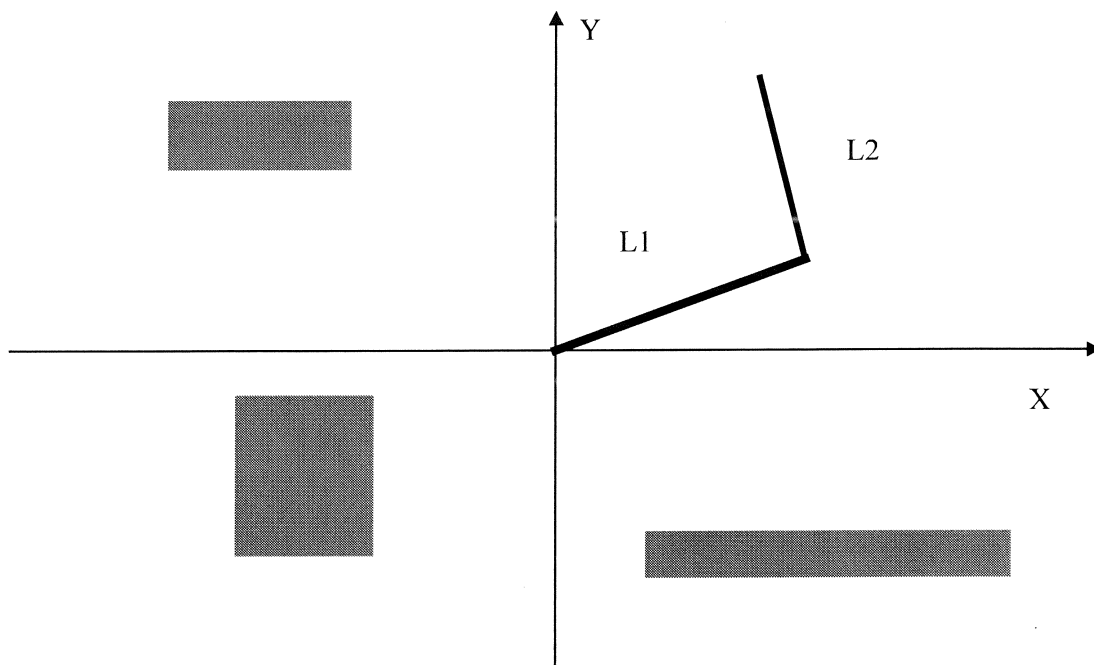


Programowanie Urządzeń Sterowanych Numerycznie
Zadanie nr 4 - Szukanie drogi dla płaskiego łańcucha kinematycznego
(2 grudnia 2016)

Termin oddania: 16 grudnia 2016 r.

Celem zadania jest implementacja prostego algorytmu szukania drogi dla dwuwymiarowego łańcucha kinematycznego. Wygląd schematyczny zadania przedstawia rysunek 1.



Rys. 1. Analizowany łańcuch kinematyczny złożony z dwóch ramion o długościach $L1$ i $L2$. Łańcuch jest płaski, tzn. obie osie obrotu są prostopadłe do płaszczyzny XY . Widoczne prostokąty to przeszkody umieszczone w przestrzeni roboczej robota.

Opis programu:

Program działa w dwóch trybach – edycji oraz szukania drogi. Po uruchomieniu programu działa on w trybie edycji przeszkód i łańcucha kinematycznego. Parametry robota ($L1$ i $L2$) oraz przeszkody są definiowane za pomocą myszki przez użytkownika w trakcie działania programu. Ponadto, kliknięcie myszką w dowolny punkt przestrzeni roboczej (x, y) powoduje ustawienie robota tak, że jego koniec (efektor) znajduje się w klikniętym punkcie (należy rozwiązać proste zadanie kinematyki odwrotnej). Pokazywane są oba możliwe rozwiązania, jedno lub żadne – jeśli położenie jest nieosiągalne (np. z powodu zasłonięcia przez przeszkodę).

Przeszkody na scenie i przestrzeń konfiguracji:

Należy zbudować tablicę współrzędnych wewnętrznych łańcucha – kąty α_1 i α_2 – tj. utworzyć dyskretyzację przestrzeni konfiguracji łańcucha. Zalecane jest przyjęcie podziału na 360 części czyli utworzenie tablicy 360×360 (uwaga na topologię przestrzeni konfiguracji!). Następnie dla każdej konfiguracji odpowiadającej wartościom indeksów w tej tablicy należy sprawdzić, czy łańcuch kinematyczny przecina się z którąś z przeszkód. Tablica będzie zatem zawierać odpowiednie wartości wskazujące, czy w dana konfiguracja jest osiągalna czy nie. Przestrzeń konfiguracji należy wizualizować tzn. należy przedstawić

w programie prostokąt, w którym każdy piksel odpowiada jednej komórce tablicy, a kolorami oznaczone są wartości.

W trybie edycji przeszkody mogą być dodawane, usuwane, zmieniane przez użytkownika. Nie trzeba wypisywać listy przeszkód, wystarczy ich wizualna reprezentacja. Nie ma także potrzeby, aby przestrzeń konfiguracji była obliczana za każdym uaktualnieniem przeszkód a tylko na żądanie użytkownika.

Metoda szukania drogi:

W trybie szukania drogi użytkownik wybiera konfigurację początkową i końcową – wybierając punkty (x, y) efektora w przestrzeni roboczej łańcucha. Następnie algorytmem flood-fill (4-spójny) znajdowana jest w przestrzeni konfiguracji droga pomiędzy tymi położeniami. Wynik obliczeń prezentowany jest użytkownikowi na 2 sposoby:

1. Animacja – łańcuch przesuający się z położenia początkowego do końcowego (animacja ma być płynna)
2. Łamana w przestrzeni konfiguracji oznaczająca drogę. Dodatkowo gradientem kolorów należy oznaczyć odległość każdego punktu od położenia początkowego znalezionej przez algorytm flood-fill podczas szukania drogi.