STUDIA INFORMATICA

Formerly: Zeszyty Naukowe Politechniki Śląskiej, seria INFORMATYKA Quarterly

Volume 32, Number 4C (102)

Jerzy S. RESPONDEK

METODY ANALIZY NUMERYCZNEJ W BADANIACH ZBIORÓW OSIĄGALNYCH UKŁADÓW DYNAMICZNYCH



Silesian University of Technology Press Gliwice 2011

STUDIA INFORMATICA

Formerly: Zeszyty Naukowe Politechniki Śląskiej, seria INFORMATYKA

Editor in Chief

Dr. Marcin SKOWRONEK Silesian University of Technology Gliwice, Poland

Editorial Board

Dr. Mauro CISLAGHI Project Automation Monza, Italy

Prof. Bernard COURTOIS Lab. TIMA Grenoble, France

Prof. Tadeusz CZACHÓRSKI Silesian University of Technology Gliwice, Poland

Prof. Jean-Michel FOURNEAU Université de Versailles - St. Quentin Versailles, France

Prof. Jurij KOROSTIL IPME NAN Ukraina Kiev, Ukraine

Dr. George P. KOWALCZYK Networks Integrators Associates, President Parkland, USA

Prof. Stanisław KOZIELSKI Silesian University of Technology Gliwice, Poland

Prof. Peter NEUMANN Otto-von-Guericke Universität Barleben, Germany **Prof. Olgierd A. PALUSINSKI** University of Arizona Tucson, USA

Prof. Svetlana V. PROKOPCHINA Scientific Research Institute BITIS Sankt-Petersburg, Russia

Prof. Karl REISS Universität Karlsruhe Karlsruhe, Germany

Prof. Jean-Marc TOULOTTE Université des Sciences et Technologies de Lille Villeneuve d'Ascq, France

Prof. Sarma B. K. VRUDHULA University of Arizona Tucson, USA

Prof. Hamid VAKILZADIAN University of Nebraska-Lincoln Lincoln, USA

Prof. Stefan WEGRZYN Silesian University of Technology Gliwice, Poland

Prof. Adam WOLISZ Technical University of Berlin Berlin, Germany

STUDIA INFORMATICA is indexed in INSPEC/IEE (London, United Kingdom)

© Copyright by Silesian University of Technology Press, Gliwice 2011 PL ISSN 0208-7286, QUARTERLY Printed in Poland The paper version is the original version

ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ OPINIODAWCY Prof. dr hab. inż. Lesław SOCHA Prof. dr hab. Andrzej MARCINIAK

KOLEGIUM REDAKCYJNEREDAKTOR NACZELNYREDAKTOR DZIAŁU- Dr inż. Marcin SKOWRONEKSEKRETARZ REDAKCJI- Mgr Elżbieta LEŚKO

Nr kol. 1852

Volume 32, Number 4C (102)

Spis Treści

1.	WPF	OWADZENIE1	1		
	1.1.	Analiza numeryczna w analizie Komitetu Informatyki PAN1	2		
	1.2.	Cel rozprawy	2		
	1.3.	Przegląd problemów rozprawy1	2		
		1.3.1. Algorytm rozwiązywania równania diofantycznego1	2		
		1.3.2. Zastosowanie algorytmu rozwiązywania równań diofantycznych w wyznaczaniu zbiorów osiągalnych wybranej klasy układów dynamicznych1	3		
		1.3.3. Algorytm numeryczny obliczania pewnej klasy wielomianów symetrycznych 1	3		
		1.3.4. Algorytm odwracania konfluentnych macierzy Vandermonde'a 1	4		
		1.3.5. Zastosowanie algorytmów numerycznych w analizie własności dynamicznych układów dynamicznych dowolnego rzędu1	4		
		1.3.6. Numeryczna analiza zbiorów osiągalnych układów dynamicznych przy obustronnym ograniczeniu funkcji wymuszającej	n 4		
2.	ANA	LIZA NUMERYCZNA WYBRANEJ KLASY RÓWNAŃ DIOFANTYCZNYCH1	5		
	2.1.	Wprowadzenie do problematyki równań diofantycznych1	5		
	2.2.	Geneza analizowanej klasy nieliniowych równań diofantycznych1	7		
	2.3. Wybrane problemy złożoności obliczeniowej algorytmów				
	2.4.	Błędy obliczeń numerycznych1	9		
		2.4.1. Lemat Wilkinsona ([133])	0		
	2.5.	Sformułowanie problemu diofantycznego2	0		
	2.6.	Konstrukcja algorytmu rozwiązania nieliniowego równania diofantycznego2	1		
		2.6.1. Przypadek 1: $S(\vec{s}_2, \vec{r}_2) \ge 0$	3		
		2.6.2. Przypadek 2: $S(\vec{s}_2, \vec{r}_2) < 0$	4		
		2.6.3. Uproszczenie przedziałów zmienności parametrów2	5		
	2.7.	Eliminacja rozwiązań symetrycznych2	6		
		2.7.1. Przypadek 1: $p_k > 0, k = 2,, n$	8		
		2.7.2. Przypadek 2: $p_k = 0, k = 2,, n$	0		
	2.8.	Algorytm finalny	0		
	2.9.	Złożoność czasowa algorytmu 2.8	1		

	2.10. Określenie krotności n-tek generujących równe sumy kwadratów	32
	2.11. Analiza błędów obliczeń	33
	2.12. Przykład wykonania algorytmu 2.8	33
	2.13. Praktyczny test efektywności algorytmu 2.8	35
	2.14. Podsumowanie	36
3.	Zastosowanie algorytmów rozwiązywania równań diofantycznych w badaniach zbiorów osiągalnych wybranych układów dynamicznych	37
	3.1. Wprowadzenie	37
	3.2. Wyjaśnienie terminologii przyjętej w rozdziale	37
	3.3. Wprowadzenie do tematyki układów dynamicznych o parametrach rozłożonych typu parabolicznego	38
	3.3.1. Ogólna postać liniowych równań różniczkowych cząstkowych rzędu drugiego	38
	3.3.2. Znaczenie fizyczne równań różniczkowych cząstkowych typu parabolicznego	38
	3.4. Model badanego układu nieskończenie wymiarowego typu parabolicznego	39
	3.5. Analiza numeryczna własności nieskończenie wymiarowego układu parabolicznego	40
	3.5.1. Transformacja układu	40
	3.5.2. Synteza algorytmu numerycznego analizy zbiorów osiągalnych badanego układu 3.5.3. Algorytm	42
	3.5.4. Przykład analizy numerycznej układu parabolicznego	44
	3.6. Podsumowanie rozdziału	46
4.	EFEKTYWNY ALGORYTM NUMERYCZNY OBLICZANIA PEWNEJ KLASY FUNKCJI	47
	SYMETRYCZNYCH	.4/
	4.1. wprowadzenie	4/
	4.2. Pojęcie wielomianow symetrycznych podstawowych	4/
	4.3. Algorytm obliczania wielomianow symetrycznych podstawowych	48
	4.4. Złożoność obliczeniowa algorytmu	49
	4.5. Przykład wykonania algorytmu	49
	4.6. Podsumowanie rozdziału	51
5.	ALGORYTM ODWRACANIA PEWNEJ KLASY MACIERZY O SPECJALNEJ POSTACI	52
	5.1. Wprowadzenie	52
	5.2. Pojęcie konfluentnej macierzy Vandermode'a	52
	5.2.1. Definicja konfluentnej macierzy Vandermonde'a ([31, 82])	53
	5.2.2. Przykłady konfluentnych macierzy Vandermonde'a	53
	5.3. Analityczna metoda odwracania konfluentnych macierzy Vandermode'a5.3.1. Twierdzenie ([31] s. 1543)	54 54
	5.4. Synteza numerycznego algorytmu odwracania konfluentnych macierzy Vandermonde'a	55
	5.4.1. Twierdzenie [82]	55

	5.5. Wyznaczanie wielkości wejściowych algorytmu odwracania konfluentnych macierz Vandermonde'a				
	5.6.	Finalne algorytmy odwracania konfluentnych macierzy Vandermonde'a	57		
		5.6.1. Algorytm wyznaczający ostatnią kolumnę szukanej macierzy odwrotnej	58		
		5.6.2. Algorytm wyznaczający pozostałe kolumny szukanej macierzy odwrotnej	61		
	5.7.	Złożoność obliczeniowa algorytmu odwracania konfluentnych macierzy Vandermonde'a	62		
	5.8.	Przykład wykonania algorytmu	63		
	5.9.	Podsumowanie	65		
6.	ZAS ODV DYN	STOSOWANIE ALGORYTMÓW OBLICZANIA WIELOMIANÓW SYMETRYCZNYCH ORA WRACANIA MACIERZY SPECJALNYCH W ANALIZIE WŁASNOŚCI UKŁADÓW NAMICZNYCH DOWOLNEGO RZĘDU	Z .66		
	6.1.	Wprowadzenie do problematyki jakościowych własności układów dynamicznych	66		
	6.2.	Cel rozdziału	67		
	6.3.	Metodologia zastosowania algorytmów numerycznych w analizie własności układów dynamicznych dowolnego rzędu	68		
	6.4.	Model matematyczny badanego układu dynamicznego	69		
	6.5.	Własności spektralne badanego układu dynamicznego	70		
	6.6.	Transformacja układu dynamicznego	71		
	6.7.	Dekompozycja Jordana macierzy stanu	73		
	6.8.	Numeryczne wyznaczenie odwrotności macierzy podobieństwa formy kanonicznej Jordana macierzy stanu	74		
		6.8.1. Twierdzenie	74		
	6.9.	Analiza własności układu dowolnego rzędu	76		
		6.9.1. Twierdzenie o aproksymacyjnej absolutnej U-sterowalności	76		
		6.9.2. Twierdzenie o aproksymacyjnej relatywnej U-sterowalności	82		
	6.10). Przykład 1	84		
		6.10.1. Przekształcenie równania różniczkowego cząstkowego	85		
		6.10.2. Aproksymacyjna sterowalność bez ograniczeń	87		
		6.10.3. Aproksymacyjna sterowalność przy nieujemnych, stożkowych ograniczeniach na sterowania	۱ 89		
		6.10.4. Podsumowanie przykładu	89		
	6.11	. Przykład 2	90		
		6.11.1. Porównanie złożoności obliczeniowej klasycznej metody badania sterowalności z metodą algorytmicznego odwracania macierzy specjalnych	91		
	6.12	2. Podsumowanie rozdziału	91		
7.	NIII	MERYCZNE WYZNACZANIE ZBIORÓW OSIAGALNYCH WYBRANEJ KLASY UKŁADÓW	V		
	DYN	NAMICZNYCH	.93		
	7.1.	Cel rozdziału	93		
	7.2.	Znaczenie metod analizy numerycznej w problemach rozdziału	93		

	7.3. Podstawy teoretyczne badania zbiorów osiągalnych	
	7.3.1. Definicja ([116] s. 761)	
	7.3.2. Twierdzenie ([116] s. 761)	94
	7.4. Model matematyczny badanego układu dynamicznego	95
	7.5. Zastosowana w rozdziale metodologia badań numerycznych	
	7.6. Dyskretyzacja badanego układu dynamicznego metodą prostych	96
	7.7. Własności spektralne macierzy stanu układu dynamicznego	
	7.8. Konstrukcja finalnego algorytmu	
	7.8.1. Finalny algorytm analizy zbiorów osiągalnych	
	7.9. Testy numeryczne	
	7.9.1. Wymuszenie eksponencjalnie ważone	
	7.9.2. Wymuszenie brzegowe	
	7.10. Podsumowanie	
8.	PODSUMOWANIE ROZPRAWY	
BI	BLIOGRAFIA	109
ST	RESZCZENIE	118
Ав	STRACT	121

CONTENTS

1.	PREFACE	11
	1.1. Numerical Analysis in the PAN Computer Science Committee analysis	. 12
	1.2. Objective of the monograph	.12
	1.3. Monograph problems survey	.12
	1.3.1. Algorithm for the diophantine equation solving	.12
	1.3.2. Application of the diophantine equations solving algorithm in the research of attainable sets of the selected class of dynamical systems	. 13
	1.3.3. Numerical algorithm for calculation of a certain class of symmetric polynomials	.13
	1.3.4. Algorithm for inverting the confluent Vandermonde matrix	. 14
	1.3.5. Application of numerical algorithms in the arbitrary order dynamical systems	.14
	1.3.6. Numerical determining of the attainable sets of dynamical systems with both side constrained force function.	.14
2.	NUMERICAL ANALYSIS OF CERTAIN CLASS OF THE DIOPHANTINE EQUATIONS	15
	2.1. Introduction to the diophantine equations domain	. 15
	2.2. Source of the nonlinear diophantine equations in question	. 17
	2.3. Selected problems of the algorithms computational complexity	.18
	2.4. Errors of numerical calculations	. 19
	2.4.1. Wilkinson's lemma ([133])	. 20
	2.5. Diophantine problem formulation	. 20
	2.6. Construction of the algorithm for the nonlinear diophantine equation solving	.21
	2.6.1. Case 1: $S(\vec{s}_2, \vec{r}_2) \ge 0$.23
	2.6.2. Case 2: $S(\vec{s}_2, \vec{r}_2) < 0$.24
	2.6.3. Simplification of the parameter iteration range.	.25
	2.7. Symmetric solutions elimination	.26
	2.7.1. Case 1: $p_k > 0, k = 2,, n$.28
	2.7.2. Case 2: $p_k = 0, k = 2,, n$. 30
	2.8. Final algorithm	. 30
	2.9. Computational complexity of the algorithm 2.8	.31

	2.10. Determining of the n-ths multiplicities generating equal sums of the squares	32
	2.11. Analysis of calculation errors	33
	2.12. Example of the algorithm 2.8 execution	33
	2.13. Practical test of the algorithm 2.8 effectiveness	35
	2.14. Summary	36
3.	APPLICATION OF THE DIOPHANTINE EQUATIONS SOLVING ALGORITHMS TO THE RESEARCH OF ATTAINABLE SETS OF THE SELECTED DYNAMICAL SYSTEMS	37
	3.1. Introduction	37
	3.2. Explanation of the terminology used in the chapter	37
	3.3. Introduction to the issues of the parabolic-type distributed parameter dynamical systems	38
	3.3.1. General form of the second order linear partial differential equations	38
	3.3.2. Physical importance of the parabolic-type partial differential equations	38
	3.4. Model of the analyzed parabolic-type infinite dimensional system	39
	3.5. Numerical analysis of the parabolic-type infinite dimensional system properties3.5.1. System transformation	40
	3.5.2. Synthesis of the numerical algorithm for the attainable sets examination of the analyzed system	42
	3.5.3. Algorithm	43
	3.5.4. Example of the parabolic system numerical analysis	44
	3.6. Summary	46
4.	EFFECTIVE NUMERICAL ALGORITHM FOR CALCULATION OF A CERTAIN CLASS OF SYMMETRICAL FUNCTIONS	47
	4.1. Introduction	
	4.2 Elementary symmetric polynomials notion	47
	4.3 Algorithm for the elementary symmetric polynomials calculation	48
	4.4 Computational complexity of the algorithm	49
	4.5 Example of the algorithm execution	49
	4.6. Summary	
5.	ALGORITHM FOR INVERTING A CERTAIN CLASS OF SPECIAL MATRICES	52
	5.1. Introduction	52
	5.2. Confluent Vandermonde matrix notion	52
	5.2.1. Confluent Vandermonde matrix definition ([31, 82])	53
	5.2.2. Confluent Vandermonde matrix examples	53
	5.3. Analytical method for the confluent Vandermonde matrix inversion	54
	5.3.1. Theorem ([31] s. 1543)	54
	5.4. Synthesis of the confluent Vandermonde matrix inversion numerical algorithm5.4.1. Theorem [82]	55 55

	5.5. Determining the input values for the confluent Vandermonde matrix inversion algorithm.	56
	5.6. Final algorithms for the confluent Vandermonde matrix inversion	57
	5.6.1. Algorithm for the last column of the searched inverse matrix calculation	58
	5.6.2. Algorithm for the remaining columns of the searched inverse matrix calculation.	61
	5.7. Computational complexity of the confluent Vandermonde matrix inversion algorithm	62
	5.8. Example of the algorithm execution	63
	5.9. Summary	65
6.	APPLICATION OF THE SYMMETRIC POLYNOMIALS CALCULATION AND SPECIAL MATRICES INVERSION ALGORITHMS IN THE ARBITRARY ORDER DYNAMICAL SYSTEM ANALYSIS	1S 66
	6.1. Introduction to the qualitative properties of the dynamical systems domain	66
	6.2. Objective of the chapter	67
	6.3. Methodology of the numerical algorithms application to the research of the arbitrary order dynamical systems properties	68
	6.4. Mathematical model of the analyzed dynamical system	69
	6.5. Spectral properties of the analyzed dynamical system	70
	6.6. Dynamical system transformation	71
	6.7. Jordan decomposition of the state matrix	73
	6.8. Numerical determining of the inversion of the Jordan canonical form matrix of the state matrix	74
	6.8.1. Theorem	74
	6.9. Arbitrary order system analysis	76
	6.9.1. Theorem for the approximate absolute U-controllability	76
	6.9.2. Theorem for the approximate relative U-controllability	82
	6.10. Example 1	84
	6.10.1. Transformation of the partial differential equation	85
	6.10.2. Onconstrained approximate controllability with non-negative cone type controls	/ 6
	6.10.4. Example summary	89
	6 11 Example 2	90
	6.11.1. Comparison of the classical method of the controllability verification computation complexity with the algorithmic method of the special matrices inversion	mal 91
	6.12. Summary	91
7.	NUMERICAL DETERMINING THE ATTAINABLE SETS OF THE SELECTED CLASS OF DYNAMICAL SYSTEMS	93
	7.1. Objective of the chapter	
	7.2. Importance of the numerical analysis in the chapter problems	93
	7.3. Theoretical basis for the research of the attainable sets	94

	7.3.1. Definition ([116] s. 761)	94	
	7.3.2. Theorem ([116] s. 761)	94	
	7.4. Mathematical model of the analyzed dynamical system	95	
	7.5. Numerical research paradigm applied in the chapter	95	
	7.6. Discretization of the analyzed dynamical system by the line method		
	7.7. Spectral properties of the analyzed dynamical system	97	
	7.8. The final algorithm construction		
	7.8.1. The final algorithm	99	
	7.9. Numerical tests		
	7.9.1. Exponentially weighted excitation	99	
	7.9.2. Border excitation	103	
	7.10. Summary		
8.	SUMMARY	108	
BI	BLIOGRAPHY	109	
SUMMARY			
Ab	STRACT	121	

1. WPROWADZENIE

Obserwowany w ostatnich dekadach rozwój systemów komputerowych powoduje systematyczny wzrost znaczenia metod numerycznych w obliczeniach inżynierskich i naukowych. Tradycyjne metody analityczne posiadają zalety, do których można zaliczyć m.in.:

- uniwersalność, tj. niezależność uzyskiwanych wyników od konkretnych wartości danych wejściowych,
- dokładność, tj. wyniki analityczne nie są obciążone błędami metod numerycznych,
- zwartość wyników wynikającą z faktu, iż dla konkretnych problemów naukowo-inżynierskich można znaleźć prostszą metodę analityczną dla konkretnego przypadku niż w przypadku zastosowania ogólnej metody numerycznej,
- możliwość stosowania abstrakcyjnego zapisu np. za pomocą analizy funkcjonalnej.
 Z drugiej strony analityczne podejście do problemów obliczeniowych posiada wady:
- klasę problemów możliwych do rozwiązania metodami analitycznymi, tzn. problemów, których rozwiązanie można przedstawić za pomocą skończonej superpozycji funkcji elementarnych, jest ograniczona; do przykładów można zaliczyć m.in. równanie algebraiczne stopnia n-tego, równania nieliniowe, całki przestępne itd.,
- dla wybranych złożonych układów fizycznych problemem jest zbudowanie modelu metodami czysto analitycznymi; przykładem pomocnej tu metody numerycznej jest metoda elementów skończonych.

Problemy te powodują konieczność zastosowania do tych zagadnień metod numerycznych. Jednocześnie obserwuje się stały wzrost mocy obliczeniowej komputerów. Powoduje to rosnący udział podejścia numerycznego w pracach i badaniach naukowo-technicznych, gdyż możliwe staje się zastosowanie w rozsądnym czasie i przy akceptowalnych kosztach algorytmów o złożonościach, które jeszcze niedawno wykluczały ich praktyczne zastosowanie w danym problemie.

1.1. Analiza numeryczna w analizie Komitetu Informatyki PAN

Szczegółową analizę metodologii numerycznej oraz jej klasyfikację w dyscyplinie informatyki przeprowadziła Sekcja Nauk Obliczeniowych Komitetu Informatyki PAN w raporcie [43]. Wskazano w nim, że symulacje komputerowe w połączeniu z tradycyjnymi narzędziami matematyki oraz prawami fizyki pozwalają na znaczne poszerzenie obszaru ich zastosowań na nauki przyrodnicze, techniczne i społeczne.

1.2. Cel rozprawy

Celem rozprawy jest przedstawienie nowych algorytmów numerycznych wraz z zastosowaniami w badaniach zbiorów osiągalnych dla wybranych klas układów dynamicznych.

1.3. Przegląd problemów rozprawy

Tematem wiodącym niniejszej rozprawy jest wprowadzenie algorytmów numerycznych do badań problemów, które dotychczas były analizowane metodami czysto analitycznymi. Wyniki przedstawione w niniejszej rozprawie podsumowują wyniki badań autora zamieszczone w publikacjach [81-87]. W rozprawie przedstawiono następujące nowe algorytmy:

- algorytm rozwiązujący wybraną klasę równań diofantycznych (rozdział 2),
- algorytm obliczania wielomianów symetrycznych podstawowych (rozdział 4),
- algorytm odwracania konfluentnych macierzy Vandermonde'a (rozdział 5).

Zastosowano je w analizie następujących problemów, dotyczących układów dynamicznych:

- wyznaczania zbiorów osiągalnych wybranej klasy układów dynamicznych określonych w wielościanie (rozdział 3),
- badania własności układów dowolnego rzędu (rozdział 6),
- wyznaczania zbiorów osiągalnych wybranej klasy układów dynamicznych przy uwzględnieniu obustronnych ograniczeń na funkcję wymuszającą (rozdział 7).

1.3.1. Algorytm rozwiązywania równania diofantycznego

W niniejszej rozprawie w rozdziale 2 skoncentrowano się na następującym równaniu diofantycznym:

$$\frac{i_1^2}{m_1} + \dots + \frac{i_k^2}{m_k} + \dots + \frac{i_n^2}{m_n} = \frac{j_1^2}{m_1} + \dots + \frac{j_k^2}{m_k} + \dots + \frac{j_n^2}{m_n}$$
(1.1)

gdzie: $n \ge 2$, $m_1, m_2, ..., m_n \in Z_+$ są danymi dodatnimi stałymi całkowitymi, natomiast $i_k, j_k \in Z_+$ są szukanymi niewiadomymi. Celem jest znalezienie algorytmu, znajdującego wszystkie niesymetryczne i nietrywialne rozwiązania równania (1.1) w liczbach całkowitych dodatnich, nie większych od ustalonej stałej całkowitej. Znalezieniu takiego algorytmu jest poświęcony rozdział 2. Wyniki przedstawione w rozdziale 2 opierają się na rezultatach z publikacji [83].

1.3.2. Zastosowanie algorytmu rozwiązywania równań diofantycznych w wyznaczaniu zbiorów osiągalnych wybranej klasy układów dynamicznych

Znalezienie zbioru rozwiązań równania diofantycznego (1.1) jest konieczne w analizie zbiorów osiągalnych układów dynamicznych, w których modelu matematycznym można wyróżnić następujący operator Laplace'a:

$$Ax(z) = \frac{\partial^2 x(z,t)}{\partial z_1^2} + \dots + \frac{\partial^2 x(z,t)}{\partial z_k^2} + \dots + \frac{\partial^2 x(z,t)}{\partial z_n^2}, \quad x(z) \in D(A)$$
(1.2)

$$D(A) = \left\{ x(z) \in L^{2}(D) : Ax(z) \in L^{2}(D), \quad x(z,t) = 0 \Big|_{z \in \Gamma} \right\}$$
(1.3)

gdzie obszar D jest n-wymiarowym wielościanem. Jednym z czynników, wpływającym na własności układów dynamicznych, w których opisie matematycznym występuje operator Laplace'a (1.2), (1.3), są krotności jego wartości własnych. W celu określenia krotności wartości własnych w niniejszej rozprawie wykorzystano algorytmy rozwiązywania równania diofantycznego (1.1). Wyniki przedstawione w rozdziale 3 opierają się na rezultatach z publikacji [83].

1.3.3. Algorytm numeryczny obliczania pewnej klasy wielomianów symetrycznych

Rozdział 4 jest poświęcony przedstawieniu nowego, efektywnego algorytmu obliczania wielomianów symetrycznych podstawowych. Same wielomiany symetryczne podstawowe są zdefiniowane za pomocą następującej formuły:

$$\begin{cases} w_1^{(n)}(\lambda_1,...,\lambda_n) = \lambda_1 + \lambda_2 + ... + \lambda_n \\ w_2^{(n)}(\lambda_1,...,\lambda_n) = \lambda_1\lambda_2 + \lambda_1\lambda_3 + ... + \lambda_1\lambda_n + ... + \lambda_{n-1}\lambda_n \\ w_3^{(n)}(\lambda_1,...,\lambda_n) = \lambda_1\lambda_2\lambda_3 + \lambda_1\lambda_2\lambda_4 + ... + \lambda_{n-2}\lambda_{n-1}\lambda_n \\ \\ w_n^{(n)}(\lambda_1,...,\lambda_n) = \lambda_1\lambda_2 \cdot ... \cdot \lambda_n \end{cases}$$
(1.4)

Pomimo elementarnej budowy wzoru definicyjnego (1.4) ich efektywne obliczenie na drodze algorytmicznej jest zagadnieniem nietrywialnym. Zaproponowany algorytm obliczania wielomianów symetrycznych podstawowych znalazł zastosowanie w wyznaczaniu odwrotności konfluentych macierzy Vandermonde'a oraz badaniu własności dynamicznych układów dowolnego rzędu, czemu poświecono kolejne rozdziały.

1.3.4. Algorytm odwracania konfluentnych macierzy Vandermonde'a

W rozdziale 5 skonstruowano nowy algorytm odwracania konfluentych macierzy Vandermonde'a. Ich budowa różni się tym od klasycznej postaci macierzy Vandermonde'a, że w kolumnach konfluentnej macierzy Vandermonde'a oprócz kolejnych potęg różnych pierwiastków znajdują się pochodne tychże kolumn. W konstrukcji algorytmu korzysta się z przedstawionego w poprzednim rozdziale algorytmu obliczania wielomianów symetrycznych podstawowych. Sam algorytm znajduje zastosowanie w analizie własności liniowych układów dynamicznych o dowolnym stopniu pochodnych względem czasu, czemu poświęcono kolejny rozdział. Wyniki przedstawione w rozdziale 5 korzystają z rezultatów opublikowanych w artykule [82].

1.3.5. Zastosowanie algorytmów numerycznych w analizie własności dynamicznych układów dynamicznych dowolnego rzędu

Myślą wiodącą rozdziału 6 jest zastosowanie algorytmów:

- obliczania wielomianów symetrycznych podstawowych,
- odwracania konfluentnych macierzy Vandermonde'a,

w analizie wybranych własności układów dynamicznych. Główną innowacją rozdziału jest przeprowadzenie badań dla dowolnego, n-tego stopnia badanego układu dynamicznego względem czasu. Dotychczas w literaturze można znaleźć pozycje, które zajmowały się jedynie szczególnymi przypadkami tych problemów. Wyniki przedstawione w rozdziale 6 stanowią podsumowanie rezultatów autora z publikacji [85-87].

1.3.6. Numeryczna analiza zbiorów osiągalnych układów dynamicznych przy obustronnym ograniczeniu funkcji wymuszającej

Przedmiotem rozdziału 7 jest numeryczne wyznaczanie zbiorów osiągalnych układów dynamicznych modelowanych za pomocą równań różniczkowych cząstkowych typu parabolicznego. Innowacją przedstawionych badań jest założenie realistycznych, obustronnych (tj. odgórnych i oddolnych) ograniczeń funkcji wymuszającej. Odpowiedni dobór metod dyskretyzacji pozwolił na zastosowanie kryteriów badania zbiorów osiągalnych przy obustronnych ograniczeniach na funkcję wymuszającą, podanych przez Schmitendorfa w pracy [116] dla równań różniczkowych zwyczajnych, do układu dynamicznego modelowanego równaniem różniczkowym cząstkowym. Wykonanie badań zbiorów osiągalnych przy realistycznych ograniczeniach na funkcję wymuszającą było możliwe dzięki zastosowaniu odpowiednio dobranej metody numerycznej rozwiązywania równań różniczkowych cząstkowych. Rozdział stanowi ilustrację połączenia w całość metod numerycznych i klasycznych wyników analitycznych. Wyniki przedstawione w rozdziale 7 stanowią podsumowanie rezultatów z publikacji [81, 84].

2. ANALIZA NUMERYCZNA WYBRANEJ KLASY RÓWNAŃ DIOFANTYCZNYCH

2.1. Wprowadzenie do problematyki równań diofantycznych

W literaturze równanie diofantyczne definiuje się jako algebraiczne równanie wielomianowe lub równanie innego rodzaju, którego rozwiązań poszukujemy w zbiorze liczb całkowitych. Problem diofantyczny na ogół ma mniej równań niż niewiadomych zmiennych. Jego rozwiązanie polega na znalezieniu wszystkich n liczb całkowitych, spełniających każde z równań problemu diofantycznego. Nazwa wywodzi się od Diofanatesa z Aleksandrii, hellenistycznego matematyka żyjącego w III wieku n.e., który badał takie równania. Dopiero w XX wieku sformułowano ogólną teorię równań diofantycznych ([4, 140]).

Analiza równań diofantycznych stara się znaleźć odpowiedzi na następujące problemy:

- czy dane równanie posiada rozwiązanie,
- czy liczba rozwiązań jest skończona czy nieskończona,
- czy wszystkie rozwiązania mogą być znalezione na drodze teoretycznej,
- czy jest możliwe praktyczne znalezienie pełnego zbioru rozwiązań?

Jako przykłady często rozważanych w literaturze równań diofantycznych można przedstawić:

- Równanie algebraiczne ax + by = c (a, b, c ∈ Z oznaczają dane liczby całkowite) jest równaniem diofantycznym liniowym. Jest to, obok równania rozważanego w Wielkim Twierdzeniu Fermata, najbardziej znane równanie diofantyczne. Ma rozwiązanie wtedy i tylko wtedy, gdy liczba c dzieli największy wspólny dzielnik liczb a i b. Analizą tego równania zajmują się autorzy: Nathanson [73] s. 37-42; Manin [60] s. 22; Yan [140] s. 54-57; Rosen, Michaels, Gross, Grossman, Shier [110] s. 314-315.
- Równanie x² ny² = 1 (n > 0), zwane równaniem Pella. Dla liczby n będącej kwadratem liczby całkowitej nie ma ono rozwiązań, zaś dla n niebędącej kwadratem liczby całkowitej ma nieskończenie wiele rozwiązań. Równaniem diofantycznym Pella zajmuje się wiele książek: Baker [4] s. 74-77; Manin [60] s. 28; Yan [140] s. 57-63; Rosen, Michaels, Gross, Grossman, Shier [110] s. 318-321.

• W literaturze były także badane równania diofantyczne, w których występowała silnia. Przykładem może być równanie:

$$x_1 !+ x_2 !+ \dots + x_n != x_{n+1} !, \quad x_i \in N$$
(2.1)

Można udowodnić, że dla $n \ge 2$ równanie (2.1) ma przynajmniej jedno rozwiązanie, ale liczba rozwiązań jest skończona, natomiast dla n = 1 równanie to ma nieskończenie wiele rozwiązań. Równanie analizuje Sandor w pracy [114] s. 68-69.

- Równanie x^y = y^x ma w liczbach naturalnych dokładnie dwa nietrywialne rozwiązania (x ≠ y): x = 4, y = 2 oraz x = 2, y = 4. Należy ono do rzadko analizowanej grupy wykładniczych równań diofantycznych. Jego analizą zajmuje się Sandor [114] s. 93-95.
- Analizuje się także układy liniowych równań diofantycznych o postaci AX = B. Sam układ równań ma analogiczną postać do, znanego z algebry, klasycznego układu równań liniowych. Różnica polega tu na tym, że wszystkie elementy zarówno danych macierzy A, B, jak i wektora niewiadomych X należą do zbioru liczb całkowitych. Wiele problemów dotyczących układów liniowych równań diofantycznych można znaleźć w monografiach: Manin [60] s. 22-24; Rosen, Michaels, Gross, Grossman, Shier [110] s. 314-315. Przykładem na to, że tematyka układów równań diofantycznych jest intensywnie badana do dziś, może być artykuł Hernando, Ledesma, Laita [29] z 2009 roku.
- Najczęściej są badane algebraiczne równania diofantyczne, jednak były także badane równania, w których występują inne funkcje elementarne. Przykładem może być następujące równanie [114]:

$$arctg \frac{1}{x_1} + arctg \frac{1}{x_2} + \dots + arctg \frac{1}{x_n} = \frac{\pi}{4}$$

gdzie niewiadome są dodatnimi liczbami całkowitymi. Odnośnie do tego równania można udowodnić następujące fakty:

- dla każdego n równanie to posiada przynajmniej jedno rozwiązanie,
- liczba rozwiązań dla danego n jest skończona.

Ponadto, wszystkie rozwiązania tego równania wyrażają się poprzez podstawowe działania arytmetyczne:

 $x_1 = 1 + 1 + 1^2, ..., x_{n-1} = 1 + (n-1) + (n-1)^2, x_n = n$

Równaniu temu jest poświęcony rozdział w monografii Sandor [114] s. 96-100.

Nie sposób nie wspomnieć tu równania diofantycznego o postaci xⁿ + yⁿ = zⁿ, gdzie n≥2 jest daną stałą całkowitą. Jest ono przedmiotem Wielkiego Twierdzenia Fermata. Twiedzenie to mówi, że nieliniowe równanie diofantyczne xⁿ + yⁿ = zⁿ nie ma rozwiązań w liczbach całkowitych dla n>2. Dla n=2 ma ono nieskończenie wiele rozwiązań,

tworzących tzw. trójki pitagorejskie (np. x = 3, y = 4, z = 5 daje $x^2 + y^2 = 3^2 + 4^2 = 25 = 5^2 = z^2$). Problem ten jest przedstawiony w następujących książkach: Baker [4] s. 84-87; Rosen, Michaels, Gross, Grossman, Shier [110] s. 317-318; Manin [60] s. 341-393; Paulo [75].

W polskiej literaturze tematyką równań diofantycznych zajmował się Wacław Sierpiński, który wydał książki [122-125]. Wśród pozycji innych autorów, przetłumaczonych na język polski, można wymienić książki Narkiewicz [72], Paulo [75], Marzantowicz [62], Koblitz [53].

W literaturze światowej problem równań diofantycznych jest poruszany w następujących monografiach: Nathanson [73] s. 37-42 (równanie liniowe); Baker [4] s. 74-91 (równania: Pella, Thue, Mordella, Fermata, Catallana); Manin [60] s. 22-49 (równanie liniowe, układ równań liniowych, równania stopnia 2 i 3, Pella); Yan [140] s. 52-62; Moser [70] s. 53-58; Rosen, Michaels, Gross, Grossman, Shier [110] s. 314-322 (równanie liniowe, problem trójek pitagorejskich, równanie Fermata, Pella, Bacheta i Catalana). Książka Sandora [114] z 2002 roku na s. 56-121 analizuje 21 typów równań diofantycznych, w tym równania wymierne, nieliniowe, wykładnicze oraz równania diofantyczne, których niewiadome są argumentami funkcji przestępnych.

2.2. Geneza analizowanej klasy nieliniowych równań diofantycznych

Rezultaty przedstawione w rozdziale 2 zostały opublikowane w artykule [83]. Niniejszy rozdział jest poświęcony znalezieniu algorytmu, rozwiązującego w efektywny sposób równanie diofantyczne o postaci:

$$\frac{i_1^2}{m_1} + \dots + \frac{i_k^2}{m_k} + \dots + \frac{i_n^2}{m_n} = \frac{j_1^2}{m_1} + \dots + \frac{j_k^2}{m_k} + \dots + \frac{j_n^2}{m_n}$$
(2.2)

Postać rozważanego równania jest ściśle związana z postacią spektralną operatora różniczkowego Laplace'a. Mianowicie, wartości własne operatora Laplace'a określonego w n-wymiarowym wielościanie, przy zerowych warunkach brzegowych typu Dirichleta, wyrażają się wprost przez następującą formułę analityczną [9]:

$$\lambda_{i_1...i_n} = -\pi^2 \left(\frac{i_1^2}{a_1^2} + ... + \frac{i_k^2}{a_k^2} + ... + \frac{i_n^2}{a_n^2} \right) \quad k = 1, 2, ..., n, \quad i_k = 1, 2, 3, ...$$
(2.3)

gdzie przez liczby a_k wyrażają się długości krawędzi obszaru określoności operatora różniczkowego Laplace'a. W przypadku gdy kwadraty tych długości należą do zbioru liczb całkowitych, do analizy krotności wartości własnych (2.3) można bezpośrednio zastosować równanie diofantyczne (2.2), gdyż stała proporcjonalności $-\pi^2$ nie ma wpływu na badane krotności.

Analiza krotności wartości własnych (2.3) sprowadza się do odpowiedzi na pytania, czy następująca kombinacja liniowa S kwadratów liczb całkowitych $i_1,...,i_n$:

$$S(i_1,...,i_n) = \frac{i_1^2}{m_1} + \dots + \frac{i_k^2}{m_k} + \dots + \frac{i_n^2}{m_n}, \quad m_k \in N_+$$
(2.4)

- może przyjmować te same wartości, dla różnych n-tek całkowitych *i*₁,...,*i*_n,
- ile istnieje różnych n-tek całkowitych i₁,...,i_n, którym odpowiada ta sama wartość kombinacji liniowej S(i₁,...,i_n) określonej przez (2.4) ?

Szerzej ten problem został opisany w rozdziale 3.

Syntezę algorytmu rozwiązywania równania diofantycznego (2.2) można podzielić na następujące główne etapy:

- parametryzacja niewiadomych,
- sprowadzenie równania do postaci bez kwadratów poprzez zamianę niewiadomych zmiennych,
- uzyskanie końcowego algorytmu przy wykorzystaniu własności podzielności liczb,
- wyznaczenie przedziałów, w jakich powinny iterować parametry,
- eliminacja rozwiązań trywialnych i symetrycznych,
- oszacowanie błędu obliczeń oraz złożoności czasowej uzyskanego algorytmu.

2.3. Wybrane problemy złożoności obliczeniowej algorytmów

Zasadnicze znaczenie dla porównywania jakości różnych algorytmów, w tym numerycznych, ma ich złożoność obliczeniowa. Jest ona ściśle związana z czasem wykonania danego algorytmu. Jednak podanie jedynie czasu wykonania algorytmu jest niewystarczające dla obiektywnej oceny jego efektywności, gdyż czas jego wykonania zależy m. in. od [1]:

- konfiguracji sprzętowej zastosowanego systemu komputerowego,
- rodzaju użytego języka programowania i kompilatora,
- używanego systemu operacyjnego,
- priorytetu procesu, w którym został uruchomiony algorytm,
- liczby i priorytetów pozostałych wątków uruchomionych w systemie,
- konfiguracji i rozmiaru danych wejściowych.

Dlatego konieczne okazało się wprowadzenie miary złożoności obliczeniowej, która będzie obiektywną miarą czasu wykonania algorytmu w funkcji rozmiaru i konfiguracji danych wejściowych i nie będzie zależała od pozostałych, wymienionych wcześniej czynników. Różnice w czasie wykonania algorytmów są szczególnie widoczne dla dużej liczby danych wejściowych, dlatego najczęściej stosowaną miarą efektywności algorytmów jest złożoność asymptotyczna. Wynika to z faktu, iż dla dużej liczby danych przybliżenie

asymptotyczne jest szczególnie bliskie złożoności dokładnej. Najczęściej stosowanym zapisem przeznaczonym do wyrażania złożoności asymptotycznej jest notacja "wielkie O". Definiuje się ją następująco [1]:

Definicja 2.1

Nieujemna funkcja $f(n): Z_+ \to Z_+$ jest rzędu funkcji $g(n): Z_+ \to Z_+$, f(n) = O[g(n)], jeżeli istnieje stała rzeczywista c, taka że:

$$\underset{\substack{c>0\\n_0\in N}}{\exists} \bigvee_{n>n_0} f(n) \leq c \cdot g(n)$$

Notacja "wielkie O" będzie stosowana w dalszej części pracy do szacowania złożoności obliczeniowej wprowadzanych algorytmów numerycznych. Do jej wyznaczania są użyteczne następujące własności [1]:

$$c \cdot O[f(n)] = O[f(n)] \tag{2.5}$$

$$O[f(n)] + O[f(n)] = O[f(n)]$$
(2.6)

$$O\left[O\left[f\left(n\right)\right]\right] = O\left[f\left(n\right)\right] \tag{2.7}$$

$$O[f(n)] \cdot O[g(n)] = O[f(n) \cdot g(n)]$$
(2.8)

2.4. Błędy obliczeń numerycznych

Jak już wspomniano, jedną z zalet obliczeń analitycznych jest dokładność obliczeń. Przeprowadzając obliczenia numeryczne, popełniamy błędy obliczeń, do których należą:

- błędy wejściowe,
- błędy obcięcia,
- błędy metody,
- błędy zaokrągleń.

Jak pokazano w książce [133], wprowadzone w dalszej części niniejszej pracy algorytmy numeryczne są obciążone błędami zaokrągleń. Błędy te wynikają z wykonywania działań na liczbach rzeczywistych, które z konieczności w pamięci komputera są reprezentowane w skończonej postaci typów zmiennoprzecinkowych.

Oznaczmy jako ε parametr charakteryzujący dokładność obliczeniową komputera. Do obliczeń może być wykorzystany typ *double*. Liczby tego typu są zapamiętywane przy podstawie 2 i zajmują 64 bity, z czego 53 jest przeznaczonych na mantysę. Stąd parametr ε charakteryzujący dokładność obliczeń wynosi ([138]):

$$\varepsilon = 2^{-\eta + 1} = 2^{-52} \tag{2.9}$$

gdzie η oznacza liczbę bitów przyjętych do reprezentowania mantysy. Jest oczywiste, iż wykonywanie działań na liczbach niedokładnych powoduje uzyskiwanie niedokładnych

wyników. Podstawową rolę w szacowaniu błędów zaokrągleń, powstałych w wyniku niedokładnej reprezentacji liczby przez komputer, odgrywa lemat Wilkinsona [133]. Służy on do oszacowania błędów zaokrągleń powstających podczas wykonywania pojedynczych działań arytmetycznych na liczbach zmiennoprzecinkowych.

2.4.1. Lemat Wilkinsona ([133])

Błędy zaokrągleń powstające podczas wykonywania działań zmiennopozycyjnych są równoważne zastępczemu zaburzeniu liczb, na których wykonujemy działania. W przypadku pojedynczych działań arytmetycznych otrzymujemy:

$$float(x_1 \pm x_2) = x_1(1 + \varepsilon_1) + x_2(1 + \varepsilon_2)$$
(2.10)

$$float(x_1 x_2) = x_1(1 + \varepsilon_3) x_2 = x_1 x_2(1 + \varepsilon_3)$$
(2.11)

$$float\left(\frac{x_1}{x_2}\right) = \frac{x_1(1+\varepsilon_4)}{x_2} = \frac{x_1}{x_2(1+\varepsilon_5)}$$
(2.12)

gdzie $|\varepsilon_i| \leq |\varepsilon|$, i = 1, 2, ..., 5.

2.5. Sformulowanie problemu diofantycznego

Rozpatrzmy następujące nieliniowe równanie diofantyczne:

$$\frac{i_1^2}{m_1} + \dots + \frac{i_k^2}{m_k} + \dots + \frac{i_n^2}{m_n} = \frac{j_1^2}{m_1} + \dots + \frac{j_k^2}{m_k} + \dots + \frac{j_n^2}{m_n}$$
(2.13)

gdzie:

$$m_1, m_2, \dots, m_n \in Z_+, \ n \ge 2$$
 (2.14)

są danymi stałymi całkowitymi, natomiast zmienne:

 $i_1, \dots, i_k, \dots, i_n \in Z_+$ (2.15)

oraz:

$$j_1, \dots, j_k, \dots, j_n \in Z_+$$
 (2.16)

są szukanymi niewiadomymi, gdzie Z_+ oznacza zbiór liczb całkowitych dodatnich.

Cel poszukiwanego algorytmu jest następujący: znaleźć wszystkie niesymetryczne i nietrywialne *2n*-tki:

$$(i_1, ..., i_k, ..., i_n, j_1, ..., j_k, ..., j_n) \in Z_+^{2n}$$
(2.17)

spełniające równanie (2.13), ograniczone z góry przez daną dodatnią liczbę całkowitą M:

$$\bigvee_{k=1,\dots,n} i_k, j_k \le M, \ M \in Z_+$$

$$(2.18)$$

Poszukiwanie niesymetrycznych rozwiązań oznacza, że odrzucamy takie rozwiązania, które są jedynie zamienione stronami.

2.6. Konstrukcja algorytmu rozwiązania nieliniowego równania diofantycznego

W celu znalezienia algorytmu znajdującego zbiór rozwiązań rozważanego równania diofantycznego (2.13) dogodne jest, aby zapisać je w równoważnej postaci bez kwadratów. W tym celu w pierwszym kroku przekształćmy dane równanie diofantyczne (2.13) do następującej postaci:

$$-\left(j_{1}^{2}-i_{1}^{2}\right)=\sum_{k=2}^{n}\frac{m_{1}}{m_{k}}\left(j_{k}^{2}-i_{k}^{2}\right)$$
(2.19)

Postać (2.19) równania diofantycznego (2.13) jest dogodna do wprowadzenia nowych zmiennych definiowanych przez równość:

$$\begin{cases} s_k = j_k + i_k \\ r_k = j_k - i_k \end{cases}, \ k = 1, ..., n$$
(2.20)

Uwzględniając znaną z algebry tożsamość:

$$j_k^2 - i_k^2 = (j_k - i_k)(j_k + i_k), \quad k = 1, ..., n$$
(2.21)

równanie (2.19) można zapisać w postaci bez kwadratów:

$$-s_1 r_1 = \sum_{k=2}^n \frac{m_1}{m_k} s_k r_k$$
(2.22)

Sprowadzając ułamki występujące w równaniu (2.22) do wspólnego mianownika, otrzymujemy:

$$r_{1} = -\frac{1}{s_{1}} \frac{\sum_{k=2}^{n} m_{1} \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_{n} s_{k} r_{k}}{m_{2} m_{3} \cdot \dots \cdot m_{n}}$$
(2.23)

Z kolei z zależności (2.20) możemy obliczyć, jak wyrażają się szukane zmienne (i_k, j_k) w funkcji parametrów (s_k, r_k) dla k = 2, ..., n:

$$\begin{cases} i_k = \frac{1}{2} (s_k - r_k) \\ j_k = \frac{1}{2} (s_k + r_k) \end{cases}, \quad k = 2, ..., n$$
(2.24)

Podstawiając równanie (2.23) do wzorów (2.24), uzyskujemy formułę (2.25) podającą wprost, jak wyrażają się wartości (i_1, j_1) w funkcji parametrów $s_k, r_k, k = 2,...,n$ oraz parametru s_1 :

$$\begin{cases} i_{1} = \frac{1}{2} \left(s_{1} + \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} m_{1} \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_{n} s_{k} r_{k}}{m_{2} m_{3} \cdot \dots \cdot m_{n}} \right) \\ j_{1} = \frac{1}{2} \left(s_{1} - \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} m_{1} \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_{n} s_{k} r_{k}}{m_{2} m_{3} \cdot \dots \cdot m_{n}} \right)$$

$$(2.25)$$

Teraz wyznaczmy postać ograniczeń dla parametrów $s_k, r_k, k = 2,...,n$, występujących w równaniu (2.25). W tym celu, po pierwsze, należy uwzględnić fakt, że wszystkie niewiadome $(i_k, j_k), k = 1,...,n$, na podstawie założenia (2.17) muszą należeć do zbioru liczb całkowitych dodatnich. Ponadto, uwzględniając fakt, że szukamy niewiadomych o wartościach maksymalnych określonych nierównością (2.18), z wykorzystaniem równania (2.24) parametry $s_k, r_k, k = 2,...,n$ muszą spełniać następujący układ podwójnych nierówności:

$$\begin{cases} 1 \le \frac{1}{2} (s_k - r_k) \le M \\ 1 \le \frac{1}{2} (s_k + r_k) \le M \end{cases}, \quad k = 2, ..., n \end{cases}$$
(2.26)

Bezpośrednio z nierówności (2.26) uzyskujemy:

$$\begin{cases} r_k + 2 \le s_k \le 2M + r_k \\ -r_k + 2 \le s_k \le 2M - r_k \end{cases}, \quad k = 2, ..., n$$
(2.27)

Nierówności w ciągu (2.27) określają w kartezjańskim układzie współrzędnych $R_k S_k$ zbiór punktów zawartych w prostokącie z wierzchołkami o współrzędnych (0,2), (M-1, M+1), (0, 2M), (-M+1, M+1). Pozostały do określenia ograniczenia dla parametru s_1 . Uwzględniając ograniczenia (2.17) oraz (2.18) oraz równanie (2.25), uzyskujemy podwójną nierówność:

$$1 \le \frac{1}{2} \left(s_1 \pm \frac{1}{s_1} \frac{\sum_{k=2}^n m_1 \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_n s_k r_k}{m_2 m_3 \cdot \dots \cdot m_n} \right) \le M$$
(2.28)

którą można przekształcić do postaci czterech nierówności kwadratowych względem parametru s_1 :

$$0 \le m_2 m_3 \cdot \dots \cdot m_n s_1^2 - 2m_2 m_3 \cdot \dots \cdot m_n s_1 \pm \sum_{k=2}^n m_1 \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_n s_k r_k$$
(2.29)

$$m_2 m_3 \cdot \dots \cdot m_n s_1^2 - 2M m_2 m_3 \cdot \dots \cdot m_n s_1 \pm \sum_{k=2}^n m_1 \cdot \dots \cdot m_{k-1} m_{k+1} \cdot \dots \cdot m_n s_k r_k \le 0$$
(2.30)

Dla uproszczenia notacji wprowadźmy nową zmienną $S(\vec{s}_2, \vec{r}_2)$:

$$S(\vec{s}_{2}, \vec{r}_{2}) = \sum_{k=2}^{n} m_{1} \cdot ... \cdot m_{k-1} m_{k+1} \cdot ... \cdot m_{n} s_{k} r_{k}$$
(2.31)

gdzie wektory \vec{s}_2 oraz \vec{r}_2 są dane przez:

$$\vec{s}_2 = (s_2, .., s_n), \ \vec{r}_2 = (r_2, .., r_n)$$
 (2.32)

W celu rozwiązania nierówności (2.29), (2.30) rozróżnijmy 2 przypadki, wymienione w kolejnych podpunktach.

2.6.1. Przypadek 1: $S(\vec{s}_2, \vec{r}_2) \ge 0$

Wyróżnik pierwszych dwóch nierówności (2.29) przy użyciu zmiennych (2.31) ma postać:

$$\Delta_1^{\pm} = 4m_2^2 m_3^2 \cdot \dots \cdot m_n^2 \pm 4m_2 m_3 \cdot \dots \cdot m_n S\left(\vec{s}_2, \vec{r}_2\right)$$
(2.33)

Z zależności (2.33) wynika, że $\Delta_1^+ > 0$ (z założenia w przypadku 1 mamy $S(\vec{s}_2, \vec{r}_2) \ge 0$), ale znak parametru Δ_1^- jest nieznany. Jeżeli $\Delta_1^- \ge 0$, wtedy rozwiązaniem nierówności (2.29) są następujące dwa przedziały zmienności dla zmiennej s_1 :

$$P_{1} = R \setminus \left(1 - \sqrt{1 - \frac{S\left(\vec{s}_{2}, \vec{r}_{2}\right)}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; 1 + \sqrt{1 - \frac{S\left(\vec{s}_{2}, \vec{r}_{2}\right)}{m_{2}m_{3} \cdot \dots \cdot m_{n}}} \right)$$
(2.34)

$$P_{2} = R \setminus \left(1 - \sqrt{1 + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot ... \cdot m_{n}}}; 1 + \sqrt{1 + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot ... \cdot m_{n}}} \right)$$
(2.35)

gdzie R to zbiór liczb rzeczywistych. Ze wzorów (2.34), (2.35) wynika następująca inkluzja:

$$P_2 \subset P_1 \tag{2.36}$$

Nierówności (2.29) muszą być spełnione jednocześnie, gdyż wynikają one z ograniczenia (2.28). Zatem, rozwiązaniem nierówności (2.29) dla $\Delta_1^- \ge 0$ jest iloczyn mnogościowy zbiorów P_1 , P_2 . Z zależności (2.36) wynika:

$$P_1 \cap P_2 = P_2 \tag{2.37}$$

Rozwiązaniem w tym przypadku, dla $\Delta_1^- \ge 0$, jest przedział (2.35). Jeżeli $\Delta_1^- < 0$, to nierówność (2.29) wzięta ze znakiem + jest spełniona dla wszystkich liczb rzeczywistych, więc będzie także spełniona dla liczb całkowitych i przedział (2.35) pozostaje rozwiązaniem nierówności (2.29) także w tym przypadku. Podsumowując, w przypadku 1 przedział (2.35) jest zbiorem rozwiązań nierówności (2.29). Oczywiście interesują nas jedynie punkty całkowite przedziału P_2 (równanie (2.20) oraz założenie (2.17)).

Przejdźmy do nierówności (2.30). Ich wyznacznik jest dany wzorem:

$$\Delta_{2}^{\pm} = 4M^{2}m_{2}^{2}m_{3}^{2}\cdot\ldots\cdot m_{n}^{2} \pm 4m_{2}m_{3}\cdot\ldots\cdot m_{n}S\left(\vec{s}_{2},\vec{r}_{2}\right)$$
(2.38)

Z zależności (2.30) wynika, że dla $\Delta_2^- < 0$ ograniczenie (2.28) nie jest spełnione. Dla przeciwnego przypadku $\Delta_2^- \ge 0$ z nierówności (2.30) wynikają dwa przedziały:

$$P_{3} = \left[M - \sqrt{M^{2} - \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; M + \sqrt{M^{2} - \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}} \right]$$
(2.39)

$$P_{4} = \left[M - \sqrt{M^{2} + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; M + \sqrt{M^{2} + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}} \right]$$
(2.40)

Można zauważyć w przedziałach (2.39), (2.40) następującą implikację:

$$P_3 \subset P_4 \Longrightarrow P_3 \cap P_4 = P_3 \tag{2.41}$$

więc częścią wspólną rozwiązań nierówności (2.30) jest przedział (2.39), przy założeniu $\Delta_2^- \ge 0$.

2.6.2. Przypadek 2: $S(\vec{s}_2, \vec{r}_2) < 0$

Po podobnych obliczeniach, jak w przypadku 1, uzyskujemy następujący przedział zmienności dla zmiennej s_1 , wynikający z nierówności (2.29):

$$P_{5} = R \setminus \left(1 - \sqrt{1 - \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; 1 + \sqrt{1 - \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}} \right)$$
(2.42)

W tym przypadku z dwóch wyróżników danych równaniem (2.38) jedynie wyróżnik Δ_2^+ przyjmuje ujemne wartości. Układ nierówności (2.30) w tym przypadku odpowiada przedziałowi (2.43) dla parametru s_1 oraz warunkowi (2.44), gdzie wyróżnik Δ_2^+ wyraża się równością (2.38):

$$P_{6} = \left[M - \sqrt{M^{2} + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; M + \sqrt{M^{2} + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}} \right]$$
(2.43)

$$\Delta_2^+ \ge 0 \tag{2.44}$$

2.6.3. Uproszczenie przedziałów zmienności parametrów

Celem niniejszego podpunktu jest znalezienie iloczynu mnogościowego przedziałów (2.35), (2.39) oraz (2.42), (2.43). W tym celu udowodnimy następujący lemat:

Lemat 2.1

Następujące nierówności są spełnione:

$$1 - \sqrt{1 - a} < M - \sqrt{M^2 + a} \tag{2.45}$$

dla $0 < a \le 1, M \ge 1$, oraz:

$$1 - \sqrt{1 - a} > M - \sqrt{M^2 + a} \tag{2.46}$$

 $dla -M^2 < a < 0, M \ge 1$, gdzie w obydwu nierównościach (2.45), (2.46) parametr $a \in R$ oraz M jest dodatnią liczbą całkowitą.

νη γενι ασααιτία περοά εαικον

Dowód lematu

Rozpatrzmy następującą funkcję:

$$f(a,M) = \left(1 - \sqrt{1 - a}\right) + \left(\sqrt{M^2 + a} - M\right)$$
(2.47)

Można zauważyć, że następujące nierówności są spełnione:

$$1 - \sqrt{1 - a} > 0, \ dla \ 0 < a \le 1$$
 (2.48)

oraz:

$$\sqrt{M^2 + a} - M > 0, \ dla \ 0 < a \le 1$$
(2.49)

więc:

$$f(a,M) > 0, \ dla \ 0 < a \le 1$$
 (2.50)

Ponadto, dla $-M^2 < a < 0$ zachodzą nierówności:

$$1 - \sqrt{1 - a} < 0, \ \sqrt{M^2 + a} - M < 0 \tag{2.51}$$

więc funkcja f(a, M) przyjmuje wartości ujemne dla $-M^2 < a < 0$. Co więcej, funkcja f(a, M) ma dokładnie jedno miejsce zerowe dla a = 0, a zatem z zależności (2.47)-(2.51) wynikają bezpośrednio nierówności (2.45) i (2.46).

C.N.D.

Wróćmy do uproszczenia przedziałów zmienności parametru s_1 . W tym celu lemat 2.1 może zostać użyty po podstawieniu:

$$a = \frac{S\left(\vec{s}_2, \vec{r}_2\right)}{m_2 m_3 \cdot \dots \cdot m_n} \tag{2.52}$$

Za pomocą nierówności (2.45), (2.46) możemy ostatecznie zredukować przedziały (2.35), (2.39) do jednego przedziału:

$$P_{7} = \left[1 + \sqrt{1 + \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; M + \sqrt{M^{2} - \frac{S(\vec{s}_{2}, \vec{r}_{2})}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}\right]$$
(2.53)

dla:

$$S(\vec{s}_2, \vec{r}_2) \ge 0, \, \Delta_2^- \ge 0$$
 (2.54)

Natomiast przedziały (2.42) i (2.43) redukują się do przedziału:

$$P_{8} = \left[1 + \sqrt{1 - \frac{S\left(\vec{s}_{2}, \vec{r}_{2}\right)}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}; M + \sqrt{M^{2} + \frac{S\left(\vec{s}_{2}, \vec{r}_{2}\right)}{m_{2}m_{3} \cdot \dots \cdot m_{n}}}\right]$$
(2.55)

dla:

$$S(\vec{s}_2, \vec{r}_2) < 0, \, \Delta_2^+ \ge 0$$
 (2.56)

2.7. Eliminacja rozwiązań symetrycznych

W sformułowaniu problemu syntezy algorytmu rozwiązywania badanego nieliniowego równania diofantycznego (punkt 2.5) założyliśmy, że interesują nas jedynie niesymetryczne rozwiązania równania (2.13). Oznacza to, że ze wszystkich par rozwiązań w postaci:

$$\begin{cases} \left(i_{1}^{(1)},...,i_{k}^{(1)},...,i_{n}^{(1)},j_{1}^{(1)},...,j_{k}^{(1)},...,j_{n}^{(1)}\right) \in Z_{+}^{2n} \\ \left(i_{1}^{(2)},...,i_{k}^{(2)},...,i_{n}^{(2)},j_{1}^{(2)},...,j_{k}^{(2)},...,j_{n}^{(2)}\right) \in Z_{+}^{2n} \end{cases}$$

$$(2.57)$$

posiadających własność:

$$i_k^{(1)} = j_k^{(2)}, j_k^{(1)} = i_k^{(2)}, \ k = 1, ..., n$$
 (2.58)

bierzemy pod uwagę jedynie jedną 2n-tkę z pary (2.57).

Celem niniejszego podpunktu jest wybranie z ciągu rozwiązań równania (2.13), którego symetryczna para wyrazów jest dana równaniami (2.24) oraz (2.25), jednego niesymetrycznego ciągu 2n-tek. Każde z rozwiązań równania (2.13) w parze 2n-tek (2.57) można podzielić na dwie grupy zmiennych:

- parę zmiennych (i_1, j_1) , daną wzorem (2.25),
- parę (n-1)-tek, danych wzorem (2.24).

Dla potrzeb eliminacji rozwiązań symetrycznych wprowadźmy nową zmienną pomocniczą p_k , zdefiniowaną następującą:

$$p_{k} = \begin{cases} r_{k}, \ dla \ r_{k} \ge 0 \\ -r_{k}, \ dla \ r_{k} < 0 \end{cases}$$

$$(2.59)$$

Zmienna r_k jest definiowana przez równanie (2.20). Uwzględniając podstawienie (2.59) we wzorze (2.24), uzyskujemy 2^{n-1} wariacji z powtórzeniami ciągów rozwiązań (znaki pod

sumą we wzorze (2.25) są determinowane przez wybór znaku we wzorze (2.24)). Podstawiając zmienną p_k daną wzorem (2.59) do równania (2.24), otrzymujemy układ równości:

$$\begin{cases} i_k = \frac{1}{2} (s_k \mp p_k) \\ j_k = \frac{1}{2} (s_k \pm p_k) \end{cases}, \quad k = 2, ..., n$$
(2.60)

Natomiast para (i_1, j_1) (2.25) po podstawieniu zmiennej p_k (2.59) przyjmuje postać:

$$\begin{cases} i_{1} = \frac{1}{2} \left(s_{1} + \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \pm m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}} \right) \\ j_{1} = \frac{1}{2} \left(s_{1} - \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \pm m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}} \right) \end{cases}$$
(2.61)

Jako znak \pm w sumach we wzorze (2.61), określającym postać pary niewiadomych (i_1, j_1) , przyjmujemy odpowiednio "+" dla wartości parametru p_k odpowiadającym nieujemnym wartościom parametru r_k , oraz "–" dla wartości p_k odpowiadającym ujemnym wartościom r_k . Załóżmy, że mamy dwie (2n-2)-tki niewiadomych (2.24) spełniających założenie symetryczności rozwiązania (2.58):

$$\left(i_{2}^{(1)},...,i_{k}^{(1)},...,i_{n}^{(1)},j_{2}^{(1)},...,j_{k}^{(1)},...,j_{n}^{(1)}\right) = \frac{1}{2}\left(s_{2}\overline{\odot}p_{2},...,s_{n}\overline{\odot}p_{n},s_{2}\odot p_{2},...,s_{n}\odot p_{n}\right)$$
(2.62)

$$\left(i_{2}^{(2)},...,i_{k}^{(2)},...,i_{n}^{(2)},j_{2}^{(2)},...,j_{k}^{(2)},...,j_{n}^{(2)}\right) = \frac{1}{2}\left(s_{2}\odot p_{2},...,s_{n}\odot p_{n},s_{2}\overline{\odot} p_{2},...,s_{n}\overline{\odot} p_{n}\right)$$
(2.63)

gdzie znak \odot oznacza operację dodawania lub odejmowania, natomiast $\overline{\odot}$ oznacza, dla odpowiadających sobie zmiennych we wzorach (2.62) i (2.63), działanie przeciwne. Z równania (2.25) wynika, że wartości niewiadomych (i_1, j_1) są zdeterminowane przez wartości pozostałych zmiennych (i_k, j_k) , k = 2,...,n, więc symetria par niewiadomych $(i_{1}^{(1)}, j_{1}^{(1)}), (i_{1}^{(2)}, j_{1}^{(2)})$ jest zdeterminowana przez wartości pozostałych niewiadomych (i_k, j_k) , k = 2,...,n. Podstawiając (2n-2)-tki (2.62) i (2.63) do wzoru (2.61), para niewiadomych (i_1, j_1) przyjmuje postać:

$$\begin{pmatrix} i_{1}^{(1)}, j_{1}^{(1)} \end{pmatrix} = \\ = \frac{1}{2} \left(s_{1} + \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \overline{\odot} m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}}, s_{1} - \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \overline{\odot} m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}} \right) = \\ = \frac{1}{2} \left(s_{1} - \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \overline{\odot} m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}}, s_{1} + \frac{1}{s_{1}} \frac{\sum_{k=2}^{n} \overline{\odot} m_{1} \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_{n} s_{k} p_{k}}{m_{2} m_{3} \cdot \ldots \cdot m_{n}} \right)$$

$$(2.64)$$

Z równania (2.64) wynika:

$$\left(i_{1}^{(1)}, j_{1}^{(1)}\right) = \left(j_{1}^{(2)}, i_{1}^{(2)}\right)$$

$$(2.65)$$

Innymi słowy: z równości par (2.65) wynika, że dwie symetryczne (2n-2)-tki (2.62), (2.63) generują także symetryczne pary $(i_1^{(1)}, j_1^{(1)}), (i_1^{(2)}, j_1^{(2)})$. Dlatego w celu określenia zbioru wartości parametrów s_k, p_k dla k = 1, ..., n, generującego zbiór wszystkich niesymetrycznych rozwiązań równania (2.13), konieczna i wystarczająca jest jedynie weryfikacja symetryczności zmiennych (i_k, j_k) dla indeksu k zmieniającego się od 2 do n. W celu eliminacji rozwiązań symetrycznych rozróżnijmy dwa przypadki, podane w kolejnych podpunktach.

2.7.1. *Przypadek* 1: $p_k > 0, k = 2, ..., n$

Weźmy pod uwagę następujący podzbiór rozwiązań równania (2.13), wyrażony w funkcji parametrów s_i oraz p_i , wynikający z równości (2.60):

$$(i_2,...,i_q) = \frac{1}{2} (s_2 \odot_2 p_2,...,s_q \odot_q p_q), \ q = 2,...,n-1$$
 (2.66)

gdzie \bigcirc_i , i = 2,...,q oznacza działanie dodawania lub odejmowania. Analogiczny podzbiór rozwiązań dla prawej strony równania (2.13), wyrażony w funkcji tych samych parametrów, ma postać (2.67). Wynika on także z równości (2.60) oraz założonej postaci rozwiązania $(i_2,...,i_q)$ (2.66).

$$(j_2,...,j_q) = \frac{1}{2} (s_2 \overline{\odot}_2 p_2,...,s_q \overline{\odot}_q p_q), q = 2,...,n-1$$
 (2.67)

gdzie $\overline{\odot}_i$ oznacza działanie przeciwne do \odot_i . Parametr q w równaniach (2.66), (2.67) generuje 2^{q-1} wariacji z powtórzeniami 2-elementowego zbioru znaków $\{+, -\}$. Przeanalizujmy, ile niesymetrycznych rozwiązań równania (2.13) uzyskamy po wzięciu pod uwagę o jedną niewiadomą więcej z jego obu stron, niż uwzględniliśmy we wzorach (2.66) i (2.67), czyli po dodaniu niewiadomych (i_{q+1}, j_{q+1}) . Dodając te niewiadome do każdej z wariacji znaków we wzorach (2.66), (2.67), otrzymujemy dwa nowe rozwiązania:

$$\begin{pmatrix} i_{2}^{(1)}, \dots, i_{q}^{(1)}, i_{q+1}^{(1)}, j_{2}^{(1)}, \dots, j_{q}^{(1)}, j_{q+1}^{(1)} \end{pmatrix} = = \frac{1}{2} \Big(s_{2} \odot_{2} p_{2}, \dots, s_{q} \odot_{q} p_{q}, s_{q+1} + p_{q+1}, s_{2} \overline{\odot}_{2} p_{2}, \dots, s_{q} \overline{\odot}_{q} p_{q}, s_{q+1} - p_{q+1} \Big), q = 2, \dots, n-1$$
 (2.68)

oraz:

$$\begin{pmatrix} i_{2}^{(2)}, \dots, i_{q}^{(2)}, i_{q+1}^{(2)}, j_{2}^{(2)}, \dots, j_{q}^{(2)}, j_{q+1}^{(2)} \end{pmatrix} = = \frac{1}{2} \Big(s_{2} \tilde{\odot}_{2} p_{2}, \dots, s_{q} \tilde{\odot}_{q} p_{q}, s_{q+1} - p_{q+1}, s_{2} \tilde{\odot}_{2} p_{2}, \dots, s_{q} \tilde{\odot}_{q} p_{q}, s_{q+1} + p_{q+1} \Big) q = 2, \dots, n-1$$
 (2.69)

gdzie $\tilde{\odot}_i$, i = 2,...,q także oznacza dowolny ze znaków dodawania lub odejmowania, ale niezależny od \odot_i . Z definicji rozwiązań symetrycznych (2.57) i (2.58), w podzbiorach rozwiązań (2.66) i (2.67) można wyróżnić dwa podzbiory rozwiązań symetrycznych - każdej z wariacji znaków w tychże podzbiorach rozwiązań (2.66) i (2.67) odpowiada jedna wariacja generująca symetryczny podzbiór rozwiązań. Więc dla każdej wariacji znaków $\odot_2 ... \odot_q$ we wzorze (2.68) można znaleźć dokładnie jedną wariację $\widetilde{\odot}_2 ... \widetilde{\odot}_q$, generującą podzbiór rozwiązań symetrycznych, jak pokazano w następującym równaniu:

$$\forall \exists \\ (j_{2}^{(1)},...,j_{q}^{(1)}) = (j_{2}^{(2)},...,j_{q}^{(2)}) \\ (j_{2}^{(1)},...,j_{q}^{(1)}) = (i_{2}^{(2)},...,i_{q}^{(2)}) \\ q = 2,...,n-1$$

$$(2.70)$$

Ponadto, ze wzorów (2.68), (2.69) widać, że:

$$\left(i_{q+1}^{(1)}, j_{q+1}^{(2)}\right) = \left(j_{q+1}^{(2)}, i_{q+1}^{(1)}\right), \quad q = 2, \dots, n-1$$

$$(2.71)$$

Innymi słowy: równanie (2.71) oznacza, że dodana para zmiennych (i_{q+1}, j_{q+1}) generuje dwa nowe rozwiązania symetryczne. Podsumowując, przy odpowiednim wyborze ciągów znaków \odot_i , i = 2,...,q oraz $\tilde{\odot}_i$, i = 2,...,q, równania (2.68) i (2.69) wyrażają dwa symetryczne i równoliczne podzbiory rozwiązań. Celem niniejszego podrozdziału jest eliminacja rozwiązań symetrycznych, dlatego w tym celu powinniśmy wziąć pod uwagę jedynie jeden podzbiór z rozwiązań (2.68) i (2.69) dla q = n-1. Arbitralnie wybierzmy podzbiór opisany zależnością (2.68). Zależność (2.68) dla q = n-1 oznacza, że dla dokładnie jednego z parametrów $p_2,...,p_n$ musimy wziąć pod uwagę jedynie jeden znak we wzorze (2.60).

2.7.2. *Przypadek 2:* $p_k = 0, k = 2, ..., n$

W tym podpunkcie obowiązują oznaczenia wprowadzone dla przypadku 1. W niniejszym przypadku równania (2.68) i (2.69) przyjmują postać:

$$\begin{cases} \left(i_{2}^{(1)},...,i_{q}^{(1)},i_{q+1}^{(1)}\right) = \frac{1}{2}\left(s_{2}\odot_{2}p_{2},...,s_{q}\odot_{q}p_{q},s_{q+1}\right)\\ \left(j_{2}^{(1)},...,j_{q}^{(1)},j_{q+1}^{(1)}\right) = \frac{1}{2}\left(s_{2}\overline{\odot}_{2}p_{2},...,s_{q}\overline{\odot}_{q}p_{q},s_{q+1}\right) \end{cases}, \quad q = 2,...,n-1$$

$$(2.72)$$

Z równań (2.72) wynika, że w tym przypadku dodawanie kolejnej zmiennej do podzbioru rozwiązań nie generuje rozwiązań symetrycznych, więc nie ma potrzeby eliminacji rozwiązań.

2.8. Algorytm finalny

Wyniki uzyskane w punktach 2.6 i 2.7 pozwalają na skonstruowanie algorytmu, który znajduje zbiór wszystkich niesymetrycznych i nietrywialnych rozwiązań równania (2.13) przy założeniach (2.14)-(2.18). Uzyskany algorytm jest w pełni deterministyczny i nie wymaga użycia przybliżonych metod iteracyjnych. Dlatego generowane wyniki są obciążone jedynie błędami zaokrągleń.

- 0. Wprowadź następujące argumenty typu całkowitego: – liczbę niewiadomych po każdej ze stron równania: n :integer,
- dziedzinę równania: M :integer,
- parametry równania: $m_1, m_2, ..., m_n$: integer.
- 1. Niech A_k^1 oznacza zbiór wszystkich punktów (r_k, s_k) o współrzędnych całkowitych zawartych wewnątrz prostokąta z wierzchołkami o współrzędnych (0,2), (M-1,M+1), (0,2N), (-M+1,M+1), k=2,..,n.
- 2. Niech $A_k^{\overline{2}}$ będzie zbiorem wszystkich punktów (r_k, s_k) o współrzędnych całkowitych zawartych wewnątrz trójkąta z wierzchołkami o współrzędnych (0,2), (M-1,M+1), (0,2M), k=2,..,n.

```
3. Dla (r_2, s_2) \in A_2^{\frac{1}{2}} powtarzaj kroki 4-15.

4. ...

5. Jeżeli punkt (r_{k-1}, s_{k-1}) leży na osi S_{k-1} płaszczyzny R_{k-1}S_{k-1}, wtedy

dla (r_k, s_k) \in A_k^{\frac{1}{2}} powtarzaj kroki 6-15

w przeciwnym przypadku

dla (r_k, s_k) \in A_k^1 powtarzaj kroki 6-15.

6. ...

7. Jeżeli punkt (r_{n-1}, s_{n-1}) leży na osi S_{n-1} płaszczyzny R_{n-1}S_{n-1}, wtedy

dla (r_n, s_n) \in A_n^{\frac{1}{2}} powtarzaj kroki 8-15
```

w przeciwnym przypadku dla $(r_n, s_n) \in A_n^1$ powtarzaj kroki 8-15. 8. Jeżeli $S(\vec{s}_2, \vec{r}_2) \ge 0$, wtedy wykonaj krok 9 w przeciwnym przypadku przejdź do kroku 10. 9. Jeżeli $\Delta_2^- \geq 0$, wtedy dla $s_1 \in P_7$ (przedział dany wzorem (2.53)) wykonuj kroki 12-15. 10. Jeżeli $S(\vec{s}_2, \vec{r}_2) < 0$, wtedy wykonaj krok 11. 11. Jeżeli $\Delta_2^{\scriptscriptstyle +} \geq 0$, wtedy dla $s_{\rm l} \in P_{\rm 8}$ (przedział dany wzorem (2.55)) wykonuj kroki 12-15. 12. Jeżeli $2|(s_k+r_k)$ oraz $2|(s_k-r_k)$, wtedy dla k = 2..n powtarzaj kroki 13-15. 13. Jeżeli zachodzą podzielności: $\left(s_1m_2m_3\cdot\ldots\cdot m_n\right)\left(\sum_{k=2}^n m_1\cdot\ldots\cdot m_{k-1}m_{k+1}\cdot\ldots\cdot m_ns_kr_k\right)$ oraz $2\left(s_1 \pm \frac{1}{s_1} \frac{\sum_{k=2}^{n} m_1 \cdot \ldots \cdot m_{k-1} m_{k+1} \cdot \ldots \cdot m_n s_k r_k}{m_2 m_3 \cdot \ldots \cdot m_n}\right)$ wtedy wykonaj kroki 14-15. 14. Oblicz $(i_1,...,i_k,...,i_n,j_1,...,j_k,...,j_n)$ na postawie wzorów (2.24), (2.25). 15.Jeżeli $(i_1,...,i_k,...,i_n) \neq (j_1,...,j_k,...,j_n)$ wtedv dodaj (2n+1)-tkę $(i_1,...,i_k,...,i_n,j_1,...,j_k,...,j_n,S)$ do zbioru rozwiązań (gdzie S jest równe kombinacji liniowej w stronach równania (2.13)). 16. Koniec.

2.9. Złożoność czasowa algorytmu 2.8

W algorytmie 2.8 iteruje się 2(n-1) zmiennych w dziedzinie trójkąta lub prostokąta. Każda ze zmiennych w parze (r_k, s_k) , k = 2, ..., n iteruje w przedziale proporcjonalnym do górnego ograniczenia M dziedziny (2.18) rozwiązywanego równania (2.13) (kroki 3-7 algorytmu 2.8). Ostatni parametr s_1 iteruje wewnątrz przedziału (2.53) lub (2.55), zależnego od znaku wyrażenia $S(\vec{s}_2, \vec{r}_2)$ (2.32). Zatem, złożoność prezentowanego algorytmu 2.8 może zostać obliczona jak następuje:

$$T(M) = O(M^{2n-2}) \cdot O\left(M + \sqrt{M^2 + \frac{S(\vec{s}_2, \vec{r}_2)}{m_2 m_3 \cdot \dots \cdot m_n}} - 1 - \sqrt{1 + \frac{S(\vec{s}_2, \vec{r}_2)}{m_2 m_3 \cdot \dots \cdot m_n}}\right) = O(M^{2n-2}) \cdot O(M) = O(M^{2n-1})$$
(2.73)

gdzie *n* jest równe liczbie niewiadomych, jakie występują w kombinacjach liniowych po obydwu stronach równania (2.13). Podsumowując, w porównaniu z algorytmem przeglądu zupełnego (ang. *brute-force*) zaprezentowany algorytm 2.8 poprawił efektywność o czynnik liniowy.

2.10. Określenie krotności n-tek generujących równe sumy kwadratów

Celem algorytmu 2.8 jest znalezienie rozwiązań równania (2.13). Wśród rozwiązań tego równania można znaleźć więcej niż dwie n-tki generujące równe sumy kwadratów, tj. n-tki spełniające następującą równość:

$$\frac{i_{1(1)}^{2}}{m_{1}} + \dots + \frac{i_{k(1)}^{2}}{m_{k}} + \dots + \frac{i_{n(1)}^{2}}{m_{n}} = \dots = \frac{i_{1(l)}^{2}}{m_{1}} + \dots + \frac{i_{k(l)}^{2}}{m_{k}} + \dots + \frac{i_{n(l)}^{2}}{m_{n}} = \dots =$$

$$= \frac{i_{1(\gamma)}^{2}}{m_{1}} + \dots + \frac{i_{k(\gamma)}^{2}}{m_{k}} + \dots + \frac{i_{n(\gamma)}^{2}}{m_{n}} = S$$
(2.74)

gdzie przez *S* oznaczono wartość kombinacji liniowych (2.74). Dzieje się tak ze względu na specyfikę algorytmu 2.8: znajdywane n-tki są obliczane przez ten algorytm w parach. Dlatego krotności n-tek generujących równe sumy kwadratów w kombinacji liniowej (2.74) nie mogą być bezpośrednio określone przez algorytm 2.8. Problem określania krotności n-tek generujących równe sumy kwadratów można sformalizować następujaco: dla każdej wartości *S* odpowiadającej zbiorowi 2n-tek $(i_1,...,i_k,...,i_n, j_1,..., j_k,..., j_n)$, wygenerowanych przez algorytm 2.8, należy określić liczbę członów γ w równości (2.74). Można ją obliczyć zauważając fakt, że algorytm 2.8 wyznacza wszystkie dwuelementowe kombinacje zbioru złożonego ze wszystkich 2n-tek $(i_1,...,i_k,...,i_n, j_1,..., j_k,..., j_n)$ o tej samej wartości sumy *S*. Stąd z kombinatoryki wynika równanie:

$$\binom{\gamma}{2} = q \tag{2.75}$$

gdzie:

- γ szukana krotność n-tek generujących równe sumy kwadratów we wzorze (2.74),
- q liczba niesymetrycznych 2n-tek (i₁,...,i_k,...,i_n, j₁,..., j_k,..., j_n), odpowiadających danej wartości *S*, zwróconych przez algorytm 2.8.

Wzór (2.75), po zastosowaniu definicji symbolu Newtona, prowadzi do klasycznego równania kwadratowego. Na podstawie wzoru (2.75) można obliczyć szukaną liczbę równych wyrazów γ we wzorze (2.74):

$$\gamma = \frac{1 + \sqrt{1 + 8q}}{2} \tag{2.76}$$

2.11. Analiza błędów obliczeń

W obliczaniu wartości sumy (2.74) popełniamy błąd zaokrągleń spowodowany skończoną reprezentacją liczb zmiennoprzecinkowych w pamięci komputera. Na podstawie lematu Wilkinsona 2.4.1 możemy oszacować dokładność obliczonej sumy (2.74).

Liczby i_k , j_k oraz m_k są z założenia liczbami całkowitymi, dlatego błąd popełniamy jedynie przy obliczaniu ilorazu, a następnie dodawaniu ułamków reprezentowanych już jako liczby zmiennoprzecinkowe. Stąd na podstawie lematu Wilkinsona suma (2.74) będzie obliczona numerycznie jako:

$$S = \sum_{k=1}^{n} \frac{i_{k}^{2}}{m_{k}} (1 + \varepsilon_{k1}) (1 + \varepsilon_{k2})$$
(2.77)

gdzie ε_{k1} reprezentuje błąd zmiennoprzecinkowy ilorazu, natomiast ε_{k2} odpowiada błędowi sumy liczb już obciążonych błędem zaokrągleń powstałych w wyniku dzielenia. Zachodzi zależność:

$$\left|\varepsilon_{k(1,2)}\right| \leq \varepsilon, \quad k=1,2,...,n$$

gdzie $\varepsilon = 2^{-\eta+1} = 2^{-52}$, co wynika ze wzoru (2.9). Obliczoną sumę (2.77) można oszacować następująco:

$$S_0(1-\varepsilon)^{2n} \le S \le S_0(1+\varepsilon)^{2n}$$
(2.78)

gdzie S_0 oznacza dokładną wartość sumy, a 2n jest liczbą niewiadomych w równaniu diofantycznym (2.13).

2.12. Przykład wykonania algorytmu 2.8

W niniejszym przykładzie zaprezentujemy sposób działania algorytmu 2.8 w przypadku następującego równania diofantycznego (2.79) z konkretnymi wartościami parametrów całkowitych $m_1 = 1$, $m_2 = 3$, $m_3 = 5$:

$$\frac{i_1^2}{1} + \frac{i_2^2}{3} + \frac{i_3^2}{5} = \frac{j_1^2}{1} + \frac{j_2^2}{3} + \frac{j_3^2}{5}$$
(2.79)

przy ograniczeniu:

$$\bigvee_{k=1,2,3} i_k, j_k \le M = 4 \tag{2.80}$$

W niniejszym przykładzie zbiory A_k^1 , deklarowane w punkcie 1 algorytmu 2.8, są prostokątami o wierzchołkach (0,2), (3,5), (0,8), (-3,5). Z kolei trójkąty $A_k^{\frac{1}{2}}$, deklarowane w punkcie 2 tego algorytmu, mają wierzchołki (0,2), (3,5), (0,8). Kolejne kroki wykonania

Tabela 1

algorytmu 2.8 przedstawiono w tabeli 1 ((x) oznacza nieskończoną liczbę powtórzeń cyfry x w rozwinięciu dziesiętnym danej liczby).

Kolejne iteracje w wykonywaniu algorytmu 2.8						
Wartości parametrów pomocniczych]	Rozwiązania	Wartość sumy	
(r_2, s_2)	(r_3, s_3)	S_1	(i_1,i_2,i_3)	(j_1, j_2, j_3)	S	
(0,2)		3	(2,1,1)	(1,1,4)	4.5(3)	
(0,4)	(35)		(2,2,1)	(1,2,4)	5.5(3)	
(0,6)	(5,5)		(2,3,1)	(1,3,4)	7.2	
(0,8)			(2,4,1)	(1,4,4)	9.5(3)	
	(-1,5)	2	(1,1,3)	(1,2,2)	3.1(3)	
(1 3)		4	(2,1,3)	(2,2,2)	6.1(3)	
(1,5)		6	(3,1,3)	(3,2,2)	11.1(3)	
		8	(4,1,3)	(4,2,2)	18.1(3)	
		3	(2,2,3)	(1,4,2)	7.1(3)	
(2,6)	(1,5)	5	(3,2,2)	(2,4,3)	11.1(3)	
	(3,5)	7	(4,2,1)	(3,4,4)	17.5(3)	
	(0,2)	- 5	(3,1,1)	(2,4,1)	9.5(3)	
	(0,4)		(3,1,2)	(2,4,2)	10.1(3)	
(3,5)	(0,6)		(3,1,3)	(2,4,3)	11.1(3)	
	(0,8)		(3,1,4)	(2,4,4)	12.5(3)	
	(3,5) 4	(3,1,1)	(1,4,4)	9.5(3)		

Kolejne iteracje w wykonywaniu algorytmu 2.8

Analizując tabelę 1 warto zauważyć, że dla dwóch różnych wartości sumy S, tj. 9.5(3) oraz 11.1(3), istnieje więcej niż jedna para trójek (i_1, i_2, i_3) , (j_1, j_2, j_3) , którym odpowiada ta sama wartość w równaniu (2.79), oznaczona jako S. Dla sumy o wartości S = 9.5(3) mamy następujące trzy pary trójek:

 $(i_1, i_2, i_3; j_1, j_2, j_3) = \begin{cases} (2, 4, 1; 1, 4, 4) \\ (3, 1, 1; 2, 4, 1) \\ (3, 1, 1; 1, 4, 4) \end{cases}$

natomiast dla S = 11.1(3) otrzymaliśmy następujące pary:

$$(i_1, i_2, i_3; j_1, j_2, j_3) = \begin{cases} (3, 1, 3; 3, 2, 2) \\ (3, 2, 2; 2, 4, 3) \\ (3, 1, 3; 2, 4, 3) \end{cases}$$

Określenia krotności trójek generujących równą wartość sumy S dokonujemy za pomocą wzoru (2.76). Dla obydwu powyższych par trójek otrzymujemy:

$$\gamma = \frac{1 + \sqrt{1 + 8 \cdot 3}}{2} = 3 \tag{2.81}$$

Oznacza to, że dla sumy o wartości S = 9.5(3) mamy trzy trójki o postaci: (3,1,1), (2,4,1), (1,4,4). Natomiast dla S = 11.1(3) występują następujące trzy trójki: (3,1,3), (3,2,2), (2,4,3).

2.13. Praktyczny test efektywności algorytmu 2.8

Na rys. 1 przedstawiono w skali logarytmicznej czas wykonania algorytmu 2.8 oraz algorytmu przeglądu zupełnego dla parametrów $m_1 = 1$, $m_2 = 3$, $m_3 = 5$ oraz ograniczeniu M (2.18), zmieniającym się w przedziale od 1 do 48. Testy wykonano na serwerze wyposażonym w procesor Xeontm 2.8GHz. Jak widać, wprowadzony algorytm 2.8 znacznie podnosi efektywność czasową, dla M = 48 zysk czasowy wynosi już ponad 3600%.



- Rys. 1. Czas wykonania algorytmu 2.8 w porównaniu z czasem wykonania algorytmu przeglądu zupełnego w funkcji górnego ograniczenia zbioru rozwiązań
- Fig. 1. The algorithm 2.8 execution time compared with the brute force execution time with respect to the upper solution set constraint

2.14. Podsumowanie

Celem rozdziału 2 jest przedstawienie algorytmu służącego do efektywnego rozwiązywania równania diofantycznego (2.13) przy założeniach (2.14)-(2.18). Działanie zaprezentowanego algorytmu 2.8 opiera się na parametryzacji niewiadomych oraz wykorzystaniu własności podzielności liczb całkowitych. Istotną jego cechą jest to, że nie zwraca on rozwiązań symetrycznych, czyli takich, gdzie rozwiązania są jedynie zamienione stronami. Eliminacji takich rozwiązań jest poświęcona analiza w punkcie 2.7. Uzyskany zysk czasowy jest znaczący w porównaniu do rozwiązania tego równania metodą przeglądu zupełnego, czego dowodzi zarówno teoretyczna złożoność czasowa algorytmu (2.73), jak i uzyskane testowe czasy wykonania. Dodatkową zaletą algorytmu 2.8 jest to, że nie wymaga on przydzielania w pamięci miejsca dla dodatkowych struktur danych, z wyjątkiem ewentualnego zapamiętania zbioru rozwiązań. W kolejnym rozdziale przedstawiono jego zastosowanie w numerycznym badaniu własności pewnej klasy nieskończenie wymiarowych układów dynamicznych.
3. ZASTOSOWANIE ALGORYTMÓW ROZWIĄZYWANIA RÓWNAŃ DIOFANTYCZNYCH W BADANIACH ZBIORÓW OSIĄGALNYCH WYBRANYCH UKŁADÓW DYNAMICZNYCH

3.1. Wprowadzenie

W niniejszym rozdziale zastosowano nieliniowe równanie diofantyczne (2.13) do analizy zbiorów osiągalnych wybranej klasy nieskończenie wymiarowych układów dynamicznych. Przeanalizowano nieskończenie wymiarowy układ dynamiczny, opisany równaniem różniczkowym cząstkowym drugiego rzędu typu parabolicznego z zerowymi warunkami brzegowymi typu Dirichleta. Dziedziną układu jest n-wymiarowy wielościan. Zbiory osiągalne takiego układu są niemożliwe do zbadania na drodze czysto analitycznej. Wynika to z własności wartości własnych operatora różniczkowego występującego w modelu tego układu, gdyż ich krotności są niemożliwe do wyznaczenia analitycznie. Dlatego znalazł zastosowanie algorytm rozwiązywania równań diofantycznych z rozdziału 2. Rezultaty przedstawione w niniejszym rozdziale pochodzą z publikacji [83].

3.2. Wyjaśnienie terminologii przyjętej w rozdziale

Zbiory osiągalne układów dynamicznych są stosunkowo rzadko badane w literaturze. Jedną z ich najczęściej badanych własności jest sterowalność. Zdefiniowana ona została przez Kalmana w pracy [40]. Mówiąc nieprecyzyjnie, oznacza ona, że zbiór osiągalny układu, który ją posiada, obejmuje wszystkie możliwe wartości. Inaczej mówiąc: poprzez swobodny dobór funkcji wymuszającej można doprowadzić rozwiązanie układu do dowolnej wartości w skończonym czasie. Precyzyjną definicję podaje [46] s. 4.

Układ dynamiczny, któremu poświęcono niniejszy rozdział, był w postaci ogólniejszej badany w pracach [46] s. 138 oraz [25]. W pracach tych podano ogólne warunki jego sterowalności dla dowolnego kształtu i wymiarów jego dziedziny przestrzennej. Jej warunkiem koniecznym jest, aby krotność wartości własnych była nie większa od liczby dostępnych składowych wymuszenia. Jednak dla badanej postaci dziedziny i warunków brzegowych krotność ta zależy od konkretnych długości boków n-wymiarowego wielościanu. Jej badaniu służy algorytm przedstawiony w rozdziale 2. Algorytm ten pozwala stwierdzić, kiedy krotność wartości własnych przekracza liczbę dostępnych wymuszeń.

Układ dynamiczny badany w rozdziale jest w ogólnym przypadku niesterowalny. W rozdziale przedstawiono metodologię zastosowania algorytmu rozwiązywania nieliniowych równań diofantycznych do wyznaczania zbioru osiągalnego badanego układu dynamicznego. Formalna definicja zbioru osiągalnego jest podana w [66] s.69 dla układów dynamicznych modelowanych równaniami różniczkowymi zwyczajnymi i w [46] s. 129 dla układów dynamicznych modelowanych równaniami różniczkowymi cząstkowymi.

3.3. Wprowadzenie do tematyki układów dynamicznych o parametrach rozłożonych typu parabolicznego

W niniejszym punkcie przedstawiono matematyczne podstawy liniowych układów dynamicznych 2 rzędu o parametrach rozłożonych typu parabolicznego oraz ich znaczenie w nauce i technice.

3.3.1. Ogólna postać liniowych równań różniczkowych cząstkowych rzędu drugiego

Liniowe równanie różniczkowe cząstkowe rzędu drugiego o niewiadomej funkcji x(z,t)dwóch zmiennych niezależnych z i t nazywamy równanie następującej postaci:

$$A\frac{\partial^2 x}{\partial z^2} + B\frac{\partial^2 x}{\partial z \partial t} + C\frac{\partial^2 x}{\partial t^2} + a\frac{\partial x}{\partial z} + b\frac{\partial x}{\partial t} + cx + d = 0$$
(3.1)

gdzie A, B, C, a, b, c, d oznaczają dane funkcje dwóch zmiennych z i t o ciągłych pochodnych w obszarze płaskim D.

W niniejszym rozdziale skupiono się na podklasie równania (3.1), tj. na liniowych równaniach cząstkowych rzędu drugiego typu parabolicznego. Typ paraboliczny równania (3.1) można przedstawić w następującej postaci ogólnej:

$$\frac{\partial^2 x}{\partial z^2} + F\left(\frac{\partial x}{\partial z}, \frac{\partial x}{\partial t}, x, z, t\right) = 0$$
(3.2)

3.3.2. Znaczenie fizyczne równań różniczkowych cząstkowych typu parabolicznego

Analizowana w niniejszym rozdziale klasa parabolicznych równań różniczkowych cząstkowych znajduje zastosowanie w modelowaniu następujących zjawisk fizycznych:

- zagadnienie przewodnictwa cieplnego,
- zagadnienie dyfuzji,
- zjawisko fal temperaturowych,
- zagadnienie krzepnięcia,
- analiza ruchów Browna,
- drgania belek i powłok elastycznych,
- mechanika kwantowa (równanie Schrödingera).

Problematyka równań różniczkowych cząstkowych jest poruszana w monografiach: Wolska-Bochenek [137], Sneddon [126], Renardy [80], Evans [20], Budak [8], Tichonow [129], Curtain [12].

3.4. Model badanego układu nieskończenie wymiarowego typu parabolicznego

Dany jest układ dynamiczny, opisany następującym liniowym równaniem różniczkowym cząstkowym drugiego rzędu, typu parabolicznego:

$$\frac{\partial x(z,t)}{\partial t} = \frac{\partial^2 x(z,t)}{\partial z_1^2} + \dots + \frac{\partial^2 x(z,t)}{\partial z_k^2} + \dots + \frac{\partial^2 x(z,t)}{\partial z_n^2} + \sum_{i=1}^p b_i(z)u_i(t)$$
(3.3)

gdzie $p \in Z_+$ jest liczbą funkcji wymuszających. Dziedziną *D* układu (3.3) jest n-wymiarowy wielościan:

$$D = \left\{ z = \left(z_1, ..., z_k, ..., z_n \right) \in \mathbb{R}^n : z_k \in [0, a_k], \ a_k^2 \in \mathbb{Z}_+, \ k = 1, 2, ..., n \right\}$$
(3.4)

Każda z funkcji $b_i(z)$ ma postać iloczynu *n* funkcji:

$$b_i(z) = \prod_{j=1}^n f_{ij}(z_j), \quad i = 1, 2, ..., p$$
 (3.5)

Czas przyjmuje wartości z przedziału:

$$0 \le t < \infty \tag{3.6}$$

Zakładamy zerowe warunki brzegowe typu Dirichleta:

$$x(z,t) = 0\Big|_{z \in \Gamma} \tag{3.7}$$

gdzie Γ jest brzegiem obszaru D. Ponadto:

$$u_i(t) \in L^2([0,\infty], R), \ i = 1, 2, ..., p$$
 (3.8)

są skalarnymi funkcjami wymuszającymi.

Problem jest następujący:

 jaki jest zbiór osiągalny nieskończenie wymiarowego układu dynamicznego opisanego parabolicznym równaniem różniczkowym cząstkowym (3.3) określonego w n-wymiarowym wielościanie przy zerowych warunkach brzegowych Dirichleta?

Definicja 3.1

Przez pojęcie zbioru osiągalnego układu dynamicznego (3.3) rozumiemy zbiór wszystkich możliwych jego rozwiązań w postaci funkcji dwóch zmiennych x(z,t), przy dowolności wyboru funkcji wymuszających $u_i(t)$ z klasy (3.8), w skończonym czasie.

3.5. Analiza numeryczna własności nieskończenie wymiarowego układu parabolicznego

3.5.1. Transformacja układu

Najpierw przekształćmy równanie różniczkowe cząstkowe (3.3) do postaci tzw. abstrakcyjnego równania różniczkowego. Wykonanie tej transformacji wymaga zdefiniowania następujących dwóch operatorów ([46 s. 138, 25]):

• operatora stanu *A* jako operatora różniczkowego drugiego rzędu danego formułami:

$$Ax(z) = \frac{\partial^2 x(z)}{\partial z_1^2} + \dots + \frac{\partial^2 x(z)}{\partial z_k^2} + \dots + \frac{\partial^2 x(z)}{\partial z_n^2}, \quad x(z) \in D(A)$$
(3.9)

$$D(A) = \left\{ x(z) \in L^{2}(D) : Ax(z) \in L^{2}(D), \ x(z,t) = 0 \Big|_{z \in \Gamma} \right\}$$
(3.10)

• operatora wejścia *B* w postaci:

$$B = \begin{bmatrix} b_1(z) & b_2(z) & \cdots & b_k(z) & \cdots & b_p(z) \end{bmatrix}$$

$$b_k(z) \in X, \ k = 1, 2, ..., p$$
(3.11)

$$B: \mathbb{R}^p \to X \tag{3.12}$$

Tak zdefiniowane operatory *A*, *B* sprowadzają nam równanie różniczkowe cząstkowe z klasycznej postaci (3.3) do zapisu w formie abstrakcyjnego równania różniczkowego.

$$\dot{x}(t) = Ax(t) + Bu(t), \quad t \ge 0$$
(3.13)

W dalszej transformacji będą konieczne własności spektralne operatora różniczkowego *A* danego wzorem (3.9). Na podstawie monografii [9] wartości własne operatora *A* mają postać:

$$\lambda_{\alpha}\left(i_{1}...i_{n}\right) = -\pi^{2}\left(\frac{i_{1}^{2}}{a_{1}^{2}} + ... + \frac{i_{k}^{2}}{a_{k}^{2}} + ... + \frac{i_{n}^{2}}{a_{n}^{2}}\right), \quad k = 1, 2, ..., n, \ i_{k} = 1, 2, 3, ...$$
(3.14)

gdzie $\lambda_{\alpha}(i_{1}...i_{n})$ oznacza wartość własną odpowiadającą danej n-tce $(i_{1}...i_{n})$. Jak pokażemy w dalszej części rozdziału, ta sama wartość własna λ_{α} może odpowiadać więcej niż jednej n-tce $(i_{1}...i_{n})$. Wtedy liczbę takowych n-tek $(i_{1}...i_{n})$, dających na podstawie wzoru (3.14) tę samą wartość własną λ_{α} , nazywamy krotnością wartości własnej λ_{α} i oznaczamy jako r_{α} . Funkcje własne operatora różniczkowego A (3.9), (3.10) mają postać następującego iloczynu funkcji trygonometrycznych:

$$\phi_{i_1\dots i_n}(z_1,\dots,z_j,\dots,z_n) = C \prod_{k=1}^n \sin\frac{\pi i_k z_k}{a_k} \in D(A) \quad k = 1,2,\dots,n, \ i_k = 1,2,3,\dots$$
(3.15)

Zbiór wszystkich funkcji własnych $\phi_{i_1...i_n}$, odpowiadających zbiorowi n-tek $(i_1...i_n)$ dających tę samą wartość własną λ_{α} (3.14) ma liczność r_{α} i oznaczymy jego elementy jako $\phi_{\alpha j}, j = 1,..,r_{\alpha}$. Ostatnim krokiem transformacji rozważanego układu parabolicznego (3.3) jest dekompozycja spektralna. Po dekompozycji spektralnej układu (3.3), zapisanego z użyciem operatorów A (3.9), (3.10) oraz B (3.11), (3.12) w postaci abstrakcyjnego równania różniczkowego (3.13), otrzymujemy następujący równoważny nieskończony ciąg równań różniczkowych zwyczajnych: ([46 s.138, 25])

$$\dot{x}_{\alpha}(t) = A_{\alpha}x_{\alpha}(t) + B_{\alpha}u(t), \quad x_{\alpha} \in R^{r_{\alpha}}, \quad \alpha = 1, 2, 3, ...$$
(3.16)

gdzie r_{α} oznacza krotność wartości własnej λ_{α} . Ponadto, macierze A_{α} oraz B_{α} mają następującą postać:

$$A_{\alpha} = \begin{bmatrix} \lambda_{\alpha} & & 0 \\ & \lambda_{\alpha} & \\ & & \ddots & \\ 0 & & \lambda_{\alpha} \end{bmatrix}, \qquad \alpha = 1, 2, 3, \dots$$

$$\dim A_{\alpha} = r_{\alpha} \times r_{\alpha}$$
(3.17)

$$B_{\alpha} = \begin{bmatrix} \langle b_{1}, \phi_{\alpha | 1} \rangle_{X} & \dots & \langle b_{l_{1}}, \phi_{\alpha | 1} \rangle_{X} & \dots & \langle b_{p}, \phi_{\alpha | 1} \rangle_{X} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \langle b_{1}, \phi_{\alpha | l_{2}} \rangle_{X} & \dots & \langle b_{l_{1}}, \phi_{\alpha | l_{2}} \rangle_{X} & \dots & \langle b_{p}, \phi_{\alpha | l_{2}} \rangle_{X} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \langle b_{1}, \phi_{\alpha | r_{\alpha}} \rangle_{X} & \dots & \langle b_{l_{1}}, \phi_{\alpha | r_{\alpha}} \rangle_{X} & \dots & \langle b_{p}, \phi_{\alpha | r_{\alpha}} \rangle_{X} \end{bmatrix}$$
(3.18)
$$\alpha = 1, 2, 3, \dots$$

gdzie:

- λ_{α} jest α -tą wartością własną operatora A,
- $\phi_{\alpha i}$ jest funkcją własną operatora A, odpowiadającą jego α -tej wartości własnej,
- r_{α} jest krotnością wartości własnej λ_{α} , $r_{\alpha} < \infty$,
- $u(t) = [u_1(t), ..., u_p(t)]^T$, $u_i(t) \in L^2([0,\infty], R)$, i = 1, 2, ..., p.

Podsumowując, przekształciliśmy wyjściowe paraboliczne równanie różniczkowe cząstkowe do równoważnej postaci nieskończonego ciągu układów skończenie wymiarowych, tj. ciągu macierzowych równań różniczkowych zwyczajnych (o skończenie

wymiarowych macierzach). Metoda ta pozwala na zastosowanie do analizy własności równań cząstkowych twierdzeń dotyczących układów skończenie wymiarowych, tj. opisywanych macierzowym równaniem różniczkowym zwyczajnym. W tym celu należy zastosować odpowiednie twierdzenie do każdego podukładu w ciągu (3.16).

3.5.2. Synteza algorytmu numerycznego analizy zbiorów osiągalnych badanego układu

Po przekształceniu w punkcie 3.5.1 badanego równania różniczkowego cząstkowego typu parabolicznego (3.3) do równoważnej postaci nieskończonego ciągu równań macierzowych (3.16) naturalne jest pytanie o operację odwrotną, tj. jak na podstawie ciągu równań (3.16) zbudować rozwiązanie x(z,t) równania cząstkowego (3.3)? Z literatury wiadomo, że funkcje własne operatora A tworzą układ ortonormalny $\{\phi_{\alpha j}, i_{\alpha} = 1, 2, 3, ..., j = 1, 2, ..., r_{\alpha}\}$ w przestrzeni Hilberta X ([21, 22, 32, 111, 25]). Dlatego dla każdego $x \in X$ jest prawdziwe następujące rozwinięcie:

$$x(z,t) = \sum_{\alpha=1}^{\infty} \sum_{j=1}^{r_{\alpha}} x_{\alpha}^{(j)}(t) \cdot \phi_{\alpha j}(z)$$
(3.19)

gdzie $x_{\alpha}^{(j)}(t)$ oznacza j-tą składową wektora $x_{\alpha}(t)$ z ciągu równań (3.16). Rozważane równanie paraboliczne (3.3) po dekompozycji ma postać zwyczajnego równania różniczkowego pierwszego rzędu (3.16). Zatem, można wyznaczyć α -ty wektor zmiennych stanu (3.20) ([38 s.32, 141]):

$$x_{\alpha}(t) = e^{A_{\alpha}t} x_{\alpha}(0) + \int_{0}^{t} e^{A_{\alpha}(t-\tau)} B_{\alpha}u(\tau)d\tau, \quad \alpha = 1, 2, 3, \dots$$
(3.20)

Przystąpmy teraz do analizy zbioru osiągalnego układu (3.3). Składają się na niego te wektory zmiennych stanu (3.20), które odpowiadają sterowalnym podukładom w nieskończonym ciągu (3.16). Jej warunkiem koniecznym i wystarczającym jest, aby rząd macierzy B_{α} był równy krotności odpowiedniej wartości własnej r_{α} ([46 s. 138, 25]):

$$rank[B_{\alpha}] = r_{\alpha}, \qquad \alpha = 1, 2, 3, \dots$$
(3.21)

gdzie macierz B_{α} jest dana przez równość (3.18).

Przeanalizujmy teraz możliwość zastosowania metodologii algorytmicznej w weryfikacji spełnienia nieskończonego ciągu równań (3.21). Wymiary macierzy (3.18) zależą od krotności wartości własnych (3.14) oraz liczby funkcji wymuszających. Krotności wartości własnych (3.14) zależą od konkretnych wartości długości krawędzi wielościanu, będącego obszarem określoności badanego równania parabolicznego (3.3). Dzieje się tak dlatego, że wartości własne (3.14) są proporcjonalne (współczynnik proporcjonalności to $-\pi^2$) do kombinacji liniowej kwadratów parametrów $(i_1...i_n)$, a współczynniki tej kombinacji liniowej zależą od długości boków wielościanu a_k , k = 1,...,n, zgodnie z formułą (3.14). Dlatego, aby zbadać rzędy macierzy w równości (3.21), konieczne jest obliczenie krotności wartości własnych r_{α} .

Zadanie to sprowadza się do znalezienia ilości n-tek generujących równe sumy kwadratów liczb całkowitych w kombinacji liniowej o dodatnich współczynnikach. Problem ten można modelować równaniem diofantycznym (2.13). Dlatego w celu wyznaczenia krotności wartości własnych r_{α} operatora różniczkowego *A* danego wzorami (3.9) i (3.10) użyjemy wprowadzonego w rozdziale 2 algorytmu 2.8 rozwiązywania równania diofantycznego (2.13). Z ciągu równości (3.21) wynika następujący warunek konieczny sterowalności każdego podukładu z ciągu (3.16):

$$\bigvee_{\alpha=1,2,3,\dots} r_{\alpha} \le p \tag{3.22}$$

Równanie (3.22) oznacza, że podukłady w ciągu (3.16) są sterowalne tak długo, jak długo krotność wartości własnych (3.14) jest nie większa od liczby dostępnych funkcji wymuszających p. Jak można zauważyć, wartości własne (3.14) systemowego operatora stanu A po wykonaniu podstawienia (3.23) są proporcjonalne do każdej ze stron równania diofantycznego (2.13):

$$m_k = a_k^2, \quad k = 1, 2, ..., n$$
 (3.23)

W kolejnym punkcie rozdziału zaproponowano algorytm 3.5.3, obliczający kres górny $M_{i_1,...,i_n}$ indeksów $i_1,...,i_n$, znajdujący zbiór osiągalny układu (3.3) przy danej liczbie wymuszeń p. Opiera się on na tym, że danemu kresowi górnemu $M_{i_1,...,i_n}$ indeksów $i_1,...,i_n$ (ograniczenie (2.18) dziedziny równania diofantycznego) odpowiada ściśle określona maksymalna krotność r_{α}^{\max} wartości własnych λ_{α} . Dlatego można podać zależność:

$$r_{\alpha}^{\max} = r_{\alpha}^{\max}(M_{i_1,\dots,i_n})$$

Ponadto, każdy kres górny $M_{i_1,...,i_n}$ indeksów $i_1,...,i_n$ determinuje liczbę odpowiadających im różnych wartości własnych λ_{α} , $\alpha = 1,...,\alpha_{max}$. Dlatego można podać kolejną zależność:

$$\alpha_{\max} = \alpha_{\max}(M_{i_1,\dots,i_n})$$

Rozumowanie można podsumować w poniższym algorytmie.

3.5.3. Algorytm

```
1.Wprowadź:
    - liczbę wymuszeń: p:integer,
```

- liczbę wymiarów przestrzennych: n:integer,
- wymiary boków prostopadłościanu: $a_k:real,\ k=1,2,\ldots,n$.
- 2. Podstaw $m_k := a_k^2, k = 1, 2, ..., n$.
- 3.Podstaw $M_{i_1..i_n} := 2$.

```
4.Wykonaj algorytm 2.8.

5.Oblicz krotność wartości własnych r_{\alpha}, \alpha = 1, ..., \alpha_{\max} korzystając ze wzoru (2.76).

6.Wyznacz maksimum r_{\alpha}^{\max} krotności wartości własnych r_{\alpha}.

7.Jeżeli r_{\alpha}^{\max} \leq p podstaw M_{i_{1}..i_{n}} \coloneqq 2*M_{i_{1}..i_{n}}, przejdź do 4.

8.Jeżeli r_{\alpha}^{\max} > p podstaw M_{i_{1}..i_{n}} \coloneqq M_{i_{1}..i_{n}}/2.

9.Koniec.
```

Algorytm 3.5.3 oblicza kres górny $M_{i_1,...,i_n}$, więc sprawdzając równość (3.21) dla skończonego zbioru indeksów $i_1,...,i_n \leq M_{i_1,...,i_n}$ możemy określić, ile wyrazów z nieskończonego szeregu Fouriera (3.19) składa się na zbiór osiągalny badanego układu dynamicznego. Mianowicie, zbiór osiągalny x(z,t) układu dynamicznego modelowanego za pomocą równania różniczkowego cząstkowego (3.3) ma postać:

$$K = \left\{ x(z,t) : x(z,t) = \sum_{\alpha=1}^{\alpha_{\max}(M_{i_1,\dots,i_n})} \sum_{j=1}^{r_{\alpha}} x_{\alpha}^{(j)}(t) \cdot \phi_{\alpha_j}(z) \right\}$$
(3.24)

3.5.4. Przykład analizy numerycznej układu parabolicznego

• Model matematyczny badanego układu

Weźmy pod uwagę układ dynamiczny opisany następującym liniowym równaniem różniczkowym cząstkowym drugiego rzędu:

$$\frac{\partial x(z,t)}{\partial t} = \frac{\partial^2 x(z,t)}{\partial z_1^2} + \frac{\partial^2 x(z,t)}{\partial z_2^2} + \frac{\partial^2 x(z,t)}{\partial z_3^2} + \sum_{l=1}^6 b_l(z) u_l(t)$$
(3.25)

gdzie $z = \begin{bmatrix} z_1 & z_2 & z_3 \end{bmatrix}$. Dziedziną D układu (3.25) jest prostopadłościan:

$$D = \left\{ \left(z_1, \ z_2, \ z_3 \right) \in \mathbb{R}^3 = \begin{bmatrix} 0, \ 1 \end{bmatrix} \times \begin{bmatrix} 0, \ \sqrt{3} \end{bmatrix} \times \begin{bmatrix} 0, \ \sqrt{5} \end{bmatrix} \right\}$$
(3.26)

Zmienna czasowa spełnia warunek:

$$0 \le t < \infty \tag{3.27}$$

Ponadto, dane są funkcje pomocnicze $b_l(z) \in L^2(D)$, l = 1,...,6 w postaci następującego iloczynu:

$$b_l(z) = f_{l1}(z_1) f_{l2}(z_2) f_{l3}(z_3), \quad l = 1,.., 6$$
 (3.28)

Założymy zerowe warunki brzegowe typu Dirichleta:

$$x(z,t) = 0\Big|_{z \in \Gamma}$$
(3.29)

gdzie Γ jest brzegiem dziedziny *D*. Celem przykładu jest znalezienie kresu górnego $M_{i_1,...,i_n}$ dla indeksów $i_1,...,i_n$, określającego postać zbioru osiągalnego dla układu (3.25) przy liczbie wymuszeń p = 6.

• Rozwiązanie

Prostopadłościan, będący dziedziną układu (3.25), ma boki o długościach:

3.5. Analiza numeryczna własności nieskończenie wymiarowego układu parabolicznego 45

$$a_1 = 1, \quad a_2 = \sqrt{3}, \quad a_3 = \sqrt{5}$$
 (3.30)

Dlatego w kroku 2 algorytmu 3.5.3 mamy:

$$m_1 = 1, \quad m_2 = 3, \quad m_3 = 5$$
 (3.31)

Wykonując drugi przebieg pętli algorytmu 3.5.3, zmienna $M_{i_1,...,i_n}$ przyjmuje wartość 4. Otrzymujemy dwie potrójne wartości własne:

$$\lambda_7 = -9.5(3)\pi^2, \ \lambda_9 = -11.1(3)\pi^2$$
(3.32)

Wszystkie uzyskane sumy kwadratów i ich krotności dla $M_{i_1,...,i_n} = 4$ przedstawiono w tabeli 2. Odpowiednie wartości własne są równe wartości sumy *S* pomnożonej przez stały czynnik $-\pi^2$.

(i_1,i_2,i_3)	(j_1, j_2, j_3)	S	α	Krotność r_{α}	
(1,1,3)	(1,2,2)	3.1(3)	1	2	
(2,1,1)	(1,1,4)	4.5(3)	2	2	
(2,2,1)	(1,2,4)	5.5(3)	3	2	
(2,1,3)	(2,2,2)	6.1(3)	4	2	
(2,2,3)	(1,4,2)	7.1(3)	5	2	
(2,3,1)	(1,3,4)	7.2	6	2	
(2,4,1)	(1,4,4)				
(3,1,1)	(2,4,1)	9.5(3)	7	3	
(3,1,1)	(1,4,4)				
(3,1,2)	(2,4,2)	10.1(3)	8	2	
(3,1,3)	(3,2,2)				
(3,2,2)	(2,4,3)	11.1(3)	9	3	
(3,1,3)	(2,4,3)				
(3,1,4)	(2,4,4)	12.5(3)	10	2	
(4,2,1)	(3,4,4)	17.5(3)	11	2	
(4,1,3)	(4,2,2)	18.1(3)	12	2	

Krotności wartości własnych

Tabela 2

Wykonanie kolejnej iteracji algorytmu 3.5.3 dla $M_{i_1,...,i_n} = 8$ generuje już wartość własną o krotności $r_{\alpha}^{\max} = 8$, więc nie jest spełniona nierówność (3.22), gdyż krotność wartości własnych jest wtedy już większa od liczby funkcji wymuszających. Zbiór osiągalny układu dynamicznego modelowanego równaniem różniczkowym cząstkowym (3.3) można wyznaczyć z następującej sumy:

$$K = \left\{ x(z,t) : x(z,t) = \sum_{\alpha=1}^{12} \sum_{j=1}^{r_{\alpha}} x_{\alpha}^{(j)}(t) \cdot \phi_{\alpha j}(z) \right\}$$
(3.33)

gdzie $x_{\alpha}(t)$ obliczamy z równania (3.20).

3.6. Podsumowanie rozdziału

Rozdział 3 jest poświęcony wprowadzeniu metod numerycznych do dziedziny badań własności wybranej klasy układów nieskończenie wymiarowych. Jako badaną własność wybrano zbiory osiągalne badanych układów. Jako klasę badanych układów wybrano układy dane równaniami różniczkowymi cząstkowymi drugiego rzędu typu parabolicznego przy zerowych warunkach brzegowych typu Dirichleta określonych w n-wymiarowym wielościanie.

W trakcie badań okazało się, że analiza zbiorów osiągalnych wybranego typu układów dynamicznych wymaga zastosowania metod algorytmicznych. Są one potrzebne do rozwiązania odpowiedniego nieliniowego równania diofantycznego, którego budowa odpowiada postaci wartości własnych różniczkowego operatora stanu badanego równania. Zaproponowany w niniejszym rozdziale algorytm wyznaczania zbiorów osiągalnych wywołuje, jako podprocedurę, algorytm rozwiązywania równań diofantycznych przedstawiony w rozdziale 2.

Podsumowując, metody analizy numerycznej pozwoliły wykonać krok naprzód w badaniach nad własnościami układów dynamicznych w stosunku do obecnego stanu wiedzy uzyskanego na drodze badań czysto analitycznych.

4. EFEKTYWNY ALGORYTM NUMERYCZNY OBLICZANIA PEWNEJ KLASY FUNKCJI SYMETRYCZNYCH

4.1. Wprowadzenie

Niniejszy rozdział jest poświęcony przedstawieniu efektywnego algorytmu obliczania tzw. wielomianów symetrycznych podstawowych. Pozwalają one na obliczenie współczynników występujących przy kolejnych potęgach argumentu wielomianu o danych pierwiastkach. Ponadto, znajdują one zastosowanie w konstrukcji algorytmu obliczania tzw. konfluentnych macierzy Vandermonde'a, czemu jest poświęcony kolejny rozdział.

W literaturze podejmowano już próby syntezy algorytmu obliczania wielomianów symetrycznych podstawowych ([19] s. 644). Jednak algorytm zaproponowany w artykule [19] zawiera błąd prowadzący do nieokreślonych wyników.

4.2. Pojęcie wielomianów symetrycznych podstawowych

Najpierw przedstawmy definicję pojęcia wielomianu symetrycznego. Dowolny wielomian nazywamy symetrycznym, jeśli przy dowolnej permutacji jego zmiennych niezależnych w wyniku uzyskujemy ten sam wielomian ([71] s. 279). Na przykład, następujące wielomiany $p(\lambda_1, \lambda_2, \lambda_3)$ trzech zmiennych niezależnych są symetryczne:

$$p(\lambda_1, \lambda_2, \lambda_3) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$$
$$p(\lambda_1, \lambda_2, \lambda_3) = \lambda_1^3 + \lambda_2^3 + \lambda_3^3$$
$$p(\lambda_1, \lambda_2, \lambda_3) = \lambda_1 \lambda_2 + \lambda_1 \lambda_3 + \lambda_2 \lambda_3$$

gdyż są spełnione tożsamościowo równości:

$$p(\lambda_1, \lambda_2, \lambda_3) = p(\lambda_1, \lambda_3, \lambda_2) = p(\lambda_2, \lambda_1, \lambda_3) =$$
$$= p(\lambda_2, \lambda_3, \lambda_1) = p(\lambda_3, \lambda_1, \lambda_2) = p(\lambda_3, \lambda_2, \lambda_1)$$

Pojęcie wielomianów symetrycznych podstawowych jest ściśle związane z następującym wielomianem:

$$p_{0n}(s) = (s - \lambda_1)(s - \lambda_2)...(s - \lambda_n)$$
(4.1)

Po rozwinięciu wielomianu $p_{0n}(s)$ i uporządkowaniu względem malejących potęg zmiennej s otrzymamy:

$$p_{0n}(s) = s^{n} - w_{1}^{(n)}(\lambda_{1},...,\lambda_{n})s^{n-1} + w_{2}^{(n)}(\lambda_{1},...,\lambda_{n})s^{n-2} + ... + (-1)^{n}w_{n}^{(n)}(\lambda_{1},...,\lambda_{n})$$
(4.2)

gdzie $w_j^{(n)}(\lambda_1,...,\lambda_n)$ są wielomianami symetrycznymi podstawowymi zdefiniowanymi przez wzór ([71] s. 280):

$$w_{1}^{(n)}(\lambda_{1},...,\lambda_{n}) = \lambda_{1} + \lambda_{2} + ... + \lambda_{n}$$

$$w_{2}^{(n)}(\lambda_{1},...,\lambda_{n}) = \lambda_{1}\lambda_{2} + \lambda_{1}\lambda_{3} + ... + \lambda_{1}\lambda_{n} + ... + \lambda_{n-1}\lambda_{n}$$

$$w_{3}^{(n)}(\lambda_{1},...,\lambda_{n}) = \lambda_{1}\lambda_{2}\lambda_{3} + \lambda_{1}\lambda_{2}\lambda_{4} + ... + \lambda_{n-2}\lambda_{n-1}\lambda_{n}$$

$$...$$

$$w_{n}^{(n)}(\lambda_{1},...,\lambda_{n}) = \lambda_{1}\lambda_{2} \cdot ... \cdot \lambda_{n}$$

$$(4.3)$$

Algorytmowi obliczania wielomianów symetrycznych podstawowych (4.3) jest poświęcony niniejszy rozdział.

4.3. Algorytm obliczania wielomianów symetrycznych podstawowych

Proponowany algorytm można zapisać w pseudokodzie następująco:

```
Enter:

\lambda[1,...,n]:real - pierwiastki wielomianu p_{0n}(s)

n:integer - liczbę pierwiastków wielomianu p_{0n}(s)

W[1,1]:=\lambda[1]

For i:=1 To n-1

W[i+1,1]:=W[i,1]+\lambda[i+1]

For j:=2 To i

W[i+1,j]:=W[i,j]+W[i,j-1]*\lambda[i+1]

Next j

W[i+1,i+1]:=W[i,i]*\lambda[i+1]

Next i

Return:

W[n,j]:real, j=1,...,n - wartości wielomianów symetrycznych

podstawowych w_i^{(n)}(\lambda_1,...,\lambda_n)
```

Dowód poprawności algorytmu

Załóżmy, że został obliczony następujący ciąg wielomianów symetrycznych podstawowych:

$$W_1^{(n)}, W_2^{(n)}, \dots, W_n^{(n)}$$
 (4.4)

Powyższy ciąg (4.4) odpowiada wielomianowi $p_{0n}(s) = (s - \lambda_1)(s - \lambda_2)...(s - \lambda_n).$ Naszym celem jest znalezienie ciągu $w_1^{(n+1)}, w_2^{(n+1)}, ..., w_{n+1}^{(n+1)}$. Na podstawie definicji odpowiada on następującemu wielomianowi:

$$p_{0(n+1)}(s) = (s - \lambda_{n+1}) p_{0n}(s)$$
(4.5)

Wykonajmy mnożenie w formule (4.5). Korzystając z zależności (4.2), otrzymujemy:

$$p_{0(n+1)}(s) = (s - \lambda_{n+1}) p_{0n}(s) = (s - \lambda_{n+1}) \left[s^{n} - w_{1}^{(n)} s^{n-1} + w_{2}^{(n)} s^{n-2} + \dots + (-1)^{n} w_{n}^{(n)} \right] =$$

$$= s^{n+1} - \left[w_{1}^{(n)} - \lambda_{n+1} \right] s^{n} + \left[w_{2}^{(n)} + w_{1}^{(n)} \lambda_{n+1} \right] s^{n-1} + \left[w_{3}^{(n)} + w_{2}^{(n)} \lambda_{n+1} \right] s^{n-2} + \dots + (-1)^{n+1} w_{n}^{(n)} \lambda_{n+1}$$
(4.6)

Jednocześnie wielomian $p_{0(n+1)}(s)$ można wyrazić w postaci:

$$p_{0(n+1)}(s) = s^{n+1} - w_1^{(n+1)}s^n + w_2^{(n+1)}s^{n-1} - w_3^{(n+1)}s^{n-2} + \dots + (-1)^{n+1}w_{n+1}^{(n+1)}$$
(4.7)

Porównując wielomian $p_{0(n+1)}(s)$ zapisany w postaci (4.6) do jego równoważnej formy (4.7), otrzymujemy bezpośrednio algorytm 4.3.

Kolejność obliczania wielomianów symetrycznych podstawowych za pomocą algorytmu 4.3 ilustruje schemat na rys. 2.

$$W = \begin{bmatrix} w_1^{(1)}(\lambda_1) & & \\ \downarrow & \searrow & \\ w_1^{(2)}(\lambda_1, \lambda_2) & w_2^{(2)}(\lambda_1, \lambda_2) & \\ \downarrow & \searrow & \downarrow & \searrow & \\ \vdots & \vdots & \ddots & \\ \downarrow & & \downarrow & & \searrow & \\ w_1^{(n)}(\lambda_1, \dots, \lambda_n) & w_2^{(n)}(\lambda_1, \dots, \lambda_n) & \cdots & w_n^{(n)}(\lambda_1, \dots, \lambda_n) \end{bmatrix}$$

Rys. 2. Kolejność obliczeń w algorytmie 4.3

Fig. 2. The calculation order in the algorithm 4.3

4.4. Złożoność obliczeniowa algorytmu

Złożoność obliczeniowa algorytmu 4.3 względem liczby zmiennych λ_i (4.3) może być obliczona następująco:

$$T(n) = \sum_{i=1}^{n-1} (i-1) = \frac{0+n-2}{2} (n-1) = O(n^2)$$
(4.8)

gdzie T(n) oznacza konieczną do wykonania liczbę mnożeń zmiennopozycyjnych. Podsumowując, algorytm 4.3 ma wielomianową złożoność obliczeniową klasy $O(n^2)$.

4.5. Przykład wykonania algorytmu

Dany jest następujący wielomian dziesiątego rzędu:

$$p_{0,10}(s) = (s+0.5)(s+3)^2(s+2)^3(s+1)^4$$
(4.9)

Wielomian ten ma następujące pierwiastki:

.

Tabela 3

	Pierwiastki wielomianu (4.9)											
i	1	2	3	4	5	6	7	8	9	10		
λ_i	-0.5	-3.0	-3.0	-2.0	-2.0	-2.0	-1.0	-1.0	-1.0	-1.0		

Zadanie jest następujące:

należy znaleźć wartości wielomianów symetrycznych podstawowych $w_i^{(10)}(\lambda_1,...,\lambda_{10}), j = 1,..,10$ danych wzorem (4.3) za pomocą algorytmu 4.3.

Tabela 4 pokazuje kolejne iteracje wykonania algorytmu 4.3. Komórka tabeli o współrzędnych *ij* zawiera wartość wielomianu $w_j^{(i)}(\lambda_1,...,\lambda_i)$.

Tabela 4

									1 uoviu	
Przykład v	vykonani	a algorytı	mu oblicz	zania wie	lomianóv	v symetry	ycznych	podstawc	wych	
:										

i j	1	2	3	4	5	6	7	8	9	10
1	-0.50									
2	-3.50	1.50								
3	-6.50	12.00	-4.50							
4	-8.50	25.00	-28.50	9.00						
5	-10.50	42.00	-78.50	66.00	-18.00					
6	-12.50	63.00	-162.50	223.00	-150.00	36.00				
7	-13.50	75.50	-225.50	385.50	-373.00	186.00	-36.00			
8	-14.50	89.00	-301.00	611.00	-758.50	559.00	-222.00	36.00		
9	-15.50	103.50	-390.00	912.00	-1369.50	1317.50	-781.00	258.00	-36.00	
10	-16.50	119.00	-493.50	1302.00	-2281.50	2687.00	-2098.50	1039.00	-294.00	36.00

W ostatnim wierszu znajdują się szukane wartości wielomianów symetrycznych podstawowych $w_j^{(10)}(\lambda_1, ..., \lambda_{10}), j = 1, ..., 10$. Są one równe:

$w_1^{(10)}(\lambda_1,,\lambda_{10}) = -16.50$	$\left(w_6^{(10)}(\lambda_1,,\lambda_{10}) = 2687.00 \right)$	
$w_2^{(10)}(\lambda_1,,\lambda_{10}) = 119.00$	$w_7^{(10)}(\lambda_1,,\lambda_{10}) = -2098.50$	
$w_3^{(10)}(\lambda_1,,\lambda_{10}) = -493.50$	$\begin{cases} w_8^{(10)}(\lambda_1,,\lambda_{10}) = 1039.00 \end{cases}$	(4.10)
$w_4^{(10)}(\lambda_1,,\lambda_{10}) = 1302.00$	$w_9^{(10)}(\lambda_1,,\lambda_{10}) = -294.00$	
$w_5^{(10)}(\lambda_1,,\lambda_{10}) = -2281.50$	$w_{10}^{(10)}(\lambda_1,,\lambda_{10}) = 36.00$	

50

4.6. Podsumowanie rozdziału

Najważniejszym rezultatem rozdziału jest zaproponowanie efektywnego algorytmu numerycznego obliczania wartości wielomianów symetrycznych podstawowych. Uzyskano wielomianową złożoność obliczeniową algorytmu równą $O(n^2)$. Zaprezentowany w rozdziale algorytm stanowi podstawę algorytmów numerycznych przedstawionych w dalszej części pracy.

5. ALGORYTM ODWRACANIA PEWNEJ KLASY MACIERZY O SPECJALNEJ POSTACI

5.1. Wprowadzenie

W wielu działach matematyki, analizy numerycznej i inżynierii znajduje zastosowanie macierz o specjalnej postaci, zwana macierzą Vandermonde'a. Niniejszy rozdział zajmuje się jej uogólnieniem, tj. konfluentną macierzą Vandermonde'a. Polega ono na tym, że w kolumnach konfluentnej macierzy Vandermonde'a oprócz kolejnych potęg różnych pierwiastków znajdują się pochodne tychże kolumn przemnożone przez pewien współczynnik. Formalną definicję konfluentnej macierzy Vandermonde'a podano w kolejnych punktach.

Celem niniejszego rozdziału jest przedstawienie efektywnego algorytmu numerycznego znajdowania odwrotności konfluentnej macierzy Vandermonde'a. W tym celu wykorzystano prace z literatury, podające twierdzenia służące jej analitycznemu wyznaczaniu. Jednak dotychczas znane w literaturze twierdzenia znajdowania tejże odwrotności wymagały obliczeń symbolicznych. Niniejszy rozdział stanowi istotny krok naprzód w tej problematyce, gdyż uzyskany algorytm numeryczny jest gotowy do efektywnego wykonania na drodze numerycznej. Wyniki niniejszego rozdziału bazują na rezultatach przedstawionych w publikacji [82].

5.2. Pojęcie konfluentnej macierzy Vandermode'a

Najpierw przytoczmy definicję klasycznej macierzy Vandermonde'a. Można ją skojarzyć z następującym wielomianem o pojedynczych pierwiastkach rzeczywistych:

$$p_{0n}(s) = (s - \lambda_1)(s - \lambda_2)...(s - \lambda_n)$$

$$(5.1)$$

Macierz Vandermonde'a dla takiego wielomianu definiuje się jako:

$$V_{0n} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \lambda_1 & \lambda_2 & \cdots & \lambda_n \\ \lambda_1^2 & \lambda_2^2 & \cdots & \lambda_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^{n-1} & \lambda_2^{n-1} & \cdots & \lambda_n^{n-1} \end{bmatrix}$$
(5.2)

Odwrotność klasycznej macierzy Vandermonde'a (5.2) jest znana i wyraża się w jawnej postaci za pomocą wzoru analitycznego ([27] s. 86).

Przejdźmy teraz do konfluentnej macierzy Vandermonde'a. Podobnie jak jej klasyczny odpowiednik (5.2), można ją skojarzyć z wielomianem, jednak sedno uogólnienia polega na występowaniu w nim pierwiastków wielokrotnych.

5.2.1. Definicja konfluentnej macierzy Vandermonde'a ([31, 82])

Niech $\lambda_1, \lambda_2, ..., \lambda_r$ oznaczają pierwiastki następującego wielomianu:

$$p(s) = (s - \lambda_1)^{n_1} \dots (s - \lambda_r)^{n_r}, \quad n_1, \dots, n_r \in Z_+$$
(5.3)

gdzie $n_1 + ... + n_r = n$. Konfluentna macierz Vandermonde'a V odpowiadająca pierwiastkom wielomianu p(s) jest $n \times n$ -wymiarową macierzą:

$$V = \begin{bmatrix} V_1 V_2 \dots V_r \end{bmatrix}$$
(5.4)

gdzie bloki V_k są macierzami o wymiarach $n \times n_k$ o następujących elementach:

$$\begin{bmatrix} V_k(\lambda_k, n_k) \end{bmatrix}_{ij} = \begin{cases} \binom{i-1}{j-1} \lambda_k^{i-j}, & dla \ i \ge j \\ 0, & w \ przeciwnym \ przypadku \end{cases}$$
(5.5)

 $dla \ k = 1, 2, ..., r; \ i = 1, 2, ..., n \ oraz \ j = 1, 2, ..., n_k$

5.2.2. Przykłady konfluentnych macierzy Vandermonde'a

• Dla wielomianu o postaci:

$$p(s) = (s+2)^3(s-3)$$

którego pierwiastki $\lambda_1 = -2$, $n_1 = 3$ oraz $\lambda_2 = 3$, $n_2 = 1$ konfluentna macierz Vandermonde'a ma postać:

$$V = \begin{bmatrix} 1 & 0 & 0 & 1 \\ -2 & 1 & 0 & 3 \\ 4 & -4 & 1 & 9 \\ -8 & 12 & -6 & 27 \end{bmatrix}$$

• Dla wielomianu o postaci:

 $p(s) = (s-1)^2(s-2)^2$

którego pierwiastki $\lambda_1 = 1$, $n_1 = 2$ oraz $\lambda_2 = 2$, $n_2 = 2$ konfluentna macierz Vandermonde'a ma postać:

$$V = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 1 \\ 1 & 2 & 4 & 4 \\ 1 & 3 & 8 & 12 \end{bmatrix}$$

5.3. Analityczna metoda odwracania konfluentnych macierzy Vandermode'a

Twierdzenie na analityczną metodę odwracania konfluentnej macierzy Vandermode'a jest omówione w publikacji [31].

5.3.1. Twierdzenie ([31] s. 1543)

Odwrotność konfluentnej macierzy Vandermonde'a V, danej wzorami (5.4) i (5.5), ma następującą postać:

$$V^{-1} = \begin{bmatrix} \mathcal{Q}_1 \\ \mathcal{Q}_2 \\ \vdots \\ \mathcal{Q}_r \end{bmatrix}$$
(5.6)

gdzie wektory kolumnowe h_i w macierzy blokowej $Q_k = [h_{kn}, ..., h_{k1}]$ można obliczyć rekurencyjnie, zgodnie z następującym schematem:

$$\begin{pmatrix}
h_{k1} = K_{k,n_k} \theta_k + K_{k,n_{k-1}} N_k \theta_k + \dots + K_{k,1} (N_k)^{n_k - 1} \theta_k \\
h_{k2} = (\lambda_k I_k + N_k) h_{k1} + a_1 h_{k1} \\
\vdots \\
h_{kn} = (\lambda_k I_k + N_k) h_{k(n-1)} + a_{n-1} h_{k1}
\end{cases}$$
(5.7)

gdzie λ_k oznacza wartość własną, I_k jest macierzą identycznościową o wymiarach $n_k \times n_k$, $N_k = J_k(0, n_k)$ (obliczaną z równości (5.8)), a_k są współczynnikami wielomianu p(s)określonego przez (5.3), $\theta_k = [0, ..., 0, 1]^T$, dim $\theta_k = n_k$ oraz $K_{k,j}$ są współczynnikami rozwinięcia ułamkowego odwrotności wielomianu p(s).

$$J_{k}(\lambda_{k},n_{k}) = \begin{bmatrix} \lambda_{k} & 1 & \cdots & 0 \\ \lambda_{k} & 1 & \vdots \\ & \ddots & \ddots \\ & & \lambda_{k} & 1 \\ 0 & & & \lambda_{k} \end{bmatrix}_{n_{k} \times n_{k}}$$
(5.8)

Konstrukcja algorytmu odwracania konfluentnej macierzy Vandermonde'a bezpośrednio na podstawie twierdzenia 5.3.1 daje nieefektywny algorytm. Najbardziej czasochłonne jest obliczenie wektora h_{k1} , wymagające obliczenia wszystkich kolejnych potęg $(N_k)^i$ macierzy N_k . Mnożenie macierzy jest operacją o złożoności $O(n^3)$, więc wyznaczenie wszystkich potęg macierzy N_k jest operacją o złożoności $O(n^4)$. Dlatego dalsza część rozdziału zajmuje się konstrukcją algorytmu odwracania konfluentnej macierzy Vandermonde'a przy wykorzystaniu optymalizacji twierdzenia 5.3.1.

5.4. Synteza numerycznego algorytmu odwracania konfluentnych macierzy Vandermonde'a

W syntezie odpowiedniego algorytmu korzystamy z twierdzenia 5.3.1. W pierwszej kolejności poddajmy optymalizacji wzór wyrażający wektor h_{k1} , tj. równość (5.7).

 Może on zostać znacznie zoptymalizowany po spostrzeżeniu, że spełniona jest następująca tożsamość macierzowa:

$$(N_k)^j \theta_k = \begin{bmatrix} 0 \cdots 1 \cdots 0 \\ \vdots \ddots \vdots \ddots \vdots \\ 0 \cdots 0 \cdots 1 \end{bmatrix}$$
$$j \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \cdots 0 1 \\ 0 \\ \vdots \\ 1 \end{bmatrix}^T, \quad j = 0, 1, ..., n_k - 1$$
(5.9)

Ponadto, z użyciem wzoru (5.8) występujący wielokrotnie wyraz (λ_kI_k + N_k) można zapisać krócej jako J_k (λ_k, n_k).

Uwzględniając powyższe dwa spostrzeżenia, twierdzenie 5.3.1 można zastąpić następującym twierdzeniem:

5.4.1. Twierdzenie [82]

Wektory kolumnowe h_{ki} macierzy blokowej $Q_k = [h_{kn}, ..., h_{k1}] w V^{-1} = [Q_1^T, ..., Q_r^T]^T$ można wyznaczyć z następującego schematu rekurencyjnego:

$$h_{k1} = \begin{bmatrix} K_{k,1} \cdots K_{k,n_k} \end{bmatrix}^T$$

$$h_{k2} = J_k (\lambda_k, n_k) h_{k1} + a_1 h_{k1}$$

$$\vdots$$

$$h_{kn} = J_k (\lambda_k, n_k) h_{k(n-1)} + a_{n-1} h_{k1}$$
(5.10)

5.5. Wyznaczanie wielkości wejściowych algorytmu odwracania konfluentnych macierzy Vandermonde'a

Weźmy pod uwagę współczynniki $K_{k,j}$ występujące w równaniu (5.10). Twierdzenie 5.3.1 podaje, że są one współczynnikami rozwinięcia ułamkowego odwrotności wielomianu p(s):

$$\frac{1}{p(s)} = \sum_{k=1}^{r} \sum_{j=0}^{n_k-1} \frac{K_{k,j+1}}{\left(s - \lambda_k\right)^{j+1}}$$
(5.11)

Można je wyznaczyć korzystając ze znanych reguł algebry liniowej poprzez różniczkowanie:

$$K_{k,j+1} = \frac{1}{(n_k - j - 1)!} \frac{d^{(n_k - j - 1)}}{ds^{(n_k - j - 1)}} \left[\frac{1}{p(s)} (s - \lambda_k)^{n_k} \right]_{s = \lambda_k}, \qquad j = 0, 1, \dots, n_k - 1 \qquad (5.12)$$

Korzystając z rozwinięcia (5.12), wektor kolumnowy (5.10) można obliczyć jak następuje:

$$h_{k1} = \begin{bmatrix} K_{k,1} \\ \vdots \\ K_{k,j+1} \\ \vdots \\ K_{k,n_k} \end{bmatrix} = \begin{bmatrix} \frac{1}{(n_k - 1)!} \frac{d^{(n_k - 1)}}{ds^{(n_k - 1)!}} \begin{bmatrix} \frac{1}{p(s)} (s - \lambda_k)^{n_k} \end{bmatrix}_{s = \lambda_k} \\ \vdots \\ \frac{1}{(n_k - j - 1)!} \frac{d^{(n_k - j - 1)}}{ds^{(n_k - j - 1)}} \begin{bmatrix} \frac{1}{p(s)} (s - \lambda_k)^{n_k} \end{bmatrix}_{s = \lambda_k} \\ \vdots \\ \begin{bmatrix} \frac{1}{p(s)} (s - \lambda_k)^{n_k} \end{bmatrix}_{s = \lambda_k} \end{bmatrix}, \quad k = 1, 2, ..., r \quad (5.13)$$

Korzystając z twierdzenia 5.4.1 można skonstruować także wzór analityczny (5.14), przedstawiający wprost postać kolejnych wektorów w odwrotności konfluentnej macierzy Vandermonde'a:

$$h_{ki} = \left[J_k^{i-1}(\lambda_k, n_k) + \sum_{j=1}^{i-1} a_j J_k^{i-j-1}(\lambda_k, n_k)\right] h_{k1}, \quad k = 1, 2, ..., r, \quad i = 2, 3, ..., n$$
(5.14)

gdzie $J_k^q(\lambda_k, n_k)$ są potęgami elementarnych klatek Jordana (5.8), których postać jest znana w literaturze ([6, 39]):

$$J_{k}^{q}(\lambda_{k},n_{k}) = \begin{bmatrix} \lambda_{k} & 1 & \cdots & 0 \\ \lambda_{k} & 1 & \vdots \\ & \ddots & \ddots & \\ & & \lambda_{k} & 1 \\ 0 & & & \lambda_{k} \end{bmatrix}_{n_{k} \times n_{k}}^{q} = \begin{bmatrix} \lambda_{k}^{q} & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(1)}}{1!} & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(2)}}{2!} & \cdots & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(n_{k}-1)}}{(n_{k}-1)!} \\ & \lambda_{k}^{q} & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(1)}}{1!} & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(2)}}{2!} & \vdots \\ & & \ddots & \ddots & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(2)}}{2!} \\ & & \lambda_{k}^{q} & \frac{(\lambda_{k}^{q})_{\lambda_{k}}^{(1)}}{1!} \\ 0 & & & \lambda_{k}^{q} \end{bmatrix}_{n_{k} \times n_{k}}^{n}$$

$$(5.15)$$

Ponadto, z algebry liniowej wiadomo, że współczynniki rozwinięcia wielomianu p(s) wyrażają się następująco:

$$a_{j} = (-1)^{j} w_{j}^{(n)}(\lambda_{1},...,\lambda_{n}), \quad j = 1, 2, ..., n$$
(5.16)

gdzie $w_j^{(n)}(\lambda_1,...,\lambda_n)$ są wielomianami symetrycznymi podstawowymi, zdefiniowanymi przez wzór (4.3). Efektywny algorytm numerycznego ich obliczania przedstawiono w rozdziale 4 niniejszej monografii.

5.6. Finalne algorytmy odwracania konfluentnych macierzy Vandermonde'a

Wyznaczanie odwrotności konfluentnej macierzy Vandermonde'a można podzielić na następujące etapy:

- wyznaczenie współczynników rozwinięcia wielomianu *p*(*s*) za pomocą algorytmu obliczania wielomianów symetrycznych podstawowych, przedstawionego w poprzednim rozdziale, oraz formuły (5.16),
- wyznaczenie ostatniej kolumny szukanej macierzy odwrotnej,
- obliczenie jej pozostałych kolumn.

W następnych podpunktach przedstawiono odpowiednie algorytmy, realizujące ostatnie dwa zadania.

5.6.1. Algorytm wyznaczający ostatnią kolumnę szukanej macierzy odwrotnej

W niniejszym podpunkcie przedstawiono algorytm obliczający wektory h_{k1} , k = 1, ..., rw macierzy blokowej $Q_k = [h_{kn}, ..., h_{k1}]$ w $V^{-1} = [Q_1^T, ..., Q_r^T]^T$. Algorytm można zapisać w pseudokodzie następująco:

```
Enter:
n:integer
                   - stopień wielomianu p(s)
                   - liczba różnych pierwiastków wielomianu p(s)
r:integer
\lambda[1,..,r]:real
                   - pierwiastki wielomianu p(s)
n[1,..,r]:integer – krotności pierwiastków wielomianu p(s)
{ Lokalne zmienne pomocnicze }
var
   i, k, q, delta:integer
   tmp, factorial, L[1,..,r], AuxTab[1,..,r]:real
{Właściwe obliczenia}
delta:=0
For k:=1 To r
   tmp:=1
   { Wyznacz K_{k,n_k} }
   For i:=1 To r
       if i=k continue
       tmp := tmp*(\lambda[k] - \lambda[i])^n[i]
   Next i
   Q[n[k]+delta,n] := 1/tmp
   factorial := 1
   For i:=1 To r AuxTab[i] := 1 Next i
    { Wykonaj schemat rekurencyjny (5.17) dla danego pierwiastka }
   For q:=0 To n[k]
       { Oblicz L_{ki}^{(q+1)}
       For i:=1 To r
          if i=k continue
          L[i] := factorial*AuxTab[i]*Q[n[k]-q+delta,n]-q*L[i]
       Next i
       tmp :=0
       { Oblicz K_{k,n_k-q-1} }
       For i:=1 To r
          if i=k continue
          AuxTab[i] := AuxTab[i] * (\lambda[k] - \lambda[i])
          tmp := tmp + n[i]*L[i]/AuxTab[i]
       Next i
       factorial := factorial*(q+1)
       Q[n[k]-q-1+delta,n] := -tmp/factorial
   Next a
   delta := delta + n[k]
Next k
Return:
Q[i,n]:real, i=1,...,n - szukana ostatnia kolumna odwrotności konfluentnej
   macierzy Vandermonde'a
```

Dowód poprawności algorytmu

W ostatniej kolumnie szukanej odwrotności konfluentnej macierzy Vandermonde'a znajdują się wektory h_{k1} , k = 1, ..., r, zbudowane ze współczynników $K_{k,j}$ rozwinięcia ułamkowego odwrotności wielomianu p(s) (równości (5.10) i (5.11)). Współczynniki

te wyrażają się wprost poprzez formułę (5.12). Aby dowieść poprawności algorytmu 5.6.1, wpierw zauważmy, że dla każdego z r pierwiastków wielomianu p(s) wykonuje on następujący schemat rekurencyjny:

$$\begin{cases} L_{ki}^{(q+1)}(\lambda_{k}) = q! (\lambda_{k} - \lambda_{i})^{q} \cdot K_{k,n_{k}-q} - q \cdot L_{ki}^{(q)}(\lambda_{k}), & i = 1,..,k-1,k+1,..,r \\ K_{k,n_{k}-q-1} = -\frac{1}{(q+1)!} \sum_{i=1,i\neq k}^{r} n_{i} \frac{L_{ki}^{(q+1)}(\lambda_{k})}{(\lambda_{k} - \lambda_{i})^{q+1}} \end{cases}$$
(5.17)

dla $q = 0, 1, ..., n_k - 2$, natomiast współczynniki K_{k,n_k} oblicza bezpośrednio ze wzoru (5.12). Należy wykazać, że schemat rekurencyjny (5.17) jest równoważny formule (5.12), wyrażającej wartości współczynników K_{k,n_k} . Dowód przeprowadzimy indukcyjnie.

• Krok wstępny: dla q = 0

Z równania (5.12) obliczamy współczynnik K_{k,n_k-1} . W tym celu korzystamy z reguły różniczkowania wielomianu danego w postaci iloczynowej:

$$f(s) = (s - \lambda_1)(s - \lambda_2) \cdot \dots \cdot (s - \lambda_n)$$

Pochodna powyższego wielomianu f(s) ma postać ([71] s.110):

$$\frac{df}{ds} = \sum_{i=1}^{n} \left(s - \lambda_1 \right) \cdot \ldots \cdot \left(s - \lambda_{i-1} \right) \left(s - \lambda_{i+1} \right) \cdot \ldots \cdot \left(s - \lambda_n \right)$$

Na tej podstawie obliczamy współczynnik K_{k,n_k-1} , zgodnie ze wzorem (5.12):

$$K_{k,n_{k}-1} = \frac{1}{1!} \frac{d}{ds} \left[\frac{1}{p(s)} (s - \lambda_{k})^{n_{k}} \right]_{s=\lambda_{k}} = \frac{-\sum_{i=1,i\neq k}^{r} n_{i} (s - \lambda_{i})^{n_{i}-1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}}{\prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{2n_{j}}} \right]_{s=\lambda_{k}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{(s - \lambda_{i})^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}}}{\left| \int_{s=\lambda_{k}}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}} \right|_{s=\lambda_{k}}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{\lambda_{k} - \lambda_{i}} K_{k,n_{k}}}{\left(s - \lambda_{i}\right)^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}} \right|_{s=\lambda_{k}}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}}{\left| \int_{s=\lambda_{k}}^{r} \frac{n_{i}}{\lambda_{k} - \lambda_{i}} K_{k,n_{k}}}{\left(s - \lambda_{i}\right)^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}} \right|_{s=\lambda_{k}}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}}{\left(s - \lambda_{i}\right)^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}} \left| \int_{s=\lambda_{k}}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}}{\left(s - \lambda_{i}\right)^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}} \right|_{s=\lambda_{k}}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}}{\left(s - \lambda_{i}\right)^{n_{i}+1} \prod_{j=1,j\neq i}^{r} (s - \lambda_{j})^{n_{j}}} \right|_{s=\lambda_{k}}} = -\sum_{i=1,i\neq k}^{r} \frac{n_{i}}{s - \lambda_{i}} K_{k,n_{k}}}$$

Zauważmy, że wynik (5.18) jest równoważny formule (5.17) dla q = 0.

• Krok główny: dla $q = 0, 1, ..., n_k - 2$

Traktując wielkości $K_{k,j}$ i $L_{ki}^{(q)}$ jako funkcje zmiennej *s*, współczynniki K_{k,n_k-q-2} można obliczyć z równości (5.12) i (5.17):

$$K_{k,n_{k}-q-2} = \frac{1}{q+2} \cdot \frac{d}{ds} \Big[K_{k,n_{k}-q-1}(s) \Big]_{s=\lambda_{k}} = \frac{1}{q+2} \cdot \frac{d}{ds} \Big[-\frac{1}{(q+1)!} \sum_{i=1, i\neq k}^{r} n_{i} \frac{L_{ki}^{(q+1)}(s)}{(s-\lambda_{i})^{q+1}} \Big]_{s=\lambda_{k}} = -\frac{1}{(q+2)!} \sum_{i=1, i\neq k}^{r} n_{i} \frac{(s-\lambda_{i}) \cdot \frac{d}{ds} L_{ki}^{(q+1)}(s) - (q+1) \cdot L_{ki}^{(q+1)}(s)}{(s-\lambda_{i})^{q+2}} \Big|_{s=\lambda_{k}}$$

$$(5.19)$$

Jest spełniony następujący ciąg równości:

$$\begin{aligned} \frac{d}{ds} L_{ki}^{(q+1)}(s) &= \frac{d}{ds} \Big[(s - \lambda_i)^q q! K_{k,n_k-q}(s) - q \cdot L_{ki}^{(q)}(s) \Big]_{s=\lambda_k} = \\ &= (s - \lambda_i)^q (q+1)! K_{k,n_k-q-1}(s) \Big|_{s=\lambda_k} + \Big[q(s - \lambda_i)^{q-1} q! K_{k,n_k-q}(s) - q \frac{d}{ds} L_{ki}^{(q)}(s) \Big]_{s=\lambda_k} = \\ &= (s - \lambda_i)^q (q+1)! K_{k,n_k-q-1}(s) \Big|_{s=\lambda_k} - \Big[q(q-1)(s - \lambda_i)^{q-2} (q-1)! K_{k,n_k-q+1}(s) - q(q-1) \frac{d}{ds} L_{ki}^{(q-1)}(s) \Big]_{s=\lambda_k} = \\ &: \\ &= (s - \lambda_i)^q (q+1)! K_{k,n_k-q-1}(s) \Big|_{s=\lambda_k} + (-1)^h \Big[q^{h+1} (s - \lambda_i)^{q-h-1} (q-h)! K_{k,n_k-q+h}(s) - q^{h+1} \frac{d}{ds} L_{ki}^{(q-h)}(s) \Big]_{s=\lambda_k} \end{aligned}$$

gdzie q^{-1} oznacza dolną silnię oraz h = 0, 1, ..., q - 1. Podstawiając w powyższej równości h = q - 1 otrzymujemy:

$$\frac{d}{ds} L_{ki}^{(q+1)}(s) \Big|_{s=\lambda_{k}} = (s-\lambda_{i})^{q} (q+1)! K_{k,n_{k}-q-1}(s) \Big|_{s=\lambda_{k}} + (-1)^{q-1} \left[q! (s-\lambda_{i})^{0} 1! K_{k,n_{k}-1}(s) - q! \frac{d}{ds} L_{ki}^{(1)}(s) \right]_{s=\lambda_{k}}$$
(5.20)

Ponadto, ze schematu rekurencyjnego (5.17) wynika następująca równość:

$$L_{ki}^{(1)}(s)\Big|_{s=\lambda_{k}} = K_{k,n_{k}}(s)\Big|_{s=\lambda_{k}}$$
(5.21)

Korzystając z równości (5.21), zależność (5.20) redukuje się do następującej postaci:

$$\frac{d}{ds} L_{ki}^{(q+1)}(s) \Big|_{s=\lambda_{k}} = (s-\lambda_{i})^{q} (q+1)! K_{k,n_{k}-q-1}(s) \Big|_{s=\lambda_{k}} + (-1)^{q-1} \left[q! (s-\lambda_{i})^{0} 1! K_{k,n_{k}-1}(s) - q! \frac{d}{ds} K_{k,n_{k}}(s) \right] \Big|_{s=\lambda_{k}} = (s-\lambda_{i})^{q} (q+1)! K_{k,n_{k}-q-1}(s) \Big|_{s=\lambda_{k}}$$
(5.22)

Wróćmy do szukanych współczynników K_{k,n_k-q-2} . Podstawiając równość (5.22) do zależności (5.19), otrzymujemy ich następującą postać:

$$K_{k,n_{k}-q-2} = -\frac{1}{(q+2)!} \sum_{i=1, i\neq k}^{r} n_{i} \frac{(q+1)!(s-\lambda_{i})^{q+1} K_{k,n_{k}-q-1}(s) - (q+1)L_{ki}^{(q+1)}(s)}{(s-\lambda_{i})^{q+2}} \bigg|_{s=\lambda_{k}} (5.23)$$

Można zauważyć, że po następującym podstawieniu:

$$L_{ki}^{(q+2)}(s)\Big|_{s=\lambda_k} = (q+1)!(s-\lambda_i)^{q+1} K_{k,n_k-q-1}(s)\Big|_{s=\lambda_k} - (q+1)L_{ki}^{(q+1)}(s)\Big|_{s=\lambda_k}$$
(5.24)

zależność (5.23) jest równoważna dowodzonemu schematowi rekurencyjnemu (5.17) dla q := q + 1. Podsumowując, udowodniliśmy prawdziwości schematu rekurencyjnego (5.17) dla dowolnego $q = 0, 1, ..., n_k - 2$.

C.N.D.

Kolejność obliczeń realizowanych przez algorytm 5.6.1, wykonujący obliczenia zgodnie ze schematem rekurencyjnym (5.17), wyjaśnia rys. 3.



k=1,2,...,r

Rys. 3. Diagram obrazujący kolejność obliczeń w algorytmie 5.6.1 Fig. 3. Scheme for the auxiliary coefficients determination algorithm 5.6.1

5.6.2. Algorytm wyznaczający pozostałe kolumny szukanej macierzy odwrotnej

W niniejszym podpunkcie przedstawiono algorytm obliczający wektory $h_{kn},...,h_{k2}, k = 1,...,r$ w macierzy blokowej $Q_k = [h_{kn},...,h_{k1}]$ w $V^{-1} = [Q_1^T,...,Q_r^T]^T$. Algorytm można zapisać w pseudokodzie następująco:

```
Enter:
n:integer
                     - stopień wielomianu p(s)
r:integer
                     – liczba różnych pierwiastków wielomianu p(s)
                     - pierwiastki wielomianu p(s)
\lambda[1,..,r]:real
n[1,..,r]:integer
                     - krotności pierwiastków wielomianu p(s)
Q[i,n]:real, i=1,...,n - szukana odwrotność konfluentnej macierzy Vandermonde'a,
   z wyznaczoną ostatnią kolumną za pomocą algorytmu 5.6.1
a[1,..,n-1]:real – współczynniki rozwinięcia wielomianu p(s) wyznaczone za
   pomocą algorytmu obliczania wielomianów symetrycznych podstawowych 4.3 oraz
   formuły (5.16)
{ Lokalne zmienne pomocnicze }
var
i, j, k, delta:integer
{ Właściwe obliczenia }
   delta :=0:
   For k=1 To r
      For i=n-1 Downto 1
          For j=1 To n[k]-1
             Q[j+delta,i]:=\lambda[k]*Q[j+delta,i+1]+
                Q[j+delta+1,i+1]+a[n-i]*Q[j+delta,n]
          Next j
          Q[n[k]+delta,i]:=\lambda[k]*W[n[k]+delta,i+1]+
            a[n-i]*Q[n[k]+delta,n]
      Next i
       delta := delta + n[k];
   Next k
Return:
Q[i,j]:real, i,j=1,...,n - szukana odwrotność konfluentnej macierzy Vandermonde'a
```

Dowód poprawności algorytmu

Algorytm wykonuje obliczenia zgodnie ze schematem rekurencyjnym (5.10). Zastosowano w tym algorytmie optymalizację czasową podczas mnożenia macierzy przez wektor o postaci $J_k(\lambda_k, n_k)h_k$. Jest to w ogólnym przypadku zadanie klasy $O(n^2)$, jednak elementarny blok Jordana (5.8) jest macierzą bi-diagonalną. Dlatego konieczne mnożenie $J_k(\lambda_k, n_k)h_k$. wykonano na poziomie skalarnym, wykorzystując szczególną postać macierzy $J_k(\lambda_k, n_k)$ uzyskując zadanie klasy O(n).

5.7. Złożoność obliczeniowa algorytmu odwracania konfluentnych macierzy Vandermonde'a

Wyznaczenie odwrotności konfluentnej macierzy Vandermonde'a można podzielić na następujące trzy etapy:

- wyznaczenie współczynników rozwinięcia wielomianu *p(s)*, realizowane za pomocą algorytmu obliczania wielomianów symetrycznych podstawowych 4.3,
- obliczenie pomocniczych współczynników $K_{k,j}$, występujących w ostatniej kolumnie szukanej odwrotności, za pomocą algorytmu 5.6.1,
- obliczenie pozostałych kolumn szukanej odwrotności za pomocą algorytmu 5.6.2.

$$T(n) = \sum_{\substack{k=1\\ obliczanie \ pomocniczych \ wspolczynnikow \ K_{k,j}}}^{r} \sum_{\substack{k=1\\ obliczanie \ pozostałych \ kolumn \ inwersji \ V^{-1}}}^{r} \sum_{\substack{k=1\\ obliczanie \ pozostałych \ kolumn \ inwersji \ V^{-1}}}^{r}$$
(5.25)

gdzie T(n) oznacza liczbę koniecznych do wykonania zmiennopozycyjnych mnożeń skalarnych. Uwzględniając zależność $\sum_{k=1}^{r} n_k = n$ (p. 5.2.1) z zależności (5.25) otrzymujemy:

$$T(n) = (7r-5)\sum_{k=1}^{r} n_k - (7r-5)\sum_{k=1}^{r} 1 + 2(n-1)\sum_{k=1}^{r} n_k = (7r-5)(n-r) + 2(n-1)n = O(n^2)$$

Podsumowując, uzyskana złożoność obliczeniowa klasy $O(n^2)$ jest o czynnik liniowy lepsza od algorytmu odwracania macierzy metodą eliminacji Gaussa.

5.8. Przykład wykonania algorytmu

Rozważmy konfluentną macierz Vandermonde'a o wymiarach 10×10 określoną przez następujący wielomian charakterystyczny:

$$p(s) = (s+0.5)(s+3)^{2}(s+2)^{3}(s+1)^{4}$$
(5.26)

Ma on tę samą postać co wielomian charakterystyczny rozważany w podpunkcie 4.5, gdzie wyznaczano jego wielomiany symetryczne podstawowe. Odpowiadająca mu konfluentna macierz Vandermonde'a, na podstawie definicji 5.2.1, ma następującą postać:

$$V = \begin{bmatrix} 1.00000 & 1.00000 & 0.00000 & 1.00000 & 0.00000 & 0.00000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ -0.50000 & -3.00000 & 1.00000 & -2.0000 & 1.00000 & 0.00000 & -1.0000 & 1.0000 & 0.0000 & 0.0000 \\ 0.25000 & 9.00000 & -6.00000 & 4.00000 & -4.0000 & 1.00000 & 1.0000 & -2.000 & 1.0000 & 0.0000 \\ -0.1250 & -27.0000 & 27.0000 & -8.0000 & 12.0000 & -6.00000 & -1.0000 & 3.0000 & -3.000 & 1.0000 \\ 0.062500 & 81.0000 & -108.000 & 16.0000 & -32.000 & 24.0000 & 1.0000 & -4.000 & 6.0000 & -4.000 \\ -0.03120 & -243.000 & 405.000 & -32.000 & 80.0000 & -80.0000 & -1.0000 & 5.0000 & -10.00 & 10.000 \\ 0.015625 & 729.000 & -1458.00 & 64.0000 & -192.00 & 240.000 & 1.0000 & -6.000 & 15.000 & -20.00 \\ -0.00780 & -2187.00 & 5103.00 & -128.00 & 448.000 & -672.000 & -1.0000 & 7.0000 & -21.00 & 35.000 \\ 0.003906 & 6561.00 & -17496.0 & 256.000 & -1024.0 & 1792.00 & 1.0000 & -8.000 & 28.000 & -56.000 \\ -0.00190 & -19683.0 & 59049.0 & -512.00 & 2304.00 & -4608.00 & -1.0000 & 9.0000 & -36.00 & 84.000 \\ \end{bmatrix}$$

Pierwszym etapem w wyznaczaniu jej odwrotności jest znalezienie współczynników rozwinięcia wielomianu *p(s)* danego wzorem (5.26). Wyrażają się one przez wielomiany symetryczne podstawowe, odpowiadające pierwiastkom wielomianu (5.26), wziętym z odpowiednim znakiem (formuła (5.16)). Same wielomiany symetryczne podstawowe

dla wielomianu (5.26) były obliczane za pomocą algorytmu 4.3 w podpunkcie 4.5. Stosując odpowiednio formułę (5.16) otrzymujemy następujące współczynniki rozwinięcia wielomianu (5.26).

Tabela 5

Tabela 6

w społczynniki rozwinięcia przykładowego wielomianu $p(s)$
--

i	1	2	3	4	5	6	7	8	9	10
a_i	16.5	119.0	493.5	1302.0	2281.5	2687.0	2098.5	1039.0	294.0	36.0

• Kolejnym etapem jest obliczenie pomocniczych współczynników $K_{k,j}$, występujących w ostatniej kolumnie szukanej odwrotności macierzy (5.27). Dokonujemy tego za pomocą algorytmu 5.6.1. Tabela 6 przedstawia przebieg obliczeń wykonywanych przez algorytm 5.6.1 dla przykładowej konfluentnej macierzy Vandermonde'a (5.27).

k	λ_{k}	n_k	q	i	$L_{ki}^{\left(q+1 ight) }\left(\lambda_{k}^{ ight) } ight)$	K_{k,n_k-q-1}													
1	-0.50000	1	-	-	-	$K_{1,1} = 0.75851$													
		2			-	-	-	$K_{2,2} = 0.02500$											
C	2 00000			1	$L_{21}^{(1)}(\lambda_2) = 0.02500$														
Z	-3.00000	2	0	3	$L_{23}^{(1)}(\lambda_2) = 0.02500$	$K_{2,1} = 0.13500$													
				4	$L_{24}^{(1)}(\lambda_2) = 0.02500$														
			-	-	-	$K_{3,3} = -0.666666$													
				1	$L_{31}^{(1)}(\lambda_3) = -0.666666$														
			0	2	$L_{32}^{(1)}(\lambda_3) = -0.666666$	$K_{3,2} = -1.77777$													
3 -2.00000	3		4	$L_{34}^{(1)}(\lambda_3) = -0.666666$															
			1	$L_{31}^{(2)}(\lambda_3) = 3.33333$															
																1	2	$L_{32}^{(2)}(\lambda_3) = -1.11111$	$K_{3,1} = -4.51851$
				4	$L_{34}^{(2)}(\lambda_3) = 2.44444$														
			-	-	-	$K_{4,4} = -0.50000$													
				1	$L_{41}^{(1)}(\lambda_4) = -0.50000$														
			0	2	$L_{42}^{(1)}(\lambda_4) = -0.50000$	$K_{4,3} = 1.00000$													
4 -1.00000	-1.00000	1		3	$L_{43}^{(1)}(\lambda_4) = -0.50000$														
	-1.00000	4		1	$L_{41}^{(2)}(\lambda_4) = 0.00000$														
			1	2	$L_{42}^{(2)}(\lambda_4) = 2.50000$	$K_{4,2} = -2.87500$													
				3	$L_{43}^{(2)}(\lambda_4) = 1.50000$														
			2	1	$L_{41}^{(3)}\left(\lambda_{4}\right) = -1.43750$	$K_{4,1} = 3.62500$													

Przebieg kolejnych iteracji algorytmu 5.6.1

cd. tabeli 6

	2	$L_{42}^{(3)}(\lambda_4) = -28.0000$	
	3	$L_{43}^{(3)}(\lambda_4) = -8.75000$	

 Ostatnim etapem jest obliczenie pozostałych kolumn szukanej odwrotności za pomocą algorytmu 5.6.2. W wyniku uzyskujemy następującą odwrotność macierzy V danej równością (5.27):

$$Q = V^{-1} = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} h_{1(10)}^{(1)} & h_{19}^{(1)} & h_{18}^{(1)} & h_{17}^{(1)} & h_{16}^{(1)} & h_{15}^{(1)} & h_{14}^{(1)} & h_{13}^{(1)} & h_{12}^{(1)} & K_{1,1} \\ h_{2(10)}^{(1)} & h_{29}^{(2)} & h_{28}^{(2)} & h_{27}^{(2)} & h_{26}^{(2)} & h_{25}^{(2)} & h_{24}^{(2)} & h_{23}^{(2)} & h_{22}^{(2)} & K_{2,2} \\ h_{2(10)}^{(1)} & h_{29}^{(2)} & h_{28}^{(2)} & h_{27}^{(2)} & h_{26}^{(2)} & h_{25}^{(2)} & h_{24}^{(2)} & h_{23}^{(2)} & h_{22}^{(2)} & K_{2,2} \\ h_{2(10)}^{(1)} & h_{39}^{(1)} & h_{38}^{(1)} & h_{37}^{(1)} & h_{36}^{(1)} & h_{35}^{(1)} & h_{34}^{(1)} & h_{31}^{(1)} & h_{32}^{(1)} & K_{3,1} \\ h_{2(10)}^{(1)} & h_{39}^{(2)} & h_{38}^{(2)} & h_{37}^{(2)} & h_{36}^{(2)} & h_{35}^{(2)} & h_{34}^{(2)} & h_{33}^{(2)} & K_{3,2} \\ h_{3(10)}^{(1)} & h_{39}^{(2)} & h_{38}^{(2)} & h_{37}^{(2)} & h_{36}^{(2)} & h_{35}^{(2)} & h_{34}^{(2)} & h_{33}^{(2)} & h_{32}^{(2)} & K_{3,2} \\ h_{3(10)}^{(3)} & h_{39}^{(3)} & h_{38}^{(3)} & h_{37}^{(3)} & h_{36}^{(3)} & h_{35}^{(3)} & h_{34}^{(3)} & h_{33}^{(3)} & h_{32}^{(2)} & K_{3,2} \\ h_{3(10)}^{(1)} & h_{49}^{(1)} & h_{48}^{(1)} & h_{47}^{(1)} & h_{46}^{(1)} & h_{45}^{(1)} & h_{44}^{(1)} & h_{43}^{(1)} & h_{42}^{(1)} & K_{4,1} \\ h_{4(10)}^{(2)} & h_{49}^{(2)} & h_{48}^{(2)} & h_{47}^{(2)} & h_{46}^{(2)} & h_{45}^{(2)} & h_{44}^{(2)} & h_{43}^{(2)} & h_{42}^{(2)} & K_{4,2} \\ h_{4(10)}^{(1)} & h_{49}^{(1)} & h_{48}^{(1)} & h_{47}^{(1)} & h_{46}^{(1)} & h_{45}^{(1)} & h_{44}^{(1)} & h_{43}^{(1)} & h_{42}^{(1)} & K_{4,3} \\ h_{4(10)}^{(4)} & h_{49}^{(4)} & h_{48}^{(4)} & h_{47}^{(4)} & h_{46}^{(4)} & h_{45}^{(4)} & h_{44}^{(4)} & h_{43}^{(4)} & h_{42}^{(4)} & K_{4,4} \\ h_{4(10)}^{(4)} & h_{49}^{(4)} & h_{48}^{(4)} & h_{47}^{(4)} & h_{46}^{(4)} & h_{45}^{(4)} & h_{44}^{(4)} & h_{43}^{(4)} & h_{42}^{(4)} & K_{4,4} \\ \end{pmatrix} \end{bmatrix}$$

 $\begin{bmatrix} 54.6133 & 336.782 & 902.637 & 1378.23 & 1319.82 & 821.475 & 332.231 & 84.1956 & 12.1362 & 0.75852 \\ 1.72000 & 13.4400 & 44.9000 & 84.4200 & 98.5875 & 74.2050 & 36.0050 & 10.8600 & 1.84750 & 0.13500 \\ 0.30000 & 2.35000 & 7.87500 & 14.8625 & 17.4375 & 13.2000 & 6.45000 & 1.96250 & 0.33750 & 0.02500 \\ -100.333 & -758.222 & -2433.04 & -4362.15 & -4820.22 & -3405.55 & -1539.11 & -429.556 & -67.2962 & -4.51852 \\ -38.0000 & -288.333 & -929.555 & -1675.22 & -1861.33 & -1322.33 & -600.667 & -168.333 & -26.4444 & -1.77778 \\ -12.0000 & -92.0000 & -300.333 & -549.333 & -621.000 & -450.000 & -209.000 & -60.0000 & -9.66667 & -0.66667 \\ 45.0000 & 408.000 & 1485.50 & 2899.50 & 3401.81 & 2509.87 & 1170.86 & 334.500 & 53.3125 & 3.62500 \\ -85.5000 & -612.750 & -1872.88 & -3222.06 & -3439.06 & -2358.75 & -1039.00 & -283.562 & -43.5625 & -2.87500 \\ 18.0000 & 147.000 & 501.500 & 938.250 & 1064.00 & 761.500 & 345.500 & 96.2500 & 15.0000 & 1.00000 \\ -18.0000 & -129.000 & -390.500 & -658.750 & -684.750 & -456.000 & -195.000 & -51.7500 & -7.75000 & -0.50000 \end{bmatrix}$

5.9. Podsumowanie

W niniejszym rozdziale na podstawie analitycznego twierdzenia służącego do odwracania konfluentnej macierzy Vandermonde'a skonstruowano algorytm wykonujący to zadanie na drodze numerycznej. Uzyskano wielomianową złożoność obliczeniową algorytmu równą $O(n^2)$. W kolejnym rozdziale zaprezentowano przykład zastosowania przedstawionego algorytmu w analizie własności liniowych układów dynamicznych o dowolnym stopniu pochodnych względem czasu.

6. ZASTOSOWANIE ALGORYTMÓW OBLICZANIA WIELOMIANÓW SYMETRYCZNYCH ORAZ ODWRACANIA MACIERZY SPECJALNYCH W ANALIZIE WŁASNOŚCI UKŁADÓW DYNAMICZNYCH DOWOLNEGO RZĘDU

6.1. Wprowadzenie do problematyki jakościowych własności układów dynamicznych

W praktyce inżynierskiej i naukowej nie zawsze konieczna jest znajomość dokładnego (lub nawet przybliżonego) rozwiązania danego równania różniczkowego. W wielu przypadkach większe znaczenie od znalezienia konkretnej funkcji spełniającej dane równanie jest znajomość pewnych ogólnych własności rozwiązania tego równania. Niniejszy rozdział zajmuje się jedną z jakościowych własności układów dynamicznych, tj. sterowalnością. Istnieje obszerna literatura dotycząca badania sterowalności, ale do dziś nie wszystkie związane z nią problemy zostały w pełni rozwiązane. Własność ta, obok stabilności, ma największe znaczenie w praktyce inżynierskiej i naukowej.

Można wyróżnić kilka głównych kierunków wcześniejszych i współczesnych badań nad problematyką sterowalności, dotyczących następujących rodzajów układów dynamicznych:

- ciągłych liniowych (Kalman [40], Bartosiewicz [5], Mitkowski [67]),
- dyskretnych liniowych (Klamka [46] s.89-126),
- stacjonarnych (Klamka [46] s.16-21, 95-97),
- niestacjonarnych (Klamka [46] s.5-15, 92-96),
- nieliniowych (Klamka [48]),
- stochastycznych (Mahmudov [58]),
- układów przemysłowych (Alotaibi [3], Vargra [132]),
- w formie kanonicznej Jordana (Chen [10], Zadeh [141] s.511),
- złożonych (Davison [14]),
- nieskończenie wymiarowych (Sakawa [111-113], Triggiani [130-131], Chen, Russel [11], Huang [32], Ito [34-35], Fattorini [21-22], Jacob [36], Miller [64]),
- ze stożkowo ograniczonymi sterowaniami (Brammer [7], Schmitendorf [117]),
- ze sterowaniami ograniczonymi do zbioru zwartego (Schmitendorf [116]),
- układów wyższych rzędów (Shi [118-119], Shubov [120-121], Xu [139]).

67

Klasyczną definicję sterowalności dla ciągłych układów liniowych o parametrach skupionych podają prace [41] s.192, [46] s. 2. Twierdzenie służące jej badaniu podają prace [41] s.201, [46] s. 16, [38] s.130, [132]. Szczególny przypadek tego twierdzenia dla układów o diagonalnej macierzy stanu podaje monografia [66] s. 195, natomiast dla układów w formie Jordana stosuje się twierdzenie Chena z prac [10] i [46] s. 25 . Dla układów z opóźnieniami w sterowaniach rozróżniamy pojęcia sterowalności absolutnej ([46] s.195) i względnej (relatywnej) ([46] s.195). Kryteria służące do badania tej pierwszej podają prace [46] s. 207, [45], natomiast do badania tej drugiej prace [46] s. 202, [45]. Wyróżniamy także tzw. U-sterowalność, nakładającą na sterowania ograniczenia w postaci stożkowego zbioru sterowań dopuszczalnych U. Jej definicję podaje monografia [46] s. 36, natomiast odpowiednie twierdzenie służące jej weryfikacji podają prace: [7, 117, 46] s. 52. Osobnym problemem jest sterowalność układów dynamicznych opisanych równaniami różniczkowymi cząstkowymi. Rozróżniamy tu sterowalność aproksymacyjną ([46] s.130).

6.2. Cel rozdziału

Głównym celem rozdziału 6 jest znalezienie warunków aproksymacyjnej sterowalności liniowych układów dynamicznych dowolnego rzędu z zastosowaniem algorytmów obliczania wielomianów symetrycznych podstawowych (z rozdziału 4) oraz odwracania konfluentnych macierzy Vandermonde'a (z rozdziału 5). Główną innowacją w stosunku do rezultatów znanych w literaturze jest przeprowadzenie analizy dla układu liniowego o wysokim stopniu ogólności:

- o dowolnym rzędzie względem czasu,
- o dowolnej krotności każdej z wartości własnych (pojedynczych, jak i wielokrotnych),
- z możliwością opóźnienia w sterowaniach.

Przyjęcie modelu matematycznego analizowanego układu dynamicznego w ogólnej postaci daje korzyści w postaci szerokiej klasy układów fizycznych potencjalnie opisywanych przez model. W literaturze można znaleźć wiele artykułów badających sterowalność dla konkretnego rzędu układu, zazwyczaj drugiego. Nieco mniej obszerna jest literatura dotycząca sterowalności układów czwartego rzędu, przykładem pozycji zajmujących się tą problematyką są [118, 119, 121]. Analiza dla dowolnego rzędu układu, przeprowadzona w niniejszym rozdziale, jest znacznie bardziej złożona niż dla układów rzędu drugiego i czwartego. Rozdział podsumowuje rezultaty autora niniejszej monografii z artykułów [85], [86] oraz [87].

6.3. Metodologia zastosowania algorytmów numerycznych w analizie własności układów dynamicznych dowolnego rzędu

Przed zastosowaniem algorytmów obliczania wielomianów symetrycznych podstawowych (rozdział 4) oraz odwracania konfluentnej macierzy Vandermonde'a (rozdział 5) jest konieczne wykonanie następujących operacji na drodze analitycznej:

- przekształcenie badanego układu do postaci ciągu macierzowych układów 1 rzędu; pomocnymi narzędziami tutaj są: dekompozycja spektralna oraz forma kanoniczna Frobeniusa (p. 6.6),
- przekształcenie układu do postaci kanonicznej Jordana oraz znalezienie macierzy podobieństwa (p. 6.7),
- przekształcenie układu do postaci bez opóźnień w sterowaniach (p. 6.9).

Algorytm odwracania konfluentnej macierzy Vandermonde'a, a tym samym także algorytm obliczania wielomianów symetrycznych podstawowych, znajduje zastosowanie w następujących problemach analizy układu dynamicznego:

- obliczenie odwrotności macierzy podobieństwa T formy kanonicznej Jordana macierzy stanu układu; znajomość odwrotności macierzy T pozwala na bezpośrednie zastosowanie twierdzenia Chena do badania obydwu rodzajów sterowalności, tj. bez ograniczeń i ze stożkowo ograniczonymi sterowaniami,
- obliczenie wektorów własnych macierzy transponowanej do macierzy stanu układu.

Metodologię analizy układów dynamicznych zastosowanej w niniejszym rozdziale podsumowano na diagramie rys. 4. Obrazuje on sposób połączenia metod analitycznych z zastosowaniem algorytmów numerycznych.



- Rys. 4. Idea metodologii zastosowania algorytmów numerycznych w analizie własności układów dynamicznych dowolnego rzędu
- Fig. 4. The paradigm of the numerical recipes application in the arbitrary order dynamical systems analysis

6.4. Model matematyczny badanego układu dynamicznego

Dany jest liniowy, stacjonarny układ dynamiczny n-tego rzędu z opóźnieniami w sterowaniach, opisany następującym równaniem różniczkowym:

$$\frac{d^{n}x(t)}{dt^{n}} + f_{n-1}(A)\frac{d^{n-1}x(t)}{dt^{n-1}} + \dots + f_{q}(A)\frac{d^{q}x(t)}{dt^{q}} + \dots + f_{1}(A)\frac{dx(t)}{dt} + f_{0}(A)x(t) =
= \sum_{k=0}^{M} B_{k}u(t-h_{k}), \quad t \ge t_{0}$$
(6.1)

gdzie $f_a(A)$ jest ciągiem następujących wyrazów tłumiących:

$$f_q(A) = \alpha_0^{(q)} + \alpha_1^{(q)}A + \alpha_2^{(q)}A^{\frac{1}{2}}, \quad q = 0, 1, ..., n-1, \quad \alpha_0^{(q)}, \, \alpha_1^{(q)}, \, \alpha_2^{(q)} \in R$$
(6.2)

przy warunkach początkowych:

$$x(0) = x_0 \in X, \quad x^{(q)}(0) = x_q \in X, \quad q = 1, 2, ..., n-1$$
(6.3)

gdzie $x(t) \in X$ (X jest przestrzenią Hilberta), stałe rzeczywiste współczynniki $\alpha_k^{(q)} \in R$. Operatory wejścia B_k są zdefiniowane jak następuje:

$$B_k u(t - h_k) = \sum_{l=1}^p b_k^{(l)} u_l(t - h_k), \quad B_k \in L(U, X)$$
(6.4)

gdzie: $b_k^{(l)} \in X$, $u_l(t) \in L^2([t_0, \infty), R)$, k = 0, 1, ..., M, l = 1, 2, ..., p, U to przestrzeń Hilberta sterowań, dim U = p. Stałe opóźnienia h_k spełniają nierówność (6.5):

$$0 = h_0 < h_1 < \dots < h_k < \dots < h_M \tag{6.5}$$

O operatorze $A: X \supset D(A) \rightarrow X$ zakładamy, że jest liniowym, nieograniczonym, samosprzężonym i dodatnio określonym operatorem o dziedzinie D(A). Fizyczna interpretacja równania (6.1) obejmuje szeroką klasę rzeczywistych układów z tłumieniem i zależy od konkretnej postaci operatora A i od współczynników oraz wykładników czynników tłumiących $f_q(A)$.

6.5. Własności spektralne badanego układu dynamicznego

Operator A posiada następujące własności spektralne ([21, 22, 32, 34, 35]):

Operator A ma czysto dyskretne widmo punktowe składające się całkowicie z odosobnionych, rzeczywistych i dodatnich wartości własnych λ_i o skończonej krotności m_i:

$$0 < \lambda_1 < \lambda_2 < \dots < \lambda_i < \lambda_{i+1} < \dots, \quad \lim_{i \to \infty} \lambda_i = \infty$$
(6.6)

Funkcje własne operatora A { φ_{ij}, i = 1,2,3,..., j = 1,2,...,m_i} tworzą zupełny i ortogonalny układ w przestrzenie Hilberta X. Stąd dla każdego x ∈ X ma miejsce rozwinięcie:

$$x = \sum_{i=1}^{\infty} \sum_{j=1}^{m_i} \langle x, \phi_{ij} \rangle_X \phi_{ij}$$
(6.7)

• Operator A ma następującą reprezentację spektralną:

$$\bigvee_{x \in D(A)} Ax = \sum_{i=1}^{\infty} \sum_{j=1}^{m_i} \lambda_i < x, \phi_{ij} >_X \phi_{ij}, \quad D(A) = \left\{ x \in X : \sum_{i=1}^{\infty} \sum_{j=1}^{m_i} \lambda_i^2 \left| < x, \phi_{ij} >_X \right|^2 < \infty \right\}$$
(6.8)

• Ułamkowa potęga operatora A jest zdefiniowana wzorami:

$$\bigvee_{x \in D(A^{\beta}), \beta \in (0,1)} A^{\beta} x = \sum_{i=1}^{\infty} \sum_{j=1}^{m_i} \lambda_i^{\beta} \langle x, \phi_{ij} \rangle_X \phi_{ij}$$
(6.9)

$$D(A^{\beta}) = \left\{ x \in X : \sum_{i=1}^{\infty} \sum_{j=1}^{m_i} \lambda_i^{2\beta} \left| \langle x, \phi_{ij} \rangle_X \right|^2 < \infty \right\}$$
(6.10)

 Operator A^β, 0 < β < 1 jest także samosprzężony i dodatnio określony o dziedzinie D(A^β) gęstej w przestrzeni X.

6.6. Transformacja układu dynamicznego

Celem niniejszego podpunktu jest przekształcenie układu nieskończenie wymiarowego (6.1) do równoważnej postaci nieskończonego ciągu skończenie wymiarowych liniowych układów pierwszego rzędu. W tym celu korzystamy z własności (6.7), (6.8) oraz (6.9) uzyskując:

$$\sum_{i=1}^{\infty} \sum_{j=1}^{m_{i}} \frac{d^{n} x_{ij}(t)}{dt^{n}} \phi_{ij} + \sum_{q=0}^{n-1} \left(\alpha_{0(q)} \sum_{i=1}^{\infty} \sum_{j=1}^{m_{i}} \frac{d^{q} x_{ij}(t)}{dt^{q}} \phi_{ij} + \alpha_{1(q)} \sum_{i=1}^{\infty} \sum_{j=1}^{m_{i}} \lambda_{i} \frac{d^{q} x_{ij}(t)}{dt^{q}} \phi_{ij} + \alpha_{2(q)} \sum_{i=1}^{\infty} \sum_{j=1}^{m_{i}} \lambda_{i}^{\frac{1}{2}} \frac{d^{q} x_{ij}(t)}{dt^{q}} \phi_{ij} \right) = \sum_{k=0}^{M} B_{k} u(t-h_{k}), \quad i = 1, 2, 3, ..., \quad j = 1, 2, ..., m_{i}, \quad t \ge t_{0}$$

$$(6.11)$$

gdzie:

$$x_{ij}(t) = \langle x(t), \phi_{ij} \rangle_X, \quad i = 1, 2, 3, ..., \quad j = 1, 2, ..., m_i$$
(6.12)

oznacza *ij*-ty współczynnik Fouriera spektralnej reprezentacji elementu X w przestrzeni stanu X, ϕ_{ij} jest *ij*-tą funkcją własną operatora stanu A, m_i jest krotnością wartości własnych tego operatora A. Kolejnym krokiem jest obliczenie iloczynu skalarnego obydwu stron ciągu równań (6.11) w przestrzeni Hilberta X:

$$\frac{d^{n}x_{ij}(t)}{dt^{n}} + \sum_{q=0}^{n-1} \left(\alpha_{0(q)} \frac{d^{q}x_{ij}(t)}{dt^{q}} + \alpha_{1(q)}\lambda_{i} \frac{d^{q}x_{ij}(t)}{dt^{q}} + \alpha_{2(q)}\lambda_{i}^{\frac{1}{2}} \frac{d^{q}x_{ij}(t)}{dt^{q}} \right) = \sum_{k=0}^{M} \sum_{l=1}^{p} \left\langle b_{k}^{(l)}, \phi_{ij} \right\rangle_{X} u_{l}(t-h_{k}), \quad t \ge t_{0}, \quad i = 1, 2, 3, ..., \quad j = 1, 2, ..., m_{i}$$
(6.13)

Ostatnim krokiem jest przekształcenie ciągu równań (6.13) do postaci Frobeniusa. W wyniku tego otrzymujemy następujące, macierzowe i skończenie wymiarowe równanie różniczkowe zwyczajne, pierwszego rzędu względem czasu:

$$\dot{\varsigma}_{i}(t) = A_{i}\varsigma_{i}(t) + \sum_{k=0}^{M} B_{ki}u(t-h_{k}), \quad i = 1, 2, 3, ..., \quad t \ge t_{0}$$
(6.14)

gdzie wektor stanu $\zeta_i(t)$ jest dany równością:

$$\varsigma_i(t) = \left[\left[\varsigma_{i1}'(t) \right]^T \cdots \left[\varsigma_{ij}'(t) \right]^T \cdots \left[\varsigma_{im_i}'(t) \right]^T \right]^T, \ i = 1, 2, 3, \dots$$
(6.15)

gdzie podwektory $\zeta'_{ij}(t)$ są dane przez:

$$\varsigma'_{ij}(t) = \begin{bmatrix} x_{ij}(t) & \cdots & \frac{d^q x_{ij}(t)}{dt^q} & \cdots & \frac{d^{n-1} x_{ij}(t)}{dt^{n-1}} \end{bmatrix}^T, \quad i = 1, 2, 3, \dots, \quad j = 1, 2, \dots, m_i$$
(6.16)

Macierz stanu A_i oraz macierz wejścia B_{ki} są odpowiednio macierzą blokowo-diagonalną oraz macierzą blokową o postaci:

$$A_{i} = \begin{bmatrix} A_{i}^{'} & & 0 \\ & A_{i}^{'} & \\ 0 & & A_{i}^{'} \end{bmatrix} \qquad B_{ki} = \begin{bmatrix} \begin{bmatrix} B_{ki1}^{'} \end{bmatrix}^{T} & \cdots & \begin{bmatrix} B_{kij}^{'} \end{bmatrix}^{T} & \cdots & \begin{bmatrix} B_{kim_{i}}^{'} \end{bmatrix}^{T} \end{bmatrix}^{T} \\ k = 0, 1, \dots, M, \quad i = 1, 2, 3, \dots$$
(6.17)

Występujące w równaniu (6.17) podmacierze A'_i są macierzami Frobeniusa:

$$A'_{i} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -f^{*}_{i0} & -f^{*}_{i1} & -f^{*}_{i2} & \cdots & -f^{*}_{i(n-2)} & -f^{*}_{i(n-1)} \end{bmatrix} \quad i = 1, 2, 3, \dots$$
(6.18)

Podmacierze B'_{kij} są dane przez:

$$B_{kij}^{'} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \\ \left\langle b_{k}^{(1)}, \phi_{ij} \right\rangle_{X} & \cdots & \left\langle b_{k}^{(l)}, \phi_{ij} \right\rangle_{X} & \cdots & \left\langle b_{k}^{(p)}, \phi_{ij} \right\rangle_{X} \end{bmatrix} \right\} n$$

$$i = 1, 2, 3, \dots \quad j = 1, 2, \dots, m_{i}, \quad k = 0, 1, \dots, M$$

$$(6.19)$$

Stałe współczynniki f_{iq}^* w podmacierzach A_i' (6.18) są zdefiniowane równościami (6.20), na podstawie (6.2):

$$f_{iq}^* = \alpha_0^{(q)} + \alpha_1^{(q)}\lambda_i + \lambda_i^{\frac{1}{2}}, \quad q = 0, 1, \dots, n-1, \quad i = 1, 2, 3, \dots$$
(6.20)

gdzie λ_i oznacza i-tą wartość własną operatora stanu A.

Jak wiadomo ([6, 39]), wyznacznik macierzy blokowo-diagonalnej jest iloczynem wyznaczników macierzy na diagonali. Dlatego równanie charakterystyczne macierzy stanu (6.17), na podstawie (6.18), (6.20), może być wyrażone przez potęgę $m_i \in Z_+$ funkcji $p_i(s_i)$:

$$p_i^{m_i}(s_i) = \left(s_i^n + f_{i(n-1)}^* s_i^{n-1} + \dots + f_{iq}^* s_i^q + \dots + f_{i1}^* s_i + f_{i0}^*\right)^{m_i} = 0, \quad i = 1, 2, 3, \dots$$
(6.21)

Układ dynamiczny (6.1) w dalszej części rozdziału będzie badany w równoważnej postaci ciągu układów (6.14).
6.7. Dekompozycja Jordana macierzy stanu

Niech $s_{i1}, s_{i2}, ..., s_{ir_i}$ będą parami różnymi zerami wielomianu charakterystycznego $p_i(s_i)$:

$$p_i\left(s_i\right) = \prod_{\gamma=1}^{r_i} \left(s_i - s_{i\gamma}\right)^{n_{i\gamma}}$$
(6.22)

gdzie $n_{i1} + ... + n_{ir_i} = n$. Niech $J_{in_{i\lambda}}(s_{i\gamma})$ będzie $(n_{i\gamma} \times n_{i\gamma})$ -wymiarową elementarną klatką Jordana:

$$J_{in_{i\gamma}}\left(s_{i\gamma}\right) = \begin{bmatrix} s_{i\gamma} & 1 & \cdots & 0 \\ s_{i\gamma} & 1 & \vdots \\ & \ddots & \ddots \\ & & s_{i\gamma} & 1 \\ 0 & & & s_{i\gamma} \end{bmatrix}_{n_{i\gamma} \times n_{i\gamma}}, \quad i = 1, 2, 3, \dots, \quad \gamma = 1, 2, \dots, r_{i}$$
(6.23)

Korzystając z klatek Jordana (6.23), można znaleźć postać kanoniczną Jordana macierzy stanu A_i . Ma ona postać macierzy blokowo-diagonalnej:

$$J(A_{i}) = diag[\underbrace{J_{in_{i1}}(s_{i1})\dots J_{in_{i1}}(s_{i1})}_{m_{i}-razy} \cdots \underbrace{J_{in_{i\gamma}}(s_{i\gamma})\dots J_{in_{i\gamma}}(s_{i\gamma})}_{m_{i}-razy} \cdots \underbrace{J_{in_{i\gamma}}(s_{i\gamma})\dots J_{in_{i\gamma}}(s_{i\gamma})}_{m_{i}-razy}]$$

$$i = 1, 2, 3, \dots$$
(6.24)

Jak wiadomo ([27]), macierz podobieństwa formy kanonicznej Jordana macierzy Frobeniusa (6.18) o wielokrotnych wartościach własnych jest konfluentną macierzą Vandermonde'a, której poświęcono rozdział 5. W przypadku blokowo-diagonalnej macierzy stanu (6.17), (6.18) macierz podobieństwa $T(A_i)$ ma nieco bardziej skomplikowaną postać macierzy blokowej:

$$T(A_i) = \begin{bmatrix} T_{i1}(s_{i1}) & \cdots & T_{i\gamma}(s_{i\gamma}) & \cdots & T_{i\gamma_i}(s_{i\gamma_i}) \end{bmatrix} \quad i = 1, 2, 3, \dots$$
(6.25)

gdzie podmacierze $T_{i\gamma}(s_{i\gamma})$ mają postać następującej blokowej macierzy antydiagonalnej:

$$T_{i\gamma}\left(s_{i\gamma}\right) = \underbrace{\begin{bmatrix}0 & \cdots & V_{i\gamma}\left(s_{i\gamma}\right)\\ \vdots & \ddots & \vdots\\ V_{i\gamma}\left(s_{i\gamma}\right) & \cdots & 0\end{bmatrix}}_{m_i - razy}, \quad i = 1, 2, 3, \dots, \quad \gamma = 1, 2, \dots, r_i$$
(6.26)

Każdej $n_{i\gamma}$ -tej krotności wartości własnej $s_{i\gamma}$ odpowiada $(n \times n_{i\gamma})$ -wymiarowa macierz $V_{i\gamma}(s_{i\gamma})$, której pierwsza kolumna ma postać taką jak w klasycznej macierzy Vandermonde'a, a pozostałe kolumny są kolejnymi pochodnymi pierwszej kolumny:

$$\begin{bmatrix} V_{i\gamma}(s_{i\gamma}) \end{bmatrix}_{qj} = \begin{cases} s_{i\gamma}^{q-1}, & dla \ j = 1\\ \frac{1}{(j-1)!} \frac{d^{j-1}}{ds_{i\gamma}^{j-1}} (s_{i\gamma}^{q-1}), & dla \ j = 2, 3, ..., n_{i\gamma} \end{cases}$$

$$i = 1, 2, 3, ..., \ \gamma = 1, 2, ..., r_i, \ q = 1, 2, ..., n$$
(6.27)

Wykonując różniczkowanie we wzorze (6.27), otrzymujemy bezpośrednio konfluentną macierz Vandermonde'a, o formie użytej w algorytmie przedstawionym w rozdziale 5:

$$\begin{bmatrix} V_{i\gamma}(s_{i\gamma}) \end{bmatrix}_{qj} = \begin{cases} \begin{pmatrix} q-1\\ j-1 \end{pmatrix} s_{i\gamma}^{q-j} & dla \ q \ge j \\ 0, & dla \ q < j \end{cases} \stackrel{i=1,2,3,..., \gamma = 1,2,...,r_i}{q=1,2,...,n, \ j=1,2,...,n_{i\gamma}}$$
(6.28)

6.8. Numeryczne wyznaczenie odwrotności macierzy podobieństwa formy kanonicznej Jordana macierzy stanu

W odwracaniu macierzy podobieństwa formy kanonicznej Jordana macierzy stanu znajduje zastosowanie algorytm obliczania odwrotności konfluentnej macierzy Vandermonde'a przedstawiony w rozdziale 5.

Postać odwrotnej konfluentnej macierzy Vandermonde'a, której algorytm obliczania podano w rozdziale 5, w kontekście badanego układu dynamicznego dowolnego stopnia, odpowiada przypadkowi, gdy podmacierze $T_{i\gamma}(s_{i\gamma})$, określone wzorem (6.26), są zbudowane tylko z jednego bloku $V_{i\gamma}(s_{i\gamma})$ zdefiniowanego w (6.28). Jest tak, gdy wszystkie wartości własne operatora stanu A (6.17) m_i są pojedyncze. W takim przypadku macierz stanu A_i (6.17) składa się tylko z jednego bloku Frobeniusa A_i' (6.18).

Dlatego teraz podamy i udowodnimy twierdzenie przedstawiające, w jaki sposób obliczyć odwrotną blokową konfluentną macierz Vandermonde'a w postaci (6.25), (6.26), (6.28) przy wykorzystaniu algorytmu numerycznego z rozdziału 5. Opowiada to najogólniejszemu przypadkowi, gdy krotności wartości własnych m_i operatora stanu A są dowolne.

6.8.1. Twierdzenie

Odwrotność macierzy blokowej $T(A_i)$, danej wzorami (6.25), (6.26) i (6.28), można wyrazić wzorem:

$$T^{-1}(A_{i}) = \left[\left[T^{o}_{i1}\left(s_{i1}\right) \right]^{T} \cdots \left[T^{o}_{i\gamma}\left(s_{i\gamma}\right) \right]^{T} \cdots \left[T^{o}_{i\gamma}\left(s_{i\gamma}\right) \right]^{T} \right]^{T} \quad i = 1, 2, 3, \dots$$
(6.29)

gdzie podmacierze $T_{i\gamma}^{o}(s_{i\gamma})$ mają postać następującej antydiagonalnej macierzy blokowej:

$$T_{i\gamma}^{o}\left(s_{i\gamma}\right) = \underbrace{\begin{bmatrix}0 & \cdots & V_{i\gamma}^{0}\left(s_{i\gamma}\right)\\ \vdots & \ddots & \vdots\\ V_{i\gamma}^{0}\left(s_{i\gamma}\right) & \cdots & 0\end{bmatrix}}_{m_{i} razy}, \quad i = 1, 2, 3, \dots, \quad \gamma = 1, 2, \dots, r_{i}$$
(6.30)

a bloki $V_{i\gamma}^0(s_{i\gamma})$ są odwrotnymi konfluentnymi macierzami Vandermonde'a, obliczanymi przez algorytmy 5.6.1, 5.6.2 zgodnie z formułą:

$$V_{i\gamma}^{0}\left(s_{i\gamma}\right) = \left[v_{i\gamma n}^{0}\left(s_{i\gamma}\right)...v_{i\gamma 1}^{0}\left(s_{i\gamma}\right)\right] = \left[h_{i}[\gamma, n], ..., h_{i}[\gamma, 1]\right], \quad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_{i} \quad (6.31)$$

gdzie $v_{i\gamma j}^0$ są $n_{i\gamma}$ -wymiarowymi pionowymi wektorami.

Dowód twierdzenia 6.8.1

Obliczmy iloczyn macierzy blokowych $T^{-1}(A_i)T(A_i)$, korzystając ze wzorów (6.25), (6.26) oraz (6.29), (6.30):

$$T^{-1}(A_{i})T(A_{i}) = \begin{bmatrix} 0 & \cdots & V_{i1}^{0}(s_{i1}) \\ \vdots & \ddots & \vdots \\ \frac{V_{i1}^{0}(s_{i1}) & \cdots & 0}{\vdots} & m_{i} \\ \frac{V_{i1}^{0}(s_{i1}) & \cdots & 0}{\vdots} & \vdots & \ddots & \vdots \\ \frac{V_{i1}^{0}(s_{i1}) & \cdots & 0}{\vdots} & m_{i} \end{bmatrix} \begin{bmatrix} 0 & \cdots & V_{i1}(s_{i1}) \\ \vdots & \ddots & \vdots & \vdots \\ \frac{V_{i1}(s_{i1}) & \cdots & 0}{m_{i} razy} \end{bmatrix} & \cdots \begin{bmatrix} 0 & \cdots & V_{ir_{i}}(s_{ir_{i}}) \\ \vdots & \ddots & \vdots \\ \frac{V_{ir_{i}}(s_{i1}) V_{i1}(s_{i1}) & 0 & \cdots & 0}{m_{i} razy} \end{bmatrix} = \begin{bmatrix} V_{i1}^{0}(s_{i1}) V_{i1}(s_{i1}) & 0 & \cdots & 0 \\ m_{i} razy & \cdots & 0 \end{bmatrix} m_{i} \end{bmatrix} \begin{bmatrix} 0 & \cdots & V_{i1}(s_{i1}) \\ \vdots & \ddots & \vdots \\ \frac{W_{ir_{i}}^{0}(s_{i1}) V_{i1}(s_{i1}) & 0 & \cdots & 0}{m_{i} - 1} & \cdots & 0 \\ m_{i} - 1 \begin{cases} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ \vdots & \cdots & 0 \\ 0 & \cdots & 0 \\ m_{i} - 1 \begin{cases} 0 & \cdots & 0 \\ \vdots & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ m_{i} - 1 \begin{cases} 0 & \cdots & 0 \\ 0$$

gdzie 0 oznacza macierz zerową o odpowiednich wymiarach. W celu obliczenia iloczynu macierzowego $V_{ik_1}^0(s_{ik_1})V_{ik_2}(s_{ik_2})$ w macierzy blokowej $T^{-1}(A_i)T(A_i)$ zauważmy, że w najprostszym przypadku, dla $m_i = 1$, odwrotność $T^{-1}(A_i)$ oraz macierz $T(A_i)$ mają postać odpowiednio:

$$T^{-1}(A_i)\Big|_{m_i=1} = \begin{bmatrix} V_{i1}^0(s_{i1}) & \cdots & V_{ir_i}^0(s_{ir_i}) \end{bmatrix}^T, \quad i = 1, 2, 3, \dots$$
(6.33)

$$T(A_{i})\Big|_{m_{i}=1} = \begin{bmatrix} V_{i1}(s_{i1}) & \cdots & V_{ir_{i}}(s_{ir_{i}}) \end{bmatrix}, \quad i = 1, 2, 3, \dots$$
(6.34)

Z założenia algorytm w rozdziale 5 oblicza macierz odwrotną, więc z definicji macierzy odwrotnej wynika, że:

$$T^{-1}(A_i)\Big|_{m_i=1} T(A_i)\Big|_{m_i=1} = I_{n \times n}$$
(6.35)

Uwzględniając ten fakt, iloczyny macierzowe $V_{ik_1}^0(s_{ik_1})V_{ik_2}(s_{ik_2})$ w macierzy blokowej (6.32) mogą być obliczone jak następuje:

$$V_{ik_{1}}^{0}\left(s_{ik_{1}}\right)V_{ik_{2}}\left(s_{ik_{2}}\right) = \begin{cases} I, \ k_{1} = k_{2} \\ 0, \ k_{1} \neq k_{2} \end{cases}, \quad \dim[I] = \dim[0] = n_{ik_{1}} \times n_{ik_{2}}, \\ k_{1}, k_{2} = 1, 2, \dots, r_{i}, \ i = 1, 2, 3, \dots \end{cases}$$
(6.36)

gdzie n_{ik_1} , n_{ik_2} oznaczają krotności wartości własnych operatora stanu (6.22). Z równania (6.36) bezpośrednio wynika, że macierz blokowa (6.32) jest macierzą jednostkową.

C.N.D

6.9. Analiza własności układu dowolnego rzędu

6.9.1. Twierdzenie o aproksymacyjnej absolutnej U-sterowalności

W niniejszym punkcie podano oraz udowodniono warunki konieczne i wystarczające aproksymacyjnej absolutnej U-sterowalności badanego układu (6.1). W tym celu wykorzystamy twierdzenia: Chena [10], o sterowalności układów przy stożkowych ograniczeniach na sterowania ([7, 117, 46 s. 52]) oraz twierdzenie z prac [46 s.207, 45] o sterowalności absolutnej. Najpierw przekształćmy nieskończenie wymiarowy układ dynamiczny (6.1), w postaci nieskończonego ciągu układów skończenie wymiarowych (6.14), do postaci wymaganej w twierdzeniu z prac [46 s. 207, 45]. W wyniku tego przekształcenia otrzymujemy układ:

$$\dot{\varsigma}_{i}(t) = A_{i}\varsigma_{i}(t) + \hat{B}_{i}u(t), \quad i = 1, 2, 3, ..., \quad t \ge t_{0}$$
(6.37)

$$\hat{B}_{i} = \sum_{k=0}^{M} e^{-A_{i}h_{k}} B_{ki}, \quad i = 1, 2, 3, \dots$$
(6.38)

gdzie macierze A_i, B_{ki} są dane przez (6.17).

• Twierdzenie [87]

Układ dynamiczny (6.1) jest globalnie aproksymacyjnie absolutnie U-sterowalny do zera w przedziale $[t_0, t_1], t_1 > t_0 + h_M$, przy sterowaniach ograniczonych do nieujemnego stożka o wierzchołku w zerze wtedy i tylko wtedy, gdy następujące warunki 1-5 są jednocześnie spełnione:

(1) Istnieje $w_i \in U$ takie, że $B_{ki}^* w_i = 0$ dla każdego i = 1, 2, 3, ..., k = 0, 1, ..., M.

- (2) Otoczka wypukła CH(U) ma niepuste wnętrze w przestrzeni \mathbb{R}^p .
- (3) Jest spełniony nieskończony ciąg równości:

$$rank \sum_{k=0}^{M} e^{-s_{i\gamma}h_k} B_{ki}^* = m_i, \quad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_i$$
(6.39)

gdzie B_{ki}^* jest dane przez (6.51), $s_{i\gamma}$ są wartościami własnymi (6.21) operatora stanu A_i (6.17), h_k są opóźnieniami oraz m_i krotnością wartości własnych operatora stanu (6.17). (4) Dla każdego $i = 1, 2, 3, ..., \gamma = 1, 2, ..., r_i$, którym odpowiada rzeczywista wartość własna $s_{i\gamma}$, musi być spełniona nierówność:

$$\begin{cases} \forall \ _{i=1,2,3,\dots} \ _{\gamma=1,2,\dots,r_{i}} \ _{d=1,2,\dots,n_{i\gamma}} \ _{l_{1},l_{2} \in \{1,2,\dots,p\}}^{\exists} \\ \operatorname{Im}[s_{i\gamma}]=0 \end{cases} \\ \left[\sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)} \left(s_{i\gamma} \right) \hat{B}_{il_{1}}'(k,l_{1}) \right] \left[\sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)} \left(s_{i\gamma} \right) \hat{B}_{il_{2}}'(k,l_{2}) \right] < 0 \end{cases}$$

$$(6.40)$$

gdzie elementy macierzy $\hat{B}'_{il}(\cdot,\cdot)$ można obliczyć ze wzoru (6.60) oraz $v^{0(d)}_{i\gamma(n-k+1)}(s_{i\gamma})$ oznacza d-ty element wektorów danych przez (6.31), $s_{i\gamma}$ są pierwiastkami równania charakterystycznego (6.22).

(5) Żaden z pierwiastków równania charakterystycznego (6.22) nie ma dodatniej części rzeczywistej:

$$\operatorname{Re}\left[s_{i\gamma}\right] \le 0, \quad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_{i}$$
(6.41)

Dowód twierdzenia

Każdy ze skończenie wymiarowych podukładów w nieskończonym ciągu (6.37) jest U-sterowalny wtedy i tylko wtedy, gdy warunki 1-5 twierdzenia o sterowalności układów przy stożkowych ograniczeniach ([7, 117, 46 s. 52]) są dla niego spełnione. Zastosujmy je kolejno.

– Warunki 1,2,5

Sprawdzając warunki 1, 2, 5 twierdzenia o sterowalności układów przy stożkowych ograniczeniach dla każdego ze skończenie wymiarowych podukładów (6.37), (6.38) rozważanego układu (6.1) bezpośrednio otrzymujemy warunki 1, 2, 5 dowodzonego twierdzenia.

– Warunek 3

Warunek 3 dowodzonego twierdzenia jest równoważny warunkowi sterowalności bez ograniczeń, który może zostać zweryfikowany za pomocą twierdzenia Chena. Zastosujmy je do układu (6.1) w postaci nieskończonego ciągu (6.37), (6.38). Najpierw obliczmy macierz \hat{B}_i (6.38) dla układu (6.14):

$$\hat{B}_{i} = \sum_{k=0}^{M} e^{-A_{i}h_{k}} B_{ki} = \sum_{k=0}^{M} T_{i} e^{-J(A_{i})h_{k}} T_{i}^{-1} B_{ki}, \quad i = 1, 2, 3, \dots$$
(6.42)

gdzie $J(A_i)$ jest dane przez wzór (6.24). Kluczową rolę w twierdzeniu Chena pełni czynnik $T_i^{-1}\hat{B}_i$. Przekształcając równanie (6.42), otrzymujemy:

$$T_i^{-1}\hat{B}_i = \sum_{k=0}^M e^{-J(A_i)h_k} T_i^{-1} B_{ki}, \quad i = 1, 2, 3, \dots$$
(6.43)

Obliczmy $e^{-J(A_i)h_k}$ w równaniu (6.43). Ze wzorów (5.8), (6.24) wynikają równości:

$$e^{-J(A_{i})h_{k}} = diag[\underbrace{e^{-J_{in_{1}}(s_{i_{1}})h_{k}} \dots e^{-J_{in_{l}}(s_{l_{1}})h_{k}}}_{m_{i}-razy} \dots \underbrace{e^{-J_{in_{i}}(s_{i_{j}})h_{k}} \dots e^{-J_{in_{i}}(s_{i_{j}})h_{k}}}_{m_{i}-razy} \dots \underbrace{e^{-J_{in_{i}}(s_{i_{j}})h_{k}} \dots e^{-J_{in_{i}}(s_{i_{j}})h_{k}}}_{m_{i}-razy}],$$

$$i = 1, 2, 3, \dots, \quad k = 0, 1, \dots, M$$
(6.44)

$$e^{-J_{iny}(s_{iy})h_{k}} = \begin{bmatrix} e^{-s_{iy}h_{k}} \frac{-h_{k}e^{-s_{iy}h_{k}}}{1!} & \cdots & \frac{(-1)^{(n_{iy}-1)}h_{k}^{n_{iy}-1}e^{-s_{iy}h_{k}}}{(n_{iy}-1)!} \\ e^{-s_{iy}h_{k}} \frac{-h_{k}e^{-s_{iy}h_{k}}}{1!} & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ & & e^{-s_{iy}h_{k}} \frac{-h_{k}e^{-s_{iy}h_{k}}}{1!} \\ 0 & & & e^{-s_{iy}h_{k}} \end{bmatrix}_{n_{ii}\times n_{ii}}^{n_{iy}-1} e^{-s_{iy}h_{k}} d^{-s_{iy}h_{k}} d^{-s_{iy}h_{k}}$$

Łącząc wzory (6.45), (6.29) możemy obliczyć:

$$e^{-J(A_{i})h_{k}}T^{-1}(A_{i}) = \\ = \left[e^{-J_{in_{i}}(s_{i})h_{k}}\left[T_{i1}^{o}\left(s_{i}\right)\right]^{T} \cdots e^{-J_{in_{i}}(s_{i\gamma})h_{k}}\left[T_{i\gamma}^{o}\left(s_{i\gamma}\right)\right]^{T} \cdots e^{-J_{in_{in}}(s_{in_{i}})h_{k}}\left[T_{i\gamma}^{o}\left(s_{in_{i}}\right)\right]^{T}\right]^{T}, \\ i = 1, 2, 3, ..., \quad k = 0, 1, ..., M$$

$$(6.46)$$

Teraz wykonajmy mnożenia macierzowe we wzorze (6.46). W wyniku tej operacji otrzymujemy, korzystając z zależności (6.30):

$$e^{-J(A_{i})h_{k}}T^{-1}(A_{i}) = \left[\left[D_{ik1}(s_{i1}) \right]^{T} \cdots \left[D_{ik\gamma}(s_{i\gamma}) \right]^{T} \cdots \left[D_{ik\gamma}(s_{i\gamma}) \right]^{T} \right]^{T}$$

$$i = 1, 2, 3, ..., \quad k = 0, 1, ..., M$$
(6.47)

gdzie $D_{ik\gamma}(s_{i\gamma})$ jest antydiagonalną macierzą blokową:

$$D_{ik\gamma}(s_{i\gamma}) = \begin{bmatrix} 0 & \cdots & Q_{ik\gamma}(s_{i\gamma}) \\ \vdots & \ddots & \vdots \\ Q_{ik\gamma}(s_{i\gamma}) & \cdots & 0 \\ \end{bmatrix}, \quad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_i, \quad k = 0, 1, ..., M (6.48)$$

oraz $Q_{ik\gamma}(s_{i\gamma}) = \{q_{ik\gamma l_1 l_2}(s_{i\gamma})\}$ we wzorze (6.48) jest następującą $(n_{i\gamma} \times n)$ -wymiarową macierzą:

$$e^{-s_{i\gamma}h_{k}} \begin{bmatrix} \sum_{d=1}^{n_{i\gamma}} \frac{(-1)^{(d-1)} h_{k}^{d-1}}{(d-1)!} v_{i\gamma n}^{0(d)}(s_{i\gamma}) \dots \sum_{d=1}^{n_{i\gamma}} \frac{(-1)^{(d-1)} h_{k}^{d-1}}{(d-1)!} v_{i\gamma l_{2}}^{0(d)}(s_{i\gamma}) \dots \sum_{d=1}^{n_{i\gamma}} \frac{(-1)^{(d-1)} h_{k}^{d-1}}{(d-1)!} v_{i\gamma l_{1}}^{0(d)}(s_{i\gamma}) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \sum_{d=l_{1}}^{n_{i\gamma}} \frac{(-1)^{(d-l_{1})} h_{k}^{d-l_{1}}}{(d-l_{1})!} v_{i\gamma n}^{0(d)}(s_{i\gamma}) \dots \sum_{d=l_{1}}^{n_{i\gamma}} \frac{(-1)^{(d-l_{1})} h_{k}^{d-l_{1}}}{(d-l_{1})!} v_{i\gamma l_{2}}^{0(d)}(s_{i\gamma}) \dots \sum_{d=l_{1}}^{n_{i\gamma}} \frac{(-1)^{(d-l_{1})} h_{k}^{d-l_{1}}}{(d-l_{1})!} v_{i\gamma l_{1}}^{0(d)}(s_{i\gamma}) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_{i\gamma n}^{0(n_{i\gamma})}(s_{i\gamma}) \dots & v_{i\gamma l_{2}}^{0(n_{i\gamma})}(s_{i\gamma}) \dots & v_{i\gamma l_{2}}^{0(n_{i\gamma})}(s_{i\gamma}) \end{pmatrix} \dots & v_{i\gamma l_{2}}^{0(n_{i\gamma})}(s_{i\gamma}) \end{bmatrix} = Q_{ik\gamma}(s_{i\gamma}) = \left\{ q_{ik\gamma l_{l_{2}}}(s_{i\gamma}) \right\}_{\substack{l_{1}=1,2,\dots,n_{n}\\ l_{2}=n,n-1,\dots,1}}, \quad i=1,2,3,\dots, \quad \gamma=1,2,\dots,r_{i}, \quad k=0,1,\dots,M \end{cases}$$

$$(6.49)$$

We wzorze (6.49) wielkości $v_{i\gamma l_2}^{(d)}(s_{i\gamma})$ są dane przez (6.31). Teraz czynnik $T_i^{-1}\hat{B}_i$ określony w (6.43) przy wykorzystaniu zależności (6.47) może być zapisany w postaci:

$$T_{i}^{-1}\hat{B}_{i} = \sum_{k=0}^{M} \left[\left[D_{ik1}(s_{i1}) \right]^{T} \cdots \left[D_{ik\gamma}(s_{i\gamma}) \right]^{T} \cdots \left[D_{ik\gamma}(s_{i\gamma}) \right]^{T} \right]^{T} B_{ki}, \quad i = 1, 2, 3, \dots (6.50)$$

Wprowadźmy pomocniczą macierz B_{ki}^* oraz wektory $B_{ki}^{*(l_1)}$, $l_1 = 1, 2, ..., m_i$:

$$B_{ki}^{*} = \begin{bmatrix} \frac{B_{ki}^{*(1)}}{\vdots} \\ \frac{B_{ki}^{*(l)}}{\vdots} \\ \frac{B_{ki}^{*(l)}}{\vdots} \\ \frac{B_{ki}^{*(m_{i})}}{\vdots} \end{bmatrix} = \begin{bmatrix} \langle b_{k}^{(1)}, \phi_{i1} \rangle_{X} & \dots & \langle b_{k}^{(l_{2})}, \phi_{i1} \rangle_{X} & \dots & \langle b_{k}^{(p)}, \phi_{i1} \rangle_{X} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \langle b_{k}^{(1)}, \phi_{il_{1}} \rangle_{X} & \dots & \langle b_{k}^{(l_{2})}, \phi_{il_{1}} \rangle_{X} & \dots & \langle b_{k}^{(p)}, \phi_{il_{1}} \rangle_{X} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \langle b_{k}^{(1)}, \phi_{im_{i}} \rangle_{X} & \dots & \langle b_{k}^{(l_{2})}, \phi_{im_{i}} \rangle_{X} & \dots & \langle b_{k}^{(p)}, \phi_{im_{i}} \rangle_{X} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \langle b_{k}^{(1)}, \phi_{im_{i}} \rangle_{X} & \dots & \langle b_{k}^{(l_{2})}, \phi_{im_{i}} \rangle_{X} & \dots & \langle b_{k}^{(p)}, \phi_{im_{i}} \rangle_{X} \end{bmatrix}$$

$$(6.51)$$

$$k = 0, 1, \dots, M, \ i = 1, 2, 3, \dots$$

Uwzględniając fakt, że jedynie co n-ty wiersz w ciągu macierzy B_{ki} (6.17) jest niezerowy, ze wzorów (6.50), (6.49) możemy obliczyć:

$$T_i^{-1}\hat{B}_i = \sum_{k=0}^{M} \left[\hat{G}_{ik1}^T \quad \cdots \quad \hat{G}_{ik\gamma}^T \quad \cdots \quad \hat{G}_{ik\gamma}^T \right]^T, \ i = 1, 2, 3, \dots$$
(6.52)

gdzie:

$$\hat{G}_{ik\gamma} = \begin{bmatrix} Q_{ik\gamma}^{(1)T} \left(s_{i\gamma} \right) \otimes B_{ki}^{*(m_i)T} & \cdots & Q_{ik\gamma}^{(1)T} \left(s_{i\gamma} \right) \otimes B_{ki}^{*(l_i)T} & \cdots & Q_{ik\gamma}^{(1)T} \left(s_{i\gamma} \right) \otimes B_{ki}^{*(1)T} \end{bmatrix}^T$$

$$i = 1, 2, 3, \dots, \ \gamma = 1, 2, \dots, r_i, \ k = 1, 2, \dots, M$$
(6.53)

oraz $B_{ki}^{*(l_1)}$ są wektorami danymi przez (6.51), $Q_{ik\gamma}^{(1)}(s_{i\gamma})$ jest ostatnią kolumną macierzy $Q_{ik\gamma}(s_{i\gamma})$ określonej w (6.49), odpowiadającą wektorowi $v_{i\gamma 1}^0(s_{i\gamma})$ zdefiniowanemu w (6.31):

$$Q_{ik\gamma}^{(1)}\left(s_{i\gamma}\right) = e^{-s_{i\gamma}h_{k}} \begin{bmatrix} \sum_{d=1}^{n_{i\gamma}} \frac{\left(-1\right)^{(d-1)} h_{k}^{d-1}}{(d-1)!} v_{i\gamma1}^{0(d)}\left(s_{i\gamma}\right) \\ \vdots \\ \sum_{d=l_{1}}^{n_{i\gamma}} \frac{\left(-1\right)^{(d-l_{1})} h_{k}^{d-l_{1}}}{(d-l_{1})!} v_{i\gamma1}^{0(d)}\left(s_{i\gamma}\right) \\ \vdots \\ v_{i\gamma1}^{0(n_{i\gamma})}\left(s_{i\gamma}\right) \end{bmatrix} \qquad \begin{array}{l} i = 1, 2, 3, \dots \\ \gamma = 1, 2, \dots, r_{i} \\ k = 0, 1, \dots, M \end{array}$$
(6.54)

Znak \otimes oznacza iloczyn Kronekera macierzy ([39] s.255). Wróćmy do weryfikacji sterowalności nieskończenie wymiarowego układu dynamicznego (6.1) w postaci (6.37) i (6.38). Twierdzenie Chena mówi o tym, że liniowo niezależne muszą być jedynie te wiersze macierzy $G_i = T_i^{-1}(A_i)\hat{B}_i$, które odpowiadają ostatnim wierszom elementarnej klatki Jordana $J_{in_{\gamma}}(s_{i\gamma})$ (5.8), dla każdej wartości własnej $s_{i\gamma}$, $i = 1, 2, 3, ..., \gamma = 1, 2, ..., r_i$. Zastosujmy twierdzenie Chena do rozważanego układu dynamicznego w postaci bez opóźnień w sterowaniach, o macierzy $G_i = T_i^{-1}(A_i)\hat{B}_i$ danej wzorami (6.52), (6.53), (6.54). Otrzymujemy, że układ (6.1) jest aproksymacyjnie sterowalny wtedy i tylko wtedy, gdy jest spełniony następujący ciąg warunków:

$$rank\left[\sum_{k=0}^{M} v_{i\gamma 1}^{0(n_{i\gamma})}(s_{i\gamma}) \cdot e^{-s_{i\gamma}h_{k}}(B_{ki}^{*})'\right] = m_{i}, \qquad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_{i}$$
(6.55)

gdzie macierz $(B_{ki}^*)'$ można uzyskać z macierzy B_{ki}^* (6.51) poprzez odwrócenie kolejności wierszy. Po zastosowaniu przekształceń algebraicznych z równości (6.55) uzyskujemy dowodzoną równość (6.39).

Warunek 4

Jak już wspomniano, wektory własne macierzy stanu A_i danej wzorami (6.17) i (6.18) mają postać blokowej macierzy Vandermonde'a (6.25)-(6.28). Z algebry liniowej wiadomo, że wektory własne macierzy transponowanej można obliczyć z transponowanej inwersji wektorów własnych danej macierzy. Pokazuje to formuła:

$$T(A_i^T) = \left[T^{-1}(A_i)\right]^T, \quad i = 1, 2, 3, ...$$
 (6.56)

Na podstawie wzoru (6.56) oraz wzorów (6.29), (6.30) i (6.31) rzeczywiste wektory własne macierzy A_i^T mają postać:

$$\psi_{i}^{(qd\gamma)}\left(s_{i\gamma}\right) = \left[\underbrace{0\cdots0}_{(q\times n)-razy} v_{i\gamma n}^{0(d)}\left(s_{i\gamma}\right)\cdots v_{i\gamma k}^{0(d)}\left(s_{i\gamma}\right)\cdots v_{i\gamma 1}^{0(d)}\left(s_{i\gamma}\right)\underbrace{0\cdots0}_{(n\times m_{i}-(n+1)q)-razy}\right]^{T}$$

$$i = 1, 2, 3, ..., \ \gamma = 1, 2, ..., r_{i}, \ d = 1, 2, ..., n_{i\gamma}, \ q = 0, 1, ..., m_{i} - 1, \ \operatorname{Im}\left[s_{i\gamma}\right] = 0$$
(6.57)

gdzie $v_{i\gamma k}^{0(d)}(s_{i\gamma})$ jest d-tym elementem wektora $v_{i\gamma k}^{0}(s_{i\gamma})$. Kolejną wielkością potrzebną w warunku 4 twierdzenia o sterowalności układów przy stożkowych ograniczeniach jest macierz \hat{B}_{i} , obliczona ze wzoru (6.38). Korzystając z macierzy podobieństwa Jordana $T(A_{i})$ (6.25), (6.26), (6.28) oraz już policzonego wyrazu $T_{i}^{-1}\hat{B}_{i}$ danego przez (6.52), (6.53), (6.54) możemy obliczyć macierz \hat{B}_{i} :

$$\hat{B}_{i} = \left[\begin{bmatrix} \hat{B}_{i1} \end{bmatrix}^{T} \cdots \begin{bmatrix} \hat{B}_{ij} \end{bmatrix}^{T} \cdots \begin{bmatrix} \hat{B}_{im_{i}} \end{bmatrix}^{T} \end{bmatrix}^{T}, \ i = 1, 2, 3, \dots$$

$$\hat{B}_{ij} = \sum_{k=0}^{M} \left[\sum_{\substack{\gamma=1 \\ \gamma=1}}^{r_{i}} \sum_{\substack{\tau=1 \\ \tau=1}}^{n_{i\gamma}} V_{i\gamma} \left(s_{i\gamma} \right)_{1\tau} \cdot q_{ik\gamma l_{1}l_{2}} \left(s_{i\gamma} \right)_{l_{1}=\tau} \\ \vdots \\ \sum_{\substack{\gamma=1 \\ \gamma=1}}^{r_{i}} \sum_{\substack{\tau=1 \\ \tau=1}}^{n_{i\gamma}} V_{i\gamma} \left(s_{i\gamma} \right)_{r\tau} \cdot q_{ik\gamma l_{1}l_{2}} \left(s_{i\gamma} \right)_{l_{1}=\tau} \\ \vdots \\ \sum_{\substack{\gamma=1 \\ \gamma=1}}^{r_{i}} \sum_{\substack{\tau=1 \\ \tau=1}}^{n_{i\gamma}} V_{i\gamma} \left(s_{i\gamma} \right)_{n\tau} \cdot q_{ik\gamma l_{1}l_{2}} \left(s_{i\gamma} \right)_{l_{1}=\tau} \\ \vdots \\ \sum_{\substack{\gamma=1 \\ \gamma=1}}^{r_{i}} \sum_{\substack{\tau=1 \\ \tau=1}}^{n_{i\gamma}} V_{i\gamma} \left(s_{i\gamma} \right)_{n\tau} \cdot q_{ik\gamma l_{1}l_{2}} \left(s_{i\gamma} \right)_{l_{1}=\tau} \\ \vdots \\ \end{bmatrix}$$

$$(6.59)$$

gdzie $q_{ik\gamma l_l l_2}(s_{i\gamma})$ są elementami ostatniej kolumny macierzy $Q_{ik\gamma}(s_{i\gamma})$ (6.49), $V_{i\gamma}(s_{i\gamma})_{r\tau}$ są blokami Vandermonde'a (6.28), *n* jest stopniem równania (6.1) oraz $B_{ki}^{*(j)}$ są dane przez (6.51). Korzystając ze wzorów (6.28) oraz (6.54), z zależności (6.59) otrzymujemy formułę wyrażającą razem z formułą (6.58) postać macierzy \hat{B}_i :

$$\hat{B}_{ij}^{'} = \sum_{k=0}^{M} \left[\sum_{\substack{\gamma=1\\\gamma=1}}^{r_{i}} \sum_{\tau=1}^{r} \left(\frac{r-1}{\tau-1} \right) s_{i\gamma}^{r-\tau} \cdot e^{-s_{i\gamma}h_{k}} \sum_{\substack{d=1\\\gamma=1}}^{n_{i\gamma}} \frac{\left(-1\right)^{(d-1)} h_{k}^{d-1}}{(d-\tau)!} v_{i\gamma1}^{0(d)} \left(s_{i\gamma}\right) \right] \otimes B_{ki}^{*(j)}, \quad \substack{i=1,2,3,\ldots\\j=1,2,3,\ldots\\j=1,2,\ldots,m_{i}} \quad (6.60)$$

$$\sum_{\substack{\gamma=1\\\gamma=1}}^{r_{i}} \sum_{\tau=1}^{n_{i\gamma}} \left(\frac{n-1}{\tau-1} \right) s_{i\gamma}^{n-\tau} \cdot e^{-s_{i\gamma}h_{k}} \sum_{\substack{d=\tau\\\gamma=1\\\gamma=1}}^{n_{i\gamma}} \frac{\left(-1\right)^{(d-\tau)} h_{k}^{d-\tau}}{(d-\tau)!} v_{i\gamma1}^{0(d)} \left(s_{i\gamma}\right) \right]$$

gdzie r = 1, 2, ..., n oznacza kolejny numer wiersza w macierzy \hat{B}_{ij} . Obliczmy wyraz $\hat{B}_i w_i$. Korzystając z zależności (6.17), (6.19), (6.58), (6.60) otrzymujemy:

$$\hat{B}_{i}w_{i} = \left[\underbrace{\sum_{l=1}^{p}\hat{B}_{i1}^{'}(1,l)u_{l}(t)}_{n}\cdots\sum_{l=1}^{p}\hat{B}_{i1}^{'}(n,l)u_{l}(t)}_{n}\cdots\sum_{l=1}^{p}\hat{B}_{im_{i}}^{'}(1,l)u_{l}(t)\cdots\sum_{l=1}^{p}\hat{B}_{im_{i}}^{'}(n,l)u_{l}(t)\right]^{T} (6.61)$$

gdzie wielkości $\hat{B}'_{ij}(v,l), v = 1,...,n, l = 1,..., p$ są odpowiednimi elementami macierzy \hat{B}'_{ij} (6.60). Teraz obliczmy czynnik $\psi_i^{(qd\gamma)T}(s_{i\gamma})\hat{B}_iw_i$ występujący w warunku 4 twierdzenia

o sterowalności układów przy stożkowych ograniczeniach. Łącząc wzór (6.61) ze wzorem (6.57), otrzymujemy:

$$\psi_{i}^{(qd\gamma)T}\left(s_{i\gamma}\right)\hat{B}_{i}w_{i} = \sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)}\left(s_{i\gamma}\right)\sum_{l=1}^{p}\hat{B}_{i(q+1)}^{'}(k,l)u_{l}(t) = \sum_{l=1}^{p} u_{l}(t)\sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)}\left(s_{i\gamma}\right)\hat{B}_{i(q+1)}^{'}(k,l)$$

$$i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_{i}, \quad d = 1, 2, ..., n_{i\gamma}, \quad q = 0, 1, ..., m_{i} - 1, \quad \mathrm{Im}\left[s_{i\gamma}\right] = 0$$
(6.62)

Uwzględniając, że sterowania są z założenia ograniczone do nieujemnego stożka oraz biorąc pod uwagę konkretną postać (6.62) czynnika $\psi_i^{(qd\gamma)T}(s_{i\gamma})\hat{B}_iw_i$ dla rozważanego układu dynamicznego (6.14), warunek 4 twierdzenia o sterowalności układów przy stożkowych ograniczeniach redukuje się do wymagania, aby wyrażenie $\psi_i^{(qd\gamma)T}(s_{i\gamma})\hat{B}_iw_i$ w dopuszczalnej przestrzeni sterowań przyjmowało wartości obu znaków. Jedynie w tym przypadku nie istnieje wektor własny $\psi_i^{(qd\gamma)T}(s_{i\gamma})$ macierzy transponowanej A_i^T spełniający warunek:

Z równania (6.62) można wywnioskować, że wyrażenie $\psi_i^{(qd\gamma)T}(s_{i\gamma})\hat{B}_iw_i$ przyjmuje wartości obu znaków przy nieujemnym zbiorze sterowań dopuszczalnych wtedy i tylko wtedy, gdy spełniona jest nierówność:

$$\bigvee_{i=1,2,3,\dots} \bigvee_{\substack{\gamma=1,2,\dots,r_{i} \\ \text{Im}[s_{i\gamma}]=0}} \forall \exists_{l_{1},l_{2} \in \{1,2,\dots,p\}} \left[\sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)} \left(s_{i\gamma} \right) \hat{B}_{il_{1}}^{'} \left(k,l_{1} \right) \right] \left[\sum_{k=1}^{n} v_{i\gamma(n-k+1)}^{0(d)} \left(s_{i\gamma} \right) \hat{B}_{il_{2}}^{'} \left(k,l_{2} \right) \right] < 0$$

$$(6.64)$$

$$C.N.D.$$

6.9.2. Twierdzenie o aproksymacyjnej relatywnej U-sterowalności

W niniejszym punkcie udowodniono warunki konieczne i wystarczające aproksymacyjnej relatywnej U-sterowalności badanego układu. W tym celu wykorzystano twierdzenie Chena oraz twierdzenia o sterowalności układów przy stożkowych ograniczeniach na sterowania ([7, 117, 46 s. 52]) oraz twierdzenie z prac [46 s. 202, 45] o sterowalności relatywnej. Przekształćmy nieskończenie wymiarowy układ dynamiczny (6.1), w postaci nieskończonego ciągu układów skończenie wymiarowych (6.14), do postaci wymaganej przez twierdzenie o sterowalności relatywnej. W wyniku tego przekształcenia otrzymujemy układ:

$$\dot{\varsigma}_i(t) = A_i \varsigma_i(t) + \tilde{B}_i u(t), \quad i = 1, 2, 3, ..., \quad t \ge t_0$$
(6.65)

$$\tilde{B}_{i} = \left[B_{0i} \mid B_{1i} \mid \dots \mid B_{(k_{0}-1)i} \right], \quad i = 1, 2, 3, \dots$$
(6.66)

gdzie macierze A_i, B_{ki} są dane wzorem (6.17).

• Twierdzenie [87]

Układ dynamiczny (6.1) jest globalnie aproksymacyjnie relatywnie U-sterowalny do zera w przedziale $t_1 > t_0$, przy sterowaniach ograniczonych do nieujemnego stożka o wierzchołku w zerze wtedy i tylko wtedy, gdy następujące warunki 1-5 są jednocześnie spełnione:

- (1) Istnieje $w_i \in U$ takie, że $B_{ki}^* w_i = 0$ dla każdego i = 1, 2, 3, ..., k = 0, 1, ..., M.
- (2) Otoczka wypukła CH(U) ma niepuste wnętrze w przestrzeni R^p .
- (3) Jest spełniony ciąg równości:

$$rank \left[B_{0i}^{*} \mid B_{1i}^{*} \mid \dots \mid B_{(k_{0}-1)i}^{*} \right] = m_{i}, \quad i = 1, 2, 3, \dots,$$
(6.67)

gdzie B_{ki}^* jest dane przez (6.51).

(4) Dla każdego $i = 1, 2, 3, ..., \gamma = 1, 2, ..., r_i$, któremu odpowiada rzeczywista wartość własna wielomianu charakterystycznego (6.21), w każdym n-tym wierszu macierzy \tilde{B}_i (6.66) muszą się znajdować dwa iloczyny skalarne przeciwnych znaków:

$$\begin{array}{l} \exists \\ k_1, k_2 \in \{1, 2, \dots, p\} \\ l_1, l_2 \in \{0, 1, \dots, k_0 - 1\} \\ (k_1, l_1) \neq (k_2, l_2) \end{array} \langle b_{k_1}^{l_1}, \phi_{iq} \rangle_X \langle b_{k_2}^{l_2}, \phi_{iq} \rangle_X < 0, \quad i = 1, 2, 3, \dots, \quad q = 1, 2, \dots, m_i$$

$$(6.68)$$

(5) Żaden z pierwiastków równania charakterystycznego (6.22) nie ma dodatniej części rzeczywistej:

$$\operatorname{Re} \left| s_{i\gamma} \right| \le 0, \quad i = 1, 2, 3, ..., \quad \gamma = 1, 2, ..., r_i \tag{6.69}$$

Dowód twierdzenia

Układ nieskończenie wymiarowy (6.1) jest aproksymacyjnie relatywnie U-sterowalny wtedy i tylko wtedy, gdy warunki 1-5 twierdzenia o sterowalności przy stożkowo ograniczonych sterowaniach są spełnione dla każdego z podukładów w nieskończonym ciągu (6.65).

– Warunki 1,2,5

Dowód tych warunków przebiega analogicznie jak dla twierdzenia 6.9.1.

Warunek 3

Dowód tego warunku jest oparty na twierdzeniu Chena. Czynnik $T_i^{-1}\tilde{B}_i$ ma postać:

$$T_{i}^{-1}\tilde{B}_{i} = \begin{bmatrix} \tilde{G}_{i1}^{T} & \cdots & \tilde{G}_{i\gamma}^{T} & \cdots & \tilde{G}_{ir_{i}}^{T} \end{bmatrix}^{T}, \ i = 1, 2, 3, \dots$$
(6.70)

$$\hat{G}_{i\gamma} = \begin{bmatrix} Q_{ik\gamma}^{(1)T}\left(s_{i\gamma}\right) \otimes \tilde{B}_{i}^{*(m_{i})T} & \cdots & Q_{ik\gamma}^{(1)T}\left(s_{i\gamma}\right) \otimes \tilde{B}_{i}^{*(l_{i})T} & \cdots & Q_{ik\gamma}^{(1)T}\left(s_{i\gamma}\right) \otimes \tilde{B}_{i}^{*(1)T} \end{bmatrix}^{T}$$

$$i = 1, 2, 3, \dots, \ \gamma = 1, 2, \dots, r_{i}$$

$$(6.71)$$

$$\tilde{B}_{i}^{*(l_{1})} = \left[B_{0i}^{*(l_{1})} \mid B_{1i}^{*(l_{1})} \mid \dots \mid B_{(k_{0}-1)i}^{*(l_{1})} \right], \quad i = 1, 2, 3, \dots, \ l_{1} = 1, 2, \dots, m_{i}$$
(6.72)

gdzie wielkości $B_{ki}^{*(l_1)}$ są wektorami określonymi przez (6.51), $Q_{ik\gamma}^{(1)}(s_{i\gamma})$ jest dana wzorem (6.49). Stosując twierdzenie Chena do czynnika $T_i^{-1}\tilde{B}_i$ określonego przez (6.70) bezpośrednio otrzymujemy ciąg równości (6.67).

- Warunek 4

Wpierw obliczmy czynnik $\tilde{B}_i w_i$. Korzystając ze wzorów (6.17), (6.19), (6.66) otrzymujemy:

$$\tilde{B}_{i}w_{i} = \left[\underbrace{0 \cdots 0}_{n-1} \sum_{k=0}^{k_{0}-1} \sum_{l=1}^{p} \left\langle b_{k}^{(l)}, \phi_{l} \right\rangle_{X} w_{(k\cdot l)}(t) \cdots \underbrace{0 \cdots 0}_{n-1} \sum_{k=0}^{k_{0}-1} \sum_{l=1}^{p} \left\langle b_{k}^{(l)}, \phi_{im_{i}} \right\rangle_{X} w_{(k\cdot l)}(t) \right]^{l}$$
(6.73)

Wektory własne są dane zależnością (6.57). Obliczmy czynnik $\psi_i^{(qd\gamma)T}(s_{i\gamma})\tilde{B}_iw_i$ używając wzorów (6.57) oraz (6.73):

$$\psi_{i}^{(qd\gamma)T}\left(s_{i\gamma}\right)\tilde{B}_{i}w_{i} = v_{i\gamma1}^{0T(d)}\left(s_{i\gamma}\right)\sum_{k=0}^{k_{0}-1}\sum_{l=1}^{p}\left\langle b_{k}^{(l)},\phi_{iq}\right\rangle_{X}w_{(k\cdot l)}(t)$$

$$i = 1, 2, 3, ..., \ \gamma = 1, 2, ..., r_{i}, \ d = 1, 2, ..., n_{i\gamma}, \ q = 1, 2, ..., m_{i}, \ \operatorname{Im}\left[s_{i\gamma}\right] = 0$$
(6.74)

Wnioskując podobnie jak w dowodzie twierdzenia 6.9.1 otrzymujemy, że w każdym wierszu macierzy \tilde{B}_i określonej przez (6.66) muszą występować dwa elementy o przeciwnym znaku.

6.10. Przykład 1

W niniejszym podpunkcie pokazano przykład zastosowań wybranych rezultatów z niniejszego rozdziału do analizy sterowalności elastycznej belki. Weźmy pod uwagę układ mechaniczny, opisany następującym liniowym równaniem różniczkowym cząstkowym, z dwiema siłami wymuszającymi $u_1(t), u_2(t) \in L^2([t_0,\infty], R)$.

$$\frac{\partial^2 x(z,t)}{\partial t^2} + \frac{\partial^4 x(z,t)}{\partial z^4} + \alpha \frac{\partial^5 x(z,t)}{\partial z^4 \partial t} - \beta \frac{\partial^3 x(z,t)}{\partial z^2 \partial t} + \gamma \frac{\partial^2 x(z,t)}{\partial z^2} = z^{n_1} u_1(t) + z^{n_2} u_2(t) \quad (6.75)$$

gdzie niezależna zmienna przestrzenna zmienia się w granicach $z \in (0, L_0)$, parametry α, β, γ są ograniczone przez nierówności $\alpha, \gamma > 0$, $\beta > 2$, czas $t \ge t_0$, a wykładniki n_1, n_2 są dodatnimi liczbami całkowitymi. Warunki początkowe mają postać:

$$x(z,0) = x_0(z), \quad \frac{\partial x(z,0)}{\partial t} = x_1(z), \quad z \in (0, L_0)$$
(6.76)

Natomiast warunki brzegowe:

$$x(0,t) = x(L_0,t) = \frac{\partial^2 x(0,t)}{\partial z^2} = \frac{\partial^2 x(L_0,t)}{\partial z^2} = 0, \quad t \ge t_0$$
(6.77)

Funkcja x(z,t) jest równa przesunięciu rozważanej belki elastycznej w kierunku osi Y w chwili czasu $t \ge t_0$ w punkcie $z \in (0, L_0)$. Pierwsze dwa człony w równaniu (6.75) są członami typowymi dla modelu idealnie sprężystej belki. Kolejne dwa czynniki reprezentują wewnętrzne tłumienie strukturalne, natomiast pozostały piąty człon odpowiada za siły osiowe działające na belkę. Dokładniejszy opis tego modelu oraz zjawisk zachodzących w belce można znaleźć w artykułach [11, 34, 112].

6.10.1. Przekształcenie równania różniczkowego cząstkowego

W podpunkcie wykonano następujące operacje: przekształcono równanie różniczkowe cząstkowe (6.75) do postaci abstrakcyjnego równania różniczkowego (6.1), a następnie do postaci nieskończonego ciągu (6.14). W tym celu zdefiniujmy liniowy nieograniczony operator A [35] o następującej postaci:

$$Ax(z) = \frac{\partial^4 x(z)}{\partial z^4}, \quad x \in D(A)$$
(6.78)

$$D(A) = \begin{cases} x(z) \in X([0, L_0], R) : \frac{d^4}{dz^4} x(z) \in L^2([0, L_0], R) \\ x(0) = x(L_0) = \frac{d^2 x}{dz^2}(0) = \frac{d^2 x}{dz^2}(L_0) = 0 \end{cases}$$
(6.79)

gdzie $X([0, L_0], R)$ oznacza przestrzeń Hilberta wszystkich funkcji całkowalnych z kwadratem na przedziale $[0, L_0]$. Można udowodnić ([35]), że wartości własne λ_i oraz funkcje własne $\phi_i(z)$ operatora *A* mają postać odpowiednio:

$$\lambda_{i} = \left(\frac{i\pi}{L_{0}}\right)^{4}, \quad \phi_{i}(z) = \sqrt{\frac{2}{L_{0}}} \sin\frac{\pi i z}{L_{0}}, \quad i = 1, 2, 3, \dots$$
(6.80)

oraz że operator A jest liniowy, samosprzężony i dodatnio określony. W szczególności można zdefiniować potęgę ułamkową operatora A ([35]):

$$A^{\frac{1}{2}}x = -\frac{\partial^2 x}{\partial z^2} \tag{6.81}$$

$$D(A^{\frac{1}{2}}) = \left\{ x \in X([0, L_0], R) : \frac{d^2}{dz^2} x(z) \in L^2([0, L_0], R) : x(0) = x(L_0) = 0 \right\}$$
(6.82)

Dla badanego układu mechanicznego (6.75) można określić operator wejścia:

$$B_0 u(t) = \sum_{l=1}^2 b_0^{(l)} u_l(t), \quad B_k \in L(U, X), \quad u_1(t), u_2(t) \in L^2\left(\left[t_0, \infty\right], R\right)$$
(6.83)

gdzie $b_0^{(1)} = z^{n_1}, b_0^{(2)} = z^{n_2}$. Stosując operatory *A* (6.78) oraz B_0 (6.83) do równania różniczkowego cząstkowego (6.75) otrzymujemy następujące operatorowe, zwyczajne równanie różniczkowe drugiego rzędu względem czasu *t* w przestrzeni Hilberta X:

$$\frac{d^2 x(t)}{dt^2} + \left(\alpha A + \beta A^{\frac{1}{2}}\right) \frac{dx(t)}{dt} + \left(A - \gamma A^{\frac{1}{2}}\right) x(t) = B_0 u(t), \quad t \ge t_0$$
(6.84)

Sprawdźmy, czy abstrakcyjne równanie różniczkowe (6.84) spełnia założenie układu (6.1) określone w sekcji 6.4. Na podstawie formuły (6.20) oraz równania (6.84) możemy obliczyć współczynniki jego równania charakterystycznego:

$$f_{i0}^{*} = \left(\frac{i\pi}{L_{0}}\right)^{4} - \gamma \left(\frac{i\pi}{L_{0}}\right)^{2}, \quad f_{i1}^{*} = \alpha \left(\frac{i\pi}{L_{0}}\right)^{4} + \beta \left(\frac{i\pi}{L_{0}}\right)^{2}, \quad i = 1, 2, 3, \dots$$
(6.85)

Pierwiastki równania charakterystycznego (6.84) można obliczyć następująco:

$$s_{i1} = \frac{-f_{i1}^* - \sqrt{f_{i1}^{*2} - 4f_{i0}^*}}{2}, \quad s_{i2} = \frac{-f_{i1}^* + \sqrt{f_{i1}^{*2} - 4f_{i0}^*}}{2}, \quad i = 1, 2, 3, \dots$$
(6.86)

Z równania (6.85) wynika, że dla $\alpha, \gamma > 0, \beta > 2$ czynnik $f_{i1}^{*2} - 4f_{i0}^{*}$ jest dodatni dla każdego $i = 1, 2, 3, \dots$ Dlatego z równania (6.86) wynika, że $s_{i1} \neq s_{i2}$ dla każdego $i = 1, 2, 3, \dots$

Biorąc pod uwagę, że operator stanu (6.78) ma jedynie pojedyncze wartości własne (równanie (6.80)), możemy wyznaczyć macierze stanu oraz wejścia A_i , B_{0i} w nieskończonym ciągu skończenie wymiarowych układów (6.14) w postaci:

$$A_{i} = \begin{bmatrix} 0 & 1 \\ -f_{i0}^{*} & -f_{i1}^{*} \end{bmatrix}, \quad B_{0i} = \begin{bmatrix} 0 & 0 \\ \left\langle b_{0}^{(1)}, \phi_{i} \right\rangle_{X} & \left\langle b_{0}^{(2)}, \phi_{i} \right\rangle_{X} \end{bmatrix}, \quad i = 1, 2, 3, \dots$$
(6.87)

gdzie ϕ_i wyraża się wzorem (6.80). Iloczyny skalarne $\langle b_0^{(l)}, \phi_i \rangle_{\chi}$ obliczamy, jak następuje:

$$\left\langle b_{0}^{(1,2)}, \phi_{i} \right\rangle_{X} = \sqrt{\frac{2}{L_{0}}} \int_{0}^{L_{0}} z^{n_{1,2}} \sin \frac{\pi i z}{L_{0}} dz, \quad i = 1, 2, 3, \dots$$
 (6.88)

Całek (6.88) nie da się wyrazić za pomocą skończonej kombinacji funkcji elementarnych dla dowolnych wykładników $n_1, n_2 \in Z_+$. Korzystając ze wzorów (6.88), możemy obliczyć macierz B_{0i}^* na podstawie wzoru (6.51):

$$B_{0i}^{*} = \left[\left\langle b_{0}^{(1)}, \phi_{i} \right\rangle_{X} \quad \left\langle b_{0}^{(2)}, \phi_{i} \right\rangle_{X} \right], \quad i = 1, 2, 3, \dots$$
(6.89)

Podsumowując, przekształciliśmy analizowany układ mechaniczny (6.75) do postaci (6.14), wymaganej przez twierdzenia o sterowalności wprowadzone w niniejszym rozdziale. Przeanalizujmy, czy ten układ jest sterowalny przy dwóch rodzajach dopuszczalnego zbioru sterowań.

6.10.2. Aproksymacyjna sterowalność bez ograniczeń

Do badania aproksymacyjnej sterowalności bez ograniczeń układu mechanicznego (6.75) użyjemy twierdzenia 6.9.2, które w tym przypadku redukuje się do warunku rzędu (6.67). W naszym przykładzie nieskończony ciąg równań (6.39) ma następującą postać:

$$rank\left[\sqrt{\frac{2}{L_0}}\int_{0}^{L_0} z^{n_1} \sin\frac{\pi i z}{L_0} dz, \quad \sqrt{\frac{2}{L_0}}\int_{0}^{L_0} z^{n_2} \sin\frac{\pi i z}{L_0} dz\right] = 1, \quad i = 1, 2, 3, \dots$$
(6.90)

Dla spełnienia warunku (6.90) konieczne jest, aby dla dowolnego i = 1, 2, 3, ...przynajmniej jeden element wektora $\left[\left\langle b_0^{(1)}, \phi_i \right\rangle_X \quad \left\langle b_0^{(2)}, \phi_i \right\rangle_X \right]$ był niezerowy.

Warunku (6.90) nie da się sprawdzić dla dowolnych wykładników n_1, n_2 poprzez proste obliczenie całek metodami:

- analityczną, gdyż nie da się ich wyrazić jawną formułą dla dowolnych wykładników $n_1, n_2 \in Z_+$,
- numeryczną, gdyż metoda numeryczna wymaga skonkretyzowania wartości wszystkich zmiennych występujących pod znakiem całki. Prawdziwość warunku (6.90) chcemy zbadać dla dowolnych wartości parametrów n₁, n₂. Ponadto, liczba równań do sprawdzenia w ciągu (6.90) jest nieskończona, a w pracy [83] udowodniono, że ciąg całek we wzorze (6.90) dąży do zera przy i → ∞.

Mimo to weryfikacja warunku (6.90) jest możliwa dla dowolnych wartości parametrów n_1, n_2 . Jest to możliwe dlatego, że do sprawdzenia rzędu wektora nie jest konieczna znajomość dokładnych wartości wszystkich jego elementów. Wystarczy oszacowanie, czy przynajmniej jeden z nich jest niezerowy. Oszacowanie takie przeprowadza poniższy lemat.

Lemat 6.1

Następująca całka:

$$\int_{0}^{\pi i} z^{n} \sin z \, dz, \quad i \in Z_{+}, \, z \in R$$
(6.91)

dla $n \in N$ znika jedynie dla parzystych wartości parametru *i*, gdy parametr *n* jest równy zeru. Dla niezerowych wartości *n* całka (6.91) jest dodatnia dla *n* nieparzystych oraz ujemna dla *n* parzystych.

Dowód lematu

• Przypadek A: i jest liczbą parzystą, n > 0.

Rozważaną całkę można przekształcić do następującej postaci:

$$\int_{0}^{\pi i} z^{n} \sin z \, dz = \sum_{j=0}^{\frac{1}{2}-1} \int_{2j\pi}^{(2j+2)\pi} z^{n} \sin z \, dz \tag{6.92}$$

Całka pod sumą może zostać przez odpowiednie podstawienie przepisana w formie:

$$\int_{2j\pi}^{(2j+2)\pi} z^n \sin z \, dz = \int_{2j\pi}^{(2j+1)\pi} [z^n - (z+\pi)^n] \sin z \, dz \tag{6.93}$$

Jest spełniona następująca oczywista nierówność:

$$z^{n} + \pi^{n} \le (z + \pi)^{n} \Longrightarrow z^{n} - (z + \pi)^{n} \le -\pi^{n}$$

$$(6.94)$$

Ponadto, w przedziale $[2j\pi, (2j+1)\pi]$, $j \in \mathbb{Z}$, funkcja $\sin(z)$ jest nieujemna. Dlatego jest spełniona następująca nierówność:

$$\int_{2j\pi}^{(2j+2)\pi} z^n \sin z \, dz \le -\pi^n \int_{2j\pi}^{(2j+1)\pi} z^n \sin z \, dz = -2\pi^n < 0 \tag{6.95}$$

Zatem, każda całka w sumie (6.92) jest ujemna, więc rozważana całka (6.91) jest ujemna i tym samym różna od zera.

• Przypadek B : *i* jest liczbą nieparzystą, n > 0

W tym przypadku całkę (6.91) można podobnie rozpisać przez następującą sumę:

$$\int_{0}^{\pi i} z^{n} \sin z \, dz = \int_{0}^{\pi} z^{n} \sin z \, dz + \sum_{j=1}^{\frac{l}{2}-1} \int_{(2j-1)\pi}^{(2j+1)\pi} z^{n} \sin z \, dz$$
(6.96)

Pierwsza z całek w równaniu (6.96) jest dodatnia. Po podobnym oszacowaniu, jak w przypadku A, uzyskujemy następującą nierówność:

$$\int_{(2j-1)\pi}^{(2j+1)\pi} z^n \sin z \, dz \ge 2\pi^n > 0 \tag{6.97}$$

Dlatego suma (6.96), a tym samym badana całka (6.91), jest dodatnia, czyli różna od zera.

• Przypadek C: n = 0.

W tym przypadku rozważana całka równa się 0 dla parzystych wartości i.

C.N.D.

Wróćmy do weryfikacji warunku rzędu (6.90). Do jego sprawdzenia zastosować można lemat 6.1. Całki występujące w równaniu (6.90) po podstawieniu $z = \frac{L_0}{\pi i} t$ przyjmują postać:

$$\int_{0}^{L_{0}} z^{n} \sin \frac{\pi i z}{L_{0}} dz = \left(\frac{L_{0}}{\pi i}\right)^{n+1} \int_{0}^{\pi i} t^{n} \sin t \, dt$$
(6.98)

Lemat 6.1 stwierdza, że dla niezerowych wartości ostatnia z całek w (6.98) jest różna od zera dla dowolnego i = 1, 2, 3, ... Wniosek: nieskończony ciąg równań (6.90) jest spełniony dla każdego i = 1, 2, 3, ... Z tego wynika, że układ mechaniczny (6.75) jest aproksymacyjnie sterowalny bez ograniczeń na sterowania.

6.10.3. Aproksymacyjna sterowalność przy nieujemnych, stożkowych ograniczeniach na sterowania

Ten rodzaj sterowalności układu mechanicznego (6.75) przeanalizowano korzystając z twierdzenia 6.9.2. W badanym układzie mechanicznym nie ma opóźnień w sterowaniach, więc mamy M = 0 i macierz \tilde{B}_i dana wzorem (6.66) jest równa macierzy B_{0i}^* określonej przez (6.89). Warunek 1 jest spełniony dla każdego i=1,2,3,..., jeżeli ustalimy $w_i = 0$. Warunek 2 wymaga, aby otoczka wypukła dopuszczalnego zbioru sterowań miała niepuste wnętrze w przestrzeni R^p , co jest spełnione. Warunek rzędu był weryfikowany w poprzednim podpunkcie. Ponadto, jak już sprawdzono w poprzednim podpunkcie, jest spełniona nierówność:

$$f_{i1}^{*2} - 4f_{i0}^{*} > 0, \quad i = 1, 2, 3, \dots$$
(6.99)

więc wszystkie pierwiastki s_{i1} , s_{i2} w tym przypadku są rzeczywiste. Dlatego warunek 4 twierdzenia 6.9.2 jest równoznaczny wymaganiu, aby dla każdego i = 1, 2, 3, ... w macierzy wejścia B_{0i} (6.87) występowała para iloczynów skalarnych przeciwnego znaku. Uwzględniając postać macierzy B_{0i} (6.87) oraz postać iloczynów skalarnych (6.88), warunek 4 twierdzenia 6.9.2 jest równoważny nierówności:

$$\bigvee_{i=1,2,3,\dots} \left[\int_{0}^{L_{0}} z^{n_{1}} \sin \frac{\pi i z}{L_{0}} dz \right] \cdot \left[\int_{0}^{L_{0}} z^{n_{2}} \sin \frac{\pi i z}{L_{0}} dz \right] < 0$$
(6.100)

Lemat 6.1 stwierdza, że całka (6.98) jest dodatnia dla *n* nieparzystych oraz ujemna dla *n* parzystych. Stąd wynika, że warunek 4 twierdzenia 6.9.2 jest spełniony wtedy i tylko wtedy, gdy wykładniki n_1, n_2 są liczbami o różnych parzystościach. Teraz przeanalizujmy warunek 5 twierdzenia 6.9.2. Z zalezności (6.85) wynika nierówność:

$$f_{i1}^* > 0, \ i = 1, 2, 3, \dots$$
 (6.101)

Zatem, ze wzorów (6.86) i (6.99) można wydedukować, że $\text{Re}[s_{i1}] < 0$ dla każdego i = 1, 2, 3, ... Teraz przeanalizujmy nierówność $\text{Re}[s_{i2}] < 0, i = 1, 2, 3, ...$ Na podstawie nierówności (6.99), (6.101) oraz (6.86) widać, że $\text{Re}[s_{i2}] < 0, i = 1, 2, 3, ...$, wtedy i tylko wtedy, gdy $f_{i0}^* > 0$. Ze wzoru (6.85) wynika, że f_{i0}^* jest dodatnie wtedy i tylko wtedy, gdy zachodzi nierówność $\gamma < \pi^2 / L_0^2$.

6.10.4. Podsumowanie przykładu

Belka elastyczna (6.75) jest:

aproksymacyjnie sterowalna,

• aproksymacyjnie sterowalna przy nieujemnych, stożkowych ograniczeniach na sterowania wtedy i tylko wtedy, gdy wykładniki n_1, n_2 są liczbami całkowitymi o różnej parzystości oraz gdy parametr γ spełnia nierówność $\gamma < \pi^2 / L_0^2$.

6.11. Przykład 2

W punkcie pokazano przykład zastosowania algorytmu odwracania konfluentnej macierzy Vandermonde'a do analizy sterowalności układu dynamicznego modelowanego następującym równaniem różniczkowym zwyczajnym dziesiątego rzędu:

$$\frac{d^{10}y(t)}{dt^{10}} + a_1 \frac{d^9y(t)}{dt^9} + \dots + a_9 \frac{dy(t)}{dt} + a_{10}y(t) = 10u(t)$$
(6.102)

gdzie wartości liczbowe współczynników $a_1,...,a_{10}$ są dane w tabeli 5. Równanie (6.102) można przekształcić do postaci Frobeniusa:

$$\frac{d}{dt}\begin{bmatrix} x_{1}(t) \\ x_{2}(t) \\ \vdots \\ x_{10}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -a_{10} & -a_{9} & \cdots & -a_{1} \end{bmatrix} \begin{bmatrix} x_{1}(t) \\ x_{2}(t) \\ \vdots \\ x_{10}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ b \end{bmatrix} u(t)$$
(6.103)

gdzie $x_{i+1}(t) = \frac{d^i y(t)}{dt^i}$, i = 0, 1, ..., 9. Analizę sterowalności przeprowadzimy na podstawie twierdzenia Chena [10]. Wymaga ono znajomości odwrotności macierzy podobieństwa formy kanonicznej Jordana badanego układu, która dla układu w formie Frobeniusa ma postać konfluentnej macierzy Vandermonde'a. Odwrotność ta, oznaczana przez $T^{-1} = Q$, została wyznaczana za pomocą algorytmów 5.6.1 oraz 5.6.2 w rozdziale 5 i wyraża się formułą (5.28).

Twierdzenie Chena wymaga, aby wszystkie komponenty wektora $T^{-1}B$ odpowiadające ostatnim wierszom każdego bloku Jordana były różne od zera. Stosując tę zasadę do macierzy T^{-1} danej formułą (5.28) widzimy, że $K_{1,1}b \neq 0$, $K_{2,2}b \neq 0$, $K_{3,3}b \neq 0$, $K_{4,4}b \neq 0$, więc warunek sterowalności dla równania różniczkowego (6.102) jest spełniony.

90

6.11.1. Porównanie złożoności obliczeniowej klasycznej metody badania sterowalności z metodą algorytmicznego odwracania macierzy specjalnych

- Klasyczna metoda badania sterowalności układu wymaga wyznaczenia macierzy blokowej o postaci $[B|AB|A^2B|...|A^{n-1}B]$. Jest to metoda bardzo nieefektywna o złożoności obliczeniowej klasy $O(N^4)$.
- Zaproponowana w niniejszej monografii metoda badania sterowalności dla układów w postaci kanonicznej Frobeniusa zamiast wyznaczenia macierzy blokowej wymaga obliczenia odwrotności konfluentnej macierzy Vandermonde'a. Przedstawiony w rozdziale 5 algorytm realizujący to zadanie posiada złożoność obliczeniową równą $O(N^2)$.

Reasumując, zastosowanie algorytmów przedstawionych w rozdziałach 4 i 5 niniejszej monografii w zagadnieniach sterowalności układów Frobeniusa poprawia złożoność obliczeniową tych problemów o czynnik kwadratowy, redukując ją z klasy $O(N^4)$ do $O(N^2)$.

6.12. Podsumowanie rozdziału

W rozdziale sformułowano oraz udowodniono warunki dwóch rodzajów sterowalności dla układów dynamicznych dowolnego rzędu. Zaproponowana metodologia od strony analitycznej polega na sprowadzeniu wyjściowego układu do postaci Frobeniusa, a następnie przekształcenie jej do postaci kanonicznej Jordana. Z literatury wiadomo, że dla układu w postaci Frobeniusa macierz podobieństwa formy kanonicznej Jordana jest konfluentną macierzą Vandermonde'a. Spostrzeżenie to pozwoliło na połączenie metod analitycznych z algorytmicznymi. Znajdują zastosowanie:

- algorytm obliczania wielomianów symetrycznych podstawowych, zaproponowany w rozdziale 4,
- algorytm odwracania konfluentnych macierzy Vandermonde'a, przedstawiony w rozdziale 5.

Zaproponowaną metodologię przedstawiono na diagramie rys. 4. Kluczem do uzyskania wyników ważnych dla dowolnego stopnia układu jest połączenie metod analitycznych z wykorzystaniem algorytmów numerycznych. Metody analityczne znalazły zastosowanie z kolei w analizie przykładowej belki elastycznej, gdyż ich ogólność pozwoliła na zbadanie warunku rzędu (6.90), pomimo że:

- ciąg warunków jest nieskończony,
- elementy badanej macierzy sterowalności dążą do zera przy parametrze *i* dążącym do nieskończoności ([83] s.1021),
- dla ogólnych wartości wykładników potęg n_1, n_2 funkcji wymuszających nie jest możliwe wyznaczenie dokładnej wartości iloczynów skalarnych (6.88). Jednak jest możliwe analityczne oszacowanie ich znaków (lemat 6.1).

Samo zadanie analizy warunku rzędu (6.90) nie jest możliwe do zbadania na drodze numerycznej. Dopiero uzupełnienie metod analitycznych o metody algorytmiczne pozwoliło na zbadanie układu (6.1) dla dowolnego, n-tego rzędu względem czasu. Stanowi to krok naprzód w tej problematyce, gdyż dotychczasowe prace dotyczyły jedynie układów drugiego i czwartego rzędu.

7. NUMERYCZNE WYZNACZANIE ZBIORÓW OSIĄGALNYCH WYBRANEJ KLASY UKŁADÓW DYNAMICZNYCH

7.1. Cel rozdziału

Celem rozdziału jest wyznaczenie metodami analizy numerycznej zbiorów osiągalnych układów dynamicznych modelowanych równaniami różniczkowymi cząstkowymi. Innowacją w stosunku do dotychczasowego stanu wiedzy dotyczącego tego problemu jest uwzględnienie realistycznej postaci ograniczeń na funkcję wymuszającą. W rozdziale są wyznaczane zbiory osiągalne układów dynamicznych przy uwzględnieniu, że wartości funkcji wymuszającej są ograniczone zarówno z góry, jak i z dołu. W rozdziale podsumowano wyniki z publikacji [81, 84].

7.2. Znaczenie metod analizy numerycznej w problemach rozdziału

Zastosowanie w rozdziale metod analizy numerycznej pozwoliło na znalezienie zbioru osiągalnego dla układu dynamicznego modelowanego równaniem różniczkowym cząstkowym, przy realistycznych ograniczeniach na funkcję wymuszającą. Jest to zagadnienie bardziej złożone niż w przypadku równania różniczkowego zwyczajnego lub równania różniczkowego cząstkowego, ale bez ograniczeń na funkcję wymuszającą. Dotychczas w literaturze to zagadnienie było badane metodami analitycznymi, co powodowało wymienione poniżej ograniczenia:

- znajdowano zbiory osiągalne rozwiązań dla równań różniczkowych cząstkowych, ale bez uwzględnienia ograniczeń na funkcję wymuszającą; powodowało to, że uzyskane wyniki były mało realistyczne, gdyż zakładały dostępność wymuszenia o dowolnie dużej wartości, co jest fizycznie niemożliwe,
- analizowano zbiory osiągalne przy obustronnie ograniczonej funkcji wymuszającej, ale tylko dla równań różniczkowych zwyczajnych; znacznie ograniczało to klasę układów fizycznych, którą można było modelować za pomocą tych metod.

Dopiero połączenie metod analizy numerycznej z metodami o charakterze analitycznym pozwoliło na przezwyciężenie powyższych ograniczeń, tj. znalezienie zbiorów osiągalnych:

- dla układów dynamicznych modelowanych równaniami różniczkowymi cząstkowymi,
- przy uwzględnieniu zarówno dolnego, jak i górnego ograniczenia na funkcję wymuszającą.

7.3. Podstawy teoretyczne badania zbiorów osiągalnych

W celu badania zbiorów osiągalnych przy obustronnie ograniczonej funkcji wymuszającej zastosujemy twierdzenia podane w pracy [116], która rozważa liniowy układ ciągły następującej postaci:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{7.1}$$

gdzie $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ oraz macierze *A*, *B* są danymi macierzami o odpowiednich wymiarach. Zbiór osiągalny *K* jest modelowany następująco:

$$K = \{x \colon Lx = c\} \tag{7.2}$$

gdzie *L* oznacza daną $p \times n$ -wymiarową macierz rzędu *p* oraz *c* jest danym *p*-wymiarowym wektorem. Zbiór funkcji wymuszających u(t) jest dany inkluzją $u(t) \in M(\Omega)$, gdzie Ω jest danym zbiorem zwartym w przestrzeni R^m oraz $M(\Omega)$ oznacza zbiór funkcji mierzalnych na przedziale $[0,\infty]$ działających z *R* w zbiór Ω .

7.3.1. Definicja ([116] s. 761)

Zbiór osiągalny układu liniowego (7.1) zawiera punkt x(T) wtedy i tylko wtedy, gdy przy danym warunku początkowym $x(0) = x_0$, istnieje funkcja wymuszająca $u(t) \in M(\Omega)$ taka, że rozwiązanie x(t) układu (7.1) spełnia równość Lx(T) = c dla pewnego $T \in [0, \infty)$.

Metody wyznaczania zbiorów osiągalnych zaproponowane w artykule [116] zostały podane za pomocą skalarnej funkcji podporowej $J: \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^p \to \mathbb{R}$, którą zdefiniowano wzorem:

$$J(x_0, T, \alpha) = \alpha^T L e^{AT} x_0 + \int_0^T \max\left\{\omega^T B^T e^{A(t-\tau)} L^T \alpha : \omega \in \Omega\right\} d\tau - \alpha^T c$$
(7.3)

Korzystając z definicji 7.3.1, z funkcji podporowej (7.3) oraz ze zbioru osiągalnego (7.2), zaprezentujmy twierdzenie służące do wyznaczenia zbioru osiągalnego przy funkcjach wymuszających o wartościach ograniczonych do zbioru zwartego.

7.3.2. Twierdzenie ([116] s. 761)

Wybierzmy dowolny podzbiór H z przestrzeni R^p , zawierający 0 jako punkt wewnętrzny. Wtedy zbiór osiągalny układu dynamicznego (7.1) zawiera punkt x(T), przy danym warunku początkowym $x(0) = x_0$, wtedy i tylko wtedy, gdy zachodzi ekstremum:

94

$$\min\left\{J(x_0, T, \alpha) : \alpha \in H\right\} = 0 \tag{7.4}$$

dla pewnego $T \in [0,\infty)$.

7.4. Model matematyczny badanego układu dynamicznego

Rozważmy układ dynamiczny o parametrach rozłożonych, modelowany następującym równaniem różniczkowym cząstkowym typu parabolicznego:

$$\frac{\partial x(z,t)}{\partial t} - \frac{\partial^2 x(z,t)}{\partial z^2} = u(z,t), \qquad \bigvee_{(z,t) \in [0,L_0] \times [0,T]} u_{\min}(z) \le u(z,t) \le u_{\max}(z)$$
(7.5)

Określony w dziedzinie:

$$D = \left\{ \left(z, t\right) : 0 \le z \le L_0, \quad 0 \le t \le T, \quad L_0 \in R_+ \right\}$$
(7.6)

O funkcji wymuszającej u(z,t) zakładamy, że jest ciągła w dziedzinie D względem obydwu parametrów z, t. Warunek początkowy ma postać:

$$x(z,0) = g(z), \quad 0 \le z \le L_0$$
(7.7)

Zakładamy następujące, niezerowe oraz niestacjonarne warunki brzegowe typu Dirichleta:

$$x(0,t) = \varphi(t), \ x(L_0,t) = \psi(t), \ 0 \le t \le T$$
(7.8)

Ponadto zakładamy, że funkcja g(z) jest ciągła w przedziale $[0, L_0]$ oraz funkcje $\varphi(t), \psi(t)$ są ciągłe w przedziale [0, T].

7.5. Zastosowana w rozdziale metodologia badań numerycznych

- Celem niniejszego rozdziału jest wyznaczenie na drodze numerycznej zbioru osiągalnego układu dynamicznego typu parabolicznego przy obustronnie ograniczonej (tzn. z dołu i z góry) funkcji wymuszającej i niezerowych warunkach brzegowych. Jako narzędzie do analizy zbiorów osiągalnych w rozdziale użyto metody numerycznej prostych. Polega ona na tym, że w równaniu różniczkowym cząstkowym dyskretyzujemy jedynie pochodną względem zmiennej przestrzennej. Dyskretyzacja metodą prostych przekształca wyjściowe równanie różniczkowe cząstkowe w skończenie wymiarowe macierzowe równanie różniczkowe zwyczajne o postaci (7.1). Dlatego metoda numeryczna prostych pozwoliła na zastosowanie pojęć i twierdzeń z punktu 7.3, gdyż kryterium Schmitendorfa i Barmisha można stosować do macierzowych równań różniczkowych zwyczajnych.
- Zastosowanie warunku 7.3.2 dla układu skończenie wymiarowego wymaga rozwiązania problemu optymalizacyjnego. Jest to możliwe po spostrzeżeniu, że metoda częściowej dyskretyzacji równań cząstkowych, tj. metoda prostych, prowadzi do trójdiagonalnej

macierzy o specjalnej postaci. Uzyskana w wyniku dyskretyzacji macierz układu ma nieujemne wszystkie elementy poza główną przekątną tzn. jest macierzą Metzlera ([39] s.336). Ważną własnością macierzy Metzlera jest fakt, że ich eksponenta macierzowa ma wszystkie elementy nieujemne ([39] s.336). Umożliwia to rozwiązanie problemu optymalizacji min-maksowej, koniecznego do wyznaczenia zbiorów osiągalnych za pomocą pojęć przedstawionych w rozdziale 7.3.

7.6. Dyskretyzacja badanego układu dynamicznego metodą prostych

Monografiami poświeconymi metodom numerycznym rozwiazywania równań różniczkowych cząstkowych są Morton [69] oraz Knaber [50]. W monografii [77] temu zagadnieniu jest poświęcony rozdział 19 s. 827-888. W literaturze są stosowane dwie podstawowe metody numeryczne rozwiazywania równań różniczkowych cząstkowych: metoda siatek i metoda prostych. Metoda siatek polega na pełnej dyskretyzacji rozważanego równania cząstkowego zarówno względem zmiennych przestrzennych, jak i względem czasu. Z kolei metoda prostych polega na tym, że w analizowanym równaniu dyskretyzuje się jedynie zmienną przestrzenną, pozostawiając w równaniu pochodną względem czasu. Metoda prostych prowadzi do aproksymacji wyjściowego równania różniczkowego cząstkowego przez odpowiadające mu macierzowe, skończenie wymiarowe równanie różniczkowe zwyczajne. Monografia poświęconą metodzie prostych jest Schiesser [115]. Metoda prostych jest omawiana także w pracach: Knaber [50] s.293-311, Tichonow [129] s.443-451, Fortuna [24] s.367-370, Klamka [49] s.173-174.

Zdyskretyzujmy badany układ dynamiczny, modelowany równaniem różniczkowym cząstkowym (7.5), za pomocą metody prostych. W tym celu dziedzinę D równania (7.5), określoną zależnością (7.6), pokrywamy rodziną równo odległych prostych o wzajemnej odległości h:

$$z_i = ih, \quad h = L_0 / N, \quad i = 0, 1, ..., N$$
 (7.9)

W wyniku otrzymujemy następujące, skończenie wymiarowe macierzowe równanie różniczkowe zwyczajne:

$$\frac{dX}{dt} = \frac{1}{h^2} A_0 X(t) + BU(t), \quad X(0) = G$$
(7.10)

gdzie wektor stanu X(t), wymuszenia U(t) oraz wektor warunków początkowych G, odpowiadające warunkom brzegowym i początkowym (7.7) i (7.8) mają odpowiednio postać:

$$X(t) = \begin{bmatrix} x_{1}(t) \\ x_{2}(t) \\ \vdots \\ x_{N-2}(t) \\ x_{N-1}(t) \end{bmatrix}, \quad U(t) = \begin{bmatrix} u(z_{1},t) + \frac{1}{h^{2}}\varphi(t) \\ u(z_{2},t) \\ \vdots \\ u(z_{N-2},t) \\ u(z_{N-1},t) + \frac{1}{h^{2}}\psi(t) \end{bmatrix}, \quad G = \begin{bmatrix} g(z_{1}) \\ g(z_{2}) \\ \vdots \\ g(z_{N-2}) \\ g(z_{N-1}) \end{bmatrix}$$
(7.11)

gdzie $x_i(t)$ oznacza przybliżone rozwiązanie równania różniczkowego cząstkowego (7.5) na prostej $z_i = ih$, natomiast $u(z_i, t) = u(ih, t)$, $g(z_i) = g(ih)$, i = 0, 1, ..., N.

Macierz wejścia B w równaniu (7.10) jest macierzą identycznościową, natomiast macierz stanu A_0 jest następującą trójdiagonalną macierzą Metzlera:

$$A_{0} = \begin{bmatrix} -2 & 1 & & 0 \\ 1 & -2 & \ddots & \\ & \ddots & \ddots & \ddots \\ & & \ddots & -2 & 1 \\ 0 & & 1 & -2 \end{bmatrix} \in R^{(N-1)\times(N-1)}$$
(7.12)

Można udowodnić ([129 s.444, 24 s.368]), że:

$$|x(z_i,t) - x_i(t)| \le Mh^2$$

gdzie M jest stałą niezależną od h. Równanie (7.10) stanowi podstawę dalszej analizy.

7.7. Własności spektralne macierzy stanu układu dynamicznego

W dalszych obliczenia będą potrzebne własności spektralne układu dynamicznego (7.10), tj. wartości własne λ_i macierzy stanu $\frac{1}{h^2}A_0$. Literatura ([6, 33]) podaje postać wartości własnych dla macierzy A_0 (7.12). Uwzględniając, że dim $A_0 = (N-1) \times (N-1)$ zachodzi:

$$\lambda_i = -2 + 2\cos\left(\frac{\pi i}{N}\right), \qquad i = 1, 2, ..., N - 1$$
(7.13)

Obliczmy teraz wartości własne macierzy układu dynamicznego (7.10), biorąc pod uwagę równość macierzową $A = \frac{1}{h^2} A_0$. Z definicji wartości własnych kolejno obliczamy:

$$\det(A_0 - \lambda I) = 0 \iff \det\left(\frac{1}{h^2}A_0 - \frac{\lambda}{h^2}I\right) = 0 \iff \det\left(A - \frac{\lambda}{h^2}I\right) = 0$$
(7.14)

Z równania (7.14) otrzymujemy wartości własne μ_i macierzy A:

$$\mu_i = \frac{\lambda_i}{h^2}, \quad i = 1, 2, ..., N - 1 \tag{7.15}$$

7.8. Konstrukcja finalnego algorytmu

Znalezienie zbioru osiągalnego dla układu dynamicznego (7.5) zostanie przeprowadzone za pomocą funkcji podporowej (7.3) oraz twierdzenia 7.3.2. Dla układu dynamicznego (7.5) funkcja podporowa (7.3) przyjmuje postać:

$$J(x_0,T,\alpha) = \alpha^T L e^{AT} x_0 - \alpha^T c + \int_0^T \max_{U(\tau) \in [U_{\min},U_{\max}]} \left\{ U^T(\tau) B^T e^{A(T-\tau)} L^T \alpha \right\} d\tau \qquad (7.16)$$

Załóżmy, że chcemy przeprowadzić układ dynamiczny (7.5) z punktu $x_q(t) = x(q \cdot \Delta z, t), q = 1, 2, ..., N-1$, do punktu docelowego $c \in R$, tj. chcemy, aby $x_q(T) = c$. Dlatego w zbiorze osiągalnym określonym wzorem K (7.2) należy przyjąć p = 1 oraz:

$$L = [\underbrace{0...0}_{q} 1 \ 0...0], \quad q = 1, 2, ..., N - 1$$
(7.17)

Ponadto, dla p = 1 podzbiór H w twierdzeniu 7.3.2 jest dowolnym przedziałem zawierającym 0 jako punkt wewnętrzny. Dlatego jako ten przedział przyjmijmy H = [-1, 1]. Oznaczmy przez $\left[e^{A(T-\tau)}\right]_{ij}$ element znajdujący się na przecięciu *i*-tego rzędu oraz *j*-tej kolumny eksponenty macierzowej. Uwzględniając, że B = I, funkcja podporowa przyjmuje następującą postać:

$$J(x_{q0},T,\alpha) = \alpha \left[e^{AT} \right]_{qq} x_{q}(0) - \alpha c + \int_{0}^{T} \max_{u(z_{i},\tau) \in \left[u_{\min}(z_{i}), u_{\max}(z_{i})\right]} \left\{ \alpha \sum_{i=1}^{N-1} u(z_{i},\tau) \left[e^{A(T-\tau)} \right]_{qi} \right\} d\tau$$

$$(7.18)$$

Postać eksponenty macierzowej $e^{A(T-\tau)}$ można znaleźć w literaturze ([66]):

$$e^{A(T-\tau)} = \begin{bmatrix} \frac{2}{N^2} \sum_{k=1}^{N-1} e^{\mu_k(T-\tau)} \sin\left(\frac{k\pi}{N}\right) \sin\left(\frac{k\pi}{N}\right) & \cdots & \frac{2}{N^2} \sum_{k=1}^{N-1} e^{\mu_k(T-\tau)} \sin\left(\frac{k\pi}{N}\right) \sin\left((N-1)\frac{k\pi}{N}\right) \\ \vdots & \ddots & \vdots \\ \frac{2}{N^2} \sum_{k=1}^{N-1} e^{\mu_k(T-\tau)} \sin\left((N-1)\frac{k\pi}{N}\right) \sin\left(\frac{k\pi}{N}\right) \cdots \frac{2}{N^2} \sum_{k=1}^{N-1} e^{\mu_k(T-\tau)} \sin\left((N-1)\frac{k\pi}{N}\right) \sin\left((N-1)\frac{k\pi}{N}\right) \end{bmatrix}$$
(7.19)

Przy użyciu eksponenty macierzowej (7.19) funkcję podporową (7.18) można zapisać wprost jako:

$$J(x_{q0},T,\alpha) = \alpha \frac{2x_{q}(0)}{N^{2}} \sum_{k=1}^{N-1} e^{\mu_{k}T} \sin^{2}\left(q \frac{k\pi}{N}\right) - \alpha c + \int_{0}^{T} \max_{u(z_{i},\tau) \in [u_{\min}(z_{i}), u_{\max}(z_{i})]} \left\{ \frac{2\alpha}{N^{2}} \sum_{i=1}^{N-1} u(z_{i},\tau) \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left(i \frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right) \right\} d\tau$$
(7.20)

Parametr α zmienia się w przedziale domkniętym [-1,1]. Warunek (7.4) twierdzenia 7.3.2, które jest stosowane jako główne narzędzie w niniejszym rozdziale, można przekształcić do postaci:

$$\min\left\{J\left(x_{q0},T,\alpha\right):\alpha\in\left[-1,1\right]\right\}=0\tag{7.21}$$

Dalsze obliczenie funkcji podporowej (7.20) może zostać przeprowadzone jedynie dla konkretnych postaci funkcji wymuszającej w analizowanym układzie dynamicznym modelowanym równaniem różniczkowym cząstkowym (7.5). Równości (7.20) i (7.21) stanowią podstawę analizy zbiorów osiągalnych przy obustronnie ograniczonej funkcji wymuszającej dla konkretnego układu dynamicznego.

7.8.1. Finalny algorytm analizy zbiorów osiągalnych

1. Wprowadź: model układu dynamicznego w postaci równania różniczkowego cząstkowego, warunki początkowe i brzegowe, funkcję wymuszającą wraz z dopuszczalnym zbiorem jej wartości. 2. Zdyskretyzuj metodą prostych badane równanie różniczkowe cząstkowe do postaci zwyczajnego równania różniczkowego macierzowego (7.10), przybliżając nieskończenie wymiarowy stan x(z,t) przez wektor X(t) oraz dwuargumentową funkcję wymuszającą u(z,t) przez wektor U(t) określony wzorem (7.11).

```
3. Znajdź konkretną postać funkcji podporowej J(x_{q0},T,\alpha) danej wzorem (7.21),
uwzględniając kształt ograniczeń na dopuszczalne wartości funkcji
wymuszającej (7.5).
4. Rozwiąż finalny warunek optymalizacyjny (7.21).
```

W kolejnym podpunkcie przedstawiono zastosowanie tego algorytmu dla dwóch konkretnych zbiorów dopuszczalnych wartości funkcji wymuszającej.

7.9. Testy numeryczne

7.9.1. Wymuszenie eksponencjalnie ważone

Poddajmy analizie następujący układ dynamiczny, modelowany parabolicznym równaniem różniczkowym cząstkowym z rozłożonym wymuszeniem postaci $u(z,t) = e^{k_0 z} u_0(t)$ jak następuje:

$$\frac{\partial x(z,t)}{\partial t} - \frac{\partial^2 x(z,t)}{\partial z^2} = e^{k_0 z} u_0(t), \qquad \bigvee_{t \in [0,T]} 0 < u_{0\min} \le u_0(t) \le u_{0\max}, \quad k_0 \in R_+$$
(7.22)

Zakładamy zerowe warunki brzegowe typu Dirichleta.

• Rozwiązanie problemu

Przeanalizujmy zbiory osiągalne układu dynamicznego (7.22) za pomocą algorytmu 7.8.1. W niniejszym przykładzie sednem analizy jest obliczenie całki oznaczonej (7.20), a w szczególności wyznaczenie jej maksimum. Najpierw przedstawmy zbiór dopuszczalnych wartości funkcji wymuszającej $f(z_i)u(\tau) \in \left[e^{k_0 z_i}u_{\min}(z_i), e^{k_0 z_i}u_{\max}(z_i)\right]$. Biorąc pod uwagę, że funkcja $f(z) = e^{k_0 z}$ jest rosnąca względem zmiennej z oraz $u_{0\min} \leq u_0(t) \leq u_{0\max}$, możemy obliczyć odpowiednie wektory ograniczeń:

$$U_{\min} = \begin{bmatrix} e^{k_0 z_1} u_{0\min} \\ e^{k_0 z_2} u_{0\min} \\ \vdots \\ e^{k_0 z_{N-2}} u_{0\min} \\ e^{k_0 z_{N-1}} u_{0\min} \end{bmatrix}, \qquad U_{\max} = \begin{bmatrix} e^{k_0 z_1} u_{0\max} \\ e^{k_0 z_2} u_{0\max} \\ \vdots \\ e^{k_0 z_{N-2}} u_{0\max} \\ e^{k_0 z_{N-1}} u_{0\max} \end{bmatrix}, \qquad z_i = ih = i\frac{L_0}{N}, \quad i = 1, 2, ..., N - 1 (7.23)$$

Uwzględnienie postaci ograniczeń na dopuszczalne wartości wymuszenia pozwala na obliczenie wartości całki oznaczonej w funkcji podporowej (7.20). Biorąc pod uwagę, że parametr α zmienia się w przedziale [-1,1] należy rozróżnić dwa przypadki:

- Przypadek A: $\alpha \in [0,1]$

Jak już wspomniano, macierz stanu *A* układu dynamicznego (7.10) jest macierzą Metzlera. Macierze Metzlera posiadają taką własność, że każdy element jej eksponenty jest nieujemny ([39]). Ten fakt znajduje zastosowanie w obliczaniu maksimum w funkcji podporowej. W naszym przypadku następujący wyraz funkcji podporowej:

$$\sum_{k=1}^{N-1} e^{\mu_k(T-\tau)} \sin\left(i\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right)$$
(7.24)

jest nieujemny. Dlatego uwzględniając ograniczenia (7.23), można obliczyć maksimum czynnika podcałkowego jak następuje:

$$\max_{\substack{u(z_{i},\tau)\in[u_{\min}(z_{i}),u_{\max}(z_{i})]}} \left\{ \frac{2\alpha}{N^{2}} \sum_{i=1}^{N-1} f(z_{i}) u(\tau) \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left(i\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) \right\} = \frac{2\alpha u_{0\max}}{N^{2}} \sum_{i=1}^{N-1} e^{k_{0}z_{i}} \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left(i\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right)$$
(7.25)

Korzystając z formuły (7.25) oraz wykonując całkowanie we wzorze (7.20), otrzymujemy następującą postać funkcji podporowej dla układu dynamicznego:

$$J(x_{q0},T,\alpha) = \alpha \frac{2x_{q}(0)}{N^{2}} \sum_{k=1}^{N-1} e^{\mu_{k}T} \sin^{2}\left(q \frac{k\pi}{N}\right) + \frac{2\alpha u_{0\max}}{N^{2}} \sum_{i=1}^{N-1} e^{k_{0}z_{i}} \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T}-1}{\mu_{k}} \sin\left(i \frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right) - \alpha c$$
(7.26)

- Przypadek B: $\alpha \in [-1, 0)$

Po podobnych obliczeniach, jak w przypadku A, funkcja podporowa przyjmuje postać:

$$J(x_{q0}, T, \alpha) = \alpha \frac{2x_{q}(0)}{N^{2}} \sum_{k=1}^{N-1} e^{\mu_{k}T} \sin^{2}\left(q \frac{k\pi}{N}\right) + \frac{2\alpha u_{0\min}}{N^{2}} \sum_{i=1}^{N-1} e^{k_{0}z_{i}} \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T} - 1}{\mu_{k}} \sin\left(i \frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right) - \alpha c$$
(7.27)

W toku dalszej analizy wygodne jest wprowadzenie pomocniczej funkcji $S(x_{q0}, T, u_0)$ zdefiniowanej przez wzór:

$$S(x_{q0}, T, u_0) = \frac{2x_q(0)}{N^2} \sum_{k=1}^{N-1} e^{\mu_k T} \sin^2\left(q \frac{k\pi}{N}\right) + \frac{2u_0}{N^2} \sum_{i=1}^{N-1} e^{k_0 z_i} \sum_{k=1}^{N-1} \frac{e^{\mu_k T} - 1}{\mu_k} \sin\left(i \frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right)$$
(7.28)

Korzystając z funkcji pomocniczej (7.28), funkcja podporowa (7.27) przyjmuje następującą postać:

$$J(x_{q0},T,\alpha) = \begin{cases} \alpha \left[S(x_{q0},T,u_{0\max}) - c \right], & \alpha \in [0,1] \\ \alpha \left[S(x_{q0},T,u_{0\min}) - c \right], & \alpha \in [-1,0) \end{cases}$$
(7.29)

Postać (7.29) jest użyteczna w stosowaniu twierdzenia 7.3.2, podającego warunki wyznaczania zbiorów osiągalnych. Można wywnioskować, że ekstremum $\min \{J(x_{q0}, T, \alpha) : \alpha \in [-1, 1]\} = 0$ zachodzi wtedy i tylko wtedy, gdy są spełnione nierówności:

$$\begin{cases} S(x_{q0}, T, u_{0\max}) - c \ge 0, \ dla \ \alpha \in [0, 1] \\ S(x_{q0}, T, u_{0\min}) - c \le 0, \ dla \ \alpha \in [-1, 0) \end{cases}$$
(7.30)

Układ nierówności (7.30) pozwala wyznaczyć zbiór osiągalny dla układu dynamicznego (7.22) przy obustronnie ograniczonym wymuszeniu. W celu jego wyznaczenia należy użyć twierdzenia 7.3.2 oraz uwzględnić postać funkcji podporowej (7.28), (7.29).

• Finalna postać algorytmu wyznaczania zbioru osiągalnego przy rozłożonym wymuszeniu brzegowym

Zbiór osiągalny węzła x = qh układu dynamicznego (7.22) modelowanego przez paraboliczne równanie różniczkowe cząstkowe, przy rozłożonym eksponencjalnie ważonym wymuszeniu $u(z,t) = e^{k_0 z} u_0(t)$ obustronnie ograniczonym do domkniętego przedziału $0 < u_{0\min} \le u_0(t) \le u_{0\max}$, wyraża się przez następującą nierówność podwójną:

$$S(x_{q0}, T, u_{0\min}) \le c \le S(x_{q0}, T, u_{0\max})$$
(7.31)

gdzie $S(x_{q0},T,u_0)$ jest funkcją pomocniczą daną wzorem (7.28) oraz c oznacza osiągalną wartość rozwiązania układu dynamicznego (7.22).

Z fizykalnego punktu widzenia nierówność (7.31) oznacza, iż *q*-ty węzeł układu dynamicznego (7.22) może osiągać wartości z przedziału $S(x_{q0}, T, u_{0\min})...S(x_{q0}, T, u_{0\max})$. Ponadto, warto zauważyć, że założenie obustronnych ograniczeń na dopuszczalne wartości funkcji wymuszającej powoduje, że zbiór osiągalny zależy zarówno od czasu ewolucji układu, jak i konkretnych wartości ograniczeń. Zależność ta nie zachodzi przy rozpatrywaniu zbiorów osiągalnych bez ograniczeń na funkcję wymuszającą.

• *Testy numeryczne*

W niniejszym podpunkcie przedstawiono numeryczny przykład wykonania algorytmu. Symulację przeprowadzono dla następujących wartości parametrów symulowanego układu dynamicznego, modelowanego równaniem różniczkowym cząstkowym (7.22):

- ograniczenia na funkcję wymuszającą: $u_{0\min} = 0, u_{0\max} = 10$,
- maksymalny czas symulacji: T = 0.5,
- liczba węzłów dyskretyzacji: N = 64,
- wzmocnienie: $k_0 = 1.5$,
- wielkość dziedziny: $L_0 = 2$,
- zerowy warunek początkowy.

Wyniki symulacji zaprezentowano na wykresach rys. 5 oraz rys. 6, przedstawiających maksymalną osiągalną wartość rozwiązania układu dynamicznego (7.22) względem: maksymalnego czasu symulacji (rys. 5) oraz górnego ograniczenia funkcji wymuszającej (rys. 6).



- Rys. 5. Maksymalna wartość osiągalna rozwiązania układu dynamicznego względem maksymalnego czasu ewolucji T
- Fig. 5. Maximum attainable state value with respect to maximum control time T





Fig. 6. Maximum attainable state value with respect to upper control constraint u_{0max}

7.9.2. Wymuszenie brzegowe

Przeanalizujmy następujący układ dynamiczny modelowany parabolicznym równaniem różniczkowym cząstkowym:

$$\frac{\partial x(z,t)}{\partial t} - \frac{\partial^2 x(z,t)}{\partial z^2} = 0, \qquad (z,t) \in [0, L_0] \times [0,T]$$
(7.32)

Zakładamy następujące wymuszenie na obydwu brzegach dziedziny przestrzennej równania $[0, L_0]$:

$$x(0,t) = \varphi(t), \quad x(L_0,t) = \psi(t), \qquad \bigvee \begin{array}{c} 0 < \varphi_{\min} \le \varphi(t) \le \varphi_{\max} \\ t \in [0,T] \end{array}, \quad 0 < \psi_{\min} \le \psi(t) \le \psi_{\max} \end{array}, \quad 0 \le t \le T$$
(7.33)

• Rozwiązanie problemu

Analizę zbiorów osiągalnych układu dynamicznego (7.32) przeprowadzamy za pomocą algorytmu 7.8.1. Dopuszczalny zbiór wartości funkcji wymuszenia przyjmuje postać:

$$U_{\min} = \begin{bmatrix} \frac{1}{h^2} \varphi_{\min} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{h^2} \psi_{\min} \end{bmatrix}, \qquad U_{\max} = \begin{bmatrix} \frac{1}{h^2} \varphi_{\max} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{h^2} \psi_{\max} \end{bmatrix}, \qquad z_i = ih = i\frac{L_0}{N}, \quad i = 1, 2, ..., N-1$$
(7.34)

Rozróżnijmy dwa przypadki. Gdy $\alpha \in [0, 1]$, maksimum funkcji podcałkowej w funkcji podporowej (7.20) można obliczyć uwzględniając zbiór dopuszczalnych wartości funkcji wymuszającej:

$$\max_{\substack{\left[\frac{1}{h^{2}}\varphi_{\min},\frac{1}{h^{2}}\varphi_{\max}\right]_{l=1}\\0, i=2,3,...,N-2}} \left\{ \frac{2\alpha}{N^{2}} \sum_{i=1}^{N-1} u(z_{i},\tau) \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left(i\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) \right\} = \frac{2\alpha}{N^{2}h^{2}} \left[\varphi_{\max} \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left(\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) + \psi_{\max} \sum_{k=1}^{N-1} e^{\mu_{k}(T-\tau)} \sin\left((N-1)\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) \right]$$
(7.35)

Za pomocą tożsamości (7.35) możemy wyrazić wprost funkcję podporową dla układu dynamicznego (7.32) przy dopuszczalnym zbiorze wartości wymuszenia o postaci:

$$J(x_{q0}, T, \alpha) = \alpha \frac{2x_{q}(0)}{N^{2}} \sum_{k=1}^{N-1} e^{\mu_{k}T} \sin^{2}\left(q \frac{k\pi}{N}\right) - \alpha c + \frac{2\alpha}{N^{2}h^{2}} \left[\varphi_{\max} \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T} - 1}{\mu_{k}} \sin\left(\frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right) + \psi_{\max} \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T} - 1}{\mu_{k}} \sin\left((N-1)\frac{k\pi}{N}\right) \sin\left(q \frac{k\pi}{N}\right) \right]^{(7.36)}$$

Obliczenia przebiegają analogicznie w przypadku, gdy $\alpha \in [-1, 0)$. W tym przykładzie numerycznym funkcja pomocnicza $S(x_{q0}, T, \varphi, \psi)$ może zostać zdefiniowana jako:

$$S(x_{q0}, T, \varphi, \psi) = \frac{2x_{q}(0)}{N^{2}} \sum_{k=1}^{N-1} e^{\mu_{k}T} \sin^{2}\left(q\frac{k\pi}{N}\right) + \frac{2}{N^{2}h^{2}} \left[\varphi \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T} - 1}{\mu_{k}} \sin\left(\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) + \psi \sum_{k=1}^{N-1} \frac{e^{\mu_{k}T} - 1}{\mu_{k}} \sin\left((N-1)\frac{k\pi}{N}\right) \sin\left(q\frac{k\pi}{N}\right) \right]^{(7.37)}$$

Przy użyciu pomocniczej funkcji $S(x_{q0}, T, \varphi, \psi)$ (7.37), funkcja podporowa przyjmuje następującą postać:

$$J(x_{q0},T,\alpha) = \begin{cases} \alpha \left[S(x_{q0},T,\varphi_{\max},\psi_{\max}) - c \right], & \alpha \in [0,1] \\ \alpha \left[S(x_{q0},T,\varphi_{\min},\psi_{\min}) - c \right], & \alpha \in [-1,0) \end{cases}$$
(7.38)

Zastosowanie kryterium optymalizacyjnego 7.3.2 do funkcji podporowej (7.38) prowadzi do szukanego zbioru osiągalnego przy obustronnie ograniczonym wymuszeniu.

• Finalna postać algorytmu wyznaczania zbioru osiągalnego przy obustronnym wymuszeniu brzegowym

Zbiór osiągalny węzła x = qh układu dynamicznego (7.32) przy obustronnym wymuszeniu brzegowym, ograniczonym do domkniętych przedziałów (7.33), wyraża się przez następujący przedział:

$$S\left(x_{q0}, T, \varphi_{\min}, \psi_{\min}\right) \le c \le S\left(x_{q0}, T, \varphi_{\max}, \psi_{\max}\right)$$
(7.39)

gdzie c oznacza konkretną wartość rozwiązania osiąganą przez układ dynamiczny (7.32).

• Testy numeryczne

W podpunkcie przedstawiono numeryczny przykład wyznaczania zbioru osiągalnego przy obustronnym wymuszeniu brzegowym. Testy przeprowadzono przy następujących parametrach symulowanego układu dynamicznego:

- dopuszczalne wartości wymuszeń brzegowych: $\varphi_{\min} = \psi_{\min} = 0$, $\varphi_{\max} = 4$, $\psi_{\max} = 10$,
- maksymalny czas symulacji: T = 0.5,
- liczba węzłów dyskretyzacji: N = 64,
- wielkość dziedziny: $L_0 = 2$,
- zerowy warunek początkowy.

Wyniki symulacji przedstawiono na wykresach rys. 7 oraz rys. 8, przedstawiających maksymalną osiągalną wartość przez rozwiązanie badanego układu dynamicznego (7.32) względem maksymalnego czasu symulacji (rys. 7) oraz maksymalnej wartości wymuszenia $\psi = 0,..,10$ (rys. 8).



- Rys. 7. Maksymalna wartość osiągalna rozwiązania układu dynamicznego względem maksymalnego czasu ewolucji *T*
- Fig. 7. Maximum attainable state value with respect to maximum control time T





Fig. 8. Maximum attainable state value with respect to upper control constraint ψ

7.10. Podsumowanie

W rozdziale zastosowano metody numeryczne w analizie jednej z jakościowych własności układów dynamicznych, tj. zbiorów osiągalnych. Główną innowacją rozdziału jest zaproponowanie nowej metodologii badań, która uwzględnia fizyczne ograniczenia na funkcję wymuszającą, tj. jej górne oraz dolne ograniczenie.

Środkiem pozwalającym na przyjęcie realistycznego modelu matematycznego jest zastosowanie metod numerycznych w analizie równań różniczkowych cząstkowych. Zostały one połączone z twierdzeniami analitycznymi służącymi badaniu zbiorów osiągalnych dla równań różniczkowych zwyczajnych. Połączenie obydwu metodologii pozwoliło na uzyskanie nowych, bardziej realistycznych rezultatów.

8. PODSUMOWANIE ROZPRAWY

Wśród nowych algorytmicznych rezultatów rozprawy można wymienić:

- znalezienie algorytmu numerycznego rozwiązującego rozważane równanie diofantyczne,
- zastosowanie tegoż algorytmu do zbadania zbiorów osiągalnych układów dynamicznych modelowanych równaniami różniczkowymi cząstkowymi typu parabolicznego,
- zbudowanie efektywnego algorytmu numerycznego obliczania wielomianów symetrycznych podstawowych,
- zbudowanie algorytmu odwracania konfluentnych macierzy Vandermonde'a,
- zastosowanie powyższych dwóch algorytmów do analizy własności układu dynamicznego dowolnego rzędu,
- zbadanie metodami numerycznymi zbiorów osiągalnych układów dynamicznych typu parabolicznego przy uwzględnieniu obustronnych ograniczeń na funkcję wymuszającą.

Komitet Informatyki PAN w raporcie [43] wskazuje metody komputerowe jako istotne narzędzie współczesnej nauki. Niniejsza rozprawa przedstawia najważniejsze rezultaty autora, które zostały uzyskane za pomocą metod komputerowych.
BIBLIOGRAFIA

- Aho A. V., Hopcroft J. E., Ullman J. D.: Data structures and algorithms. Addison-Wesley, New York 1983.
- Alagić S., Arbib M. A.: Projektowanie programów poprawnych i dobrze zbudowanych. WNT, Warszawa 1982.
- 3. Alotaibi S., Sen M., Goodwine B., Yang K. T.: Controllability of cross-flow heat exchangers. International Journal of Heat and Mass Transfer, 2004, Vol. 47, s. 913÷924.
- 4. Baker A.: A concise introduction to the theory of numbers. Cambridge University Press, Cambridge 1984.
- 5. Bartosiewicz Z.: Partially defined control systems. Controllability and stabilizability. Applied Mathematics and Computer Science, 1995, Vol. 5, s. 481÷490.
- 6. Bellman R.: Introduction to matrix analysis. Mcgraw-Hill Book Company, New York 1960.
- 7. Brammer R. F.: Controllability in linear autonomous systems with positive controllers. SIAM Journal on Control and Optimization, 1972, Vol. 10 (2), s. 339÷353.
- 8. Budak B. M., Samarski A. A., Tichonow A. N.: Zadania i problemy fizyki matematycznej. PWN, Warszawa 1965.
- Butkowskij A. G.: Charakteristiki sistiem s raspriedieliennymi paramietrami. Sprawocznoje posobie, Nauka, Gławnaja redakcja fizyko-matematiczeskoj literatury, Moskwa 1979.
- Chen C.: Introduction to linear systems theory. Holt, Rinehard and Winston Inc., New York 1970.
- 11. Chen S., Russell D.: A mathematical model for linear elastic systems with structural damping. Quaterly of Applied Mathematics, 1982, Vol. 39, s. 433÷454.
- 12. Curtain R., Zwart H.: An introduction to infinite-dimensional linear systems theory. Springer-Verlag, New York 1995.
- 13. Dahl O. J., Dijkstra E. W., Hoare C. A. R.: Structured programming. Academic Press, London 1972.
- 14. Davison E. J., Wang S. H.: New results on the controllability and observability of general composite systems. IEEE Transactions on Automatic Control, 1975, Vol. 20, s. 123÷128.

15.	Delfour M., Mitter S.: Controllability, observability for infinite dimensional systems.
	SIAM Journal on Control and Optimization, 1972, Vol. 10, s. 329÷333.
16.	Demidowicz B. L., Maron I.A.: Metody numeryczne. PWN, Warszawa 1965, T. 1.
17.	Demidowicz B. L., Maron I.A., Szuwałowa E.J.: Metody numeryczne. PWN,
	Warszawa 1965, T. 2.
18.	Dijkstra E. W.: Umiejętność programowania. WNT, Warszawa 1985.
19.	El-Mikkawy M. E. A.: Explicit inverse of a generalized Vandermonde matrix. Applied
	Mathematics and Computation, 2003, Vol. 146, s. 643÷651.
20.	Evans L. C.: Partial differential equations. American Mathematical Society, Graduate
	Studies in Mathematics, Vol. 19, Rhode Island 2010.
21.	Fattorini H.: Some remarks on complete controllability. SIAM Journal on Control and
	Optimization, 1966, Vol. 4, s. 686÷694.
22.	Fattorini H.: On complete controllability of linear systems. Journal of Differential
	Equations, 1967, Vol. 3, s. 391÷402.
23.	Fichtencholtz M.: Kurs differiencialnowo i integralnowo iscislenia. Nauka, Gławnaja
	redakcja fizyko-matematiczeskoj literatury, Moskwa 1979.
24.	Fortuna Z., Macukow B., Wąsowski J.: Metody numeryczne. WNT,
	Wyd. 4, Warszawa 1998.
25.	Fuji N.: Feedback stabilization of distributed parameter systems by a functional observer.
	SIAM Journal on Control and Optimization, 1980, Vol. 18, s. 108÷120.
26.	Gerald C. F.: Applied numerical analysis. Addison-Wesley, New York 1973.
27.	Górecki H.: Optymalizacja systemów dynamicznych. WNT, Warszawa 1986.
28.	Harel D.: Rzecz o istocie informatyki. Algorytmika. WNT, Warszawa 1992.
29.	Hernando A., Ledesma L., Laita L.: Showing the non-existence of solutions in systems of
	linear diophantine equations. Mathematics and Computers in Simulation, 2009,
	Vol. 79 (11), s. 3211÷3220.
30.	Hildebrand F. B.: Introduction to numerical analysis. McGraw-Hill, New York 1956.
31.	Hou S., Pang W.: Inversion of confluent Vandermonde matrices. An International Journal
	of Computers & Mathematics with Applications, 2002, Vol. 43, s. 1539÷1547.
32.	Huang F.: On the mathematical model for linear elastic systems with analytic damping.
	SIAM Journal on Control and Optimization, 1988, Vol. 26, s. 714÷724.
33.	Ilin W, Kuzniecow J.: Triechdiagonalnyje matricy i ich priłożenia. Izdatielstwo Nauka,
	Moskwa 1985.
34.	Ito K., Kunimatsu N.: Semigroup model of structurally damped beam with boundary
	input. International Journal of Control, 1991, Vol. 54, s. 367÷391.
35.	Ito K., Kunimatsu N.: Stabilization of non-linear distribuded parameter vibratory system.
	International Journal of Control, 1988, Vol. 48, s. 2389÷2415.

- Jacob B., Partington J. R.: On controllability of diagonal systems with one-dimensional input space. Systems & Control Letters, 2006, Vol. 55, s. 321÷328.
- Jankowski M., Woźniakowski H.: O złożoności obliczeniowej w analizie numerycznej. Matematyka Stosowana, 1975, Vol. 5, s. 5÷27.
- 38. Kaczorek T.: Teoria sterowania i systemów. PWN, Wyd. 2, Warszawa 1996.
- 39. Kaczorek T.: Wektory i macierze w automatyce i elektrotechnice. WNT, Warszawa 1998.
- 40. Kalman R. E.: On the general theory of control systems. Proc. I IFAC Congress, London 1960, s. 481÷493.
- 41. Kalman R. E, Ho Y. C., Narendra K. S.: Controllability of linear dynamical systems. Contributions to Differential Equations, 1963, Vol. 1 (2), s. 189÷213.
- 42. Kącki E.: Problemy optymalnego sterowania systemami o rozłożonych parametrach. PWN, Warszawa 1991.
- 43. Kleiber M., Burczyński T.: Badania bazujące na symulacji komputerowej; symulacja komputerowa jako ważny element współczesnej metodologii prowadzenia badań naukowych. Raport, Komitet Informatyki PAN, Sekcja Nauk Obliczeniowych. Warszawa 2010.
- 44. Klamka J.: Absolute controllability of linear systems with time-variable delays in control. International Journal of Control, 1977, Vol. 26 (1), s. 57÷63.
- 45. Klamka J.: Controllability of linear systems with time-variable delays in control. International Journal of Control, 1976, Vol. 24 (6), s. 869÷878.
- 46. Klamka J.: Controllability of dynamical systems. Kluwer Academic Publishers, Dordrecht 1991.
- 47. Klamka J.: Controllability, observability and stabilizability of distributed systems with infinite domain. Proceedings of the 3rd IFAC Symposium on Control of Distributed Parameter Systems, Toulouse 7-11 VI 1982.
- 48. Klamka J.: Schauder's fixed point theorem in nonlinear controllability problems. Control and Cybernetics, 2000, Vol. 29, s. 1377÷1393.
- 49. Klamka J., Pawełczyk M., Wyrwał J.: Numerical methods. Wydawnictwo Politechniki Śląskiej, Gliwice 2001.
- 50. Knabner P., Angerman L.: Numerical methods for elliptic and parabolic partial differential equations. Springer, Texts in Applied Mathematics 44, New York 2003.
- 51. Knuth D.: The art of computer programming. Vol. 2: Seminumerical Algorithms. Addison-Wesley, New York 1973.
- 52. Kobayashi T.: Some remarks on controllability and observability of flexible beam model. International Journal of Systems Science, 1992, Vol. 23, s. 2349÷2357.
- 53. Koblitz N.: Wykład z teorii liczb i kryptografii. WNT, Warszawa 1995.

- Korbicz J., Gawlowicz P.: Sensors location problem for stochastic non-linear discrete-time distributed parameter systems. Control of distributed parameter systems. IFAC Symposia series. 1990, Pergamon Press. N 3, s. 479÷484.
- 55. Kudrewicz J.: Analiza funkcjonalna dla automatyków i elektroników. PWN, Warszawa 1976.
- Lasiecka I., Triggiani R.: Exact controllability with controls in Dirichlet and Neumann boundary conditions: a non conservative code. SIAM Journal Control and Optimization, 1989, Vol. 19, s. 263÷290.
- 57. Lipski W.: Kombinatoryka dla programistów. Wyd. 2, PWN, Warszawa 1989.
- Mahmudov N. I., Zorlu S.: Controllability of semilinear stochastic systems. International Journal of Control, 2005, Vol. 78 (13), s. 997÷1004.
- 59. Manber U.: Introduction to algorithms. A creative approach. Addison-Wesley, New York 1989.
- 60. Manin Yu. I., Panchishkin A. A.: Introduction to modern number theory fundamental problems, ideas and theories. Second Edition, Springer-Verlag, Berlin 2004.
- 61. Manna Z.: Mathematical theory of computation. McGraw-Hill, New York 1974.
- 62. Marzantowicz W., Zakrzycki P.: Elementy teorii liczb. Wydawnictwo Naukowe UAM, Poznań 1999.
- McGlothin G.: Controllability, observability and duality in a distributed parameter systems with continuous and point spectrum. IEEE Transactions on Automatic Control, 1978, Vol. 23, s. 687÷690.
- 64. Miller L.: Non-structural controllability of linear elastic systems with structural damping. Journal of Functional Analysis, 2006, Vol. 236 (2), s. 592÷608.
- Mitkowski W.: Feedback stabilization of second order evolution equations with damping by discrete-time input-output data. Proc. IMACS-IFAC Symp. Model. Simul. Contr. Lumped and Distributed Parameter Systems, Lille 1986, France, s. 355÷358.
- 66. Mitkowski W.: Stabilizacja systemów dynamicznych. WNT, Warszawa 1991.
- 67. Mitkowski W.: Dynamical properties of Metzler systems. Bulletin of the Polish Academy of Sciences. Technical Sciences, 2008, Vol. 56 (4), s. 309÷312.
- 68. Mizohata S.: The theory of partial differential equations. Cambride University Press, Cambride 1973.
- 69. Morton K. W., Mayers D. F.: Numerical solution of partial differential equations:an introduction. Cambridge University Press, Cambridge 1996.
- 70. Moser L.: An introduction to the theory of numbers. Trillia Group, West Lafayette, Indiana 2004.
- 71. Mostowski A., Stark M.: Elementy algebry wyższej. PWN, Warszawa 1972.

- 72. Narkiewicz W.: Teoria liczb. Biblioteka Matematyczna, PWN, Wyd. 2, Warszawa 1990, T. 50.
- 73. Nathanson M. B.: Elementary methods in number theory. Graduate Texts in Mathematics, Springer, New York 2000.
- 74. Nievergelt J., Farrar J.C., Reingold E.M.: Informatyczne rozwiązywanie zadań matematycznych. WNT, Warszawa 1978.
- 75. Paulo R.: Wielkie Twierdzenie Fermata dla laików. WNT, Warszawa 2001.
- Phillips G. M.: Theory and applications of numerical analysis. Academic Press, New York 1973.
- 77. Press W. H., Teukolsky S. A., Vettering T. W., Flannery B. P.: Numerical recipes in C. Cambridge University Press, Cambridge 1992.
- 78. Ralston A., Wilf H. S.: Mathematical metods for digital computers. John Wiley, New York 1960.
- 79. Reid J. K.: Software for numerical mathematics. Ed. Evans, Academic Press, London 1974, s. 29÷47.
- 80. Renardy M., Rogers R.: A first graduate course in partial differential equations. Texts in Applied Mathematics, Springer, New York 1993.
- Respondek J.: Numerical simulation in the partial differential equations controllability analysis with physically meaningful constraints. Mathematics and Computers in Simulation, 2010, Vol. 81 (1), s. 120÷132.
- 82. Respondek J.: On the confluent Vandermonde matrices calculation algorithm. Applied Mathematics Letters, 2011, Vol. 24, s. 103÷106.
- Respondek J.: Numerical approach to the non-linear diofantic equations with applications to the controllability of infinite dimensional dynamical systems. International Journal of Control, 2005, Vol. 78 (13), s. 1017÷1030.
- Respondek J.: Numerical analysis of controllability of diffusive-convective system with limited manipulating variables. International Communications in Heat and Mass Transfer, 2007, Vol. 34 (8), s. 934÷944.
- Respondek J.: Controllability of dynamical systems with constraints. Systems & Control Letters, 2005, Vol. 54 (4), s. 293÷314.
- Respondek J.: Approximate controllability of infinite dimensional systems of the n-th order. International Journal of Applied Mathematics and Computer Science, 2008, Vol. 18 (2), s. 199÷212.
- Respondek J.: Approximate controllability of the n-th order infinite dimensional systems with controls delayed by the control devices. International Journal of Systems Science, 2008, Vol. 39 (8), s. 765÷782.

- Respondek J.: Coupled algorithm approach for the certain partial differential equations properties research. 2nd European Seminar on Coupled Problems, Pilsen, Czech Republic 2010, s. 76.
- 89. Respondek J.: The applications of the decomposition of the real numbers to the linear combination of three natural number's squares to the investigations of the controllability of an infinite dimensional system in the three dimensional space domain. Archiwum Informatyki PAN, 2003, Vol. 15 (2), s. 143÷167.
- 90. Respondek J.: Semidiscretization methods in certain controllability measures of a parabolic PDE system. 13th WSEAS International Conference on Applied Mathematics, Puerto De La Cruz, Canary Islands 2008, s. 144÷147.
- Respondek J.: Parallel implementation of algorithm for solving certain nonlinear diofantic equations. V Konferencja computer methods and systems, Kraków 2005, Vol. 2, Oprogramowanie Naukowo-Techniczne, s. 389÷392.
- Respondek J.: Numerical approach to the null controllability of parabolic systems with compact constraints. VI konferencja computer methods and systems, Kraków 2007, s. 295÷300.
- Respondek J.: Dynamic task dividing in the parallel implementation of algorithm for solving certain nonlinear diofantic equations. VI konferencja computer methods and systems, Kraków 2007, s. 345÷348.
- Respondek J.: Task dividing method in diophantine equation parallel Solving. Polish Journal of Environmental Studies, Proceedings of the Congress of Young IT Scientists, Świnoujście 2007, Vol. 16 (4A), s. 264÷267.
- 95. Respondek J.: Parallel algorithm designing for the simulation of the properties of the distributed parameters systems. Polish Journal of Environmental Studies, Proceedings of the Congress of Young IT Scientists, Świnoujście 2008, Vol. 17 (3B), s. 370÷374.
- 96. Respondek J.: Numerical analysis of controllability of a parabolic system with delayed controls and non zero boundary condition. 7th International Conference On Computer Science-Research And Applications, Kazimierz Dolny 2008, Proc. in: Annales Informatica, Lublin 2008, Vol. 8 (2), s. 119÷128.
- 97. Respondek J.: Algorithm for certain control problems arising in a hyperbolic type PDE. 2nd International Conference on Information Technology, Gdańsk 2010, Proc. in: ZN Politechniki Gdańskiej, Gdańsk 2010, Information Technology, s. 503÷507.
- 98. Respondek J.: Numerical algorithm for the controllability level of an elliptic distributed parameters system. 29th IASTED International Conference on Modelling, Identification, and Control, Innsbruck, Austria 2010, s. 376÷380.

- Respondek J.: Równoległy algorytm znajdowania par liczb całkowitych generujących równe sumy kwadratów w kombinacji liniowej. XI Konferencja Sieci Komputerowe, Zakopane 2004, Współczesne problemy sieci komputerowych – Zastosowanie i bezpieczeństwo, WNT, Warszawa 2004, s. 309÷316.
- Respondek J.: Zastosowanie protokołu TCP/IP w algorytmie rozwiązywania wybranej klasy nieliniowych równań diofantycznych. XIII Konferencja Sieci Komputerowe, Zakopane 2006, Tom 1-Nowe Technologie, WKŁ, Warszawa 2006, s. 243÷250.
- 101. Respondek J.: The effective algorithm for solving the quadratic diofantic equation with three unknowns. Konferencja Informatyka-Badania i Zastosowania. Kazimierz Dolny 2005, Proc. in: Annales UMCS Informatica, Vol. 3, Lublin 2005, s. 57÷64.
- 102. Respondek J.: A parallel implementation in the investigations of the controllability of an infinite dimensional system in the three dimensional rectangular prism. Konferencja Informatyka-Badania i Zastosowanie, Kazimierz Dolny 2005. Proc. in: Annales UMCS Informatica, Vol. 6, Lublin 2007, s.15÷22.
- 103. Respondek J.: Znajdowanie par liczb całkowitych generujących równe sumy kwadratów w kombinacji liniowej. ZN Pol. Śl. Studia Informatica, Vol. 24, No. 4 (56), s. 85÷100, Gliwice 2003.
- 104. Respondek J.: Problemy numeryczne związane z badaniem sterowalności układu parabolicznego w obszarze prostokątnym. ZN Pol. Śl. Studia Informatica, Vol. 24, No. 4 (56), s. 119÷132, Gliwice 2003.
- 105. Respondek J.: Porównanie efektywności algorytmów badania krotności wartości własnych operatora Laplace'a określonego w prostokącie. XXXIV Ogólnopolska Konferencja Zastosowań Matematyki, Zakopane 2005, s. 74.
- 106. Respondek J.: Metody dyskretyzacji częściowej w analizie wybranych własności układów o parametrach rozłożonych, XXXVII Ogólnopolska Konferencja Zastosowań Matematyki, Zakopane 2008, s. 63.
- 107. Respondek J.: Synteza oraz optymalizacja algorytmów odwracania uogólnionych macierzy Vandermonde'a. XXXIX Ogólnopolska Konferencja Zastosowań Matematyki, Zakopane 2010, s. 60.
- 108. Respondek J.: Numeryczne algorytmy rozwiązywania wybranych klas równań diofantycznych. X Środowiskowa Konferencja Matematyczno-Informatyczna, Materiały Konferencyjne, Korytnica 2004, s. 50.
- 109. Richter J.: Advanced Windows. Microsoft Press, Wyd. 3, Redmont 2001.
- 110. Rosen K. H., Michaels J. G., Gross J. L., Grossman J. W., Shier D. R.: Handbook of discrete and combinatorial mathematics. CRC Press, Londyn 2000.
- 111. Sakawa Y.: Controllability for partial differential equations of parabolic type. SIAM Journal on Control and Optimization, 1974, Vol. 12, s. 389÷400.

112.	Sakawa Y .: Feedback control of second order evolution equations with damping. SIAM
	Journal Control and Optimization, 1984, Vol. 22, s. 343÷361.
113.	Sakawa Y, Matsuno F.: Modelling and control of flexible manipulator with a parallel drive
	mechanism. International Journal of Control, 1986, Vol. 44, s. 299÷313.
114.	Sandor J.: Geometric theorems, diophantine equations and arithmetic functions. American
	Research Press, Rehoboth 2002.
115.	Schiesser W.E.: The numerical method of lines integration of partial differential
	equations. Academic Press, San Diego 1991.
116.	Schmitendorf W., Barmish B.: Controlling a constrained linear system to an affinite target.
	IEEE Transactions on Automatic Control, 1981, Vol. 26, s. 761÷763.
117.	Schmitendorf W., Barmish B.: Null controllability of linear systems with constrained
	controls. SIAM Journal Control and Optimization, 1980, Vol. 18 (4), s. 327÷345.
118.	Shi D. H, Hou S. H, Feng D .: Feedback stabilization of a Timoshenko beam with an end
	mass. International Journal of Control, 1998, Vol. 69 (2), s. 285÷300.
119.	Shi D. H., Feng D., Yan Q.: Feedback stabilization of rotating Timoshenko beam with
	adaptive gain. International Journal of Control, 2001, Vol. 74 (3), s. 239÷251.
120.	Shubov M. A.: Exact controllability of damped Timoshenko beam. IMA Journal of
	Mathematical Control and Information, 2000, Vol. 17, s. 375÷395.
121.	Shubov M. A.: Spectral operators generated by Timoshenko beam model. Systems &
	Control Letters, 1999, Vol. 38, s. 249÷258.
122.	Sierpiński W.: 250 zadań z elementarnej teorii liczb. Wydawnictwa Szkolne
	i Pedagogiczne, Warszawa 1987.
123.	Sierpiński W.: Arytmetyka teoretyczna. PWN, Warszawa 1968.
124.	Sierpiński W.: O stu prostych, ale trudnych zagadnieniach arytmetyki. PZWS,
	Warszawa 1959.
125.	Sierpiński W.: Wstęp do teorii liczb. Państwowe Zakłady Wydawnictw Szkolnych,
	Warszawa 1965.
126.	Sneddon N.: Równania różniczkowe cząstkowe. PWN, Warszawa 1962.
127.	Sontag E. D.: Mathematical control theory. Springer Verlag, New York 1990.
128.	Stoer J., Bulirsch R.: Introduction to numerical analysis. Springer-Verlag, NewYork 1980.
129.	Tichonow A. N., Samarski A. A.: Równania fizyki matematycznej. PWN,
	Warszawa 1963.
130.	Triggiani R.: Controllability and observability in Banach spaces with bounded operators.
	SIAM Journal on Control and Optimization, 1975, Vol. 13, s. 462÷491.
131.	Triggiani R.: Extensions of rank conditions for controllability and observability to Banach
	spaces with unbounded operators. SIAM Journal on Control and Optimization, 1976,
	Vol. 14, s. 313÷338.

- 132. Vargra E. I., Hangos K. M., Szigeti F.: Controllability and observability of heat exchangers networks in the time varying case. Control Engineering Practice, 1995, Vol. 3 (10), s. 1409÷1419.
- 133. Wilkinson J. H.: Błędy zaokrągleń w procesach algebraicznych. PWN, Warszawa 1967.
- 134. Wirth N.: Algorytmy + struktury danych = programy. WNT, Warszawa 1989.
- 135. Wirth N.: Programming development by stepwise refinement. Communications of the ACM, 1971, Vol. 14, s. 221÷227.
- 136. Wirth N.: Wstęp do programowania systematycznego. WNT, Warszawa 1978.
- 137. Wolska-Bochenek J., Borzymowski A., Chmaj J., Tryjarska M.: Zarys teorii równań całkowych i równań różniczkowych cząstkowych. PWN, Warszawa 1981.
- 138. www.ieee.com
- 139. Xu G. Q.: Boundary feedback exponential stabilization of a Timoshenko beam with both ends free. International Journal of Control, 2005, Vol. 78 (4) 10, s. 286÷297.
- 140. Yan S. Y.: Number theory for computing. Second Edition, Springer-Verlag, Berlin 2002.
- 141. Zadeh L., Desoer C.: Linear system theory. the state space approach. McGraw-Hill, New York 1963.

METODY ANALIZY NUMERYCZNEJ W BADANIACH ZBIORÓW OSIĄGALNYCH UKŁADÓW DYNAMICZNYCH

STRESZCZENIE

Przedmiotem rozprawy jest wprowadzenie metod numerycznych do zagadnień dotyczących jakościowych własności układów dynamicznych, które dotychczas były przedmiotem rozważań jedynie na drodze czysto analitycznej.

W rozprawie wykorzystano zarówno klasyczne metody numeryczne znane z literatury (jak metoda prostych wykorzystana do dyskretyzacji równania różniczkowego cząstkowego w rozdziale 7), jak i wprowadzono zupełnie nowe algorytmy służące numerycznemu rozwiązaniu wybranej klasy równań diofantycznych, obliczaniu wielomianów symetrycznych podstawowych oraz odwracaniu konfluentnych macierzy Vandermonde'a.

Rozdział 1 jest wprowadzeniem do rozprawy. Przedstawiono w nim opracowania Komitetu Informatyki PAN, określające analizę numeryczną jako dyscyplinę w dziedzinie informatyki. Następnie sformułowano cel rozprawy. Rozdział kończy przegląd problemów niniejszej rozprawy.

Rozdział 2 jest poświęcony znalezieniu algorytmu rozwiązującego pewną klasę równań diofantycznych. Równanie będące przedmiotem analizy modeluje problem znajdywania n-tek liczb naturalnych, generujących równe sumy kwadratów liczb całkowitych w kombinacji liniowej o dodatnich współczynnikach. Podzbiór liczb naturalnych, w którym poszukujemy rozwiązań równania, jest ograniczony odgórnie przez daną stałą. Na początku rozdziału 2 przypomniano najważniejsze z pojęć analizy algorytmów (złożoność obliczeniowa) oraz teorii błędów (błędy zaokrągleń) koniecznych do analizy własności uzyskanego w rozdziałe algorytmu, rozwiązującego rozważane równanie diofantyczne. Zasadniczą częścią rozdziału jest poszukiwanie oraz optymalizacja odpowiedniego algorytmu. Ważną częścią rozdziału jest eliminacja rozwiązań symetrycznych równania diofantycznego, tj. takich, w których występuje ta sama para n-tek, ale zamienionych stronami. Rozważania dopełnia treść algorytmu w pseudokodzie, wyznaczenie jego złożoności czasowej oraz analiza błędów obliczeń. Oprócz tego w rozdziałe pokazano, jak wprowadzony algorytm wykorzystać do określenia krotności n-tek, generujących równe sumy kwadratów w kombinacji liniowej. Znalazło to zastosowanie w kolejnych rozdziałach rozprawy. Działanie algorytmu pokazano

na przykładzie liczbowym, przedstawiając w tabeli wszystkie kolejne wartości zmiennych roboczych algorytmu oraz odpowiadających im rozwiązań. Rozdział kończy praktyczny test efektywności obliczeniowej algorytmu, przedstawiony w postaci wykresu obrazującego czasy jego wykonania w funkcji górnego ograniczenia dziedziny równania.

W rozdziale 3 przedstawiono zastosowanie algorytmu rozwiązywania równania diofantycznego do analizy zbiorów osiagalnych układu dynamicznego o parametrach rozłożonych, danego równaniem różniczkowym cząstkowym typu parabolicznego. Rozważono zerowe warunki brzegowe typu Dirichleta oraz dziedzinę równania w postaci n-wymiarowego wielościanu. Jak wykazano, wartości własne operatora różniczkowego Laplace'a, występującego w badanym równaniu, są proporcjonalne do stron rozwiązywanego w rozdziale 2 równania diofantycznego. Dzięki temu jest możliwe wyznaczenie ich krotności na podstawie algorytmu z rozdziału 2. W rozdziale 3 przedstawiono model matematyczny rozważanego układu parabolicznego, zdefiniowano odpowiednie operatory: różniczkowy stanu oraz macierzowy wejścia i wykonano dekompozycje spektralna układu. W ten sposób wyjściowy układ nieskończenie wymiarowy zastąpiono równoważnym nieskończonym ciągiem układów skończenie wymiarowych, który stanowił punkt wyjścia do dalszych badań. Następnie zbudowano algorytm numeryczny, służący badaniu zbiorów osiągalnych rozważanego układu dynamicznego. Wykorzystano w nim algorytm rozwiązywania równań diofantycznych z rozdziału 2. Na zakończenie wyznaczono zbiory osiagalne dla konkretnego przykładu układu dynamicznego.

Rozdział 4 jest poświęcony zbudowaniu efektywnego algorytmu obliczania wielomianów symetrycznych podstawowych. Wielomiany te pozwalają na obliczenie współczynników występujących przy kolejnych potęgach argumentu wielomianu. Ponadto, znajdują one zastosowanie w konstrukcji algorytmu odwracania konfluentnych macierzy Vandermonde'a.

W rozdziale 5 skonstruowano algorytm odwracania konfluentnych macierzy Vandermonde'a. Ich budowa różni się od klasycznej postaci macierzy Vandermonde'a tym, że w kolumnach konfluentnej macierzy Vandermonde'a, oprócz kolejnych potęg różnych pierwiastków, znajdują się pochodne tychże kolumn. W konstrukcji algorytmu korzysta się z przedstawionego w poprzednim rozdziale algorytmu obliczania wielomianów symetrycznych podstawowych. Sam algorytm znajduje zastosowanie w analizie własności liniowych układów dynamicznych o dowolnym stopniu pochodnych względem czasu, czemu poświęcono kolejny rozdział.

W rozdziale 6 zastosowano algorytm obliczania wielomianów symetrycznych podstawowych oraz algorytm odwracania uogólnionych macierzy Vandermonde'a w analizie wybranych własności układów dynamicznych. Jako badaną własność obrano kilka rodzajów sterowalności. Główną innowacją rozdziału jest przeprowadzenie badań dla dowolnego, n-tego stopnia badanego układu dynamicznego względem czasu. Ponadto, udowodnione

kryteria stosuje się do najogólniejszej postaci rozważanego układu, o dowolnej krotności każdej z wartości własnych jego równania charakterystycznego. Uzyskanie tak ogólnych wyników było możliwe dzięki zastosowaniu wyżej wymienionych algorytmów. Na koniec rozdziału zastosowano uzyskane w rozdziale wyniki do analizy własności elastycznej belki.

Rozdział 7 zajmuje się numerycznym wyznaczaniem zbiorów osiągalnych układów dynamicznych, modelowanych za pomocą równań różniczkowych cząstkowych typu parabolicznego. Główną innowacją przeprowadzonych badań jest założonie realistycznych, obustronnych (tj. odgórnych i oddolnych) ograniczeń funkcji wymuszającej. Najpierw zdyskretyzowano analizowane równanie różniczkowe cząstkowe za pomocą metody numerycznej prostych. Następnie wyznaczono funkcję podporową układu oraz spektrum macierzy stanu oraz zastosowano do nich odpowiednie kryterium wyznaczania zbiorów osiągalnych przy ograniczonej funkcji wymuszającej. Uzyskany wynik zilustrowano kilkoma wykresami dla dwóch konkretnych postaci funkcji wymuszających. Należy podkreślić, że główna innowacja rozdziału, tj. wykonane badania zbiorów osiągalnych przy realistycznych ograniczeniach na funkcję wymuszającą, była możliwa dzięki zastosowaniu odpowiednio dobranej metody numerycznej rozwiązywania równań różniczkowych cząstkowych. Rozdział stanowi ilustrację połączenia w całość metod numerycznych i klasycznych wyników analitycznych, prowadzących do nowych, bardziej realistycznych wyników w nauce.

Rozdział 8 stanowi podsumowanie rezultatów przedstawionych w niniejszej rozprawie.

NUMERICAL ANALYSIS IN THE DYNAMICAL SYSTEM ATTAINABLE SETS RESEARCH

ABSTRACT

The objective of this book is introducing numerical methods into issues pertaining the qualitative dynamical systems properties, which so far have been analyzed only by purely analytical methods.

In the book we used classical numerical methods known from literature (i.e. the line method used for the discretization of the partial differential equation in chapter 7) and introduced new algorithms, designed for numerical solving of a selected class of the diophantine equations, elementary symmetrical polynomials calculation and the confluent Vandermonde matrices inversion calculating.

Chapter 1 contains the introduction to the book. It features the works of the Computer Science Committee of the Polish Academy of Sciences defining the numerical analysis as a part of the computer science discipline. Next, the objective of the book is formulated. The chapter is finalized by the survey of the issues of the book.

Chapter 2 is devoted to finding an algorithm for solving a certain class of diophantine equations. The analyzed equation models the problem of finding the n-ths which generate equal sums of the squares of the integer numbers in the linear combination with positive coefficients. The subset of natural numbers, in which we are looking for solutions, is constrained to a given maximum number. At the beginning of chapter 2 we referred to the basic notions of the algorithms analysis (computational complexity) and the errors theory (rounding errors) necessary in the analysis of equation properties presented in the chapter, solving the considered diophantine equation. The main part of the chapter is searching for and optimization of the proper algorithm. A separate item is devoted to the elimination of symmetrical solutions, i.e. solutions with the same couple of n-ths but with swapped sides. Those calculations are followed by the code of the algorithm, calculation complexity determination and computational errors analysis. Additionally, in this chapter we showed how the application of the presented algorithm is used to calculate the multiplicities of the

n-ths which generate equal sums of the squares in the linear combination. We showed the operation of the algorithm on a particular example, i.e. placing in the table all values of the loops working variables and solutions in the sequence corresponding with those variables. The chapter is completed by the computational effectiveness test presented in the form of a graph. The graph shows the times of the algorithm execution in the dependency of the upper constraint of the equation domain.

In chapter 3 we presented the application of the diophantine equation solving algorithm in the attainable sets analysis of a distributed parameter dynamical system, described by a parabolic-type partial differential equation. We considered the zero Dirichlet-type boundary condition and n-dimensional rectangular prism equation domain. We showed that the eigenvalues of the Laplace differential operator, existing in the analyzed equation, are proportional to the sides of the diophantine equation, solved in chapter 2. This way, one can calculate the multiplicities with the use of the algorithm presented in chapter 2. In chapter 3 we presented a mathematical model of the considered parabolic system, defined the proper state differential operator and matrix input operator, and performed the spectral decomposition of the system. Thus we have the infinite dimensional system in the corresponding form of the infinite series of finite dimensional systems, convenient for further analysis. Next, we constructed the numerical algorithm for the attainable sets analysis of the dynamical system in question. In that algorithm we made use of the algorithm presented in chapter 2. Finally, we calculated the attainable sets for the particular dynamical system.

Chapter 4 devoted to building the effective elementary symmetrical polynomials calculation algorithm building. Those polynomials enable to calculate the subsequent powers polynomial argument coefficients. Moreover, they are used in the construction of the confluent Vandermonde matrix inverse calculation algorithm.

In chapter 5 we constructed the confluent Vandermonde matrices inverse calculation algorithm. Their structure differs from the classical Vandermonde matrix, as in the columns of the confluent Vandermonde matrix, apart from the subsequent powers of the different roots, there are also their derivatives. In the construction of the algorithm we made use of the elementary symmetrical polynomials calculation algorithm, presented in the previous chapter. The constructed algorithm can be applied in the analysis of the arbitrary order with respect to time derivatives dynamical system. Next chapter is devoted to that issue.

In chapter 6 we applied the elementary symmetrical polynomials calculation algorithm as well as the confluent Vandermonde matrix inverse calculation algorithm to the research of the selected dynamical systems properties. For the investigated property we chose a few kinds of controllability. The main innovation of the chapter is that the research results hold true for the arbitrary order with respect to time derivatives. Moreover, the given criteria pertain to the most general form of the analyzed system, with arbitrary characteristic equation eigenvalues multiplicities. Deriving such general results was possible thanks to the above described algorithms. At the end of the chapter we showed how to use the obtained results in the elastic beam properties analysis.

In chapter 7 we performed a numerical analysis of the dynamical systems attainable sets, modeled by the parabolic-type partial differential equations. The main innovation of the performed research is taking into account realistic, both-side (i.e. upper and lower) constraints of the excitation function. First, we discretized the analyzed partial differential equation by means of the line numerical method. Next, the support function and the spectrum of the state matrix were derived and the attainable sets determination criterion with constrained excitation function was applied. The received outcome was illustrated by two graphs for two particular cases of the excitation function. It should be pointed out that the main innovation of the chapter, i.e. performed attainable sets research considering realistic excitation function constraints, was received thanks to the use of a precisely selected numerical method of the partial differential equations solving. The chapter can be an illustration for combining the numerical methods and classical analytical results, leading to new, more realistic outcomes in science.

Chapter 8 is the summary of the book results.

INFORMATION FOR AUTHORS

The journal *STUDIA INFORMATICA* publishes both fundamental and applied Memoirs and Notes in the field of informatics. The Editors' aim is to provide an active forum for disseminating the original results of theoretical research and applications practice of informatics understood as a discipline focused on the investigations of laws that rule processes of coding, storing, processing, and transferring of information or data.

Papers are welcome from fields of informatics inclusive of, but not restricted to Computer Science, Engineering, and Life and Physical Sciences.

All manuscripts submitted for publication will be subject to critical review. Acceptability will be judged according to the paper's contribution to the art and science of informatics.

In the first instance, all text should be submitted as hardcopy, conventionally mailed, and for accepted paper accompanying with the electronically readable manuscript to:

Dr. Marcin SKOWRONEK Institute of Informatics Silesian University of Technology ul. Akademicka 16 44-100 Gliwice, Poland Tel.: +48 32 237-12-15 Fax: +48 32 237-27-33

e-mail: marcin.skowronek@polsl.pl

MANUSCRIPT REQUIREMENTS

All manuscripts should be written in Polish or in English. Manuscript should be typed on one side paper only, and submitted in duplicate. The name and affiliation of each author should be followed by the title of the paper (as brief as possible). An abstract of not more than 50 words is required. The text should be logically divided under numbered headings and subheadings (up to four levels). Each table must have a title and should be cited in the text. Each figure should have a caption and have to be cited in the text. References should be cited with a number in square brackets that corresponds to a proper number in the reference list. The accuracy of the references is the author's responsibility. Abbreviations should be used sparingly and given in full at first mention (e.g. "Central Processing Unit (CPU)"). In case when the manuscript is provided in Polish (English) language, the summary and additional abstract (up to 300 words with reference to the equations, tables and figures) in English (Polish) should be added.

After the paper has been reviewed and accepted for publication, the author has to submit to the Editor a hardcopy and electronic version of the manuscript.

It is strongly recommended to submit the manuscript in a form downloadable from web site http://zti.polsl.pl/makiety/.

To subscribe: *STUDIA INFORMATICA* (PL ISSN 0208-7286) is published by Silesian University of Technology Press (Wydawnictwo Politechniki Śląskiej) ul. Akademicka 5, 44-100 Gliwice, Poland, Tel./Fax +48 32 237-13-81. 2011 annual subscription rate: US\$60. Single number price approx. US\$10-20 according to the issue volume.

INSTYTUT INFORMATYKI prowadzi:

- □ Studia stacjonarne I stopnia (inżynierskie)
- □ Studia stacjonarne II stopnia (magisterskie)
- □ Studia niestacjonarne I stopnia (inżynierskie)
- □ Studia niestacjonarne II stopnia (magisterskie)
- □ Studia podyplomowe:
 - Sieci i systemy komputerowe, bazy danych
 - Systemy informacji geograficznej
 - Teleinformatyka w transporcie lotniczym
 - *Technologie internetowe i technologie mobilne*
 - Metody eksploracji baz danych przedsiębiorstw
- □ Studia doktoranckie

Informacje:

POLITECHNIKA ŚLĄSKA Instytut Informatyki

44-100 Gliwice, ul. Akademicka 16 tel. (032) 237 24 05; 237 21 51; fax (032) 237 27 33 (czynny całą dobę)

e-mail: rau2@polsl.pl

http://www.inf.polsl.pl (dydaktyka)