|  |  |  |  |
| --- | --- | --- | --- |
| **Numer ćwiczenia** | **-** | **Tytuł ćwiczenia:**  **Optymalizacja sterowania 2-osiowym robotem SCARA** | |
| **Data wykonania ćwiczenia:** | | 03-05.2022 | **Imię, nazwisko i numer albumu:** |
| **Data oddania sprawozdania:** | | 06.06.2022 | 1) Mateusz Głuch 305744 2) Kamil Głoski 3) Olgierd Duda |
| **Nr grupy lab./ /Kierunek:** | | **1a/ISS/AiR** |

#### **Spis treści**

##### [1. Cel ćwiczenia 2](#__RefHeading___Toc1340_3806494507)

##### [2. Wstęp 2](#__RefHeading___Toc2064_3806494507)

##### [3. Modelowanie systemu 2](#__RefHeading___Toc1342_3806494507)

##### [4. Kinematyka prosta i odwrotna 3](#__RefHeading___Toc1336_3806494507)

[4.1. Równania kinematyki prostej 3](#__RefHeading___Toc1344_3806494507)

[4.2. Równania kinematyki odwrotnej 3](#__RefHeading___Toc1388_3806494507)

##### [5. Sterowanie czasooptymalne 3](#__RefHeading___Toc1334_3806494507)

[5.1. Idea regulatora czasooptymalnego – założenia 3](#__RefHeading___Toc1378_3806494507)

[5.2. Funkcja celu 3](#__RefHeading___Toc1376_3806494507)

[5.3. Hamiltonian i równania sprzężone 4](#__RefHeading___Toc1374_3806494507)

[5.4. Sterowanie bang-bang i jego konsekwencje 4](#__RefHeading___Toc1372_3806494507)

[5.5. Metodyka rozwiązywania numerycznego zadania optymalizacji 5](#__RefHeading___Toc1370_3806494507)

[5.6. Algorytm porządkowania czasów przełączeń 5](#__RefHeading___Toc1368_3806494507)

[5.7. Wyniki i analiza rezultatów – dla różnych rozdzielczości rozwiązania 5](#__RefHeading___Toc1366_3806494507)

[5.8. Obszary osiągalne 5](#__RefHeading___Toc1364_3806494507)

##### [6. Sterowanie z optymalizacją kwadratowej funkcji celu 5](#__RefHeading___Toc1332_3806494507)

[6.1. Idea regulatora z kwadratową funkcją celu 5](#__RefHeading___Toc1362_3806494507)

[6.2. Funkcja celu 5](#__RefHeading___Toc1360_3806494507)

[6.3. Hamiltonian i równania sprzężone – modyfikacja 5](#__RefHeading___Toc1358_3806494507)

[6.4. Gradient hamiltonianu 5](#__RefHeading___Toc1356_3806494507)

[6.5. Metodyka rozwiązania równań różniczkowych 5](#__RefHeading___Toc1354_3806494507)

[6.6. Analiza rozwiązania zadania w zależności od wag f-cji celu 5](#__RefHeading___Toc1352_3806494507)

[6.7. Analiza rozwiązania zadania w zależności od zadanego czasu symulacji 5](#__RefHeading___Toc1350_3806494507)

[6.8. Analiza rozwiązania zadania w zależności od rozdzielczości solvera 5](#__RefHeading___Toc1348_3806494507)

[6.9. Wyniki dla różnych trajektorii dla zadania podążania za trajektorią 5](#__RefHeading___Toc1346_3806494507)

##### [7. Sterowanie predykcyjne – Model Predictive Control 5](#__RefHeading___Toc1330_3806494507)

[7.1. Idea regulacji predykcyjnej o skończonym horyzoncie 5](#__RefHeading___Toc1386_3806494507)

[7.2. Warianty regulacji predykcyjnej 5](#__RefHeading___Toc1384_3806494507)

[7.3. Implementacja MPC ze sterowaniem optymalnym 5](#__RefHeading___Toc1382_3806494507)

[7.4. Analiza jakościowa regulacji MPC – horyzont i krok predykcji 5](#__RefHeading___Toc1380_3806494507)

##### [8. Wnioski 5](#__RefHeading___Toc1328_3806494507)

#### Cel ćwiczenia

Celem ćwiczenia jest zaprojektowanie oprogramowania pozwalającego na wyznaczenie sterowania optymalnego (tj. minimalizującego zadaną funkcję celu) dla modelu matematycznego 2-osiowego robota SCARA. Regulator ten został rozpatrzony w wariancie czasooptymalnym oraz w wariancie z kwadratową funkcją celu. Następnie został zaprojektowany regulator Model Predictive Control, pozwalający zarówno na skrócenie czasu obliczeń poprzez zastosowanie skończonego horyzontu, ale także na zastosowanie sprzężenia zwrotnego od stanu obiektu.

#### Wstęp

#### Modelowanie systemu

Robot 2-osiowy SCARA został zamodelowany jako obiekt nieliniowy z dwoma sterowaniami (ze względu na ruchliwość mechanizmu robota równą 2). Przy modelowaniu uwzględniono zarówno niezerową masę członów robota, jak i nieliniowy model tarcia.

Równanie ruchu manipulatora ma postać:

Oznaczenia:

Macierz odwrotna do macierzy momentów bezwładności została oznaczona symbolem i jest równa:

W modelu manipulatora został zastosowany przybliżony, ciągły model tarcia:

Poprzez oznaczenie pozostałych zmiennych stanu jako , można zapisać równania stanu manipulatora w postaci jednorodnej względem sterowania:

Gdzie:

Parametry manipulatora ustawiono zgodnie z poniższą tabelą:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Numer członu | I [kg\*m2] | m[kg] | d[m] | xci[m] | yci[m] | si[Nm] | f**i**[W-1] | umin[Nm] | umax[Nm] |
| 1 | 1.00 | 10.0 | 1.00 | 0.50 | 0.00 | 0.10 | 0.01 | -1.00 | 1.00 |
| 2 | 1.00 | 5.0 | 0.70 | 0.35 | 0.00 | 0.10 | 0.01 | -1.00 | 1.00 |

#### Kinematyka prosta i odwrotna

##### Równania kinematyki prostej

Robot SCARA porusza się po przestrzeni roboczej odnosząc się do położeń swoich przegubów, wyrażonych jako odpowiednie kąty obrotu (Rys. 4.1). Aby możliwe było wyznaczenie położenia robota w przestrzeni kartezjańskiej , konieczne jest rozwiązanie tzw. zadania prostego kinematyki robota, czyli przekształcenia współrzędnych kątowych na kartezjańskie. Dla 2-osiowego płaskiego robota o konfiguracji SCARA równania kinematyki prostej przedstawiają się następująco (poprzez macierze rotacji):

Oznaczenia:

- położenie kątowe przegubu 1

- położenie kątowe przegubu 2

Macierz rotacji przegubu 1:

Macierz rotacji przegubu 2:

Równania kinematyki prostej:

##### Równania kinematyki odwrotnej

Aby możliwe było przekształcenie zadanej trajektorii w postaci parametrycznej , konieczne jest rozwiązanie tzw. zadania odwrotnego kinematyki, tj. przekształcenie współrzędnych kartezjańskich na położenia kątowe odpowiednich przegubów robota. Dodatkowym problemem jest fakt, że o ile równania kinematyki prostej robota mają zawsze dokładnie jedno rozwiązanie (dla każdego położenia kątowego robota istnieje dokłanie jeden punkt i orientacja, w której robot się znajduje), o tyle równania kinematyki odwrotnej na ogół mają kilka rozwiązań – konfiguracji robota. W tym przypadku mamy do czynienia z dwoma konfiguracjami – „above” i „below” (zob. rys. 4.2). Równania kinematyki odwrotnej przedstawione są poniżej:

Gdzie zmienne wyrażone są za pomocą:

#### Sterowanie czasooptymalne

##### Idea regulatora czasooptymalnego – założenia

##### Funkcja celu

Funkcja celu w algorytmie czasooptymalnym została dobrana w ten sposób, aby gwarantować zarówno optymalizację pod kątem czasu sterowania, ale także pod kątem osiągnięcia założonego celu; tj. stan w chwili końcowej symulacji musi odpowiadać stanowi referencyjnemu. Funkcja celu przedstawia się w następujący sposób:

Gdzie:

W – macierz wag związana ze stanem układu ()

T – czas symulacji

x(T) – stan obiektu w chwili końcowej

xf – założony stan obiektu w chwili końcowej

##### Hamiltonian i równania sprzężone

Hamiltonian dla tego przypadku jest wyrażony wzorem:

Gdzie funkcje f, g1 i g2 są wyrażone jak w punkcie 3.

Równania sprzężone mają postać:

Gdzie pochodna hamiltonianu względem stanu obiektu przybiera postać:

Gdzie poszczególne współczynniki macierzy:

##### Sterowanie bang-bang i jego konsekwencje

W związku z tym, że celem sterowania jest osiągnięcie minimalnego czasu sterowania układu, zasadnym jest założenie, że sterowanie optymalne będzie przybierało wartości z dyskretnego zbioru (zob. tw. Feldbauma):

Oznacza to, że mimo osiągnięcia przez hamiltonian swojego maksimum (zgodnie z zasadą maksimum Pontriagina), pochodna hamiltonianu względem sterowań nie będzie równa 0 – przyjmie ona następujące wartości:

Ze względu na powyższy fakt, zasadnym jest nazwanie funkcji funkcją przełączającą, ponieważ dla sterowania optymalnego zachodzą poniższe zależności (rzecz jasna, **jedynie dla pewnego szczególnego przypadku, gdy** , natomiast jest to wygodny sposób na przedstawienie funkcji przełączającej – poprzez funkcję signum):

Jawnie, dla tego problemu funkcja przełączająca wyrazi się wzorami:

##### Metodyka rozwiązywania numerycznego zadania optymalizacji

Zadanie optymalizacji jest rozwiązywane numerycznie – ze względu na to, że trajektorię stanu złożonego, nieliniowego modelu można w rozsądny sposób otrzymać jedynie jako rezultat zastosowania metod numerycznych.

Do rozwiązania równań stanu, jak i równań różniczkowych zastosowano metodę Rungego-Kutty 4-go rzędu, opisaną w następujący sposób: jeśli równanie różniczkowe ma postać oraz równanie różniczkowe rozwiązywane jest z krokiem h, to kolejne wartości zmiennych stanu równania różniczkowego wyrażają się poprzez równania:

Gdzie k1, k2, k3, k4 są wyrażone poprzez:

Do przeprowadzenia odpowiedniego algorytmu optymalizacyjnego konieczne jest także zdefiniowanie odpowiednich gradientów ze względu na czasy przełączeń . Gradient ten wyraża się poprzez:

Algorytm obliczania kolejnych sterowań odbywa się wg następującego schematu:

* rozwiązanie równań stanu dla obecnego sterowania ( ),
* obliczenie końcowych wartości zmiennych sprzężonych ( ),
* rozwiązanie równań sprzężonych (uwaga: równania sprzężone rozwiązuje się z czasem do tyłu),
* obliczenie gradientów hamiltonianu
* wyznaczenie nowych czasów przełączeń – wykonywane przez funkcję *fmincon* w Matlabie, bądź wyznaczane za pomocą metody gradientu prostego:

Kryterium zatrzymania algorytmu jest albo domyślnie ustalone w implementacji funkcji *fmincon*, albo jest nim maksymalna ilość iteracji wykonana przez algorytm gradientu prostego.

##### Wyniki i analiza rezultatów – dla różnych rozdzielczości rozwiązania

##### Obszary osiągalne

#### Sterowanie z optymalizacją kwadratowej funkcji celu

##### Idea regulatora z kwadratową funkcją celu – założenia

##### Funkcja celu

Funkcja celu w tym wariancie problemu została dobrana tak, żeby gwarantować Funkcja celu w tym problemie jest zadana wzorem:

##### Hamiltonian i równania sprzężone – modyfikacja

Hamiltonian układu, w przypadku wystąpienia w funkcji celu składnika całkowego, przyjmie w tym problemie postać zadaną wzorem poniżej – wyrażenie podcałkowe staje się składnikiem hamiltonianu:

Równania sprzężone mają postać:

Gdzie macierz A jest zdefiniowana jak w zadaniu czasooptymalnym, natomiast dodatkowy składnik jest równy:

Funkcja przełączająca (której nie można już nazywać funkcją przełączającą, ze względu na to, że rozwiązaniem zadania optymalizacji będzie zazwyczaj *sterowanie osobliwe*), dla tego przypadku przybierze postać:

##### Metodyka rozwiązania równań różniczkowych

Do rozwiązania zadania optymalizacji konieczne jest zastosowanie pewnej aproksymacji sterowania – w tym zadaniu skorzystano z faktu, że obliczenia są wykonywane z dość dużą dokładnością dla metody Rungego-Kutty 4 rzędu już dla dużych kroków (0.1s) i uznano, że do skutecznej optymalizacji wystarczy wyznaczyć sterowanie dla każdej chwili czasowej z rozdzielczością równą rozdzielczości symulacji.

Aby w efektywny sposób rozwiązać zadanie optymalizacji, konieczne jest także obliczenie całki znajdującej się w definicji funkcji celu. Jest to osiągnięte poprzez dołożenie dodatkowego równania do równań stanu:

Rozwiązanie tak zdefiniowanych równań stanu sprawi, że oprócz odpowiedzi układu na zadane sterowanie, zostanie także otrzymana aproksymacja całki występującej w def. funkcji celu.

Do skutecznego rozwiązania zadania optymalizacji konieczne jest także zdefiniowanie odpowiednich gradientów hamiltonianu:

W tym przypadku również konieczne jest obliczenie pewnej całki, w definicji równań sprzężonych zostały dołączone dwa dodatkowe równania:

Pozostała część obliczeń wykonywana jest w taki sam sposób, co w przypadku regulatora czasooptymalnego.

##### Analiza rozwiązania zadania w zależności od wag f-cji celu

##### Analiza rozwiązania zadania w zależności od zadanego czasu symulacji

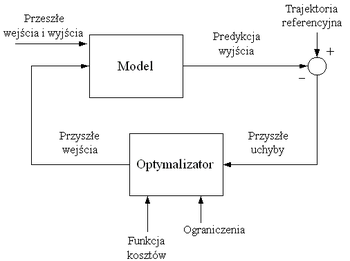
##### Analiza rozwiązania zadania w zależności od rozdzielczości solvera

##### Wyniki dla różnych trajektorii dla zadania podążania za trajektorią

#### Sterowanie predykcyjne – Model Predictive Control

##### Idea regulacji predykcyjnej o skończonym horyzoncie

Regulator *Model Predictive Control*, w przeciwieństwie do tradycyjnych regulatorów ze sprężeniem zwrotnym, dostosowuje swoje działanie z pewnym wyprzedzeniem, zanim nastąpią stosowne zmiany w wielkościach mierzonych (obserwowanych) układu. Metoda ta polega na cyklicznym rozwiązywaniu zadania optymalizacji z warunkiem początkowym równym zaobserwowanemu stanowi obiektu.

  
Rys 7.1 Schemat działania regulatora *Model Prediictive Control*

Algorytm działania regulatora Model Predictive Control opiera się na iteracyjnym wykonaniu następującego algorytmu:

* rozwiązanie zadania optymalizacji przy zadanym *horyzoncie* optymalizacji,
* zastosowanie na wejście obiektu części obliczonego, optymalnego sterowania – w wesji dyskretnej MPC jest to jedna wartość, natomiast w tym problemie zaimplementowano zastosowanie na wejście obiektu określonej długości sygnału sterowania (*step*),
* przesunięcie horyzontu o wartość kroku algorytmu (*step),*
* Jeśli został osiągnięty zadany czas regulacji, algorytm automatycznie skraca horyzont predykcji, natomiast w przypadku, gdy horyzont predykcji osiąga wartość 0 – algorytm kończy swoje działanie.

Ze względu na skończony horyzont predykcji, wymiarowość zadania optymalizacji zmniejsza się, dzięki czemu, w przeciwieństwie do algorytmów omawianych w poprzednich punktach, algorytm ten pozwala na obliczanie, przy odpowiednio zaawansowanym kontrolerze sterującym, kolejnych sterowań w czasie rzeczywistym.

##### Implementacja MPC ze sterowaniem optymalnym

##### Analiza jakościowa regulacji MPC – horyzont i krok predykcji

#### Wnioski