

Adam SŁOTA¹

WYZNACZANIE WSPÓŁRZĘDNYCH ZŁĄCZOWYCH DLA SKOORDYNOWANYCH RUCHÓW ROBOTÓW CZTEROOSIOWYCH

W pracy przedstawiono algorytm generowania trajektorii dla zadania transportowego realizowanego przez dwa roboty. Wyznaczone w układach współrzędnych zadania, położenia i orientacje punktów TCP robotów przekształcono do układów współrzędnych robotów, dla przyjętego ustawienia robotów na stanowisku. Dla wybranych modeli robotów czteroosiowych rozwiązano zadanie odwrotne. Uzyskane rozwiązanie zaimplementowano i zweryfikowano w systemie LabVIEW. Przedstawiono rozwiązanie dla wybranego zadania i konfiguracji stanowiska.

CALCULATION OF JOINT COORDINATES FOR COORDINATED MOTION OF FOUR AXES ROBOTS

In the paper an algorithm of generation trajectories for transport tasks executed by two robots is presented. Calculated in task coordinated systems, positions and orientations of robots' TCP are transformed into robots' coordinated systems for defined robots' positions. For selected model of four axes robots inverse kinematics is solved. Obtained solution is implemented and verified in LabVIEW system. The solution of a sample transport task and robot cell configuration is presented.

1. WSTĘP

Zastosowania przemysłowe robotów wymagają, poza automatycznym sterowaniem poszczególnymi robotami w celu wykonania wybranych fragmentów procesu technologicznego, koordynacji czasowej zadań realizowanych przez różne roboty w gnieździe. Koordynacja taka może mieć charakter dyskretny – rozpoczęcie danego zadania warunkowane jest określonym stanem urządzeń gniazda, zakończenie realizacji zadania (lub jego fragmentu) skutkuje zmianą stanu. Koordynacja taka może być zrealizowana za pomocą dyskretnych sygnałów wejścia/wyjścia. Zadania takie jak transport dużego i ciężkiego przedmiotu przed dwa roboty, czy spawanie z udziałem pozycjonera (lub robota przemysłowego pełniącego rolę pozycjonera) wymagają ciągłej koordynacji ruchów robotów. W tej dziedzinie rozwijane są dwa podejścia. Pierwsze to sterowanie robotami wymagającymi ciągłej koordynacji ruchów za pomocą jednego układu sterowania.

¹ Politechnika Krakowska, Wydział Mechaniczny, Zakład Zautomatyzowanych Systemów Produkcyjnych
31-864 Kraków, Al. Jana Pawła II 37. Tel: + 48 12 374-32-50, Fax: + 48 12 374-32-02,
e-mail: slota@mech.pk.edu.pl

Rozwiązania takie dostarczane są przez producentów robotów (Fanuc [4], ABB [1]). W tym przypadku jeden układ sterowania realizuje interpolację ruchów np. dwunastu osi napędowych. Podejście drugie to bezpośrednie sterowanie robotami przez oddzielne układy sterowania, z uwzględnieniem sprzężenia pomiędzy układami sterowania w celu zapewnienia wymaganej koordynacji ruchów.

Dotychczasowe prace autora dotyczyły koordynacji trajektorii dwóch robotów wykonujących wspólnie transport jednego przedmiotu. Założono, że obydwie roboty sterowane są przez oddzielne układy sterowania, które w celu zapewnienia koordynacji ruchów punktów *TCP* robotów wykorzystują sprzężenie położeniowe.

Opracowany algorytm wyznacza, dla zadanych pozycji początkowych i docelowych punktów *TCP* robotów, trajektorie punktów *TCP* robotów. Kolejne pozycje punktów trajektorii wyrażone są poprzez współrzędne położenia oraz orientacji. Położenie określone jest w kartezjańskim układzie współrzędnych, orientację określają kąty obrotu wokół stałych osi układu współrzędnych. Opracowany algorytm został zaimplementowany i zweryfikowany w systemie *LabVIEW* z wykorzystaniem modułu *Simulation Module*. W algorytmie wyznaczania trajektorii nie uwzględniono konfiguracji stanowiska zrobotyzowanego. W celu weryfikacji realizowalności otrzymanych trajektorii na stanowisku rzeczywistym zaproponowano wykorzystanie systemu Delmia [3]. System ten udostępnia biblioteki modeli robotów przemysłowych różnych producentów, umożliwia konfigurację stanowiska zrobotyzowanego, definiowanie zadań dla robotów, weryfikację realizowalności zadań i wygenerowanie programów sterujących. W pracy [5] przedstawiono procedurę przeniesienia uzyskanych w programie *LabVIEW* i zapisanych w plikach tekstowych współrzędnych położenia i orientacji punktów *TCP* do systemu Delmia. Dane te posłużyły do zdefiniowania zadań robotów dla przyjętej konfiguracji stanowiska z wykorzystaniem robotów czteroosiowych. Weryfikację działania stanowiska w środowisku wirtualnym dla robotów sześciuosiowych opisano w pracy [2].

Zaproponowany sposób weryfikacji algorytmu generowania skoordynowanych ruchów robotów jest sprawdzeniem offline. Sprawdzane są wygenerowane uprzednio trajektorie. Wykryte w ten sposób niezgodności z możliwościami rzeczywistego stanowiska zrobotyzowanego powodują konieczność ponownego wygenerowania trajektorii dla zmodyfikowanych parametrów i kolejnej weryfikacji wyników. Aby uniknąć takiego postępowania, należy uzupełnić algorytm generowania skoordynowanych ruchów robotów o możliwość weryfikacji rozwiązania w trybie online. Ponieważ podstawowe ograniczenia dotyczące ruchów robotów określone są w przestrzeni złączowej (zakresy ruchów, prędkości, przyspieszenia), dlatego konieczne jest wyrażenie pozycji punktów trajektorii we współrzędnych złączowych.

Prezentowana praca ma na celu uzupełnienie modelu o możliwość definiowania konfiguracji stanowiska (wzajemne położenie i orientacja robotów stanowiska, lokalizacja pozycji początkowych i docelowych punktów *TCP* robotów) oraz rozbudowa algorytmu wyznaczania skoordynowanych ruchów robotów o moduł rozwiązywania zadania odwrotnego dla robotów czteroosiowych.

W rozdziale 2.1 przedstawiono zarys zaproponowanego algorytmu koordynacji ruchów w układzie współrzędnych zadania. Przekształcenie współrzędnych trajektorii z układu zadania do układu współrzędnych robota zamieszczono w rozdziale 2.2, a w rozdziale 2.3 przedstawiono rozwiązanie zadania odwrotnego kinematyki dla robota czteroosiowego ZD 130S firmy Kawasaki.

2. GENEROWANIE SKOORDYNOWANYCH RUCHÓW ROBOTÓW

2.1 Algorytm generowania trajektorii we współrzędnych kartezjańskich

Trajektoria punktów *TCP* obydwu robotów wyznaczana jest przez ten sam algorytm – jedna instancja algorytmu wylicza współrzędne pozycji punktu *TCP* robota *A*, druga – współrzędne punktu *TCP* robota *B*. Algorytm wyznacza współrzędne kartezjańskie położenia punktu *TCP* robota *A* (*B*) $A_t^{ZA}(x_{tA}^{ZA} y_{tA}^{ZA} z_{tA}^{ZA})$ ($B_t^{ZB}(x_{tB}^{ZB} y_{tB}^{ZB} z_{tB}^{ZB})$) w układzie współrzędnych zadania robota *A* (*B*) $X_{ZA} Y_{ZA} Z_{ZA}$ ($X_{ZB} Y_{ZB} Z_{ZB}$). Przyjęto, że układy zadania obydwu robotów mają taką samą orientację, a ich wzajemne położenie określone jest danymi wejściowymi. Założono, że dla algorytmu wyznaczającego trajektorię punktu *TCP* jednego robota dostępne są współrzędne położenia i orientacji punktu *TCP* drugiego robota.

Kolejne współrzędne położenia $A_t^{ZA}(x_{tA}^{ZA} y_{tA}^{ZA} z_{tA}^{ZA})$ ($B_t^{ZB}(x_{tB}^{ZB} y_{tB}^{ZB} z_{tB}^{ZB})$) wyznaczane są jako złożenie dwóch ruchów:

- ruchu zaprogramowanego z położenia bieżącego do położenia docelowego (w aktualnej wersji algorytmu ruch zaprogramowany to ruch wzdłuż linii prostej ze stałą prędkością V_A^P (V_B^P)),
- ruchu korekcyjnego wzdłuż kierunku łączącego punkty *TCP* obydwu robotów.

Celem ruchu korekcyjnego jest minimalizacja zmiany odległości pomiędzy punktami *TCP* robotów. Odległość referencyjna, która ma być zachowana, to odległość pomiędzy punktami *TCP* $A_{T0}^{ZA}(x_{TA0}^{ZA} y_{TA0}^{ZA} z_{TA0}^{ZA})$ oraz $B_{T0}^{ZB}(x_{TB0}^{ZB} y_{TB0}^{ZB} z_{TB0}^{ZB})$ w chwili początkowej. Ze względu na „nieznajomość” celu drugiego robota, do wyliczenia prędkości ruchu korekcyjnego wykorzystano strukturę układu automatycznej regulacji, ze zmianą odległości pomiędzy punktami *TCP* jako uchybem. Wartość prędkości ruchu korekcyjnego robota *A* (*B*) V_A^C (V_B^C) obliczana jest na podstawie zmiany odległości pomiędzy punktami *TCP* $\Delta l(t)$ jako suma dwóch składników: części proporcjonalnej do wyliczonej zmiany odległości $\Delta l(t)$ oraz części proporcjonalnej do całki ze zmiany odległości $\Delta l(t)$.

W celu wyznaczenia orientacji punktów *TCP* robotów przyjęto takie ustawienie wykorzystywanych robotów czteroosiowych, że orientacja punktu *TCP* robota może być zmieniana tylko przez obrót wokół osi Z_{ZA} (Z_{ZB}). Orientacja ta jest wyrażona poprzez kąt γ_{tA}^{ZA} (γ_{tB}^{ZB}), jaki z osią X_{ZA} (X_{ZB}) tworzy rzut wektora $B_t^{ZA} - A_t^{ZA}$ ($A_t^{ZB} - B_t^{ZB}$) na płaszczyznę $X_{ZA} Y_{ZA}$ ($X_{ZB} Y_{ZB}$). Wyznaczone według przedstawionego powyżej algorytmu położenia punktów *TCP* robotów *A* (*B*) A_t^{ZA} (B_t^{ZB}) oraz początkowa orientacja punktów *TCP* γ_{TA0}^{ZA} (γ_{TB0}^{ZB}) względem rzutu wektora $B_t^{ZA} - A_t^{ZA}$ ($A_t^{ZB} - B_t^{ZB}$) wykorzystywane są do wyliczenia wartości kątów γ_{tA}^{ZA} i γ_{tB}^{ZB} .

Dane wejściowe algorytmu to:

- położenie i orientacja początkowa punktu *TCP* robota *A* (*B*) - $A_{T0}^{ZA}(x_{TA0}^{ZA} y_{TA0}^{ZA} z_{TA0}^{ZA})$ i γ_{TA0}^{ZA} ($B_{T0}^{ZB}(x_{TB0}^{ZB} y_{TB0}^{ZB} z_{TB0}^{ZB})$ i γ_{TB0}^{ZB}),
- położenie i orientacja docelowa punktu *TCP* robota *A* (*B*) - $A_{TD}^{ZA}(x_{TAD}^{ZA} y_{TAD}^{ZA} z_{TAD}^{ZA})$ i γ_{TAD}^{ZA} ($B_{TD}^{ZB}(x_{TBD}^{ZB} y_{TBD}^{ZB} z_{TBD}^{ZB})$ i γ_{TBD}^{ZB}),

- zaprogramowana prędkość ruchu punktu TCP robota A (B) - V_A^p (V_B^p)
- parametry korektora trajektorii robota A (B) – współczynnik wzmocnienia k_A (k_B), czas zdwojenia T_A (T_B).

2.2 Konfiguracja stanowiska

W celu przekształcenia wygenerowanych współrzędnych trajektorii skoordynowanych ruchów w układach kartezjańskich zadania na współrzędne złączowe robotów należy określić konfigurację stanowiska. Ustawienie robota A (B) w gnieździe określone jest przez układ współrzędnych $X_{RA}Y_{RA}Z_{RA}$ ($X_{RB}Y_{RB}Z_{RB}$). Położenie układu współrzędnych zadania względem układu współrzędnych robota A (B) określa przesunięcie $X_{ZA}^{RA}Y_{ZA}^{RA}Z_{ZA}^{RA}$ ($X_{ZB}^{RB}Y_{ZB}^{RB}Z_{ZB}^{RB}$). Przyjęto, że orientacja układu współrzędnych zadania względem układu współrzędnych robota A (B) określona jest przez kąt obrotu wokół osi Z_{RA} (Z_{RB}) - γ_{ZA}^{RA} (γ_{ZB}^{RB}). Dla tak określonej konfiguracji stanowiska wyznaczane są współrzędne trajektorii w układzie robota – zależności od (1) do (4).

$$x_{TA}^{RA} = x_{ZA}^{RA} + x_{TA}^{ZA} \cos \gamma_{ZA}^{RA} - y_{TA}^{ZA} \sin \gamma_{ZA}^{RA} \quad (1)$$

$$y_{TA}^{RA} = y_{ZA}^{RA} + x_{TA}^{ZA} \sin \gamma_{ZA}^{RA} + y_{TA}^{ZA} \cos \gamma_{ZA}^{RA} \quad (2)$$

$$z_{TA}^{RA} = z_{ZA}^{RA} + z_{TA}^{ZA} \quad (3)$$

$$\gamma_{TA}^{RA} = \gamma_{ZA}^{RA} + \gamma_{TA}^{ZA} \quad (4)$$

2.3 Wyznaczanie współrzędnych złączowych

W pracy przyjęto, że zadanie transportu realizowane jest przez dwa identyczne roboty czteroosiowe firmy Kawasaki ZD130S. Schemat kinematyczny robota przedstawiono na rysunku 1.

Wartości współrzędnych złączowych określają zależności od (5) do (8):

$$\varphi_1 = \arctan \frac{y_{TA}^{RA}}{x_{TA}^{RA}} \quad (5)$$

$$\varphi_2 = \arccos \frac{l_1^2 - l_2^2 - d^2}{2l_1d} + \arctan \frac{z_{TA}^{RA} - l_4 + l_6}{x_{TA}^{RA} - l_3 - l_5} - \frac{\pi}{2} \quad (6)$$

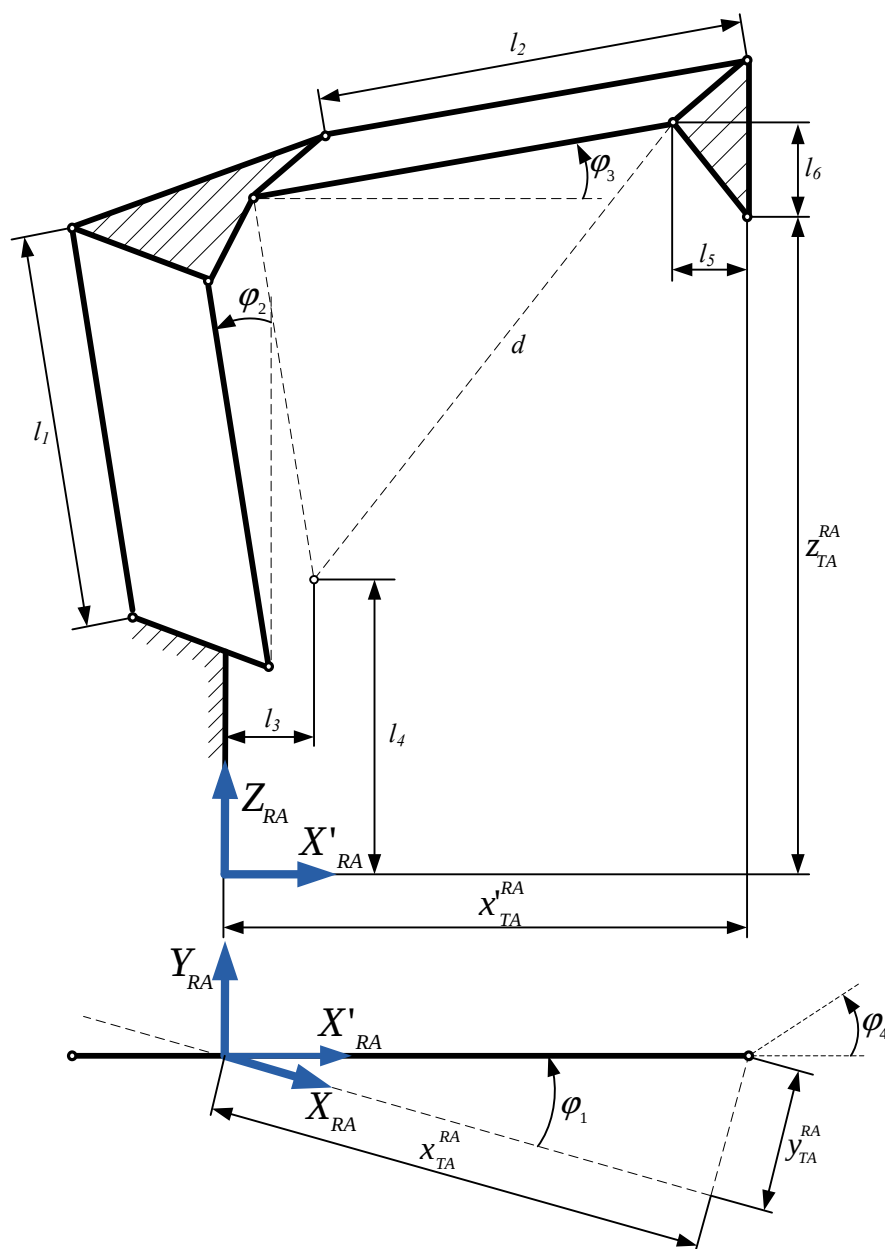
$$\varphi_3 = \arccos \frac{l_1^2 + l_2^2 - d^2}{2l_1l_2} + \varphi_2 - \frac{\pi}{2} \quad (7)$$

$$\varphi_4 = \gamma_{TA}^{RA} - \varphi_1 \quad (8)$$

gdzie:

$$d = \sqrt{(x_{TA}^{RA} - l_3 - l_5)^2 + (z_{TA}^{RA} + l_4 + l_6)^2} \quad (9)$$

$$x_{TA}^{RA} = \sqrt{x_{TA}^{RA}^2 + y_{TA}^{RA}^2} \quad (10)$$



Rys.1. Schemat układu kinematycznego robota ZD130S

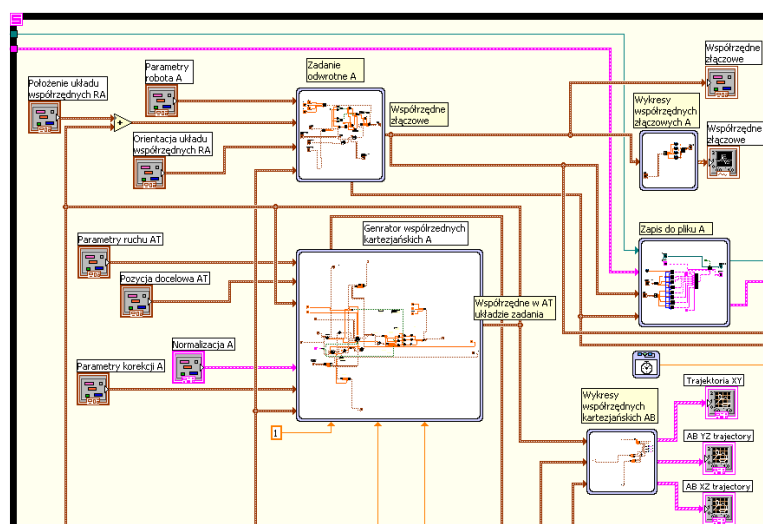
Wartości parametrów występujących w zależnościach (2) oraz (3) dla wybranego modelu ZD130S przedstawiono w tabeli 1.

Tab. 1. Wartości parametrów robota

Parametr	l_1	l_2	l_3	l_4	l_5	l_6
Wartość [mm]	1250	1450	300	940	255	300

3. PRZYKŁAD

Przedstawione w punkcie 2.3 zależności zostały wprowadzone do algorytmu wyznaczania skoordynowanych trajektorii. Na rysunku 2 przedstawiono fragment schematu blokowego algorytmu wyznaczania współrzędnych kartezjańskich i złączowych dla robota A, zbudowanego w module *Simulation Module* systemu *LabVIEW*.



Rys.2. Schemat blokowy algorytmu w systemie LabVIEW

W modelu występują bloki:

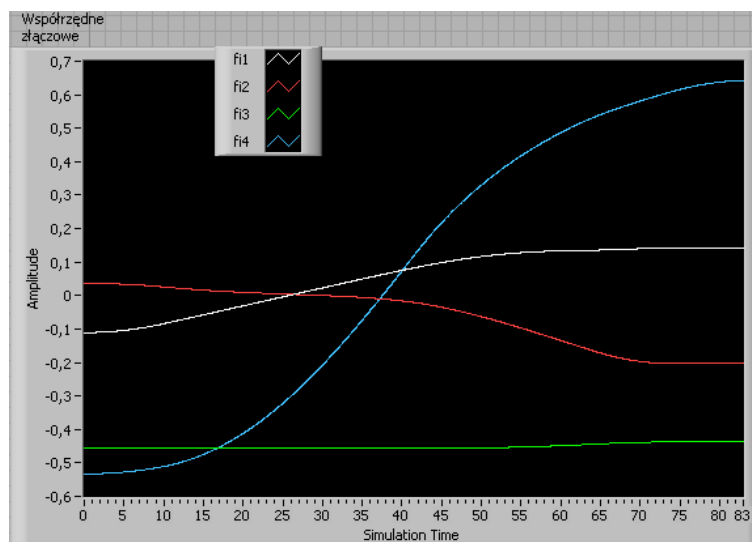
- *Generator współrzędnych kartezjańskich* – blok ten wyznacza trajektorie skoordynowanych ruchów punktów *TCP* robotów w ich układach współrzędnych zadania,
- *Zadanie odwrotne* – blok odpowiedzialny za przekształcenie współrzędnych pozycji *TCP* z układu współrzędnych zadania do układu współrzędnych robota, zgodnie z zależnościami od (1) do (4) oraz rozwiązania zadania odwrotnego według wzorów od (5) do (8),
- *Wykresy współrzędnych kartezjańskich* – blok służący do przedstawienia trajektorii w układzie współrzędnych zadania,
- *Wykresy współrzędnych złączowych* – blok służący do przedstawienia na wykresie wyznaczonych współrzędnych złączowych,

- *Zapis do pliku* – blok zapisu do pliku tekstowego współrzędnych kartezjańskich i złączowych trajektorii.
Działanie algorytmu zostało zweryfikowane dla danych przedstawionych w tabeli 2.

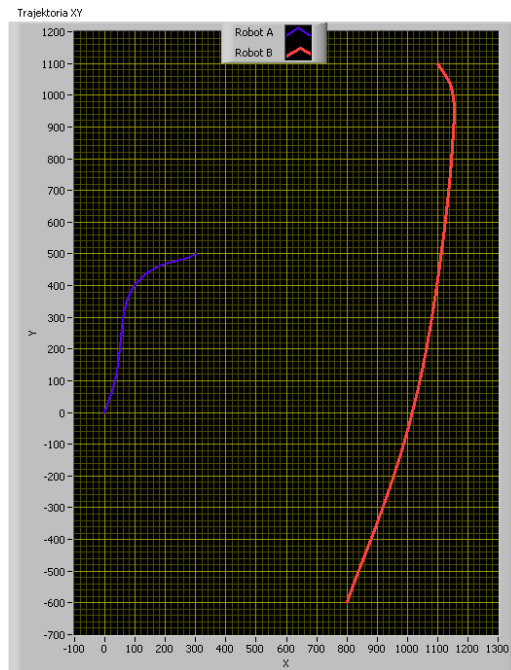
Tab. 2. Wartości danych wejściowych

Robot A		Robot B	
$X_{ZA} Y_{ZA} Z_{ZA}$ [mm]	0;0;0	$X_{ZB} Y_{ZB} Z_{ZB}$ [mm]	800;-600;0
$A_{T0}^{ZA} (x_{TA0}^{ZA} y_{TA0}^{ZA} z_{TA0}^{ZA})$ [mm]	0;0;100	$B_{T0}^{ZB} (x_{TB0}^{ZB} y_{TB0}^{ZB} z_{TB0}^{ZB})$ [mm]	0;0;100
γ_{TA0}^{ZA} [deg]	0	γ_{TB0}^{ZB} [deg]	0
$A_{TD}^{ZA} (x_{TAD}^{ZA} y_{TAD}^{ZA} z_{TAD}^{ZA})$ [mm]	300;500;100	$B_{TD}^{ZB} (x_{TBD}^{ZB} y_{TBD}^{ZB} z_{TBD}^{ZB})$ [mm]	300;1700;100
γ_{TAD}^{ZA} [deg]	0	γ_{TBD}^{ZB} [deg]	0
V_A^P [mm/s]	10	V_B^P [mm/s]	200
$X_{ZA}^{RA} Y_{ZA}^{RA} Z_{ZA}^{RA}$ [mm]	1800;-200;1100	$X_{ZB}^{RB} Y_{ZB}^{RB} Z_{ZB}^{RB}$ [mm]	2000;400;1100
γ_{ZA}^{RA} [deg]	0	γ_{ZB}^{RB} [deg]	180
k_A [1/s]	20	k_B [1/s]	20
T_A [s]	0,2	T_B [s]	0,2

Poniżej zamieszczono uzyskane wyniki. Wyznaczone współrzędne złączowe dla robota ZD130S, ustawionego według danych zamieszczonych w tabeli 2, pokazano na rysunku 3. Rysunek 4 przedstawia widok wygenerowanych trajektorii na płaszczyźnie $X_{ZA} Y_{ZA}$ układu współrzędnych zadania robota A.



Rys.3. Wykresy współrzędnych złączowych robota A



Rys.4. Wykresy trajektorii punktów TCP w układzie współrzędnych zadania robota A

4. PODSUMOWANIE

Opracowany algorytm koordynacji trajektorii punktów TCP dwóch robotów, realizujących wspólnie zadanie transportowe, wyznaczał współrzędne trajektorii w układzie współrzędnych zadania. Zaproponowany we wcześniejszych pracach autora sposób weryfikacji algorytmu miał charakter sprawdzenia offline. Uzupełnienie algorytmu o wyznaczanie współrzędnych złączowych, zaprezentowane w niniejszej pracy, umożliwi uwzględnienie konfiguracji stanowiska oraz parametrów robotów w trakcie generowania trajektorii robotów. Dalsze prace dotyczyć będą uwzględnienia w algorytmie generowania skoordynowanych trajektorii ograniczeń, wynikających z konfiguracji stanowiska i zastosowanych robotów.

5. LITERATURA

- [1] Application manual – MultiMove, ABB, MultiMove 3HAC021272-001_RevG_en.pdf
- [2] Słota A.: *Generowanie skoordynowanych ruchów robotów w przestrzeni 3D – weryfikacja w środowisku wirtualnym*, praca przyjęta do publikacji w *Pomiary Automatyka Robotyka* 2/2011
- [3] http://ol.cadfamily.com/delmia/online/DELMIA_default.htm
- [4] http://www.dinsaonline.com/aplicaciones/pdf/dualram_system/dualarmSystem.pdf
- [5] http://www.procax.org.pl/pliki/Artykul_9_Slota.pdf