

Jarosław Warczyński
Politechnika Poznańska

SYSTEM NIEJAWNEGO PROGRAMOWANIA ROBOTÓW

Streszczenie. W pracy przedstawiono koncepcję systemu automatycznego programowania robotów dla celów montażu, ze szczególnym uwzględnieniem zagadnienia formalizacji wyznaczania ograniczeń kolejnościowych w zbiorze operacji montażowych. Projektowany system przeznaczony jest do pełnienia funkcji interfejsu pomiędzy projektowaniem wspomaganym komputerem - CAD a automatycznym montażem za pomocą robota.

1. Wstęp

Jak wiadomo, jednym z najbardziej perspektywicznych kierunków rozwoju zautomatyzowanych systemów wytwórczych okazały się tzw. Elastyczne Systemy Produkcyjne w połączeniu z filozofią C I M [2,4]. Istotnym elementem takich systemów są roboty przemysłowe. Należy jednak zauważyć, że prawdziwie efektywne zastosowanie robotów do wielu podstawowych zadań wytwórczych wiąże się z koniecznością rozwiązania wielu niełatwych problemów z dziedziny programowania, sztucznej inteligencji, sensoryki itp.

Jednym z najtrudniejszych zadań stawianych przed robotyką jest zadanie montażu, w którym ogniskują się prawie wszystkie z wymienionych zagadnień badawczych.

W niniejszej pracy, prezentującej ogólną koncepcję systemu automatycznego generowania programów montażu dla robotów, chcielibyśmy zwrócić uwagę na niektóre aspekty tego problemu, ze szczególnym uwzględnieniem zagadnienia formalizacji wyznaczania ograniczeń kolejnościowych w zbiorze operacji montażowych.

Proponowany system stanowi rozwinięcie, wykonanego w ramach CPBR 7.1 "Roboty przemysłowe", systemu symulacji i programowania robotów [5,6]. Dodajmy, że pod pojęciem programowania niejawnego (ang. *implicit programming*) rozumiemy taki sposób programowania robota, który nie wymaga układania szczegółowego programu jego czynności przez programistę, lecz ogranicza się do ogólnego wytyczenia zadania. Szczegółowy program wykonania zadania jest generowany automatycznie.

Programowanie niejawne pozwala na zredukowanie nakładów ponoszonych przy przygotowywaniu programów dla robotów metodami tradycyjnymi (*teach-in* lub w języku wysokiego poziomu), wymaga jednak zastosowania wyspecjalizowanego systemu przetwarzania wiedzy oraz bazuje na zaawansowanych systemach sensorycznych.

2. Ogólny opis systemu

Zadanie automatycznej generacji programu montażu można rozłożyć na kilka podzadań, do których zaliczamy:

- formalizację zadania,
- zdefiniowanie kolejności czynności prowadzących do wykonania zadania,
- zaplanowanie sposobu wykonania czynności,
- wyznaczenie trajektorii ruchów w ramach każdej czynności,
- wygenerowanie kodu sterującego dla zaplanowanej sekwencji ruchów.

Zgodnie z powyższym założeniem, w ramach systemu programowania niejawnego powinny znajdować się moduły funkcjonalne odpowiedzialne za realizację wyszczególnionych podzadań.

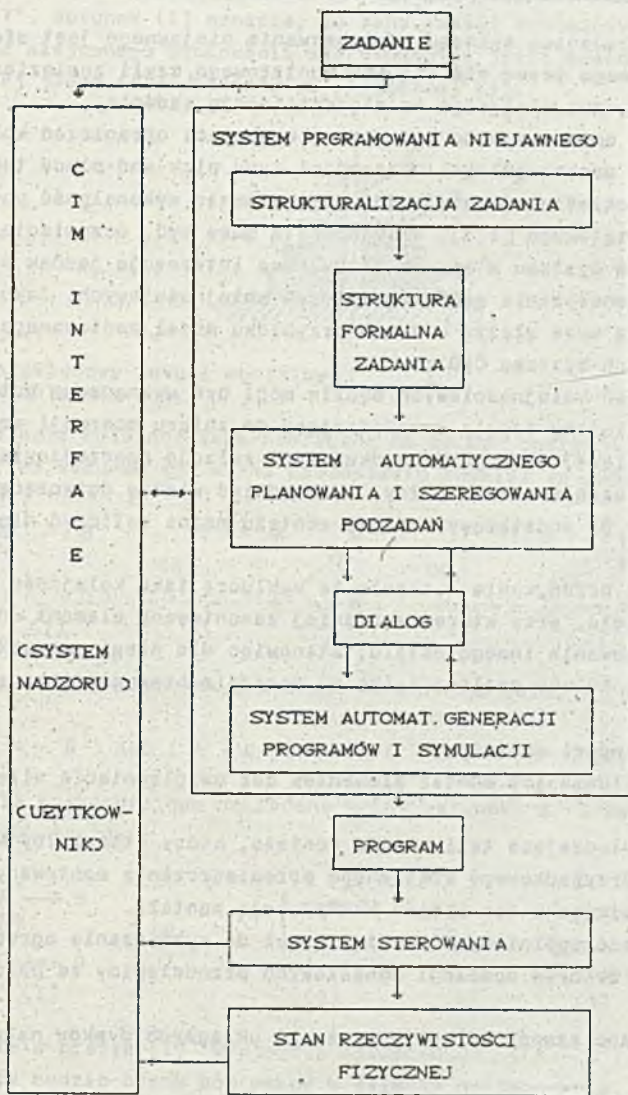
Koncepcję takiego systemu obrazuje schemat rys. 1, przedstawiający jego ogólną strukturę oraz współdziałanie jego modułów funkcjonalnych.

Istotnym założeniem koncepcyjnym dla projektowanego systemu jest możliwość włączenia go do szerszej struktury tzw. komputerowo zintegrowanego wytwarzania, co na schemacie z rys. 1 odzwierciedla interfejs CIM. Celem takiego założenia jest uzyskanie możliwości automatycznego wygenerowania programu montażu dla robota, na podstawie modelu przeznaczonego do zmontowania zespołu montażowego. Przez model ten może być tutaj rozumiana struktura danych powstała przy projektowaniu danego zespołu w ramach systemu projektowania wspomaganego komputerem - CAD.

W tym kontekście projektowany system programowania pełniłby funkcję analogiczną do tej, jaką spełniają systemy generacji programów obróbki dla obrabiarek sterowanych numerycznie, będące "przedłużeniem" systemów CAD, czy też spełniające funkcję interfejsu CAD - CAM.

Automatyczna generacja programów montażu dla robota jest zagadnieniem, z bardzo wielu względów, znacznie trudniejszym niż generacja programu obróbki dla obrabiarki i dlatego też przewidujemy stopniowe jego rozwiązywanie, poczynając od pewnej ograniczonej, aczkolwiek dostatecznie szerokiej klasy zadań, w której wszystkie niezbędne czynności montażowe są operacjami typu "pobierz i umieść" (ang. pick-and-place). Będą to więc operacje, w których nie ma potrzeby manipulowania np. z elementami elastycznymi, czy sprężystymi lub elementami wymagającymi połączeń gwintowych.

Powyzsze zawęzenie rozpatrywanego zagadnienia automatycznego generowania programów montażu pozwala na wyodrębnienie problemu, którego rozwiązanie leży bardziej w gestii metod automatycznego wnioskowania niż szeroko rozumianej sensoryki, a jednocześnie ma dostatecznie szerokie odniesienie do praktyki zrobotyzowanego montażu.



Rys. 1. Ogólna struktura systemu
 Fig. 1. The global structure of the system

3. Formalizacja zadania - graf ograniczeń kolejnościowych

Podstawowym zadaniem systemu programowania niejawnego jest sformalizowanie postawionego przed nim zadania montażowego, czyli znalezienie jego opisu formalnego, pozwalającego na algorytmizację zadania.

Zakładamy, że opis ten może być podany w postaci ograniczeń kolejnościowych pomiędzy poszczególnymi operacjami typu pick-and-place tak, aby dokonanie ich w określonej kolejności gwarantowało wykonalność postawionego zadania montażowego [1,3]. Kolejność ta może być, oczywiście, zadana przez użytkownika systemu w sposób jawny, nas interesuje jednak możliwość automatycznego generowania grafu ograniczeń kolejnościowych. Jako dane wejściowe zadania może służyć w takim przypadku model montowanego zespołu utworzony w ramach systemu CAD.

Graf ograniczeń kolejnościowych będzie mógł być wyznaczony automatycznie, jeśli zdefiniujemy reguły wprowadzające do zbioru operacji montażowych (pick-and-place) relację porządkującą - relację poprzedzania. Reguły te powinny, w sposób możliwie pełny, uwzględniać wiedzę dotyczącą procesów montażowych. Do podstawowych reguł montażu można zaliczyć między innymi następujące:

1. Reguła zakazu przenikania - reguła ta wyklucza taką kolejność montażu detali w zespole, przy której wcześniej zamontowany element uniemożliwiłby zamontowanie innego detalu, stanowiąc dla niego przeszkodę, której ominięcie byłoby możliwe tylko na zasadzie nierealistycznego przenikania.
2. Reguły stabilności montażu -
 - a/ reguła wykluczająca montaż elementów bez uwzględniania siły grawitacji,
 - b/ reguła wykluczająca taki sposób montażu, który stwarzałby możliwość łatwego, przypadkowego wzajemnego przemieszczania montowanych detali, uniemożliwiając w ten sposób kontynuację montażu.

Zastosowanie wyszczególnionych powyżej reguł do wyznaczania ograniczeń kolejnościowych w zbiorze operacji montażowych przedstawimy za pomocą przykładu:

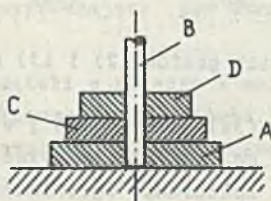
Na rys. 2 pokazano zespół składający się z okrągłych dysków nałożonych na pręt.

Należy wyznaczyć relacje poprzedzania dla procesu montażu zespołu z rys.2, które będą stanowiły podstawę do określenia właściwej kolejności montażu. Właściwa kolejność czynności montażowych musi spełniać wszystkie warunki wynikające z formalnego zastosowania reguł montażu 1, 2a i 2b.

Reguła 1. prowadzi do następującej alternatywy relacji:

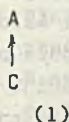
$$(C - A) \text{ OR } (C - B) \text{ OR } (C - D) \quad (1)$$

gdzie (X - Y) oznacza: "zamontowanie elementu X poprzedza zamontowanie elementu Y". Warunek (1) oznacza, że dany zespół montażowy może być zmontowany bez wzajemnego blokowania się elementów, jeśli spełniony będzie co najmniej jeden człon alternatywy logicznej (1).

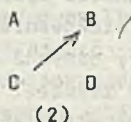


Rys. 2. Przykładowy zespół montażowy
Fig. 2. Example assembly

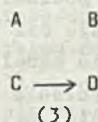
Ograniczenia kolejnościowe nakładane na zbiór operacji montażowych A, B, C, D przez warunek (1) można przedstawić również za pomocą grafów:



OR



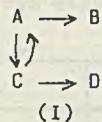
OR



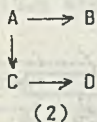
Reguła 2a produkuje warunek o postaci koniunkcji relacji poprzedzania.

$$(A \sim B) \text{ AND } (A \sim C) \text{ AND } (C \sim D) \quad (2)$$

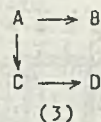
Ograniczenia kolejnościowe nakładane przez warunek 1 i 2a mają postać następujących grafów:



OR



OR

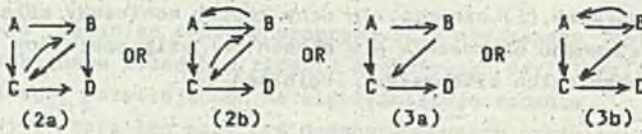


Zauważmy, że w grafie (1) występuje sprzeczność ((A ~ C) AND (C ~ A)). Graf ten będzie brany pod uwagę w dalszym postępowaniu, gdyż spełnienie sprzecznych ograniczeń kolejnościowych nie jest możliwe, a co za tym idzie, graf ten nie może posłużyć do wyznaczania kolejności czynności montażowych, która musi spełniać łącznie warunki wynikające z reguł 1, 2a i 2b.

Reguła 2b produkuje alternatywę koniunkcji relacji poprzedzania:

$$(B \sim C) \text{ AND } (B \sim D) \text{ OR } (B \sim A) \text{ AND } (B \sim C) \quad (3)$$

Ograniczenia kolejnościowe wynikające z koniunkcji warunków (1 - 3) można przedstawić za pomocą następujących grafów:



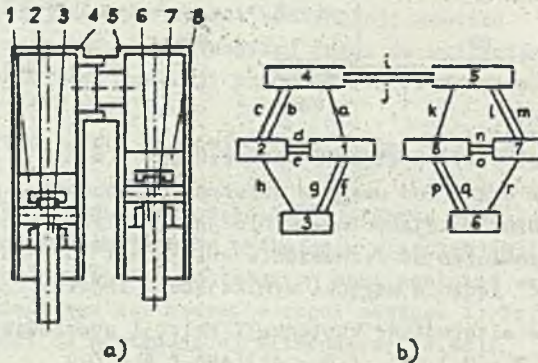
które powstają z poprzednich grafów (2) i (3) po dodaniu ograniczeń wynikających z warunku (3).

Zauważmy, że jedynie graf (3a) jest grafem, w którym nie ma sprzecznych ograniczeń kolejnościowych, co oznacza, że tylko ograniczenia tego grafu będą mogły być w praktyce spełnione. Ograniczenia te jednoznacznie wyznaczają kolejność operacji montażowych w postaci sekwencji $A \rightarrow B \rightarrow C \rightarrow D$.

Istotną trudnością w omawianym powyżej postępowaniu jest wyznaczanie ograniczeń kolejnościowych dla operacji montażowych bezpośrednio z modelem montowanego zespołu o postaci struktury danych, istniejącej w systemie CAD.

Innym sposobem poszukiwania kolejności operacji montażowych, gwarantującej wykonanie montażu bez wzajemnego blokowania się detali, jest metoda polegająca na analizie modelu (struktury danych) montowanego zespołu pod kątem możliwości rozłożenia (demontażu) zespołu. Wyznaczona sekwencja czynności demontażu może stanowić podstawę do określenia kolejności operacji w montażu zespołu, gdyż jest ona odwrotna niż przy jego rozkładaniu.

W podejściu tym istotne jest znalezienie algorytmu demontażu, który gwarantowałby całkowite rozłożenie zespołu. Niewłaściwe podejście może prowadzić tutaj do sytuacji, w której pomimo że demontaż jest fizycznie możliwy, dochodzi się do stanu, z którego nie można uzyskać dalszego rozłożenia zespołu. Przykład takiej sytuacji mamy na rys. 3a, gdzie przedstawiono zespół składający się z dwóch cylindrów i dwóch tłoków.



Rys. 3 a - Przykładowy zespół montażowy, b - Graf połączeń zespołu
Fig. 3 a - Example assembly, b - Graph of assembly linkages

łatwo zauważyć, że w przypadku strategii demontażu polegającej na każdorazowym wymontowywaniu jednego detalu z zespołu, rozłożenie zespołu okaże się niemożliwe.

Poniżej prezentujemy ogólny algorytm analizujący możliwość demontażu dowolnego zespołu oraz wyznaczający ten demontaż.

Algorytm demontażu:

Krok 1 Wyznacz liczbę detali w zbiorze Z wszystkich detali sprawdzanego zespołu. Jeśli liczba ta jest równa 1, to zakończ z komunikatem o możliwości demontażu zespołu. W przeciwnym przypadku przejdź do Kroku 2.

Krok 2 Wybierz jeden detal d ze zbioru Z i utwórz zbiór $D = \{d\}$. Utwórz zbiór P wszystkich połączeń detalu d z pozostałymi detalami w zbiorze Z .

Krok 3 Wybierz ze zbioru P jedno połączenie p .

Krok 4 Sprawdź, czy można odłączyć od zespołu Z zbiór detali D , przerywając jedynie połączenie p . Jeśli odłączenie jest możliwe, to zakończ z komunikatem o możliwości demontażu oraz z oddzielnymi częściami zespołu D i $Z - D$ jako wynikami demontażu. W przeciwnym przypadku przejdź do kroku 5.

Krok 5 Utwórz zbiór NR połączeń, które nie ulegają przerwanemu jednocześnie z rozerwaniem połączenia p i dołącz do zbioru D jeden detal nie należący do D i związany z D takim połączeniem p . Jeśli $D \neq Z$, to powróć do kroku 4. W przeciwnym przypadku zakończ z komunikatem o niemożliwości demontażu.

Zastosowanie tego algorytmu do zespołu z rys. 3a przedstawia tabela 1, w której pokazano wyniki kolejnych kroków działania algorytmu. Rys. 3b obrazuje model zespołu z rys. 3a, w postaci grafu połączeń detali w zespole. Model ten stanowi dane wejściowe algorytmu demontażu.

Tabela 1.

Krok	d	P	p	D	NR	Z - D
1						
2	1	{a, d, e, f, g}		{1}		
3	1	{a, d, e, f, g}	a	{1}		
4	1	{a, d, e, f, g}	a	{1}		
5	1	{a, d, e, f, g}	a	{1}	{d, e}	
4	1	{a, d, e, f, g}	a	{1}	{d, e}	
5	1	{a, d, e, f, g}	a	{1, 2}	{h}	
4	1	{a, d, e, f, g}	a	{1, 2}	{h}	
5	1	{a, d, e, f, g}	a	{1, 2, 3}	{ }	{4, 5, 6, 7, 8}

4. Planowanie wykonania zadania

W ramach rozpatrywanej klasy zadań montażowych każdą z czynności (operacji) montażowych można rozłożyć na szereg podoperacji, które mogą być zaklasyfikowane jako:

- uchwycenie detalu (przedmiotu manipulacji), obejmujące odnalezienie właściwego miejsca i sposobu uchwycenia,
- manipulacja precyzyjna obejmująca ruchy niezbędne do wydostania detalu z jego położenia początkowego (podajnika, palety, itp.),
- przeniesienie obejmujące ruch (z reguły szybki i prostoliniowy) mający na celu przybliżenie detalu do miejsca jego przeznaczenia,
- manipulacja precyzyjna obejmująca faktyczny montaż, tj. wmontowywanie detalu w zespół,
- odseparowanie chwytaaka od detalu (puszczenie detalu) po umieszczeniu go w montowanym zespole.

Planowanie wykonania zadania polega na określeniu parametrów ruchu dla każdej z wymienionych mikrooperacji, a ponadto musi ono uwzględniać również wewnętrzne warunki wykonania każdej z mikrooperacji.

W sytuacji, gdy niemożliwe jest bezpośrednie wykonanie określonej czynności montażowej (całego cyklu jej mikrooperacji) zadaniem modułu planującego jest uzupełnienie sekwencji operacji o nowe czynności przygotowujące możliwość jej bezpośredniego wykonania. Przykładem może służyć tutaj sytuacja, w której przeznaczony do zamontowania detal jest przesłonięty lub zablokowany jakąś przeszkodą uniemożliwiającą bezpośrednio jego uchwycenie. Jeśli przeszkoda ta jest usuwalna, tj. istnieje możliwość przeniesienia jej na inne miejsce, to zadanie można wykonać po dodaniu do wygenerowanej wcześniej sekwencji operacji pewnej nowej sekwencji (również składającej się z zbioru mikrooperacji tworzących operację pick-and-place), mającej na celu odsłonięcie detalu przeznaczonego do zamontowania.

Pośród innych ważnych funkcji planowania wykonania zadania należy wymienić planowanie przydziału środków wykonania, co wiąże się z analizą możliwych sposobów wykonania montażu w ramach konkretnej implementacji gniazda wytwórczego (montażowego), oraz geometryczne planowanie trajektorii poszczególnych faz ruchu w ramach każdej z zaplanowanych operacji.

LITERATURA

- [1] Frommherz B., Hornberger J. : Automatic Generation of Precedence Graphs. ESPRIT N° 623. Working Paper No. IP-UKA-2.87/1, 1987.
- [2] Rembold U., Blume Ch., Dillmann R. : Computer - Integrated Manufacturing Technology and Systems. M. Dekker Inc. , New York, Basel. 1985.

- [3] Sanderson A., C. Homem - de - Mello L.S. : Task Planning and Control Synthesis for Flexible Assembly Systems. Proc. of Nato Advanced Research Workshop on Machine Intelligence and Knowledge Engineering for Robotic Applications. Edited by A.K.C.Wong and A. Pugh, Springer Nato ASI Series , vol. F33, 1987.
- [4] Smith R.M. : Computer - Aided Fully Generative Process Planning, Manuf. Eng. May 1981.
- [5] Warczyński J. : Założenia dla komputerowego systemu symulacji kinematyki i dynamiki robotów. Raport I.A. Politechniki Poznańskiej, Poznań. 1987.
- [6] Warczyński J. : Computer - Aided Programming Tool for Robotics. Proc. of 6-th IFAC Symp. Information Control Problems in Manufacturing Technology, Madrid, Sept. 26-29, 1989.

Recenzent: Prof.dr inż.H.Kowalowski

Wpłynęło do Redakcji do 1990-04-30.

IMPLICIT PROGRAMMING SYSTEM FOR ROBOTS

S u m m a r y

The paper presents an implicit programming system for robots aimed at programming of assembly processes. Special attention has been paid to the problem of formal determining the sequence constraints among the assembly operations. The developed system has been devoted to function as an interface between the computer-aided design- (CAD) system and the robotized assembly processes.

СИСТЕМА АВТОМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ РОБОТОВ

Р е з ю м е

В работе представлена идея системы автоматического программирования роботов для целей монтажа. В частности, особое внимание уделено проблеме формального определения ограничений по следованию монтажных операций. Разрабатываемая система предназначена для выполнения функций интерфейса между системой автоматического проектирования - САПР и автоматическим монтажом с помощью робота.