

Universal Robot UR3: wykorzystanie czujnika siły*

Katarzyna Zadarnowska, Michał Dołharz[†]
Laboratorium Robotyki
Wydział Elektroniki
Politechnika Wrocławska

1 Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z możliwościami wynikającymi z zastosowania czujnika siły FT 300 zamontowanego na manipulatorze UR3.

2 Wymagania wstępne

Przed przystąpieniem do realizacji ćwiczenia należy:

- zapoznać się z Zasadami bezpiecznej pracy w Pracowni Robotyki, s.010 C-3,
- znać mechanizmy awaryjnego wyłączenia robota,
- zapoznać się z instrukcją K. Zadarnowska, J. Ratajczak "Universal Robot UR3",
- zapoznać się z niniejszą instrukcją.

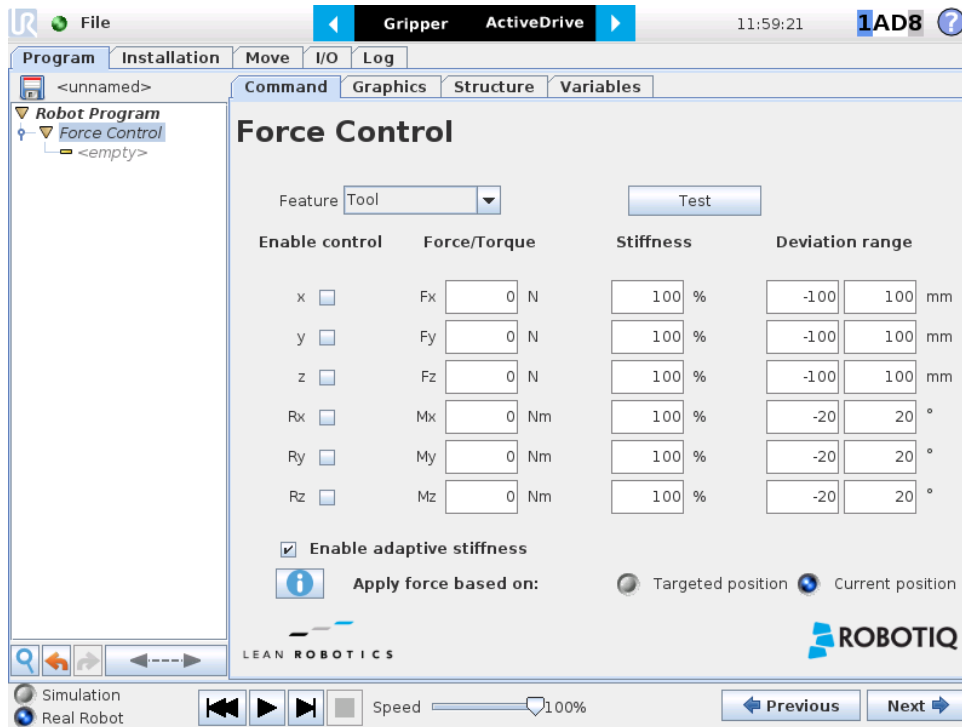
3 Informacje ogólne

Ponieważ manipulator UR3 jest cobotem, posiada wbudowane zabezpieczenia umożliwiające współpracę z człowiekiem. Jednym z takich zabezpieczeń jest szacowanie przykładanej siły na podstawie pomiarów wartości prądów w przegubach. Gdy te pośrednie pomiary siły przekroczą wartości zdefiniowane w ustawieniach bezpieczeństwa robota, na przykład gdy cobot nieoczekiwanie natrafi na człowieka, natychmiastowo przerywa pracę i wycofuje się od miejsca wykrycia siły, a na ekranie panelu operatorskiego pojawia się komunikat `Protective stop`. Pomiary siły uzyskane w ten sposób są przydatne do zgrubnego wykrywania kolizji, jednak wartości są bardzo niedokładne. Z tego powodu został stworzony czujnik siły Robotiq FT 300.

Czujnik Robotiq FT 300 [3,5] to czujnik sił i momentów obrotowych w tym przypadku przeznaczony do pracy z robotami Universal Robots serii CB. Został stworzony z myślą o wykrywaniu z wysoką powtarzalnością zdarzeń wynikających z wystąpienia siły, takich jak lokalizowanie przedmiotów lub elementów otoczenia i podejmowanie stosownych akcji, nie jest natomiast precyzyjnym narzędziem pomiarowym. Jest starszą wersją czujnika Robotiq FT 300-S,

*Ćwiczenie laboratoryjne przeznaczone do realizacji w ramach kursu Robotyka (3) – data ostatniej modyfikacji dokumentu 26 września 2022

[†]Katedra Cybernetyki i Robotyki



Rysunek 1: Interfejs polecenia Force Control

który różni się głównie klasą szczelności. Oba czujniki działają z wykorzystaniem tego samego oprogramowania – Copilot URCaps Package, opartego na rozwiązaniu URCaps od Universal Robots dla producentów osprzętu manipulatorów, takich jak Robotiq.

4 Kontrola sił i momentów sił (Force Control)

Polecenie Force Control pozwala na ustawienie kontroli siły w poszczególnych osiach układu współrzędnych robota (x, y, z) lub momentów sił w obrotach wokół tych osi. Instrukcje, które mają być wykonywane z siłami zaprogramowanymi w Force Control, muszą być umieszczone w węzle omawianego polecenia. Warto zaznaczyć, że polecenia MoveJ, MoveL, MoveP, a więc podstawowe polecenia służące pozycjonowaniu robota, nie mogą działać pod wpływem kontroli sił, a próba zaprogramowania robota w taki sposób zakończy się komunikatem o błędzie. Do pozycjonowania robota z jednoczesną kontrolą siły zostały wprowadzone polecenia Path, Multipoint Path oraz Path Generator i to właśnie te polecenia są pierwszorzędnie przeznaczone do pracy z Force Control.

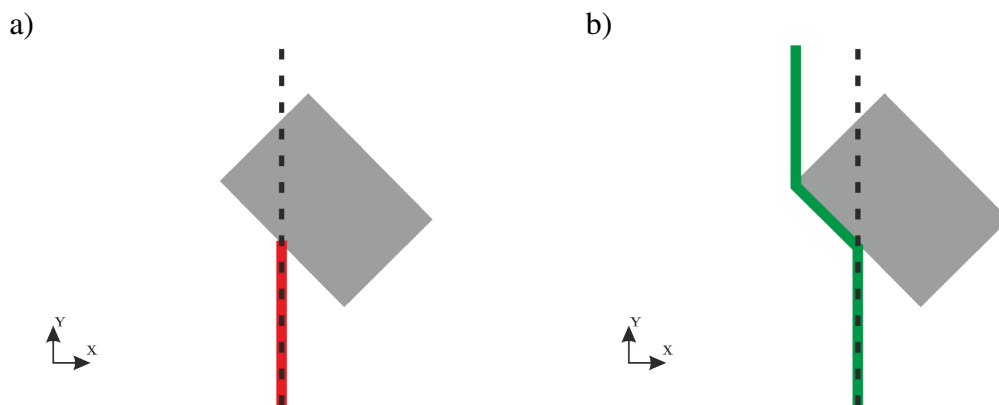
Polecenie Force Control samo w sobie może spowodować ruch robota. Nakazując robotowi utrzymać siłę o podanej przez użytkownika wartości, robot będzie do niej dążył, a więc poruszy się w kierunku wyznaczonym przez tę siłę. Taki ruch jest uzależniony od parametrów polecenia. Interfejs polecenia Force Control z domyślnymi wartościami parametrów został przedstawiony na rysunku 1.

Kontrola każdej z sił oraz każdego momentu obrotowego musi zostać aktywowana, nie wystarczy samo wpisanie żądanych wartości. Do aktywacji służą pola wyboru w kolumnie Enable control. Wartości kontrolowanych sił poszczególnych osi wprowadza się do pól zawartych w kolumnie Force/Torque. Przed każdym polem znajduje się również nazwa zmiennej, do której zapisywane są wartości. Zmienne te są wyświetlane w zakładce Variables oraz w zakładce Installation/Copilot i w trakcie wykonywania programu są na bieżą-

co aktualizowane. Wartości kontroli sił w osiach dozwolone są w zakresie od $-150N$ do $150N$, a wartości momentów obrotowych w zakresie od $-50Nm$ do $50Nm$.

Parametr *Stiffness* określa sztywność ruchu robota w poszczególnych osiach. Sztywność osi można definiować jako stopień podatności ruchu robota na działające siły zewnętrzne. Domyślna wartość to 100% dla każdej osi, a więc całkowita sztywność. W takiej sytuacji ruch robota nie będzie zależał od zewnętrznych sił, robot nie będzie więc miał możliwości adaptacji do otoczenia. Przykładowa sytuacja zastosowania sztywności 100% została przedstawiona na rysunku 2a.

Im mniejsza wartość sztywności, tym bardziej ruch robota jest elastyczny i zależny od otoczenia. Wartość 0% oznacza całkowity brak sztywności. Ruch robota w osi z zerową sztywnością będzie całkowicie zależny od czynników zewnętrznych, będzie więc dostosowywał się do otoczenia. Gdy robot natrafi na przeszkodę, w zależności od konfiguracji polecenia oraz samej przeszkody, istnieje szansa, że ją ominie. W takim przypadku jednak po ominięciu przeszkody nie powróci na dawną pozycję, ponieważ jego ruch w tej osi zależy w pełni od sił zewnętrznych. Taka sytuacja została przedstawiona na rysunku 2b.



Rysunek 2: Ruch robota po zaprogramowaniu trajektorii wzdłuż osi OY (czarna przerywana linia) – a) Sztywność 100% w osi OX. Robot nie może dostosować się do otoczenia, więc po natrafieniu na przeszkodę nie może się dalej poruszyć; b) Sztywność 0% w osi OX. Robot po natrafieniu na przeszkodę dostosuje się i będzie poruszał się wzdłuż przeszkody, jednak ze względu na brak innych sił w osi OX nie będzie potrafił wrócić i kontynuować ruchu wzdłuż pierwotnej trajektorii

Zaznaczenie pola wyboru *Enable adaptive stiffness* włącza adaptacyjną sztywność w osiach, w których aktywna jest kontrola sił. Wartości wprowadzone w polach kolumny *Stiffness* tych osi nie będą stosowane.

Na początek jest stosowana sztywność 0%. Przy pierwszym kontakcie z obiektem i wystąpieniu siły, która została wprowadzona do kolumny *Force/Torque* dla danej osi, robot wycofa efektor, a następnie procedurę będzie powtarzał dla coraz większej wartości sztywności, aż zaadaptuje się do badanego obiektu. W ten sposób odbywa się uwzględnienie odpowiedniej sztywności dla obiektu, z którym robot ma do czynienia, a co za tym idzie ustawienie odpowiedniej sztywności w poszczególnych osiach robota tak, by uwzględnić obecność obiektu/przeszkody.

Ze względu na dokładność wyznaczonej trajektorii ruchu robota, należy wybrać jedną z dwóch opcji, *Current position* bądź *Targeted position*. Jeżeli trajektoria ruchu jest zadana precyzyjnie, należy wybrać opcję *Targeted position*. Robot najpierw poruszy się do punktu startowego i dopiero wtedy zacznie kontrolować siłę/momenty sił. Ruch odbędzie się zgodnie z programem. Jeżeli jednak trajektoria nie jest zadana dokładnie, należy wybrać opcję *Current*

position. W takim przypadku robot ignoruje te współrzędne punktów trasy, które odpowiadają osiom, w których kontrolowana jest siła lub moment obrotowy. Zakłada się więc, że robot dotrze do prawidłowej pozycji w tych osiach na skutek działania sił pochodzących z Force Control. Pozostałe współrzędne nadal obowiązują i robot przesunie się w ich kierunku, zanim rozpocznie kontrolę sił.

Układ współrzędnych, którego dotyczy kontrola sił, można umiejscowić w trzech miejscach.

- Base – układ współrzędnych przy podstawie robota.
- Tool – układ współrzędnych narzędzia (przy chwytaku). Przy korzystaniu z tej opcji bardzo wygodne jest oznakowanie poszczególnych osi na chwytaku.
- Motion – układ współrzędnych wzdłuż trajektorii ruchu. W tym przypadku oś X jest zawsze skierowana wzdłuż kierunku ruchu, a oś Z jest zbliżona do osi Z robota.

Możliwe jest przetestowanie wprowadzonych wartości, bez konieczności uruchamiania programu. Służy do tego przycisk Test. Testując zaprogramowane polecenie, ruch robota może odbyć się w sposób nieoczekiwany, dlatego zawsze należy zachować ostrożność.

5 Zerowanie sensora (Zero FT Sensor)

Ważne jest, by przed każdym poleceniem związanym z pomiarem sił wyzerować czujnik. Służy do tego polecenie Zero FT Sensor. Niektóre z dostępnych poleceń automatycznie dodają do programu to polecenie. Inne posiadają w swoim interfejsie pole wyboru, po którego zaznaczeniu czujnik również będzie się zerował, natomiast polecenie zerowania nie zostanie dodane do drzewa programu. Niektóre polecenia nie mają żadnej z tych opcji i wówczas to użytkownik powinien pamiętać o zerowaniu wartości czujnika.

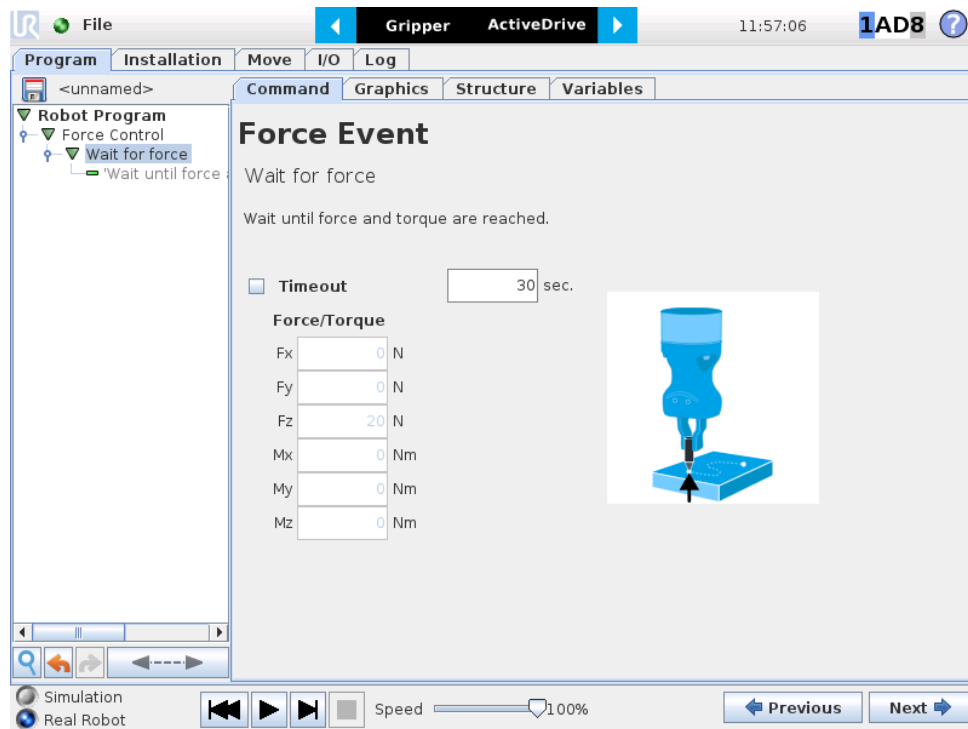
Interfejs polecenia zawiera jedynie pole wyboru, w którym użytkownik może zdecydować, czy zerowanie czujnika powinno odbyć się dopiero gdy ustabilizują się wskazania czujnika. Należy pamiętać, że odczyty nigdy nie będą równe 0. Nawet po wyzerowaniu zawsze będą oscylować wokół zera, pokazując zazwyczaj ułamki niutonów. Czujnik nie powinien być zerowany w trakcie wykonywania ruchu robota. Nie powinien być również zerowany w momencie, gdy na czujnik oddziałuje jakaś siła.

6 Oczekiwanie na wykrycie siły (Wait for force)

Polecenie Wait for force realizuje funkcję oczekiwania na wykrycie siły o wartości zadanej przez użytkownika. Jest jedną z dwóch możliwości konfiguracji polecenia Force event.

Interfejs polecenia jest przedstawiony na rysunku 3. Instrukcja ta co prawda posiada pola z wartościami poszczególnych sił, ale nie pozwala na ich edycję, a jedynie wyświetlanie. Wartości sił są zamiast tego pobierane z konfiguracji polecenia Force control, pod którego wpływem znajduje się funkcja Wait for force. Jedyнным parametrem dostępnym z poziomu interfejsu tego polecenia jest Timeout - maksymalny czas oczekiwania na wykrycie sił, jego domyślna wartość to 30sec.

W domyślnie skonfigurowanym poleceniu robot zacznie poruszać się w osi, w której siła ma zostać wykryta. Jest to efekt działania polecenia Force Control. Możliwe jest jednak unieruchomienie efektora w taki sposób, aby oczekiwanie na wykrycie siły odbywało się zawsze w tym samym miejscu przestrzeni roboczej. Aby tego dokonać, należy ustawić odpowiedni parametr kolumny Deviation range na wartość zero milimetrów. Osiągnięcie wyznaczonej wartości granicznej, w tym przypadku 0[mm], nie spowoduje zakończenia wykonywania



Rysunek 3: Interfejs polecenia Wait for force

polecenia, więc oczekiwanie na siłę może zostać zakończone poprzez faktyczne wykrycie siły lub poprzez fakt ograniczenia czasu działania Timeout.

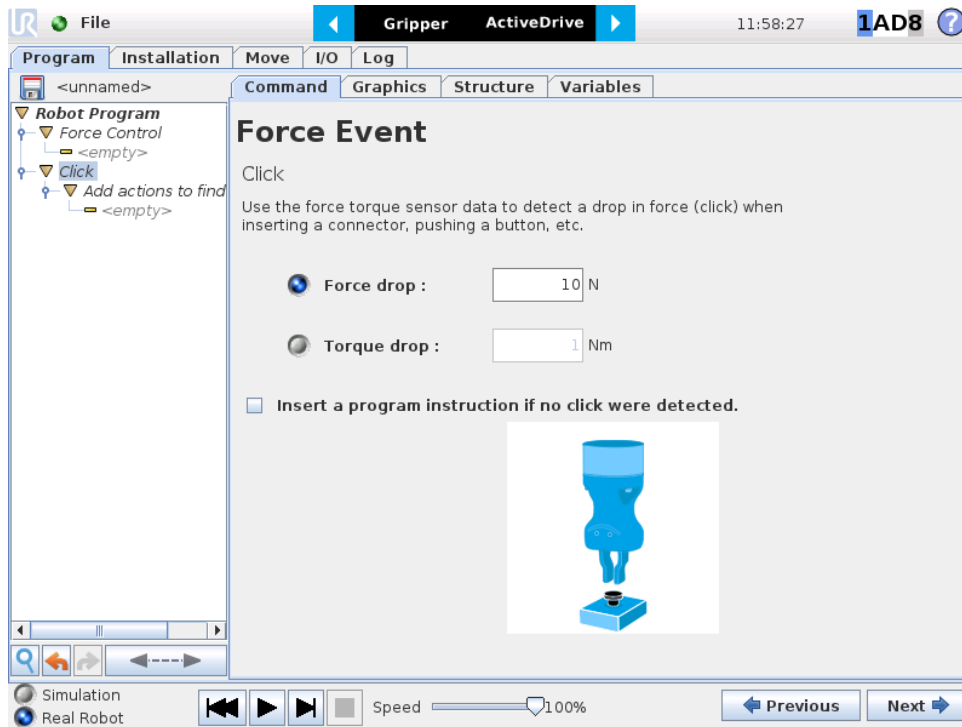
7 Wykrywanie kliknięcia (Click detection)

Wykrywanie kliknięcia odbywa się na podstawie nagłego spadku odczytywanej wartości siły. Służy do tego polecenie Click detection będące jedną z dwóch możliwych konfiguracji polecenia Force event. Interfejs polecenia jest przedstawiony na rysunku 4.

Interfejs pozwala zdecydować użytkownikowi, czy ma być wykryte kliknięcie pojawiające się na skutek działania sił wzdłuż osi Z w układzie współrzędnych narzędzia Tool, czy momentów sił działających wokół wymienionej osi. Nie jest możliwe jednoczesne wykrywanie kliknięcia pojawiającego się na skutek działania obu tych sił. Wyboru dokonuje się przyciskami opcji Force drop i Torque drop. Zaraz obok znajdują się pola, w których definiuje się wartość spadku siły oznaczającego wykrycie kliknięcia. Domyślna wartość dla siły to 10N, a wartości dozwolone są z zakresu od 0N do 100N. Domyślna wartość dla momentu obrotowego to 1Nm, a wartości dozwolone są z zakresu od 0Nm do 50Nm.

Instrukcja posiada opcjonalną możliwość podjęcia kroków w przypadku, gdy kliknięcie nie zostanie wykryte. Po zaznaczeniu tej opcji w interfejsie polecenia, do programu zostaną dodane dodatkowe linijki pod instrukcją warunkową podpisaną If not clicked.

Po skonfigurowaniu polecenia Force Event jako Click detection, do programu zostanie dodana linia Add actions to find a click. Pod nią powinny znaleźć się instrukcje, które powinny prowadzić do kliknięcia, które ma zostać wykryte. Innymi słowy, jest to część programu, w trakcie wykonywania której będzie wykrywane kliknięcie. Warto zaznaczyć, że do zaprogramowania akcji prowadzącej do wykrycia kliknięcia mogą zostać wykorzystane polecenia MoveJ, MoveL oraz MoveP.



Rysunek 4: Interfejs polecenia Click detection

8 Wykrywanie powierzchni (Find Surface)

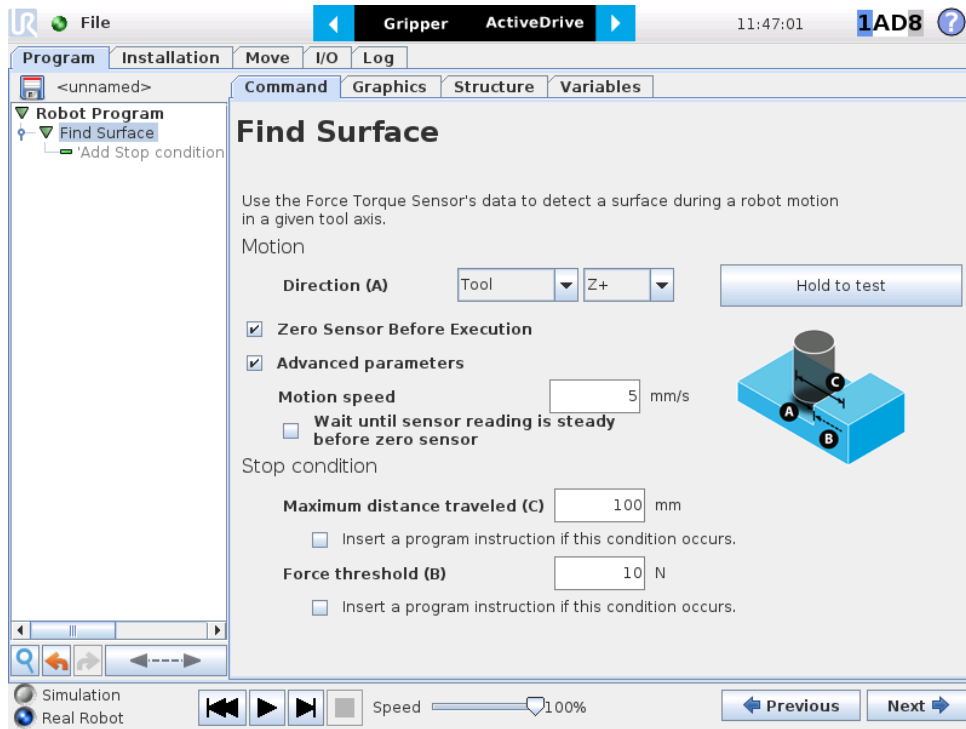
Odpowiedni dobór parametrów polecenia pozwala na ogólne wykrywanie obecności obiektów, dzięki czemu może być fundamentem pod bardziej złożone programy. Funkcja zapewnia wysoką powtarzalność w dokładnym pozycjonowaniu obiektów. Polecenie odnajdowania powierzchni może przypominać polecenie oczekiwania na siłę `Wait for force`, jednak oba polecenia różnią się interfejsem i parametrami oraz przeznaczeniem. W swojej prostszej formie polecenie posiada jedynie trzy parametry, natomiast po zaznaczeniu pola wyboru `Advanced parameters` wyświetlone zostaną dodatkowe parametry definiujące ruch i zachowanie robota w sposób bardziej szczegółowy. Interfejs polecenia jest przedstawiony na rysunku 5.

Aby ułatwić użytkownikowi konfigurację polecenia, po prawej stronie interfejsu znajduje się grafika obrazująca niektóre z parametrów. Parametry te są oznaczone na grafice (animacji) wielkimi literami i te same oznaczenia są wykorzystywane w interfejsie.

Wciśnięty przycisk `Hold to test` pozwala przetestować skonfigurowane przez użytkownika polecenie w praktyce (bez uruchamiania całego programu).

Kierunek ruchu robota jest zadawany poprzez dwie listy rozwijane, z których pierwsza dotyczy środka współrzędnych układu (narzędzia `Tool` lub podstawy robota `Base`) względem którego ma się odbyć ruch, a druga - kierunku ruchu w postaci oznaczenia osi wraz ze znakiem. Domyślnie parametr ten jest ustalony jako ruch w kierunku `Z+` dla środka współrzędnych narzędzia `Tool`.

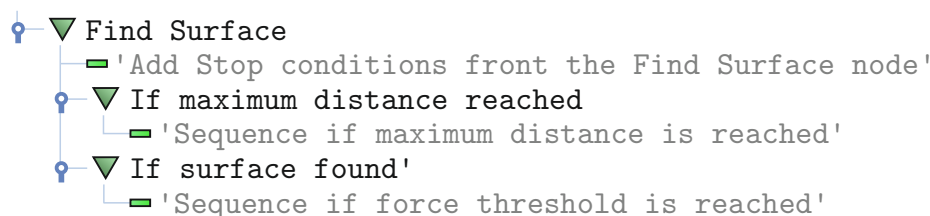
Przed użyciem czujnika zawsze należy go wyzerować. Interfejs polecenia dostarcza opcje zerowania czujnika w postaci pojedynczego pola wyboru, podpisanego `Zero Sensor Before Execution`. Opcja ta jest równoznaczna z poleceniem `Zero FT Sensor`, jednak w przeciwieństwie do niego nie dodaje kolejnej linii do drzewa programu robota. Domyślnie opcja zerowania czujnika jest włączona. Dostępna jest również opcja odczekania z wyzerowaniem czujnika do momentu, gdy odczyty przestaną się znacząco zmieniać. Wyświetlana jest po włączeniu zaawansowanych parametrów interfejsu, w postaci pola wyboru podpisanego `Wait until`



Rysunek 5: Interfejs polecenia Find Surface

sensor reading is steady before zero sensor. Jest odpowiednikiem pola wyboru `Wait until sensor reading is steady` polecenia `Zero FT Sensor`.

Prędkość ruchu robota określana jest w milimetrach na sekundę. Domyślnie jest to wartość 5mm/s , natomiast dozwolone są wartości z zakresu od 0.1mm/s do 150.0mm/s . Maksymalny przebyty dystans to parametr oznaczony jako `Maximum distance traveled`. Jego dozwolone wartości są z zakresu od 0.5mm do 1900.0mm . Domyślna wartość to 100mm . Jeżeli zostanie osiągnięty maksymalny możliwy dystans, robot uzna polecenie wyszukiwania powierzchni za zakończone i przejdzie do dalszej realizacji programu. Dostępna jest jednak opcja dodania do programu instrukcji, które zostaną wykonane, gdy warunek maksymalnego przebytego dystansu zostanie spełniony. Służy do tego pole wyboru poniżej omawianego parametru. Po zaznaczeniu w drzewie programu pojawi się węzeł z dodatkowym warunkiem podpisanym `If maximum distance is reached` i miejscem do zdefiniowania dalszych instrukcji przez użytkownika, przedstawiony na rysunku 6.



Rysunek 6: Struktura węzła polecenia Find Surface z dodatkowymi węzłami obsługi przebycia maksymalnego dystansu oraz wykrycia powierzchni

Wartość siły, której odczyt jest uznawany za kontakt z powierzchnią, a więc jej wykrycie, nazwany jest `Force threshold` i jest z zakresu od 0.2N do 200.0N , a jej domyślna wartość to 10N . Po wykryciu powierzchni robot przejdzie do dalszej realizacji programu. Istnieje opcja dodania specjalnych linii w programie, gdy powierzchnia zostanie odnaleziona. Służy do tego

pole wyboru poniżej omawianego parametru. Po jego zaznaczeniu, w drzewie programu pojawi się węzeł z dodatkowym warunkiem `If surface found` i miejscem do zdefiniowania dalszych instrukcji przez użytkownika (patrz rysunek 6).

9 Monitorowanie kolizji (**Collision Detection**)

Wykrywanie kolizji jest przydatną funkcją, gdy ważne jest zapewnienie bezpieczeństwa zarówno dla otoczenia, jak i dla samego robota. Należy jednak pamiętać, że odczyty dla tego polecenia pobierane są z czujnika FT 300, więc tylko kolizje końcówki robota zostaną wykryte.

Polecenie `Collision Detection` działa na zasadzie wątku pracującego w tle programu, musi więc zostać włączony (przycisk `Start monitoring`) i może być wyłączony (przycisk `Stop monitoring`). Wyłączenie wątku nie jest konieczne, zakończenie wykonywanego programu automatycznie wyłącza wątek monitorujący kolizje. Ponieważ pola wyboru nie mogą zostać zaznaczone jednocześnie, program wymaga zastosowania dwóch instrukcji `Collision Detection` do aktywacji i deaktywacji wątku wykrywającego kolizję.

Obowiązkowe jest podanie zarówno wartości sił, jak i momentów obrotowych, których wystąpienie będzie jednoznaczne z wystąpieniem kolizji. Dla sił domyślna wartość to $140N$, a dozwolone są wartości z zakresu od $1N$. Dla momentów sił domyślna wartość to $14Nm$ natomiast dozwolone wartości są z zakresu od $0.02Nm$ do $30Nm$.

Wykrycie kolizji może skutkować jedną z dwóch opcji, wybieranych za pomocą przycisków opcji:

- `Halt on collision detection` – program zakończy się po wykryciu kolizji.
- `Suspend on collision detection` – program zatrzyma się po wykryciu kolizji i będzie możliwe jego wznowienie.

10 Umieszczanie obiektów w otworach (**Insertion**)

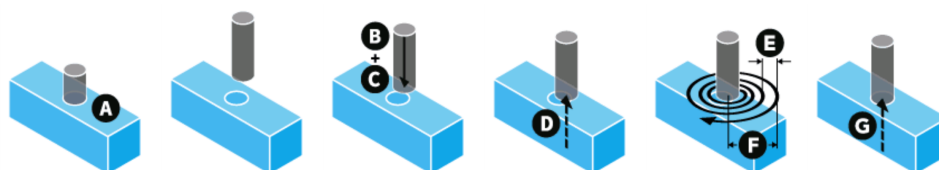
Oprogramowanie czujnika dostarcza trzy polecenia obsługujące wkładanie obiektów do przeznaczonych im gniazd.

- Wyszukiwanie otworu (`Spiral`) – kiedy obiekt jest dopasowany do otworu, ale niekoniecznie wycentrowany względem niego.
- Dopasowywanie do otworu (`Rotational`) – kiedy obiekt jest wycentrowany względem otworu, ale niekoniecznie dobrze obrócony względem osi otworu.
- Wkładanie do otworu (`Linear`) – kiedy obiekt jest dopasowany do otworu i wycentrowany względem niego, ale nie jest znany dystans, o jaki trzeba obiekt opuścić.

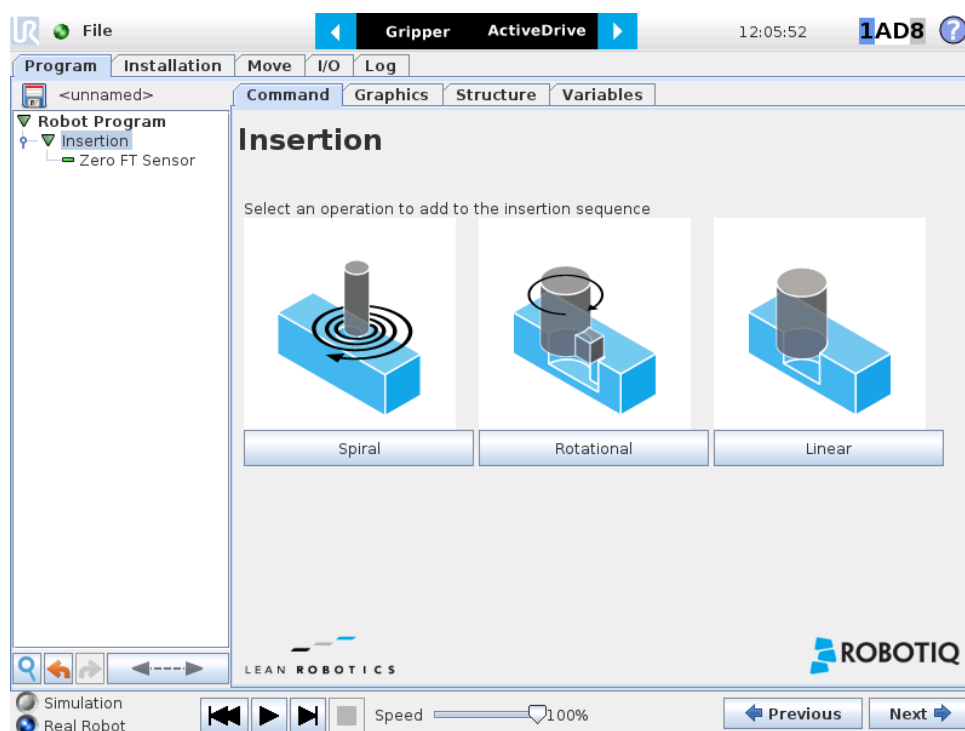
Omawiane polecenia są zgrupowane pod ogólnym poleceniem `Insertion`. Interfejs tego polecenia składa się z trzech przycisków pod animacjami symbolizującymi omawiane funkcje (patrz rysunek 8). Kliknięcie każdej z nich dodaje odpowiednie polecenia do programu. Każda z nich posiada również parametry określane jako zaawansowane, do których dostęp uzyskuje się po zaznaczeniu pola wyboru `Advanced parameters`.

Przed każdą z tych funkcji automatycznie dodawane jest polecenie `Zero FT Sensor`. Wszystkie posiadają również instrukcje podejmowane w razie niepowodzenia wykonywania, dodawane automatycznie do drzewa programu. Domyślnie jest to zakończenie wykonywania programu.

Wszystkie trzy funkcje posiadają w swoim interfejsie poklatkową animację przedstawiającą niektóre z parametrów. Przykładowa animacja została przedstawiona (w postaci poszczególnych klatek) na rysunku 7. Jeżeli przy nazwie parametru znajduje się oznaczenie w postaci wielkiej litery w nawiasie, oznacza to, że taki parametr jest przedstawiony w animacji.



Rysunek 7: Poszczególne klatki animacji pomocniczej polecenia *Spiral*



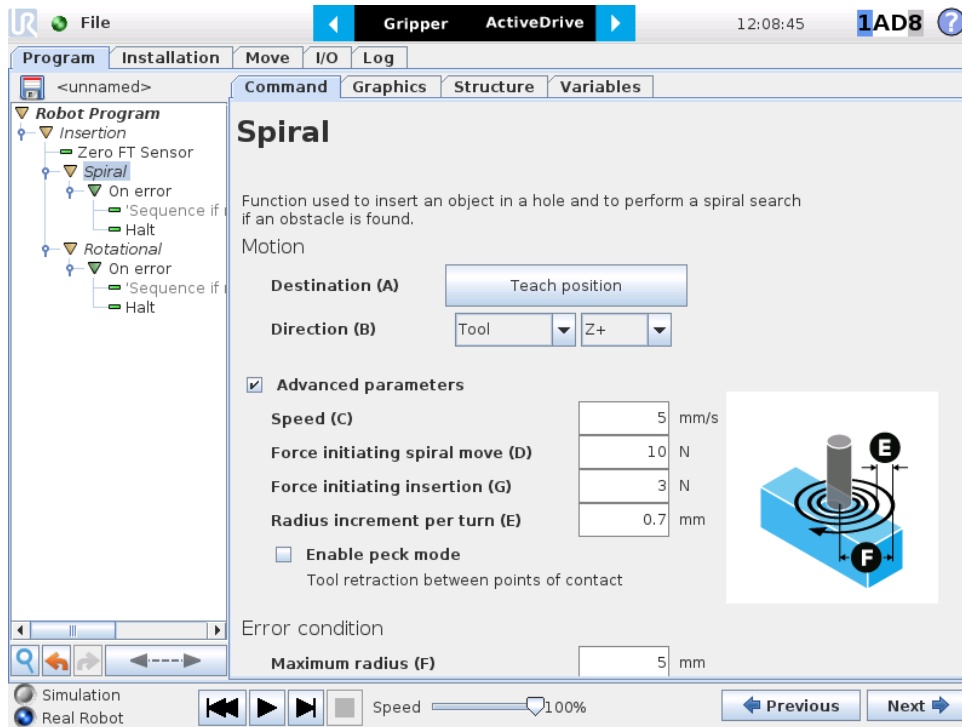
Rysunek 8: Interfejs polecenia *Insertion*

10.1 Wyszukiwanie otworu (*Spiral*)

Wyszukiwanie otworu odbywa się dzięki poleceniu *Spiral*. Po wstępnym naprowadzeniu na otwór robot zaczyna wykonywać spiralny ruch i kontynuuje go, dopóki wykrywa siłę reakcji podłoża. Brak tej reakcji oznacza, że otwór został znaleziony i rozpoczyna opuszczanie obiektu do gniazda. Interfejs polecenia jest przedstawiony na rysunku 9.

Parametr *Destination* (A) to docelowa pozycja, w której obiekt ma zostać umieszczony w gnieździe. Ustawiając ten punkt warto zwrócić uwagę w jaki sposób obiekt został schwytyany, a więc czy uda się zapewnić powtarzalność wykonywania programu.

Kierunek ruchu robota jest określany poprzez dwie listy rozwijane, z których pierwsza dotyczy środka układu współrzędnych (narzędzia lub podstawy robota), a druga kierunku ruchu w postaci oznaczenia osi wraz ze znakiem. Parametr ten jest podpisany jako *Direction* (B). Domyślnie parametr ten jest ustalony jako ruch w kierunku Z+ dla środka współrzędnych narzędzia *Tool*, a więc ruch pionowo w dół.



Rysunek 9: Interfejs polecenia Insertion Spiral

Prędkość ruchu robota *Speed (C)* określana jest w milimetrach na sekundę. Domyślnie jest to wartość 5mms , natomiast dozwolone są wartości z zakresu od 0.1mms do 20mms .

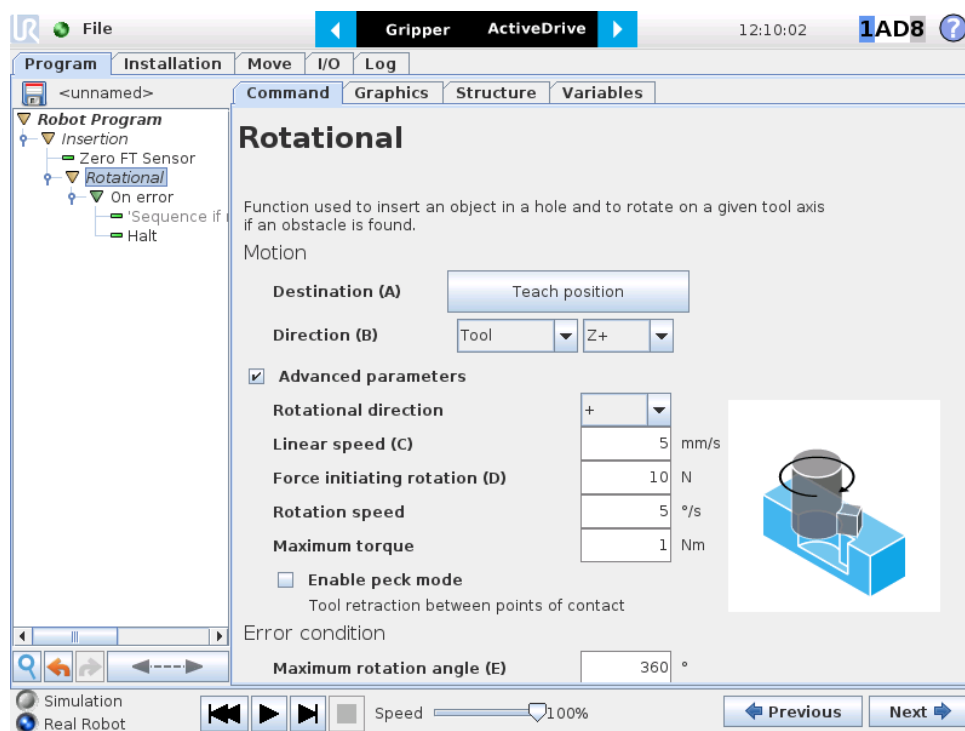
Parametr *Force initiating spiral move (D)* określa wartość siły reakcji powierzchni na nacisk robota, w wyniku wykrycia której robot rozpocznie ruch spiralny, a więc wyszukiwanie otworu. Jej domyślna wartość to 10N . Dozwolone są wartości z zakresu od 0.5N do 150N

Robot wykonuje ruch spiralny ze stałym naciskiem, którego wartość ustala parametr *Force initiating spiral move (D)*. Wykrycie otworu następuje, gdy wartość odczytywanej siły spadnie o wartość podaną w parametrze *Force initiating insertion (G)*. Jego domyślna wartość to 3N . Dozwolone są wartości z zakresu od 0.5N do 40N . Po wykryciu otworu robot rozpocznie opuszczanie obiektu do gniazda.

Parametr *Radius increment per turn (E)* pozwala określić, o ile milimetrów ma zostać zwiększony promień ruchu spiralnego na każdy obrót. Domyślnie jest to 0.7mm . Dozwolone są wartości z zakresu od 0.05mm do 5mm .

Standardowo, podczas wykonywania spiralnego ruchu utrzymywany jest stały kontakt obiektu trzymanego przez robota z powierzchnią, w której jest wyszukiwany otwór. Ponieważ taki kontakt nie zawsze może być utrzymywany, na przykład ze względu na wrażliwą konstrukcję powierzchni, dostępna jest opcja ograniczenia kontaktu obiektu z powierzchnią. Dokonuje się tego zaznaczając pole wyboru *Enable peck mode*. W tym trybie po każdym kontakcie robot odsunie się od powierzchni, przesunie o pewną odległość i ponownie doprowadzi do kontaktu w celu sprawdzenia obecności otworu.

Wykonywanie ruchu spiralnego ograniczone jest maksymalnym promieniem tego ruchu, zadawanym w parametrze *Maximum radius (F)*. Parametr jest wyrażany w milimetrach, a jego domyślna wartość to 5mm . Dozwolone są wartości z zakresu od 0.5mm do 10mm . Osiągnięcie tej wartości promienia ruchu spiralnego spełnia warunek wystąpienia błędu zadeklarowanego automatycznie (w efekcie wybrania polecenia *Insertion Spiral*) w drzewie programu.



Rysunek 10: Interfejs polecenia Insertion Rotational

10.2 Dopasowywanie do otworu (Rotational)

Dopasowywanie do otworu odbywa się dzięki poleceniu `Rotational`. Po wycentrowaniu obiektu względem otworu robot rozpoczyna ruch obrotowy i kontynuuje go, dopóki wykrywa siłę reakcji powierzchni. Brak tej reakcji oznacza, że obiekt został dopasowany do otworu i robot może rozpocząć opuszczanie obiektu do środka. Interfejs polecenia jest przedstawiony na rysunku 10.

Parametr `Destination (A)` to docelowa pozycja, w której obiekt jest umieszczony w otworze. Ustawiając ten punkt warto zwrócić uwagę w jaki sposób obiekt został schwytyany, a więc czy uda się zapewnić powtarzalność wykonywania programu.

Kierunek ruchu robota jest określany poprzez dwie listy rozwijane, z których pierwsza dotyczy środka układu współrzędnych (narzędzia lub podstawy robota), a druga kierunek ruchu w postaci oznaczenia osi wraz ze znakiem. Parametr ten jest podpisany jako `Direction (B)`. Domyślnie parametr ten jest ustalony jako ruch w kierunku `Z+` dla środka współrzędnych narzędzia `Tool`, a więc ruch pionowo w dół.

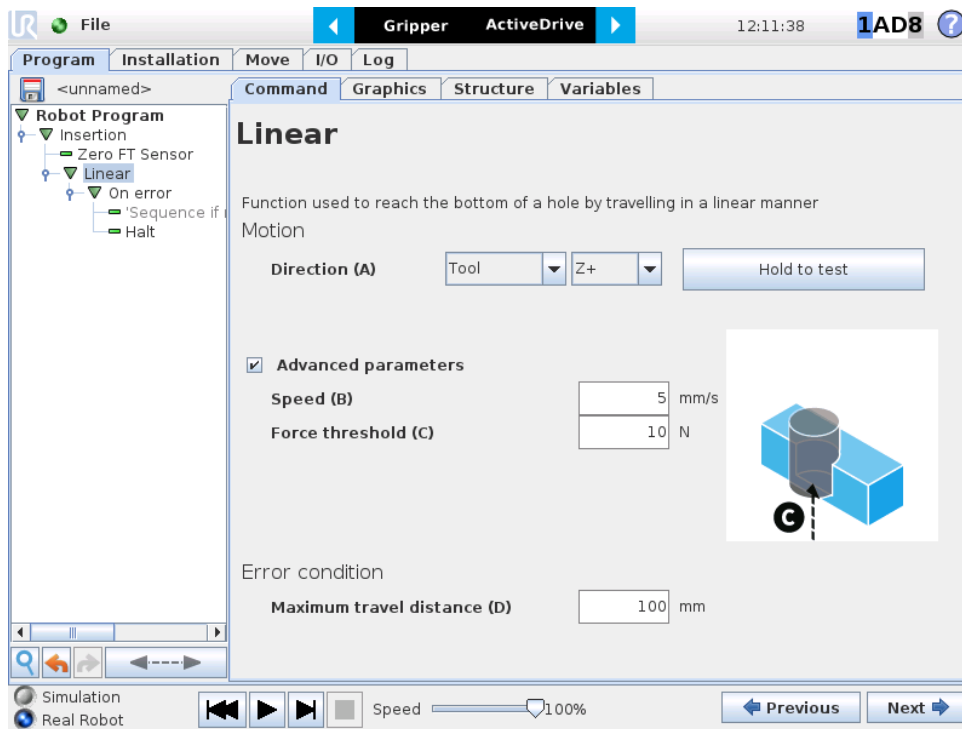
Kierunek obracania obiektu wokół osi wybranej w parametrze `Direction (B)` jest zależny od wybranej opcji z listy rozwijanej `Rotational direction`. Dostępne są dwie możliwości, "+" oraz "-", odpowiadające kierunkowi obrotów wokół osi.

Prędkość ruchu wkładającego obiekt do otworu `Linear speed (C)` przyjmuje domyślną wartość 5mm/s , natomiast dozwolone są wartości z zakresu od 0.1mm/s do 150mm/s .

Parametr `Force initiating rotation (D)` określa wartość siły reakcji powierzchni na nacisk robota, w wyniku wykrycia której robot rozpocznie ruch obrotowy, a więc dopasowywanie obiektu do otworu. Parametr ten jest wyrażony w niutonach, a jego domyślna wartość to 10N , natomiast dozwolone są wartości z zakresu od 0.5N do 200N .

Prędkość ruchu obrotowego `Rotation speed` przyjmuje domyślną wartość 5deg/s , natomiast dozwolone są wartości z zakresu od 0.1deg/s do 45deg/s .

W trakcie wykonywanego ruchu obrotowego może dojść do niechcianych kolizji, dlatego



Rysunek 11: Interfejs polecenia Insertion Linear

parametr `Maximum torque` określa maksymalną wartość momentu obrotowego, z jakim obraca się robot. Domyślnie jest to wartość $1Nm$, natomiast dozwolone są wartości z zakresu od $0.01Nm$ do $15Nm$.

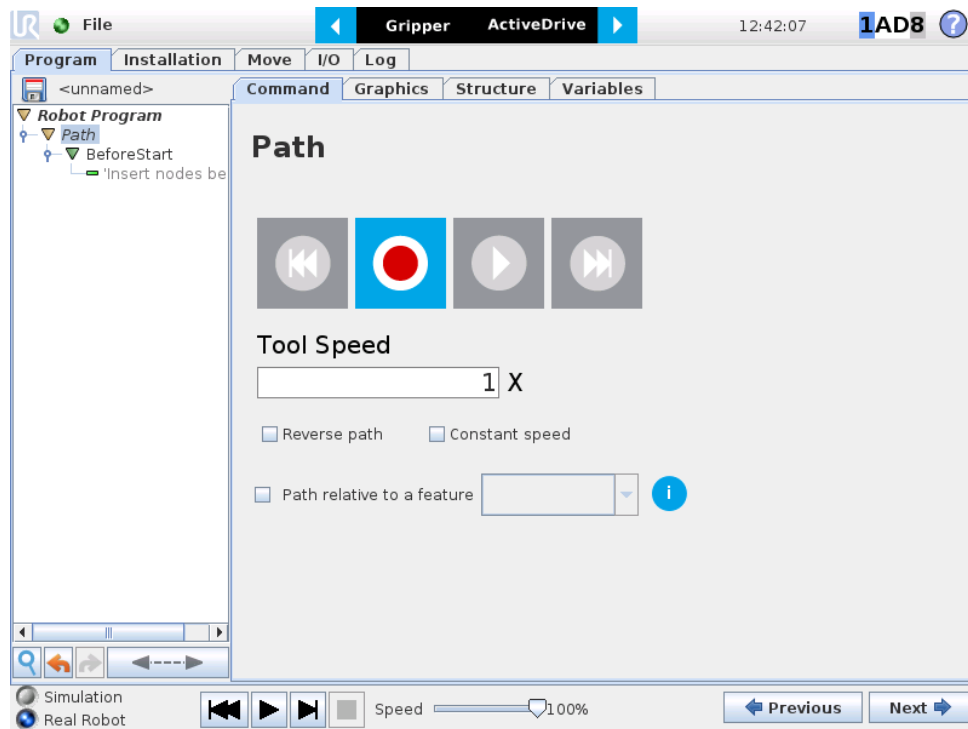
Standardowo, podczas wykonywania ruchu obrotowego utrzymywany jest stały kontakt obiektu trzymanego przez robota z powierzchnią. Ponieważ taki kontakt nie zawsze może być utrzymywany, na przykład ze względu na wrażliwą konstrukcję powierzchni, dostępna jest opcja ograniczenia kontaktu obiektu z powierzchnią. Dokonuje się tego zaznaczając pole wyboru `Enable peck mode`. W tym trybie po każdym kontakcie robot odsunie się od powierzchni, obróci o niewielki kąt i ponownie doprowadzi do kontaktu w celu sprawdzenia dopasowania do otworu.

Parametr `Maximum rotation angle (E)` określa maksymalny kąt wyrażony w stopniach, o jaki robot może obrócić obiekt w celu dopasowania do otworu. Domyślnie jest to wartość 360° , natomiast dozwolone są wartości z zakresu od 1° do 360° .

10.3 Umieszczanie w otworze (Linear)

Umieszczanie obiektu w otworze odbywa się dzięki poleceniu `Linear`. Polecenie to jest przydatne, gdy obiekty lub gniazda mają różne wysokości, przez co nie da się programowo zapewnić powtarzalności. Po naprowadzeniu na otwór robot rozpoczyna opuszczanie obiektu i kontynuuje ten ruch, dopóki nie wykryje siły reakcji podłoża. Wykrycie tej siły oznacza, że obiekt został już opuszczony do gniazda. Interfejs polecenia jest przedstawiony na rysunku 11.

Kierunek ruchu robota jest określany poprzez dwie listy rozwijane, z których pierwsza dotyczy środka układu współrzędnych (narzędzia lub podstawy robota), a druga kierunek ruchu w postaci oznaczenia osi wraz ze znakiem. Parametr ten jest podpisany jako `Direction (A)`. Domyślnie parametr ten jest ustalony jako ruch w kierunku `Z+` dla środka współrzędnych narzędzia `Tool`, a więc ruch pionowo w dół.



Rysunek 12: Interfejs polecenia Path dla narzędzia zamontowanego na robocie

Prędkość ruchu robota `Speed` (B) domyślnie wynosi 5mm/s , natomiast dozwolone są wartości z zakresu od 0.1mm/s do 150mm/s .

W przeciwieństwie do instrukcji `Spiral` i `Rotational`, polecenie `Linear` nie ma zdefiniowanej pozycji końcowej obiektu w otworze, ponieważ dno otworu jest wykrywane poprzez odczyt siły reakcji. Wartość tej siły uznawana za wykrytą powierzchnię definiuje parametr `Force threshold` z domyślną wartością 10N , i dozwolonymi wartościami z zakresu od 0.1N do 200N .

Maksymalny przebyty dystans `Maximum travel distance` (D) przyjmuje dozwolone wartości z zakresu od 0.5mm do 1500.0mm . Domyślna wartość to 100mm .

11 Ścieżka nagrywana (Path)

Polecenie `Path` jest jednym z trzech poleceń stworzonych w ramach zastąpienia poleceń `MoveJ`, `MoveL` i `MoveP`, które nie mogą być używane pod wpływem kontroli sił polecenia `Force Control`. To polecenie pozwala na zapisywanie na bieżąco ruchu robota odbywającego się w trybie `Freedrive mode` (tryb ręcznego sterowania). Tak nagrany ruch będzie następnie odtwarzany w programie.

Polecenie `Path` wymaga podania, czy narzędzie, które będzie używane, jest zamontowane na robocie, czy może jest niezależnym, zewnętrznym urządzeniem, do którego robot będzie przykładał obiekt.

Interfejs polecenia (patrz rys. 12) dostarcza przyciski do nagrywania ruchu, a także jego odtwarzania, cofania i stopowania. Przyciski oznaczone są tradycyjnymi ikonami. Ta część interfejsu jest taka sama zarówno w przypadku narzędzia zamontowanego na robocie, jak i dla zewnętrznego narzędzia.

Zaznaczając pole wyboru `Reverse Path`, ruch robota będzie odbywał się w kierunku przeciwnym, niż nagrany. Ta możliwość jest dostępna zarówno dla narzędzia zamontowanego

na robocie, jak i dla zewnętrznego narzędzia.

11.1 Parametry dla narzędzia zamontowanego na robocie

W tym przypadku zmiana prędkości ruchu robota odbywa się na dwa sposoby. Domyślny sposób to podanie mnożnika, przez który zostanie przeskalowana prędkość wynikająca z nagranego ruchu. Drugi sposób, stała prędkość, wymaga zaznaczenia pola wyboru `Constant speed`. Pole, do którego wprowadzi się mnożnik, zostanie zamienione na pole z prędkością podaną w milimetrach na sekundę.

Zaznaczając pole wyboru `Path relative to a feature` można zastosować nagraną trajektorię ruchu robota w odniesieniu do punktu wybranego z listy rozwijanej. Na uwagę zasługuje pozycja z listy o nazwie `Actual position`. Wybór tej opcji sprawi, że ruch zawsze odbędzie się względem bieżącej pozycji robota.

11.2 Parametry dla zewnętrznego narzędzia

Korzystając z pola tekstowego `Tool Speed` możliwa jest zmiana prędkości ruchu robota. W przeciwieństwie do prędkości z narzędziem zamontowanym na robocie, dla zewnętrznego narzędzia możliwe jest wprowadzanie prędkości robota tylko poprzez stałą wartość wyrażoną w milimetrach na sekundę.

Ponadto z listy rozwijanej `External Tool Feature` należy wybrać punkt, który jest punktem styku zewnętrznego narzędzia ze środkiem narzędzia robota TCP (Tool Center Point). Taki punkt należy zdefiniować w zakładce `Installation/Features`.

12 Ścieżka wielopunktowa (Multipoint Path)

Polecenie `Multipoint Path` jest jednym z trzech poleceń stworzonych w ramach zastąpienia poleceń `MoveJ`, `MoveL` i `MoveP`, które nie mogą być używane pod wpływem kontroli sił polecenia `Force Control`. To polecenie pozwala na zaprogramowanie ścieżki składającej się z wielu punktów. Polecenie `Multipoint Path` wymaga podania czy narzędzie, które będzie używane, jest zamontowane na robocie, czy jest niezależnym, zewnętrznym urządzeniem, do którego robot będzie przykładał obiekt.

Interfejs dostarcza przyciski do odtwarzania ruchu, a także jego cofania i wstrzymywania. Przyciski oznaczone są tradycyjnymi ikonami. W przeciwieństwie do polecenia `Path`, nagrywanie ścieżki nie jest dostępne. Ta część interfejsu jest taka sama zarówno w przypadku narzędzia zamontowanego na robocie, jak i dla zewnętrznego narzędzia.

Korzystając z pola tekstowego `Tool Speed` możliwa jest zmiana prędkości ruchu robota. W przeciwieństwie do prędkości z narzędziem zamontowanym na robocie, dla zewnętrznego narzędzia możliwe jest wprowadzanie prędkości robota tylko poprzez stałą wartość wyrażoną w milimetrach na sekundę.

Przy wykorzystywaniu narzędzia zamontowanego na robocie, zaznaczając pole wyboru `Path relative to a feature` można zastosować zaprogramowaną trajektorię ruchu robota w odniesieniu do punktu wybranego z listy rozwijanej. Na uwagę zasługuje pozycja z listy o nazwie `Actual position` (ruch zawsze odbędzie się względem aktualnej pozycji robota).

Przy wykorzystywaniu zewnętrznego narzędzia z listy rozwijanej `External Tool Feature` należy wybrać punkt, który jest punktem styku zewnętrznego narzędzia ze środkiem

narzędzia robota TCP (Tool Center Point). Taki punkt należy zdefiniować w zakładce *Installation/Features*.

Pierwszy punkt trajektorii jest automatycznie dodawany i różni się od pozostałych. Wymaga jedynie ustawienia punktu docelowego przyciskiem *Set waypoint*. Niestety żaden z interfejsów polecenia *Multipoint Path* nie daje możliwości podglądu ustalonego punktu (ani tego ani żadnego innego), możliwe jest natomiast przesunięcie efektora do tej pozycji przyciskiem *Move here*.

Dodawanie kolejnych punktów trasy odbywa się za pomocą przycisków *Add line* (ruch prostoliniowy) oraz *Add curve* (ruch po krzywej Béziera). Do programu zostanie dodany punkt, w którym aktualnie znajduje się robot. Rodzaj ruchu nie wpływa na nic innego poza trajektorią ruchu. W dowolnym momencie można swobodnie przełączać się między typami ruchów. Robot w swoim ruchu zawsze przejdzie przez każdy z zaprogramowanych punktów.

Przyciski, jakie występują w interfejsie punktów ścieżki:

- *Edit waypoint* – pozwala na zmianę punktu na nowy.
- *Delete* – usuwa punkt. Punkt początkowy nie zostanie usunięty, a oznaczony jako nieskonfigurowany.
- *Move here* – przesuwa robota do zaznaczonego punktu. Przycisk należy przytrzymać.
- *Add line* – dodaje do trasy punkt, w którym aktualnie znajduje się robot. Ruch do tego punktu odbędzie się w linii prostej.
- *Add curve* – dodaje do trasy punkt, w którym aktualnie znajduje się robot. Ruch do tego punktu odbędzie się po krzywej Béziera.

13 Generator ścieżki (Path Generator)

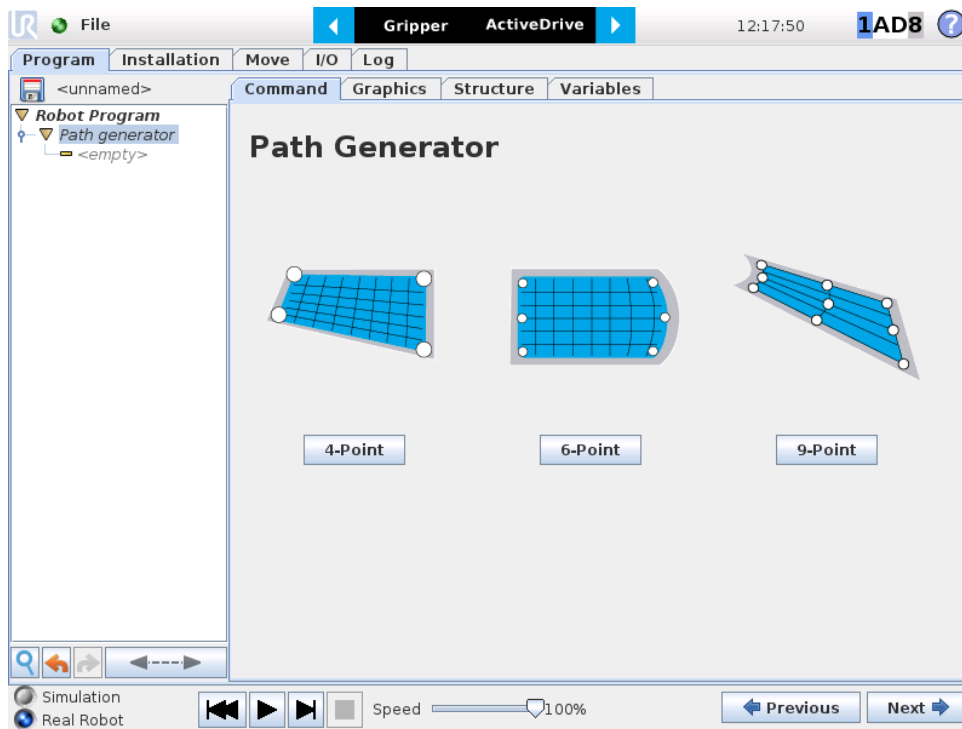
Polecenie *Path Generator* jest używane pod wpływem kontroli sił polecenia *Force Control*. Tworzy ono trasę na podstawie struktury składającej się z czterech, sześciu lub dziewięciu punktów. Interfejs wyboru struktury przedstawiony na rysunku 13 składa się z trzech przycisków, nad którymi wyświetlana jest animacja poklatkowa z propozycjami wykorzystania wspomnianych struktur do wygenerowania różnych kształtów. Odpowiednie ustawienie punktów ścieżki pozwala na uzyskanie trójwymiarowych kształtów. Warto zwrócić uwagę na możliwość ustawienia kilku punktów w tym samym miejscu, przez co można uzyskać dodatkowe kształty.

Wybór odpowiedniej struktury powinien zależeć od zapotrzebowania:

- struktura czteropunktowa – przestrzeń zawarta między dwiema liniami prostymi,
- struktura sześciopunktowa – przestrzeń zawarta między dwiema liniami krzywymi,
- struktura dziewięciopunktowa – przestrzeń zawarta między czterema liniami krzywymi.

Polecenie *Path Generator* wymaga ustalenia, czy narzędzie, które będzie używane, jest zamontowane na robocie, czy może jest niezależnym, zewnętrznym urządzeniem, do którego robot będzie przykładał obiekt.

Interfejs struktury czteropunktowej dla narzędzia zamontowanego na robocie jest dużo bardziej rozbudowany niż reszta możliwości, dlatego, mimo niektórych parametrów wspólnych, zostanie omówiony osobno i niezależnie.



Rysunek 13: Interfejs polecenia Path Generator. Animacja wyświetla propozycje wykorzystania struktur

W sytuacji, gdy narzędzie jest zamontowane na robocie, w interfejsie znajduje się parametr `Path relative to a feature`, umożliwiający wykonanie zaprogramowanej trajektorii w odniesieniu do punktu wybranego z listy rozwijanej. Jest on opcjonalny, dlatego obok znajduje się pole wyboru służące do włączania i wyłączania go. Jeżeli jednak robot korzysta z zewnętrznego narzędzia, w interfejsie znajduje się parametr `External tool feature`, dla którego również trzeba wybrać punkt z rozwijanej listy, ale musi to być punkt styku zewnętrznego narzędzia ze środkiem narzędzia robota TCP (Tool Center Point). Parametr ten jest obowiązkowy.

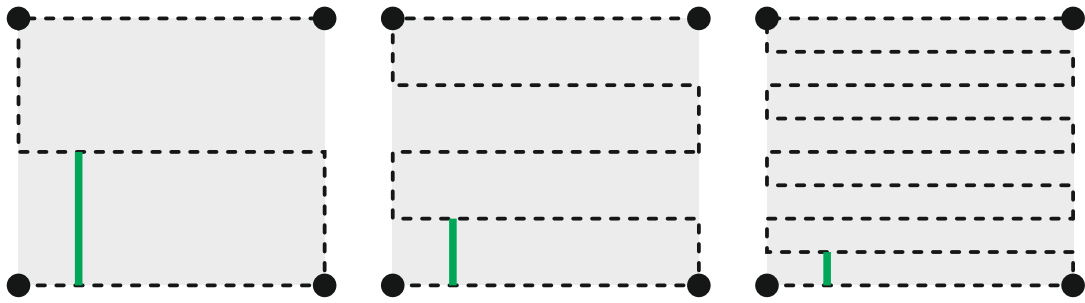
Prędkość ruchu robota `Tool speed` domyślnie wynosi 60mm/s , natomiast dozwolone są wartości z zakresu od 1mm/s do 250mm/s .

Ruch odbywający się jedynie od punktu do punktu byłby bardzo ograniczony i niewiele różniłby się od polecenia `Multipoint Path`. Interfejs dostarcza jednak możliwość wpływu na gęstość ruchów robota, a więc na odległości pomiędzy kolejnymi przejściami od jednej krzywej ograniczającej powierzchnię do drugiej. Służy do tego parametr `Spacing`, wyrażany w milimetrach. Zazwyczaj robot nie zastosuje dokładnie takiej wartości, jaką podał użytkownik, ale przybliżoną, która umożliwi zakończenie ruchu w punkcie wyznaczonym przez użytkownika. Przykładowe konfiguracje są przedstawione na rysunku 14.

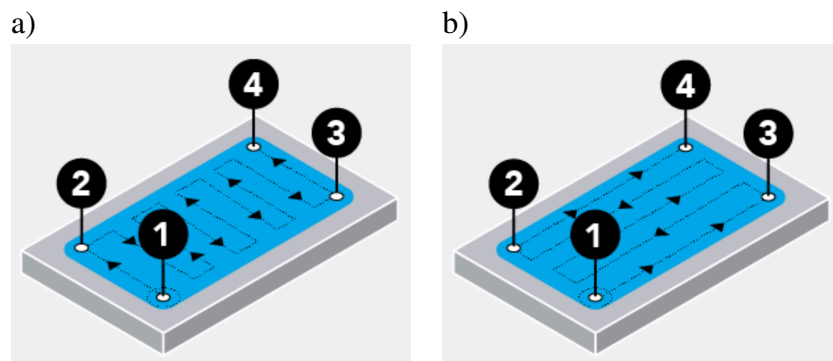
W strukturze czteropunktowej robot może poruszać się na dwa sposoby, przedstawione na rysunku 15.

1. `Vertical passes` – ruch przez środek płaszczyzny odbywa się pionowo. Ruch po obrzeżach płaszczyzny odbywa się poziomo.
2. `Horizontal passes` – ruch przez środek płaszczyzny odbywa się poziomo. Ruch po obrzeżach płaszczyzny odbywa się pionowo.

W strukturze sześciopunktowej robot może poruszać się na dwa sposoby, przedstawione na rysunku 16.

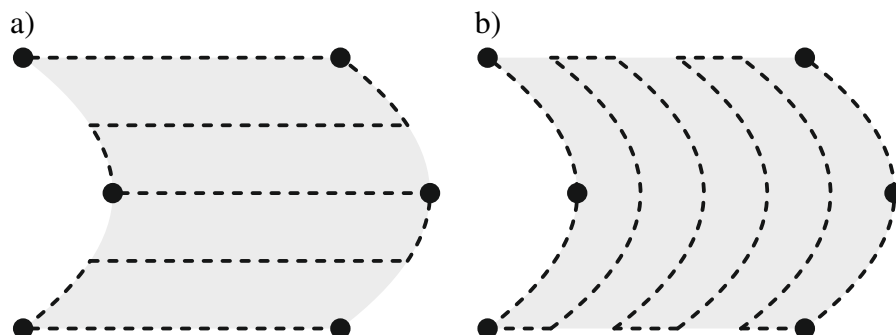


Rysunek 14: Trzy różne wartości parametru `Spacing` (zaznaczone zielonym kolorem) i ich spodziewany rezultat dla struktury czteropunktowej



Rysunek 15: Dwa kierunki ruchu robota na przykładzie struktury czteropunktowej: a) Vertical passes; b) Horizontal passes

1. `Linear passes` – ruch w linii prostej od jednej krzywej wyznaczającej powierzchnię do drugiej.
2. `Curved passes` – ruch zaokrąglony dostosowujący się do kształtu krzywych wyznaczających powierzchnię.



Rysunek 16: Kierunki ruchu w strukturze sześciopunktowej: a) Linear passes; b) Curved passes

Interfejs pozwala na wybór punktu końcowego trasy parametrem `End path on`. Wyboru dokonuje się poprzez przyciski opcji z numerami odpowiednich punktów obok. W zależności od struktury oraz kierunku ruchu mogą to być punkty o różnych numerach.

13.1 Parametry interfejsu Path Generator dla struktury czteropunktowej z narzędziem zamontowanym na robocie

Interfejs polecenia dla tej struktury jest podzielony na zakładki. Każda zakładka posiada grafikę pomocniczą, na której zaznaczone są niektóre z parametrów, oznaczone standardowo literami.

- `Path` – główne ustawienia trajektorii ruchu. W zależności od ustawień aktywuje lub deaktywuje zakładki `Surface` oraz `Perimeter`.
- `Surface` – ustawienia ruchu w środkowej części powierzchni.
- `Perimeter` – ustawienia ruchu wzdłuż obrzeży powierzchni.
- `Dimensions` – rozmiar powierzchni.
- `Margins` – marginesy powierzchni.

Zaznaczając pole wyboru `Path relative to a feature` w zakładce `Path` można zastosować nagrany trajektorię ruchu robota w odniesieniu to punktu wybranego z listy rozwijanej. Na uwagę zasługuje pozycja z listy o nazwie `Actual position` (ruch zawsze odbędzie się względem pozycji, w której robot będzie znajdował się po odbyciu wszystkich poprzednich instrukcji programu, czyli względem pozycji bieżącej).

Ruch robota może się odbywać (ustawienia w zakładce `Path`):

- `Surface only` – tylko po powierzchni.
- `Perimeter only` – tylko po obrzeżach powierzchni.
- `Perimeter and surface` – zarówno po powierzchni jak i jej obrzeżach.

Robot może poruszać się na dwa sposoby, przedstawione na rysunku 15 (ustawienia w zakładce `Path`):

1. `Vertical passes` – ruch przez środek płaszczyzny odbywa się pionowo. Ruch po obrzeżach płaszczyzny odbywa się poziomo.
2. `Horizontal passes` – ruch przez środek płaszczyzny odbywa się poziomo. Ruch po obrzeżach płaszczyzny odbywa się pionowo.

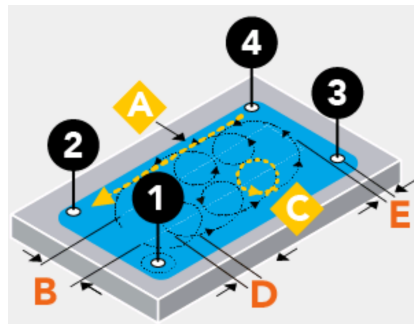
Prędkość ruchu `Feed A` z zakładki `Surface` domyślnie wynosi 60mm/s , natomiast dozwolone są wartości z zakresu od 1mm/s do 250mm/s .

Interfejs dostarcza możliwość wpływu na gęstość ruchów robota, a więc na odległości pomiędzy kolejnymi przejściami od jednego boku powierzchni do drugiego (patrz rys. 14). Służy do tego parametr `Spacing ~ B` w zakładce `Surface`. Zazwyczaj robot nie stosuje dokładnie takiej wartości, jaką podał użytkownik, ale przybliżoną, która umożliwi zakończenie ruchu w punkcie wyznaczonym przez użytkownika. Dozwolone są wartości z zakresu od 1mm do 800mm .

Interfejs pozwala na wybór punktu końcowego trasy parametrem `End path on` z zakładki `Surface`. Wyboru dokonuje się poprzez przyciski opcji. Możliwe jest wybranie punktu trzeciego lub czwartego, ale można również wybrać automatyczne dopasowanie `Spacing best fit`. Korzystając z tej opcji, punkt końcowy zostanie dopasowany w taki sposób, aby robot zastosował gęstość ruchu możliwie najbliższą do wartości parametru `spacing`. W tym przypadku ruch robota skończy się w punkcie drugim lub czwartym.

Standardowo ruch robota odbywa się po liniach prostych, jednak poprzez zaznaczenie pola wyboru *Add circular weaving* z zakładki *Surface*, można zamienić ten ruch na zaokrąglony na kształt pętel, przedstawiony na rysunku 17. Po wybraniu ruchu zaokrąglonego można go skonfigurować trzema parametrami.

- *Weaving radius* $\sim C$ – prędkość, z jaką zostanie wykonany każdy obrót. Wartości są z zakresu od 45° do 1080° .
- *Weaving radius* D – promień każdego obrotu. Wartości są z zakresu od $1mm$ do $100mm$.
- *Weaving margin* E – minimalna odległość od krawędzi wyznaczonej powierzchni, jaka ma być zachowana w trakcie wykonywania ruchu. Wartości są z zakresu od $0mm$ do $200mm$.



Rysunek 17: Ruch zaokrąglony

Prędkość ruchu robota na obrzeżach powierzchni *Perimeter speed* z zakładki *Perimeter* domyślnie przyjmuje wartość $30mm/s$, natomiast dozwolone są wartości z zakresu od $1mm/s$ do $400mm/s$.

Możliwe jest też określenie, jak wiele razy ma się odbyć ruch po obrzeżach powierzchni. Służy do tego parametr *Number of passes* z zakładki *Perimeter*. Minimalna liczba to 1, a maksymalna 100 okrążeń.

Domyślnie wymiary powierzchni określa się na podstawie czterech punktów ustawianych przez użytkownika. Odpowiadają za to parametry *Width* (odległość między punktami 1 i 2) oraz *Length* (odległość między punktami 1 i 3) z zakładki *Dimensions*, które domyślnie są ustawione na opcje *Fixed*. Oba parametry można ustawić niezależnie od siebie na opcje *Dynamic*, która pozwala na ustalenie rozmiarów na podstawie odległości wyrażonych w milimetrach *Taught part* W_i punktów orientacyjnych wybieranych z listy rozwijanej *Current part* A . Punkty można dodawać w zakładce *Installation/Features*.

W zakładce *Margins* znajdują się ustawienia marginesów powierzchni. Marginesy to odległości faktycznej powierzchni użytkowej od prostych łączących punkty wyznaczone przez użytkownika. Każdy bok posiada osobne ustawienia marginesu, podpisane i zaznaczone na grafice pomocniczej. Domyślna wartość każdego marginesu to $0mm$, maksymalna wartość to $1000mm$.

Po wyborze struktury odpowiednia liczba punktów zostanie automatycznie dodana do programu. Przechodząc do każdego z nich należy ustawić robota w miejscu, gdzie punkt ma się znajdować i wcisnąć przycisk *Set waypoint*. Można przesunąć robota do już ustawionych punktów przytrzymując przycisk *Move here*.

14 Wykonanie programu względem obiektu o wykrytym położeniu i orientacji (Find Contact Offset oraz Apply Contact Offset)

Planując pracę manipulatora i pisząc program, nie zawsze możliwe jest zagwarantowanie powtarzalności pozycjonowania obiektów, przy których robot będzie pracował. W takich sytuacjach pomocne jest zautomatyzowane określanie położenia bądź orientacji przedmiotów przez robota, a następnie wykonanie dalszej części programu w nowo obliczonych współrzędnych. Oprogramowanie czujnika dostarcza rozwiązania w postaci zestawu dwóch poleceń.

- `Find Contact Offset` – ustalenie pozycji (odległości) bądź orientacji obiektu. Kombinacje tych dwóch możliwości są dozwolone.
- `Apply Contact Offset` – wykorzystanie nowo ustalonych współrzędnych w wydzielonej części programu.

Polecenie `Find Contact Offset` wykrywa położenie względem punktu znajdującego się w zakładce `Installation/Features`. Punkt ten może być ulokowany w dowolnym miejscu przestrzeni roboczej. Wszystkie polecenia zawarte w węźle `Apply Contact Offset` muszą być wykonywane względem tego samego punktu. Jeżeli któreś polecenie nie posiada takiej możliwości, prawdopodobnie nie będzie działać poprawnie.

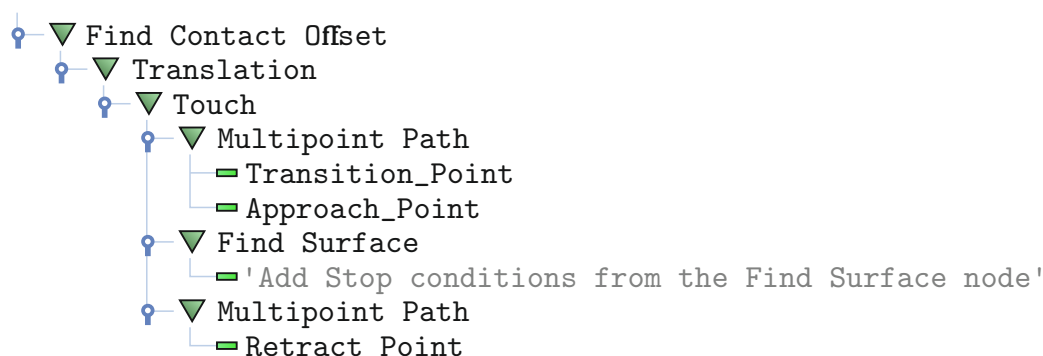
Sam proces konfigurowania polecenia `Find Contact Offset` jest długotrwały, a w przypadku potrzeby wniesienia poprawek, część procesu należy powtórzyć. Z tego względu zalecane jest zaprojektowanie całego polecenia przed przystąpieniem do programowania. Warto zwrócić uwagę na samo umiejscowienie polecenia w programie. Polecenie może znajdować się w głównej pętli programu, gdzie częstość wykonywania będzie zależeć od programu, lub może znajdować się w sekcji `BeforeStart`, gdzie wykona się tylko jeden raz przy starcie programu. Na podstawie kontaktu manipulatora z obiektem i odczytów siły z sensora, robot oblicza położenie obiektu względem wybranego punktu z przestrzeni roboczej.

W celu zaprogramowania wykrywania odległości obiektu należy w interfejsie polecenia `Find Contact Offset` wybrać `Add translation` (jeżeli przycisk jest niedostępny, należy najpierw wybrać punkt `Feature to Offset`).

Do drzewa programu zostanie dodany węzeł `Translation` o strukturze przedstawionej na rysunku 18. Pierwszy węzeł `Multipoint Path` zawiera dwa punkty służące do zaprogramowania bezpiecznego podejścia do pozycji startowej kolejnego polecenia. Źle zaprogramowany ruch może spowodować przesunięcie obiektu, a więc zaburzyć całą sekwencję. Do wykrycia punktu kontaktu służy standardowe polecenie wykrywania powierzchni. Ze wszystkich parametrów najważniejszy jest kierunek `Direction`. Do ustalenia poprawnego kierunku pomocne mogą być oznakowania na chwytaku oraz przycisk `Hold to test`. Drugi węzeł ścieżki wielopunktowej ma na celu bezpieczne wycofanie efektora. Ponownie, źle zaprogramowany ruch może spowodować przesunięcie obiektu, a więc zaburzyć całą sekwencję.

Wykrywanie orientacji obiektu odbywa się poprzez dwukrotny kontakt efektor z obiektem, na którego podstawie ustalane są dwa punkty, które z kolei służą do obliczenia orientacji. Polecenie nadaje się więc do obiektów, w których występują powierzchnie płaskie. Nieregularne obiekty, takie jak np. rzeźby lub figurki, nie mają powierzchni, które umożliwiają wykrycie orientacji. Takie powierzchnie nie muszą być jednak zintegrowane z obiektami.

W celu zaprogramowania wykrywania orientacji obiektu należy w interfejsie polecenia `Find Contact Offset` wybrać `Add orientation` (jeżeli przycisk jest niedostępny,



Rysunek 18: Struktura węzła Find Contact Offset

należy najpierw wybrać punkt Feature to Offset). Do drzewa programu zostanie dodany węzeł Orientation z dwoma węzłami Touch. Oba konfiguruje się dokładnie w ten sam sposób, jak dla wyszukiwania odległości Add translation:

- Multipoint Path – dwa punktu służące do zaprogramowania bezpiecznego podejścia do pozycji startowej kolejnego polecenia.
- Find Surface – standardowe polecenie wykrywania powierzchni. Ze wszystkich parametrów najważniejszy jest kierunek Direction. Do ustalenia poprawnego kierunku pomocne mogą być oznakowania na chwytaku oraz przycisk Hold to test.
- Multipoint Path – jeden punkt służący do zaprogramowania bezpiecznego wycofania efektora.

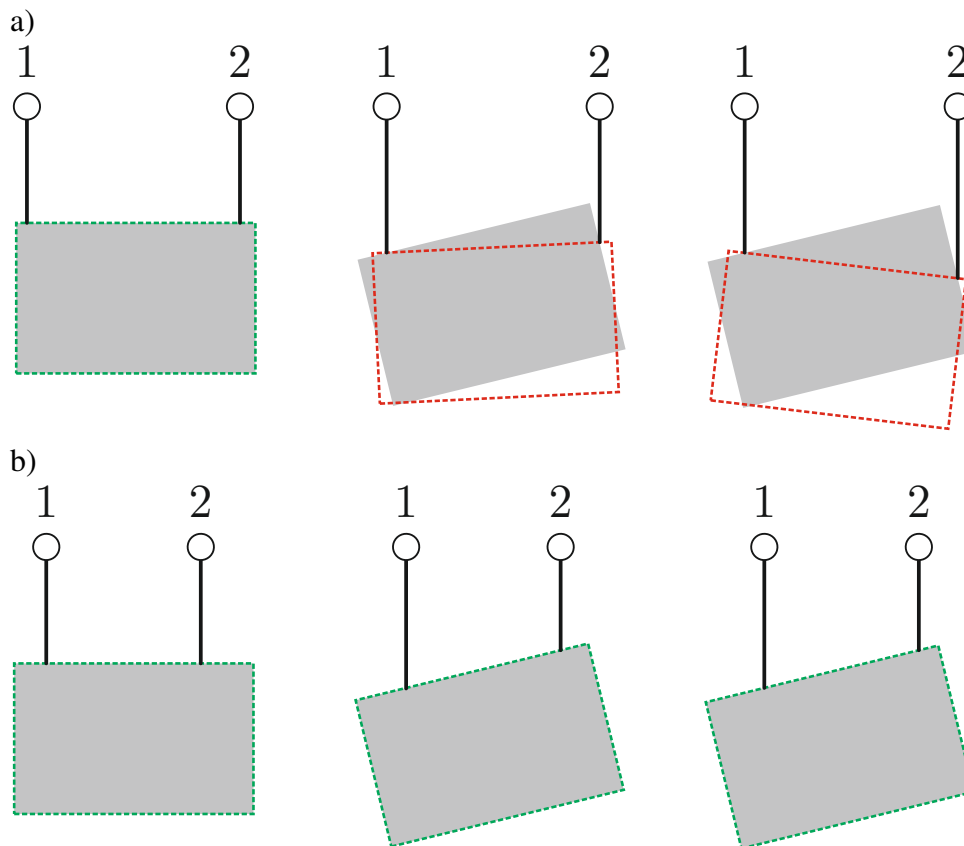
Producent zaleca jak największy możliwy odstęp między dwoma punktami kontaktu w celu jak najdokładniejszego obliczenia orientacji. Należy jednak zachować ostrożność przy wyznaczaniu tych punktów i zastanowić się nad sytuacją przedstawioną na rysunku 19a. Punkty w zbyt dużej odległości od siebie będą wymagać dokładniejszego pozycjonowania obiektów. Warto rozważyć zaprogramowanie punktów podejścia bliżej siebie, jak na rysunku 19b.

Po skonfigurowaniu wszystkich węzłów polecenia Find Contact Offset należy powrócić do interfejsu tego polecenia i wcisnąć przycisk Teach the contact offset reference. Zostanie wyświetlone ostrzeżenie, że robot poruszy się celem wykonania sekwencji. Po zakończonym ruchu zostanie wyświetlone zapytanie o akceptację wykonanej sekwencji i zapisanie jej jako pozycji referencyjnej. Jeżeli wszystko odbyło się prawidłowo, należy zaakceptować sekwencję. W interfejsie powinna pojawić się informacja, że punkt referencyjny odpowiada punktowi obliczonemu podczas wykonywania sekwencji.

Wykonanie tej sekwencji nie oznacza, że wykryte położenie i orientacja obiektu zostały już ustalone na stałe. Przeciwnie, cała sekwencja pomiarów będzie odbywała się za każdym razem, gdy polecenie zostanie wywołane w programie, nadpisując poprzedni offset. Wykonanie sekwencji w tym momencie ma na celu wstępne nauczenie robota przestrzeni roboczej podlegającej offsetowi. Z tego powodu po wykonaniu tej sekwencji należy przejść do polecenia Apply Contact Offset.

14.1 Wykorzystanie wykrytego offsetu

Po skonfigurowaniu polecenia Find Contact Offset i nauczeniu robota sekwencji można przystąpić do programowania działań podlegających wykrytemu offsetowi. W interfejsie polecenia Apply Contact Offset oraz w każdym poleceniu wewnątrz tego węzła należy



Rysunek 19: Dwa rozstawy punktów kontaktu i ich efekty, Położenie obiektu powinno być dodatkowo wykrywane poprzez wykrycie translacji (odległości), ale w ramach uproszczenia rysunku nie zostało to zaznaczone: a) Prawie maksymalny rozstaw. Kolorem czerwonym zaznaczono błędnie wykryte położenie obiektu; b) Mniejszy, bezpieczny rozstaw

wybrać z listy rozwijanej ten sam punkt z zakładki `Installation/Features`, który został wykorzystany w poleceniu `Find Contact Offset`. W przeciwnym razie lub w przypadku wyboru polecenia bez możliwości wykonania go względem punktu, polecenie nie zostanie poddane offsetowi.

W momencie wybrania odpowiedniego punktu referencyjnego w interfejsie powinna wyświetlić się informacja, że wybrany punkt odpowiada punktowi obliczonemu w poprzednio wykonanej sekwencji `Find Contact Offset`. Program, który ma się odbyć względem wykrytego offsetu należy umieścić wewnątrz węzła `Apply Contact Offset` pamiętając o zaznaczeniu odpowiedniego punktu referencyjnego dla każdego polecenia.

Może jednak wystąpić sytuacja, gdy komunikat mówi o tym, że węzeł jest niedostępny, a punkt referencyjny nie odpowiada obliczonemu punktowi w poprzednio wykonanej sekwencji. Nie powinno się tak stać bezpośrednio po skonfigurowaniu sekwencji `Find Contact Offset`, ale na pewno taki komunikat pojawi się po wykonaniu innych akcji przez robota.

Po zaprogramowaniu obu poleceń `Find Contact Offset` i `Apply Contact Offset`, oraz po dalszym programowaniu robota, może pojawić się potrzeba poprawienia programu odbywającego się pod wpływem offsetu lub zmiany samej sekwencji wykrywania położenia i orientacji obiektu. Jest to dosyć uciążliwa sytuacja, ponieważ oba polecenia prawdopodobnie zostały zablokowane i nie można do nich wносить poprawek. Interfejs polecenia `Apply Contact Offset` poinformuje użytkownika o niezgodności punktu referencyjnego z ostatnio obliczonym punktem w sekwencji `Find Contact Offset`. Wyświetlona będzie również informacja, że aby naprawić błąd, należy przejść do interfejsu `Find Contact Offset`



Rysunek 20: Zakładka ActiveDrive

i zastosować się do trzykrokowej instrukcji odblokowania węzłów

1. Execute the Touch sequence – po kliknięciu przycisku Execute sequence robot ponownie wykona sekwencję polecenia Find Contact Offset.
2. Move the robot to the calculated feature position – przytrzymanie przycisku Hold to move robot sprawi, że robot poruszy się do punktu obliczonego podczas wykonywania sekwencji.
3. Go to the installation tab > Features > your_feature > Change/Set point > OK – zadaniem użytkownika jest udanie się do zakładki Installation/Features, wybranie używanego punktu referencyjnego (w górnej części interfejsu Find Contact Offset znajduje się nazwa tego punktu) i przypisanie mu aktualnej pozycji robota, a więc tej obliczonej podczas wykonywania sekwencji.

Po wykonaniu tych trzech kroków oba węzły poleceń Find Contact Offset oraz Apply Contact Offset powinny zostać odblokowane. Drugą możliwością odblokowania węzłów jest zaprogramowanie sekwencji pomiarów od nowa korzystając z przycisku Modify the Touch sequence program.

15 Zakładka ActiveDrive

Jeżeli czujnik wykrywa przykładaną siłę, w momencie otwarcia zakładki ActiveDrive (patrz rys. 20) zostanie wyświetlony komunikat z propozycją wyzerowania sensora. Po upewnieniu się, że żadna siła nie działa na czujnik, należy go wyzerować.

Tryb ręcznego sterowania z wykorzystaniem czujnika siły włącza się przyciskiem On/Off. Ikona zmieni się, informując o aktywowanym trybie. Tryb automatycznie wyłącza się po włączeniu programu robota, po dokonaniu zmiany w programie, po poruszeniu robotem w dowolny inny sposób oraz po trzydziestu sekundach bezczynności.

Standardowy tryb ręcznego sterowania Freedrive mode pozwala na poruszanie robotem poprzez przykładanie siły do ramion robota. W trybie ręcznego sterowania z wykorzystaniem czujnika siły należy przykładać właśnie do czujnika. Taki ruch może być dodatkowo definiowany poprzez opcje Mode oraz Contact z zakładki ActiveDrive.

Zazwyczaj, w momencie włączenia omawianego trybu sterowania, jeżeli czujnik wykrywa przykładaną siłę, zostanie wyświetlony stosowny komunikat z zapytaniem o wyzerowanie sensora. Może się jednak zdarzyć, że taki komunikat się nie pojawi, a po włączeniu trybu sterowania ręcznego z wykorzystaniem czujnika siły, robot sam zacznie się poruszać. W obu przypadkach należy wyzerować czujnik, oczywiście w sytuacji, w której do czujnika nie jest przykładana żadna siła.

Przycisk Mode otwiera zakładkę, w której znajdują się sposoby ruchu w trybie ręcznego sterowania z wykorzystaniem czujnika siły:

- `Free` – ruch translacyjny we wszystkich osiach, obroty wokół wszystkich osi.
- `Scara` – ruch translacyjny we wszystkich osiach, obroty tylko wokół jednej (pionowej) osi.
- `Translation` – ruch translacyjny we wszystkich osiach, żadnych obrotów wokół osi
- `Plane` – ruch translacyjny tylko w dwóch poziomych osiach, obroty tylko w trzeciej, pionowej, osi (ruch w płaszczyźnie).
- `Dominant` – ruch translacyjny we wszystkich osiach, ale w danym momencie tylko w tej osi, do której przykładana jest największa siła, żadnych obrotów wokół osi.

Przycisk `Zero` służy do zerowania czujnika. Korzystając z tego przycisku, można również pozbyć się ewentualnych drgań robota wynikających z odczytów sensora.

Przycisk `Contact` służy do przełączania pomiędzy szybkim (ikona królika) a wolnym (ikona ślimaka) ruchem w trybie ręcznego sterowania z wykorzystaniem czujnika siły. Wolna prędkość ma na celu zwiększenie precyzji sterowania. Czujnik automatycznie przełącza się w tryb wolnej prędkości po wykryciu dużych sił.

Przytrzymanie przycisku `Reorient` sprawi, że robot poruszy się do najbliższej ortogonalnej pozycji względem głównego układu współrzędnych `Base`. W większości przypadków oznacza to, że ustawi się prostopadle do podłoża, co może być istotne do poprawnego odczytu sił przy wykonywaniu niektórych poleceń.

16 Opis stanowiska

Na potrzeby realizacji następujących zadań

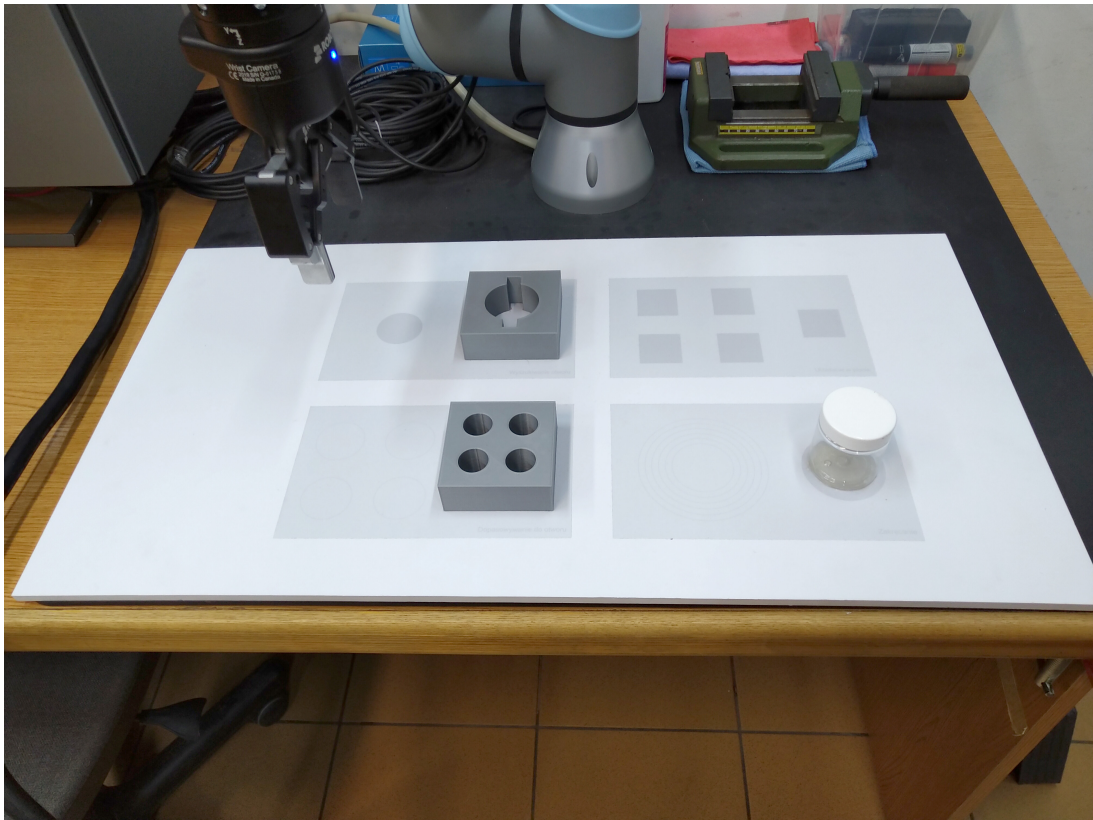
- układanie obiektów w pionie,
- spiralne wyszukiwanie otworu,
- rotacyjne dopasowywanie do otworu,
- zakręcanie i odkręcanie nakrętki,

została stworzona plansza (rys. 21) z przygotowanym miejscem na każde zadanie. Na planszy są oznaczane miejsca poboru obiektów, mające na celu ułatwienie pracy z robotem. Znajdują się tam również dwa gniazda z otworami na klocki wkładane przez robota oraz pojemnik, do którego robot może przykręcać i odkręcać nakrętkę.

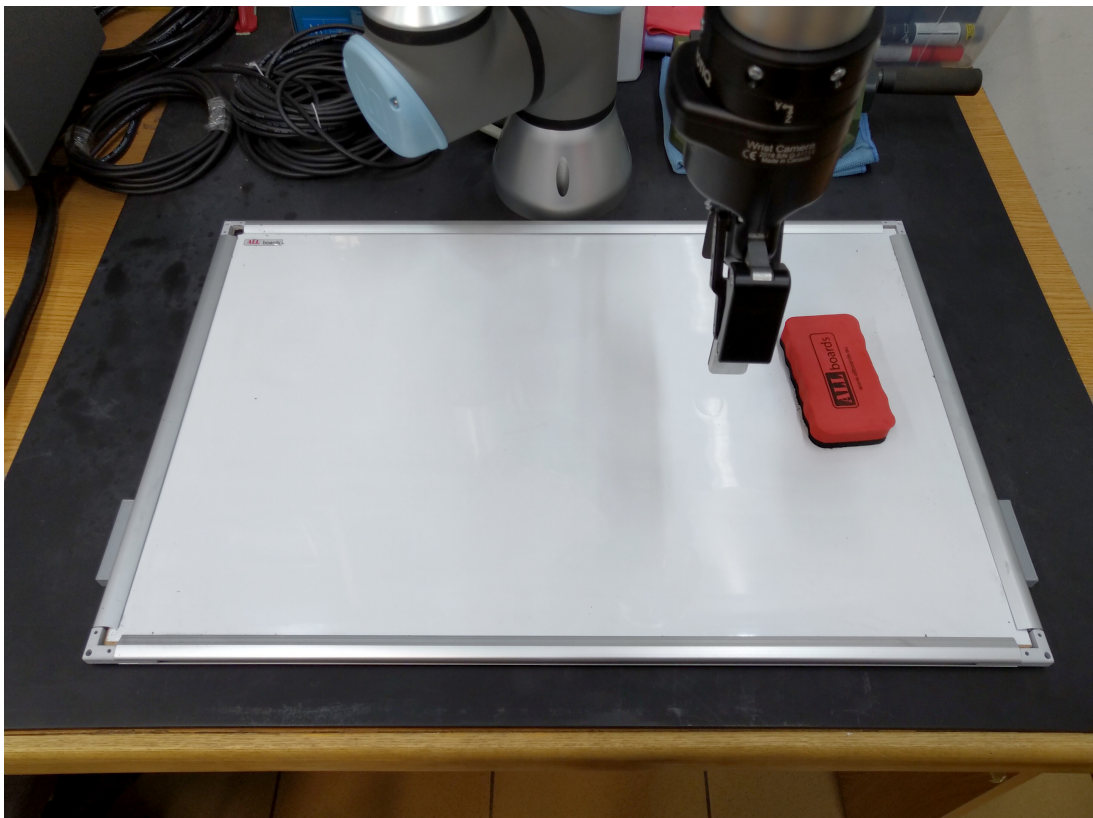
Na potrzeby realizacji zadania czyszczenia tablicy suchościernalnej (polerowania powierzchni) została przygotowana tablica ze specjalnymi uchwytami do zamontowania na płycie robota (zdjęcie 22).

17 Zadania

Poniższe zadania zostały zaprojektowane w taki sposób, aby zawrzeć jak najwięcej informacji na temat funkcjonalności czujnika oraz pomóc zrozumieć jego działanie i wykorzystanie zjawiska siły. Przed każdym zadaniem warto się upewnić, że chwytak jest ustawiony prostopadle do podłoża. Można skorzystać z opcji `Reorient` w zakładce `ActiveDrive`.



Rysunek 21: Stanowisko z planszą do ćwiczeń

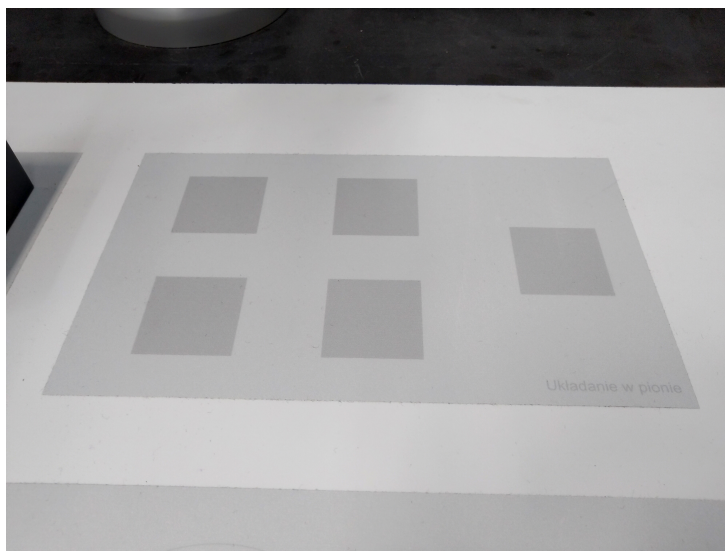


Rysunek 22: Stanowisko z tablicą suchościeralną

17.1 Układanie klocków w pionie

Ćwiczenie polega na pobieraniu klocków przez robota i układaniu ich jeden na drugim. Klocki są prostopadłościenne, mogą być różnych wymiarów (w granicach możliwości chwytaka), w szczególności klocki mogą być o różnych wysokościach (polecenie `Find Surface` sprawdzi wysokość klocka - chwytak porusza się w kierunku klocka dopóki czujnik nie wykryje zadanej siły (warto przetestować różne wartości siły i ich wpływ na działanie programu) wynikające z kolizji z klockiem; maksymalny możliwy dystans do przebycia powinien być tak zaprogramowany, aby robot nie pomylił podłoża z klockiem w przypadku jego braku). To założenie czyni zadanie bardzo problematycznym dla zwykłego robota bez czujników, natomiast jest trywialne dla robota wyposażonego w czujnik siły. Kolejnym założeniem jest pobieranie klocków z czterech miejsc, wykorzystując polecenie paletyzacji `Pallet`. **Uwaga:** Ze względu na niewielkie odległości pomiędzy klockami na planszy, chwytak nie może zostać w pełni otwarty. Wartość rozstawu szczęk chwytaka 40% wystarczy, aby chwycić klocek.

Strefa planszy przeznaczona do realizacji zadania jest przedstawiona na zdjęciu 23.



Rysunek 23: Strefa zadania układania klocków w pionie

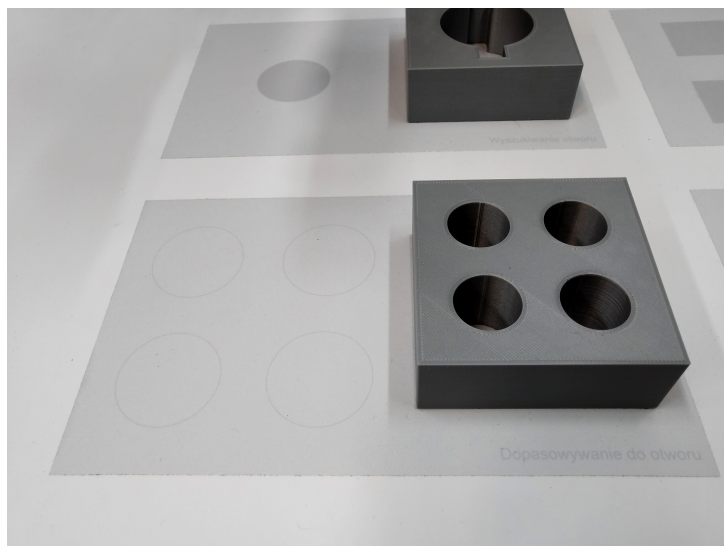
Program można rozwinąć o sprawdzanie obecności klocka wykorzystując maksymalny dystans przebyty w polecenie `Find Surface` i podejmując odpowiednią akcję. Ponadto, warto wykorzystać polecenie `Grip check` do sprawdzenia, czy klocek został poprawnie schwytyany.

17.2 Spiralne wyszukiwanie otworu

Ćwiczenie polega na pobieraniu cylindrycznych klocków i wkładaniu ich do poszczególnych otworów. Klocki w miejscu ich poboru nie są dokładnie ułożone, przez co chwytak zawsze złapie każdy klocek w innym miejscu. Tak złapane klocki muszą zostać dopasowane do otworów, do których odnalezienia należy wykorzystać polecenie `Spiral` z węzła `Insertion`.

Strefa planszy przeznaczona do realizacji zadania jest przedstawiona na zdjęciu 24.

Wskazówki: W celu pobierania i odkładania klocków należy wykorzystać polecenie paletyzacji `Pallet`. Warto wykorzystać komunikat informujący o zakończeniu programu. Z poziomu okna komunikatu można program zakończyć lub kontynuować, co może pozwolić na bezpieczne dołożenie nowych klocków i zabranie tych ułożonych, bez konieczności kończenia programu. Przy programowaniu punktów podejścia do otworów, należy zastosować polecenie



Rysunek 24: Strefa zadania wyszukiwania otworu

Insertion (Warto sprawdzić niezawodność działania w zależności od ustawionych parametrów, w szczególności inkrementacji promienia ruchu na jeden obrót `Radius increment per turn` (E) oraz siły inicjalizującej ruch spiralny `Force initiating spiral move` (D)). Tworząc program, należy dobrać punkty trajektorii w taki sposób, aby uniknąć wszelkich kolizji klocków w czasie transportowania. Można to zrobić poprzez dobranie odpowiednich punktów w przestrzeni lub zastosowanie polecenia wykrywania kolizji `Collision Detection`. Przydatne może być wykorzystanie ruchu typu `Relative Position` polecenia `MoveL` do przemieszczania chwytaka o zadaną odległość. Przy programowaniu pozycji docelowej wewnątrz otworów pomocne mogą być funkcje z zakładki `ActiveDrive`.

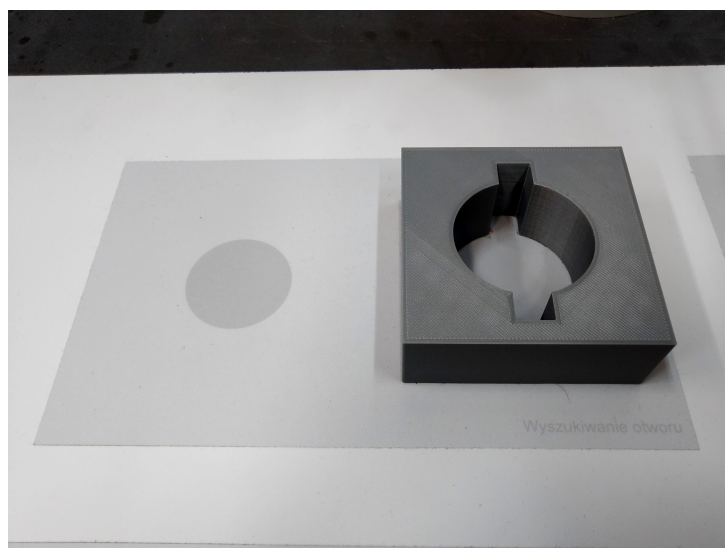
17.3 Rotacyjne dopasowywanie obiektu do otworu

Ćwiczenie polega na pobraniu cylindrycznego klocka z dwiema wypustkami i włożeniu go do specjalnego otworu. Kłoczek posiada dwie identyczne wypustki po przeciwległych stronach, więc musi zostać dopasowany do otworu. Strefa planszy przeznaczona do realizacji zadania jest przedstawiona na zdjęciu 25.

Przy programowaniu pozycji docelowej wewnątrz otworu pomocne mogą być funkcje z zakładki `ActiveDrive`. Warto sprawdzić niezawodność działania w zależności od ustawionych parametrów polecenia `Rotational` węzła `Insertion`, w szczególności prędkości obrotów `Rotation speed` oraz siły inicjalizującej ruch obrotowy `Force initiating rotation` (D).

17.4 Polerowanie/czyszczenie powierzchni

Zadanie polerowania powierzchni w warunkach laboratoryjnych można zastąpić zadaniem ściernia tablicy suchościeralnej za pomocą gąbki umieszczonej w chwytaku. Przez cały czas robot powinien wywierać stały nacisk na tablicę. Parametry należy dobrać w taki sposób, aby zapewnić gwarancję całkowitego wyczyszczenia tablicy bez jej uszkodzenia oraz w rozsądnym czasie. Zadanie idealnie obrazuje ideę wykorzystania kontroli sił `Force Command` i dowolnej ścieżki `Path`, `Multipoint Path` lub `Path Generator`. Dodatkowo, charakteryzuje się dużą liczbą parametrów poleceń, dzięki czemu problemy jakie wystąpią, mogą zostać rozwiązane



Rysunek 25: Strefa zadania dopasowywania obiektu do otworu

na wiele sposobów.

Stanowisko do realizacji zadania jest przedstawione na zdjęciu 22. Do czyszczenia wykorzystywana jest standardowa gąbka do tablic suchościeralnych. W związku z powyższym pojawia się potrzeba schwywania gąbki chwytakiem. Takie rozwiązanie charakteryzuje się jednak pewną komplikacją pod postacią niewystarczającej siły tarcia występującej pomiędzy gąbką a chwytakiem. W efekcie, przy dużym nacisku lub długim czasie czyszczenia tablicy, chwytak coraz bardziej przesuwa się po gąbce w dół. Całkowite zbliżenie się chwytaka do tablicy w ten sposób oznacza porysowanie powierzchni. Rozwiązanie zaproponowane w programie to chwytanie gąbki przed rozpoczęciem wycierania tablicy i odkładanie jej po ukończeniu. W ten sposób za każdym razem robot rozpoczyna operację, trzymając gąbkę na odpowiedniej wysokości, a czas jednego czyszczenia jest zbyt krótki, aby chwytak zbliżył się do powierzchni. Innym rozwiązaniem może być wykorzystanie parametru `Deviation range` polecenia `Force Control` do ograniczenia odległości chwytaka od tablicy względem pozycji początkowej. Dodatkowo poleceniem `Grip Check` należy sprawdzić, czy gąbka została poprawnie schwyta. Próba czyszczenia tablicy bez gąbki również skończyłaby się porysowaniem tablicy.

Warto wykorzystać polecenie generatora ścieżki `Path Generator` o strukturze cztero-punktowej, której punkty powinny wyznaczać obszar czyszczenia. Warto programować punkty trzymając chwytakiem gąbkę. Szczególnie gdy punkty znajdują się w narożnikach tablicy, gąbka pomaga zachować odpowiednie odległości od ramy tablicy. Zalecane jest również umieszczenie punktów w takich miejscach, aby robot dotykał gąbką tablicy. W przeciwnym razie może (w zależności od innych parametrów) wystąpić sytuacja, gdy robot docierając do punktów oddalonych od tablicy, nie doczyści jej w tych miejscach. W celu zapewnienia poprawnego przyłożenia siły, chwytak powinien być skierowany prostopadle do tablicy.

Samą ścieżkę ruchu robota dobrze jest zaprogramować, testując różne ustawienia parametrów w praktyce. Warto rozpocząć od czyszczenia samej powierzchni bez obrzeży i przetestowania ruchu zaokrąglonego `Add circular weaving`, a następnie doboru prędkości ruchu i wartości parametru `Spacing`.

Aby zapewnić czyszczenie tablicy ze stałym naciskiem, polecenie ścieżki powinno znajdować się w węźle `Force Control`. Przykładana siła nie musi być duża, jej dobór powinien odbywać się w tym samym czasie co dobór parametrów trajektorii ruchu robota. Podanie zbyt dużej wartości siły może przyspieszyć opadanie chwytaka wynikające z wartości tarcia gąbki

i chwytaka. Należy pamiętać o wyzerowaniu czujnika przed kontrolą sił. Czujnik nie powinien być zerowany w momencie, gdy chwytak jest w bezpośrednim kontakcie z powierzchnią tablicy.

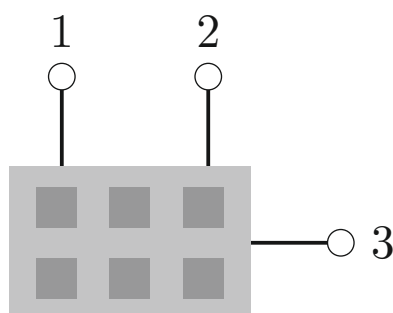
Można przyjąć następujące założenia: jeżeli w momencie rozpoczęcia kontroli sił chwytak znajduje się już przy tablicy i dotyka gąbką powierzchni, można wybrać `Current position`. W przeciwnym razie, jeżeli w momencie rozpoczęcia kontroli sił chwytak znajduje się w innej pozycji niż początek ścieżki, ponieważ np. punkt `Approach` jest ustawiony znacznie ponad ścieżką, należy użyć `Targeted position`. Takie założenia powinny stanowić jedynie punkt wyjściowy, a finalny wybór powinien zostać przetestowany w praktyce.

17.5 Pobieranie klocków z palety znajdującej się w różnych położeniach

Ćwiczenie polega na pobraniu klocków z palety. Zakłada się, że docelowa pozycja palety jest jedynie przybliżona (nie zawsze taka sama), a zatem nie ma pewności, że robot zawsze poprawnie pobierze klocki. Pozycja palety oraz jej orientacja powinny zostać ustalone programowo po każdym dołożeniu nowej palety. Do wykrywania orientacji i położenia palety należy użyć poleceń `Find Contact Offset` oraz `Apply Contact Offset`. Zadanie wymaga wykrycia dokładnej lokalizacji i orientacji palety, a więc pozycji w dwóch osiach płaszczyzny oraz kąta obrotu w trzeciej, pionowej osi. Struktura węzła `Find Contact Offset` do określenia pozycji palety powinna być następująca.

1. `Orientation` – wykrycie orientacji to złożenie dwóch węzłów `Touch`. Nieformalnie można więc powiedzieć, że jest to zastosowanie dwa razy instrukcji `Translation`, która wykrywa odległość obiektu, rozbudowane o obliczenie orientacji na podstawie dwóch punktów kontaktu. W rezultacie, zastosowanie wykrywania orientacji pozwala uzyskać orientację palety oraz jej przesunięcie w jeden osi płaszczyzny.
2. `Translation` – wykrycie odległości w drugiej osi płaszczyzny, prostopadłej do pierwszej.

Graficzna reprezentacja rozmieszczenia punktów podejścia węzłów `Orientation` oraz `Translation` jest przedstawiona na rysunku 26.



Rysunek 26: Konfiguracja punktów węzła `Find Contact Offset`. Punkty 1 i 2 należą do węzła `Orientation`, natomiast punkt 3 należy do węzła `Translation`

Po odbyciu wstępnego nauczania sekwencji w węźle `Find Contact Offset` należy przystąpić do zaprogramowania instrukcji pobierania klocków z palety w węźle polecenia `Apply Contact Offset`. Niestety polecenie `Pallet` nie posiada możliwości ustawienia punktu względem którego ma się odbyć paletyzacja, więc nie może zostać wykorzystana do odbierania klocków. Ważne, by zaimplementowane w ramach polecenia `Apply Contact`

Offset instrukcje ruchu (Move) dotyczące poboru klocka odbywały się na podstawie ruchu *Relative Position* (nie *Fixed Position*), który z definicji jest ruchem względem aktualnej pozycji chwytaka.

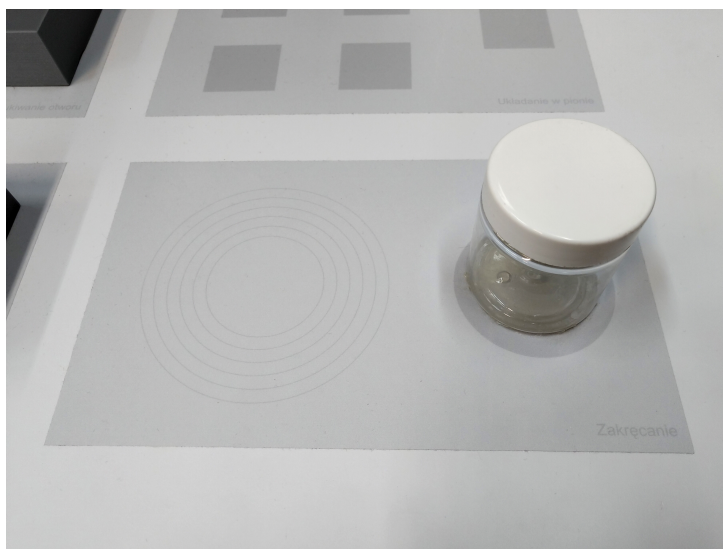
Instrukcje odkładania pobranych klocków nie muszą znajdować się wewnątrz węzła *Apply Contact Offset*, ponieważ zakłada się, że klocki są odkładane w znane miejsce, a nie w inne miejsce palety. Zalecane jest wykorzystanie polecenia paletyzacji do odkładania klocków. Zaproponowany program może, w ramach polecenia paletyzacji, wykorzystać polecenie *Spiral* z węzła *Insertion* pozwalające wyszukać gniazda do których będą odkładane obiekty. Ewentualnie, jeśli klocki będą odkładane na powierzchnię o różnej wysokości, można wykorzystać polecenie *Find Surface* do zbadania wysokości, na którą mają być odłożone klocki.

17.6 Zakręcanie i odkręcanie nakrętki

Ćwiczenie może mieć trzy postaci, przedstawione poniżej w kolejności od najtrudniejszej do najłatwiejszej w realizacji. Każda realizacja zadania powinna odbywać się poprzez kontrolę odpowiednich momentów sił oraz ewentualne usprawnienia.

1. Pobranie nakrętki, przykręcenie jej do pojemnika, a następnie odkręcenie. Przed odkręceniem nakrętki proponuje się programowe odsunięcie robota i wstrzymanie jego pracy poleceniem *Popup* w celu bezpiecznego ręcznego potwierdzenia zakręcenia nakrętki.
2. Pobranie nakrętki i przykręcenie jej do pojemnika. Po zakręceniu proponuje się zakończenie wykonywania programu w celu bezpiecznego ręcznego potwierdzenia zakręcenia nakrętki.
3. Odkręcenie nakrętki.

Strefa planszy przeznaczona do realizacji zadania jest przedstawiona na zdjęciu 27.



Rysunek 27: Strefa zadania zakręcania i odkręcania pojemnika

Ważne jest, aby punkt poboru nakrętki był jak najbardziej wycelowany względem jej punktu środkowego. Podobnie istotne jest, aby punkt podejścia do zakręcania pojemnika oraz

sam punkt zakręcania były wycelowane względem punktu środkowego pojemnika. W przeciwnym wypadku punkt środkowy nakrętki oraz punkt środkowy pojemnika przy próbie zakręcenia nie będą się pokrywać i nakrętka lub pojemnik mogą zostać uszkodzone.

Należy ostrożnie dobrać wysokość chwytania nakrętki, aby uniknąć kolizji z podłożem. Wykorzystywany chwytak posiada wprawdzie niezależny od czujnika mechanizm składania palców w razie potrzeby, jednak w ogólności nie jest to często spotykana funkcjonalność. Zaleca się, aby w trakcie chwytania nakrętki parametry `Force` oraz `Speed` miały wartość `0%`. Zbyt duża siła chwytaka lub jego prędkość zamykania spowoduje wygięcie nakrętki, co w rezultacie utrudni lub wręcz uniemożliwi jej poprawne zakręcenie, a także będzie zbędnym ryzykiem jej uszkodzenia.

Punkt podejścia do zakręcania powinien być ustawiony powyżej pojemnika. Zalecane jest ustawienie pozycji chwytaka prostopadle do podłoża, można do tego wykorzystać opcję `Reorient` z zakładki `ActiveDrive`. Przejście do punktu zakręcania pojemnika może zostać wykonane poprzez zwykłą instrukcję ruchu lub poprzez instrukcję `Wait for Force`, instrukcję oczekiwania na wykrycie siły reakcji pojemnika. Przyłożenie nakrętki do pojemnika jednym ruchem może być jednak problematyczne, nakrętka może zahaczać o krawędzie pojemnika, dlatego warto rozważyć skorzystanie z polecenia `Spiral` z węzła `Insertion`. Dobierając odpowiednio małą wartość parametru `Radius increment per turn (E)` można wprowadzić niewielkie ruch spiralny, który w rezultacie może dopasować nakrętkę do pojemnika.

Tuż przed przystąpieniem do zakręcania warto zastosować instrukcję zamykania chwytaka z parametrem `Force` ustawionym na `100%`. Zwiększy to tarcie pomiędzy chwytakiem a nakrętką, a będąc już przy pojemniku, nie będzie groziło wykrzywieniem nakrętki. Przed instrukcją rozpoczęcia zakręcania zaleca się również umieścić w programie polecenie `Popup`. Dzięki temu użytkownik będzie mógł sam podjąć decyzję o rozpoczęciu zakręcania po potwierdzeniu, że nakrętka została odpowiednio dopasowana do pojemnika (przycisk `Continue` w okienku komunikatu). Takie podejście może zapobiec uszkodzeniu nakrętki bądź słodka.

Stanowisko pracy nie pozwala na przeprowadzenie jednego obrotu chwytakiem o duży kąt ze względu na przewody podłączone do efektora, które mogą się zawinać, a nawet uszkodzić. Standardowe słodka i nakrętki są skonstruowane w taki sposób, aby do pełnego zakręcenia wystarczył obrót o maksymalnie 90° . Pojemnik na stanowisku niestety nie jest tak skonstruowany. W najlepszym przypadku do prawidłowego zakręcenia wymagany jest obrót o pewien bliżej nieokreślony kąt, którego wartość jest z przedziału od ponad 180° do 270° . Należy wziąć to pod uwagę, dobierając pozycję początkową zakręcania. Ponieważ nie można wykonać jednego obrotu o duży kąt, zakręcanie powinno znajdować się w pętli, w której nakrętka zostaje przekręcona o pewien kąt, następnie jest wypuszczana, chwytak obraca się w przeciwnym kierunku i ponownie łapie nakrętkę i przekręca ją. Obrócenie się chwytaka w przeciwnym kierunku może zostać wykonane na podstawie ruchu `Relative position`, będącego różnicą dwóch punktów, a konkretnie pozycji chwytaka w tych punktach.

Zakręcanie pojemnika powinno odbywać się na podstawie kontroli momentu obrotowego poleceniem `Force Control`. W praktyce sprawdzają się wartości z zakresu od $0.5Nm$ do $1Nm$. Ruch wynikający z momentu obrotowego o takiej wartości jest jednak bardzo wolny, zakręcenie pojemnika może trwać nawet do kilku minut. W związku z tym można wykorzystać fakt, że do całkowitego zakręcenia wymagany jest obrót o ponad 180° i zaprogramować pierwszy obrót właśnie o kąt półpełny z momentem obrotowym o wartości maksymalnej, czyli $50Nm$. Nakrętka w trakcie zakręcania porusza się w kierunku pojemnika, dlatego należy zadać ruch wynikający z kontroli siły w osi o niewielkiej wartości siły, rzędu kilku niutonów. Należy zwrócić uwagę na parametr `Deviation Range` dla momentu obrotowego, ustawiony domyślnie na zakres od -20° do 20° .

Polecenie `Force Control` zostało stworzone z myślą o pracy głównie w połączeniu z poleceniami `Path`, `Multipoint Path` oraz `Path Generator`, jednak w tym zadaniu chwytak jedynie obraca się, a nie przesuwa. Pierwszy obrót może więc zostać wykonany na dwa sposoby.

1. Obrót powinien być ograniczony do wartości 180° , do czego można wykorzystać parametr `Deviation Range`, a wewnątrz węzła kontroli sił wystarczy zadać instrukcję oczekania, `Wait`. Czas czekania powinien zostać dobrany tak, aby robot nie stał bezczynnie, ale żeby obrócić się o zadany kąt. Ten sposób został zaimplementowany w zaproponowanym programie.
2. Obrót może zostać zadany jako ścieżka `Path` lub `Multipoint Path` składająca się z dwóch punktów, gdzie pozycja robota w drugim punkcie jest obrócona względem pozycji w pierwszym o wartość poniżej 180° . Pozycja w drugim punkcie nie może być obrócona o wartość większą niż 180° , ponieważ w trakcie wykonywania programu robot wybierze obrót o mniejszą liczbę stopni, czyli obróciłby się w przeciwnym kierunku.

Kolejne obroty, znajdujące się w pętli, muszą zapewnić możliwość zakończenia zakręcania po wykryciu momentu obrotowego o zadanej wartości. Idealnym poleceniem jest `Wait for force` z węzła `Force Event`. Polecenie to pobierze wartości momentu obrotowego i siły docisku do pojemnika z węzła kontroli sił. Pamiętając, że obroty są wykonywane w pętli z powodu niemożności wykonania jednego obrotu o duży kąt, jedyny sposób to wykorzystanie parametru `Timeout` do wyznaczenia maksymalnego czasu wykonywania pojedynczego obrotu. Czas powinien być dobrany tak, aby wykonany obrót był o kąt podobny do obrotu, który robot wykona w kolejnej instrukcji w przeciwnym kierunku. W przeciwnym wypadku z każdym przebiegiem pętli, czyli dwoma obrotami w obie strony, chwytak będzie coraz bardziej obrócony w jedną ze stron, co grozi zawinięciem przewodów. Parametr `Deviation Range` nie powinien ograniczać w tym przypadku obrotów.

Podsumowując krótko etap zakręcania, składa się on z dwóch faz.

1. Obrót nakrętki o 180° w celu przyspieszenia wykonywania programu. Przy takim obrocie nakrętka nie zostanie zakręcona, więc nie jest wymagana kontrola sił. Implementacja takiego obrotu jest opcjonalna.
2. Obroty wykonywane w pętli aż do momentu wykrycia sił o zadanych wartościach. Obroty są zadawane poprzez czas wykonywania, a nie przez liczbę stopni. Po każdym obrocie nakrętka jest wypuszczana i chwytak obraca się o taką samą liczbę stopni w przeciwnym kierunku, po czym z powrotem łapie nakrętkę.

Sygnalem informującym o zakręceniu nakrętki jest wykrycie momentu sił o zadanej wartości. W przypadku odkręcania nakrętki nie ma żadnej informacji zwrotnej o zakończonym zadaniu. Wykrywanie momentu sił przy odkręcaniu nie jest konieczne, jednakże zastosowanie polecenia `Force Control` to wygodny sposób zaprogramowania ruchu obrotowego z wolnym ruchem podnoszącym nakrętkę. Ustawienie maksymalnego momentu sił zapewni najszybszy ruch obrotowy. Do zakręcenia pojemnika w przypadku optymalnie ustawionej nakrętki wymagany jest obrót o pewien kąt z zakresu od 180° do 270° . Oznacza to, że obrót zakręconej nakrętki o -270° jest gwarancją jej odkręcenia. Tę wartość wystarczy wprowadzić do parametru `Deviation Range`, a wewnątrz węzła kontroli sił zaprogramować oczekiwanie o odpowiednio dobrany czas poleceniem `Wait`. Przed instrukcją odkręcania należy ustawić robota w odpowiedniej pozycji, aby kontrolować kierunek obrotu w zaproponowanym programie, odbywa się to poprzez podwójne wykonanie ruchu `Relative Position`, odpowiedzialnego za sam obrót o 135° .

Zaleca się, aby w trakcie odkręcanie nakrętka była schwytaana z siłą 100%, natomiast po odkręceniu zalecane jest zmniejszenie siły chwytu do 0%.

18 Przydatne materiały

1. Universal Robots UR3 User Manual
https://kcir.pwr.edu.pl/~kz/Robotyka2/robotyka_zdalnie/ur3_v3.pdf,
2. Materiały szkoleniowe Universal Robots
www.universal-robots.com/pl/akademia/,
3. Robotiq Copilot & FT 300-S Sensor Instruction Manual
<https://robotiq.com/products/force-copilot>,
4. K. Zadarnowska, J. Ratajczak, Universal Robot UR3
<https://kcir.pwr.edu.pl/~kz/Robotyka3.html>,
5. Copilot, kurs internetowy
<https://elearning.robotiq.com/>.