

Regulatory niecałkowitego rzędu w manipulatorach

D. Ostaszewicz ; M. Szarało

I. STRESZCZENIE

W artykule przedstawiono zastosowanie rachunku różniczkowego niecałkowitego rzędu w projektowaniu układów sterowania manipulatorami o sześciu stopniach swobody. Omówiono teoretyczne podstawy tego podejścia oraz narzędzia wykorzystywane do modelowania i symulacji, takie jak MATLAB, Simulink, Simscape, FOMCON, Robotics Toolbox i ROS2. Szczególną uwagę poświęcono regulatorowi niecałkowitego rzędu. Wyniki badań wskazują na potencjał rachunku niecałkowitego rzędu w zaawansowanych zastosowaniach robotycznych.

II. WSTĘP

RACHUNEK niecałkowitego rzędu, rozwijany na przestrzeni ostatnich dekad, zyskuje na znaczeniu w obszarze inżynierii sterowania. Dzięki swojej elastyczności i możliwości modelowania systemów z pamięcią, stanowi doskonałe narzędzie do opisu i projektowania układów o złożonej dynamice. W porównaniu z klasycznym rachunkiem różniczkowym, oferuje on możliwość dostosowania charakterystyki dynamicznej układów poprzez precyzyjną manipulację rzędami pochodnych i całek, co otwiera nowe kierunki w optymalizacji i analizie systemów sterowania. Podstawy rachunku i jego zastosowania w elektrotechnice, automatyce i teorii sterowania są szeroko opisane, przez specjalistów w dziedzinie [11]–[14].

Praktyczne zastosowania układów niecałkowitego rzędu obejmują szeroki wachlarz dziedzin, takich jak sterowanie procesami przemysłowymi, modelowanie dynamiczne w biologii czy inżynieria mechaniczna [15]–[18]. W kontekście układów sterowania, rozwiązania te pozwalają na zwiększenie dokładności regulacji, redukcję przeregulowań i poprawę tłumienia zakłóceń.

Celem niniejszego artykułu jest analiza zastosowania rachunku niecałkowitego rzędu w projektowaniu układów sterowania manipulatora o sześciu stopniach swobody. W pracy przedstawiono metody oraz narzędzia użyte w procesie modelowania i symulacji. Szczególny nacisk położono na wykorzy-

stanie oprogramowania MATLAB i Simulink, które umożliwiły przeprowadzenie symulacji oraz obliczeń numerycznych.

Biblioteka Simscape, będąca częścią środowiska Simulink, zapewniła możliwość szybkiego i efektywnego tworzenia modeli elektrycznych, mechanicznych oraz mechatronicznych systemów wielodzielnicowych. Do modelowania i analizy układów niecałkowitego rzędu wykorzystano bibliotekę FOMCON, która oferuje zaawansowane narzędzia do projektowania i symulacji takich systemów.

W kontekście robotyki zastosowano Robotics Toolbox, który umożliwił przeprowadzanie operacji na modelach manipulatora, w tym analizę kinematyki i dynamiki. Dodatkowo, wykorzystano system ROS2 (Robot Operating System 2), który pozwala na modułowe projektowanie i prototypowanie systemów robotycznych. Komunikację na odległość z systemem ROS2 zrealizowano za pomocą Hamachi, zapewniającego bezpieczne połączenie przez VPN.

Integracja symulacji z ROS2 została osiągnięta dzięki ROS Toolbox, który umożliwił przesyłanie danych między Simulinkiem a ROS2, a także stworzenie zdalnej wizualizacji ruchu manipulatora. Przedstawione narzędzia i metody stanowią kompleksowe podejście do projektowania i analizy zaawansowanych układów sterowania manipulatorów, podkreślając zalety wykorzystania rachunku niecałkowitego rzędu.

III. METODY

A. Matlab i Simulink

MATLAB to zaawansowane środowisko obliczeniowe, które umożliwia przeprowadzanie analiz numerycznych, przetwarzanie danych, modelowanie matematyczne oraz wizualizację wyników. Jego wszechstronność i szeroka gama funkcji sprawiają, że jest wykorzystywany w wielu dziedzinach, takich jak inżynieria, nauki ścisłe czy finanse. MATLAB wspiera użytkownika poprzez gotowe biblioteki i narzędzia, które upraszczają implementację zaawansowanych algorytmów i umożliwiają szybkie prototypowanie rozwiązań. [1]

Simulink to rozszerzenie MATLAB-a dedykowane modelowaniu, symulacji i analizie systemów dynamicznych. Dzięki

graficznemu interfejsowi użytkownika umożliwia intuicyjne tworzenie modeli za pomocą bloków reprezentujących elementy systemu, co znacząco przyspiesza proces projektowania. Simulink jest szczególnie ceniony w automatyce, robotyce oraz przemyśle motoryzacyjnym, gdzie znajduje zastosowanie w projektowaniu i testowaniu systemów sterowania, przetwarzaniu sygnałów czy analizie systemów mechatronicznych. [2]

B. FOPID

Regulator FOPID (ang. Fractional-Order Proportional-Integral-Derivative) jest rozszerzeniem klasycznego regulatora PID, który wykorzystuje pojęcia rachunku różniczkowego i całkowego niecałkowitego rzędu. W przeciwieństwie do standardowego PID, gdzie działania różniczkowania i całkowania są określone jako operatory pierwszego i zerowego rzędu, w FOPID ich rzędy są dowolnymi liczbami rzeczywistymi, oznaczonymi jako λ (rząd całkowania) oraz μ (rząd różniczkowania) (1).

$$G(s) = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu \quad (1)$$

Przedstawiane badania wykorzystują bibliotekę FOMCON, opisaną w sekcji E.

C. Robotics Toolbox

Robotics Toolbox to biblioteka dostępna dla MATLAB-a, zaprojektowana z myślą o analizie, symulacji i projektowaniu systemów robotycznych. Narzędzie to dostarcza zestawu funkcji i obiektów wspomagających modelowanie manipulatorów, pojazdów mobilnych oraz systemów autonomicznych, umożliwiając kompleksowe badania nad różnymi aspektami robotyki. Główne funkcjonalności Robotics Toolbox obejmują:

- modelowanie manipulatorów – definiowanie struktur kinematycznych i dynamicznych przy użyciu reprezentacji Denavita-Hartenberga (DH) lub standardu URDF,
- kinematykę – analiza kinematyki prostej i odwrotnej manipulatorów,
- planowanie trajektorii – generowanie trajektorii ruchu manipulatorów i pojazdów mobilnych w przestrzeni operacyjnej,
- symulację i wizualizację – możliwość symulowania ruchu robotów oraz wizualizacji 3D,
- analizę dynamiki – symulacja sił i momentów działających w systemach robotycznych.

Robotics Toolbox wspiera badania nad algorytmami sterowania, umożliwia testowanie nowych metod planowania ruchu

oraz integrację z innymi narzędziami, takimi jak Simulink czy ROS. Dzięki swoim wszechstronnym możliwościom, jest szczególnie ceniony w projektach badawczo-rozwojowych i edukacji z zakresu automatyki i robotyki. [3]

D. Simscape

Simscape to narzędzie rozszerzające możliwości MATLAB-a i Simulinka, które umożliwia modelowanie, symulację oraz analizę systemów fizycznych w sposób intuicyjny, za pomocą komponentów graficznych. Jest szczególnie przydatne w dziedzinach, gdzie systemy mają złożoną strukturę wielodomenową, łączącą elementy mechaniczne, elektryczne, hydrauliczne czy termiczne [4]. Główne cechy Simscape:

- modelowanie wielodomenowe – możliwość integracji elementów z różnych dziedzin fizyki w jednym środowisku symulacyjnym,
- komponenty gotowe do użycia – bogata biblioteka bloków reprezentujących elementy, takie jak rezystory, silniki, zawory czy tłoki,
- podejście opierające się na fizyce – modele bazują na rzeczywistych równaniach opisujących zjawiska fizyczne, co ułatwia dokładne odwzorowanie systemów,
- symulacja dynamiczna – możliwość analizowania zachowań systemów w czasie, z uwzględnieniem nieliniowości i sprzężeń między domenami.

E. FOMCON

FOMCON jest biblioteką Matlaba i Simulinka, która wprowadza do środowisk możliwość dodania numerycznej implementacji układów regulacji bazujących na funkcjach niecałkowitego rzędu. Biblioteka bazuje na definicji dyskretnego Grünwald'a-Letnikov'a (2).

$$\Delta^\alpha x_k = \sum_{j=0}^k (-1)^j \binom{\alpha}{j} x_{k-j} \quad (2)$$

Do odtwarzania wyników ciągu dyskretnego - w bibliotece - wykorzystywana jest aproksymacja Oustaloupa, która jest budowana na podstawie transmitancji operatorowej rachunku, w zdefiniowanym zakresie częstotliwości i ograniczeniu pamięci próbek N (3).

$$s^\alpha = G_\omega \prod_{k=-N}^N \frac{s + \omega'_k}{s + \omega_k} \quad (3)$$

Szczegóły transformacji i aproksymacji są dokładniej opisane przez autorów biblioteki [5], [6]. Gotowy blok regulatora; Fig 1; niecałkowitego rzędu może zostać wstawiony bezpośrednio do wcześniej przygotowanych modeli.

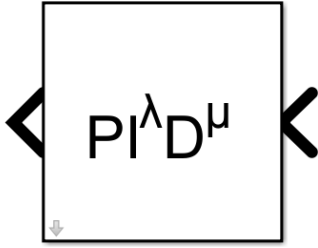


Fig. 1. Blok regulatora FOPID

Wewnątrz bloku można określić parametry tj.:

- wzmocnienie proporcjonalne K_p ,
- wzmocnienie całkowania K_i ,
- rząd całkowania λ ,
- wzmocnienie różniczkowania K_d ,
- rząd różniczkowania μ ,
- zakres częstotliwości aproksymacji,
- rząd aproksymacji.

F. ROS2

Robot Operating System 2 (ROS2) to otwarte środowisko programistyczne zaprojektowane do tworzenia aplikacji dla systemów robotycznych. Stanowi rozwinięcie pierwotnej wersji ROS, wprowadzając znaczące ulepszenia w zakresie wydajności, skalowalności oraz bezpieczeństwa. ROS2 umożliwia projektowanie systemów modularnych, które mogą działać na wielu platformach, takich jak Linux, Windows czy macOS, a także na urządzeniach wbudowanych. Dzięki wsparciu dla komunikacji typu publish-subscribe, ROS2 pozwala na efektywną wymianę danych pomiędzy komponentami systemu robotycznego [9].

Wizualizacja manipulatora została stworzona w aplikacji RVIZ systemu ROS2.

G. Hamachi

Hamachi to narzędzie do tworzenia wirtualnych sieci prywatnych (VPN), które umożliwia bezpieczne połączenie komputerów przez internet, tak jakby znajdowały się one w tej samej lokalnej sieci. Jest to rozwiązanie typu peer-to-peer, często używane do zdalnej gry, pracy zespołowej i w sytuacjach, gdy tradycyjne rozwiązania VPN są zbyt skomplikowane lub kosztowne. Połączenia w Hamachi są szyfrowane, co zapewnia bezpieczeństwo przesyłanych danych. Umożliwia to dostęp do zasobów w sieci lokalnej, nawet jeśli komputery znajdują się w różnych miejscach na świecie. Hamachi działa bez

potrzeby ręcznej konfiguracji routerów, co upraszcza połączenie w środowiskach, w których nie ma pełnej kontroli nad ustawieniami sieciowymi. Może być używane do współpracy na plikach i aplikacjach działających w sieci lokalnej, a także do dostępu do urządzeń, takich jak drukarki czy kamery sieciowe. Umożliwia także zdalny dostęp do zasobów firmowych lub domowych. Hamachi działa na systemach Windows, macOS i Linux, a jego wersja podstawowa jest bezpłatna, z ograniczeniem liczby użytkowników w wirtualnej sieci. Dla większej liczby połączeń dostępna jest wersja płatna. W artykule sieć VPN została wykorzystana do przesyłania pozycji stawów i wizualizacji procesu, symulacji manipulatora w RVIZ dostępnym wraz z systemem ROS2. W systemie opartym na Linux, tj. Ubuntu Desktop Hamachi nie posiada GUI. Żeby zaimplementować interfejs graficzny należy pobrać Haguichi (GUI Hamachi w Ubuntu) [8]. Proces instalacji dla systemu Ubuntu, Linux Mint i innych systemów pochodnych:

```
sudo add-apt-repository -y ppa:ztefn/haguichi-stable
sudo apt update
sudo apt install -y haguichi
```

H. ROS Toolbox

ROS Toolbox [10] to narzędzie dostępne w MATLAB, które umożliwia integrację z systemami opartymi na Robot Operating System (ROS) oraz ROS 2. Ułatwia projektowanie, symulację i analizę algorytmów dla systemów robotycznych, a także pozwala na komunikację z rzeczywistymi robotami. Toolbox oferuje funkcje takie jak wysyłanie i odbieranie wiadomości ROS, subskrypcję tematów, wywoływanie usług i akcji, a także generowanie niestandardowych typów wiadomości. Umożliwia również symulację i wizualizację za pomocą narzędzi takich jak Gazebo czy Rviz oraz integrację z Simulink. Blok Publisher umożliwia wysyłanie danych środowiska Simulink; w tym wypadku przez VPN; do systemu ROS2; Rysunek 2.

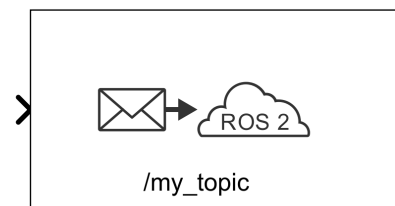


Fig. 2. Blok publikujący na temacie

Blok odbierający dane z tematu; Rysunek 3; odbiera dane, które zwraca system ROS2 (nasłuchuje danych na określonym temacie).

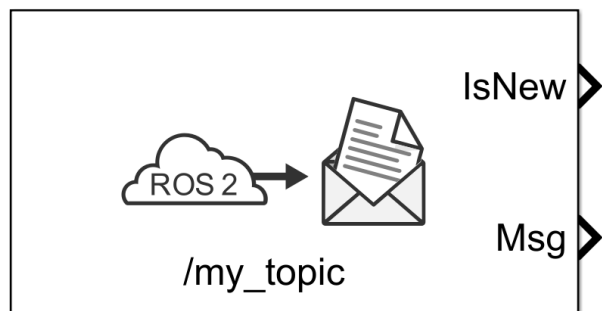


Fig. 3. Subskrybent w Simulinku

I. Model Manipulatora

Modele 3D elementów manipulatora zostały opracowane w programie Fusion 360. Następnie zostały zaimportowane do środowiska Simulink, gdzie przeprowadzono ich odpowiednie rozmieszczenie, uwzględniając także właściwości inercyjne oraz parametry materiałowe poszczególnych elementów robota. W rezultacie tych działań uzyskano implementację modelu matematycznego manipulatora, gotowego do dalszej analizy i symulacji.

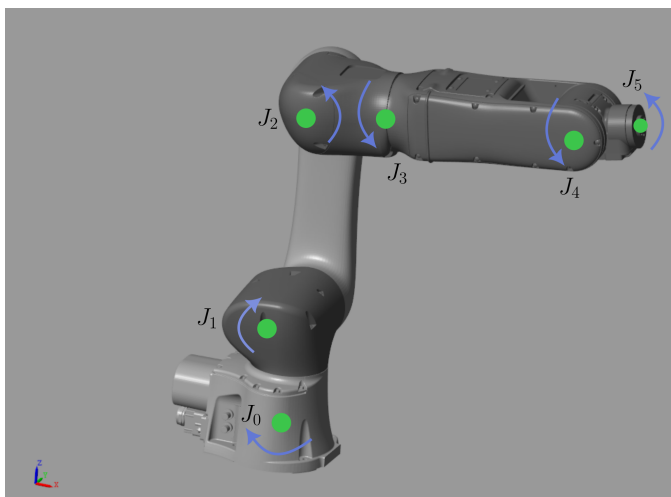


Fig. 4. Model 3D manipulatora

Robot posiada 6 stawów typu revolute, co implikuje ruch o sześciu stopniach swobody. Układ sterowania; Fig 5 został zbudowany w oparciu o technikę kaskadowych układów regulacji. Nadrzędną pętlę stanowi układ regulacji położenia kątownego. Wyjściem regulatora położenia jest zadana prędkość kątowna, która reaguje w pętli podrzędnej z regulatorem prędkości kątownej. Dla każdego stawu układ regulacji został skopiowany sześciokrotnie. Podczas tworzenia modelu nie zostały uwzględnione modele układów napędowych, ponieważ wtedy poziom skomplikowania i czas budowy modelu zdecydowanie

by się wydłużył. Dynamika samych napędów nie ma wpływu na przeprowadzane badanie.

J. Proces strojenia regulatorów

Ręczne strojenie większego, nieliniowego modelu z początku może wydawać się skomplikowane jednak w tym rozdziale zostaną zestawione; empiryczne spostrzeżenia związane z całym procesem. Na początku trzeba ustawić wartości wzmocnień tak, aby regulatory utrzymywały się samoistnie w początkowym położeniu równowagi lub mechanicznie zablokować położenie wszystkich stawów. Strojenie rozpoczyna się od stawu końcowego, ponieważ układ regulacji położenia efektora jest najmniej zależny od dynamiki pozostałych członów robota. Proces ręcznego doboru nastaw powinien się rozpocząć od pętli wewnętrznych (prędkości kątowych). Na wyjściu układ zapewnia moment, który powinien przewyższyć opory ruchu i aperiodycznie doprowadzić do osiągnięcia zadanego położenia. Nadrzędne układy regulacji muszą zawierać regulatory PID, ponieważ z tego co zostało zauważone podczas strojenia, jeśli dynamika członu zależy również od innego, niezależnego członu, żeby przeciwdziałać szybkim zmianom i ewentualnym drganiom robota, potrzebny jest człon różniczkujący, który zapewni dużą reakcję na szybkie zmiany położenia układu. Po procesie strojenia można przejść do wykonania właściwych badań. Nastrojone układy regulacji całkowitego rzędu stanowią część porównawczą badań.

IV. WYNIKI

W bieżącym rozdziale szczegółowo omówiono wyniki badań dotyczących wpływu zastosowania regulatorów niecałkowitego rzędu w podrzędnej pętli regulacji prędkości kątowej stawów. Celem tych badań było porównanie efektywności regulatorów niecałkowitego rzędu z klasycznymi regulatorami całkowitego rzędu. W trakcie eksperymentów, podczas modulacji rzędu regulatorów, nie zmieniano nastaw samych regulatorów, co pozwoliło na zachowanie spójności parametrów i umożliwiło bezpośrednie porównanie charakterystyk dynamicznych całego układu. W ten sposób badano, jak zmiana rzędu regulatorów wpływa na stabilność i dokładność śledzenia zadanej nieliniowej trajektorii w układzie regulacji.

A. Charakterystyki częstotliwościowe regulatorów

Zadanie sinusoidalnie zmieniającego się położenia kątownego, dla każdego ze stawów prowadzi do realizacji nieliniowej trajektorii. Amplituda sinusoidy wynosząca 10 stopni rozszerza zakres badanych częstotliwości, umożliwiając analizę

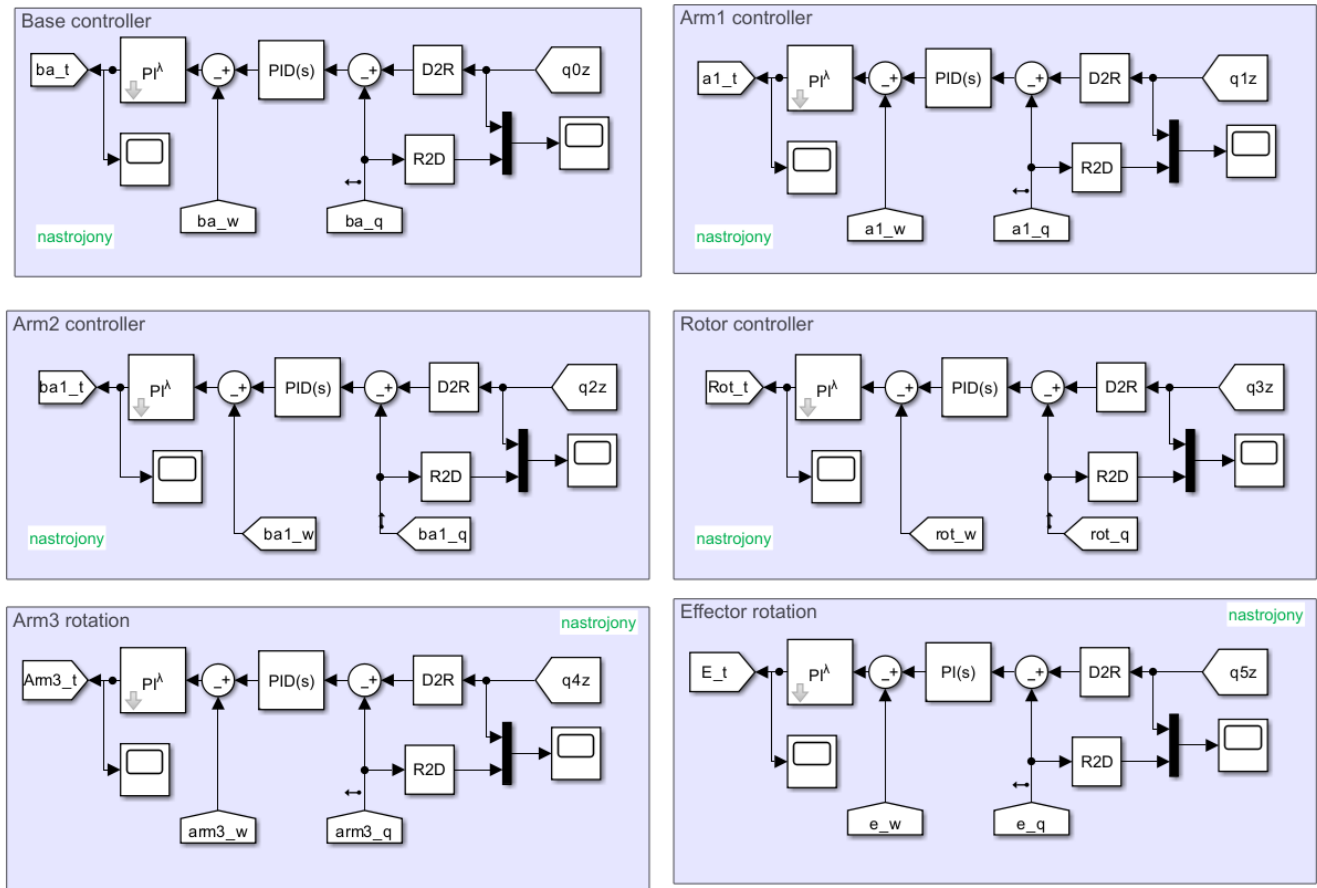


Fig. 5. Układ sterowania położeniem kątowym stawów manipulatora

systemu w szerszym zakresie dynamiki. Stosunek otrzymanej do zadanej amplitudy zmiany położenia kąowego dla pięciu znaczących ramion i różnych rzędów całkowania w stanie ustalonym, pod względem częstotliwości zestawiają Tabele I. W miejscach, gdzie zamiast wyników pojawiają się myślniki, oznacza to, że dany człon układu nie był stabilny, nie osiągał wymaganej wartości zadanej lub z powodu specyfiki jego działania występowała składowa zerowa. Ta składowa uniemożliwiała poprawne odwzorowanie sygnału zadanego, co mogło prowadzić do braku pełnej realizacji oczekiwanego przebiegu. W Tabeli I, przy częstotliwości 4 Hz, odczytanie wartości amplitudy nie jest możliwe ze względu na występowanie znaczącej składowej zerowej, która uniemożliwia odtworzenie wartości zadanej. Badanie dla $\lambda = 0.5$ ogranicza się do jeszcze mniejszych częstotliwości sygnałów zadanych; Tabela II. Zmniejszenie rzędu całkowania dodatkowo ogranicza możliwości odtworzenia sygnału zadanego przez układ. Ograniczenie rzędu całkowania w pętli regulacji prędkości kątovej zmniejsza możliwość do odtworzenia momentu zadanego na wyjściu regulatora, w wyniku czego zwiększa się błąd składowej stałej i zmniejsza stabilność układu dla większych

TABLE I
CHARAKTERYSTYKA CZĘSTOTLIWOŚCIOWA DLA $\lambda = 1$

f_z	$\frac{I_0}{I_z}$	$\frac{I_1}{I_z}$	$\frac{I_2}{I_z}$	$\frac{I_3}{I_z}$	$\frac{I_4}{I_z}$
Hz	-	-	-	-	-
$\frac{1}{50}$	1	0.96	1	1	1
$\frac{1}{40}$	1	0.96	1	1	1
$\frac{1}{30}$	1	0.96	0.95	1	0.95
$\frac{1}{20}$	1	0.96	0.92	1	0.94
$\frac{1}{10}$	1	0.96	0.92	1	0.94
$\frac{1}{5}$	1	0.97	0.92	0.96	0.96
$\frac{1}{4}$	0.98	0.97	0.94	0.94	0.97
$\frac{1}{3}$	0.95	0.97	0.95	0.94	0.98
$\frac{1}{2}$	0.86	0.97	0.96	0.91	1.02
1	0.76	0.97	0.90	0.87	0.92
2	0.76	0.97	0.74	0.78	0.60
3	0.72	0.94	0.67	-	0.40
4	-	-	-	-	-

częstotliwości. Tabela III zestawia dane dotyczące badania odtwarzania sinusoidalnego sygnału zadanego, dla $\lambda = 1.5$. Ogólnie, częstotliwości powyżej 1 Hz są silnie tłumione przez układ; dynamika jest ciężka i układ nie nadąża za szybkimi zmianami. Dodatkowo podczas szybkich zmian sygnał

TABLE II
CHARAKTERYSTYKA CZĘSTOTLIWOŚCIOWA DLA $\lambda = 0.5$

f_z	$\frac{I_0}{I_z}$	$\frac{I_1}{I_z}$	$\frac{I_2}{I_z}$	$\frac{I_3}{I_z}$	$\frac{I_4}{I_z}$
Hz	-	-	-	-	-
$\frac{1}{50}$	1	0.96	0.95	1	1
$\frac{1}{40}$	1	0.96	0.97	1	0.95
$\frac{1}{30}$	1	0.96	0.97	1	0.95
$\frac{1}{20}$	1	0.96	0.97	1	0.94
$\frac{1}{10}$	1.1	0.96	0.97	1	0.94
$\frac{1}{5}$	1	0.97	0.93	0.96	0.96
$\frac{1}{4}$	0.98	0.97	0.94	0.95	0.98
$\frac{1}{3}$	0.95	0.98	0.95	0.94	1
$\frac{1}{2}$	0.86	0.97	0.96	0.92	1.02
1	0.76	0.97	0.90	-	-
2	-	-	-	-	-
3	-	-	-	-	-
4	-	-	-	-	-

wyjściowy podlega nieliniowym deformacjom. Następne wyniki przedstawia analizę składowych harmoniczných sygnałów wyjściowych.

TABLE III
CHARAKTERYSTYKA CZĘSTOTLIWOŚCIOWA DLA $\lambda = 1.5$

f_z	$\frac{I_0}{I_z}$	$\frac{I_1}{I_z}$	$\frac{I_2}{I_z}$	$\frac{I_3}{I_z}$	$\frac{I_4}{I_z}$
Hz	-	-	-	-	-
$\frac{1}{50}$	1	0.97	0.97	0.98	0.95
$\frac{1}{40}$	1	0.97	0.96	0.98	0.94
$\frac{1}{30}$	1	0.97	0.94	0.98	0.94
$\frac{1}{20}$	1	0.97	0.92	0.97	0.93
$\frac{1}{10}$	1	0.97	0.90	0.97	0.83
$\frac{1}{5}$	1	0.97	0.92	0.95	0.90
$\frac{1}{4}$	1	0.97	0.94	0.95	1
$\frac{1}{3}$	0.95	0.97	0.95	0.94	1
$\frac{1}{2}$	0.86	0.97	0.95	0.94	1.2
1	0.76	0.97	0.89	0.88	0.96
2	0.78	0.97	0.75	-	-
3	-	-	-	-	-
4	-	-	-	-	-

B. Analiza składowych harmoniczných

V. BIBLIOGRAFIA

- [1] Matlab - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://www.mathworks.com/help/matlab/>
- [2] Simulink - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://www.mathworks.com/help/simulink/>
- [3] Robotics Toolbox - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://www.mathworks.com/help/robotics/>
- [4] Simscape - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://www.mathworks.com/help/robotics/>
- [5] A. Tepljakov, E. Petelnikov, and J. Belikov, "A flexible MATLAB tool for optimal fractional-order PID controller design subject to specifications,"*Proc. 31st Chinese Control Conf.*, Hefei, China, Jul. 2012, pp. 4876–4881. [Online]. Dostęp (22.11.2024 r.): <https://www.researchgate.net/publication/259741861>
- [6] FOMCON - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://fomcon.net/>
- [7] Hamachi - strona [Online]. Dostęp (22.11.2024 r.): <https://www.vpn.net/>
- [8] Haguichi - instrukcja pobrania [Online]. Dostęp (22.11.2024 r.): <https://haguichi.net/download/>
- [9] ROS2 foxy - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://docs.ros.org/en/foxy/index.html>
- [10] ROS Toolbox - dokumentacja [Online]. Dostęp (22.11.2024 r.): <https://www.mathworks.com/help/ros/>
- [11] K. Rogowski, T. Kaczorek, "Fractional Linear Systems and Electrical Circuits", Studies in Systems, Decision and Control, Springer, NY, 2015
- [12] T. Kaczorek, "Selected Problems of Fractional Systems Theory", Lecture Notes in Control and Information Sciences 411, 2011 Springer-Verlag Berlin Heidelberg
- [13] E. Pawłuszewicz, A. Koszewnik, P. Burzyński, "ON GRÜNWALD-LETINKOV FRACTIONAL OPERATOR WITH MEASURABLE ORDER ON CONTINUOUS-DISCRETE TIME SCALE", acta mechanica et automatica, vol.14 no.3 (2020)
- [14] M. Popolizio, "On the Matrix Mittag-Leffler Function: Theoretical Properties and Numerical Computation", Mathematics, 2019
- [15] Pushpendra Kumar, Dumitru Baleanu, Vedat Suat Erturk, Mustafa Inc and V. Govindaraj, "A delayed plant disease model with Caputo fractional derivatives", Advances in Continuous and Discrete Models, Springer, 2022
- [16] Mustafa Turkyilmazoglu, Mohamed Altanji, "Fractional models of falling object with linear and quadratic frictional forces considering Caputo derivative", Chaos, Solitons and Fractals, Elsevier, 2023
- [17] Muhammad Farman, Rabia Sarwar, Ali Akgul, "Modeling and analysis of sustainable approach for dynamics of infections in plant virus with fractal fractional operator", Chaos, Solitons and Fractals, Elsevier, 2023
- [18] Georgios Tzounas, Ioannis Dassios, Mohammed Ahsan Adib Murad, and Federico Milano, "Theory and Implementation of Fractional Order Controllers for Power System Applications", IEEE TRANSACTIONS ON POWER SYSTEMS, VOL. 35, NO. 6, NOVEMBER 2020