

ANALIZA ŚCIEŻKI NARZĘDZIA W OPERACJI MONTAŻU Z ZASTOSOWANIEM PROGRAMU ROBOT-STUDIO

Analysis of the tool path in assembly using RobotStudio program

Marcin WIŚNIEWSKI, Katarzyna PETA, Arkadiusz PIETROWIAK,

Olaf CISZAK, Kamil GROCHALSKI

Streszczenie: W artykule zaprezentowano możliwości środowiska RobotStudio firmy ABB oraz przedstawiono budowę dydaktycznego zrobotyzowanego stanowiska do montażu reduktora wraz z jego modelem wirtualnym, wspierającym programowanie. Przedstawiono trajektorię ruchu narzędzia i poddano ją analizie w module Signal Analyzer. Dzięki temu możliwa jest optymalizacja ścieżki narzędzia w procesie przemysłowym.

Słowa kluczowe: procesy przemysłowe, roboty przemysłowe, RobotStudio, Signal Analyzer

Abstract: The article presents the possibilities of the RobotStudio environment by ABB company and presents the construction of the didactic robotic workstation for the reducer assembly together with its virtual model that supports programming. The trajectory of the tool movement is presented and analyzed in the Signal Analyzer module. That makes it possible to optimize the tool path in the industrial process.

Key words: industrial processes, industrial robots, RobotStudio, Signal Analyze

Wprowadzenie

Współcześnie od gotowego wyrobu oczekuje się najwyższej jakości, szczególnie w przypadku wyrobów oferowanych przez przemysł motoryzacyjny, elektroniczny i metalowy [8]. W celu sprostania wymaganiom rynku coraz powszechniej stosuje się roboty przemysłowe, które stają się w ten sposób jednym z ważniejszych ogniw procesu produkcyjnego, gwarantujących zarówno jego właściwy przebieg, jak i prawidłowe działanie gotowego wyrobu.

Zrobotyzowany proces montażu ma bezpośredni i znaczący wpływ na parametry jakościowe wyrobu, tj.: niezawodność (zdolność do pracy bezusterkowej), trwałość (okres zachowania cech użytkowych) i bezpieczeństwo użytkownika [4]. Cechami charakteryzującymi pracę robota przemysłowego (element zrobotyzowanego stanowiska montażowego) jest dokładność (AP) [9] i powtarzalność (RP) [5] pozycjonowania. Do dalszych rozważań przyjęto ich stałe, niezmiennie wartości – założenie to umożliwi analizę przebiegu danej operacji montażu pod względem stałych podczas ruchu parametrów pracy robota, np. prędkości, położenia punktów lub sygnałów I/O.

Budowa dydaktycznego zrobotyzowanego stanowiska montażowego

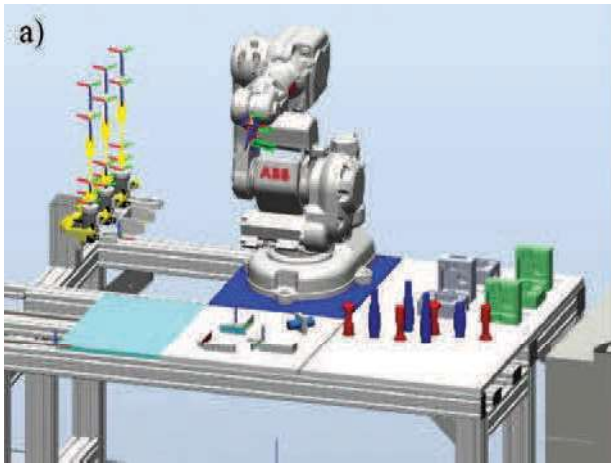
Głównym elementem dydaktycznego stanowiska zrobotyzowanego jest robot IRB140T firmy ABB programowany za pomocą kontrolera IRC5. Obecne stanowisko

zrobotyzowane (rys. 1b) zamodelowano w programie RobotStudio 6.03 (rys. 1a) – takie rozwiązanie pozwoliło na programowanie robota, testowanie i symulowanie jego pracy, a po pozytywnej weryfikacji uruchomienie programu na rzeczywistym obiekcie.

Praca w środowisku wirtualnym umożliwia zaprogramowanie poszczególnych pozycji robota, wynikających z przebiegu procesu technologicznego, a w przypadku nieprawidłowego jego działania, np. kolizji, szybką korektę w celu uniknięcia dodatkowych strat i kosztów na stanowisku produkcyjnym. Na przedstawionym stanowisku zrobotyzowanym możliwe jest przeprowadzenie symulacji procesu montażu reduktora składającego się korpusu dolnego, górnego i dwóch wałków. Na rys. 2a przedstawiono przykładową zaprojektowaną ścieżkę narzędzia (chwytaka) dla operacji montażu reduktora, z kolei rys. 2b stanowi zdjęcie z jej realizacji na istniejącym stanowisku laboratoryjnym.

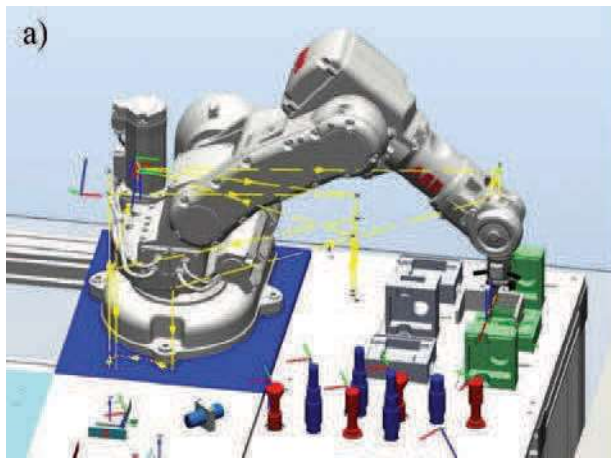
Zaprojektowana ścieżka narzędzia w środowisku wirtualnym może być następnie analizowana i optymalizowana. W omawianym przypadku optymalizacja [6] polegająca na skróceniu ścieżki narzędzia i zmniejszeniu czasu jej trwania, realizowana była za pomocą: zmiany położenia punktów tworzących trajektorię ruchu robota; zwiększenia prędkości ruchu; zmiany interpolacji ruchu oraz określenia parametru z [2] (*zonedata* – tzw. zaokrąglenia trajektorii, rys. 3).

Zmieniając prędkość ruchu i parametr z lub *fine* (określający sposób osiągnięcia punktu), uzyskano skrócenie czasu trwania operacji. Przykładowe zapisy programu wraz z graficznym przedstawieniem ścieżki



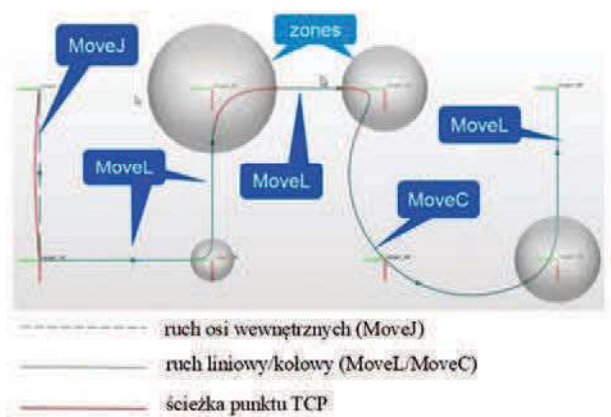
Rys. 1. Dydaktyczne stanowisko zrobotyzowane: a) zamodelowane w programie RobotStudio, b) istniejące stanowisko laboratoryjne [7]

Fig. 1. Didactic Robotized Position: a) Modified in RobotStudio, b) Existing laboratory post [7]



Rys. 2. Montaż reduktora: a) przykładowa zaprojektowana ścieżka narzędzia (chwytaka) operacji montażu reduktora, b) zdjęcie z realizacji zaprojektowanej ścieżki na stanowisku laboratoryjnym [7]

Fig. 2. Assembly of the reducer: a) Sample design of tool path (Gripper) of the reducer mounting operation, b) Photo of the projected path in the Laboratory station [7]



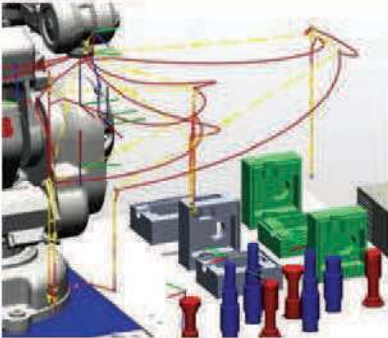
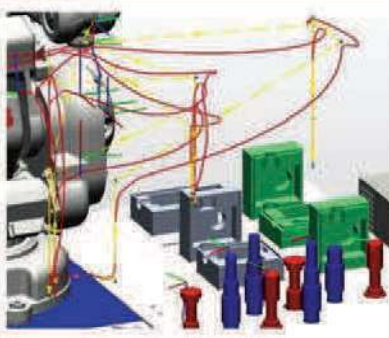
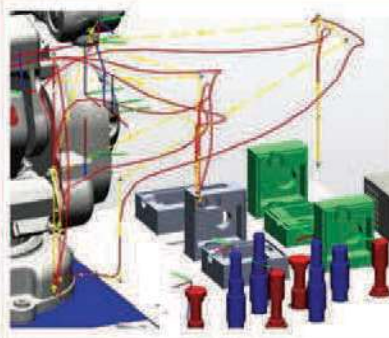
Rys. 3. Graficzne przedstawienie parametru z (zonedata – zaokrąglenie trajektorii) [2]

Fig. 3. Graphical representation of the parameter z (zonedata – rounding trajectory) [2]

narzędzia i wynikami symulacji ruchu przedstawiono w tab. I.

Porównując otrzymane wyniki, można zauważyć, że zmiana parametru $z80$ na $z200$ oraz ustawienie prędkości ruchu z $v1000$ na $vmax$ spowodowały skrócenie czasu montażu o 3 s. Według danych producenta prędkość, jaką może osiągnąć robot IRB140T, wynosi ok. 2,5 m/s. Można zatem przyjąć, że ustawienie prędkości $vmax$ znacząco skróci czas montażu. Wiadomo jednak, że na osiągnięcie zadanej prędkości ruchu duży wpływ ma orientacja narzędzia w punktach tworzących trajektorię ruchu. Układ sterowania ogranicza prędkość ruchu do takiej wartości, przy której jest w stanie osiągnąć zadane położenie, dlatego ważne jest prawidłowe ustawienie narzędzia w poszczególnych punktach trajektorii oraz dobór odpowiedniej prędkości ruchu. Może mieć to istotne znaczenie w przypadku operacji nakładania kleju, mas uszczelniających, lakierowania

Tabela I. Zapis programu i wyniki symulacji
Tabela I. Program recording and simulation results

<i>v1000, fine</i>	<i>v1000, z80</i>	<i>vmax, z200</i>
<pre> PROC Path_20_f() zerowanie; CH2P; MoveJ home10,v1000,fine,tool0; MoveJ home30,v1000,fine,tool0; MoveL home40_2,v1000,fine,tool0; MoveL home40,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home50,v30,fine,tool0; MoveJ home60,v1000,fine,tool0; MoveL home70,v500,fine,tool0; MoveL home80,v30,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; MoveL home90,v1000,fine,tool0; Set DO14_1; Zamkniecie_imadla; MoveJ home10,v1000,z100,tool0; MoveJ home340,v1000,z50,tool0; MoveL home350,v1000,fine,tool0; MoveL home360,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home370,v1000,fine,tool0; MoveJ home380,v1000,z20,tool0; MoveJ home390,v1000,z20,tool0; MoveL home400,v1000,z10,tool0; MoveL home410,v1000,fine,tool0; MoveJ home420,v500,z10,tool0; MoveJ home440,v400,z10,tool0; MoveJ home450,v300,z10,tool0; MoveJ home460,v200,z10,tool0; MoveL home470,v100,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; Set DO14_2; MoveL home480,v1000,fine,tool0; MoveL home490,v1000,z20,tool0; MoveL home480,v1000,fine,tool0; Otwarcie_imadla; MoveL home510,v30,fine,tool0; CH_2_zamknij; Set DO14_3; MoveL home520,v500,fine,tool0; MoveL home530,v1000,z50,tool0; MoveJ home30,v1000,fine,tool0; MoveL home560,v1000,fine,tool0; MoveL home570,v30,fine,tool0; CH_2_otworz; Set DO14_4; MoveL home580,v1000,fine,tool0; MoveJ home10,v1000,fine,tool0; CH2O; MoveJ home10,v1000,fine,tool0; Set DO14_5; zerowanie; ENDPROC </pre>	<pre> PROC Path_20_z80() zerowanie; CH2P; MoveJ home10,v1000,z80,tool0; MoveJ home30,v1000,z80,tool0; MoveL home40_2,v1000,z80,tool0; MoveL home40,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home50,v30,fine,tool0; MoveJ home60,v1000,z80,tool0; MoveL home70,v500,z80,tool0; MoveL home80,v30,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; MoveL home90,v1000,z80,tool0; Set DO14_1; Zamkniecie_imadla; MoveJ home10,v1000,z100,tool0; MoveJ home340,v1000,z80,tool0; MoveL home350,v1000,z80,tool0; MoveL home360,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home370,v1000,fine,tool0; MoveJ home380,v1000,z80,tool0; MoveJ home390,v1000,z80,tool0; MoveL home400,v1000,z80,tool0; MoveL home410,v1000,z80,tool0; MoveJ home420,v500,z80,tool0; MoveJ home440,v400,z80,tool0; MoveJ home450,v300,z80,tool0; MoveJ home460,v200,z80,tool0; MoveL home470,v100,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; Set DO14_2; MoveL home480,v1000,fine,tool0; MoveJ home490,v1000,z80,tool0; MoveL home480,v1000,fine,tool0; Otwarcie_imadla; MoveL home510,v30,fine,tool0; CH_2_zamknij; Set DO14_3; MoveL home520,v500,fine,tool0; MoveL home530,v1000,z80,tool0; MoveJ home30,v1000,z80,tool0; MoveL home560,v1000,z80,tool0; MoveL home570,v30,fine,tool0; CH_2_otworz; Set DO14_4; MoveL home580,v1000,fine,tool0; MoveJ home10,v1000,z80,tool0; CH2O; MoveJ home10,v1000,fine,tool0; Set DO14_5; zerowanie; ENDPROC </pre>	<pre> PROC Path_20_z200_vmax() zerowanie; CH2P; MoveJ home10,vmax,z200,tool0; MoveJ home30,vmax,z200,tool0; MoveL home40_2,vmax,z200,tool0; MoveL home40,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home50,v30,fine,tool0; MoveJ home60,vmax,z200,tool0; MoveL home70,vmax,z200,tool0; MoveL home80,v30,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; MoveL home90,vmax,z200,tool0; Set DO14_1; Zamkniecie_imadla; MoveJ home10,vmax,z200,tool0; MoveJ home340,vmax,z200,tool0; MoveL home350,vmax,z200,tool0; MoveL home360,v30,fine,tool0; Set DO10_15_przypnij_odepnij_CH2; CH_2_zamknij; MoveL home370,vmax,fine,tool0; MoveJ home380,vmax,z200,tool0; MoveJ home390,vmax,z200,tool0; MoveL home400,vmax,z200,tool0; MoveL home410,vmax,z200,tool0; MoveJ home420,vmax,z200,tool0; MoveJ home440,vmax,z200,tool0; MoveJ home450,vmax,z20,tool0; MoveJ home460,v200,z200,tool0; MoveL home470,v100,fine,tool0; CH_2_otworz; Reset DO10_15_przypnij_odepnij_CH2; Set DO14_2; MoveL home480,vmax,fine,tool0; MoveJ home490,vmax,z20,tool0; MoveL home480,vmax,fine,tool0; Otwarcie_imadla; MoveL home510,v30,fine,tool0; CH_2_zamknij; Set DO14_3; MoveL home520,v500,fine,tool0; MoveL home530,vmax,z200,tool0; MoveJ home30,vmax,z200,tool0; MoveL home560,vmax,z200,tool0; MoveL home570,v30,fine,tool0; CH_2_otworz; Set DO14_4; MoveL home580,vmax,fine,tool0; MoveJ home10,vmax,z200,tool0; CH2O; MoveJ home10,vmax,fine,tool0; Set DO14_5; zerowanie; ENDPROC </pre>
 <p>Czas symulacji: 52,4 s</p>	 <p>Czas symulacji: 47,5 s</p>	 <p>Czas symulacji: 44,5 s</p>

lub spawania, gdzie dokonywana przez robota zmiana prędkości (najczęściej redukcja) wykonywania tych operacji może spowodować nieprawidłowy przebieg procesu (np.: przegrzanie spoiny, zwiększenie grubości kleju, lakieru lub spłynięcie masy uszczelniającej itp.).

Analiza ścieżki narzędzia w programie RobotStudio

Narzędzie Signal Analyzer programu RobotStudio służy do wyświetlania i analizy sygnałów z kontrolera robota [3], co umożliwi dalszą optymalizację pracy stanowiska zrobotyzowanego. W zależności od jego



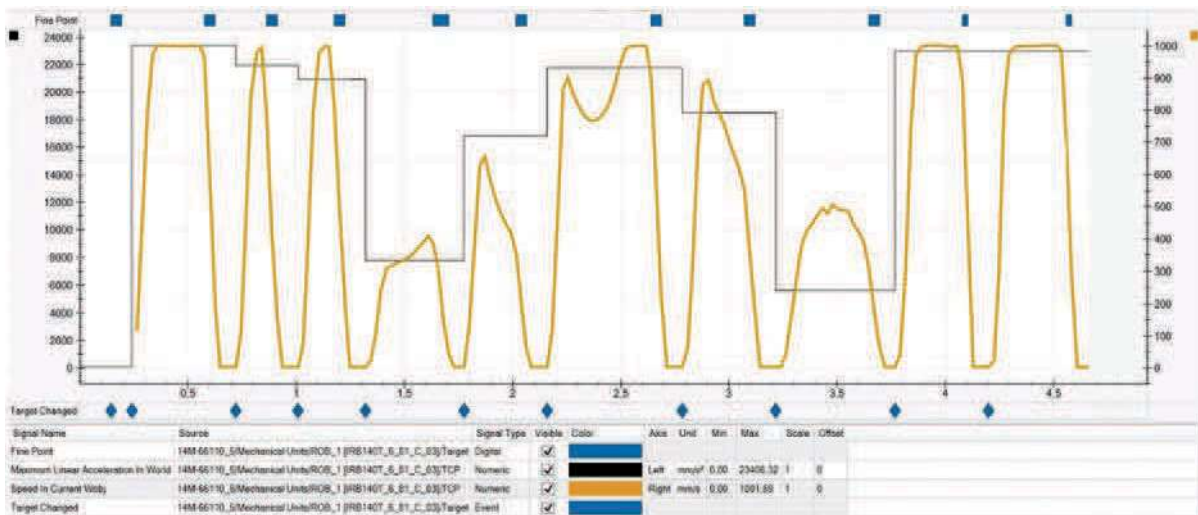
Rys. 4. Ścieżka narzędzia do nakładania masy uszczelniającej [7]

Fig. 4. Tool path for sealant applying [7]

budowy oraz potrzeb analizy możliwe jest śledzenie w czasie rzeczywistym przebiegu sygnałów wejść/wyjść oraz zmian: położenia (narzędzia, osi robota), prędkości ruchu, wymiany narzędzia lub układów współrzędnych.

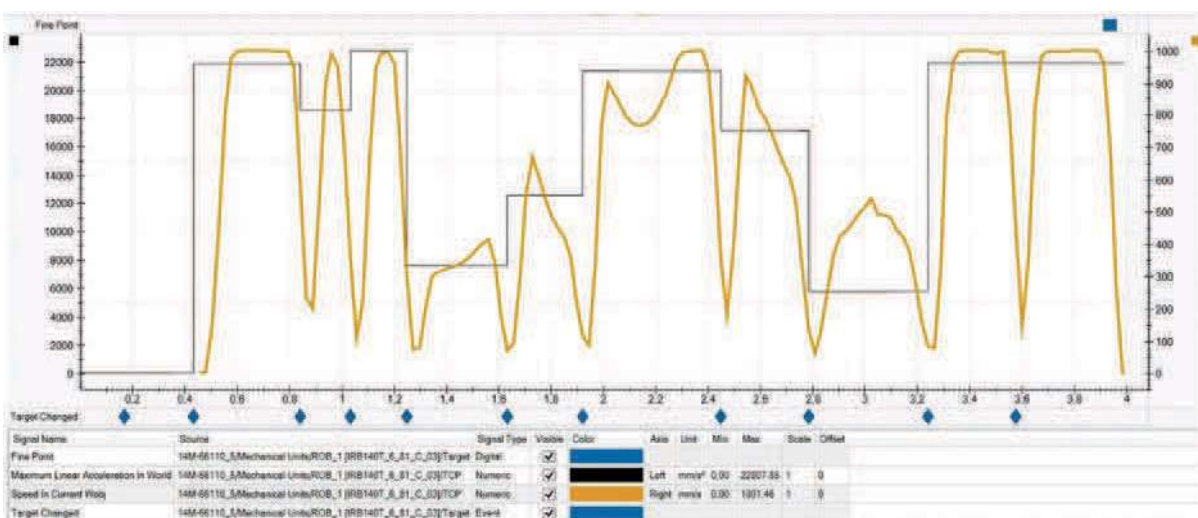
Na rys. 4 przedstawiono ścieżkę narzędzia (wraz z położeniem narzędzia w punktach tworzących trajektorię) do nakładania masy uszczelniającej, która następnie została poddana analizie za pomocą narzędzia Signal Analyzer. W narzędziu tym ustawiono wyświetlanie na wykresach: prędkości i przyspieszenia punktu TCP, punktów z parametrem *fine* oraz zmianę punktu.

Wynik analizy ruchu dla położenia punktów z parametrem *fine* i *v1000* przedstawiono na wykresie (rys. 5) z czego wynika, że ustawiona stała prędkość ruchu (*v1000*) dla wszystkich punktów nie została osiągnięta. Przyczyną może być położenie narzędzia



Rys. 5. Wynik analizy ruchu dla położenia punktów z parametrem *fine* i *v1000* [7]

Fig. 5. The Result of traffic analysis for position of points with *fine* and *v1000* parameters [7]



Rys. 6. Wynik analizy ruchu dla położenia punktów z parametrem *z0* i *v1000* [7]

Fig. 6. The Result of traffic analysis for position of points with *z0* and *v1000* parameters [7]

w punktach trajektorii oraz odległości pomiędzy punktami. Istotne jest również zastosowanie parametru *fine*, który wymusza zatrzymanie robota w każdym punkcie trajektorii. W celu skrócenia czasu przejazdu i poprawienia płynności ruchu w kolejnym kroku zamieniono parametr *fine* na *z0*, a wynik analizy przedstawiono na rys. 6.

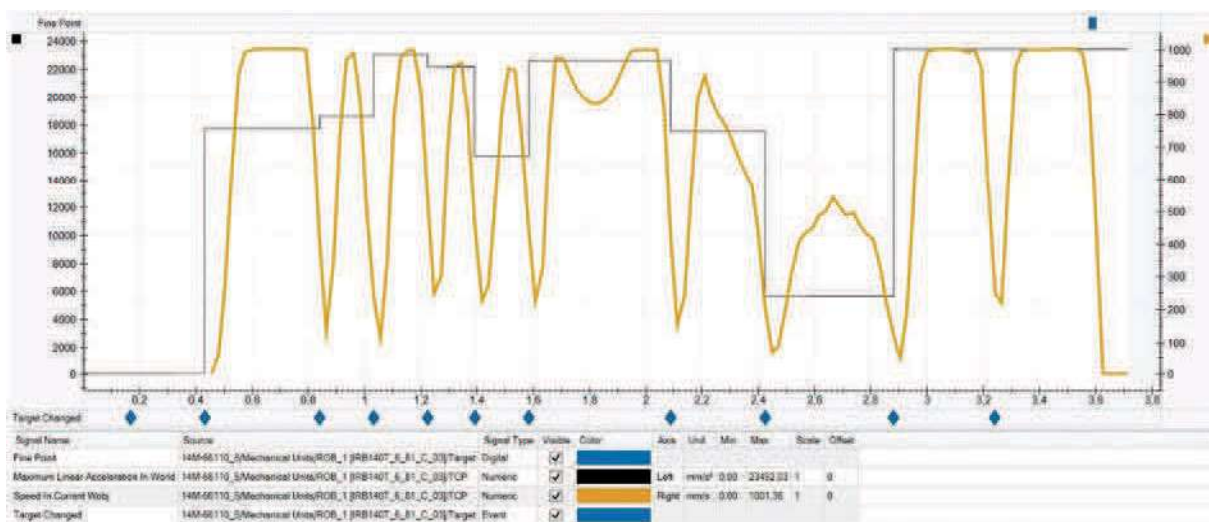
Z analizy rys. 6 wynika, że płynność ruchów uległa poprawie (nie dochodzi do zatrzymania w punktach trajektorii – prędkość przejścia pomiędzy punktami wynosi od 60 do 230 mm/s), jednak nadal nie uzyskano prędkości 1000 mm/s dla całej ścieżki narzędzia. Ponadto nie można zmienić parametru *z* (tj. zwiększyć, tak jak w przypadku montażu korpusów reduktora), gdyż nie uzyska się żądanego kształtu trajektorii (zwiększenie parametru *z* powoduje zaokrąglenie ścieżki). Można natomiast zmienić orientację narzędzia w punkcie, tj. tam, gdzie to możliwe, ustawić jego stałą orientację oraz

zastosować interpolację liniową lub absolutną, która modyfikuje orientację narzędzia podczas ruchu wzdłuż trajektorii.

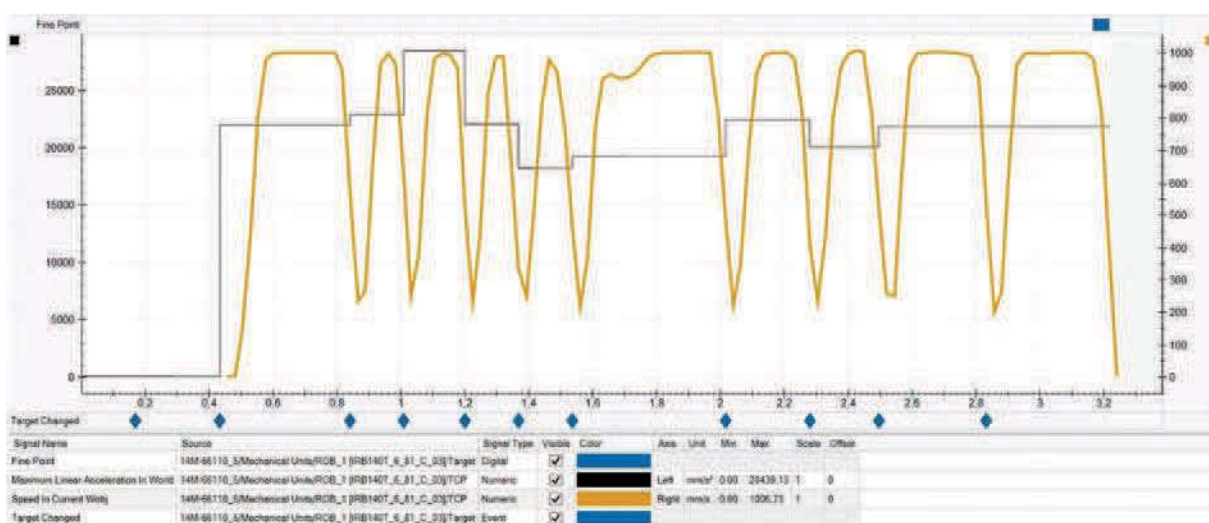
Interpolacja liniowa rozkłada różnicę orientacji równomiernie na podstawie pozycji docelowej wzdłuż długości ścieżki, natomiast interpolacja absolutna rozprowadza równomiernie różnicę orientacji na podstawie sekwencji celów w ścieżce [1].

Na rys. 7 i 8 przedstawiono wykresy analizy w przypadku zastosowania interpolacji liniowej dla wybranych punktów. Na rys. 7 pokazano zastosowanie interpolacji do punktów od 30 do 90, a na rys. 8 interpolacja dotyczyła całej ścieżki narzędzia.

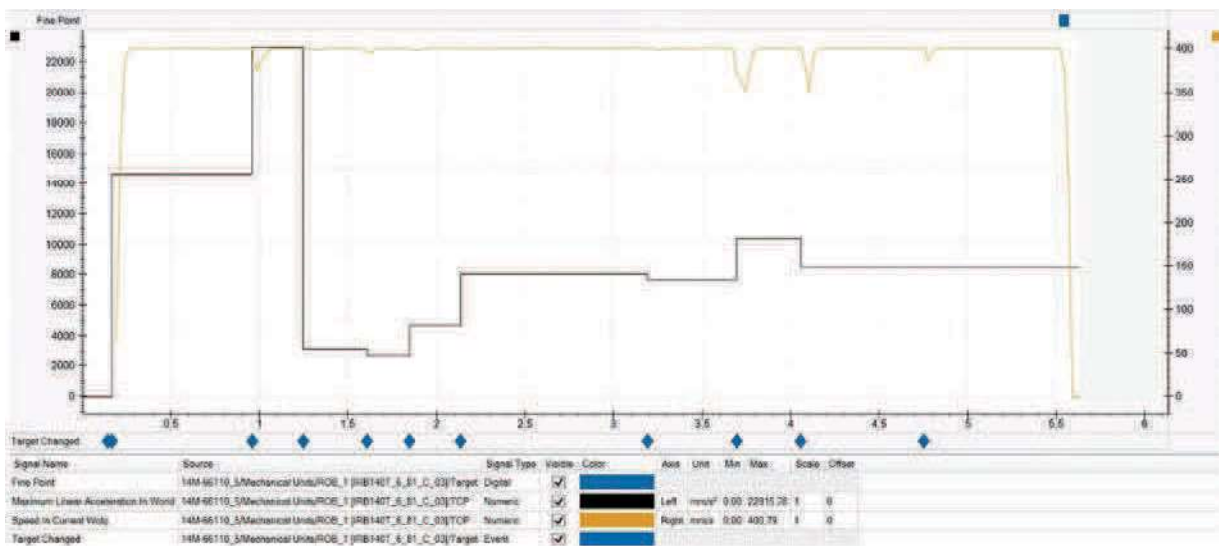
Z rys. 8 wynika, że udało się zwiększyć prędkość ruchu w punktach pośrednich do ok. 200 mm/s. Na rys. 9 przedstawiono wykres ruchu dla prędkości 200 mm/s. Redukcja prędkości spowodowała zmniejszenie charakterystycznych pików oraz poprawę stabilności



Rys. 7. Wynik analizy ruchu w przypadku zastosowania interpolacji liniowej – punkty od 30 do 100 [7]
Fig. 7. The result of traffic analysis for the use of linear interpolation – 30 to 100 points [7]



Rys. 8. Wynik analizy ruchu w przypadku zastosowania interpolacji liniowej – cała ścieżka v1000 [7]
Fig. 8. The result of traffic analysis for the use of linear interpolation – the entire path v1000 [7]



Rys. 9. Wynik analizy ruchu w przypadku zastosowania interpolacji liniowej – cała ścieżka v200 [7]
 Fig. 9. The result of traffic analysis for the use of linear interpolation – the entire path v200 [7]



Rys. 10. Wynik analizy ruchu (inna ścieżka) w przypadku zastosowania interpolacji liniowej – cała ścieżka v400 [7]
 Fig. 10. The result of traffic analysis (other path) for the use of linear interpolation – the entire path v400 [7]

procesu, natomiast znacząco wydłużyła czas operacji, natomiast rys. 10 stanowi przykład innej, już zoptymalizowanej ścieżki o stałej prędkości ruchu wynoszącej 400 mm/s.

Wnioski i podsumowanie

Ustalono, że zastosowanie programu RobotStudio (oprogramowanie do komputerowego wspomaganie programowania robotów przemysłowych) umożliwia pełne odzwierciedlenie elementów stanowiska produkcyjnego oraz zasymulowanie zadań przewidzianych do wykonania. Analiza parametrów prędkości zaprojektowanej ścieżki narzędzia do dozowania masy uszczelniającej umożliwiła skrócenie czasu wykonywania operacji oraz określenie maksymalnej stałej

prędkości, którą należy przypisać do trajektorii ruchu. Narzędzie Signal Analyzer umożliwia również zbieranie informacji z rzeczywistego stanowiska, które następnie mogą być porównane z uzyskanymi z programu RobotStudio.

LITERATURA

- [1] Dokumentacja programu RobotStudio firmy ABB.
- [2] <https://library.e.abb.com/public/e6c1a4a021094f4c9bf614c4b9d956db/ReleaseNotesRobotStudio6-03.pdf>, dostęp 9.04.2017.
- [3] <https://library.e.abb.com/public/980b0e01f1272562c1257b59004271ec/WhatsNewRobotStudio51501.pdf>, dostęp 9.04.2017.
- [4] <https://pl.wikipedia.org/wiki/Jako%C5%9B%C4%87>, dostęp 25.03.2016

- [5] ISO 9283.
- [6] Kost G. G. Reclik D. 2008. „Dynamiczne planowanie trajektorii,” *Przegląd Mechaniczny* (1): 23–29.
- [7] Wiśniewski M., Opracowanie własne.
- [8] World Robotics 2015 Industrial Robots.
- [9] Żurek J. „Metodyka i warunki badań dokładności i powtarzalności pozycjonowania robota przemysłowego Fanuc M-16iB,” *Technologia i Automatyizacja Montażu*, nr 1, pp. 31–34, 2013.

dr inż. Marcin Wiśniewski – Zakład Projektowania Technologii Instytutu Technologii Mechanicznej Politechniki Poznańskiej, ul. Piotrowo 3, 61-138 Poznań, e-mail: marcin.wisniewski@put.poznan.pl

mgr inż. Katarzyna Peta – Zakład Projektowania Technologii Instytutu Technologii Mechanicznej Politechniki Poznańskiej, ul. Piotrowo 3, 61-138 Poznań, e-mail: katarzyna.peta@put.poznan.pl

mgr inż. Arkadiusz Pietrowiak – Zakład Projektowania Technologii Instytutu Technologii Mechanicznej Politechniki Poznańskiej, ul. Piotrowo 3, 61-138 Poznań, e-mail: arkadiusz.pietrowiak@put.poznan.pl

dr hab. inż. Olaf Ciszak – Zakład Projektowania Technologii Instytutu Technologii Mechanicznej Politechniki Poznańskiej, ul. Piotrowo 3, 61-138 Poznań, e-mail: olaf.ciszak@put.poznan.pl

mgr inż. Karol Grochalski – Zakład Metrologii i Systemów Pomiarowych Instytutu Technologii Mechanicznej Politechniki Poznańskiej, ul. Jana Pawła II 24, 60-965 Poznań, e-mail: karol.grochalski@wp.pl