

Г. В. ШАБАНОВ, Ю. М. АНДРЕЄВ

АНАЛІТИЧНЕ РІШЕННЯ ОБЕРНЕНОГО І ПРЯМОГО ЗАВДАННЯ КІНЕМАТИКИ ПРОСТОРОВОГО КУТОВОГО МАНІПУЛЯТОРА АBB З ОДАЛЬШИМ 3D-МОДЕЛЮВАННЯМ

У статті пропонується алгоритм вирішення оберненої задачі кінематики шестиступеневого кутового маніпулятора на базі моделі робота АBB. Алгоритм заснований на отриманих авторами аналітичних викладках, які становлять послідовність виразів, що дозволяють за заданим законом руху вихідної ланки – захвату визначити всі 6 кутів повороту частин кінематичних пар, що з'єднують ланки. Спочатку з умов виконання захватом виробничого завдання визначається закон його руху – функціями часу декартових координат полюса та орієнтаційних параметрів. За полюс може бути обрана будь-яка точка захвату. Як орієнтаційні параметри можуть використовуватися кватерніон або матриця повороту. Показано, як можна побудувати матрицю повороту та кватерніон за законами зміни декартових координат трьох точок захвату. Особливістю маніпулятора, що розглядається тут, є наявність загальної точки осей повороту трьох останніх ланок. Це дозволяє із закону руху захвату визначити закон руху цієї точки. Після чого однозначно визначаються закони зміни кутів повороту трьох перших ланок. При цьому одночасно вирішується орієнтаційне завдання цих ланок. Для визначення кутів трьох останніх ланок складається матричне чи кватерніонне рівняння, виходячи з еквівалентності двох послідовностей поворотів ланок. Перша – сукупність вже відомих поворотів від третьої ланки через другу та першу ланку до абсолютної системи координат i , нарешті, від абсолютної системи до захвату. Параметри цих поворотів відомі. Друга послідовність – це 3 повороти на невідомі кути трьох крайніх ланок від третьої знову до захвату. Ці невідомі кути визначаються з такого рівняння. Виклад складу запропонованого алгоритму ведеться з урахуванням маніпулятора АBB, але може бути прикладений до маніпуляторів класу ПУМА. Алгоритм реалізований за допомогою спеціальної системи комп'ютерної алгебри КіДиМ та розробленої програми на C++, яка використовує можливості OpenGL та SolidWorks, що дозволяють 3D-візуалізацію результатів розрахунків.

Ключові слова: кутовий маніпулятор, обернена задача кінематики, комп'ютерне 3D-моделювання, АBB-маніпулятор.

H. SHABANOV, YU. ANDRIEIEV.

ANALYTICAL SOLUTION OF THE INVERSE AND DIRECT KINEMATICS PROBLEM FOR THE SPATIAL ANGULAR MANIPULATOR ABB WITH SUBSEQUENT 3D MODELING

This article proposes an algorithm for solving the inverse kinematics problem of a six-degree angular manipulator based on the ABB robot model. The algorithm is based on analytical derivations obtained by the authors, which consist of a sequence of expressions that allow determining all six rotation angles of the kinematic pair parts that connect the chains by a given law of movement of the output chain – the gripper. Initially, the law of its movement – functions of time of Cartesian coordinates of the pole and orientation parameters – is determined from the conditions of performing a production task by the gripper. Any point of the gripper can be chosen as the pole. Quaternion or rotation matrix can be used as orientation parameters. It is shown how a rotation matrix and quaternion can be constructed according to the laws of change of Cartesian coordinates of three gripper points. A distinctive feature of the manipulator discussed here is the presence of a common rotation axis point for the last three chains. This allows determining the law of movement of this point from the law of movement of the gripper. After which the laws of change of rotation angles of the first three chains are uniquely determined. At the same time, the orientation task of these chains is solved. To determine the angles of the last three links, a matrix or quaternion equation is compiled, based on the equivalence of two sequences of chain rotations. The first sequence is a combination of already known rotations from the third chain through the second and first chain to the absolute coordinate system and, finally, from the absolute system to the gripper. The parameters of these rotations are known. The second sequence consists of 3 rotations at unknown angles of the last three chain from the third again to the gripper. These unknown angles are determined from such an equation. The presentation of the proposed algorithm takes into account the ABB manipulator but can be applied to manipulators of the PUMA class. The algorithm is implemented using a special computer algebra system KidyM and a developed C++ program, which uses OpenGL and SolidWorks engineering graphic packet that allow 3D visualization of the calculation results.

Keywords: angular manipulator, inverse kinematics problem, computer 3D modeling, ABB manipulator.

Вступ. Розв'язання задач оберненої кінематики є одним із основних факторів розробки систем керування роботами та маніпуляторами [1-5]. Також ці завдання відіграють важливу роль у проектуванні систем захоплення руху, анімації персонажів у комп'ютерній графіці [6].

Сьогодні задачі оберненої кінематики роботів і маніпуляторів розв'язуються як аналітичними, так і чисельними методами. Їх поєднання також є одним з можливих варіантів. Останнім часом розроблені і в деяких випадках показали свою ефективність методи, які при використанні традиційних підходів також залучають методи штучного інтелекту [7-9].

Незважаючи на це, класичні аналітичні та чисельні методи залишаються основним засобом вирішення цих завдань. Серед підходів, що набули поширення останніми роками, є реалізація кватерніонної алгебри [6, 10], використання якої в ряді випадків зменшує обсяг обчислень і виключає появу сингулярностей [10].

У багатьох випадках використовуються чисельні методи [5, 6, 11, 12]. Хоча в деяких, найбільш загальних випадках, неможливо отримати рішення без їх використання, одним з їхніх недоліків залишається відсутність чисельної стабільності та велика кількість обчислювальних операцій для досягнення збіжності рішень.

Починаючи з роботи [13], яка встановила стандарт формального опису механізмів з кінематичними парами п'ятого класу, більшість досліджень у цьому напрямі використовують такий опис (див., наприклад, останню публікацію [4]). Однак D-X методика не завжди є оптимальною через використовувані обмеження (обов'язкове позначення осей кінематичних пар осями аплікат пов'язаних з ланками систем координат ланки). Це призводить до врахування зайвих для перетворення систем координат поворотів на 90° і 180° . Алгоритми, розроблені авторами даної статті, використовують прямий опис переміщень і поворотів

систем координат ланок одна відносно іншої із зазначенням осей трансляції та повороту [9, 14-16].

У даній роботі для розв'язання оберненої задачі кінематики шестиступеневого кутового маніпулятора з шістьма ступенями свободи (DOF) запропоновано використовувати суто аналітичні методи, на результатах яких можуть бути побудовані уточнюючі чисельні. Такий підхід спирається на наявність так званого принципу Пейпера [17, 18] – наявність загальної точки перетину трьох останніх ланок і передбачає наявність досить точної механічної моделі робота. Що досягається визначенням креслень усіх деталей, використання Solid Works з подальшим уточненням параметрів моделі алгоритмами оптимального пошуку (наприклад, методом Левенберга-Марквардта [19]).

Слід сказати, що при виконанні Спочатку вирішується завдання позиціонування захвату з інструментом, або без нього, у початкове для робочого процесу положення з вихідного, наприклад, парковочного положення. Аналітичний розв'язок базується на вирішенні двох пов'язаних завдань: спочатку за заданими координатами трьох точок робочого інструменту визначаються параметри його орієнтації в просторі. Вони пов'язані з кутами орієнтації (кутами повороту трьох останніх ланок). За заданими або визначеними координатами точки перетину осей обертання двох-трьох останніх ланок вирішується задача визначення транспортних кутів робота (кутів повороту перших трьох ланок). З умови спряженості орієнтації четвертої ланки відносно абсолютної системи відліку, що розглядається в транспортному та орієнтаційному кутах, вирішується вся обернена задача кінематики маніпулятора. Якщо в конкретній моделі припущення для вирішення такої задачі не виконуються повністю, обернена задача уточнюється методом Ньютона з початковим наближенням, отриманим аналітичним рішенням.

Постановка задачі. В статті пропонується аналітичний алгоритм розв'язання оберненої задачі кінематики кутових 6-ступеневих маніпуляторів, що виконані за схемою ПУМА-560 [17, 18]. Показано, що наявність загальної точки перетину осей трьох останніх ланок дозволяє розбити задачу на дві – транспортну та орієнтаційну. Транспортна визначає кути повороту перших трьох ланок, а орієнтаційна – останніх трьох. Розглянуто також випадок, коли вісь четвертої ланки проходить поруч з точкою перетину п'ятої і шостої, що є доволі типовим. Вирішення завдання в цьому випадку зводиться до розв'язання нелінійного рівняння з одним невідомим. Наведено, як отримати таке рівняння і розв'язати його методом Ньютона або бінарним пошуком. Демонструється використання запропонованого алгоритму рахунками реального маніпулятора в спеціальній системі комп'ютерної алгебри КіДиМ (ССКА КіДиМ).

Кінематична схема маніпулятора та отримання положення вихідної ланки – захвату з умов виконання робочого процесу.

На рис. 1 представлено зовнішній вид маніпулятора.

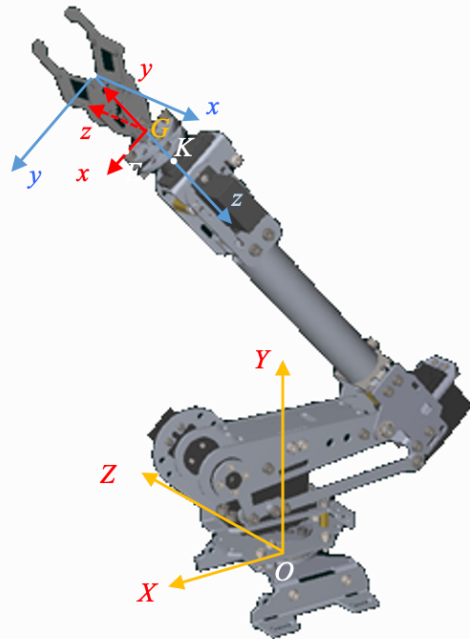


Рис. 1 – Модель маніпулятора ABB

Тут показано ще три системи координат – зовнішня нерухома (далі – абсолютна) $OXYZ$, $Dxyz$ – пов'язана з захватом і початком в т. D – положення такої системи відносно легко визначити під час руху в робочому процесі, $Gxyz$ – система координат 6-ї ланки механізму. Точка K , через яку проходить вісь аплікат системи $Dxyz$ – це точка перетину осей трьох останніх ланок робота. Точка E належить захвату і задає напрям осі ординат системи $Dxyz$ (наприклад, перпендикуляр до осі аплікат Dz , що проходить через неї, дає напрям осі ординат Dy)

За положенням точок K , E , D можна побудувати орти системи координат $Dxyz$ (отримати їх проекції на осі абсолютної системи координат)

$$\vec{k} = \frac{\overline{DK}}{DK}; \quad \vec{i} = \frac{\overline{DE}}{DE \sin(\angle KDE)} \times \vec{k}; \quad \vec{j} = \vec{k} \times \vec{i};$$

Тут $DE \sin(\angle KDE)$ – відстань т. E від осі Dz .

Таким чином, матриця, стовпцями якої є визначені орти

$$\mathbf{S}_{(Dxyz)}^{(abs)} = [\vec{i} \quad \vec{j} \quad \vec{k}] \quad (1)$$

буде матрицею повороту, що буде переводити вектори, задані в системі координат $Dxyz$, в вектори з координатами в абсолютній системі координат з початком в т. D . Для отримання абсолютних координат точки, координати якої x , y , z задано в системі $Dxyz$ в системі з початком в деякій точці O очевидна формула

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_D \\ Y_D \\ Z_D \end{bmatrix} + \mathbf{S}_{(Dxyz)}^{(abs)} \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad (2)$$

де $[X_D \ Y_D \ Z_D]^T$ – вектор т. D в системі $OXYZ$.

Тим самим, матриця $\mathbf{S}_{(Dxyz)}^{(abs)}$ задає орієнтацію захвату в абсолютній системі координат. Замість неї можна використовувати для того ж самого кватерніон.

Покажемо, як його можна побудувати з заданого положення в абсолютній системі точок K, E, D . Розглянемо рис. 2.

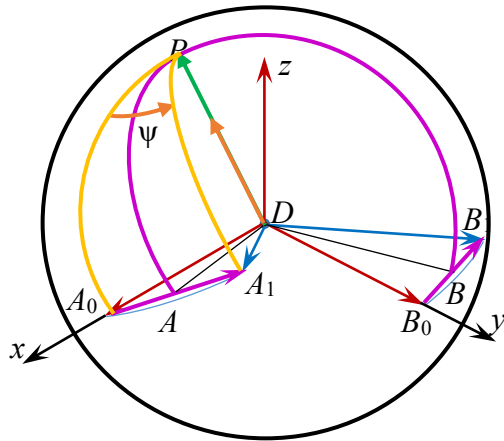


Рис. 2 – До отримання кватерніону орієнтації

Тут показана система координат $Dxyz$, її орти осей абсцис ($\vec{i} = \overline{DA_0}$) та ординат ($\vec{j} = \overline{DB_0}$) в початковому та повернутому стані ($\vec{i} = \overline{DA_1}$, $\vec{j} = \overline{DB_1}$) в деякий момент часу t . Треба знайти вісь та кут кінцевого повороту, що переводить систему координат $Dxyz$ з початкового до кінцевого положення.

Позначимо координати точок $A_0 = \{1, 0, 0\}$, $A_1 = \{x_A, y_A, z_A\}$, $B_0 = \{0, 1, 0\}$, $B_1 = \{x_B, y_B, z_B\}$.

З'єднаємо точки A_0 і A_1 вектором $\overline{A_0A_1}$, а точки B_0 і B_1 вектором $\overline{B_0B_1}$. Очевидно, що точки A_0 і A_1 (B_0 і B_1) належать дугам, які описують кінці ортів \vec{i} та \vec{j} при кінцевому повороті. А, значить, точки осі кінцевого повороту знаходяться на однаковій відстані від точок A_0 і A_1 (B_0 і B_1). Тобто вони належать площинам, що перпендикулярні векторам $\overline{A_0A_1}$ та $\overline{B_0B_1}$. Таким чином ось кінцевого повороту зразу знаходиться з векторного добутку нормалей цих площин (векторів $\overline{A_0A_1}$ та $\overline{B_0B_1}$)

$$\overline{OP} = \overline{A_0A_1} \times \overline{B_0B_1} = \begin{bmatrix} y_1z_2 - (y_2 - 1)z_1 \\ x_2z_1 - (x_1 - 1)z_2 \\ (x_1 - 1)(y_2 - 1) - x_2z_1 \end{bmatrix}.$$

Тоді визначаємо орт осі кінцевого повороту

$$\vec{e} = \frac{\overline{DP}}{|\overline{DP}|}$$

Для визначення кута кінцевого повороту – кута між площинами A_0OP і A_1OP (або – між площинами B_0OP і B_1OP) треба знайти для них нормалі. Це буде

$$\text{або} \quad \vec{n}_0 = \vec{e} \times \overline{DA_0}, \quad \vec{n}_1 = \vec{e} \times \overline{DA_1},$$

$$\text{або} \quad \vec{n}_0 = \vec{e} \times \overline{DB_0}, \quad \vec{n}_1 = \vec{e} \times \overline{DB_1}$$

в залежності які з них не дорівнюють нулю.

Тоді шуканий кут визначиться за формулою

$$\psi = \arccos \left(\frac{|\vec{n}_0 \cdot \vec{n}_1|}{|\vec{n}_0| \cdot |\vec{n}_1|} \right).$$

За цим кватерніон кінцевого повороту отримає вираз

$$\Lambda = \cos \frac{\psi}{2} + \vec{e} \sin \frac{\psi}{2}. \quad (3)$$

Замість формули (1) можна тоді використовувати формулу

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_D \\ Y_D \\ Z_D \end{bmatrix} + \text{Vect} \left(\Lambda \begin{bmatrix} 0 \\ x \\ y \\ z \end{bmatrix} \tilde{\Lambda} \right), \quad (4)$$

де функція Vect повертає векторну частину кватерніону.

Розв'язання транспортної задачі маніпулятора.

Отримані формули (1) або (3) дають змогу визначити потрібне положення т. K в абсолютній системі координат. А це дає змогу розв'язати задачу визначення кутів повороту одна від одної перших трьох ланок механізму. Тобто – розв'язати транспорту задачу – задачу доставки точки K – перетину осей трьох останніх ланок у визначене положення.

Розглянемо рис. 3. Тут наведено дві проекції схеми маніпулятора – на вертикальну площину його симетрії ($O\xi Y$, вверху) та на горизонтальну (OXY , внизу). Показано розміри – $l_1=AB$, $l_2=BC$, $d_3=CD$, $l_3=DK$, $l_4=CK$, кути – $\alpha, \beta, \gamma, \varphi_1, \varphi_2, \varphi_3$. Причому кути α, β – конструктивні, γ – проміжний, $\varphi_1, \varphi_2, \varphi_3$ – узагальнені координати.

Кут φ_1 легко знаходиться з рис. 3 нижньої проекції (використано функцію **atan2** для отримання кута в межах $(-\pi, +\pi)$)

$$\varphi_1 = \text{atan2}(Z_K, X_K); \quad (5)$$

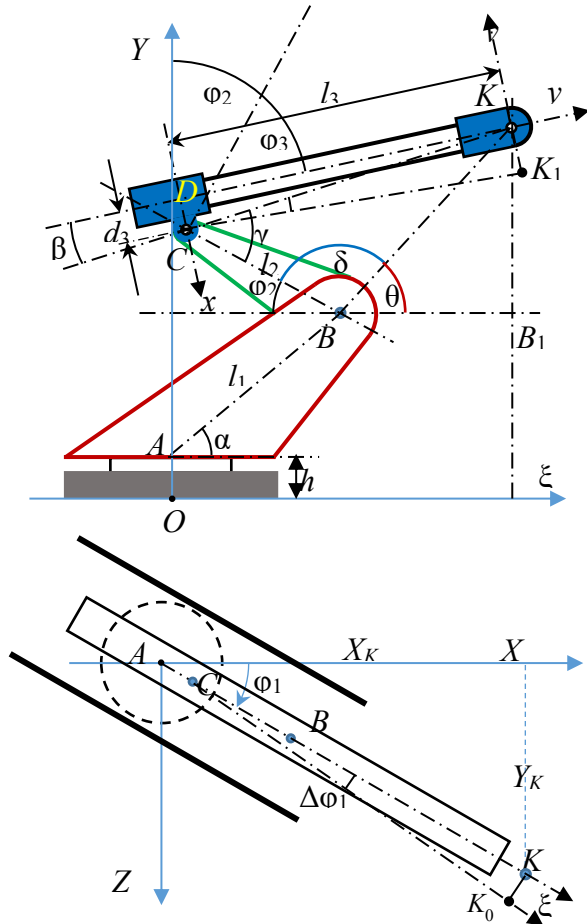


Рис. 3 – Проекції схеми робота на вертикальну (вверху) і горизонтальну площину

Для визначення кутів φ_2, φ_3 , що задають положення ланок BC та CDK , вирішимо трикутники ΔBB_1K та ΔBCK

$$BK^2 = (\xi_K - \xi_B)^2 + (Y_K - Y_B)^2; \quad \theta = \arctg\left(\frac{Y_K - Y_B}{\xi_K - \xi_B}\right);$$

$$\gamma = \arccos\left(\frac{l_2^2 + l_4^2 - BK^2}{2l_2l_4}\right); \quad \delta = \arccos\left(\frac{l_2^2 + BK^2 - l_4^2}{2l_2BK}\right);$$

$$\varphi_2 = \pi - \delta - \theta; \quad \varphi_3 = \pi/2 - \gamma + \beta. \quad (6)$$

$$\text{Тут } Y_B = h + l_1 \sin \alpha; \quad \xi_B = l_1 \sin \alpha; \quad \xi_K = \sqrt{X_K^2 + Z_K^2}.$$

Розв'язання орієнтаційної задачі.

Отримані кути $\varphi_1, \varphi_2, \varphi_3$ визначають, крім віддаленості т. K від стійки маніпулятора, орієнтацію у просторі ланки CDK (див. рис. 3). Запишемо вираз для її матриці повороту відповідно до послідовності поворотів на кути (5), (6) множенням матриць елементарних поворотів

$$\mathbf{S}_{(CDK)}^{(abs)} = \mathbf{S}_y(-\varphi_1) \mathbf{S}_z(-\varphi_2 - \varphi_3), \quad (7)$$

де позначення $\mathbf{S}_\xi(\varphi)$, $\xi=x, y, z$ – тут і далі означають матрицю елементарного повороту навколо відповідно однієї з осей x, y, z на кут, вказаний у дужках.

Виходячи з цього, відповідна матриця повороту матиме вигляд

$$\mathbf{S}_{(CDK)}^{(abs)} = \begin{bmatrix} \cos \varphi_1 \sin(\varphi_2 + \varphi_3) & -\cos \varphi_1 \cos(\varphi_2 + \varphi_3) & -\sin \varphi_1 \\ \cos(\varphi_2 + \varphi_3) & \sin(\varphi_2 + \varphi_3) & 0 \\ \sin \varphi_1 \sin(\varphi_2 + \varphi_3) & -\sin \varphi_1 \cos(\varphi_2 + \varphi_3) & \cos \varphi_1 \end{bmatrix} \quad (8)$$

Для отримання орієнтації «захвату» систему координат 3-ї ланки треба повернути (див. рис. 1, 3) на кути φ_4 (у шарнірі CD) та φ_5 і φ_6 у шарнірах, осі яких проходять через точку. Система координат захвату $Gxyz$, яка при цьому отримується, позначена червоним кольором на рис. 1. Її треба ще послідовно повернути щодо осі Gy на 90° та – нового положення осі Gx теж на 90° . Тоді її орієнтація збігається з орієнтацією системи координат захвату, орієнтація якої була задана в абсолютній СК з умов робочого процесу матрицею $\mathbf{S}_{(Dxyz)}^{(abs)}$ (1). Матриця $\mathbf{S}_{(Dxyz)}^{(abs)}$ може визначатися у робочому процесі, або через кватерніон (4). Для розв'язання орієнтаційної задачі, очевидно треба отримати матрицю, яка сформується послідовними поворотами на кути $\varphi_4, \varphi_5, \varphi_6$. Позначимо її так

$$\mathbf{S}_{(Gxyz)}^{(CDK)} = \mathbf{S}_y(\varphi_4) \mathbf{S}_z(\varphi_5) \mathbf{S}_y(\varphi_6) \quad (9)$$

Запишемо рівняння матриць повороту від абсолютної до системи координат $Gxyz$. З одного боку – це поворот від системи $Kxyz$ на кути -90° щодо осі Kx та -90° щодо нового положення осі Gy . З другого боку – це послідовність поворотів на кути $\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6$. В наших позначеннях це буде матричне рівняння

$$\mathbf{S}_{(CDK)}^{(abs)} \mathbf{S}_{(Gxyz)}^{(CDK)} = \mathbf{S}_{(Dxyz)}^{(abs)} \mathbf{S}_x(-\pi/2) \mathbf{S}_y(-\pi/2)$$

Звідси, з урахуванням (8),

$$\mathbf{S}_{(Gxyz)}^{(CDK)} = \left[\mathbf{S}_{(CDK)}^{(abs)} \right]^T \mathbf{S}_{(Dxyz)}^{(abs)} \mathbf{S}_x(-\pi/2) \mathbf{S}_y(-\pi/2). \quad (10)$$

Тут вигідно отримати вираз не для матриці (9) – лівої частині виразу (10), а для матриці

$$\mathbf{S}_{(Dxyz)}^{(CDK)} = \mathbf{S}_{(Gxyz)}^{(CDK)} \mathbf{S}_y(\varphi_2 + \varphi_3) \mathbf{S}_x(\varphi_1) \quad (11)$$

так як вона може рахуватися простіше

$$\mathbf{S}_{(Dxyz)}^{(CDK)} = \left[\mathbf{S}_{(CDK)}^{(abs)} \right]^T \mathbf{S}_{(Dxyz)}^{(abs)}.$$

Згідно з формулою (10) вона має наступну структуру

$$\mathbf{S}_{(Dxyz)}^{(CDK)} = \begin{bmatrix} S_{\varphi_5} C_{\varphi_6} & C_{\varphi_5} & S_{\varphi_5} S_{\varphi_6} \\ S_{\varphi_4} C_{\varphi_5} C_{\varphi_6} + C_{\varphi_4} S_{\varphi_6} & -S_{\varphi_4} S_{\varphi_5} & S_{\varphi_4} C_{\varphi_5} S_{\varphi_6} - C_{\varphi_4} C_{\varphi_6} \\ -(C_{\varphi_4} C_{\varphi_5} C_{\varphi_6} - S_{\varphi_4} S_{\varphi_6}) & C_{\varphi_4} S_{\varphi_5} & -(C_{\varphi_4} C_{\varphi_5} S_{\varphi_6} + S_{\varphi_4} C_{\varphi_6}) \end{bmatrix}. \quad (12)$$

Звідси знаходимо всі кути

$$\varphi_5 = \arccos(s_{12}); \quad \varphi_4 = -\text{atan2}(s_{22}, s_{32}); \quad \varphi_6 = \text{atan2}(s_{13}, s_{11}), \quad (13)$$

де в дужках стоятиме елементи матриці (12).

Формули (13) мають сенс тільки для кута $\varphi_5 \neq 0$ та $\varphi_5 \neq \pm\pi$. Конструктивно, можливе тільки значення $\varphi_5 = 0$. Тоді матриця (12) перетвориться до виду

$$\mathbf{S}_{(xyz)}^{CDK} = \begin{bmatrix} 0 & 1 & 0 \\ \sin(\varphi_4 + \varphi_6) & 0 & -\cos(\varphi_4 + \varphi_6) \\ -\cos(\varphi_4 + \varphi_6) & 0 & -\sin(\varphi_4 + \varphi_6) \end{bmatrix}. \quad (14)$$

Це означає, що осі кінематичних пар №4 та №6 стануть тотожними, або паралельними (див. далі). В такому випадку зі значень елементів матриці (14) можна визначити суму кутів

$$\varphi_4 + \varphi_6 = \text{atan2}(s_{13}, s_{33}). \quad (15)$$

повороту абсолютної СК OXYZ до СК «схвата» Kхуz представляється в цьому випадку твором матриці повороту від СК OXYZ до СК B2хуz «ліктя» та матриці повороту від СК B2хуz «ліктя» до СК Kхуz «схвата».

Залежність кутів повороту кривошипу та ланки №3 відносно ланки №2

Отримані формули (5), (6), (13), (15) визначають, взагалі кажучи, кути між ланками в кінематичних парах, що їх з'єднують. Але в цьому маніпуляторі, що тут розглядається є одна особливість, що потребує її розглянути окремо.

На рис. 4 показано розташування чотириланки, що приводить до руху ланку №3. Їїго ведуча ось A змонтована ні на ланці №2, а на ланці №1.

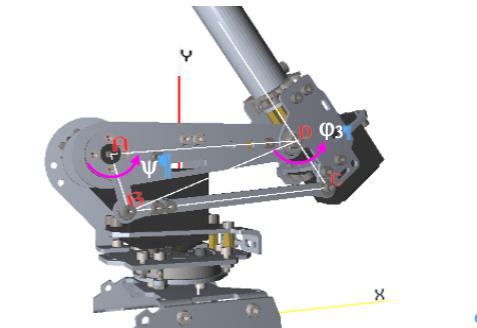


Рис. 4 – Чотириланковий механізм

Таким чином потрібно розв'язати задачу визначення кута ψ повороту кривошипа AB, який забезпечить визначений в результаті розв'язання транспортної задачі кут φ_3 . Тобто потрібно побудувати залежність $\psi(\varphi_3)$.

Розглянемо рис. 5, де показано два положення чотириланки ABCD: початкове – AB_0C_0D , відповідного куту повороту $\varphi_3 = 0$, і поточне, ABCD, що визначається довільним значенням кута φ_3 . Нехай задані розміри ланок чотириланки (в мм):

$$AB = r = 25,96;$$

$$BC = b = 104,33;$$

$$CD = R = 29,27;$$

$$AD = d = 95,00.$$

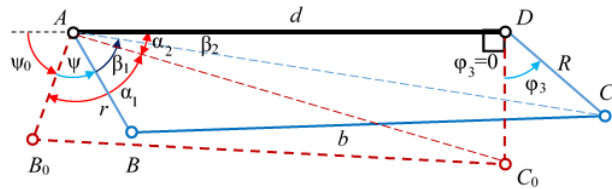


Рис. 5 – До визначення кінематики чотириланки

Знайдемо кут ψ_0 , що відповідає «нульовому» ($\varphi_3 = 0$) положенню (червоний колір, див. рис. 5):

$$\alpha_2 = \arctg\left(\frac{R}{d}\right), \quad AC_0 = \sqrt{d^2 + R^2},$$

$$\alpha_1 = \arccos\left(\frac{r^2 + d^2 + R^2 - b^2}{2r \cdot AC_0}\right), \quad \psi_0 = \pi - \alpha_1 - \alpha_2.$$

Після чого знайдемо кут ψ (або $\psi + \psi_0$, див. рис. 5), що відповідає будь-якому значенню кута φ_3

$$AC = \sqrt{d^2 + R^2 + 2d \cdot R \sin \varphi_3}, \quad \beta_1 = \arccos\left(\frac{AC^2 + r^2 - b^2}{2AC \cdot r}\right),$$

$$\beta_2 = \arccos\left(\frac{AC^2 + d^2 - R^2}{2AC \cdot d}\right), \quad \psi = \pi - \psi_0 - \beta_1 - \beta_2.$$

Випадок відсутності перетину осі 4-ї кінематичної пари з точкою перетину осей 5-ї та 6-ї.

В практиці розрахунків кінематики маніпуляторів такого класу, які тут розглядаються, зустрічаються випадки, коли ось повороту 3-ї ланки проходить поруч точки K – точки перетину, через яку проходять осі 5-ї та 6-ї (рис. 6).

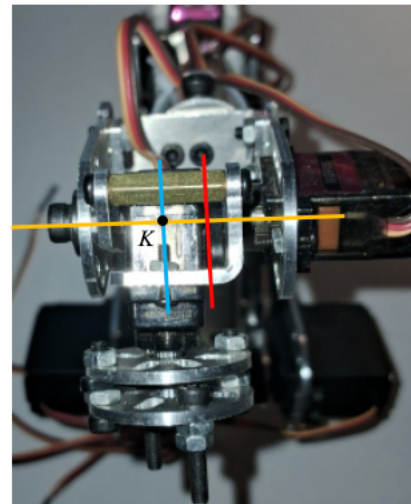


Рис. 6 – Вигляд вздовж 4-ї ланки на т. K

Це дещо ускладнює розглянутий алгоритм. Зупинимося на цьому.

Вказаний факт приводить до того, що при розв'язанні транспортної задачі фізичну точку K – точку перетину осей 5-ї та 6-ї ланок – треба поєднати з визначеною розрахунками з урахуванням такого неузгодження. І це неузгодження визначається цілком кутом повороту 4-ї

ланки, який на даний момент є невідомим. І це дає проблему.

Розглянемо рис. 7, де показана ось ординат Dz_3 системи координат 3-ї ланки (див. рис. 3), т. K – вузлова точка в умовах відсутності неузгодження, т. K_1 – існуюче положення вузлової точки, φ_4 – кут повороту системи координат 4-ї ланки щодо 3-ї, e – відстань між точками K і K_1 , Kx_3 , Kz_3 – паралельні осі осям 3-ї ланки.

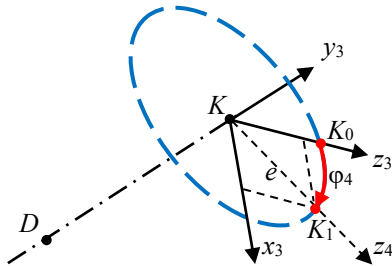


Рис. 7 – До ліквідації неузгодження тт. K

Тут слід підкреслити, що в результаті поворотів перших трьох ланок вісь Kz_3 залишиться горизонтальною. Тим самим проекція відстані KK_1 на цю ось ($z_{K_1} = e \cos \varphi_4$) буде теж горизонтальною (див. рис. 3 внизу). Друга проекція на вертикальну площину ($x_{K_1} = e \sin \varphi_4$) приводить до зміни відстані точки K від точки A та її аплікати в абсолютній системі координат. Все це приводить до того, що змінюються формули для кутів θ , δ , γ та відстань BK вони стають функціями від кута φ_4 .

$$\begin{aligned} \varphi_1 &= \operatorname{atan2}(Z_K, X_K) - 2 \operatorname{arctg} \frac{e \sin \varphi_4}{AK} \\ \varphi_2 &= \pi - \delta(\varphi_4) - \theta(\varphi_4); \\ \varphi_3 &= \pi/2 - \gamma(\varphi_4) + \beta; \end{aligned} \quad (16)$$

Це, в свою чергу, веде до того, що елементи матриці $S_{(CDK)}^{(abs)}$ теж стають функціями кута φ_4 . І ми приходимо до того, формули (13) набувають вигляду

$$\begin{aligned} \varphi_5 &= \arccos(s_{12}(\varphi_4)); \\ \varphi_4 &= -\operatorname{atan2}(s_{22}(\varphi_4), s_{32}(\varphi_4)); \\ \varphi_6 &= \operatorname{atan2}(s_{13}(\varphi_4), s_{11}(\varphi_4)), \end{aligned} \quad (17)$$

Тепер зрозуміло, що для остаточного розв'язання задачі оберненої кінематики у випадку неузгодженості осей останніх трьох ланок треба знайти корінь рівняння

$$\varphi_4 = -\operatorname{atan2}(s_{22}(\varphi_4), s_{32}(\varphi_4)),$$

після чого все знайдеться з побудованих формул (16) і (17).

3D моделювання та налагоджування математичної моделі. Для дослідження математичної моделі доцільно використовувати сучасні технології для 3D моделювання. Це дозволяє отримувати швидкі результати у порівнянні з використанням фізичної моделі. 3D моделювання дозволяє спростити аналіз та

усунення геометричних похибок у підрахунках, що на фізичних моделях призвело до руйнівних процесів.

Необхідною умовою при створенні подібних автоматизацій є можливість внесення та автоматична зміна кутів за допомоги табличних даних або мікропрограм-скриптів описуючих траєкторії руху.

Загальний вигляд головного екрану програми можна побачити на рисунку (3)

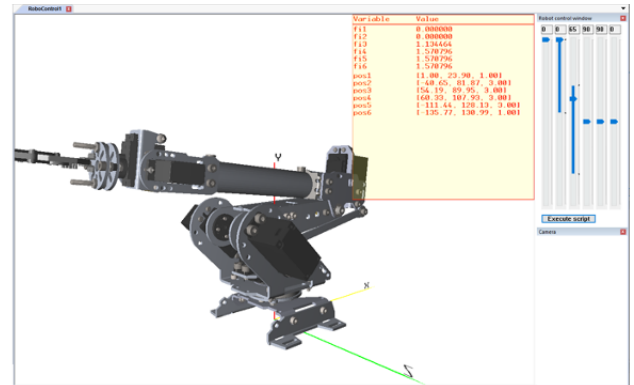


Рис. 8 – Загальний вигляд додатку 3D симуляції моделі робота

За допомоги панелі керування (правий верхній кут на рис. 3) є можливість змінювати кути в ручному режимі, змінюючи кути відповідних ланцюгів робота. Для розрахунку положення ланок використовується два методи, які можна обрати у інтерфейсі. Перший використовує підхід для розрахунку через кути Ейлера, другий використовує кватерніонну алгебру. Механізм робота, що тут розглядається, не може отримувати ефект так званого «шарнірного замку» або ефекту складання рамок, з причини фізичної конфігурації. Однак метод було додано у якості оптимізації обчислювальних операцій, та для подальшого розвитку.

Додатковою можливістю, є вбудований інтерфейс Python, який дозволяє користувачам писати та виконувати скрипти безпосередньо в середовищі без компіляції всього модуля. Ця функція дозволяє автоматизувати складні послідовності, де робот може виконувати завдання, такі як обертання та нахили на основі попередньо визначених скриптів. Спроможність написання скриптів є потужним інструментом для розробників, які прагнуть впровадити складні автоматизаційні процедури або симулювати різні сценарії перед впровадженням робота в реальні завдання.

Для інтерфейсу роботи скриптів використовується інжектований у середовище Python модуль. Через цей модуль відбувається зв'язок інтерпретованого коду з внутрішніми керуючими класами програми.

На рис. 9 зображено приклад керуючої програми на мові Python, яка виконує простий рух. Через модуль інтерфейсу **mathmodel** ми маємо доступ до відповідних кутів математичної моделі. Ми змінюємо кути ланок 1, 2 та 4 по лінійному закону. Пауза між ітераціями виконує роль затримки і потрібна для

завдання швидкості руху. У нашому випадку це 10 мс на 1° , або приблизно 1.745 рад/с .

```
import sys
import math
import mathmodel

def checkScript():
    print("PY> RoboControl script - OK")
    return True;

class CRoboControl:
    __refres_rate = 10
    # define method
    #
    def getRefreshRate():
        return __refres_rate;
    #
    # script loop
    #
    def RunScript():
        i = 0
        while (i < 70):
            mathmodel.chain1(i)
            mathmodel.chain2(i + 2)
            mathmodel.chain4(i)
            mathmodel.pause(20)
            i = i + 1
        return True
```

Рис. 9 – Приклад керуючої програми

Так як ми можемо використовувати всі можливості Python, то ці рухи можна моделювати довільним чином, наприклад задати табличною формою.

Наведемо приклад виконання подібного скрипту. На рис. 10, показані перетворення які відбулися в моделі роботу після переміщення з початкового положення, що показано на рис. 8.

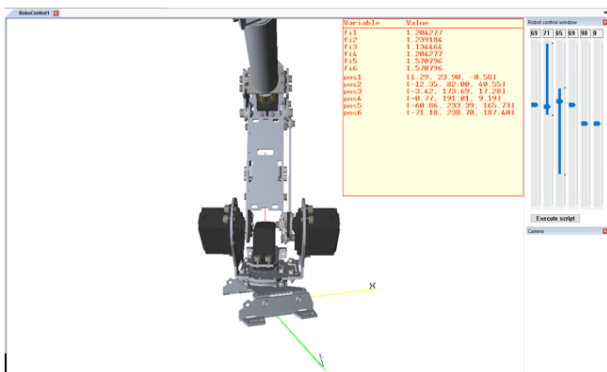


Рис. 10 – Кінцеве положення маніпулятора, що відбулося під впливом виконання скрипту.

Ми можемо бачити (див. рис. 8 та рис. 10), що відбулись зміни кутів у порівнянні зі встановленими. Відповідно до закладеної форми руху.

Іншою властивістю додатку можна означити можливість програмно-апаратного з'єднання з фізичним пристроєм. Між розробленою програмою та програмною частиною управляючого контролеру моделі робота існує протокол, який дозволяє передавати інформаційні пакети стану. Таким чином ті зміни які відбуваються з 3D моделлю передаються і

впливають на фізичні виконавчі механізми. Відповідність фізичної та математичної моделі встановлюється через крайні положення, виходячи з того, що двигун може змінювати свій кут у межах 180 градусів.

Обернена кінематика.

Для обчислення оберненої математики механізму у програму закладене визначення координати вказівника миші з подальшим використанням пошуку перетинання з об'єктом: алгоритм Моллера-Трумбора, або так званий алгоритм Ray-Picking. Для цього з вказівника миші будеться пряма яка перпендикулярна екранній площині. Таким чином ми отримуємо необхідний елемент моделі, який можна рухати.

На рис. 11 ми можемо бачити «доторкання» по елементу моделі, яка підсвічена фіолетовим кольором. Подальша логіка аналізує проекцію напрямку руху, таким чином передаючи напрямок зміни кутів в математичну модель.

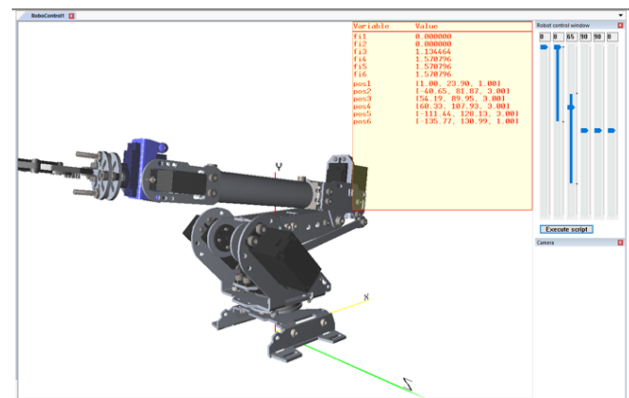


Рис.11 – Використання алгоритму Моллера-Трумбора для пошуку виділеного елемента

Висновки. У статті побудовано повністю аналітичне розв'язання оберненої задачі кінематики для просторового маніпулятора з шістьма степенями свободи. Такий алгоритм буде справедливим для всього класу роботів типу ПУМА, якщо використовувати розроблений підхід, а не отримані тут формули. Також розроблена програма реалізованих алгоритмів вирішення прямого завдання кінематики (що використовують рівняння Ейлера, або кватерніони з використанням можливостей OpenGL для тестування і дослідження моделі робота ABB).

Планується в наступній статті розкрити алгоритми кватерніонного представлення моделей маніпуляторів вказаного класу для розв'язання прямої кінематики.

Список літератури

1. S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, "Closed-form inverse kinematics for continuum manipulators," *Adv. Robot.*, vol. 23, no. 15, pp. 2077–2091, 2009. <https://doi.org/10.1163/016918609X12529299964101>
2. S. Kucuk and Z. Bingul, "Robot kinematics: Industrial robotics: forward and inverse kinematics," in *Industrial Robotics. Theory, Modelling and Control*, London, UK: INTECH OA Publ, 2006, pp. 117-148. <https://doi.org/10.5772/5015>

3. S. Kucuk and Z. Bingul, "The inverse kinematics solutions of industrial robot manipulators," in *Proc. IEEE Conf. on Mechatronics*, Istanbul, Turkey, June 2004, 2004, pp. 274-279. <https://doi.org/10.1109/ICMECH.2004.1364451>
4. F. Xiao, G. Li, D. Jiang et al., "An effective and unified method to derive the inverse kinematics formulas of general six-dof manipulator with simple geometry", *Mech. Mach. Theory*, vol. 159, pp. 104265, 2021. <https://doi.org/10.1016/j.mechmachtheory.2021.104265>
5. K. Harada, E. Yoshida, and K. Yokoi, "Motion Planning for Humanoid Robots", Springer, 2010. <https://doi.org/10.1007/978-1-84996-220-9>
6. K. Erleben, S. Andrews. "Solving inverse kinematics using exact Hessian matrices." *Computers & Graphics*. Feb; vol. 78, pp.1-11, 2019. <https://doi.org/10.1016/j.cag.2018.10.012>
7. S. Momani, Z. S., Z. S. Abo-Hammour, and O. M. Alsmadi, "Solution of inverse kinematics problem using genetic algorithms", *Appl. Math. Inf. Sci.*, vol.10, no.1, pp.225, 2016. <https://doi.org/10.18576/amis/100122>
8. A. Csiszar, J. Eilers, and A. Verl, "On solving the inverse kinematics problem using neural networks." in *Proc. 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, 2017, pp. 1-6. <https://doi.org/10.1109/M2VIP.2017.8211457>
9. Y. Andrijev, D. Breslavsky, S. Pashchenko and O. Tatarinova, "Development the Algorithms of Anthropomorphic Robot's Motion Control by Use of AI Algorithms", in *Proc IEEE KhPI Week on Advanced Technology, KhPI Week 2020*, Kharkiv, Ukraine, 2020, pp. 82-85. <https://doi.org/10.1109/KhPIWeek51551.2020.9250175>
10. L. Chen, T. Zielinska, J. Wang, and W. Ge, "Solution of an inverse kinematics problem using dual quaternions". *Int. J. Appl. Math. Comput.*, vol 30, no. 2, pp. 351-361, 2020. <https://doi.org/10.1109/KhPIWeek51551.2020.9250175>
11. A. El-Sherbiny, M. A. Elhosseni, A. Y. Haikal, "A comparative study of soft computing methods to solve inverse kinematics problem", *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 2535-2548, 2018. <https://doi.org/10.1016/j.asej.2017.08.001>
12. S. Xie, L. Sun, Z. Wang, and G. Chen, "A speedup method for solving the inverse kinematics problem of robotic manipulators", *Int. J. Adv. Robotic Syst.*, vol 19, no 3, pp.17298806221104602, 2022. <https://doi.org/10.1177/17298806221104602>
13. J. Denavit, R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices", *Int. Appl. Mech.*, vol. 22, no.2, pp. 215-221, 1955. <https://doi.org/10.1115/1.4011045>
14. Yu. M. Andreev and O. K. Morachkovsky, "Dynamics of holonomic systems of rigid bodies", *Int. Appl. Mech.*, vol. 41, pp. 817-824, 2005. <https://doi.org/10.1007/s10778-005-0150-0>
15. Yu. Andrijev, D. Breslavsky, A. Larin, and V. Mietelov, "Computer Modelling of UAV Flight", in *Proc 2021 IEEE 2nd KhPI Week on Advanced Technology (KhPIWeek)*, 13-17 Sept. 2021, Kharkiv, Ukraine, IEEE, 2021, pp. 455-459. <https://doi.org/10.1109/KhPIWeek53812.2021.9569991>
16. Yu. Andriev, D. Breslavsky, H. Chystilina, and K. Ivanchenko, "Algorithm for kinematic design of anthropomorphic robot's with a free foot walking control", in *Proc 2022 IEEE 3rd KhPI Week on Advanced Technology Conference (KhPIWeek)*, October 3-7, 2022, Kharkiv, Ukraine, IEEE, 2022, pp. 324-329. <https://doi.org/10.1109/KhPIWeek53812.2021.9569991>
17. H. S. Liu, W. N. Zhou, X. B. Lai, and S. Q. Zhu, "An efficient inverse kinematic algorithm for a PUMA560-structured robot manipulator, *International Journal of Advanced Robotic Systems*, vol. 10, 236, 2013. doi: <https://doi.org/10.5772/56403>.
18. Liu, Novel inverse kinematic approaches for robot manipulators with Pieper-Criterion based geometry, *Int. J. Control. Autom.*, № 13, c. 1242? 2015. doi: <https://doi.org/10.1007/s12555-013-0440-y>
19. Андреев Ю. М. Реализация и использование алгоритма Левенберга-Марквардта в задачах калибровки роботоманипуляторов / Ю. М. Андреев // Вісник НТУ «ХПІ». Серія: Динаміка та міцність машин. – Харків : НТУ «ХПІ», 2021. – № 2 (2021). – С. 86 – 93.
20. Андреев Ю. М. Новая система компьютерной алгебры для исследования колебаний структурно-сложных голономных и неголономных систем твердых тел / Ю. М. Андреев, О. К. Морачковский // Надежность и долговечность машин и сооружений: междунар. науч.-техн. сбор. НАН Украины. — К.: ИПП им. Писаренко Г. С., Ассоциация «Надежность машин и сооружений», 2006. — Вып. 26. — С. 11-18.

References (transliterated)

1. S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, "Closed-form inverse kinematics for continuum manipulators," *Adv. Robot.*, vol. 23, no. 15, pp. 2077-2091, 2009. <https://doi.org/10.1163/016918609X12529299964101>
2. S. Kucuk and Z. Bingul, "Robot kinematics: Industrial robotics: forward and inverse kinematics," in *Industrial Robotics. Theory, Modelling and Control*, London, UK: INTECH OA Publ, 2006, pp. 117-148. <https://doi.org/10.5772/5015>
3. S. Kucuk and Z. Bingul, "The inverse kinematics solutions of industrial robot manipulators," in *Proc. IEEE Conf. on Mechatronics*, Istanbul, Turkey, June 2004, 2004, pp. 274-279. <https://doi.org/10.1109/ICMECH.2004.1364451>
4. F. Xiao, G. Li, D. Jiang et al., "An effective and unified method to derive the inverse kinematics formulas of general six-dof manipulator with simple geometry", *Mech. Mach. Theory*, vol. 159, pp. 104265, 2021. <https://doi.org/10.1016/j.mechmachtheory.2021.104265>
5. K. Harada, E. Yoshida, and K. Yokoi, "Motion Planning for Humanoid Robots", Springer, 2010. <https://doi.org/10.1007/978-1-84996-220-9>
6. K. Erleben, S. Andrews. "Solving inverse kinematics using exact Hessian matrices." *Computers & Graphics*. Feb; vol. 78, pp.1-11, 2019. <https://doi.org/10.1016/j.cag.2018.10.012>
7. S. Momani, Z. S., Z. S. Abo-Hammour, and O. M. Alsmadi, "Solution of inverse kinematics problem using genetic algorithms", *Appl. Math. Inf. Sci.*, vol.10, no.1, pp.225, 2016. <https://doi.org/10.18576/amis/100122>
8. A. Csiszar, J. Eilers, and A. Verl, "On solving the inverse kinematics problem using neural networks." in *Proc. 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, 2017, pp. 1 - 6. <https://doi.org/10.1109/M2VIP.2017.8211457>
9. Y. Andrijev, D. Breslavsky, S. Pashchenko and O. Tatarinova, "Development the Algorithms of Anthropomorphic Robot's Motion Control by Use of AI Algorithms", in *Proc IEEE KhPI Week on Advanced Technology, KhPI Week 2020*, Kharkiv, Ukraine, 2020, pp. 82-85. <https://doi.org/10.1109/KhPIWeek51551.2020.9250175>
10. L. Chen, T. Zielinska, J. Wang, and W. Ge, "Solution of an inverse kinematics problem using dual quaternions". *Int. J. Appl. Math. Comput.*, vol 30, no. 2, pp. 351-361, 2020. <https://doi.org/10.1109/KhPIWeek51551.2020.9250175>
11. A. El-Sherbiny, M. A. Elhosseni, A. Y. Haikal, "A comparative study of soft computing methods to solve inverse kinematics problem", *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 2535-2548, 2018. <https://doi.org/10.1016/j.asej.2017.08.001>
12. S. Xie, L. Sun, Z. Wang, and G. Chen, "A speedup method for solving the inverse kinematics problem of robotic manipulators", *Int. J. Adv. Robotic Syst.*, vol 19, no 3, pp.17298806221104602, 2022. <https://doi.org/10.1177/17298806221104602>
13. J. Denavit, R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices", *Int. Appl. Mech.*, vol. 22, no.2, pp. 215-221, 1955. <https://doi.org/10.1115/1.4011045>
14. Yu. M. Andreev and O. K. Morachkovsky, "Dynamics of holonomic systems of rigid bodies", *Int. Appl. Mech.*, vol. 41, pp. 817-824, 2005. <https://doi.org/10.1007/s10778-005-0150-0>
15. Yu. Andrijev, D. Breslavsky, A. Larin, and V. Mietelov, "Computer Modelling of UAV Flight", in *Proc 2021 IEEE 2nd KhPI Week on Advanced Technology (KhPIWeek)*, 13-17 Sept. 2021, Kharkiv, Ukraine, IEEE, 2021, pp. 455-459. <https://doi.org/10.1109/KhPIWeek53812.2021.9569991>
16. Yu. Andriev, D. Breslavsky, H. Chystilina, and K. Ivanchenko, "Algorithm for kinematic design of anthropomorphic robot's with a free foot walking control", in *Proc 2022 IEEE 3rd KhPI Week on Advanced Technology Conference (KhPIWeek)*, October 3-7, 2022, Kharkiv, Ukraine, IEEE, 2022, pp. 324-329. <https://doi.org/10.1109/KhPIWeek53812.2021.9569991>
17. H. S. Liu, W. N. Zhou, X. B. Lai, and S. Q. Zhu, "An efficient inverse kinematic algorithm for a PUMA560-structured robot manipulator, *International Journal of Advanced Robotic Systems*, vol. 10, 236, 2013. doi: <https://doi.org/10.5772/56403>.
18. Liu, Novel inverse kinematic approaches for robot manipulators with Pieper-Criterion based geometry, *Int. J. Control. Autom.*, № 13, c. 1242? 2015. doi: <https://doi.org/10.1007/s12555-013-0440-y>
19. Andreev Yu. M. Realizatsiya i ispol'zovanie algoritma Levenberga-Markvarda v zadachakh kalibrovki robotov-manipulyatorov /

- Yu. M. Andreev // Visnik NTU "KhPI". Seriya: Dinamika i mitsnost' mashin. - Kharkiv : NTU "KhPI", 2021. - № 2 (2021). - S. 86 – 93.
20. Andreev Yu. M. Novaia systema kompiuternoї alhebyr dlia yssledovanyia kolebanyı strukturno-slozhnykh holonomnykh y nehonomnykh system tverdykh tel / Yu. M. Andreev, O. K. Morachkovskiy // Nadezhnost y dolhovechnost mashyn y

sooruzheniy : mezhdunar. nauch.-tekh. sbor. NAN Ukrayny. — K.: YPP ym. Pysarenko H. S., Assotsyatsiya «Nadezhnost mashyn y sooruzheniy», 2006. — Vyp. 26. — S. 11-18.

Надійшла (received) 08.12.2024

Відомості про авторів / About the Authors

Андрєєв Юрій Михайлович – доктор технічних наук, професор, професор кафедри комп'ютерного моделювання процесів та систем, Національний технічний університет «Харківський політехнічний інститут», м. Харків; Україна; ORCID: <https://orcid.org/0000-0003-3213-8496>; e-mail: andrjejev@gmail.com.

Andrieiev Yuriy – Doctor of Technical Sciences, Professor, Professor of the Department of Computer Modeling of Processes and Systems, National Technical University "Kharkiv Polytechnic Institute", Kharkiv; Ukraine; ORCID: <https://orcid.org/0000-0003-3213-8496>; e-mail: andrjejev@gmail.com.

Шабанов Геннадій Вікторович – аспірант кафедри комп'ютерного моделювання процесів і систем, Національний технічний університет «Харківський політехнічний інститут», м. Харків; тел.: (050) 596-88-86; e-mail: gennady.shabanov@gmail.com ORCID: <https://orcid.org/0009-0003-8555-4693>

Shabanov Hennadii Viktorovich – aspirant at the Department of Computer Modeling of Processes and Systems, National Technical University "Kharkiv Polytechnic Institute", Kharkiv; tel.: (050) 596-88-86; e-mail: gennady.shabanov@gmail.com.

Бібліографічні описи / Bibliographic descriptions

Шабанов Г. В. Аналітичне рішення оберненого і прямого завдання кінематики просторового кутового маніпулятора АBB з подальшим 3D-моделюванням / Г.В. Шабанов, Ю. М. Андрєєв // Вісник НТУ «ХПІ». Серія: Динаміка та міцність машин. – Харків : НТУ «ХПІ», 2024. – № 2 (2024). – С. XX – XX. Бібліогр.: 20 назв. – ISSN 2078-9130.

Shabanov H. Analytical solution of the inverse and direct kinematics problem for the spatial angular manipulator ABB with subsequent 3D modeling / H. Shabanov, Yu. Andrieiev // Bulletin of National Technical University «KhPI» Series: Dynamics and strength of machines. – Kharkiv : NTU «KhPI», 2024. – № 2 (2024). – pp. XX – XX. Bibliog.: 20 titles. – ISSN 2078-9130.