

**Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie**

Wydział Elektrotechniki, Automatyki, Informatyki
i Inżynierii Biomedycznej

KATEDRA AUTOMATYKI I ROBOTYKI



ROZPRAWA DOKTORSKA

WALDEMAR BAUER

**IMPLEMENTACJA UKŁADÓW NIECAŁKOWITEGO
RZĘDU W SYSTEMACH WBUDOWANYCH**

AUTOR: Waldemar Bauer

DYSCYPLINA NAUKOWA:

Automatyka i Robotyka

PROMOTOR:

Prof. dr hab. inż.

Wojciech Mitkowski

Kraków 2019

OŚWIADCZENIE AUTORA PRACY

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

.....

PODPIS

AGH
University of Science and Technology in Krakow

Faculty of Electrical Engineering, Automatics, Computer Science
and Biomedical Engineering

DEPARTMENT OF AUTOMATICS AND ROBOTICS



DOCTORAL DISSERTATION

WALDEMAR BAUER

**IMPLEMENTATION OF THE FRACTIONAL ORDER
SYSTEMS IN THE EMBEDDED SYSTEMS**

AUTHOR: Waldemar Bauer

SCIENTIFIC DISCIPLINE:

Automation and Robotics

SUPERVISOR:

Professor dr hab. eng.

Wojciech Mitkowski

Krakow 2019

Pracę dedykuję moim bliskim, szczególnie Rodzicom.

Pragnę złożyć serdeczne podziękowania promotorowi Profesorowi Wojciechowi Mitkowskiemu za trud włożony w opiekę naukową, okazaną życzliwość, cierpliwość oraz poświęcony czas.

Pragnę podziękować również dr. hab. inż. Jerzemu Baranowskiemu, dr. inż. Pawłowi Piątkowi oraz mgr inż. Marcie Zagórowskiej za okazaną mi pomoc, wsparcie i mobilizowanie do działania.

Streszczenie

Autor rozprawy sprawdził w niej tezę dotyczącą możliwości zdefiniowania postaci aproksymacji układów niecałkowitego rzędu które po dyskretyzacji będzie można zaimplementować w układach wbudowanych czasu rzeczywistego.

Rozprawa została podzielona na siedem rozdziałów zasadniczych w których przedstawiono podstawy teoretyczne dotyczące rachunku różniczkowego niecałkowitego rzędu. Opisano w niej również znane sposoby implementacji tej klasy układów wraz z metodami opracowanymi przez autora rozprawy - czasowa metoda Oustaloupa oraz równoległa metoda Oustaloupa.

Przedstawiono rodzaje stosowanych obecnie układów niecałkowitego rzędu. Zaprezentowano teorię dotyczącą układów wbudowanych w szczególności bazujących na systemach czasu rzeczywistego. Opis metody implementacji opracowanych aproksymacji na układy wbudowane i sprawdzenie ich poprawności na platformach sprzętowych Arduino Uno oraz STM32F0-DISCOVERY.

W następnym rozdziale rozprawy opisano eksperymenty weryfikujące poprawność zaproponowanych implementacji metod aproksymacji układów niecałkowitego rzędu poprzez ich realizację na laboratoryjnych układach nagrzewnicy powietrza i lewitacji magnetycznej.

Ostatni rozdział zasadniczy zawiera podsumowanie osiągniętych rezultatów oraz opis dalszych możliwych kierunków rozwoju. Dodatkowo na końcu rozprawy znajduje się spis tabel oraz ilustracji zawartych w rozprawie. Rozprawę zamyka spis bibliograficzny zawierający 192 publikacje.

Abstract

The author of the dissertation checked the thesis on the possibility of defining the form of approximation of non-integer order systems that after discretization can be implemented in embedded real-time systems.

The dissertation was divided into seven main chapters in which the theoretical basis for the non-integer differential calculus was presented. It also describes the state of art of implementing this class of systems together with the method developed by the author of the dissertation - the time domain Oustaloup method and the parallel Oustaloup method.

Presented are selected types of currently used non-integer order systems. The theory concerning embedded systems, in particular, those based on real-time systems, has been presented. Description of the implementation method of the developed approximations for embedded circuits and checking their correctness on the Arduino Uno and STM32F0-DISCOVERY hardware platforms.

The next chapter of the dissertation describes experiments verifying the correctness of proposed implementations of approximation methods of partial order systems by their implementation on laboratory systems of air heater and magnetic levitation.

The last main chapter contains a summary of the results achieved and a description of further possible directions of researchs. In addition, at the end of the hearing, there is a list of tables and illustrations contained in the dissertation. The dissertation closes the bibliographic list containing 192 publications.

Spis treści

1. Wstęp	15
1.1. Zarys historyczny	15
1.2. Cele i tezy pracy	16
1.3. Motywacja	16
1.4. Układ i zakres pracy	20
2. Wstęp matematyczny	23
2.1. Funkcje specjalne	23
2.2. Definicja operatora różniczko-całki niecałkowitego rzędu	24
2.3. Definicja całki i różniczki Grünwalda-Letnikova (GL).....	25
2.4. Definicja całki i różniczki Riemanna-Liouville'a (RL).....	26
2.5. Definicja różniczki Caputo (C).....	27
2.6. Transformata Laplace'a układów ułamkowych.....	27
2.6.1. Ogólne własności transformaty Laplace'a	28
2.6.2. Transformata Laplace'a dla operatora niecałkowitego rzędu.....	28
2.6.3. Metody dyskretyzacji	29
3. Cyfrowa realizacja układów niecałkowitego rzędu	31
3.1. Rozwiązanie układu równań liniowych niecałkowitego rzędu z krótką pamięcią	31
3.2. Aproksymacja CFE (Continued Fraction Expansion)	33
3.2.1. Bazowa metoda aproksymacji CFE operatora różniczko-całki niecał-	
kowitego rzędu.....	34
3.2.2. Metoda Carlsona.....	34
3.2.3. Metoda Matsuda	35
3.3. Metoda Charefa	35
3.4. Metoda Djouambi	36
3.5. Metoda rozwinięcia w szereg potęgowy - Power Series Expansion (PSE).....	36
3.6. Aproksymacja LIRA	37
3.6.1. Skończenie wymiarowa aproksymacja układu niecałkowitego rzędu....	38

3.7.	Aproksymacja Oustaloupa.....	39
3.8.	Aproksymacje oparte na aproksymacji Oustaloupa	41
3.8.1.	Równoległa aproksymacja Oustaloupa	41
3.8.2.	Czasowa aproksymacja Oustaloupa	42
4.	Stosowane układy niecałkowitego rzędu.....	46
4.1.	Regulator $PI^{\alpha}D^{\mu}$	46
4.2.	Filtr niecałkowitego rzędu dolnoprzepustowy.....	48
4.3.	Filtr niecałkowitego rzędu górnoprzepustowy	48
4.4.	Filtr niecałkowitego drugiego rzędu.....	49
4.5.	Filtr niecałkowitego rzędu pasmowo przepustowy.....	49
5.	Układy wbudowane.....	52
5.1.	Układy wbudowane wybrane do testów	52
5.1.1.	Arduino Uno.....	52
5.1.2.	Mikrokontroler STM32	54
5.2.	Czas rzeczywisty w układach wbudowanych.....	56
5.2.1.	Systemy czasu rzeczywistego.....	56
5.2.2.	Systemy wbudowane	58
5.2.3.	Rodzaje systemów czasu rzeczywistego	59
5.3.	Implementacja elementów niecałkowitego rzędu w układach wbudowanych ...	62
5.3.1.	Algorytmy.....	62
5.4.	Wyniki implementacji na układach wbudowanych	64
5.4.1.	Implementacja Arduino Uno	64
5.4.2.	Implementacja STM32	69
5.5.	Wyniki eksperymentu	70
6.	Wykorzystanie ułamkowych regulatorów ułamkowych typu PID w sterowaniu rzeczywistymi obiektami.....	80
6.1.	Realizacja regulatora ułamkowego PID dla systemu nagrzewnicy powietrznej	80
6.1.1.	Model nagrzewnicy powietrznej.....	81
6.1.2.	Funkcja czułości dla nagrzewnicy powietrznej	83
6.1.3.	Implementacja regulatora $PI^{\alpha}D^{\mu}$ i wyniki eksperymentu	85
6.2.	Realizacja regulatora $PI^{\lambda}D$ dla układu lewitacji magnetycznej	89
6.2.1.	Matematyczny model lewitacji magnetycznej	89
6.2.2.	Dobór parametrów regulatora.....	90
6.2.3.	Implementacja zaprojektowanego regulatora	92
6.2.4.	Redukcja całkowitego rzędu aproksymacji s^{γ}	95

6.2.5. Eksperymenty na systemie laboratoryjnym.....	99
7. Podsumowanie.....	102

Zbiór oznaczeń

\mathbb{R} - zbiór liczb rzeczywistych

\mathbb{N} - zbiór liczb naturalnych

a, b - Wartości skalarne:

Wektory: a, b , wszystkie wektory w pracy są wektorami pionowymi to znaczy o wymiarze $n \times 1$

Macierze: A, B

$x(0)$ - wektor wartości początkowych systemu

D - dziedzina funkcji

$f : D \rightarrow Y$ - funkcja f przekształcająca dziedzinę funkcji D w zbiór Y

$f(x)$ - funkcja f , o wartościach w zbiorze liczb rzeczywistych, przyjmująca jako argumenty wartości rzeczywiste x

$\det A$ - wyznacznik macierzy A

$f(x)$ - funkcjonal f , o wartościach w zbiorze liczb rzeczywistych, przyjmujący jako argument wektor liczb rzeczywistych x

$f^{(n)}(x), f'(x)$ - pochodna funkcji $f(x)$ stopnia n

$\frac{d^n}{dt^n}f(x)$ - pochodna funkcji, zdefiniowana jako:

$$\frac{d^q}{dt^q} = \frac{d^{n-m}f(t)}{dt^{n-m}} = \frac{1}{\Gamma(m)} \frac{d^n}{dt^n} \int_0^t (t-y)^{m-1} f(y) dy$$

$\Gamma(m)$ - funkcja zdefiniowana jako:

$$\Gamma(m) = \int_0^{\infty} e^{-u} u^{m-1} du$$

$\binom{n}{k}$ - uogólniony symbol Newtona:

$$\binom{n}{k} = \frac{n(n-1) \cdots (n-k+1)}{k!} = \frac{(n)_k}{k!}$$

$E_{\alpha,\beta}(z)$ - funkcja Mittag-Leffler, zdefiniowana jako:

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\alpha + \beta)}$$

$\operatorname{erfc}(x)$ - funkcja błędu Gaussa, zdefiniowana jako:

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

${}_a D_t^\alpha$ - ciągły operator różniczko-całki niecałkowitego rzędu zdefiniowana jako:

$${}_a D_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \alpha > 0, \\ 1 & \alpha = 0, \\ \int_a^t (d\tau)^{-\alpha} & \alpha < 0 \end{cases}$$

gdzie: α rząd pochodnej, a i t wartości graniczne dla czasu

${}^GL_a D_t^\alpha$ - różniczko-całka niecałkowitego rzędu Grünwalda-Letnikova (GL) zdefiniowana jako:

$${}^GL_a D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t - jh)$$

gdzie: α rząd pochodnej, a i t wartości graniczne dla czasu

${}^{RL}_a D_t^\alpha$ - różniczko-całka niecałkowitego rzędu Riemann-Liouville (RL) zdefiniowana jako:

$${}^{RL}_a D_t^\alpha f(t) = \frac{1}{\Gamma(\alpha - n)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t - \tau)^{\alpha - n + 1}} d\tau$$

gdzie: α rząd pochodnej, a i t wartości graniczne dla czasu

${}^C_a D_t^\alpha$ - różniczko-całka niecałkowitego rzędu Caputto (C) zdefiniowana jako:

$${}^C_a D_t^\alpha f(t) = \frac{1}{\Gamma(\alpha - n)} \int_a^t \frac{f^{(n)}(\tau)}{(t - \tau)^{\alpha - n + 1}} d\tau$$

gdzie: α rząd pochodnej, a i t wartości graniczne dla czasu

\mathcal{L} - transformata Laplace'a

1. Wstęp

1.1. Zarys historyczny

Pojęcie operatora różniczkowania $\frac{d}{dx}$ jest ogólnie znanym pojęciem stosowany powszechnie w matematyce jak i w inżynierii. Dla odpowiednich funkcji n -ta pochodna funkcji $f(x)$, opisana operatorem $\frac{d^n}{dx^n}$ jest dobrze zdefiniowana, gdy n jest dodatnią liczbą całkowitą. W 1695 r. L'Hopital zapytał Leibniza, jakie znaczenie można przypisać operatorowi $\frac{d^n}{dx^n}$, jeśli n jest ułamkiem, więcej szczegółów zostało opisanych przez Weilbeer (2006).

Od tego czasu rachunek różniczkowy niecałkowitego (ułamkowego) rzędu był rozwijany przez wielu słynnych matematyków, takich jak Euler, Laplace, Fourier, Abel, Liouville, Weilbeer, Riemann i Laurent. W 1730 r. tematem rachunku niecałkowitego rzędu zajmował się Eulera. J.L. Lagrange w 1772 przyczynił się do rozwoju tego rodzaju rachunku różniczkowego w sposób pośredni, poprzez opracowanie prawa wykładników dla operatorów różnicowych. W 1812 roku P. S. Laplace zdefiniował pochodną niecałkowitego rzędu za pomocą całki, a w 1819 S.F. Lacroix rozważał własności pochodnej *arbitralnego rzędu*, a następnie własnościami tej pochodnej zainteresował się J. B. J. Fourier w 1822 r.. Pierwszym znanym zastosowaniem rachunku niecałkowitego rzędu do rozwiązania problemu fizycznego było rozwiązanie problemu tautochronu przez N. H. Abel w 1823 r.. J. Liouville opisał studium rachunku niecałkowitego rzędu w 1832 r., w którym zastosował swoje definicje pochodnej niecałkowitego rzędu do problemów teoretycznych. W 1867 r. A.K. Grünwald pracował nad rozwojem rachunku niecałkowitego rzędu. G.F.B. Riemann w okresie swoich studiów pracował nad teorią całkowania niecałkowito rzędowego, którą opublikował w 1892 roku. A.V. Letnikov napisał kilka artykułów na ten temat w latach 1868 - 1872. W okresie od 1900 do 1970 r. rachunek niecałkowitego rzędu głównie rozwijał się dzięki: H.H. Hardy, S. Samko, H. Weyl, M. Riesz, S. Blair. Od 1970 r. W późniejszych latach za rozwój tej dziedziny odpowiedzialni byli: J. Spanier, K.B. Oldham, B. Ross, K. Nishimoto, O. Marichev, A. Kilbas, H.M. Srivastava, R. Bagley, K.S. Miller, M. Caputo, I. Podlubny, T. Kaczorek i wielu innych.

Obecnie zauważalny jest wzrost zainteresowaniem zastosowania układów niecałkowitego rzędu w rozwiązaniach problemów z zakresu modelowania procesów cieplnych, superdyfuzji, modelowaniu układów elektrycznych i tworzenia filtrów.

1.2. Cele i tezy pracy

Celem niniejszej pracy jest przedstawienie własnych propozycji metod aproksymacji układów niecałkowitego rzędu bazujących na metodzie Oustaloupa i ich implementacji w układach cyfrowych. Praca ma również podjąć próbę usystematyzowania wiedzy w zakresie implementacji układów ułamkowych w układach rzeczywistych.

Tezę niniejszej rozprawy doktorskiej sformułować można w następujący sposób:

”Możliwe jest zdefiniowanie postaci aproksymacji układów niecałkowitego rzędu którą po dyskretyzacji będzie można zaimplementować w układach wbudowanych w szczególności pracujących w czasie rzeczywistym.”

1.3. Motywacja

W dzisiejszych czasach szeroko zakrojonym problemem jest realizacja systemów niecałkowitego rzędu na platformy cyfrowe w szczególności pracujących w czasie rzeczywistym. Teoria tego typu układów jest dobrze ugruntowana, jednak wiele problemów związanych z implementacją na platformy sprzętowe jest nadal otwartych. Stworzenie stabilnej i odpornej numerycznie implementacji systemów niecałkowitego rzędu w postaci ich aproksymacji jest jedną z oczywistych potrzeb.

W ostatnich dziesięcioleciach XX wieku wzrosło zainteresowanie realizacją układów niecałkowitego rzędu zarówno w dziedzinie teorii sterowania. Jedną z osób które w tym okresie opisała teorie układów ułamkowych był Podlubny (1998). W pracy tej autor przedstawił zarówno zagadnienia związane z formalnymi definicjami różnych typów pochodnych niecałkowitego rzędu, definicjami stabilności tej klasy układów oraz zagadnienia implementacji w środowiskach cyfrowych. W późniejszym okresie tematyką teorii układów z pochodną niecałkowitego rzędu zajmowali się między innymi Diethelm (2010); Mitkowski (2011); Kaczorek (2011); Baranowski et al. (2015c).

Do najbardziej znaczących grup badawczych z tego okresu zajmujących się tematyką implementacji układów niecałkowitego rzędu należy zaliczyć zespół CRONE (**fr.** Commande Robuste d’Ordre Non Entier - Sterowanie odporne systemami niecałkowitego rzędu) którego pracami kierował Oustaloup. Oustaloup et al. (1999) przedstawili opis metod do identyfikacji i układów regulacji systemów dynamicznych niecałkowitego rzędu. Za szczególnie istotne osiągnięcie grupy CRONE należy uznać metodę estymacji wartości ułamkowego równania różniczkowego na podstawie definicji pochodnej Grünwalda-Letnikova metodą najmniejszych kwadratów Oustaloup et al. (1996). Natomiast Trigeassou et al. (1999) zaprezentował metodę aproksymacji układu niecałkowitego rzędu przy pomocy układu całkowitego rzędu.

W zakresie metod identyfikacji Hartley i Lorenzo (2003) zaproponowali by dla dziedziny częstotliwości zastosować metodę bazującą o ciągłych rozkład współczynników, a w

(Valerio i Sa da Costa, 2005) opisano metodę identyfikacji Levy'ego rozszerzoną na układy niecałkowitego rzędu.

Definicję kontrolera PID rzędu ułamkowego wprowadził Podlubny (1999, 1998). W (Čech i Schlegel, 2006; Monje et al., 2008; Xue et al., 2006) pokazano, że kontroler PID niecałkowitego rzędu posiada lepsze własności adaptacyjne ze względu na możliwość szerszego dopasowania charakterystyki częstotliwościowej tego układu. Valério i da Costa (2006) przedstawili przegląd metod strojenia ułamkowego kontrolera PID. Metody opisane w tej pracy można podzielić na analityczne, oparte na definicji stabilności oraz numeryczne, które zależą od przyjętej metody aproksymacji układu. Należy tutaj zaznaczyć, że metody strojenia bazujące na analitycznych zasadach są rzadkie ze względu na brak w teorii formalnych metod wyznaczania zapasów wzmocnień i fazy dla tego typu układów. Ze względu na to Monje et al. (2010) oraz Valério i da Costa (2006) zaproponowali metody strojenia tego typu regulatorów na podstawie klasycznych metod analitycznych ale bez dowodów formalnych. Natomiast numeryczne metody strojenia regulatora ułamkowego PID zostały zaprezentowane w wielu praktycznych zagadnieniach przez Monje et al. (2010). Do ich strojenia używano między innymi globalnych metod optymalizacji opisanych między innymi w (Chang i Chen, 2009; Cao i Cao, 2006; Bauer et al., 2013; Mitkowski i Oprzedkiewicz, 2013; Oprzedkiewicz i Dziedzic, 2017). Metody automatycznego strojenia układów niecałkowitego rzędu zostały również przedstawione przez Monje et al. (2008) gdzie do procesu strojenia bazuje się na metodzie relay feedback-based.

Coraz większe zainteresowanie budzi również uwzględnienie opóźnień w pętli sterowania z regulatorem ułamkowym PID, problem ten opisali między innymi (Åström i Hägglund, 2006; Yu, 2006) . Natomiast proces strojenia tego typu regulatora został przedstawiony w pracach Feliu-Batlle i Castillo-García (2014); Malek et al. (2013); Padula i Visioli (2015); Ruszewski (2008) . Jednak proces automatycznego strojenia regulatorów ułamkowego rzędu nie został tak dobrze opracowany jak ten sam proces dla klasycznych regulatorów PID Åström i Murray (2008).

Przegląd istniejących ciągłych i dyskretnych aproksymacji operatorów ułamkowych zaprezentowali Vinagre et al. (2000b). Cyfrowe aproksymacje operatorów ułamkowych zwykle opierają się metodach Continued Fraction Expansion (CFE) lub Power Series Expansion (PSE). Metodami implementacji układów niecałkowitego rzędu metodą CFE zajmowali się między innymi (Oprzedkiewicz i Mitkowski, 2018a; Oprzedkiewicz et al., 2015; Chen i Moore, 2002). Natomiast metoda PSE była używana przez (Oprzedkiewicz et al., 2018a; Stanisławski et al., 2017; Rydel et al., 2016b; Stanisławski et al., 2011). Cyfrowe realizacja układów niecałkowitego rzędu najczęściej przyjmują postać filtra FIR (filtru o skończonej odpowiedzi impulsowej) lub IIR (filtru o nieskończonej odpowiedzi impulsowej), przykład takich realizacji podał Petras et al. (2003b); Petras (2011a). W tym miejscu należy również zaznaczyć, że istnieją implementacje filtrów ułamkowych stochastycznych, dla przykładu w

pracach Sierociuk (2013); Sierociuk et al. (2016) przedstawiono uogólnią postać filtru Kalmana na układy niecałkowitego rzędu.

Jednak najważniejsze wyniki realizacji układów niecałkowitego rzędu zostały przedstawione w dziedzinie analizy sygnałów szczególnie w realizacji za pomocą układów analogowych. AbdelAty et al. (2016b) zaprezentowali wariant ułamkowy dla dobrze znanego równania różniczkowego Hermite'a dla filtrów. Zastosowano tam metodę rozwiązywania równania z pochodną niecałkowitego rzędu za pomocą ułamkowej metody szeregów Taylora. Tsirimokou et al. (2016) przedstawiają nowe topologiczne uogólnienie filtrów niecałkowitego rzędu. Metody opisane w tej pracy pozwalają na realizację filtrów niecałkowitego rzędu: dolnoprzepustowych, górnoprzepustowych, pasmowo-przepustowych i pasmo - zaporowych, przy użyciu tej samej topologii układu elektrycznego. W pracy Psychalinos et al. (2016) opisali proces projektowania filtru dolnoprzepustowego z przełączaniem kondensatorów dla niecałkowitego rzędu kroku przełączeń o charakterystyce Butterwortha i jego implementację na platformie CMOS. W omawianej pracy zostały przedstawione wyniki zarówno obliczeń analitycznych jak i wyniki eksperymentalne. Khateb et al. (2016) przedstawili sposób projektowania i implementacji filtrów rzędu ułamkowego w oparciu o strukturę CMOS z różnicowym przenośnikiem prądu (**ang.** *Differential Difference Current Conveyor - DDCC*). Tepljakov et al. (2016, 2015, 2014) przedstawili cyfrową implementację kontrolera PID niecałkowitego rzędu w oparciu o strukturę filtra o nieskończonej odpowiedzi impulsowej uzyskaną przez zastosowanie przybliżenia Oustaloup. Petras et al. (2003a) zaprezentowali metodę realizacji regulatorów ułamkowych na mikroprocesorze PIC i zaimplementowanych w języku PIC Basic.

Poza przedstawionym zastosowaniami układy niecałkowitego rzędu służą do modelowania dynamiki zjawisk. Jako najważniejsze typy zjawisk które są modelowane przy użyciu tego aparatu matematycznego można podać:

1. **Systemy elektryczne:** przetwornica prądu stałego DC (Radwan et al., 2018), indukcyjność (Foupouapouognigni et al., 2017), obwody elektryczne (Gómez-Aguilar et al., 2016), transformator różnicowy liniowy (Veeraian et al., 2017), silnik synchroniczny z magnesem trwałym (Thakar et al., 2016), silnik indukcyjny (Jalloul et al., 2013), silnik prądu stałego (Cipin et al., 2014), fotowoltaiczny moduł słoneczny (AbdelAty et al., 2016a).
2. **Akumulatory:** akumulatory litowo-jonowe (Wang et al., 2015), ogniwa paliwowe (Taleb et al., 2017), superkondensatory (Lewandowski i Orzyłowski, 2017).
3. **Procesy cieplne:** nieliniowy system termiczny (Oprzedkiewicz i Mitkowski, 2018b; Mitkowski i Obrączka, 2011; Maachou et al., 2014), rozkład temperatury w wirującym satelicie (Prajapati et al., 2016), przewodzenie ciepła (Oprzedkiewicz et al., 2018b; Obrączka i Mitkowski, 2014; Zecova i Terpak, 2015), wymiana ciepła w bezkontak-

towych uszczelnieniach czołowych (Blasiak, 2016), dynamika termiczna budynków (Długosz i Skruch, 2015).

4. **Modelowanie reaktorów jądrowych:** zasad tworzenia neutronów (Espinosa-Paredes et al., 2008; Vyawahare i Nataraj, 2013a), modele kinetyki reaktora punktowego (Espinosa-Paredes et al., 2011; Vyawahare i Nataraj, 2013b), model funkcji przeniesienia liniowego (Vyawahare et al., 2016).
5. **Dyfuzje:** dyfuzja anomalna (Metzler i Klafter, 2000), nieliniowy proces reakcji dyfuzji Fisher'a (Atangana, 2016), proces dyfuzji (Sierociuk et al., 2015; Mitkowski, 2011).
6. **Systemy mechaniczne:** nieliniowe model w mechanice (Grzesikiewicz et al., 2013), model potrójnego wahadła (Coronel-Escamilla et al., 2016), układ przekładni zębatej z luzami (Hedrih i Nikolić-Stanojević, 2010), turbina gazowa (Jadhav et al., 2016), przepływ płynu w silnikach spalinowych (Lino et al., 2015).
7. **Urządzenia nano-elektroniczne:** elektrochemiczny nano-biochip (Djouambi et al., 2013).

Poza zastosowaniem w inżynierii układy niecałkowitego rzędu służą obecnie do modelowania dynamiki zjawisk w dziedzinach takich jak:

1. **Biologia:** bioinżynierii (Magin, 2012), tkanek biologicznych (Magin, 2010; Biswas et al., 2017), układu oddechowego i płuc (Ionescu i Kelly, 2017; Ionescu et al., 2011), DNA (Machado, 2015), dynamiki białek (Glockle i Nonnenmacher, 1995).
2. **Medycyna:** farmakokinetyce i farmakodynamice (Verotta, 2010; Ionescu i Copot, 2017), oddziaływania leków (Copot et al., 2014).
3. **Modele epidemiologiczne:** epidemii niestacjonarnych (Ahmed i Elgazzar, 2007), Zapalenie wątroby typu C (Ahmed i El-Saka, 2010), Grypy A(H1N1) (González-Parra et al., 2013), HIV (Arafa et al., 2014).
4. **Materiały lepkosprężyste i polimery:** zachowania materiałów lepkosprężyste (Torvik i Bagley, 1984; Adolfsson et al., 2005), mechanicznych układów lepkosprężystych (Lazopoulos i Karaoulanis, 2016), modelowanie żeli spożywczych (Faber et al., 2017a,b), modelowanie polimerów (Caponetto et al., 2013).
5. **Ekonomia i rynki finansowe:** wycena opcji (Song, 2017), migracja zarobkowa (Balci, 2017), ekonomia finansowa (Fallahgoul et al., 2016), modele zmienności (Mendes, 2008), modele wzrostów ekonomicznych (Tejado et al., 2015), ciągłe rynki finansowe (Scalas et al., 2000).

6. **Rozprzestrzenianie się zanieczyszczeń i zanieczyszczeń:** dyspersja zanieczyszczeń w atmosferze (Goulart et al., 2017), zmian klimatycznych (Zhang et al., 2017b), podziemne rozprzestrzenianie się zanieczyszczeń (Zhang et al., 2017a), modele zmian ukształtowania terenów (Schumer et al., 2009).

1.4. Układ i zakres pracy

Rozprawa została podzielona na siedem rozdziałów zasadniczych w których przedstawiono podstawy teoretyczne dotyczące rachunku różniczkowego niecałkowitego rzędu. Opisano w niej również znane sposoby implementacji tej klasy układów wraz z metodami opracowanymi przez autora rozprawy - czasowa metoda Oustaloupa oraz równoległa metoda Oustaloupa. Przedstawiono rodzaje stosowanych obecnie układów niecałkowitego rzędu. Zaprezentowano teorię dotyczącą układów wbudowanych w szczególności bazujących na systemach czasu rzeczywistego. Opis metody implementacji opracowanych aproksymacji na układy wbudowane i sprawdzenie ich poprawności na platformach sprzętowych Arduino Uno oraz STM32F0-DISCOVERY. W następnym rozdziale rozprawy opisano eksperymenty weryfikujące poprawność zaproponowanych implementacji metod aproksymacji układów niecałkowitego rzędu poprzez ich realizację na laboratoryjnych układach nagrzewnicy powietrza i lewitacji magnetycznej. Ostatni rozdział zasadniczy zawiera podsumowanie osiągniętych rezultatów oraz opis dalszych możliwych kierunków rozwoju. Dodatkowo na końcu rozprawy znajduje się spis tabel oraz ilustracji zawartych w rozprawie. Rozprawę zamyka spis bibliograficzny zawierający 192 publikacje.

W rozdziałach rozprawy poruszono następujące zagadnienia:

- **Rozdział 1.** zawiera opis zarysu historycznego dotyczącego rachunku pochodnej niecałkowitego rzędu. Zaprezentowano w nim również cele i tezę pracy oraz przedstawiono motywację do podjęcia tego tematu wraz z przeglądem literatury.
- **Rozdział 2.** opisuje wstęp do aparatu matematycznego wykorzystywanego w rachunku pochodnej niecałkowitego rzędu. W szczególności podano definicję funkcji specjalnych: funkcji gamma, operatora newtona dla liczb rzeczywistych oraz funkcji jedno- i dwu-parametrycznej Mittag-Lefflera. Dla jasności podano definicję operatora różniczko-całki niecałkowitego rzędu oraz definicję pochodnych Grünwalda-Letnikova, Riemanna-Liouville'a, Caputo. Następnie w rozdziale znajduje się opis transformaty Laplace'a dla układów niecałkowitego rzędu. Rozdział kończy opis metod dyskretyzacji wykorzystanych w eksperymentach numerycznych.
- **Rozdział 3.** przedstawia przegląd wybranych metod aproksymacji układów niecałkowitego rzędu. Jako pierwszą zaprezentowano w nim metodę rozwiązania układu równań liniowych niecałkowitego rzędu z krótką pamięcią. Następnie opisano aprok-

symacje oparte o Continued Fraction Expansion (CFE). Kolejnymi opisanymi metodami implementacji są aproksymacji Charefa, metoda funkcji dystrybucji, Power Series Expansion (PSE) oraz metodę LIRA. Ostatni podrozdział skupia się na opisie własności klasycznej metody Oustaloupa oraz opracowanych przez autora metod bazujących na niej - czasowa metoda Oustaloupa i równoległa metoda Oustaloupa.

- **Rozdział 4.** stanowi opis wybranych struktur dynamicznych bazujących na pochodnej niecałkowitego rzędu. Opisane elementy to regulator niecałkowitego rzędu, filtry dolno- i górnoprzepustowe oraz filtr drugiego rzędu oraz filtru pasmowo przepustowego. Ich własności zostały zobrazowane poprzez charakterystyki bodego.
- **Rozdział 5.** zawiera przedstawienie wybranych systemów wbudowanych użytych do testów zaproponowanych metod aproksymacji - Arduino Uno oraz STM32F0-DISCOVERY. Opisano w nim teorię systemów wbudowanych i systemów czasu rzeczywistego. Przedstawiono algorytmy implementacji układów niecałkowitego rzędu na układy cyfrowe wraz z ich analizą. Rozdział kończy się opisem eksperymentów weryfikujących poprawność zaproponowanych metod aproksymacji i ich implementacji. Na platformie zaimplementowano regulator ułamkowy PID oraz ułamkowy filtr drugiego rzędu, otrzymane wyniki zostały omówione w formie porównania z referencyjną implementacją w środowisku Matlab Simulink. Na układzie STM32F0-DISCOVERY zaimplementowano filtr drugiego rzędu jako układ czasy rzeczywistego dla wybranych rzędów aproksymacji oraz częstotliwości próbkowania. Wyniki zostały opracowane w formie opracowania statystycznego opisującego zachowanie się błędu ze względu na wybrane parametry filtra.
- **Rozdział 6.** opisuje metod wyznaczania nastaw i implementacji regulatorów ułamkowego rzędu. Pierwszym opisanym systemem dla którego opracowano regulator ułamkowym jest laboratoryjny system nagrzewnicy powietrznej. Dla tego układu dobór nastaw regulatora odbył się poprzez przeprowadzenie optymalizacji globalnej za pomocą metody symulowanego wyżarzania. Do implementacji regulatora niecałkowitego rzędu użyto metody równoległej Oustaloupa a wyniki regulacji zostały porównane z klasycznym regulatorem PID. Drugim systemem rozważanym w tym rozdziale jest laboratoryjny układ lewitacji magnetycznej. W tym przypadku dobór regulatora odbywał się poprzez linearyzację układu w otoczeniu punktu pracy i dobraniu współczynników tak by pozostał stabilny. Regulator dla tego zadania sterowania został zaimplementowany przy użyciu czasowej metody Oustaloupa. Na końcu rozdziału opisano również porównanie implementacji regulatora ułamkowego rzędu przy użyciu metody redukcji rzędów układu z metodą implementacji opartej o czasową metodę Oustaloupa.

- **Rozdział 7.** zawiera podsumowanie rezultatów otrzymanych w trakcie przygotowania rozprawy oraz opis zidentyfikowanych dalszych kierunków rozwoju.

2. Wstęp matematyczny

Celem tego rozdziału jest wprowadzenie niezbędnych pojęć matematycznych dla uporządkowania informacji na temat rachunku różniczkowego niecałkowitego rzędu oraz metod wykorzystywanych w pracy do ich realizacji na układach cyfrowych.

2.1. Funkcje specjalne

Na początku rozważań wprowadzimy definicję funkcji gamma, zwanej również gammą Eulera (Davis, 1959):

$$\Gamma(m) = \int_0^{\infty} e^{-u} u^{m-1} du \quad (2.1)$$

Funkcję tą można traktować jako uogólnienie silni na liczby rzeczywiste:

$$\Gamma(n) = (n - 1)! \quad (2.2)$$

Do dalszych rozważań niezbędne jest wprowadzenie definicji tak zwanego uogólnionego symbolu Newtona, (Coolidge, 1949):

$$\binom{n}{k} = \frac{n(n-1) \cdots (n-k+1)}{k!} = \frac{(n)_k}{k!} \quad (2.3)$$

Kolejnymi funkcjami stosowanymi w rachunku różniczkowym niecałkowitego rzędu są jedno- i dwu- parametryczna funkcja Mittag-Lefflera. Funkcja Mittag-Lefflera jedno-parametryczna została zdefiniowana w pracy Mittag-Leffler (1905) w następujący sposób:

$$E_{\alpha}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + 1)} \quad (2.4)$$

Następnie Humbert i Agarwal (1953) wprowadzili definicję funkcji dwu-parametrycznej. Funkcja ta jest zdefiniowana w następujący sposób:

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\alpha + \beta)} \quad (2.5)$$

gdzie $\alpha, \beta > 0$. Funkcja ta przyjmuje następujące postacie dla określonych wartości parametrów, (Podlubny, 1999, str. 17):

$$\begin{aligned} E_{1,1}(z) &= e^z, & E_{1,2}(z) &= \frac{e^z - 1}{z}, \\ E_{2,1}(z) &= \cosh(\sqrt{z}), & E_{2,1}(-z^2) &= \cos(z), \\ E_{0.5,1}(\sqrt{z}) &= \frac{2}{\sqrt{\pi}} e^{-z} \operatorname{erfc}(-\sqrt{z}). \end{aligned}$$

Funkcje te są wykorzystywane przy rozwiązywaniu układów równań różniczkowych niecałkowitego rzędu postaci:

$${}_a D_t^\alpha x(t) + bx(t) = u(t)$$

przy podaniu warunków początkowych.

2.2. Definicja operatora różniczko-całki niecałkowitego rzędu

Rachunek różniczkowy niecałkowitego rzędu jest uogólnieniem operacji różniczkowania i całkowania poprzez operator ${}_a D_t^\alpha$, gdzie a i t są wartościami granicznymi dla czasu a $\alpha \in \mathbb{R}$. Ciągły operator różniczko-całki definiujemy jako (Petras, 2011b, str. 9):

$${}_a D_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \alpha > 0, \\ 1 & \alpha = 0, \\ \int_a^t (d\tau)^{-\alpha} & \alpha < 0 \end{cases} \quad (2.6)$$

W historii istnienia rachunku różniczkowego niecałkowitego rzędu powstało wiele definicji operatora różniczko-całki, opracowanych przez znakomitych matematyków, między innymi: Weyl, Fourier, Cauchy, Abel, Nishimoto. Jednak najczęściej stosowanymi obecnie definicjami są definicje: Grünwalda-Letnikova (GL), Riemann-Liouville (RL) oraz Caputo (C). W rozprawie tej skupiono się właśnie na tych trzech definicjach które przy zerowych warunkach początkowych wszystkie trzy definicje operatora różniczko-całki są sobie równoważne.

2.3. Definicja całki i różniczki Grünwalda-Letnikova (GL)

Niech będzie dana funkcja $f(t)$, której pierwszą pochodną możemy zdefiniować jako (Podlubny, 1998, str. 43):

$$\frac{df(t)}{dt} \equiv f'(t) = \lim_{h \rightarrow 0} \frac{f(t) - f(t-h)}{h} \quad (2.7)$$

Używając dwukrotnie (2.7) na funkcji $f(t)$ otrzymujemy pochodną drugiego stopnia tej funkcji postaci:

$$\begin{aligned} \frac{d^2 f(t)}{d^2 t} &\equiv f''(t) = \lim_{h \rightarrow 0} \frac{f'(t) - f'(t-h)}{h} \\ &= \lim_{h \rightarrow 0} \frac{1}{h} \left\{ \frac{f(t) - f(t-h)}{h} - \frac{f(t-h) - f(t-2h)}{h} \right\} = \lim_{h \rightarrow 0} \frac{f(t) - 2f(t-h) + f(t-2h)}{h^2}. \end{aligned} \quad (2.8)$$

Korzystając z (2.7) i (2.8) możemy podać wzór na pochodną trzeciego rzędu funkcji $f(t)$:

$$\frac{d^3 f(t)}{d^3 t} \equiv f'''(t) = \lim_{h \rightarrow 0} \frac{f(t) - 3f(t-h) + 3f(t-2h) - f(t-3h)}{h^3} \quad (2.9)$$

Bazując na powyższym rozumowaniu możemy podać ogólne równanie na pochodną n -tego rzędu funkcji $f(t)$ dla $n \in \mathbb{N}$ i $j > n$:

$$\frac{d^n f(t)}{d^n t} \equiv f^n(t) = \lim_{h \rightarrow 0} h^{-n} \sum_{j=0}^n (-1)^j \binom{n}{j} f(t-jh) \quad (2.10)$$

Jeżeli rozważane jest $-n$, dwumian przyjmuje następujące wartości:

$$\binom{-n}{j} = \frac{-n(-n-1)(-n-2) \cdots (-n-j+1)}{j!} = (-1)^j \begin{bmatrix} n \\ j \end{bmatrix} = (-1)^j \frac{n(n+1) \cdots (n+j-1)}{j!} \quad (2.11)$$

Jeżeli do równania (2.10) podstawimy $-n$, to otrzymujemy:

$$\frac{d^{-n} f(t)}{d^{-n} t} \equiv f^{-n}(t) = \lim_{h \rightarrow 0} h^{-n} \sum_{j=0}^n \begin{bmatrix} n \\ j \end{bmatrix} f(t-jh) \quad (2.12)$$

Bazując na równaniach (2.10)-(2.12) możemy zapisać równanie operatora różniczko-całki dla rzędu $\alpha \in \mathbb{R}$:

$$D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t - jh) \quad (2.13)$$

Korzystając z zależności (2.2) możemy zapisać następującą równość:

$$\binom{\alpha}{j} = \frac{\alpha!}{j!(\alpha-j)!} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \quad (2.14)$$

Jeżeli rozważymy dodanie ograniczenia $n = \frac{t-a}{h}$, gdzie a jest stałą liczbą rzeczywistą, to dostajemy formułę na operator różniczkowo-całkowy niecałkowitego rzędu Grünwalda-Letnikova postaci:

$${}_a^{GL}D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t - jh) \quad (2.15)$$

2.4. Definicja całki i różniczki Riemanna-Liouville'a (RL)

Do wyprowadzenia operatora różniczko-całki Riemann-Liouville należy rozważyć n krotną całkę funkcji $f(t)$, dla $n \in \mathbb{N}$, zdefiniowaną w następujący sposób (Petras, 2011b, str. 11):

$$\underbrace{\int_a^t \int_a^{t_n} \int_a^{t_{n-1}} \cdots \int_a^{t_3} \int_a^{t_2} f(t_1) dt_1 dt_2 \cdots dt_{n-1} dt_n}_{n\text{-krotna}} = \frac{1}{\Gamma(n)} \int_a^t \frac{f(\tau)}{t-\tau} {}_{1-n} \quad (2.16)$$

Całkę niecałkowitego rzędu $\alpha \in \mathbb{R}$ i $\alpha < 0$ dla funkcji $f(t)$ można zatem wyrazić poprzez równanie (2.16) w następujący sposób:

$${}_a I_t^\alpha = {}_a D_t^{-\alpha} f(t) = \frac{1}{\Gamma(-\alpha)} \int_a^t \frac{f(\tau)}{t-\tau} {}_{\alpha+1} \quad (2.17)$$

Bazując na (2.17) możemy podać definicję różniczko-całki RL postaci:

$${}_a^{RL}D_t^\alpha f(t) = \frac{1}{\Gamma(\alpha-n)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \quad (2.18)$$

Zarówno w przypadku definicji GL jak i RL do rozwiązania układu równań różniczkowych niecałkowitego rzędu należy podać warunki początkowe dla pochodnych całkowitych

jak i ułamkowych (Herrmann, 2011, str. 14).

2.5. Definicja różniczki Caputo (C)

Caputo (1967) podał definicję różniczki niecałkowitego rzędu następującej postaci:

$${}^C D_t^\alpha f(t) = \frac{1}{\Gamma(\alpha - n)} \int_a^t \frac{f^{(n)}(\tau)}{(t - \tau)^{r-n+1}} d\tau \quad (2.19)$$

przy warunku początkowym:

$$f^{(n)}(0) = \frac{df}{dt} \quad (2.20)$$

dla $0 \leq n - 1 < \alpha < n$ i $n \in \mathbb{N}$. Jak widać warunki początkowe dla równań różniczkowych niecałkowitego rzędu z różniczką Caputo mają tę samą postać co w przypadku równań różniczkowych całkowitego rzędu. Jest to zaletą, ponieważ w trakcie formułowania i rozwiązywania problemu nie wymagają definicji wartości początkowych dla ułamkowych pochodnych.

Zauważmy również, że między operatorem Riemann-Liouville a operatorem Caputo zachodzi następująca zależność:

$${}^{RL} D_t^\alpha f(t) = {}^C D_t^\alpha f(t) + \sum_{k=0}^{n-1} \frac{(t-a)^{k-\alpha}}{\Gamma(k-\alpha+a)} f^{(k)}(a) \quad (2.21)$$

dla $f^{(k)}(0) = 0$ przy $k = 0, 1, \dots, n-1$, gdzie $0 \leq n - 1 < \alpha < n$ i $n \in \mathbb{N}$ (Li et al., 2011).

2.6. Transformata Laplace'a układów ułamkowych

W analizie zachowań dynamiki układów rzędu całkowitego powszechnie wykorzystuje się opis transmitancyjny układów w oparciu o transformatę Laplace'a. Ponieważ w przypadku zerowych warunków początkowych różnice pomiędzy transformatą Laplace'a w układach całkowitego rzędu a niecałkowitego są niewielkie. Zaznaczyć należy, że dla przypadku gdy warunki początkowe są różne od zerowych analiza w domenie Laplace'a znacznie się komplikuje. W sekcji tej zostaną przypomniane podstawowe informacje na ten temat. Dodatkowo większość metod aproksymacji układów niecałkowitego rzędu które można zaimplementować podane są w postaci transmitancji (Monje et al., 2010, str. 12).

2.6.1. Ogólne własności transformaty Laplace'a

Funkcja $F(s)$ gdzie s jest liczbą zespoloną, zdefiniowana przez:

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} e^{-st} f(t) dt \quad (2.22)$$

nazywamy transformatą Laplace'a funkcji $f(t)$. Całka opisana równaniem (2.22) dla funkcji $f(t)$ jest zbieżna jeżeli istnieje ograniczenie wykładnicze funkcji pod całkowej, czyli istnieją takie parametry M, β, T , że prawdziwa jest następująca nierówność (Podlubny, 1998, str. 103):

$$|f(t)| < M e^{\beta t}, \text{ dla } t > T \quad (2.23)$$

Odwrotną transformatę Laplace'a dla $F(s)$ możemy zdefiniować jako:

$$f(t) = \mathcal{L}^{-1} F(s) = \int_{c-j\infty}^{c+j\infty} e^{st} F(s) ds, \quad c = \operatorname{Re}(s) > c_0, \quad (2.24)$$

gdzie c_0 jest liczbą rzeczywistą taką, że wszystkie punkty osobliwe funkcji podcałkowej (2.22) leżą w lewej półpłaszczyźnie.

Bazując na definicji odwrotnej transformaty Laplace'a (2.24) łatwo zauważyć, że splot dwóch funkcji $f(t)$ i $g(t)$ zdefiniowany jako:

$$f(t) * g(t) = \int_0^t f(t-\tau) g(\tau) d\tau = \int_0^t f(\tau) g(t-\tau) d\tau \quad (2.25)$$

jeżeli transformaty Laplace'a funkcji $f(t)$ i $g(t)$ istnieją, to prawdziwa jest równość:

$$\mathcal{L}\{f(t) * g(t)\} = F(s)G(s). \quad (2.26)$$

Ostatnia omawiana własność transformaty Laplace'a przydatna w dalszych rozważaniach dotyczyć będzie transformaty dla pochodnej stopnia n -tego funkcji $f(t)$:

$$\mathcal{L}\{f^n(t)\} = s^n F(s) - \sum_{k=0}^{n-1} s^{n-k-1} f^k(0) \quad (2.27)$$

2.6.2. Transformata Laplace'a dla operatora niecałkowitego rzędu

Transformata Laplace'a dla operatora niecałkowitego rzędu Reimanna-Liouville dla $p > 0$ zdefiniowanej przez równanie (2.16) dla postaci splotowej funkcji $g(t) = t^{p-1}$ i

$f(t)$ (Petras, 2011b, str. 13):

$${}_0D_t^{-p}f(t) = \frac{1}{\Gamma(p)} \int_0^t (t-\tau)^{p-1} f(\tau) d\tau = t^{p-1} * f(t) \quad (2.28)$$

i dla t^{p-1} ma postać (Podlubny, 1999, str. 105):

$$G(s) = \mathcal{L}\{t^{p-1}\} = \Gamma(p)s^{-p}. \quad (2.29)$$

Stąd używając wzoru na transformatę Laplace'a splotu (2.26) otrzymujemy transformację Laplace'a dla operatorów Grünwalda-Letnikova, Riemann-Liouville i Caputo przy zerowych warunkach początkowych:

$$\mathcal{L}\{{}_0D_t^{-p}f(t)\} = s^{-p}F(s) \quad (2.30)$$

2.6.3. Metody dyskretyzacji

W dalszej części prac zostaną użyte metody dyskretyzacji Tustina i Eulera do dyskretyzacji opracowanych aproksymacji w postaci:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t) + Du(t) \end{aligned} \quad (2.31)$$

Zasadniczo metody te nie zachowują zmiennych stanów systemu bazowego, jednak nowy stan zachowuje informacje o wejściu i stanie bazowym (Middleton i Goodwin, 1990, str. 472). Po dyskretyzacji system przyjmie postać (dla uproszczenia $u(Tk)$ jest zapisywane jako $u(k)$)

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{\Phi}\mathbf{w}(k) + \mathbf{\Gamma}u(k) \\ y(k) &= \mathbf{H}\mathbf{w}(k) + Ju(k) \end{aligned} \quad (2.32)$$

gdzie dla metody Eulera:

$$\mathbf{w} = (\mathbf{I} - \mathbf{A}T)\mathbf{x} - T\mathbf{B}u \quad (2.33)$$

$$\mathbf{\Phi} = (\mathbf{I} - \mathbf{A}T)^{-1} \quad (2.34)$$

$$\mathbf{\Gamma} = (\mathbf{I} - \mathbf{A}T)^{-1}\mathbf{B}T \quad (2.35)$$

$$\mathbf{H} = \mathbf{C}(\mathbf{I} - \mathbf{A}T)^{-1} \quad (2.36)$$

$$J = D + \mathbf{C}(\mathbf{I} - \mathbf{A}T)^{-1}\mathbf{B}T \quad (2.37)$$

i dla metody Tustina:

$$\sqrt{T}\mathbf{w} = (\mathbf{I} - \mathbf{A}\frac{T}{2})\mathbf{x} - \frac{T}{2}\mathbf{B}u \quad (2.38)$$

$$\Phi = (\mathbf{I} + \mathbf{A}\frac{T}{2})(\mathbf{I} - \mathbf{A}\frac{T}{2})^{-1} \quad (2.39)$$

$$\Gamma = (\mathbf{I} - \mathbf{A}\frac{T}{2})^{-1}\mathbf{B}\sqrt{T} \quad (2.40)$$

$$\mathbf{H} = \sqrt{T}\mathbf{C}(\mathbf{I} - \mathbf{A}\frac{T}{2})^{-1} \quad (2.41)$$

$$J = D + \mathbf{C}(\mathbf{I} - \mathbf{A}\frac{T}{2})^{-1}\mathbf{B}\frac{T}{2} \quad (2.42)$$

Jak można zaobserwować w obu metodach macierz Φ zachowuje strukturę macierzy trójkątnej dolnej. Dlatego na diagonalnej znajdują się wartości własne dyskretnego systemu, które są dyskretyzowane indywidualnie. Zatem błędy zaokrągleń są znacznie zmniejszone. Ponadto dlatego, że metoda jest jawna, najszybsza wartość własna jest również mapowana w taki sposób że znajduje się w kole jednostkowym.

3. Cyfrowa realizacja układów niecałkowitego rzędu

Rozdział ten zawiera opis wybranych przez autora metod aproksymacji układów niecałkowitego rzędu poprzez układy rzędu całkowitego.

3.1. Rozwiązanie układu równań liniowych niecałkowitego rzędu z krótką pamięcią

Dla układów równań liniowych niecałkowitego rzędu, które mogą służyć do realizacji między innymi wielu rodzajów filtrów niecałkowitego rzędu o współmiernych rzędach pochodnych (Piątek i Baranowski, 2011), zdefiniowanych jako:

$$\begin{aligned} {}_0^C D_t^\alpha \mathbf{x}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad 0 < \alpha < 1 \\ \mathbf{x}(0) &= \mathbf{x}_0, \end{aligned} \quad (3.1)$$

Do rozwiązania numerycznego takiego typu układu równań może posłużyć następująca metoda, zob. Monje et al. (2010):

$$\mathbf{x}(t) = (\mathbf{I} - h^\alpha \mathbf{A})^{-1} \left(h^\alpha \mathbf{B}\mathbf{u}(t) - \sum_{k=1}^p c_k \mathbf{x}(t - kh) \right) \quad (3.2)$$

$$h = \frac{T}{m} \quad (3.3)$$

$$t = ph, \quad p = 0, 1, \dots, m \quad (3.4)$$

$$c_k = (-1)^k \binom{\alpha}{k} \quad (3.5)$$

W metodzie tej wraz ze wzrostem czasu t niezbędne jest zapamiętywanie coraz większej ilości próbek do obliczenia rozwiązania, innymi słowy do obliczenia $t \rightarrow \infty$ potrzebne jest nieskończenie wiele pamięci. Do rozwiązania tego problemu Podlubny (1998) (str. 203) zaproponował tak zwaną zasadę krótkiej pamięci (**ang.** *short memory principle*). Zasada ta

bazuje na obserwacji zachowania się wartości współczynników w pochodnej Grünwalda-Letnikova dla dużych wartości t , które są zależne od wartości współczynników przy $t = 0$, lub też innym warunku początkowym. Współczynniki te mają również niewielki wpływ na jakość rozwiązania. Fakt ten pozwala nam na przybliżenie rozwiązania numerycznego za pomocą wartości funkcji blisko rozważanego czasu t , czyli do obliczeń użyjemy jedynie współczynników z przedziału czasowego $[t - L, t]$, gdzie L jest długością pamięci, czyli liczba próbek która jest zapamiętywana w czasie obliczeń.

Stąd możemy zapisać:

$${}^GL D_t^\alpha f(t) \approx {}^GL D_{t-L}^\alpha f(t), \quad t > L \quad (3.6)$$

gdzie długość pamiętanych współczynników jest ograniczona przez L/h . Błąd przybliżenia przy tych założeniach dla $|f(t)| \leq M$, przy zakresie $(0 < t \leq t_1)$ jest ograniczony przez zależność:

$$\epsilon(t) = \left| {}^C D_t^\alpha f(t) - {}^C D_{t-L}^\alpha f(t) \right| \leq \frac{ML^\alpha}{|\Gamma(1-\alpha)|}, \quad L \leq t \leq t_1 \quad (3.7)$$

Zależność ta pozwala obliczyć najkrótszą długość pamięci L_ϵ przy zadanym maksymalnym błędzie ϵ i jest wyrażona przez:

$$L_\epsilon \geq \left(\frac{M}{\epsilon |\Gamma(1-\alpha)|} \right)^{1/\alpha} \quad (3.8)$$

Dla przypadku gdy wartość pochodnej α jest stała, można obliczyć współczynniki c_k w następujący sposób:

$$c_0^\alpha = 1, \quad c_k^\alpha = \left(1 - \frac{\alpha + 1}{k} \right) c_{k-1}^\alpha, \quad k = 1, 2, \dots \quad (3.9)$$

Dla zobrazowania działania tej metody rozważmy równanie różniczkowe niecałkowitego rzędu postaci:

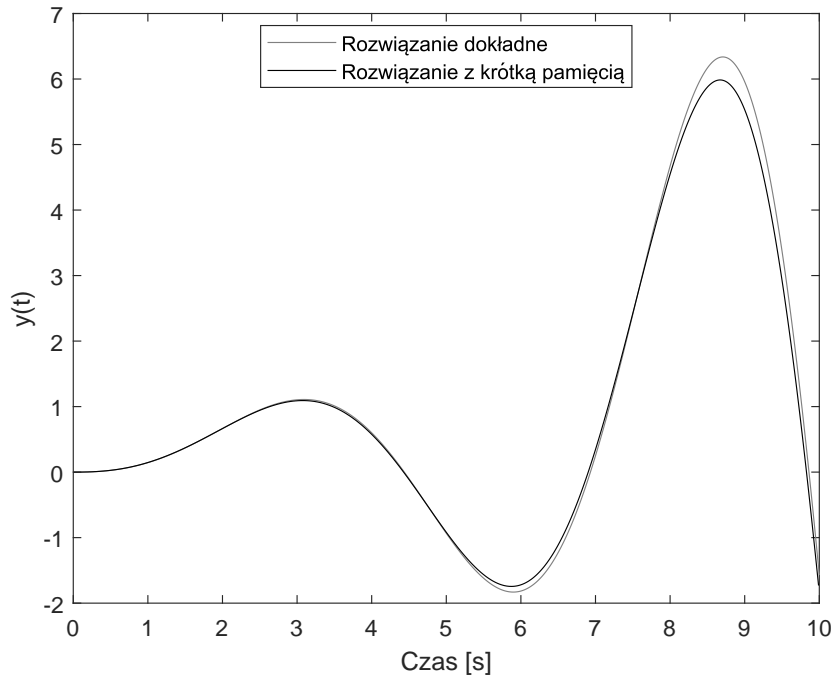
$${}^C D_t^{2.5} \mathbf{x}(t) = 2\mathbf{x}(t) + 3\mathbf{u}(t), \quad (3.10)$$

$$\mathbf{x}(0) = 0. \quad (3.11)$$

W tym przypadku rozwiązanie analityczne ma postać:

$$y(t) = \frac{t^{2.5}}{2} E_{2.5, 3.5}(-1.5t^{2.5})$$

Porównanie rozwiązań obrazuje wykres 3.1. Z łatwością można zauważyć, że metoda z krótką pamięcią już po upływie 6 sekund zaczyna w znaczący sposób różnić się od me-



Rysunek 3.1: Porównanie rozwiązania dokładnego z metodą z krótką pamięcią

tody dokładnej. Ze względu na ten fakt w dalszej części pracy nacisk zostanie położony na aproksymacje w dziedzinie Laplace'a.

3.2. Aproksymacja CFE (Continued Fraction Expansion)

Aproksymacja CFE - Continued Fraction Expansion - jest używana w procesie przybliżenia funkcji niewymiernej $f(s)$ przez funkcję wymierną. Metoda ta dla tej klasy funkcji często jest szybciej zbieżna niż metody oparte na rozwinięciu w szereg potęgowy, na przykład rozwinięcie w szereg Taylora, (Press et al., 2007). Dla operatora $G(s) = s^\alpha$ możemy zapisać następujące przybliżenie CFE:

$$G(s) \approx a_0(s) + \frac{b_1(s)}{a_1(s) + \frac{b_2(s)}{a_2(s) + \frac{b_3(s)}{a_3(s) + \dots}}} = a_0(s) + \frac{b_1(s)}{a_1(s)} + \frac{b_2(s)}{a_2(s)} + \frac{b_3(s)}{a_3(s)} + \dots \quad (3.12)$$

gdzie a_k i b_k są funkcjami wymiernymi zmiennej s lub stałymi.

Z drugiej strony funkcję nie wymierną możemy przybliżyć przy pomocy ilorazu dwóch wielomianów wymiernych otrzymanych w procesie interpolacji. Wynika to bezpośrednio z możliwości modelowania charakterystyki częstotliwościowej bazując na lokalizacji zer i biegunów transmitancji (Vinagre et al., 2000a). Bazując na tym założeniu można zapisać

metodę CFE za pomocą następującej funkcji przybliżającej $G(s)$:

$$G(s) \approx R_{i(i+1)\dots(m+1)} = \frac{P_\mu(s)}{Q_\nu(s)} = \frac{p_0 + p_1s + \dots + p_\mu s^\mu}{q_0 + q_1s + \dots + q_\nu s^\nu} \quad (3.13)$$

gdzie $m + 1 = \mu + \nu + 1$ i przybliżenie przechodzi przez punkty $(s_i, G(s_i)), \dots, (s_{i+m}, G(s_{i+m}))$.

3.2.1. Bazowa metoda aproksymacji CFE operatora różniczko-całki niecałkowitego rzędu

W artykule Roy (1967) zaproponował ogólną metodę przybliżenia funkcji $G(s) = s^{-\alpha}$ dla $\alpha \in (0, 1)$ przy pomocy szeregowego połączenia aproksymacji CFE dwóch funkcji:

$$G_h(s) = \frac{1}{(1 + sT)^\alpha} \quad (3.14)$$

$$G_l = \left(1 + \frac{1}{s}\right)^\alpha \quad (3.15)$$

gdzie $G_h(s)$ odpowiada za odwzorowanie przebiegu funkcji aproksymowanej dla wysokich częstotliwości $\omega T \gg 1$, natomiast $G_l(s)$ odwzorowuje funkcję dla częstotliwości $\omega \ll 1$.

3.2.2. Metoda Carlsona

Carlson (1960) (str. 18) zaproponował metodę przybliżenia s^α za pomocą metody Newtona do znajdowania rozwiązań równań nieliniowych. Metoda ta bazuje na następujących założeniach:

$$(H(s))^{1/\alpha} - G(s) = 0 \quad (3.16)$$

stąd

$$H(s) = G(s)^\alpha \quad (3.17)$$

dla zdefiniowanych wartości $\alpha = 1/q$, $m = q/2$ oraz warunku początkowego $H_0 = 1$. Można zapisać iteracyjną metodę przybliżania $G(s)^\alpha$ w następujący sposób:

$$H_i(s) = H_{i-1}(s) \frac{(q-m)(H_{i-1}(s))^2 + (q+m)G(s)}{(q+m)(H_{i-1}(s))^2 + (q-m)G(s)} \quad (3.18)$$

3.2.3. Metoda Matsuda

Inne podejście zostało przedstawione w artykule Matsuda i Fujii (1993). Autorzy zaproponował by operator s^α przybliżać przy pomocy wyrażenia dla znanych punktów s_k :

$$H(s) = a_0 + \frac{s - s_0}{a_1} + \frac{s - s_1}{a_2} + \frac{s - s_2}{a_3} + \dots \quad (3.19)$$

gdzie

$$a_i = v_i(s), \quad v_0(s) = H(s), \quad v_{i+1}(s) = \frac{s - s_i}{v_i(s) - a_i} \quad (3.20)$$

3.3. Metoda Charefa

Aproksymacja zaproponowana przez Charef et al. (1992) dla systemu opisanego następującą transmitancją

$$H(s) = \frac{1}{\left(1 + \frac{s}{pT}\right)^\alpha} \quad (3.21)$$

tworzy jej przybliżenie za pomocą następującej zależności:

$$G_t(s) = \frac{\prod_{i=0}^{N-1} \left(1 + \frac{s}{z_i}\right)}{\prod_{i=0}^N \left(1 + \frac{s}{p_i}\right)} \quad (3.22)$$

gdzie zera i bieguny transmitancji aproksymującej zadaną funkcję są dane następującymi zależnościami:

$$p_0 = pT\sqrt{b} \quad (3.23)$$

$$p_i = p_0(ab)^i \quad (3.24)$$

$$z_i = ap_0(ab)^i \quad (3.25)$$

Współczynniki a oraz b są obliczane w taki sposób by maksymalne odchylenie od oryginalnej amplitudy w dziedzinie częstotliwości wynosiło y dB, osiąga się to za pomocą zależności:

$$a = 10^{\frac{y}{10(1-\alpha)}}, \quad (3.26)$$

$$b = 10^{\frac{y}{10\alpha}}. \quad (3.27)$$

Natomiast rząd aproksymacji N jest określony za pomocą zależności:

$$N = \left\lfloor \frac{\log(\omega_{max}T)}{\log(ab)} \right\rfloor + 1 \quad (3.28)$$

gdzie ω_{max} jest maksymalną częstotliwością dla której aproksymacja będzie używana.

3.4. Metoda Djouambi

Djouambi et al. (2007) zaproponowali aproksymację dla członu:

$$G_f(s) = \frac{1}{(T_s)^\alpha + 1} \quad (3.29)$$

gdzie T to stała czasowa a $\alpha \in R$ jest rzędem pochodnej, poprzez następującą formułę:

$$\widehat{G}_N(s) = \sum_{i=1}^{2N-1} \frac{h(\tau_i)}{\tau_i s + 1} \quad (3.30)$$

gdzie N jest rzędem aproksymacji natomiast $h(\tau_i)$ jest dane formułą:

$$h(\tau_i) = \frac{1}{2\pi} \frac{\sin((1-\alpha)\pi)}{\cosh\left(\alpha \log\left(\frac{\tau}{T}\right)\right) - \cos((1-\alpha)\pi)} \quad (3.31)$$

gdzie:

$$\tau_i = T\lambda^{(N-i)} \quad (3.32)$$

Dla wyznaczenia λ należy rozwiązać zagadnienie min-max postaci:

$$\lambda = \arg \min \left\{ \max_{\omega \in [\omega_b, \omega_h]} \{|G_N(j\omega) - G(j\omega)|\} \right\} \quad (3.33)$$

gdzie $\omega \in [\omega_b, \omega_h]$ to przedział częstotliwości na którym aproksymujemy człon podstawowy.

Jednak metoda ta ze względu na aproksymowanie konkretnej klasy systemu jest rzadko używana.

3.5. Metoda rozwinięcia w szereg potęgowy - Power Series Expansion (PSE)

Poza metodami aproksymującymi ciągłe systemy niecałkowitego rzędu istnieją metody które korzystają z własności, iż każda funkcja $f(t)$ jest aproksymowana poprzez funkcję dys-

kretną $f(nh)$, gdzie h to długość kroku dyskretyzacji a n numer próbki. Gorenflo (1997) podał formalną definicję aproksymacji pochodnej niecałkowitego rzędu przy pomocy układu dyskretnego w następującej formie:

$$y_n(nh) = h^{-\alpha} (\omega(\zeta^{-1}))^\alpha f_h(nh) \quad (3.34)$$

gdzie ζ^{-1} jest operatorem przesunięcia natomiast $\omega(\zeta^{-1})$ jest funkcją generującą. Przyjęta funkcja generująca i operator przesunięcia w sposób jednoznaczny określają formę aproksymacji jak i jej współczynniki.

Metoda PSE bazuje na opisanym powyżej założeniu. Używając funkcji generującej opartej na schemacie różnicowym dla układów niecałkowitego rzędu z krokiem wstecz:

$$\omega(z^{-1}) = (1 - z^{-1}) \quad (3.35)$$

i rozwinięcia w szereg potęgowy wyrażenia $(1 - z^{-1})^\alpha$ jako operatora przesunięcia, otrzymujemy aproksymację dyskretną dla różniczki Grünwalda-Letnikova postaci:

$$\Delta_T^\alpha f(nT) = T^{-\alpha} \sum_{k=0}^{\infty} (-1)^k \binom{\alpha}{k} f((n-k)T) \quad (3.36)$$

gdzie T to okres dyskretyzacji.

3.6. Aproksymacja LIRA

Metoda aproksymacji poprzez przybliżenie odpowiedzi impulsowej układu niecałkowitego rzędu, została zaproponowana przez Bania i Baranowski (2013). Metoda ta może być analizowana dla równań typu (Podlubny, 1998, str. 122):

$${}^C D_t^{\sigma_n} x(t) + \left(\sum_{j=1}^{n-1} p_{n-j} {}^C D_t^{\sigma_{n-j}} x(t) \right) + p_0 x(t) = \quad (3.37)$$

$$q_m {}^C D_t^{\gamma_m} u(t) + \left(\sum_{j=1}^{m-1} q_{m-j} {}^C D_t^{\gamma_{m-j}} u(t) \right) + q_0 u(t) \quad (3.38)$$

gdzie $j \leq \sigma_j \leq j+1$, $j = 1, 2, \dots, n$, $j \leq \gamma_j \leq j+1$, $j = 1, 2, \dots, m$, $p_j, q_j \in \mathbb{R}$, przy zerowych warunkach początkowych. Przyjmijmy również, że $|u(t)| \leq u_{max}$ for $t \geq 0$ i $u(t) = 0$ dla $t < 0$. Ponieważ zakłada się, że warunki początkowe wynoszą zero, operator różniczkowy może być typu Riemanna-Liouville'a lub Caputo.

Teraz możliwe jest zastosowanie transformacji Laplace'a do równania (3.38)

$$\hat{x} = \hat{g}\hat{u} \quad (3.39)$$

gdzie

$$\hat{g}(s) = \frac{q_m s^{\gamma_m} + q_{m-1} s^{\gamma_{m-1}} + \dots + q_0}{s^{\sigma_n} + p_{n-1} s^{\sigma_{n-1}} + \dots + p_0} \quad (3.40)$$

Funkcja \hat{g} przyjmuje postać transformaty Laplace'a pewnej funkcji $g : [0, \infty) \rightarrow \mathbb{R}$, która spełnia $g \in \mathcal{L}_1(0, \infty) \cup \mathcal{L}_2(0, \infty)$. Odwrotna transformata dla (3.39) określa rozwiązania dla (3.38) podane przez splot:

$$x(t) = u * g = \int_0^t u(t - \theta) g(\theta) d\theta \quad (3.41)$$

3.6.1. Skończenie wymiarowa aproksymacja układu niecałkowitego rzędu

Można pokazać, że rozwiązanie (3.41) równania (3.38) może być przybliżone za pomocą rozwiązania układu równań n liniowych równań różniczkowych (Bania i Baranowski, 2013). Aproksymacja opiera się na bazie ortonormalnej w przestrzeni $\mathcal{L}_2(0, \infty)$

$$e_k(\theta, \mu) = \sqrt{2\mu} e^{-\mu\theta} L_k(2\mu\theta), \quad k = 0, 1, 2, \dots \quad (3.42)$$

gdzie μ jest dowolną stałą dodatnią, a L_k jest k -tym wielomianem Laguerre'a postaci:

$$L_k(z) = \frac{e^z}{k!} \frac{d^k}{dz^k} (e^{-z} z^k) \quad (3.43)$$

Twierdzenie 3.1 podaje warunki, które muszą zostać spełnione, aby znaleźć aproksymację układu przy minimalnym błędzie.

Twierdzenie 3.1 *Jeżeli $g \in \mathcal{L}_1(0, \infty) \cup \mathcal{L}_2(0, \infty)$ i $|u(t)| \leq u_{max}$ wtedy:*

1. *Rozwiązanie (3.38) może być aproksymowane przez:*

$$x_n(t) = \sum_{k=0}^n \beta_k \xi_k(t) \quad (3.44)$$

gdzie funkcje $\xi_k(t) : [0, \infty) \rightarrow \mathbb{R}$ są rozwiązaniem układu:

$$\dot{\xi}_k = -\mu \xi_k - 2\mu \sum_{i=0}^{k-1} \xi_i + \sqrt{2\mu} u \quad (3.45)$$

$$\xi_k(0) = 0, \quad k = 0, 1, 2, \dots, n$$

i

$$\beta_k = \int_0^{\infty} g(\theta) e_k(\theta, \mu) d\theta. \quad (3.46)$$

2. Dla każdego $\varepsilon > 0$ istnieje liczba n_0 zależna od g , ε i u_{max} , że błąd aproksymacji $e_n(t) = x(t) - x_n(t)$ spełnia nierówność:

$$|e_n(t)| < \varepsilon \quad (3.47)$$

for all $n \geq n_0$ and $t \geq 0$

Dowód 3.1 Dowód powyższego twierdzenia opracowali Bania i Baranowski (2013).

Formuła (3.46) nie jest wymagana do implementacji metody w środowiskach numerycznych. W Bania i Baranowski (2013) autorzy przedstawiają formułę rekurencyjną na wyliczenie współczynników w następującej formie:

$$\beta_k = \frac{\sqrt{2\mu}}{k!} \sum_{j=0}^k \binom{k}{j} c_j^k(\mu) \hat{g}^{(k-j)}(\mu) \quad (3.48)$$

gdzie

$$c_j^k = \frac{k-j+1}{2\mu} c_{j-1}^k, \quad c_0^k(\mu) = (2\mu)^k, \quad j = 1, 2, \dots, k \quad (3.49)$$

$$\text{i } \hat{g}^{(j)}(s) = \frac{d^j \hat{g}(s)}{ds^j}.$$

3.7. Aproksymacja Oustaloupa

Jedną z najczęściej wykorzystywanych metod do realizacji układów niecałkowitego rzędu jest metoda Oustaloupa (Oustaloup et al., 2000). Aproksymacja ta przybliża transmitancję $G(s) = s^\alpha$ za pomocą formuły:

$$G_t(s) = K \prod_{i=1}^N \frac{s + \omega'_i}{s + \omega_i} \quad (3.50)$$

gdzie:

$$\omega'_i = \omega_b \omega_u^{(2i-1-\alpha)/N} \quad (3.51)$$

$$\omega_i = \omega_b \omega_u^{(2i-1+\alpha)/N} \quad (3.52)$$

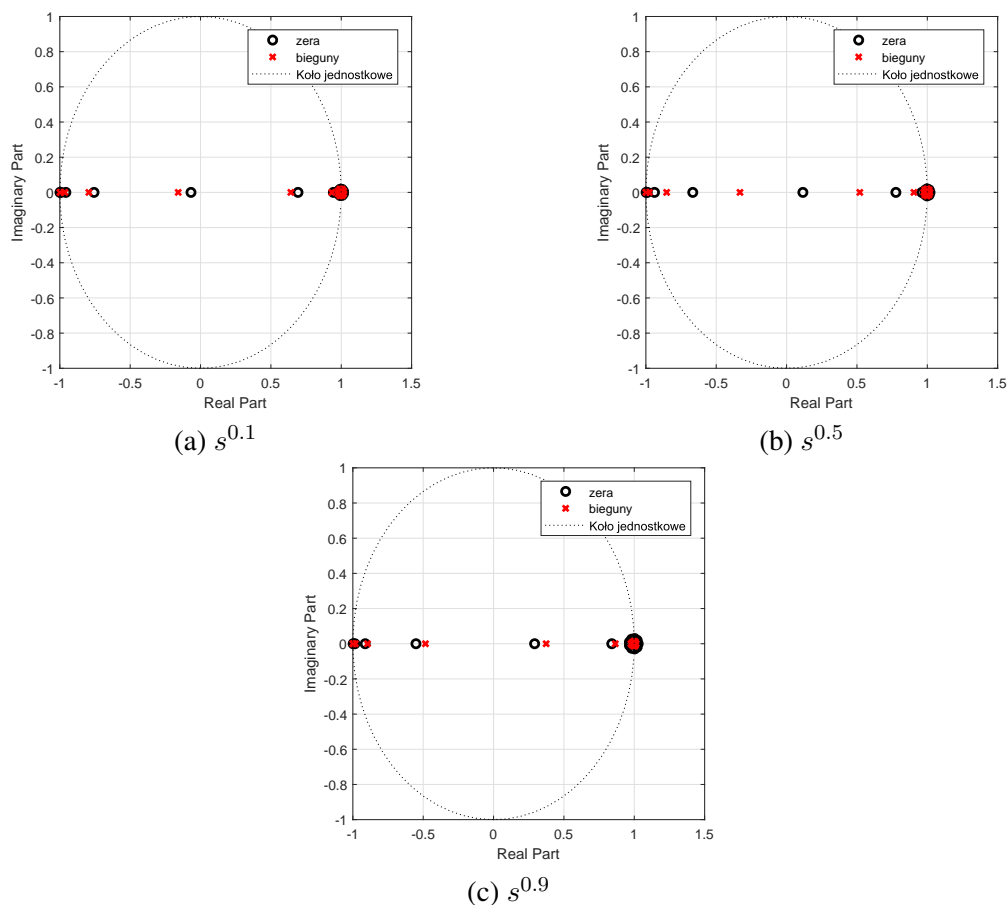
$$K = \omega_h^\alpha \quad (3.53)$$

$$\omega_u = \sqrt{\frac{\omega_h}{\omega_b}} \quad (3.54)$$

Aproksymacja działa dla częstotliwości z przedziału $\omega \in [\omega_b, \omega_h]$ i N jest rzędem aproksymacji. Aproksymacja ta realizuje przybliżenie systemu s^α poprzez serię asymptotycznie stabilnych linowych systemów pierwszego rzędu. Jak można zaobserwować wybór szerokiego

pasmo częstotliwości skutkuje dużymi wartościami ω_u i wysokimi rzędami N , co skutkuje odstępami między biegunami w zakresie od bliskich $-\omega_h$ do bardzo bliskich $-\omega_b$. Odstępami te nie są liniowe (istnieje grupa w pobliżu $-\omega_b$), co w rezultacie powoduje problemy z zachowaniem asymptotycznej stabilności układów w procesie dyskretyzacji na układach liczących. Dzieje się tak ponieważ wszystkie liczby zmiennoprzecinkowe pojedynczej lub podwójnej precyzji używane w dzisiejszych systemach informatycznych określone w normie IEEE754 (zobacz IEEE (1985)) posiadają ograniczenia związane z reprezentacją liczb, kolejnością wykonywania działań, błędami zaokrągleń, przepełnień i niedomiarów.

W przypadku cyfrowej reprezentacji aproksymacji Oustaloupa mamy styczność z błędem powiązaniem z dodawaniem liczb zmiennoprzecinkowych, które reprezentują bieguny transmitancji, o cechach różniących się od siebie o więcej niż ilość bitów przeznaczonych na mantysy. W takim przypadku po denormalizacji mantysa będzie miała wartość 0, a liczba o mniejszym wykładniku nie wpłynie na wynik dodawania bądź odejmowania. Przypadek niestabilności numerycznej tej aproksymacji w procesie dyskretyzacji obrazują wykresy 3.2a, 3.2b i 3.2c.



Rysunek 3.2: Rozkład zer i biegunów systemów dyskretnych dla aproksymacji czasowego Oustaloupa rzędu 15

3.8. Aproksymacje oparte na aproksymacji Oustaloupa

3.8.1. Równoległa aproksymacja Oustaloupa

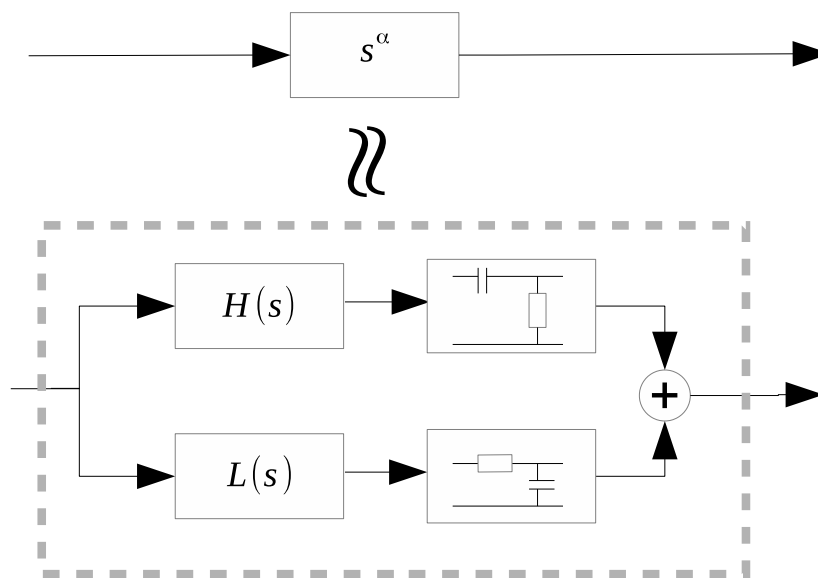
Równoległa aproksymacja Oustaloupa jest modyfikacją podstawowej Metody Oustaloupa opisywanej równaniem (3.50), pozwalającą na rozmieszczenie biegunów aproksymacji w taki sposób, by po dyskretyzacji układ pozostawał asymptotycznie stabilny.

Metoda ta opiera się na koncepcji by zamiast aproksymacji wyższego rzędu (dla $N > 5$) w całym przedziale $[\omega_b, \omega_h]$ utworzyć ją przy pomocy sumy dwóch aproksymacji dla niższych częstotliwości ($L(s)$) i wyższych częstotliwości ($H(s)$), obie rzędu $n = \lfloor N/2 \rfloor$.

$$L(j\omega) \approx j\omega^\alpha, \quad \omega \in [\omega_b, \omega_c] \quad (3.55)$$

$$H(j\omega) \approx j\omega^\alpha, \quad \omega \in [\omega_c, \omega_h] \quad (3.56)$$

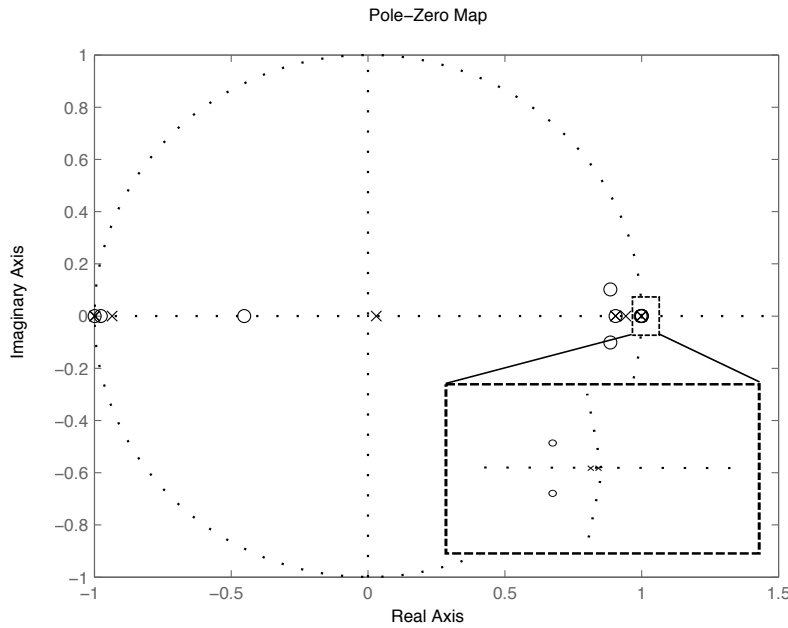
Punkt dzielący zbiór $[\omega_b, \omega_h]$ znajduje się w jego centrum ω_c . Obie te aproksymacje powinny być połączone szeregowo z filtrami dolnoprzepustowymi i górnoprzepustowymi (oba z częstotliwością odcięcia ω_c). Te połączenia szeregowo są następnie połączone równoległe. Taką konstrukcję przedstawiono na rysunku 3.4.



Rysunek 3.3: Schemat równoległej aproksymacji Oustaloupa

Ponieważ pasma niskie i wysokie są aproksymowane oddzielnie, takie równoległe połączenie jest zgodne z aproksymacją całego pasma, jednak z różnie rozmieszczonymi biegunami. Do odfiltrowania niechcianych pasm z aproksymacji $L(s)$ i $H(s)$ zalecane są klasyczne filtry pierwszego rzędu, ponieważ nie będą powodowały powstawania nowych błędów numerycznych.

Metoda ze względu na dużą złożoność komponentów niezbędnych do jej realizacji może jedynie służyć do realizacji podstawowych członów niecałkowitego rzędu.



Rysunek 3.4: Rozmieszczenie zer i biegunów w kole jednostkowym dla $s^{0.5}$ przy aproksymacji równoległej Oustaloupa

3.8.2. Czasowa aproksymacja Oustaloupa

Proponowane podejście polega na realizacji każdego bloku transmitancji (3.50) w postaci systemu przestrzeni stanów, zgodnie z zależnością opisaną w równaniu (3.57). Asymptotycznie stabilne systemy pierwszego rzędu zostaną następnie zebrane w postaci macierzowego równania różniczkowego z trójkątną dolną macierzą stanu, postaci danej przez (3.59). Taka postać umożliwi stabilniejszą numerycznie realizację dyskretyzacji tego układu. Cały proces transformacji został przedstawiony poniżej.

Przy zerowych warunkach początkowych można każdy ze członów transmitancji (3.50) zapisać jako:

$$\frac{s + \omega'_k}{s + \omega_k} \Rightarrow \begin{cases} \dot{x}_k = A_k x_k + B_k u_k \\ y_k = x_k + u_k \end{cases} \quad (3.57)$$

gdzie:

$$\begin{aligned} A_k &= -\omega_k \\ B_k &= \omega'_k - \omega_k \\ C &= 1 \\ D &= 1 \end{aligned} \quad (3.58)$$

gdzie ω'_k jest dana (3.51) a ω_k jest dana przez (3.52). Uwzględniając równania współczynników można zapisać:

$$\begin{aligned}
B_k &= \omega_{\min} \omega_u^{\frac{2k-1}{N}} (\omega_u^{-\frac{\gamma}{N}} - \omega_u^{\frac{\gamma}{N}}) \\
&= \omega_{\min} \omega_u^{\frac{2k-1}{N}} (e^{-\frac{\gamma}{N} \log \omega_u} - e^{\frac{\gamma}{N} \log \omega_u}) \\
&= -2\omega_{\min} \omega_u^{\frac{2k-1}{N}} \sinh\left(\frac{\gamma}{N} \log \omega_u\right)
\end{aligned}$$

Następnie zapisując szeregowo zależności pomiędzy wejściami i wyjściami poszczególnych układów otrzymujemy układ równań:

$$\left\{ \begin{array}{l} \dot{x}_1 = A_1 x_1 + B_1 K u \\ y_1 = x_1 + K u \\ \dot{x}_2 = A_2 x_2 + B_2 y_1 \\ y_2 = x_2 + y_1 \\ \dot{x}_3 = A_3 x_3 + B_3 y_2 \\ y_3 = x_3 + y_2 \\ \vdots \\ \dot{x}_k = A_k x_k + B_k y_{k-1} \\ y_k = x_k + y_{k-1} \end{array} \right.$$

Grupując zależności możemy zapisać wzory na \dot{x}_N i y_N :

$$\left\{ \begin{array}{l} \dot{x}_1 = A_1 x_1 + B_1 K u \\ y_1 = x_1 + K u \\ \dot{x}_2 = A_2 x_2 + B_2 (x_1 + K u) \\ y_2 = x_2 + x_1 + K u \\ \dot{x}_3 = A_3 x_3 + B_3 (x_2 + x_1 + K u) \\ y_3 = x_3 + x_2 + x_1 + K u \\ \vdots \\ \dot{x}_N = A_N x_N + B_N \left(\sum_{k=1}^{N-1} x_k + K u \right) \\ y_N = \sum_{k=1}^N x_k + K u \end{array} \right.$$

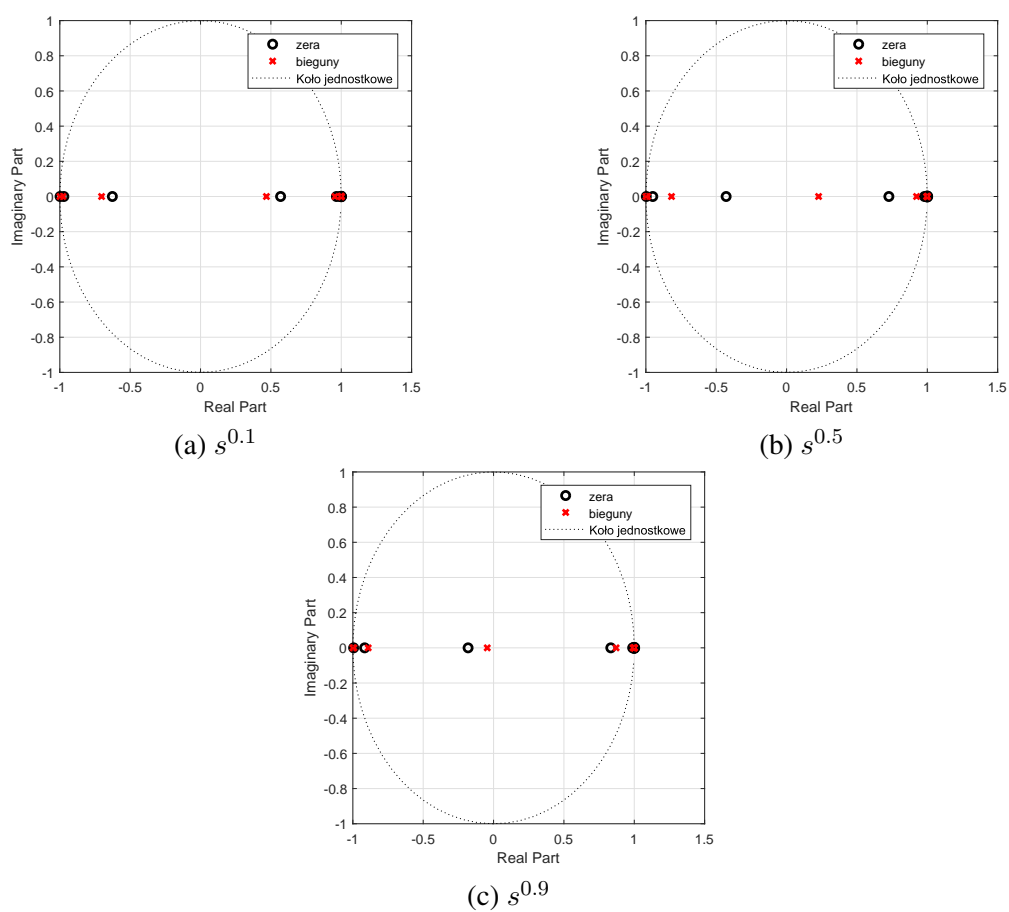
Stąd możemy zapisać aproksymację w postaci układu równań różniczkowych równoważną do transmitancji (3.50):

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} A_1 & 0 & 0 & \dots & 0 \\ B_2 & A_2 & 0 & \dots & 0 \\ B_3 & B_3 & A_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ B_N & B_N & \dots & B_N & A_N \end{bmatrix} \mathbf{x} + \begin{bmatrix} KB_1 \\ KB_2 \\ KB_3 \\ \vdots \\ KB_N \end{bmatrix} u \\ \mathbf{y} &= \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \end{bmatrix} \mathbf{x} + Ku \end{aligned} \quad (3.59)$$

lub krócej:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ y &= \mathbf{Cx} + Du \end{aligned} \quad (3.60)$$

Można od razu zauważyć, że macierz \mathbf{A} jest macierzą trójkątną dolną. Jest to niezwykle ważne w przypadku tego problemu, ponieważ wszystkie jego wartości własne (bieguny transmitancji (3.50) znajdują się na przekątnej. Zatem nie ma potrzeby wyliczania wartości własnych tego układu, co prowadziłyby do błędów numerycznych. Dlatego metody dyskretyzacji muszą zachować strukturę macierzy z układu (3.59). Dla porównania stabilności tej metody z klasyczną przedstawiono wykresy 3.5a, 3.5b i 3.5c. Jak łatwo zauważyć wszystkie zera i bieguny znajdują się w kole jednostkowym, co oznacza, że układ po dyskretyzacji w tej metodzie pozostaje asymptotycznie stabilny w przeciwieństwie do klasycznej metody aproksymacji Oustaloupa.



Rysunek 3.5: Rozkład zer i biegunów systemów dyskretnych dla aproksymacji czasowego Oustaloupa rzędu 15

4. Stosowane układy niecałkowitego rzędu

Układy czasowe bazujące na definicji pochodnej niecałkowitego rzędu stały się w ostatnich latach popularnym tematem badań ze względu na dużą elastyczność w procesie kształtowania ich charakterystyki częstotliwościowej. Klasyczne układy regulacji i filtracji umożliwiają tworzenie filtrów bazujących na spadku 20dB na dekadę, jeżeli chce się uzyskać większe nachylenie niezbędne jest tworzenie filtrów wysokich rzędów. W układach niecałkowitego rzędu do uzyskania tego samego efektu wystarczy jedynie zmienić rząd układu. Używanie tego typu układów w praktyce jest jednak utrudnione ponieważ do obliczenia dokładnych odpowiedzi układów musielibyśmy dysponować nieskończoną pamięcią, natomiast metody przybliżonego obliczania są na ogół niestabilne numerycznie dla wysokich rzędów aproksymacji.

W rozdziale tym pokazane zostały charakterystyki podstawowych układów regulacji i filtracji dla układów niecałkowitego rzędu.

4.1. Regulator $PI^\alpha D^\mu$

Regulator typu PID jest jednym z najbardziej znanych struktur regulacji. W pracy Podlubny (1998) (str. 249) została przedstawiona koncepcja uogólnienia tego regulatora poprzez dodanie do niego członu różniczkującego i całkującego niecałkowitego rzędu - $PI^\alpha D^\mu$. Działanie regulator tego typu opisane jest przez następującą funkcję:

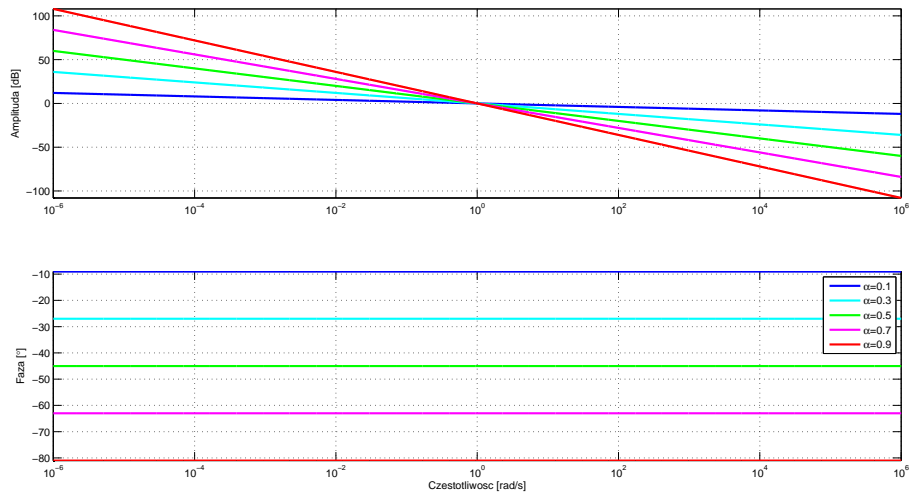
$$u(t) = K_p e(t) + K_i {}_0^C D_t^{-\alpha} e(t) + K_d {}_e^C D_t^\mu(t) \quad (4.1)$$

gdzie:

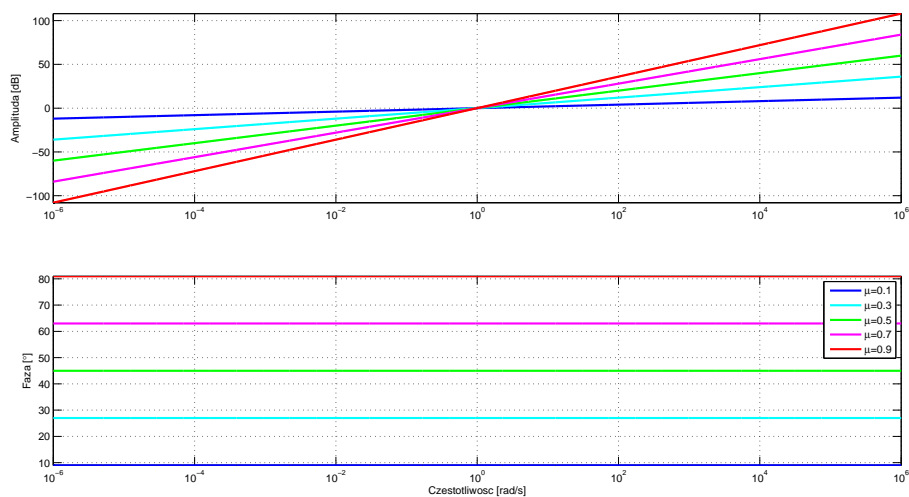
- K_p wzmocnienie członu proporcjonalnego,
- K_i wzmocnienie członu całkującego,
- K_d wzmocnienie członu różniczkującego,
- $e(t)$ uchyb sterowania,
- α rząd pochodnej członu całkującego,

– μ rząd pochodnej członu różniczkującego.

Na wykresach 4.1 i 4.2 widzimy charakterystyki amplitudowo-fazowe przy różnym niecałkowitym rzędzie, odpowiednio dla członu całkującego i różniczkującego.



Rysunek 4.1: Porównanie członu całkującego dla różnych wartości stopnia pochodnej ułamkowej



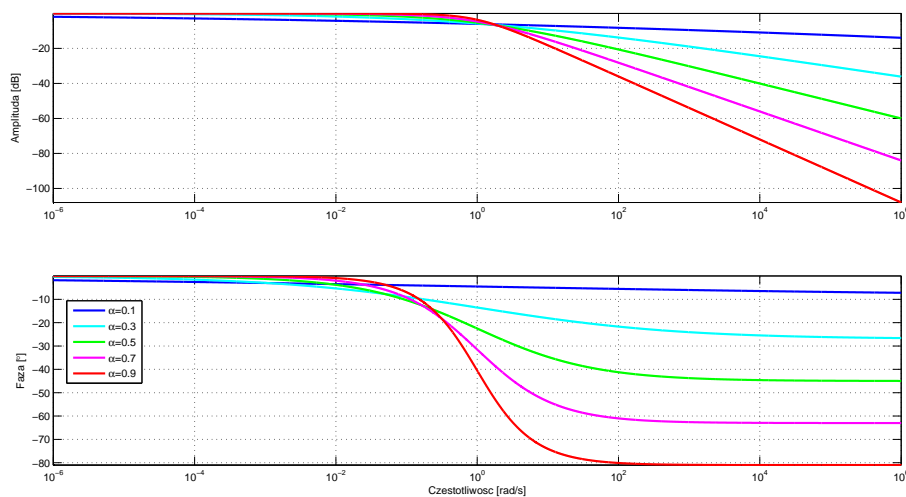
Rysunek 4.2: Porównanie członu różniczkującego dla różnych wartości stopnia pochodnej ułamkowej

4.2. Filtr niecałkowitego rzędu dolnoprzepustowy

Ogólna postać transmitancji układu opisująca filtr niecałkowitego rzędu dolnoprzepustowego jest dana równaniem:

$$G(s) = \frac{K}{Ts^\alpha + 1} \quad (4.2)$$

gdzie K jest wartością wzmocnienia niskich częstotliwości a T jest stałą czasową filtra. Zauważyć należy, że dla tej klasy filtrów pasmo tłumienia opada zgodnie z zależnością -6α dB/oct (Tsirimokou et al., 2017, str. 17). Natomiast częstotliwość zaporowa filtra wynosi $\omega_0 = 1/T$. Zachowanie się pasma zaporowego w zależności od rzędu pochodnej obrazuje wykres 4.3.



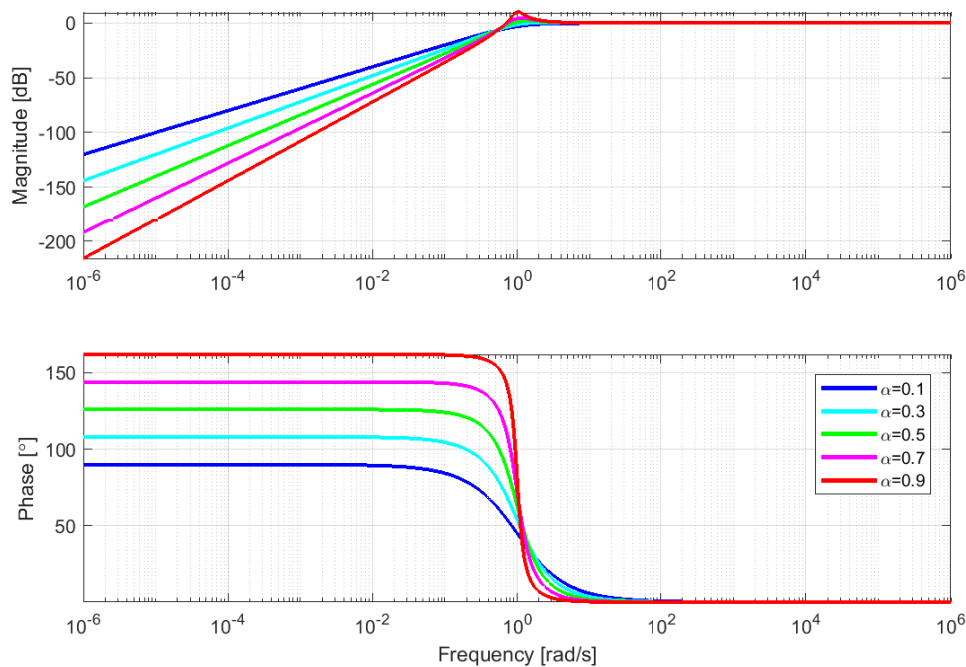
Rysunek 4.3: Porównanie charakterystyk filtra dolnoprzepustowego dla różnych wartości stopnia pochodnej ułamkowej

4.3. Filtr niecałkowitego rzędu górnoprzepustowy

W tym przypadku ogólna postać transmitancji dla tej klasy filtrów ma postać:

$$G(s) = \frac{Ks^\alpha}{Ts^\alpha + 1} \quad (4.3)$$

Przez analogię K jest wartością wzmocnienia wysokich częstotliwości a T jest stałą czasową filtra. W tym przypadku pasmo wzmocnienia rośnie zgodnie z zależnością 6α dB/oct (Tsirimokou et al., 2017, str. 17). Zachowanie tego filtra dla różnych wartości rzędu pochodnej obrazuje wykres 4.4.



Rysunek 4.4: Porównanie charakterystyk filtru górnoprzepustowego dla różnych wartości stopnia pochodnej ułamkowej

4.4. Filtr niecałkowitego drugiego rzędu

Filtr niecałkowitego drugiego rzędu jest opisywany za pomocą trzech parametrów:

- α - podstawowy rząd pochodnej ułamkowej
- b - współczynnik tłumienia
- c - wyraz wolny

i dany jest przez transmitancję:

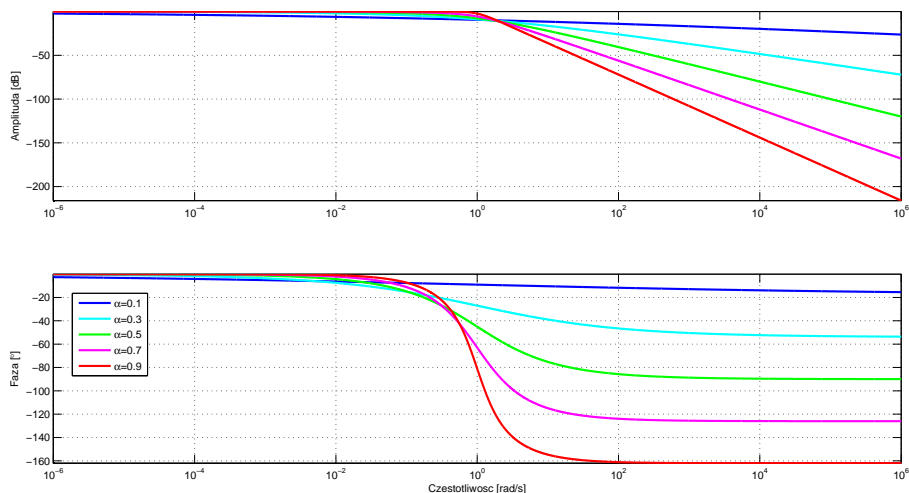
$$G(s) = \frac{1}{s^{2\alpha} + bs^\alpha + c} \quad (4.4)$$

Szczegóły dotyczące jego własności zostały opisane w (Baranowski et al., 2014d,e,b,f). Zachowanie tego typu filtra dla różnych wartości rzędu pochodnej obrazuje wykres 4.5.

4.5. Filtr niecałkowitego rzędu pasmowo przepustowy

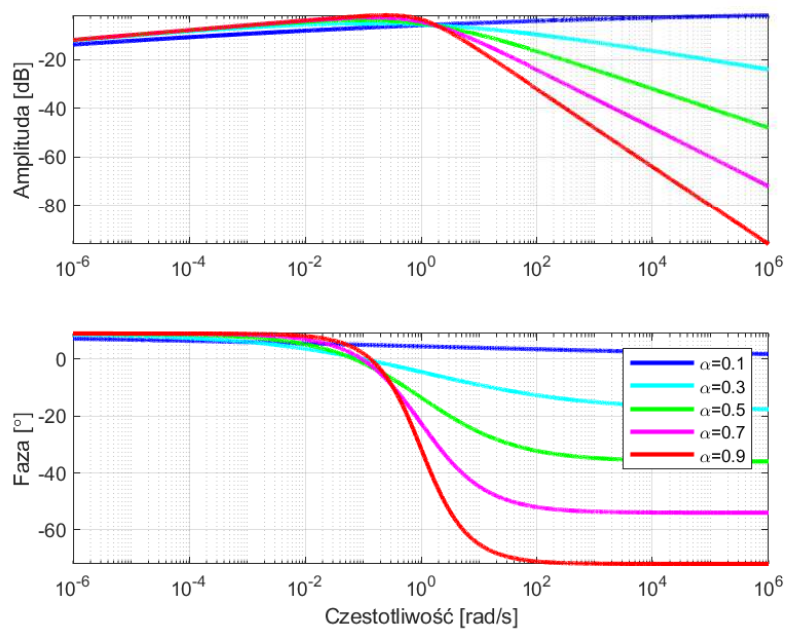
Ogólną postać filtru niecałkowitego rzędu pasmowo przepustowy jest dana transmitancją (Tsimokou et al., 2017, str. 18):

$$G(s) = \frac{Ts^\beta}{Ts^\alpha + 1} \quad (4.5)$$



Rysunek 4.5: Porównanie charakterystyk filtra ułamkowego drugiego rzędu dla różnych wartości stopnia pochodnej ułamkowej

Gdzie T jest stałą czasową filtra. Należy zauważyć, że tłumienie na częstotliwości zaporowej górnych częstotliwości wynosi $6(\alpha - \beta)$ dB/oct, podczas gdy dla niskich częstotliwości odcięcia jest to 6β dB/oct, co umożliwia konstrukcję filtra o różnych stopniach nachylenia zarówno wysokich jak i niskich częstotliwości. W przypadku, gdy $\alpha = 2\beta$ nachylenie pasm zaporowych jest równe 6β dB/oct dla wysokich częstotliwości i 6β dB/oct dla niskich częstotliwości. Przykład zachowań charakterystyk bodego dla tego typu filtra obrazuje wykres 4.6.



Rysunek 4.6: Porównanie charakterystyk filtra ułamkowego pasmowo przepustowego dla różnych wartości stopnia pochodnej α i $\beta = 0.1$

5. Układy wbudowane

Celem rozprawy jest opracowanie metod implementacji członów dynamicznych niecałkowitego rzędu na układy wbudowane. W poprzednim rozdziale została zaproponowana czasowa metoda aproksymacji Oustaloupa, która zostanie wykorzystana do implementacji członów niecałkowitego rzędu na układach wbudowanych. Do przetestowania poprawności działania tej metody zostaną użyte dwa układy wbudowane Arduino Uno oraz platforma STM32F051R8T6 oparta na mikroprocesorze STM32.

5.1. Układy wbudowane wybrane do testów

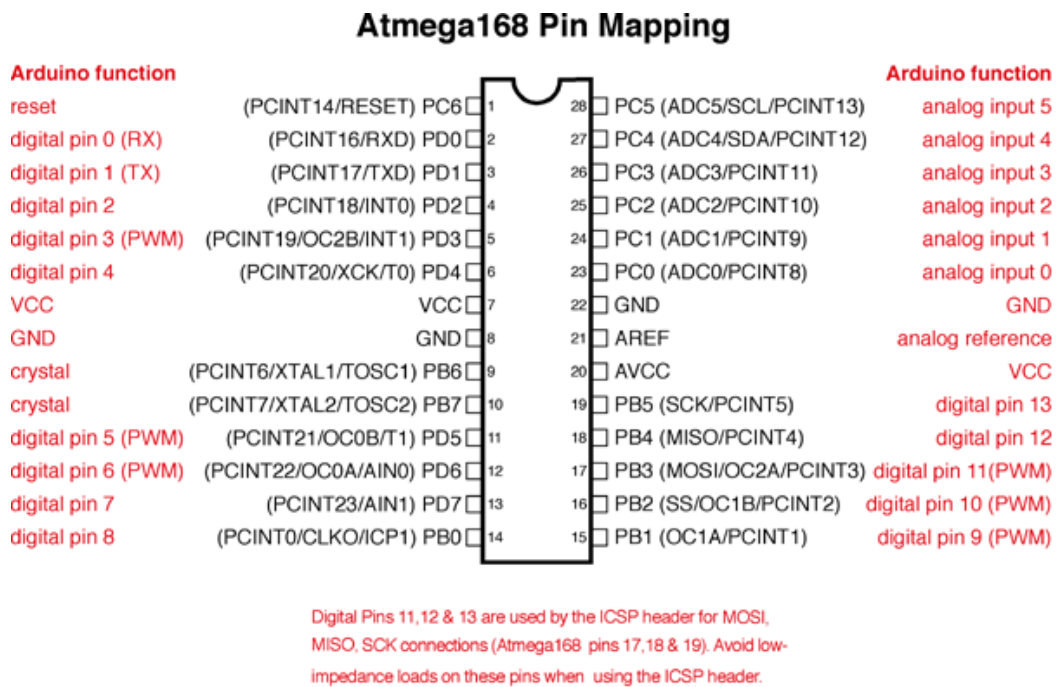
5.1.1. Arduino Uno

Arduino Uno to płyta oparta na mikroprocesorze ATmega328P. Posiadająca 14 pinów wejściowych/wyjściowych (z których 6 może być używanych jako wyjścia PWM), 6 wejść analogowych, kryształ kwarcowy 16 MHz, złącze USB, gniazdo zasilania, nagłówek ICSP i przycisk resetowania.

Omawiana platforma sprzętowa może być zasilana przez złącze USB lub z zewnętrznego źródła zasilania. Źródło zasilania jest wybierane automatycznie, pomiędzy zasilaniem z portu USB, zasilaczem prądu przemiennego (poprzez gniazdo zasilania) lub akumulatora (poprzez piny GND i Vin złącza Power). Układ może działać na zasilaniu zewnętrznym od 6 do 20 woltów. Jeśli jednak zasilany jest napięciem mniejszym niż 7 V, pin 5 V może dostarczać mniej niż 5 woltów, a układ może stać się niestabilny. Pin uziemienia IOREF zapewnia napięcie odniesienia, z którym działa mikrokontroler. Prawidłowo skonfigurowana osłona może odczytać napięcie podawane na IOREF i wybrać odpowiednie źródło zasilania lub włączyć translatory napięcia na wyjściach do pracy z napięciem 5V lub 3, 3V.

Pamięć na tym układzie zapewnia mikrokontroler ATmega328, który posiada 32 KB pamięci wewnętrznej, z czego 0,5 KB zajmuje bootloader. Dodatkowo układ wyposażony jest w 2 KB pamięci SRAM i 1 KB pamięci EEPROM (którą można zarządzać przy użyciu biblioteki EEPROM).

Układ obsługuje wszystkie 14 cyfrowych pinów mikrokontrolera, które można używać jako wejścia lub wyjścia. Wszystkie piny działają przy napięciu 5V. Każdy pin może dostar-



Rysunek 5.1: Schemat pinów ATmega328, źródło: <https://www.arduino.cc/en/Hacking/PinMapping168>

czyć lub odebrać 20mA, zabezpieczone są przez wewnętrzny rezystor podciągający (domyślnie odłączony) o wartości 20 – 50kΩ. Ponadto niektóre piny mają specjalistyczne funkcje:

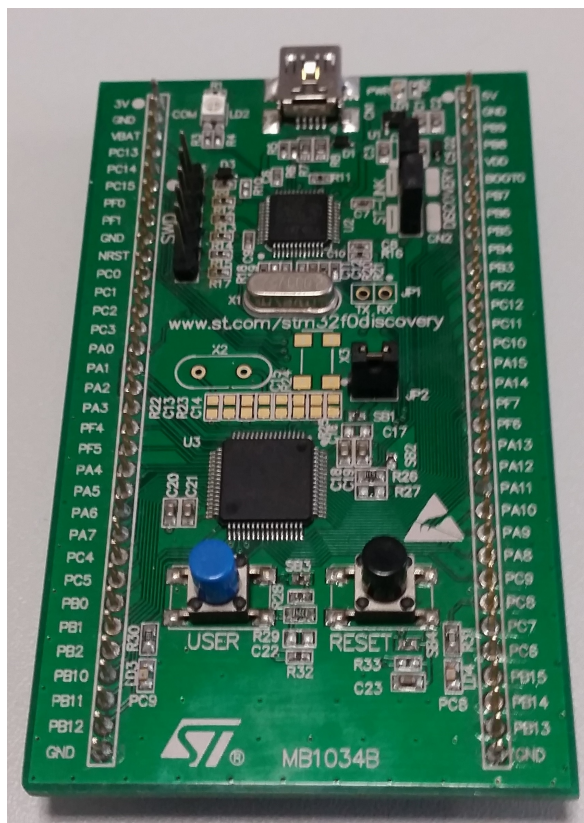
- Port szeregowy: służący do odbierania (RX) i przesyłania (TX) danych szeregowych TTL. Piny te są podłączone do odpowiednich pinów układu szeregowego USB-to-TTL ATmega8U2.
- Przerwania zewnętrzne: styki mogą być skonfigurowane do wyzwalania przerwania na niskiej wartości zbocza rosnącego lub opadającego lub zmiany wartości na porcie.
- Generator PWM: dostarcza 8-bitowego wyjścia PWM.
- SPI: piny do obsługi komunikacji SPI.
- LED: wbudowana dioda LED, którą można sterować.
- TWI: obsługa komunikacji TWI.
- 6 wejść analogowych z 10 bitową rozdzielczością. Domyślny pomiar wartości od wartości referencyjnej IOREF do 5 woltów, istnieje możliwość zmiany górnego zakresu za pomocą pinu AREF na planszy.

Arduino Uno jest zbudowany w taki sposób, by zapewnić wszystkie niezbędne peryferia do obsługi mikrokontrolera. Do jego programowania i ustawiania należy zainstalować dostarczane przez producenta IDE oraz połączyć układ do komputera poprzez port USB.

Arduino Uno zostało wybrane jako sprzętowa platforma do testów ze względu na omówioną konfigurację. Układ ten będzie służył do określenia możliwości implementacji filtrów niecałkowitego rzędu na platformach sprzętowych z ograniczonym zasobem pamięciowym i mocy obliczeniowej. Przetestowanie takiej konfiguracji pokaże czy istnieje możliwość implementacji układów niecałkowitego rzędu w tanich urządzeniach (na przykład tanie układy śledzenia ruchu, pulsu, itp.) lub w takich gdzie moc obliczeniowa jest ograniczona poprzez rozmiary urządzenia, zasilanie lub inne warunki.

5.1.2. Mikrokontroler STM32

Drugim wybranym do testów układem jest STM32F0-DISCOVERY. Moduł pozwala na realizację projektów w oparciu o popularny 32-bitowy mikrokontroler rodziny STM32. Dzięki dzięki temu układowi możliwe jest wypróbowanie możliwości rdzenia ARM Cortex M0. Dodatkowo firma ST będąca twórcą tego układu zapewnia pełne wsparcie procesu konfiguracji i programowania urządzenia poprzez dostarczenie darmowego oprogramowania STM32Cube (ST, 2019a).

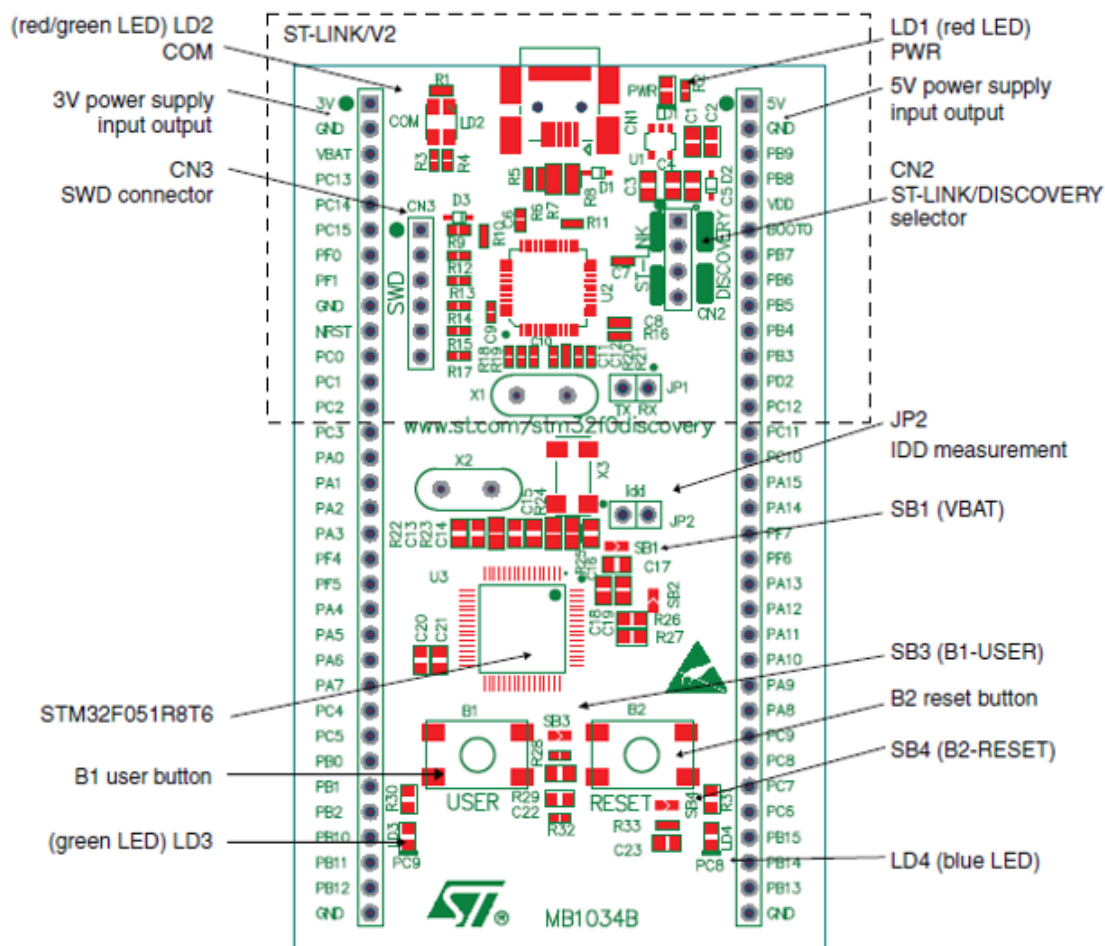


Rysunek 5.2: STM32F0-DISCOVERY Board

Jest to nowoczesny, tani i bardzo popularny układ scalony charakteryzujący się rozbudowanymi urządzeniami peryferyjnymi, w tym interfejsami komunikacyjnymi (takimi jak SPI, I2C, UART), licznikami, licznikami czasu i wieloma uniwersalnymi liniami wejścia-wyjścia ogólnego przeznaczenia. Licznik czasu umożliwia implementację na nim systemu

spełniające ograniczenia systemu czasu rzeczywistego dowolnej kategorii, zob. 5.2.3. Maksymalna częstotliwość robocza to 48 MHz. Układ ma pamięć flash 64 kB i pamięć RAM 8 kB. Układ można zasilać przy pomocy portu USB lub też zewnętrznego źródła zasilania 5V/3,3V. Debuggera ST-Link/V2 znajdujący się na układzie umożliwia pracę układu jako oddzielne urządzenie z wyjściem SWD. Dodatkowo na układzie znajdują się cztery diody LED, dwie z nich może programować użytkownik. Dwa przyciski jeden pełniący funkcję resetu oraz drugi który można oprogramować. Na płycie zostały również wyprowadzone goldpiny do wszystkich portów wejściowych i wyjściowych mikrokontrolera.

Szczegółowe rozmieszczenie elementów na układzie prezentuje rysunek 5.3.



Rysunek 5.3: Schemat rozmieszczenia elementów na STM32F0-DISCOVERY Board (ST, 2019b)

5.2. Czas rzeczywisty w układach wbudowanych

5.2.1. Systemy czasu rzeczywistego

Systemów czasu rzeczywistego jako pojęcie uformowało się równoległe z budową dużych systemów komercyjnych i rządowych w Stanach Zjednoczonych w okresie lat 60-tych. Martin (1965) w *Programming Real-time Computer Systems* jako jeden z pierwszych omówił zagadnienia:

- Algorytmów dynamicznego przydzielania zasobów (**ang.** *dynamic scheduling*).
- Dynamicznej alokacji rdzeni procesora (**ang.** *dynamic core allocation*).
- Przydzielania priorytetów wątków i procesów (**ang.** *allocation of priorities*).
- Współbieżność procesów (**ang.** *multi-programming*).
- Struktury danych i funkcjonalności takie jak: przerwania (**ang.** *interrupts*), kolejki (**ang.** *queues*), przeciążenia (**ang.** *overloads*), wielozadaniowość (**ang.** *multi-processing*).
- Mechanizmy komunikacji w systemie i procesie: linie komunikacyjne (**ang.** *communication lines*), pliki o swobodnym dostępie (**ang.** *random-access files*), programy nadzorcze (**ang.** *supervisory programs*),
- Zdefiniowanie komunikacji z innymi komputerami (**ang.** *communication with other computers*), kryteriów wysokiej niezawodności (**ang.** *high reliability*), redundancji (**ang.** *duplexing*) i przełączania awaryjnego (**ang.** *switchover fall-back*).

Dodatkowo w pracy tej zostały podjęte tematy związane z tematyką inżynierii oprogramowania na systemy wbudowane - testowanie oprogramowania w sposób automatyczny, problemy z zarządzaniem procesem wytwarzania oprogramowania i projektowania systemów. Wszystkie problemy omówione przez Martina są obecne do dzisiaj w procesie tworzenia oprogramowania dla systemów czasu rzeczywistego. W definiowanie własności systemów czasu rzeczywistego wkład swój wniósł również Dijkstra (1962/1963.), który w pracy *Over de sequentialiteit van procesbeschrijvingen* zaproponował koncepcję semafora. Koncepcja ta opierała się o zmienną używaną do kontrolowania dostępu do współdzielonego zasobu, aby skutecznie poradzić sobie z synchronizacją. Wraz z wprowadzeniem semaforów i innych struktur do kontroli procesów w systemach czasu rzeczywistego, takich jak wzajemne wykluczanie i serializację, poprawność ich działania może być formalnie udowodniona (Wang, 2014). Dzięki tym własnościom oraz rozwojowi elektroniki w ciągu ostatnich dwóch dekad systemy czasu rzeczywistego przeniosły się w sferę produktów konsumenckich (przenośne urządzenia, automatyka domowa, automatyka samochodowa itp.), tym samym wymuszając

zarówno na środowisku akademickim jak i komercyjnym prace nad rozwojem tych systemów.

Większość oprogramowania i systemów z których korzystamy na co dzień odpowiada na żądania użytkownika bez ograniczeń czasowych, z tak zwanymi opóźnieniami. Dla przykładu można przytoczyć tutaj sytuację gdy program który próbujemy uruchomić zostaje wywołany przez system jednego dnia po kilku sekundach a następnego po kilkunastu. Systemy które nie mają zdefiniowanych ograniczeń czasowych i wydajnościowych nazywamy systemami przeznaczenia ogólnego. Natomiast systemy, które muszą odpowiedzieć na zgłoszenie w określonym czasie nazywane są systemami czasu rzeczywistego (Kopetz, 2011; Buttazzo, 2011). System czasu rzeczywistego traktuje każde przychodzące żądanie jako zadanie, które musi zostać obsłużone z czasowym ograniczeniem. W przeciwieństwie do systemów ogólnego przeznaczenia, systemy czasu rzeczywistego są przeznaczone do monitorowania, interakcji, sterowania lub reagowania na środowisko fizyczne. "Komunikacja ze światem zewnętrznym" odbywa się poprzez czujniki, systemy komunikacyjne i inne wejścia oraz urządzenia zewnętrzne. W tym kontekście konieczne jest obsłużenie żądania w zadanym czasie. Opóźnienia mogą okazać się niebezpieczne, a nawet katastrofalne. W związku z tym system czasu rzeczywistego musi posiadać następujące cechy: 1. czas dostarczenia odpowiedzi na żądanie jest równie ważny jak poprawność tej odpowiedzi, 2. konsekwencje opóźnienia odpowiedzi na żądanie jest równie niebezpieczne, jak błędna odpowiedzi.

Inne wymagania są uogólnione i opisane jako wymagania niefunkcjonalne, a opisują one wytyczne dotyczące bezpieczeństwa i wydajności systemów czasu rzeczywistego - zostały one zebrane i opisane w tabeli 5.1.

Wymagania niefunkcjonalne	Opis
Bezpieczeństwo	System ma reagować na żądania w taki sposób, by zapobiec uszkodzeniom.
Wydajność	Dotyczy to zarówno czasu reakcji, jak i przepustowości niezbędnej do ochrony systemu przed uszkodzeniem lub innymi niepożądanymi skutkami.
Odporność na uszkodzenia	Możliwość ochrony systemu przed uszkodzeniami wynikającymi z błędów projektowych.
Odporność	Zdolność systemu do ochrony przed uszkodzeniami wynikającymi z zewnętrznych zakłóceń.
Skalowalność	Możliwość łatwego rozbudowywania systemu w zależności od wymogów wydajnościowych.
Zabezpieczenia	Działania systemu, aby zapobiec umyślnemu uszkodzeniu systemu.

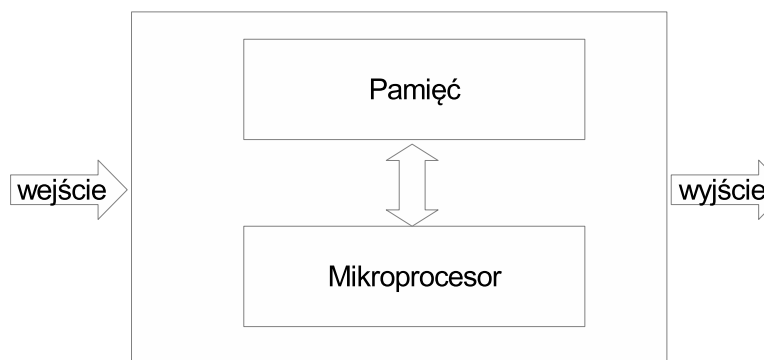
Tablica 5.1: Wymagania niefunkcjonalne dla systemów czasu rzeczywistego

5.2.2. Systemy wbudowane

W naukach technicznych termin "system wbudowany" jest używany w odniesieniu do systemów elektronicznych zaprojektowanych do wykonywania dedykowanych funkcji i często "wbudowany" (będących częścią) większych systemów.

Systemy wbudowane różnią się od urządzeń obliczeniowych ogólnego przeznaczenia głównie w dwóch aspektach:

1. System wbudowany jest zaprojektowany dla określonej funkcji. Natomiast urządzenie komputerowe ogólnego przeznaczenia, takie jak smartfon, laptop lub komputer stacjonarny, mogą być używane w różnych celach na przykład jako serwery WWW, hurtownie danych, do prowadzenia prac biurowych, czytania wiadomości, itd.
2. System wbudowany jest zazwyczaj budowany wraz z oprogramowaniem, które ma na nim działać. Taki równoległy model tworzenia sprzętu i oprogramowania jest znany jako współprojektowanie sprzętu i oprogramowania (**ang.** *hardware-software co-design*). W przeciwieństwie do tego, urządzenie obliczeniowe ogólnego przeznaczenia jest zazwyczaj budowane niezależnie od aplikacji, które mogą na nim działać.



Rysunek 5.4: Uproszczony schemat architektury systemów wbudowanych

System wbudowany to połączenie komponentów elektronicznych i oprogramowania, a czasem także elementów mechanicznych. Schemat 5.4 przedstawia ogólny koncept architektury systemów wbudowanych. Oprogramowanie wbudowane jest zazwyczaj przechowywane w pamięciach nieulotnych takich jak: pamięć tylko do odczytu (ROM), kasowalna pamięć programowalna (EPROM) i pamięci flash. Mikroprocesor wymaga również pamięci podręcznej o dostępie swobodnym (RAM) do obliczeń w czasie wykonywania.

Po włączeniu systemu wbudowanego jego mikroprocesor odczytuje instrukcje oprogramowania przechowywane w pamięci, wykonuje instrukcje przetwarzania informacji wej-

ściowych z komponentów peryferyjnych (za pomocą czujników, sygnałów, przycisków itp.) i generuje dane wyjściowe potrzebnych innym zewnętrzny komponentom systemu.

Biorąc pod uwagę, że obecnie możliwe jest wybranie gotowych systemów sprzętowych do realizacji konkretnych zadań, większość wysiłku tworzenia systemów wbudowanych dotyczy oprogramowania: w tym aplikacji i sterowników urządzeń. W wielu przypadkach możliwe jest zbudowanie niestandardowego układu scalonego, który jest funkcjonalnie równoważny systemowi wbudowanemu. Rozwiązanie oparte na układach scalonych to rozwiązanie sprzętowe, które nie zawiera oprogramowania i mikroprocesora. Jednak oparcie rozwiązania konkretnego problemu na systemach wbudowanych jest bardziej elastyczne i tańsze, zwłaszcza gdy produkt musi być często aktualizowany w celu uwzględnienia nowych zmian. W odpowiedzi na nową zmianę, dla rozwiązania sprzętowego, nowy układ scalony musi zostać zaprojektowany, skonstruowany i dostarczony. Natomiast w systemie wbudowanym często wystarczy zaprojektować lub uaktualnić nowy program jeżeli warstwa sprzętowa nieuległa zmianie. Przez co proces ten jest dużo tańszy i może być wykonywany zdalnie lub przez użytkownika.

5.2.3. Rodzaje systemów czasu rzeczywistego

Ograniczenie czasowe zadania jest powiązane z wieloma czynnikami za zwyczaj z wymaganą częstotliwością pracy systemu która wynika z dynamiki systemu. Ograniczenie czasowe mówiąc inaczej to wymagany czas wykonania konkretnego zadania po którym wynik ten nie jest akceptowalny. W zależności od konsekwencji na działanie systemu powodowane przez nie spełnienie tego warunku wyróżnić można dwa typy ograniczeń:

- *bezwzględne* - gdy odpowiedź systemu musi nastąpić w określonej chwili czasu po zdarzeniu,
- *względne* - gdy odpowiedź musi nastąpić w określonym czasie po zdarzeniu.

Konsekwencje niespełnienia ograniczeń pozwalają na klasyfikację systemów czasu rzeczywistego jako:

1. *Systemy z twardymi ograniczeniami czasowymi (ang. hard real-time)* - system w którym niespełnienie ograniczenia jest traktowane jako niepowodzenie, przy czym niema znaczenia czas przekroczenia odpowiedzi. Oznacza to, że w tym typie systemów jeżeli system odpowiedział poprawnie po czasie taka odpowiedź nie jest uznawana.
2. *Systemy o mocnych ograniczeniach czasowych (ang. firm real-time)* - system w którym "sporadyczne" niedotrzymanie ograniczeń nie doprowadzi do awarii systemu, jednak każda odpowiedź systemu po terminie nie ma wartości dla jego komponentów, powodując pogorszenie jakości działania całego systemu.

3. *System o miękkich ograniczeniach czasowych (ang. soft real-time)* - system w którym przekroczenie czasu na odpowiedź jest akceptowalne jednak powoduje zmniejszenie jakości pracy systemu lub doprowadza do jego awarii.

W pierwszych dwóch przypadkach, jeśli można ustalić a priori, że termin wykonanie zadania nie zostanie dotrzymany, lepszym rozwiązaniem może być nie podejmowanie tego zadania i zastosowanie procedur bezpieczeństwa. Bardziej złożone systemy czasu rzeczywistego będą składały się podsystemów z każdej z omówionych kategorii. System w którym zadania mają miękkie ograniczenia czasowe klasyfikujemy jako Soft Real Time. System jest nazywany Hard Real Time jeśli jego wszystkie kluczowe zadania mają twarde ograniczenia czasowe.

Soft Real Time

Systemy typu Soft Real Time charakteryzują się tym, że obsługa żądania jest prawie zawsze zakończona w znanym skończonym czasie, które odpowiada ograniczeniu czasowemu i pewnej tolerancji na jej nie spełnienie. Oznacza to, że w tej klasie systemów dopuszczalne jest pominięcie zadanej liczby odpowiedzi. Warto zauważyć, że chociaż brak odpowiedzi systemu nie spowoduje krytycznych błędów w jego działaniu jednak może ono ulec pogorszeniu. Przykłady tego typu systemu zostały opisane w tabeli 5.2.

System	Przykład ograniczenia czasowego	Skutki nie spełnienia ograniczenia
Kamera Cyfrowa	Czas otwarcia migawki ustawiony jest na 0.5s	Prześwietlone lub zbyt ciemne zdjęcie
GPS	Odbieranie informacji o pozycji z częstotliwością 1Hz	Złe odwzorowanie trajektorii
Router bezprzewodowy	Średnia wartość opóźnionych lub utraconych wynosi 2/min	Opóźniana w transmisji danych

Tablica 5.2: Przykładowe systemy czasu rzeczywistego z miękkimi ograniczeniami czasowymi

Firm Real Time

W tym typie systemu każde zadanie ma określone wcześniej ograniczenie czasowe, przed którym wymagane jest uzyskanie odpowiedzi systemu. Jednak w przeciwieństwie do systemów z miękkimi ograniczeniami czasowymi odpowiedzi które nie spełniają ograniczeń czasowych są odrzucane. Innymi słowy, wyniki obliczone przez system po czasie ograniczenia stają się nieprzydatne. Tego typu systemy zostały opisane w tabeli 5.3

System	Przykład ograniczenia czasowego	Skutki nie spełnienia ograniczenia
Wideo konferencja	Różnica w dekompresji dźwięku i wizji nie może przekroczyć 0.1 ms.	Niezsynchronizowanie obrazu i dźwięku
Śledzenie wizyjne	Algorytm śledzenia musi wykonać się w czasie 1ms od przyjścia zdjęcia	Zgubienie śledzonego obiektu
Nagrywarka TV	Odczyt strumienia danych ma odbywać się z częstotliwością 100Hz	Niepoprawnie nagrany program

Tablica 5.3: Wymagania нефункционалне dla systemów czasu rzeczywistego

Hard Real Time

W ostatnim z typów omawianych systemów czasu rzeczywistego to znaczy z twardymi ograniczeniami czasowymi nie dotrzymanie jakichkolwiek ograniczeń jest całkowicie nie do przyjęcia, ponieważ może to spowodować uszkodzenie systemu lub zaprzestanie poprawnego jego działania. Tego typu systemu czasu rzeczywistego gwarantują zarówno poprawność przeprowadzonych obliczeń jak i czasu odpowiedzi. Mówimy w tym przypadku o poprawności funkcjonalnej i czasowej. Poprawność czasowa systemu jest definiowana jako obsługa żądania w zadanym ograniczeniu czasowym. W praktyce poprawność czasowa jest ważniejsza niż poprawność funkcjonalna, to znaczy zakłada się, że obliczenia potrzebne do uzyskania odpowiedzi mają zadaną dokładność w taki, sposób by spełniać ograniczenia czasowe.

Ponieważ nie spełnienie twardych ograniczeń czasowych niesie za sobą poważne konsekwencje w działaniu systemu, zwykle w procesie tworzenia takiego systemu jest rygorystycznie sprawdzana poprawność systemu. W literaturze techniki sprawdzania poprawności działania systemu czasu rzeczywistego obejmują harmonogramowanie zadań (**ang.** *design-time schedulability analysis*), symulacje działania systemu i kombinatoryczne testowanie (**ang.** *combinatorial performance testing*). Tabela 5.4 podaje niektóre przykłady twardych systemów czasu rzeczywistego.

System	Przykład ograniczenia czasowego	Skutki nie spełnienia ograniczenia
ABS	Koło, które się blokuje, powinno przestać się obracać w mniej niż 1 s	Wypadek samochodowy
Rozrusznik serca	Dolna granica czasu oczekiwania wynosi 0,1 s, a górna granica czasu oczekiwania wynosi 0,2 s	Śmierć pacjenta
FTSE 100 Index	Obliczenie nowego indeksu i opublikowanie co 15s	Kryzys finansowy

Tablica 5.4: Wymagania нефункционалне dla systemów czasu rzeczywistego

5.3. Implementacja elementów niecałkowitego rzędu w układach wbudowanych

Ponieważ rozpatrywane układy są układami cyfrowymi, przed implementacją na nich układów niecałkowitego rzędu należy poddać je dyskretyzacji. We wszystkich eksperymentach opisanych w tej rozprawie będziemy korzystać z metody dyskretyzacji Tustin, a opisanej wzorami (2.38)–(2.42).

5.3.1. Algorytmy

Podstawowa forma algorytmu umożliwiająca realizację układów omawianych w tej pracy ma postać, (Bauer i Kawala-Janik, 2017):

Data: N – rząd aproksymacji
 A, B, C, D – macierze układu dyskretnego
 xp – stan członu w chwili t
 B_u, A_x – zmienne pomocnicze
 u_t – wartość sygnału wejściowego w chwili t
 x – wektor pomocniczy stanu
Result: y_t – odpowiedź układu w chwili t ;
 $y_t = 0$;
for $i = 0$ *to* $N - 1$ **do**
 $B_u = B[i] * u_t$;
 $A_x = 0$;
 for $j = 0$ *to* $N - 1$ **do**
 $A_x = A_x + A[i][j] * xp[j]$;
 end
 $x[i] = A_x + B_u$;
 $y_t = y_t + C[i] * x[i]$;
end
 $xp = x$;
return $y_+ = D * u_t$

Algorithm 1: Ogólny algorytm realizacji układu dyskretnego

Algorytm ten jest wersją najbardziej ogólną przez co jest złożoność wynosi $O(n^2)$. Zaletą tego rozwiązania jest to, że jeżeli jesteśmy w stanie podać macierze stanu dyskretnego systemu zadziała on niezależnie od metody aproksymacji. Jednak ze względu na ograniczenia sprzętowe i czasowe na układach wbudowanych warto wykorzystać własności czasowej aproksymacji Oustaloupa (3.59). Zauważmy, że w tej metodzie macierz stanu jest macierzą trójkątną dolną, więc jej niezerowe współczynniki możemy zapisać w wektorze w następu-

jącej kolejności:

$$\mathbf{A} = \left[A_1, B_2, A_2, \dots, B_N, A_N \right], \quad (5.1)$$

możemy zmodyfikować algorytm 1 tak, by wykorzystał tę informację. W takim przypadku otrzymujemy:

Data: N – rząd aproksymacji

A, B, C, D – wektory układu dyskretnego

x_p – stan filtru w chwili t

B_u, A_x – zmienne pomocnicze

u_t – wartość sygnału w chwili t

x – wektor pomocniczy stanu

Result: y_t – odpowiedź układu w chwili t ;

$y_t = 0$;

for $i = 0$ *to* $N - 1$ **do**

$B_u = B[i] * u_t$;

$A_x = 0$;

for $j = 0$ *to* $i + 1$ **do**

$A_x = A_x + A[j + i] * x_p[j]$;

end

$x[i] = A_x + B_u$;

$y_t = y_t + C[i] * x[i]$;

end

$x_p = x$;

return $y_t = D * u_t$

Algorithm 2: Algorytm realizacji układu dyskretnego dla macierzy trójkątnych

Modyfikacja ta po pierwsze pozwala na zaoszczędzenie pamięci. Dzieje się tak ponieważ bez straty dla poprawności działania nie musimy pamiętać zerowych wartości z komórek macierzy. Dodatkowo warto zauważyć, że oszczędzona pamięć rośnie wraz z rzędem aproksymacji N . Dla $N = 2$ oszczędzamy 1 pole, co stanowi 25% całej pamięci potrzebnej na zapamiętanie macierzy stanu. W przypadku $N = 3$ są to już 3 pola a więc około 33%, wartości zaoszczędzonych zasobów będą rosły wraz ze wzrostem rzędu aproksymacji. Drugą istotną zaletą tej modyfikacji algorytmu jest obniżenie jego złożoności do $O(n)$, co bezpośrednio przekłada się na przyspieszenie odpowiedzi układu.

5.4. Wyniki implementacji na układach wbudowanych

5.4.1. Implementacja Arduino Uno

W celu przetestowania jakości implementacji zaproponowanego rozwiązania na układzie Arduino Uno wybrano filtr drugiego rzędu (4.4) dla rzeczywistego sygnału EEG oraz implementację ułamkowego regulatora PID.

Filtr drugiego rzędu dla sygnału EEG

Testowany sygnał EEG był zarejestrowany przy pomocy zestawu EEG - Emotiv EPOC. Czas trwania testowanych sygnałów wynosił około 700 sekund i rejestrował aktywność mózgu pacjenta dla ruchów lewą i prawą ręką, zobacz (Kawała-Janik et al., 2013). Schemat opracowanego mechanizmu filtracji za pomocą Arduino Uno prezentuje rys.5.5.

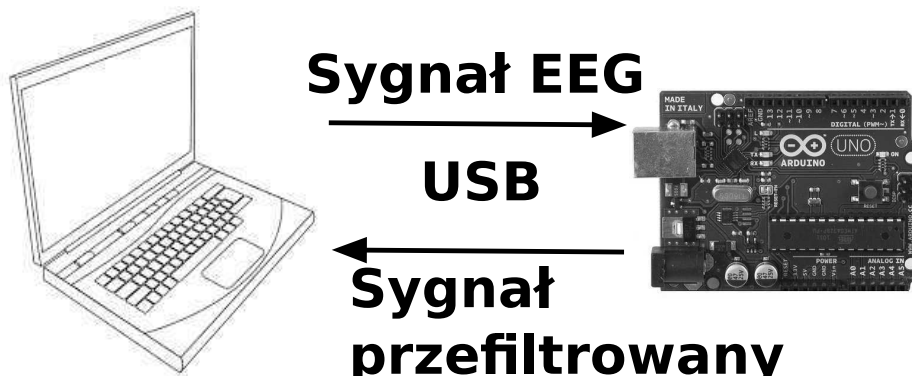
Dla porównania filtr ten został również zaimplementowany w środowisku Matlab Simulink, który posłużył za rozwiązanie referencyjne do określenia jakości filtracji na platformie Arduino Uno.

Do eksperymentu przyjęto następujące wartości dla parametrów filtra:

- $N = 7$,
- $\omega = [10^{-6}, 10^6]$,
- $\alpha = \{0.1, 0.7\}$,
- $b = 1.5$,
- $c = 2.24$,

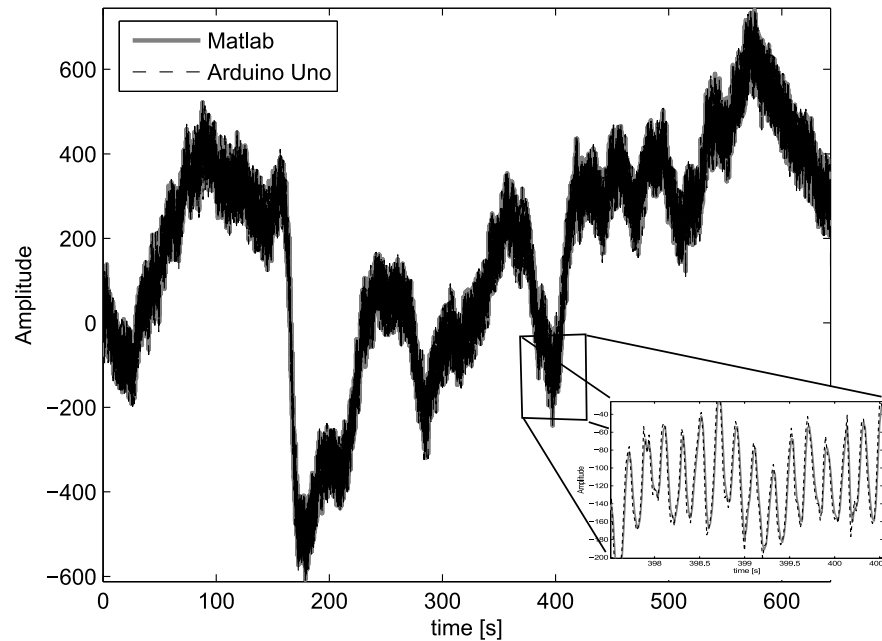
natomiast czas dyskretyzacji wynosił 0.01s.

Wynik filtracji opracowanego filtra dla $\alpha = 0.1$ jest pokazany na rysunku 5.6, 5.7, a wyniki uzyskane dla $\alpha = 0.7$ ilustrują rysunki 5.6, 5.7. Średni błąd między implementacją Matlab Simulink i Arduino Uno dla filtrowanego sygnału w pełnym zakresie czasu ma wartość 9, 5.



Rysunek 5.5: Arduino Uno/PC connection schema.

Biorąc pod uwagę fakt, że Arduino Uno reprezentuje liczby zmiennoprzecinkowe w pojedynczej precyzji, otrzymany wynik należy uznać za bardzo dobry. Drugi przypadek, gdy $\alpha = 0,7$ daje średnią wartość błędu 12,48. Poprawność działania tej metody filtracji potwierdzają również wyniki przeprowadzone w dziedzinie częstotliwości przedstawione na rys. 5.7 i 5.9. Jak łatwo zauważyć w obu przypadkach widmowa wartość mocy sygnałów po filtracji jest prawie identyczna. Widoczne różnice pojawiają się jedynie blisko końca przedziału częstotliwości aproksymacji, co można w przyszłości niwelować jego przesunięciem.

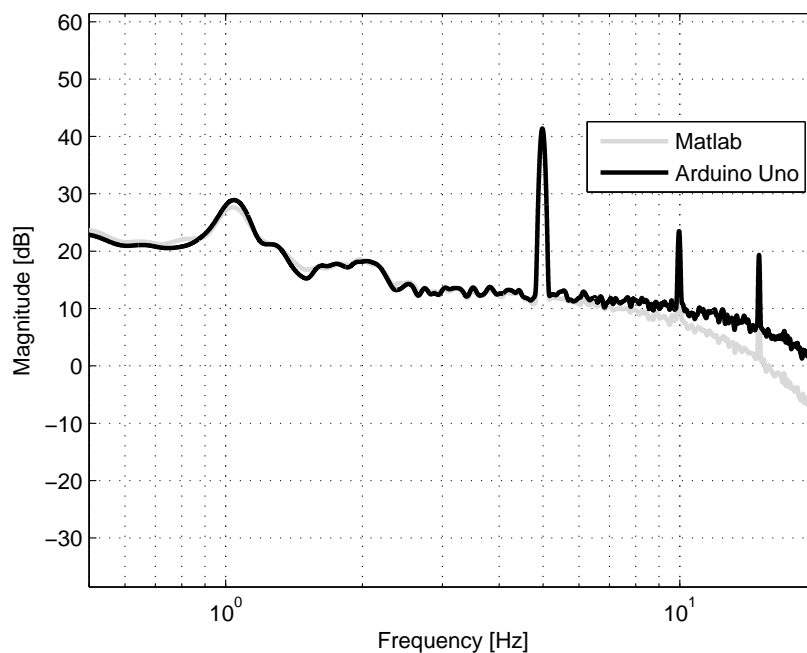


Rysunek 5.6: Porównanie wyników filtracji na platformie Arduino Uno z Matlab Simulink dla $\alpha = 0.1$

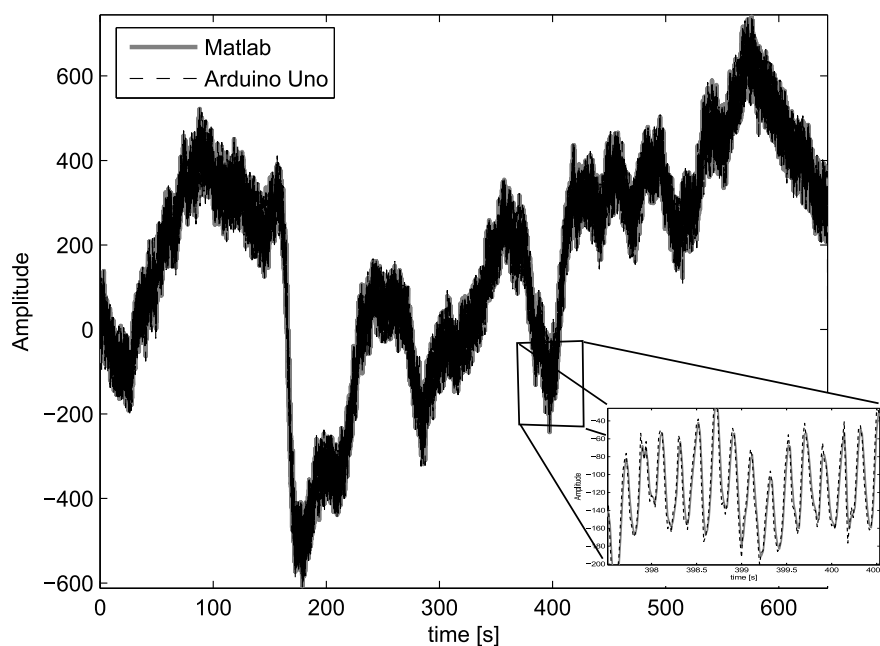
Ułamkowy regulator PID

W tym eksperymencie jak i w poprzednim regulator został zaimplementowany na układzie wbudowanym i w Matlab Simulink, który również w tym przypadku pełnił funkcję środowiska referencyjnego. Do zaimplementowani regulatora zostały użyte następujące wartości parametrów:

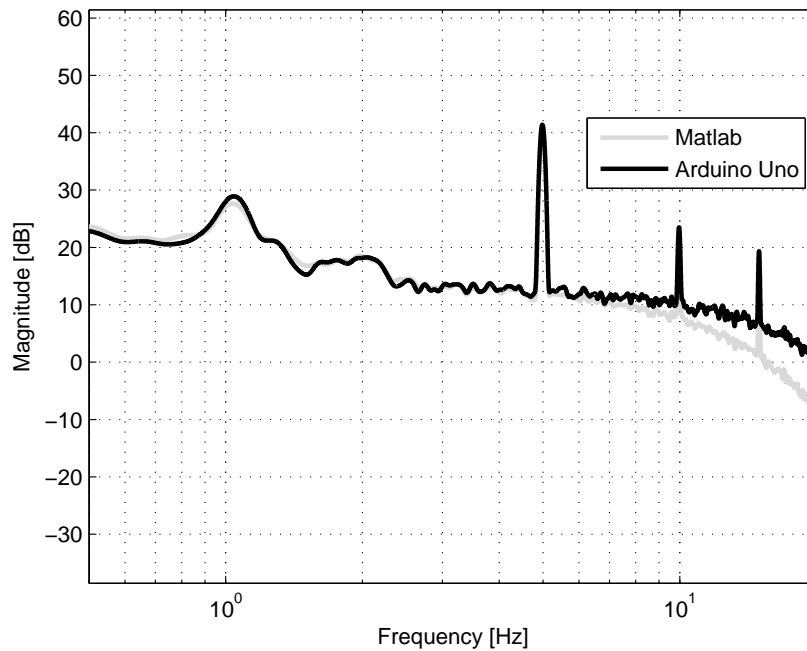
- $K_p, K_i, K_d = 1$
- $\alpha \in \{0.3, 0.7\}$
- $\mu \in \{0.3, 0.7\}$
- $N \in \{5, 10\}$
- $\omega \in [10^{-6}, 10^6]$



Rysunek 5.7: Porównanie gęstość mocy sygnału po filtracji przez Arduino Uno i Matlab Simulink dla $\alpha = 0.1$



Rysunek 5.8: Porównanie wyników filtracji na platformie Arduino Uno z Matlab Simulink dla $\alpha = 0.7$



Rysunek 5.9: Porównanie gęstość mocy sygnału po filtracji przez Arduino Uno i Matlab Simulink dla $\alpha = 0.1\alpha = 0.7$

Natomiast czas dyskretyzacji to 0.01 s. Parametry wzmocnień zostały ustawione w taki sposób by jak najłatwiej sprawdzić właściwości numeryczne proponowanego rozwiązania. Rząd, zakres częstotliwościowy aproksymacji oraz rzędy pochodnych wybrano tak by sprawdzić zachowanie algorytmu dla wysokich rzędów aproksymacji oraz szerokiego zakresu częstotliwości.

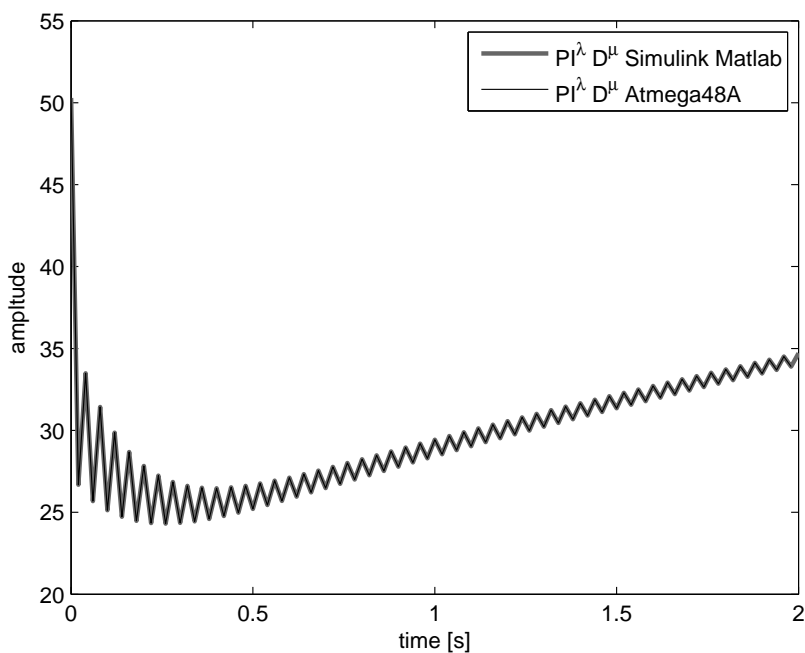
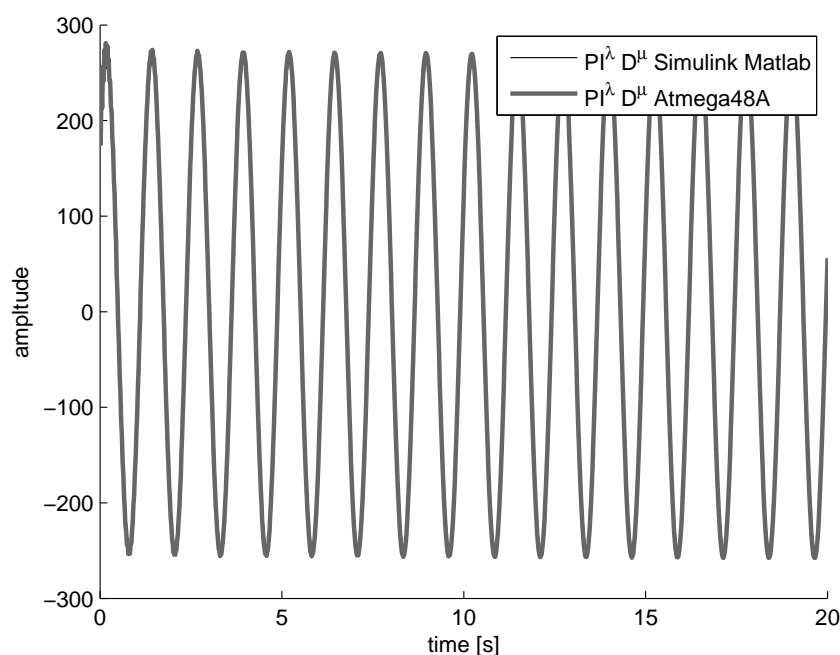
Do oceny jakości wyników posłużymy się następującym wskaźnikiem:

$$Error = \sum_0^T |M_i - A_i| \quad (5.2)$$

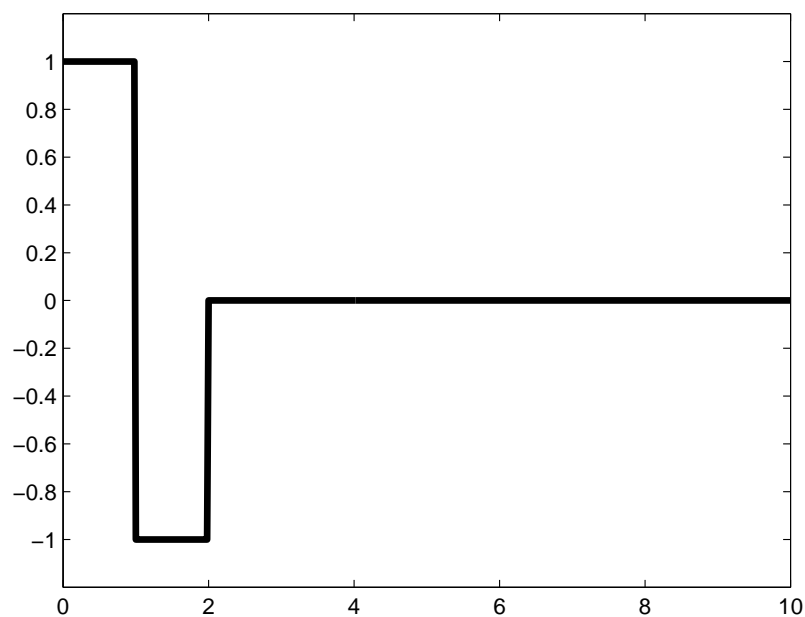
gdzie T oznacza długość sygnału, M to sygnał uzyskany z Matlab Simulink and A to sygnał z Atmega328P.

Odpowiedź krokowa systemu w otwartej pętli dla 2 sekund jest pokazana na rysunku 5.10. W tym przypadku błąd (5.2) wynosi 0,047 i z punktu widzenia rzeczywistych układów sterowania na tym przedziale czasowym jest akceptowalna. Drugi eksperyment przedstawiony został na rysunku 5.11, polegał na zbadaniu własności nadążania za sygnałem sinusoidalnym przez regulator w czasie 10 sekund. Wartość błędu na całym odcinku czasu w tym przypadku wynosił 1,25, co również zważając na różnice w reprezentacji liczb zmienoprzecinkowych jest akceptowalne.

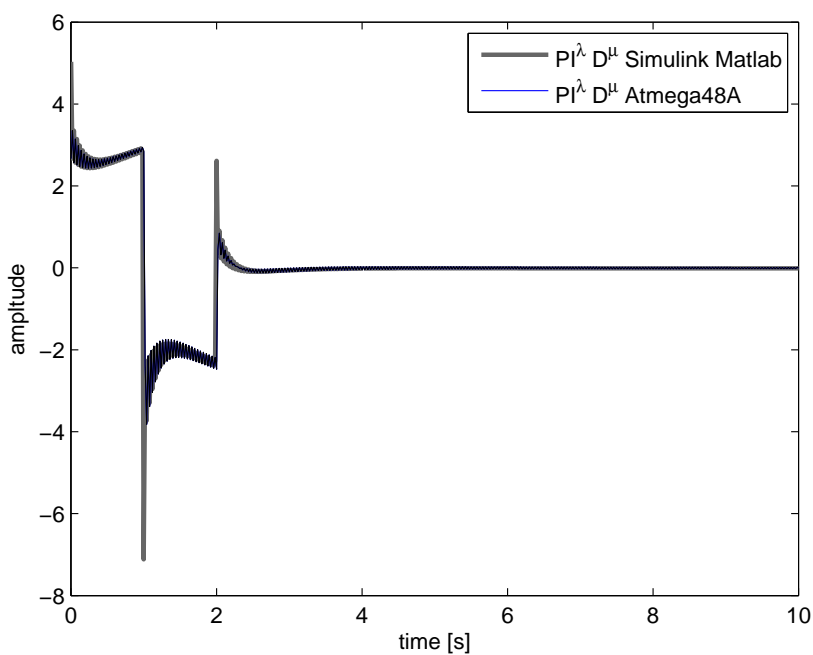
Wyniki ostatniego eksperymentu polegającego na śledzeniu sygnału referencyjnego postaci 5.12 trwającego 10 sekund zestawione są w tabeli 5.5 oraz na rysunku 5.12. Porówna-

Rysunek 5.10: Odpowiedź skokowa regulatora $PI^{\lambda}D^{\mu}$ w Matlab Simulink i ATmega328PRysunek 5.11: Odpowiedź regulatora $PI^{\lambda}D^{\mu}$ w Matlab Simulink i ATmega328P na sygnał sinusoidalny

nie to pokazuje, że implementacja na Arduino Uno jest stabilna numerycznie w przypadku zmiany parametrów aproksymacji.



Rysunek 5.12: Sygnał referencyjny

Rysunek 5.13: Odpowiedź regulatora $PI^\lambda D^\mu$ w Matlab Simulink i ATmega328P na sygnał referencyjny

5.4.2. Implementacja STM32

Do sprawdzenia poprawności opracowanych algorytmów użyto platformę sprzętową opartą na mikrokontrolerze ARM Cortex-M0 - STM32, patrz rys. 5.2.

Rząd	Błąd			
	$\alpha = 0.3$ $\mu = 0.3$	$\alpha = 0.3$ $\mu = 0.7$	$\alpha = 0.7$ $\mu = 0.3$	$\alpha = 0.7$ $\mu = 0.7$
5	0.03	0.69	0.09	0.69
10	0.09	0.67	0.08	0.67

Tablica 5.5: Porównanie błędów regulatora $PI^{\lambda}D^{\mu}$

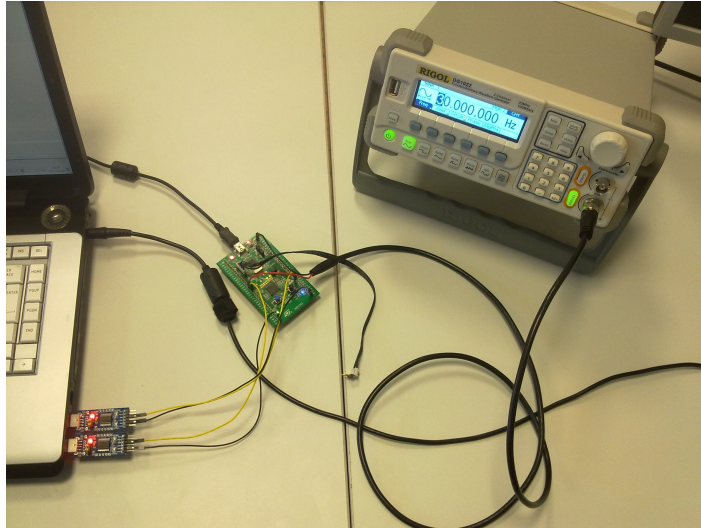
Główny program został napisany w języku C. Składa się z definicji globalnych tablic wymaganych do prawidłowego działania algorytmu, pustej pętli głównej i właściwej funkcji algorytmu, która jest wykonywana w podprogramie przzerwania zsynchronizowanym z przepełnieniem licznika wewnętrznego. Częstotliwość przepełnienia licznika skonfigurowano tak by odpowiadało 600 Hz. Gdy wystąpi przerwa związane przepełnienia zegara, próbkowany sygnał testowy zostanie odczytany przez wejście przetwornika analogowo-cyfrowego. Po zakończeniu konwersji analogowo-cyfrowej wywoływane jest drugie przerwanie. W tym podprogramie cyfrowa reprezentacja (po konwersji) jest przekazywana jako argument funkcji algorytmu filtracji.

Należy zauważyć, że algorytm w oryginalnej postaci działa na typie danych zmiennoprzecinkowych. Mikrokontroler STM32F0 nie ma rozszerzenia w postaci sprzętowej jednostki peryferyjnej zmiennoprzecinkowej. Dlatego wszystkie obliczenia dokonane na tym typie zmiennych absorbują znaczne ilości czasu i zasobów obliczeniowych. Niewątpliwie przyspieszenie algorytmu można osiągnąć za pomocą mikrokontrolera z wbudowanym FPU (jednostka zmiennoprzecinkowa).

Próbkowanie sygnału wejściowego o częstotliwości 600 Sa / s odbywa się za pomocą wbudowanego 12-bitowego przetwornika analogowo-cyfrowego SAR. Ze względu na brak precyzyjnego źródła napięcia odniesienia, maksymalna liczba bitów, które można realistycznie osiągnąć na platformie sprzętowej, wynosi około 10 bitów. Ta wartość jest wystarczająca (kwant próbkowania ok. 3 mV), aby zilustrować prawidłowe działanie algorytmu. Ustawienia sprzętowe użyte do testów algorytmów przedstawiono na rys. 5.14.

5.5. Wyniki eksperymentu

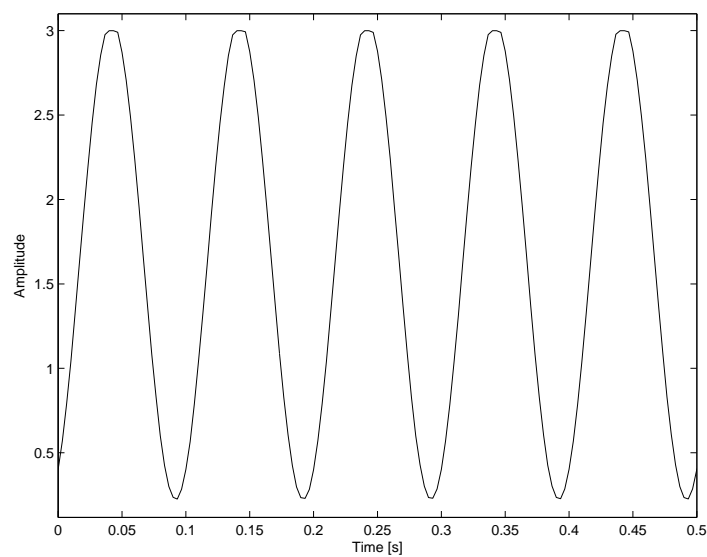
W tej sekcji zostanie przedstawiona implementacja aproksymacji Oustaloup w domenie czasu na platformie sprzętowej. Bez utraty ogólności analiza jest przeprowadzana dla filtru ułamkowego drugiego rzędu z $\alpha = 0.1$, $b = 0.1$ i $c = 0.1$. Parametry b i c zostały wybrane w taki sposób by zmaksymalizować błąd numeryczny. Częstotliwość próbkowania wybrano jako $f = \{15Hz, 150Hz, 300Hz\}$. Dla metody aproksymacji Oustaloupa pasmo częstotliwości zostało wybrane jako $[10^{-6}, 10^6]$, a rząd aproksymacji został wybrany w następujący sposób $N = 6, 10, 14$. Sygnały testowe wybrano jako:



Rysunek 5.14: Konfiguracja systemu pomiarowego

– Sinusoidalny (sin) (rys. 5.15)

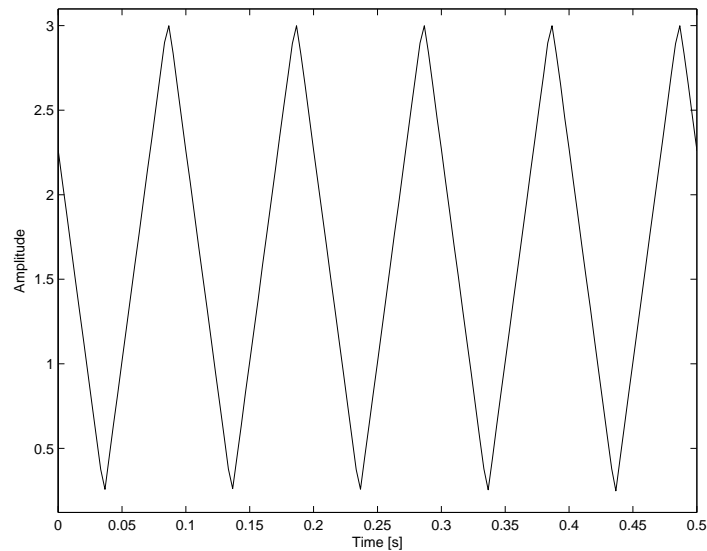
$$S(t) = A \sin(f\pi t)$$



Rysunek 5.15: Sinusoidalny sygnał testowy

– Trójkątny (triangle) (rys. 5.16)

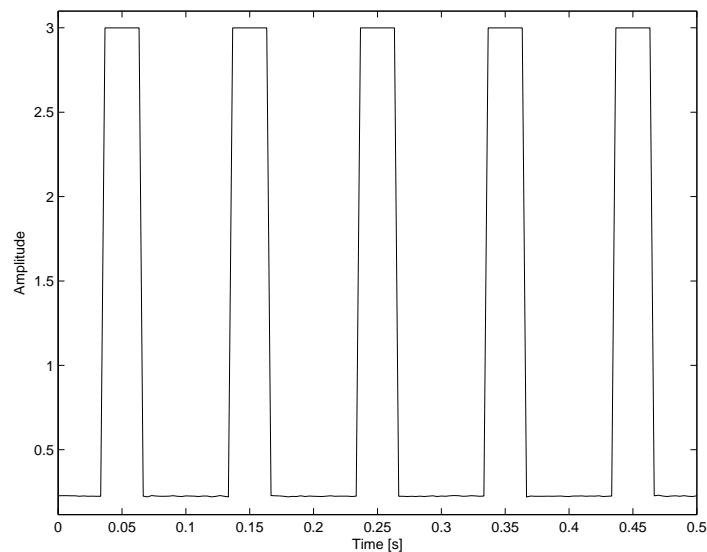
$$T(t) = \frac{2A}{\pi} \sin^{-1}[\sin(f\pi t)]$$



Rysunek 5.16: Trójkątny sygnał testowy

– Prostokątny (pulse) (rys. 5.17)

$$P(t) = A(-1)^{\lfloor 2(t-0.05)f \rfloor}$$



Rysunek 5.17: Prostokątny sygnał testowy

gdzie $f = \{15Hz, 150Hz, 300Hz\}$ i $A = 3V$.

Opisana aproksymacja Oustaloup w dziedzinie czasu została zdyskretyzowana przy użyciu metody Tustina.

Wynik przeprowadzonego filtrowania z PSD dla częstotliwości sygnałów 15 Hz i rzędów aproksymacji 6 i 14 pokazano na rys. 5.19 i 5.20 i rzędy przybliżenia 6 i 14 ilustrują rys.

5.23 i 5.24. Jak łatwo zauważyć we wszystkich prezentowanych przypadkach, sygnał nie jest znacząco zniekształcony.

Do oceny wyników zdefiniowano następujący wskaźnik jakości:

$$Q(SIM, STM) = \frac{|SIM - STM|}{L} \quad (5.3)$$

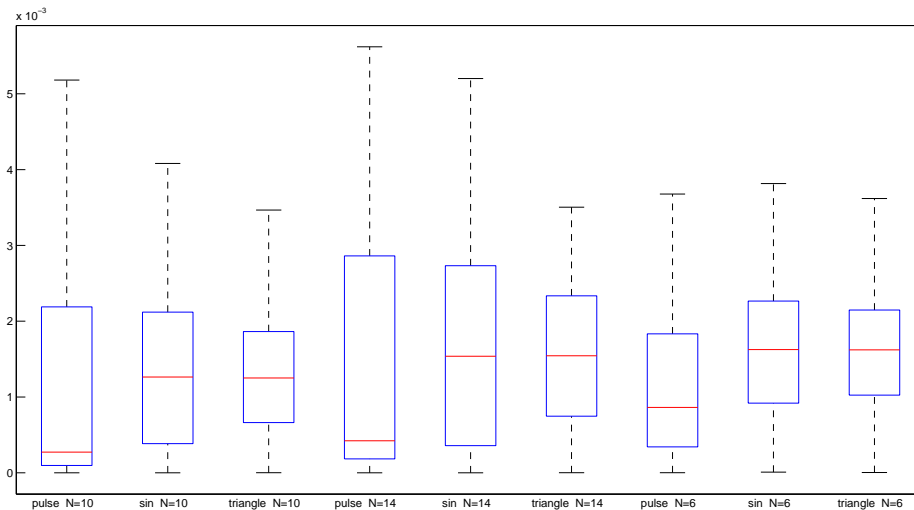
gdzie L jest długością sygnału, SIM jest sygnałem uzyskanym przez symulacje Matlab Simulink i STM jest rzeczywistym sygnałem z układu STM. Parametry statystyczne błędów między implementacją Matlab Simulink i mikrokontrolera STM dla przefiltrowanego sygnału przedstawiono w tab. 5.6. Ponieważ rozważane sygnały obejmują stany przejściowe do analiz wyników eksperymentów użyto analizę statystyczną której, wyniki obrazują wykresy pudełkowe. Przedstawiają one błędy z podziałem na częstotliwość próbkowania na rys.5.18, rys. 5.21 i rys. 5.22.

Na wykresie 5.18 można zauważyć, że maksymalna wartość błędu aproksymacji dla tego przypadku wynosi jedynie około 0.005 w przypadku sygnału prostokątnego z rzędem aproksymacji 15. Jednocześnie należy zauważyć że wartość błędów maksymalnych, dla sygnałów prostokątnych, zawsze jest najwyższa w zależności od rzędu. Natomiast wartość średnia dla tych sygnałów jest zawsze najniższa. Można więc sformułować wniosek, że sygnał prostokątny ze względu na swoją charakterystykę generuje duże błędy przy zmianie zbocza, jednak w pozostałych punktach pracy zaimplementowana metoda działa bardzo dobrze. Analiza wykresu 5.18 pozwala stwierdzić, że dla częstotliwości sygnałów 15Hz zaimplementowana metoda jest stabilna numerycznie i spełnia ograniczenia czasowe systemu czasu rzeczywistego. Dodatkowo na wykresach 5.19 i 5.20 obrazujących gęstość mocy sygnałów po filtracji w obydwóch środowiskach widać, że różnice pomiędzy nimi są znikome. Świadczy to o tym, że opracowana metoda implementacji układów niecałkowitego rzędu na układy wbudowane jest poprawna

Wykresy 5.21 i 5.22 obrazują bardzo zbliżone zachowanie błędów odpowiedzi filtra z układu wbudowanego co na wykresie 5.18. Zauważalne są dwie różnice po pierwsze maksymalny błąd zmniejszył się do wartości około 0.0042 dla częstotliwości sygnału 150Hz oraz do wartości około 0.0024 dla częstotliwości 300Hz. Wskazuje to na to, że przeprowadzone eksperymenty są zgodne z teorią przetwarzania sygnałów cyfrowych. Dodatkowo pojawiły się wartości odstające błędów które można pominąć w analizie ponieważ nie wpływają na poprawność otrzymanych rezultatów a ich występowanie jest związane z działaniem przetwornik analogowo-cyfrowego. Fakt ten potwierdzają wykresy gęstości mocy 5.23 i 5.24, które pokazują jednoznacznie, że filtr na systemie wbudowanym ma te same własności co filtr zaimplementowany w środowisku Matlab Simulink.

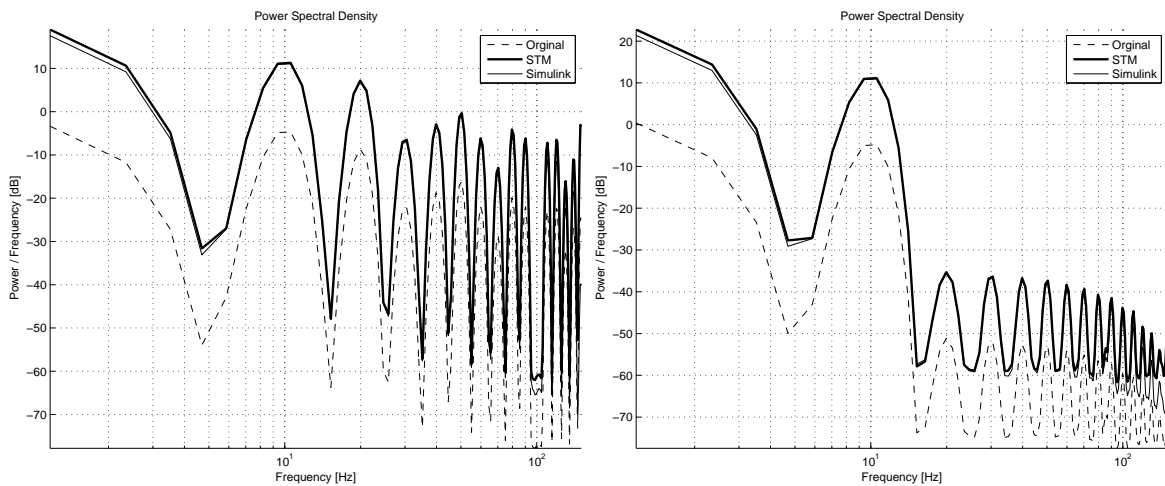
Hz	Rząd Aproksy.	Typ	Maksymalna wartość błędu	Średnia wartość błędu	Odchylenie standardowe błędu	Błąd IQR
15 Hz	6	Prostokątny	0.003	0.001	0.0006	0.0007
		Sinusoidalny	0.002	0.002	0.0004	0.0006
		Trójkątny	0.002	0.001	0.0003	0.0006
	10	Prostokątny	0.005	0.0004	0.0011	0.0014
		Sinusoidalny	0.002	0.0008	0.0007	0.0011
		Trójkątny	0.004	0.0008	0.0006	0.0007
	14	Prostokątny	0.004	0.0004	0.0011	0.0019
		Sinusoidalny	0.003	0.0008	0.0007	0.0014
		Trójkątny	0.004	0.0008	0.0007	0.0011
150 Hz	6	Prostokątny	0.004	0.0009	0.0009	0.0015
		Sinusoidalny	0.004	0.0016	0.0008	0.0013
		Trójkątny	0.004	0.0016	0.0007	0.0011
	10	Prostokątny	0.005	0.0003	0.0012	0.002
		Sinusoidalny	0.004	0.0003	0.0009	0.0017
		Trójkątny	0.003	0.0012	0.0007	0.0012
	14	Prostokątny	0.004	0.0004	0.0016	0.0027
		Sinusoidalny	0.005	0.0015	0.0011	0.0024
		Trójkątny	0.003	0.0015	0.0009	0.0016
300 Hz	6	Prostokątny	0.0036	0.0009	0.0006	0.0007
		Sinusoidalny	0.002	0.0012	0.0005	0.0006
		Trójkątny	0.0019	0.0012	0.0004	0.0006
	10	Prostokątny	0.0024	0.0005	0.0006	0.0009
		Sinusoidalny	0.0014	0.0005	0.0004	0.0006
		Trójkątny	0.0019	0.0005	0.0003	0.0006
	14	Prostokątny	0.002	0.0003	0.0005	0.0003
		Sinusoidalny	0.0011	0.0001	0.0001	0.0002
		Trójkątny	0.0017	0.0001	0.0001	0.0001

Tablica 5.6: Analiza statystyczna przeprowadzonych eksperymentów



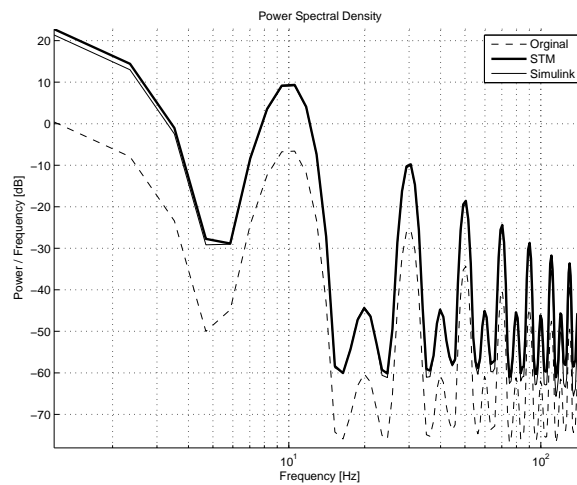
Rysunek 5.18: Analiza błęd częstotliwości próbkowania 15 Hz

Podsumowując z przeprowadzonych eksperymentów jednoznacznie wynika, że zaproponowane metody aproksymacji układów niecałkowitego rzędu odwzorowują własności dynamiczne przybliżanych układów. Pokazano również, że zaproponowana metoda dyskretyzacji zachowuje stabilność systemu. Na końcu potwierdzono również, że opracowane algorytmy implementacji dyskretnego systemu na systemie wbudowanym są poprawne. Należy zaznaczyć, że otrzymane wyniki są satysfakcjonujące, ponieważ żaden z testowanych układów wbudowanych nie posiadał reprezentacji liczb zmiennoprzecinkowej podwójnej precyzji. Mimo to różnice w odpowiedziach ze środowiska Matlab Simulink dysponującym tym typem zmiennych a implementacjami na układach wbudowanych były niewielkie.



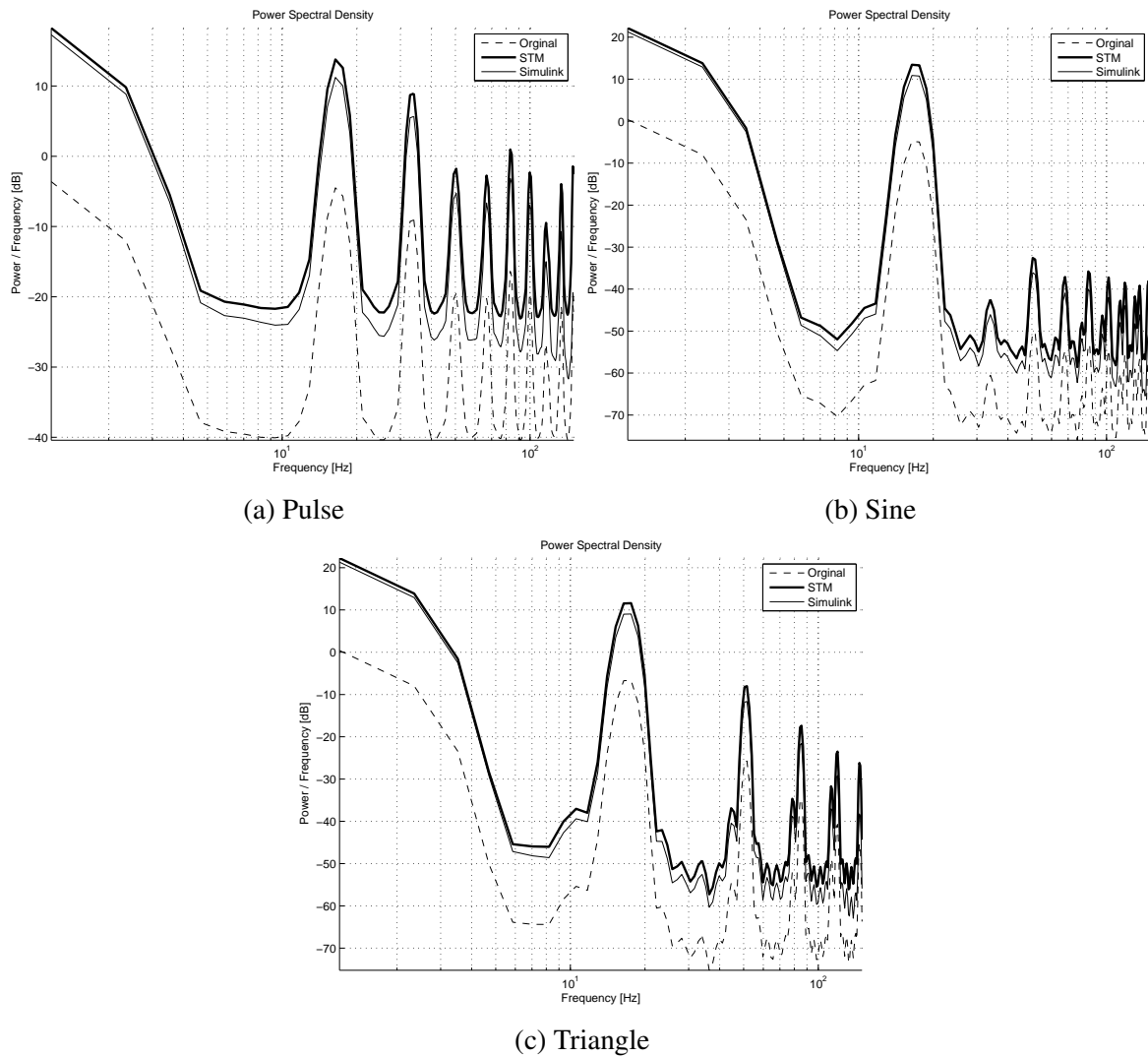
(a) Sygnał prostokątny

(b) Sygnał sinusoidalny

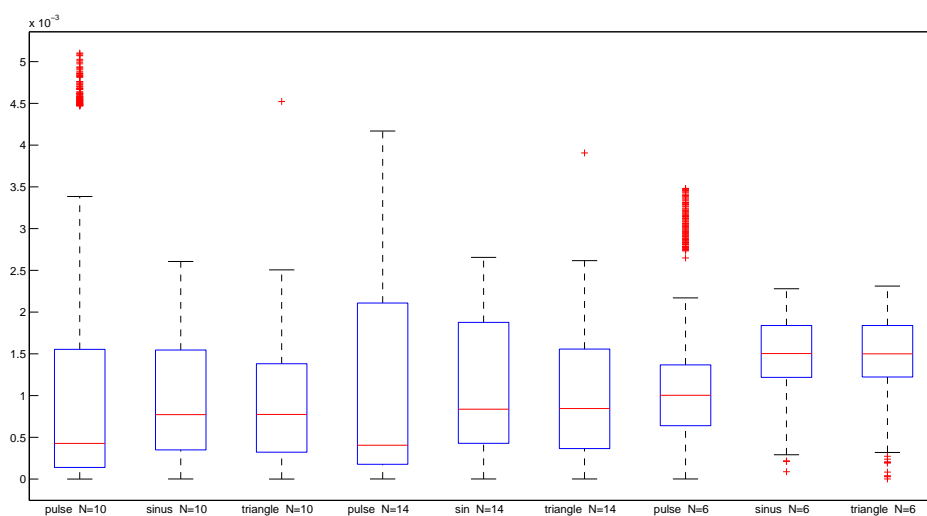


(c) Sygnał trójkątny

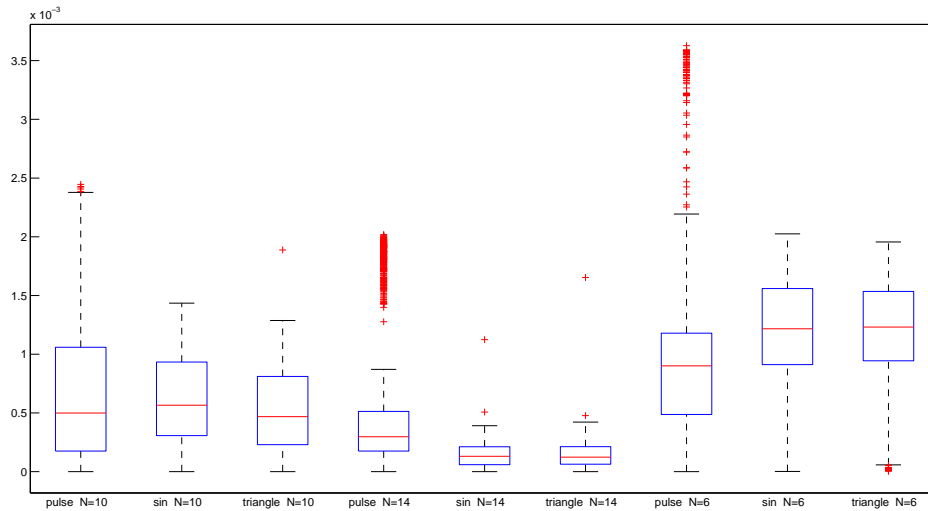
Rysunek 5.19: Porównanie działania aproksymacji w Matlab Simulink i STM dla 15 Hz i rzędzie aproksymacji 6



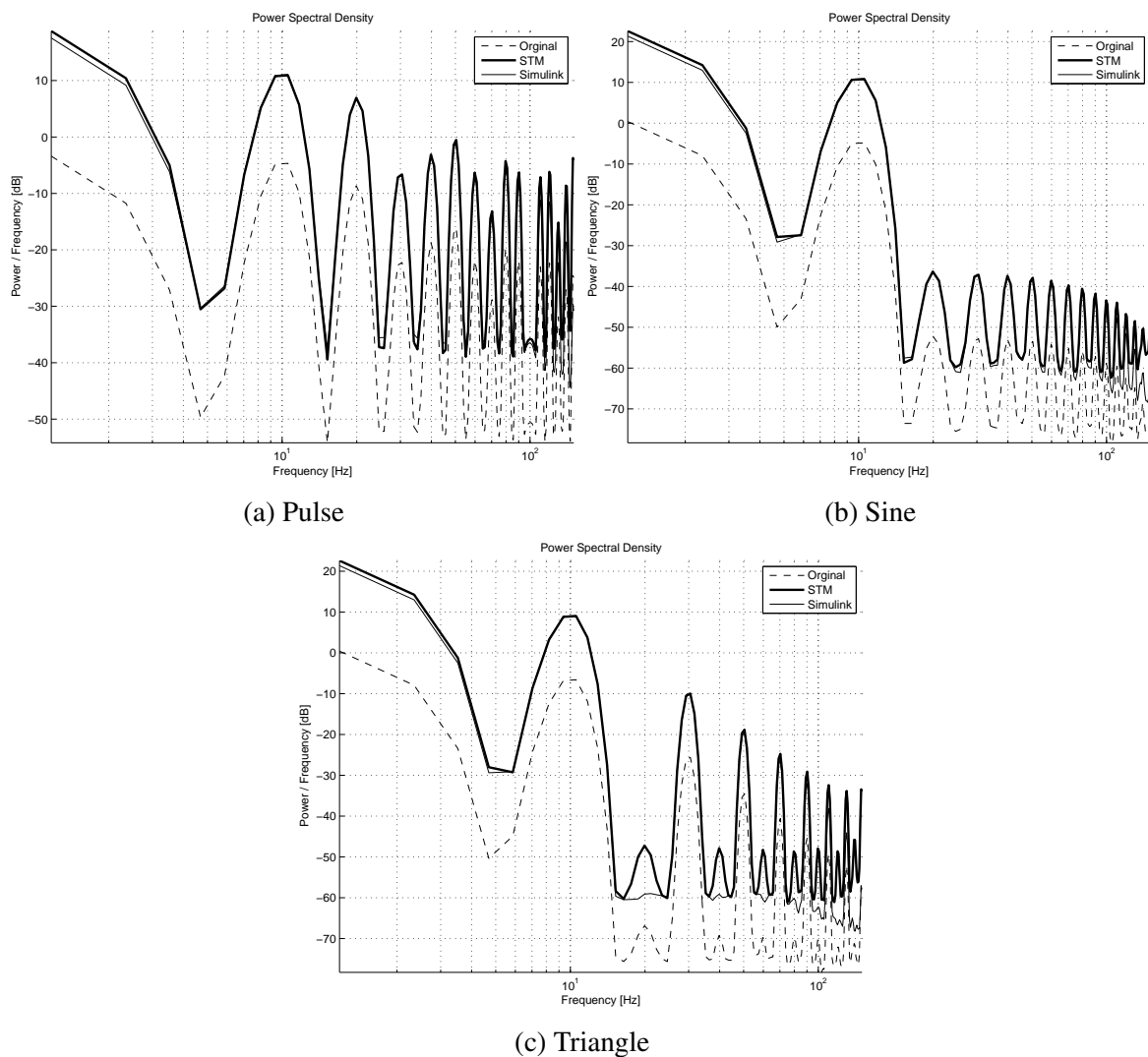
Rysunek 5.20: Porównanie działania aproksymacji w Matlab Simulink i STM dla 15 Hz i rzędzie aproksymacji 15



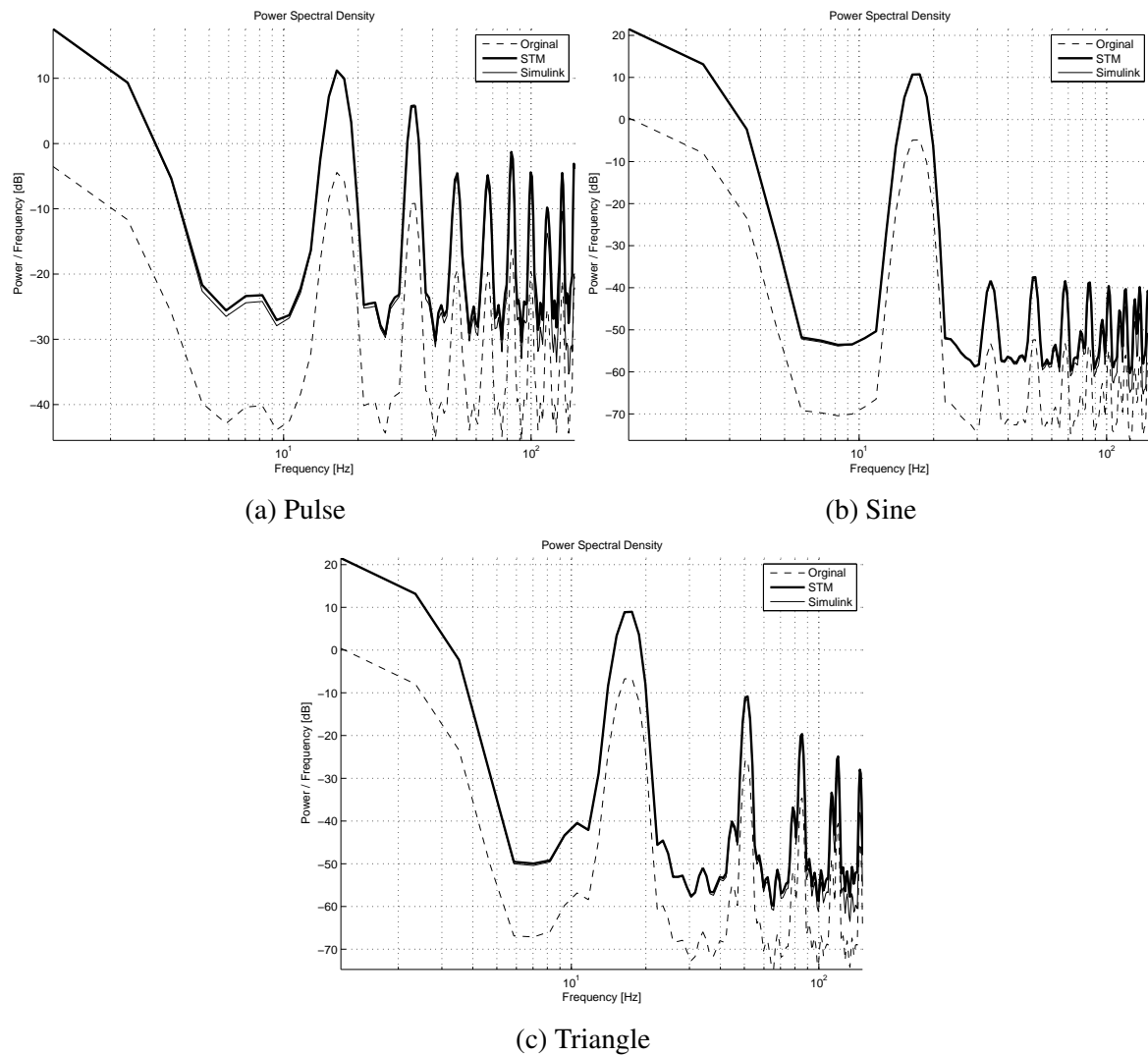
Rysunek 5.21: Analiza błędów częstotliwości próbkowania 150 Hz



Rysunek 5.22: Analiza błędów częstotliwości próbkowania 300 Hz



Rysunek 5.23: Porównanie działania aproksymacji w Matlab Simulink i STM dla 300 Hz i rzędzie aproksymacji 6



Rysunek 5.24: Porównanie działania aproksymacji w Matlab Simulink i STM dla 300 Hz i rzędzie aproksymacji 14

6. Wykorzystanie ułamkowych regulatorów ułamkowych typu PID w sterowaniu rzeczywistymi obiektami

6.1. Realizacja regulatora ułamkowego PID dla systemu nagrzewnicy powietrznej

Projektowanie systemu sterowania dla systemu nagrzewnicy powietrznej wiąże się z dwoma głównymi problemami. Po pierwsze, system dynamiczny który go opisuje jest niestacjonarny, ze względu na nagrzewanie się linia przesyłowej ciepłego powietrza. Po drugie problem optymalizacji staje się problem nieskończenie wymiarowym, przez obecność ciągłego opóźnienia czasowego w systemie dynamicznym. Dlatego projektowany układ regulacji powinien być odporny na zmiany parametrów instalacji w trakcie trwania procesu grzania.

Bode w pracy (Bode, 1945) omawia idealny kształt funkcji transmitancji dla pętli sprzężenia zwrotnego, aby system był stabilny i niewrażliwy na zmiany wzmocnień w systemie. Proponowana funkcja transferu została później nazwana idealną charakterystyką odcięcia Bodego, (Åström, 2000), i jest opisana przy pomocy zależności:

$$L(s) = \left(\frac{s}{\omega_{gc}} \right)^n \quad (6.1)$$

W tej pracy odporność regulatora zyskuje się przez zwiększenie złożoności regulatora. Zamiast klasycznego regulatora PID użyto regulatora ułamkowego. Metodę strojenia regulatora ułamkowego rzędu typu PID za pomocą optymalizacji wielokryterialnej, bazującej na warunkach stabilności układu zamkniętego została omówiona przez Meng i Xue (2009). Dobór parametrów regulatora $PI^\alpha D^\mu$ bazuje na rozważaniach opisanych przez Dziwiński et al. (2014a) oraz Bauer et al. (2013) dla wskaźnika jakości typu całki z modułu uchybu regulacji. Należy zaznaczyć, że w literaturze przedstawione są również inne metody rozwiązania tego zagadnień, jak w pracy Barbosa et al. (2004), gdzie autor rozważa optymalizację z ograniczeniami do wyliczenia odpornego systemu sterowania. Bazując na tych pracach dla tego

układu wybrano, a następnie wykorzystano trzy kryteria jakości: płaskości i marginesu fazy oraz zapasu wzmocnienia.

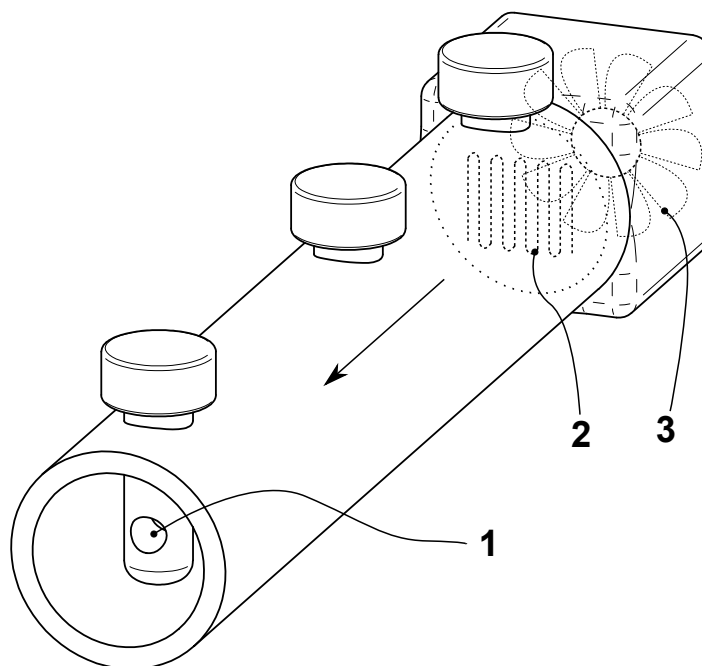
6.1.1. Model nagrzewnicy powietrznej

Proces nagrzewania powietrznego można w uproszczeniu przedstawić jako system liniowy z opóźnieniem (LTI), który jest opisany przy pomocy transmitancji postaci:

$$P(s) = \frac{K}{(T_1s + 1)(T_2s + 1)} e^{-s\tau}, \quad (6.2)$$

gdzie $K=18.8$, $T_1=7.783$, $T_2=0.0014$ and $\tau=0.5842$. Parametry zostały zidentyfikowane w eksperymentach, a proces dokładnie opisany w pracy Dziwiński et al. (2014a).

Rozważany układ przedstawiony jest na rysunku 6.1 i składa się z pustej rurki, czujnika temperatury, nagrzewnicy i wentylatora wymuszającego ruch ogrzanego powietrza. Na zdjęciu 6.2 znajduje się układ nagrzewnicy powietrznej, która została użyta w eksperymentach. Charakterystykę częstotliwościową omawianego układu obrazuje rys. 6.3

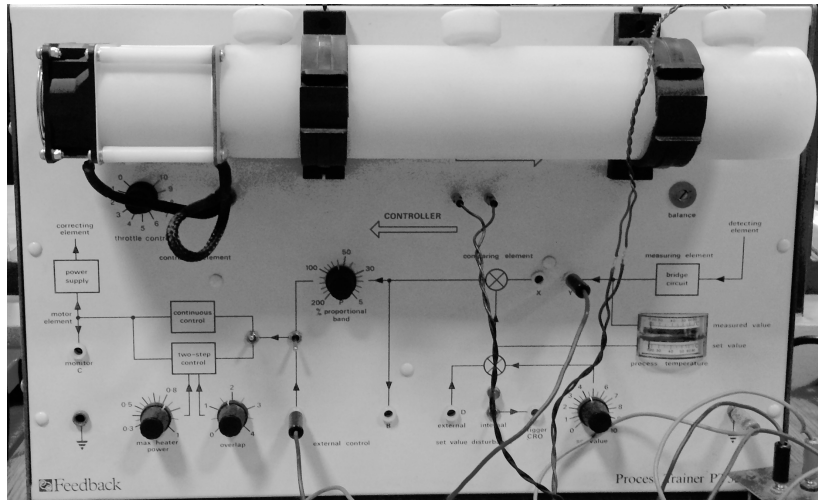


Rysunek 6.1: Schemat układu: 1 – termo rezystor, 2 – element grzewczy, 3 – wiatrak

Schemat blokowy systemu sterowania sprzężeniem zwrotnym pokazano na rysunku 6.4.

Do rozwiązania problemu doboru optymalnych nastaw regulatora sformułowane zostały następujące wymagania dotyczące charakterystyki częstotliwościowej regulatora:

1. zdefiniowany margines fazy ϕ_m i margines wzmocnienia g_m ,
2. odporność na zmiany parametrów w czasie pracy układu,
3. odporność na zakłócenia dla wysokich częstotliwości,



Rysunek 6.2: Nagrzewnica powietrzna PT326

4. odporność na zakłócenia wyjścia,
5. anulowanie błędu w stanie ustalonym.

Stabilność układu zamkniętego została obliczona poprzez klasyczną analizę zapasu fazy i wzmocnienia w układzie otwartym sterowania. Ponieważ oczywistym jest, że chcemy by układ posiadał jak największy zapas fazy i wzmocnienia przyjęto następujące wskaźniki jakości, bazujące na założeniach idealnej charakterystyki Bodego (Luo et al., 2014):

- Zapas wzmocnienia

$$|C(j\omega)P(j\omega)| \Big|_{\omega=\omega_{pm}} = 12\text{dB} \quad (6.3)$$

- Zapas fazy

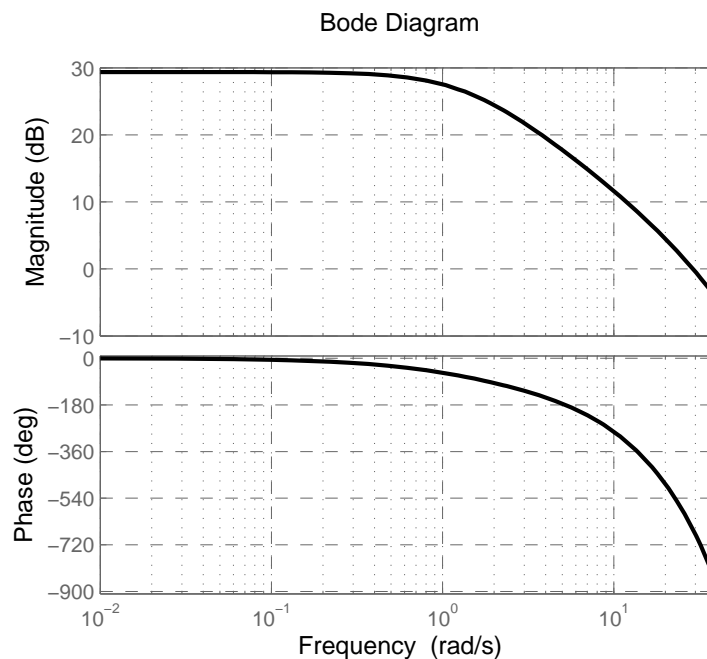
$$\arg(C(j\omega)P(j\omega)) \Big|_{\omega=\omega_{gm}} = 45^\circ \quad (6.4)$$

- Współczynniki płaskości fazy z częstotliwością rozgraniczającą wzmocnienia (**ang.** *iso-damping*)

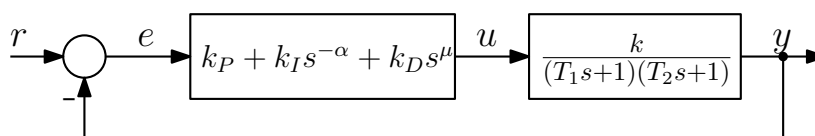
$$\frac{d \arg(C(j\omega)P(j\omega))}{d\omega} \Big|_{\omega=\omega_{gm}} = 0 \quad (6.5)$$

Kryteria (6.3) i (6.4) określają zapas stabilności przyjęty do eksperymentów. Natomiast warunek (6.5), odpowiedzialny jest za płaskość fazy z częstotliwością rozgraniczającą wzmocnienia, który określa odporność układu regulacji po zamknięciu pętli sprzężenia zwrotnego na zmianę wzmocnienia w czasie procesu nagrzewania (Barbosa et al., 2004).

Do przeprowadzenia optymalizacji regulatora $PI^\alpha D^\mu$ użyto metody symulowanego wyżarzania (**ang.** *Simulated Annealing Optimisation*, (Kirkpatrick et al., 1983)). W zadanym procesie optymalizacji minimalizował funkcję celu, która poprzez wagi na poszczególnych funkcjach opisujących idealną charakterystykę Bodego, odzwierciedlała stopień w jakim chciano by wpływały one na zachowanie kontrolera PID niecałkowitego rzędu. Poprzez ten



Rysunek 6.3: Odpowiedź częstotliwościowa dla rozpatrywanego systemu



Rysunek 6.4: Układ sterowania z regulatorem PID ułamkowego rzędu dla nagrzewnicy powietrznej

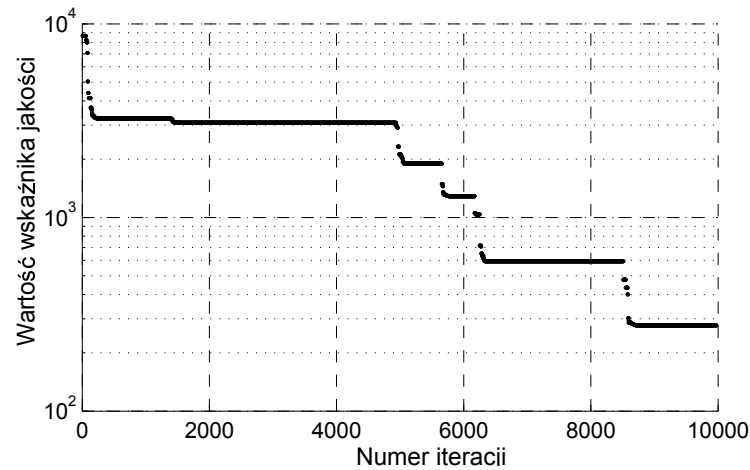
proces udało się uzyskać system z kontrolerem odpornym na zakłócenia procesowe, którego charakterystyka Nyquista pokazana jest na wykresie 6.14. Zmianę wartości funkcji celu w poszczególnych iteracjach pokazano na rysunku 6.5.

6.1.2. Funkcja czułości dla nagrzewnicy powietrznej

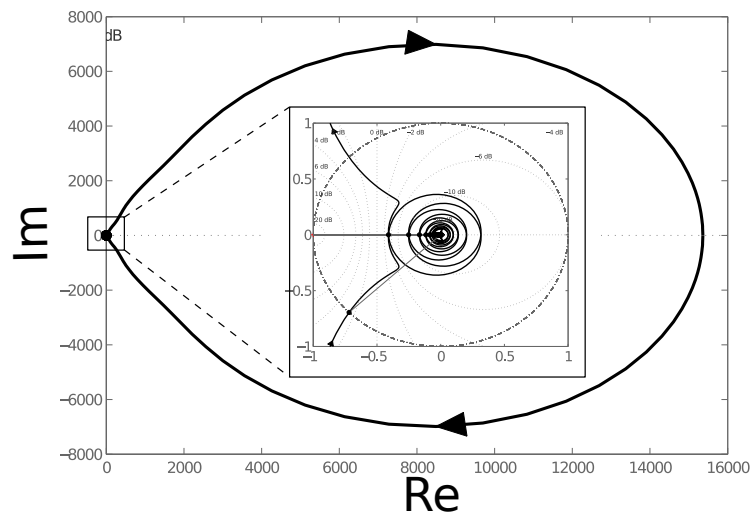
Funkcja czułości dla nagrzewnicy powietrznej S , daną równaniem (6.7), opisuje wpływ zakłóceń na sterowanie w układzie ze sprzężeniem zwrotnym. Mówiąc bardziej precyzyjnie opisuje ona czy zakłócenia w układzie zamkniętym są wzmacniane czy tłumione. Natomiast

Margines Fazy	45.956°
Częstotliwość graniczna dla fazy	1.15 Hz
Margines wzmocnienia	-8.052 dB
Częstotliwość graniczna dla wzmocnienia	0.32 Hz
Wartość kątowna fazy w ω_{cg}	-0.398°

Tablica 6.1: Uzyskane marginesy stabilności



Rysunek 6.5: Postęp optymalizacji



Rysunek 6.6: Charakterystyka Nyquista z zaznaczonym marginesem stabilności

komplementarna funkcja czułości T , opisana równaniem (6.6), określa w jaki sposób błędy pomiaru wpływają na działanie sprzężenia zwrotnego.

$$S(s) = \frac{E(s)}{R(s)} = \frac{1}{1 + C(s)P(s)} \quad (6.6)$$

$$T(s) = \frac{Y(s)}{R(s)} = \frac{C(s)P(s)}{1 + C(s)P(s)} \quad (6.7)$$

Maksymalne wartości omówionych funkcji mogą służyć jako miara odporności układu sterowania, którą możemy zdefiniować jako:

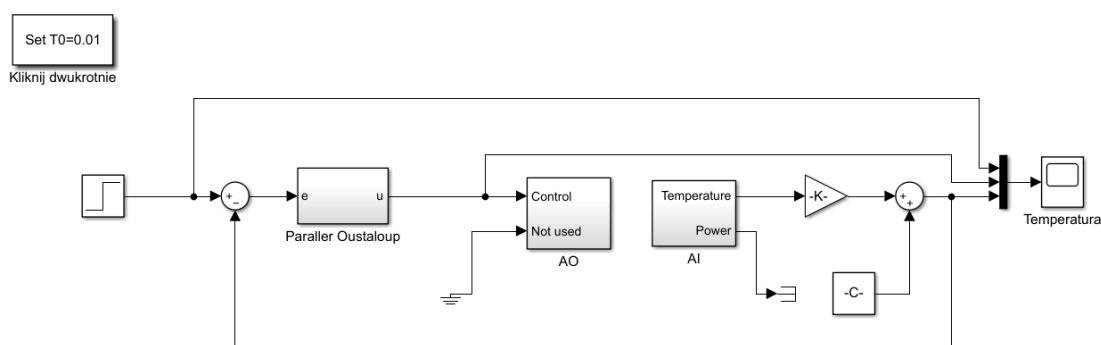
$$M_S = \sup_{\omega} |S(j\omega)| \quad M_T = \sup_{\omega} |T(j\omega)| \quad (6.8)$$

K_P	0.0035
K_I	0.1268
K_D	0.0206
α	0.7229
μ	0.7307

Tablica 6.2: Optymalne parametry kontrolera

6.1.3. Implementacja regulatora $PI^\alpha D^\mu$ i wyniki eksperymentu

Do sterowania laboratoryjnym układem nagrzewnicy powietrza regulator $PI^\alpha D^\mu$ zostały zaimplementowany dwiema metodami: klasyczną metodą Oustaloupa (3.7) oraz równoległą metodą Oustaloupa(3.8.1). Regulatory te zostały zaimplementowane w taki sposób by rząd aproksymacji był sobie równy i wynosił $N = 8$ oraz by wzmocnienia poszczególnych członów były wyznaczone w procesie optymalizacji. Do porównania zaimplementowano również regulator PID całkowitego rzędu.

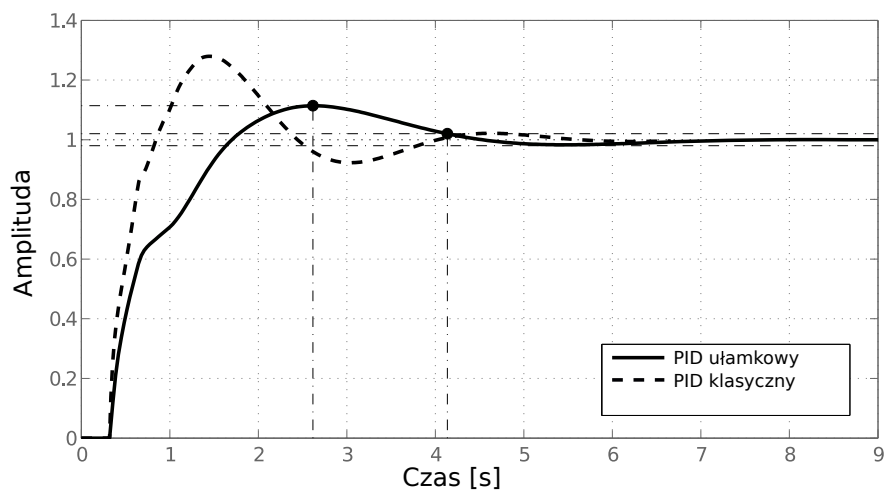


Rysunek 6.7: Realizacja układu regulacji niecałkowitego rzędu dla laboratoryjnego systemu nagrzewnicy powietrza w Matlab Simulink

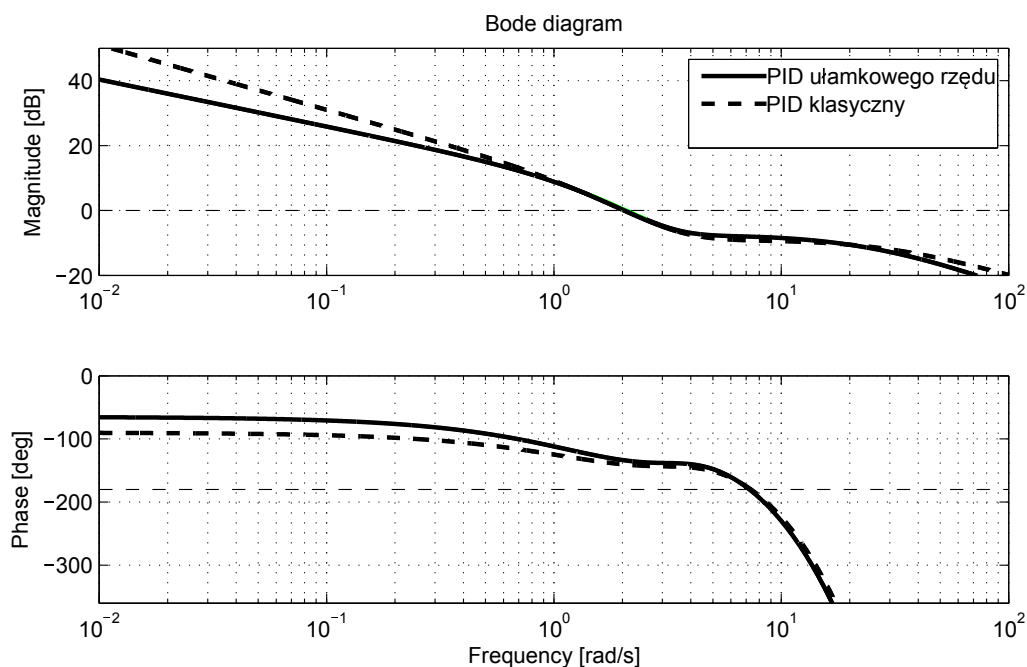
Optymalne ustawienia regulatora zebrano w tabeli 6.2, natomiast wskaźniki odporności przedstawione są w tabeli 6.1. Odpowiedź skokowa symulowanego systemu zamkniętego jest pokazana na rysunku 6.8. Charakterystyka Bodego dla otwartej pętli sterowania systemu przedstawiona jest na rysunku 6.9. Można zauważyć, że charakterystyka wzmocnienia układu sterowania z idealnym regulatorem niecałkowitego rzędu ma nachylenie 14,5dB na dekadę w paśmie niskich częstotliwości, podczas gdy klasyczny regulator PID 20dB na dekadę. Szczegółowe porównanie PID i $PI^\alpha D^\mu$ przedstawiono w tabeli 6.3.

Po analizie powyższej tabeli można zauważyć, że regulator typu $PI^\alpha D^\mu$ osiąga lepsze wyniki w trzech z czterech parametrów zdefiniowanych do opisu jakości sterowania. Jednak należy zauważyć również, że różnica wartości we wskaźniku opisującym czułość układu na zakłócenia M_S jest niewielka.

Wskaźnik	PID	$PI^\alpha D^\mu$
M_S	1.64	1.66
M_T	1.50	1.30
σ [%]	28	11
tr [s]	0.72	0.68

Tablica 6.3: Porównanie zoptymalizowanego PID i zoptymalizowanego $PI^\alpha D^\mu$ 

Rysunek 6.8: Porównanie odpowiedzi skokowych

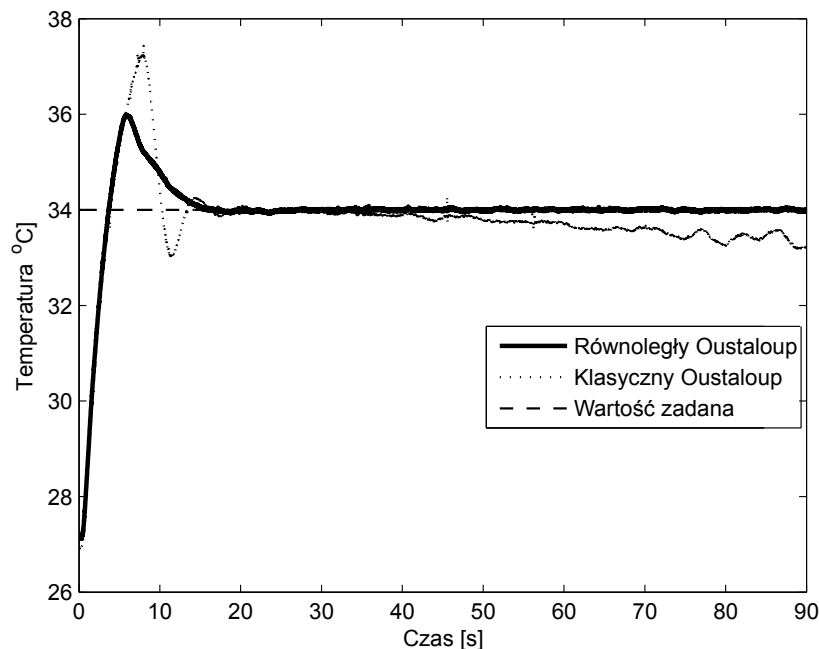


Rysunek 6.9: Porównanie odpowiedzi częstotliwościowych

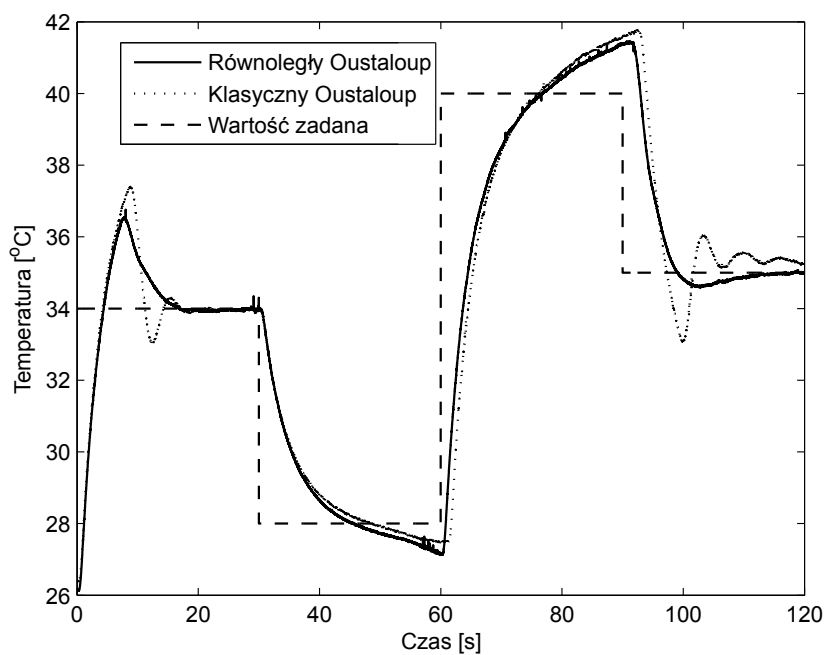
Jakość zaimplementowanych regulatorów dla laboratoryjnego systemu grzewczego zbadano na podstawie szeregu eksperymentów na urządzeniu. Pierwszy z nich sprawdzał za-

chowanie się układu na wymuszenie w postaci skoków wartości wartości zadanej, wynik zaprezentowano w rysunkach 6.10 i 6.11. Jak można zauważyć implementacja przy pomocy metody równoległej Oustaloupa dla tych samych parametrów strojenia regulatora posiada większą stabilność numeryczną niż metoda klasyczna. Można to zaobserwować na rysunku 6.10 w 40 sekundzie. Wynika to z faktu grupowania biegunów w pobliżu $-\omega_b$ w klasycznej metodzie Oustaloupa, implementacja równoległej implementacji problem ten nie występuje.

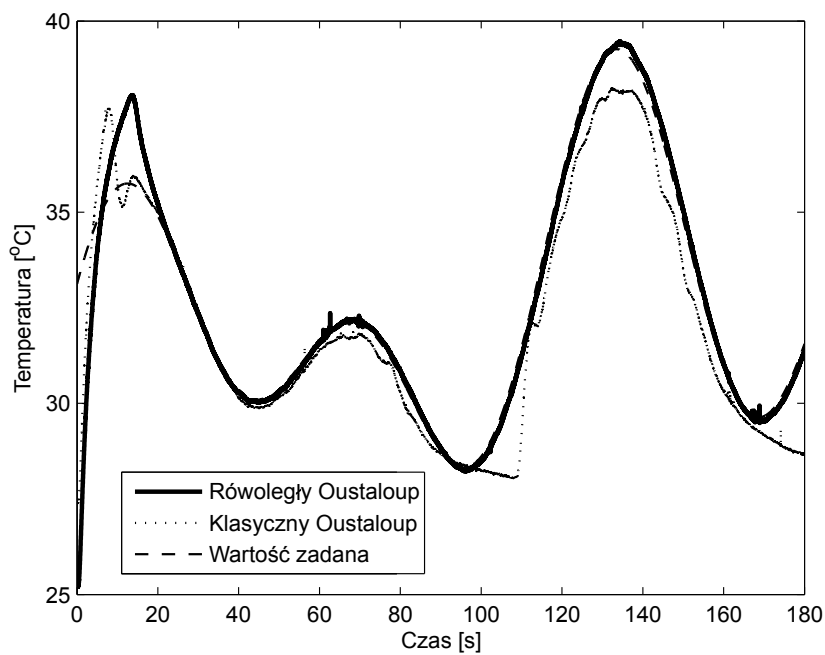
Wynik ostatniego eksperymentu przedstawiony na rysunku 6.12 obrazuje zadanie nadążania sterowania za sygnałem referencyjnym. Po analizie wykresu można stwierdzić, że implementacja równoległa radzi sobie lepiej z tym zadaniem. To znowu dowodzi, że jej odporność numeryczna jest większa niż w przypadku metody klasycznej. Dodatkowe publikacje autora na ten temat można znaleźć w (Kania et al., 2014; Dziwiński et al., 2015; Dziwiński et al., 2014a,b; Bauer, 2015).



Rysunek 6.10: Porównanie wydajności kontrolerów dla stałej wartości wymuszenia



Rysunek 6.11: Porównanie na szereg odpowiedzi skokowych



Rysunek 6.12: Porównanie dla zadania nadążania

6.2. Realizacja regulatora $PI^\lambda D$ dla układu lewitacji magnetycznej

W tej sekcji zostanie omówiony proces implementacji regulatora niecałkowitego rzędu typu $PI^\lambda D$ dla laboratoryjnego układu lewitacji magnetycznej. Przedstawione zostaną zasady tworzenia tego typu regulatora. Na końcu zostanie również opisany proces redukcji rzędu aproksymacji regulatora, a następnie zredukowany regulator zostanie porównany z regulatorem otrzymanym bezpośrednio z aproksymacji.

6.2.1. Matematyczny model lewitacji magnetycznej

W sekcji tej omówiony zostanie model układu lewitacji magnetycznej, składający się z elektromagnesu, kuli ferromagnetycznej, przetwornika prądu oraz układu pomiarowego pozycji wykorzystującego czujnik światła i źródło światła.

Model matematyczny tego obiektu, opiera się na podstawowej zależności z drugiej zasady dynamiki Newtona. Jeżeli zdefiniujemy $x_1(t)$ jako pozycję piłki rozumianą jako odległość między piłką a elektromagnesem i $x_3(t)$ będzie prądem cewki elektromagnesu to $F_l(x_1(t), x_3(t))$ rozumiana jest jako siła generowana przez elektromagnes, m jest masą kuli oraz g jest przyspieszenie grawitacyjne. Więc siła generowana przez elektromagnes można opisać następującą relacją (Piłat, 2002):

$$F_l(x_1(t), x_3(t)) = -\frac{a}{mb} \exp\left(-\frac{x_1(t)}{b}\right) x_3^2(t) + g \quad (6.9)$$

Parametry a i b określono przez analizę szeregu punktów stanu ustalonego układu z zamkniętą pętlą sprzężenia zwrotnego. Funkcja wykładnicza została dopasowana do tych punktów poprzez minimalizację najmniejszych kwadratów.

Prąd cewki w układzie zazwyczaj jest zależny od wielu czynników, takich jak zmiany indukcyjności, prędkości i inne. Jednak system laboratoryjny lewitacji jest wyposażony w dedykowany sterownik prądu co pozwala na zaniechanie wpływu tych zakłóceń w rozpatrywanym modelu. Takie podejście jest bardzo popularne (Baranowski i Piątek, 2008), ponieważ umożliwia opracowanie modelu niższego rzędu oraz znacznie upraszcza model. W sytuacji optymalnej sterownik prądu cewki powinien pozwolić na pełną kontrolę prądu w elektromagnesie, jednak w rzeczywistości element ten posiada własną dynamikę, którą należało by zamodelować. W rozważanym systemie dynamika ta może zostać opisana za pomocą systemu pierwszego rzędu, podanego w następujący sposób:

$$\dot{x}_3(t) = \frac{\exp\left(\frac{x_1(t)}{p}\right)}{T_s} (k_s u(t) + i_s - x_3(t)) \quad (6.10)$$

gdzie $u(t)$ jest napięciem sterowania, k_s jest wzmacnieniem sterowania cewki, T_s jest stałą czasową regulatora cewki i i_s jest wartością prądu na cewce przy zerowym błędzie jej regulatora, natomiast p jest dodatnią stałą. Prędkość sfery ferromagnetycznej x_2 jest pierwszą pochodną pozycji więc możemy zbudować model przestrzeni stanów. Nieliniowy system lewitacji magnetycznej opisany przez układ nieliniowych równań różniczkowych ma postać:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} x_2(t) \\ -\frac{a}{mb} \exp\left(-\frac{x_1(t)}{b}\right) x_3^2(t) + g \\ \frac{\exp\left(\frac{x_1(t)}{p}\right)}{T_s} (k_s u(t) + i_s - x_3(t)) \end{bmatrix} \quad (6.11)$$

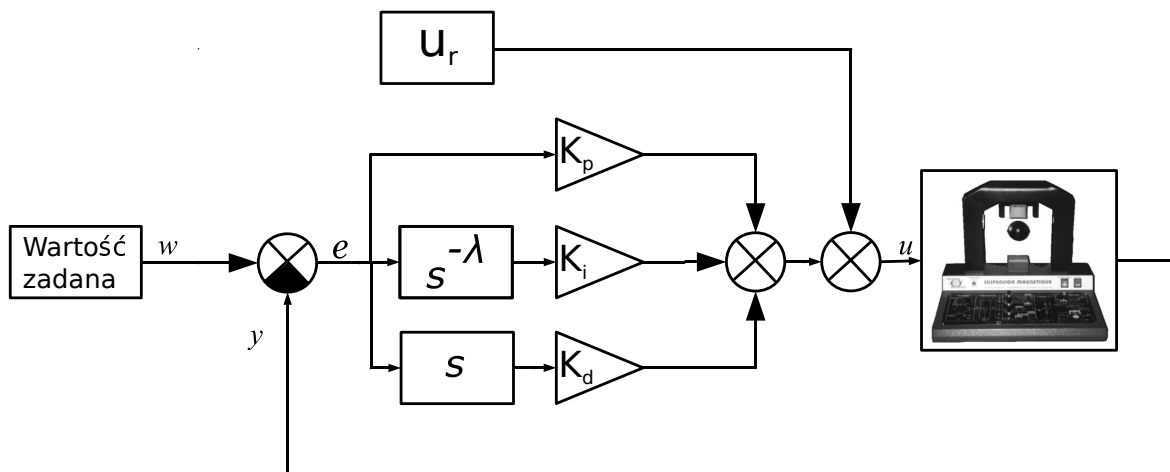
gdzie :

- $x_1(t)$ - pozycja sfery,
- $x_2(t)$ - prędkość sfery,
- $x_3(t)$ - prąd cewki,
- $u(t)$ - napięcie sterowania,
- m - waga sfery,
- g - przyspieszenie grawitacyjne,
- k_s - wzmacnienie regulatora prądu na cewce,
- T_s - stała czasowa liniowego sterownika prądu na cewce,
- i_s - wartość prądu na cewce przy zerowym błędzie regulatora,
- a, b, p - dodatnie zmienne pomocnicze.

6.2.2. Dobór parametrów regulatora

Do wyliczenia nastaw regulatora zostanie użyte liniowe przybliżenie układu w punkcie równowagi odpowiadającemu zadanej pozycji pracy. Zlinearyzowany system lewitacji magnetycznej w takim punkcie można opisać za pomocą następującej transmitancji:

$$G(s) = \frac{ve^{x_{1r}/p} k_s}{T_s s^3 + e^{x_{1r}/p} s^2 - T_s n s - ne^{x_{1r}/p}} \quad (6.12)$$

Rysunek 6.13: System lewitacji magnetycznej z regulatorem ułamkowym typu $PI^\lambda D$

Gdzie x_{1r} , x_{3r} , u_r są wartościami stanu ustalonego wybranego punktu pracy i v, n mają wartość:

$$v = -2 \frac{a}{bm} \exp\left(\frac{-x_{1r}}{b}\right) x_{3r}^2 \quad (6.13)$$

$$n = \frac{a}{b^2 m} \exp\left(\frac{-x_{1r}}{b}\right) x_{3r}^2 \quad (6.14)$$

Do sterowania omawianego układu został użyty regulator typu $PI^\lambda D$, opisany przy pomocy zależności:

$$G_{PI^\lambda D}(s) = K_p + K_i s^{-\lambda} + K_d s \quad (6.15)$$

Regulator ten został wybrany ponieważ układ lewitacji magnetycznej jest niestabilny w sposób naturalny, dlatego też niezbędne będzie użycie członu różniczkującego. Natomiast człon całkujący może być ułamkowego rzędu ponieważ zakłócenia w systemie nie są typu obciążeniowego.

Do wyznaczenia dopuszczalnych wartości nastaw regulatora podjęto następujące kroki:

1. Został zaprojektowany klasyczny kontroler PD, którego zadaniem jest ustabilizowanie systemu.

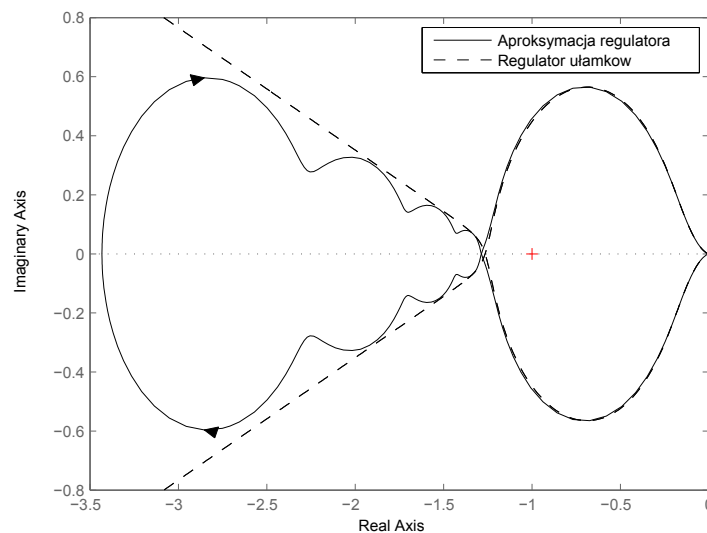
2. Zależności pomiędzy wzmocnieniami regulatora dla osiągnięcia stanu asymptotycznej stabilności mają postać:

$$K_p < \frac{n}{k_s p} \quad (6.16)$$

$$K_d < \frac{T_s n}{k_s p} e^{-\frac{x_{1r}}{p}} \quad (6.17)$$

$$K_d < T_s \cdot K_p e^{-\frac{x_{1r}}{p}} \quad (6.18)$$

3. Na podstawie rysunku 6.14 i kryterium Nyquista został dobrany rząd λ i wartość wzmocnienia K_i dla integratora niecałkowitego rzędu tak, by uzyskać rozsądny zakres stabilności.



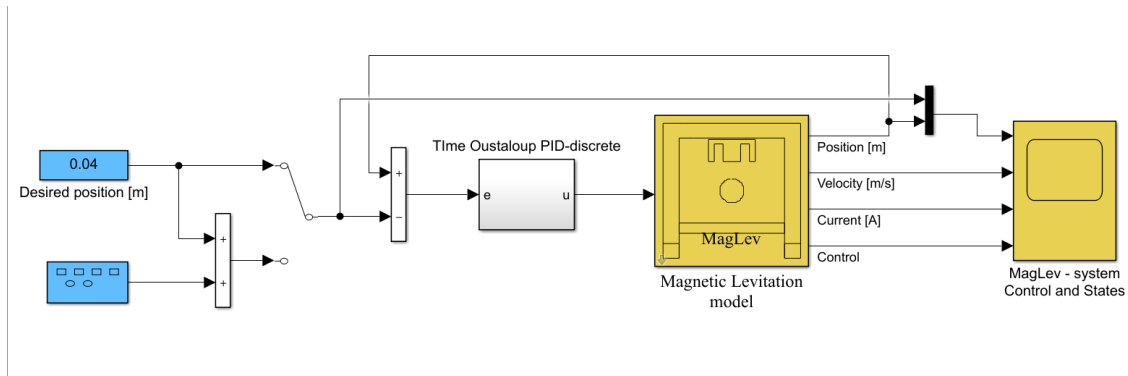
Rysunek 6.14: Wykres Nyquista dla rzeczywistej i aproksymowanej postaci $PI^\lambda D$ controller

Wykres Nyquista zaprojektowanego idealnego (nieaproksymowanego) regulatora niecałkowitego rzędu jest przedstawiony na rysunku 6.14 linią przerywaną.

6.2.3. Implementacja zaprojektowanego regulatora

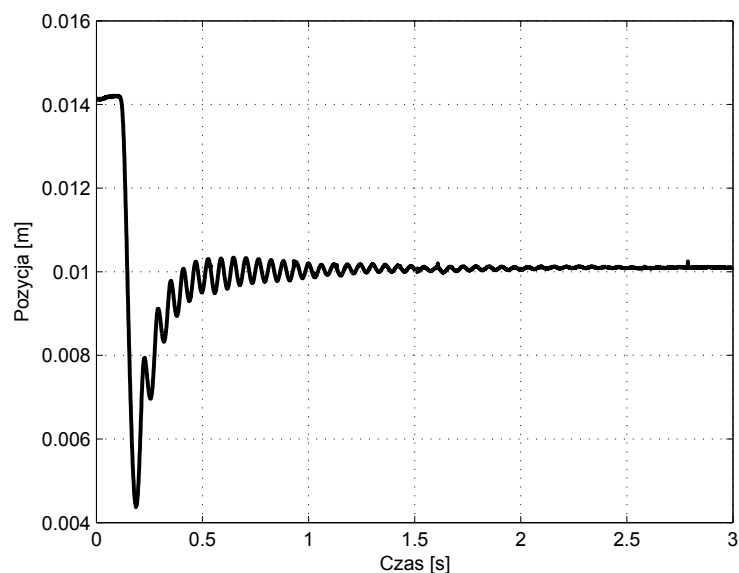
Na tym etapie prac ustalono następujące zasady eksperymentów dla określenia jakości opracowanego regulatora: do implementacji integratora niecałkowitego rzędu posłuży czasowa metoda Oustaloupa, rząd aproksymacji będzie wynosił $N = 8$, zakres częstotliwości dla których przeprowadzona zostanie aproksymacją będzie znajdował się w przedziale $\omega \in [10^{-6}, 10^6]$ natomiast czas dyskretyzacji wyniesie 0.001 sekundy. Na rysunku 6.14

można łatwo zauważyć, że ta metoda dyskretyzacji daje zadowalające wyniki pod względem odporności regulatora.



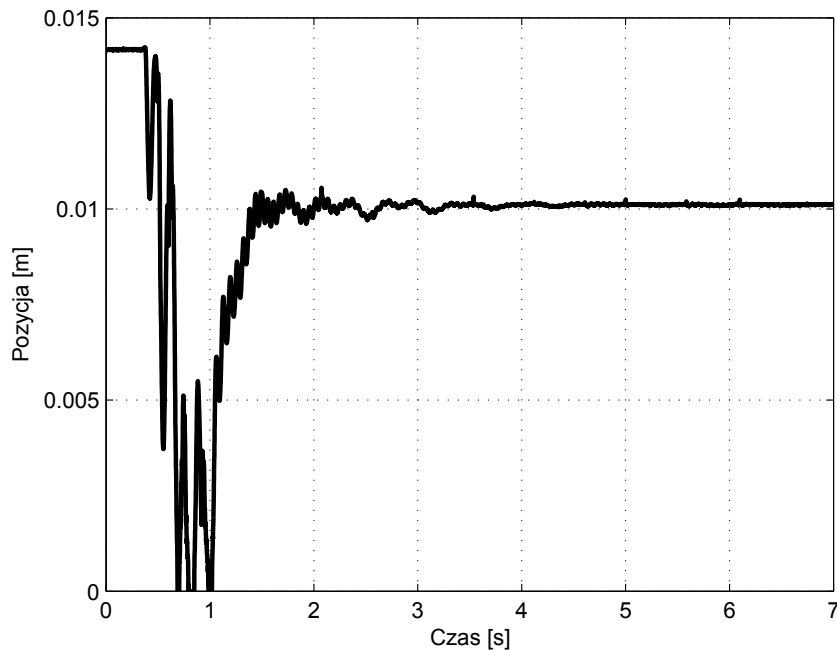
Rysunek 6.15: Realizacja układu regulacji niecałkowitego rzędu dla laboratoryjnego systemu lewitacji magnetycznej w Matlab Simulink

Przeprowadzono eksperymenty mające na celu sprawdzenie poprawność działania zaprojektowanego regulatora. Czasowa aproksymacja Oustaloup została zaimplementowana w środowisku czasu rzeczywistego przy użyciu karty pomiarowej RT-DAC i biblioteki MATLAB RT-CON. Na rysunku 6.16 można zaobserwować proces osiągania pozycji w zadanym punkcie 10^{-3} m przy starcie z bazy urządzenia. Jak widać $PI^\lambda D$ stabilizuje pozycję systemu w czasie 1,5 sekundę co stanowi dobry wynik.

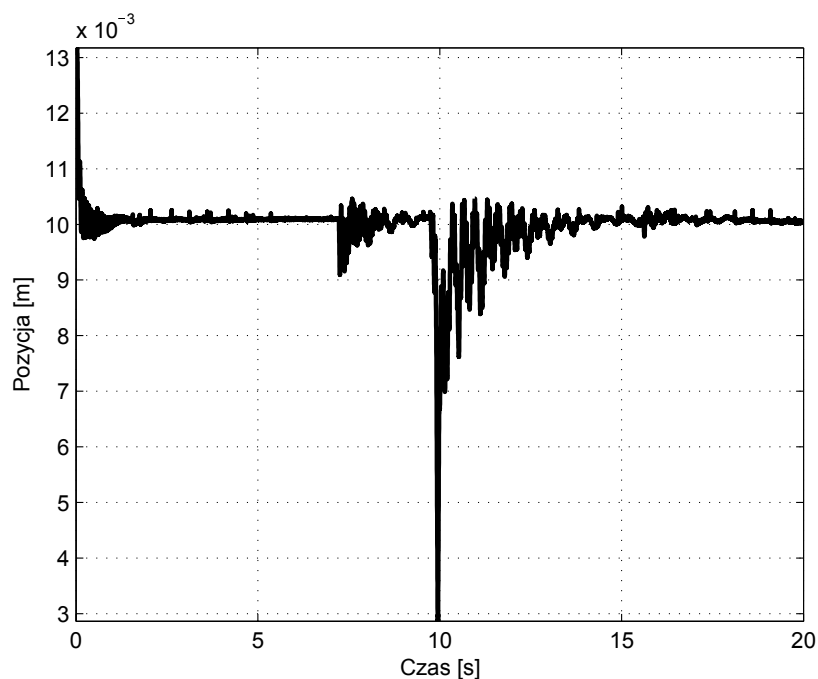


Rysunek 6.16: Rzeczywista pozycja obiektu w systemie lewitacji magnetycznej z $PI^\lambda D$ - rozpoczęcie z bazy

Wykres przedstawiony na rysunku 6.17 obrazuje sytuację, gdy sfera została gwałtownie wrzuconego pod elektromagnes. W tym przypadku regulator ustabilizował system, ale czas tego procesu był dłuższy niż w pierwszym przypadku, zobacz rysunek 6.16.



Rysunek 6.17: Rzeczywista pozycja obiektu wrzuconego pod elektromagnes



Rysunek 6.18: Rzeczywista pozycja obiektu w czasie zakłóceń

Wyniki ostatniego eksperymentów przedstawiono na wykresie 6.18 pokazuje jak opracowany regulator radzi sobie ze znacznymi zakłóceniami, gdy system został zakłócony około 7 i 10 sekundy.

Przeprowadzone eksperymenty dowiodły, że opracowany regulatora jest bardzo odporny



Rysunek 6.19: Pozycja sfery dla której projektowany był regulator $PI^{\lambda}D$

na zakłócenia. Dodatkowo można to zaobserwować poprzez zmianę obiektu przedstawionego na zdjęciach 6.19, 6.20 i 6.21.

6.2.4. Redukcja całkowitego rzędu aproksymacji s^{γ}

Ze względu na ograniczone zasoby sprzętowe platform na których istnieje możliwość zaimplementowania regulatorów niecałkowitego rzędu oraz przede wszystkim ze względu na błędy numeryczne które pojawiają się przy dyskretyzacji wyższych rzędów aproksymacji, postanowiono sprawdzić czy metody redukcji rzędów układów pozwolą na rozwiązanie tych problemów.

Aproksymacja układów niecałkowitego rzędu jest zdefiniowana dla określonego zakresu



Rysunek 6.20: Pozycja sfery dla której nie był projektowany regulator $PI^\lambda D$

częstotliwości $[\omega_{min}, \omega_{max}]$ i rzędu aproksymacji N . Cechą pożądaną jest to by zakres częstotliwości był jak najszerszy, ponieważ aproksymacja zachowuje się najlepiej we wnętrzu interwału. Dodatkowo chcemy by rząd był jak najwyższy gdyż gwarantuje to lepsze przybliżenie projektowanego członu. W praktyce wartość ω_{min} , ω_{max} i N jest określana metodą eksperymentalną, co czasami może być nieefektywne. Istnieje również możliwość oszacowania parametrów zaprezentowana przez Merrikh-Bayat (2012).

Zadanie redukcji modelu liniowego można przedstawić w następujący sposób, dla stabilnego modelu rzędu N zredukowany model rzędu k jest określony przez równanie (6.19), gdzie $k < N$, dla normy błędu redukcji danej jako $\|y(t) - y_r(t)\|$ osiąga minimum.

$$\begin{aligned} \dot{x}_r(t) &= A_r x_r(t) + B_r u(t) \\ y_r(t) &= C_r x_r(t) + D_r u(t) \end{aligned} \tag{6.19}$$

Rysunek 6.21: Nakrętki lewitujące w systemie z regulatorem $PI^{\lambda}D$

Istnieje kilka technik kompleksowej redukcji modelu. Wśród nich dużą popularność uzyskała metoda BTA wprowadzona w pracach (Moore, 1981). Metoda ta opiera się na koncepcji realizacji modelu zrównoważonego, który łatwo można określić dominującą częścią modelu i redukcji poprzez "cięcie" macierzy opisującej dynamikę modelu w przestrzeni stanu.

Aby zrównoważyć model, konieczne jest znalezienie takiej macierzy transformacji T , że Macierze Grama modelu zrównoważonego są identycznymi macierzami diagonalnymi, z malejącymi wartościami osobliwymi Hankela (σ_i) na głównej diagonalu (Antoulas, 2005; Moore, 1981).

$$TPT^T = (T^T)^{-1}QT^{-1} = \text{diag}(\sigma_i) \quad (6.20)$$

gdzie: P and Q są macierzami Grama sterowalnymi i obserwowalnymi dla zredukowanego modelu i można je wyznaczyć przy pomocy równań Lyapunova:

$$\begin{aligned} AP + PA^T + BB^T &= 0 \\ A^TQ + QA + C^TC &= 0 \end{aligned} \quad (6.21)$$

W modelu redukcji zrównoważonej można łatwo określić części dominujące (rzędu k) i słabe (rzędu $N - k$) w następujący sposób :

$$\begin{aligned}\dot{x}(t) &= TAT^{-1}x(t) + TBu(t) \\ &= \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} x(t) + \begin{bmatrix} \bar{B}_1 \\ \bar{B}_2 \end{bmatrix} u(t) \\ y(t) &= CT^{-1}x(t) + Du(t) = [\bar{C}_1 \bar{C}_2] x(t) + Du(t)\end{aligned}\quad (6.22)$$

Zredukowany model uzyskuje się przez wyeliminowanie "słabych" podsystemów z modelu zrównoważonego za pomocą zależności:

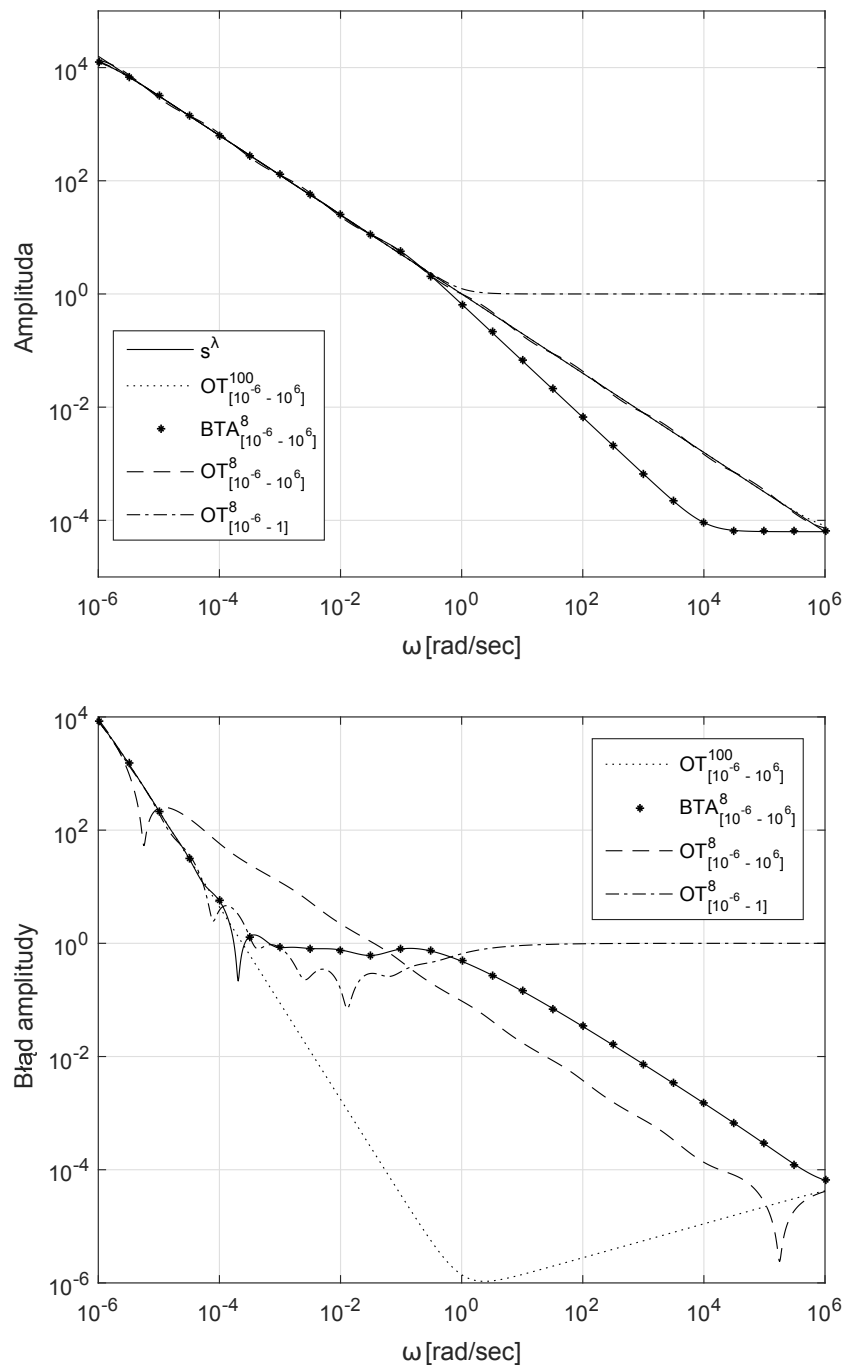
$$A_r = \bar{A}_{11} \quad B_r = \bar{B}_1 \quad C_r = \bar{C}_1 \quad D_r = D \quad (6.23)$$

Zbilansowana metoda skracania umożliwia również "a priori" zdefiniowanie normy błędu aproksymacji. Norma \mathcal{H}_∞ różnicy między modelem pełnego rzędu a modelem o zredukowanym rzędzie jest ograniczona dwukrotnością sumy pominiętych pojedynczych wartości własnych Hankela Antoulas (zob. 2005) i ta formuła matematyczna może być przydatna aby określić rząd zredukowanego modelu.

$$\|\mathbf{G}(j\omega) - \mathbf{G}_r(j\omega)\|_{\mathcal{H}_\infty} \leq 2 \sum_{i=k+1}^N \sigma_i \quad (6.24)$$

Transformacja macierzy Grama nie jest operacją jednoznaczną, dlatego też istnieje wiele algorytmów wyznaczenia macierzy transformacji T i jej odwrotności T^{-1} , jak użyty w tej pracy algorytm BFSR (Antoulas, 2005; Varga, 1991; Rydel et al., 2016a).

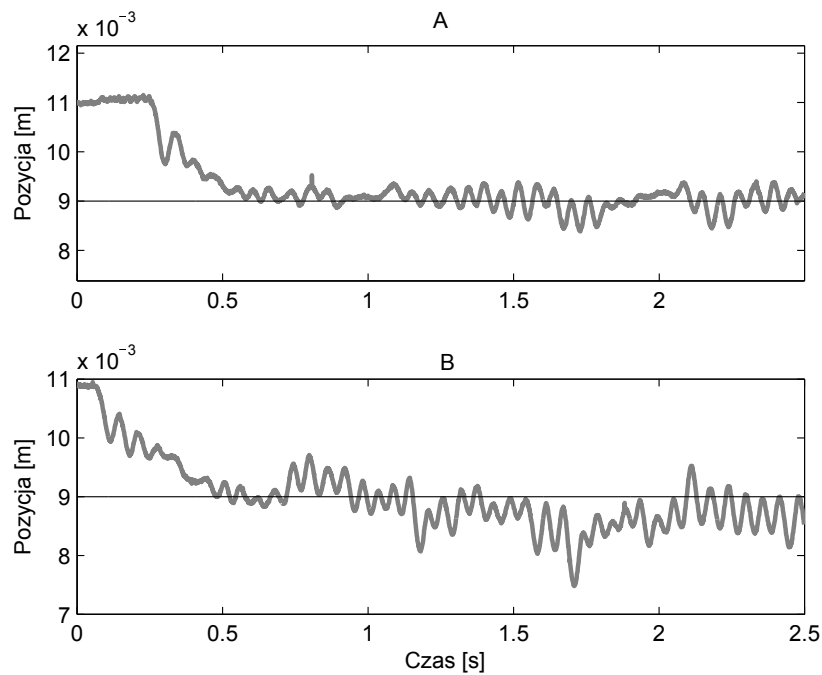
Przeprowadzono porównanie aproksymacji czasowej Oustaloup rzędu $N = 8$ i zredukowanych aproksymacji tego samego typu z rzędu $N = 100$ do $k = 8$ dla zakresu częstotliwości $\omega \in [10^{-6}, 10^6]$. Otrzymane wyniki w dziedzinie częstotliwości są przedstawione na wykresie 6.22. Jak widać, charakterystyki otrzymanych wyników są podobne i dość dobrze przybliżają integrator niecałkowitego rzędu. Jednak lepszą aproksymację przy niskich częstotliwościach otrzymuje się dla modelu uzyskanego w wyniku redukcji metody czasowej Oustalupa wyższego rzędu. Wynika to z faktu, że dominująca część zrównoważonego modelu jest związana z największymi wartościami własnymi odpowiedzi częstotliwościowej modelu. Podobny efekt można uzyskać zmieniając zakres częstotliwości aproksymacji Oustaloup na $\omega \in [10^{-6}, 1]$, ale uzyskany model daje znacznie większy błąd aproksymacji dla częstotliwości powyżej górnego limitu.



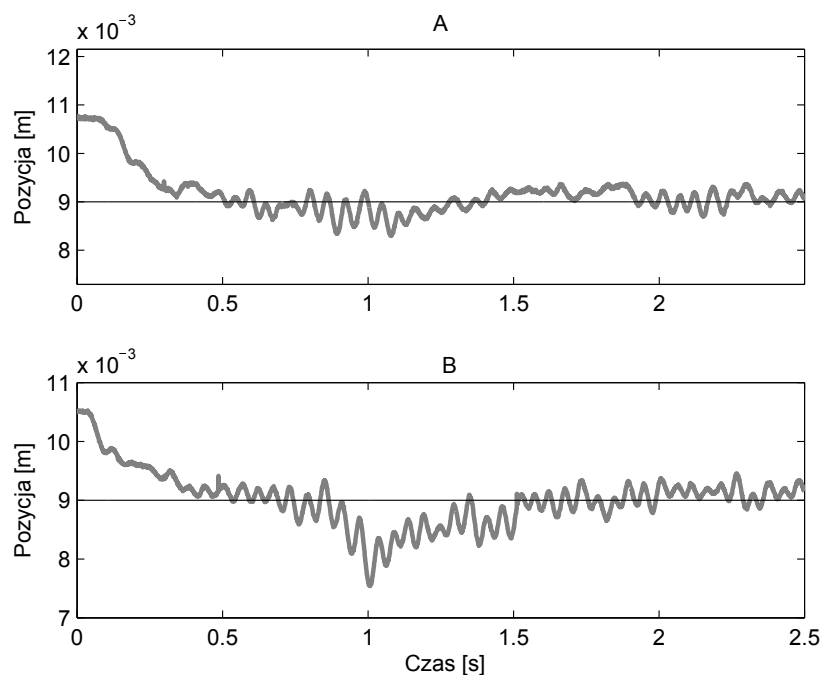
Rysunek 6.22: Porównanie charakterystyk częstotliwościowych i błędów aproksymacji dla aproksymacji członu całkującego niecałkowitego rzędu

6.2.5. Eksperymenty na systemie laboratoryjnym

Eksperymenty dotyczące określenia jakości sterowania regulatora typu $PI^\lambda D$ z integratorem niecałkowitego rzędu po redukcji zostały przeprowadzone dla tych samych wartości parametrów co w poprzedniej sekcji. Różnica polega jedynie na tym, że człon całkujący został zaimplementowany na podstawie modelu zredukowanego z rzędu $N = 100$ do $k = 8$. Został on porównany z regulatorem zaimplementowanym we wcześniejszej sekcji rozprawy.



Rysunek 6.23: Pozycja sfery dla której projektowany był regulator $PI^\lambda D$ A) Czasowy Oustaloup B) BTA



Rysunek 6.24: Pozycja sfery dla której projektowany był regulator $PI^\lambda D$ A) Czasowy Oustaloup B) BTA

Wykres 6.23 obrazuje zadanie pozycjonowania sfery na pozycji 9×10^{-3} m. Można zauważyć, że regulatory $PI^\lambda D$ radzą sobie z zadaniem ustabilizowania sfery na pozycji w podobnym czasie. Jednak po analizie sygnałów sterowania oraz błędu pozycji w punkcie docel-

	Błąd pozycji	
	Nominalna Sfera	Nienominalna sfera
$OT_{[10^{-6}, 10^6]}^8$	4.4739	4.2562
$BTA_{[10^{-6}, 10^6]}^8$	4.3656	4.8083

Tablica 6.4: Porównanie błędu pozycji dla różnych metod realizacji członu całkującego

	Moc sygnału sterującego	
	Nominalna Sfera	Nienominalna sfera
$OT_{[10^{-6}, 10^6]}^8$	0.0840	0.0513
$BTA_{[10^{-6}, 10^6]}^8$	0.0677	0.0605

Tablica 6.5: Porównanie mocy sygnału sterującego dla różnych metod realizacji członu całkującego

wym regulator z zredukowanym członem całkującym niecałkowitego rzędu metodą BTA ma mniejszy sumaryczny błąd pozycji oraz mniejszy koszt sterowania tabela 6.4 i tabela 6.5.

Eksperyment jeszcze raz potwierdził że zmiana obiektu nie wpływa znacząco na jakość regulacji, co oznacza, że regulator posiada dobrą odporność na zakłócenia, wykres 6.24. Jednak po analizie wykresów i tabel 6.4 i 6.5 można zauważyć, że zmiana obiektu lewitującego wpływa bardziej na regulator opracowany z użyciem metody redukcji BTA.

Temat ten został również omówiony przez autora w następujących publikacjach: Bauer (2016b); Bauer i Rydel (2016); Bauer et al. (2014, 2015b).

7. Podsumowanie

Rozprawa doktorska stanowi efekt pracy badawczej autora nad implementacją członów dynamicznych niecałkowitego rzędu na układach wbudowanych. Na podstawie przeprowadzonych badań udowodniono tezę postawioną na początku rozprawy. Jako jeden z pierwszych wniosków który można sformułować to to że ze względu na własności dynamiczne, pozwalające na elastyczniejsze formowanie charakterystyk częstotliwościowych, niż ma to miejsce w systemach klasycznych układy z członami niecałkowitymi będą obiektem dalszych badań w celu wykorzystania ich w zastosowaniach praktycznych. Opracowane przez autora metody i przeprowadzone eksperymenty dowodzą, że możliwe jest stworzenie stabilnych numerycznie i odpornych na zakłócenia metod realizacji dynamicznych układów niecałkowitego rzędu na układy wbudowane. Jak pokazano systemy te nadają się zarówno jako kontrolery jak i filtry. Należy zaznaczyć jednak, że przedstawione metody ze względu na swoją strukturę są trudne w implementacji przy bardziej złożonych strukturach członów wykonawczych.

Niektóre rezultaty dotyczące bezpośrednio tematyki rozprawy jak i tematów pobocznych były przedstawiane przez autora na konferencjach naukowych bądź też publikowane w czasopismach naukowych, zobacz: Bauer (2016b); Bauer i Rydel (2016); Kania et al. (2014); Bauer i Kawala-Janik (2017); Dziwiński et al. (2015); Baranowski et al. (2015b); Zagórska et al. (2015); Baranowski et al. (2016e); Zagórska et al. (2014b, 2015); Bauer et al. (2015a); Baranowski et al. (2014f,b); Zagórska et al. (2014c); Dziwiński et al. (2014a); Piątek et al. (2014b); Kawala-J.ik et al. (2014); Baranowski et al. (2014g, 2016b); Bauer et al. (2013); Zagórska et al. (2014d); Piątek et al. (2014a); Dziwiński et al. (2014b); Baranowski et al. (2014c); Bauer et al. (2014); Zagórska et al. (2014a); Baranowski et al. (2016c, 2015c); Bauer (2015); Zagórska et al. (2015); Baranowski et al. (2015a, 2016d,a, 2014a); Bauer (2016a).

Według autora jako najważniejsze oryginalne osiągnięcia w pracy należy zaliczyć:

1. Opracowanie dwóch metod aproksymacji elementu s^α opartej na metodzie Oustaloupa:
 - 1.1. Równoległej metody Oustaloupa, str. 41 ,
 - 1.2. Czasowej metody Oustaloupa - wspólnie z mgr inż. Martą Zagórską i dr. hab.inż. Jerzy Baranowskim, str. 42.

2. Podanie stabilnych numerycznie metod dyskretyzacji dla opracowanych metody dyskretyzacji - str. 29.
3. Implementacja w Matlabie metod do symulacji i wyliczania postaci aproksymacji testowanych układów ciągłych i dyskretnych.
4. Implementacja i przeprowadzenie testów odporności numerycznej członów dynamicznych niecałkowitego rzędu na Arduino Uno str. 64 .
5. Implementacja i przeprowadzenie testów odporności numerycznej członów dynamicznych niecałkowitego rzędu na mikroprocesorze STM32, str. 69.
6. Opracowanie i implementacja regulatora ułamkowego PID dla laboratoryjnego systemu nagrzewnicy powietrznej z opóźnieniem w tym:
 - 6.1. przeprowadzenie procesu strojenia regulatora przy użyciu metody symulowanego wyżarzania, str. 83,
 - 6.2. implementacja regulatora z użyciem aproksymacji równoległej Oustaloupa i klasycznej metody Oustaloupa, str. 85,
 - 6.3. porównanie otrzymanych wyników regulacji w eksperymentach na rzeczywistym obiekcie laboratoryjnym, str. 87.
7. Opracowanie i implementacja regulatora ułamkowego typu $PI^\lambda D$ dla laboratoryjnego systemu lewitacji magnetycznej w tym:
 - 7.1. opracowanie i opisanie procesu strojenia regulatora przy użyciu linearyzacji w punkcie pracy, str. 90,
 - 7.2. implementacja regulatora z użyciem czasowej aproksymacji Oustaloupa, str. 92,
 - 7.3. testy zaimplementowanego rozwiązania na rzeczywistym obiekcie laboratoryjnym, str. 93,
 - 7.4. porównanie we współpracy z dr inż. Markiem Rydlem implementacji czasowej aproksymacji Oustaloupa w dwóch wariantach: niskiego rzędu oraz zredukowanej z wysokiego rzędu aproksymacji. Wyniki potwierdzono na laboratoryjnym systemie lewitacji magnetycznej, str. 95.

Przygotowana rozprawa na pewno nie wyczerpuje tematów związanych z implementacją elementów niecałkowitego rzędu na układach wbudowanych. Przede wszystkim ze względu na wybraną tematykę pracy nie podjęto rozważań dotyczących oceny jakości zachowania się dynamiki tej klasy układów. Ze względu na to autor widzi kilka otwartych problemów badawczych dotyczących układów dynamicznych niecałkowitego rzędu. Pierwszym z nich jest opracowanie metod określania stabilności układów czasowych z członami niecałkowitego

rzędu po zamknięciu sprzężenia zwrotnego (metoda analogiczna do Kryterium Nyquista), umożliwiło by to określanie zapasów wzmocnień i fazy dla opracowywanych rozwiązań. Rozwój takich metod umożliwiłby ustalenie reguł strojenia regulatorów opartych o układy niecałkowitego rzędu w szczególności doboru zbioru ograniczeń.

W dalszym ciągu otwartym tematem badawczym jest również opracowanie metod aproksymacji dla złożonych układów niecałkowitego rzędu oraz metod implementacji członów niecałkowitego rzędu o mniejszej złożoności obliczeniowej. Dodatkowo należy stworzyć narzędzia do szybkiego prototypownia układów niecałkowitego rzędu na platformy symulacyjne jak i sprzętowe.

Spis tablic

5.1	Wymagania niefunkcjonalne dla systemów czasu rzeczywistego	57
5.2	Przykładowe systemy czasu rzeczywistego z miękkimi ograniczeniami czasowymi	60
5.3	Wymagania niefunkcjonalne dla systemów czasu rzeczywistego	61
5.4	Wymagania niefunkcjonalne dla systemów czasu rzeczywistego	61
5.5	Porównanie błędów regulatora $PI^\lambda D^\mu$	70
5.6	Analiza statystyczna przeprowadzonych eksperymentów	74
6.1	Uzyskane marginesy stabilności	83
6.2	Optymalne parametry kontrolera	85
6.3	Porównanie zoptymalizowanego PID i zoptymalizowanego $PI^\alpha D^\mu$	86
6.4	Porównanie błędu pozycji dla różnych metod realizacji członu całkującego .	101
6.5	Porównanie mocy sygnału sterującego dla różnych metod realizacji członu całkującego	101

Spis rysunków

3.1	Porównanie rozwiązania dokładnego z metodą z krótką pamięcią	33
3.2	Rozkład zer i biegunów systemów dyskretnych dla aproksymacji czasowego Oustaloupa rzędu 15	40
3.3	Schemat równoległej aproksymacji Oustaloup	41
3.4	Rozmieszczenie zer i biegunów w kole jednostkowym dla $s^{0.5}$ przy aproksymacji równoległej Oustaloupa	42
3.5	Rozkład zer i biegunów systemów dyskretnych dla aproksymacji czasowego Oustaloupa rzędu 15	45
4.1	Porównanie członu całkującego dla różnych wartości stopnia pochodnej ułamkowej	47
4.2	Porównanie członu różniczkującego dla różnych wartości stopnia pochodnej ułamkowej	47
4.3	Porównanie charakterystyk filtra dolnoprzepustowego dla różnych wartości stopnia pochodnej ułamkowej	48
4.4	Porównanie charakterystyk filtra górnoprzepustowego dla różnych wartości stopnia pochodnej ułamkowej	49
4.5	Porównanie charakterystyk filtra ułamkowego drugiego rzędu dla różnych wartości stopnia pochodnej ułamkowej	50
4.6	Porównanie charakterystyk filtra ułamkowego pasmowo przepustowego dla różnych wartości stopnia pochodnej α i $\beta = 0.1$	51
5.1	Schemat pinów ATmega328, źródło: https://www.arduino.cc/en/Hacking/PinMapping168	53
5.2	STM32F0-DISCOVERY Board	54
5.3	Schemat rozmieszczenia elementów na STM32F0-DISCOVERY Board (ST, 2019b)	55
5.4	Uproszczony schemat architektury systemów wbudowanych	58
5.5	Arduino Uno/PC connection schema.	64
5.6	Porównanie wyników filtracji na platformie Arduino Uno z Matlab Simulink dla $\alpha = 0.1$	65

5.7	Porównanie gęstość mocy sygnału po filtracji przez Arduino Uno i Matlab Simulink dla $\alpha = 0.1$	66
5.8	Porównanie wyników filtracji na platformie Arduino Uno z Matlab Simulink dla $\alpha = 0.7$	66
5.9	Porównanie gęstość mocy sygnału po filtracji przez Arduino Uno i Matlab Simulink dla $\alpha = 0.1$ $\alpha = 0.7$	67
5.10	Odpowiedź skokowa regulatora $PI^\lambda D^\mu$ w Matlab Simulink i ATmega328P .	68
5.11	Odpowiedź regulatora $PI^\lambda D^\mu$ w Matlab Simulink i ATmega328P na sygnał sinusoidalny	68
5.12	Sygnał referencyjny	69
5.13	Odpowiedź regulatora $PI^\lambda D^\mu$ w Matlab Simulink i ATmega328P na sygnał referencyjny	69
5.14	Konfiguracja systemu pomiarowego	71
5.15	Sinusoidalny sygnał testowy	71
5.16	Trójkątny sygnał testowy	72
5.17	Prostokątny sygnał testowy	72
5.18	Analiza błędu częstotliwości próbkowania 15 Hz	75
5.19	Porównanie działania aproksymacji w Matlab Simulink i STM dla 15 Hz i rzędzie aproksymacji 6	76
5.20	Porównanie działania aproksymacji w Matlab Simulink i STM dla 15 Hz i rzędzie aproksymacji 15	77
5.21	Analiza błędu częstotliwości próbkowania 150 Hz	77
5.22	Analiza błędu częstotliwości próbkowania 300 Hz	78
5.23	Porównanie działania aproksymacji w Matlab Simulink i STM dla 300 Hz i rzędzie aproksymacji 6	78
5.24	Porównanie działania aproksymacji w Matlab Simulink i STM dla 300 Hz i rzędzie aproksymacji 14	79
6.1	Schemat układu: 1 – termo rezystor, 2 – element grzewczy, 3 – wiatrak . . .	81
6.2	Nagrzewnica powietrzna PT326	82
6.3	Odpowiedź częstotliwościowa dla rozpatrywanego systemu	83
6.4	Układ sterowania z regulatorem PID ułamkowego rzędu dla nagrzewnicy powietrznej	83
6.5	Postęp optymalizacji	84
6.6	Charakterystyka Nyquista z zaznaczonym marginesem stabilności	84
6.7	Realizacja układu regulacji niecałkowitego rzędu dla laboratoryjnego systemu nagrzewnicy powietrza w Matlab Simulink	85

6.8	Porównanie odpowiedzi skokowych	86
6.9	Porównanie odpowiedzi częstotliwościowych	86
6.10	Porównanie wydajności kontrolerów dla stałej wartości wymuszenia	87
6.11	Porównanie na szereg odpowiedzi skokowych	88
6.12	Porównanie dla zadania nadążania	88
6.13	System lewitacji magnetycznej z regulatorem ułamkowym typu $PI^{\lambda}D$	91
6.14	Wykres Nyquista dla rzeczywistej i aproksymowanej postaci $PI^{\lambda}D$ controller	92
6.15	Realizacja układu regulacji niecałkowitego rzędu dla laboratoryjnego systemu lewitacji magnetycznej w Matlab Simulink	93
6.16	Rzeczywista pozycja obiektu w systemie lewitacji magnetycznej z $PI^{\lambda}D$ - rozpoczęcie z bazy	93
6.17	Rzeczywista pozycja obiektu wrzuconego pod elektromagnes	94
6.18	Rzeczywista pozycja obiektu w czasie zakłóceń	94
6.19	Pozycja sfery dla której projektowany był regulator $PI^{\lambda}D$	95
6.20	Pozycja sfery dla której nie był projektowany regulator $PI^{\lambda}D$	96
6.21	Nakrętki lewitujące w systemie z regulatorem $PI^{\lambda}D$	97
6.22	Porównanie charakterystyk częstotliwościowych i błędów aproksymacji dla aproksymacji członu całkującego niecałkowitego rzędu	99
6.23	Pozycja sfery dla której projektowany był regulator $PI^{\lambda}D$ A) Czasowy Oustaloup B) BTA	100
6.24	Pozycja sfery dla której projektowany był regulator $PI^{\lambda}D$ A) Czasowy Oustaloup B) BTA	100

Bibliografia

- A. M. AbdelAty, A. G. Radwan, A. Elwakil, i C. Psychalinos. A fractional-order dynamic PV model. In *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2016a.
- A. M. AbdelAty, A. Soltan, W. A. A., i A. G. Radwan. Hermite polynomials in the fractional order domain suitable for special filters design. In *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 1–4, June 2016b.
- K. Adolfsson, M. Enelund, i P. Olsson. On the fractional order model of viscoelasticity. *Mechanics of Time-Dependent Materials*, 9(1):15–34, 2005.
- E. Ahmed i H. A. El-Saka. On fractional order models for hepatitis c. *Nonlinear Biomedical Physics*, 4(1), mar 2010. doi: 10.1186/1753-4631-4-1.
- E. Ahmed i A. Elgazzar. On fractional order differential equations model for nonlocal epidemics. *Physica A: Statistical Mechanics and its Applications*, 379(2):607–614, 2007.
- A. Antoulas. *Approximation of Large-Scale Dynamical System*. Society for Industrial and Applied Mathematics, Philadelphia, 2005.
- A. Arafa, S. Rida, i M. KhA.I. A fractional-order model of HIV infection with drug therapy effect. *Journal of the Egyptian Mathematical Society*, 22(3):538–543, 2014.
- K. Åström i T. Hägglund. *Advanced Pid Control*. ISA-The Instrumentation, Systems, and Automation Society, 2006.
- K. J. Åström. Model uncertainty and robust control. *Lecture Notes on Iterative Identification and Control Design*, pages 63–100, 2000.
- K. J. Åström i R. M. Murray. *Feedback Systems*. Princeton University Press, 2008.
- A. Atangana. On the new fractional derivative and application to nonlinear fisher’s reaction–diffusion equation. *Applied Mathematics and Computation*, 273:948–956, 2016.

- B. Wang, S. E. Li, H. Peng, i Z. Liu. Fractional-order modeling and parameter identification for lithium-ion batteries. *Journal of Power Sources*, 293:151–161, 2015.
- M. A. Balci. Time fractional capital-induced labor migration model. *Physica A: Statistical Mechanics and its Applications*, 477:91–98, 2017.
- P. Bania i J. Baranowski. Laguerre polynomial approximation of fractional order linear systems. In W. Mitkowski, J. Kacprzyk, and J. Baranowski, editors, *Advances in the Theory and Applications of Non-integer Order Systems: 5th Conference on Non-integer Order Calculus and Its Applications, Cracow, Poland*, pages 171–182. Springer, 2013.
- J. Baranowski i P. Piątek. Nonlinear dynamical feedback for motion control of magnetic levitation system. In *Proceedings of 13th International Power Electronics and Motion Control Conference EPE-PEMC*, pages 1469–1476, Poznań, 1–3 September 2008.
- J. Baranowski, P. Piątek, W. Bauer, T. Dziwiński, i M. Zagórska. Bi-fractional filters, part 2: Right half-plane case. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference on*, pages 369–373. IEEE, 2014a.
- J. Baranowski, P. Piątek, W. Bauer, T. Dziwiński, i M. Zagórska. Bi-fractional filters, part 2: Right half-plane case. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference on*, pages 369–373, 2014b.
- J. Baranowski, P. Piątek, A. Kawala-Jik, M. Zagórska, W. Bauer, i T. Dziwiński. Non-integer order filtration of electromyographic signals. In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Order Calculus and its Applications*. Springer, 2014c.
- J. Baranowski, M. Zagórska, P. Bania, W. Bauer, T. Dziwiński, i P. Piątek. Impulse response approximation method for bi-fractional filter. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference*, pages 379–383. IEEE, 2014d.
- J. Baranowski, M. Zagórska, P. Bania, W. Bauer, T. Dziwiński, i P. Piątek. Impulse response approximation method for bi-fractional filter. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference*, 2014e.
- J. Baranowski, M. Zagórska, P. Bania, W. Bauer, T. Dziwiński, i P. Piątek. Impulse response approximation method for bi-fractional filter. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference On*, 2014f.
- J. Baranowski, M. Zagórska, W. Bauer, T. Dziwiński, i P. Piątek. Remarks on impulse response of non-integer order systems. In *Materiały XXXVII Międzynarodowej konferencji*

- z podstaw elektrotechniki i teorii obwodów IC-SPETO, pages 50–54, Ustroń, 21–24. 05. 2014g.
- J. Baranowski, W. Bauer, M. Zagórowska, T. Dziwiński, i P. Piątek. Time-domain oustaloup approximation. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference On*, pages 116–120. IEEE, 2015a.
- J. Baranowski, W. Bauer, M. Zagórowska, T. Dziwiński, i P. Piątek. Time-domain oustaloup approximation. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference On*, 2015b.
- J. Baranowski, M. Zagórowska, W. Bauer, T. Dziwiński, i P. Piątek. Applications of direct lyapunov method in caputo non-integer order systems. *Elektronika ir Elektrotechnika*, 21 (2):10–13, 2015c.
- J. Baranowski, W. Bauer, i M. Zagórowska. Stability properties of discrete time-domain oustaloup approximation. In S. Domek and P. Dworak, editors, *Theoretical Developments and Applications of Non-Integer Order Systems*, volume 357 of *Lecture Notes in Electrical Engineering*, pages 93–103. Springer International Publishing, 2016a.
- J. Baranowski, W. Bauer, i M. Zagórowska. Stability properties of discrete time-domain oustaloup approximation. In S. Domek and P. Dworak, editors, *Theoretical Developments and Applications of Non-Integer Order Systems*, volume 357 of *Lecture Notes in Electrical Engineering*, pages 93–103. Springer International Publishing, 2016b.
- J. Baranowski, W. Bauer, M. Zagórowska, A. Kawala-J.ik, T. Dziwiński, i P. Piątek. Adaptive non-integer controller for water tank system. In S. Domek and P. Dworak, editors, *Theoretical Developments and Applications of Non-Integer Order Systems*, volume 357 of *Lecture Notes in Electrical Engineering*, pages 271–280. Springer International Publishing, 2016c.
- J. Baranowski, W. Bauer, M. Zagórowska, i P. Piątek. On digital rea.zations of non-integer order filters. *Circuits Syst Signal Process*, 2016d.
- J. Baranowski, W. Bauer, M. Zagórowska, i P. Piątek. On digital rea.zations of non-integer order filters. *Circuits Syst Signal Process*, 2016e.
- R. S. Barbosa, J. Machado, and I. M. Ferreira. Tuning of PID controllers based on Bode's ideal transfer function. *Nonlinear Dynamics*, 38(1-4):305–321, 2004.
- W. Bauer. Implementation of non-integer order controller using oustaloup parallel approximation for air heating process trainer. In *Theoretical Developments and Applications of Non-Integer Order Systems: 7th Conference on Non-Integer Order Calculus and Its Applications, Szczecin, Poland*, volume 357, page 239. Springer, 2015.

- W. Bauer. Implementation of non-integer $pi^\lambda d^\mu$ controller for the atmega328p microcontroller. In *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 118–121, 2016a.
- W. Bauer. Implemetation of Non-integer $PI^\lambda D^\mu$ controller for the ATmega328P Microcontroller. In *Methods and Models in Automation and Robotics (MMAR), 2016 21st International Conference On*, pages 118–121, 29.08-1.09 2016b.
- W. Bauer i A. Kawala-Janik. Implementation of bi-fractional filtering on the arduino uno hardware platform. *Lecture Notes in Electrical Engineering*, 407:419–428, 2017.
- W. Bauer i M. Rydel. Application of reduced models of non-integer order integrator to the reazation $PI^\lambda D$ controller. In *2016 39th International Conference on Telecommunications and Signal Processing*, pages 611–614, 27-29 06 2016.
- W. Bauer, J. Baranowski, i W. Mitkowski. Non-integer order $PI^\alpha D^\mu$ control ICU-MM. In W. Mitkowski, J. Kacprzyk, and J. Baranowski, editors, *Advances in the Theory and Applications of Non-integer Order Systems: 5th Conference on Non-integer Order Calculus and Its Applications, Cracow, Poland*, pages 295–303. Springer, 2013.
- W. Bauer, T. Dziwiński, J. Baranowski, P. Piątek, i M. Zagórska. Comparison of performance indices for tuning of $pi^\lambda d^\mu$ controller for magnetic levitation system. In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Orfer Caculus and its Applications*. Springer, 2014.
- W. Bauer, J. Baranowski, T. Dziwiński, P. Piątek, i M. Zagórska. Oustalup parallel approximation. In *Materiały XXXVIII Międzynarodowej konferencji z podstaw elektrotechniki i teorii obwodów IC-SPETO*, pages 51–52, 2015a.
- W. Bauer, J. Baranowski, T. Dziwiński, P. Piątek, i M. Zagórska. Stabilisation of magnetic levitation with a $pi^\lambda d^\mu$ controller. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference*, pages 638–642. IEEE, 2015b.
- K. Biswas, G. Bohannan, R. Caponetto, A. M. Lopes, and J. A. T. Machado. Fractional-order models of vegetable tissues. In *Fractional-Order Devices*, pages 73–92. Springer International Publishing, 2017.
- S. Blasiak. Time-fractional heat transfer equations in modeling of the non-contacting face seals. *International Journal of Heat and Mass Transfer*, 100:79–88, sep 2016.
- H. W. Bode. *Network analysis and feedback amplifier design*. 1945.
- G. C. Buttazzo. *Hard Real-Time Computing Systems*. Springer US, 2011.

- J. Cao i B. Cao. Design of fractional order controllers based on particle swarm optimization. In *2006 1st IEEE Conference on Industrial Electronics and Applications*. IEEE, 2006.
- R. Caponetto, S. Graziani, F. Sapuppo, i V. Tello. An enhanced fractional order model of ionic polymer-metal composites actuator. *Advances in Mathematical Physics*, 2013:1–6, 2013.
- M. Caputo. Linear models of dissipation whose q is almost frequency independent–II. *Geophysical Journal International*, 13(5):529–539, nov 1967. doi: 10.1111/j.1365-246x.1967.tb02303.x.
- G. Carlson. Simulation of the fractional derivative operator \sqrt{s} and the fractional integral operator $1/\sqrt{s}$, 1960.
- L. Chang i H. Chen. Tuning of fractional pid controllers using adaptive genetic algorithm for active magnetic bearing system. *WSEAS Trans. Syst*, 8(1):158–167, 2009.
- A. Charef, H. H. Sun, Y. Y. Tsao, i B. Onaral. Fractal system as represented by singularity function. *IEEE Transactions on Automatic Control*, 37(9):1465–1470, 1992.
- Y. Q. Chen i K. Moore. Discretization schemes for fractional-order differentiators and integrators. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 49(3):363–367, 2002.
- R. Cipin, C. Ondrusek, i R. Huzlík. Fractional-order model of DC motor. In *Mechatronics 2013*, pages 363–370. Springer International Publishing, 2014.
- J. L. Coolidge. The story of the binomial theorem. *The American Mathematical Monthly*, 56(3):147, 1949.
- D. Copot, C. M. Ionescu, i R. D. Keyser. Modelling drug interaction using a fractional order pharmacokinetic model. In *ICFDA 14 International Conference on Fractional Differentiation and Its Applications 2014*. IEEE, 2014.
- A. Coronel-Escamilla, J. Gómez-Aguilar, M. López-López, V. Alvarado-Martínez, and G. Guerrero-Ramírez. Triple pendulum model involving fractional derivatives with different kernels. *Chaos, Solitons & Fractals*, 91:248–261, 2016.
- P. J. Davis. Leonhard euler’s integral: A historical profile of the gamma function: In memoriam: Milton abramowitz. *The American Mathematical Monthly*, 66(10):849, 1959.
- K. Diethelm. *The Analysis of Fractional Differential Equations: An Application-Oriented Exposition Using Differential Operators of Caputo Type*. Number no. 2004 in Lecture Notes in Mathematics. Springer, 2010.

- E. W. Dijkstra. Over de sequentialiteit van procesbeschrijvingen. circulated privately, 1962/1963. URL <http://www.cs.utexas.edu/users/EWD/ewd00xx/EWD35.PDF>.
- A. Djouambi, A. Charef, i A. Besançon. Optimal approximation, simulation and analog realization of the fundamental fractional order transfer function. *International Journal of Applied Mathematics and Computer Science*, 17(4):455–462, 2007.
- A. Djouambi, A. Voda, P. Grangeat, i P. Mailley. Fractional order modeling and identification for electrochemical nano-biochip. In *Micro, Nanosystems and Systems on Chips*, pages 197–220. John Wiley & Sons, Inc., 2013.
- M. Długosz i P. Skruch. The application of fractional-order models for thermal process modelling inside buildings. *Journal of Building Physics*, 39(5):440–451, 2015.
- T. Dziwiński, W. Bauer, J. Baranowski, P. Piątek, i M. Zagórowska. Robust non-integer order controller for air heater. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference On*, pages 434–438, 2014a.
- T. Dziwiński, W. Bauer, J. Baranowski, P. Piątek, i M. Zagórowska. Robust non-integer order controller for air heating process trainer. In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Orfer Caculus and its Applications*. Springer, 2014b.
- T. Dziwiński, P. Piątek, J. Baranowski, W. Bauer, i M. Zagórowska. On the practical implementation of non-integer order filters. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference On*, pages 921–924, 2015.
- G. Espinosa-Paredes, J. B. Morales-Sandoval, R. Vázquez-Rodríguez, i E.-G. Espinosa-Martínez. Constitutive laws for the neutron density current. *Annals of Nuclear Energy*, 35(10):1963–1967, 2008.
- G. Espinosa-Paredes, M. Polo-Labarrios, E. Espinosa-Martínez, i E. del Valle-Gallegos. Fractional neutron point kinetics equations for nuclear reactor dynamics. *Annals of Nuclear Energy*, 38(2-3):307–330, 2011.
- T. Faber, A. Jaishankar, i G. McKinley. Describing the firmness, springiness and rubberiness of food gels using fractional calculus. part i: Theoretical framework. *Food Hydrocolloids*, 62:311–324, 2017a.
- T. Faber, A. Jaishankar, i G. McKinley. Describing the firmness, springiness and rubberiness of food gels using fractional calculus. part II: Measurements on semi-hard cheese. *Food Hydrocolloids*, 62:325–339, 2017b.

- H. Fallahgoul, S. Focardi, i F. Fabozzi. *Fractional calculus and fractional processes with applications to financial economics: theory and application*. Academic Press, 2016.
- V. Feliu-Battle i F. J. Castillo-García. On the robust control of stable minimum phase plants with large uncertainty in a time constant. a fractional-order control approach. *Automatica*, 50(1):218–224, 2014.
- O. Foupouapouognigni, C. N. D. Buckjohn, M. S. Siewe, i C. Tchawoua. Nonlinear electromechanical energy harvesters with fractional inductance. *Chaos, Solitons & Fractals*, 103:12–22, 2017.
- W. Glockle i T. Nonnenmacher. A fractional calculus approach to self-similar protein dynamics. *Biophysical Journal*, 68(1):46–53, 1995.
- J. Gómez-Aguilar, H. Yépez-Martínez, R. Escobar-Jiménez, C. Astorga-Zaragoza, i J. Reyes-Reyes. Analytical and numerical solutions of electrical circuits described by fractional derivatives. *Applied Mathematical Modelling*, 40(21-22):9079–9094, 2016.
- G. González-Parra, A. J. Arenas, i B. M. Chen-Charpentier. A fractional order epidemic model for the simulation of outbreaks of influenza a(h1n1). *Mathematical Methods in the Applied Sciences*, 37(15):2218–2226, 2013.
- R. Gorenflo. *Fractional Calculus*, pages 277–290. Springer Vienna, Vienna, 1997.
- A. Goulart, M. Lazo, J. Suarez, i D. Moreira. Fractional derivative models for atmospheric dispersion of pollutants. *Physica A: Statistical Mechanics and its Applications*, 477:9–19, 2017.
- W. Grzesikiewicz, A. Wakulicz, i A. Zbiciak. Non-linear problems of fractional calculus in modeling of mechanical systems. *International Journal of Mechanical Sciences*, 70: 90–98, 2013.
- T. T. Hartley i C. F. Lorenzo. Fractional-order system identification based on continuous order-distributions. *Signal Processing*, 83(11):2287–2300, 2003.
- K. S. Hedrih i V. Nikolić-Stanojević. A model of gear transmission: Fractional order system dynamics. *Mathematical Problems in Engineering*, 2010:1–23, 2010.
- R. Herrmann. *Fractional Calculus*. WORLD SCIENTIFIC, 2011.
- P. Humbert i R. Agarwal. Sur la fonction de mittag-leffler e quelques-unes de ses generations. *Bull. Sci. Math*, (77):180–185, 1953.
- IEEE. *IEEE standard for binary floating-point arithmetic*. Institute of Electrical and Electronics Engineers, New York, 1985. Note: Standard 754–1985.

- C. Ionescu i D. Copot. On the use of fractional order PK-PD models. *Journal of Physics: Conference Series*, 783:12–50, 2017.
- C. Ionescu i J. F. Kelly. Fractional calculus for respiratory mechanics: Power law impedance, viscoelasticity, and tissue heterogeneity. *Chaos, Solitons & Fractals*, 102:433–440, 2017.
- C. M. Ionescu, J. A. T. Machado, i R. D. Keyser. Modeling of the lung impedance using a fractional-order ladder network with constant phase elements. *IEEE Transactions on Biomedical Circuits and Systems*, 5(1):83–89, 2011.
- S. P. Jadhav, R. Adik, R. J. H. Chile, S. G. G. Singhji, S. T. Hamde, i S. G. G. Singhji. Reduced-parameter fractional-order modeling of large dynamical system: Application to gas turbine. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*. IEEE, 2016.
- A. Jalloul, J. Trigeassou, K. Jelassi, i P. Melchior. Fractional order modeling of rotor skin effect in induction machines. *Nonlinear Dynamics*, 73(1-2):801–813, 2013.
- T. Kaczorek. *Selected Problems of Fractional Systems Theory*. Lecture Notes in Control and Information Sciences. Springer, 2011.
- B. Kania, T. Dziwiński, W. Bauer, M. Zagórska, i J. Baranowski. A comparison between integer order and non-integer order controllers applied to a water levelling system. *Automatyka/Automatics*, 18(2):71–82, 2014.
- A. Kawala-Janik, M. Podpora, M. Pelc, J. Baranowski, i P. Piątek. Implementation of an inexpensive eeg headset for the pattern recognition purpose. In *The 7th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, 2013.
- A. Kawala-Janik, M. Podpora, J. Baranowski, W. Bauer, i M. Pelc. Innovative approach in analysis of eeg and emg signals - comparison of the two novel methods. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference on*, pages 804–807. IEEE, 2014.
- F. Khateb, D. Kubanek, G. Tsirimokou, i C. Psychalinos. Fractional-order filters based on low-voltage {DDCCs}. *Microelectronics Journal*, 50:50 – 59, 2016. ISSN 0026-2692.
- S. Kirkpatrick, C. D. Gelatt, i M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.
- H. Kopetz. *Real-Time Systems*. Springer US, 2011.
- K. Lazopoulos i D. Karaoulanis. On fractional modelling of viscoelastic mechanical systems. *Mechanics Research Communications*, 78:1–5, 2016.

- M. Lewandowski i M. Orzyłowski. Fractional-order models: The case study of the supercapacitor capacitance measurement. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 65(4):449–457, 2017.
- C. Li, D. Qian, i Y. Chen. On riemann-liouville and caputo derivatives. *Discrete Dynamics in Nature and Society*, 2011:1–15, 2011.
- P. Lino, G. Maione, i F. Saponaro. Fractional-order modeling of high-pressure fluid-dynamic flows: An automotive application. *IFAC-PapersOnLine*, 48(1):382–387, 2015.
- Y. Luo, T. Zhang, B. Lee, C. Kang, i Y. Chen. Fractional-order proportional derivative controller synthesis and implementation for hard-disk-drive servo system. 2014.
- A. Maachou, R. Malti, P. Melchior, J. Battaglia, A. Oustaloup, i B. Hay. Nonlinear thermal system identification using fractional volterra series. *Control Engineering Practice*, 29: 50–60, 2014.
- J. T. Machado. Fractional order description of DNA. *Applied Mathematical Modelling*, 39 (14):4095–4102, 2015.
- R. L. Magin. Fractional calculus models of complex dynamics in biological tissues. *Computers & Mathematics with Applications*, 59(5):1586–1593, 2010.
- R. L. Magin. Fractional calculus in bioengineering: A tool to model complex dynamics. In *Proceedings of the 13th International Carpathian Control Conference (ICCC)*. IEEE, 2012.
- H. Malek, Y. Luo, i Y. Chen. Identification and tuning fractional order proportional integral controllers for time delayed systems with a fractional pole. *Mechatronics*, 23(7):746–754, 2013.
- J. Martin. *Programming real-time computer systems*. Prentice-Hall series in automatic computation. Prentice-Hall, 1965.
- K. Matsuda i H. Fujii. h_∞ optimized wave-absorbing control: Analytical and experimental results. *Journal of Guidance, Control, and Dynamics*, 16(6):1146–1153, 1993.
- R. V. Mendes. A fractional calculus interpretation of the fractional volatility model. *Nonlinear Dynamics*, 55(4):395–399, 2008.
- L. Meng and D. Xue. Design of an optimal fractional-order pid controller using multi-objective ga optimization. In *Control and Decision Conference, 2009. CCDC'09. Chinese*, pages 3849–3853. IEEE, 2009.

- F. Merrikh-Bayat. Rules for selecting the parameters of oustaloup recursive approximation for the simulation of linear feedback systems containing $\pi^{\lambda}d^{\mu}$ controller. *Communications in Nonlinear Science and Numerical Simulation*, 17:1852–1861, 2012.
- R. Metzler i J. Klafter. The random walks guide to anomalous diffusion: a fractional dynamics approach. *Physics Reports*, 339(1):1–77, 2000.
- R. H. Middleton i G. C. Goodwin. *Digital Control and Estimation: A Unified Approach*. Prentice-Hall International, Englewood Cliffs, New Jersey, 1990.
- W. Mitkowski. Approximation of fractional diffusion-wave equation. *acta mechanica et automatica*, 5:65–68, 2011.
- W. Mitkowski i A. Obrączka. Simple identification of fractional differential equation. *Diffusion and Defect Data-Solid State Data-Part B-Solid State Phenomena*, 1486(180):331, 2011.
- W. Mitkowski i K. Oprzedkiewicz. Fractional-order $p2d \beta$ controller for uncertain parameter DC motor. In *Lecture Notes in Electrical Engineering*, pages 249–259. Springer International Publishing, 2013.
- G. Mittag-Leffler. Sur la representation analytique d’une branche uniforme d’une fonction monogène: cinquieme note. *Acta Mathematica*, 1905(02403200), 1905.
- C. A. Monje, B. M. Vinagre, V. Feliu, i Y. Chen. Tuning and auto-tuning of fractional order controllers for industry applications. *Control Engineering Practice*, 16(7):798–812, 2008.
- C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, i V. Feliu. *Fractional-order systems and controls. Fundamentals and applications*. Advances in Industrial Control. Springer-Verlag, London, 2010.
- B. Moore. Principal component analysis in linear systems: Controllability, observability and model reduction. *IEEE Transactions on Automatic Control*, AC–26(1):17–32, 1981.
- A. Obrączka i W. Mitkowski. The comparison of parameter identification methods for fractional, partial differential equation. *Solid State Phenomena*, 210:265–270, 2014.
- K. Oprzędkiewicz i K. Dziedzic. A tuning of a fractional order PID controller with the use of particle swarm optimization method. In *Artificial Intelligence and Soft Computing*, pages 394–407. Springer International Publishing, 2017.
- K. Oprzedkiewicz i W. Mitkowski. Parameter identification for non integer order, discrete, state space model of heat transfer process using CFE approximation. In *2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*. IEEE, 2018a.

- K. Oprzedkiewicz i W. Mitkowski. A memory-efficient noninteger-order discrete-time state-space model of a heat transfer process. *International Journal of Applied Mathematics and Computer Science*, 28(4):649–659, 2018b.
- K. Oprzedkiewicz, M. Chochol, W. Bauer, i T. Meresinski. Modeling of elementary fractional order plants at PLC SIEMENS platform. In *Lecture Notes in Electrical Engineering*, pages 265–273. Springer International Publishing, 2015.
- K. Oprzedkiewicz, E. Gawin, i W. Mitkowski. A PLC implementation of PSE approximant for fractional order operator. In *Lecture Notes in Electrical Engineering*, pages 102–112. Springer International Publishing, 2018a.
- K. Oprzedkiewicz, W. Mitkowski, E. Gawin, i K. Dziedzic. The caputo vs. caputo-f. operators in modeling of heat transfer process. *Bulletin of the Polish Academy of Sciences: Technical Sciences*, 2018b.
- A. Oustaloup, L. Le Lay, i B. Mathieu. Identification of non integer order system in the time domain, ieeecesa'96, smc imacs multiconference, computational engineering in systems application. In *Symposium on Control, Optimisation and Supervision, Lille, France, 1996*.
- A. Oustaloup, J. Sabatier, i P. Lanusse. From fractal robustness to the CRONE control. *FCAA*, 2(1):1–30, 1999.
- A. Oustaloup, F. Levron, B. Mathieu, i F. M. Nanot. Frequency-band complex noninteger differentiator: characterization and synthesis. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 47(1):25–39, 2000.
- F. Padula i A. Visioli. *Advances in Robust Fractional Control*. Springer International Publishing, 2015.
- I. Petras. Practical aspects of tuning and implementation of fractional-order controllers. In *Volume 3: 2011 ASME/IEEE International Conference on Mechatronic and Embedded Systems and Applications, Parts A and B*. ASME, 2011a.
- I. Petras. *Fractional-Order Nonlinear Systems: Modeling, Analysis and Simulation*. Nonlinear Physical Science. Springer, 2011b. ISBN 9783642181009.
- I. Petras, S. Grega, i L. Dorcak. Digital fractional order controllers realization by pic microprocessor: Experimental results. *arXiv preprint math/0306016*, 2003a.
- I. Petras, S. Grega, and L. Dorcak. Digital fractional order controllers realization by pic microprocessor: Experimental results. *arXiv preprint math/0306016*, 2003b.

- P. Piątek i J. Baranowski. Investigation of Fixed-Point Computation Influence on Numerical Solutions of Fractional Differential Equations. *Acta Mechanica et Automatica*, 5(2):101–107, 2011.
- P. Piątek, J. Baranowski, M. Zagórska, W. Bauer, i T. Dziwiński. Bi-fractional filters, part 1: Left half-plane case. In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Order Calculus and its Applications*. Springer, 2014a.
- P. Piątek, M. Zagórska, J. Baranowski, W. Bauer, i T. Dziwiński. Discretisation of different non-integer order system approximations. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference on*, pages 429–433. IEEE, 2014b.
- A. Piłat. *Sterowanie układami magnetycznej lewitacji*. PhD thesis, Akademia Górniczo-Hutnicza im. Stanisława Staszica, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Kraków, 2002. Promotor: W. Grega.
- I. Podlubny. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. Mathematics in Science and Engineering. Elsevier Science, 1998.
- I. Podlubny. *Fractional Differential Equations*. Academic Press, 1999.
- J. C. Prajapati, K. B. Kachhia, i S. P. Kosta. Fractional calculus approach to study temperature distribution within a spinning satellite. *Alexandria Engineering Journal*, 55(3): 2345–2350, 2016.
- W. H. Press, S. A. Teukolsky, W. T. Vetterling, i B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.
- C. Psychalinos, G. Tsirimokou, i A. S. Elwakil. Switched-capacitor fractional-step butterworth filter design. *Circuits, Systems, and Signal Processing*, 35(4):1377–1393, 2016. ISSN 1531-5878.
- A. G. Radwan, A. A. Emira, A. M. AbdelAty, i A. T. Azar. Modeling and analysis of fractional order DC-DC converter. *ISA Transactions*, 82:184–199, 2018.
- S. D. Roy. On the realization of a constant-argument immittance or fractional operator. *IEEE Transactions on Circuit Theory*, 14(3):264–274, 1967.
- A. Ruszewski. Stability regions of closed loop system with time delay inertial plant of fractional order and fractional order pi controller, 2008.

- M. Rydel, R. Stanisławski, G. Bialic, i K. Latawiec. Modeling of discrete-time fractional-order state space systems using the balanced truncation method. In *Theoretical Developments and Applications of Non-Integer Order Systems*, pages 119–127. Springer International Publishing, 2016a.
- M. Rydel, R. Stanisławski, M. Gałek, i K. J. Latawiec. Modeling of fractional-order integrators and differentiators using tustin-based approximations and model order reduction techniques. In *Lecture Notes in Electrical Engineering*, pages 277–286. Springer International Publishing, 2016b.
- E. Scalas, R. Gorenflo, and F. Mainardi. Fractional calculus and continuous-time finance. *Physica A: Statistical Mechanics and its Applications*, 284(1-4):376–384, 2000.
- R. Schumer, M. M. Meerschaert, i B. Baeumer. Fractional advection-dispersion equations for modeling transport at the earth surface. *Journal of Geophysical Research*, 114, 2009.
- D. Sierociuk. Fractional kalman filter algorithms for correlated system and measurement noises. *Control and Cybernetics*, 42(2), 2013.
- D. Sierociuk, T. Skovranek, M. Macias, I. Podlubny, I. Petras, A. Zielinski, i P. Ziubinski. Diffusion process modeling by using fractional-order models. *Applied Mathematics and Computation*, 257:2–11, 2015.
- D. Sierociuk, M. Macias, W. Malesza, i G. Sarwas. Dual estimation of fractional variable order based on the unscented fractional order kalman filter for direct and networked measurements. *Circuits, Systems, and Signal Processing*, 35(6):2055–2082, 2016.
- L. Song. A space-time fractional derivative model for european option pricing with transaction costs in fractal market. *Chaos, Solitons & Fractals*, 103:123–130, 2017.
- C. ST. Stm32cubemx - stm32cube initialization code generator - stmicroelectronics, 2019a. URL <https://www.st.com/en/development-tools/stm32cubemx.html>.
- C. ST. Um1525 user manual. Technical report, Company ST, 2019b.
- R. Stanislawski, W. P. Huneke, i K. J. Latawiec. Finite approximations of a discrete-time fractional derivative. In *2011 16th International Conference on Methods & Models in Automation & Robotics*. IEEE, 2011.
- R. Stanisławski, M. Rydel, i K. J. Latawiec. Modeling of discrete-time fractional-order state space systems using the balanced truncation method. *Journal of the Franklin Institute*, 354(7):3008–3020, 2017.

- M. A. Taleb, O. Béthoux, i E. Godoy. Identification of a PEMFC fractional order model. *International Journal of Hydrogen Energy*, 42(2):1499–1509, 2017.
- I. Tejado, D. Valério, E. Pérez, i N. Valério. Fractional calculus in economic growth modeling: the spanish and portuguese cases. *International Journal of Dynamics and Control*, 5(1):208–222, 2015.
- A. Tepljakov, E. Petlenkov, i J. Belikov. Embedded system implementation of digital fractional filter approximations for control applications. In *The 21th International Conference Mixed Design of Integrated Circuits and Systems*, page 441–445, Lublin, Poland, June 2014.
- A. Tepljakov, E. Petlenkov, i J. Belikov. Robust FOPI and FOPID controller design for FFOPDT plants in embedded control applications using frequency-domain analysis. In *The 54rd American Control Conference*, page 3868–3873, Chicago, IL, USA, July 2015.
- A. Tepljakov, E. Petlenkov, and J. Belikov. Digital implementation of retuning fractional controllers for an existing closed-loop magnetic levitation control system. In *Proc. of 39th International Conference on Telecommunications and Signal Processing*, Vienna, Austria, 2016.
- U. Thakar, V. Joshi, i V. Vyawahare. Design of fractional-order PI controllers and comparative analysis of these controllers with linearized, nonlinear integer-order and nonlinear fractional-order representations of PMSM. *International Journal of Dynamics and Control*, 5(1):187–197, 2016.
- P. J. Torvik i R. L. Bagley. On the appearance of the fractional derivative in the behavior of real materials. *Journal of Applied Mechanics*, 51(2):294, 1984.
- J. C. Trigeassou, T. Poinot, J. Lin, A. Oustaloup, i F. Levron. Modeling and identification of a non integer order system. In *1999 European Control Conference (ECC)*. IEEE, aug 1999.
- G. Tsirimokou, C. Psychalinos, i A. S. Elwakil. Emulation of a constant phase element using operational transconductance amplifiers. *Analog Integrated Circuits and Signal Processing*, 85(3):1–12, 2016.
- G. Tsirimokou, C. Psychalinos, i A. Elwakil. *Design of CMOS Analog Integrated Fractional-Order Circuits*. Springer International Publishing, 2017. doi: 10.1007/978-3-319-55633-8.
- D. Valério i J. da Costa. Tuning of fractional PID controllers with ziegler–nichols-type rules. *Signal Processing*, 86(10):2771–2784, 2006.

- D. Valerio i J. Sa da Costa. Levy's identification method extended to commensurate fractional order transfer functions. In *Fifth EUROMECH Nonlinear Dynamics Conference*, pages 1357–1366, 2005.
- A. Varga. Balancing-free square-root algorithm for computing singular perturbation approximations. In *30th IEEE Conference on Decision and Control*, volume 2, pages 1062–1065, 1991.
- M. Čech i M. Schlegel. The fractional-order pid controller outperforms the classical one. pages 1–6, Pardubice, 2006. Technical University. ISBN 80-7194-860-8.
- P. Veeraian, U. Gandhi, i U. Mangalanathan. Fractional order linear variable differential transformer: Design and analysis. *AEU - International Journal of Electronics and Communications*, 79:141–150, 2017.
- D. Verotta. Fractional dynamics pharmacokinetics–pharmacodynamic models. *Journal of Pharmacokinetics and Pharmacodynamics*, 37(3):257–276, 2010.
- B. Vinagre, I. Podlubny, A. Hernandez, i V. Feliu. Some approximations of fractional order operators used in control theory and applications. *Fractional calculus and applied analysis*, 3(3):231–248, 2000a.
- B. M. Vinagre, I. Podlubny, i V. Feliu. Some approximations of fractional order operators used in control theory and applications. *Journal of Fractional Calculus and Applied Analysis*, pages 231–248, 2000b.
- V. A. Vyawahare i P. Nataraj. Fractional-order modeling of neutron transport in a nuclear reactor. *Applied Mathematical Modelling*, 37(23):9747–9767, 2013a.
- V. A. Vyawahare i P. Nataraj. Development and analysis of some versions of the fractional-order point reactor kinetics model for a nuclear reactor with slab geometry. *Communications in Nonlinear Science and Numerical Simulation*, 18(7):1840–1856, 2013b.
- V. A. Vyawahare, P. S. V. Nataraj, G. Espinosa-Paredes, i R.-I. Cázares-Ramírez. Nuclear reactor with subdiffusive neutron transport: development of linear fractional-order models. *International Journal of Dynamics and Control*, 5(4):1182–1200, 2016.
- J. Wang. *Real-Time Embedded Systems*. Wiley Publishing, 1st edition, 2014.
- M. Weilbeer. *Efficient Numerical Methods for Fractional Differential Equations and Their Analytical Background*. Papierflieger, 2006.
- D. Xue, C. Zhao, i Y. Chen. Fractional order PID control of a DC-motor with elastic shaft: a case study. In *2006 American Control Conference*. IEEE, 2006.

- C. Yu. *Autotuning of PID Controllers*. Springer-Verlag, 2006.
- M. Zagórowska, J. Baranowski, P. Bania, P. Piątek, W. Bauer, i T. Dziwiński. Impulse response approximation method for "fractional order lag". In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Order Calculus and its Applications*. Springer, 2014a.
- M. Zagórowska, J. Baranowski, W. Bauer, T. Dziwiński, i P. Piątek. Lyapunov direct method for non-integer order systems. In *Materiały XXXVII Międzynarodowej konferencji z podstaw elektrotechniki i teorii obwodów IC-SPE TO*, pages 64–65, Ustroń, 21–24. 05. 2014b.
- M. Zagórowska, J. Baranowski, W. Bauer, T. Dziwiński, i P. Piątek. Simulational analysis of non-integer order system behaviour under relay control. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference On*, pages 374–378. IEEE, 2014c.
- M. Zagórowska, J. Baranowski, W. Bauer, T. Dziwiński, P. Piątek, i W. Mitkowski. Lapunov direct method for non-integer order systems. In *Advances in Modelling and Control of Noninteger-order Systems - 6th Conference on Non-Integer Order Calculus and its Applications*. Springer, 2014d.
- M. Zagórowska, J. Baranowski, P. Bania, W. Bauer, T. Dziwiński, i P. Piątek. Parametric optimization of pd controller using laguerre approximation. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference On*, pages 104–109. IEEE, 2015.
- M. Zagórowska, J. Baranowski, W. Bauer, T. Dziwiński, i P. Piątek. Hurwitz-like criterion for nonlinear non-integer order systems. In *Materiały XXXVIII Międzynarodowej konferencji z podstaw elektrotechniki i teorii obwodów IC-SPE TO*, pages 53–54, 20–23. 05. 2015.
- M. Zagórowska, J. Baranowski, W. Bauer, T. Dziwiński, P. Piątek, i W. Mitkowski. Lyapunov direct method for non-integer order systems. In K. J. Latawiec, M. Łukaniszyn, and R. Stanisławski, editors, *Advances in Modelling and Control of Non-integer-Order Systems*, volume 320 of *Lecture Notes in Electrical Engineering*, pages 221–228. Springer International Publishing, 2015.
- M. Zecova i J. Terpak. Heat conduction modeling by using fractional-order derivatives. *Applied Mathematics and Computation*, 257:365–373, 2015.
- Y. Zhang, H. Sun, B. Lu, R. Garrard, i R. M. Neupauer. Identify source location and release time for pollutants undergoing super-diffusion and decay: Parameter analysis and model evaluation. *Advances in Water Resources*, 107:517–524, 2017a.

Y. Zhang, H. Sun, H. H. Stowell, M. Zayernouri, i S. E. Hansen. A review of applications of fractional calculus in earth system dynamics. *Chaos, Solitons & Fractals*, 102:29–46, 2017b.

OŚWIADCZENIE AUTORA PRACY

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ ROZPRAWĘ DOKTORSKĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

.....

PODPIS