****

AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,

INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ

KATEDRA AUTOMATYKI I INŻYNIERII BIOMEDYCZNEJ

Optymalizacja w systemach sterowania

*Stabilizacja położenia sześciennej kostki „Cubli”*

Autorzy: Jakub Wąsik, Michał Jasiński

Kierunek studiów: Automatyka i Robotyka, studia magisterskie I stopnia

Prowadzący: Dr hab. inż. Adam Korytowski

Kraków 2017

Spis treści

[Wstęp 4](#_Toc485053681)

[1. Opis badanego obiektu 4](#_Toc485053682)

[2. Model matematyczny 5](#_Toc485053683)

[2.1. Oznaczenia 5](#_Toc485053684)

[2.2. Opis matematyczny modelu 7](#_Toc485053685)

[3. Model symulacyjny 7](#_Toc485053686)

[4. Optymalizacja dynamiczna 7](#_Toc485053687)

[4.1. Sformułowanie problemu 7](#_Toc485053688)

[4.2. Wskaźnik jakości 7](#_Toc485053689)

[4.3. Funkcje sprzężone oraz Hamiltonian 8](#_Toc485053690)

[4.4. Testy funkcji sprzężonych 8](#_Toc485053691)

[5. Algorytm BFGS 8](#_Toc485053692)

[5.1. Informacje wstępne 8](#_Toc485053693)

[5.2. Gradient wskaźnika jakości 8](#_Toc485053694)

[5.3. Opis algorytmu 8](#_Toc485053695)

[6. Testy aplikacji 8](#_Toc485053696)

[6.1. Optymalizacja kodu 9](#_Toc485053697)

[6.2. Testy działania 9](#_Toc485053698)

[6.3. Wnioski 9](#_Toc485053699)

[Podsumowanie 9](#_Toc485053700)

[Dodatki 9](#_Toc485053701)

[Dodatek A: Struktura programu 9](#_Toc485053702)

[Dodatek B: Rozwiązywanie równań różniczkowych metodą RK4 w MATLAB-ie 9](#_Toc485053703)

[Dodatek C: Algorytm BFGS 10](#_Toc485053704)

[Literatura 10](#_Toc485053705)

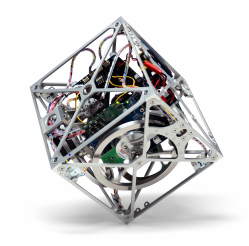
# Wstęp

Celem projektu jest znalezienie sterowania optymalnego dla problemu stabilizacji sześciennej kostki Cubli z wykorzystaniem optymalizacji dynamicznej. Cubli jest projektem realizowanym od 2011 roku na Instytucie Sterowania i Systemów Dynamicznych Politechniki Federalnej ETHZ w Zurychu przez Michaela Muehlebacha i prof. Raffaello D’Andrea [1]. Jedną z wielu możliwości tej kostki o wymiarach 15x15x15 cm jest balansowanie na jednym z jej rogów. Ponadto kostka ta potrafi także podskakiwać, chodzić, itd. Zdecydowano, że sterowanie optymalne będzie szukane dla pierwszego problemu, czyli dla balansowania na jednym z rogów kostki.

W pracy zostanie przedstawiony obiekt i jego właściwości. Zaprezentowane będą także równania matematyczne systemu. Następnie omówione będą zagadnienia dotyczące poszukiwania sterowania optymalnego z wykorzystaniem optymalizacji dynamicznej, wykorzystanego gradientowego algorytmu BFGS oraz zaprezentowane będą testy i wyniki.

# Opis badanego obiektu

*Cubli*, czyli kostka o wymiarach 15x15x15 cm, która potrafi balansować na jednym z jej rogów, bazuje na efekcie trójwymiarowego odwróconego wahadła. Kostka posiada trzy koła reakcyjne po jednym na każdą z osi, które obracając się z zadaną prędkością kątową wytwarzają moment pędu, któremu przeciwstawia się kostka, aby zachować zasadę zachowania pędu. Efektem tego jest to, że kostka potrafi stać stabilnie na jednym z jej rogów, oraz reagować na delikatne próby wytrącenia jej z równowagi. Wykonana kostka *Cubli* jest zaprezentowana na rysunku 1.1.



Rys. 1.1 – Sześcienna kostka Cubli balansująca na jednym z jej rogów

*Cubli* jest w stanie przeciwstawić się tylko niewielkim siłą próbującym wytrącić ją z położenia równowagi, ponieważ moment pędu wytworzony przez koła reakcyjne rozpędzające się do maksymalnej prędkości jest ograniczony. Można także rozważyć sytuację, w której kostka w początkowym momencie jest odchylona o pewien kąt od pozycji zadanej (zerowej), w której będzie stabilizowana. Zadaniem kół reakcyjnych jest doprowadzenie kostki do położenia zadanego. Właśnie tego typu problem, czyli doprowadzenie kostki do punktu równowagi i stabilizacja w tym punkcie (na jednym z jej rogów) będzie rozważany w tym projekcie. Założenie jest takie, że kąt o jaki będzie odchylona kostka będzie dostatecznie mały tak, aby koła reakcyjne rozpędzające się do maksymalnej prędkości były w stanie doprowadzić tą kostkę do położenia równowagi.

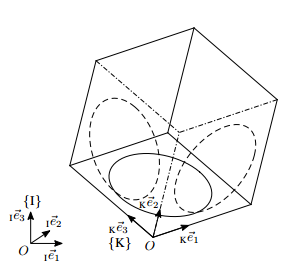
Jak wspomniano wcześniej, kostka ta potrafi także podskoczyć z leżącej. Jak łatwo się domyślić, do podniesienia kostki wymagana jest bardzo duża siła. Na tyle duża, że rozpędzenie kół reakcyjnych do maksymalnej prędkości byłoby niewystarczające. Tutaj jednak projektanci zastosowali inną technikę uzyskiwania dużego momentu. Rozpędzają oni odpowiednie koła reakcyjne do maksymalnej prędkości i następnie zatrzymują gwałtownie te koła z wykorzystaniem blokad mechanicznych. Tego typu sytuacja nie będzie przedmiotem rozważań w ramach tego projektu.

# Model matematyczny

W tej sekcji opisany będzie model matematyczny trójwymiarowego odwróconego wahadła. Po przedstawieniu oznaczeń użytych w modelu, opisane będą równania ruchu opisane w pracy [2] oraz [3].

## Oznaczenia

Aby można było rozsądnie zdefiniować ruch kostki, należy rozróżnić dwa układy współrzędnych: układ współrzędnych odniesienia oraz układ współrzędnych kostki . Podane układy współrzędnych zostały zaprezentowane na rysunku 2.1.



Rys. 2.1. – Układy współrzędnych przyjęte w modelu matematycznym

Na powyższym rysunku układ współrzędnych odniesienia jest przesunięty w lewo w stosunku do układu współrzędnych kostki. To przesunięcie jest zastosowane tylko dla celów ilustracji, gdyż punkt podparcia O jest wspólny dla tych obu układów współrzędnych. Różnica między tymi układami jest taka, że układ jest nieruchomy, natomiast jest związany z obracającą się kostką. Wektory K oraz I , gdzie , oznaczają i-ty wersor układu współrzędnych oraz .

Należy także oznaczyć momenty bezwładności występujące w systemie:

* - tensor bezwładności ramki sześciennej kostki
* - moment bezwładności koła reakcyjnego

Ponadto moment bezwładności całego systemu wokół punktu podparcia O jest równy . Warto zdefiniować także macierz antysymetryczną odpowiadającą wektorowi oznaczoną jako . Macierz ta spełnia równanie:

(2.1)

gdzie , natomiast .

Pozostałe oznaczenia:

* - prędkość kątowa sześciennej kostki w odniesieniu do układu odniesienia
* - prędkości kątowe kół reakcyjnych
* - moment pędu sześciennej kostki,
* - moment pędu wytworzony przez koła reakcyjne
* - wektor pozycji z punktu podparcia do środka ciężkości pomnożony przez całkowita masę
* - wektor grawitacji
* - macierz transformacji, która przekształca wektor wyrażony w układzie współrzędnych kostki na wektor z układu współrzędnych odniesienia .
* - moment obrotowy silników napędzających koła reakcyjne

## Opis matematyczny modelu

Równania matematyczne modelu prezentują się następująco:

(2.2)

Ponadto i są z definicji równe:

(2.3)

Z zależności opisanych w układzie równań (2.3) można wyznaczyć użyte w równaniach (2.2) jako zależność i .

# Model symulacyjny

Dzięki uprzejmości i udostępnieniu modelu symulacyjnego przez Michael Muehlebach i prof. Raffaello D’Andrea z Politechniki Federalnej ETHZ w Zurychu. Został on zaimplementowany w Simulinku. Model potraktowano jako wzorcowy i przerobiono go tak, aby wszystkie operacje, włącznie z rozwiązywaniem równań różniczkowych, były wykonywane w Matlabie

Jako solver równań różniczkowych użyto metodę Rungego-Kutty rzędu 4. Równanie postaci z wartością początkową i okresem próbkowania h można numerycznie rozwiązać wykorzystując poniższe wzory wspomnianej metody:

gdzie:

W głównym skrypcie *main.m* odpowiadał za wywołanie procedur oraz funkcji. W celu zdefiniowania wszystkich potrzebnych parametrów wywołano procedurę *init.m.* Za wyliczenie optymalnego sterowania metodą BFGS odpowiada funkcja *bfgs()* zwracająca optymalne sterowanie. W funkcji *rk4()* zawarto wspomiany powyżej solver równań różniczkowych, który wyliczał również równania sprzężone oraz gradient. W tej funkcji obliczana była prawa strona równania różniczkowego w funkcji *rhs()*, która zawierała model obiektu. Pozostała część skryptu głównego odpowiada za zebranie obliczeń oraz wyświetlenie przebiegów.

W celu przetestowania stworzonego modelu, przeprowadzono symulację dla układu bez sterowania, odchylonego od położenia równowagi, a następnie porównano z modelem wzorcowym.



Rys 3.1. Odpowiedź układu modelu referencyjnego bez sterowania



Rys 3.2. Odpowiedź stworzonego modelu bez sterowania

Analizując powyższe wykresy, są one niemal identyczne, co dowodzi poprawności implementacji modelu symulacyjnego.

# Optymalizacja dynamiczna

W tym rozdziale zostanie przedstawiona optymalizacja dynamiczna, której celem będzie znalezienie sterowania optymalnego dla problemu stabilizacji położenia sześciennej kostki „Cubli”.

## Sformułowanie problemu

Problem sterowania optymalnego, w którym dynamika systemu jest opisana równaniem różniczkowym stanu

(4.1)

gdzie: ,

wskaźnik jakości natomiast przyjmuje postać

(4.2)

Dla takiego problemu można zdefiniować równania sprzężone (4.3) oraz Hamiltonian (4.4):

(4.3)

W zapisie macierzowym jest to równoważne:

Macierz pochodnych funkcji nazywana jest transponowaną macierzą Jacobiego.

(4.4)

Mając zdefiniowany Hamiltonian, zasadę maksimum Pontriagina można określić następująco:

Załóżmy, że jest sterowaniem optymalnym (czyli dopuszczalnym i takim dla którego   
 ), wygenerowaną przez nie optymalną trajektorią stanu, a - odpowiednią trajektorią sprzężoną. Przy tych założeniach dla wszystkich zachodzi warunek   
, to znaczy hamiltonian jest maksymalizowany przez sterowanie optymalne.

## Wskaźnik jakości

Zadaniem stawianym układowi regulacji jest stabilizacja położenia sześciennej kostki w niestabilnym punkcie równowagi. Ze względu na to, że położenie kątowe kostki nie jest zmienną stanu, należało je wyliczyć na podstawie pozostałych zmiennych, jak to też zostało zrobione w modelu wzorcowym. Jednak w podanych wzorach występowała funkcja arccos, która ma ograniczony zbiór wartości [-pi/2,pi/2]. Dlatego też, zamiast kąta wychylenia, we wskaźniku jakości uwzględniono cosinus tego kąta.

Zdecydowano się na przyjęcie całkowego wskaźnika jakości postaci:

(4.2)

Gdzie:

Q=diag(0.5,0.5,0.5),

-kąty nachylenia kostki odpowiednio w płaszczyznach x, y oraz z,

- prędkość obrotowa kół reakcyjnych.

## Funkcje sprzężone oraz Hamiltonian

Ze względu na skomplikowaną postać macierzy Jacobiego, zostanie ona przedstawiona w dodatku D. Macierz ta jest wymiaru 14x14 i do jej obliczenia posłużono się przybornikiem obliczeń symbolicznych z programu Matlab.

## Testy funkcji sprzężonych

W celu sprawdzenia poprawności działania i implementacji równań sprzężonych przeprowadzono testy, w których porównano pochodną wskaźnika jakości względem stanu początkowego, wyliczoną dwoma sposobami.

Gradient wskaźnika jakości można przybliżyć za pomocą ilorazów różnicowych:

(4.5)

Porównawcze wartości dla pochodnej wskaźnika jakości względem stanu początkowego przedstawiono na rys 4.4.1. Pierwsza kolumna to wektor gradientu obliczony jako równanie 4.4, natomiast druga - 4.5. Ostatnia kolumna to różnica pomiędzy odpowiednimi wierszami w tabeli.



Rys.4.4.1. Gradient wskaźnika jakości względem stanu obliczony dwoma sposobami.

Z rys 4.4.1 wynika, że wartości wektora obliczonego z równania 4.4 zgadzają się z tymi obliczonymi sposobem 4.5. Wartości wynikowe pokrywają się na co najmniej 3 miejscach znaczących.

# Algorytm BFGS

W tej sekcji opisany będzie algorytm BFGS.

## Informacje wstępne

Informacje wstępne

## Gradient wskaźnika jakości

Gradient

## Opis algorytmu

Opis algorytmu

# Testy aplikacji

Testy aplikacji

## Optymalizacja kodu

Optymalizacja kodu

## Testy działania

Dla wszystkich eksperymentów kąty zostały wyrażone w radianach, jednak dla lepszej wizualizacji na wykresach postanowiono przejść na stopnie. Na wykresach przedstawiono wartość wskaźnika jakości, kąt nachylenia kostki w trzech płaszczyznach, prędkość kątową kostki oraz prędkość obrotową kół reakcyjnych.

Założenia dla pierwszego eksperymentu:

* Czas symulacji: 1s
* Liczba węzłów strukturalnych: 5
* Warunki początkowe na kąt wychylenia sześciennej kostki w płaszczyznach x,y,z:
* Ograniczenia na sterowanie: Lb=-2 [Nm] Ub=2 [Nm]
* Maksymalna ilość iteracji algorytmu: 400



Rys. 6.2.1. Przebiegi czasowe dla pierwszego eksperymentu

Założenia dla drugiego eksperymentu:

* Czas symulacji: 1.5 s
* Liczba węzłów strukturalnych: 5
* Warunki początkowe na kąt wychylenia sześciennej kostki w płaszczyznach x,y,z:
* Ograniczenia na sterowanie: Lb=-1 [Nm] Ub=1 [Nm]
* Maksymalna ilość iteracji algorytmu: 200



Rys. 6.2.2. Przebiegi czasowe dla drugiego eksperymentu

Założenia dla trzeciego eksperymentu:

* Czas symulacji: 1.5 s
* Liczba węzłów strukturalnych: 10
* Warunki początkowe na kąt wychylenia sześciennej kostki w płaszczyznach x,y,z:
* Ograniczenia na sterowanie: Lb=-1 [Nm] Ub=1 [Nm]
* Maksymalna ilość iteracji algorytmu: 200



Rys. 6.2.3. Przebiegi czasowe dla trzeciego eksperymentu

## Wnioski

Wnioski

# Podsumowanie

Wnioski

# Dodatki

Wnioski

## Dodatek A: Struktura programu

Dodatek A

## Dodatek B: Rozwiązywanie równań różniczkowych metodą RK4 w MATLAB-ie

Dodatek B

## Dodatek C: Algorytm BFGS

Dodatek C

# Literatura

|  |  |
| --- | --- |
| [1] | ETH Zurich Institute for Dynamic Systems and Control, „Cubli,” 2017. [Online]. Available: http://www.idsc.ethz.ch/research-dandrea/research-projects/cubli.html. |
| [2] | M. Muehlebach i R. D'Andrea, „Nonlinear Analysis and control of a Reaction-Wheel-Based 3-D Inverted Pendulum,” *IEEE Transactions on Control Systems Technology,* 2016. |
| [3] | M. Gajamohan, M. Muehlebach, T. Widmer i R. D'Andrea, „The Cubli: A reaction wheel based 3D inverted pendulum,” w *European Control Conference*, 2013. |