

Zastosowanie algorytmów regulacji
wielowymiarowej w problematyce
wydajnej pracy sieci komputerowych
z przełączaniem pakietów

Praca Doktorska

mgr inż. Piotr Dzierwa

Wydział Elektrotechniki, Automatyki i Informatyki
Politechnika Opolska



POLITECHNIKA
OPOLSKA

Promotor: dr hab. inż. Wojciech Hunek, prof. Politechniki Opolskiej

OPOLE 2018

mojej Żonie, Dzieciom oraz Rodzicom poświęcam

Podziękowanie

Prof. Wojciechowi Hunkowi pragnę złożyć najserdeczniejsze podziękowanie za udzielenie mi niezawodnego wsparcia oraz za nieustającą pracę nad przekazaniem mi swojego perfekcjonizmu badawczego.

POLITECHNIKA OPOLSKA
WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI

PRACA DOKTORSKA

na temat:

Zastosowanie algorytmów regulacji wielowymiarowej w problematyce wydajnej
pracy sieci komputerowych z przełączaniem pakietów

autor:

mgr inż. Piotr Dzierwa

promotor:

dr hab. inż. Wojciech Hunek, prof. PO

OPOLE 2018

STRESZCZENIE

Rozprawa doktorska przedstawia nowe podejście do problematyki wydajnej pracy sieci komputerowych z przełączaniem pakietów. Pokazano w niej, że wskutek zastosowania wielowymiarowej regulacji dedykowanej liniowym stacjonarnym obiektom opisanym modelem typu wejście-wyjście z czasem dyskretnym, w efektywny sposób można zarządzać procesem kolejkowania pakietów we wspomnianych sieciach. Połączenie mechanizmów identyfikacji i adaptacji z zaawansowanym aparatem matematycznym, głównie w kontekście użytych procedur sterowania, zaowocowało autorską nową metodą potwierdzoną przeprowadzonymi badaniami symulacyjnymi. Pozytywnie przyjęta przez międzynarodową społeczność automatyków publikacja tzw. ‘filadelfijska’ z poruszanej tu tematyki świadczyć może o aktualności problemów zawartych w ramach niniejszej dysertacji doktorskiej.

OPOLE UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF ELECTRICAL, CONTROL AND COMPUTER
ENGINEERING

Ph. D. Thesis

on:

An application of multivariable control algorithms in the problem of efficient
work of computer networks with packet switching

by

Piotr Dzierwa, M.Sc., Eng.

Advisor:

**Wojciech Hunek, Ph.D., Eng.,
Assoc. Prof. at the Opole University of Technology**

OPOLE 2018

ABSTRACT

The Ph.D. thesis presents a new approach to the problem of efficient work of computer networks with packet switching. It is shown, that through an application of multivariable control dedicated to linear time-invariant systems described by an input-output model in discrete-time domain we can manage the queuing process in the above-mentioned networks in an effective way. A connection of the identification and adaptation mechanisms with an advanced mathematical calculus, mainly in the context of the applied control laws, resulted in a new author's method confirmed by the simulation tests. Accepted by the international control community a publication from the ISI Master Journal List related to the subject discussed under this doctoral dissertation can testify to the current problems contained here.

Symbole

$(\cdot)^R$	symbol niejednoznacznej inwersji prawostronnej
$(\cdot)_0^R$	symbol jednoznacznej inwersji prawostronnej
$\underline{A}'(q^{-1})$	wielomian macierzowy zmiennej q^{-1} równy $\underline{A}(q^{-1})(1 - q^{-1})$
$\underline{A}, \underline{B}, \underline{C}, \underline{D}$	macierze z elementami w formie wielomianów zmiennej q^{-1}
A^+	uogólniona odwrotność Moore'a-Penrose'a macierzy A
A, B, C, D	macierze z elementami w R
$A \in R^{n_y \times n_y}[w],$	macierze lewostronnie względnie pierwsze z elementami w formie wielomianów zmiennej w
$B \in R^{n_y \times n_u}[w]$	
$A_{m \times n}$	macierz o elementach rzeczywistych o wymiarze $m \times n$
$d = n - m$	opóźnienie obiektu, gdzie n i m są rzędami odpowiednio wielomianów $A(z)$ i $B(z)$
$e(t)$	n_y -wektor nieskorelowanego szumu białego o zerowej wartości oczekiwanej w dziedzinie t
γ, α	stałe przyjęte podczas procesu estymacji parametrów modelu
$G \in R^{n_y \times n_u}(p)$	$n_y \times n_u$ -macierzowa transmitancja operatorowa w dziedzinie w
I_n	macierz jednostkowa o wymiarze n
n	liczba zmiennych stanu (obiekt)
n_u	liczba wejść (obiekt)
n_y	liczba wyjść (obiekt)
p	operator zespolony
$P(q^{-1}), Q(q^{-1})$	arbitralnie dobierane macierze parametryczne lub wielomianowe używane w sterowaniu GMVC

$p(t)$	funkcja Drop/Mark Probability opisująca prawdopodobieństwo odrzutu pakietów
$\underline{q}(t + d)$	stochastyczny predyktor wyjścia wielowymiarowej sieci komputerowej z przełączaniem pakietów
\underline{q}_{ss}	błąd w stanie ustalonym
q^{-1}	operatory przesunięcia wstecz
$q_{ref}(t + d)$	wartość referencyjna/zadana długości kolejki
R	ciało liczb rzeczywistych
$R^{m \times n}[w]$	macierz o wymiarach $m \times n$ zmiennej zespolonej w
$R(w)$	ciało funkcji wymiernych zmiennej zespolonej w o rzeczywistych współczynnikach
$R[w]$	pierścień wielomianów zmiennej zespolonej w
s, w, z	operatory zespolone
$S(A, B, C, D)$	obiekt dyskretny opisany w przestrzeni stanów
t	czas ciągły lub dyskretny
$\theta(t)$	wektor parametrów modelu ARMAX
$\hat{\theta}(t)$	estymator wektora parametrów $\theta(t)$
$u(t)$	wektor wejść o wymiarze n_u w dziedzinie t
$x(t)$	wektor stanu o wymiarze n w dziedzinie t
$y(t)$	wektor wyjść o wymiarze n_y w dziedzinie t

Skróty

ABR	maksymalna możliwa przepustowość (ang. Available Bit Rate)
AQM	aktywne zarządzanie kolejką (ang. Active Queue Management)
AR	model autoregresyjny (ang. AutoRegressive)
ARMA	model autoregresyjny ze średnią ruchomą (ang. AutoRegressive Moving Average)
ARMAX	model autoregresyjny ze średnią ruchomą i zewnętrznym wejściem (ang. AutoRegressive Moving Average with eXogenous Input)
ATM	asynchroniczny tryb transmisji (ang. Asynchronous Transfer Mode)
CNN	sieć neuronowo-komórkowa (ang. Cellular Neural Network)
DGMAVC	dyskretne uogólnione adaptacyjne sterowanie minimalnowariancyjne (ang. Discrete-Time Generalized Minimum Variance Adaptive Control)
DGMVC	dyskretne uogólnione sterowanie minimalnowariancyjne (ang. Discrete-Time Generalized Minimum Variance Control)
DMVAC	dyskretne adaptacyjne sterowanie minimalnowariancyjne (ang. Discrete-Time Minimum Variance Adaptive Control)
DMVC	dyskretne sterowanie minimalnowariancyjne (ang. Discrete-Time Minimum Variance Control)
DTSS	systemy symulacji zdarzeń w czasie dyskretnym (ang. Discrete-Time Simulation Systems)
ERLS	rozszerzona rekurencyjna metoda najmniejszych kwadratów (ang. Extended Recursive Least-Squares)
FIFO	kolejka pierwszy na wejściu/pierwszy na wyjściu (ang. First-In/First-Out)

GPC	uogólnione sterowanie predykcyjne (ang. Generalized Predictive Control)
LTI	liniowy stacjonarny (ang. Linear Time-Invariant)
MFD	macierzowy opis frakcyjny (ang. Matrix Fraction Description)
MIAC	regulator adaptacyjny oparty na identyfikacji modelu (ang. Model Identification Adaptive Controller)
MIMO	wiele-wejść/wiele-wyjść (ang. Multi-Input/Multi-Output)
MPC	sterowanie predykcyjne z przesuwany horyzontem (ang. Model Predictive Control)
MRAC	regulator adaptacyjny z modelem odniesienia (ang. Model Reference Adaptive Controller)
NS2	symulator sieci dla zdarzeń dyskretnych (ang. Network Simulator)
OMNeT++	modułowa, oparta na komponentach biblioteka symulacji C++ oraz zestaw bibliotek (framework), głównie do budowania symulatorów zachowań sieci (ang. Objective Modular Network Testbed in C++)
Opnet	narzędzia do zarządzania wydajnością sieci (ang. Optimized Network Engineering Tools)
QoS	jakość usługi (ang. Quality of Service)
QualNet	platforma do symulacji komunikacji w sieciach
RED	algorytm kolejowania oraz unikania zakleszczeń (ang. Random Early Detection)
RHC	sterowanie z przesuwany horyzontem (ang. Receding Horizon Control)
RLS	rekurencyjna metoda najmniejszych kwadratów (ang. Recursive Least-Squares)
RM	komórka informacji w usłudze ABR (ang. Rate Marking)
SISO	jedno-wejście/jedno-wyjście (ang. Single-Input/Single-Output)
TCP/IP	teoretyczny model warstwowej struktury protokołów komunikacyjnych (ang. Transmission Control Protocol/Internet Protocol)
WWW	hipertekstowy, multimedialny, internetowy system informacyjny (ang. World Wide Web)

Spis treści

Symbole	vi
Skróty	viii
Rozdział 1	
Wstęp	1
1.1 Sformułowanie problemu	2
1.2 Cel, teza i zakres pracy	3
1.3 Aktualny stan wiedzy	4
Rozdział 2	
Model dynamicznych	
liniowych stacjonarnych	
układów wielowymiarowych	6
2.1 Inwersje niekwadratowych macierzy wielomianowych	9
2.1.1 Uogólniona inwersja Moore'a-Penrose'a	9
2.1.2 T -inwersja	10
Rozdział 3	
Dyskretne sterowanie minimalnowariancyjne	13
3.1 Sterowanie minimalnowariancyjne	13
3.2 Przykłady symulacyjne	15
3.2.1 Przykład 1	15
3.2.2 Przykład 2	17

3.2.3	Przykład 3	18
3.3	Uogólnione sterowanie minimalnowariancyjne	21
Rozdział 4		
	Istota sterowania adaptacyjnego	22
4.1	Struktury sterowania adaptacyjnego	23
4.2	Model komputerowych sieci kolejkowych z przełączaniem pakietów	26
4.3	Modelowanie zdarzeń w sieciach komputerowych	27
Rozdział 5		
	Sterowanie adaptacyjne sieci komputerowych SISO	
	z wykorzystaniem Drop/Mark Probability Function	30
5.1	Dyskretne minimalnowariancyjne sterowanie adaptacyjne w pętli zamkniętej	30
5.1.1	Przykład symulacyjny 1	32
5.1.2	Dyskretne uogólnione sterowanie adaptacyjne w pętli zamkniętej	36
5.1.3	Przykład symulacyjny 2	39
Rozdział 6		
	Sterowanie adaptacyjne sieci komputerowych MIMO	
	z wykorzystaniem Drop/Mark Probability Function	41
6.1	Przykład symulacyjny 1	42
6.2	Przykład symulacyjny 2	44
Rozdział 7		
	Podsumowanie i problemy otwarte	47
Literatura		49
Załącznik A.		
	Schemat układu sterowania MV/perfekcyjnego obiektem	
	o trzech wejściach i dwóch wyjściach	59

Załącznik B.	
Procedura wyznaczania algorytmu sterowania minimalnowa- riancyjnego/perfekcyjnego	61
Załącznik C	
Autorskie procedury wymagane podczas prowadzenia badań symulacyjnych	64
Spis rysunków	101

Rozdział 1

Wstęp

Proces ulepszenia niezawodności funkcjonowania sieci komputerowych został zdezeterminowany przez dynamiczny rozwój technologii szybkiej transmisji danych. Problematykę tę w głównej mierze podjęły systemy inteligentne, odpowiedzialne za przełączanie pakietów. Wysokie normy jakościowe, narzucone na przykład przez standard jakości usług, *ang. Quality of Service (QoS)*, są wyznacznikiem jakości transmisji i ich spełnienie staje się w dzisiejszym, szybko rozwijającym się społeczeństwie informacyjnym, coraz trudniejsze [1, 2]. Problem zapewnienia wydajnej pracy sieci staje się jednym z najważniejszych problemów funkcjonowania współczesnego Internetu. W celu stabilnej pracy sieci pakietowych stosuje się szeroką gamę mechanizmów analitycznych i heurystycznych w zadaniach optymalnej pracy sieci komputerowych z przełączaniem pakietów. Znamiennym jest, że z punktu widzenia odpornej pracy, problem zakleszczeń pakietów, zdaniem autora, został dotychczas niewystarczająco poruszony, co otwiera duże możliwości badawcze w podnoszonej problematyce. Dlatego w ramach niniejszej pracy podjęto próbę implementacji znanych z zagadnień teorii sterowania i systemów algorytmów adaptacyjnej regulacji predykcyjnej do zagadnień optymalnej pracy wspomnianych sieci. Rzecz o tyle ważna, bo analizie i syntezie zostały poddane sieci komputerowe z nierówną liczbą wejść i wyjść, tzw. niekwadratowe układy wielowymiarowe. Struktura taka wymusza zastosowanie deterministycznych algorytmów sterowania, wykorzystujących niekwadratowe macierze wielomianowe. Okazuje się, że problem użycia wspomnianych algorytmów jest coraz częściej po-

ruszany, co z powodzeniem osadza niniejszą pracę doktorską w praktyce, powodując jednocześnie znaczną ekspolację zagadnień pogranicza automatyki i informatyki.

Reasumując, dysertacja doktorska podsumowuje oraz, zdaniem autora, znacząco rozszerza tematykę implementacji znanych z problematyki teorii sterowania algorytmów wielowymiarowej regulacji predykcyjnej w zadaniach optymalnego przełączania pakietów w sieciach komputerowych. Wskazuje także na możliwe kierunki rozwoju pokazanych tu nowych rozwiązań we współczesnych systemach kolejkowych.

1.1 Sformułowanie problemu

W celu poprawy funkcjonowania sieci z przełączaniem pakietów i spełnienia wysokich norm jakościowych, implementowane są różnego rodzaju mechanizmy regulujące. Do najbardziej popularnych z nich można zaliczyć rozwiązania oparte na systemach kolejkowych [3, 4]. Analizując różne typy modeli kolejkowych należy zwrócić uwagę, aby przepustowość modelowanej sieci komputerowej była w jak największym stopniu wykorzystana, przy jednoczesnym ograniczeniu potrzeby retransmisji przesyłanych pakietów oraz minimalizowaniu czasu przebywania danych w systemie. Najbardziej interesującymi mechanizmami, wychodzącymi na przeciw wspomnianym wymaganiom, wydają się być te, wykorzystujące technikę aktywnego zarządzania kolejką, *ang. Active Queue Management* (AQM), zarówno podejścia analityczne, jak i rozwiązania heurystyczne [5, 6, 7, 8, 9, 10, 11]. Klasycznym sposobem zapobiegania problemom zarządzania kolejką jest organizacja bufora za pomocą mechanizmu, tzw. *ang. First-In/First-Out* (FIFO) [12]. Mechanizm ten 'odcina' pakiety przychodzące w momencie zapełnienia się przestrzeni bufora. Jest to rozwiązanie najprostsze, a jednocześnie najczęściej stosowane w routerach sieciowych. Zarządzając kolejką przy pomocy tego algorytmu, dopuszcza się możliwość zakleszczenia sieci [13, 14] w sytuacjach częstego przepełnienia bufora, dodatkowo reguluje on szybkość nadawania pakietów dopiero po przepełnieniu bufora. Takie pasywne zachowania w konsekwencji przy-

noszą opóźnienia działania kolejki i ostatecznie nieefektywne wykorzystanie łącza [13]. W literaturze przedmiotu spotkać można interesujące, równie wydajne, a w licznych przypadkach, rozwiązania angażujące technologię AQM i jednowymiarowe, tzw. *ang. Single-Input/Single-Output* (SISO), modele sieci komputerowych [13, 15, 16]. Znamiennym jest, że niniejsza praca doktorska efektywnie wykorzystuje przytoczony mechanizm AQM, efektywnie użyty w wielowymiarowych systemach niekwadratowych o różnej liczbie wejść i wyjść, tzw. obiektach *ang. Multiple-Input/Multiple-Output* (MIMO). W takich systemach istnieje możliwość wykorzystania dostępnej maksymalnej przepustowości sieci poprzez odpowiednie zarządzanie różnymi grupami użytkowników. Otrzymać to można wskutek zastosowania wielowymiarowych algorytmów regulacji predykcyjnej, angażujących tzw. inwersje macierzy niekwadratowych [17, 18], w sposób istotny realizujący 'odrzut' pakietów poszczególnych grup użytkowników przy zachowaniu przyjętych norm jakościowych.

1.2 Cel, teza i zakres pracy

Celem niniejszej pracy jest przeniesienie złożonych, spotykanych we współczesnej teorii sterowania, algorytmów regulacji predykcyjnej, w szczególności 'okołominimalnowariancyjnej', na grunt nowoczesnych metod zarządzania kolejkowymi sieciami komputerowymi, a także konfrontacja nowego podejścia z istniejącymi mechanizmami.

Wobec powyższego, teza pracy została sformułowana w sposób następujący:

Teza. *Zastosowanie algorytmów regulacji wielowymiarowej stwarza możliwości poprawy przepustowości sieci komputerowych z przełączaniem pakietów wykorzystujących mechanizm kolejowania.*

Celem naukowym pracy jest zatem wprowadzenie nowego podejścia do problematyki przełączania pakietów w kolejkowych sieciach komputerowych z użyciem

algorytmów sterowania dedykowanych niekwadratowym obiektom wielowymiarowym. W szczególności zostaną użyte inwersje niekwadratowych macierzy, w sposób pośredni determinujący retransmisję pakietów, przy zachowaniu maksymalnej wydajności wspomnianych sieci.

W pracy, obok jednowymiarowych, zostaną użyte głównie (niekwadratowe) wielowymiarowe algorytmy adaptacyjnego sterowania minimalnowariancyjnego i adaptacyjnego uogólnionego sterowania minimalnowariancyjnego. Użycie analitycznych algorytmów *ang. Random Early Detection* (RED), wydaje się być alternatywną, a nawet konkurencją dla istniejących dotychczas rozwiązań dedykowanych problematyce optymalnej pracy sieci komputerowych. Znamiennym jest, że nowy aparat może być użyty nie tylko w sieciach TCP/IP, ale także w sieciach, gdzie nie występuje mechanizm retransmisji danych. Potwierdzeniem tego są szeroko zakrojone badania, częściowo zobrazowane przykładami symulacyjnymi w ramach niniejszej pracy.

1.3 Aktualny stan wiedzy

Typową metodą zarządzania kolejką w routerach sieciowych jest wspomniany na wstępie mechanizm FIFO, gdzie urządzenie sieciowe buforuje maksymalną ilość pakietów przychodzących, ucinając te, które są pakietami nadmiarowymi. Zarządzając kolejką przy pomocy tego algorytmu, dopuszcza się możliwość zakleszczenia sieci w sytuacjach częstego przepełnienia bufora. Jednakże w literaturze przedmiotu spotkać można bardziej zaawansowane metody zarządzania kolejką [19]. Alternatywą do rozwiązania FIFO wydaje się być algorytm kolejkowania oraz unikania zakleszczeń RED, redukujący problem zakleszczeń sieci poprzez monitorowanie średniej wartości bufora kolejki. Bazuje on na wyznaczeniu prawdopodobieństw statystycznych odrzucenia pakietu. Prawdopodobieństwo bliskie zeru mówi, że bufor jest prawie pusty i dozwolone jest przyjmowanie pakietów przychodzących z podsieci. Podczas procesu zapełniania bufora prawdopodobieństwo odrzucenia pakietu rośnie, osiągając wartość liczbową 1, gdy bufor jest pełny. Zastosowanie algorytmów RED gwarantuje wcześniejsze wykrycie zaklesz-

czeń, przy jednoczesnym wypełnieniu normy QoS. Znane są różne metody rozwiązania problemu wydajnego zarządzania przełączaniem pakietów i, co za tym idzie, poprawy wydajności sieci. Podejście oparte na sieciach neuronowo-komórkowych, *ang. Cellular Neural Network* (CNN) [20], czy neuronowo-rozmytych [21], może być efektywnie stosowane do szeregowania w czasie rzeczywistym komórek w szybkich komutatorach ATM, w szczególności w komutatorach opartych na architekturze matrycowej z wirtualizacją buforowania na wyjściu. Przytoczona metoda jest jednak dedykowana sieciom typu ATM i jej stosowalność sprowadza się do ograniczeń wynikających z charakterystyki samej sieci neuronowo-komórkowej. Natomiast klasyczna sieć neuronowa [22] także może służyć jako narzędzie odpowiadające za rozwiązywanie problemów związanych z wydajną pracą sieci komputerowych. Możliwość adaptacji do zmieniających się warunków pracy różnych architektur sieciowych przemawia za korzyścią użycia tego ciekawego podejścia. Niestety, modele odwzorowujące badaną rzeczywistość powstają tutaj w drodze adaptacji, bazującej często na niedokładnej wiedzy eksperckiej. Sieć neuronowa może zatem nie spełniać restrykcyjnych ram czasowych przełączania pakietów. Innym, niemniej interesującym rozwiązaniem, jest podejście oparte na zastosowaniu zmiennych niepewnych do oceny stabilności systemów sterowania przeciążeniem w sieciach ATM [23]. Propozycja ta oparta jest na rozkładzie niepewności zadany przez ekspertów dla obiektu, którym jest cała badana sieć komputerowa. Sterowanie obciążeniem odbywa się tutaj przy wykorzystaniu komórek RM [24] i monitorowaniu transmisji źródeł, tzw. technika sterowania przepływem, *ang. Available Bit Rate* (ABR) [25]. Implementujący w tym podejściu model jest modelem bez opóźnienia, co nieznacznie upraszcza badaną rzeczywistość, dając pole do działania innym metodom, niekoniecznie umocowanym w technologii ATM. Podsumowując, w literaturze przedmiotu spotkać można wiele mechanizmów służących poprawie wydajności pracy sieci komputerowych z przełączaniem pakietów. Większość z nich bazuje na podejściu heurystycznym, znikoma liczba wykorzystuje szeroko poruszany w ramach niniejszej dysertacji doktorskiej aparat analityczny [26].

Rozdział 2

Model dynamicznych liniowych stacjonarnych układów wielowymiarowych

W celu analizy i syntezy wszystkich wartości badanych, wynikających z zastosowania algorytmów regulacji wielowymiarowej w problematyce wydajnej pracy sieci komputerowych z przełączaniem pakietów, wprowadźmy w bieżącym rozdziale kilka preliminarii matematycznych pozwalających zrozumieć naturę wspomnianych algorytmów. Zostaną tu omówione głównie modele liniowych stacjonarnych układów wielowymiarowych, pozwalające na wierny opis własności dynamicznych omawianych w pracy sieci komputerowych. Zaczniemy nasze rozważania od wprowadzenia pojęć stowarzyszonych z nomenklaturą obliczeniową macierzy niekwadratowych. Oczywiście, przez macierz niekwadratową należy tutaj rozumieć każdą macierz, składającą się z nierównej liczby kolumn i wierszy [27]. Znamiennym jest, że tematykę macierzy wielomianowych, będąca przewodnią w niniejszej dysertacji doktorskiej, będzie szerzej poruszona w ramach jej kolejnych rozdziałów.

Niech $R[w]$ będzie pierścieniem wielomianów zmiennej zespolonej w o rzeczywistych współczynnikach, gdzie $w = s$. Zbiór wszystkich macierzy o wymiarach $m \times n$ z elementami w $R[w]$ oznaczać będziemy przez $R^{m \times n}[w]$, że $w = s$ dla układów ciągłych oraz $w = z$ dla układów dyskretnych. Natomiast macierzą

wielomianową nazywa się dowolny element należący do zbioru $R^{m \times n}[w]$. Alternatywną realizacją macierzy wielomianowych są wielomiany macierzowe.

Niech ciało funkcji wymiernych zmiennej zespolonej w o rzeczywistych współczynnikach będzie reprezentowane przez $R(w)$. Stąd, zbiór wszystkich macierzy o wymiarach $m \times n$ z elementami w $R(w)$ będziemy oznaczać jako $R^{m \times n}(w)$. Natomiast dowolny element należący do zbioru $R^{m \times n}(w)$ nazywać będziemy macierzą wymierną.

Rozważmy liniowy stacjonarny (LTI) obiekt dyskretny lub ciągły o n_u -wejściach i n_y -wyjściach. Stanowiących odpowiednio wektory $u(t)$ oraz $y(t)$. Dla opisu takich obiektów niekwadratowych wykorzystano tutaj macierzową transmitancję operatorową $G \in R^{n_y \times n_u}(w)$ wyrażoną za pomocą macierzowego opisu frakcyjnego (MFD) $G(w) = A^{-1}(w)B(w) = \tilde{B}(w)\tilde{A}^{-1}(w)$ [28, 29, 30, 31], gdzie macierze wielomianowe $A \in R^{n_y \times n_y}[w]$ i $B \in R^{n_y \times n_u}[w]$ nazywać będziemy lewostronnie względnie pierwszymi, natomiast $\tilde{A} \in R^{n_y \times n_y}[w]$ i $\tilde{B} \in R^{n_u \times n_y}[w]$ są prawostronnie względnie pierwsze.

Przedmiotem badań niniejszej rozprawy doktorskiej będą głównie układy dyskretne opisane formami wielomianowymi w postaci $\underline{A}(z) = z^n A(z^{-1})$ oraz $\underline{B}(z) = z^m B(z^{-1})$, gdzie n i m są rzędami odpowiednio wielomianów $A(z)$ i $B(z)$ (także $\underline{A}(z^{-1}) = I_{n_y} + \dots + \underline{a}_n z^{-n}$ i $\underline{B}(z^{-1}) = \underline{b}_0 + \dots + \underline{b}_n z^{-m}$), gdzie I_{n_y} jest macierzą jednostkową o wymiarze n_y . Zauważmy, że $G(z) = A^{-1}(z)B(z) = z^{-d} \underline{A}^{-1}(z^{-1}) \underline{B}(z^{-1})$, gdzie $d = n - m$ jest opóźnieniem obiektu. Należy zaznaczyć, że chociaż w pracy, jak już wcześniej wspomniano, analizowane są modele opisane operatorem $w = z$, to, zdaniem autora, wszystkie otrzymane w ramach dysertacji doktorskiej rezultaty można bezpośrednio przenieść do problematyki optymalnego sterowania systemami kolejkowymi, opisanymi modelami wykorzystującymi dziedzinę $w = s$. Na koniec wprowadzimy typowy operator przesunięcia wstecz q^{-1} , który w zależności od kontekstu użytkowany będzie zamiennie z formą z^{-1} .

Rozważmy jeszcze typowe obiekty opisane w przestrzeni stanów $S(A, B, C, D)$

układu dyskretnego

$$\begin{aligned}x(t+1) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}, \quad (2.1)$$

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}, \quad (2.2)$$

gdzie $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n_u}$, $C \in \mathbb{R}^{n_y \times n}$, $D \in \mathbb{R}^{n_y \times n_u}$, $x(t)$ jest n -wektorem stanu oraz t jest odpowiednio czasem dyskretnym oraz ciągłym, można więc zapisać $G(w) = C(wI_n - A)^{-1}B + D$.

Chociaż w ramach niniejszej pracy doktorskiej syntetyzuje się algorytmy sterowania bazujące głównie na typowym opisie typu wejście-wyjście, to prowadzone rozważania, zdaniem autora, z powodzeniem można ująć za pomocą aparatu statystycznego. Ta wykraczająca poza ramy dysertacji interesująca równoległość, uwzględniająca często używany w problematyce optymalnego sterowania sieciami komputerowymi rozkład Pareto, warta jest przyszłych prac badawczych.

Przejdźmy zatem do omówienia kluczowych z punktu widzenia poprawnej realizacji dysertacji zagadnień dotyczących algorytmów sterowania. W celu lepszego zrozumienia istoty regulacji na początku omówione zostaną postulowane rozwiązania dedykowane modelom stacjonarnym. Natomiast część druga pracy zawierać będzie nowe podejścia, wykorzystujące modele adaptacyjne. Szeroko zostaną omówione algorytmy (adaptacyjnego) sterowania tzw. 'okołominimalnowariancyjnego' *ang. Minimum Variance Control* (MVC) oraz uogólnionego sterowania minimalnowariancyjnego, *ang. Generalized Minimum Variance Control* (GMVC), także ich wielowymiarowe wersje. Należy jednak zaznaczyć, że o ile algorytmy implementujące modele SISO wykorzystują klasyczny aparat obliczeniowy, to niekwadratowe modele MIMO (składające się z nierównej liczby wejść i wyjść) angażować będą tzw. inwersje macierzowe. Charakterystykę T -inwersji niekwadratowych macierzy wielomianowych, biorących udział w syntezy sieci komputerowych z przełączaniem pakietów prowadzonej w ramach niniejszej pracy doktorskiej, podano w krótkim Podrozdziale 2.1. Natomiast, wyczerpujący prze-

gląd inwersji wraz z ich zastosowaniem w projektowaniu odpornych struktur sterowania minimalnowariancyjnego/perfekcyjnego znaleźć można w [18, 32, 33, 34, 35, 36].

2.1 Inwersje niekwadratowych macierzy wielomianowych

Zagadnienie wyznaczania i wykorzystywania uogólnionych odwrotności macierzy (wielomianowych) jest bardzo ważnym elementem nowoczesnej teorii sterowania. Ponadto z zagadnieniem tym możemy się spotkać między innymi w syntezie sieci elektrycznych i sieci neuronowych, a także w statystyce matematycznej, pozycjonowaniu statków, czy podczas rozwiązywania liniowego zadania najmniejszych kwadratów lub wyznaczaniu rozkładu biegunowego macierzy [37, 38, 39, 40, 41, 42].

2.1.1 Uogólniona inwersja Moore'a-Penrose'a

Najbardziej rozpowszechnionym, a co za tym idzie stosunkowo starszym rozwiązaniem biorącym udział w zadaniu wyznaczenia uogólnionych macierzy odwrotnych, jest metoda Moore'a-Penrose'a [43]. Zgodnie z prezentowanym podejściem dla dowolnej macierzy prostokątnej $A_{m \times n}$ o elementach rzeczywistych istnieje dokładnie jedna macierz $X_{x \times m}$ o elementach rzeczywistych, która spełnia poniższe cztery warunki (tzw. warunki Moore'a-Penrose'a):

$$AXA = A, \tag{2.3}$$

$$XAX = X, \tag{2.4}$$

$$(AX)^T = AX, \tag{2.5}$$

$$(XA)^T = XA. \quad (2.6)$$

Tak otrzymaną macierz X , charakteryzująca się prezentowanymi własnościami, nazywamy pseudoodwrotnością lub odwrotnością Moore'a-Penrose'a macierzy A i oznaczamy jako $X = A^+$, podczas gdy macierze AX i XA są macierzami symetrycznymi [44].

Na przestrzeni ostatnich lat można zaobserwować wiele publikacji dotyczących wyznaczania uogólnionych odwrotności macierzy niekwadratowych, tj. macierzy składających się z nierównej liczby kolumn i wierszy [38, 39, 40, 41, 42, 45, 46]. Analizując mnogość rozwiązań dotyczących odwrotności niekwadratowych macierzy wielomianowych (MIMO) wraz z jednoznaczną inwersją Moore'a-Penrose'a, intrygującym wydaje się prezentowany w pozycji [17, 18] mechanizm wyznaczania odwrotności macierzowych, wprowadzający notację zer sterowniczych, (*ang. Control Zeros*) [18]. Znamiennym jest, że wspomniane narzędzie wykorzystuje tzw. stopnie swobody [18, 47], daje to możliwość uzyskania nieskończonej liczby rozwiązań. Fakt ten, dobrze znany z problematyki analizy i syntezy różnych sterowań 'okołominimalnowariancyjnych' [33, 47, 48, 49], został wykorzystany wstępnie w niniejszej dysertacji doktorskiej w stosunku do równania (3.10). W późniejszych badaniach, nieobjętych ramami rzeczonyj pracy, pozwoli to na głębszą analizę i wykorzystanie prezentowanych na jej łamach rozwiązań.

2.1.2 T -inwersja [50]

Należy w tym miejscu zaznaczyć, że wspomniane inwersje należy zaliczyć do szczególnych przypadków będącej uogólnieniem inwersji Moore'a-Penrose'a, gdzie podlegające inwersji macierze niekwadratowe są pełnego rzędu. Co więcej, klasyczne podejście implementujące macierze z elementami rzeczywistymi - w ramach niniejszej pracy użyto ich wielowymiarowych 'odpowiedników' wprowadzonych w [18]. Stąd poniżej zestawiono dwie kluczowe, z punktu widzenia poprawnej realizacji pracy doktorskiej, definicje T -inwersji. Pierwsza z nich odnosi się do przypadku bardziej ogólnego, tj. dedykowanej obiektom niepełnego normalnego

rzędu T -inwersji Moore'a-Penrose'a, natomiast druga traktuje o T -inwersji prawostronnej o minimalnej normie i T -inwersji lewostronnej minimalno-kwadratowej, gdzie odwracane macierze są pełnego normalnego rzędu [18].

Definicja 2.1.1. *Rozważmy macierz wielomianową $\underline{B}(q^{-1}) = \underline{b}_0 + \underline{b}_1q^{-1} + \dots + \underline{b}_mq^{-m}$ niepełnego normalnego rzędu r . Przeprowadźmy szkieletową faktoryzację $\underline{B}(q^{-1}) = \underline{C}(q^{-1})\underline{D}(q^{-1})$, gdzie macierze wielomianowe $\underline{B}(q^{-1})$, $\underline{C}(q^{-1})$, $\underline{D}(q^{-1})$ są odpowiednio o wymiarach $n_y \times n_u$, $n_y \times r$, $r \times n_u$. Jednoznaczną T -inwersję Moore'a-Penrose'a macierzy $\underline{B}(q^{-1})$ definiuje się jako $\underline{B}_0^\#(q^{-1}) = \underline{D}_0^R(q^{-1})\underline{C}_0^L(q^{-1})$, gdzie $\underline{D}_0^R(q^{-1}) = \underline{D}^T(q^{-1})[\underline{D}(q^{-1})\underline{D}^T(q^{-1})]^{-1}$ i $\underline{C}_0^L(q^{-1}) = [\underline{C}^T(q^{-1})\underline{C}(q^{-1})]^{-1}\underline{C}^T(q^{-1})$.*

Definicja 2.1.2. *Rozważmy macierz wielomianową $\underline{B}(q^{-1}) = \underline{b}_0 + \underline{b}_1q^{-1} + \dots + \underline{b}_mq^{-m}$ pełnego normalnego rzędu n_y (lub n_u). Prawostronną T -inwersję o minimalnej normie (lub lewostronną minimalnokwadratową) macierzy $\underline{B}(q^{-1})$ definiuje się jako $\underline{B}_0^{R1}(q^{-1}) = \underline{B}^T(q^{-1})[\underline{B}(q^{-1})\underline{B}^T(q^{-1})]^{-1}$ (lub $\underline{B}_0^{L1}(q^{-1}) = [\underline{B}^T(q^{-1})\underline{B}(q^{-1})]^{-1}\underline{B}^T(q^{-1})$).*

Należy podkreślić, że wyczerpujący przegląd użycia nowych inwersji we współczesnej syntezy sterowań 'okołominimalnowariancyjnych', syntezy sterowań odpornych na uszkodzenia, syntezy sterowań obiektami ułamkowego rzędu, syntezy sterowań kolejkowymi sieciami komputerowymi z przełączaniem pakietów, znaleźć można w pozycjach [26, 33, 34, 35, 36, 51, 52, 53, 54, 55, 56].

O ile podejście angażujące mechanizm tzw. niejednoznaczności wydaje się być interesującą alternatywą, to w ramach dysertacji ograniczono się jedynie do inwersji prawostronnej o minimalnej normie i inwersji lewostronnej minimalno-kwadratowej. Tylko jeden przypadek symulacyjny z kolejnego Rozdziału 3 obrazuje zastosowanie σ -inwersji wielomianowej, jako skutecznego narzędzia w procesie projektowania algorytmów skutecznego sterowania minimalnowariancyjnego/perfekcyjnego [17, 18, 33, 34, 35, 36]. Oczywiście, metodę tę z pewnością przenieść można do zadań optymalnego sterowania w sieciach komputerowych z przełączaniem pakietów. Ten interesujący wątek badawczy wart jest przyszłej eksploracji naukowej. W nawiązaniu do wspomnianego przykładu symulacyjnego

poniżej podano nową definicję σ -inwersji macierzy wielomianowych pełnego normalnego rzędu. Starsza jej wersja została podana w pozycji [17, 18].

Definicja 2.1.3. [34] *Rozważmy macierz wielomianową $\underline{B}(q^{-1}) = \underline{b}_0 + \underline{b}_1 q^{-1} + \dots + \underline{b}_m q^{-m}$ pełnego normalnego rzędu n_y (lub n_u). Niech $z^s \underline{\beta}(z^{-1}) = \beta(z) \in \mathbb{R}^{n_y \times n_u}[z]$ będzie dowolnym wielomianem macierzowym stopnia s . Załóżmy także, że iloczyn $\underline{B}(q^{-1}) \underline{\beta}^T(q^{-1})$ jest pełnego normalnego rzędu n_y (lub n_u). Wtedy prawostronna σ -inwersja przyjmuje postać*

$$\underline{B}^R(q^{-1}) = \underline{\beta}^T(q^{-1}) [\underline{B}(q^{-1}) \underline{\beta}^T(q^{-1})]^{-1}. \quad (2.7)$$

Uwaga 2.1.1. *Należy zaznaczyć, że wielomian macierzowy $\underline{\beta}(q^{-1})$ zawiera stopnie swobody.*

Uwaga 2.1.2. *Dla $\underline{\beta}(q^{-1}) = \underline{B}(q^{-1})$ σ -inwersja sprowadza się do T -inwersji.*

Mając już definicję T -inwersji wielomianowych przejdźmy zatem do ich użycia w wielowymiarowym sterowaniu minimalnowariancyjnym/perfekcyjnym.

Rozdział 3

Dyskretne sterowanie minimalnowariancyjne

3.1 Sterowanie minimalnowariancyjne

Przejdźmy zatem do omówienia kluczowych z punktu widzenia realizacji niniejszej dysertacji doktorskiej sterowań adaptacyjnych. Aby dobrze zrozumieć istotę adaptacyjnego sterowania minimalnowariancyjnego, zapoznajmy się najpierw z jego nieadaptacyjną wersją. W tym celu posłużymy się znanym z literatury przedmiotu modelem ARMAX [57, 58, 59].

$$\underline{A}(q^{-1})y(t) = q^{-d}\underline{B}(q^{-1})u(t) + \underline{C}(q^{-1})e(t), \quad (3.8)$$

z wielomianami $\underline{B}(q^{-1}) = \underline{b}_0 + \underline{b}_1q^{-1} + \dots + \underline{b}_mq^{-m}$, $\underline{A}(q^{-1}) = I_{n_y} + \underline{a}_1q^{-1} + \dots + \underline{a}_nq^{-n}$ i $\underline{C}(q^{-1}) = \underline{c}_0 + \underline{c}_1q^{-1} + \dots + \underline{c}_kq^{-k}$. Załóżmy, że $d = n - m$ jest czasem opóźnienia, a $e(t)$ jest nieskorelowanym szumem białym o zerowej wartości oczekiwanej, gdzie t jest czasem dyskretnym. Dodatkowo załóżmy, że wielomian macierzowy $\underline{B}(q^{-1})$ jest pełnego normalnego rzędu n_y . Oczywiście wektory $y(t)$ i $u(t)$ reprezentują odpowiednio wyjścia i wejścia obiektu opisanego modelem ARMAX.

Mając już model ARMAX zminimalizujemy wskaźnik jakości

$$\min_{u(t)} E \left\{ [y(t+d) - y_{ref}(t+d)]^T [y(t+d) - y_{ref}(t+d)] \right\}, \quad (3.9)$$

gdzie $y(t+d) = \underline{\tilde{C}}^{-1}(q^{-1})[\underline{\tilde{F}}(q^{-1})\underline{B}(q^{-1})u(t) + \underline{\tilde{H}}(q^{-1})y(t)] + \underline{F}(q^{-1})e(t)$ oraz $y_{ref}(t+d)$ są odpowiednio stochastycznym predyktorem wyjścia i wartością referencyjną/zadaną jednocześnie otrzymując prawo dyskretnego sterowania MV/perfekcyjnego [17, 49, 60, 61, 62]

$$u(t) = \underline{B}^R(q^{-1})\underline{y}(t), \quad (3.10)$$

gdzie $\underline{y}(t) = \underline{\tilde{F}}^{-1}(q^{-1}) \left[\underline{\tilde{C}}(q^{-1})y_{ref}(t+d) - \underline{\tilde{H}}(q^{-1})y(t) \right]$.

Natomiast $(n_y \times n_y)$ -macierze wielomianowe $\underline{\tilde{F}}(q^{-1}) = I_{n_y} + \underline{\tilde{f}}_1 q^{-1} + \dots + \underline{\tilde{f}}_{d-1} q^{-d+1}$ i $\underline{\tilde{H}}(q^{-1}) = \underline{\tilde{h}}_0 + \underline{\tilde{h}}_1 q^{-1} + \dots + \underline{\tilde{h}}_{n-1} q^{-n+1}$ wyznaczone są w oparciu o tożsamość wielomianowo-macierzową (zwaną równaniem diofantycznym)

$$\underline{\tilde{C}}(q^{-1}) = \underline{\tilde{F}}(q^{-1})\underline{A}(q^{-1}) + q^{-d}\underline{\tilde{H}}(q^{-1}), \quad (3.11)$$

oraz faktoryzację

$$\underline{\tilde{C}}(q^{-1})\underline{F}(q^{-1}) = \underline{\tilde{F}}(q^{-1})\underline{C}(q^{-1}), \quad (3.12)$$

gdzie $\underline{F}(q^{-1}) = I_{n_y} + \underline{f}_1 q^{-1} + \dots + \underline{f}_{d-1} q^{-d+1}$, $\underline{\tilde{C}}(q^{-1}) = \underline{\tilde{c}}_0 + \underline{\tilde{c}}_1 q^{-1} + \dots + \underline{\tilde{c}}_k q^{-k}$. Dla systemów prawostronnie odwracalnych symbol $\underline{B}^R(q^{-1})$ oznacza nieskończoną liczbę prawostronnych inwersji macierzy wielomianowej $\underline{B}(q^{-1})$ [17, 34, 36, 63, 64].

Uwaga 3.1.1. *Znamiennym jest, że symbol $(\cdot)^R$ w równaniu (3.10) oznacza każdą niejednoznacznie inwersję prawostronnie odwracalnego wielomianu macierzowego $\underline{B}(q^{-1})$. Jednak w ramach niniejszej dysertacji doktorskiej oparto się głównie na T -inwersji wielomianowej z Rozdziału 2.1.2, oznaczonej symbolem $(\cdot)_0^R$. Stąd, równania 3.10 przyjmuje następującą postać*

$$u(t) = [\underline{B}(q^{-1})]_0^R \underline{y}(t) \quad (3.13)$$

Jednak użycie niejednoznacznych inwersji [17, 18, 26, 33, 34, 35, 36, 49, 63, 64] w zadaniach optymalnego kolejgowania stanowi może ciekawy wątek warty podjęcia wysiłku badawczego w przyszłości.

Uwaga 3.1.2. Należy podkreślić, że sterowanie MV (3.10) sprowadza się do jego deterministycznej postaci, czyli sterowania perfekcyjnego w przypadku braku zakłóceń. W takim scenariuszu w równaniu (3.11) należy położyć $\tilde{C}(q^{-1}) = I_{n_y}$.

W celu zobrazowania omawianych w bieżącym rozdziale treści zaproponowano kilka reprezentatywnych układów sterowania MV/perfekcyjnego złożonych obiektami niekwadratowymi opisanymi modelem ARMAX. W tym celu posłużono się klasyczną prawostronną T -inwersją i, jak wcześniej już wspomniano, niejednoznaczoną σ -inwersję wielomianową zdefiniowano w Rozdziale 2.1.

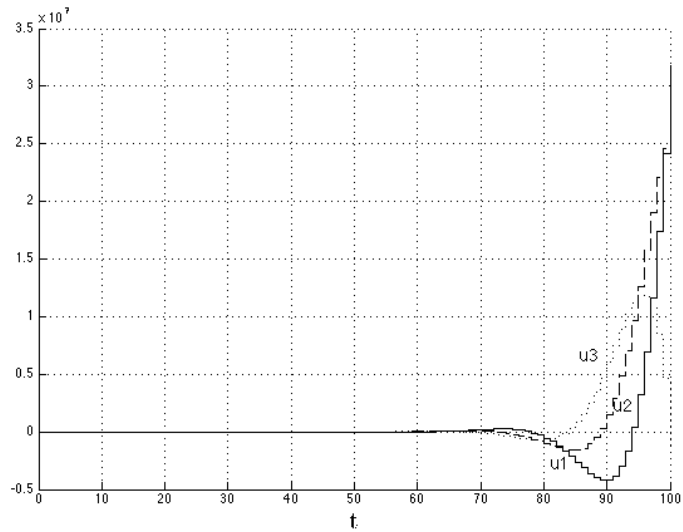
3.2 Przykłady symulacyjne

3.2.1 Przykład 1

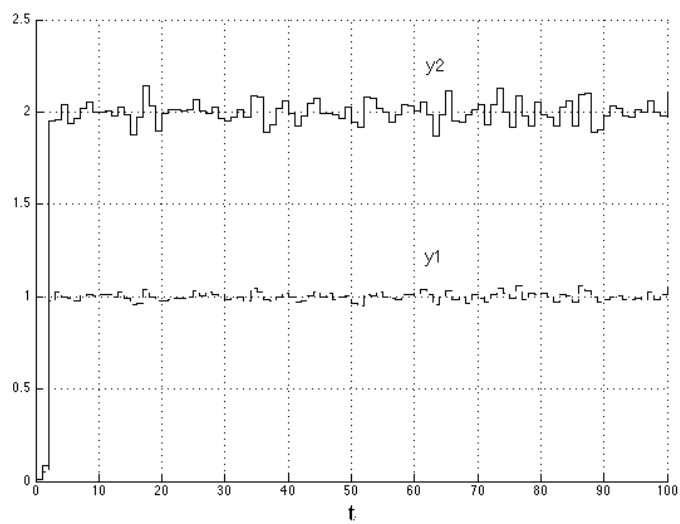
Rozważmy złożony system niestabilny o 3 wejściach i 2 wyjściach z czasem opóźnienia $d = 2$ opisany modelem ARMAX z

$$\begin{aligned} \underline{A}(q^{-1}) &= \begin{bmatrix} 1 - \frac{3}{5}q^{-1} & \frac{2}{25}q^{-1} \\ \frac{1}{20}q^{-1} & 1 - \frac{26}{25}q^{-1} \end{bmatrix}, \\ \underline{B}(q^{-1}) &= \begin{bmatrix} 1 - \frac{101}{100}q^{-1} & \frac{1}{10} - \frac{2}{5}q^{-1} & \frac{1}{100} - \frac{3}{100}q^{-1} \\ \frac{17}{50} - \frac{2}{25}q^{-1} & \frac{1}{10} - \frac{11}{50}q^{-1} & 1 - \frac{107}{100}q^{-1} \end{bmatrix}, \\ \underline{C}(q^{-1}) &= \begin{bmatrix} 1 - \frac{1}{100}q^{-1} & 2 - \frac{1}{50}q^{-1} \\ 3 - \frac{1}{10}q^{-1} & 2 - \frac{1}{10}q^{-1} \end{bmatrix}. \end{aligned}$$

Dla podanego systemu wyznaczamy sterowanie minimalnowariancyjne (3.10), używając prawostronnej T -inwersji wielomianowej z Rozdziału 2.1.2. Zauważmy, że zera sterownicze typu 1 [18] równe $z_1 = 1.1866$, $z_2 = 0.8906$ pociągają za sobą niestabilność sterowania MV. Dla $y_{ref1} = 1$ i $y_{ref2} = 2$ otrzymujemy odpowiednio $var\{y_1(t)\} = 0.019$ oraz $var\{y_2(t)\} = 0.088$. Wykresy sterowań minimalnowariancyjnych $u_1(t)$, $u_2(t)$ i $u_3(t)$ oraz wyjść $y_{ref1}(t)$ i $y_{ref2}(t)$ pokazano na rysunkach 3.1 i 3.2.



Rysunek 3.1: Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 1)

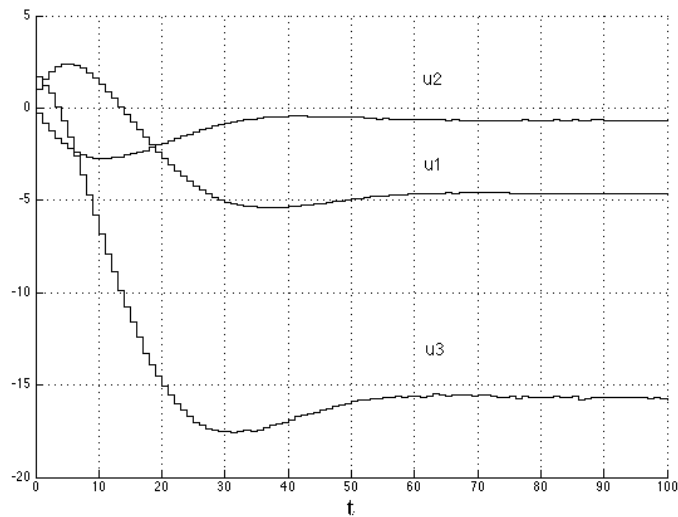


Rysunek 3.2: Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 1)

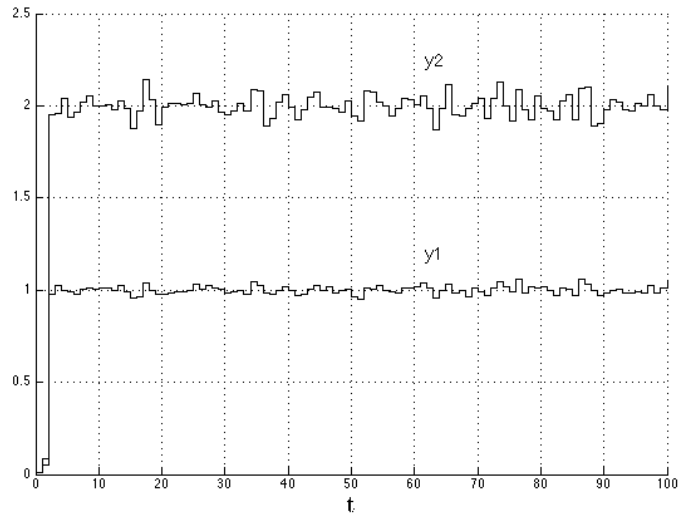
3.2.2 Przykład 2

W celu stabilizacji układu sterowania minimalnowariancyjnego z przykładu 3.2.1 użyto σ -inwersji wielomianowej z parametrycznymi stopniami swobody [34].

Dla $\underline{B}(q^{-1}) = \begin{bmatrix} 0.8 & -1.4 & -3.1 \\ 1 & -1 & -1 \end{bmatrix}$ otrzymano zera sterownicze typu 2 [18] równe $z_{1,2} = 0.9272 \pm 0.0912i$ oraz stabilizację sterowań minimalnowariancyjnych. Sygnały wejść i wyjść pokazano odpowiednio na wykresach 3.3 i 3.4.



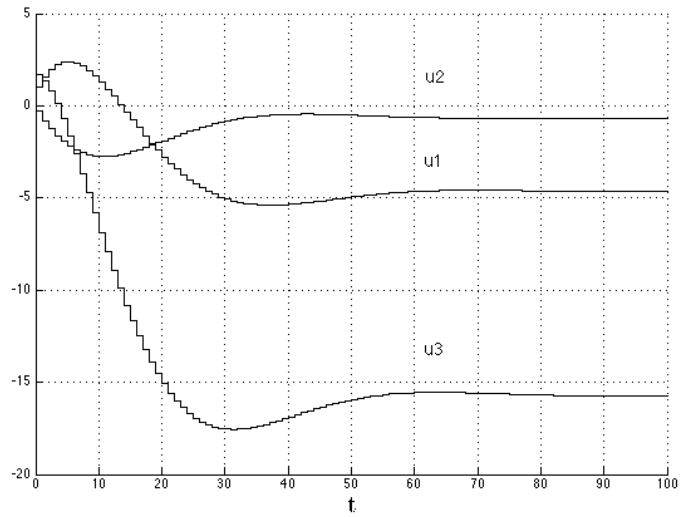
Rysunek 3.3: Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 2)



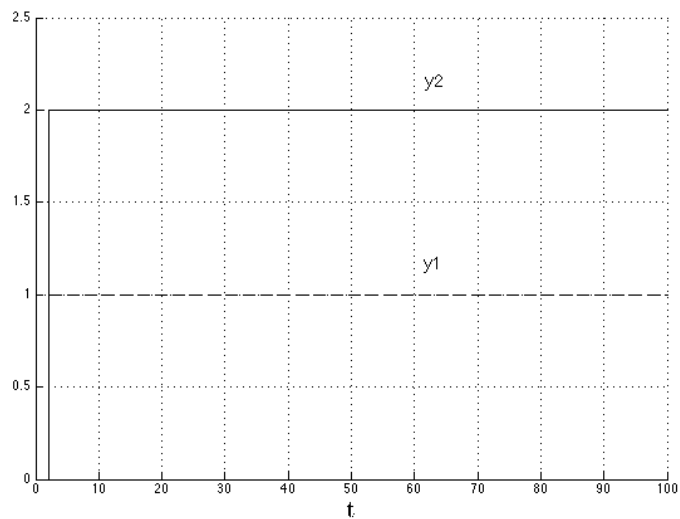
Rysunek 3.4: Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 2)

3.2.3 Przykład 3

Następnie wielowymiarowy układ sterowania minimalnowariancyjnego sprowadzono do układu sterowania perfekcyjnego zakładając $\underline{C}(q^{-1}) = I_n$. Działanie otrzymanego systemu deterministycznego pokazano na wykresach 3.5 i 3.6.



Rysunek 3.5: Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 3)



Rysunek 3.6: Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 3)

Znamiennym jest, że badania symulacyjne układów sterowania perfekcyjnego-

/minimalnowariancyjnego z przykładów 3.2.1-3.2.3 wykonano z wykorzystaniem autorskich procedur implementujących wielowymiarowe równania różnicowe, natomiast weryfikację otrzymanych rezultatów przeprowadzono w przyborniku Simulink pakietu Matlab. Schemat układu regulacji stworzony przy pomocy przybornika Simulink pokazano w Załączniku A (patrz rysunek 7.22). Walidację wprowadzonych równań różnicowych przeprowadzono w oparciu o przybornik Polynomial Toolbox. Pozwoliło to nie tylko na rozwiązanie skomplikowanych równań diofantycznych, ale na znacznie przyspieszenie czasu obliczeń. W ten sposób powstało uniwersalne narzędzie, pozwalające na szybkie projektowanie układów sterowania perfekcyjnego/minimalnowariancyjnego obiektami dyskretnymi o różnej liczbie wejść i wyjść. Omawianą autorską procedurę pokazano w Załączniku B (tutaj parametry początkowe są zgodne z przykładami symulacyjnymi 3.2.1). Warto w tym miejscu wspomnieć, że dla przykładów 3.2.1 i 3.2.2 macierze wielomianowe $\underline{F}(q^{-1})$ i $\underline{H}(q^{-1})$ wyniosły

$$\underline{F}(q^{-1}) = \begin{bmatrix} \frac{49}{100}q^{-1} + 1 & \frac{99}{50}q^{-1} + 2 \\ \frac{8}{5}q^{-1} + 3 & \frac{87}{50}q^{-1} + 2 \end{bmatrix}, \quad (3.14)$$

i

$$\underline{H}(q^{-1}) = \begin{bmatrix} 0.2 & 2 \\ 0.87 & 1.7 \end{bmatrix}. \quad (3.15)$$

Zauważmy, że dobór stopni swobody np. σ -inwersji wielomianowej, przy nieskończonej T -inwersji prawostronnej o minimalnej normie, może potencjalnie destabilizować układ sterowania minimalnowariancyjnego/perfekcyjnego. W sukurs temu przychodzi uogólnione sterowanie minimalnowariancyjne/perfekcyjne *ang. generalized minimum variance control* (GMVC). Stanowiący nadklasę MVC okołominimalnowariancyjny algorytm GMVC wykazuje się większą odpornością w kontekście uzyskiwanej stabilności strukturalnej. W dalszej części dysertacji doktorskiej prawo sterowania GMVC będzie kluczowe, z punktu widzenia odpornego sterowania sieciami komputerowymi z przełączaniem pakietów, co zostanie zweryfikowane przez liczne symulacje. W tej części pracy przedstawiona będzie jedynie reguła GMVC dedykowana obiektom stacjonarnym. Wraz z implementacją

mechanizmu adaptacji powstanie skuteczne narzędzie z pogranicza współczesnej automatyki i informatyki - zostanie ono omówione w kolejnych rozdziałach dysertacji.

3.3 Uogólnione sterowanie minimalnowariancyjne [18]

Rozważmy obiekt LTI MIMO opisany modelem ARMAX (3.8). Minimalizując wskaźnik jakości

$$\min_{u(t)} E \{ \| y(t+d) - y_{ref}(t+d) \|_P^2 + \| u(t) \|_Q^2 \}, \quad (3.16)$$

gdzie arbitralnie dobierane $P(q^{-1})$ i $Q(q^{-1})$ są wielomianami macierzowymi, otrzymujemy prawo sterowania GMVC

$$u(t) = \{ Q(q^{-1}) + \underline{b}_0^T P(q^{-1}) \tilde{F}(q^{-1}) \underline{B}(q^{-1}) \}^{-1} \underline{b}_0^T P(q^{-1}) [\tilde{C}(q^{-1}) y_{ref}(t+d) - \tilde{H}y(t)], \quad (3.17)$$

z macierzami $\tilde{F}(q^{-1})$ i $\tilde{H}(q^{-1})$ wyznaczanymi w oparciu o równanie (3.11) i \underline{b}_0 stanowiącym współczynnik wiodący macierzy wielomianowej $\underline{B}(q^{-1})$.

Uwaga 3.3.1. *Algorytm GMVC sprowadza się do algorytmu MVC w przypadku $P(q^{-1}) = I_{n_y}$.*

Uwaga 3.3.2. *W celu eliminacji błędu w stanie ustalonym wprowadza się operator Δu w równaniach (3.8), (3.16), (3.17). Natomiast równania (3.8) i (3.11) należy rozpatrywać w kontekście $\underline{A}'(q^{-1}) = \underline{A}(q^{-1})(1 - q^{-1})$ podstawionej za $\underline{A}(q^{-1})$.*

Uwaga 3.3.3. *Zmodyfikowany algorytm GMVC znaleźć można w pozycji [18].*

Znając już istotę sterowania okołominimalnowariancyjnego dedykowanego obiektom stacjonarnym, przejdźmy do kluczowego, z punktu widzenia niniejszej dysertacji doktorskiej, problemu regulacji dla obiektów niestacjonarnych. Rozważania nasze zaczniemy od wyjaśnienia podstawowych pojęć spotykanych w podnoszonej tematyce.

Rozdział 4

Istota sterowania adaptacyjnego

Algorytmy adaptacyjne pracują w oparciu o istniejący algorytm sterowania i poprzez dodatkowe informacje korygują pracę algorytmu bazowego. Rozwiązanie oparte na adaptacji służy przede wszystkim lepszemu dopasowaniu wyznaczonego algorytmu sterowania do obiektu, na którym algorytm pracuje. Decydując się na użycie algorytmów regulacji adaptacyjnej, chcemy osiągnąć stabilną pracę układu oraz optymalne charakterystyki tłumienia zakłóceń, jak i nadążania wielkości regulowanej za wielkością zadaną. Układy z ujemnym sprzężeniem zwrotnym charakteryzują się samoistną odpornością na błędy wynikające z doboru parametrów modelu. Poprawa i redukcja błędów w układzie jest ograniczona, jeśli bazować jedynie na mechanizmie ujemnego sprzężenia zwrotnego. Przy dużych wahaaniach charakterystyk, badany obiekt może nie spełnić kryteriów stabilności. Przy badaniu sieci komputerowych, które ze swej natury są obiektami bardzo często i silnie zmieniającymi się w kontekście właściwości dynamicznych, naturalnym wydaje się zastosowanie dodatkowego mechanizmu dostrajania regulatora, jakim jest mechanizm adaptacji. Sterowanie adaptacyjne wykorzystuje się w momencie, gdy nie posiadamy pełnych informacji o badanym obiekcie. W takich sytuacjach działanie regulatora o stałych parametrach może być niewystarczające do spełnienia zadania sterowania. Aby uzyskać więcej informacji o obiekcie stosuje się identyfikację strukturalną [65, 66]. Przeprowadzając identyfikację parametryczną w czasie rzeczywistym dla regulatora, o zadanej strukturze, jesteśmy w stanie realizować proces adaptacji [67]. Oczywiście skutkuje to odpowiednim doбором

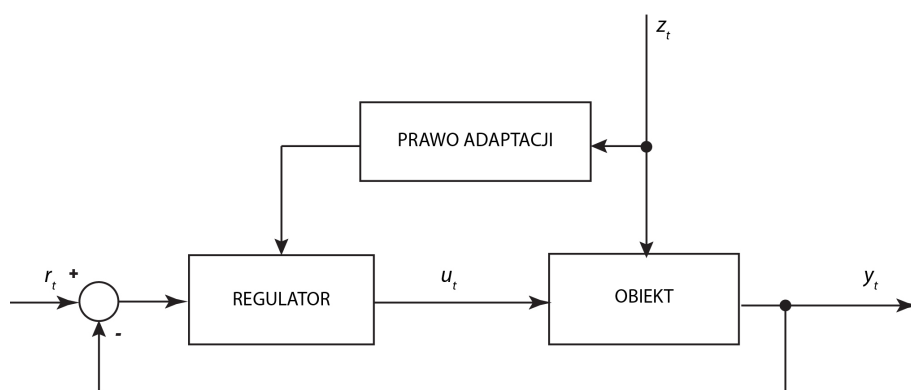
parametrów arbitralnie dobranego modelu, np. w modelu ARMAX (3.8) procedurze 'strojenia' ulec mogą składowe macierzy wielomianowych $\underline{A}(q^{-1})$, $\underline{B}(q^{-1})$ oraz $\underline{C}(q^{-1})$.

Podążając za [65] oraz [67], można przyjąć różne struktury adaptacyjnych układów regulacji. W kolejnym rozdziale przedstawiono te najbardziej reprezentatywne.

4.1 Struktury sterowania adaptacyjnego

- sterowanie ze sprzężeniem do przodu (*feedforward control*)

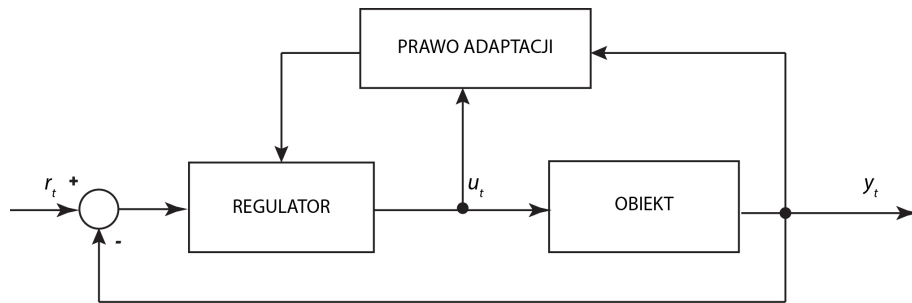
Parametry regulatora przestrajane są na podstawie zbudowanej tablicy przeglądowej, charakteryzującej wpływ mierzalnego sygnału z_t na zachowanie się obiektu, którą pokazano na rysunku (4.7).



Rysunek 4.7: Schemat blokowy układu ze sprzężeniem do przodu

- sterowanie ze sprzężeniem zwrotnym (*feedback control*)

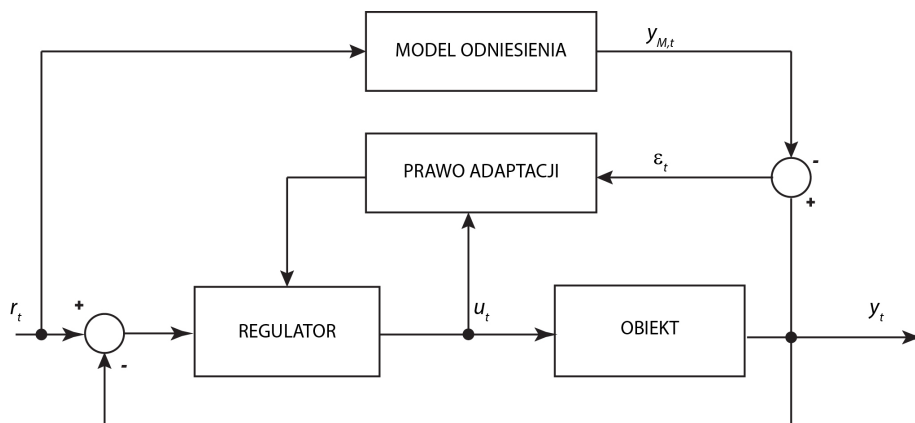
Parametry regulatora zmieniane są dzięki istnieniu zewnętrznej pętli sprzężenia zwrotnego, pozwalając poprzez identyfikację reagować, w ogólnym przypadku, na zmiany właściwości obiektu, zwiększając złożoność układu (4.8).



Rysunek 4.8: Schemat blokowy układu ze sprzężeniem zwrotnym

- sterowanie z modelem odniesienia (*Model Reference Adaptive Controller - MRAC*)

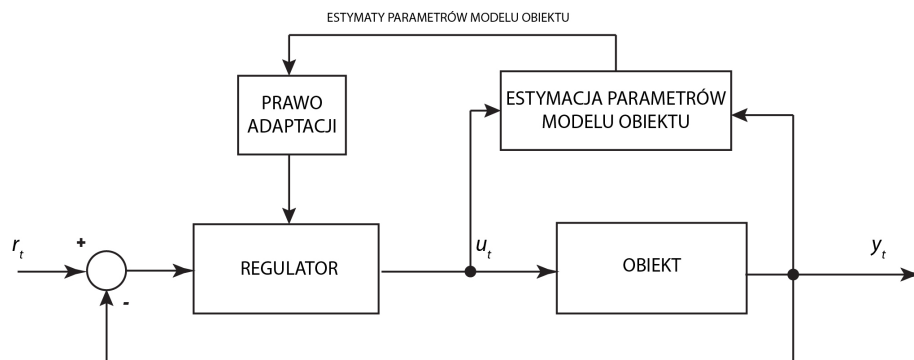
Na rysunku (4.9) pokazano model odniesienia przez porównanie. Sygnałami porównywanymi są odpowiednio sygnał wejściowy $y_{M,t}$ z modelu odniesienia oraz sygnał wyjściowym y_t obiektu. Dzięki takiemu rozwiązaniu możliwe jest przestrojenie regulatora tak, aby układ zamknięty miał właściwości zbliżone do modelu odniesienia.



Rysunek 4.9: Schemat blokowy układu z modelem odniesienia

- sterowanie z estymacją parametrów modelu (*Model Identification Adaptive Controller - MIAC*)

Rysunek (4.10) przedstawia w jaki sposób informacja o obiekcie uzyskana w procesie identyfikacji skutkuje wyznaczeniem modelu charakteryzującego się parametrami wyznaczonymi w procesie estymacji. Wyestymowane parametry służą w kolejnym etapie do syntezy regulatora.



Rysunek 4.10: Schemat blokowy układu z estymacją parametrów modelu

Zagadnienie identyfikacji nierozzerwalnie łączy się z eksperymentalnym wyznaczaniem parametrów badanego modelu. Podczas procesu identyfikacji dąży się do minimalizacji różnicy pomiędzy obiektem, a modelem, który go reprezentuje. Reasumując, zagadnienie identyfikacji jest bardzo istotnym elementem sterowania adaptacyjnego z punktu widzenia poprawy parametrów układów regulacji wielowymiarowej. Zanim jednak przejdziemy do przedstawienia wybranych w ramach dysertacji doktorskiej metod identyfikacji, omówmy kluczowy paradygmat modelowania zdarzeń sieci komputerowych z przełączaniem pakietów. Jak zaraz zostanie pokazane, wspomniane rozwiązanie oparto na modelu ARMAX i przytoczonych algorytmach regulacji okołominimalnowariancyjnych. Wraz z warstwą mechanizmu identyfikacji parametrycznej powstanie kompleksowe narzędzie dedykowane projektowaniu odpornych sterowań wielowymiarowymi sieciami kolejkowymi.

4.2 Model komputerowych sieci kolejkowych z przełączaniem pakietów

Podążając za postulatami zamieszczonymi w Rozdziale 1.2. niniejszej dysertacji doktorskiej, w tym podejściem do optymalnego przełączania pakietów do modelowania kolejkowych sieci komputerowych, wykorzystano model ARMAX. Koncepcję oparto na metodologii podanej w pracy [13]. Jednak w omawianej pozycji ograniczono się jedynie do modelowania sieci o jednym wyjściu i jednym wejściu, tj. sieci opisanych strukturą SISO. Zaproponowany niekwadratowy model MIMO posłuży implementacji wspomnianych algorytmów adaptacyjnej regulacji okołominimalnowariancyjnej. Należy w tym miejscu silnie zaznaczyć, że wprowadzone inwersje niekwadratowych macierzy wielomianowych z powodzeniem przyjdzie w sukurs projektowaniu odpornych sterowań zarządzania sieciami kolejkowymi. Poprzez optymalizację np. wskaźników jakości 3.9 bezpośrednio wpływać można na postać sterowania minimalnowariancyjnego/perfekcyjnego 3.10, tj. kształtować jego charakterystyki czasowo-częstotliwościowe. W pracy doktorskiej ograniczono się wyłącznie do charakterystyk czasowych z implementacją wspomnianej T -inwersji prawostronnej, tj. inwersji skojarzonej z wielowymiarowymi macierzami prawostronnymi. Zaproponowano, że wyznaczony w oparciu o kryterium wektor sterowania $u(t)$ reprezentowany jest przez znaną z literatury przedmiotu funkcję Mark/Drop Probability [13]. Wskutek zastosowania niekwadratowych inwersji macierzy wielomianowych w sposób istotny kreować można przebiegi czasowe sterowań w systemach kolejkowych, tj. zarządzać różnymi grupami użytkowników. Paradygmat wzięty z teorii sterowania, gdzie poprzez dobór optymalnych stopni swobody można wpływać na odporność/energetyczność sterowań okołominimalnowariancyjnych i tutaj znaleźć można swoje ugruntowanie [17, 33, 34, 35, 36, 47, 52, 56, 63, 68, 69]. Jednak w pracy doktorskiej ograniczono się jedynie do jednoznacznej T -inwersji wielomianowej, która dla modeli parametrycznych gwarantuje minimalną energię sterowań, tj. optymalny 'odrzut' zgłoszeń grup użytkowników. Po uczynieniu krótkich wyjaśnień dotyczących głów-

nych założeń dysertacji doktorskiej, użyjmy model ARMAX pierwszego rzędu 3.8 z następującej postaci [13, 26]

$$\underline{A}(q^{-1})\underline{q}(t) = q^{-d}\underline{B}(q^{-1})p(t) + \underline{C}(q^{-1})e(t), \quad (4.18)$$

gdzie $\underline{A}(q^{-1}) = I_{n_y} + \underline{a}_1q^{-1}$, $\underline{B}^{n_y \times n_u}(q^{-1}) = \underline{b}_0$, $\underline{C}(q^{-1}) = I_{n_y} + \underline{c}_1q^{-1}$, $d = 1$ i $\underline{q}(t)$ jest wektorem opisującym długość kolejki, natomiast wektor $p(t)$ implikuje funkcję Drop/Mark Probability [13]. Znamiennym jest, że wszystkie kanały typu wyjście-wejście mają te same opóźnienie d . Mając już model wielowymiarowej sieci komputerowej z przełączaniem pakietów przejdźmy w następnym rozdziale do omówienia kluczowych z punktu realizacji pracy doktorskiej mechanizmów identyfikacji parametrycznej. Spośród wielu narzędzi dedykowanych estymacji parametrów modeli [70], wybrano podejście efektywnie prezentowane w pozycji [13], czyli zmodyfikowanego algorytmu RLS Kaczmarza [67, 71, 72, 73].

4.3 Modelowanie zdarzeń w sieciach komputerowych

Zagadnienie modelowania i przewidywania zdarzeń jest jednym z fundamentalnych elementów problematyki symulacji sieci. Poprawny dobór i implementacja modelu determinują poprawność działania mechanizmów sieciowych. Strumień zdarzeń może być modelowany przy wykorzystaniu przyrostowej ich liczby lub zapisanych ciągów czasowych pomiędzy sąsiadującymi zdarzeniami. Literatura, ukierunkowana na narzędzia informatyczne, dostarcza szeroką gamę rozwiązań modelowania strumienia zdarzeń. Klasyczne podejście wykorzystuje rozwiązanie bazujące na modelu Poissona [74], jednakże przytaczana metoda nie modeluje wewnętrznej korelacji zdarzeń w strumieniach, co jest obserwowalne przy analizie funkcjonowania rzeczywistych sieci. Wykładniczy rozwój infrastruktury i przepustowości determinuje jeszcze większą wewnętrzną zależność zdarzeń. Konsekwencją obserwowanych przyczynowości był rozwój modeli ruchu opartych o samopodobieństwo [75], gdzie głównym reprezentantem jest rozkład Pareto [76, 77].

Znajduje on zastosowanie w bardzo szerokim spektrum zagadnień sieciowych, między innymi w modelowaniu i kontroli zakleszczeń, prognozowaniu rozmiarów przesyłanych plików oraz modelowaniu ruchu webowego w serwisach WWW [78, 79, 80, 81, 82]. W celu zaakcentowania potencjalnych zbieżności rozkładu Pareto z wykorzystanym w ramach niniejszej pracy modelem matematycznym, poniżej przytoczone zostaną podstawowe parametry omawianego rozkładu.

Niech zmienna losowa X ma rozkład Pareto $Pa(x_m, \alpha)$ z parametrami $x_m > 0$ i $\alpha > 0$

$$f(x) = \frac{\alpha x_m^\alpha}{x^{\alpha+1}} \quad \text{dla } x \in \langle x_m, \infty \rangle. \quad (4.19)$$

Wtedy dystrybuanta, wariancja, wartość oczekiwana wyrażone są odpowiednio następującymi formułami:

$$f(x) = 1 - \left(\frac{x_m}{x}\right)^\alpha, \quad (4.20)$$

$$f(x) = \frac{x_m^2 \alpha}{(\alpha-1)^2 (\alpha-2)} \quad \text{dla } \alpha > 2, \quad (4.21)$$

$$f(x) = \frac{\alpha x_m}{\alpha-1} \quad \text{dla } \alpha > 1. \quad (4.22)$$

Znamiennym jest, że klasyczne podejście do modelowania dynamiki sieci komputerowych z przełączaniem pakietów angażuje powyższy rozkład Pareto. Zdaniem autora pracy, omawiany rozkład utożsamić można z przebiegiem sterowania GMVC, szczególnie podczas działania akcji całkującej. Tę intrygującą zależność, nieobjętą ramami niniejszej dysertacji doktorskiej, należy traktować jako kluczowy problem otwarty. Przy doborze odpowiednich parametrów estymacji podejścia/mechanizmy mogą być zbieżne.

Mając już wielowymiarowy model sieci komputerowych z przełączaniem pakietów, jak w równaniu 4.18, gdzie funkcja Drop/Mark Probability $p(t)$ odgrywa istotne znaczenie, przejdźmy do krótkiego omówienia wspomnianego algorytmu identyfikacji opartego na mechanizmie *Recursive Least Squares* (RLS). W celu

przyśpieszenia obliczeń, w pracy wykorzystano jego uproszczoną wersję w postaci zmodyfikowanej przez Kaczmarza. Znany z angielskiej literatury przedmiotu tzw. *normalized projection algorithm* [26, 71] jest następującej postaci

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\gamma\varphi(t)}{\alpha + \varphi^T(t)\varphi(t)}(\underline{q}(t) - \varphi^T(t)\hat{\theta}(t-1)), \quad (4.23)$$

gdzie $\hat{\theta}(t)$ jest estymatorem wektora parametrów modelu $\theta(t) = \begin{bmatrix} a & b & c \end{bmatrix}^T$, zaś $\varphi(t) = \begin{bmatrix} -\underline{q}(t) & p(t-1) & \varepsilon(t-1) \end{bmatrix}^T$ i $\varepsilon(t) = \underline{q}(t) - \varphi^T(t-1)\theta(t-1)$. Natomiast $\gamma \in (0, 2)$ i $\alpha \geq 0$ są odpowiednio przyjętymi stałymi, przy czym $\alpha > 0$ w przypadku $\varphi^T(t)\varphi(t) = 0$.

Po kompletnym omówieniu narzędzi używanych we współczesnej problematyce teorii sterowania i identyfikacji, przejdźmy do ich implementacji zgodnie z założeniami przedłożonymi w ramach dysertacji doktorskiej. Rozważania nasze przeprowadzimy najpierw dla jednowymiarowych obiektów sieci komputerowych z przełączaniem pakietów z $n_y = n_u = 1$ opisanych zmodyfikowanym modelem

$$\underline{A}(q^{-1})\underline{q}(t) = q^{-d}\underline{B}(q^{-1})p(t) + \underline{C}(q^{-1})e(t), \quad (4.24)$$

z $\underline{A}(q^{-1}) = 1 + \underline{a}_1q^{-1}$, $\underline{B}^{1 \times 1}(q^{-1}) = \underline{b}_0$, $\underline{C}(q^{-1}) = 1 + \underline{c}_1q^{-1}$, $d = 1$, gdzie wektory $\underline{q}(t)$ i $\underline{p}(t)$ są jednowymiarowymi wektorami opisującymi odpowiednio wyjście i wejście obiektu/sieci komputerowej z przełączaniem pakietów.

W kolejnym Rozdziale 5 przeprowadzono kompleksowe badania symulacyjne dedykowane obiektom SISO, uwzględniając wszystkie dotychczas przedstawione rozwiązania korespondujące z tezą i głównymi założeniami pracy doktorskiej. Po szczegółowym omówieniu otrzymanych rezultatów dalsze kroki zostaną skoncentrowane na sterowaniu adaptacyjnym sieciami komputerowymi MIMO.

Rozdział 5

Sterowanie adaptacyjne sieci komputerowych SISO z wykorzystaniem Drop/Mark Probability Function

Nasze rozważania zacznijmy od adaptacyjnej wersji podanego w Rozdziale 3.1 algorytmu MVC dedykowanego jednowymiarowym obiektom stacjonarnym opisanym czasem dyskretnym 4.24. Jak wcześniej wspomniano, użycie modeli nie-stacjonarnych pozwala na elastyczny opis zjawisk zachodzących w sieciach komputerowych manifestowanych między innymi zmianą liczby lub charakteru użytkowników. Poniżej przedstawiono autorski algorytm *Discrete-Time Minimum Variance Adaptive Control* (DMVAC).

5.1 Dyskretne minimalnowariancyjne sterowanie adaptacyjne w pętli zamkniętej

Jednowymiarowe dyskretne sterowania adaptacyjne MVC uzyskuje się w oparciu o model kolejki SISO jak w równaniu 4.24. Nawiązując do problematyki projektowania sterowania MVC dla obiektów stacjonarnych przedstawionych w Roz-

dziale 3.1 oraz treści związanych z identyfikacją parametryczną (patrz równanie 4.23), wspomniane sterowanie adaptacyjne uzyskuje się minimalizując wskaźnik jakości dla obiektów MIMO sieci kolejkowych

$$\min_{p(t)} E \left\{ [\underline{q}(t+d) - q_{ref}(t+d)]^T [\underline{q}(t+d) - q_{ref}(t+d)] \right\}, \quad (5.25)$$

gdzie $\underline{q}(t+d) = \underline{\tilde{C}}^{-1}(q^{-1})[\underline{\tilde{F}}(q^{-1})\underline{B}(q^{-1})p(t) + \underline{\tilde{H}}(q^{-1})\underline{q}(t)] + \underline{F}(q^{-1})e(t)$ oraz $q_{ref}(t+d)$ są odpowiednio stochastycznym predyktorem wyjścia wielowymiarowej sieci komputerowej i wartości referencyjnej/zadanej długości kolejki, zaś $p(t)$ oznacza funkcję opisującą prawdopodobieństwo odrzutu pakietów (funkcja Drop/Mark Probability). Znamiennym jest, że prezentowane powyżej uniwersalne podejście obejmuje modele z czasem opóźnienia d . W ramach niniejszej pracy doktorskiej przyjęto $d = 1$, co w znaczny sposób upraszcza prowadzone w jej ramach obliczenia. Jednakże podejście naturalnie obejmuje obiekty z $d > 1$.

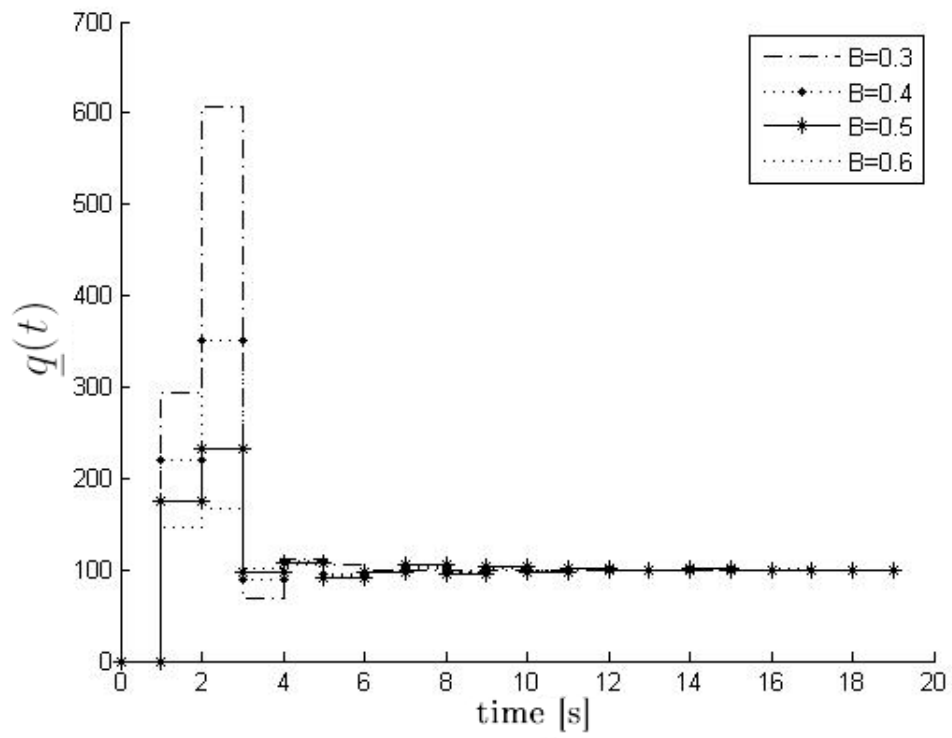
Biorąc pod uwagę wszystkie powyższe założenia, funkcja $p(t)$ dla obiektów SISO, jest następującej postaci [26]

$$p(t) = \frac{1}{\underline{b}_0} [(1 + \underline{c}_1 q^{-1})q_{ref}(t+1) - (\underline{c}_1 - \underline{a}_1)\underline{q}(t)]. \quad (5.26)$$

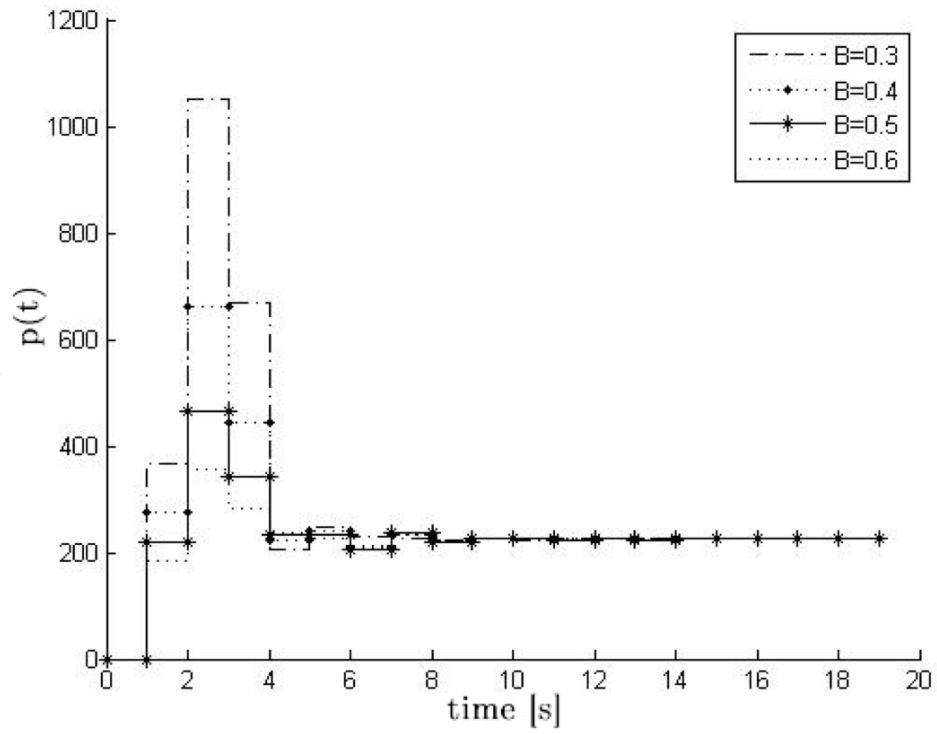
Teraz równanie 5.26 wraz z podejściem oferowanym przez Kaczmarza (patrz formuła 4.23) tworzą kompleksowe rozwiązanie problemu sterowania adaptacyjnego MVC w zadaniach optymalnego odrzutu pakietów w kolejkowych sieciach komputerowych. W celu weryfikacji przytoczonych rozważań analitycznych poniżej zostały zaprezentowane przykłady symulacyjne przeprowadzone w środowisku Matlab. Należy w tym miejscu zaznaczyć, że z powodu drobnych komplikacji wynikających z zastosowania zmodyfikowanej metody RLS, podczas symulacji oparto się na autorskich równaniach różnicowych opisujących dynamikę całego układu regulacji automatycznej.

5.1.1 Przykład symulacyjny 1 [26]

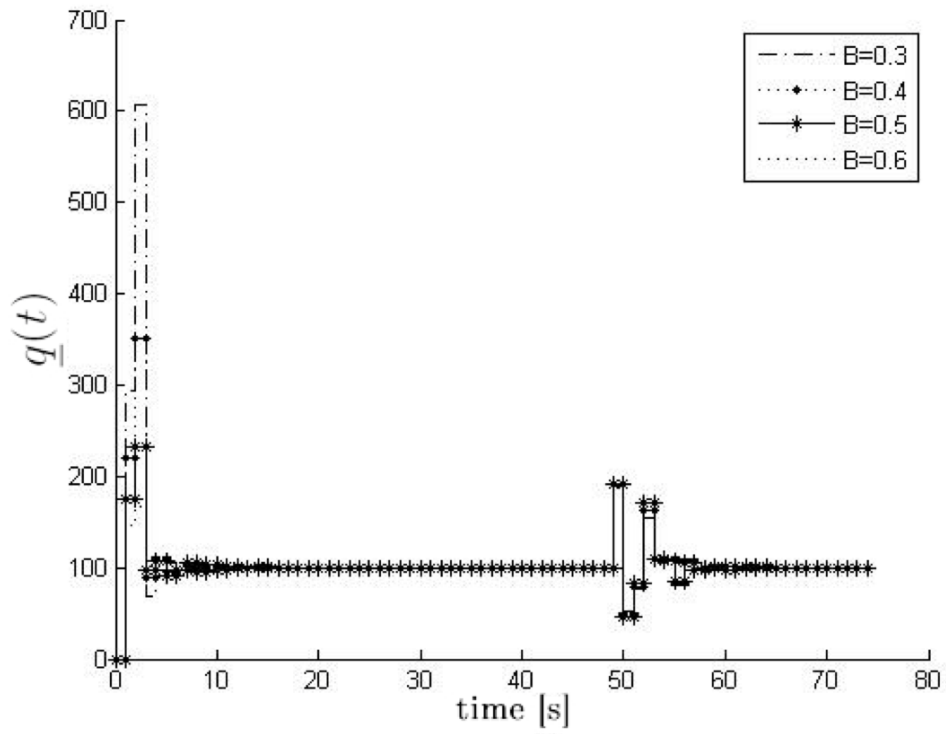
Rozważmy jednowymiarowy obiekt kolejkowej sieci komputerowej opisany modelem (4.24) z $\underline{A}(q^{-1}) = 1 + 0.8q^{-1}$, $\underline{C}(q^{-1}) = 1 + 0.1q^{-1}$ oraz zmiennym $\underline{B}(q^{-1}) = \underline{b}_0$. Załóżmy, że zmieniający się współczynnik \underline{b}_0 odpowiada różnym obciążeniom generowanym przez grupy użytkowników sieci komputerowej. W odpowiedzi na zapotrzebowanie dobierana jest taka funkcja $p(t)$ optymalizująca wskaźnik jakości 5.25 w ujęciu jednowymiarowym, przy zachowaniu ograniczeń infrastrukturalnych. Podczas deterministycznego procesu sterowania DMVAC ($e(t) = 0$) niestacjonarną kolejkową siecią komputerową przyjęto, że w 50 sekundzie symulacji następuje skokowa zmiana obciążenia sieci wskutek zmiennej wartości parametru \underline{b}_0 . Na rysunkach 5.11 i 5.12 przedstawiono wyniki symulacji dla stacjonarnego działania kolejkowej sieci komputerowej, natomiast na rysunkach 5.13 i 5.14 przy zmiennym jej obciążeniu. W obydwu przypadkach otrzymano statyczny błąd $\underline{q}_{ss} = 0$ dla $q_{ref}(t) = 100$. Oczywiście prezentowany tutaj wykres zmian funkcji $p(t)$ odpowiada czasowemu zakłóceniu na zasoby sieci (patrz rysunek 5.14).



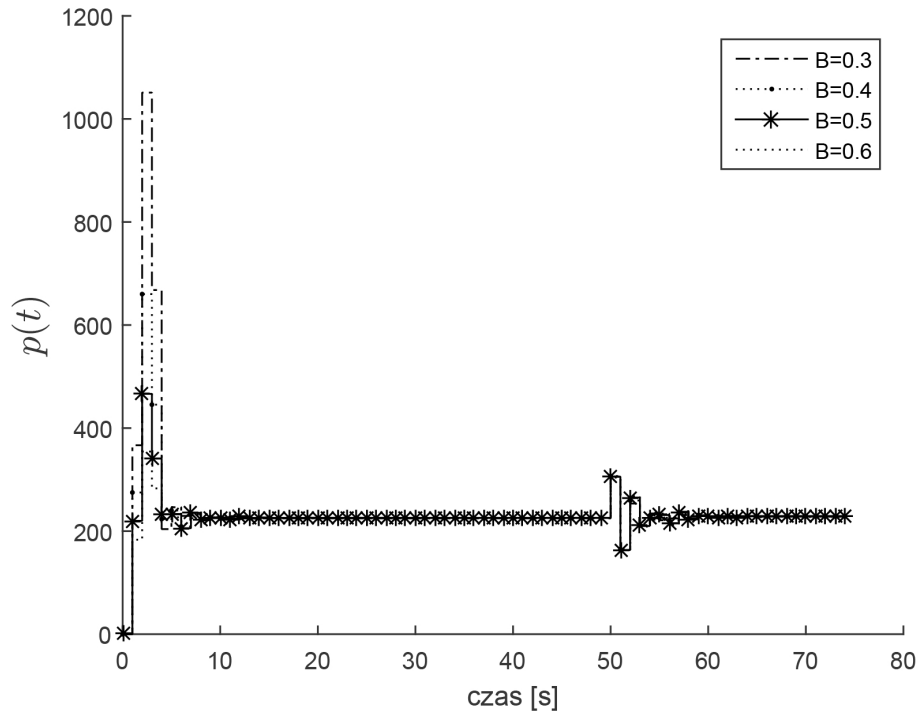
Rysunek 5.11: DMVAC: Wykres zmian wyjścia sieci komputerowej $\bar{q}(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny



Rysunek 5.12: DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny



Rysunek 5.13: DMVAC: Wykres zmian wyjścia sieci komputerowej $q(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny



Rysunek 5.14: DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny

Po krótkiej, ale treściwej, weryfikacji symulacyjnej otrzymanych rezultatów analitycznych, przejdźmy do zastosowania bardziej odpornego niż sterowanie MVC uogólnionego sterowania minimalnowariancyjnego, zwanego *Generalized Minimum Variance Control* (GMVC). W celu pokazania problematyki działania uogólnionego sterowania w wielowymiarowych kolejkowych sieciach komputerowych w rozdziale następnym zaprezentowano jego wersję dedykowaną obiektom SISO.

5.1.2 Dyskretne uogólnione sterowanie adaptacyjne w pętli zamkniętej

Podobnie jak w przypadku sterowania MVC jednowymiarowe dyskretne uogólnione sterowanie adaptacyjne MVC uzyskuje się na podstawie nieznacznie zmodyfikowanego modelu kolejki SISO (patrz równanie 4.24). Bazując na proble-

matyce sterowania GMVC umocowanej w Rozdziale 3.3 i treściach dotyczących identyfikacji parametrycznej rzezone sterowanie dla obiektów MIMO uzyskuje się w drodze minimalizacji wskaźnika jakości

$$\min_{p(t)} E\{\|\underline{q}(t+d) - q_{ref}(t+d)\|_P^2 + \|p(t)\|_Q^2\}, \quad (5.27)$$

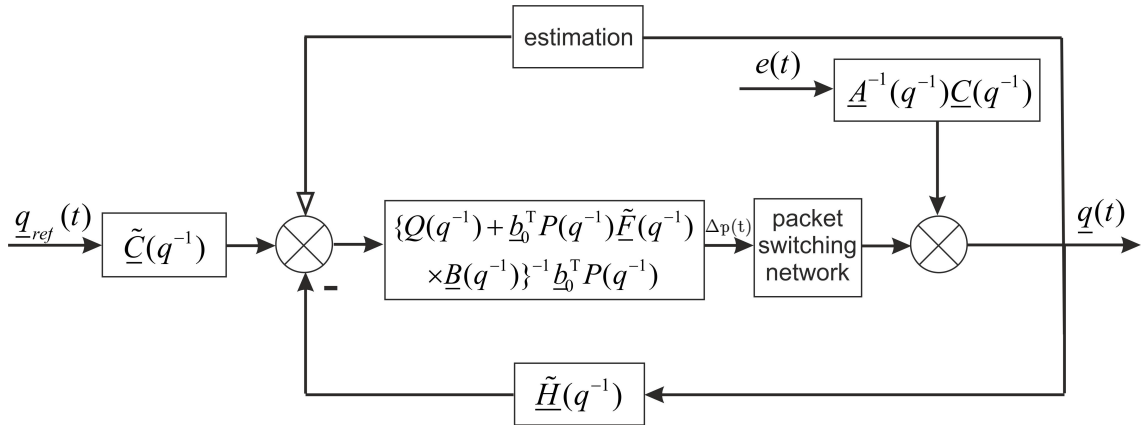
gdzie $P(q^{-1})$ i $Q(q^{-1})$ są arbitralnie dobranymi macierzami parametrycznymi lub wielomianami macierzowymi. Biorąc pod uwagę równanie 5.27 sterowanie *Discrete-Time Generalized Minimum Variance Control* (DGMVC) jest następującej postaci

$$p(t) = \{Q(q^{-1}) + \underline{b}_0^T P(q^{-1}) \tilde{F}(q^{-1}) \underline{B}(q^{-1})\}^{-1} \underline{b}_0^T P(q^{-1}) \\ \times [\tilde{C}(q^{-1}) q_{ref}(t+d) - \tilde{H}y(t)], \quad (5.28)$$

gdzie macierze wielomianowe $\tilde{F}(q^{-1})$ i $\tilde{H}(q^{-1})$ są obliczane, jak w przypadku sterowania DMVAC, z równania diofantycznego 3.11 i faktoryzacji szkieletowej 3.12, zaś \underline{b}_0 jest współczynnikiem wiodącym wielomianu macierzowego $\underline{B}(q^{-1})$. W celu otrzymania prawa sterowania DGMVAC równanie 5.28 należy osadzić w obszarze prezentowanym w Rozdziale 4.3

Uwaga 5.1.1. *Warto w tym miejscu zaznaczyć, że dla $Q(q^{-1}) = 0$ i $P(q^{-1}) = I_{n_y}$ sterowanie DGMVAC sprowadza się do sterowania DMVAC.*

Uwaga 5.1.2. *W celu eliminacji uchybu statycznego wprowadza się całkę uzupełniającą, zgodnie z treścią Rozdziału 3 [18]. Skutkuje to zastąpieniem wyrażenia $p(t)$ wyrażeniem $\Delta p(t)$, zarówno w modelu sieci kolejkowej i wskaźniku jakości, jak również w prawie sterowania 5.28. poglądowy schemat sterowania DGMVAC pokazano na poniższym rysunku.*



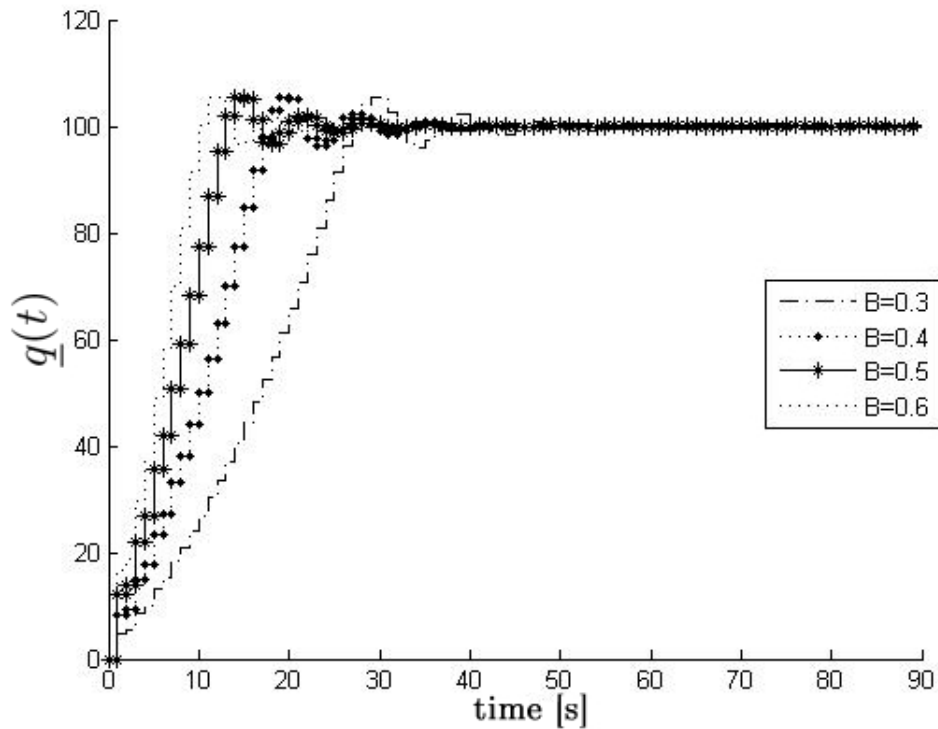
Rysunek 5.15: Schemat sterowania DGMVAC dla sieci komputerowej z dynamiką $G(z)$

Uwaga 5.1.3. Oczywiście, jak poprzednio, funkcja $p(t)$ oznacza zachowanie różnych podsieci/grup użytkowników w kontekście utraty/retransmisji pakietów. Poprzez użycie, w przypadku MIMO, niekwadratowych inwersji różnie możemy kreować odporną już teraz funkcję $p(t)$. Okazało się, że paradygmat bezpośrednio wzięty z tematyki teorii sterowania i systemów można z powodzeniem zaimplementować na potrzeby optymalnej pracy kolejkowych sieci komputerowych z przełączaniem pakietów. Znamiennym jest, że odporny przebieg funkcji $p(t)$ można uzyskać w drodze doboru odpowiednich wielomianów macierzowych $Q(q^{-1})$ i $P(q^{-1})$.

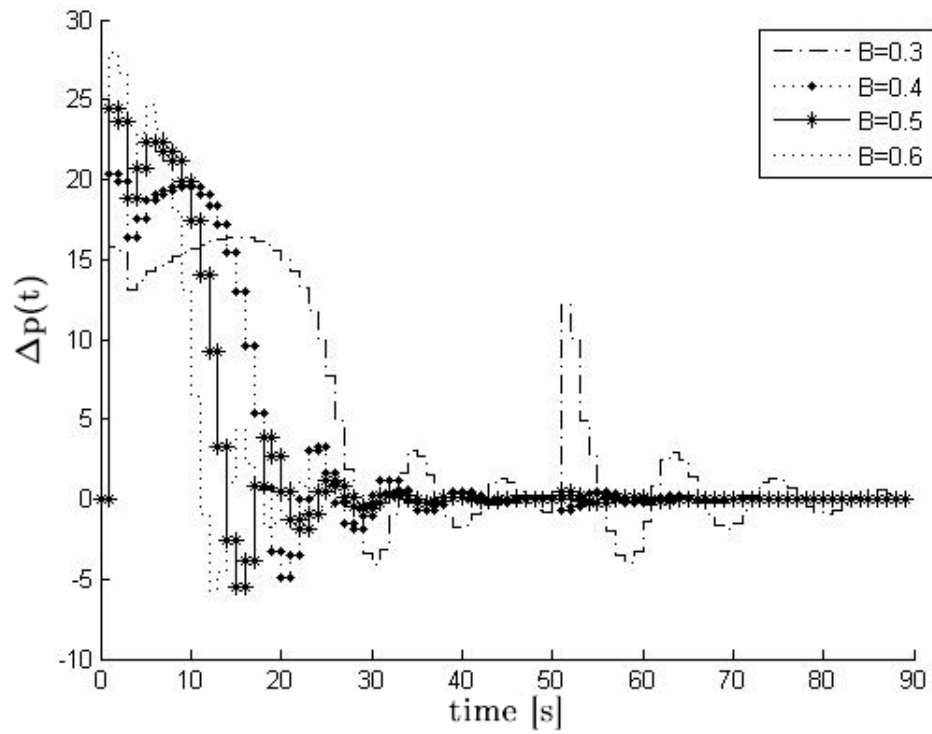
Mając już wszystkie potrzebne narzędzia i mechanizmy, przeprowadźmy weryfikujące badania symulacyjne nad istotą sterowania DGMVAC. Należy jeszcze raz zaznaczyć, że zarówno przykład symulacyjny 1, jak i kolejne przykłady realizowane były w pakiecie Matlab z wykorzystaniem autorskich równań różnicowych. Chociaż w literaturze przedmiotu spotkać można liczne środowiska dedykowane symulacjom sieci komputerowych, to nie implementują one tak oczekiwanego tutaj mechanizmu adaptacji. Klasyczne pakiety OMNeT++, NS2, QualNet, czy Opnet zorientowane na modelowanie zadań sieciowych *Discrete-Time Simulation Systems* (DTSS) nie umożliwiają projektowania sterowania obiektami zmiennymi w czasie.

5.1.3 Przykład symulacyjny 2

Sterowanie DGMVAC przeprowadzono w oparciu o model kolejki z przykładu symulacyjnego 1 dla $Q(q^{-1}) = 0.1$, $P(q^{-1}) = 1$, $\alpha = 0.1$, $\gamma = 1$ i zmiennego $\underline{B}(q^{-1}) = \underline{b}_0$. W tym celu przyjęto $\underline{A}'(q^{-1}) = \underline{A}(q^{-1})(1 - q^{-1})$ i wyznaczono $\underline{\tilde{F}}(q^{-1}) = 1$ oraz $\underline{\tilde{H}}(q^{-1}) = (\underline{c}_1 - \underline{a}_1 + 1) + \underline{a}_1 q^{-1}$. Wykresy 5.16 i 5.17 przedstawiają odpowiednio funkcje $\underline{q}(t)$ i $p(t)$. Zauważyć tu można dłuższy niż w przypadku DMVAC czas osiągnięcia stanu ustalonego ($\underline{q}_{ss} = 0$). Oczywiście spowodowane jest to użyciem akcji całkującej.



Rysunek 5.16: DGMVAC: Wykres zmian wyjścia sieci komputerowej $\underline{q}(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny



Rysunek 5.17: DGMVAC: Wykres zmian funkcji $\Delta p(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny

Uwaga 5.1.4. *Na koniec należy zaznaczyć, że użycie funkcji czasu dyskretnego $\Delta p(t)$ wynika z zastosowania w sterowaniu DGMVC akcji całkującej.*

Rozdział 6

Sterowanie adaptacyjne sieci komputerowych MIMO z wykorzystaniem Drop/Mark Probability Function

Biorąc pod uwagę wszystkie dotychczas otrzymane rezultaty wynikające z przeprowadzonych studiów nad problematyką kolejowania żądań w sieciach komputerowych z przełączaniem pakietów, przejdźmy do implementacji postulowanej nowej metody w zadaniach optymalnej pracy obiektów MIMO. Chociaż przytoczone poniżej wyniki symulacji uzyskano korzystając z modeli z jednym wyjściem, to paradygmat autorskiej metody z powodzeniem można przenieść na obiekty z większą liczbą zmiennych wyjściowych i wejściowych. Naturalnym jest, że zastosowanie obiektów typu MIMO utożsamiać można z rzeczywistą infrastrukturą, gdzie wiele grup użytkowników, wykorzystujących różne zasoby sieciowe, ostatecznie dzierżawi rzeczywistą przepustowość łączy. W celu zastosowania tego istotnego z punktu realizacji dysertacji doktorskiej wątku prześledźmy kilka zaproponowanych poniżej scenariuszy symulacyjnych. Znamienne jest, że tak jak poprzednio, badania oparto na algorytmach sterowania DMVAC z różnymi parametrami technologicznymi. Należy zaznaczyć, że prezentowany mechanizm adaptacji/identyfikacji dedykowany obiektom SISO został w bieżącym rozdziale

rozszerzony do przypadku niekwadratowego MIMO wskutek zastosowania oryginalnych autorskich rozwiązań. Jednym z kluczowych wydaje się być modyfikacja zależności 4.23 do przypadku obejmującego obiekt wielowymiarowy. Poprzez implementację nowych postaci formuł

$$\theta(t) = \left[a \quad b^1 \dots b^{n_u} \quad c \right]^T \quad (6.29)$$

oraz

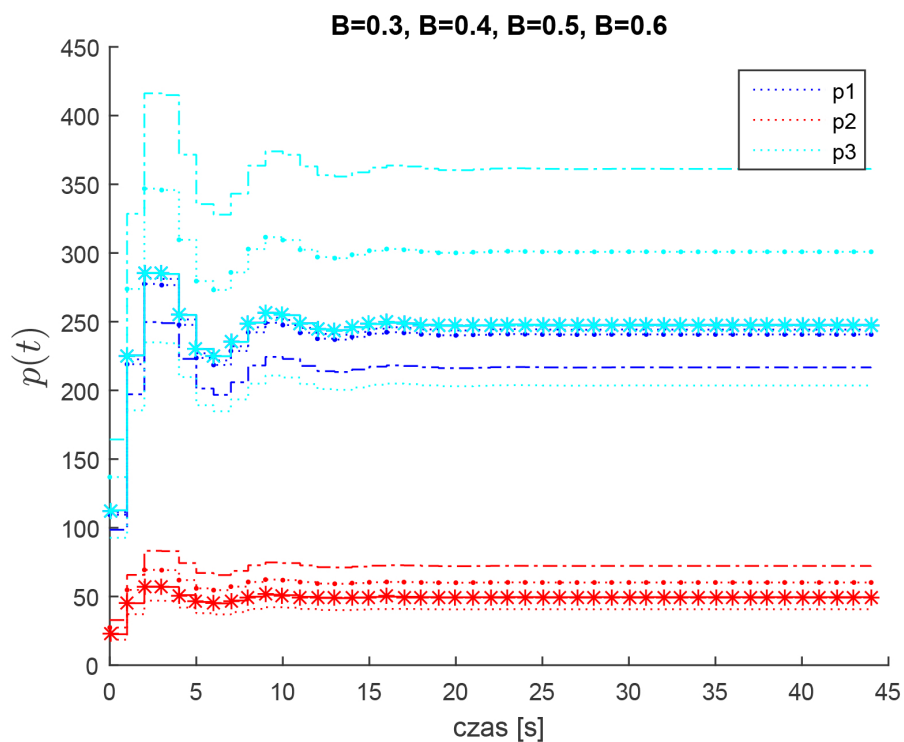
$$\varphi(t) = \left[-\underline{q}(t) \quad p^1(t-1) \dots p^{n_u}(t-1) \quad \varepsilon(t-1) \right]^T, \quad (6.30)$$

w prosty sposób można rozszerzyć proponowane nowe podejście na klasy obiektów o wielu wejściach i wielu wyjściach. Rzecz jasna, większa niż jeden liczba wyjść determinuje użycie n_y -wektora $\underline{q}(t)$, przy $n_u \geq n_y$. Kolejnym kluczowym zagadnieniem jest użycie różnych inwersji macierzy niekwadratowych z Rozdziału 2.1 w prawie sterowania 5.26. Następnym ważnym rozwiązaniem z pewnością jest implementacja autorskich równań różnicowych. Opracowanie wspomnianych równań w istotny sposób usprawniło ich projektowanie niż miałyby to miejsce za pomocą oryginalnych mechanizmów osadzonych w środowisku Matlab/Simulink. Poniżej przedstawiono otrzymane rezultaty w zakresie przeprowadzonych badań dla obiektów wielowymiarowych. I tutaj zachowano nomenklaturę korespondującą z przykładami symulacyjnymi dedykowanym obiektom jednowymiarowym.

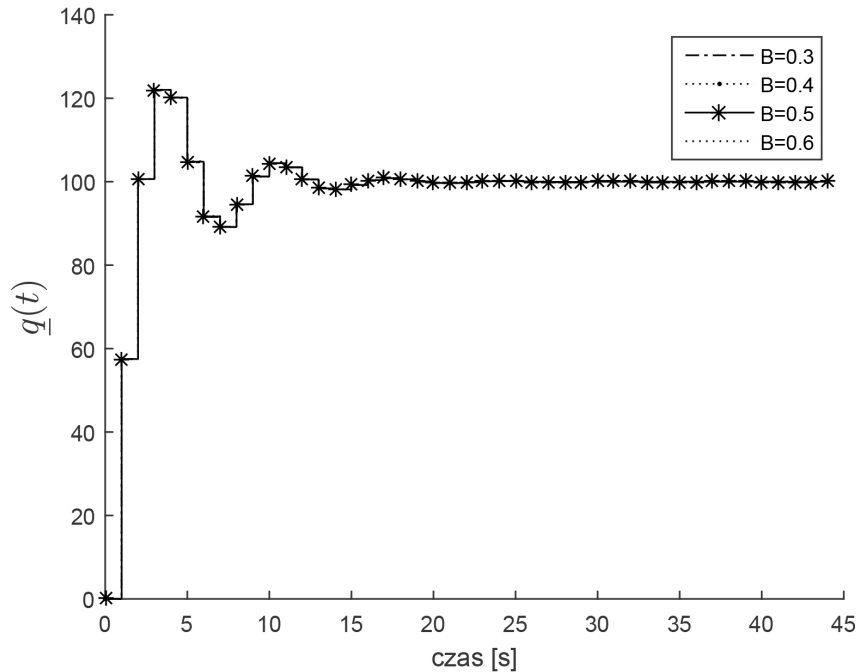
6.1 Przykład symulacyjny 1

Rozważmy obiekt kolejkowej sieci komputerowej z trzema wejściami i jednym wyjściem opisany modelem 4.24 z $\underline{B}^{1 \times 3}(q^{-1}) = [\underline{b}_0^1 \quad \underline{b}_0^2 \quad \underline{b}_0^3]$, $\underline{A}(q^{-1}) = 1 + 0.05q^{-1}$, $\underline{C}(q^{-1}) = 1 + 0.15q^{-1}$. Załóżmy, że podczas symulacji zmianie zostaje poddany każdy z elementów $\underline{B}^{1 \times 3}(q^{-1}) = [\underline{b}_0^1 \quad \underline{b}_0^2 \quad \underline{b}_0^3]$, odpowiednio reprezentujący obciążenie sieci generowanej przez różne grupy użytkowników. Podobnie jak poprzednio, wyznaczana jest funkcja $p(t)$, w oparciu o algorytm DMVAC, tak aby maksymalnie wykorzystać dostępną przepustowość wyrażoną m. in. założoną/preferowaną długością kolejki. Rysunek 6.18 przedstawia wykres funkcji

$p(t) = [p^1(t) \ p^2(t) \ p^3(t)]^T$ dla obiektu stacjonarnego, przy założeniu $q_{ref} = 100$. Rezultaty badań pokazano odpowiednio dla $\underline{B}(q^{-1}) = [0.2 \ 0.1 \ 0.5]$ oraz zmieniającego się parametru b_0^1 z interwałem zmian wynoszącym 0.1. Jako graniczną wartość parametru przyjęto wartość 0.6. Natomiast rysunek 6.19 obrazuje wykresy zmian wyjść sieci komputerowej $\underline{q}(t)$ dla wyznaczonej wcześniej funkcji Drop/Mark Probability $p(t)$. Wykresy $\underline{q}(t)$ przeprowadzone dla $\underline{B}(q^{-1}) = [0.2 \ 0.1 \ 0.5]$ oraz zmieniającego się parametru b_0^1 schematycznie opisano odpowiednio symbolami $B = 0.3, B = 0.4, B = 0.5, B = 0.6$.



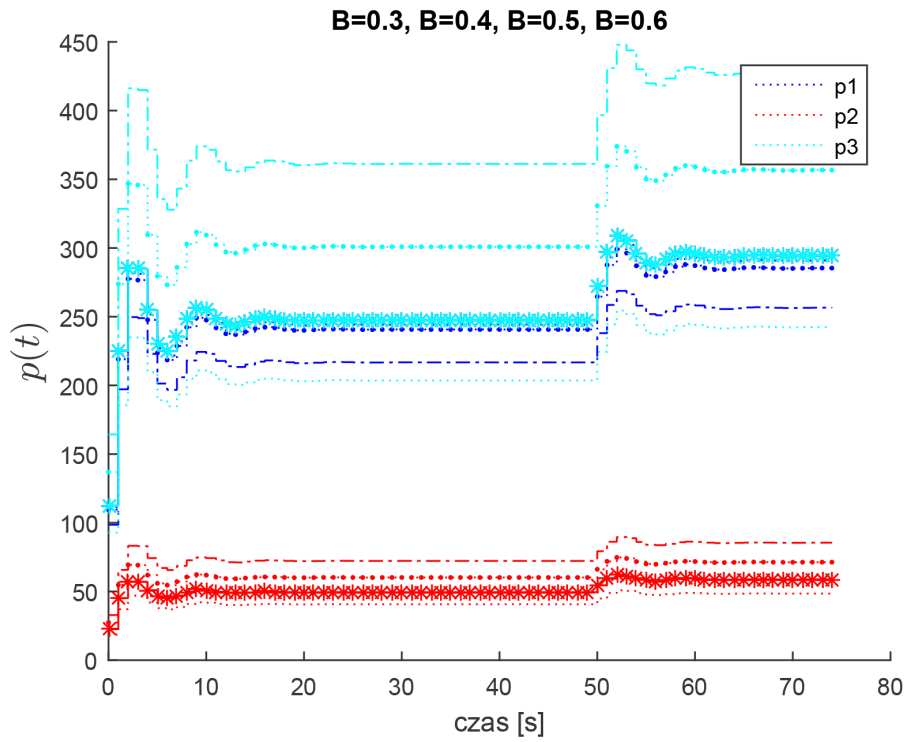
Rysunek 6.18: DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny



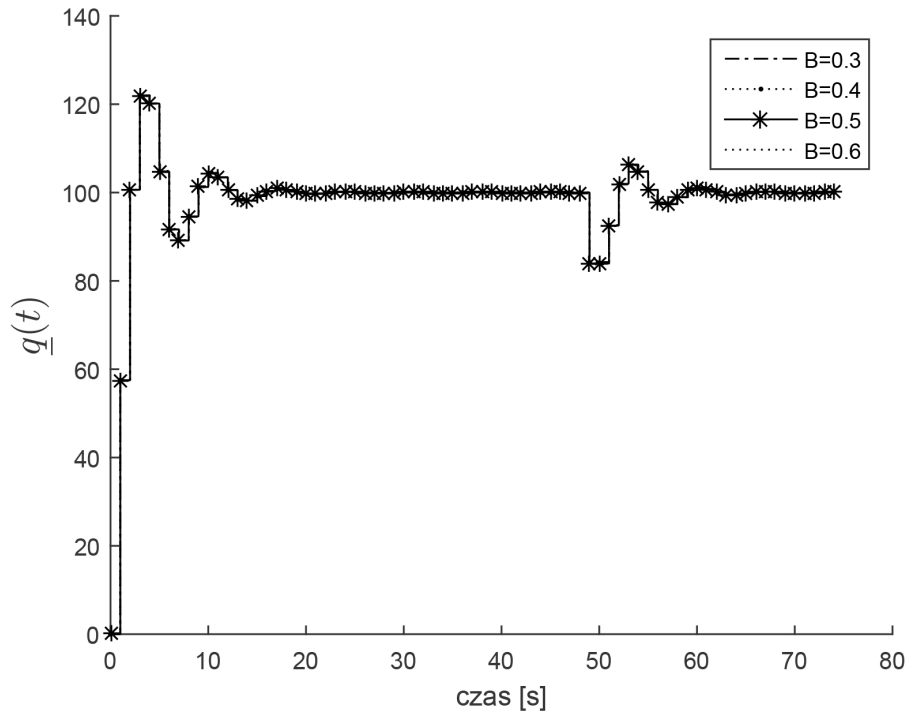
Rysunek 6.19: DMVAC: Wykres zmian wyjścia sieci komputerowej $\underline{q}(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny

6.2 Przykład symulacyjny 2

W celu pokazania wszystkich wartości dodanych wynikających z zastosowania nowej metody w zadaniach przełączania pakietów, przejdźmy w bieżącym przykładzie symulacyjnym do scenariusza implementującego wielowymiarową sieć komputerową o nieskorelowanych parametrach. Podczas symulacji założono, że w 50 sekundzie następuje skokowa zmiana jej obciążenia. I znowu wyznaczona w oparciu o mechanizm identyfikacji/adaptacji funkcja $p(t)$ kompensuje zachowanie sieci warunkowane zadaniem $q_{ref} = 100$. Na rysunkach 6.20 i 6.21 pokazano odpowiednio funkcję $p(t)$ i $\underline{q}(t)$ wyznaczone w oparciu o algorytmy DMVAC z parametrami zestawionymi w przykładzie symulacyjnym 6.1. W załączniku C zebrano wszystkie autorskie procedury wymagane podczas prowadzenia badań symulacyjnych.



Rysunek 6.20: DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$ przy zmianie obciążeń sieci w 50 sek., obiekt niestacjonarny



Rysunek 6.21: DMVAC: Wykres zmian sieci komputerowej $q(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny

Uwaga 6.2.1. *Należy w tym miejscu zaznaczyć, że badania symulacyjne dla obiektów wielowymiarowych przeprowadzono w oparciu o T -inwersję wielomianu macierzowego $\underline{B}(q^{-1})$ (patrz Rozdział 2.1.2). Wspomnianą inwersję osadzono w prawie sterowania DMVAC 5.26.*

Rozdział 7

Podsumowanie i problemy otwarte

W ramach niniejszej dysertacji doktorskiej przedstawiono autorskie podejście do problemu optymalnego sterowania w kolejkowych sieciach komputerowych z przełączaniem pakietów. W tym celu posłużono się znanymi z tematyki współczesnej teorii systemów i sterowania modelami liniowymi o różnej liczbie wejść i wyjść zdefiniowanymi w dyskretnej dziedzinie czasu. Pokazano, że używając znanych z literatury przedmiotu algorytmów regulacji wielowymiarowej, w znaczący sposób wpływać można na wydajność pracy wspomnianych sieci opisanych modelami stacjonarnymi i niestacjonarnymi typu wejście-wyjście. Połączenie mechanizmów identyfikacji oraz adaptacji z regulacją tzw. 'okołominimalnowariancyjną' daje wymierne efekty w szczególności dla niekwadratowych obiektów MIMO. Przeprowadzone w ramach rozprawy badania symulacyjne wykazały, że istnieje możliwość równomiernego obciążenia sieci, tj. użycia jej maksymalnej przepustowości, wskutek implementacji nowej metody. Stąd naturalnym wydaje się przeprowadzenie w ramach przyszłych badań weryfikacji praktycznej otrzymanych rezultatów heurystycznych. Należy zaznaczyć, że rzeczona walidacja wiązałaby się z adaptacją szeregu protokołów transmisyjnych, co dodatkowo nadawałoby złożoności poruszanej problematyce. Kolejnym interesującym wątkiem badawczym mogłoby okazać się zagadnienie analizy i syntezy obiektów sieci kolejkowych posiadających więcej niż jedno wyjście. W tym celu należałoby użyć suplementarnego aparatu matematycznego, uzupełnionego o dodatkowe zależności macierzowo-wektorowe. Ciekawym zagadnieniem z pewnością będzie wykorzystanie innych niż T -inwersja

wielomianowo-macierzowa niejednoznacznych odwrotności. Podejście takie bez wątplenia zaowocowałoby uodpornieniem całego mechanizmu przełączania pakietów, głównie w kontekście wymaganej minimalnej energii dostarczanej do procesu. Rezultaty równoległe prowadzonych badań z zakresu minimalnoenergetycznego sterowania minimalnowariancyjnego/perfekcyjnego dla obiektów LTI MIMO z czasem dyskretnym w pewnym stopniu już potwierdzałyby postawioną tezę. Ostatnim ważnym problemem, wartym poruszenia w ramach przyszłych badań, wydaje się być przedstawienie postulowanego w ramach dysertacji doktorskiej nowego podejścia w ujęciu stochastycznym, gdzie główną rolę odgrywałby np. powszechnie już używany rozkład Pareto.

Konkludując, zdaniem autora rozprawy doktorskiej, wszystkie postawione w niej tezy zostały z powodzeniem dowiedzione. Zestawione powyżej problemy otwarte każą podjąć wysiłek badawczy w przyszłości.

Literatura

- [1] A. Domański, J. Domańska, T. Czachórski, and F. Kłębczyk. *Sieci Komputerowe*, chapter Suitability study of Differentiated Service for Quality of Service in wireless LAN. Wydawnictwa Komunikacji i Łączności, 2008. 1
- [2] A. Domański, J. Domańska, T. Czachórski, and F. Kłębczyk. Some aspects of QoS in wireless LAN. *Studia Informatica*, 2008. 1
- [3] P. Dzierwa. A Review of Control Algorithms in the Packet Switching Networks. *Scientific Bulletin of the Opole University of Technology*, 62[329]:19–20, 2009. 2
- [4] M. J. Blesa. Deciding Stability in Packet-Switched FIFO Networks Under the Adversarial Queuing Model in Polynomial Time. *Lecture Notes in Computer Science*, 3724:429–441, 2005. 2
- [5] J. Klamka and J. Tańcula. Examination of robust D-stability of TCP-DCR protocol. *Theoretical and Applied Informatics*, 24(4):327–344, 2012. 2
- [6] S. Floyd and V. Jacobson. Random Early Detection Gateways for congestion Avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, 1993. 2
- [7] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang. Recommendations on queue management and congestion avoidance in the Internet. *IETF RFC 2309*, 1998. 2

- [8] B. Abbasov and S. Korukoğlu. An active queue management algorithm for reducing packet loss rate. *Mathematical and Computational Applications*, 14(1):65–72, 2009. 2
- [9] J. Domańska and A. Domański. Adaptive RED in AQM. In Andrzej Kwiecień, Piotr Gaj, and Piotr Stera, editors, *Computer Networks*, pages 174–183, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. 2
- [10] J. Domańska, A. Domański, and S. Nowak. *Performance Modeling of Selected AQM Mechanisms in TCP/IP Network*, pages 11–20. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. 2
- [11] J. Domańska and A. Domański. AQM in Linux based routers - comparing with analytical and simulation results. *Theoretical and Applied Informatics*, Vol. 20, No. 4:277–292, 2008. 2
- [12] C. Hollot, V. Misra, D. I. Towsley, and W. B. Gong. Analysis and design of controllers for AQM routers supporting TCP flows. *IEEE Transaction on Automatic Control*, 47(6):945–959, 2002. 2
- [13] M. S. Herfeh and M. Haeri. Active Queue Management in TCP Networks Based on Minimum Variance Control. *Proc. of International Symposium on Communications, Control and Signal Processing (ISCCSP), Marrakech, Morocco*, 2006. 2, 3, 26, 27
- [14] H. Sirisena and M. Hassan. Generalized Minimum Variance Control of Queues in Packet Switching Networks. *Communications in Information and Systems*, 2(4):385–398, 2002. 2
- [15] P. Dzierwa. Minimum variance control in the packet switching computer networks with a queuing mechanism. (in Polish). *Proc. of the Seminář Odborné Naučný, Opole*, pages 13–17, 2010. 3
- [16] P. Dzierwa. Minimum Variance Adaptive Control in Packet Switching Networks. *Scientific Bulletin of the Opole University of Technology*, 66[342]:17–18, 2011. 3

- [17] W. P. Hunek and K. J. Latawiec. A Study on New Right/Left Inverses of Nonsquare Polynomial Matrices. *International Journal of Applied Mathematics and Computer Science*, 21(2):331–348, 2011. 3, 10, 11, 12, 14, 26
- [18] W. P. Hunek. *Towards a General Theory of Control Zeros for LTI MIMO Systems*. Opole University of Technology Press, 2011. 3, 9, 10, 11, 12, 14, 15, 17, 21, 37
- [19] B. Filipowicz and J. Kwiecień. Queuing systems and networks. Models and applications. *Bulletin of the Polish Academy of Sciences – Technical Sciences*, 56(4):379–390, 2008. 4
- [20] J. Bilal. *Application of Cellular Neural Network for Optimization of Fast Packet Switching in Data Communication Systems with Asynchronous Transfer ATM(in Polish)*. PhD thesis, Łódź University of Technology, 2005. 5
- [21] K. Zatwarnicki. Adaptive control of cluster-based Web systems using neuro-fuzzy models. *International Journal of Applied Mathematics and Computer Science*, 22(2):365–377, 2012. 5
- [22] P. Świątek. Prediction of Queue Lengths in the Multi-Queue-Single-Processor Queuing System Based on Fuzzy-Neural Approach. *Proc. of the 19th International Conference on System Engineering(ICSEng), 2008, Las Vegas, USA*, pages 171–176, 2008. 5
- [23] M. Turowska. Zastosowanie zmiennych niepewnych do oceny stabilności systemów sterowania przeciążeniem w sieci ATM (in Polish). *Proc. of the XIV Krajowa Konferencja Automatyki (KKA), Zielona Góra, Polska*, pages 787–792, 2002. 5
- [24] P. Ignaciuk and A. Bartoszewicz. *Congestion Control in Data Transmission Networks: Sliding Mode and Other Designs*. Springer-Verlag, London, 2013. 5

- [25] K. I. Park. *QoS in Packet Networks*. PhD thesis, The MITRE Corporation USA, 2005. 5
- [26] W. P. Hunek and P. Dzierwa. New Results in Generalized Minimum Variance Control of Computer Networks. *Information Technology and Control*, 43(3):315–320, 2014. 5, 11, 14, 27, 29, 31, 32
- [27] T. Kaczorek. *Wektory i macierze w automatyce i elektrotechnice*. Wydawnictwo Naukowo-Techniczne, 1998. 6
- [28] C. C. MacDuffy. *The Theory of Matrices*. Dover Publications Inc., Mineola New York, 2004. 7
- [29] H. H. Rosenbrock. *State-space and Multivariable Theory*. Nelson-Wiley, New York, 1970. 7
- [30] W. A. Wolowich. *Linear Multivariable Systems*. Springer-Verlag, New York, 1974. 7
- [31] R. Patel. Computation of matrix fraction descriptions of linear time-invariant systems. *IEEE Transaction on Automatic Control*, 26(1):148–161, 1981. 7
- [32] K. J. Latawiec. *Contributions to Advanced Control and Estimation for Linear Discrete-Time MIMO Systems*. Opole University of Technology Press, Opole, Poland, 1998. 9
- [33] W. P. Hunek. An Application of New Polynomial Matrix σ -Inverse in Minimum-Energy Design of Robust Minimum Variance Control for Non-square LTI MIMO Systems. *Proc. of the 8th IFAC Symposium on Robust Control Design ROCOND 2015*, 48(14):150 – 154, 2015. 9, 10, 11, 14, 26
- [34] W. P. Hunek, K. Latawiec, R. Stanisławski, M. Łukaniszyn, and P. Dzierwa. A new form of a σ -inverse for nonsquare polynomial matrices. *Proc. of the 18th International Conference on Methods and Models in Automation and Robotics (MMAR'2013), Międzyzdroje, Polska*, pages 282–286, 2013. 9, 11, 12, 14, 17, 26

- [35] W. P. Hunek, K. J. Latawiec, P. Majewski, and P. Dzierwa. An application of a new matrix inverse in stabilizing state-space perfect control of nonsquare LTI MIMO systems. *Proc. of the 19th International Conference on Methods and Models in Automation and Robotics (MMAR'2014), Międzyzdroje, Polska*, pages 451–455, 2014. 9, 11, 14, 26
- [36] W. P. Hunek. An application of polynomial matrix σ -inverse in minimum-energy state-space perfect control of nonsquare LTI MIMO systems. *Proc. of the 20th International Conference on Methods and Models in Automation and Robotics (MMAR'2015), Międzyzdroje, Polska*, pages 252–255, 2015. 9, 11, 14, 26
- [37] S. Bańka. *Multivariable control systems – A polynomial approach (in Polish)*, volume 11. Szczecin University of Technology Press, Szczecin, 2007. 9
- [38] A. Witkowska. Zastosowanie macierzy pseudoodwrotnej w metodach alokacji pędników układu dynamicznego pozycjonowania statku. *Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej*, 40:145–148, 2014. 9, 10
- [39] F. Soleimani, F. Soleymani, and others. Some Matrix Iterations for Computing Generalized Inverses and Balancing Chemical Equations. *Algorithms*, 8(4):982–998, 2015. 9, 10
- [40] Z. Zhang. Pseudo-inverse multivariate/matrix-variate distributions. *Journal of Multivariate Analysis*, 98(8):1684 – 1692, 2007. 9, 10
- [41] Z. Zhang, X. Hu, and L. Zhang. Least-squares solutions of inverse problem for hermitian generalized Hamiltonian matrices. *Applied Mathematics Letters*, 17(3):303 – 308, 2004. 9, 10
- [42] E. Pawelec, A. Smoktunowicz, and I. Wróbel. Metody iteracyjne wyznaczania odwrotności Moore’a-Penrose’a. *Proc. of the XLI Ogólnopolska Konferencja Zastosowań Matematyki, Zakopane, Polska*, 2012. 9, 10

- [43] T. Söderström and G. W. Stewart. On the numerical properties of an iterative method for computing the moore-penrose generalized inverse. *SIAM J.*, 11:61–74, 1974. 9
- [44] I. Wróbel. Własności numeryczne algorytmów wyznaczania uogólnionej odwrotności Moore'a-Penrose'a. *Proc. of the XL Ogólnopolska Konferencja Zastosowań Matematyki, Zakopane, Polska*, 2011. 10
- [45] X. Pan and M. Qin. The Discriminance for FLDcirc_r Matrices and the Fast Algorithm of Their Inverse and Generalized Inverse. *Advances in Linear Algebra & Matrix Theory*, 5:54–61, 2015. 10
- [46] S. K. Bose and D. T. Nguyen. Efficient Generalized Inverse for Solving Simultaneous Linear Equations. *Journal of Applied Mathematics and Physics*, 4:16–20, 2016. 10
- [47] W. P. Hunek. A New General Class of MVC-Related Inverses of Nonsquare Polynomial Matrices Based on the Smith Factorization. *Proc. of: 14th IFAC Conference on Methods and Models in Automation and Robotics*, 42(13):372 – 377, 2009. 10, 26
- [48] W. P. Hunek. Towards robust minimum variance control of nonsquare LTI MIMO systems. *Archives of Control Sciences*, 18(LIV)(1):59–71, 2008. 10
- [49] W. P. Hunek. Pole-free vs. stable-pole designs of minimum variance control for nonsquare LTI MIMO systems. *Bulletin of the Polish Academy of Sciences – Technical Sciences*, 59(2):201–211, 2011. 10, 14
- [50] K. J. Latawiec and W. P. Hunek. Inwersje macierzy wielomianowych a sterowanie minimalnowariancyjne. *Proc. of: 15th National Automatic Control Conference (KKA'2005), Warszawa, Polska*, 2005. 10
- [51] P. Majewski and W. P. Hunek. Perfekcyjna rekonstrukcja sygnału - nowe analityczne podejście. *Proc. of: 18th National Automatic Control Conference (KKA'2014), Wrocław, Polska*, pages 737–742, 2014. 11

- [52] W. P. Hunek and J. Korniak. Polynomial Matrix Inverse Approach to Fault-Tolerant Perfect Control. *Proc. of the 8th IFAC Symposium on Robust Control Design (ROCOND'2015), Bratislava, Slovakia*, 48(14):203 – 207, 2015. 11, 26
- [53] P. Majewski and W. P. Hunek. Increasing the capacity of wireless data transmission using polynomial matrix calculus (in Polish). *Proc. of the 38th International Conference on Fundamentals of electrotechnics and Circuit Theory (SPETO'2015), Gliwice-Ustroń, Poland*, pages 95 – 96, 2015. 11
- [54] J. Korniak and W. P. Hunek. Polynomial Matrix Inverse Approach to Minimum-Energy Fault-Tolerant Control, Advanced Systems for Automation and Diagnostics, Control and Computer Science. *Proc. of the 12th International Conference on Diagnostics of Processes and Systems (DPS'2015), Ustka, Poland*, pages 33 – 38, 2015. 11
- [55] Ł. Wach and W. P. Hunek. Perfect Control for Fractional-Order Multivariable Discrete-Time Systems, Theoretical Developments and Applications of Non-Integer Order Systems, Lecture Notes in Electrical Engineering 357. *Proc. of the 7th Conference on Non-Integer Order Calculus and Its Applications (RRNR'2015), Szczecin, Poland*, pages 233 – 237, 2015. 11
- [56] W. P. Hunek. Contribution a new matrix H -inverse to stabilize the perfect control for LTI MIMO state-space systems. *test*, 99(99):99, 2016. 11, 26
- [57] William S. Levine. *The Control Handbook, (Three Volume Set) (Electrical Engineering Handbook)*. CRC Press, Inc., Boca Raton, FL, USA, 2nd edition, 2010. 13
- [58] H. Jones Richard. Maximum likelihood fitting of arma models to time series with missing observations. *Technometrics*, 22(3):389–395, 1980. 13
- [59] J. Durbin and S. J. Koopman. *Time Series Analysis by State Space Methods: Second Edition*. Oxford Statistical Science Series. OUP Oxford, 2012. 13

- [60] W. P. Hunek and K. J. Latawiec. Minimum variance control of discrete-time and continuous-time LTI MIMO systems – a new unified framework. *Control and Cybernetics*, 38(3):609–624, 2009. 14
- [61] W. P. Hunek. Towards robust minimum variance control of nonsquare LTI MIMO systems. *Archives of Control Sciences*, 18 (LIV)(1):59–71, 2008. 14
- [62] K. J. Latawiec. *The Power of Inverse Systems in Linear and Nonlinear Modeling and Control*. Opole University of Technology Press, 2004. 14
- [63] W. P. Hunek. Contribution of a new matrix H -inverse to stabilizing the perfect control for LTI MIMO state-space systems. In *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 64–69, August 2016. 14, 26
- [64] W. P. Hunek. New SVD-based matrix H -inverse vs. minimum-energy perfect control design for state-space LTI MIMO systems. In *Proc. of the 20th IEEE International Conference on System Theory, Control and Computing (ICSTCC'16), 13-15 October 2016, Sinaia, Romania, 2016*, 2016. 14
- [65] R. Inseman, K-H. Lachmann, and D. Matko. *Adaptive Control Systems*. Prentice Hall International, Wielka Brytania, 1992. 22, 23
- [66] T. Söderström and P. Stoica. *Identyfikacja systemów*. Wydawnictwo PWN, Warszawa, 1997. 22
- [67] D. Horla. *Sterowanie adaptacyjne ćwiczenia laboratoryjne*. Wydawnictwo Politechniki Poznańskiej, Poznań, 2003. 22, 23, 27
- [68] W. P. Hunek. Contribution of a new matrix H -inverse to stabilizing the perfect control for LTI MIMO state-space systems. In *Proc. of the 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 64–69, Aug 2016. 26

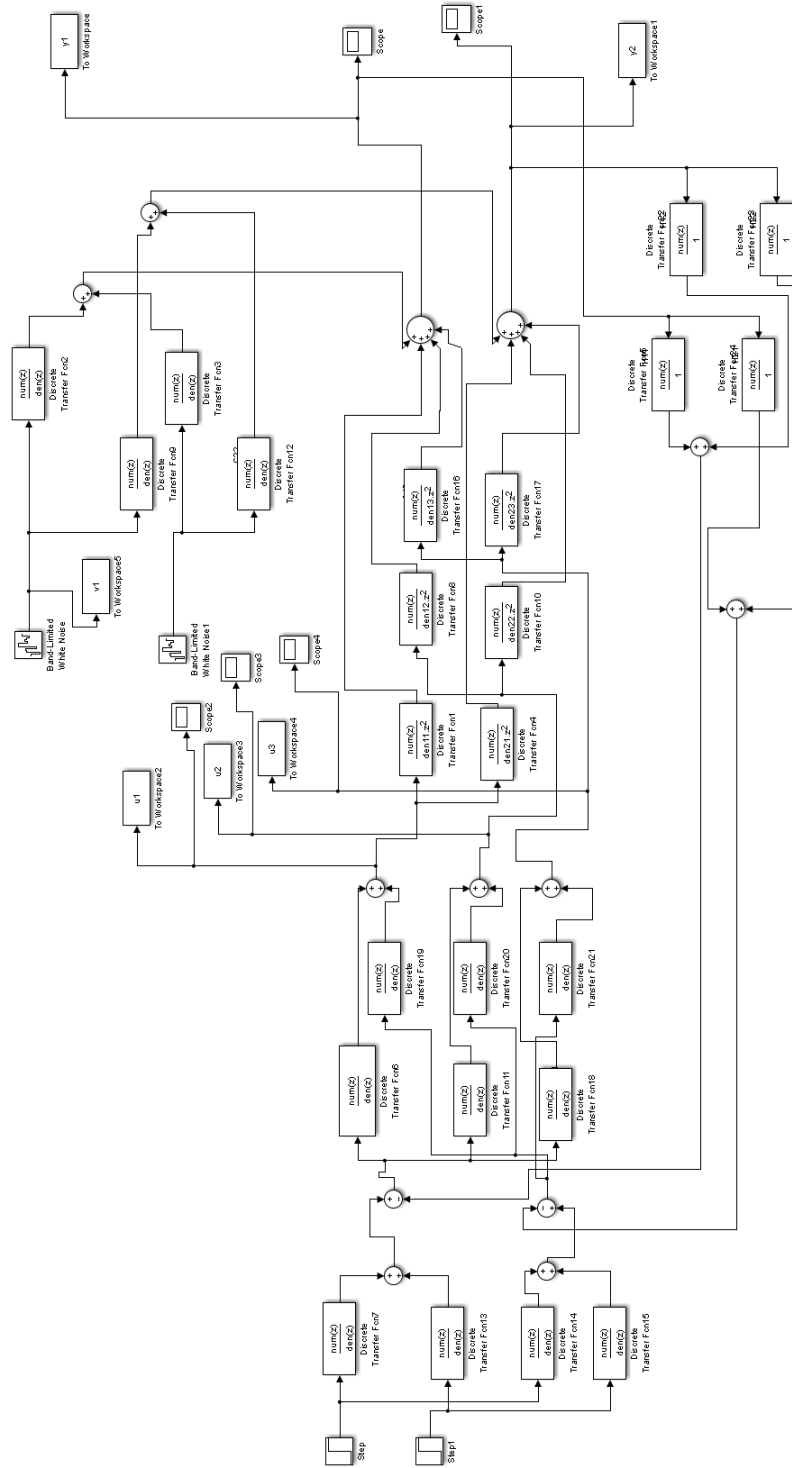
- [69] W. P. Hunek. New SVD-based matrix H -inverse vs. minimum-energy perfect control design for state-space LTI MIMO systems. In *Proc. of the 20th International Conference on System Theory, Control and Computing (ICSTCC)*, pages 14–19, Oct 2016. 26
- [70] L. Ljung. *System Identification: Theory for the User*. Prentice Hall information and system sciences series. Prentice Hall PTR, 1999. 27
- [71] K. J. Åström and B. Wittenmark. *Adaptive Control*. Addison-Wesley Publishing Company, Inc., 2 edition, 1995. 27, 29
- [72] S. Kaczmarz. Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres*, 35:355–357, 1937. 27
- [73] T. Strohmer and R. Vershynin. A Randomized Kaczmarz Algorithm with Exponential Convergence. *Journal of Fourier Analysis and Applications*, 15(2):262, 2008. 27
- [74] R. Wójcicki. New method of modeling self-similar data traffic using Markov Modulated Poisson Process. *Studia Informatica*, 26(2):23–39, 2005. 27
- [75] T. Czachórski, J. Domańska, and A. Sochan. Samopodobny charakter natężenia ruchu w sieciach komputerowych. *Studia Informatica*, 22, 2001. 27
- [76] W. Willinger, M. S. Taqqu, W. E. Leland, and D. V. Wilson. Self-similarity in high-speed packet traffic: Analysis and modeling of ethernet traffic measurements. *Statistical Science*, 10(1):67–85, 1995. 27
- [77] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the Self-Similar Nature of Ethernet Traffic. *Proc. of SIGCOMM '93, September*, pages 183–193, 1993. 27
- [78] K. L. Mills, J. Filliben, D. Cho, E. Schwartz, and D. Genin. *Study of Proposed Internet Congestion Control Mechanisms*. NIST Special Publication 500-282. NIST, 2010. 28

- [79] Choi, Y. and Silvester, J. A. Observing the effect of TCP Congestion Control on Network Traffic. *Proc. of: Applied Telecommunications Symposium, Hyatt Regency Crystal City, Arlington, VA, USA*, pages 65–68, 2004. 28
- [80] M. Mitzenmacher. Dynamic Models for File Sizes and Double Pareto Distributions. *Internet Mathematics*, 1(3):305–333, 2003. 28
- [81] G. Suchacka. Modelowanie ruchu webowego dla potrzeb badania wydajności biznesowego serwisu WWW. *Pomiary Automatyka Kontrola*, 12:57–60, 2010. 28
- [82] V. Paxson. Empirically-Derived Analytic Models of Wide-Area TCP Connections. *IEEE/ACM Transactions on Networking*, 2(4):1–25, 1994. 28

Załącznik A.

Schemat układu sterowania MV/perfekcyjnego obiektem o trzech wejściach i dwóch wyjściach

Układ sterowania MV/perfekcyjnego stworzony w środowisku Matlab/Simulink



Rysunek 7.22: Schemat układu sterowania MV/perfekcyjnego obiektami o trzech wejściach i dwóch wyjściach

Załącznik B.

Procedura wyznaczania algorytmu sterowania minimalnowariancyjnego/perfekcyjnego

```
%----- Procedura wyznaczania algorytmu
%          sterowania minimalnowariancyjnego/perfekcyjnego

pinit
w=sym('w','real');

%----- Wartości macierzy
A=[1-0.6*w 0.08*w;0.05*w 1-1.04*w];
B=[1-1.01*w 0.1-0.4*w 0.01-0.03*w;0.34-0.08*w 0.1-0.22*w 1-1.07*w];
C=[1-0.01*w 2-0.02*w;3-0.1*w 2-0.1*w];

%----- rozpisanie równan diofantycznych
%----- zamiana na macierze wielomianowe
A1=[mat2pol(sym2poly(A(1,1))) mat2pol(sym2poly(A(1,2))));
    mat2pol(sym2poly(A(2,1))) mat2pol(sym2poly(A(2,2)))]];
C1=[mat2pol(sym2poly(C(1,1))) mat2pol(sym2poly(C(1,2))));
    mat2pol(sym2poly(C(2,1))) mat2pol(sym2poly(C(2,2)))]];
B1=[mat2pol([1 0 0]) 0;0 mat2pol([1 0 0])];

[F,H]=xaybc(A1,B1,C1);

%----- zamiana na postać matematyczną zgodnie ze zmienną w
Fs=[poly2sym(pol2mat(F(1,1)),'w') poly2sym(pol2mat(F(1,2)),'w')];
```

```

poly2sym(pol2mat(F(2,1)), 'w') poly2sym(pol2mat(F(2,2)), 'w')];
Hs=[pol2mat(H(1,1)) pol2mat(H(1,2)); pol2mat(H(2,1)) pol2mat(H(2,2))];
Cs=[poly2sym(pol2mat(C(1,1)), 'w') poly2sym(pol2mat(C(1,2)), 'w');
poly2sym(pol2mat(C(2,1)), 'w') poly2sym(pol2mat(C(2,2)), 'w')];

%————— zamiana na postac z "w" na "q^{-1}"
q=sym('q', 'real');
Csq=[limit(Cs(1,1),w,q^(-1)) limit(Cs(1,2),w,q^(-1));
limit(Cs(2,1),w,q^(-1)) limit(Cs(2,2),w,q^(-1))];
Fsq=[limit(Fs(1,1),w,q^(-1)) limit(Fs(1,2),w,q^(-1));
limit(Fs(2,1),w,q^(-1)) limit(Fs(2,2),w,q^(-1))];
Asq=[limit(A(1,1),w,q^(-1)) limit(A(1,2),w,q^(-1));
limit(A(2,1),w,q^(-1)) limit(A(2,2),w,q^(-1))];
Bsq=[limit(B(1,1),w,q^(-1)) limit(B(1,2),w,q^(-1))
limit(B(1,3),w,q^(-1));limit(B(2,1),w,q^(-1))
limit(B(2,2),w,q^(-1)) limit(B(2,3),w,q^(-1))];

%————— przygotowanie zmiennych
[num,den]=numden(simplify(Csq));

numC11=sym2poly(num(1,1));
numC12=sym2poly(num(1,2));
numC21=sym2poly(num(2,1));
numC22=sym2poly(num(2,2));
denC11=sym2poly(den(1,1));
denC12=sym2poly(den(1,2));
denC21=sym2poly(den(2,1));
denC22=sym2poly(den(2,2));

inve=inv(Asq)*Bsq;
[num,den]=numden(simplify(inve));

num11=sym2poly(num(1,1));
num12=sym2poly(num(1,2));
num13=sym2poly(num(1,3));
num21=sym2poly(num(2,1));
num22=sym2poly(num(2,2));
num23=sym2poly(num(2,3));

den11=sym2poly(den(1,1));
den12=sym2poly(den(1,2));
den13=sym2poly(den(1,3));
den21=sym2poly(den(2,1));
den22=sym2poly(den(2,2));
den23=sym2poly(den(2,3));

```

```

%————— inwersja macierzy B

b=[0.8 -1.4 -3.1;1 -1 -1];
BR=b'*inv(Bsq*b');
wyz=det(Bsq*b');
[num,den]=numden(simplify(wyz));
bieguny=roots(sym2poly(num(1,1)))
invee=BR*inv(Fsq);

[num,den]=numden(simplify(invee));
numBF11=sym2poly(num(1,1));
numBF12=sym2poly(num(1,2));
numBF21=sym2poly(num(2,1));
numBF22=sym2poly(num(2,2));
numBF31=sym2poly(num(3,1));
numBF32=sym2poly(num(3,2));
denBF=sym2poly(den(1,1));

%————— uruchomienie symulacji

options = simset('SrcWorkspace','current');
sim('pd_nx3_ny2MIMO',[ ], options);

hAcs = getActiveConfigSet(gcs);
Stop_time=get_param(hAcs, 'StopTime');

%————— wykresy
run plot_mimo_u.m
run plot_mimo_y.m

```

Załącznik C

Autorskie procedury wymagane podczas prowadzenia badań symulacyjnych

1. Procedura

```
clear all
close all
clc

T=100;
A=0.8;
B=0.2;
C=0.10;

zakres_A=1;
zakres_B=0.6;
zakres_C=3;

krok_zmiany_A=0.04;
krok_zmiany_B=0.1;
krok_zmiany_C=0.02;

% 1 - zmieniamy A
% 2 - zmieniamy B
% 3 - zmieniamy C
% 4 - zmieniamy wszystkie

zmiennik = 2;

switch zmiennik
```

```

case 1
disp('Symulacja zmiany parametrow glownych modelu przy zadnym starcie zmiana A')

pom1=1;
pom2=round((zakres_A-A)/krok_zmiany_A);

p_c=zeros(pom2,1,T);
p_licz_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
A=A+krok_zmiany_A;
[p1, p_licz, q1, a1, b1, c1]=test_shakiba(A,B,C);

p_c(pom1,1,:)= p1(1,:);
p_licz_c(pom1,1,:)= p_licz(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);

max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));
pom1=pom1+1;
end

figure(1);
hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';

```

```

A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian A');
legenda = cell(1, pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian A');
legenda = cell(1, pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(4)
hold on

```

```

myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

case 2
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie parametr B')
pom1=1;
pom2=round((zakres_B-B)/krok_zmiany_B);
p_c=zeros(pom2,1,T);
p_licz_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom1);

A_legenda=A;
A_stale=A;

```



```

B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end

B=B+krok_zmiany_B;
[p1 , p_licz , q1 , a1 , b1 , c1]=test_shakiba (A,B,C);

p_c(pom1,1,:)= p1 (1 ,:);
p_licz_c(pom1,1,:)= p_licz (1 ,:);
q_c(pom1,1,:)= q1 (1 ,:);
a_c(pom1,1,:)= a1 (1 ,:);
b_c(pom1,1,:)= b1 (1 ,:);
c_c(pom1,1,:)= c1 (1 ,:);
max_p(1,pom1)=max(p_c(pom1,1 ,:));
min_p(1,pom1)=min(p_c(pom1,1 ,:));
pom1=pom1+1;
end

myMap = rand(pom2, 3);
legenda = cell(1,pom2);
txt_pom='B=';
figure(1);
hold on
title('Wykres zmian parametru p w zaleznosci od zmian B');
xlabel('czas [s$]', 'Interpreter','latex');
ylabel('$p(t)$', 'Interpreter','latex');
txt_pom='B=';
B_legenda=B_stale;
for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)= cellstr( strcat(txt_pom, num2str(B_legenda, 3)));
end

legend(legenda{:},4);
B_legenda=B_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);

```

```

title('Wykres zmian parametru q w zaleznosci od zmian B');
xlabel('czas [ $$ ]', 'Interpreter', 'latex');
ylabel('$\underline{q}(t)$', 'Interpreter', 'latex');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end
as2=cat(2, legenda{:});
legend(legenda{:}, 4);
B_legenda=B_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian B');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end
legend(legenda{:}, 4);
B_legenda=B_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian B');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end

```

```

legend(legenda{:},4);
B_legenda=B_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian B');
legenda = cell(1,pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end
legend(legenda{:},4);
B_legenda=B_stale;
hold off

case 3
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie zmiana C')
pom1=1;
pom2=round((zakres_C-C)/krok_zmiany_C);

p_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
C=C+krok_zmiany_C;
[p1, q1, a1, b1, c1]=test_shakiba(A,B,C);

```

```

p_c(pom1,1,:)= p1(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);

max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));
pom1=pom1+1;
end

figure(1);
hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;
for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)= cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;
for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)= cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(3)

```

```

hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);

```

```

C_legenda=C_stale;
hold off

case 4
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie zmiana A,B,C')
pom1=1;

zak=zeros(1,3);
zak(1)=round((zakres_A-A)/krok_zmiany_A);
zak(2)=round((zakres_B-B)/krok_zmiany_B);
zak(3)=round((zakres_C-C)/krok_zmiany_C);

pom2=round(max(zak));

p_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
A=A+krok_zmiany_A;
B=B+krok_zmiany_B;
C=C+krok_zmiany_C;
[p1,q1,a1,b1,c1]=test_shakiba(A,B,C);

p_c(pom1,1,:)= p1(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);

```

```

max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));

pom1=pom1+1;

end

figure(1);
hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA='A=';
txt_pomB='B=';
txt_pomC='C=';
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
% wykres zmian parametru p w zaleznosci od zmian A,B,C
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;

```

```

B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)= cellstr( strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru a w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)= cellstr( strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru b w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

```



```

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3), txt_pomB, ...
num2str(B_legenda,3), txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru c w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3), txt_pomB, ...
num2str(B_legenda,3), txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off
end

```

2. Procedura

```
function [p1, p_licz, q1, a1, b1, c1]=test_shakiba(A,B,C)
```

```

T=100;
p1=zeros(1,T);
p_licz=zeros(1,T);
q1=zeros(1,T);
a1=zeros(1,T);
b1=zeros(1,T);
c1=zeros(1,T);
q_ref = 100;
alfa = 0.1;
gamma = 1;

a=A;
b=B;
c=C;

q_t=zeros(1,2);
p_t=zeros(1,2);
p_licz_t=zeros(1,2);
e_t=zeros(1,2);
x_t=zeros(1,2);
w_t=zeros(1,2);
esty_t=zeros(1,3)';

pom_esty=0;

fi_t=[-q_t(2),p_t(2),e_t(2)]';
teta=[a,b,c]';
fi_t_1=[0,0,0]';
teta_1=[a,b,c]';
teta_st=teta;
esty_t_1=teta;
i=2;

a1(1)=a;
b1(1)=b;
c1(1)=c;
while 1>0
if i==T

p1(i)=p_t(2);
p_t(2)=p_t(1);

for pp=1:T

```

```

if pp+1<=T
p1(pp)=p1(pp+1);
end
end

q1(i)=q_t(2);
q_t(2)=q_t(1);

for ppp=1:2
for pp=1:T
if pp+1<=T
q1(pp)=q1(pp+1);
end
end
end
e_t(2)=e_t(1);

esty_t_1=esty_t;

fi_t_1=fi_t;

teta_1=teta;
teta=esty_t;

a=teta(1);
b=teta(2);
c=teta(3);

a1(i)=a;
b1(i)=b;
c1(i)=c;
break;
end

q_t(1)=a*p_t(2)-b*q_t(2);
p_t(1)=-((c-a)/b)*(q_t(1)-q_ref)+((1+a)/b)*q_ref;
p_licz_t(1)=p_t(1)+p_licz_t(2);
fi_t=[-q_t(2),p_t(2),e_t(2)]';
e_t(1)=q_t(1)-fi_t_1'*teta_1;
esty_t=esty_t_1+(((gamma*fi_t)/(alfa+fi_t'*fi_t))*(q_t(1)-fi_t'*esty_t_1));
p1(i)=p_t(2);
p_t(2)=p_t(1);
p_licz(i)=p_licz_t(2);
p_licz_t(2)=p_licz_t(1);
q1(i)=q_t(2);
q_t(2)=q_t(1);

```

```

e_t(2)=e_t(1);
esty_t_1=esty_t;
fi_t_1=fi_t;
teta_1=teta;
teta=esty_t;
a=teta(1);
b=teta(2);
c=teta(3);
a1(i)=a;
b1(i)=b;
c1(i)=c;
if i==T/2
a=1;
end

i=i+1;
end

stairs(q1);
grid

```

3. Procedura

```

clear all
close all
clc

T=100;

A=0.8;
B=0.2;
C=0.10;

zakres_A=1;
zakres_B=0.6;
zakres_C=3;

krok_zmiany_A=0.04;
krok_zmiany_B=0.1;
krok_zmiany_C=0.02;
Q=1;

zmiennik = 2;

```

```

switch zmiennik
case 1
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie zmiana A')

pom1=1;
pom2=round((zakres_A-A)/krok_zmiany_A);

p_c=zeros(pom2,1,T);
p_licz_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
A=A+krok_zmiany_A;
[p1,q1,a1,b1,c1]=test_shakiba_GMVAC(A,B,C,Q);

p_c(pom1,1,:)= p1(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);

max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));

pom1=pom1+1;

end

figure(1);

```

```

hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;

```

```

hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian A');
legenda = cell(1,pom2);
txt_pom='A=';
A_legenda=A_stale;

for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(A_legenda, 3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
hold off

case 2
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie parametr B')
pom1=1;
pom2=round((zakres_B-B)/krok_zmiany_B);

p_c=zeros(pom2,1,T);
p_licz_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);

```

```

max_p=zeros(1,pom2);
min_p=zeros(1,pom1);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
B=B+krok_zmiany_B;
[p1 , p_licz , q1 , a1 , b1 , c1]=test_shakiba_GMVAC(A,B,C,Q);

p_c(pom1,1,:)= p1(1,:);
p_licz_c(pom1,1,:)= p_licz(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);
max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));
pom1=pom1+1;

end

myMap = rand(pom2, 3);
legenda = cell(1,pom2);
txt_pom='B=';
figure(1);
hold on
title('Wykres zmian parametru p w zaleznosci od zmian B');
xlabel('czas [$$]', 'Interpreter', 'latex');
ylabel('$p(t)$', 'Interpreter', 'latex');
txt_pom='B=';
B_legenda=B_stale;
for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end

legend(legenda{:}, 4);
B_legenda=B_stale;

```



```

hold off
figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian B');
xlabel('czas [ $$ ]', 'Interpreter', 'latex');
ylabel('$\underline{q}(t)$', 'Interpreter', 'latex');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;
for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end
as2=cat(2, legenda{:});
legend(legenda{:}, 4);
B_legenda=B_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian B');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(B_legenda, 3)));
end
legend(legenda{:}, 4);
B_legenda=B_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian B');
legenda = cell(1, pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))

```

```

legenda(ppp)=cellstr(strcat(txt_pom,num2str(B_legenda,3)));
end
legend(legenda{:},4);
B_legenda=B_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian B');
legenda = cell(1,pom2);
txt_pom='B=';
B_legenda=B_stale;

for ppp=1:pom2
B_legenda=B_legenda+krok_zmiany_B;
stairs(c_c(ppp,:), 'r-', 'Color',myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom,num2str(B_legenda,3)));
end
legend(legenda{:},4);
B_legenda=B_stale;
hold off

case 3
disp('Symulacja zmiany parametrow glownych modelu przy zadanym starcie zmiana C')
pom1=1;
pom2=round((zakres_C-C)/krok_zmiany_C);
p_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
C=C+krok_zmiany_C;

```

```

[p1 , q1 , a1 , b1 , c1]=test_shakiba_GMVAC(A,B,C,Q);

p_c(pom1,1,:)= p1 (1 ,:);
q_c(pom1,1,:)= q1 (1 ,:);
a_c(pom1,1,:)= a1 (1 ,:);
b_c(pom1,1,:)= b1 (1 ,:);
c_c(pom1,1,:)= c1 (1 ,:);

max_p(1 ,pom1)=max(p_c(pom1 ,1 ,:));
min_p(1 ,pom1)=min(p_c(pom1 ,1 ,:));

pom1=pom1+1;

end
figure (1);
hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian C');
legenda = cell(1 ,pom2);
txt_pom='C=';
C_legenda=C_stale;
for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(p_c(ppp ,:), 'r-', 'Color', myMap(ppp ,:))
legenda (ppp)= cellstr ( strcat (txt_pom, num2str(C_legenda ,3)));
end
legend(legenda {:} ,4);
C_legenda=C_stale;
hold off

figure (2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian C');
legenda = cell(1 ,pom2);
txt_pom='C=';
C_legenda=C_stale;
for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(q_c(ppp ,:), 'r-', 'Color', myMap(ppp ,:))
legenda (ppp)= cellstr ( strcat (txt_pom, num2str(C_legenda ,3)));
end
legend(legenda {:} ,4);
C_legenda=C_stale;
hold off

figure (3)

```

```

hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej a w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej b w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);
C_legenda=C_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian zmiennej c w zaleznosci od zmian C');
legenda = cell(1,pom2);
txt_pom='C=';
C_legenda=C_stale;

for ppp=1:pom2
C_legenda=C_legenda+krok_zmiany_C;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pom, num2str(C_legenda, 3)));
end
legend(legenda{:}, 4);

```

```

C_legenda=C_stale;
hold off

case 4
disp('Symulacja zmiany parametrow glownych modelu przy zadnym starcie zmiana A,B,C')
pom1=1;

zak=zeros(1,3);
zak(1)=round((zakres_A-A)/krok_zmiany_A);
zak(2)=round((zakres_B-B)/krok_zmiany_B);
zak(3)=round((zakres_C-C)/krok_zmiany_C);

pom2=round(max(zak));

p_c=zeros(pom2,1,T);
q_c=zeros(pom2,1,T);
a_c=zeros(pom2,1,T);
b_c=zeros(pom2,1,T);
c_c=zeros(pom2,1,T);
max_p=zeros(1,pom2);
min_p=zeros(1,pom2);

A_legenda=A;
A_stale=A;
B_legenda=B;
B_stale=B;
C_legenda=C;
C_stale=C;

while 1>0
if pom1>pom2
break
end
A=A+krok_zmiany_A;
B=B+krok_zmiany_B;
C=C+krok_zmiany_C;
[p1,q1,a1,b1,c1]=test_shakiba_GMVAC(A,B,C,Q);

p_c(pom1,1,:)= p1(1,:);
q_c(pom1,1,:)= q1(1,:);
a_c(pom1,1,:)= a1(1,:);
b_c(pom1,1,:)= b1(1,:);
c_c(pom1,1,:)= c1(1,:);

```

```

max_p(1,pom1)=max(p_c(pom1,1,:));
min_p(1,pom1)=min(p_c(pom1,1,:));

pom1=pom1+1;

end

figure(1);
hold on
myMap = rand(pom2, 3);
title('wykres zmian parametru p w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA='A=';
txt_pomB='B=';
txt_pomC='C=';
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(p_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(2)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru q w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;

```

```

C_legenda=C_legenda+krok_zmiany_C;
stairs(q_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(3)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru a w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(a_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3),txt_pomB,...
num2str(B_legenda,3),txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:},4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(4)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru b w zaleznosci od zmian A,B,C');
legenda = cell(1,pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;

```

```

B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(b_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3), txt_pomB, ...
num2str(B_legenda,3), txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off

figure(5)
hold on
myMap = rand(pom2, 3);
title('Wykres zmian parametru c w zaleznosci od zmian A,B,C');
legenda = cell(1, pom2);
txt_pomA=' A=';
txt_pomB=' B=';
txt_pomC=' C=';

A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
for ppp=1:pom2
A_legenda=A_legenda+krok_zmiany_A;
B_legenda=B_legenda+krok_zmiany_B;
C_legenda=C_legenda+krok_zmiany_C;
stairs(c_c(ppp,:), 'r-', 'Color', myMap(ppp,:))
legenda(ppp)=cellstr(strcat(txt_pomA, num2str(A_legenda,3), txt_pomB, ...
num2str(B_legenda,3), txt_pomC, num2str(C_legenda,3)));
end
legend(legenda{:}, 4);
A_legenda=A_stale;
B_legenda=B_stale;
C_legenda=C_stale;
hold off
end

```

4. Procedura

```

function [p1, p_licz, q1, a1, b1, c1]=test_shakiba_GMVAC(A,B,C,Q)
T=100;

```



```

p1=zeros(1,T);
p_licz=zeros(1,T);
q1=zeros(1,T);
a1=zeros(1,T);
b1=zeros(1,T);
c1=zeros(1,T);
q_t_new=zeros(1,2);
alfa=0.1;
gamma=1;
P=1;

q_ref=zeros(1,2);
q_ref(1)=100;
q_ref(2)=100;

a=A;
b=B;
c=C;

b_0=b;
h0=0;
h1=0;
q_t=zeros(1,3);
p_t=zeros(1,2);
p_licz_t=zeros(1,2);
e_t=zeros(1,2);
x_t=zeros(1,2);
w_t=zeros(1,2);
esty_t=zeros(1,3)';
pom_esty=0;

fi_t=[-q_t(2),p_t(2),e_t(2)]';
teta=[a,b,c]';
fi_t_1=[0,0,0]';
teta_1=[a,b,c]';
teta_st=teta;
esty_t_1=teta;
i=2;
a1(1)=a;
b1(1)=b;
c1(1)=c;
while 1>0
if i==T
p1(i)=p_t(2);
p_t(2)=p_t(1);

for pp=1:T

```

```

if pp+1<=T
p1(pp)=p1(pp+1);
end
end

q1(i)=q_t(2);
q_t(2)=q_t(1);

for ppp=1:3
for pp=1:T
if pp+1<=T
q1(pp)=q1(pp+1);
end
end
end
e_t(2)=e_t(1);
esty_t_1=esty_t;
fi_t_1=fi_t;
teta_1=teta;
teta=esty_t;
a=teta(1);
b=teta(2);
c=teta(3);
a1(i)=a;
b1(i)=b;
c1(i)=c;
break;
end

q_t(1)=b*p_t(2)-(a-1)*q_t(2)+a*q_t(3);
e_t(1)=q_ref(1)+c*q_ref(2)-(-a+1+c)*q_t(1)-a*q_t(2);
p_t(1)=b*e_t(1)/(Q+b^2);
p_licz_t(1)=p_t(1)+p_licz_t(2);
fi_t=[-q_t(2),p_t(2),e_t(2)]';
esty_t=esty_t_1+(((gamma*fi_t)/(alfa+fi_t'*fi_t))*(q_t(1)-fi_t'*esty_t_1));
p1(i)=p_t(2);
p_t(2)=p_t(1);
p_licz(i)=p_licz_t(2);
p_licz_t(2)=p_licz_t(1);
q1(i)=q_t(3);
q_t(3)=q_t(2);
q_t(2)=q_t(1);
e_t(2)=e_t(1);
esty_t_1=esty_t;
fi_t_1=fi_t;

teta_1=teta;

```

```

teta=esty_t;
a=teta(1);
b=teta(2);
c=teta(3);
q_ref(2)=q_ref(1);
a1(i)=a;
b1(i)=b;
c1(i)=c;
if i==T/2
a=100;
end
i=i+1;
end
grid

```

5. Procedura

```

clear all
close all
clc
A=0.8;
B1=0.2;
B2=0.1;
B3=0.5;
C=0.1;
alfa=0.1;
gamma=0.5;
T=100;
%sciezki nalezy uzupelnic !

sciezka_alfa='C:\wykresy_alfa\';
sciezka_gamma='C:\wykresy_gamma\';
sciezka_A='C:\wykresy_A\';
sciezka_B1='C:\wykresy_B1\';
sciezka_B2='C:\wykresy_B2\';
sciezka_B3='C:\wykresy_B3\';
sciezka_C='C:\wykresy_C\';

zakres_alfa=9;
krok_alfa=0.9;

zakres_gamma=0.8;
krok_gamma=0.1;

zakres_A=0.3;
krok_A=0.05;

```

```

zakres_B1=0.7;
krok_B1=0.1;

zakres_B2=0.9;
krok_B2=0.1;

zakres_B3=0.9;
krok_B3=0.1;

zakres_C=1;
krok_C=0.2;

% 1 - zmiana alfa
% 2 - zmiana gamma
% 3 - zmiana A
% 4 - zmiana B1
% 5 - zmiana B2
% 6 - zmiana B3
% 7 - zmiana C
% 8 -
x=4;
switch x
case 1
for i=alfa:krok_alfa:zakres_alfa
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,B1,B2,B3,C,i,gamma,T);
pom1=plot_q_t(q1,A,B1,B2,B3,C,i,gamma);
pom2=plot_p_t(p1,p2,p3,A,B1,B2,B3,C,i,gamma);
txt_alfa1=strcat(sciezka_alfa,'q_t','alfa_',num2str(i),'png');
txt_alfa2=strcat(sciezka_alfa,'p_t','alfa_',num2str(i),'png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 2
for i=gamma:krok_gamma:zakres_gamma
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,B1,B2,B3,C,alfa,i,T);
pom1=plot_q_t(q1,A,B1,B2,B3,C,alfa,i);
pom2=plot_p_t(p1,p2,p3,A,B1,B2,B3,C,alfa,i);
txt_alfa1=strcat(sciezka_gamma,'q_t','gamma_',num2str(i),'png');
txt_alfa2=strcat(sciezka_gamma,'p_t','gamma_',num2str(i),'png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 3
for i=A:krok_A:zakres_A
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(i,B1,B2,B3,C,alfa,gamma,T);
pom1=plot_q_t(q1,i,B1,B2,B3,C,alfa,gamma);

```

```

pom2=plot_p_t(p1,p2,p3,i,B1,B2,B3,C,alfa,gamma);
txt_alfa1=strcat(sciezka_A,'q_t','A',num2str(i),'.png');
txt_alfa2=strcat(sciezka_A,'p_t','A',num2str(i),'.png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 4
for i=B1:krok_B1:zakres_B1
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,i,B2,B3,C,alfa,gamma,T);
pom1=plot_q_t(q1,A,i,B2,B3,C,alfa,gamma);
pom2=plot_p_t(p1,p2,p3,A,i,B2,B3,C,alfa,gamma);
txt_alfa1=strcat(sciezka_B1,'q_t','B1',num2str(i),'.png');
txt_alfa2=strcat(sciezka_B1,'p_t','B1',num2str(i),'.png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 5
for i=B2:krok_B2:zakres_B2
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,B1,i,B3,C,alfa,gamma,T);
pom1=plot_q_t(q1,A,B1,i,B3,C,alfa,gamma);
pom2=plot_p_t(p1,p2,p3,A,B1,i,B3,C,alfa,gamma);
txt_alfa1=strcat(sciezka_B2,'q_t','B2',num2str(i),'.png');
txt_alfa2=strcat(sciezka_B2,'p_t','B2',num2str(i),'.png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 6
for i=B3:krok_B3:zakres_B3
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,B1,B2,i,C,alfa,gamma,T);
pom1=plot_q_t(q1,A,B1,B2,i,C,alfa,gamma);
pom2=plot_p_t(p1,p2,p3,A,B1,B2,i,C,alfa,gamma);
txt_alfa1=strcat(sciezka_B3,'q_t','B3',num2str(i),'.png');
txt_alfa2=strcat(sciezka_B3,'p_t','B3',num2str(i),'.png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
case 7
for i=C:krok_C:zakres_C
[p1,p2,p3,q1,a1,b1,b2,b3,c1]=test_shakiba_miso(A,B1,B2,B3,i,alfa,gamma,T);
pom1=plot_q_t(q1,A,B1,B2,B3,i,alfa,gamma);
pom2=plot_p_t(p1,p2,p3,A,B1,B2,B3,i,alfa,gamma);
txt_alfa1=strcat(sciezka_C,'q_t','C',num2str(i),'.png');
txt_alfa2=strcat(sciezka_C,'p_t','C',num2str(i),'.png');
figureprint(pom1,12,10,'-r500',txt_alfa1);
figureprint(pom2,12,10,'-r500',txt_alfa2);
end
end
end

```

6. Procedura

```
function [p1,p2,p3,q1,a1,b1,b2,b3,c1]= test_shakiba_miso(A,B1,B2,B3,C, alfa ,gamma,T)
close all
clc
p1=zeros(1,T);
p2=zeros(1,T);
p3=zeros(1,T);
q1=zeros(1,T);
a1=zeros(1,T);
b1=zeros(1,T);
b2=zeros(1,T);
b3=zeros(1,T);
c1=zeros(1,T);
q_ref =50;
a=A;
b1=B1;
b2=B2;
b3=B3;
c=C;
q_t=zeros(1,2);
p1_t=zeros(1,2);
p2_t=zeros(1,2);
p3_t=zeros(1,2);
e_t=zeros(1,2);
BB=zeros(1,3)';
esty_t=zeros(1,5)';
pom_esty=0;
fi_t=[-q_t(2),p1_t(2),p2_t(2),p3_t(2),e_t(2)]';
teta=[a,b1,b2,b3,c]';
fi_t_1=[-q_t(2),p1_t(2),p2_t(2),p3_t(2),e_t(2)]';
teta_1=[a,b1,b2,b3,c]';
esty_t_1=teta;
i=1;
b11=b1;
b22=b2;
b33=b3;
a1(1)=a;
b1(1)=b1;
b2(1)=b2;
b3(1)=b3;
c1(1)=c;
while 1>0
if i==T
p1(i)=p1_t(2);
p1_t(2)=p1_t(1);
p2(i)=p2_t(2);
p2_t(2)=p2_t(1);
```

```

p3(i)=p3_t(2);
p3_t(2)=p3_t(1);
for pp=1:T
if pp+1<=T
p1(pp)=p1(pp+1);
p2(pp)=p2(pp+1);
p3(pp)=p3(pp+1);
end
end
q1(i)=q_t(2);
q_t(2)=q_t(1);
for ppp=1:2
for pp=1:T
if pp+1<=T
q1(pp)=q1(pp+1);
end
end
end
e_t(2)=e_t(1);
esty_t_1=esty_t;
fi_t_1=fi_t;
teta_1=teta;
teta=esty_t;
a=teta(1);
b11=teta(2);
b22=teta(3);
b33=teta(4);
c=teta(5);
a1(i)=a;
b1(i)=b11;
b2(i)=b22;
b3(i)=b33;
c1(i)=c;
break;
end
q_t(1)=fi_t'*teta+e_t(2);
fi_t=[-q_t(2),p1_t(2),p2_t(2),p3_t(2),e_t(2)]';
esty_t=esty_t_1+(((gamma*fi_t)/(alfa+fi_t'*fi_t))*(q_t(1)-fi_t'*esty_t_1));
BB=[esty_t(2) esty_t(3) esty_t(4)];
BR=pinv(BB);
p1_t(1)=BR(1)*((1+esty_t(5))*q_ref-(esty_t(5)-esty_t(1))*q_t(1));
p2_t(1)=BR(2)*((1+esty_t(5))*q_ref-(esty_t(5)-esty_t(1))*q_t(1));
p3_t(1)=BR(3)*((1+esty_t(5))*q_ref-(esty_t(5)-esty_t(1))*q_t(1));
e_t(1)=q_t(1)-fi_t_1'*esty_t;
p1(i)=p1_t(2);
p1_t(2)=p1_t(1);
p2(i)=p2_t(2);

```

```

p2_t(2)=p2_t(1);
p3(i)=p3_t(2);
p3_t(2)=p3_t(1);
q1(i)=q_t(2);
q_t(2)=q_t(1);
e_t(2)=e_t(1);
esty_t_1=esty_t;
fi_t_1=fi_t;
teta_1=teta;
teta=esty_t;
a=teta(1);
b11=teta(2);
b22=teta(3);
b33=teta(4);
c=teta(5);
a1(i)=a;
b1(i)=b11;
b2(i)=b22;
b3(i)=b33;
c1(i)=c;
if i==T/2
end
i=i+1;
end
close all

```

7. Procedura

```

function [hFig_p_t]=plot_p_t(p1,p2,p3,A,B1,B2,B3,C,alfa,gamma)
hFig_p_t=figure('Menubar','none');
txt_pom='Wykres zmian wyjscia p(t) dla: ';
strcat(txt_pom,' A=',num2str(A));
hold on
txt_A=strcat(' A=',num2str(A));
txt_B1=strcat(' B1=',num2str(B1));
txt_B2=strcat(' B2=',num2str(B2));
txt_B3=strcat(' B3=',num2str(B3));
txt_C=strcat(' C=',num2str(C));
txt_alfa=strcat(' \alpha=',num2str(alfa));
txt_gamma=strcat(' \gamma=',num2str(gamma));
title({txt_pom; strcat(txt_A,txt_B1,txt_B2,txt_B3,txt_C,txt_alfa,txt_gamma)});
xlabel('czas [s]', 'Interpreter','latex');
ylabel('$p(t)$', 'Interpreter','latex');
hyLabel=ylabel('$p(t)$', 'Interpreter','latex');
hxLabel=xlabel('czas [s]', 'Interpreter','latex');
set(hyLabel,'Fontangle','italic');
set(hyLabel,'Fontname','Helvetica');

```



```

set (hxLabel, 'Fontname', 'Helvetica ');
stairs(p1, 'k-');
stairs(p2, 'k*');
stairs(p3, 'k+');
legend('p1', 'p2', 'p3');
grid
hold off

```

8. Procedura

```

function [hFig_q_t]=plot_q_t(q1,A,B1,B2,B3,C, alfa ,gamma)
hFig_q_t=figure('Menubar','none');
txt_pom='Wykres zmian wyjścia q(t) dla: ';
strcat(txt_pom,' A=',num2str(A));
hold on
txt_A=strcat(' A=',num2str(A));
txt_B1=strcat(' B1=',num2str(B1));
txt_B2=strcat(' B2=',num2str(B2));
txt_B3=strcat(' B3=',num2str(B3));
txt_C=strcat(' C=',num2str(C));
txt_alfa=strcat(' \alpha=',num2str(alfa));
txt_gamma=strcat(' \gamma=',num2str(gamma));
title({txt_pom ; strcat(txt_A,txt_B1,txt_B2,txt_B3,txt_C,txt_alfa,txt_gamma)});
hyLabel=ylabel('$\underline{q}(t)$','Interpreter','latex');
hxLabel=xlabel('czas [s]','Interpreter','latex');
set(hyLabel,'Fontangle','italic');
set(hyLabel,'Fontname','Helvetica');
set(hxLabel,'Fontname','Helvetica');
stairs(q1,'k-');
legend('q(t)');
grid
hold off

```

9. Procedura

```

function figureprint(hFig, width, height, res, filename)
set(hFig, 'PaperUnits','centimeters');
set(hFig, 'PaperSize',[width height]);
set(hFig, 'PaperPosition',[0 0 width height]);
set(hFig, 'PaperOrientation','portrait');
print(hFig, '-dpng', res, filename);

```

Spis rysunków

3.1	Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 1)	16
3.2	Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 1)	16
3.3	Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 2)	17
3.4	Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 2)	18
3.5	Sterowanie minimalnowariancyjne: sygnały sterowań $u_1(t)$, $u_2(t)$, $u_3(t)$ (przykład 3)	19
3.6	Sterowanie minimalnowariancyjne: sygnały wyjść $y_1(t)$, $y_2(t)$ (przykład 3)	19
4.7	Schemat blokowy układu ze sprzężeniem do przodu	23
4.8	Schemat blokowy układu ze sprzężeniem zwrotnym	24
4.9	Schemat blokowy układu z modelem odniesienia	24
4.10	Schemat blokowy układu z estymacją parametrów modelu	25
5.11	DMVAC: Wykres zmian wyjścia sieci komputerowej dla $q_{ref}(t) = 100$	33
5.12	DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny	34
5.13	DMVAC: Wykres zmian wyjścia sieci komputerowej $q(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny	35
5.14	DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny	36
5.15	Schemat sterowania DGMVAC dla sieci komputerowej z dynamiką $G(z)$	38
5.16	DGMVAC: Wykres zmian wyjścia sieci komputerowej dla $q_{ref}(t) = 100$	39
5.17	DGMVAC: Wykres zmian funkcji $\Delta p(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny	40
6.18	DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$, obiekt stacjonarny	43
6.19	DMVAC: Wykres zmian wyjścia sieci komputerowej dla $q_{ref}(t) = 100$	44
6.20	DMVAC: Wykres zmian funkcji $p(t)$ dla $q_{ref}(t) = 100$ przy zmianie obciążeń sieci w 50 sek., obiekt niestacjonarny	45
6.21	DMVAC: Wykres zmian sieci komputerowej $q(t)$ dla $q_{ref}(t) = 100$, zmiana obciążenia w 50 sek., obiekt niestacjonarny	46
7.22	Schemat układu sterowania MV/perfekcyjnego obiektami o trzech wejściach i dwóch wyjściach	60