

dr inż. Daniel Reclik, dr hab. inż. Gabriel Kost, prof. Pol. Śl
Instytut Automatykacji Procesów Technologicznych
i Zintegrowanych Systemów Wytwarzania
Wydziału Mechanicznego Technologicznego
Politechniki Śląskiej w Gliwicach

INTEGRACJA DANYCH PLANERA TRAJEKTORII BEZKOLIZYJNYCH ROBOTA Z SYSTEMEM EMPOWER ROBCAD

Brak dostępności na rynku oprogramowania pozwalającego na automatyczne generowanie trajektorii ruchu robotów przemysłowych spowodował rozpoczęcie w Instytucie Automatykacji Procesów Technologicznych i Zintegrowanych Systemów Wytwarzania prac nad utworzeniem takiego systemu. Pierwsza postać algorytmu planowania ruchu robotów została opracowana w 2008 roku. Powstała wówczas aplikacja PLANER, która była praktyczną implementacją opracowanej metody. Początkowo aplikacja PLANER pozwalała na eksportowanie programów roboczych jedynie do aplikacji FANUC ROBOGUIDE. Rozwiązanie to pozwoliło na przetestowanie metody, jednakże ograniczało możliwości stosowania jedynie do robotów FANUC. Przeprowadzone testy, a także rozpoznanie rynku aplikacji programowania robotów off-line pokazały, że utworzony system wspomagający pracę programistów może znaleźć bardzo szerokie spektrum odbiorców, jednakże uwarunkowane jest to rozszerzeniem liczby współpracujących systemów off-line. W pracy omówiono nowy moduł aplikacji PLANER, dzięki któremu możliwe stało się eksportowanie programów roboczych do manipulatorów innych firm niż FANUC, poprzez zastosowanie uniwersalnego systemu programowania off-line jakim jest eMPower RobCAD. W niniejszej publikacji opisany został sposób realizacji wymiany danych programowych pomiędzy aplikacją PLANER i eMPower RobCAD, jak również sposób dalszego ich wykorzystania w programie roboczym dowolnego manipulatora.

DATA EXCHANGE INTEGRATION BETWEEN THE COLLISION FREE ROBOTS' MOVEMENT PLANNING SOFTWARE AND THE EMPOWER ROBCAD SYSTEM

Lack of availability of the software which allows on automatically generating the robot motion trajectory caused that the scientists in the Institute of Engineering Processes Automation and Integrated Manufacturing Systems have decided to work out this kind of software. The first form of robot motion planning algorithm has been created in 2008. PLANER application, which was then created, has been a practical implementation of worked out method. At first, the PLANER application has allowed on export the working programs only to FANUC ROBOGUIDE software. This solution allowed on testing the method, however, it limited it's usage only for FANUC robots. Carried out tests and robot off-line programming market identification show that the created system could find wide range of recipients, however, it must be enlarge for more number of cooperated off-line systems. In this paper the new module of PLANER application has been described. This module allows on export of working programs to others manipulators (not only for FANUC robots) by the use of eMPower RobCAD universal off-line programming system. In this paper the data exchange method between PLANER application and eMPower RobCAD software has been shown. The way of use in working program any manipulator is also described.

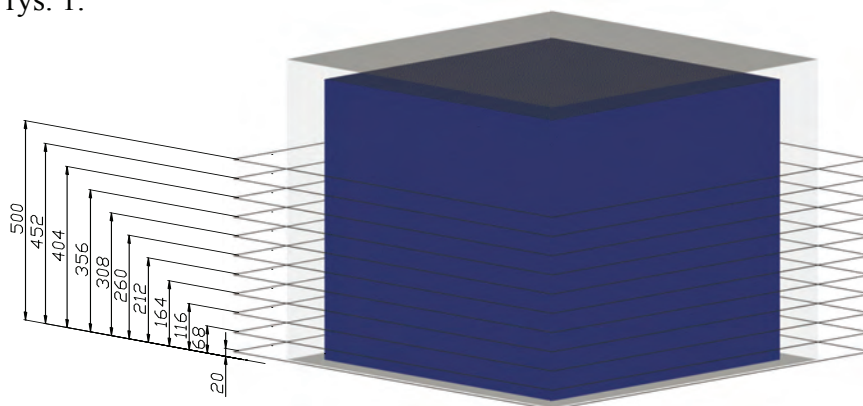
1. WPROWADZENIE

Wykorzystanie możliwości współczesnych komputerów oraz odpowiedniego oprogramowania wspomagającego programowanie robotów metodami off-line pozwala na zwiększenie wydajności pracy systemu produkcyjnego. Dzieje się tak dlatego, ponieważ dzięki zastosowaniu komputerowego wspomagania procesu przygotowania programów roboczych, w tym także sekwencji ruchu manipulatorów (systemy off-line), możliwe jest przygotowanie nowej produkcji bez zatrzymywania linii technologicznych, co pozwala maksymalizować wydajność urządzeń i tym samym obniżyć cenę końcową wyrobu. Można zauważyć, że w grupie środków wspomagających prace z zakresu programowania off-line robotów przemysłowych szczególnie słabo rozwinięty jest obszar zagadnień dotyczących planowania dróg bezkolizyjnych przebiegu robota pomiędzy obiektami technologicznymi znajdującymi się w jego przestrzeni zadaniowej. Brak dostępności na rynku oprogramowania pozwalającego na automatyczne generowanie trajektorii ruchu robotów przemysłowych spowodował rozpoczęcie w Instytucie Automatyki i Zintegrowanych Systemów Wytwarzania prac nad utworzeniem takiego systemu. Pierwszą postacią aplikacji PLANER powstała w 2008 roku. Od tamtej pory system PLANER był ciągle rozwijany, zarówno pod kątem szybkości działania, jak i możliwości integracji z istniejącymi systemami programowania robotów off-line. Początkowo istniała jedynie możliwość eksportu danych do systemu ROBOGUIDE firmy FANUC, co z punktu widzenia przeprowadzanych testów poprawności działania było wystarczające, jednakże w warunkach przemysłowych stanowiło istotną wadę systemu [8, 9, 11, 12, 13].

Ponieważ przeprowadzona seria testów działania pokazała, że utworzony system może w znaczący sposób wypełnić lukę na rynku oprogramowania wspomagającego technologiczne przygotowanie produkcji dla systemów zrobotyzowanych, autorzy postanowili rozszerzyć możliwości wymiany danych na inne systemy programowania robotów off-line. Przeprowadzona analiza rynku pokazała, że trudno byłoby dostosować format danych do systemów off-line wszystkich producentów robotów posiadających istotny udział w rynku sprzedaży w Polsce. Dlatego też podjęto próbę integracji aplikacji PLANER z oprogramowaniem eMPower RobCAD, gdyż umożliwiałoby to przygotowanie programu roboczego dla większości tych robotów [14, 15].

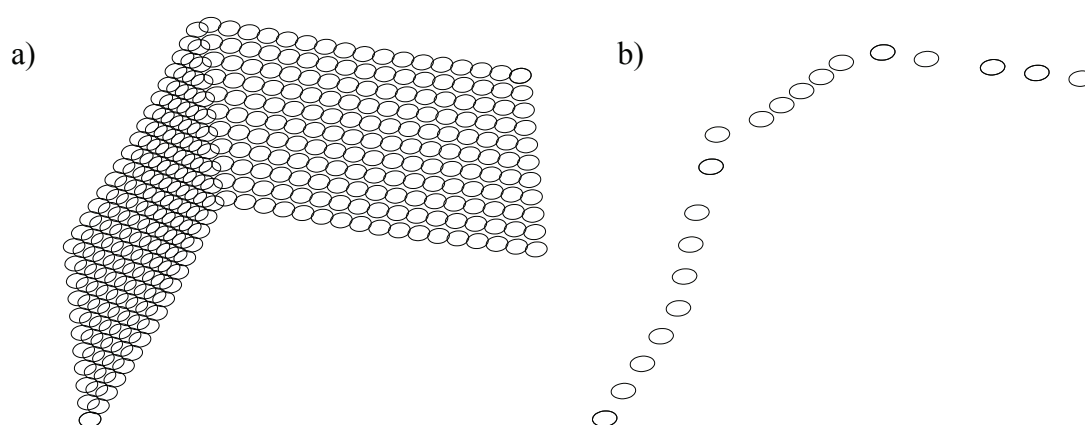
2. PODSTAWY MATEMATYCZNE APLIKACJI PLANER

Aplikacja PLANER była podstawą wielu publikacji, zarówno na konferencji Automation [8, 10, 11], jak i na innych konferencjach obejmujących zakresem tematycznym zagadnienia programowania i eksploatacji robotów, a także technologiczne przygotowanie produkcji w systemach zrobotyzowanych, stąd też w niniejszej publikacji zostały przedstawione jedynie jej główne założenia. Główną ideą opracowanego algorytmu obliczeniowego systemu PLANER było generowanie kolejnych przekrojów przestrzennej sceny robota, co pozwalało na uproszczenie obliczeń bezpiecznej ścieżki ruchu do zadania planowego. Przykładowy rozkład przekrojów sceny pokazano na rys. 1.



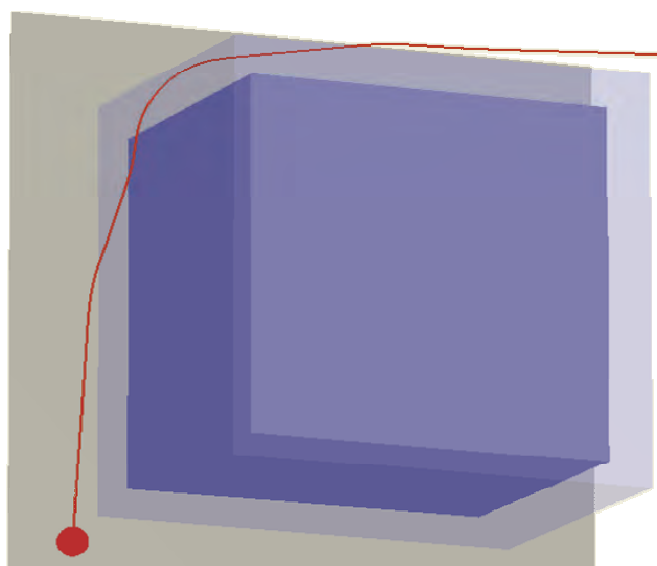
Rys. 1. Rozkład przekrojów sceny w przypadku pojedynczego obiektu otoczenia robota (aplikacja PLANER)

Wynikowe zbiory punktów podporowych są przetwarzane na graf możliwych przejść, którego optymalizacja pozwala na wytypowanie danych podporowych interpolacji B-Spline [2, 5], dzięki czemu końcowa postać trajektorii zapewnia utrzymanie gładkości klasy C^2 [5]. Gęstość rozkładu kolejnych przekrojów sceny uzależniona została od wielkości przemieszczanego przedmiotu, który na potrzeby zadania planowania ruchu aproksymowany jest za pomocą opisanej na nim kuli. Wprowadzana przez użytkownika wartość średnicy kuli opisanej na przedmiocie stanowi jednocześnie wielkość strefy bezpieczeństwa, przez co ma znaczący wpływ na przeprowadzaną analizę danego przekroju. Przekroje analizowane są metodą grafu widoczności ze wspomaganiami algorytmem Floyda-Warshalla [12] od najwyższej położonych w kierunku płaszczyzny sceny. Zgodnie z przyjętą metodą przestrzennego planowania ruchu robotów, aplikacja PLANER po przeprowadzeniu analizy wszystkich przekrojów danej sceny generuje pokazany na rys. 2a graf możliwych przejść (suma wyników analiz w wszystkich przekrojów), który w dalszej części jest optymalizowany pod względem minimalnej długości ruchu (rys. 2b).



Rys. 2. Graf możliwych przejść przed i po optymalizacji

Wynikowy zbiór punktów podporowych stanowi podstawę wykonywanej interpolacji B-Spline, której końcową postać pokazano na rys. 3.



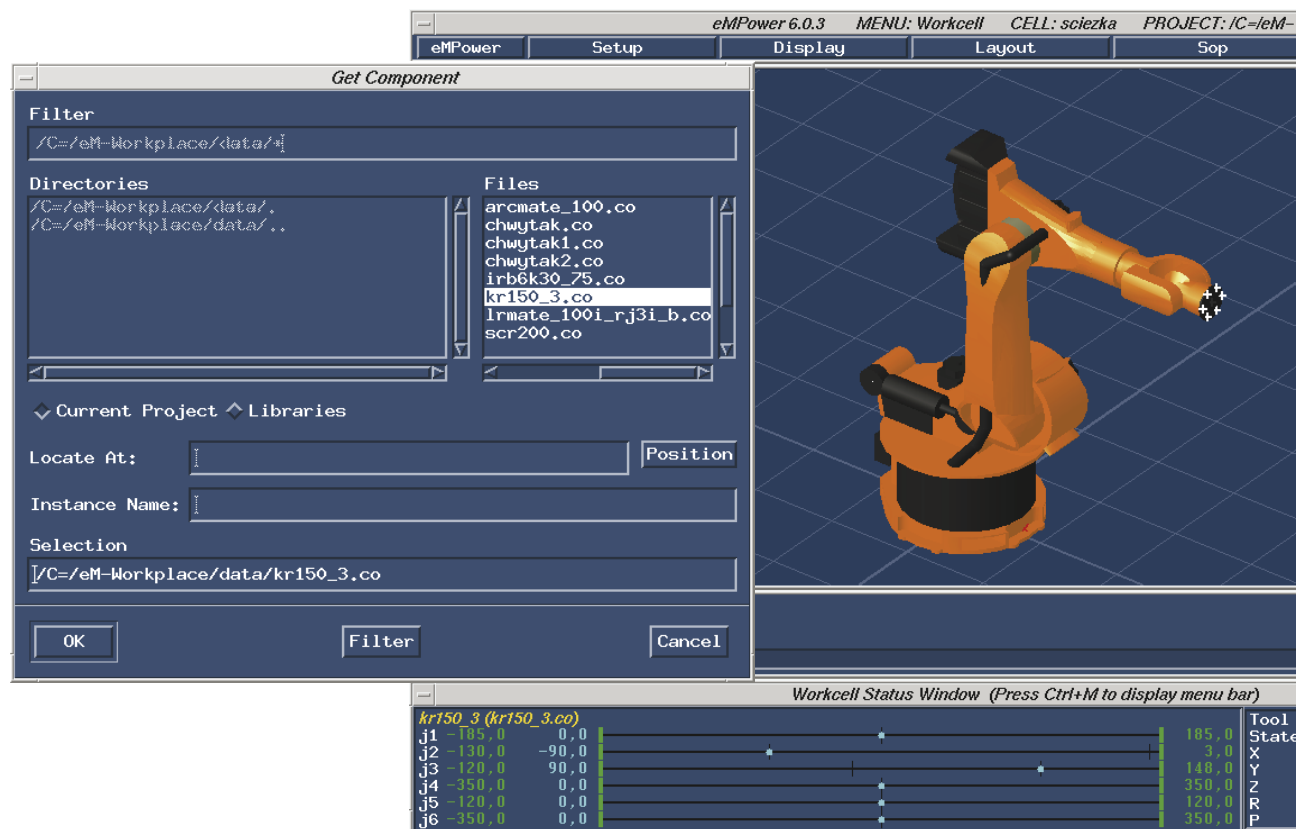
Rys. 3. Wynikowa ścieżka ruchu (aplikacja PLANER)

3. PODSTAWY OBSŁUGI SYSTEMU EMPOWER ROBCAD

eMPower RobCAD jest jednym z najstarszych systemów programowania robotów metodą off-line. Pierwsze wersje RobCADa pracowały na stacjach roboczych w środowisku Unix. Obecnie, wraz ze zwiększeniem możliwości graficznych komputerów PC, pojawiły się wersje przeznaczone do pracy w środowisku Windows. RobCAD jest środowiskiem bardzo rozbudowanym i w odróżnieniu od innych, dedykowanych do konkretnych robotów, systemów off-line (np. ABB-Robot Studio, FANUC ROBOGUIDE itp.), pozwala na generowanie programów roboczych dla większości współcześnie stosowanych robotów przemysłowych. W związku z wieloletnim procesem rozwoju systemu RobCAD, posiada on bardzo rozbudowany zestaw funkcji. Najważniejszymi z punktu widzenia programowania robotów są moduły:

- moduł *Workcell*,
- moduł *Modeling*,
- moduł *Kinematics*,
- narzędzie SOP (*Sequence Of Operations*).

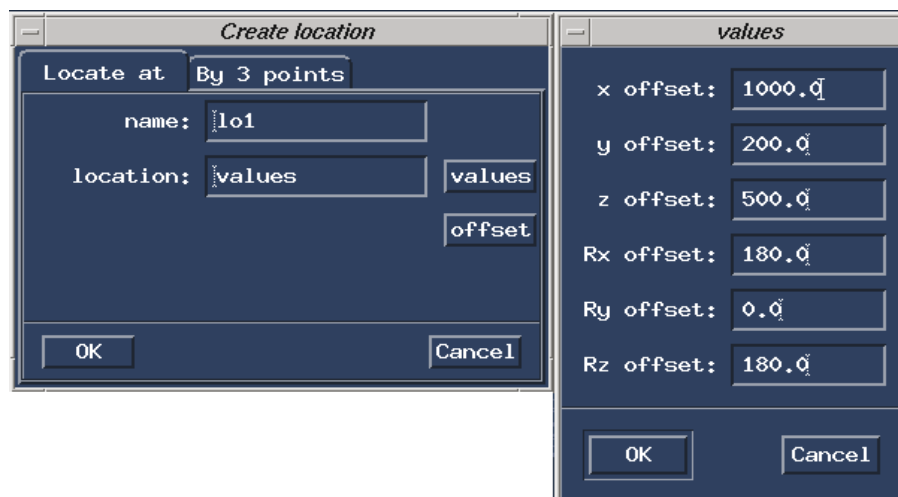
Pierwszym etapem przygotowania programu roboczego w systemie RobCAD jest utworzenie gniazda zrobotyzowanego. W tym celu należy posłużyć się modułem *Workcell* i przy użyciu narzędzia *Get Component* umieścić w gnieździe właściwy typ robota. Na potrzeby niniejszej publikacji autorzy postanowili zastosować robota KUKA KR150. Widok okna definicji typu robota pokazano na rys. 4.



Rys. 4. Okno wyboru nowego robota w systemie eMPower RobCAD

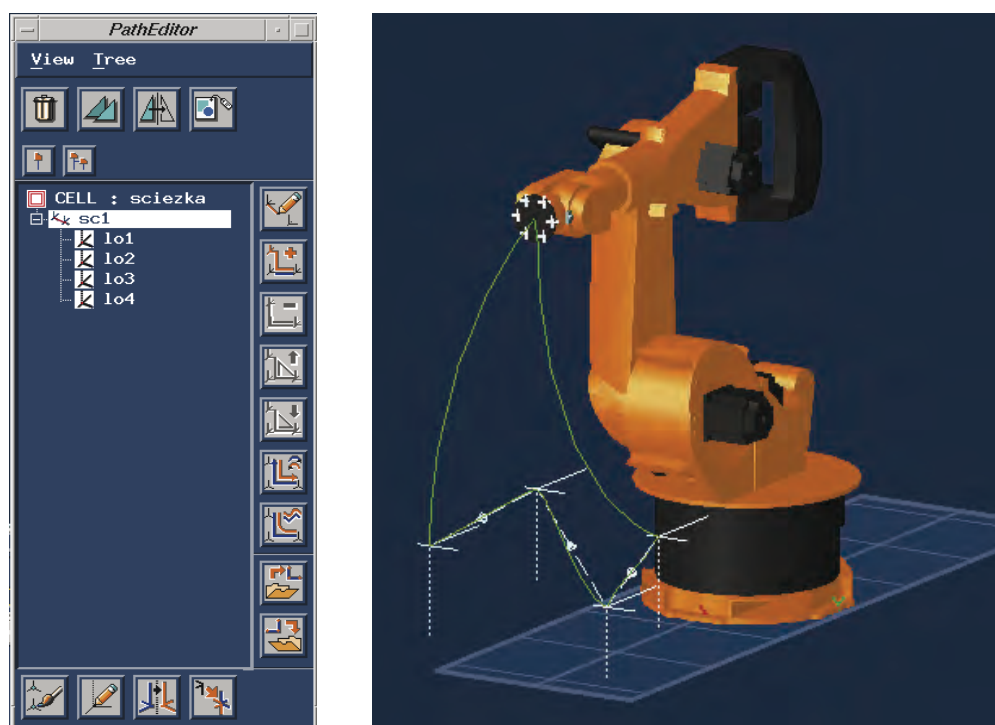
Aby wstawiony robot mógł być programowany, konieczne jest ustawienie aktywności obiektu. W tym celu należy wykorzystać pole *Active mech* i wskazać właściwy obiekt. Po wykonaniu wszystkich tych czynności możliwe jest przystąpienie do programowania ruchów robota. Aby przygotować program roboczy (sekwencje ruchów robota) w systemie eMPower RobCAD, należy

posłużyć się narzędziem *Path Editor*. Za pomocą edytora ścieżek możliwe jest wprowadzenie zestawu punktów (lokalizacji) robota, jak to pokazano na rys. 5.



Rys. 5. Wprowadzanie nowego punktu przyszłej ścieżki ruchu za pomocą metody 3-punktowej

Mając wprowadzony zestaw punktów podporowych ścieżki ruchu, możliwe jest połączenie wszystkich punktów w ścieżkę ruchu, którą następnie można przypisać do konkretnego robota umieszczonego w gnieździe (rys. 6).



Rys. 6. Nowa ścieżka ruchu robota KUKA KR150 w programie RobCAD

Przygotowana ścieżka może zostać przypisana do odpowiedniego zdarzenia modułu symulacyjnego, dzięki czemu możliwe jest sprawdzenie zachowania robota, jak również wygenerowanie stosownego podprogramu roboczego (sekwencji ruchu) na konkretny typ układu sterowania. W warunkach poprawności wygenerowanego kodu źródłowego sekwencji ruchu jest umieszczenie w katalogu programu odpowiedniej biblioteki (po przetworzeniu łączącego) dla wybranego typu robota.

4. AUTOMATYCZNE GENEROWANIE SEKWENCJI RUCHU – INTEGRACJA Z ZEWNĘTRZNYM NARZĘDZIEM PLANOWANIA TRAJEKTORII

Jak zostało pokazane w pkt. 3, programowanie ruchu robota z użyciem systemu eMPower RobCAD jest bardzo zbliżone do innych systemów off-line istniejących na rynku. Głównym zamierzeniem autorów nie było jednak ręczne programowanie ruchu robotów, ale automatyzacja tego procesu. W tym celu postanowiono opracować procedurę generowania plików ścieżek ruchu przez zewnętrzne narzędzia obliczeniowe w taki sposób, aby składowanie generowanych plików była zgodna z formatem RobCAD (aby istniała możliwość ich bezpośredniego wczytania do systemu eMPower). W tym celu konieczne było wykonanie zapisu pliku ścieżki o znanych punktach podporowych, a następnie dokonanie szczegółowej analizy uzyskanego pliku (*.path).

Jak pokazały przeprowadzone badania, pliki zapisu ścieżek ruchu z systemem eMPower RobCAD można podzielić na 3 główne części [14]. Należą do nich:

- nagłówek,
- właściwa część programu roboczego z zapisem współrzędnych (lokalizacji),
- część końcowa, która zawiera znaczniki końca ścieżki i końca programu [14].

Aby możliwe było opracowanie poprawnej procedury generowania pliku ścieżek ruchu (*.path) konieczne okazało się zweryfikowanie istoty kolejnych parametrów pliku. Analiza ta była konieczna, z racji uzyskania informacji o sposobie adaptowania pliku ścieżki do warunków sceny na etapie w czytania pliku (*.path). Wyniki przeprowadzonych badań zestawiono na rys. 7. Po lewej stronie przedstawiono plik źródłowy przed zaimportowaniem do środowiska eMPower RobCAD, po prawej stronie plik po wykonaniu automatycznej adaptacji do zainstalowanego w gnieździe robota KUKA KR150, zgodnie z przyjętą pozycją i orientacją robota (w tym przypadku nie występuje odsunięcie punktu referencyjnego narzędzia względem kiści – tcpf = 0, 0, 0, 0, 0, 0).

SKŁADNIA NAGŁÓWKA PLIKU *.path

prog_start no_name robot_name no_name tcpf 0.000000,0.000000,0.000000,0.000000,0.000000,0.000000 path_start no_name	prog_start sc4 robot_name kr150_3 tcpf 0.000000,0.000000,0.000000,0.000000,0.000000,0.000000 path_start sc_4
--	---

SKŁADNIA WŁAŚCIWA PLIKU *.path

color 0 display 0 wp_start lo1 FXS_PATH_ATTACH_STR no_name #ATTR COLLISION_CHECK 0 relative 1000.000000,200.000000,500.000000,180.000000,0.000000,180.000000 reltcpf 0,0,0,0,0,0 absolute 0,0,0,0,0,0 attach no_name color 0 display 0 wp_start lo2 FXS_PATH_ATTACH_STR no_name #ATTR COLLISION_CHECK 0 relative 1400.000000,200.000000,300.000000,180.000000,0.000000,180.000000 reltcpf 0,0,0,0,0,0 absolute 0,0,0,0,0,0 attach no_name color 0 display 0 wp_start lo3 FXS_PATH_ATTACH_STR no_name #ATTR COLLISION_CHECK 0 relative 1600.000000,0.000000,800.000000,180.000000,0.000000,180.000000 reltcpf 0,0,0,0,0,0 absolute 0,0,0,0,0,0 attach no_name	color 7 display 1 wp_start lo1 FXS_PATH_ATTACH_STR sciezka #ATTR COLLISION_CHECK 0 relative 1000.000000,200.000000,500.000000,180.000006,0.000000,180.000006 reltcpf 1410.000122,200.000000,-619.998901,0.000000,90.000003,0.000000 absolute 1000.000000,200.000000,500.000000,180.000006,0.000000,180.000006 attach sciezka color 7 display 1 wp_start lo2 FXS_PATH_ATTACH_STR sciezka #ATTR COLLISION_CHECK 0 relative 1400.000000,200.000000,300.000000,180.000006,0.000000,180.000006 reltcpf 1610.000122,200.000000,-219.998901,0.000000,90.000003,0.000000 absolute 1400.000000,200.000000,300.000000,180.000006,0.000000,180.000006 attach sciezka color 7 display 1 wp_start lo3 FXS_PATH_ATTACH_STR sciezka #ATTR COLLISION_CHECK 0 relative 1600.000000,0.000000,800.000000,180.000006,0.000000,180.000006 reltcpf 1110.000122,0.000000,-19.998901,0.000000,90.000003,0.000000 absolute 1600.000000,0.000000,800.000000,180.000006,0.000000,180.000006 attach sciezka
--	---

SKŁADNIA KOŃCA PLIKU *.path

path_end no_name prog_end	path_end no_name prog_end
------------------------------	------------------------------

Rys. 7. Składnia zapisu plików definicji ścieżek ruchu w programie eMPower RobCAD

Jak pokazały badania, proces eksportu danych z zewnętrznego narzędzia obliczeniowego (w tym przypadku aplikacji PLANER) musi zapewnić zapis sztywno zdefiniowanego nagłówka pliku, sztywno określonego znacznika końca pliku, a także sekwencji związanej z definicją kolejnych punktów podporowych zgodnie ze schematem pokazanym na rys. 8.

```

color 0
display 0
wp_start [red] p [green]
FXS_PATH_ATTACH_STR no_name
#ATTR COLLISION_CHECK 0
relative [yellow], [yellow], [yellow], 180.000000, 0.000000, 180.000000
reltcpf 0, 0, 0, 0, 0, 0
absolute 0, 0, 0, 0, 0, 0
attach no_name

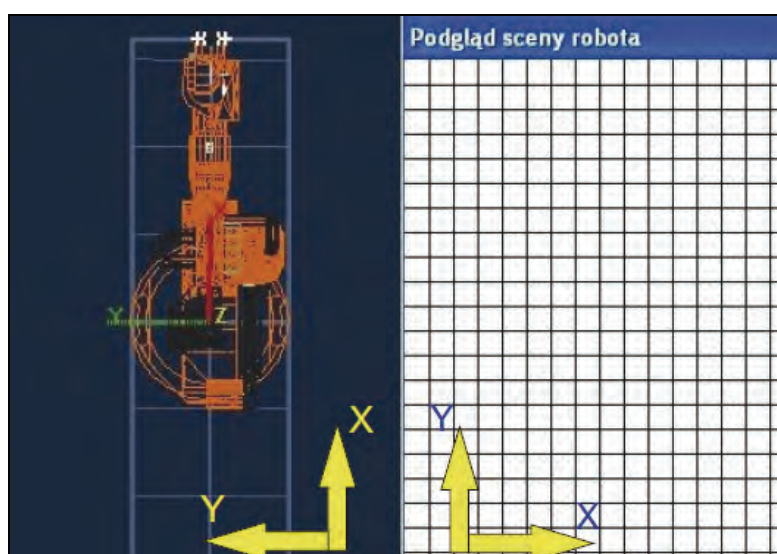
```

Rys. 8. Cykl wprowadzania pojedynczej lokalizacji do pliku definiującego ścieżkę

Podczas tworzenia pliku źródłowej definicji ścieżki kolejne punkty podporowe (współrzędne x, y, z) przyszłej trajektorii wstawiane są z dokładnością 6 miejsc po przecinku w pola oznaczone kolorem żółtym (rys. 8). W miejsce pola czerwonego (rys. 8) wstawiana jest nazwa pliku ścieżki. Aby kolejne punkty mogły tworzyć ścieżkę muszą posiadać unikatową nazwę. W związku z tym przyjęto, że nazwy kolejnych punktów podporowych (lokalizacji) będą tworzone poprzez połączenie dwóch łańcuchów tekstowych. W miejsce spójnika wstawiany będzie natomiast kolejny numer punktu podporowego przyszłej trajektorii. Przykładowo zapis czwartego punktu podporowego trajektorii o nazwie „TEST” będzie następujący:

wp_start TESTp4

Po uruchomieniu procedury odczytu pliku **.path* wartości współrzędnych *reltcpf* oraz *absolute* ulegają automatycznemu dopasowaniu do umieszczonego w gnieździe robota, dzięki czemu nie ma potrzeby każdorazowego przeliczania tych wartości. Aby zapewnić pełną kompatybilność z utworzoną aplikacją PLANER konieczne było dostosowanie orientacji bazowych globalnych układów współrzędnych. Operacja ta związana była z różnym przyjęciem położenia początku globalnego układu współrzędnych. Na rys. 9 pokazano zastosowaną transformację układów współrzędnych.



Rys. 9. Transformacja układu współrzędnych: po lewej orientacja globalnego układu współrzędnych systemu RobCAD, po prawej przyjęta orientacja w aplikacji PLANER

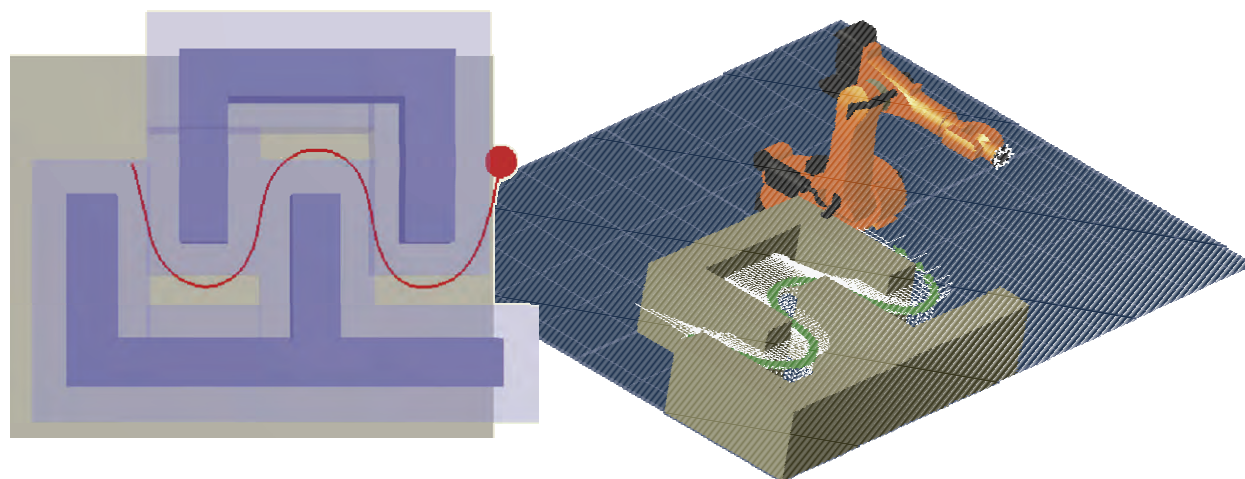
System eMPower RobCAD pozwala na dowolne przesunięcia, obroty i odbicia (funkcja *Mirror*) wczytywanej ścieżki ruchu, jednakże, jak już wspomniano wcześniej, nadrzędnym celem autorów było zapewnienie całościowej automatyzacji importu danych. Zamieniono zatem wartości współrzędnych według zależności:

$$\begin{cases} X_{ROBCAD} = Y_{PLANER} \\ Y_{ROBCAD} = -X_{PLANER} \\ Z_{ROBCAD} = Z_{PLANER} \end{cases} \quad (4.1)$$

Wprowadzenie funkcji przeliczającej wynikowe punkty podporowe aplikacji PLANER na właściwe położenie względem sceny system u RobCAD pozwoliło na bezproblemowe importowanie wyników analizy, dzięki czemu system RobCAD został rozbudowany o możliwość automatycznego generowania i optymalizacji bezkolizyjnych trajektorii ruchu robotów w oparciu o dane geometryczne ich otoczenia technologicznego.

5. PRZYKŁAD IMPORTU DANYCH Z APLIKACJI PLANER DO SYSTEMU EMPOWER ROBCAD

Chcąc zobrazować korzyści, jakie daje zastosowanie integracji wymiany danych między aplikacją PLANER a systemem eMPower RobCAD przygotowano przykładowe gniazdo zrobotyzowane, a następnie wykonano automatyczne generowanie trajektorii ruchu przy wykorzystaniu mechanizmu planowania trajektorii bezkolizyjnych zaimplementowanych w aplikacji PLANER. Wynik obliczeń (PLANER) oraz zaimportowaną ścieżkę ruchu (RobCAD) zestawiono na rys. 10.



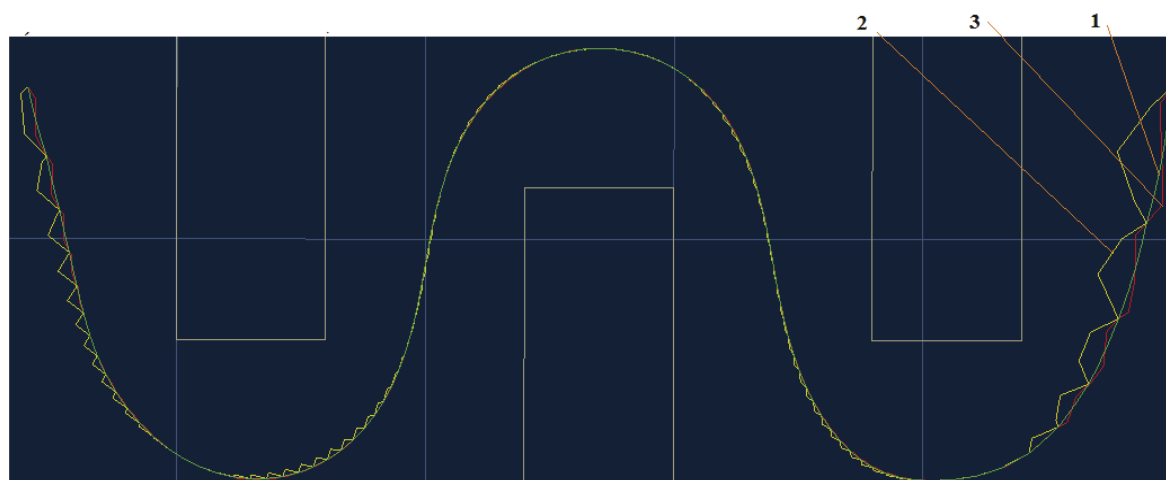
Rys. 10. Wygenerowana przykładowa trajektoria bezkolizyjna: a) aplikacja PLANER, b) wynik importu danych do systemu eMPower RobCAD

Aby zobrazować korzyści, jakie uzyskuje się poprzez zastosowanie algorytmicznej optymalizacji wynikowych trajektorii ruchu, postanowiono wygenerować przykładowy harmonogram pracy gniazda, a następnie porównać wyniki (czasy przemieszczania robota) pomiędzy dwoma punktami sceny po ścieżce przygotowanej z wykorzystaniem aplikacji PLANER, z wynikami uzyskanymi na drodze ręcznej definicji programu roboczego. Do tego celu wykorzystano możliwości narzędzia SOP. Porównywano czas potrzebny na wykonanie operacji nr 4 „podjazd_do_tok” (rys. 11). Okazało się, że czas realizacji ruchu jest znacząco krótszy, w przypadku realizacji ścieżek ruchu pochodzących z aplikacji PLANER. Realizacja krótszych czasów przejazdu pozwala na skrócenie czasu cyklu całego gniazda i tym samym zwiększenie efektywności wykorzystania maszyn. Czas realizacji ruchu po wytyczonej w aplikacji PLANER bezkolizyjnej ścieżce ruchu wynosił 12,9 s. Dla porównania czas tego samego ruchu po ścieżce zadanej manualnie w systemie RobCAD wynosił 13,8 s.



Rys. 11. Zastosowanie wykresów Gantta przy porównywaniu czasu realizacji ruchu

Dodatkowe skrócenie czasu przemieszczenia końcówki robota można uzyskać, rezygnując z interpolacji ruchu [14, 15]. Jest to jednak zabieg, który wiąże się z niebezpieczeństwem występowania kolizji, zwłaszcza w przypadku niedostatecznie gęsto zdefiniowanej ścieżki. Jeżeli ścieżka ruchu jest zdefiniowana za pomocą dużej liczby punktów podporowych prawdopodobieństwo wystąpienia kolizji znacząco maleje. Jak pokazały przeprowadzone testy, przy założeniu średniej wielkości sceny (1500×1500 mm), zdefiniowanie 200 punktów podporowych daje 100-procentową pewność poprawności realizacji wytyczonej trajektorii. Niestety, bez komputerowego wspomagania wytyczanie punktów podporowych wprowadzenie 200 punktów przyspieszenia ścieżki jest bardzo czasochłonne i raczej programiści ograniczają tę liczbę poniżej wyznaczonego poziomu bezpieczeństwa, przez co wykonywany ruch wymiaga zastosowania przynajmniej interpolacji liniowej. Zmianę zachowania końcówki robota podczas ruchu przy różnych typach interpolacji dla przykładu z rys. 10 przedstawiono na rys. 12.



Rys. 12. Ścieżka wykonana w trzech różnych konfiguracjach: 1 – ruch z interpolacją liniową (kolor zielony), 2 – ruch z interpolacją kołową (kolor żółty) 3 – ruch z punktu do punktu (kolor czerwony)

6. PODSUMOWANIE I WNIOSKI

Uzyskane wyniki importowania bezkolizyjnych trajektorii ruchu z aplikacji PLANER pokazały, że przyjęte rozwiązanie pozwala w znaczący sposób usprawnić proces programowania robotów w systemie eMPower RobCAD. Możliwość, jakie niesie za sobą możliwość rezygnacji ze stosowania ruchów interpolowanych, jak również znacząco krótsze czasy przejść robota pomiędzy poszczególnymi punktami i sceny pozwalają na stwierdzenie, że zastosowanie zewnętrznych

narzędzi obliczeniowych (w tym przypadku opracowanej aplikacji PLANER) pozwala nie tylko przyspieszyć proces przygotowania programu roboczego, ale także w sposób znaczący poprawić jego jakość. Zwiększenie zagęszczenia punktów podporowych, wyłączenie i optymalizacja samej ścieżki ruchu, krótsze czasy realizacji wygenerowanych programów to tylko niektóre zalety, jakie uzyskano na skutek integracji aplikacji PLANER z systemem eMPower RobCAD. Dodatkową i chyba najważniejszą zaletą jest skrócenie czasu cyklu linii produkcyjnej, a tym samym zwiększenie wydajności zrobotyzowanego systemu produkcyjnego.

Dodatkowym atutem przemawiającym za skutecznością wyboru aplikacji eMPower RobCAD jest zaimplementowany w niej mechanizm eksportu danych do większości systemów programowania robotów metodami off-line [14, 15].

Praca finansowana z grantu badawczo-rozwojowego nr PBR-6/RMT2/2010: Mechatroniczny integrator procedur sterowania pojazdem przez osoby niepełnosprawne (decyzja MNiSzW nr 0894/R/T02/2010/10)

BIBLIOGRAFIA

1. Dulęba I.: Metody i algorytmy planowania ruchu robotów mobilnych i manipulacyjnych. Akademia Oficyna Wydawnicza EXIT, Warszawa 2001.
2. Fortuna Z., Macukow B., Wąsowski J.: Metody numeryczne. Podręczniki Akademickie. Elektronika. Informatyka. Telekomunikacja. Wydawnictwa Naukowe PWN. Warszawa.
3. Jankowski B.: Programowanie w praktyce. Wydawnictwo MIKOM, Warszawa 1999.
4. Latombe J.-C.: Robot motion planning. Kluwer Academic Publishers. London. 1993.
5. Majchrzak E., Mochacki B.: Metody numeryczne. Podstawy teoretyczne, aspekty praktyczne i algorytmy. Wydawnictwo Politechniki Śląskiej, Gliwice 2004.
6. Reclik D., Kost G.: The comparison of elastic band and B-Spline polynomial methods in smoothing process of collision-less robot trajectory, Journal of Achievements in Materials and Manufacturing Engineering, Volume 29, Issue 2, 2008.
7. Reclik D., Kost G.: The 2½D Algorithm in Robot Workspace Analysis, 4th Conference Mechatronic Systems and Materials 2008, MSM 2008, Acta Mechanica et Automatica, Volume 2, No. 3, Białystok Technical University, 2008.
8. Reclik D., Kost G.: Zastosowanie metody 2½D w planowaniu trajektorii robotów manipulacyjnych, Pomiary Automatyka Robotyka, Nr 2/2008, Wydanie specjalne z okazji X Konferencji Naukowo Technicznej AUTOMATION 2008. Warszawa. 2008.
9. Reclik D., Kost G.: Dynamiczne planowanie trajektorii robotów manipulacyjnych. Przegląd mechaniczny Nr 1/08. s. 23–29.
10. Reclik D., Kost G.: A rational B-Spline curves in robot collision-free movement planning, Journal of Automation, Mobile Robotics and Intelligent Systems, Volume 2, No. 3, Industrial Research Institute for Automation and Measurements PIAP, Warszawa 2008.
11. D. Reclik, G. Kost, *Zastosowanie grafiki 3D i bibliotek OpenGL do symulacji stanów kolizyjnych robota*, Pomiary Automatyka Robotyka, Nr 2/2010. PAR. Warszawa. 2010. + CD
12. K. Foit, G. Kost, D. Reclik, *Automatyczne programowanie i wizualizacja 3D ruchów robota Mitsubishi Movemaster RV-M1*, Materiały Konferencyjne XI Krajowej Konferencji Robotyki, 2010.
13. G. Kost, D. Reclik, *Planowanie bezkolizyjnej trajektorii robota manipulacyjnego w przestrzeni zadaniowej*, Wydawnictwo Politechniki Śląskiej, Gliwice 2010.
14. <http://www.rob cad.de>
15. http://www.plm.automation.siemens.com/en_us/products/tecnomatix