

---

# Manipulacja dwuręczna w zadaniu układania kostki Rubika\*

Wojciech Szynkiewicz<sup>1</sup>

---

## Streszczenie

W artykule omówiono realizację zadania układania kostki Rubika jako przykładu manipulacji dwuręcznej wykonywanej przez dwa manipulatory wyposażone w proste chwytaki o dwóch szczękach równoległych. Wykonanie zadania składa się z szeregu czynności poczynając od lokalizacji kostki i rozpoznania jej stanu początkowego na podstawie analizy obrazu, przez wyznaczenie sekwencji obrotów warstw prowadzących do ułożenia kostki, na właściwej manipulacji kończąc. W pracy wykorzystano podejście polegające na dekompozycji złożonego zadania manipulacji na ciąg prostszych operacji sensomotorycznych zwanych prymitywami manipulacyjnymi lub umiejętnościami. Dla realizacji poszczególnych umiejętności kluczowe znaczenie ma właściwe wykorzystanie informacji z różnych czujników m.in. kamer wizyjnych, czujników sił i momentów, czujników dotykowych i optoelektronicznych czujników położenia.

## 1. WSTĘP

Człowiek wykonując wiele czynności wymagających manipulowania przedmiotami, posługując się dwiema rękami. Ogólnie zadania manipulacji dwuręcznej można podzielić na nieskoordynowane i skoordynowane [6]. Do pierwszej grupy zalicza się zadania, w których nie ma potrzeby koordynacji ruchów rąk. Wśród zadań skoordynowanych, wymagających synchronizacji ruchów rąk można wyróżnić zadania symetryczne i asymetryczne [5]. W zadaniach symetrycznych obie ręce wspólnie manipulują jednym obiektem, tworząc wraz nim zamknięty łańcuch kinematyczny [12]. W skoordynowanych zadaniach asymetrycznych obie ręce współpracują ze sobą, ale manipulują różnymi obiektami. Przykładem może być nalewanie wody z butelki do szklanki. W badaniach nad manipulacją dwuręczną robotów pewnych inspiracji oraz pomysłów rozwiązań może dostarczyć lektura prac poświęconych analizie czynności manualnych wykonywanych przez człowieka za pomocą obu rąk, w szczególności dotyczących zagadnień interakcji człowiek-komputer, np. [5, 6].

W robotyce, większość prac z zakresu manipulacji dotyczy tzw. manipulacji zręcznej za pomocą palców wielopalczastej sztucznej dłoni [1, 9]. Stosunkowo mniej zbadanym obszarem jest manipulacja dwuręczna [13, 15, 18], oczywiście, nie bierze się tutaj pod uwagę prac dotyczących wspólnego przenoszenia sztywno uchwyconego obiektu przez dwa lub więcej robotów (zagadnienie to jest od wielu lat przedmiotem intensywnych badań). Badania w zakresie manipulacji dwuręcznej dotyczą głów-

---

\*Praca jest finansowana przez grant MNiI: 4T11A 003 25

<sup>1</sup>Instytut Automatyki i Informatyki Stosowanej, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska, ul. Nowowiejska 15/19, 00-665 Warszawa, W.Szynkiewicz@ia.pw.edu.pl

nie operacji bimanualnych wykonywanych przez roboty humanoidalne [15], które są uczone wykonywania złożonych czynności manipulacyjnych metodą programowania przez demonstrację [18]. Prowadzone są również badania dotyczące wspólnej manipulacji wykonywanej przez człowieka i robota [14]. Wykorzystanie dwóch rąk poszerza klasę obiektów, którymi można manipulować, o przedmioty długie, elastyczne, o nieregularnych kształtach lub złożone takie jak kostka Rubika. Za pomocą dwóch rąk można wykonywać także zadania montażowe bez konieczności stosowania specjalizowanych uchwytów lub skomplikowanych urządzeń pozycjonujących [16].

Należy zauważyć, iż w pracach poświęconych problematyce manipulacji dominuje podejście bazujące na założeniu dokładnej znajomości geometrycznych relacji układu obiekt/manipulator, lokalizacji kontaktów, geometrii obiektu, końcówek palców i ogniw manipulatora oraz modelu kinematyki układu [10]. Takie założenie jest zazwyczaj uzasadnione dla robotów manipulacyjnych pracujących w otoczeniu strukturalnym (uporządkowanym). W przypadku autonomicznych robotów działających w środowisku naturalnym, lokalizacja i cechy obiektu, takie jak kształt i rozmiary są zazwyczaj nieznane lub częściowo znane [4].

Jedno z podejść do realizacji złożonych zadań manipulacji polega tworzeniu sekwencji składających się z prostych czynności (stanowiących zbiór *prymitywów manipulacyjnych – umiejętności*) bazujących na danych z różnych czujników [8]. W artykule omówiono przykład manipulacji dwuręcznej realizowanej przez dwa współpracujące manipulatory wyposażone w proste chwytaki o dwóch szczękach równoległych. Jako zadanie testowe, wymagające manipulacji wykonywanej za pomocą dwóch rąk, wybrano układanie kostki Rubika. Zadanie to zostało podzielone na kilka podzadań, dla których opracowano zbiór podstawowych umiejętności. Istotą proponowanego podejścia jest stworzenie zbioru umiejętności, które będą mogły być wykorzystane do realizacji pewnej klasy zadań, a nie tylko do jednego konkretnego zadania (w tym przypadku układania kostki Rubika). Właściwa czynność obracania ścian kostki jest przykładem skoordynowanej, symetrycznej manipulacji dwuręcznej pojedynczym obiektem. Jednakże dojście do tego etapu wymaga rozwiązania szeregu problemów, takich jak:

- Wyznaczenie początkowej pozycji kostki (lokalizacja obiektu na podstawie informacji wizyjnej).
- Zaplanowanie i realizacja bezkolizyjnej trajektorii dojścia do kostki oraz jej uchwycenia kostki (wykorzystanie serwomechanizmu wizyjnego oraz informacji z czujników nacisku).
- Rozpoznanie stanu kostki na podstawie obrazów ścian kostki (analiza obrazu).
- Rozwiązanie zadania układania kostki Rubika – określenie sekwencji ruchów ścian kostki (w miarę możliwości znalezienie rozwiązania optymalnego).
- Zaplanowanie sekwencji nominalnych chwytów i trajektorii ruchu chwytaków na podstawie wyznaczonej sekwencji ruchów ścian kostki.
- Manipulacja z wykorzystaniem bieżącej informacji z czujników (lokalne modyfikacje, zarówno chwytów jak i trajektorii ruchu chwytaków wynikające m.in. z niedokładności użytych modeli, podatności, luzów, itp.).

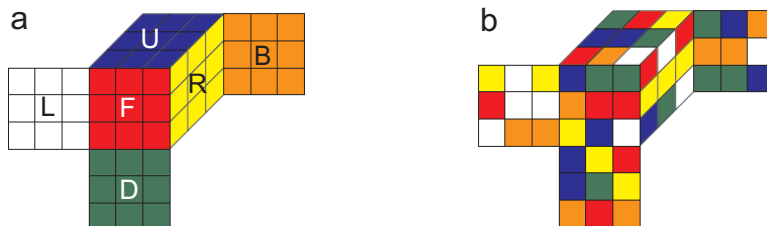
W dalszej części artykułu zostaną omówione tylko niektóre z powyższych podzadań, w szczególności dotyczące wyznaczenia planu układania kostki tj. sekwencji obro-

tów ścian kostki prowadzących do jej ułożenia oraz zaplanowania sekwencji chwytów i trajektorii chwytaków koniecznych do wykonania tego planu. Do implementacji sterowników realizujących wszystkie powyższe zadania wykorzystano programową strukturę ramową MRROC++ [17].

## 2. PLAN UKŁADANIA KOSTKI

Zakłada się, że z modułu analizy obrazów otrzymano informację o aktualnym stanie tzn. układzie ścian kostki. Kolejnym krokiem jest wyznaczenie sekwencji obrotów ścian (warstw) kostki prowadzących do jej ułożenia.

Standardowa kostka Rubika  $3 \times 3 \times 3$  jest zbudowana z 27 małych kafelków – sześciianów (26 widocznych). Z tego 6 kafelków w środku warstw nie zmienia swego położenia (zawsze pozostają w środku), zaś 20 kafelków zmienia swe położenie – 8 narożnych z trzema widocznymi ścianami i 12 na krawędziach z dwoma widocznymi ścianami. Sześcián narożny w każdej pozycji może być orientowany na 3 sposoby, każdy sześcián krawędziowy może być zorientowany na dwa sposoby. Stan kostki jednoznacznie określa podanie położenia i orientacji 20 sześciánów. Całkowita liczba potencjalnych stanów wynosi  $8! \cdot 3^8 \cdot 12! \cdot 2^{12}$ . Więzy wynikające z konstrukcji mechanicznej kostki redukują 12 razy tę liczbę. Graf przestrzeni stanów składa się z 12 oddzielnych, ale izomorficznych podgrafów (np. lustrzanych odbić), bez dopuszczalnych przejść między nimi, stąd liczba stanów osiągalnych z każdego stanu wynosi około  $4.3252 \times 10^{19}$ . Człowiek, zaczynając od losowo wybranego stanu początkowego, potrafi ułożyć kostkę wykonując zazwyczaj od 40 do 100 ruchów. Szacuje się jednak, aczkolwiek nie podano jeszcze formalnego dowodu, iż każdą kostkę można rozwiązać w co najwyżej 18 ruchów [7]. Przyjmując powszechnie stosowane oznaczenia ścian kostki (jak na rys. 1a) i oznaczając przez (F, U, D, L, R, B) – obrót odpowiedniej warstwy o kąt  $90^\circ$  (zgodnie z ruchem wskazówek zegara), (F2, U2, D2, L2, R2, B2) – obrót o kąt  $180^\circ$  oraz (F', U', D', L', R', B') obrót o kąt  $-90^\circ$ , jako ruch elementarny, to wykonując taki ruch dla jednej z sześciu ścian możemy uzyskać jeden z 18 stanów. Wykonując kolejne ruchy elementarne otrzymamy drzewo



**Rys. 1.** Kostka Rubika  $3 \times 3 \times 3$ : a – oznaczenia ścian odpowiadające kolorom środkowych kafelków, b – losowo wybrany stan kostki

stanów, w którym korzeniem jest początkowy stan zadany, a współczynnik rozgałęzienia jest równy 18. Jednak można wyeliminować pewne fragmenty sekwencji ruchów prowadzących do zdublowania stanów w drzewie, np. przez usunięcie dwóch

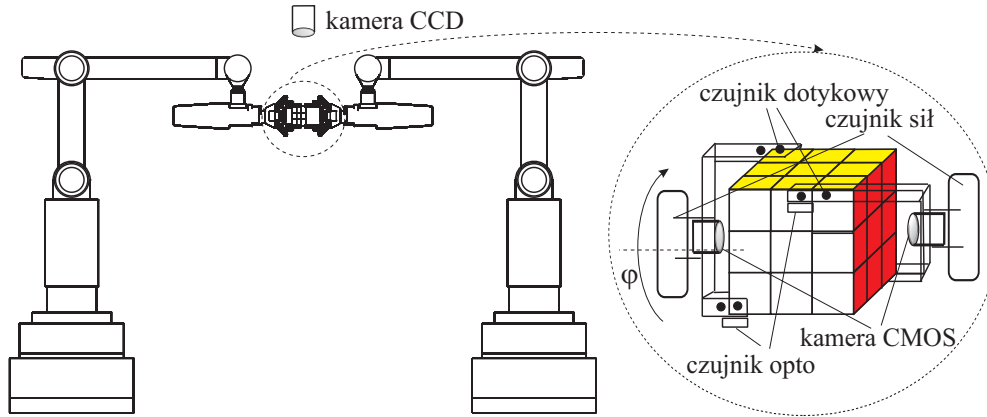
kolejnych obrotów tej samej warstwy o  $90^\circ$ , gdyż ten sam efekt uzyska się wykonując pojedynczy obrót o  $180^\circ$ . W efekcie, dzięki wykorzystaniu tego typu własności, średnia wartość współczynnika rozgałęzienia drzewa stanów wynosi 13,34847 [7]. Niestety nie da się uniknąć wszystkich powtórzeń stanów i dlatego liczba stanów w drzewie dla poziomu 18 wynosi  $2,46 \times 10^{20}$ , czyli jest prawie 6-krotnie większa od liczby wszystkich możliwych kombinacji kostki. Z powyższych rozważań wynika że, dla obliczenia optymalnego (lub suboptymalnego, jeśli czas potrzebny do znalezienia rozwiązania optymalnego jest zbyt długi<sup>†</sup>) tego zadania jest konieczne zastosowanie efektywnego algorytmu przeszukiwania drzewa stanów. W rozważanym przypadku zastosowano algorytm *Iterative Deepening A\** (IDA\*) z funkcją heurystyczną wykorzystującą obliczoną wcześniej bazę wzorców [7]. Algorytm IDA\* jest modyfikacją algorytmu A\* i bazuje na ciągu iteracji wykorzystujących przeszukiwanie włąb [11]. W każdej iteracji jest przeszukiwana przestrzeń stanów tylko do wybranego poziomu, a następnie z każdą kolejną iteracją zwiększa się numer poziomu. Zastosowanie funkcji heurystycznej  $h(n)$  (będącej optymistycznym oszacowaniem odległości węzła  $n$  od węzła docelowego) polega na obcinaniu rozgałęzień pochodzących od węzła, dla którego szacowana długość ścieżki do rozwiązania przekracza odległość tego węzła do poziomu, na którym kończy się przeszukiwanie w danej iteracji. Innymi słowy, są rozwijane tylko te węzły, dla których wartość funkcji heurystycznej  $h(n)$  spełnia nierówność  $h(n) \leq (i - p)$ , gdzie  $i$  jest numerem iteracji, a  $p$  poziomem, na którym znajduje się dany węzeł. W aktualnie zaimplementowanej wersji algorytmu znalezienie rozwiązania optymalnego wymaga kilku sekund (dla obliczeń wykonywanych na komputerze typu PC z procesorem 3GHz i pojemnością pamięci operacyjnej 512MB) dla przypadków wymagających co najwyżej 15 ruchów (obrotów warstw kostki). Jeśli rozwiązanie wymaga więcej niż 15 ruchów, to poszukiwanie rozwiązania optymalnego jest zbyt czasochłonne i jest wyznaczane rozwiązanie suboptymalne (w czasie nie przekraczającym 2 min.), które zazwyczaj nie przekracza 20-21 ruchów. Przykładowo, dla losowo wybranej kostki, której stan początkowy jest przedstawiony na rys. 1b, rozwiązaniem jest następujący ciąg (B2 R2 U' F2 U' R2 D' L2 B' L' F' D U' R' U' L F D F2 R) obrotów odpowiednich warstw. Zgodnie z założeniem nie występują dwa kolejne obroty tej samej ścianki. Kolejnym krokiem jest transformacja ruchów ścian na ruchy chwytaków robotów.

### 3. MANIPULACJA DWURECZNA

#### 3.1. Stanowisko badawcze

Stanowisko badawcze składa się z dwóch robotów IRp-6 (o 6 stopniach swobody każdy) wyposażonych w chwytaki dwupalczaste (każdy o 1 stopniu swobody) jak pokazano na rys. 2. Szczęki chwytaka pozostają równoległe w całym zakresie otwierania/zamykania. W nadgarstkach obu ramion zamontowano czujniki sił (czujnik sił/momentów Gamma firmy ATI SI-65-5) mierzące 6 składowych siły uogólnionej (3 składowe siły oraz 3 składowe momentów sił). W każdej z dwóch szczęk chwytaka są umieszczone 4 rezystancyjne czujniki siły nacisku służące do wykrycia kontaktu z

<sup>†</sup>Zakładamy, że w praktycznej realizacji akceptowalne są czasy obliczeń nie przekraczające 1-2 minut.



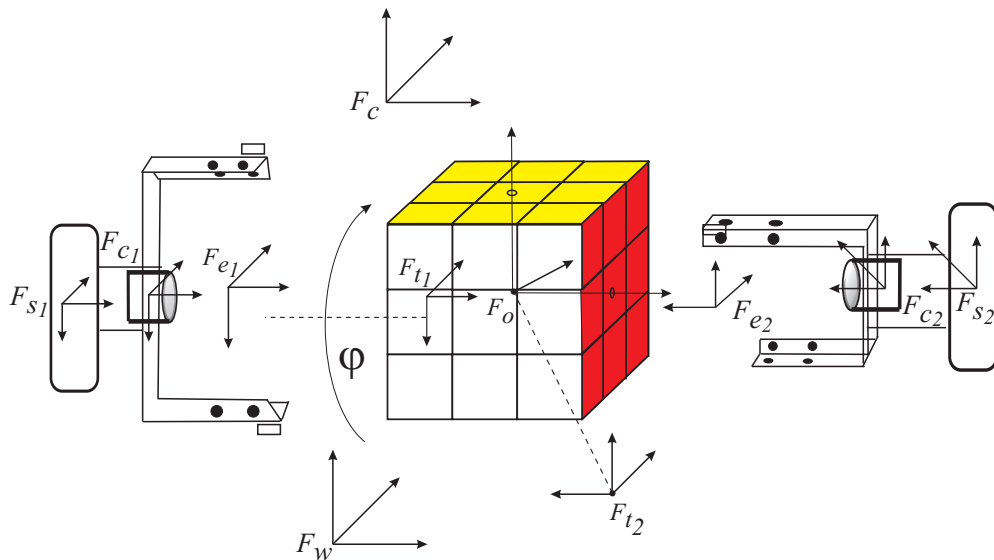
Rys. 2. Stanowisko do manipulacji dwuręcznej

chwytanym obiektem. Ich rozmieszczenie dobrano tak, aby dwa czujniki (dla każdej szczęki) były aktywne w momencie, gdy chwytak uchwycił jedną warstwę kostki. Analogicznie sygnalizacja kontaktu z czterech czujników informuje o uchwyceniu dwóch warstw kostki. Na szczękach zamontowano także dwa (po jednym na szczęcie) optoelektroniczne czujniki położenia do wykrywania naprzeciwległego położenia dwóch chwytaków. Ponadto w chwytakach umieszczono miniaturowe kamery kolorowe CMOS. Stacjonarna kamera CCD jest wykorzystywana do lokalizacji kostki.

### 3.2. Opis zadania

W zadaniu układania kostki Rubika planowanie manipulacji wymaga wyznaczenia bezkolizyjnych trajektorii dojścia obu chwytaków do obiektu, a następnie obliczenia sekwencji składającej się z chwytów, trajektorii ruchów chwytaków przy zmianie chwytu (przechwytywaniu), oraz trajektorii ruchu chwytaków realizujące kolejne obroty warstw kostki. Dla potrzeb opisu zadania manipulacji zdefiniowano prawoskrętne, kartezjańskie układy współrzędnych jak na rys. 3. Pozycję układu  $\mathcal{F}_j$  w układzie  $\mathcal{F}_i$  opisuje wektor  ${}^i\mathbf{p}_j \in \mathbb{R}^3$  położenia początku układu  $\mathcal{F}_j$  w układzie  $\mathcal{F}_i$  i macierz obrotu  ${}^i\mathbf{R}_j \in \text{SO}(3)$  ( $\text{SO}(3)$ —specjalna grupa obrotów [9]) układu  $\mathcal{F}_j$  względem układu  $\mathcal{F}_i$ . Jeśli macierze rotacji  $\mathbf{R}_i$  i  $\mathbf{R}_j$  opisują orientację układów  $\mathcal{F}_i$  oraz  $\mathcal{F}_j$  w układzie bazowym, to wzajemna orientacja między tymi układami jest opisana macierzą  ${}^i\mathbf{R}_j = \mathbf{R}_i^T \mathbf{R}_j$ . W dalszej części, gdy układem odniesienia będzie układ globalny  $\mathcal{F}_w$  lewy górny indeks będzie pomijany. Dla określenia pozycji obiektu manipulacji wprowadzono lokalny układ  $\mathcal{F}_o$  związany z kostką.

Do opisu geometrycznego i kinematycznego zadania manipulacji, a w szczególności sterowania pozycyjno-siłowego zastosowano rozszerzoną postać formalizmu TFF (*Task Frame Formalizm* [2]). Modyfikacja polegała na wprowadzeniu dwóch układów współrzędnych  $\mathcal{F}_i$ ,  $i = 1, 2$  ściśle związane z geometrią i kinematyką zadania w przestrzeni operacyjnej tzw. układów zadania. W trakcie manipulacji kierunki zadawania prędkości i siły będą pokrywały się z wersorami osi tych układów. Z chwytakami związane układy współrzędnych  $\mathcal{F}_{e_i}$ ,  $i = 1, 2$ , aktualna pozycja układu



Rys. 3. Układy współrzędnych wykorzystywane w opisie zadania manipulacji dwuręcznej

chwybaka (efektora)  $i$ -tego manipulatora jest określona przez  $(\mathbf{p}_i(\mathbf{q}_i), \mathbf{R}_i(\mathbf{q}_i))$ , gdzie  $\mathbf{q}_i$  jest wektorem  $(n_i \times 1)$  współrzędnych przegubowych  $i$ -tego manipulatora. Pozycja zadana dla chwytaków będzie natomiast określana przez położenie i orientację układów  $\mathcal{F}_i$ . Ponadto wprowadzono układy związane z czujnikami, z czujnikami siły  $\mathcal{F}_s$ , oraz kamerami w chwytakach  $\mathcal{F}_c$ ,  $i = 1, 2$ .

Wyznaczona sekwencja obrotów warstw kostki może być opisana w układzie  $\mathcal{F}_o$  jako ciąg kolejnych obrotów wokół wektorów  $\hat{\mathbf{x}}_o, \hat{\mathbf{y}}_o, \hat{\mathbf{z}}_o$  osi tego układu, tj.  ${}^o\mathbf{R}(\mathbf{u}, \varphi) = \mathbf{Rot}(\mathbf{u}, \varphi)$ , gdzie  $\mathbf{u} = \hat{\mathbf{x}}_o, \hat{\mathbf{y}}_o, \hat{\mathbf{z}}_o$ , a  $\varphi = -\frac{\pi}{2}, -\pi, \frac{\pi}{2}, \pi$ . Dlatego też, zadana trajektoria ruchu dla chwytaka może być opisana w układzie kostki jako obrót wokół jednej z osi tego układu o zadany kąt  $\varphi$ . Naturalną, w takim przypadku, parametryzacją orientacji jest reprezentacja oś-kąt (dla uniknięcia osobliwości reprezentacji można wykorzystać wersję z kwaternionami jednostkowymi tj. parametrami Eulera). Obrót wokół osi określonej przez wektor jednostkowy  $\mathbf{u}$  o kąt  $\varphi$  można reprezentować za pomocą kwaternionu jednostkowego  $Q$  zdefiniowanego jako [3]:

$$Q = \{\eta, \mathbf{e}\}, \quad \text{gdzie } \eta = \cos \frac{\varphi}{2}, \quad \mathbf{e} = \mathbf{u} \sin \frac{\varphi}{2}, \quad (1)$$

z częścią skalarną kwaternionu  $\eta \geq 0$  dla  $\varphi \in [-\pi, \pi]$ . Należy zauważyć, że obrót  $(-\mathbf{u}, -\varphi)$  oraz obrót  $(\mathbf{u}, \varphi)$  odpowiadają temu samemu kwaternionowi, zaś obrót  $(\mathbf{u}, -\varphi)$  daje część wektorową  $\mathbf{e}$  kwaternionu o przeciwnym znaku niż, obrót  $(\mathbf{u}, \varphi)$ , podczas gdy część skalarna  $\eta$  pozostaje bez zmian. Macierz obrotu odpowiadającą danemu kwaternionowi można wyznaczyć ze wzoru

$$\mathbf{R}(\eta, \mathbf{e}) = (\eta^2 - \mathbf{e}^T \mathbf{e}) \mathbf{I}_{3 \times 3} + 2\mathbf{e}\mathbf{e}^T - 2\eta \tilde{\mathbf{e}}, \quad (2)$$

gdzie  $\mathbf{I}_{3 \times 3}$  jest macierzą jednostkową, a  $\tilde{\mathbf{e}} = \mathcal{S}(\mathbf{e})$  macierzą stowarzyszoną z wektorem  $\mathbf{e}$ , czyli macierzą skośnosymetryczną [9]. Ponieważ  $\|Q\| = 1$ , to parametry Eulera

nie są niezależne

$$\|Q\|^2 = \eta^2 + e_1^2 + e_2^2 + e_3^2 = 1 \quad (3)$$

Mając daną macierz obrotu można wyznaczyć odpowiadający jej kwaternion jednostkowy (uwzględniając warunek (3)) z

$$\eta = \frac{1}{2} \sqrt{\text{tr}(\mathbf{R}) + 1} \quad (4)$$

$$\mathbf{e} = \begin{bmatrix} \frac{1}{2} \text{sgn}(r_{32} - r_{23}) \sqrt{r_{11} - r_{22} - r_{33} + 1} \\ \frac{1}{2} \text{sgn}(r_{13} - r_{31}) \sqrt{r_{22} - r_{33} - r_{11} + 1} \\ \frac{1}{2} \text{sgn}(r_{21} - r_{12}) \sqrt{r_{33} - r_{22} - r_{11} + 1} \end{bmatrix}, \quad (5)$$

gdzie  $\text{tr}(\mathbf{R})$  jest śladem macierzy  $\mathbf{R}$ , zaś  $r_{ii}$  są elementami na jej diagonalu. Niech wektor  $\mathbf{p}_{di}$  opisuje położenie zadane dla  $i$ -tego chwytaka. Zdefiniujmy uchyb położenia jako

$$\varepsilon_{pi} = \mathbf{p}_{di} - \mathbf{p}_i(\mathbf{q}_i), \quad i = 1, 2 \quad (6)$$

Niech kwaterniony  $Q_{di} = \{\eta_{di}, \mathbf{e}_{di}\}$  i  $Q_i = \{\eta_i, \mathbf{e}_i\}$  związane odpowiednio z macierzami  $\mathbf{R}_{di}$  i  $\mathbf{R}_i$  opisują orientację zadaną i aktualną  $i$ -chwytaka. Orientacja względna może być wyrażona za pomocą kwaternionu jednostkowego  $\Delta Q_i = \{\Delta \eta_i, \Delta \mathbf{e}_i\}$ , gdzie

$$\Delta \eta_i = \eta_i \eta_{di} + \mathbf{e}_i^T \mathbf{e}_{di} \quad (7)$$

$$\Delta \mathbf{e}_i = \eta_i \mathbf{e}_{di} - \eta_{di} \mathbf{e}_i - \tilde{\mathbf{e}}_{di} \mathbf{e}_{di} \quad (8)$$

Ponieważ łatwo można wykazać, że  $\Delta Q_i = \{1, \mathbf{0}\}$  wtedy i tylko wtedy, gdy układy pokrywają się, tj.  $\mathbf{R}_{di}^T \mathbf{R}_i = \mathbf{I}_{3 \times 3}$ , stąd wystarczy uwzględnić tylko  $\Delta \mathbf{e}_i$  dla określenia uchybu orientacji  $i$ -tego chwytaka. Uchyb orientacji można zatem opisać zależnością

$$\varepsilon_{O_i} = \eta_i(\mathbf{q}_i) \mathbf{e}_{di} - \eta_{di} \mathbf{e}_i(\mathbf{q}_i) - \tilde{\mathbf{e}}_{di} \mathbf{e}_{di}(\mathbf{q}_i), \quad i = 1, 2 \quad (9)$$

Bezpośrednie obliczenie  $\eta_i(\mathbf{q}_i)$  i  $\mathbf{e}_i(\mathbf{q}_i)$  nie jest możliwe, najpierw należy wyznaczyć macierz rotacji  $\mathbf{R}_i$  rozwiązując proste zadanie kinematyki dla  $i$ -tego manipulatora, a następnie wyznaczyć jednostkowy kwaternion z zależności (4) i (5).

Dla opisu skoordynowanych ruchów manipulatorów (gdy po uchwyceniu kostki powstaje zamknięty łańcuch kinematyczny) jest konieczne określenie wzajemnego położenia i orientacji chwytaków. Wzajemne położenie jest określone jako wektor

$$\Delta \mathbf{p}_{12} = \mathbf{p}_1 - \mathbf{p}_2 \quad (10)$$

Wzajemna orientacja chwytaków jest definiowana w  $\mathcal{F}_{e_2}$  jako układzie odniesienia za pomocą macierzy  ${}^2\mathbf{R}_1$ , a następnie w postaci iloczynu kwaternionów

$$Q_{12} = Q_2^{-1} * Q_1, \quad (11)$$

gdzie  $Q_2^{-1} = \{\eta_2, -\mathbf{e}_2\}$  jest kwaternionem sprzężonym z  $Q_2$ , czyli kwaternionem jednostkowym odpowiadającym macierzy  $\mathbf{R}_2^T$  [3].

### 3.3. Planowanie i realizacja zadania manipulacji

W niniejszej pracy wykorzystano podejście polegające na dekompozycji złożonego zadania manipulacji na ciąg prostszych operacji sensomotorycznych zwanych prymitywami manipulacyjnymi lub umiejętnościami. Termin operacja sensomotoryczna oznacza tutaj połączenie czucia (odczyty z czujników) i akcji (ruchy robota) w czynność będącą realizacją pewnego elementarnego zadania. Zdefiniowano następujące elementarne umiejętności wykorzystujące informacje z różnych czujników *Move(pose)*, *CompliantMove(pose)*, *Gripper(distance)*, *LocateObject(object)*. Do pełnego opisu umiejętności potrzebna jest informacja o użytych czujnikach, warunkach początkowym i końcowym, pozycji docelowej, itp. Implementacja powyższych umiejętności w ramowej strukturze MRROC++ nie nastęrcza większych trudności, gdyż stanowi naturalne rozszerzenie dotychczasowych jej możliwości, zaś do tworzenia umiejętności wykorzystuje się już istniejące elementy struktury MRROC++ (np. generatory ruchu [17]).

Do zgrubnej lokalizacji kostki jest wykorzystywana umiejętność *LocateObject(kostka)*, w której źródłem informacji jest stacjonarna kamera CCD. Na podstawie analizy obrazu jest estymowana bieżąca pozycja układu kostki  $\mathcal{F}_o$  względem układu  $\mathcal{F}_c$  związanego z stacjonarną kamerą. Dzięki znajomości pozycji układu  $\mathcal{F}_c$  w globalnym układzie odniesienia  $\mathcal{F}_w$  (kalibracja układu kamera-robot) można obliczyć pozycję kostki  $(\mathbf{p}_o, \mathbf{R}_o)$  w układzie globalnym  $\mathcal{F}_w$ .

Obraz z kamer umieszczonych w chwytakach jest wykorzystywany w operacji *Move* $(\mathbf{p}_{di}, Q_{di})$  (realizującej serwomechanizm wizyjny) do obliczania bieżącej pozycji kostki, a następnie do ustawienia  $i$ -tego chwytaka względem kostki w pozycji umożliwiającej poprawny chwyt. Istotnym problemem jest wybór takiej konfiguracji obu ramion, kiedy tworzą one wraz z uchwyconą kostką zamknięty łańcuch kinematyczny, aby możliwe było wykonanie całego obrotu o zadany kąt bez konieczności zmiany uchwytu (przechwytywania). Pozycja  $(\mathbf{p}_{di}, Q_{di})$  układu  $\mathcal{F}_i$  określa zadaną konfigurację chwytu dla  $i$ -tego chwytaka (w idealnym przypadku układ  $\mathcal{F}_{e_i}$  pokrywa się z układem  $\mathcal{F}_i$ ). Poprawność chwytu – zacisk szczęk jest realizowany za pomocą operacji *Gripper* $(d_k)$  – jest określana na podstawie odczytów z rezystancyjnych czujników nacisku.

Umiejętność *CompliantMove* $(\mathbf{p}_{di}, Q_{di})$  realizuje sterowanie pozycyjno-siłowe w układzie zadania  $\mathcal{F}_i$  (zgodnie z TFF) i jest wykorzystywana w trakcie obrotów warstw kostki, gdy ruchy chwytaków muszą być ściśle koordynowane. Umiejętność ta wykorzystuje informację o sile oddziaływań między kostką a chwytakami w celu unikania zakleszczeń oraz z dane czujników optoelektronicznych położenia dla wykrycia warunku zakończenia obrotu warstwy (naprzeciwległego ustawienia szczęk chwytaków). W aktualnej wersji zakładamy, że tylko jeden z chwytaków obraca ściankę, drugi jest sterowany siłowo w taki sposób, aby niwelować naprężenia w zamkniętym łańcuchu kinematycznym i zapobiegać zakleszczeniom warstw kostki Rubika. Przejście do wykonania obrotu kolejną ścianką wymaga zmiany chwytu, to z kolei wymaga zaplanowania bezkolizyjnej trajektorii przechwytywania. Przy jej planowaniu wykorzystuje się znajomość modelu geometrycznego chwytaka.



#### 4. PODSUMOWANIE

Wykonanie stosunkowo złożonego zadania manipulacji jakim jest układanie kostki za pomocą dwóch manipulatorów IRp-6 było możliwe dzięki podziałowi zadania na prostsze czynności zwane prymitywami manipulacyjnymi bądź umiejętnościami. Umiejętności bazują na danych uzyskiwanych z różnych czujników, m.in. kamer (stacjonarnej i ruchomych), czujników sił i momentów, czujników nacisku. Dzięki wykorzystaniu własności programowej struktury ramowej MRROC++ i już istniejących w niej mechanizmów, implementacja powyższych umiejętności nie następuje poważnych problemów i stanowi naturalne rozszerzenie możliwości tego środowiska. Umiejętności te mogą być wykorzystane do realizacji innych zadań manipulacji.

#### LITERATURA

- [1] A. Bicchi, V. Kumar. Robotic grasping and manipulation. In: *Ramsete: Articulated and Mobile Robots for Services and Technology* Red. S. Nicosia et al, Vol. 270, s. 55–74. Springer-Verlag 2001.
- [2] H. Bruyninckx, J. De Schutter. Specification of force-controlled actions in the task frame formalism: A synthesis.
- [3] J. C. K. Chou. Quaternion kinematic and dynamic differential equations. *IEEE Trans. on Robotics and Automation*, 192, Vol. 8, No. 1, s. 53–64.
- [4] A. Dollar, R. Howe. Towards grasping in unstructured environments: Grasper compliance and configuration optimization. *Advanced Robotics*, 2005, Vol. 19, No. 5, s. 523–543.
- [5] Y. Guiard. Asymmetric division of labor in skilled bimanual action: the kinematic chain as a model. *Journal of Motor Behavior*, 1987, Vol. 19, No. 4, s. 486–517.
- [6] K. Hinckley et al. Two-handed virtual manipulation. *ACM Transactions on Computer-Human Interaction*, 1998, Vol. 5, No. 3, s. 260–302.
- [7] R. Korf. Finding optimal solutions to Rubik's cube using pattern databases. In: *National Conf. on Artificial Intelligence (AAAI-97). Proceedings*, July, 1997, s. 700–705.
- [8] J.D. Morrow, P. K. Khosla. Manipulation task primitives for composing robot skills. In: *IEEE Int. Conf. on Robotics and Automation. Proceedings*, April, 1997, s. 3354–3359.
- [9] R. M. Murray, Z. Li, S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press 1994.
- [10] A.M. Okamura, N. Smaby, M.R. Cutkosky. An overview of dexterous manipulation. In: *IEEE Int. Conf. on Robotics and Automation. Proceedings*, April, 2000, s. 255–262.
- [11] S. Russell, P. Norvig. *Artificial Intelligence: A Modern Approach*. Upper Saddle River, N.J., Prentice Hall 1995.
- [12] W. Szynekiewicz. Motion planning for multi-robot systems with closed kinematic chains. In: *9th IEEE Int. Conf. on Methods and Models in Automation and Robotics. Proceedings*, Międzyzdroje, Poland, August 25-28, 2003, s. 779–786.

- [13] W. Szyrkiewicz. Planowanie manipulacji dwuręcznej. In: *Postępy robotyki. Sterowanie robotów z percepcją otoczenia* Red. K. Tchoń, s. 187–196. Warszawa, WKŁ 2005.
- [14] T. Takubo et al. Human-robot cooperative manipulation using a virtual non-holonomic constraint. *International Journal of Robotics Research*, May-June, wolumen 21, numer 5–6,.
- [15] E. Yoshida, P. Blazevic, V. Hugel. Pivoting manipulation of a large object: A study of application using humanoid platform. In: *IEEE International Conference on Robotics and Automation. Proceedings*, April 18-22, 2005, s. 1052–1057.
- [16] J. Zhang, Y. von Collani, A. Knoll. Interactive assembly by a two-arm-robot agent. *Robotics and Autonomous Systems*, 1999, Vol. 29, No. 1, s. 91–100.
- [17] C. Zieliński. The MRROC++ system. In: *1st Workshop on Robot Motion and Control, RoMoCo'99. Proceedings*, Kiekrz, Poland, June 28–29, 1999, s. 147–152.
- [18] R. Zöllner, T. Asfour, R. Dillmann. Programing by demonstration: Dual-arm manipulation tasks for humanoid robots. In: *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2004)*, Sendai, Japan. *Proceedings*, Sep. 28–Oct. 2, 2004. Vol. 1, s. 479–484.

#### **A TWO-HANDED MANIPULATION FOR SOLVING RUBIK'S CUBE PUZZLE**

The paper describes a solution of the Rubik's cube as an example of a complex two-handed manipulation. The manipulation task has been decomposed into a sequence of sensorimotor primitives – skills. Skills represent higher-level programming primitives which are task-relevant.