Parallel Mechanism Using Intelligent Data Processing. 2023 IEEE 5th International Conference on Modern Electrical and Energy System (MEES). Kremenchuk, Ukraine. 2023. P. 1–5. doi: 10.1109/MEES61502.2023.10402530

2. Автоматизована система діагностики внутрішніх поверхонь промислових автоклавів з використанням сферичного паралельного механізму. Вісник ХНТУ. 2025. №2. С. 1–6. doi: https://doi.org/10.35546/kntu2078-4481.2025.2.1.6

3. Дімітрова-Бурлаєнко С. Д., Бурлаєнко В. М., Гиря Н. П. Розв'язання задач аналітичної геометрії векторним методом : навч.-метод. посібник. 2-ге вид., випр. і доп. Харків : НТУ «ХПІ», 2020. 50 с.

4. Wang K. A theoretical analysis method of spatial analytic geometry and mathematics under digital twins. Advances in Civil Engineering. 2021. Article ID 8910274. doi: 10.1155/2021/8910274

5. Georgiev S. G., Zennir K., Boukarou A. Multiplicative analytic geometry. Taylor & Francis. 2022. 280 p. doi: 10.1201/9781003325284

6. Lebedenko Y., Polyvoda O., Derkachenko A., Modlo Y., Demishonkova S., Pylypenko Y. (2022). Research of Control Systems for Robotic Spatial Planning Platforms. 2022 IEEE 4th International Conference on Modern Electrical and Energy System (MEES). Kremenchuk, Ukraine. pp. 1–4. doi: 10.1109/MEES58014.2022.10005765

7. Numpy: Fundamental package for scientific computing with Python. URL: https://numpy.org/ (дата звернення: 14.09.2025).

8. Matplotlib: Visualization with Python. URL: https://matplotlib.org/ (дата звернення: 14.09.2025).

References

1. Derkachenko A., Polyvoda O., Lebedenko Y., Kalinina K. (2023). Research of Control Methods of a Spherical Parallel Mechanism Using Intelligent Data Processing. *2023 IEEE 5th International Conference on Modern Electrical and Energy System (MEES)*. Kremenchuk, Ukraine. pp. 1–5. doi: 10.1109/MEES61502.2023.10402530
2. Automated system for diagnostics of internal surfaces of industrial autoclaves using a spherical parallel mechanism. Visnyk KNTU. (2025). № 2. pp. 48–55. [in Ukrainian] doi: https://doi.org/10.35546/kntu2078-4481.2025.2.1.6
3. Dmytrova-Burlaienko S. D., Burlaienko V. M., Hyria N. P. (2020). Rozviazannia zadach analitychnoi heometrii vektornym metodom: navch.-metod. posibnyk. 2nd ed., rev. and add. Kharkiv : NTU "KhPI". 50 p.
4. Wang K. (2021). A theoretical analysis method of spatial analytic geometry and mathematics under digital twins. Advances in Civil Engineering. Article ID 8910274. [in Ukrainian] doi: 10.1155/2021/8910274.
5. Georgiev S. G., Zennir K., Boukarou A. (2022). Multiplicative analytic geometry. Taylor & Francis. 280 p. doi: 10.1201/9781003325284
6. Lebedenko Y., Polyvoda O., Derkachenko A., Modlo Y., Demishonkova S., Pylypenko Y. (2022). Research of Control Systems for Robotic Spatial Planning Platforms. 2022 IEEE 4th International Conference on Modern Electrical and Energy System (MEES). Kremenchuk, Ukraine. pp. 1–4. doi: 10.1109/MEES58014.2022.10005765
7. Numpy: Fundamental package for scientific computing with Python. Retrieved from https://numpy.org/ (accessed: September 14, 2025)
8. Matplotlib: Visualization with Python. Retrieved from https://matplotlib.org/ (accessed: September 14, 2025).

Дата першого надходження рукопису до видання: 27.09.2025
Дата прийнятого до друку рукопису після рецензування: 24.10.2025
Дата публікації: 28.11.2025