POLITECHNIKA BIAŁOSTOCKA
WYDZIAŁ ELEKTRYCZNY

ROZPRAWA DOKTORSKA

mgr inż. Jarosław Forenc

Analiza spekulacyjna stanów nieustalonych w układach elektrycznych

Promotor:
prof. zw. dr hab. inż. Andrzej Jordan

BIAŁYSTOK 2006

Dziękuję Prof. Andrzejowi Jordanowi
za pomoc i cenne uwagi
podczas pisania tej pracy

Spis treści

1. Wstęp 6

2. Modelowanie i analiza stanów nieustalonych układów elektrycznych opisanych równaniami różniczkowymi zwyczajnymi 8

2.1. Wstęp 8
2.2. Metoda klasyczna 10
2.3. Metoda operatorowa 12
2.4. Metoda zmiennych stanu 14
2.5. Podsumowanie 18

3. Sekwencyjne metody numeryczne analizy stanów nieustalonych 20

3.1. Wstęp 20
3.2. Metody jednokrokowe 23
3.3. Metody wielokrokowe 29
3.4. Metody predyktor-korektor 33
3.5. Podsumowanie 35

4. Równoległe metody numeryczne analizy stanów nieustalonych 37

4.1. Wstęp 37
4.2. Równoległość układu 41
4.3. Równoległość metody 42
4.4. Równoległość czasu (kroków) 47
4.5. Podsumowanie 48

5. Systemy klaster 49

5.1. Ogólna charakterystyka systemów klaster 49
5.2. Opis systemu klaster KETIM2 51

6. Cele, teza i oryginalne elementy rozprawy 55

6.1. Cele pracy 55
6.2. Teza rozprawy 56
6.3. Oryginalne elementy rozprawy 56

7. Metoda spekulacyjna analizy stanów nieustalonych 58

7.1. Wstęp 58
7.2. Podstawy teoretyczne metody spekulacyjnej 59
7.3. Metoda spekulacyjna ze stałym krokiem całkowania 60
7.4. Zastosowanie metody spekulacyjnej ze stałym krokiem całkowania do analizy dynamiki silnika asynchronicznego pierścieniowego 69
7.4.1. Wstęp 69
7.4.2. Analiza dynamiki silnika asynchronicznego pierścieniowego 69

7.4.3. Zastosowanie metody spekulacyjnej 71
7.4.4. Wyniki obliczeń 74
7.4.5. Wnioski 84
7.5. Metoda spekulacyjna ze zmiennym krokiem całkowania 84
7.6. Zastosowanie metody spekulacyjnej ze zmiennym krokiem całkowania do analizy dynamiki silnika asynchronicznego pierścieniowego 91
7.6.1. Wstęp 91
7.6.2. Zastosowanie metody spekulacyjnej 91
7.6.3. Wyniki obliczeń 93
7.6.4. Wnioski 97

8. Zmodyfikowana metoda spekulacyjna analizy stanów nieustalonych 98

8.1. Wstęp 98
8.2. Algorytm zmodyfikowanej metody spekulacyjnej 98
8.2.1. Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały 99
8.2.2. Wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych 99
8.2.3. Wyznaczenie warunków początkowych na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą 102
8.2.4. Określenie rozwiązania końcowego 107
8.3. Zastosowanie metody spekulacyjnej w linearyzacji globalnej 108
8.3.1. Wstęp 108
8.3.2. Nieliniowy model silnika prądu stałego zasilanego z generatora fotowoltaicznego 109
8.3.3. Linearyzacja globalna równań opisujących silnik 110
8.3.4. Zastosowanie zmodyfikowanej metody spekulacyjnej - wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych 113
8.3.4.1. Wyniki obliczeń 115
8.3.4.2. Wnioski 120
8.3.5. Zastosowanie zmodyfikowanej metody spekulacyjnej - wyznaczenie warunków początkowych na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą 120
8.3.5.1. Wyniki obliczeń 121
8.3.5.2. Wnioski 126
8.4. Zastosowanie zmodyfikowanej metody spekulacyjnej do analizy stanów nieustalonych w linii długiej 126
8.4.1. Wstęp 126
8.4.2. Model linii długiej 127
8.4.3. Zastosowanie zmodyfikowanej metody spekulacyjnej 130
8.4.4. Wyniki obliczeń 131
8.4.5. Wnioski 135

9. Uwagi końcowe i wnioski 136

Literatura 142

1. Wstęp

Stany nieustalone występują w większości układów elektrycznych - w obwodach RLC, układach elektromechanicznych (maszyny elektryczne) oraz układach elektroenergetycznych. Takie stany pracy mogą wynikać z normalnej eksploatacji powyższych układów (np. rozruch silników elektrycznych, zmiana wymuszenia układu lub jego punktu pracy), a także z występowania sytuacji awaryjnych (np. zwarcia w systemach elektroenergetycznych). W wielu układach, zwłaszcza automatyki (regulacji), stan nieustalony jest normalnym stanem pracy. Analiza stanów nieustalonych pełni zatem dużą rolę w elektrotechnice i jest często przedmiotem badań tego działu nauki.

Zależnie od typu układu elektrycznego oraz od rodzaju zjawisk w nim występujących, stany nieustalone opisywane są przy użyciu równań różniczkowych (ciągłe zmiany zmiennych stanu) lub układów równań różnicowych (układy dyskretne). Przedstawione w pracy rozważania będą dotyczyły układów ciągłych opisanych równaniami różniczkowymi zwyczajnymi lub równaniami różniczkowymi cząstkowymi, zastąpionymi równaniami różniczkowymi zwyczajnymi poprzez aproksymację pochodnych cząstkowych za pomocą różnic skończonych. W przypadku analizy stanów nieustalonych w obwodach elektrycznych zazwyczaj stosowana jest jedna z trzech metod: metoda klasyczna, metoda operatorowa lub metoda zmiennych stanu [11], [19], [100]. Analiza ta polega najczęściej na zdefiniowaniu równań różniczkowych lub różniczkowo-całkowych opisujących obwód, wyznaczeniu warunków komutacji, a następnie rozwiązaniu tych równań jedną z powyższych metod. Metoda klasyczna i operatorowa stosowane są do analizy układów liniowych niskiego rzędu. Metoda zmiennych stanu może być stosowana zarówno do układów liniowych jak i nieliniowych. Zastosowanie w metodzie zmiennych stanu algebry macierzowej do rozwiązywania układów liniowych jest praktycznie wykonalne dla układów do trzeciego rzędu. W przypadku układów liniowych wyższych rzędów i układów nieliniowych najczęściej są stosowane metody numeryczne.

Problem rozwiązywania numerycznego równań stanu jest znany od bardzo dawna, toteż istnieją wydajne metody i algorytmy numeryczne przeznaczone do tego celu [9], [20], [90]. W przypadkach, gdy obliczanie prawej strony równania stanu jest kosztowne, liczba równań jest bardzo duża, przedział całkowania jest długi, układ równań wymaga wielokrotnego całkowania lub otrzymanie dużej dokładności obliczeń i zapewnienie ich stabilności wymaga bardzo małego kroku całkowania, wskazanym jest zastosowanie algorytmów równoległych rozwiązywania równań stanu [85], [93].

Istniejące metody numeryczne rozwiązywania równań stanu (metody rozwiązywania tzw. zagadnień początkowych równań różniczkowych zwyczajnych) są to w większości przypadków typowe metody sekwencyjne. Wyznaczenie w nich wartości zmiennych stanu w kolejnym kroku wymaga znajomości wartości zmiennych stanu z jednego lub kilku kroków bezpośrednio poprzedzających. Pomimo tego, iż komputery równoległe i obliczenia równoległe znane są i stosowane od bardzo dawna, to liczba algorytmów równoległych rozwiązywania zagadnień początkowych jest niewielka w porównaniu z np. liczbą algorytmów równoległych algebry liniowej [85].

Istniejące algorytmy równoległe analizy zagadnień początkowych w większości przypadków wymagają komunikacji między procesorami w każdym kroku obliczeniowym. Opłacalne jest zatem ich stosowanie tylko w systemach równoległych posiadających możliwość szybkiej wymiany informacji między procesorami. Od pewnego czasu dużą popularność zdobywają obliczenia równoległe wykonywane w systemach klastrowych. Klastry zazwyczaj nie posiadają bardzo szybkich interfejsów sieciowych. Pojawiła się zatem konieczność opracowania takich metod i algorytmów numerycznych analizy równań stanu, które będzie można wydajnie stosować w systemach klastrowych.

2. Modelowanie i analiza stanów nieustalonych układów elektrycznych opisanych równaniami różniczkowymi zwyczajnymi

2.1. Wstęp

Układy elektryczne zazwyczaj pracują w tzw. *stanie ustalonym*. Taki stan pracy obwodu występuje w chwili czasu nieskończenie odległej od momentu jego powstania [100]. W przypadku układów liniowych w stanie ustalonym, jeśli wymuszenia mają charakter stały, sinusoidalny lub okresowy, to odpowiedzi mają także taki sam charakter [11], [12], [19], [100]. Powyższa definicja stanu ustalonego nie jest na ogół prawdziwa w przypadku układów parametrycznych i nieliniowych. Jeśli w obwodzie w chwili $t = t_0$ (lub $t = 0$) zachodzą zmiany dotyczące parametrów źródeł, wartości elementów obwodu, modyfikacji jego struktury, to w obwodzie występuje zjawisko nazywane *komutacją* [19], [81], [100]. Komutacja stanowi zaburzenie względem stanu obwodu występującego przed czasem $t = t_0$. Po komutacji, napięcia i prądy w obwodzie nie osiągają natychmiast wartości ustalonych, ale dopiero po pewnym czasie. Zjawiska występujące w tym czasie w obwodzie nazywane są *stanem nieustalonym*. Obwód znajduje się w stanie nieustalonym przy przejściu od jednego stanu ustalonego do drugiego [11], [19].

Stan obwodu w chwili, w której następuje komutacja ($t = t_0$ lub $t = 0$), nazywany jest *stanem początkowym*, zaś wartości napięć na elementach i prądów płynących przez elementy obwodu - *warunkami początkowymi*. Jeśli wszystkie powyższe napięcia i prądy mają wartości równe zeru, to warunki początkowe nazywane są *zerowymi*, w przeciwnym przypadku mamy *niezerowe* warunki początkowe.

Występowanie stanu nieustalonego związane jest ze zmianami w czasie energii pola magnetycznego i elektrycznego magazynowanej w elementach *L*, *M* i *C*. Energia ta nie może zmieniać się w sposób skokowy, gdyż moc chwilowa w obwodzie osiągnęłaby wtedy wartość nieskończoną, co jest fizycznie niemożliwe [101]. Wynika

z tego zasada ciągłości w chwili komutacji strumienia ψ skojarzonego z cewką, równoważna zasadzie ciągłości prądu elektrycznego płynącego przez cewkę, oraz zasada ciągłości ładunku elektrycznego kondensatora, równoważna zasadzie ciągłości napięcia na jego zaciskach. Powyższe dwie zasady nazywane są *prawami komutacji* [11]. Prawa te odgrywają dużą rolę w analizie stanów nieustalonych obwodów elektrycznych, gdyż są wykorzystywane do wyznaczania warunków początkowych.

W obwodach zawierających wyłącznie rezystancje i źródła idealne stany nieustalone nie występują, a przejście od jednego stanu ustalonego do drugiego zachodzi natychmiastowo [19], [100].

W analizie stanów nieustalonych w obwodach elektrycznych stosowane są [100]:

1. *Metody numeryczne* - stosowane do analizy układów wysokiego rzędu i/lub układów nieliniowych, umożliwiają efektywne rozwiązywanie wielu problemów dynamiki, lecz otrzymywane wyniki analizy nie mają postaci jawnych wzorów analitycznych.
2. *Metody analityczne* - stosowane do obwodów liniowych niskich rzędów, pozwalają na otrzymanie wyników analizy w postaci jawnych wzorów, co umożliwia badanie wpływu parametrów obwodu na jego dynamikę.

Przeprowadzając analizę stanu nieustalonego zazwyczaj wykonywane są trzy kroki [100]:

1. Określenie równań różniczkowych lub różniczkowo-całkowych opisujących obwód na podstawie praw Kirchhoffa i równań definicyjnych elementów *R*, *L*, *M* i *C*.
2. Określenie warunków komutacji.
3. Rozwiązanie otrzymanych równań różniczkowych stosując jedną z poniższych metod:

- *Metodę klasyczną* - polegającą na rozwiązywaniu równań różniczkowych z wykorzystaniem metod typowo matematycznych. Metoda ta stosowana jest do układów opisanych niewielką liczbą równań różniczkowych.
- *Metody operatorowe* - polegające na rozwiązywaniu równań różniczkowych poprzez zastosowanie przekształceń całkowych. Podobnie jak metoda klasyczna stosowane są do układów opisanych niewielką liczbą równań różniczkowych.
- *Metodę zmiennych stanu* - polegającą na wykorzystaniu metod algebry liniowej do rozwiązywania równań różniczkowych liniowych lub metod numerycznych do rozwiązywania równań różniczkowych liniowych oraz nieliniowych. Metoda ta stosowana jest do układów zawierających dużą liczbę elementów *L*, *M*, *C*, a szczególnie do układów nieliniowych.

W kolejnych rozdziałach scharakteryzowane zostaną powyższe trzy metody analizy stanów nieustalonych.

2.2. Metoda klasyczna

Metoda klasyczna analizy stanów nieustalonych [11], [12], [81], [100], [101] polega na rozwiązywaniu równań różniczkowych przy zastosowaniu metod typowo matematycznych. Znając parametry obwodu można napisać dla niego równania napięciowe, zgodnie z drugim prawem Kichhoffa, oraz równania prądowe, zgodnie z pierwszym prawem Kirchhoffa. Otrzymuje się w ten sposób układ równań różniczkowych lub różniczkowo-całkowych. W ogólnym przypadku jest to niejednorodne równanie różniczkowe liniowe *n*-tego rzędu o stałych współczynnikach [12], [19], [103]:

$$\frac{d^n x(t)}{dt^n} + a_{n-1}\frac{d^{n-1}x(t)}{dt^{n-1}} + \cdots + a_1\frac{dx(t)}{dt} + a_0 x(t) = f(t) \tag{2.1}$$

gdzie współczynniki $a_{n-1},\ldots,a_1,a_0$ zawierają parametry obwodu *R*, *L*, *C* i *M*, *x*(*t*) jest niewiadomą funkcją (*odpowiedzią* układu) określającą najczęściej wybrany prąd lub napięcie, zaś *f*(*t*) jest *wymuszeniem* będącym najczęściej źródłem napięcia lub prądu.

Jeśli dla równania (2.1) określony jest warunek początkowy:

$$x_0 = x(t_0), \;\; t_0 \in \langle 0, \infty)$$ (2.2)

to równanie to ma zawsze jednoznaczne rozwiązanie.

Na podstawie teorii równań różniczkowych liniowych rozwiązanie ogólne równania (2.1) można przedstawić w postaci sumy algebraicznej dwóch składowych:

$$x(t) = x_p(t) + x_u(t)$$ (2.3)

gdzie: $x_p(t)$ - całka ogólna równania jednorodnego nazywana *składową przejściową* lub *swobodną* (opisuje stan nieustalony obwodu), $x_u(t)$ - całka szczególna równania (2.1) nazywana *składową wymuszoną* lub *ustaloną* (opisuje stan ustalony obwodu).

Składową przejściową $x_p(t)$ określa się rozwiązując równanie jednorodne:

$$\frac{d^n x_p(t)}{dt^n} + a_{n-1}\frac{d^{n-1} x_p(t)}{dt^{n-1}} + \cdots + a_1 \frac{dx_p(t)}{dt} + a_0 x_p(t) = 0$$ (2.4)

Całka szczególna $x_u(t)$ wyznaczana jest poprzez rozwiązanie obwodu w stanie ustalonym.

Równanie (2.1) można sprowadzić do równoważnego układu równań rzędu pierwszego, który w postaci macierzowej zapisuje się jako:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{f}(t), \quad \mathbf{x}_0 = \mathbf{x}(t_0)$$ (2.5)

gdzie $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T$ - wektor odpowiedzi, $\mathbf{f}(t) = [f_1(t), f_2(t), \ldots, f_n(t)]^T$ - wektor wymuszeń, **A**, **B** - macierze współczynników, zaś $\mathbf{x}_0$ - wektor warunków początkowych.

Podobnie jak dla równania (2.1), także i dla układu równań (2.5), rozwiązanie można przedstawić w postaci sumy dwóch składowych:

$$\mathbf{x}(t) = \mathbf{x}_p(t) + \mathbf{x}_u(t)$$ (2.6)

gdzie $\mathbf{x}_p(t)$ - wektor składowych przejściowych, zaś $\mathbf{x}_u(t)$ - wektor składowych ustalonych.

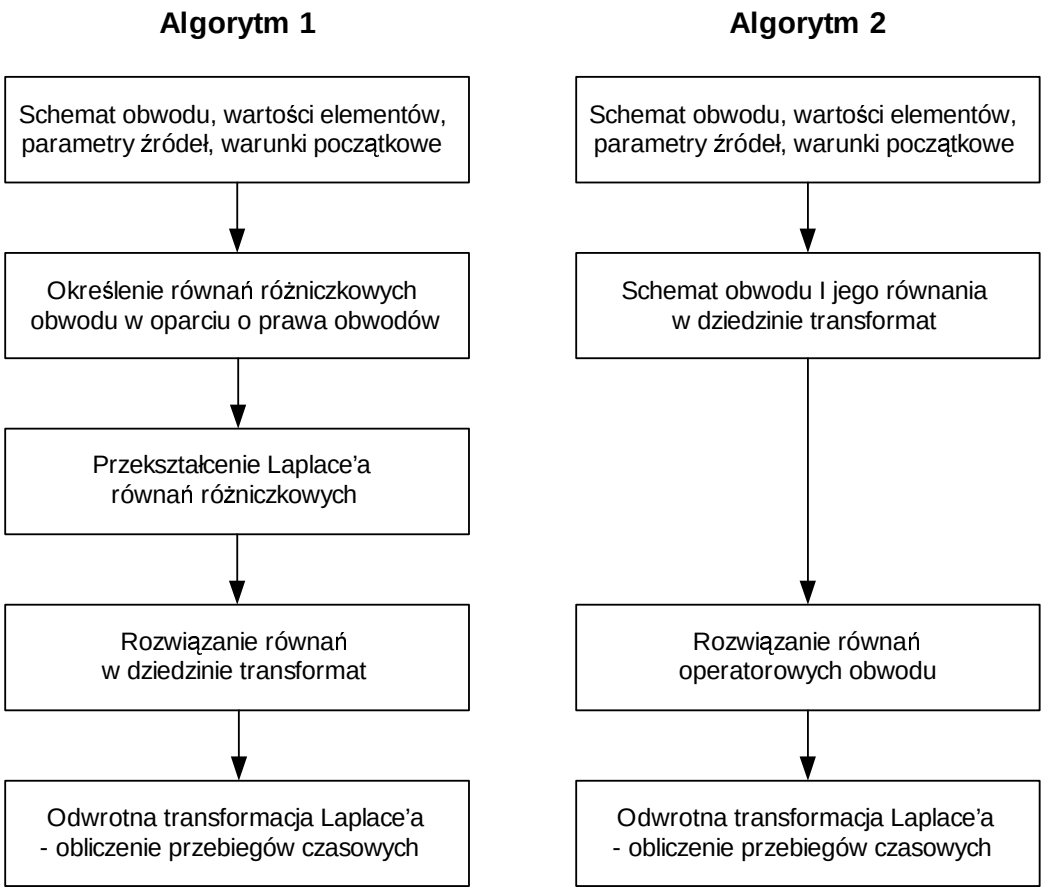
Składową przejściową $\mathbf{x}_p(t)$ wyznacza się rozwiązując równanie wektorowe jednorodne:

$$\frac{d\mathbf{x}_p(t)}{dt} = \mathbf{A}\mathbf{x}_p(t) \tag{2.7}$$

z zadanym warunkiem początkowym $\mathbf{x}_{p,0} = \mathbf{x}_p(t_0)$, zaś całkę szczególną - na podstawie rozwiązania obwodu w stanie ustalonym.

2.3. Metoda operatorowa

Metoda operatorowa [11], [12], [81], [100], [101] analizy stanów nieustalonych polega na algebraizacji równań różniczkowo-całkowych opisujących układ za pomocą przekształcenia Laplace'a. W pracy [100] przedstawiono dwa algorytmy analizy stanów nieustalonych w obwodach elektrycznych metodą operatorową (rys. 2.1).



Rys. 2.1 Algorytmy analizy stanów nieustalonych w obwodach elektrycznych metodą operatorową [100].

W algorytmie pierwszym, na podstawie schematu obwodu, zapisujemy równania różniczkowe obwodu oraz warunki początkowe. Następnie przeprowadzamy obustronną transformację Laplace'a równań różniczkowych, wyznaczamy poszukiwane wartości i stosujemy odwrotną transformację Laplace'a przechodząc do dziedziny czasu.

W drugim algorytmie przedstawiamy schemat obwodu w dziedzinie transformat i na podstawie tego schematu zapisujemy równania. Następnie rozwiązujemy równania operatorowe obwodu i stosując odwrotną transformację Laplace'a przechodzimy do dziedziny czasu.

Przedstawiając schemat obwodu w dziedzinie transformat stosuje się modele operatorowe elementów obwodu. Opis rezystora w dziedzinie czasu i dziedzinie transformat jest taki sam. W przypadku elementów L i C, mających niezerowe warunki początkowe, ich modele operatorowe uwzględniają te warunki w postaci dodatkowych źródeł napięciowych lub prądowych. Natomiast dla cewek sprzężonych magnetycznie w zastępczym schemacie operatorowym pojawiają się źródła sterowane.

W analizie stanów nieustalonych metodą operatorową słuszne są wszystkie twierdzenia dotyczące analizy obwodów metodą symboliczną [74], [100], [101]: zasada połączenia szeregowego i równoległego elementów obwodu, przekształcenia gwiazda-trójkąt i trójkąt-gwiazda, zasada superpozycji, metoda prądów oczkowych i metoda potencjałów węzłowych, zasada wyodrębniania i zasada przenoszenia źródeł, twierdzenia Thevenina i twierdzenie Nortona.

Oryginały funkcji operatorowych można wyznaczyć kilkoma sposobami [11]:

- *używając tablic transformat i ich oryginałów* - metoda stosowana dla prostych przykładów, w których transformaty przedstawiane są w postaci występującej w tablicy transformat i ich oryginałów,
- *metodą residuów* - stosowana w przypadkach, gdy transformata ma postać ułamka (funkcji wymiernej), którego licznik i mianownik są wielomianami zmiennej zespolonej *s*,

- *na podstawie wzorów Heaviside'a* - metoda stosowana, gdy funkcja operatorowa ma bieguny jednokrotne i można ją rozłożyć na ułamki proste.

Analizując stany nieustalone można połączyć metodę operatorową i metodę klasyczną [12]. W takim przypadku składową przejściową wyznacza się metodą operatorową, zaś składową ustaloną - metodą klasyczną.

2.4. Metoda zmiennych stanu

Metoda zmiennych stanu [11], [12], [20], [59], [61], [103] (nazywana także metodą *równań stanu* lub metodą *przestrzeni stanów*) ogólnie polega na [100]:

1. Formułowaniu równań różniczkowych, tzw. równań stanu (np. komputerowo w oparciu o metody topologiczne [20]).
2. Rozwiązaniu równań stanu:
   - metodami algebry liniowej dla układów liniowych,
   - metodami numerycznymi dla układów liniowych i nieliniowych.
3. Opcjonalnym określeniu wniosków jakościowych dotyczących otrzymanych rozwiązań równań stanu.

W pracach [59], [61] jako definicję *stanu układu (procesu)* określono zbiór liniowo niezależnych wielkości $x_1, x_2, \ldots, x_n$, określających w pełni skutki przeszłych oddziaływań na układ, który jest wystarczający do wyznaczenia przebiegów chwilowych dowolnych wielkości w tym układzie dla $t > t_0$, gdy znane są wymuszenia i parametry tego układu. Wielkości wchodzące w skład tego zbioru nazywane są *zmiennymi* lub *współrzędnymi stanu*, natomiast wektor będący zbiorem tych zmiennych nazywany jest *wektorem stanu*. Zbiór wszystkich możliwych wartości wektora stanu $\mathbf{x}(t)$ tworzy *przestrzeń stanów*.

Można spotkać także inną definicję *stanu układu* [100], według której stanem układu nazywany jest zbiór funkcji $x_1(t), x_2(t), \ldots, x_n(t)$ pozwalających określić dynamikę układu dla czasu $t \in \langle t_0, \infty)$, gdzie t_0 jest to umownie przyjęty czas

początkowy. Przy takiej definicji wektor stanu będzie wektorem $\mathbf{x}(t)$ złożonym z funkcji $x_1(t), x_2(t), \ldots, x_n(t)$.

W metodzie zmiennych stanu układ liniowy przedstawiany jest w zapisie macierzowym w następującej postaci [1], [20], [59], [61]:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \end{aligned} \tag{2.8}$$

gdzie: **x** - wektor stanu, **u** - wektor wymuszeń, **y** - wektor wyjść układu, **A** - macierz stanu (obwodu, układu) o wymiarze $n \times n$, **B** - macierz wymuszeń o wymiarze $n \times p$ (p - liczba wymuszeń), **C** - macierz odpowiedzi o wymiarze $q \times n$ (q - liczba wyjść), **D** - macierz transmisyjna układu o wymiarze $q \times p$.

Pierwsze równanie układu (2.8) nazywane jest *równaniem stanu* zaś drugie *równaniem wyjścia* układu (*odpowiedzi*).

W liniowych obwodach elektrycznych liczba zmiennych stanu jest równa liczbie liniowo niezależnych prądów płynących przez indukcyjności lub strumieni skojarzonych z tymi indukcyjnościami i napięć na pojemnościach lub ładunków przyporządkowanych tym pojemnościom [100]. Wartości zmiennych stanu dla $t = t_0$ określane są na podstawie praw komutacji.

Dla układów nieliniowych zapis równań stanu ma postać [82], [103]:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \end{aligned} \tag{2.9}$$

gdzie: **x** - wektor stanu, **u** - wektor wejść (wymuszeń), **y** - wektor wyjść (odpowiedzi).

Równania stanu można formułować metodą bezpośrednią lub metodą dekompozycji [100]. Metoda bezpośrednia polega na przyjęciu zmiennych stanu i zmiennych wyjściowych, utworzeniu opisu układu z wykorzystaniem praw Kirchhoffa i równań elementów, przekształceniu równań do postaci normalnej i ich rozwiązaniu. Metoda ta w praktyce stosowana jest do układów niskiego rzędu. W metodzie dekompozycji tworzona jest wstępna postać równań, której wektor stanu złożony jest ze wszystkich napięć na pojemnościach i wszystkich prądów płynących

przez indukcyjności, a następnie równania przekształcane są do postaci normalnej, w której wektor stanu składa się z liniowo niezależnych zmiennych stanu.

W przypadku układów liniowych, równania stanu mogą być rozwiązywane w dziedzinie czasu (metoda macierzy $e^{\mathbf{A}t}$, metoda diagonalizacji wektorów własnych, metody różnicowe) lub w dziedzinie częstotliwości (metoda transformacji Laplace'a) [100].

Metoda macierzy $e^{\mathbf{A}t}$

Rozwiązanie równania (2.8) z zadanym warunkiem początkowym $\mathbf{x}(t_0)$ określa poniższy wzór [59], [61], [62], [100], [103]:

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}(t_0) + \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau \tag{2.10}$$

gdzie $e^{\mathbf{A}t}$ jest macierzą podstawową (tranzycyjną) układu.

Powyższe rozwiązanie można zapisać w postaci sumy dwóch składników:

$$\mathbf{x}(t) = \mathbf{x}_p(t) + \mathbf{x}_u(t) \tag{2.11}$$

gdzie:

$$\mathbf{x}_p(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0, \quad \mathbf{x}_{p,0} = \mathbf{x}_p(t_0) \tag{2.12}$$

jest wektorem składowej przejściowej odpowiedzi układu i zależy tylko od macierzy tranzycyjnej i wektora warunków początkowych, a jest rozwiązaniem równania różniczkowego jednorodnego $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$. Natomiast $\mathbf{x}_u(t)$ jest składową wymuszoną odpowiedzi, zależną tylko od wektora wymuszeń układu $\mathbf{u}(t)$.

Rozwiązanie powyższych równań wymaga obliczenia macierzy $e^{\mathbf{A}t}$ [11], [12], [61], [62], [103], do czego stosowane jest rozwinięcie w szereg nieskończony, metoda przekształcenia Laplace'a, wzór interpolacyjny Sylvestera lub metoda wektorów własnych.

Metoda diagonalizacji

Metoda diagonalizacji polega na przekształceniu układu równań stanu z macierzą **A** do innego układu równań stanu, którego macierz **A'** będzie diagonalna. Rozwiązanie tak otrzymanego układu jest proste, gdyż każde równanie może być rozwiązywane niezależnie.

Metody różnicowe

Metody te opierają się na różnych sposobach przybliżenia pochodnych wektora stanu. Otrzymywane równania różnicowe rozwiązywane są najczęściej metodami numerycznymi. Z tego powodu rozwiązania równań różnicowych uzyskuje się w dyskretnych chwilach czasu.

Do podstawowych zalet metody zmiennych stanu należą:

- możliwość zastosowania do szerokiej klasy układów, a więc zarówno obwodów liniowych, jak i nieliniowych oraz niestacjonarnych [11], [103]; w odniesieniu do układów nieliniowych jest w zasadzie jedyną możliwą do stosowania w sposób efektywny,
- względnie prosta algorytmizacja metod formułowania i rozwiązywania równań stanu układów o dużej liczbie elementów dynamicznych (L, M, C) [100],
- możliwość zastosowania technik komputerowych do analizy dużych układów elektrycznych [100],
- łatwość oprogramowania, mała złożoność obliczeniowa i duża dokładność algorytmów analizy [11], [103].

Do wad metody zmiennych stanu zalicza się:

- niejednoznaczność opisu za pomocą równań stanu [103],
- trudności z formułowaniem równań stanu w przypadku niektórych układów nieliniowych [103].

2.5. Podsumowanie

Metoda klasyczna analizy stanów nieustalonych układów elektrycznych najczęściej stosowana jest do układów liniowych niskich rzędów, gdyż rozwiązanie układów wyższych rzędów przy jej zastosowaniu jest dość trudne. Otrzymane rozwiązanie ma postać wzoru analitycznego. W przypadku układów nieliniowych metoda klasyczna umożliwia otrzymanie rozwiązania analitycznego tylko w nielicznych przypadkach. Podobnie jest z metodą operatorową, którą stosuje się tylko do układów liniowych, zawierających niewielką liczbą równań. Metoda zmiennych stanu stosowana może być zarówno do układów liniowych, jak i nieliniowych. Zastosowanie w metodzie zmiennych stanu algebry macierzowej do rozwiązania układów liniowych jest praktycznie wykonalne dla układów do trzeciego rzędu. Natomiast w przypadku układów liniowych wyższych rzędów i układów nieliniowych najczęściej stosowane są metody numeryczne. Problem numerycznego rozwiązania równań stanu jest znany od bardzo dawna, toteż istnieją dobrze opracowane, przetestowane i wydajne metody i algorytmy numeryczne przeznaczone do tego celu. W następnym rozdziale przedstawiony zostanie przegląd tego typu metod.

W przypadku rozwiązywania układów nieliniowych stosowane są także metody graficzne oraz metody graficzno-analityczne. Metody graficzne umożliwiają otrzymanie informacji nie tylko ilościowych (wartości badanych wielkości), ale także jakościowych (dynamika badanych układów). Można je stosować tylko do układów nieliniowych niskiego rzędu.

W analizie stanów nieustalonych układów nieliniowych są stosowane również inne metody [12]:

1. *Metoda aproksymacji analitycznej* - polega na aproksymacji charakterystyk elementów nieliniowych obwodu za pomocą funkcji analitycznych, dostatecznie dobrze przybliżających charakterystyki rzeczywiste. Funkcje aproksymujące powinny być stosunkowo proste. Stosowane są różne aproksymacje: liniowa, paraboliczna, szeregiem wykładniczym, szeregiem trygonometrycznym.

2. *Metoda linearyzacji* - polega na przybliżeniu rzeczywistej charakterystyki elementu nieliniowego występującego w obwodzie prostą lub kilkoma odcinkami prostej (łamaną). Dzięki temu równania różniczkowe opisujące stan nieustalony stają się równaniami różniczkowymi liniowymi, do których możemy stosować metody analizy obwodów liniowych.
3. *Metoda iteracyjna* - w metodzie tej rozwiązanie wyznaczane jest poprzez iterację, czyli wielokrotne wykonywanie tych samych działań matematycznych. Po każdej iteracji otrzymywane jest lepsze przybliżenie rozwiązania równania różniczkowego. Stąd wzięła się druga nazwa tej metody - metoda kolejnych przybliżeń.
4. Inne metody: *metoda kolejnych odcinków prądowych, metoda kolejnych odcinków czasowych, metoda płaszczyzny fazowej*.

3. Sekwencyjne metody numeryczne analizy stanów nieustalonych

3.1. Wstęp

W rozdziale tym przedstawione zostaną sekwencyjne metody numeryczne rozwiązywania równań stanu czyli metody rozwiązywania układów równań różniczkowych zwyczajnych (ang. *ODEs - Ordinary Differential Equations*) o postaci:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2, \ldots, x_N, t) \\ \dot{x}_2 &= f_2(x_1, x_2, \ldots, x_N, t) \\ &\ldots \\ \dot{x}_N &= f_N(x_1, x_2, \ldots, x_N, t) \end{aligned} \tag{3.1}$$

z zadanym warunkiem początkowym w punkcie t_0:

$$x_1(t_0) = x_{10}, \quad x_2(t_0) = x_{20}, \quad \ldots, \quad x_N(t_0) = x_{N0} \tag{3.2}$$

W przypadku analizy stanów nieustalonych numeryczne rozwiązanie układu równań różniczkowych zwyczajnych ma zazwyczaj na celu wyznaczenie wartości rozwiązania w określonym przedziale $\langle a, b \rangle$. Przedstawiając metody numeryczne ograniczymy się tylko do jednego równania, gdyż upraszcza to znacznie zapis i jest powszechnie stosowaną praktyką w literaturze [71].

Rozwiązywane jest zatem równanie:

$$\frac{dx}{dt} = f(x,t), \quad a \le t \le b, \quad x(a) = x_0 \tag{3.3}$$

gdzie x_0 jest znanym warunkiem początkowym w punkcie $a = t_0$.

Zakładając że:

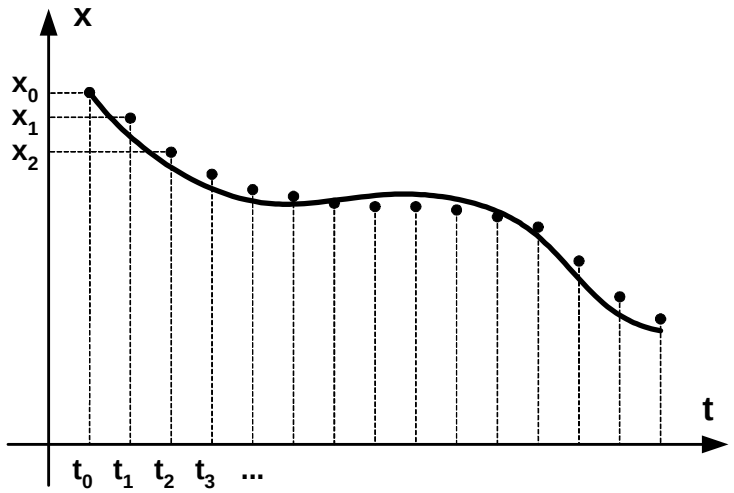
- funkcja $f(x,t)$ jest określona i ciągła w obszarze $a \le t \le b$, $-\infty < x < \infty$, gdzie a i b są skończone,
- istnieje stała $L > 0$ taka, że dla każdego $t \in \langle a, b \rangle$ i dowolnych liczb x_1 i x_2 zachodzi nierówność (warunek Lipschitza):

$$\|f(x_1,t)-f(x_2,t)\| \le L\|x_1-x_2\| \tag{3.4}$$

można udowodnić, że w przedziale $\langle a,b\rangle$ istnieje dokładnie jedna funkcja ciągła i różniczkowalna $x(t)$ spełniająca równanie (3.3) z danym warunkiem początkowym [43], [51], [65], [67], [91], [95].

Numeryczne rozwiązanie równania (3.3) przy zadanym warunku początkowym nazywane jest także numerycznym rozwiązaniem zagadnienia początkowego (ang. *IVP - Initial Value Problem*) lub numerycznym całkowaniem zagadnienia początkowego [71].

Ogólna idea numerycznego całkowania zagadnienia początkowego jest następująca. Rozpoczynając od znanej wartości x_0 w punkcie t_0 przechodzimy przez cały przedział $\langle a,b\rangle$ obliczając wartości przybliżone rozwiązania dokładnego w wybranych punktach $t_1,t_2,t_3,\ldots$, aż do osiągnięcia końca przedziału (rys. 3.1).



Rys. 3.1 Rozwiązanie dokładne (linia ciągła) i wartości przybliżone rozwiązania dokładnego w punktach $t_1,t_2,t_3,\ldots$.

Obliczenie wartości przybliżonej x_{n+1} w kolejnym punkcie t_{n+1} nazywane jest jednym *krokiem obliczeń*, zaś sposób obliczenia tej wartości (wyrażony najczęściej wzorem) w jednym kroku - *metodą numeryczną* [71]. Odległość pomiędzy dwoma kolejnymi punktami t_n,t_{n+1}, $n=0,1,\ldots$ nazywana jest *długością kroku całkowania* i oznaczana przez h_n. Jeśli odległości pomiędzy wszystkimi punktami t_n,t_{n+1} są takie

same ($h = const.$), to mówimy, że obliczenia wykonywane są ze *stałą długością kroku całkowania*. Jeśli zaś odległości te zmieniają się, to mamy obliczenia ze *zmienną długością kroku całkowania*.

Do obliczenia wartości przybliżenia x_{n+1} w punkcie t_{n+1} wykorzystywane są przybliżenia obliczone we wcześniejszych punktach. Jeśli do wykonania jednego kroku obliczeń wykorzystywane jest tylko jedno przybliżenie z poprzedniego kroku, to mówimy, że metoda numeryczna jest *metodą jednokrokową* (ang. *onestep method*). Oznacza to, że do obliczenia x_1 wykorzystujemy warunek początkowy x_0, do obliczenia x_2 - uprzednio obliczoną wartość x_1, do obliczenia x_3 - wartość x_2, itd. W przypadku, gdy do wykonania jednego kroku obliczeń wykorzystywanych jest klika przybliżeń z poprzednich kroków, to metodę taką nazywamy *metodą wielokrokową* (ang. *multistep method*). Jeśli będzie to *k* przybliżeń, to mówimy, że jest to metoda *k-krokowa*. Metody, w których podany wzór pozwala od razu wyznaczyć wartość x_{n+1} nazywane są *metodami jawnymi*, *ekstrapolacyjnymi* lub *otwartymi* (ang. *explicit methods*), zaś metody, w których obliczenie x_{n+1} wymaga dodatkowo rozwiązania nieliniowego równania algebraicznego (prawa strona wzoru metody zawiera zmienne występujące po lewej stronie), nazywane są *metodami niejawnymi, interpolacyjnymi* lub *zamkniętymi* (ang. *implicit methods*). Praktyczną realizacją metod wielokrokowych niejawnych są tzw. *metody predyktor-korektor* (ang. *predictor-corrector methods*). Jeśli we wzorach metody liczba zmiennych pośrednich jest nie większa od jednej, to metodę nazywamy *jednoetapową* (ang. *onestage method*), w przeciwnym przypadku - *wieloetapową* (ang. *multistage method*).

Wyznaczanie kolejnych wartości rozwiązania przybliżonego obarczone jest błędem. W analizie metod całkowania zagadnienia początkowego zasadniczo nie analizuje się błędu całkowitego, gdyż jest to dość trudne i ma niewielkie znaczenie praktyczne [71]. Poddaje się natomiast analizie błąd nazywany *błędem aproksymacji* (*błędem odcięcia, lokalnym błędem obcięcia*). Błąd ten definiowany jest jako różnica wartości rozwiązania dokładnego $x(t_{n+1})$ w punkcie t_{n+1} i rozwiązania numerycznego x_{n+1} w tym samym punkcie [20]. Przyjmuje się przy tym, że dla chwili t_n wyrażenie $x_n = x(t_n)$ jest rozwiązaniem dokładnym. Błąd aproksymacji jest zatem błędem jaki

wnosi sama metoda w jednym kroku obliczeń bez uwzględniania wpływu błędów popełnionych we wcześniejszych krokach [71]. Błąd aproksymacji wykorzystywany jest często w metodach ze zmiennym krokiem całkowania do regulacji długości tego kroku. Poza błędem aproksymacji istnieją także błędy wynikające z reprezentacji liczb w arytmetyce danej maszyny cyfrowej (błędy zaokrągleń). Wielkość tego błędu jest zazwyczaj o wiele rzędów mniejsza od wielkości błędu aproksymacji i w większości rozważań nie jest uwzględniana [71], chociaż może mieć znaczenie w przypadku bardzo dużej liczby kroków obliczeniowych.

W kolejnych rozdziałach przedstawione zostaną najczęściej stosowane i najbardziej popularne metody jednokrokowe i wielokrokowe oraz metody predyktor-korektor przeznaczone do całkowania zagadnień początkowych.

3.2. Metody jednokrokowe

Metody jednokrokowe [20], [25], [71], [95] są to metody, w których do obliczenia wartości przybliżonej x_{n+1} w punkcie t_{n+1} wykorzystywane jest tylko jedno przybliżenie x_n w punkcie t_n z kroku bezpośrednio poprzedzającego. Większość metod jednokrokowych swój rodowód ma w rozwinięciu w szereg Taylora.

Metoda rozwinięcia w szereg Taylora

Metoda rozwinięcia w szereg Taylora (metoda Taylora) [20], [22], [23], [47], [91] powstaje poprzez rozwinięcie rozwiązania dokładnego $x(t)$ w szereg Taylora w otoczeniu punktu $t = t_n$ i obliczeniu rozwinięcia dla $t = t_{n+1}$. Po zastąpieniu rozwiązania dokładnego przez rozwiązanie przybliżone otrzymujemy wzór ogólny metody Taylora [20]:

$$x_{n+1} = x_n + \frac{h}{1!} f(x_n, t_n) + \frac{h^2}{2!} f^{(1)}(x_n, t_n) + \ldots + \frac{h^p}{p!} f^{(p-1)}(x_n, t_n) \tag{3.5}$$

Równanie (3.5) jest często przedstawiane w następującej postaci [20], [22]:

$$x_{n+1} = x_n + h \cdot T_p(x_n, t_n; h) \tag{3.6}$$

gdzie:

$$T_p(x_n,t_n;h) = f(x_n,t_n) + \frac{h}{2!} f^{(1)}(x_n,t_n) + \ldots + \frac{h^{p-1}}{p!} f^{(p-1)}(x_n,t_n) \tag{3.7}$$

Ograniczając liczbę członów rozwinięcia do *p* otrzymujemy metodę *p-tego rzędu*. Dla $p=1$ jest to jedna z najprostszych metod - metoda Eulera. Przyjmując natomiast $p=2$ otrzymujemy metodę Taylora rzędu drugiego, tzw. ulepszoną metodę Eulera [20], [24], [83], [96]:

$$x_{n+1} = x_n + h \cdot f(x_n,t_n) + \frac{h^2}{2}\left[f_x(x_n,t_n) f(x_n,t_n) + f_t(x_n,t_n)\right] \tag{3.8}$$

gdzie f_x i f_t są pochodnymi cząstkowymi, odpowiednio, po *x* i *t*. Konieczność obliczania pochodnych cząstkowych jest największą wadą tej metody i z tego względu metoda Taylora nie jest praktycznie stosowana [22].

Metoda Eulera

Ekstrapolacyjna metoda Eulera (metoda łamanych) [1], [20], [22], [51], [67], [75], [95], [97], [98] jest najprostszą metodą całkowania zagadnień początkowych. Wykorzystuje ona tylko dwa pierwsze wyrazy rozwinięcia w szereg Taylora:

$$x_{n+1} = x_n + h \cdot f(x_n,t_n) \tag{3.9}$$

Powyższy wzór można otrzymać także w inny sposób, np. poprzez aproksymację pochodnej w punkcie (t_n, x_n) za pomocą ilorazu różnicowego $(x_{n+1} - x_n)/h$ [25], [96].

W praktyce metoda Eulera jest rzadko stosowana, gdyż jest bardzo wolno zbieżna i otrzymanie dobrej dokładności obliczeń wymaga bardzo małego kroku *h* [51]. Zwiększenie liczby kroków powoduje natomiast narastanie błędów zaokrągleń.

Metody Rungego-Kutty

Metody Rungego-Kutty [20], [21], [43], [71], [75], [85] powstają z rozwinięcia w szereg Taylora, poprzez zastąpienie $T_p(x_n,t_n;h)$ z równania (3.6) inną funkcją $K_p(x_n,t_n;h)$, dla której nie jest wymagana znajomość pochodnych cząstkowych $f(x,t)$, oraz, która zapewni ten sam rząd wartości błędu odcięcia co odpowiedni algorytm Taylora [20].

Metody Rungego-Kutty określane są wzorem ogólnym [65], [71], [83], [97]:

$$\begin{aligned} &x_{n+1} = x_n + \sum_{i=1}^{p} w_i k_i \\ &k_1 = h \cdot f(x_n, t_n), \quad k_i = h \cdot f(x_n + \sum_{j=1}^{i-1} a_{ij} k_j, t_n + c_i h) \;\; dla\; i > 1 \end{aligned} \tag{3.10}$$

gdzie $p \geq 1$ i nazywane jest rzędem metody Rungego-Kutty, zaś w_i, a_{ij}, c_i są odpowiednimi współczynnikami liczbowymi. Współczynniki te mogą być określane w różny sposób, toteż istnieją różne metody Rungego-Kutty tego samego rzędu [67], [71], [89]. Poniżej przedstawione są najczęściej stosowane wzory metod Rungego-Kutty dla $p = 1,2,3,4$:

- dla $p = 1$ otrzymujemy metodę Eulera określoną wzorem (3.9).
- dla $p = 2$ otrzymujemy metodę drugiego rzędu (metodę Heuna, zmodyfikowany algorytm trapezów) [1], [7], [8], [9], [20], [22], [25], [97]:

$$\begin{aligned} &k_1 = h \cdot f(x_n, t_n) \\ &k_2 = h \cdot f(x_n + k_1, t_n + h) \\ &x_{n+1} = x_n + (k_1 + k_2)/2 \end{aligned} \tag{3.11}$$

- dla $p = 3$ otrzymujemy metodę trzeciego rzędu [7], [8], [9]:

$$\begin{aligned} &k_1 = h \cdot f(x_n, t_n) \\ &k_2 = h \cdot f(x_n + k_1/2, t_n + h/2) \\ &k_3 = h \cdot f(x_n - k_1 + 2k_2, t_n + h) \\ &x_{n+1} = x_n + (k_1 + 4k_2 + k_3)/6 \end{aligned} \tag{3.12}$$

- dla $p = 4$ otrzymujemy metodę czwartego rzędu [1], [7], [8], [9], [18], [22], [84]:

$$\begin{aligned} &k_1 = h \cdot f(x_n, t_n) \\ &k_2 = h \cdot f(x_n + k_1/2, t_n + h/2) \\ &k_3 = h \cdot f(x_n + k_2/2, t_n + h/2) \\ &k_4 = h \cdot f(x_n + k_3, t_n + h) \\ &x_{n+1} = x_n + (k_1 + 2k_2 + 2k_3 + k_4)/6 \end{aligned} \tag{3.13}$$

Metoda czwartego rzędu jest najpopularniejszą i najczęściej stosowaną metodą Rungego-Kutty, zapewniającą dobrą dokładność obliczeń przy stosunkowo dużych wartościach kroku *h*. Wadą tej metody jest konieczność czterokrotnego obliczania

wartości funkcji $f(x,t)$ w czasie każdego kroku. Dodatkowo wartości te nie są wykorzystywane w żadnych następnych obliczeniach.

Wzór (3.13) jest klasycznym wzorem metody Rungego-Kutty czwartego rzędu. Oprócz niego można spotkać także inne wzory metody tego samego rzędu, np. wzór „trzech-ósmych", Gilla, Ralstona [43], [83], [91].

Zastosowanie zmiennego kroku całkowania w metodach Rungego-Kutty wymaga sprawdzenia po każdym kroku obliczeniowym, czy dla przyjętego kroku *h* otrzymane rozwiązanie jest wystarczająco dokładne. Najczęściej stosowanym sposobem szacowania błędu i rozstrzygania o dokładności obliczeń jest ekstrapolacja [71]. Idea tego sposobu polega na wykonaniu obliczeń z krokiem *h*, a następnie wykonaniu dwukrotnie obliczeń z krokiem *h*/2 i porównaniu otrzymanych wyników. Powyższa metoda nazywana jest także zasadą Rungego [43], [71]. W takim przypadku oszacowanie błędu ma postać [22], [43], [51], [95]:

$$R = \frac{\left(x_{n+1,h/2} - x_{n+1,h}\right)}{2^p - 1} \tag{3.14}$$

gdzie: $x_{n+1,\,h/2}$ - wartość otrzymana podczas dwukrotnych obliczeń z krokiem *h*/2, $x_{n+1,h}$ - wartość otrzymana podczas obliczeń z krokiem *h*, *p* - rząd metody.

W metodzie Rungego-Kutty czwartego rzędu algorytm oszacowania błędu oraz wyboru kroku całkowania jest następujący [43], [71]. Na podstawie wzoru (3.14) szacujemy błąd:

$$R = \frac{1}{15}\left(x_{n+1,h/2} - x_{n+1,h}\right) \tag{3.15}$$

Następnie przyjmujemy kryterium błędu względnego:

$$\frac{R}{x_{\max}} < eps \tag{3.16}$$

gdzie *eps* jest założoną tolerancją błędu, a $x_{\max}$ największą co do modułu z dotychczas obliczonych wartości rozwiązania. Jeśli powyższe kryterium jest spełnione, to długość kroku *h* jest zaakceptowana i jako wartość końcową przyjmuje się:

$$x_{n+1} = \frac{16x_{n+1,h/2} - x_{n+1,h}}{15} \tag{3.17}$$

Dodatkowo jeśli stosunek błędu R do x_{max} jest dużo mniejszy od *eps* to krok h jest zwiększany. Jeśli kryterium błędu nie jest spełnione, to następuje zmniejszenie długości kroku całkowania i powtórzenie obliczeń dla nowej, odpowiednio mniejszej, wartości h. W najprostszym przypadku przy zwiększaniu kroku, jako nowy krok przyjmuje się $2 \cdot h$, zaś przy zmniejszaniu - $h/2$. Inne metody określania długości kroku całkowania na podstawie ekstrapolacji przedstawiono w pracach [7], [8], [9], [25], [43], [86], [95], [96].

Ekstrapolacja nie jest jedyną metodą oszacowania błędu aproksymacji w metodach Rungego-Kutty. W kolejnych punktach przedstawione zostaną metody Rungego-Kutty-Fehlberga i Rungego-Kutty-Mersona, w których oszacowanie to odbywa się w inny sposób.

Metoda Rungego-Kutty-Fehlberga

W metodzie Rungego-Kutty-Fehlberga [7], [8], [9], [22], [26], [27], [47], [71], [88], [96] zastosowano sposób szacowania błędu realizowany za pomocą metod włożonych (ang. *embedding methods*). Są to metody, w których nakład obliczeń niezbędny do uzyskania wyniku za pomocą metody rzędu $p+1$ wystarcza do uzyskania wyniku za pomocą metody rzędu p [71]. Obliczone wartości współczynników k wykorzystywane są do wyznaczenia wartości x_{n+1} w kolejnym punkcie oraz do oszacowania błędu aproksymacji. Poniżej przedstawione są przykładowe metody włożone wraz z błędami aproksymacji R [7], [8], [9], [71], [88]:

- para metod włożonych pierwszego i drugiego rzędu:

$$\begin{aligned} k_1 &= h \cdot f(x_n, t_n) \\ k_2 &= h \cdot f(x_n + k_1/2, t_n + h/2) \\ k_3 &= h \cdot f(x_n + k_1/256 + 255k_2/256, t_n + h) \\ R &= (-k_1 + k_3)/512 \\ x_{n+1} &= x_n + (k_1 + 510k_2 + k_3)/512 \end{aligned} \tag{3.18}$$

- para metod włożonych drugiego i trzeciego rzędu:

$$\begin{aligned}
k_1 &= h \cdot f(x_n, t_n) \\
k_2 &= h \cdot f(x_n + k_1/4, t_n + h/4) \\
k_3 &= h \cdot f(x_n - 189k_1/800 + 729k_2/800, t_n + 27h/40) \\
k_4 &= h \cdot f(x_n + 214k_1/891 + k_2/33 + 650k_3/891, t_n + h) \\
R &= 23k_1/1782 - k_2/33 + 350k_3/11583 - k_4/78 \\
x_{n+1} &= x_n + (533k_1 + 1600k_3 - 27k_4)/2106
\end{aligned} \tag{3.19}$$

- para metod włożonych czwartego i piątego rzędu:

$$\begin{aligned}
k_1 &= h \cdot f(x_n, t_n) \\
k_2 &= h \cdot f(x_n + k_1/4, t_n + h/4) \\
k_3 &= h \cdot f(x_n + 3k_1/32 + 9k_2/32, t_n + 3h/8) \\
k_4 &= h \cdot f(x_n + 1932k_1/2197 - 7200k_2/2197 + 7296k_3/2197, t_n + 12h/13) \\
k_5 &= h \cdot f(x_n + 439k_1/216 - 8k_2 + 3680k_3/513 - 845k_4/4104, t_n + h) \\
k_6 &= h \cdot f(x_n - 8k_1/27 + 2k_2 - 3544k_3/2565 + 1859k_4/4104 - 11k_5/40, t_n + h/2) \\
R &= k_1/360 - 128k_3/4275 - 2197k_4/75240 + k_5/50 + 2k_6/55 \\
x_{n+1} &= x_n + 16k_1/135 + 6656k_3/12825 + 28561k_4/56430 - 9k_5/50 + 2k_6/55
\end{aligned} \tag{3.20}$$

Najczęściej stosowana jest ostatnia para metod włożonych - czwartego i piątego rzędu. Wykonanie jednego kroku obliczeń za pomocą tej pary wymaga sześciu obliczeń wartości funkcji prawej strony, przy czym od razu otrzymujemy oszacowanie błędu aproksymacji. Przykładowe algorytmy metody Rungego-Kutty-Fehlberga ze zmianą długości kroku całkowania zostały przedstawione w pracach [7], [8], [9], [71].

Metoda Rungego-Kutty-Mersona

Metoda Rungego-Kutty-Mersona [71], [90] jest udoskonaloną pięcioetapową wersją metody Rungego-Kutty czwartego rzędu:

$$\begin{aligned}
k_1 &= h \cdot f(x_n, t_n) \\
k_2 &= h \cdot f(x_n + k_1/3, t_n + h/3) \\
k_3 &= h \cdot f(x_n + k_1/6 + k_2/6, t_n + h/3) \\
k_4 &= h \cdot f(x_n + k_1/8 + 3k_3/8, t_n + h/2) \\
k_5 &= h \cdot f(x_n + k_1/2 - 3k_3/2 + 2k_4, t_n + h) \\
x_{n+1} &= x_n + (k_1 + 4k_4 + k_5)/6
\end{aligned} \tag{3.21}$$

Błąd aproksymacji dla tej metody wynosi:

$$R = \left| -2k_1 + 9k_3 - 8k_4 + k_5 \right| / 30 \tag{3.22}$$

Dużą zaletą metody Rungego-Kutty-Mersona jest to, że do oszacowania błędu aproksymacji wykorzystuje się już obliczone współczynniki k_1, k_3, k_4, k_5. Jest to bardzo wygodny i ekonomiczny sposób obliczania błędu aproksymacji.

Zastosowanie zmiennego kroku całkowania w metodzie Rungego-Kutty-Mersona przedstawiono w pracy [90]. Przed każdą iteracją błąd aproksymacji *R* porównywany jest z wymaganą dokładnością względną *eps*. Jeśli $|R| > eps$, to obliczenia są powtarzane z krokiem dwukrotnie krótszym. Gdy $|R| < eps/32$ to następuje podwojenie długości kroku. W pozostałych przypadkach długość kroku nie ulega zmianie.

Przedstawione powyżej metody Rungego-Kutty są to metody jawne. Oprócz nich stosowane są także niejawne metody Rungego-Kutty [16], [71].

3.3. Metody wielokrokowe

Metody wielokrokowe charakteryzują się tym, że w celu wykonania jednego kroku obliczeń wykorzystywane są przybliżenia obliczone w kilku kolejnych, bezpośrednio poprzedzających krokach [71], [75]. Jeśli wykorzystujemy *k* kolejnych przybliżeń, to mówimy, że metoda jest *k-krokowa*. Metody wielokrokowe oparte są na interpolacji wielomianowej. Wzór ogólny metody wielokrokowej ma postać [16], [43], [65], [88], [97]:

$$x_{n+1} = \sum_{i=1}^{k} a_i x_{n-i+1} + h \sum_{i=0}^{k} b_i f(x_{n-i+1}, t_{n-i+1}) \tag{3.23}$$

gdzie a_i i b_i są odpowiednimi współczynnikami liczbowymi.

Jeśli we wzorze (3.23) $b_0 = 0$, to metoda jest jawna, jeśli $b_0 \neq 0$, to metoda jest niejawna. Rozpoczęcie obliczeń w metodzie *k-krokowej* wymaga znajomości $x_0, x_1, \ldots, x_{k-1}$. Warunek początkowy x_0 jest znany z założenia, natomiast pozostałe wartości muszą być obliczone. W przypadku metody jawnej wykonanie jednego kroku obliczeń jest proste - od razu otrzymujemy poszukiwaną wartość x_{n+1}. Gdy stosujemy

metodę niejawną, to wyznaczenie wartości x_{n+1} jest bardziej skomplikowane, gdyż musimy dodatkowo rozwiązać nieliniowe równanie algebraiczne o postaci [71]:

$$x_{n+1} = C_{n+1} + hb_0 f(x_{n+1}, t_{n+1}) \tag{3.24}$$

gdzie C_{n+1} jest sumą składników zawierających współczynniki a_i, b_i, $i = 1,2,\ldots,k$.

Do rozwiązania tego równania najczęściej stosowana jest metoda iteracji prostej lub metoda Newtona [71].

Najbardziej znanymi metodami wielokrokowymi są: jawna metoda Adamsa-Bashfortha, niejawna metoda Adamsa-Moultona i metoda wstecznego różniczkowania.

Metoda Adamsa-Bashfortha

Metoda Adamsa-Bashfortha [1], [7], [8], [9], [21], [22], [71], [97] jest metodą wielokrokową jawną. Wzór ogólny metody jest następujący [71]:

$$x_{n+1} = x_n + h\sum_{i=1}^{k} b_{ki} f(x_{n-i+1}, t_{n-i+1}) \tag{3.25}$$

gdzie b_{ki} są odpowiednimi współczynnikami liczbowymi.

Dla kolejnych wartości k otrzymujemy metody:

- dla $k = 1$ otrzymujemy metodę Eulera opisaną wzorem (3.9),
- dla $k = 2$ otrzymujemy metodę dwukrokową:

$$x_{n+1} = x_n + h \cdot [3 f(x_n, t_n) - f(x_{n-1}, t_{n-1})]/2 \tag{3.26}$$

- dla $k = 3$ otrzymujemy metodę trzykrokową:

$$x_{n+1} = x_n + h \cdot [23 f(x_n, t_n) - 16 f(x_{n-1}, t_{n-1}) + 5 f(x_{n-2}, t_{n-2})]/12 \tag{3.27}$$

- dla $k = 4$ otrzymujemy metodę czterokrokową:

$$\begin{aligned} x_{n+1} = x_n + h \cdot [&55 f(x_n, t_n) - 59 f(x_{n-1}, t_{n-1}) + \\ &+ 37 f(x_{n-2}, t_{n-2}) - 9 f(x_{n-3}, t_{n-3})]/24 \end{aligned} \tag{3.28}$$

- dla $k = 5$ otrzymujemy metodę pięciokrokową:

$$x_{n+1} = x_n + h \cdot [1901 f(x_n, t_n) - 2774 f(x_{n-1}, t_{n-1}) + \\ + 2616 f(x_{n-2}, t_{n-2}) - 1274 f(x_{n-3}, t_{n-3}) + 251 f(x_{n-4}, t_{n-4})] / 720 \tag{3.29}$$

- dla $k = 6$ otrzymujemy metodę sześciokrokową:

$$x_{n+1} = x_n + h \cdot [4277 f(x_n, t_n) - 7923 f(x_{n-1}, t_{n-1}) + 9982 f(x_{n-2}, t_{n-2}) + \\ - 7298 f(x_{n-3}, t_{n-3}) + 2877 f(x_{n-4}, t_{n-4}) - 475 f(x_{n-5}, t_{n-5})] / 1440 \tag{3.30}$$

Metoda Adamsa-Moultona

Metoda Adamsa-Moultona [7], [8], [9], [71], [97] jest metodą wielokrokową niejawną. Wzór ogólny metody jest następujący [71]:

$$x_{n+1} = x_n + h \sum_{i=0}^{k} b_{ki}^{*} f(x_{n-i+1}, t_{n-i+1}) \tag{3.31}$$

gdzie b_{ki}^{*} są odpowiednimi współczynnikami liczbowymi.

Dla kolejnych wartości *k* otrzymujemy metody:

- dla $k = 0$ otrzymujemy metodę interpolacyjną Eulera pierwszego rzędu (wsteczną metodę Eulera) [44]:

$$x_{n+1} = x_n + h \cdot f(x_{n+1}, t_{n+1}) \tag{3.32}$$

- dla $k = 1$ otrzymujemy metodę jednokrokową (tzw. metodę trapezów) [1], [44], [97]:

$$x_{n+1} = x_n + h \cdot [f(x_{n+1}, t_{n+1}) + f(x_n, t_n)] / 2 \tag{3.33}$$

- dla $k = 2$ otrzymujemy metodę dwukrokową:

$$x_{n+1} = x_n + h \cdot [5 f(x_{n+1}, t_{n+1}) + 8 f(x_n, t_n) - f(x_{n-1}, t_{n-1})] / 12 \tag{3.34}$$

- dla $k = 3$ otrzymujemy metodę trzykrokową:

$$x_{n+1} = x_n + h \cdot [9 f(x_{n+1}, t_{n+1}) + 19 f(x_n, t_n) + \\ - 5 f(x_{n-1}, t_{n-1}) + f(x_{n-2}, t_{n-2})] / 24 \tag{3.35}$$

- dla $k = 4$ otrzymujemy metodę czterokrokową:

$$x_{n+1} = x_n + h \cdot [251 f(x_{n+1}, t_{n+1}) + 646 f(x_n, t_n) + \\ - 264 f(x_{n-1}, t_{n-1}) + 106 f(x_{n-2}, t_{n-2}) - 19 f(x_{n-3}, t_{n-3}) +] / 720 \tag{3.36}$$

- dla $k = 5$ otrzymujemy metodę pięciokrokową:

$$x_{n+1} = x_n + h \cdot [475 f(x_{n+1}, t_{n+1}) + 1427 f(x_n, t_n) - 798 f(x_{n-1}, t_{n-1}) + \\ + 482 f(x_{n-2}, t_{n-2}) - 173 f(x_{n-3}, t_{n-3}) + 27 f(x_{n-4}, t_{n-4})] / 1440 \tag{3.37}$$

Metoda wstecznego różniczkowania

Metoda wstecznego różniczkowania [71], [88] powstaje poprzez przybliżenie rozwiązania zagadnienia początkowego za pomocą odpowiedniego wielomianu interpolacyjnego, a następnie na przybliżeniu pochodnej rozwiązania za pomocą pochodnej tego wielomianu.

Wzór ogólny jawnej metody wstecznego różniczkowania ma postać:

$$x_{n+1} = \sum_{i=1}^{k} a_{ki} x_{n-i+1} + h b_{k1} f(x_n, t_n) \tag{3.38}$$

gdzie a_{ki} i b_{k1} są odpowiednimi współczynnikami liczbowymi [71], [88], np.:

- dla $k = 1$: $x_{n+1} = x_n + h \cdot f(x_n, t_n)$ (3.39)
- dla $k = 2$: $x_{n+1} = x_{n-1} + h \cdot 2 f(x_n, t_n)$ (3.40)
- dla $k = 3$: $x_{n+1} = -3x_n / 2 + 3x_{n-1} - x_{n-2} / 2 + h \cdot 3 f(x_n, t_n)$ (3.41)

Wzór ogólny niejawnej metody wstecznego różniczkowania (metody Geara) ma postać [7], [17], [43]:

$$x_{n+1} = \sum_{i=1}^{k} a_{ki}^{*} x_{n-i+1} + h b_{k0}^{*} f(x_{n+1}, t_{n+1}) \tag{3.42}$$

gdzie a_{ki}^{*} i b_{k0}^{*} są odpowiednimi współczynnikami liczbowymi [71], [88], np.:

- dla $k = 1$: $x_{n+1} = x_n + h \cdot f(x_{n+1}, t_{n+1})$ (3.43)
- dla $k = 2$: $x_{n+1} = 4x_n / 3 - x_{n-1} / 3 + 2/3 \cdot h \cdot f(x_{n+1}, t_{n+1})$ (3.44)
- dla $k = 3$: $x_{n+1} = 18x_n / 11 - 9x_{n-1} / 11 + 2x_{n-2} / 11 + 6/11 \cdot h \cdot f(x_{n+1}, t_{n+1})$ (3.45)

Metoda Geara przeznaczona jest do rozwiązywania układów źle uwarunkowanych. W pracy [7] przedstawiono algorytm Geara umożliwiający dobór kroku całkowania i rzędu metody.

3.4. Metody predyktor-korektor

Metoda predyktor-korektor (ekstrapolacyjno-interpolacyjna) [22], [43], [71], [89] jest sposobem realizacji metody wielokrokowej niejawnej. Składa się z dwóch etapów. Pierwszy etap obliczeń nazywany jest predykcją, następuje w nim obliczenie przybliżenia początkowego za pomocą metody jawnej. Drugi etap obliczeń nazywany jest korekcją, następuje w nim wykonanie kilku iteracji za pomocą metody niejawnej (najczęściej jest to od 1 do 3 iteracji). Określane są dwa podstawowe typy algorytmów predyktor-korektor [71], [88]:

- metoda typu **P(EC)$^{\mathbf{m}}$E** postaci:

$$\begin{aligned}
&\mathbf{P}: && x_{n+1}^{[0]} = \sum_{i=1}^{k} a_i^* x_{n-i+1}^{[m]} + h\sum_{i=1}^{k} b_i^* f(x_{n-i+1}^{[m]}, t_{n-i+1}) \\
&\mathbf{E}: && f_{n+1}^{[s]} = f(x_{n+1}^{[s]}, t_{n+1}) \\
&\mathbf{C}: && x_{n+1}^{[s+1]} = \sum_{i=1}^{k} a_i x_{n-i+1}^{[m]} + h\sum_{i=1}^{k} b_i f(x_{n-i+1}^{[m]}, t_{n-i+1}) + hb_0 f(x_{n+1}^{[s]}, t_{n+1}) \\
&\mathbf{E}: && f_{n+1}^{[m]} = f(x_{n+1}^{[m]}, t_{n+1})
\end{aligned} \tag{3.46}$$

$$\text{dla } s = 0,1,...,m-1$$

- metoda typu **P(EC)$^{\mathbf{m}}$** postaci:

$$\begin{aligned}
&\mathbf{P}: && x_{n+1}^{[0]} = \sum_{i=1}^{k} a_i^* x_{n-i+1}^{[m]} + h\sum_{i=1}^{k} b_i^* f(x_{n-i+1}^{[m-1]}, t_{n-i+1}) \\
&\mathbf{E}: && f_{n+1}^{[s]} = f(x_{n+1}^{[s]}, t_{n+1}) \\
&\mathbf{C}: && x_{n+1}^{[s+1]} = \sum_{i=1}^{k} a_i x_{n-i+1}^{[m]} + h\sum_{i=1}^{k} b_i f(x_{n-i+1}^{[m-1]}, t_{n-i+1}) + hb_0 f(x_{n+1}^{[s]}, t_{n+1})
\end{aligned} \tag{3.47}$$

$$\text{dla } s = 0,1,...,m-1$$

gdzie a_i^* i b_i^* są współczynnikami metody jawnej, a_i, b_i - współczynnikami metody niejawnej, zaś symbole **PCE** oznaczają: **P** - predykcja, tj. określenie przybliżenia początkowego za pomocą metody jawnej, **C** - korekcja, tj. wykonanie *m* iteracji za pomocą metody niejawnej, **E** - obliczenie prawej strony równania różniczkowego.

Przykładami metod typu predyktor-korektor są: metoda Adamsa w postaci algorytmu Krogha oraz metoda Adamsa i metoda wstecznego różniczkowania w postaci algorytmu Geara.

Metoda Adamsa [71] jest metodą predyktor-korektor typu $\mathbf{P_kEC_kE}$, dla której predyktorem jest *k*-krokowa metoda Adamsa-Bashfortha, a korektorem *k*-krokowa metoda Adamsa-Moultona. Metoda ta najczęściej stosowana jest w postaci algorytmu Krogha, umożliwiającego zmianę długości kroku całkowania i rzędu aproksymacji oraz zawierającego efektywny sposób oszacowania błędu.

Algorytm Geara [43], [44], [71] jest realizacją metody predyktor-korektor dla metody Adamsa i metody wstecznego różniczkowania. Jest to jeden algorytm różniący się dla powyższych metod jedynie wartościami liczbowymi pewnych współczynników. Przedstawiony został w postaci procedury DIFSUB w języku Fortran [44]. Metoda Adamsa zrealizowana została za pomocą metody predyktor-korektor typu $\mathbf{P(EC)^m}$. Rząd aproksymacji metody Adamsa można zmieniać w zakresie od 1 do 7. Dla $k=1$ predyktorem jest jednokrokowa metoda Adamsa-Bashfortha (metoda Eulera), a korektorem jednokrokowa metoda Adamsa-Moultona (wsteczna metoda Eulera). Dla $k \geq 2$ predyktorem jest *k*-krokowa metoda Adamsa-Bashfortha, zaś korektorem (*k*-1)-krokowa metoda Adamsa-Moultona. Metoda wstecznego różniczkowania także zrealizowana została jako metoda predyktor-korektor typu $\mathbf{P(EC)^m}$. Rząd aproksymacji dla tej metody można zmieniać w zakresie od 1 do 6. Predyktorem jest *k*-krokowa metoda jawna, zaś korektorem jest *k*-krokowa metoda niejawna wstecznego różniczkowania. W algorytmie Geara zastosowana została technika interpolacyjna zmiany długości kroku całkowania określana jako sposób Nordsiecka zmiany długości kroku [71]. Procedura DIFSUB jest bardzo dobra do rozwiązywania układów źle uwarunkowanych (typu *stiff*).

Do innych metod typu predyktor-korektor należą także: metoda Hamminga, której algorytm wraz z wyborem kroku całkowania przedstawiono w pracy [43], oraz metoda Milne'a [89].

3.5. Podsumowanie

Oprócz przedstawionych w poprzednich rozdziałach metod jednokrokowych i wielokrokowych oraz metod typu predyktor-korektor występują także inne metody. Wśród nich najbardziej popularne są metody ekstrapolacyjne, np. metoda Gragga-Bulirscha-Stoera (GBS) [71], [96]. Jest to jedyna metoda ekstrapolacyjna stosowana w praktyce. W metodzie tej całkowanie numeryczne jest wykonywane za pomocą metody Gragga, zaś ekstrapolacja realizuje się za pomocą algorytmu Bulirscha i Stoera. Program realizujący algorytm metody przedstawiony jest w pracach [8], [9], [44]. Metoda ta jest pracochłonna, gdyż liczba obliczeń prawej strony jest duża, toteż stosowana jest tylko w przypadku, gdy obliczanie prawej strony nie jest kosztowne.

Poza metodami ekstrapolacyjnymi stosowana jest również metoda Rosenbrocka oraz metody hybrydowe [71], będące połączeniem metod wielokrokowych i metod Rungego-Kutty. Istnieją także metody półanalityczne rozwiązywania równań różniczkowych. Do metod tych należą [67], [75], [89]: metoda współczynników nieoznaczonych, metoda kolejnego różniczkowania (metoda jednego punktu) oraz metoda kolejnych przybliżeń - Picarda.

Przedstawione metody charakteryzują się określonymi własnościami, które determinują zastosowanie danej metody do rozwiązania określonego problemu. Metody Rungego-Kutty są metodami samostartującymi, tzn. nie wymagają dodatkowych obliczeń w celu wykonania jednego kroku. Jednakże wadą tych metod, jak i większości metod jednokrokowych, jest duży koszt obliczeń (dużą dokładność można uzyskać tylko przy stosowaniu małej długości kroków [95]) oraz trudności w oszacowaniu błędu. Zastosowanie metody rzędu p wymaga co najmniej p-krotnego obliczenia w jednym kroku funkcji prawej strony równania różniczkowego. Metody

Rungego-Kutty nie nadają się do stosowania w przypadku układów źle uwarunkowanych, gdyż mają skończony i mały obszar stabilności absolutnej.

Zaletą metod wielokrokowych jest niewielka liczba obliczeń - w każdym kroku prawa strona równania różniczkowego musi być obliczana tylko raz. Zaleta ta ujawnia się szczególnie w przypadku metod ze zmienną długością kroku całkowania. Do niedawna za główną wadę metod wielokrokowych uważano to, że nie były to metody samostartujące. Rozpoczęcie obliczeń w metodzie *k*-krokowej wymaga uprzedniego określenia przybliżeń początkowych $x_1, x_2, \ldots, x_{k-1}$. Stosowano do tego celu np. metodę Rungego-Kutty lub rozwinięcie funkcji $x(t)$ w szereg Taylora w otoczeniu punktu t_0 [65]. Obecnie stosowane algorytmy wielokrokowe są samostartujące dzięki możliwości zmiany rzędu metody. W takim przypadku w pierwszym kroku obliczeń stosowana jest metoda jednokrokowa, w drugim dwukrokowa, itd. [71].

4. Równoległe metody numeryczne analizy stanów nieustalonych

4.1. Wstęp

Opisane w poprzednim rozdziale metody numeryczne rozwiązywania równań stanu są to metody sekwencyjne. **W metodach tych wyznaczenie wartości rozwiązania przybliżonego x_{n+1} w punkcie t_{n+1} wymaga znajomości przybliżeń z jednego lub kilku punktów poprzedzających. W przypadku dużych układów równań (zwłaszcza nieliniowych), otrzymanie numerycznego rozwiązania w rozsądnym czasie wymaga zastosowania systemu komputerowego o dużej wydajności oraz odpowiedniego algorytmu numerycznego**. Dużą wydajność może zapewnić zastosowanie komputera równoległego lub systemu klaster. Pomimo tego, iż komputery równoległe stosowane są już od wielu lat, to liczba algorytmów równoległych rozwiązywania zagadnień początkowych układów równań różniczkowych jest stosunkowo mała, w porównaniu z np. liczbą algorytmów równoległych algebry liniowej [16], [85]. Zastosowanie algorytmów równoległych, a mówiąc ogólniej - programowania równoległego, w analizie zagadnień początkowych jest opłacalne, gdy [16], [48], [85], [93]:

- obliczanie funkcji prawej strony równania różniczkowego jest kosztowne, np. gdy wymaga rozwiązania dodatkowego problemu,
- liczba równań układu jest bardzo duża, np. w przestrzennej dyskretyzacji równań różniczkowych cząstkowych lub w analizie dużych układów elektrycznych,
- przedział całkowania $\langle a, b \rangle$ jest długi,
- układ równań różniczkowych musi być wielokrotnie całkowany, np. przy dopasowywaniu parametrów układu lub w zadaniach optymalizacji,
- otrzymanie dobrej dokładności obliczeń wymaga zastosowania bardzo małego kroku całkowania.

Spełnienie przynajmniej jednego z powyższych warunków umożliwia otrzymanie wydajnego algorytmu równoległego. Jednakże opracowanie takiego algorytmu jest dużo bardziej skomplikowane niż opracowanie algorytmu sekwencyjnego, gdyż wymaga uwzględnienia i rozwiązania kilku dodatkowych problemów, do których należą [15], [85]:

- automatyczne równoważenie obciążenia (ang. *load balancing*) procesorów wykonujących obliczenia,
- unikanie zakleszczeń (ang. *deadlock*), czyli sytuacji, w których wszystkie procesory z pewnego zbioru procesorów są zablokowane i oczekują na zdarzenie, które może być spowodowane tylko przez jakiś inny procesor z tego zbioru [102],
- wybór sposobu komunikacji między procesorami (synchroniczna lub asynchroniczna),
- unikanie „wąskich gardeł” (ang. *bottleneck*) w komunikacji między procesorami, które w znaczący sposób obniżają wydajność algorytmu,
- zastosowanie odpowiednich zagadnień testowych, gdyż zbiory testów przeznaczone dla algorytmów sekwencyjnych nie mają zazwyczaj wystarczająco dużej liczby równań oraz odpowiednio długich przedziałów całkowania.

Istnieją dwa podstawowe podejścia [85] do opracowania algorytmów równoległych. W pierwszym z nich, klasyczne i dobrze znane metody, które są zbyt kosztowne w obliczeniach sekwencyjnych, zostają ponownie przeanalizowane i odpowiednio przekształcone przy uwzględnieniu ich wewnętrznej równoległości. Drugie podejście polega natomiast na opracowaniu zupełnie nowych algorytmów numerycznych. Ten drugi sposób jest dużo bardziej przyszłościowy, gdyż umożliwia dodatkowo uwzględnienie natury rozważanego zagadnienia oraz charakterystyki docelowej architektury równoległej.

W pracy [85] przedstawiono bardziej szczegółową, 4-poziomową, hierarchię zrównoleglania metod analizy zagadnień początkowych:

- **Poziom 1** - zrównoleglanie programów sekwencyjnych.
  Zrównoleglanie programów sekwencyjnych odbywa się przy wykorzystaniu odpowiednich kompilatorów, stosowane jest do algorytmów już zaimplementowanych jako programy sekwencyjne. Wadą tej metody jest możliwość automatycznego zrównoleglenia tylko pewnych części programu (pętle, obliczanie dużych i długich wyrażeń arytmetycznych). Często kompilatory tego typu dopuszczają stosowanie odpowiednich dyrektyw zrównoleglających wpisywanych przez programistę do programów sekwencyjnych [64]. Na zwykłych (tzn. nierównoległych) komputerach dyrektywy te mogą być traktowane jak komentarze, a wówczas kod jest przenośny - nie trzeba pisać odrębnych programów na różne maszyny.

- **Poziom 2** - zrównoleglanie algorytmów sekwencyjnych.
  Zrównoleglanie algorytmów sekwencyjnych polega na analizie i zrównoleglaniu sekwencyjnego kodu bezpośrednio przez programistę. Umożliwia to otrzymanie wysokiego stopnia zrównoleglenia. Największe problemy mogą pojawić się natomiast przy równoległej implementacji algorytmów zmiany kroku całkowania.

- **Poziom 3** - modyfikacja algorytmów sekwencyjnych.
  Metoda ta polega na modyfikacji już istniejących algorytmów lub opracowaniu nowych algorytmów w celu optymalnego wykorzystania komputerów równoległych.

- **Poziom 4** - dekompozycja problemu.
  Dekompozycja problemu polega na podziale analizowanego układu na mniejsze części zawierające jedno lub kilka równań. Każda z tak powstałych części traktowana jest oddzielnie, do jej całkowania można zastosować inny krok całkowania, rząd metody oraz dodatkowe parametry dostosowane do właściwości rozpatrywanej części.

Pierwsze badania dotyczące równoległych algorytmów rozwiązywania zagadnień początkowych prowadzone były m.in. przez C.W. Geara. W pracach [45],

[46] wprowadził on klasyfikację algorytmów równoległych, wyróżniając dwie ich grupy:

1. **Równoległość układu** (**zagadnienia**) (ang. *parallelism across the system - problem*) lub **równoległość przestrzeni** (ang. *parallelism across space*).
2. **Równoległość metody** (ang. *parallelism across the method*) lub **równoległość czasu** (ang. *parallelism across time*).

W późniejszych publikacjach innych autorów [15], [16], [49], [85] podział ten został rozszerzony do trzech grup:

1. **Równoległość układu** (**przestrzeni**) (ang. *parallelism across the system - space*) - jest to równoległość obliczania prawej strony równania różniczkowego. Oznacza podzielenie układu równań różniczkowych na bloki składające się z jednego lub kilku równań, przypisanie bloków do procesorów i wykonanie równolegle obliczeń.
2. **Równoległość metody** (ang. *parallelism across the method*) - oznacza przypisanie różnych części algorytmu metody numerycznej do różnych procesorów.
3. **Równoległość czasu** (**kroków**) (ang. *parallelism across time - steps*) - oznacza możliwość jednoczesnego obliczania dużej liczby kroków całkowania.

Rzeczywiste algorytmy mogą wykorzystywać jednocześnie wszystkie trzy podejścia [16], [17].

W kolejnych rozdziałach przedstawione zostaną trzy powyższe grupy metod analizy zagadnień początkowych, przy czym należy zwrócić uwagę na to, że:

- większość dostępnej literatury poświęcona jest opracowaniu użytecznych metod numerycznych, brakuje natomiast opisu efektywnych implementacji tych metod i ich porównania,
- w polskojęzycznej literaturze poświęconej programowaniu równoległemu i rozproszonemu [10], [13], [64], [68], [72], [102] nie są przedstawiane równoległe metody numeryczne rozwiązywania zagadnień początkowych -

z tego względu opisując poszczególne metody w części przypadków podano tylko ich angielskie nazwy,

- przy przedstawieniu metod opisana zostanie tylko ich ogólna zasada działania, nie będą natomiast przedstawiane konkretne wzory, gdyż nie można, tak jak w przypadku metod sekwencyjnych, wyróżnić najczęściej stosowanych i najbardziej popularnych metod.

4.2. Równoległość układu

Równoległość układu (ang. *parallelism across the system*) jest typem algorytmów równoległych opartych na rozdzieleniu obliczania funkcji prawej strony równania różniczkowego na procesory. Układ równań różniczkowych dzielony jest na bloki przydzielane do rozwiązujących je procesorów. Liczba bloków zależna jest od liczby dostępnych procesorów oraz liczby równań tworzących oryginalny układ. Z uwagi na to algorytmy tego typu najlepiej nadają się do stosowania w dużych systemach (ang. *large-scale parallelism*) [16], [49], składających się z dużej liczby równań. Jeśli bloki powstałe po podziale są niezależne, to mogą być rozwiązywane niezależnie od siebie przy zastosowaniu różnych metod numerycznych, np. już istniejących metod sekwencyjnych. Zaletą takiej sytuacji jest brak opóźnień w komunikacji między procesorami.

Gear [45] podzielił układy równań przeznaczone do rozwiązywania tego typu algorytmami równoległymi na dwie grupy:

1. Układy homogeniczne - wszystkie bloki mają podobną strukturę. Czas obliczeń w jednym kroku w każdym bloku jest taki sam, a więc obliczenia są bardzo efektywne. Układy o takiej strukturze powstają m.in. po dyskretyzacji równań różniczkowych cząstkowych.
2. Układy heterogeniczne - układy o nieregularnej strukturze. Czas obliczeń w poszczególnych blokach różni się, a zatem dużym problemem staje się równomierne obciążenie procesorów, komunikacja między nimi

i synchronizacja. Układy heterogeniczne najczęściej występują podczas modelowania układów VLSI.

Zastosowanie równoległości układu jest efektywne w metodach jawnych, w których występuje częste obliczanie prawej strony równania różniczkowego, natomiast nie jest efektywne w metodach niejawnych, wymagających dużej ilości komunikacji między procesorami w każdym kroku całkowania.

Wśród metod realizujących równoległość układu jedną z głównych grup są metody określane jako *multirate methods* [85]. Ogólna ich idea polega na dekompozycji układu równań na bloki w zależności od szybkości zmian zmiennych stanu. Ma to szczególnie znaczenie w przypadku układów źle uwarunkowanych (typu *stiff*), w których zmienne stanu mają bardzo różniące się stałe czasowe. Podzielony układ rozwiązywany jest przy zastosowaniu różnych metod numerycznych i przy różnych długościach kroku całkowania, dostosowanych do szybkości zmian zmiennych stanu. Obliczenia w poszczególnych podukładach są synchronizowane w ten sposób, że dłuższe kroki całkowania są zawsze wielokrotnością najmniejszego kroku. Przykładami tego typu metod są [85]: odsprzężona, niejawna metoda Eulera (ang. *decoupled implicit Euler method*), odsprzężony schemat różnic wstecznych (ang. *decoupled backward differentiation scheme*) i ekstrapolacyjna, odsprzężona, niejawna metoda Eulera (ang. *extrapolated decoupled implicit Euler metod*). *Multirate methods* są prekursorami ogólniejszej klasy metod dekompozycji, której najbardziej znanym przedstawicielem jest metoda relaksacyjna (ang. *waveform relaxation metod*) [16]. Metoda ta zaliczana jest także do grupy metod realizujących równoległość czasu i tam zostanie szczegółowo opisana.

4.3. Równoległość metody

W algorytmach realizujących równoległość metody (ang. *parallelism across the method*) następuje przypisanie różnych części algorytmu metody do różnych procesorów. Może to np. dotyczyć wielokrotnego obliczania funkcji prawej strony równania różniczkowego w jednym kroku całkowania [48], [49].

Metody tego typu można podzielić na dwie kategorie [85]:

1. Do pierwszej należą algorytmy, w których równolegle wykonywane są pewne etapy metod, wynikające bezpośrednio z ich wzorów. Zaletą takiego podejścia jest niezależność stopnia równoległości od rozmiaru rozwiązywanego układu oraz brak problemów z równoważeniem obciążenia procesorów. Wadą jest natomiast niewielki stopień równoległości zależny od zastosowanej metody numerycznej, a tym samym niewielkie przyspieszenie obliczeń.
2. W drugiej grupie znajdują się metody, w których obliczenia wykonywane są równolegle w podprzedziałach powstałych po podziale przedziału całkowania. W metodach tych trudno jest otrzymać wysoki stopień równoległości, chyba że wykonywanych jest bardzo dużo nadmiarowych obliczeń (np. metoda dekompozycji domeny zaproponowana przez Nivergelta [79]). Metody tego typu mają sens praktyczny jeśli uzyska się przyspieszenie obliczeń.

Metody należące do tej grupy przeznaczone są dla małej skali równoległości (ang. *small-scale parallelism*), czyli gdy do obliczeń wykorzystywanych jest od 2 do 10 procesorów, wymagających dodatkowo bardzo szybkiej komunikacji [49]. Czasami metody tego typu stosowane są w celu zwiększenia dokładności rozwiązania otrzymywanego w tym samym czasie, co mniej dokładne metody sekwencyjne.

Równoległość metody realizowana jest w metodach bezpośrednich (jawne i niejawne metody Rungego-Kutty, schematy interpolacyjne, metody blokowe) i metodach iteracyjnych (równoległe schematy predyktor-korektor, niejawne metody blokowe, metoda iteracyjnej korekcji błędu - ang. *iterated defect correction method*), które zostaną opisane poniżej.

Metody bezpośrednie (ang. *direct methods*) - **metody Rungego-Kutty**

Sekwencyjne metody bezpośrednie nie mogą być zrównoleglane bez odpowiedniej modyfikacji ich algorytmów. Równoległe obliczanie kilku funkcji prawej strony równania różniczkowego na różnych procesorach jest możliwe do realizacji m.in. w wieloetapowych metodach Rungego-Kutty. Metody te są zasadniczo sekwencyjne,

ale w niektórych przypadkach można wyodrębnić dwa lub więcej etapów, które da się wykonać równolegle. W takim przypadku przeprowadzana jest analiza polegająca na zapisaniu współczynników liczbowych metody w odpowiedniej tabeli i sprawdzeniu zależności pomiędzy kolejnymi etapami metody. Metodę Rungego-Kutty opisaną ogólnymi wzorami (3.10) można przedstawić w postaci następującej tabeli [48]:

c_1	a_{11}	$\cdots$	a_{1p}
$\vdots$	$\vdots$	$\cdots$	$\vdots$
c_p	a_{p1}	$\cdots$	a_{pp}
	w_1	$\cdots$	w_p

Rozpatrzmy klasyczny wzór metody Rungego-Kutty czwartego rzędu (3.13), dla którego $p = 4$. Powyższa tabela w takim przypadku ma postać:

c_1	a_{11}	a_{12}	a_{13}	a_{14}
c_2	a_{21}	a_{22}	a_{23}	a_{24}
c_3	a_{31}	a_{32}	a_{33}	a_{34}
c_4	a_{41}	a_{42}	a_{43}	a_{44}
	w_1	w_2	w_3	w_4

Zaś wzór ogólny metody (3.10) dla $p = 4$:

$$\begin{aligned} k_1 &= h \cdot f(x_n, t_n) \\ k_2 &= h \cdot f(x_n + a_{21}k_1, t_n + c_2 h) \\ k_3 &= h \cdot f(x_n + a_{31}k_1 + a_{32}k_2, t_n + c_3 h) \\ k_4 &= h \cdot f(x_n + a_{41}k + a_{42}k_2 + a_{43}k_3, t_n + c_4 h) \\ x_{n+1} &= x_n + (w_1 k_1 + w_2 k_2 + w_3 k_3 + w_4 k_4) \end{aligned} \tag{4.1}$$

Porównując powyższy wzór ogólny z klasycznym wzorem metody czwartego rzędu (3.13) otrzymujemy następującą tabelę współczynników:

0	0	0	0	0
1/2	1	0	0	0
1/2	0	1	0	0
1	0	0	1	0
	1/6	1/3	1/3	1/6

Metody powyższej nie można zrównoleglić stosując równoległość metody, gdyż do obliczenia k_2 trzeba znać k_1 (informuje o tym współczynnik $a_{21}=1$, w drugim wierszu tabeli), do obliczenia k_3 trzeba znać k_2, a do obliczenia k_4 trzeba znać k_3.

Poniżej przedstawiona jest tabela niejawnej 4-etapowej metody [85], którą można zaimplementować na dwu procesorach. Współczynniki k_1 i k_2 oraz k_3 i k_4 są niezależne od siebie, zatem mogą być obliczane równolegle na oddzielnych procesorach.

$1/2$	$1/2$	0	0	0
$2/3$	0	$2/3$	0	0
$1/2$	$-5/2$	$5/2$	$1/2$	0
$1/3$	$-5/3$	$4/3$	0	$2/3$
	-1	$3/2$	-1	$3/2$

Zależnie od postaci macierzy współczynników a_{ij} z powyższych tabel, opracowano różne równoległe metody Rungego-Kutty (RK) [16], [85] znane pod nazwami: ERK (*Explicit RK*), DIRK (*Diagonally Implicit RK*), BDIRK (*Block Diagonally Implicit RK*), MIRK (*Multi Implicit RK*), SIRK (*Singly Implicit RK*), SRK (*Symmetric RK*) i FIRK (*Fully Implicit RK*).

Metody bezpośrednie - schematy ekstrapolacyjne (ang. *extrapolation schemes*)

Algorytmy ekstrapolacyjne są efektywne w przypadku, gdy funkcja prawej strony równania różniczkowego jest aproksymowana wielomianami lub funkcjami wymiernymi. Ogólna idea schematów ekstrapolacyjnych polega na podziale przedziału całkowania na podprzedziały i wykorzystaniu do obliczeń dowolnej metody z serią różnych, stałych pod względem długości, kroków całkowania. Efektywna implementacja równoległa wymaga w tym przypadku specjalnych metod równoważenia obciążenia procesorów.

Metody bezpośrednie - metody blokowe (ang. *block methods*)

W metodach blokowych przedział całkowania dzielony jest na serie bloków, przy czym w każdym bloku należy wyznaczyć określoną liczbę punktów rozwiązania. Obliczenia

wykonywane są wewnątrz bloków i opierają się na wynikach obliczeń poprzednich *k*-bloków, nie zależą natomiast od wyników obliczeń z bieżącego bloku. W każdym bloku możliwe jest zatem przypisanie zadań obliczeniowych w każdym kroku do pojedynczego procesora. Obliczenia w ramach jednego bloku mogą być zatem wykonywane jednocześnie (równolegle).

Metody iteracyjne (ang. *iterative methods*) - **metody predyktor-korektor**

Metody wielokrokowe jawne mają tę zaletę, że wymagają minimalnego kosztu obliczeń w jednym kroku. Ich wadą są natomiast problemy ze stabilnością, powodujące konieczność zmniejszenia kroku *h*. Metody niejawne nie mają problemów ze stabilnością, więc *h* może być większe, ale wymagają dodatkowych obliczeń zmiennej niezależnej. Rozsądnym kompromisem pomiędzy tymi zaletami i wadami są metody predyktor-korektor. Wymagają one zazwyczaj większej liczby obliczeń prawej strony równania różniczkowego niż metody jawne, ale są od nich dokładniejsze. Ogólna idea równoległych metod predyktor-korektor polega na równoległym obliczaniu predyktora i korektora, ale dla różnych punktów rozwiązania. Wadą równoległych metod predyktor-korektor jest konieczność komunikacji w każdym kroku zmniejszająca wydajność oraz mały przedział stabilności w porównaniu z odpowiadającymi im metodami sekwencyjnymi.

Metoda iteracyjnej korekcji błędu (*ang. Iterated Defect Correction method - IDeC*)

Metoda IDeC jest to specjalna technika umożliwiająca przyspieszenie zbieżności, a tym samym dokładności, procedur rozwiązywania układów równań różniczkowych. W metodzie tej do otrzymania rozwiązania przybliżonego stosowana jest pewna metoda podstawowa. Następnie na podstawie tego rozwiązania konstruowany jest nowy układ (ang. *neighboring problem*), którego rozwiązanie dokładne jest znane. Układ ten rozwiązywany jest metodą podstawową. Na podstawie różnicy pomiędzy rozwiązaniem dokładnym a przybliżonym nowego układu szacowany jest nieznany błąd oryginalnego układu, wykorzystywany do korekcji (zwiększenia dokładności) podstawowego rozwiązania. Powyższe kroki powtarzane są aż do zwiększenia dokładności rozwiązania. Równoległe metody IDeC przedstawione są w pracy [85].

Opierają się one na podziale przedziału całkowania na podprzedziały i zastosowaniu algorytmu do kolejnych podprzedziałów lub na równoległości w ramach wykonania jednej iteracji.

4.4. Równoległość czasu (kroków)

W metodach realizujących równoległość czasu (kroków) (ang. *parallelism across time - steps*) prawa strona równania różniczkowego rozwiązywana jest jednocześnie w wielu punktach. Liczba jednocześnie obliczanych punktów jest stopniem równoległości metody. Do ich najbardziej znanych przedstawicieli należy metoda Picarda oraz *waveform relaxation method*.

Metoda Picarda

Metoda Picarda jest najprostszą metodą iteracyjną realizującą równoległość czasu. Metoda ta generuje sekwencję iteracyjnych rozwiązań. Równoległość w metodzie Picarda wprowadzana jest poprzez rozdzielenie układu na określoną liczbę równoległych zagadnień kwadraturowych. Wadą metody Picarda jest wolna zbieżność, którą poprawia się stosując np. podział przedziału całkowania na serię okien i zastosowanie procesu iteracyjnego oddzielnie dla każdego okna [16].

Metoda relaksacyjna

W przypadku dużych układów równań poszczególne zmienne stanu mogą mieć bardzo różniące się stałe czasowe. Rozwiązaniem tego problemu może być podział układu na bloki o podobnych stałych czasowych, które rozwiązywane są przy wykorzystaniu różnych metod i kroków całkowania, bardziej lub mniej niezależnie od siebie. Metoda realizująca powyższą koncepcję nazywana jest metodą relaksacyjną (ang. *waveform relaxation*) lub dynamiczną iteracją (ang. *dynamic iteration*). Metoda ta została oryginalnie opracowana do analizy dużych układów elektrycznych i tam jest najbardziej wydajna. W metodzie relaksacyjnej analizowany układ dekomponowany jest na kilka bloków. Następnie w całym przedziale całkowania wyznaczane jest rozwiązanie przybliżone. Od tego momentu każdy blok całkowany jest niezależnie

przez określona liczbę kroków (ang. *step sweep*). Po wykonaniu tych kroków następuje wymiana informacji pomiędzy blokami i ponownie wykonywana jest określona liczba kroków, itd. Obliczenia w każdym bloku wykonywane są przez oddzielny procesor. Istnieje kilka wariantów metody relaksacyjnej [16], [85]: WRJ (*Waveform Relaxation Jacobi*), WRGS (*Waveform Relaxation Gauss-Seidel*), WRSOR (*Waveform Relaxation Succesive Overrelaxation*) i WRN (*Waveform Relaxation Newton*).

4.5. Podsumowanie

Wybór metody równoległej analizy równań stanu powinien być dostosowany do docelowej architektury, na której będzie dana metoda zaimplementowana oraz do rozmiaru analizowanego problemu.

Algorytmy realizujące równoległość układu najbardziej wydajne są w przypadku rozwiązywania układów o dużej liczbie równań, gdyż wtedy obliczanie funkcji prawej strony równania różniczkowego jest kosztowne i zajmuje dużo czasu. W tego typu algorytmach informacje między procesorami wymieniane są w każdym kroku całkowania, zatem komunikacja między procesorami musi być bardzo szybka. Liczba procesorów, które mogą zostać wykorzystane w obliczeniach zależna jest od liczby równań rozwiązywanego układu.

W algorytmach realizujących równoległość metody także wymagane są systemy równoległe zapewniające bardzo szybką komunikację między procesorami, gdyż odbywa się ona w każdym kroku całkowania. Algorytmy te umożliwiają wykorzystanie niewielkiej liczby procesorów, toteż i osiągane przyspieszenia obliczeń nie są duże.

W trzecim typie algorytmów, równoległości czasu, nie jest wymagana bardzo szybka komunikacja między procesorami, gdyż nie jest ona związana z każdym krokiem całkowania. Z tego względu ten typ algorytmów najbardziej nadaje się do zastosowania w klastrach, będących obecnie jednym z najczęściej stosowanych systemów równoległych. Wydaje się, iż algorytmy realizujące równoległość czasu (kroków) mają największą przyszłość. W dalszej części rozprawy przedstawiona zostanie metoda spekulacyjna należąca do tej grupy algorytmów równoległych.

5. Systemy klaster

5.1. Ogólna charakterystyka systemów klaster

Od kilku lat obserwuje się duże zainteresowanie obliczeniami wykonywanymi przy wykorzystaniu systemów klastrowych. Klastry stały się interesującą alternatywą w stosunku do komputerów masywnie równoległych. Świadczy o tym chociażby fakt, iż na liście 500 systemów o największej mocy obliczeniowej [99], aż 360 są to systemy klaster, a ich liczba sukcesywnie wzrasta (listopad 2002 - 93 systemy, czerwiec 2003 - 149 systemów, listopad 2003 - 208 systemów, czerwiec 2004 - 291 systemów, listopad 2004 - 296 systemów, czerwiec 2005 - 304 systemy, listopad 2005 - 360 systemów). O dużej popularności klastrów zdecydowały m.in. [3]:

- niski koszt zakupu pojedynczego komputera klasy PC lub stacji roboczej,
- bardzo szybki wzrost mocy obliczeniowej pojedynczych komputerów oraz wzrost szybkości sieci komputerowych,
- możliwość wykorzystania do budowy klastra już dostępnych komputerów,
- prosta integracja i przystosowanie do istniejących sieci komputerowych,
- duża ilość bezpłatnego oprogramowania (biblioteki komunikacyjne, biblioteki numeryczne, kompilatory, itp.).

Klaster jest typem równoległego lub rozproszonego systemu składającego się z komputerów PC lub stacji roboczych połączonych siecią, a używanego jako pojedynczy, zintegrowany zespół obliczeniowy [2], [3]. Pojedynczy komputer wchodzący w skład klastra nazywany jest węzłem. Jako węzły w klastrach stosowane są komputery klasy PC lub stacje robocze pracujące pod kontrolą systemów operacyjnych Unix/Linux, a także, choć w znacznie mniejszym stopniu - Microsoft Windows. Węzły połączone są szybką siecią komputerową wykorzystującą standardowe protokoły stosowane w Internecie (*TCP/IP*) jak i protokoły o małych opóźnieniach, specjalnie stworzone do zastosowań w klastrach (*Active Messages*, *Fast Messages*, *VMMC - The Virtual Memory Mapped Communication System*, *U-net*, *BIP* -

Basic Interface for Parallelism) [4]. Wśród technologii sieciowych najczęściej stosowany jest *Ethernet* (*Fast Ethernet* - 100 Mb/s, *Gigabit Ethernet* - 1 Gb/s, *10 Gigabit Ethernet* - 10 Gb/s), charakteryzujący się niskim kosztem zakupu, ale i dużymi opóźnieniami w komunikacji, oraz *Myrinet* (wysoki koszt zakupu, małe opóźnienia). Do mniej popularnych technologii należą: *Giganet cLan* (1,25 Gb/s), *QsNet*, *ServerNet*, *SCI - Scalable Coherent Interface*, *ATM - Asynchronous Transmission Mode*, *Fibre Channel*, *HIPPI - High Performance Parallel Interface* i *ATOLL - Atomic Low Latency Network*. Komunikacja między węzłami realizowana jest przy wykorzystaniu jednej ze standardowych bibliotek: MPI (ang. *Message Passing Interface*) [77], [78] lub PVM (ang. *Parallel Virtual Machine*) [87].

Istnieje kilka klasyfikacji systemów klaster. Ze względu na przeznaczenie komputerów PC lub stacji roboczych tworzących klaster, wyróżniane są [72]:

- *klastry dedykowane* - węzły tworzące klaster są specjalnie przeznaczone do tego systemu, często umieszczone są w jednej dużej obudowie i pozbawione dodatkowych urządzeń takich jak monitory, klawiatury, myszy,
- *klastry niededykowane* - tworzone są przez komputery PC lub stacje robocze wykorzystywane normalnie do codziennej pracy, a obliczenia na nich wykonywane są w stanie bezczynności.

Ze względu na architekturę węzłów klastry dzielą się na [72]:

- *klastry komputerów PC* (NOWs - ang. *Network of Workstations*),
- *klastry stacji roboczych* (COWs - ang. *Cluster of Workstations*),
- *klastry komputerów SMP* (CLUMPs - ang. *Cluster of SMPs*).

Ze względu na konfigurację węzłów, wyróżniamy [72]:

- *klastry homogeniczne* - węzły tworzące klaster mają taką samą konfigurację sprzętową (typ procesora, ilość pamięci operacyjnej, płytę główną, dysk twardy) oraz pracują pod kontrolą tego samego systemu operacyjnego. Zazwyczaj *klastry dedykowane* są *klastrami homogenicznymi*,

- *klastry heterogeniczne* - są to klastry, których węzły mają różną konfigurację sprzętową oraz mogą pracować pod kontrolą różnych systemów operacyjnych.

Ze względu na system operacyjny zainstalowany na węzłach klastra, wyróżniamy [72]:

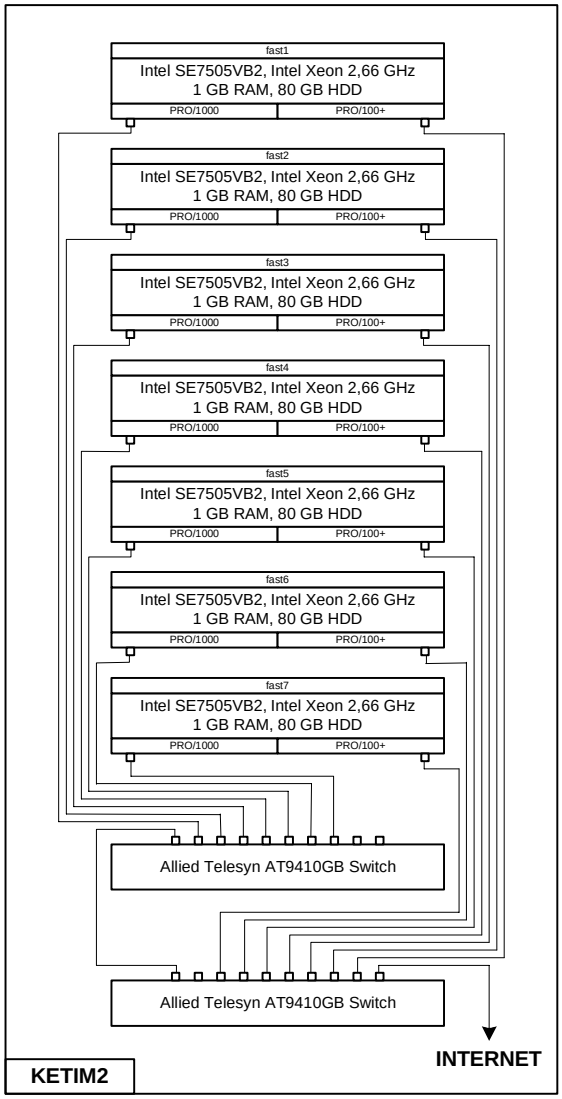
- klastry pracujące pod kontrolą systemu Linux: *Beowulf* [76] i *Mosix*,
- klastry pracujące pod kontrolą systemu Novel NetWare: *NCS - NetWare Cluster Services*,
- klastry pracujące pod kontrolą systemu Microsoft Windows NT: *Illinois HPVM*,
- klastry pracujące pod kontrolą systemu SUN Solaris: *Berkeley NOW*,
- klastry pracujące pod kontrolą systemu IBM AIX: *IBM SP2*,
- klastry pracujące pod kontrolą systemu HP UX: *Illinois-PANDA*,
- klastry pracujące pod kontrolą systemów: *Solaris MC, SCO UnixWare*.

5.2. Opis systemu klaster KETIM2

Klaster KETIM2 znajdujący się w Katedrze Elektrotechniki Teoretycznej i Metrologii składa się z siedmiu węzłów. Jest to klaster typu *Beowulf, homogeniczny* (wszystkie węzły klastra zbudowane są z takich samych podzespołów), *dedykowany* (przeznaczony jest tylko do prowadzenia badań naukowych). Na rys. 5.1 przedstawiony jest schemat ogólny klastra.

Każdy węzeł klastra posiada następującą konfigurację sprzętową:

- płyta główna Intel ServerBoard Vero Beach SE7505VB2 ATX (Socket 604x2 - do dwóch procesorów Intel Xeon, zintegrowane karty sieciowe: Intel PRO/100+ i Intel PRO/1000, zintegrowana karta graficzna ATI Rage XL 8 MB),
- procesor Intel Xeon 2,66 GHz,
- 1 GB pamięci RAM (2 x ECC DDR266 512 MB),
- dysk twardy 80 GB ATA/100,
- obudowa Intel SC5200.



Rys. 5.1 Schemat ogólny klastra KETIM2.

Węzły umieszczone są w szafie montażowej 19'' (rys. 5.2). Każdy węzeł posiada dwa interfejsy sieciowe (*Gigabit Ethernet* - 1 Gb/s i *Fast Ethernet* - 100 Mb/s) podłączone do dwóch Gigabitowych przełączników Allied Telesyn AT-9410GB. Interfejsy *Gigabit Ethernet* wykorzystywane są do komunikacji podczas obliczeń, natomiast interfejsy *Fast Ethernet* służą do kontroli i sterowania klastrem. Jeden węzeł pełni rolę węzła-serwera. Podłączony jest do niego monitor, klawiatura i mysz. Pozostałe węzły pozbawione są powyższych urządzeń. Na wszystkich węzłach zainstalowany jest system operacyjny Linux (dystrybucja Fedora Core 1) oraz system plików *NFS* (*Network File System*), dzięki czemu wszystkie węzły mają dostęp do tej samej przestrzeni dyskowej. Komunikacja między węzłami zrealizowana została przy

wykorzystaniu standardu MPI. Do tego celu zastosowano bibliotekę LAM/MPI v7.1.1 [14], [73], [94]. Biblioteka ta umożliwia zastosowanie standardu komunikacji MPI w programach w językach C, C++ i Fortran.

Rys. 5.2 Klaster KETIM2.

Na klastrze można pracować bezpośrednio wykorzystując monitor, klawiaturę i mysz podłączone do węzła-serwera lub też zdalnie - poprzez sieć Internet.

Wykonanie obliczeń na klastrze wymaga w pierwszej kolejności uruchomienia środowiska LAM/MPI. Służy do tego polecenie **`lamboot`**, dla którego jako parametr podaje się nazwę pliku tekstowego zawierającego konfigurację klastra. Po uruchomieniu środowiska LAM/MPI można przystąpić do kompilacji programu. Pliki źródłowe mogą być przygotowane bezpośrednio na klastrze lub na innym komputerze, a następnie przekopiowane na klaster. W przypadku języka C++ pliki źródłowe kompilowane są poleceniem **`mpiCC`**. Następnie należy uruchomić program poleceniem **`mpirun`** podając jako parametry liczbę węzłów, na których ma być uruchamiany

program oraz jego nazwę, np. **`mpirun -np 3 my_prog`** uruchamia program o nazwie **`my_prog`** na trzech węzłach. Poszczególne procesy uruchamiane są na węzłach klastra w takiej kolejności, jaka została określona w pliku konfiguracyjnym. Zakończenie pracy z systemem następuje po wykonaniu polecenia **`lamhalt`**.

6. Cele, teza i oryginalne elementy rozprawy

6.1. Cele pracy

Zasadniczym celem pracy jest:

Opracowanie równoległej metody numerycznej analizy stanów nieustalonych układów elektrycznych, umożliwiającej skrócenie czasu analizy stanu nieustalonego.

Opracowana metoda, nazywana metodą spekulacyjną, ma umożliwić otrzymanie numerycznego rozwiązania układu równań opisujących stan nieustalony, w przypadku, gdy jest to układ równań różniczkowych zwyczajnych, liniowych lub nieliniowych. Numeryczne rozwiązanie układu równań będzie odbywało się równolegle. Opracowując algorytm powyższej metody szczególną uwagę należy zwrócić na jego dostosowanie do wykorzystania w systemach równoległych nie posiadających bardzo szybkiej komunikacji między procesorami, np. w klastrach. Realizacja celu zasadniczego i przedstawionych powyżej założeń wymaga osiągnięcia czterech **celów cząstkowych**:

1. Opracowanie metody podziału czasu analizy stanu nieustalonego na podprzedziały, zapewniającej równomierne obciążenie procesorów liczących, w przypadku stosowania zmiennego kroku całkowania.
2. Opracowanie metod wyznaczania wartości zmiennych stanu na początku każdego podprzedziału, niezbędnych do równoległego przeprowadzenia obliczeń.
3. Opracowanie programów komputerowych implementujących algorytm równoległej metody numerycznej analizy stanów nieustalonych w sieci klaster komputerów PC.
4. Weryfikacja poprawności działania stworzonego oprogramowania na wybranych przykładach praktycznych.

6.2. Teza rozprawy

Realizacja wymienionych powyżej celów umożliwiła sformułowanie następującej tezy rozprawy:

Algorytm spekulacyjny umożliwia skrócenie czasu analizy stanów nieustalonych w układach elektrycznych.

6.3. Oryginalne elementy rozprawy

Z powyższymi celami i tezą związane są następujące elementy pracy, które autor uważa za oryginalne:

1. Opracowanie metody spekulacyjnej - równoległej metody numerycznej analizy stanów nieustalonych układów elektrycznych, opartej na dekompozycji przedziału całkowania.
2. Opracowanie zmodyfikowanej metody spekulacyjnej wymagającej mniejszej liczby węzłów obliczeniowych w porównaniu z metodą spekulacyjną.
3. Zastosowanie analizy rozwiązań przybliżonych do wyznaczania wartości zmiennych stanu na początku każdego podprzedziału.
4. Zastosowanie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą do wyznaczania wartości zmiennych stanu na początku każdego podprzedziału.
5. Zastosowanie analizy rozwiązań przybliżonych do podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały, który to podział zapewnia równomierne obciążenie procesorów.

Autor od dłuższego czasu zajmuje się realizacją zadań wynikających z powyższych celów pracy. W wyniku tych badań opublikowane zostały artykuły w czasopiśmie COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering oraz w materiałach konferencyjnych: PARELEC - International Conference on Parallel Computing in

Electrical Engineering, NATO Advanced Research Workshop - Concurrent Information Processing and Computing, IEEE-SPIE Symposium on Photonics and Web Engineering, Euro PVM/MPI - European Parallel Virtual Machine and Message Passing Interface Conference, SPETO - Międzynarodowa Konferencja z Podstaw Elektrotechniki i Teorii Obwodów, ZKwE - Zastosowania Komputerów w Elektrotechnice, OWD - Ogólnopolskie Warsztaty Doktoranckie i w innych [28]-[42], [54]-[56]. Wymienione publikacje tworzą cykl prac, na które powoływano się w niniejszej rozprawie.

7. Metoda spekulacyjna analizy stanów nieustalonych

7.1. Wstęp

Metoda spekulacyjna należy do grupy równoległych algorytmów numerycznych realizujących równoległość czasu (kroków). Ogólna jej idea opiera się na dekompozycji przedziału czasu, w którym analizowany jest stan nieustalony na podprzedziały. Wartości zmiennych stanu w poszczególnych podprzedziałach obliczane są równolegle (jednocześnie), dzięki czemu możliwa jest minimalizacja czasu potrzebnego do otrzymania rozwiązania końcowego. Wykonanie równolegle obliczeń wymaga wyznaczenia wartości zmiennych stanu na początku każdego podprzedziału. W metodzie spekulacyjnej zamiast jednej wartości, dla każdej zmiennej stanu wyznaczanych jest kilka wartości początkowych. Obliczenia w każdym podprzedziale wykonywane są zatem wielokrotnie, a z otrzymanego zbioru rozwiązań jedno wybierane jest jako rozwiązanie końcowe. Podobna koncepcja równoległego rozwiązywania zagadnień początkowych równań różniczkowych zwyczajnych została przedstawiona w pracy [79], jednakże nie była ona dalej rozwijana.

W metodzie spekulacyjnej rozwiązanie końcowe tworzone jest w taki sposób, że w każdym, kolejnym podprzedziale, wybierany jest ten przebieg, spośród kilku obliczonych, którego wartość początkowa (warunek początkowy) jest najbliższa wartości końcowej z poprzedniego podprzedziału. A zatem z wielu obliczonych rozwiązań wybierane jest tylko jedno. Stąd wzięła się nazwa prezentowanej metody - **metoda spekulacyjna**.

W rozdziale tym przedstawione zostaną podstawy teoretyczne metody spekulacyjnej, a następnie dwa jej algorytmy: metoda spekulacyjna ze **stałym krokiem całkowania**, w której obliczenia w podprzedziałach wykonywane są przy wykorzystaniu sekwencyjnej metody numerycznej ze stałą długością kroku całkowania i metoda spekulacyjna ze **zmiennym krokiem całkowania**, wykorzystująca sekwencyjną metodę numeryczną z automatyczną kontrolą długości kroku całkowania.

Jako przykład zastosowania obu metod przedstawiona zostanie analiza dynamiki silnika asynchronicznego pierścieniowego.

7.2. Podstawy teoretyczne metody spekulacyjnej

Metoda spekulacyjna znajduje zastosowanie w analizie stanów nieustalonych układów elektrycznych, w przypadku, gdy stan nieustalony opisany jest układem równań różniczkowych zwyczajnych nieliniowych:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{7.1}$$

lub liniowych:

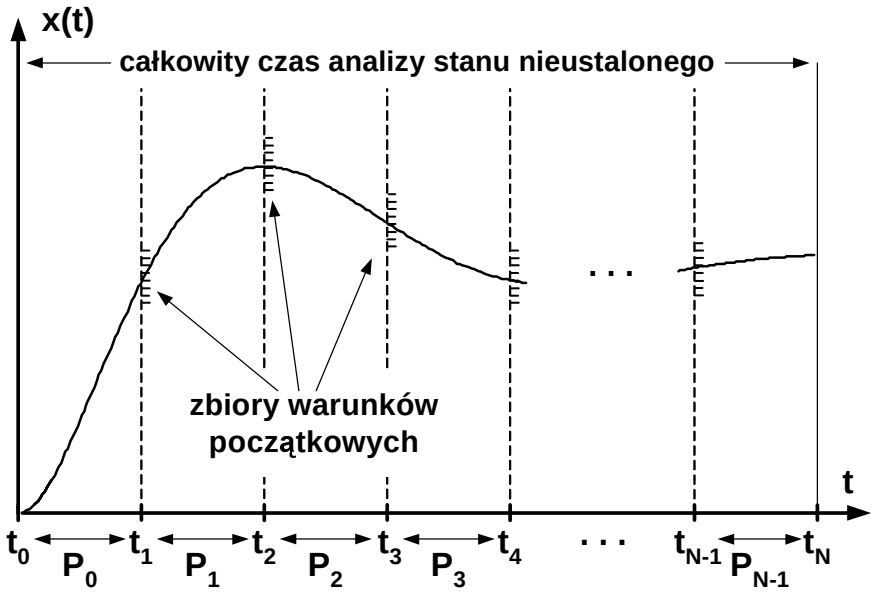
$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{7.2}$$

gdzie: $\mathbf{x} \in R^n$ jest wektorem zmiennych stanu, $\mathbf{f}(\mathbf{x}, \mathbf{u}, t)$ jest wektorem prawej strony (zbiorem funkcji nieliniowych), $\mathbf{A}, \mathbf{B}$ są macierzami stałych współczynników, **u** reprezentuje wektor wymuszeń, $\mathbf{x}(t_0) = \mathbf{x}_0$ jest wektorem warunków początkowych.

W celu dokładnego przedstawienia metody spekulacyjnej zostaną wprowadzone następujące definicje (rys. 7.1) [29], [36], [37]:

1. **Całkowity czas analizy stanu nieustalonego** - jest to przedział czasu (t_0, t_N), w którym analizowany jest stan nieustalony.
2. **Zbiór procesów równoległych** (**zbiór procesów, *P***) - jest to zbiór procesów w każdym podprzedziale $(t_i, t_{i+1}),\ i = 0,1,\ldots, N-1$ całkowitego czasu analizy stanu nieustalonego. Jako proces należy w tym przypadku rozumieć program realizujący algorytm obliczania wartości zmiennych stanu w jednym podprzedziale z określonym zbiorem warunków początkowych.
3. **Zbiór warunków początkowych** - jest to określona liczba warunków początkowych w jednym zbiorze procesów równoległych dla jednaj zmiennej stanu. Zbiory warunków początkowych określane są dla chwil czasowych $t_i,\ i = 1,2,\ldots, N-1$.

4. **Zbiór podstawowych warunków początkowych** $(\mathbf{x}(t_0) = \mathbf{x}_0)$ - jest to wektor warunków początkowych $\mathbf{x}_0$ w chwili czasowej t_0, znany z założenia.



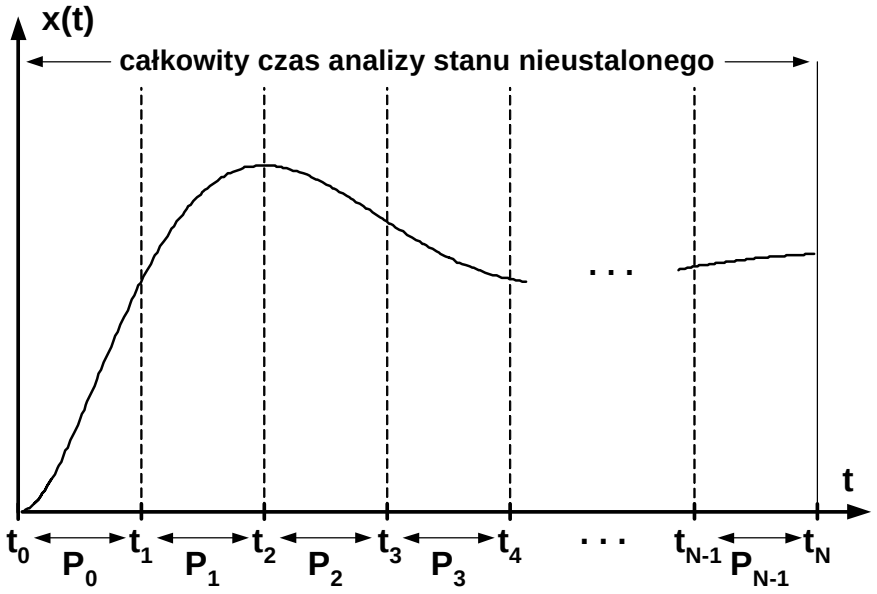
Rys. 7.1 Podstawowe definicje w metodzie spekulacyjnej.

7.3. Metoda spekulacyjna ze stałym krokiem całkowania

Algorytm metody spekulacyjnej ze stałym krokiem całkowania [28], [29], [31], [36], [54], [55], [56] składa się z dwóch głównych etapów: sekwencyjnego i równoległego. W etapie sekwencyjnym, na podstawie obliczeń przybliżonych wykonanych z dużym krokiem całkowania zapewniającym ich stabilność, wyznaczane są wartości zmiennych stanu na początku każdego podprzedziału. Natomiast w drugim etapie wykonywane są równolegle obliczenia w poszczególnych podprzedziałach oraz wyznaczane jest rozwiązanie końcowe.

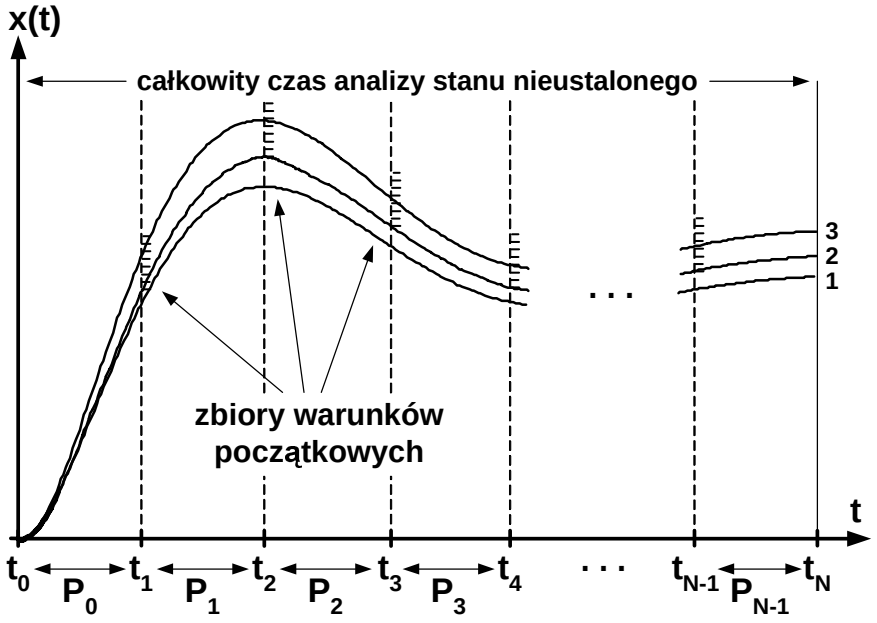
W metodzie spekulacyjnej ze stałym krokiem całkowania we wszystkich obliczeniach stosowane są sekwencyjne metody numeryczne rozwiązywania układów równań różniczkowych, jedno- lub wielokrokowe ze stałą długością kroku całkowania *h*. W niniejszym rozdziale algorytm spekulacyjny zostanie omówiony z wykorzystaniem metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania [7], [8], [9], [43], [44], [71]. Zastosowanie innej metody numerycznej wymaga wprowadzenia niewielkich zmian w algorytmie metody spekulacyjnej.

Pierwszy etap algorytmu prezentowanej metody wykonywany jest sekwencyjnie przez jeden procesor. Procesor ten nazywany jest procesorem nadrzędnym (procesorem *master*). W etapie tym całkowity czas analizy stanu nieustalonego, czyli przedział czasu (t_0, t_N), dzielony jest na określoną liczbę N równych pod względem długości podprzedziałów (t_i, t_{i+1}), $i = 0,1,\ldots,N-1$ (rys. 7.2). Liczba podprzedziałów zależna jest od liczby dostępnych procesorów, na których będą wykonywane obliczenia. Poszczególnym podprzedziałom przypisywane są zbiory procesów równoległych P_i, $i = 0,1,\ldots,N-1$.



Rys. 7.2 Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały w metodzie spekulacyjnej ze stałym krokiem całkowania.

Wykonanie równolegle obliczeń w drugim etapie algorytmu metody spekulacyjnej wymaga znajomości warunków początkowych na początku każdego podprzedziału, tj. w chwilach czasowych $t_0, t_1, \ldots, t_{N-1}$. W chwili czasowej t_0, tzn. na początku pierwszego podprzedziału (t_0, t_1), warunki początkowe znane są z założenia, gdyż jest to *zbiór podstawowych warunków początkowych*. Nieznane są natomiast warunki początkowe w pozostałych chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$. W metodzie spekulacyjnej zamiast jednego warunku początkowego, w powyższych chwilach czasowych, dla każdej zmiennej stanu wyznaczanych jest kilka warunków początkowych (*zbiór warunków początkowych*) (rys. 7.3).



Rys. 7.3 Wyznaczenie zbiorów warunków początkowych w metodzie spekulacyjnej ze stałym krokiem całkowania: 1 - rozwiązanie przybliżone otrzymane z krokiem h_1, 2 - rozwiązanie przybliżone otrzymane z krokiem h_2, 3 - nieznane rozwiązanie dokładne.

W celu wyznaczenia zbiorów warunków początkowych procesor *master* wykonuje obliczenia w całym przedziale (t_0, t_N) z dużą wartością kroku całkowania h_1, znajdującą się na granicy lub w pobliżu granicy stabilności sekwencyjnej metody numerycznej stosowanej do rozwiązania układu równań różniczkowych zwyczajnych. Otrzymywane jest w ten sposób pierwsze rozwiązanie przybliżone $x_{i,j,h1}$, $i = 1,2,\ldots,N-1$, $j = 1,2,\ldots,n$ (gdzie n jest liczbą zmiennych stanu). W przypadku metody Rungego-Kutty, gdy analizowany jest układ liniowy (7.2), warunek stabilności określa poniższa nierówność [71]:

$$\left|1 + h\lambda + \ldots + \frac{1}{p!}(h\lambda)^p\right| < 1, \qquad p = 1,2,3,4 \tag{7.3}$$

gdzie p jest rzędem metody Rungego-Kutty, h jest krokiem całkowania, zaś λ określa wartości własne macierzy **A** występującej po prawej stronie równania liniowego (7.2).

Warunek (7.3) powinien być spełniony dla wszystkich wartości własnych λ ($\mathrm{Re}\,\lambda < 0$) macierzy układu liniowego. Jeśli wartości własne są liczbami zespolonymi, to lewa strona nierówności (7.3) reprezentuje moduł liczby zespolonej.

W przypadku gdy rozważane równanie różniczkowe jest nieliniowe, to przed wyznaczeniem wartości kroku h_1 należy aproksymować je przez formę liniową (7.2), linearyzując, np. poprzez rozwinięcie w szereg Taylora wokół punktu równowagi [59]. Punkt równowagi może być warunkiem początkowym analizowanego układu lub jego stanem ustalonym. Wyznaczenie punktu równowagi wymaga rozwiązania układu nieliniowych równań algebraicznych, co w przypadku zastosowania metod iteracyjnych jest dodatkowym kosztem obliczeniowym. Jednakże etap ten należy traktować jako wstępne przygotowania do zastosowania metody spekulacyjnej, podobnie jak samą linearyzację układu nieliniowego i wyznaczenie kroku h_1 na podstawie warunku stabilności. Macierze **A** i **B** równania liniowego przyjmują następującą postać [59], [61]:

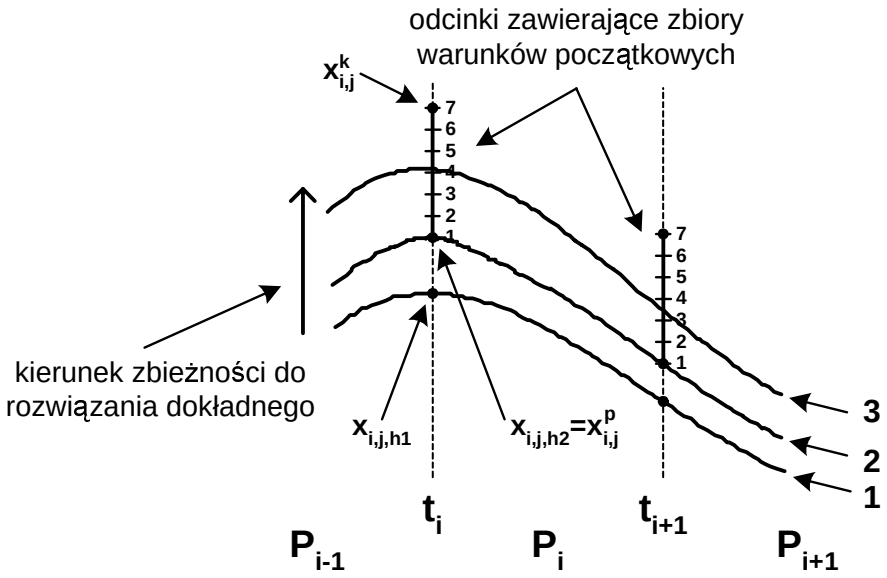
$$\mathbf{A} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\substack{x=x_{eq}\\u=u_{eq}}}, \qquad \mathbf{B} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\substack{x=x_{eq}\\u=u_{eq}}} \tag{7.4}$$

gdzie x_{eq} i u_{eq} są wartościami zmiennych stanu i wektora wymuszeń w punkcie równowagi. Operatory różniczkowania zdefiniowane są następująco [59], [61]:

$$\left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\substack{x=x_{eq}\\u=u_{eq}}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}_{\substack{x=x_{eq}\\u=u_{eq}}} \qquad \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\substack{x=x_{eq}\\u=u_{eq}}} = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_m} \end{pmatrix}_{\substack{x=x_{eq}\\u=u_{eq}}} \tag{7.5}$$

gdzie f_i i x_i, $(i = 1,2,\ldots,n)$ są składowymi wektora **f** i wektora **x**.

Po otrzymaniu pierwszego rozwiązania przybliżonego z krokiem h_1, procesor *master* powtarza te same obliczenia z krokiem h_2 spełniającym warunek $h_2 < h_1$ (h_1 powinno być wielokrotnością h_2, np. $h_2 = h_1 / 2$). Wynikiem tych obliczeń jest drugie rozwiązanie przybliżone $x_{i,j,h2}$, $i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n$. Otrzymane wartości zmiennych stanu z obu rozwiązań przybliżonych w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$ wykorzystywane są do wyznaczenia odcinków (oddzielnie dla każdej zmiennej stanu) zawierających zbiory warunków początkowych (rys. 7.4).



Rys. 7.4 Wyznaczenie odcinków zawierających zbiory warunków początkowych: 1 - rozwiązanie przybliżone otrzymane z krokiem h_1, 2 - rozwiązanie przybliżone otrzymane z krokiem h_2, 3 - nieznane rozwiązanie dokładne.

Jako długość każdego wyznaczanego odcinka ($\Delta x_{i,j}$) przyjmowana jest wartość bezwzględna różnicy wartości zmiennych stanu obliczonych z krokami h_2 i h_1 w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$, pomnożona przez, przyjmowany „a priori", współczynnik bezpieczeństwa *k*:

$$\Delta x_{i,j} = \left| x_{i,j,h2} - x_{i,j,h1} \right| \cdot k \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.6}$$

Zadaniem współczynnika bezpieczeństwa *k* jest takie określenie długości odcinka, aby zawierał się w nim dokładny (nieznany) warunek początkowy. Wartość tego współczynnika jest zależna od wielu czynników, toteż w celu jej oszacowania przeprowadzono szereg eksperymentów numerycznych polegających na rozwiązaniu analizowanych układów z różnymi krokami całkowania. Na podstawie tych eksperymentów stwierdzono, że w przypadku zastosowania metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania najbardziej odpowiednią wartością jest $k = 2{,}0$.

Po wyznaczeniu długości każdego odcinka określane jest jego położenie, czyli przedział wartości zmiennych stanu. Jako początek odcinka - $x^p_{i,j}$ (7.7) przyjmowana

jest wartość obliczona z krokiem h_2, zaś koniec odcinka - $x_{i,j}^k$ (7.8) jest to przesunięcie o jego długość ($\Delta x_{i,j}$) w kierunku zbieżności do rozwiązania dokładnego.

$$x_{i,j}^p = x_{i,j,h2}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.7}$$

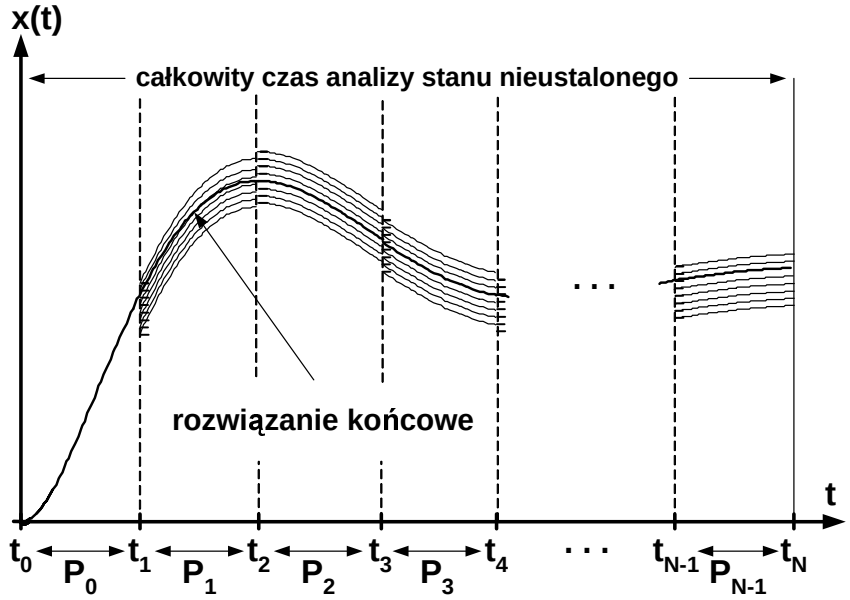
$$x_{i,j}^k = x_{i,j,h2} \pm \Delta x_{i,j}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.8}$$

Kierunek zbieżności do rozwiązania dokładnego określany jest na podstawie analizy wartości zmiennych stanu obliczonych z krokami h_1 i h_2. Zgodnie z definicją zbieżności metody numerycznej [71] jeśli równanie różniczkowe rozwiązywane jest wielokrotnie z coraz mniejszą długością kroku całkowania h ($h \to 0$) to otrzymany ciąg rozwiązań przybliżonych zbiega do wartości dokładnych. Zatem w rozpatrywanym przypadku, jeśli $x_{i,j,h1} \geq x_{i,j,h2}$, to we wzorze (7.8) występuje znak „-". Oznacza to, że rozwiązanie dokładne znajduje się „poniżej" rozwiązań przybliżonych $x_{i,j,h1}$ i $x_{i,j,h2}$. Jeśli zaś $x_{i,j,h1} < x_{i,j,h2}$, to we wzorze (7.8) występuje znak „+" i rozwiązanie dokładne znajduje się „powyżej" rozwiązań przybliżonych (por. rys. 7.4). W powyższych rozważaniach założono monotoniczną zbieżność ciągu rozwiązań przybliżonych do wartości dokładnych. Nie zawsze jednak tak będzie. Znajomość w każdym punkcie rozwiązania tylko dwóch wartości zmiennych stanu obliczonych z krokami h_1 i h_2 nie pozwala na stwierdzenie czy zbieżność jest monotoniczna. Rozwiązaniem tego problemu może być wykonanie jeszcze jednych obliczeń na całym przedziale (t_0, t_N) z krokiem h_3 ($h_3 < h_2$) i analiza wartości zmiennych stanu obliczonych z trzema krokami h_1, h_2 i h_3. Jeśli i w tym przypadku nie będzie można określić kierunku zbieżności, to jako warunek początkowy należy przyjąć wartość zmiennej stanu obliczoną z krokiem h_3.

Po wyznaczeniu dla każdej zmiennej stanu długości odcinka i jego położenia, procesor *master* dzieli odcinki na określoną liczbę przedziałów, których granice wyznaczają wartości zmiennej stanu **x** będące poszukiwanymi zbiorami warunków początkowych (rys. 7.4).

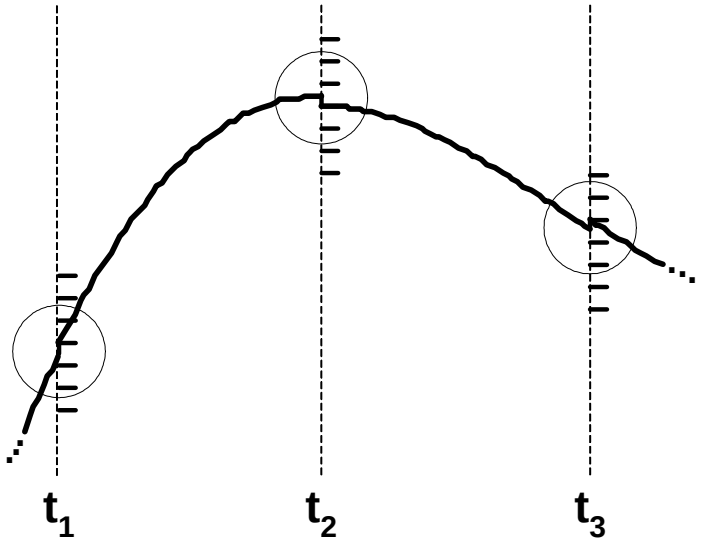
W drugim etapie algorytmu metody spekulacyjnej określona liczba procesorów wykonuje równolegle obliczenia w poszczególnych podprzedziałach. W pierwszym podprzedziale (t_0, t_1) warunki początkowe znane są z założenia (jest to *zbiór podstawowych warunków początkowych*), zatem obliczenia w tym podprzedziale wykonywane są tylko przez jeden procesor liczący - procesor *master*. W pozostałych podprzedziałach, $(t_1, t_2), \ldots, (t_{N-1}, t_N)$, obliczenia wykonywane są przez procesory *slave*. Liczba tych procesorów w każdym podprzedziale jest równa liczbie warunków początkowych w zbiorach warunków początkowych. Jeśli całkowity czas analizy stanu nieustalonego podzielony został na N podprzedziałów (zbiorów procesów równoległych), a w każdym (poza pierwszym) podprzedziale jest p warunków początkowych, to całkowita liczba procesorów wykorzystywanych w obliczeniach wynosi: $L = (N-1) \cdot p + 1$, gdzie dodatkowy jeden procesor jest to procesor *master*. W przypadku dużej liczby warunków początkowych oraz dużej liczby podprzedziałów wykorzystanie metody spekulacyjnej wymaga zastosowania systemu komputerowego zawierającego bardzo dużą liczbę procesorów, co można uznać za wadę tej metody.

Przed rozpoczęciem obliczeń w drugim etapie algorytmu metody spekulacyjnej procesor *master* przekazuje procesom *slave* odpowiednie dane (warunki początkowe, granice podprzedziałów, długość kroku całkowania *h*). Wszystkie procesory (*master* i *slave*) wykonują równolegle obliczenia w swoich podprzedziałach z małą długością kroku całkowania *h*, odpowiednią do analizowanego problemu. Po zakończeniu obliczeń w pierwszym podprzedziale wyznaczone jest tylko jedno rozwiązanie, zaś w pozostałych podprzedziałach - szereg rozwiązań mających różne warunki początkowe (rys. 7.5).



Rys. 7.5 Wyznaczenie rozwiązania końcowego w metodzie spekulacyjnej ze stałym krokiem całkowania.

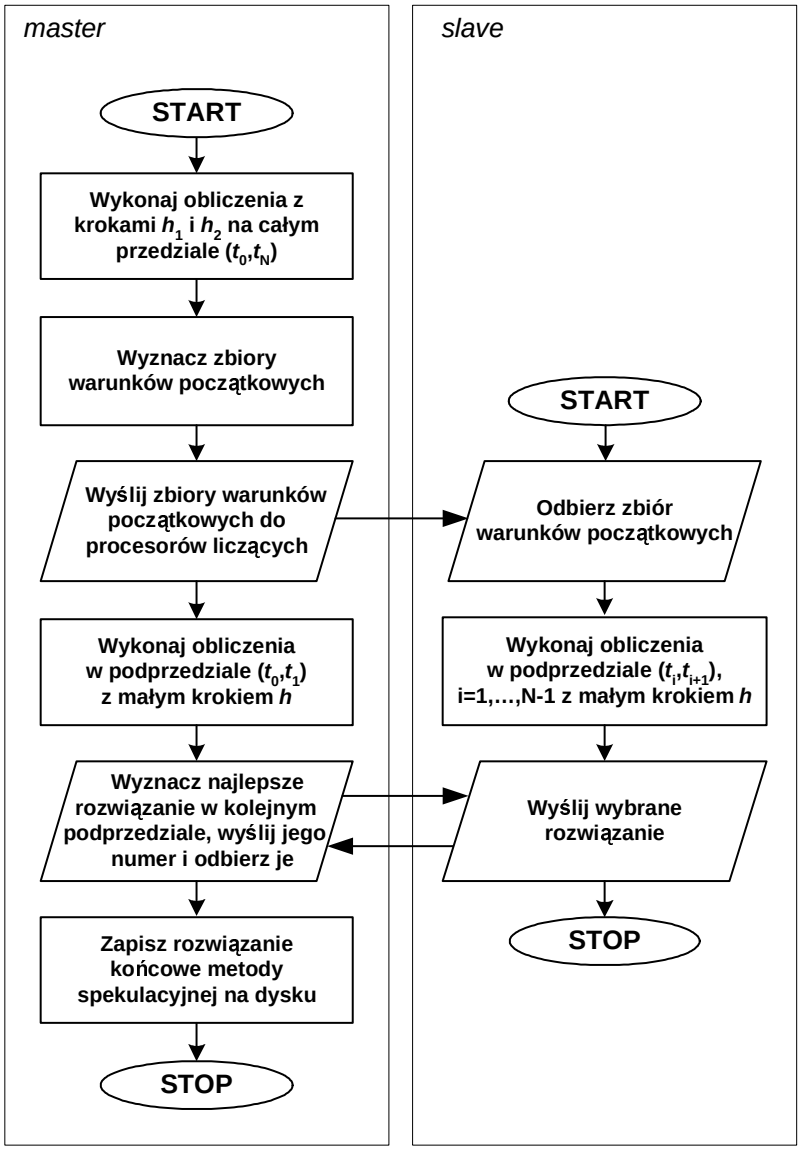
Po zakończeniu obliczeń procesor *master* przystępuje do określenia rozwiązania końcowego składającego się z wybranych przebiegów z poszczególnych podprzedziałów. W podprzedziale (t_0, t_1) dla każdej zmiennej stanu jest tylko jedno rozwiązanie wyznaczone z warunkami początkowymi znanymi z założenia. W kolejnym podprzedziale (t_1, t_2) procesor *master* wybiera to rozwiązanie, którego warunek początkowy jest najbliższy wartości w chwili czasowej t_1 otrzymanej podczas obliczeń w poprzednim podprzedziale (t_0, t_1) (rys. 7.6).



Rys. 7.6 Określenie rozwiązania końcowego w punktach będących granicami podprzedziałów.

W podprzedziale (t_2, t_3) wybierane jest to rozwiązanie, którego warunek początkowy jest najbliższy wartości w chwili czasowej t_2 będącej ostatnim punktem rozwiązania wybranego w poprzednim podprzedziale. W pozostałych podprzedziałach postępowanie przebiega w ten sam sposób, aż do osiągnięcia ostatniego punktu t_N.

Schemat blokowy algorytmu metody spekulacyjnej ze stałym krokiem całkowania przedstawiony jest na rys. 7.7. W obliczeniach wykorzystywany jest tylko jeden procesor *master* i $(N-1)\cdot p$ procesorów *slave* (N - liczba podprzedziałów, p - liczba warunków początkowych w jednym zbiorze warunków początkowych).



Rys. 7.7 Schemat blokowy algorytmu metody spekulacyjnej ze stałym krokiem całkowania.

7.4. Zastosowanie metody spekulacyjnej ze stałym krokiem całkowania do analizy dynamiki silnika asynchronicznego pierścieniowego

7.4.1. Wstęp

Jako przykład zastosowania metody spekulacyjnej ze stałym krokiem całkowania przedstawiona zostanie analiza dynamiki silnika asynchronicznego pierścieniowego [28], [56]. Stan nieustalony w tym silniku, reprezentowany przez przebieg prądów stojana i wirnika oraz przebieg prędkości kątowej wirnika, opisany jest układem pięciu nieliniowych równań różniczkowych zwyczajnych. Przykład ten został wybrany do testowania metody spekulacyjnej ze względu na interesującą dynamikę zmiennych stanu, których przebiegi w pewnych przedziałach czasu mają charakter oscylacyjny (są szybkozmienne), a w innych monotoniczny (są wolnozmienne).

7.4.2. Analiza dynamiki silnika asynchronicznego pierścieniowego

Analizowany silnik asynchroniczny pierścieniowy [7] ma następujące dane znamionowe:

$$\begin{array}{llll} P_N = 10\,kW, & n_N = 1445\,obr/min, & I_{1N} = 21A, & I_{2N} = 28\,A \\ U_N = 220V, & M_{KN}/M_N = 3{,}5, & \eta_N = 88{,}5\%, & \cos\varphi_N = 0{,}82 \\ p = 2, & \upsilon = z_s k_s / z_w k_w = 1{,}61 & & \end{array} \tag{7.9}$$

gdzie: P_N - moc znamionowa, n_N - prędkość obrotowa znamionowa, I_{1N} - znamionowy prąd uzwojenia stojana, I_{2N} - znamionowy prąd uzwojenia wirnika, U_N - napięcie znamionowe, M_{KN}/M_N - stosunek momentu znamionowego krytycznego do momentu znamionowego, η_N - sprawność, $\cos\varphi_N$ - znamionowy współczynnik mocy, p - liczba par biegunów uzwojeń stojana i wirnika, υ - przekładnia zwojowa, z_s - liczba zwojów uzwojenia stojana, z_w - liczba zwojów uzwojenia wirnika, k_s - współczynnik uzwojeń stojana, k_w - współczynnik uzwojeń wirnika.

Przebiegi przejściowe prądów stojana i wirnika oraz prędkości kątowej wirnika zostaną przedstawione w przypadku zasilania silnika z sieci symetrycznej. Równania

opisujące stan nieustalony silnika zostały sprowadzone do postaci standardowej poprzez transformację układu współrzędnych za pomocą macierzy ortogonalnej [7]. Transformacja ta upraszcza model matematyczny silnika oraz wprowadza stałe współczynniki w opisujących go równaniach [50]. Równania mają postać [7]:

$$\begin{aligned}
\frac{dx_1(t)}{dt} &= -a_1x_1 - a_5x_2 + a_4x_3 - b_4x_2x_5 - b_3x_4x_5 + e_1 \\
\frac{dx_2(t)}{dt} &= a_5x_1 - a_1x_2 + a_4x_4 + b_4x_1x_5 + b_3x_3x_5 + e_2 \\
\frac{dx_3(t)}{dt} &= a_2x_1 - a_3x_3 - a_5x_4 + b_2x_2x_5 + b_1x_4x_5 - e_3 \\
\frac{dx_4(t)}{dt} &= a_2x_2 + a_5x_3 - a_3x_4 - b_2x_1x_5 - b_1x_3x_5 - e_4 \\
\frac{dx_5(t)}{dt} &= -c_2x_5 + c_1x_1x_4 - c_1x_2x_3 - M_{obc} \\
x_i(0) &= 0, \quad i = 1,2,...,5
\end{aligned} \tag{7.10}$$

gdzie poszczególne zmienne stanu oznaczają:

$x_1(t)$, $x_2(t)$ - prądy stojana,

$x_3(t)$, $x_4(t)$ - prądy wirnika,

$x_5(t)$ - prędkość kątowa wirnika,

natomiast $x_i(0) = 0$ jest wektorem warunków początkowych, a M_{obc} - momentem obciążenia silnika.

Współczynniki występujące w równaniach (7.10) opisane są następującymi wzorami [7]:

$$\begin{aligned}
&a_1 = \frac{R_sL_w}{d}, && a_2 = \frac{R_sM}{d}, && a_3 = \frac{R_wL_s}{d}, && a_4 = \frac{R_wM}{d} \\
&b_1 = \frac{pL_sL_w}{d}, && b_2 = \frac{pL_sM}{d}, && b_3 = \frac{pL_wM}{d}, && b_4 = \frac{pM^2}{d} \\
&c_1 = \frac{pM}{J}, && c_2 = \frac{H}{J}, && c_3 = \frac{1}{J}, && a_5 = \omega, \\
&e_1 = \frac{L_wU_{w1}}{d}, && e_2 = \frac{L_wU_{w2}}{d}, && e_3 = \frac{MU_{w1}}{d}, && e_4 = \frac{MU_{w2}}{d} \\
&d = L_sL_w - M^2, && L_s = L_{rw} + \frac{3}{2}M_s, && L_w = L_{rs} + \frac{3}{2}M_w, && M = \frac{3}{2}M_{sw} \\
&U_{w1} = \frac{3}{2}U_{sm}\sin\beta, && U_{w2} = \frac{3}{2}U_{sm}\cos\beta
\end{aligned} \tag{7.11}$$

gdzie: R_s - rezystancja uzwojeń stojana, R_w - rezystancja uzwojeń wirnika, L_{rs} - indukcyjność rozproszenia uzwojeń stojana, L_{rw} - indukcyjność rozproszenia uzwojeń wirnika, M_s - indukcyjność własna uzwojeń stojana związana z głównym obwodem magnetycznym, M_w - indukcyjność własna uzwojeń wirnika związana z głównym obwodem magnetycznym, M_{sw} - indukcyjność wzajemna pomiędzy uzwojeniami stojana i wirnika w przypadku, gdy osie tych uzwojeń pokrywają się, p - liczba par biegunów uzwojeń stojana i wirnika, ω - prędkość kątowa wirnika, J - moment bezwładności sprowadzony do osi wirnika, H - współczynnik tarcia, U_{sm} - amplituda napięcia zasilającego, β - kąt przesunięcia fazowego napięcia zasilającego.

Dla silnika opisanego danymi znamionowymi (7.9) parametry obwodowe można oszacować następująco [7]:

$$\begin{aligned} &R_s = 0{,}3747\,\Omega, \qquad R_w = 0{,}112\,\Omega \\ &L_{rs} = 0{,}0023\,H, \qquad L_{rw} = 0{,}000887\,H \\ &M_s = 0{,}0475\,H, \qquad M_w = 0{,}01832\,H \\ &M_{sw} = 0{,}0295\,H, \quad U_{sm} = 220\sqrt{2}\,V, \qquad \beta = \frac{\pi}{2} \end{aligned} \tag{7.12}$$

Zakładając do obliczeń:

$$J = 1\,kgm^2, \quad M_{obc} = 5\,Nm, \quad H = 0{,}01\,Nms \tag{7.13}$$

współczynniki występujące w równaniu (7.10) przyjmują wartości:

$$\begin{aligned} &a_1 = 82{,}826, \quad a_2 = 129{,}201, \quad a_3 = 64{,}190, \quad a_4 = 38{,}619, \quad a_5 = 314{,}159, \\ &b_1 = 32{,}515, \quad b_2 = 50{,}722, \quad b_3 = 19{,}562, \quad b_4 = 30{,}515, \\ &c_1 = 0{,}0885, \quad c_2 = 0{,}01, \\ &e_1 = 72945, \quad e_2 = 72945, \quad e_3 = 113788, \quad e_4 = 113788, \\ &M_{obc} = 5 \end{aligned} \tag{7.14}$$

7.4.3. Zastosowanie metody spekulacyjnej

W pierwszym etapie algorytmu metody spekulacyjnej ze stałym krokiem całkowania wyznaczane są zbiory warunków początkowych. W tym celu wykonywane

są obliczenia z dużym krokiem całkowania h_1 znajdującym się na granicy stabilności lub w pobliżu granicy stabilności zastosowanej metody numerycznej rozwiązywania układów równań różniczkowych zwyczajnych. W przypadku analizy dynamiki silnika asynchronicznego pierścieniowego, jako sekwencyjną metodę numeryczną rozwiązywania układów równań różniczkowych zwyczajnych zastosowano metodę Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania. Warunek stabilności tej metody (7.3) określony jest dla równań liniowych, natomiast stan nieustalony w silniku asynchronicznym pierścieniowym opisany jest układem pięciu równań nieliniowych (7.10). Należy zatem układ nieliniowy aproksymować przez rozwinięcie w szereg Taylora wokół punktu równowagi. Jako punkt równowagi przyjmujemy stan ustalony. Wartości zmiennych stanu w stanie ustalonym wyznaczamy rozwiązując układ pięciu nieliniowych równań algebraicznych:

$$\begin{cases} -a_1 x_{1,eq} - a_5 x_{2,eq} + a_4 x_{3,eq} - b_4 x_{2,eq} x_{5,eq} - b_3 x_{4,eq} x_{5,eq} + e_1 = 0 \\ a_5 x_{1,eq} - a_1 x_{2,eq} + a_4 x_{4,eq} + b_4 x_{1,eq} x_{5,eq} + b_3 x_{3,eq} x_{5,eq} + e_2 = 0 \\ a_2 x_{1,eq} - a_3 x_{3,eq} - a_5 x_{4,eq} + b_2 x_{2,eq} x_{5,eq} + b_1 x_{4,eq} x_{5,eq} - e_3 = 0 \\ a_2 x_{2,eq} + a_5 x_{3,eq} - a_3 x_{4,eq} - b_2 x_{1,eq} x_{5,eq} - b_1 x_{3,eq} x_{5,eq} - e_4 = 0 \\ -c_2 x_{5,eq} + c_1 x_{1,eq} x_{4,eq} - c_1 x_{2,eq} x_{3,eq} - M_{obc} = 0 \end{cases} \tag{7.15}$$

gdzie $x_{i,eq}$ $(i = 1,2,\ldots,5)$ są poszukiwanymi wartościami zmiennych stanu w stanie ustalonym.

Następnie, zgodnie ze wzorem (7.5), wyznaczamy elementy macierzy **A** prawej strony równania liniowego. Macierz ta będzie miała postać:

$$\mathbf{A} = \begin{bmatrix} -a_1 & -a_5 - b_4 x_{5,eq} & a_4 & -b_3 x_{5,eq} & -b_4 x_{2,eq} - b_3 x_{4,eq} \\ a_5 + b_4 x_{5,eq} & -a_1 & b_3 x_{5,eq} & a_4 & b_4 x_{1,eq} + b_3 x_{3,eq} \\ a_2 & b_2 x_{5,eq} & -a_3 & -a_5 + b_1 x_{5,eq} & b_2 x_{2,eq} + b_1 x_{4,eq} \\ -b_2 x_{5,eq} & a_2 & a_5 - b_1 x_{5,eq} & -a_3 & -b_2 x_{1,eq} - b_1 x_{3,eq} \\ c_1 x_{4,eq} & -c_1 x_{3,eq} & -c_1 x_{2,eq} & c_1 x_{1,eq} & -c_2 \end{bmatrix} \tag{7.16}$$

Sprawdzając warunek stabilności metody Rungego-Kutty czwartego rzędu (7.3) dla różnych wartości kroku *h* otrzymano wyniki, które przedstawia tabela 7.1.

Tabela 7.1 Wartości lewej strony nierówności (7.3) dla różnych wartości kroku *h*.

h	**0,05**	**0,02**	**0,01**	**0,005**	**0,004**	**0,0025**	**0,002**	**0,001**
$h\lambda_1$	2121,253	44,54405	1,475468	**0,667053**	0,720564	0,809943	0,844115	0,918475
$h\lambda_2$	2121,253	44,54405	1,475468	**0,667053**	0,720564	0,809943	0,844115	0,918475
$h\lambda_3$	0,568179	0,497595	0,706884	**0,840830**	0,870498	0,916970	0,933006	0,965922
$h\lambda_4$	0,568179	0,497595	0,706884	**0,840830**	0,870498	0,916970	0,933006	0,965922
$h\lambda_5$	0,917860	0,346611	0,579792	**0,761210**	0,803897	0,872467	0,896601	0,946890

Warunek stabilności jest spełniony, gdy wartość lewej strony nierówności (7.3) jest mniejsza od jedności dla wszystkich wartości $h\lambda_i$. W analizowanym przykładzie sytuacja taka występuje dla $h = 5 \cdot 10^{-3}\, s$, a zatem w metodzie spekulacyjnej do wyznaczenia zbiorów warunków początkowych przyjęto krok całkowania $h_1 = 5 \cdot 10^{-3}\, s$ i $h_2 = 10^{-3}\, s$. Pozostałe parametry obliczeń były następujące:

- sekwencyjna metoda numeryczna rozwiązywania układów równań różniczkowych zwyczajnych: metoda Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania *h*,
- całkowity czas analizy stanu nieustalonego (t_0, t_N): $0{,}0 \div 1{,}25\, s$,
- podział całkowitego czasu analizy stanu nieustalonego na 5 podprzedziałów,
- obliczenia właściwe z podstawowym krokiem całkowania: $h = 10^{-7}\, s$,
- liczba warunków początkowych w jednym podprzedziale: 10,
- współczynnik bezpieczeństwa: $k = 2{,}0$.

Podział całkowitego czasu analizy stanu nieustalonego na 5 podprzedziałów wynikał z liczby dostępnych procesorów oraz metody wykonywania obliczeń. Długość kroku całkowania $h = 10^{-7}\, s$ została wyznaczona poprzez założenie, iż maksymalna różnica pomiędzy wartościami zmiennych stanu obliczonymi z kolejnymi krokami całkowania h_j i h_{j-1} $(h_j < h_{j-1})$, różniącymi się o rząd wielkości, jest mniejsza od założonej wartości $eps = 10^{-6}$ [56]:

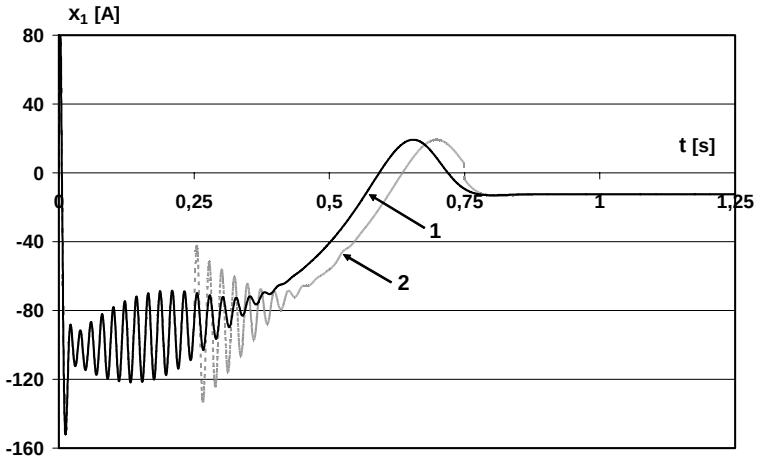
$$\left| x_{i,h_j} - x_{i,h_{j-1}} \right|_{\max} < eps, \quad i = 1,2,\ldots,5 \tag{7.17}$$

W rozpatrywanym przykładzie powyższy warunek jest spełniony dla kroku $h = 10^{-6}\, s$, natomiast w celu zwiększenia dokładności obliczeń przyjęto krok $h = 10^{-7}\, s$.

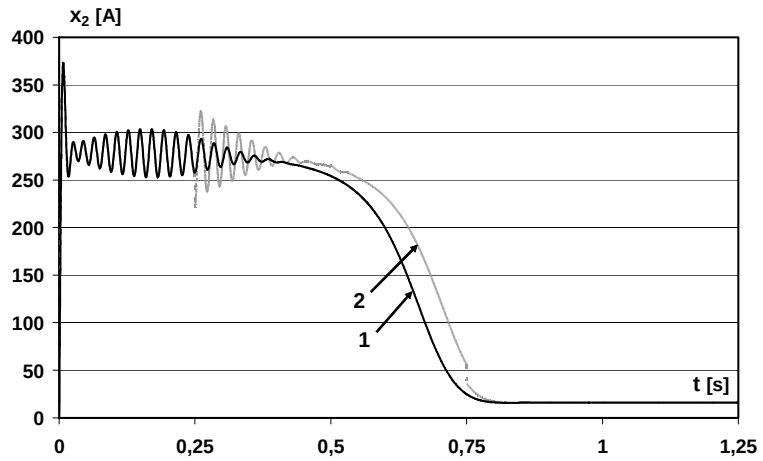
Liczba warunków początkowych w jednym podprzedziale jest kompromisem pomiędzy dokładnością wyznaczenia warunku początkowego (im większa liczba, tym dokładniejsze wyznaczenie warunku początkowego), a liczbą procesorów stosowanych w obliczeniach (im więcej warunków, tym więcej procesorów). Jak już wspomniano w opisie metody spekulacyjnej, w przypadku zastosowania metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania, najbardziej odpowiednią wartością współczynnika bezpieczeństwa jest $k = 2{,}0$.

7.4.4. Wyniki obliczeń

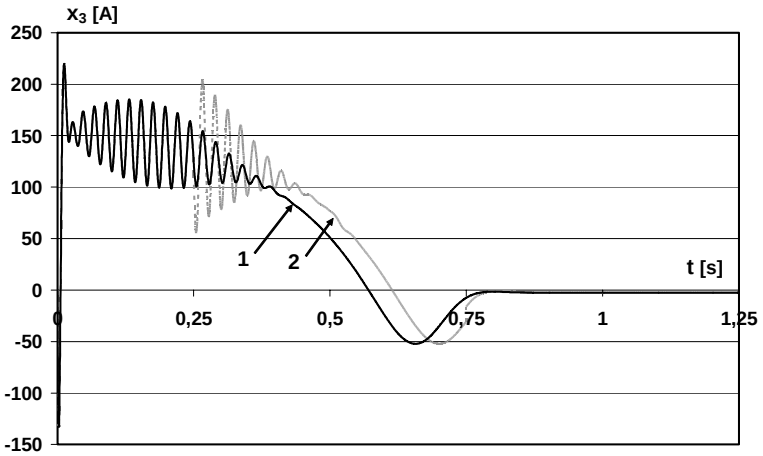
Obliczenia przeprowadzono na pięciu węzłach klastra KETIM2, który został opisany w rozdziale 5.2. Otrzymane przebiegi poszczególnych zmiennych stanu przedstawiono na rys. 7.8 - rys. 7.12.



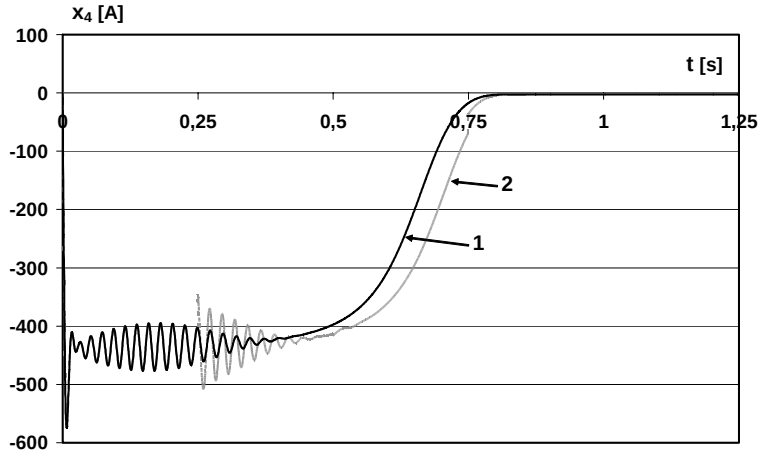
Rys. 7.8 Przebieg prądu stojana $x_1(t)$ w układzie współrzędnych wirnika: 1 - rozwiązanie sekwencyjne i rozwiązanie spekulacyjne, 2 - najgorsze rozwiązanie spekulacyjne.



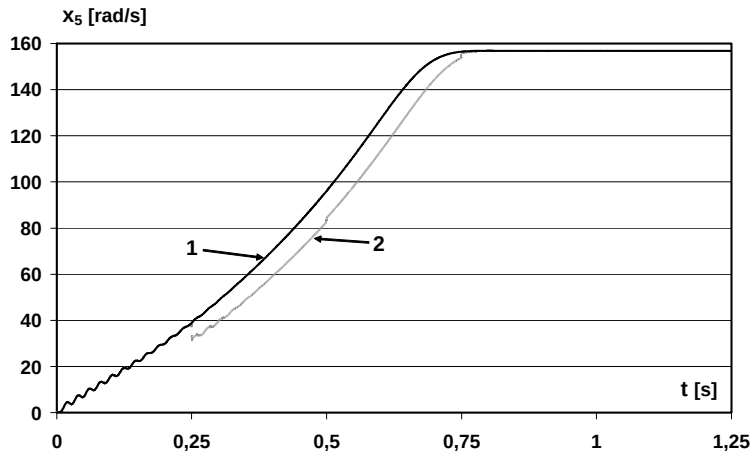
Rys. 7.9 Przebieg prądu stojana $x_2(t)$ w układzie współrzędnych wirnika: 1 - rozwiązanie sekwencyjne i rozwiązanie spekulacyjne, 2 - najgorsze rozwiązanie spekulacyjne.



Rys. 7.10 Przebieg prądu wirnika - $x_3(t)$ w układzie współrzędnych wirnika: 1 - rozwiązanie sekwencyjne i rozwiązanie spekulacyjne, 2 - najgorsze rozwiązanie spekulacyjne.



Rys. 7.11 Przebieg prądu wirnika - $x_4(t)$ w układzie współrzędnych wirnika: 1 - rozwiązanie sekwencyjne i rozwiązanie spekulacyjne, 2 - najgorsze rozwiązanie spekulacyjne.



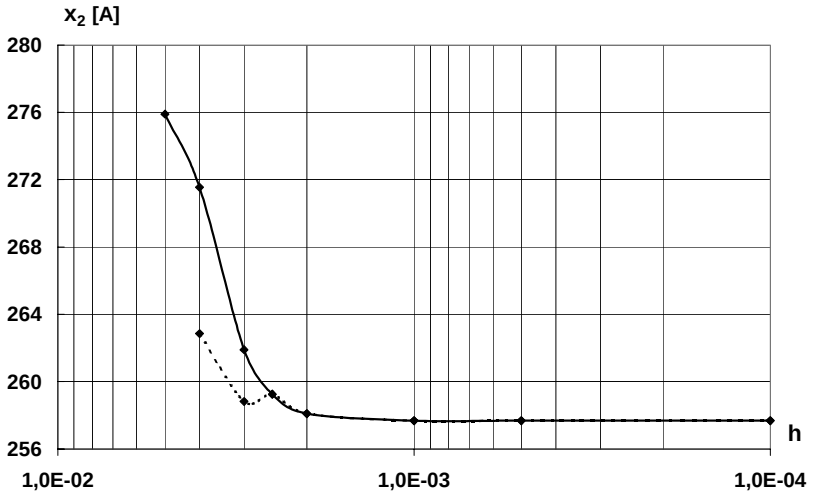
Rys. 7.12 Przebieg prędkości kątowej wirnika - $x_5(t)$; 1 - rozwiązanie sekwencyjne i rozwiązanie spekulacyjne, 2 - najgorsze rozwiązanie spekulacyjne.

Na powyższych rysunkach krzywa 1 reprezentuje rozwiązanie sekwencyjne i rozwiązanie metody spekulacyjnej (przebiegi pokrywają się w zakresie szerokości linii wykresu), zaś krzywa 2 - najgorsze rozwiązanie metody spekulacyjnej. Rozwiązanie sekwencyjne jest to rozwiązanie układu (7.10) przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-7}\ s$, taką samą jak w metodzie spekulacyjnej.

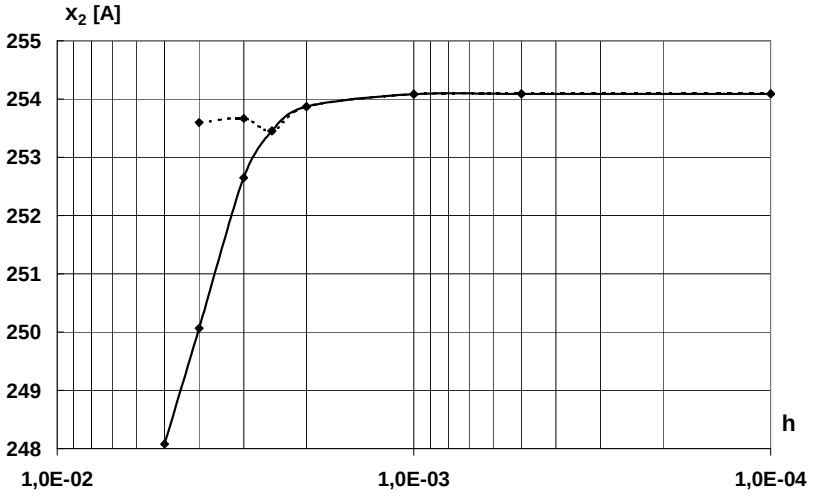
Najgorsze rozwiązanie metody spekulacyjnej jest to rozwiązanie otrzymane w przypadku, gdy przy określaniu rozwiązania końcowego, w kolejnych podprzedziałach wybierano przebieg, którego warunek początkowy jest najbardziej oddalony od ostatniej wartości z poprzedniego podprzedziału.

W celu scharakteryzowania zbieżności i dokładności rozwiązania metody spekulacyjnej przeprowadzono szereg eksperymentów numerycznych. Zbadano zbieżność sekwencyjnego algorytmu metody Rungego-Kutty i metody spekulacyjnej w chwilach czasowych, będących punktami początkowymi poszczególnych podprzedziałów. Badając zbieżność sekwencyjnego algorytmu metody Rungego-Kutty, układ (7.10) rozwiązano wielokrotnie w całym przedziale czasu (t_0, t_N) z różnymi wartościami kroku całkowania *h*. W przypadku metody spekulacyjnej badanie jej zbieżności polegało także na wielokrotnym rozwiązaniu układu (7.10), ale przy założeniu stałej długości kroków całkowania $h_1 = 5 \cdot 10^{-3}\ s$ i $h = 10^{-7}\ s$ oraz przy

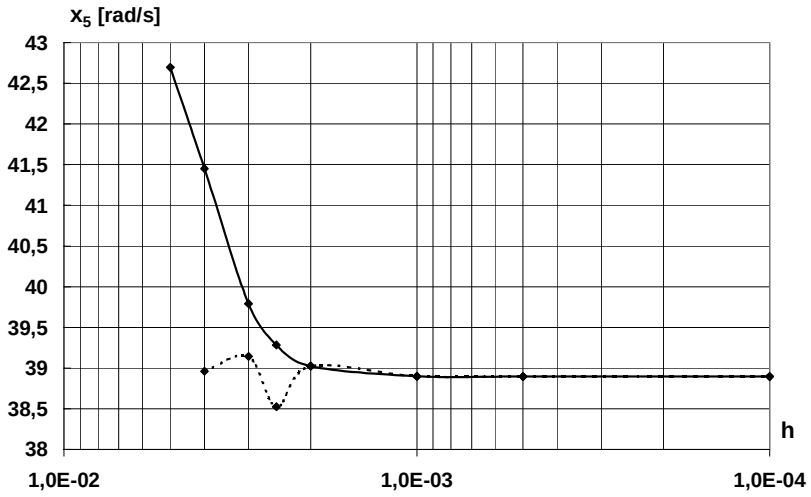
zmianie długości kroku całkowania h_2. Otrzymane wykresy zbieżności dla zmiennych stanu $x_2(t)$ i $x_5(t)$ w chwilach czasowych $t_1 = 0{,}25\,s$ i $t_2 = 0{,}5\,s$ przedstawiono na rys. 7.13 - rys. 7.16. Jako długość kroku h w algorytmie sekwencyjnym i kroku h_2 w algorytmie spekulacyjnym przyjmowano wartości umieszczone na osiach odciętych wykresów zbieżności.



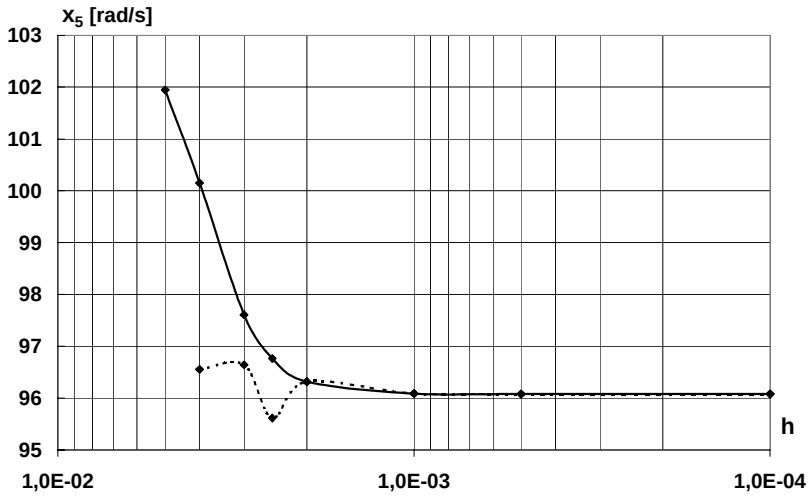
Rys. 7.13 Zbieżność metody Rungego-Kutty (linia ciągła) i metody spekulacyjnej (linia przerywana) dla zmiennej stanu $x_2(t)$ w chwili czasowej $t_1 = 0{,}25\,s$.



Rys. 7.14 Zbieżność metody Rungego-Kutty (linia ciągła) i metody spekulacyjnej (linia przerywana) dla zmiennej stanu $x_2(t)$ w chwili czasowej $t_2 = 0{,}5\,s$.



Rys. 7.15 Zbieżność metody Rungego-Kutty (linia ciągła) i metody spekulacyjnej (linia przerywana) dla zmiennej stanu $x_5(t)$ w chwili czasowej $t_1 = 0{,}25\,s$.



Rys. 7.16 Zbieżność metody Rungego-Kutty (linia ciągła) i metody spekulacyjnej (linia przerywana) dla zmiennej stanu $x_5(t)$ w chwili czasowej $t_2 = 0{,}5\,s$.

Analiza wykresu zbieżności metody Rungego-Kutty w punktach będących początkami podprzedziałów umożliwia sprawdzenie, czy wybrana do obliczeń w metodzie spekulacyjnej wartość współczynnika bezpieczeństwa k została prawidłowo określona. Wykres zbieżności zawiera wartości zmiennych stanu w jednej chwili czasowej, obliczone z różnymi krokami całkowania, a w tym z krokami h_1 i h_2. Kroki te stosowane są do wyznaczenia rozwiązań przybliżonych, wykorzystywanych do określenia odcinków zawierających zbiory warunków początkowych. Różnica wartości zmiennych stanu obliczonych z krokami h_1 i h_2 pomnożona przez

współczynnik *k* określa położenie końca powyższego odcinka. Analiza wykresu zbieżności metody Rungego-Kutty pozwala zatem sprawdzić, czy odcinek ten będzie obejmował także wartość, do której dąży rozwiązanie przy zmniejszaniu długości kroku całkowania. Badanie zbieżności metody Rungego-Kutty nie jest częścią algorytmu metody spekulacyjnej, gdyż wymagałoby wielokrotnego rozwiązania analizowanego układu z różnymi krokami całkowania przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty.

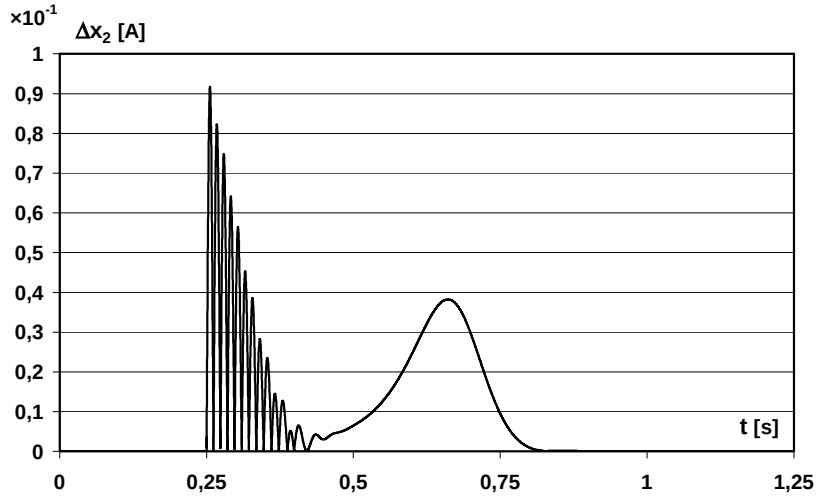
Błąd bezwzględny metody spekulacyjnej (err_{TOT}) jest zależny od błędu metody Rungego-Kutty z krokiem *h* (err_{RK}) i błędu wynikającego z wyznaczenia warunków początkowych w metodzie spekulacyjnej (err_{MSP}):

$$err_{TOT} = f(err_{RK}, err_{MSP}) \tag{7.18}$$

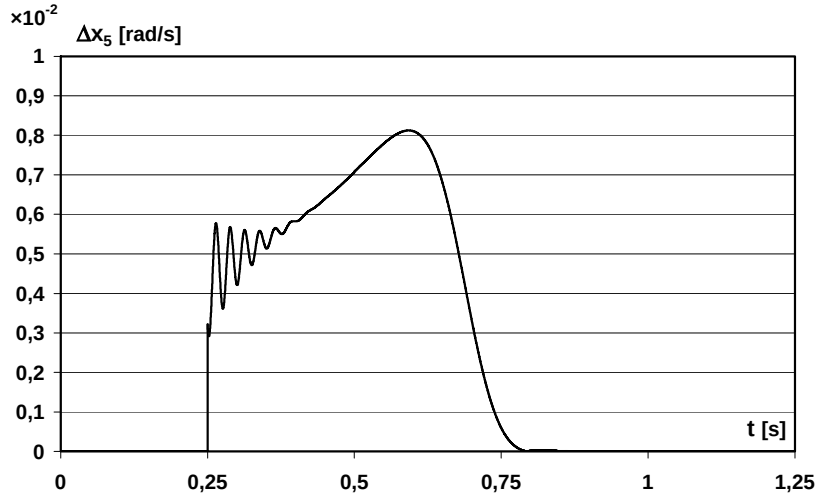
Na rys. 7.17 i rys. 7.18 przedstawiono błędy bezwzględne Δx_i (7.19) metody spekulacyjnej dla zmiennych stanu $x_2(t)$ i $x_5(t)$.

$$\Delta x_i = \left| x_{i,RK} - x_{i,MSP} \right|, \quad i = 1,2,\ldots,5 \tag{7.19}$$

gdzie: $x_{i,RK}$ jest rozwiązaniem sekwencyjnym otrzymanym przy wykorzystaniu metody Rungego-Kutty ze stałą długością kroku całkowania $h = 10^{-7}\,s$, zaś $x_{i,MSP}$ jest rozwiązaniem metody spekulacyjnej ($h_1 = 5 \cdot 10^{-3}\,s$, $h_2 = 10^{-3}\,s$, $h = 10^{-7}\,s$).



Rys. 7.17 Błędy bezwzględne metody spekulacyjnej dla zmiennej stanu $x_2(t)$.



Rys. 7.18 Błędy bezwzględne metody spekulacyjnej dla zmiennej stanu $x_5(t)$.

Ponieważ w pierwszym podprzedziale $(0 \div 0{,}25)$ obliczenia w metodzie spekulacyjnej wykonywane są z warunkami początkowymi znanymi z założenia oraz z krokiem całkowania takim samym jak w algorytmie sekwencyjnym, więc wyznaczone błędy bezwzględne w tym podprzedziale są równe zeru.

W tabeli 7.2 zestawiono maksymalne wartości błędów bezwzględnych (7.20) rozwiązania metody spekulacyjnej, określone dla poszczególnych zmiennych stanu.

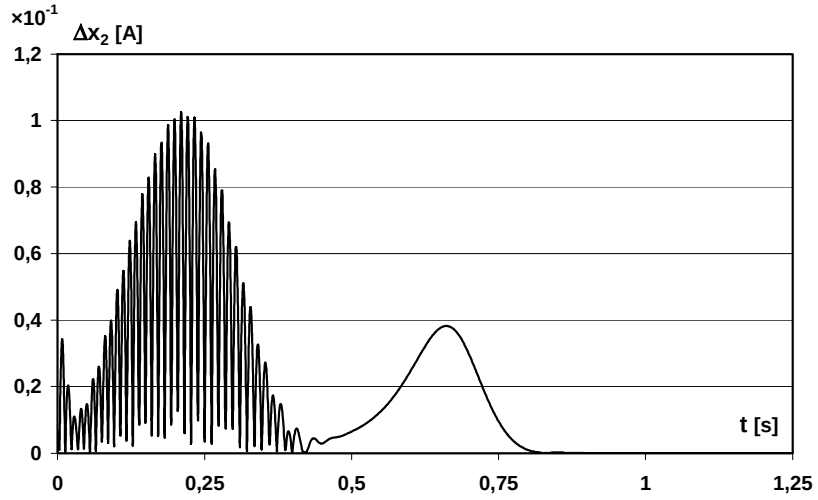
$$\Delta_{i,\max} = \left\| x_{i,RK} - x_{i,MSP} \right\|_{\max}, \quad i = 1,2,\ldots,5 \tag{7.20}$$

gdzie: $x_{i,RK}$ jest rozwiązaniem sekwencyjnym, a $x_{i,MSP}$ jest rozwiązaniem metody spekulacyjnej.

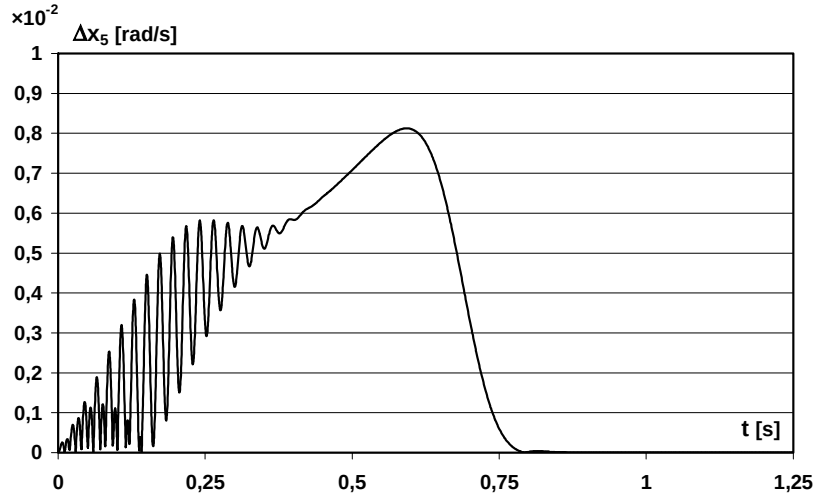
Tabela 7.2 Maksymalne błędy bezwzględne (7.20) rozwiązania metody spekulacyjnej

$\Delta_{1,\max}$	$\Delta_{2,\max}$	$\Delta_{3,\max}$	$\Delta_{4,\max}$	$\Delta_{5,\max}$
0,0927	0,0917	0,1501	0,1490	0,0081

Otrzymane wartości błędów bezwzględnych porównano z błędami bezwzględnymi metody Rungego-Kutty z krokiem całkowania $h = 10^{-3}\ s$ (rys. 7.19 i rys. 7.20). Stwierdzono, że błędy metody spekulacyjnej spełniają warunek $err_{TOT} < err_{RK}$ dla kroku $h = 10^{-3}\ s$ w metodzie Rungego-Kutty.



Rys. 7.19 Błędy bezwzględne metody Rungego-Kutty z krokiem $h = 10^{-3}\ s$ dla zmiennej stanu $x_2(t)$.

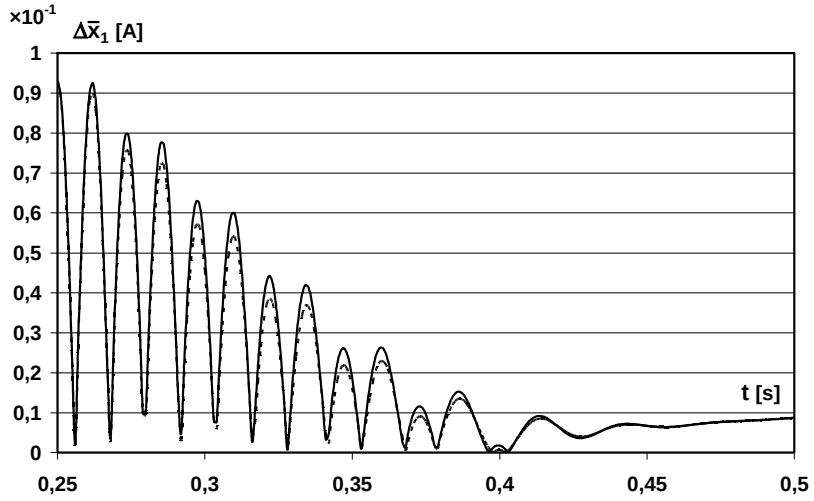


Rys. 7.20 Błędy bezwzględne metody Rungego-Kutty z krokiem $h = 10^{-3}\ s$ dla zmiennej stanu $x_5(t)$.

Przeprowadzając ilościową analizę dokładności rozwiązania metody spekulacyjnej zdefiniowano błąd $\overline{\Delta}_j$ w całym przedziale czasu (t_0, t_N) i w poszczególnych podprzedziałach $(t_j, t_{j+1}),\ j = 0,1,\ldots,N-1$. Błąd ten ma następującą wartość:

$$\overline{\Delta}_j = \frac{1}{(t_{j+1} - t_j)} \int_{t_j}^{t_{j+1}} \left| x_{i,RK} - x_{i,MSP} \right| dt, \quad i = 1,2,\ldots,5 \tag{7.21}$$

gdzie: $x_{i,RK}$ jest rozwiązaniem sekwencyjnym dla zmiennej stanu x_i, zaś $x_{i,MSP}$ jest rozwiązaniem spekulacyjnym ($h_1 = 5 \cdot 10^{-3}\, s$, $h_2 = 10^{-3}\, s$, $h = 10^{-7}\, s$). Wykres błędu $\overline{\Delta}_1$ w podprzedziale (t_1, t_2) przedstawiono na rys. 7.21.



Rys. 7.21 Błąd $\overline{\Delta}_1$ metody Rungego-Kutty (linia ciągła) dla $h = 10^{-3}\, s$ i błąd metody spekulacyjnej (linia przerywana) dla $h_1 = 5 \cdot 10^{-3}\, s$, $h_2 = 10^{-3}\, s$ i $h = 10^{-7}\, s$ dla zmiennej stanu $x_1(t)$ w podprzedziale (t_1, t_2).

Otrzymane wartości błędów dla metody Rungego-Kutty ($h = 10^{-3}\, s$) i metody spekulacyjnej ($h_1 = 5 \cdot 10^{-3}\, s$, $h_2 = 10^{-3}\, s$, $h = 10^{-7}\, s$), policzone według wzoru (7.21), w wybranych podprzedziałach i w całym przedziale (t_0, t_N) zestawiono w tabeli 7.3.

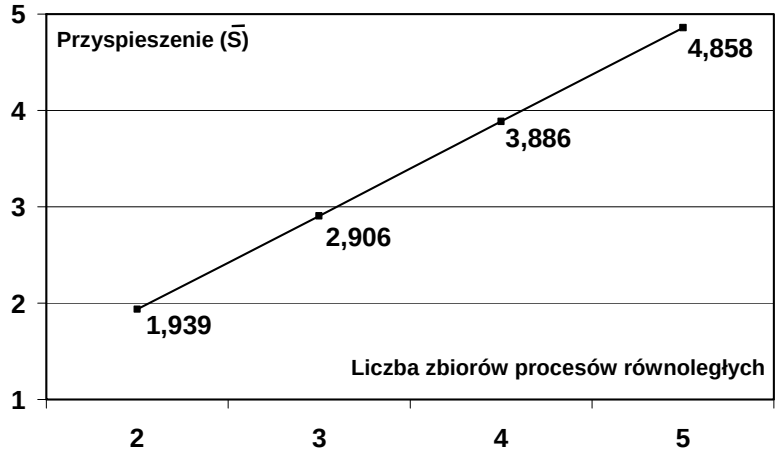
Tabela 7.3 Błędy metody Rungego-Kutty ($\overline{\Delta}_{RK}$) i metody spekulacyjnej ($\overline{\Delta}_{MSP}$).

	(0,0 ÷ 0,25 s)		***(0,25 ÷ 0,5 s)***		***(0,5 ÷ 1,25 s)***		***(0,0 ÷ 1,25 s)***	
	$\overline{\Delta}_{RK}$	$\overline{\Delta}_{MSP}$	$\overline{\Delta}_{RK}$	$\overline{\Delta}_{MSP}$	$\overline{\Delta}_{RK}$	$\overline{\Delta}_{MSP}$	$\overline{\Delta}_{RK}$	$\overline{\Delta}_{MSP}$
x_1	0,039076	0	0,021171	0,019708	0,003086	0,003086	0,063333	0,022794
x_2	0,039031	0	0,019743	0,018244	0,007911	0,00791	0,066685	0,026154
x_3	0,063457	0	0,034402	0,032042	0,005142	0,005143	0,103001	0,037184
x_4	0,063245	0	0,032176	0,029763	0,01312	0,013119	0,108541	0,042882
x_5	0,002004	0	0,005627	0,005614	0,002021	0,002021	0,009652	0,007636

Otrzymane wyniki pokazują, że zarówno w poszczególnych podprzedziałach, jak i w całym przedziale (t_0, t_N) błędy metody spekulacyjnej są mniejsze od błędów

metody Rungego-Kutty ($h=10^{-3}\ s$). Szczególnie duże różnice można zaobserwować w przypadku całego przedziału (t_0, t_N).

Podczas eksperymentów numerycznych wyznaczono przyspieszenie obliczeń. Wykorzystano do tego celu od 1 do 5 węzłów systemu klaster KETIM2, który został opisany w rozdziale 5.2. W algorytmie metody spekulacyjnej zastosowano parametry obliczeń przedstawione w rozdziale 7.4.3 ($h_1=5\cdot10^{-3}\ s$, $h_2=10^{-3}\ s$, $h=10^{-7}\ s$), natomiast obliczenia na jednym procesorze przeprowadzono stosując sekwencyjny algorytm metody Rungego-Kutty czwartego rzędu, ze stałym krokiem całkowania $h=10^{-7}\ s$. Otrzymane wartości przyspieszenia przedstawione są na rys. 7.22.



Rys. 7.22 Przyspieszenie obliczeń w metodzie spekulacyjnej.

Zastosowana w tym przypadku definicja przyspieszenia różni się od definicji stosowanej w klasycznych obliczeniach równoległych [64], [72]. Jeśli *t*(1) jest czasem obliczeń na jednym procesorze, a *t*(*n*) - czasem obliczeń na *n* procesorach, to klasyczne przyspieszenie jest definiowane jako [64], [72]:

$$S=\frac{t(1)}{t(n)} \tag{7.22}$$

W przypadku metody spekulacyjnej zamiast czasu *t*(*n*) stosujemy czas *t*(P_i) będący czasem obliczeń przy podziale całkowitego czasu analizy stanu nieustalonego na *i* podprzedziałów (zbiorów procesów równoległych - P_i). W takim przypadku przyspieszenie $\overline{S}$ definiowane jest jako [56]:

$$\overline{S} = \frac{t(1)}{t(P_i)} \tag{7.23}$$

Powyższa definicja precyzyjniej określa przyspieszenie metody spekulacyjnej.

7.4.5. Wnioski

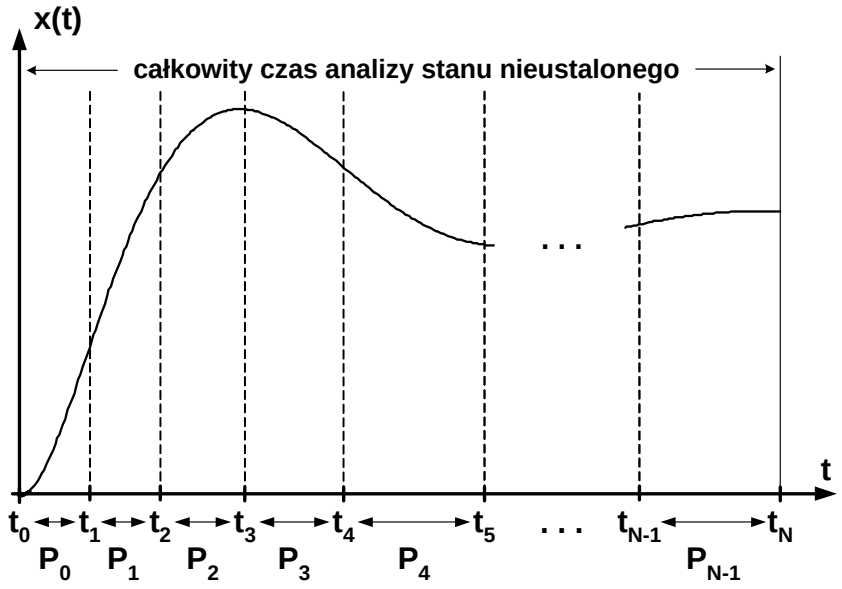
W opisanym przykładzie analizy dynamiki silnika asynchronicznego pierścieniowego przedstawiono szereg badań i eksperymentów numerycznych. Pokazały one, że wyniki obliczeń przy zastosowaniu metody spekulacyjnej ze stałym krokiem całkowania są zadawalające. Zastosowanie metody spekulacyjnej umożliwiło skrócenie czasu analizy dynamiki silnika asynchronicznego pierścieniowego, co jest **potwierdzeniem tezy rozprawy**. Należy podkreślić, że przyspieszenie obliczeń w metodzie spekulacyjnej zależne jest od liczby podprzedziałów, na które został podzielony całkowity czas analizy stanu nieustalonego, i wzrasta wraz ze wzrostem liczby tych podprzedziałów. Zwiększenie liczby podprzedziałów powoduje skrócenie ich długości oraz konieczność zastosowania większej liczby procesorów, co pociąga za sobą wydłużenie czasu komunikacji. Jednakże można przypuszczać, że ze względu na to, że w metodzie spekulacyjnej komunikacja między procesorami występuje tylko przed rozpoczęciem równoległego etapu algorytmu tej metody oraz po jego zakończeniu, to zwiększenie czasu komunikacji będzie miało niewielki wpływ na przyspieszenie obliczeń..

7.5. Metoda spekulacyjna ze zmiennym krokiem całkowania

W metodzie spekulacyjnej ze zmiennym krokiem całkowania [29], [33], [34], [35], [37], [38], [39] wykorzystywane są sekwencyjne metody numeryczne rozwiązywania układów równań różniczkowych zwyczajnych z automatycznym doborem (kontrolą) długości kroku całkowania. W niniejszym rozdziale algorytm metody spekulacyjnej zostanie opisany w przypadku zastosowania metody Rungego-Kutty czwartego rzędu z algorytmem doboru kroku całkowania opartym na ekstrapolacji, a przedstawionym w pracy [43].

Algorytm metody spekulacyjnej ze zmiennym krokiem całkowania jest podobny do przedstawionego w rozdziale 7.3 algorytmu metody spekulacyjnej ze stałym krokiem całkowania. Podstawowe różnice występują w sposobie podziału *całkowitego czasu analizy stanu nieustalonego* na podprzedziały oraz w sposobie wyznaczania *zbiorów warunków początkowych.*

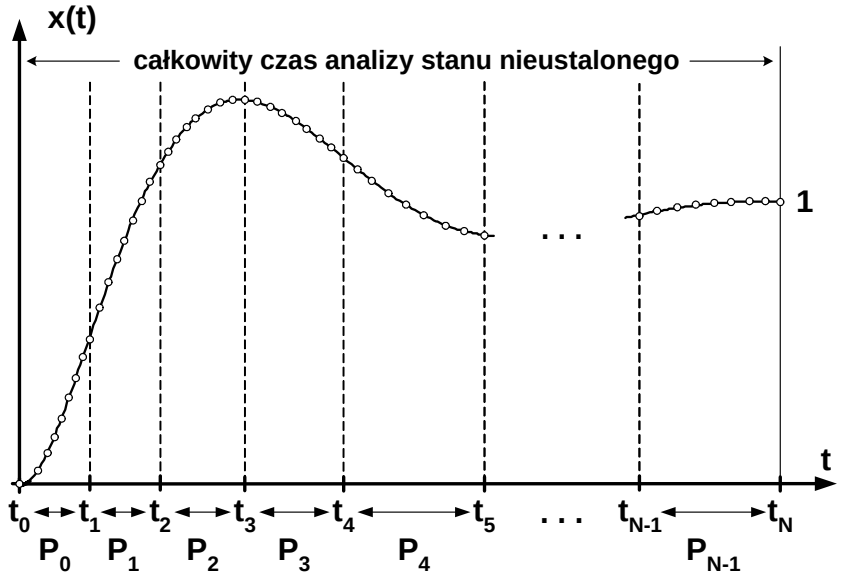
W metodzie spekulacyjnej ze zmiennym krokiem całkowania pierwszy etap algorytmu wykonywany jest sekwencyjnie przez jeden procesor (*master*). W etapie tym całkowity czas analizy stanu nieustalonego (t_0, t_N) dzielony jest na określoną liczbę podprzedziałów czasu (t_i, t_{i+1}), $i = 0,1,\ldots, N-1$ o różnej długości (rys. 7.23).



Rys. 7.23 Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały w metodzie spekulacyjnej ze zmiennym krokiem całkowania.

Obliczenia w poszczególnych podprzedziałach wykonywane są równolegle. Otrzymanie dobrego przyspieszenia obliczeń wymaga takiego samego lub porównywalnego czasu obliczeń w każdym podprzedziale. Czas obliczeń związany jest z liczbą kroków całkowania zastosowanej metody numerycznej. Należy zatem dążyć do tego, aby w każdym podprzedziale wykonywana była taka sama liczba tych kroków. W przypadku metod z automatycznym doborem długości kroku całkowania, długość kroku h, a tym samym i ich liczba w poszczególnych podprzedziałach zależna jest od szybkości zmian zmiennych stanu. W częściach szybkozmiennych (oscylacyjnych) algorytm metody numerycznej zmniejsza długość kroku całkowania. W przypadku,

gdy przebiegi są wolnozmienne (aperiodyczne), długość kroku całkowania jest zwiększana. Zastosowanie w takiej sytuacji podziału całkowitego czasu analizy stanu nieustalonego na równe podprzedziały jest nieoptymalne i nie zapewni porównywalnych czasów obliczeń w poszczególnych podprzedziałach. Dlatego w celu dokonania odpowiedniego podziału, procesor *master* wykonuje obliczenia w całym przedziale czasu (t_0, t_N) z założoną dużą wartością tolerancji błędu eps_1. W wyniku tych obliczeń otrzymywane jest rozwiązanie przybliżone (*rp*) $x_0^{rp}, x_1^{rp}, \ldots, x_m^{rp}$ w chwilach czasowych $t_0^{rp}, t_1^{rp}, \ldots, t_m^{rp}$ (przy czym $t_0^{rp} = t_0$ i $t_m^{rp} = t_N$). Następnie otrzymana liczba punktów *m* rozwiązania przybliżonego dzielona jest na *N* zbiorów punktów (gdzie *N* jest liczbą podprzedziałów). Chwile czasowe $t_i, i = 1,2,\ldots,N-1$, odpowiadające punktom znajdującym się na granicach takiego podziału, stają się granicami podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały (rys. 7.24).

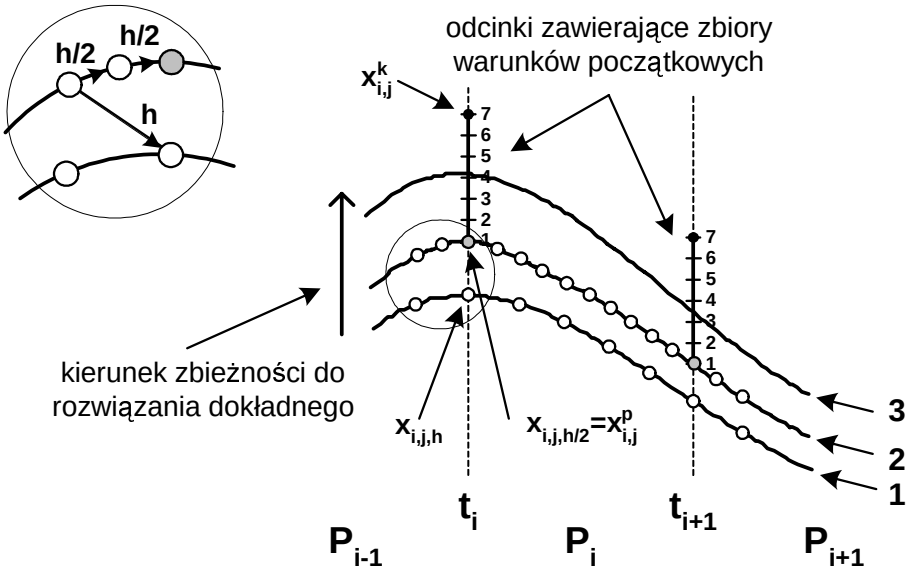


Rys. 7.24 Wyznaczenie granic podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały w metodzie spekulacyjnej ze zmiennym krokiem całkowania: 1 - *m* punktów rozwiązania przybliżonego obliczonego z dużą wartością tolerancji błędu eps_1.

Podział taki zapewnia taką samą liczbę kroków całkowania w każdym podprzedziale, ale tylko przy obliczeniach z dużą wartością tolerancji błędu eps_1. Wraz ze wzrostem dokładności obliczeń, liczba kroków całkowania szybciej rośnie w podprzedziałach zawierających oscylacje zmiennych stanu. Zjawisko to może mieć

niekorzystny wpływ na czas obliczeń w poszczególnych podprzedziałach (czasy te zaczynają się coraz bardziej różnić od siebie), a tym samym na przyspieszenie obliczeń. Częściowym rozwiązaniem tego problemu jest dodatkowa modyfikacja długości poszczególnych podprzedziałów. W tym celu, po dokonaniu podziału *m* punktów rozwiązania przybliżonego na *N* zbiorów, z każdego zbioru usuwamy określoną liczbę punktów. Następnie wszystkie usunięte punkty przydzielamy ponownie do podprzedziałów, ale w ilości proporcjonalnej do długości podprzedziałów z pierwszego podziału. Na podstawie licznych eksperymentów numerycznych stwierdzono, iż z każdego podprzedziału należy usunąć od 10% do 15% punktów do ponownego podziału. Taka modyfikacja długości podprzedziałów powoduje zmniejszenie długości krótszych podprzedziałów i zwiększenie dłuższych. W przypadku przebiegów bardzo wolnozmiennych powyższa modyfikacja nie będzie miała wpływu na długość poszczególnych podprzedziałów.

Wykonanie równolegle obliczeń w poszczególnych podprzedziałach wymaga znajomości warunków początkowych w chwilach czasowych $t_0, t_1, \ldots, t_{N-1}$. W chwili czasowej t_0 warunki te znane są z założenia - jest to zbiór podstawowych warunków początkowych. W pozostałych chwilach czasowych, $t_1, t_2, \ldots, t_{N-1}$, dla każdej zmiennej stanu, zamiast jednego warunku początkowego, wyznaczany jest zbiór warunków początkowych. W tym celu procesor *master* wykorzystuje wyniki wcześniejszych obliczeń z dużą wartością tolerancji błędu eps_1. W metodzie Rungego-Kutty czwartego rzędu wykorzystującej ekstrapolację do automatycznej kontroli długości kroku całkowania, wyznaczenie rozwiązania w kolejnej chwili czasowej wymaga wykonania obliczeń z krokiem *h*, a następnie, startując z poprzedniej chwili czasowej, dwukrotnych obliczeń z krokiem $h/2$. W ten sposób, w każdej chwili czasowej otrzymywane są dwie wartości zmiennych stanu, obliczone z krokiem *h* (oznaczane przez $x_{i,j,h}$, $i = 1,2,\ldots,N-1$, $j = 1,2,\ldots,n$, gdzie *n* jest liczbą zmiennych stanu) oraz obliczone z krokiem $h/2$ (oznaczane przez $x_{i,j,h/2}$). Wartości te wykorzystywane są do określenia dla każdej zmiennej stanu, w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$, odcinków, zawierających zbiory warunków początkowych (rys. 7.25).



Rys. 7.25 Wyznaczenie zbiorów warunków początkowych w metodzie spekulacyjnej ze zmiennym krokiem całkowania: 1 - punkty rozwiązania przybliżonego wyznaczonego z krokiem *h*, 2 - punkty rozwiązania przybliżonego wyznaczonego z krokiem *h/2*, 3 - nieznane rozwiązanie dokładne.

Wartość bezwzględna różnicy wartości zmiennych stanu obliczonych z krokami h i $h/2$, pomnożona przez współczynnik bezpieczeństwa k, określa długość poszukiwanego odcinka:

$$\Delta x_{i,j} = \left| x_{i,j,h/2} - x_{i,j,h} \right| \cdot k \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.24}$$

Współczynnik bezpieczeństwa k pełni taką samą rolę jak w przypadku metody spekulacyjnej ze stałym krokiem całkowania. Jego wartość została określona na podstawie eksperymentów numerycznych. W przypadku zastosowania metody Rungego-Kutty czwartego rzędu oraz ekstrapolacji do automatycznej kontroli długości kroku całkowania, najbardziej odpowiednią wartością jest $k = 10$. Jako położenie początku odcinka ($x^p_{i,j}$) przyjmowane są wartości zmiennych stanu obliczone z krokiem $h/2$:

$$x^p_{i,j} = x_{i,j,h/2}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.25}$$

zaś położenie końca odcinka ($x^k_{i,j}$) jest to przesunięcie o jego długość w kierunku zbieżności do rozwiązania dokładnego:

$$x^k_{i,j} = x_{i,j,h/2} \pm \Delta x_{i,j}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{7.26}$$

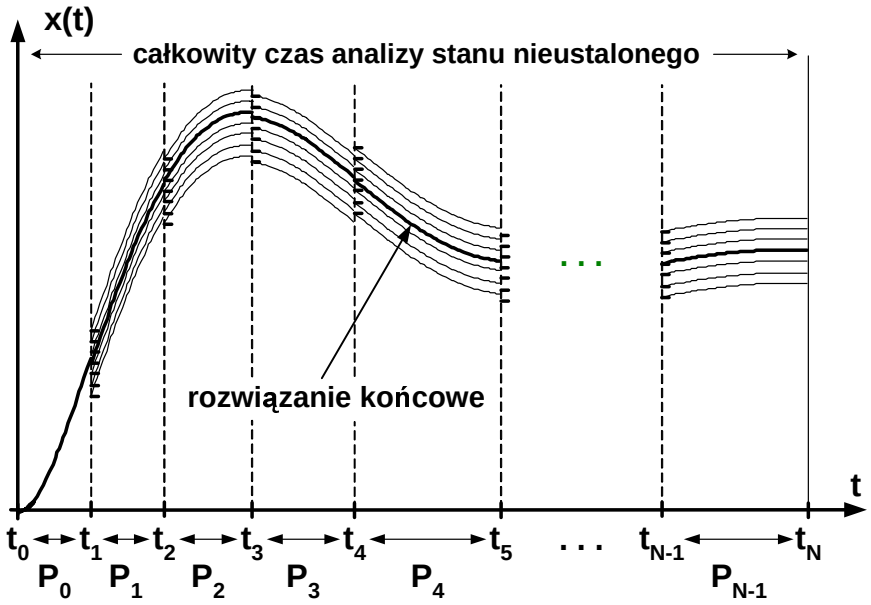
Kierunek zbieżności do rozwiązania dokładnego wyznaczany jest na podstawie analizy wartości zmiennych stanu obliczonych z krokami h i $h/2$, przy czym jest on określany oddzielnie dla każdej chwili czasowej $t_1, t_2, \ldots, t_{N-1}$ i oddzielnie dla każdej zmiennej stanu. Jeśli $x_{i,j,h} \geq x_{i,j,h/2}$, to we wzorze (7.26) występuje znak „-", co oznacza, że rozwiązanie dokładne znajduje się na wykresie „poniżej" rozwiązań przybliżonych $x_{i,j,h}$ i $x_{i,j,h/2}$. Jeśli natomiast $x_{i,j,h} < x_{i,j,h/2}$, to we wzorze (7.26) występuje znak „+" i rozwiązanie dokładne znajduje się na wykresie „powyżej" rozwiązań przybliżonych (por. rys. 7.25).

Otrzymany odcinek dzielony jest na określoną liczbę przedziałów. Wartości zmiennych stanu powstałe podczas tego podziału są poszukiwanymi zbiorami warunków początkowych (rys. 7.25).

Po wyznaczeniu warunków początkowych przechodzimy do drugiego etapu algorytmu metody spekulacyjnej. W etapie tym wykonywane są równolegle obliczenia właściwe we wszystkich podprzedziałach z małą wartością tolerancji błędu *eps*. W pierwszym podprzedziale (t_0, t_1) warunki początkowe znane są z założenia, toteż obliczenia w tym podprzedziale wykonywane są tylko jednokrotnie przez procesor nadrzędny *master*. W pozostałych podprzedziałach, dla każdej zmiennej stanu określony jest zbiór warunków początkowych. Obliczenia w nich wykonywane są przez procesory *slave*, przy czym liczba tych procesorów w każdym podprzedziale jest równa liczbie warunków początkowych w zbiorze warunków początkowych. Jeśli zatem całkowity czas analizy stanu nieustalonego podzielony został na N podprzedziałów, a w każdym podprzedziale jest p warunków początkowych dla jednej zmiennej stanu, to całkowita liczba procesorów wykorzystywanych w obliczeniach wynosi: $L = (N-1) \cdot p + 1$, przy czym $(N-1) \cdot p$ jest liczbą procesorów *slave*. Przed rozpoczęciem obliczeń w drugim etapie algorytmu, procesor *master* wysyła dane konieczne do rozpoczęcia obliczeń (m.in. warunki początkowe, granice podprzedziałów) do procesorów *slave*. Wszystkie procesory rozpoczynają równolegle obliczenia w swoich podprzedziałach z małą wartością tolerancji błędu *eps*. Po zakończeniu obliczeń w pierwszym podprzedziale (t_0, t_1) otrzymywane jest jedno

rozwiązanie dla każdej zmiennej stanu, zaś w pozostałych podprzedziałach - szereg rozwiązań mających różne warunki początkowe.

Następnie procesor *master* określa rozwiązanie końcowe składające się z wybranych przebiegów z poszczególnych podprzedziałów (rys. 7.26).



Rys. 7.26 Wyznaczenie rozwiązania końcowego w metodzie spekulacyjnej ze zmiennym krokiem całkowania.

W pierwszym podprzedziale jest tylko jedno rozwiązanie wyznaczone z warunkami początkowymi znanymi z założenia. W kolejnym podprzedziale (t_1, t_2) dla każdej zmiennej stanu określony jest szereg rozwiązań o różnych warunkach początkowych. Jako rozwiązanie końcowe w tym podprzedziale procesor *master* wybiera ten przebieg, którego warunek początkowy jest najbliższy wartości w chwili czasowej t_1 otrzymanej podczas obliczeń w pierwszym podprzedziale (t_0, t_1). W kolejnych podprzedziałach procesor *master* postępuje w ten sam sposób, aż do osiągnięcia ostatniego podprzedziału (t_{N-1}, t_N).

W metodzie spekulacyjnej ze stałym krokiem całkowania wartości wszystkich zmiennych stanu są zawsze wyznaczane dla tych samych chwil czasowych. Natomiast w metodzie spekulacyjnej ze zmiennym krokiem całkowania wartości zmiennych stanu mogą być obliczane dla różnych chwil czasowych. Wymaga to zapamiętania, oddzielnie dla każdej zmiennej stanu, jej wartości, oraz chwili czasowej, w której ta

wartość została wyznaczona. Nie dotyczy to pierwszego podprzedziału, w którym obliczenia wykonywane są tylko jednokrotnie przez jeden procesor. Konieczność oddzielnego zapamiętania wartości zmiennych stanu oraz chwil czasowych jest spowodowana tym, że obliczenia w poszczególnych podprzedziałach wykonywane są wielokrotnie z różnymi warunkami początkowymi, więc w każdym przypadku długość kroku całkowania może inaczej zmieniać się. W takiej sytuacji przeprowadzenie obliczeń z jednakowymi długościami kroku całkowania wymagałoby komunikacji między procesorami przed wykonaniem każdego kroku, co znacząco wydłużyłoby całkowity czas obliczeń.

7.6. Zastosowanie metody spekulacyjnej ze zmiennym krokiem całkowania do analizy dynamiki silnika asynchronicznego pierścieniowego

7.6.1. Wstęp

Jako przykład zastosowania metody spekulacyjnej ze zmiennym krokiem całkowania przedstawiona zostanie, opisana w rozdziale 7.4.2, analiza dynamiki silnika asynchronicznego pierścieniowego.

7.6.2. Zastosowanie metody spekulacyjnej

W algorytmie metody spekulacyjnej ze zmiennym krokiem całkowania przyjęto następujące parametry obliczeń:

- sekwencyjna metoda numeryczna rozwiązywania układów równań różniczkowych zwyczajnych: metoda Rungego-Kutty czwartego rzędu z automatyczną kontrolą długości kroku całkowania,
- całkowity czas analizy stanu nieustalonego (t_0, t_N): $0{,}0 \div 1{,}0\,s$,
- podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów (zbiorów procesów równoległych: $P_0, P_1, \ldots, P_5$),

- obliczenia z dużą wartością tolerancji błędu: $eps_1 = 10^{-2}$,
- obliczenia z małą wartością tolerancji błędu: $eps = 10^{-8}$,
- liczba warunków początkowych w jednym podprzedziale: 10,
- współczynnik bezpieczeństwa: $k = 10$.

Podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów wynikał z liczby dostępnych procesorów oraz metody wykonywania obliczeń. Duża wartość tolerancji błędu $eps_1 = 10^{-2}$ została wybrana w ten sposób, aby rozwiązanie przybliżone było stabilne. Otrzymanie dokładnego rozwiązania metody spekulacyjnej wymagało wykonania obliczeń z mała wartością tolerancji błędu $eps = 10^{-8}$. Liczba dziesięciu warunków początkowych w jednym podprzedziale jest kompromisem pomiędzy dokładnością wyznaczania warunku początkowego, a liczbą procesorów stosowanych w obliczeniach. Wartość współczynnika bezpieczeństwa k wynika z zastosowania metody Rungego-Kutty czwartego rzędu z automatyczną kontrolą długości kroku całkowania.

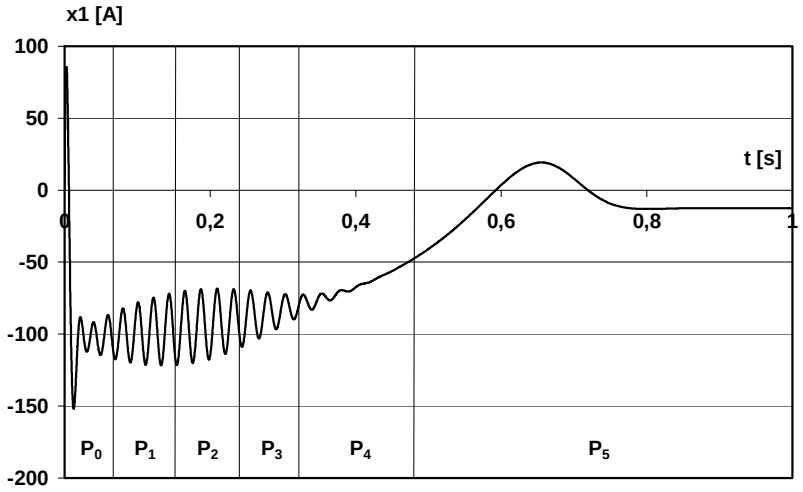
Obliczenia przeprowadzono na sześciu węzłach systemu klaster KETIM2, który został opisany w rozdziale 5.2. Na podstawie obliczeń z dużą wartością tolerancji błędu $eps_1 = 10^{-2}$ całkowity czas analizy stanu nieustalonego podzielony został na 6 podprzedziałów. Długości poszczególnych podprzedziałów zostały dodatkowo zmodyfikowane w celu otrzymania porównywalnego czasu obliczeń w poszczególnych podprzedziałach (tabela 7.4). Do modyfikacji wykorzystano 15% punktów rozwiązania przybliżonego otrzymanego podczas obliczeń z dużą wartością tolerancji błędu eps_1.

Tabela 7.4 Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały.

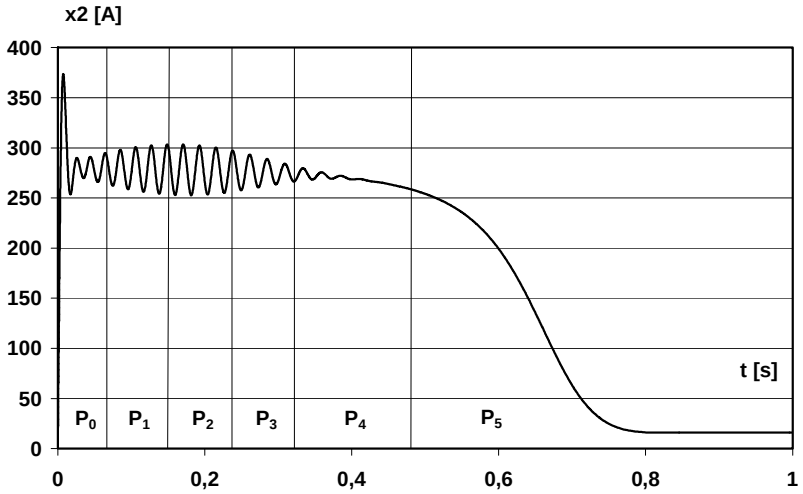
	Długości podprzedziałów					
	(t_0, t_1)	(t_1, t_2)	(t_2, t_3)	(t_3, t_4)	(t_4, t_5)	(t_5, t_6)
Przed modyfikacją	0,0675 s	0,0925 s	0,0925 s	0,09 s	0,22 s	0,4375 s
Po modyfikacji	0,0625 s	0,0875 s	0,0875 s	0,0825 s	0,1625 s	0,5175 s

7.6.3. Wyniki obliczeń

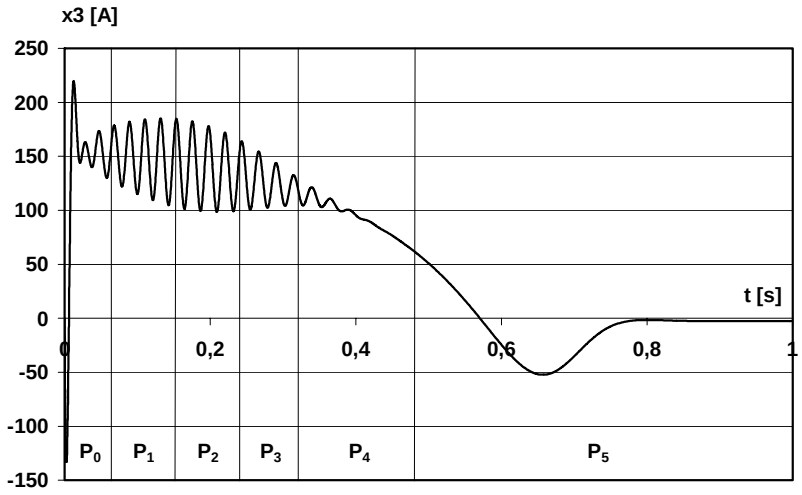
Na rys. 7.27 - rys. 7.31 przedstawiono otrzymane rozwiązania dla wszystkich zmiennych stanu, wraz z zaznaczeniem granic podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały. Na rysunkach tych można zaobserwować wpływ charakteru rozwiązania na długość poszczególnych podprzedziałów. Podprzedziały, w których zmienne stanu oscylują (przebiegi są szybkozmienne) mają mniejszą długość w stosunku do podprzedziałów, w których przebiegi są gładkie (wolnozmienne).



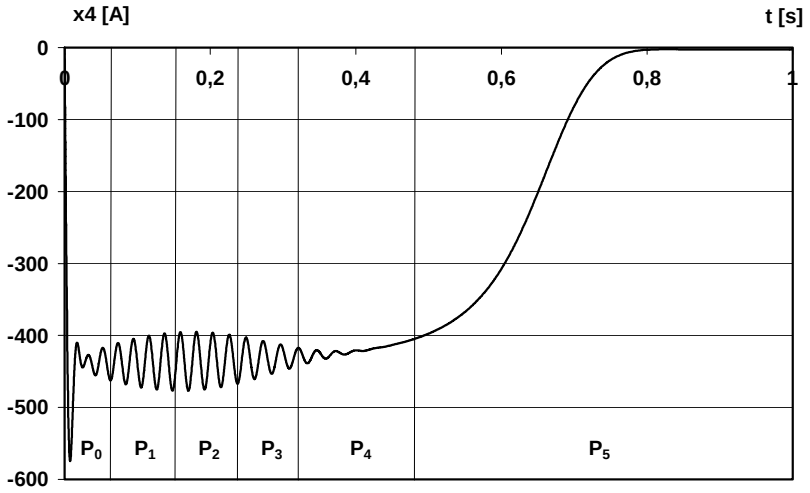
Rys. 7.27 Przebieg prądu stojana $x_1(t)$ w układzie współrzędnych wirnika.



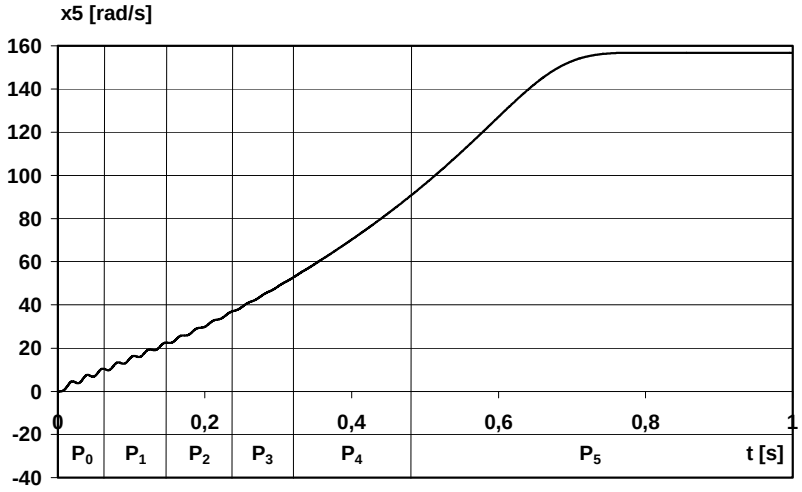
Rys. 7.28 Przebieg prądu stojana $x_2(t)$ w układzie współrzędnych wirnika.



Rys. 7.29 Przebieg prądu wirnika $x_3(t)$ w układzie współrzędnych wirnika.



Rys. 7.30 Przebieg prądu wirnika $x_4(t)$ w układzie współrzędnych wirnika.



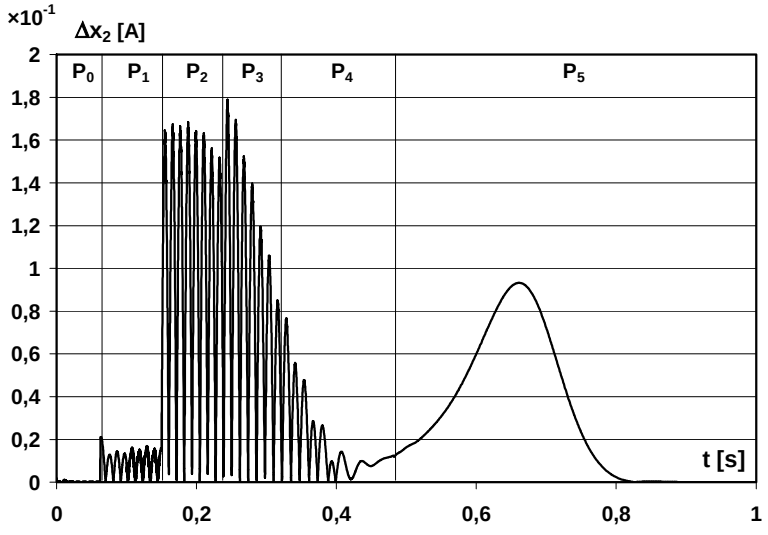
Rys. 7.31 Przebieg prędkości kątowej $x_5(t)$.

W celu oszacowania dokładności rozwiązania metody spekulacyjnej, układ nieliniowy (7.10) został rozwiązany przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-9}\ s$. W przypadku rozwiązania tego samego przykładu przy zastosowaniu metody spekulacyjnej ze stałym krokiem całkowania, obliczenia sekwencyjne zostały wykonane z krokiem $h = 10^{-7}\ s$. Zmiana długości kroku całkowania została spowodowana koniecznością otrzymania rozwiązania sekwencyjnego w dokładnie tych samych chwilach czasowych co rozwiązanie metody spekulacyjnej ze zmiennym krokiem całkowania. Zastosowanie w tym przypadku kroku $h = 10^{-7}\ s$ spowodowałoby pominięcie dużej liczby punktów rozwiązania.

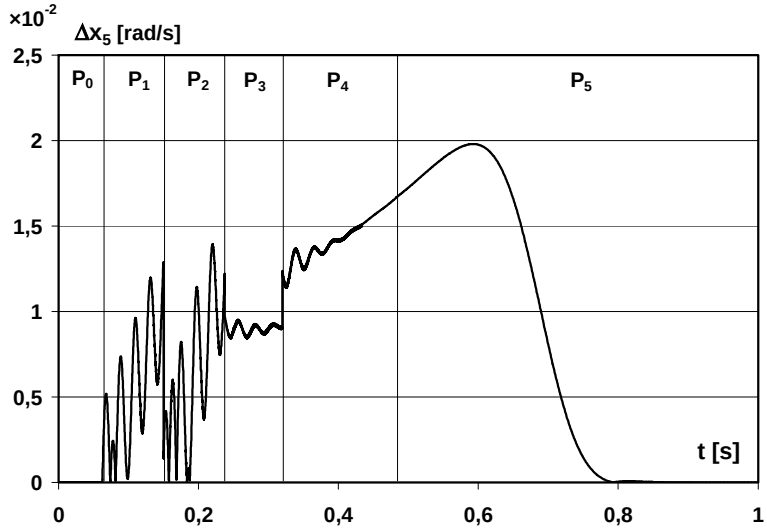
Na rys. 7.32 i rys. 7.33 przedstawiono błędy bezwzględne Δx_i (7.27) metody spekulacyjnej dla dwóch wybranych zmiennych stanu $x_2(t)$ i $x_5(t)$.

$$\Delta x_i = \left| x_{i,RK} - x_{i,MSP} \right|, \quad i = 1,2,\ldots,5 \tag{7.27}$$

gdzie: $x_{i,RK}$ jest rozwiązaniem sekwencyjnym, zaś $x_{i,MSP}$ jest rozwiązaniem metody spekulacyjnej ($eps_1 = 10^{-2}$, $eps = 10^{-8}$).



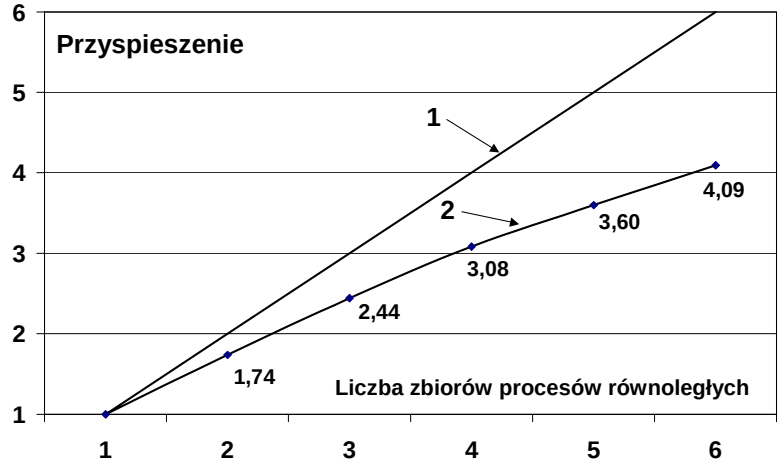
Rys. 7.32 Błędy bezwzględne metody spekulacyjnej ze zmiennym krokiem całkowania dla zmiennej stanu $x_2(t)$.



Rys. 7.33 Błędy bezwzględne metody spekulacyjnej ze zmiennym krokiem całkowania dla zmiennej stanu $x_5(t)$.

Największe zmiany wartości błędów można zaobserwować na początku każdego podprzedziału. Jest to spowodowane tym, iż obliczenia w każdym podprzedziale rozpoczynane są ze spekulacyjnie wyznaczonymi warunkami początkowymi.

Na rys. 7.34 przedstawione zostały wartości oszacowanego przyspieszenia ($\overline{S}$), otrzymane na podstawie wyników obliczeń wykonanych na klastrze KETIM2, opisanym w rozdziale 5.2.



Rys. 7.34 Oszacowane przyspieszenie w metodzie spekulacyjnej;
1 - przypadek idealny, 2 - otrzymane przyspieszenie.

W celu wyznaczenia przyspieszenia, całkowity czas analizy stanu nieustalonego podzielony został na 2, 3, 4, 5 i 6 podprzedziałów. Podczas tego oszacowania pod uwagę została wzięta tylko liczba kroków całkowania metody Rungego-Kutty, natomiast czas komunikacji między procesorami został pominięty. Rzeczywiste wartości przyspieszenia będą tylko nieznacznie mniejsze od przedstawionych na rys. 7.34, gdyż w metodzie spekulacyjnej komunikacja między procesorami odbywa się tylko przed rozpoczęciem i po zakończeniu drugiego etapu algorytmu metody.

7.6.4. Wnioski

Zastosowanie metody spekulacyjnej umożliwiło skrócenie czasu analizy stanu nieustalonego w stosunku do algorytmu sekwencyjnego, co jest **potwierdzeniem tezy rozprawy**. W przedstawionym przykładzie otrzymane przyspieszenie wyniosło 4,09 (przy podziale całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów).

Podstawowym problemem w metodzie spekulacyjnej ze zmiennym krokiem całkowania jest odpowiedni podział całkowitego czasu analizy stanu nieustalonego na podprzedziały, zapewniający taką samą liczbę kroków całkowania w każdym podprzedziale, a tym samym taki sam czas obliczeń. Podział ten ma znaczący wpływ na przyspieszenie obliczeń. W tabeli 7.5 zestawiono liczby kroków całkowania wykonanych przed i po modyfikacji długości poszczególnych podprzedziałów. Analizując poniższą tabelę można zauważyć, że największa poprawa nastąpiła w przypadku ostatniego podprzedziału (t_5, t_6).

Tabela 7.5 Liczba kroków całkowania przed i po modyfikacji długości podprzedziałów.

	Liczba kroków całkowania					
	(t_0, t_1)	(t_1, t_2)	(t_2, t_3)	(t_3, t_4)	(t_4, t_5)	(t_5, t_6)
Przed modyfikacją	559	592	592	576	554	398
Po modyfikacji	527	560	560	528	570	526

8. Zmodyfikowana metoda spekulacyjna analizy stanów nieustalonych

8.1. Wstęp

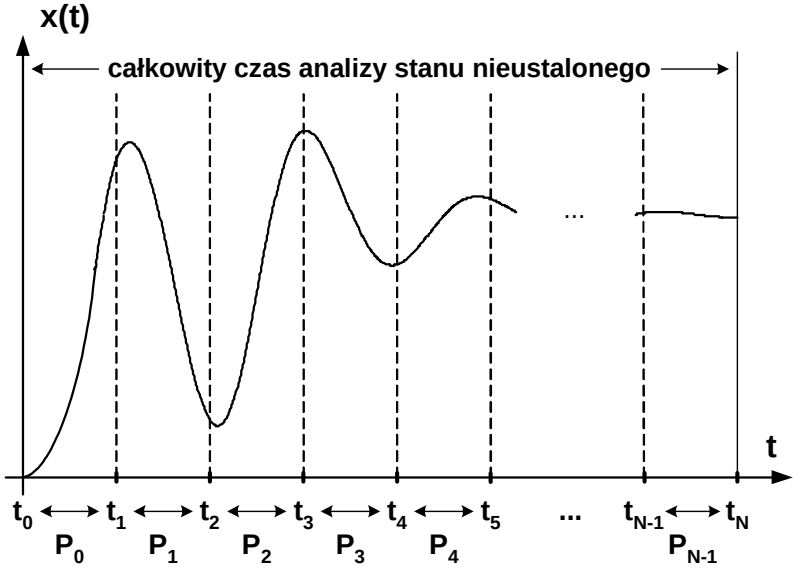
Opisana w poprzednim rozdziale metoda spekulacyjna (ze stałym lub zmiennym krokiem całkowania), przy podziale całkowitego czasu analizy stanu nieustalonego na N podprzedziałów i przy p warunkach początkowych w jednym zbiorze warunków początkowych, wymaga $L = (N-1) \cdot p + 1$ procesorów liczących, czyli systemu komputerowego o dużej liczbie węzłów obliczeniowych. Opracowano zatem koncepcję zmodyfikowanej metody spekulacyjnej [30], [32], [40], [41], [42], wymagającej mniejszej liczby węzłów obliczeniowych. W metodzie tej, podobnie jak w metodzie spekulacyjnej, dla każdej zmiennej stanu wyznaczany jest zbiór warunków początkowych, ale do obliczeń wybierany jest tylko jeden warunek początkowy z tego zbioru. Obliczenia w każdym podprzedziale wykonywane są zatem jednokrotnie przez jeden procesor liczący. Przy opisie algorytmu zmodyfikowanej metody spekulacyjnej przedstawione zostaną dwie metody wyboru zbiorów warunków początkowych. Pierwsza metoda oparta jest na analizie rozwiązań przybliżonych, natomiast druga na przybliżeniu wykresu zbieżności rozwiązania funkcją wykładniczą.

8.2. Algorytm zmodyfikowanej metody spekulacyjnej

Algorytm zmodyfikowanej metody spekulacyjnej przedstawiony zostanie w przypadku zastosowania metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania, jako sekwencyjnej metody numerycznej rozwiązywania układów równań różniczkowych zwyczajnych. W algorytmie zmodyfikowanej metody spekulacyjnej wyróżniane są dwa etapy: sekwencyjny i równoległy. W pierwszym etapie całkowity czas analizy stanu nieustalonego dzielony jest na podprzedziały oraz wyznaczane są warunki początkowe niezbędne do wykonania obliczeń. W drugim etapie wykonywane są równolegle obliczenia właściwe oraz wyznaczane jest rozwiązanie końcowe.

8.2.1. Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały

Pierwszy etap algorytmu zmodyfikowanej metody spekulacyjnej wykonywany jest sekwencyjnie przez procesor *master*. Dzieli on całkowity czas analizy stanu nieustalonego (t_0, t_N) na N podprzedziałów (t_i, t_{i+1}), $i = 0,1,\ldots,N-1$. W przypadku stosowania metody numerycznej ze stałą długością kroku całkowania, długość wszystkich podprzedziałów jest taka sama (rys. 8.1).

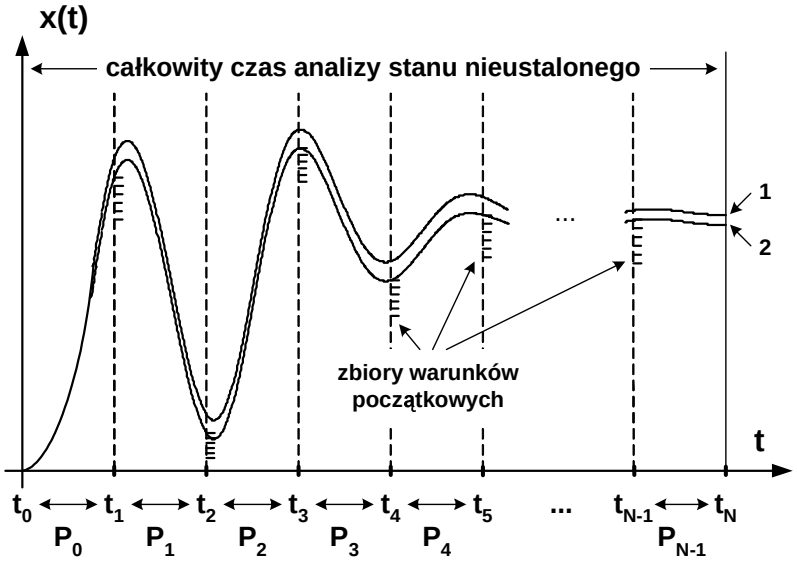


Rys. 8.1 Podział całkowitego czasu analizy stanu nieustalonego na podprzedziały w zmodyfikowanej metodzie spekulacyjnej.

8.2.2. Wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych

Po podziale całkowitego czasu analizy stanu nieustalonego procesor *master* przystępuje do wyznaczenia zbiorów warunków początkowych na początku poszczególnych podprzedziałów, tj. w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$ (w chwili czasowej t_0 warunki te znane są z założenia). W przypadku wyznaczania zbiorów warunków początkowych na podstawie analizy rozwiązań przybliżonych, algorytm jest taki sam jak w metodzie spekulacyjnej. Procesor *master* wykonuje obliczenia w całym przedziale czasu (t_0, t_N) z dużą długością kroku całkowania h_1 (np. na granicy lub w pobliżu granicy stabilności zastosowanej metody numerycznej rozwiązywania

układów równań różniczkowych). W wyniku tych obliczeń otrzymywane jest pierwsze rozwiązanie przybliżone $x_{i,j,h1}$, $i = 1,2,\ldots,N-1$, $j = 1,2,\ldots,n$ (gdzie n jest liczbą zmiennych stanu) w chwilach czasowych t_i, $i = 1,2,\ldots,N-1$ (rys. 8.2).



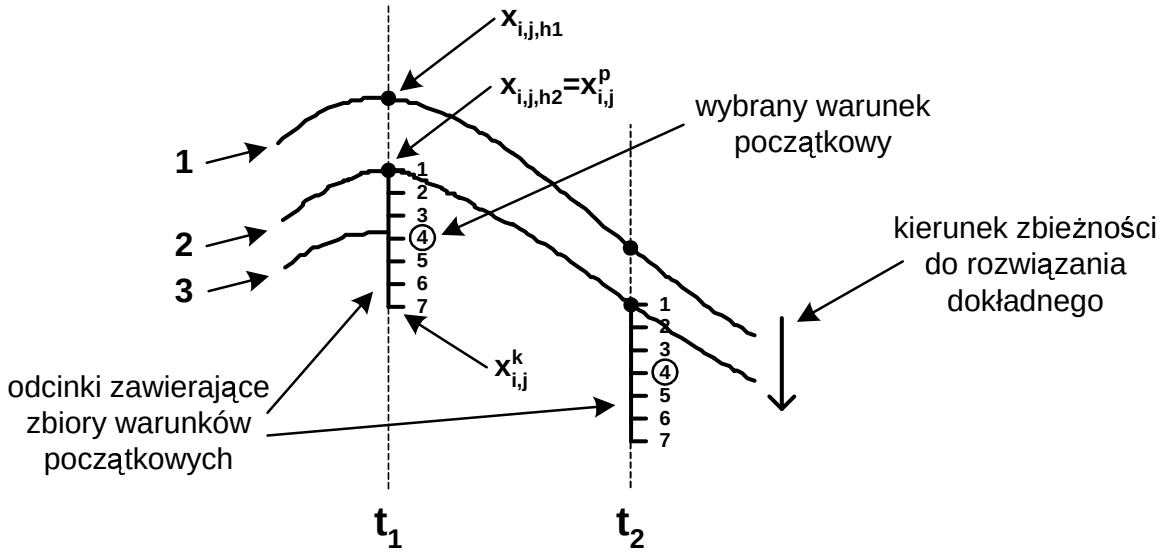
Rys. 8.2 Wyznaczenie zbiorów warunków początkowych w zmodyfikowanej metodzie spekulacyjnej: 1 - rozwiązanie przybliżone otrzymane z krokiem h_1, 2 - rozwiązanie przybliżone otrzymane z krokiem h_2.

Następnie obliczenia są powtarzane z krokiem h_2 spełniającym warunek $h_2 < h_1$ (h_1 powinno być wielokrotnością h_2, np. $h_2 = h_1/2$) i otrzymywane jest drugie rozwiązanie przybliżone $x_{i,j,h2}$, $i = 1,2,\ldots,N-1$, $j = 1,2,\ldots,n$. Wartości obu rozwiązań wykorzystywane są do określenia w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$, oddzielnie dla każdej zmiennej stanu, odcinków, zawierających warunki początkowe. Jako długość odcinka ($\Delta x_{i,j}$) przyjmowana jest wartość bezwzględna różnicy wartości obliczonych z krokami h_1 i h_2, pomnożona przez współczynnik bezpieczeństwa k:

$$\Delta x_{i,j} = \left| x_{i,j,h2} - x_{i,j,h1} \right| \cdot k \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{8.1}$$

Położenie odcinka określane jest przy zastosowaniu następującej metody (rys. 8.3). Jako początek odcinka - $x_{i,j}^p$ (8.2) przyjmowane są wartości zmiennych stanu obliczone z krokiem h_2:

$$x_{i,j}^p = x_{i,j,h2}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{8.2}$$



Rys. 8.3 Wyznaczenie odcinków zawierających zbiory warunków początkowych w zmodyfikowanej metodzie spekulacyjnej; 1 - rozwiązanie przybliżone otrzymane z krokiem h_1, 2 - rozwiązanie przybliżone otrzymane z krokiem h_2, 3 - rozwiązanie wyznaczone przez procesor *master* w podprzedziale (t_0, t_1).

Koniec odcinka - $x^k_{i,j}$ (8.3) jest to przesunięcie o jego długość ($\Delta x_{i,j}$) w kierunku zbieżności do rozwiązania dokładnego.

$$x^k_{i,j} = x_{i,j,h2} \pm \Delta x_{i,j}, \qquad i = 1,2,\ldots,N-1, \quad j = 1,2,\ldots,n \tag{8.3}$$

Kierunek zbieżności do rozwiązania dokładnego określany jest na podstawie analizy wartości zmiennych stanu obliczonych z krokami h_1 i h_2. Jeśli $x_{i,j,h1} \geq x_{i,j,h2}$, to nieznane rozwiązanie dokładne znajduje się „poniżej” rozwiązań przybliżonych i we wzorze (8.3) występuje znak „+”. W przeciwnym przypadku, gdy $x_{i,j,h1} < x_{i,j,h2}$, to rozwiązanie dokładne znajduje się „powyżej” rozwiązań przybliżonych i we wzorze (8.3) występuje znak „-”. Otrzymany odcinek dzielony jest na określoną liczbę przedziałów (rys. 8.3). Punkty podziału odcinka, czyli wartości zmiennej stanu **x**, stają się poszukiwanym zbiorem warunków początkowych. Każdy warunek początkowy w zbiorze warunków początkowych otrzymuje kolejny numer (rys. 8.3).

Do obliczeń właściwych wybierany jest tylko jeden warunek ze zbioru warunków początkowych. W celu wyznaczenia numeru tego warunku procesor *master* wykonuje obliczenia w pierwszym podprzedziale (t_0, t_1) z małą długością kroku

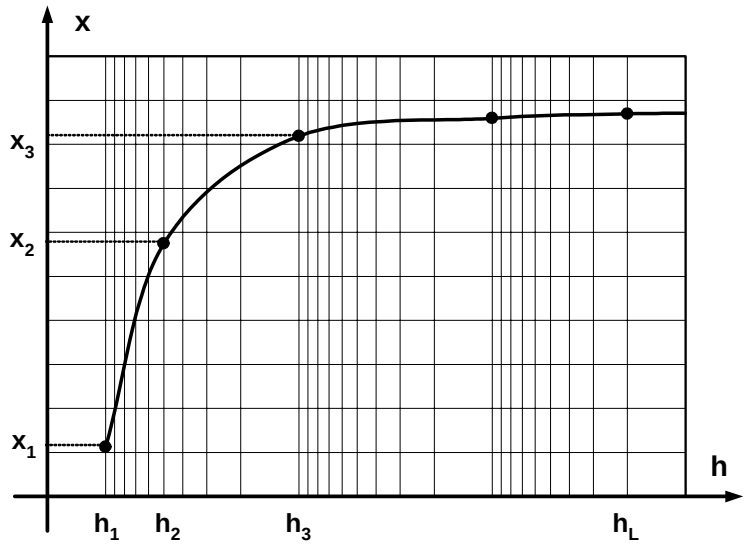
całkowania h (taką samą jak obliczenia właściwe w drugim etapie algorytmu zmodyfikowanej metody spekulacyjnej). Następnie porównuje otrzymane wartości zmiennych stanu w chwili czasowej t_1 z wyznaczonymi dla tej chwili czasowej warunkami początkowymi, zapamiętując, oddzielnie dla każdej zmiennej stanu, numery tych warunków początkowych, które są najbliższe wartościom zmiennych stanu w chwili czasowej t_1. W pozostałych chwilach czasowych $t_2, t_3, \ldots, t_{N-1}$ wybierane są warunki początkowe mające takie same numery w zbiorach warunków początkowych, jak zapamiętane dla chwili czasowej t_1. W ten sposób w każdym podprzedziale wyznaczony zostaje dla każdej zmiennej stanu tylko jeden warunek początkowy wykorzystywany w obliczeniach właściwych w drugim etapie algorytmu.

8.2.3. Wyznaczenie warunków początkowych na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą

W metodzie spekulacyjnej i zmodyfikowanej metodzie spekulacyjnej przy określaniu długości odcinków zawierających zbiory warunków początkowych wprowadzany jest tzw. współczynnik bezpieczeństwa k. Wartość tego współczynnika zależna jest od zastosowanej sekwencyjnej metody numerycznej rozwiązywania równań różniczkowych, a określana na podstawie eksperymentów numerycznych. Zadaniem współczynnika k jest takie określenie długości odcinka, aby zawierał się w nim dokładny (nieznany) warunek początkowy. Poprawność wyboru wartości współczynnika k można sprawdzić analizując wykresy zbieżności rozwiązania zastosowanej metody numerycznej w punktach będących początkami podprzedziałów podziału całkowitego czasu analizy stanu nieustalonego. Analizując powyższe wykresy stwierdzono, że wykres zbieżności przedstawiony w skali logarytmicznej można przybliżyć funkcją wykładniczą (rys. 8.4) o postaci:

$$x(t) = A(1 - e^{-\alpha t}) \tag{8.4}$$

Znając zatem wartości zmiennych stanu w określonej chwili czasowej obliczone z dużymi krokami całkowania (h_1, h_2, h_3), można wyznaczyć parametry α i A równania (8.4) i na tej podstawie obliczyć przybliżone wartości zmiennych stanu w tej samej chwili czasowej, ale dla małego kroku całkowania h_L (rys. 8.4).



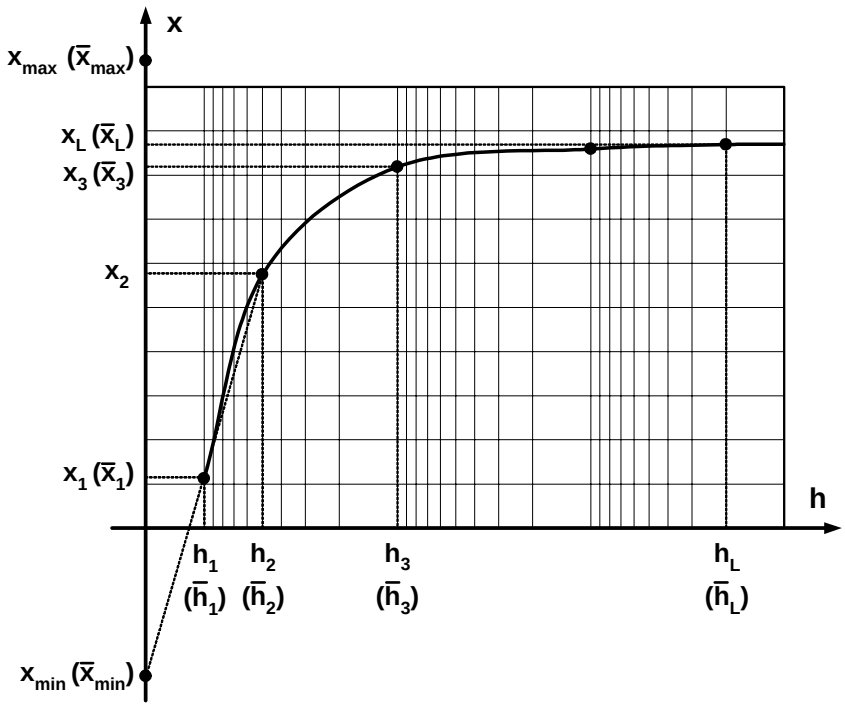
Rys. 8.4 Wykres zbieżności rozwiązania metody numerycznej w wybranej chwili czasowej dla jednej zmiennej stanu ($h_1 > h_2 > h_3 > h_L$).

Powyższe spostrzeżenie było podstawą do opracowania nowej metody wyznaczania warunków początkowych [42] w metodzie spekulacyjnej.

W metodzie tej warunki początkowe wyznaczane są przez procesor *master*. Wykonuje on trzykrotnie obliczenia w całym przedziale (t_0, t_N) z krokami całkowania h_1, h_2 i h_3 ($h_1 > h_2 > h_3$) znajdującymi się na granicy lub w pobliżu granicy stabilności zastosowanej metody numerycznej rozwiązywania układów równań różniczkowych. Wynikiem tych obliczeń są wartości zmiennych stanu (x_1, x_2, x_3) w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$, będących granicami podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały (rys. 8.5).

Przybliżenie wykresu zbieżności wzorem (8.4) wymaga takiego przeskalowania obu osi, aby wykres zbieżności przechodził przez początek układu współrzędnych. Na osi odciętych znajdują się długości kroków całkowania umieszczone w skali logarytmicznej. Ich przeskalowanie na współrzędne w skali dziesiętnej jest bardzo proste i prowadzi do otrzymania wartości $\bar{h}_1$, $\bar{h}_2$ i $\bar{h}_3$. W przypadku przeskalowania osi rzędnych należy w pierwszej kolejności określić punkt przecięcia wykresu zbieżności z tą osią, czyli wartość $x_{\min}$. Wartość ta wyznaczana jest z równania prostej przechodzącej przez punkty $(\bar{h}_1, x_1)$ i $(\bar{h}_2, x_2)$:

$$x_{\min} = \frac{x_2 \bar{h}_1 - x_1 \bar{h}_2}{\bar{h}_1 - \bar{h}_2} \tag{8.5}$$



Rys. 8.5 Przeskalowanie wartości na osiach wykresu zbieżności.

Następnie określany jest górny zakres wartości do przeskalowania - $x_{\max}$, za który przyjmowana jest wartość x_3 obliczona z krokiem h_3, powiększona o różnicę wartości obliczonych z krokami h_3 i h_2:

$$x_{\max} = x_3 + (x_3 - x_2) \tag{8.6}$$

Dla nowej skali zakładane jest, że $\bar{x}_{\min}$ odpowiada wartości $x_{\min}$ będącej punktem przecięcia wykresu zbieżności z osią rzędnych, zaś $\bar{x}_{\max}$ odpowiada wartości $x_{\max}$, określającej górny zakres wartości do przeskalowania, czyli punkt leżący ponad wykresem zbieżności. Przeskalowanie wartości x odbywa się na podstawie proporcji:

$$\frac{x - x_{\min}}{x_{\max} - x_{\min}} = \frac{\bar{x} - \bar{x}_{\min}}{\bar{x}_{\max} - \bar{x}_{\min}} \tag{8.7}$$

stąd poszukiwane $\bar{x}$ wynosi:

$$\bar{x} = \frac{\bar{x}_{\max} - \bar{x}_{\min}}{x_{\max} - x_{\min}} \left(x - x_{\min} \right) + \bar{x}_{\min} \tag{8.8}$$

Zakładając, że $\bar{x}_{\min} = 0$ i $\bar{x}_{\max} = X$, gdzie X jest dodatnią stałą, otrzymywane jest przeskalowanie wartości x_1 i x_3 o następującej postaci:

$$\bar{x}_1 = \frac{X}{x_{\max} - x_{\min}}(x_1 - x_{\min}), \qquad \bar{x}_3 = \frac{X}{x_{\max} - x_{\min}}(x_3 - x_{\min}) \tag{8.9}$$

Po przeskalowaniu x_1 i x_3 wyznaczane są parametry α i A równania (8.4):

$$\alpha = \frac{\ln(\bar{x}_3 - \bar{x}_1) + \ln \bar{x}_1 - \ln \bar{x}_3}{\bar{h}_3 - \bar{h}_1} \tag{8.10}$$

$$A = \frac{\bar{x}_3}{1 - e^{-\alpha \bar{h}_3}} \tag{8.11}$$

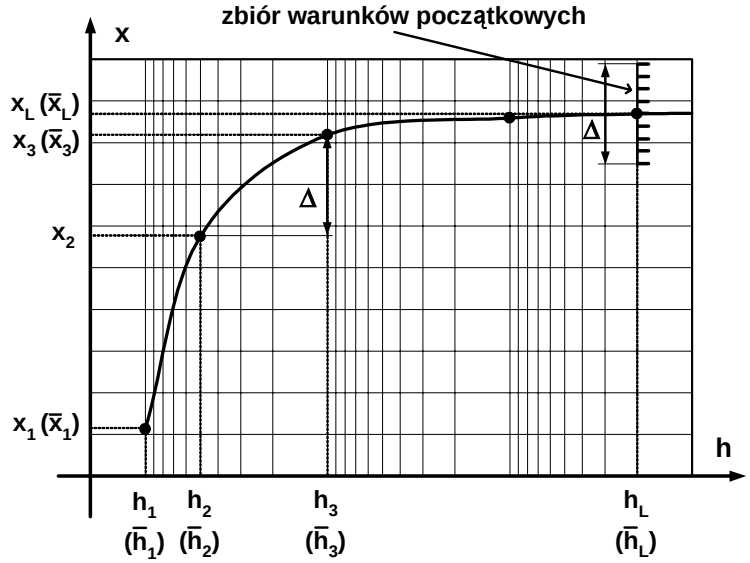
Znając powyższe parametry można obliczyć przybliżone wartości zmiennych stanu $\bar{x}_L$ dla małego kroku całkowania $\bar{h}_L$. Dla uproszczenia można przyjąć, że wartość $\bar{x}_L$ jest równa parametrowi A i wtedy x_L wyznaczane jest po ponownym przeskalowaniu:

$$x_L = \frac{A}{X}(x_{\max} - x_{\min}) + x_{\min} \tag{8.12}$$

W powyższy sposób wyznaczane są przybliżone wartości zmiennych stanu w chwilach czasowych $t_1, t_2, \ldots, t_{N-1}$, oddzielnie dla każdej zmiennej stanu. Następnie wartości te przyjmowane są jako środki odcinków, które to odcinki podzielone na określoną liczbę podprzedziałów, wyznaczają poszukiwane zbiory warunków początkowych (rys. 8.6). Jako długości odcinków przyjmowane są różnice wartości zmiennych stanu obliczonych z krokami h_2 i h_3 czyli $|x_2 - x_3|$. Wybór jednego warunku początkowego ze zbioru tak wyznaczonych warunków odbywa się w ten sam sposób jak zostało to opisane w rozdziale 8.2.2.

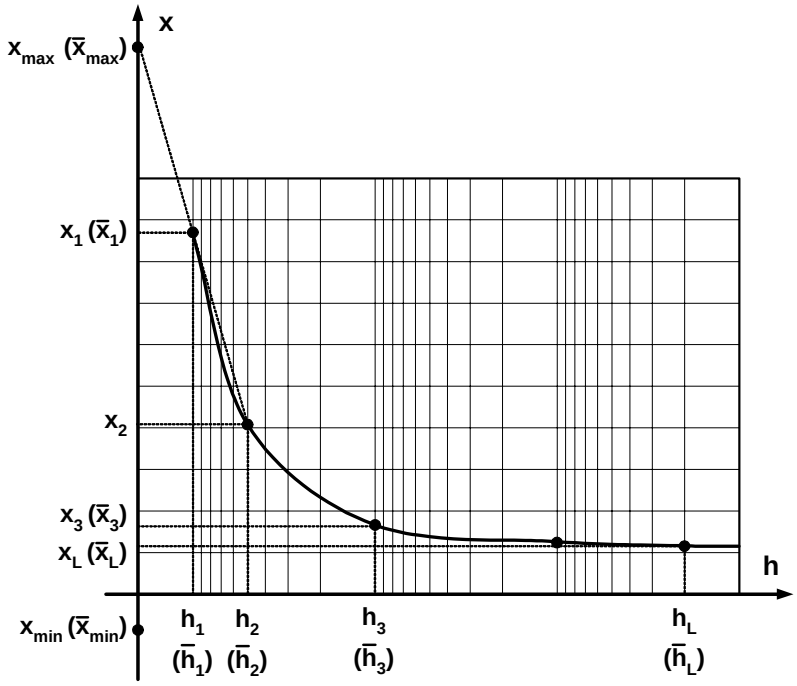
Warunkiem poprawności działania powyższego algorytmu jest monotoniczność zbieżności ciągu rozwiązań przybliżonych do wartości dokładnych, co oznacza, że jeśli $h_1 > h_2 > h_3$, to $x_1 > x_2 > x_3$ lub $x_1 < x_2 < x_3$. Jeśli warunek ten nie jest spełniony, to wykonywane są dodatkowe obliczenia z krokiem $h_4 < h_3$ i ponownie sprawdzana jest monotoniczność zbieżności dla wartości x_2, x_3, x_4 wyznaczonych z krokami h_2, h_3, h_4.

Jeśli ponownie nie będzie monotonicznej zbieżności, to jako warunek początkowy przyjmowana jest wartość obliczona z krokiem h_4.



Rys. 8.6 Wyznaczenie zbiorów warunków początkowych.

Przedstawiony powyżej sposób wyznaczania warunków początkowych dotyczy sytuacji, gdy wykres zbieżności jest funkcją rosnącą. W przypadku, gdy wykres zbieżności opisany jest funkcją malejącą (rys. 8.7) należy odpowiednio zmodyfikować wzory (8.5) - (8.12).



Rys. 8.7 Przeskalowanie wartości na wykresie zbieżności w przypadku, gdy jest on funkcją malejącą.

Przy przeskalowaniu wartości na osi rzędnych wartość maksymalna $x_{\max}$ wyznaczana jest z równania prostej:

$$x_{\max} = \frac{x_2\bar{h}_1 - x_1\bar{h}_2}{\bar{h}_1 - \bar{h}_2} \tag{8.13}$$

zaś wartość minimalna $x_{\min}$:

$$x_{\min} = x_3 - (x_2 - x_3) \tag{8.14}$$

Przeskalowanie wartości x_1 i x_3 ma postać:

$$\bar{x}_1 = X - \frac{X}{x_{\max} - x_{\min}}(x_1 - x_{\min}), \qquad \bar{x}_3 = X - \frac{X}{x_{\max} - x_{\min}}(x_3 - x_{\min}) \tag{8.15}$$

gdzie X jest dodatnią stałą.

Parametry α i A wyznaczane są na podstawie wzorów:

$$\alpha = \frac{\ln(\bar{x}_1 - \bar{x}_3) + \ln\bar{x}_1 - \ln\bar{x}_3}{\bar{h}_3 - \bar{h}_1} \tag{8.16}$$

$$A = X - \frac{\bar{x}_3}{1 - e^{-\alpha\bar{h}_3}} \tag{8.17}$$

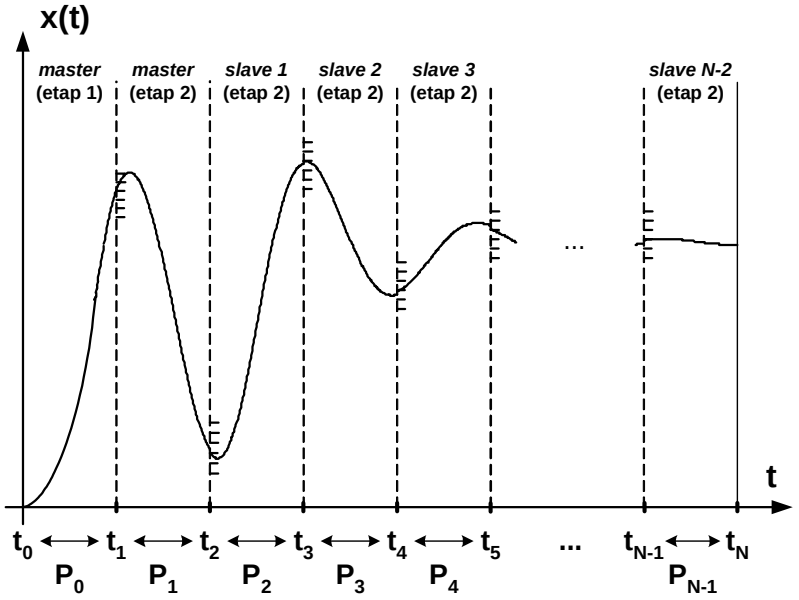
Po wyznaczeniu powyższych parametrów obliczane są przybliżone wartości zmiennych stanu x_L dla małego kroku całkowania h_L:

$$x_L = \frac{A}{X}(x_{\min} - x_{\max}) + x_{\max} \tag{8.18}$$

8.2.4. Określenie rozwiązania końcowego

W drugim etapie algorytmu zmodyfikowanej metody spekulacyjnej wykonywane są równolegle obliczenia w podprzedziałach $(t_1,t_2),\ldots,(t_{N-1},t_N)$ z małą długością kroku całkowania h. Przed rozpoczęciem obliczeń procesor *master* wysyła dane początkowe do pozostałych procesorów (procesorów *slave*). Wszystkie procesory rozpoczynają równolegle obliczenia, przy czym procesor *master* wykonuje obliczenia w podprzedziale (t_1,t_2), pierwszy procesor *slave* - w podprzedziale (t_2,t_3), drugi

procesor *slave* - w podprzedziale (t_3, t_4), itd. (rys. 8.8). Po zakończeniu obliczeń procesory *slave* zwracają wyniki (wartości zmiennych stanu) do procesora *master*, który zapisuje rozwiązanie końcowe zmodyfikowanej metody spekulacyjnej na lokalnym dysku komputera.



Rys. 8.8 Wyznaczenie rozwiązania końcowego w zmodyfikowanej metodzie spekulacyjnej.

8.3. Zastosowanie metody spekulacyjnej w linearyzacji globalnej

8.3.1. Wstęp

Głównym celem stosowania metody spekulacyjnej jest minimalizacja czasu analizy stanu nieustalonego, czyli mówiąc inaczej, obniżenie kosztów analizy układu równań różniczkowych opisujących ten stan nieustalony. Ma to szczególne znaczenie w przypadku długotrwałych obliczeń. Sytuacja taka występuje m.in. w analizie układów nieliniowych po zastosowaniu linearyzacji. Linearyzacja układu nieliniowego [59] jest to zastąpienie układu nieliniowego jego liniowym przybliżeniem. Dla układu liniowego można dokładnie zdefiniować istnienie i jednoznaczność jego rozwiązania oraz warunki stabilności. Istnieje kilka metod linearyzacji układów nieliniowych: rozwinięcie w szereg Taylora wokół punktu równowagi [59], linearyzacja optymalna [52], [53] i linearyzacja globalna [60]. W rozdziale tym przedstawiona zostanie analiza

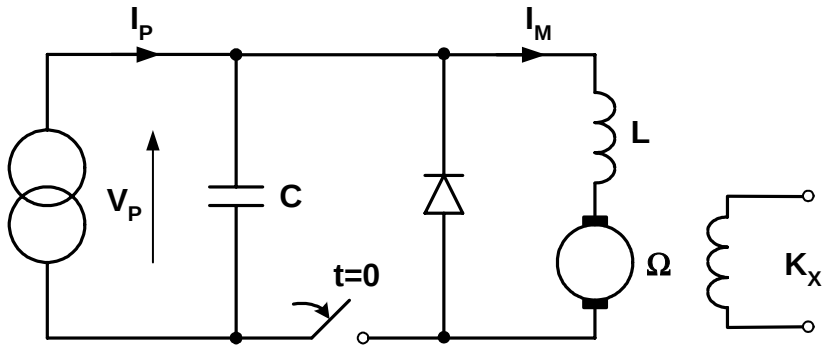
układu zawierającego silnik prądu stałego zasilany z generatora fotowoltaicznego. Równania nieliniowe opisujące stan nieustalony w tym układzie zostały zlinearyzowane przy zastosowaniu linearyzacji globalnej. Otrzymanie dokładnego rozwiązania numerycznego układu zlinearyzowanego wymaga zastosowania małego kroku całkowania, co znacznie wydłuża czas obliczeń. Zastosowanie zmodyfikowanej metody spekulacyjnej umożliwia skrócenie tego czasu z ok. 4,5 godziny do 1,5 godziny. Obliczenia przeprowadzone zostaną przy zastosowaniu obu metod wyznaczania warunków początkowych w zmodyfikowanej metodzie spekulacyjnej.

8.3.2. Nieliniowy model silnika prądu stałego zasilanego z generatora fotowoltaicznego

Schemat układu nieliniowego zawierającego model silnika prądu stałego zasilanego z generatora fotowoltaicznego [5], [52], [58] przedstawiony jest na rys. 8.9. Nieliniowość układu jest wynikiem przyjętej do analizy charakterystyki napięciowo-prądowej generatora fotowoltaicznego:

$$I_P = I_0 - I_S\left(e^{\gamma V_P} - 1\right) \tag{8.19}$$

gdzie I_P, I_0 i I_S są prądami charakteryzującymi generator, V_p jest napięciem generatora, a γ jest stałą.



Rys. 8.9 Schemat elektryczny układu nieliniowego zawierający generator fotowoltaiczny i silnik prądu stałego.

Stan nieustalony w powyższym układzie opisany jest przez trzy nieliniowe równania różniczkowe:

$$\begin{aligned}
\frac{dV_P}{dt} &= -\frac{I_S}{C} e^{\gamma V_P} - \frac{I_M}{C} + u \\
\frac{dI_M}{dt} &= \frac{V_P}{L} - \frac{R_M}{L} I_M - \frac{K_X}{L} \Omega \\
\frac{d\Omega}{dt} &= \frac{K_X}{J} I_M - \frac{K_R}{J} \Omega
\end{aligned} \tag{8.20}$$

gdzie: I_M - prąd silnika, Ω - prędkość obrotowa silnika, K_R, K_X - parametry charakteryzujące silnik, J - moment bezwładności silnika.

Wymuszenie u opisane jest wzorem:

$$u = \frac{I_0 + I_S}{C} \tag{8.21}$$

W analizie przyjęto następujące oznaczenia zmiennych stanu:

$$x_1 = V_P, \quad x_2 = I_M, \quad x_3 = \Omega \tag{8.22}$$

oraz parametry:

$$\begin{aligned}
a_1 &= \frac{I_S}{C}, \quad a_2 = \frac{1}{C}, \quad a_3 = \frac{1}{L}, \quad a_4 = \frac{R_M}{L} \\
a_5 &= \frac{K_X}{L}, \quad a_6 = \frac{K_X}{J}, \quad a_7 = \frac{K_R}{J}, \quad a = \gamma
\end{aligned} \tag{8.23}$$

Obecnie układ równań (8.20) ma nastepującą postać:

$$\begin{aligned}
\dot{x}_1 &= -a_1 e^{a x_1} - a_2 x_2 + u \\
\dot{x}_2 &= a_3 x_1 - a_4 x_2 - a_5 x_3 \\
\dot{x}_3 &= a_6 x_2 - a_7 x_3
\end{aligned} \tag{8.24}$$

z warunkami początkowymi:

$$x_1(0) = V_{P,0}, \quad x_2(0) = 0, \quad x_3(0) = 0 \tag{8.25}$$

8.3.3. Linearyzacja globalna równań opisujących silnik [58][60]

Przed zastosowaniem zmodyfikowanej metody spekulacyjnej układ nieliniowy (8.24) zostanie zlinearyzowany [57], [58] przy wykorzystaniu linearyzacji globalnej [60]. W metodzie tej układ nieliniowy należy przedstawić w postaci ogólnej:

$$\begin{aligned}
\dot{x}_1 &= \phi_1(x_1) + x_2 + g_1(x,u) \\
\dot{x}_2 &= \phi_2(x_1, x_2) + x_3 + g_2(x,u) \\
\dot{x}_3 &= \phi_3(x_1, x_2, x_3) + g_3(x,u) \\
x(0) &= x_0
\end{aligned} \tag{8.26}$$

Następnie definiowana jest zamiana zmiennych stanu:

$$\begin{aligned}
z_1 &= x_1 \\
z_2 &= \phi_1(x_1) + x_2 \\
z_3 &= \phi_2(x_1, x_2) + x_3 \\
z(0) &= \phi[x(0)]
\end{aligned} \tag{8.27}$$

Po zamianie zmiennych stanu i linearyzacji układu (8.26) otrzymywany jest układ równań liniowych (8.28), w którym funkcje $g_1(x,u)$, $\bar{g}_2(x,u)$ i $\bar{g}_3(x,u)$ zawierają wymuszenia oraz nieliniowe kombinacje zmiennych stanu x_1, x_2 i x_3:

$$\begin{aligned}
\dot{z}_1 &= \dot{x}_1 = z_2 + g_1(x,u) \\
\dot{z}_2 &= \frac{\partial \phi_1}{\partial x_1} \cdot \dot{x}_1 + \dot{x}_2 = z_3 + \bar{g}_2(x,u) \\
\dot{z}_3 &= \frac{\partial \phi_2}{\partial x_1} \cdot \dot{x}_1 + \frac{\partial \phi_2}{\partial x_2} \cdot \dot{x}_2 + \dot{x}_3 = -b_1 z_1 - b_2 z_2 - b_3 z_3 + \bar{g}_3(x,u)
\end{aligned} \tag{8.28}$$

gdzie:

$$\begin{aligned}
\bar{g}_2(x,u) &= \frac{\partial \phi_1}{\partial x_1} \cdot \left[\phi_1(x_1) + x_2 + g_1(x,u)\right] + g_2(x,u) \\
\bar{g}_3(x,u) &= \frac{\partial \phi_2}{\partial x_1} \cdot \left[\phi_1(x_1) + x_2 + g_1(x,u)\right] + \frac{\partial \phi_2}{\partial x_2} \cdot \left[\phi_2(x_1, x_2) + x_3 + g_2(x,u)\right] \\
&\quad + \phi_3(x_1, x_2, x_3) + g_3(x,u) + b_1 z_1 + b_2 z_2 + b_3 z_3
\end{aligned} \tag{8.29}$$

Poszukiwane wartości zmiennych stanu $\tilde{x}(t)$ wyznaczane są na podstawie transformacji odwrotnej:

$$\begin{aligned}
\tilde{x}_1 &= z_1 \\
\tilde{x}_2 &= z_2 - \phi_1(z_1) \\
\tilde{x}_3 &= z_3 - \phi_2(z_1, z_2 - \phi_1(z_1))
\end{aligned} \tag{8.30}$$

W przypadku modelu silnika prądu stałego opisanego równaniem (8.24) postać ogólna trzech równań nieliniowych jest następująca:

$$\begin{aligned}\dot{x}_1 &= \phi_1(x_1) - a_2 x_2 + g_1(x,u)\\ \dot{x}_2 &= \phi_2(x_1, x_2) - a_5 x_3 + g_2(x,u)\\ \dot{x}_3 &= \phi_3(x_1, x_2, x_3) + g_3(x,u)\end{aligned} \tag{8.31}$$

gdzie:

$$\begin{aligned}&\phi_1(x_1) = -a_1 e^{ax_1} && g_1(x,u) = u\\ &\phi_2(x_1, x_2) = a_3 x_1 - a_4 x_2 && g_2(x,u) = 0\\ &\phi_3(x_1, x_2, x_3) = a_6 x_2 - a_7 x_3 && g_3(x,u) = 0\end{aligned} \tag{8.32}$$

Po wprowadzeniu zamiany zmiennych:

$$\begin{aligned}z_1 &= x_1\\ z_2 &= -a_1 e^{ax_1} - a_2 x_2\\ z_3 &= a_3 x_1 - a_4 x_2 - a_5 x_3\end{aligned} \tag{8.33}$$

otrzymywany jest układ liniowy:

$$\begin{aligned}\dot{z}_1 &= z_2 + u\\ \dot{z}_2 &= z_3 + \bar{g}_2(x,u)\\ \dot{z}_3 &= -b_1 z_1 - b_2 z_2 - b_3 z_3 + \bar{g}_3(x,u)\end{aligned} \tag{8.34}$$

gdzie:

$$\begin{aligned}\bar{g}_2(x,u) &= a a_1 e^{ax_1}(a_1 e^{ax_1} + a_2 x_2 - u) - (1 + a_2)\cdot(a_3 x_1 - a_4 x_2 - a_5 x_3)\\ \bar{g}_3(x,u) &= a_6 x_1 - a_1 a_4 e^{ax_1} - (a_2 a_4 + a_5 a_6) x_2 + a_5 a_7 x_3 + a_3 u\end{aligned} \tag{8.35}$$

Otrzymanie macierzy **A** prawej strony równania liniowego (8.34) w postaci (8.36) wymagało dokonania dodatkowych przekształceń funkcji $\bar{g}_2(x,u)$.

$$A = \begin{bmatrix} 0 & 1 & 0\\ 0 & 0 & 1\\ -b_1 & -b_2 & -b_3 \end{bmatrix} \tag{8.36}$$

Nowy warunek początkowy wynosi:

$$z_1(0) = V_{p,0}, \quad z_2(0) = -a_1 e^{aV_{p,0}}, \quad z_3(0) = a_3 V_{p,0} \tag{8.37}$$

Współczynniki b_i, $i = 1,2,3$ zostały tak dobrane, aby zapewniały stabilność otrzymanych równań. Wartości tych współczynników wynoszą:

$$\begin{aligned} b_1 &= a_6 \\ b_2 &= a_4 - a_3 \\ b_3 &= a_4 \end{aligned} \tag{8.38}$$

Natomiast przekształcenie odwrotne:

$$\begin{aligned} \tilde{x}_1 &= z_1 \\ \tilde{x}_2 &= -\frac{1}{a_2}(z_2 + a_1 e^{az_1}) \\ \tilde{x}_3 &= \frac{1}{a_5}(-z_3 + a_3 z_1 + \frac{a_4}{a_2}(z_2 + a_1 e^{az_1})) \end{aligned} \tag{8.39}$$

Otrzymanie dokładnego rozwiązania numerycznego zlinearyzowanego układu (8.34) wymaga zastosowania bardzo małego kroku całkowania, co wymusza długi czas obliczeń. W takim przypadku zastosowanie zmodyfikowanej metody spekulacyjnej umożliwi znaczne skrócenie tego czasu.

W obliczeniach numerycznych przyjęto następujące wartości parametrów nieliniowego modelu silnika prądu stałego [52]:

$$\begin{aligned} &I_0 = 2\,A, \quad I_S = 1{,}28 \cdot 10^{-5}\,A, \quad \gamma = 0{,}54 V^{-1} \\ &L = 0{,}1 H, \quad R_M = 12{,}045\,\Omega, \quad C = 500\,\mu F \\ &K_X = 0{,}5 Vs, \quad K_R = 0{,}1 Ws^2 \\ &J = 10^{-3} Ws^3, \quad V_{p,0} = 22{,}15 V \end{aligned} \tag{8.40}$$

8.3.4. Zastosowanie zmodyfikowanej metody spekulacyjnej - wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych

W algorytmie zmodyfikowanej metody spekulacyjnej przyjęto następujące parametry obliczeń:

- sekwencyjna metoda numeryczna rozwiązywania układów równań różniczkowych zwyczajnych: metoda Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania,
- całkowity czas analizy stanu nieustalonego (t_0, t_N): $0{,}0 \div 0{,}09\,s$,
- podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów,
- obliczenia z dużym krokiem całkowania (w celu wyznaczenia warunków początkowych): $h_1 = 5 \cdot 10^{-4}\,s$ i $h_2 = 10^{-4}\,s$,
- obliczenia właściwe z małym krokiem całkowania: $h = 10^{-11}\,s$,
- liczba warunków początkowych w jednym podprzedziale: 10,
- współczynnik bezpieczeństwa: $k = 2{,}0$.

Podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów wynikał z zastosowania w obliczeniach 5 procesorów. Do wyznaczenia warunków początkowych zastosowano krok całkowania $h_1 = 5 \cdot 10^{-4}\,s$ i $h_2 = 10^{-4}\,s$, gdyż krok h_1 jest w rozpatrywanym przykładzie granicą stabilności metody Rungego-Kutty czwartego rzędu i musi być wielokrotnością kroku h_2.

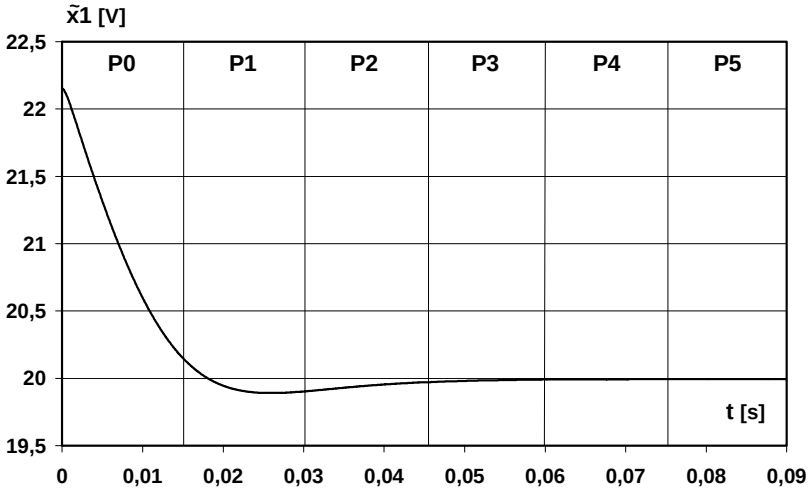
Długość kroku całkowania $h = 10^{-11}\,s$ wyznaczono przy założeniu, że wartość bezwzględna maksymalnej różnicy pomiędzy wartościami zmiennych stanu obliczonymi z kolejnymi krokami całkowania h_j i h_{j-1} ($h_j < h_{j-1}$), różniącymi się o rząd wielkości, jest dla każdej zmiennej stanu mniejsza od założonej wartości $eps = 10^{-6}$:

$$\left| x_{i,h_j} - x_{i,h_{j-1}} \right|_{\max} < eps, \qquad i = 1,2,3 \tag{8.41}$$

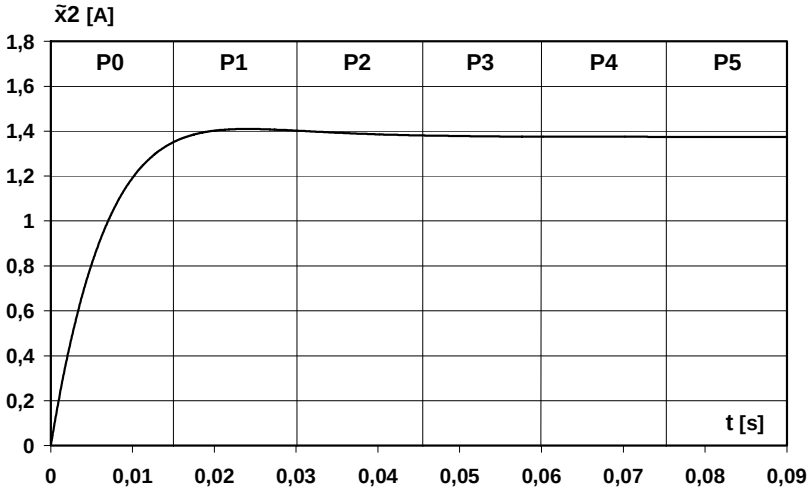
W każdym podprzedziale określono 10 warunków początkowych, gdyż, jak stwierdzono eksperymentalnie, zwiększenie tej liczby nie ma znaczącego wpływu na dokładność rozwiązania. Wartość współczynnika bezpieczeństwa k wynika z zastosowania metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania, dla której $k = 2{,}0$.

8.3.4.1. Wyniki obliczeń

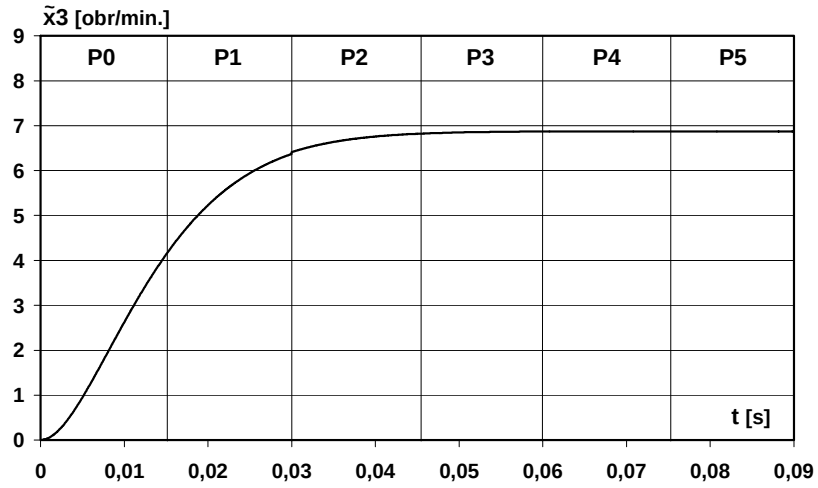
Obliczenia przeprowadzono na pięciu węzłach systemu klaster KETIM2, który został opisany w rozdziale 5.2. Rys. 8.10 - rys. 8.12 przedstawiają otrzymane rozwiązania dla wszystkich trzech zmiennych stanu, w przypadku podziału całkowitego czasu analizy stanu nieustalonego na sześć podprzedziałów (zbiorów procesów równoległych - $P_0, P_1, \ldots, P_5$). Granice podprzedziałów zaznaczono na wykresach.



Rys. 8.10 Przebieg napięcia generatora fotowoltaicznego - $\tilde{x}_1(t)$.



Rys. 8.11 Przebieg prądu silnika - $\tilde{x}_2(t)$.



Rys. 8.12 Przebieg prędkości obrotowej silnika - $\tilde{x}_3(t)$.

W celu oszacowania dokładności rozwiązania zmodyfikowanej metody spekulacyjnej układ trzech równań nieliniowych (8.24) został rozwiązany przy wykorzystaniu algorytmu sekwencyjnego metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-11}\,s$, taką samą jak w zmodyfikowanej metodzie spekulacyjnej.

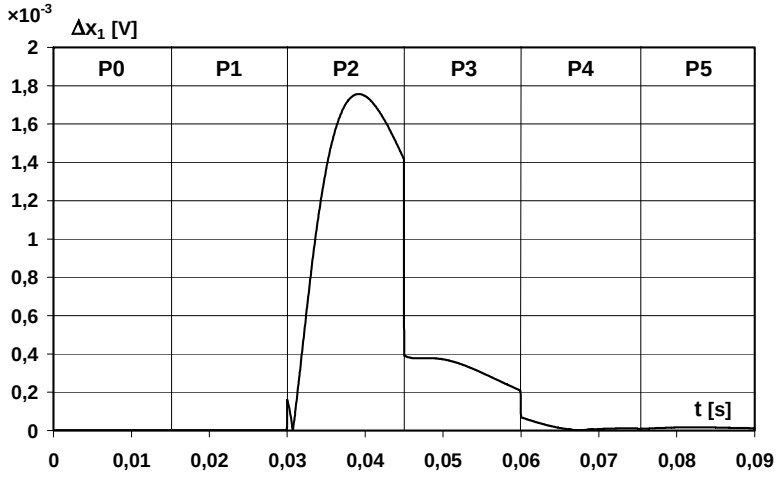
Na podstawie otrzymanych wyników obliczeń sekwencyjnych oraz spekulacyjnych wyznaczono wartości błędów bezwzględnych (8.42) i względnych (8.43) na całej długości przedziału całkowania (t_0, t_N).

$$\Delta x_i = \left| x_i - \tilde{x}_i \right|, \qquad i = 1,2,3 \tag{8.42}$$

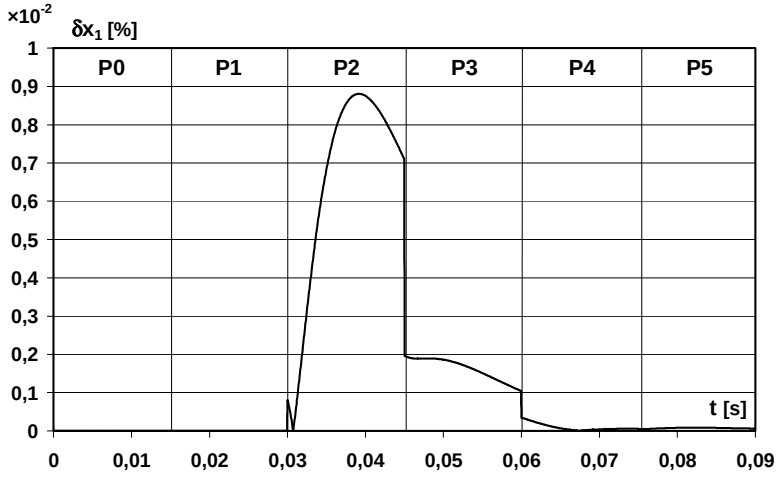
$$\delta x_i = \frac{\left| x_i - \tilde{x}_i \right|}{x_i} \cdot 100\,\%, \qquad i = 1,2,3 \tag{8.43}$$

gdzie: x_i - wartości obliczone przy wykorzystaniu algorytmu sekwencyjnego, $\tilde{x}_i$ - wartości obliczone przy wykorzystaniu algorytmu spekulacyjnego.

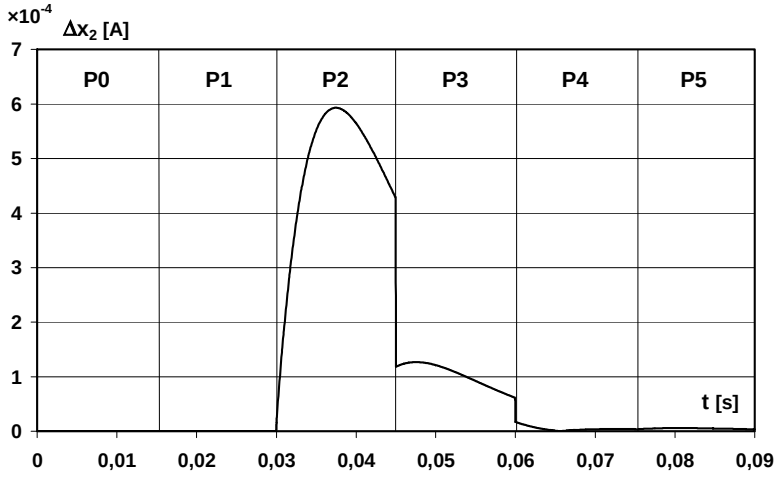
Otrzymane błędy bezwzględne i względne rozwiązania spekulacyjnego przedstawiono na rys. 8.13 - rys. 8.18.



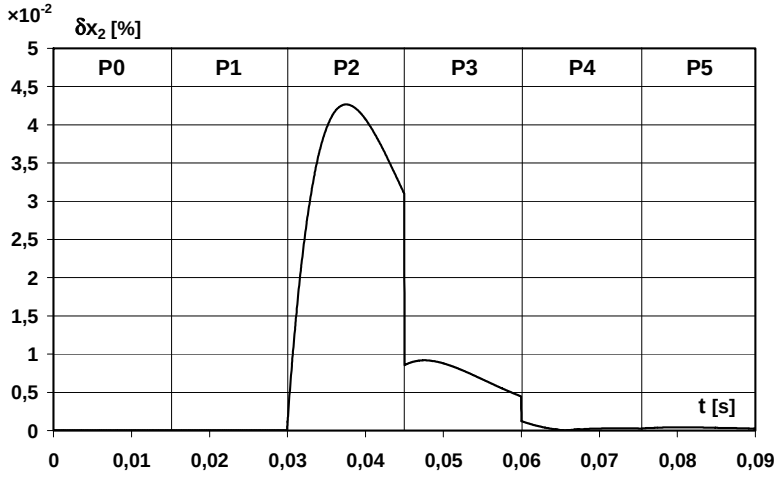
Rys. 8.13 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_1(t)$.



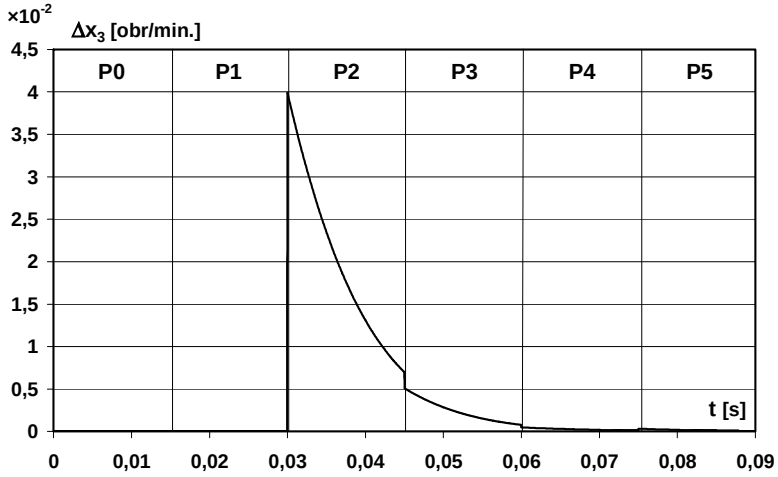
Rys. 8.14 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_1(t)$.



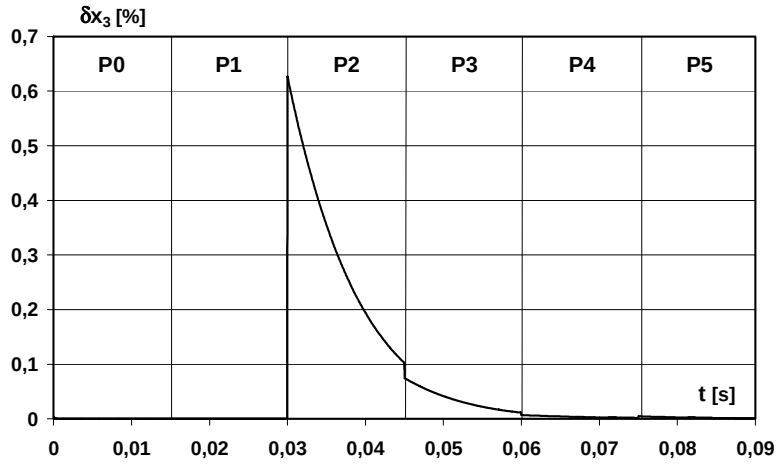
Rys. 8.15 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_2(t)$.



Rys. 8.16 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_2(t)$.



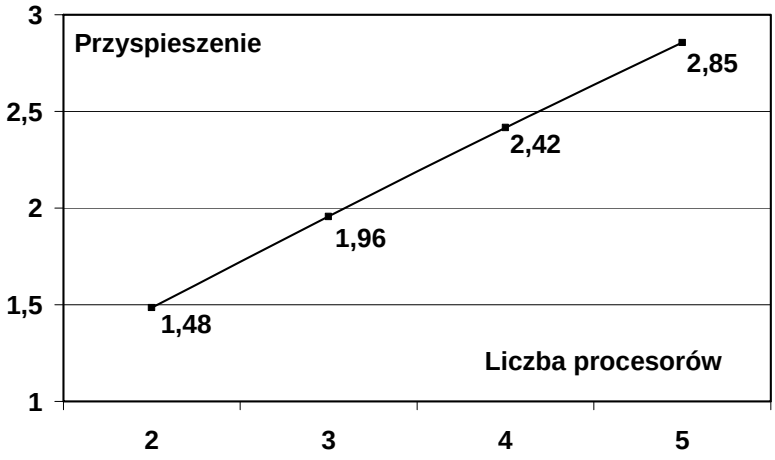
Rys. 8.17 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_3(t)$.



Rys. 8.18 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_3(t)$.

Z analizy powyższych wykresów błędu wynika, że otrzymane rozwiązanie jest dokładne - maksymalny błąd względny w przypadku wszystkich zmiennych stanu nie przekroczył 1 %. Praktycznie zerowe wartości błędów w pierwszych dwóch podprzedziałach wynikają z tego, iż obliczenia w pierwszym podprzedziale wykonywane są z warunkami początkowymi znanymi z założenia oraz z takim samym krokiem całkowania jak obliczenia sekwencyjne. Podobnie jest w przypadku drugiego podprzedziału. Jako warunki początkowe w tym podprzedziale brane są wartości obliczone na końcu pierwszego podprzedziału.

Do wyznaczenia przyspieszenia obliczeń całkowity czas analizy stanu nieustalonego podzielono na 3, 4, 5 i 6 podprzedziałów (w przypadku 2 podprzedziałów wykorzystywany jest tylko jeden procesor). Obliczenia sekwencyjne przeprowadzono rozwiązując układ zlinearyzowany (8.34) przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty ze stałą długością kroku całkowania $h = 10^{-11}\,s$. Wykres przyspieszenia przedstawiono na rys. 8.19.



Rys. 8.19 Przyspieszenie obliczeń w zmodyfikowanej metodzie spekulacyjnej (wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych).

Otrzymane wartości przyspieszenia są zadawalające. Różnice w stosunku do przypadku idealnego (wartość przyspieszenia równa liczbie zastosowanych procesorów) są spowodowane tym, iż pierwszy etap algorytmu zmodyfikowanej metody spekulacyjnej wykonywany jest sekwencyjnie (wyznaczanie warunków początkowych, obliczenia w podprzedziale (t_0, t_1) z małym krokiem całkowania),

a dopiero obliczenia w pozostałych podprzedziałach $(t_1,t_2),\ldots,(t_5,t_6)$, w drugim etapie algorytmu metody, wykonywane są równolegle.

8.3.4.2. Wnioski

Otrzymanie dokładnego rozwiązania numerycznego układu zlinearyzowanego wymaga zastosowania bardzo małego kroku całkowania, co powoduje wydłużenie czasu obliczeń. Zastosowanie zmodyfikowanej metody spekulacyjnej umożliwiło znaczne skrócenie tego czasu, co **potwierdza tezę rozprawy**. W przedstawionym przykładzie analizy nieliniowego modelu silnika prądu stałego zasilanego z generatora fotowoltaicznego otrzymano czasy obliczeń przedstawione w tabeli 8.1.

Tabela 8.1 Zestawienie czasów obliczeń w analizie nieliniowego modelu silnika prądu stałego zasilanego z generatora fotowoltaicznego

Liczba procesorów	**Czas obliczeń [s]**	**Czas obliczeń [hh : mm : ss]**
1	15 961	04 : 26 : 01
2	10 754	02 : 59 : 14
3	8 160	02 : 16 : 00
4	6 607	01 : 50 : 07
5	5 591	01 : 33 : 11

8.3.5. Zastosowanie zmodyfikowanej metody spekulacyjnej - wyznaczenie warunków początkowych na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą

W obliczeniach numerycznych przyjęto wartości parametrów nieliniowego modelu silnika prądu stałego i generatora fotowoltaicznego przedstawione w rozdziale 8.3.4 - wzór (8.40). W algorytmie zmodyfikowanej metody spekulacyjnej przyjęto następujące parametry obliczeń:

- sekwencyjna metoda numeryczna rozwiązywania układów równań różniczkowych zwyczajnych: metoda Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania,

- całkowity czas analizy stanu nieustalonego (t_0, t_N): $0{,}0 \div 0{,}09\,s$,
- podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów,
- obliczenia z dużym krokiem całkowania (wyznaczenie warunków początkowych): $h_1 = 5 \cdot 10^{-4}\ s$, $h_2 = 10^{-4}\ s$ i $h_3 = 5 \cdot 10^{-5}\ s$,
- obliczenia właściwe z małym krokiem całkowania: $h = 10^{-11}\ s$,
- liczba punktów podziału odcinka w celu wyznaczenia warunków początkowych: 10.

Uzasadnienie wyboru wartości powyższych parametrów jest takie samo jak w rozdziale 8.3.4.

8.3.5.1. Wyniki obliczeń

Obliczenia przeprowadzono przy wykorzystaniu pięciu węzłów systemu klaster KETIM2, który został opisany w rozdziale 5.2. Tabela 8.2 zawiera wartości zmiennych stanu na początku każdego podprzedziału, tj. w chwilach czasowych $t_1, t_2, ..., t_5$, otrzymane w algorytmie spekulacyjnym, w przypadku podziału całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów: $(t_0, t_1), (t_1, t_2), ..., (t_5, t_6)$. Wartości te są warunkami początkowymi wyznaczonymi na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą.

Tabela 8.2 Warunki początkowe wyznaczone w zmodyfikowanej metodzie spekulacyjnej

	t_1 = **0,015**	t_2 = **0,03**	t_3 = **0,045**	t_4 = **0,06**	t_5 = **0,075**
x_1 **[V]**	20,14984	19,90160	19,96913	19,99004	19,99349
x_2 **[A]**	1,35105	1,40317	1,38168	1,37566	1,37472
x_3 **[obr/min.]**	4,13246	6,37895	6,81463	6,86986	6,87362

W celu oceny dokładności wyznaczenia warunków początkowych i dokładności rozwiązania metody spekulacyjnej układ nieliniowy (8.24) rozwiązano przy wykorzystaniu sekwencyjnego algorytmu metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-11}\ s$. Otrzymane wartości w punktach

będących granicami podprzedziałów zawiera tabela 8.3, zaś tabela 8.4 i tabela 8.5 zawierają wartości błędów bezwzględnych (8.44) i względnych (8.45) wyznaczenia warunków początkowych.

$$\Delta x_i = \left| x_i - \tilde{x}_i \right|, \qquad i = 1,2,3 \tag{8.44}$$

$$\delta x_i = \frac{\left| x_i - \tilde{x}_i \right|}{x_i} \cdot 100\,\%, \qquad i = 1,2,3 \tag{8.45}$$

gdzie: x_i - wartości obliczone przy wykorzystaniu algorytmu sekwencyjnego (rozwiązanie układu nieliniowego), $\tilde{x}_i$ - wartości obliczone przy wykorzystaniu algorytmu spekulacyjnego (rozwiązanie układu zlinearyzowanego).

Tabela 8.3 Wartości wyznaczone przy wykorzystaniu sekwencyjnego algorytmu metody Rungego-Kutty z krokiem całkowania $h = 10^{-11}\ s$

	t_1 = **0,015**	t_2 = **0,03**	t_3 = **0,045**	t_4 = **0,06**	t_5 = **0,075**
x_1 **[V]**	20,14984	19,90166	19,96911	19,99003	19,99349
x_2 **[A]**	1,35105	1,40264	1,38156	1,37565	1,37472
x_3 **[obr/min.]**	4,13246	6,37078	6,81346	6,86967	6,87354

Tabela 8.4 Błędy bezwzględne wyznaczenia warunków początkowych

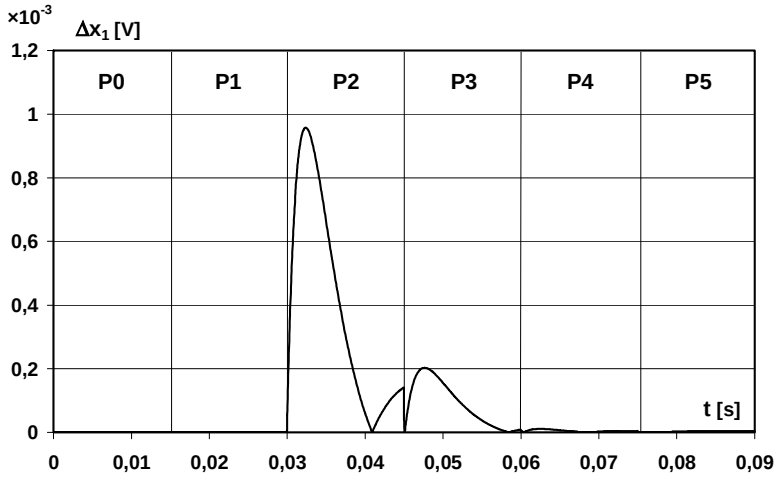
	t_1 = **0,015**	t_2 = **0,03**	t_3 = **0,045**	t_4 = **0,06**	t_5 = **0,075**
Δx_1 **[V]**	$3{,}0\cdot10^{-10}$	$5{,}6\cdot10^{-5}$	$2{,}3\cdot10^{-5}$	$6{,}2\cdot10^{-6}$	$6{,}3\cdot10^{-7}$
Δx_2 **[A]**	0	$5{,}3\cdot10^{-4}$	$1{,}1\cdot10^{-4}$	$7{,}8\cdot10^{-6}$	$2{,}8\cdot10^{-7}$
Δx_3 **[obr/min.]**	$6{,}7\cdot10^{-9}$	$8{,}2\cdot10^{-3}$	$1{,}2\cdot10^{-3}$	$1{,}9\cdot10^{-4}$	$8{,}0\cdot10^{-5}$

Tabela 8.5 Błędy względne wyznaczenia warunków początkowych

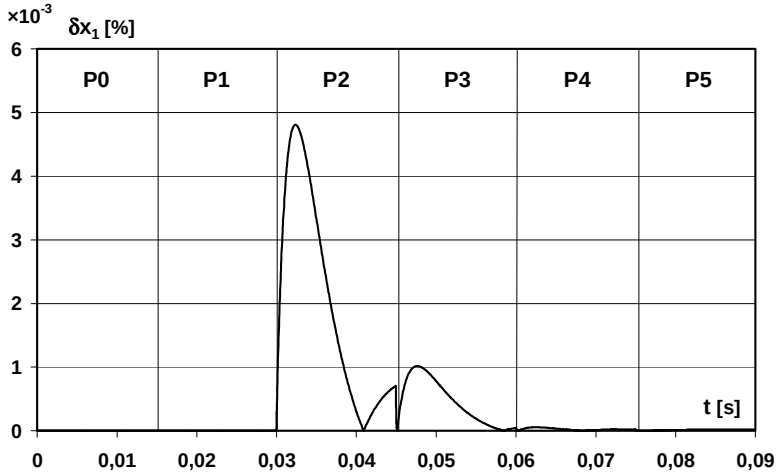
	t_1 = **0,015**	t_2 = **0,03**	t_3 = **0,045**	t_4 = **0,06**	t_5 = **0,075**
δx_1 **[%]**	$1{,}5\cdot10^{-9}$	$2{,}8\cdot10^{-4}$	$1{,}2\cdot10^{-4}$	$3{,}1\cdot10^{-5}$	$3{,}2\cdot10^{-6}$
δx_2 **[%]**	0	$3{,}8\cdot10^{-2}$	$8{,}3\cdot10^{-3}$	$5{,}7\cdot10^{-4}$	$2{,}0\cdot10^{-5}$
δx_3 **[%]**	$1{,}6\cdot10^{-7}$	$1{,}3\cdot10^{-1}$	$1{,}7\cdot10^{-2}$	$2{,}8\cdot10^{-3}$	$1{,}2\cdot10^{-3}$

Najmniejsze wartości błędów występują w chwili czasowej t_1, gdyż do obliczeń w tym punkcie nie jest brana wartość wyznaczona na podstawie przybliżenia wykresu zbieżności funkcją wykładniczą, ale wartość obliczona w pierwszym etapie algorytmu metody spekulacyjnej, czyli z warunkiem początkowym znanym z założenia.

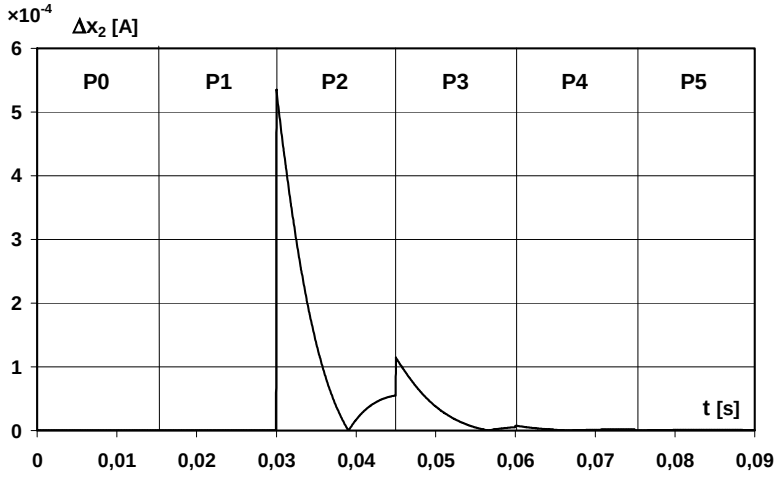
Otrzymane rozwiązania zmiennych stanu są takie same jak w przypadku zastosowania zmodyfikowanej metody spekulacyjnej (rys. 8.10 - rys. 8.12) dlatego zostaną pominięte. Przedstawione natomiast zostaną wykresy błędów bezwzględnych (rys. 8.20, rys. 8.22, rys. 8.24) i błędów względnych (rys. 8.21, rys. 8.23, rys. 8.25) rozwiązania spekulacyjnego w całym przedziale (t_0, t_N).



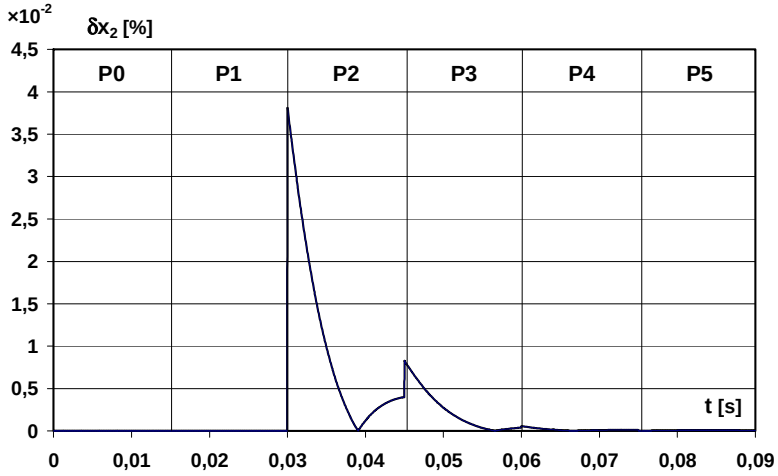
Rys. 8.20 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_1(t)$.



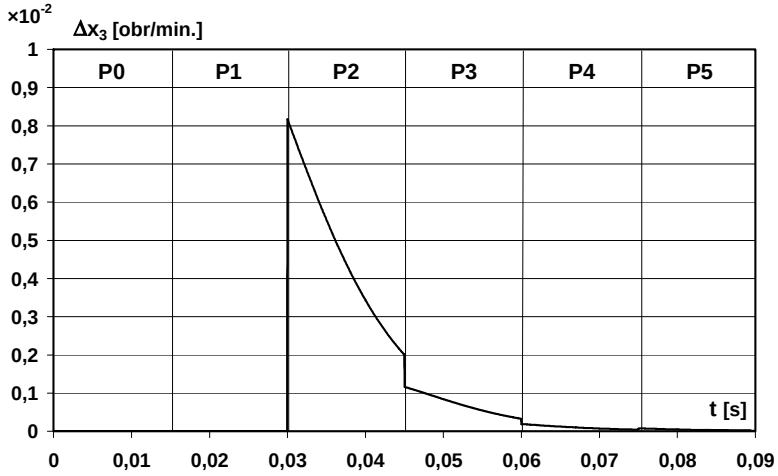
Rys. 8.21 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_1(t)$.



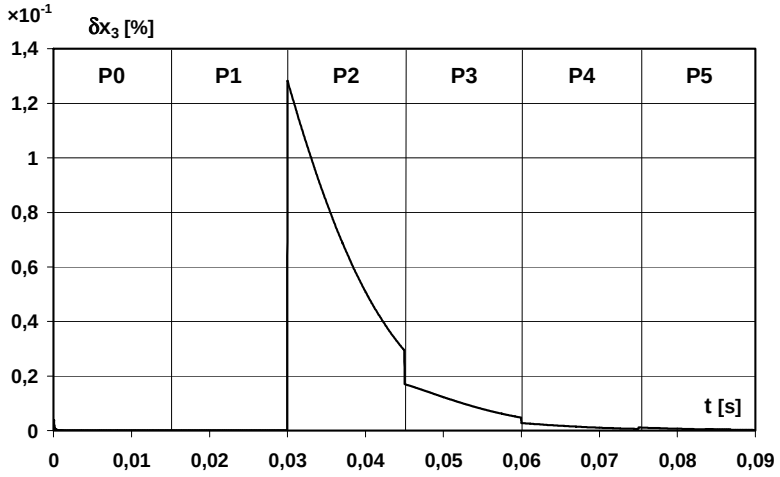
Rys. 8.22 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_2(t)$.



Rys. 8.23 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_2(t)$.

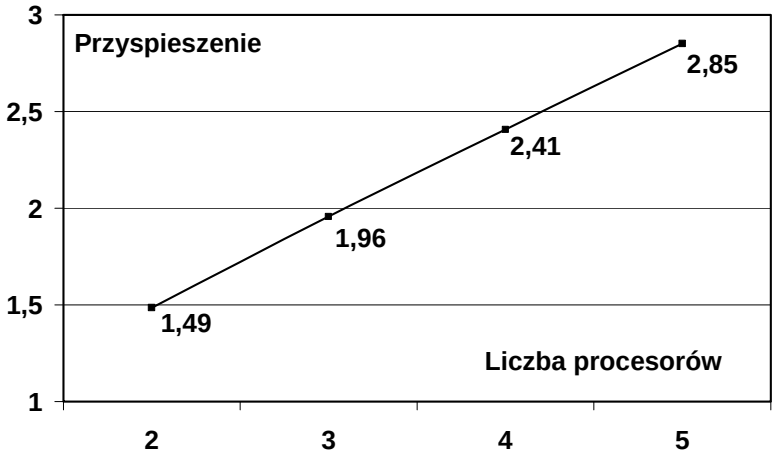


Rys. 8.24 Błędy bezwzględne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_3(t)$.



Rys. 8.25 Błędy względne rozwiązania spekulacyjnego dla zmiennej stanu $\tilde{x}_3(t)$.

Wyznaczając przyspieszenie obliczeń całkowity czas analizy stanu nieustalonego podzielono na 3, 4, 5 i 6 podprzedziałów. Obliczenia sekwencyjne przeprowadzono rozwiązując układ zlinearyzowany przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-11}\,s$. Otrzymane wartości przyspieszenia przedstawiono na rys. 8.26. Ponieważ wyznaczenie warunków początkowych ma niewielki wpływ na całkowity czas obliczeń, więc otrzymane wartości przyspieszenia (rys. 8.26) są zbliżone do wartości przedstawionych w rozdziale 8.3.4.1 (rys. 8.19).



Rys. 8.26 Przyspieszenie obliczeń w zmodyfikowanej metodzie spekulacyjnej (wyznaczenie warunków początkowych na podstawie przybliżenia wykresu zbieżności rozwiązania funkcją wykładniczą).

8.3.5.2. Wnioski

Zastosowanie metody wyznaczania warunków początkowych opartej na przybliżeniu wykresu zbieżności rozwiązania funkcją wykładniczą umożliwia wyznaczenie warunków początkowych bliskich wartościom dokładnym. Zapewnia to dobrą dokładność rozwiązania zmodyfikowanej metody spekulacyjnej. Konieczność wykonania obliczeń w pierwszym etapie algorytmu z dużymi wartościami kroków h_1, h_2 i h_3 na całym przedziale nie wpływa znacząco na całkowity czas obliczeń - dla obu algorytmów wyznaczania zbiorów warunków początkowych otrzymano prawie takie same wartości przyspieszenia obliczeń oraz ich skrócenie w stosunku do obliczeń sekwencyjnych - **potwierdza to tezę rozprawy**.

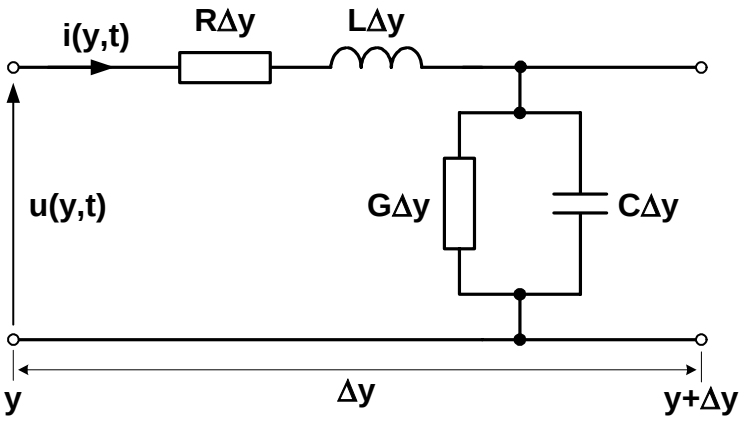
8.4. Zastosowanie zmodyfikowanej metody spekulacyjnej do analizy stanów nieustalonych w linii długiej

8.4.1. Wstęp

Jako drugi przykład zastosowania zmodyfikowanej metody spekulacyjnej przedstawiona zostanie analiza stanów nieustalonych w linii długiej. Rozkład prądu i napięcia w linii długiej opisany jest układem dwóch równań różniczkowych cząstkowych pierwszego rzędu. Równania te zastępowane są układem równań różniczkowych zwyczajnych poprzez aproksymację pochodnych cząstkowych za pomocą różnic skończonych. Liczba równań różniczkowych zwyczajnych jest dwa razy większa od liczby segmentów, na które dzielona jest linia. Duża dokładność obliczeń wymaga dużej liczby segmentów, a zatem i dużej liczby równań. Rozwiązanie numeryczne takiego układu może być czasochłonne. Zastosowanie zmodyfikowanej metody spekulacyjnej umożliwi skrócenie tego czasu. Stany nieustalone w linii długiej zostaną zbadane przy założeniu wymuszenia w postaci impulsu napięciowego i zwarcia linii na jej końcu.

8.4.2. Model linii długiej

Analiza stanów nieustalonych w linii elektrycznej polega na określeniu przebiegów zmian napięcia między przewodami linii $u(y,t)$ i prądu linii $i(y,t)$ w każdym punkcie *y* linii (gdzie *y* jest odległością od początku linii) i w każdej chwili czasowej *t*. Linia elektryczna stanowi obwód o parametrach rozłożonych. Dostatecznie krótki odcinek linii o długości Δy można zastąpić układem o parametrach skupionych. Najczęściej odcinek ten przybliża się za pomocą czwórnika skupionego [66], [81], [90] (rys. 8.27). W czwórniku tym występują parametry skupione: $R\Delta y$, $L\Delta y$, $G\Delta y$, $C\Delta y$, gdzie *R* jest rezystancją jednostkową linii (Ω/m), *L* impedancją jednostkową linii (H/m), *G* konduktancją jednostkową linii (S/m), zaś *C* pojemnością jednostkową linii (F/m).



Rys. 8.27 Przybliżenie odcinka linii o długości Δy za pomocą czwórnika skupionego.

Wyznaczenie zależności między napięciami i prądami w powyższym czwórniku i przejście do granicy $\Delta y \to 0$, umożliwia otrzymanie układu dwóch równań różniczkowych cząstkowych pierwszego rzędu opisujących rozkład napięcia $u(y,t)$ i prądu $i(y,t)$ w linii [66], [81]:

$$\begin{aligned} -\frac{\partial u(y,t)}{\partial y} &= R(y)\cdot i(y,t) + L(y)\frac{\partial i(y,t)}{\partial t} \\ -\frac{\partial i(y,t)}{\partial y} &= G(y)\cdot u(y,t) + C(y)\frac{\partial u(y,t)}{\partial t} \end{aligned} \tag{8.46}$$

Powyższe równania nazywane są równaniami *linii długiej*, a linia elektryczna opisywana tymi równaniami - *linią długą*.

Linię długą można rozpatrywać jako połączenie łańcuchowe czwórników skupionych, stanowiących modele dostatecznie krótkich odcinków linii, gdy długość tych odcinków zmierza do zera, a ich liczba rośnie do nieskończoności [81]. Zakładając podział linii na *K* segmentów o węzłach y_k, $k = 0,1,\ldots,K$ równania (8.46) zastępowane są układem równań różniczkowych zwyczajnych poprzez aproksymację pochodnych cząstkowych za pomocą różnic skończonych [6]:

$$\left.\frac{\partial u(y,t)}{\partial y}\right|_{y=y_k} \cong \frac{u(y_{k+1},t)-u(y_{k-1},t)}{y_{k+1}-y_{k-1}}$$
$$\left.\frac{\partial i(y,t)}{\partial y}\right|_{y=y_k} \cong \frac{i(y_{k+1},t)-i(y_{k-1},t)}{y_{k+1}-y_{k-1}} \tag{8.47}$$

Oznaczając $u_k(t)$ jako napięcie $u(y_k,t)$ w węźle y_k, a $i_k(t)$ jako prąd $i(y_k,t)$ w węźle y_k definiowane są zmienne stanu napięciowe i prądowe dla każdego węzła podziału linii długiej na segmenty. Równania (8.46) po zastosowaniu podstawienia (8.47) zostają przybliżone układem równań różniczkowych zwyczajnych [6]:

$$\frac{du_k(t)}{dt} = -\frac{G(y_k)}{C(y_k)}u_k(t) - \frac{i_{k+1}(t)-i_{k-1}(t)}{C(y_k)\cdot(y_{k+1}-y_{k-1})}$$
$$\frac{di_k(t)}{dt} = -\frac{R(y_k)}{L(y_k)}i_k(t) - \frac{u_{k+1}(t)-u_{k-1}(t)}{L(y_k)\cdot(y_{k+1}-y_{k-1})} \tag{8.48}$$

Warunki brzegowe na początku linii (dla $y = 0$) $u(0,t)$ i $i(0,t)$ dla układu równań różniczkowych cząstkowych stają się członem wymuszającym dla układu równań różniczkowych zwyczajnych [6]:

$$i_0(t) = i(0,t)$$
$$u_0(t) = u(0,t) \tag{8.49}$$

Natomiast warunki początkowe (dla $t = 0$) układu równań różniczkowych cząstkowych $u(y,0)$ i $i(y,0)$ stają się warunkami początkowymi dla układu równań różniczkowych zwyczajnych [6]:

$$\begin{aligned} i_k(0) &= i(y_k,0) \\ u_k(t) &= u(y_k,0) \end{aligned} \qquad k = 0,1,\ldots,K \tag{8.50}$$

W przypadku modelowania linii długiej w stanie zwarcia napięcie na końcu linii jest równe zeru - $u_K(t) = 0$, natomiast na początku linii podawane jest wymuszenie napięciowe $u_0(t)$. Wektor stanu ma zatem następującą postać:

$$x(t) = \left[i_0(t), i_1(t), u_1(t), i_2(t), ..., i_k(t), u_k(t), ..., i_{K-1}(t), u_{K-1}(t), i_K(t)\right]^T \tag{8.51}$$

Ten sam wektor po ponumerowaniu zmiennych:

$$x(t) = \left[x_1(t), x_2(t), x_3(t), x_4(t), ..., x_{2k}(t), x_{2k+1}(t), ..., x_{2K-2}(t), x_{2K-1}(t), x_{2K}(t)\right]^T \tag{8.52}$$

Kolejne równania opisujące rozkład napięcia i prądu w linii długiej mają następującą postać:

$$\begin{aligned}
\frac{dx_1(t)}{dt} &= -\frac{R(y_0)}{L(y_0)}x_1(t) - \frac{x_3(t) - u_0(t)}{L(y_0)\cdot(y_1 - y_0)} \\
\frac{dx_2(t)}{dt} &= -\frac{R(y_1)}{L(y_1)}x_2(t) - \frac{x_5(t) - u_0(t)}{L(y_1)\cdot(y_2 - y_0)} \\
\frac{dx_3(t)}{dt} &= -\frac{G(y_1)}{C(y_1)}x_3(t) - \frac{x_4(t) - x_1(t)}{C(y_1)\cdot(y_2 - y_0)} \\
&\ldots \\
\frac{dx_{2k}(t)}{dt} &= -\frac{R(y_k)}{L(y_k)}x_{2k}(t) - \frac{x_{2k+3}(t) - x_{2k-1}(t)}{L(y_k)\cdot(y_{k+1} - y_{k-1})} \\
\frac{dx_{2k+1}(t)}{dt} &= -\frac{G(y_k)}{C(y_k)}x_{2k+1}(t) - \frac{x_{2k+2}(t) - x_{2k-2}(t)}{C(y_k)\cdot(y_{k+1} - y_{k-1})} \\
&\ldots \\
\frac{dx_{2K-2}(t)}{dt} &= -\frac{R(y_{K-1})}{L(y_{K-1})}x_{2K-2}(t) - \frac{u_K(t) - x_{2K-3}(t)}{L(y_{K-1})\cdot(y_K - y_{K-2})} \\
\frac{dx_{2K-1}(t)}{dt} &= -\frac{G(y_{K-1})}{C(y_{K-1})}x_{2K-1}(t) - \frac{x_{2K}(t) - x_{2K-4}(t)}{C(y_{K-1})\cdot(y_K - y_{K-2})} \\
\frac{dx_{2K}(t)}{dt} &= -\frac{R(y_K)}{L(y_K)}x_{2K}(t) - \frac{u_K(t) - x_{2K-1}(t)}{L(y_K)\cdot(y_K - y_{K-1})}
\end{aligned} \tag{8.53}$$

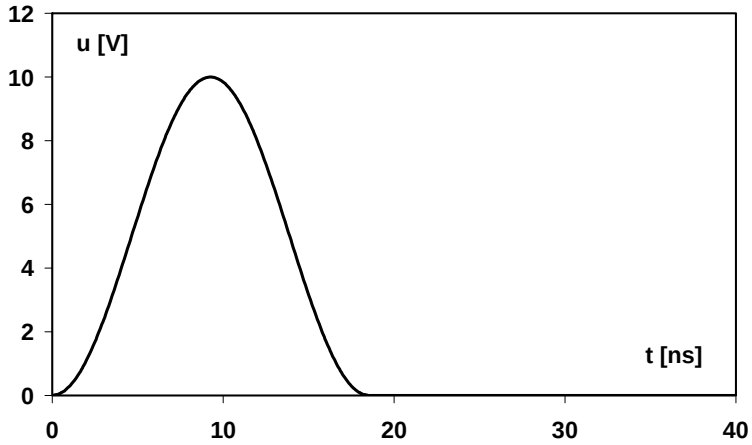
Liczba tych równań jest dwukrotnie większa od liczby segmentów, na które została podzielona linia. Równania z indeksami parzystymi opisują prądy, natomiast równania z indeksami nieparzystymi (poza pierwszym równaniem) - napięcia.

8.4.3. Zastosowanie zmodyfikowanej metody spekulacyjnej

Do obliczeń przyjęto linię długą współosiową (kablową), jednorodną, o długości 10 m, średnicy przewodów 2 mm i średnicy rdzenia 4 mm. Parametry pierwotne linii wynoszą [6]:

$$\begin{aligned} R &= 19{,}733\, m\Omega / m, \qquad L = 318{,}848\, nH / m \\ C &= 104{,}307\ pF / m, \qquad G = 6{,}554\, nS / m \end{aligned} \tag{8.54}$$

Zakładając obserwację wartości napięć oraz prądów linii co 1 cm, linię podzielono na $K = 1000$ segmentów otrzymując układ 2000 równań różniczkowych zwyczajnych o postaci (8.53). Jako wymuszenie na początku linii podano impuls napięciowy o amplitudzie 10 V i czasach narastania i opadania równych 9,25 ns. Przebieg impulsu wymuszającego przedstawiony jest na rys. 8.28.



Rys. 8.28 Przebieg impulsu wymuszającego.

W algorytmie zmodyfikowanej metody spekulacyjnej (rozdział 8.2) przyjęto następujące parametry obliczeń:

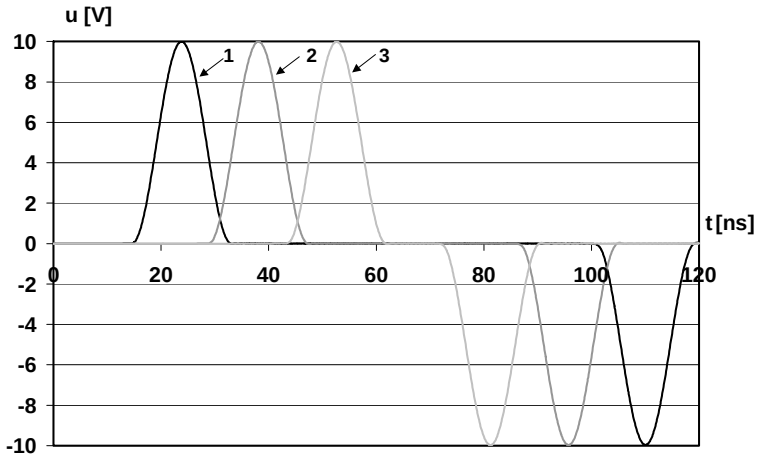
- metoda numeryczna rozwiązywania układów równań różniczkowych: metoda Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania *h*,

- całkowity czas analizy stanu nieustalonego (t_0, t_N): $0 \div 120\,ns$,
- podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów,
- wyznaczenie warunków początkowych na podstawie analizy rozwiązań przybliżonych, wyznaczonych z krokami: $h_1 = 10^{-10}\,s$ i $h_2 = 5 \cdot 10^{-11}\,s$,
- obliczenia właściwe z małym krokiem całkowania: $h = 10^{-13}\,s$,
- liczba warunków początkowych w jednym podprzedziale: 10,
- współczynnik bezpieczeństwa: $k = 2{,}0$.

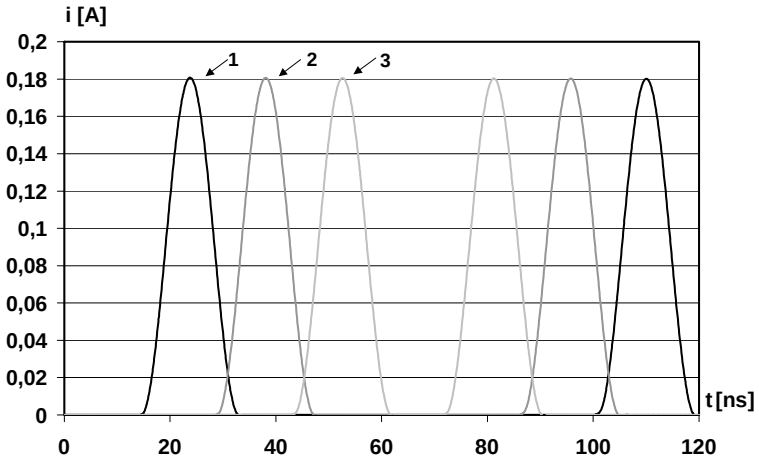
Podział całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów wynikał z liczby 5 procesorów zastosowanych w obliczeniach. W analizowanym przykładzie graniczną wartością kroku całkowania, dla której rozwiązanie jest stabilne, jest krok $h_1 = 10^{-10}\,s$. Z tego powodu warunki początkowe zostały wyznaczone na podstawie rozwiązań przybliżonych, otrzymanych z krokami całkowania $h_1 = 10^{-10}\,s$ i $h_2 = 5 \cdot 10^{-11}\,s$. Obliczenia właściwe przeprowadzono z krokiem całkowania $h = 10^{-13}\,s$ ze względu na bardzo małą długość przedziału całkowania ($120\,ns$) oraz ze względu na długość impulsu wymuszającego. Dla każdej zmiennej stanu określono 10 warunków początkowych, gdyż, jak stwierdzono eksperymentalnie, zwiększenie tej liczby nie ma znaczącego wpływu na dokładność rozwiązania. Wartość współczynnika bezpieczeństwa $k = 2{,}0$ wynika z zastosowania w obliczeniach metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania.

8.4.4. Wyniki obliczeń

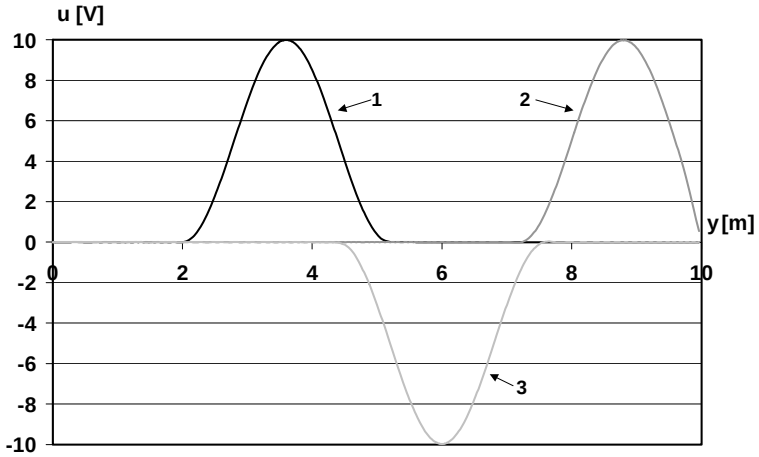
Obliczenia wykonano przy wykorzystaniu systemu klaster KETIM2. Rys. 8.29 i rys. 8.30 przedstawiają rozkłady napięcia i prądu w trzech wybranych punktach linii długiej. Wyniki te zostały otrzymane w przypadku podziału całkowitego czasu analizy stanu nieustalonego na 6 podprzedziałów. Kolejne rysunki, rys. 8.31 i rys. 8.32, przedstawiają rozkład napięcia i prądu w całej linii długiej w trzech wybranych chwilach czasowych.



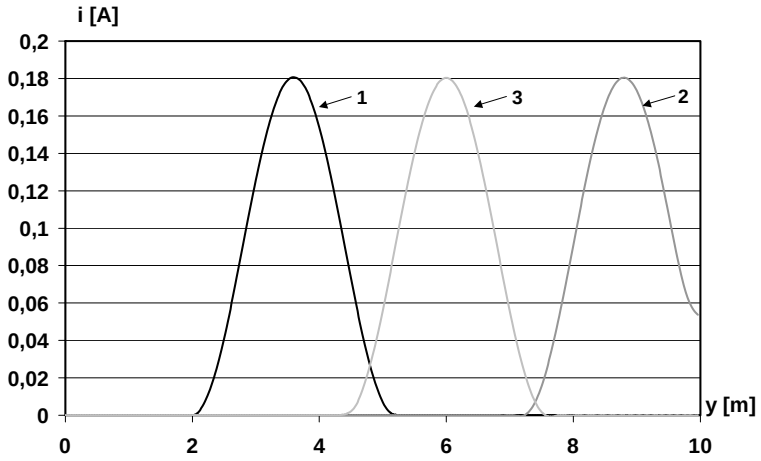
Rys. 8.29 Rozkład napięcia w wybranych punktach linii długiej:
1 - $y = 2{,}5\,m$, 2 - $y = 5{,}0\,m$, 3 - $y = 7{,}5\,m$.



Rys. 8.30 Rozkład prądu w wybranych punktach linii długiej:
1 - $y = 2{,}5\,m$, 2 - $y = 5{,}0\,m$, 3 - $y = 7{,}5\,m$.



Rys. 8.31 Rozkład napięcia w całej linii długiej w trzech wybranych chwilach czasowych: 1 - $t = 3\,ns$, 2 - $t = 6\,ns$, 3 - $t = 9\,ns$.



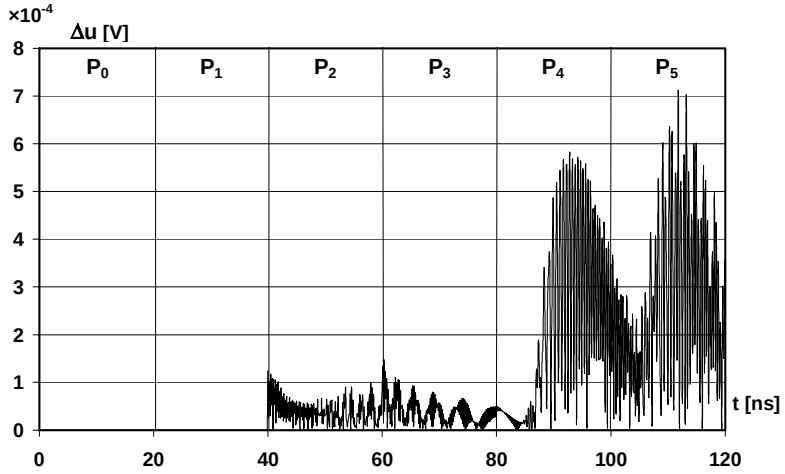
Rys. 8.32 Rozkład prądu w całej linii długiej w trzech wybranych chwilach czasowych: 1 - $t = 3ns$, 2 - $t = 6ns$, 3 - $t = 9ns$.

W celu oceny dokładności rozwiązania zmodyfikowanej metody spekulacyjnej układ (8.53) rozwiązano przy zastosowaniu sekwencyjnego algorytmu metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania $h = 10^{-13}\,s$, a następnie wyznaczono wartości błędów bezwzględnych:

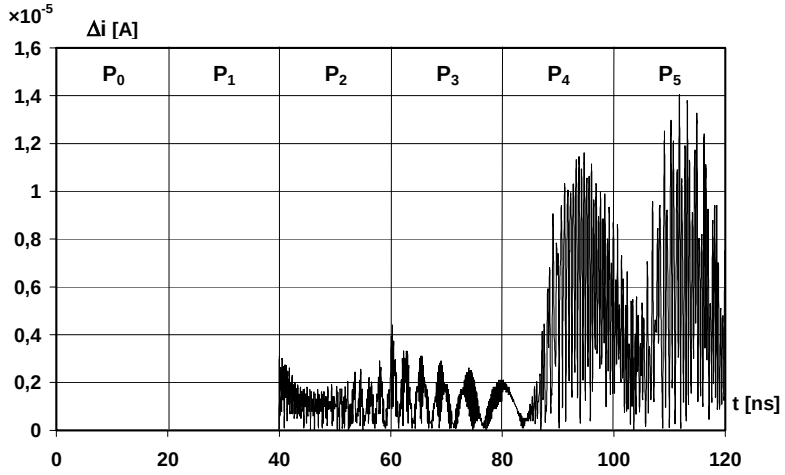
$$\Delta x_i = \left| x_{i,RK} - x_{i,MSP} \right|, \quad i = 1,2,\ldots,2 \cdot K \tag{8.55}$$

gdzie: $x_{i,RK}$ jest rozwiązaniem sekwencyjnym, zaś $x_{i,MSP}$ jest rozwiązaniem metody spekulacyjnej.

Otrzymane wartości błędów dla dwóch zmiennych stanu opisujących przebieg zmian napięcia i prądu w linii, w odległości $y = 5{,}0\,m$ od jej początku przedstawiają rys. 8.33 i rys. 8.34. Na rysunkach tych zaznaczono również granice podziału całkowitego czasu analizy stanu nieustalonego na podprzedziały.

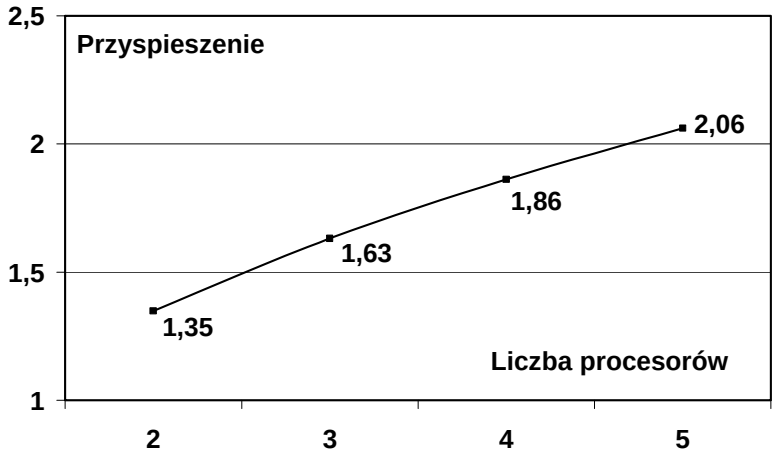


Rys. 8.33 Błędy bezwzględne zmiennej stanu opisującej napięcie w linii długiej w odległości $y = 5{,}0\,m$ od jej początku.



Rys. 8.34 Błędy bezwzględne zmiennej stanu opisującej prąd w linii długiej w odległości $y = 5{,}0\,m$ od jej początku.

Rys. 8.35 przedstawia przyspieszenie obliczeń w zmodyfikowanej metodzie spekulacyjnej. W celu otrzymania tego przyspieszenia całkowity czas analizy stanu nieustalonego podzielono na 3, 4, 5 i 6 podprzedziałów. Podczas wyznaczania wartości przyspieszenia czas rozwiązania zmodyfikowanej metody spekulacyjnej odniesiono do czasu rozwiązania sekwencyjnego metody Rungego-Kutty czwartego rzędu ze stałą długością kroku całkowania wynoszącą $h = 10^{-13}\,s$. Otrzymanie stosunkowo małych wartości przyspieszenia (2,06 dla 5 procesorów) jest spowodowane tym, iż w zmodyfikowanej metodzie spekulacyjnej pierwsza część algorytmu metody wykonywana jest sekwencyjnie (wyznaczenie warunków początkowych, obliczenia w przedziale (t_0, t_1)).



Rys. 8.35 Przyspieszenie obliczeń w zmodyfikowanej metodzie spekulacyjnej.

8.4.5. Wnioski

Zastosowanie zmodyfikowanej metody spekulacyjnej w analizie stanów nieustalonych w linii długiej umożliwiło skrócenie czasu analizy stanu nieustalonego co jest **potwierdzeniem tezy rozprawy**. Ma to szczególne znaczenie w przypadku, gdy wymagana jest duża dokładność obliczeń pociągająca za sobą konieczność rozwiązania dużego układu równań różniczkowych zwyczajnych. Tabela 8.6 zawiera zestawienie czasów obliczeń w przypadku podziału linii na określoną liczbę segmentów. We wszystkich przypadkach obliczenia wykonano przy tym samym kroku całkowania $h = 10^{-13}\,s$.

Tabela 8.6 Czas analizy stanu nieustalonego w linii długiej w przypadku podziału linii na określoną liczbę segmentów

Liczba segmentów	**Liczba równań**	**Czas obliczeń [s] przy określonej liczbie procesorów**				
		1 proc.	**2 proc.**	**3 proc.**	**4 proc.**	**5 proc.**
1 000	2 000	315	234	193	169	153
2 000	4 000	620	461	379	330	298
5 000	10 000	1 548	1 150	952	797	689
10 000	20 000	4 225	3 125	2 455	2 047	1 695
20 000	40 000	10 556	8 202	6 511	5 688	4 857

9. Uwagi końcowe i wnioski

Niniejszą rozprawę poświęcono opracowaniu równoległej metody numerycznej przeznaczonej do analizy stanów nieustalonych układów elektrycznych, w przypadku, gdy stan nieustalony opisany jest układem równań różniczkowych zwyczajnych, liniowych bądź nieliniowych lub układem równań różniczkowych cząstkowych zastąpionych układem równań różniczkowych zwyczajnych poprzez aproksymację pochodnych cząstkowych za pomocą różnic skończonych. W wyniku tych prac zaproponowano algorytm metody spekulacyjnej ze stałym krokiem całkowania oraz algorytm metody spekulacyjnej ze zmiennym krokiem całkowania. Realizacja obu algorytmów wymagała opracowania metod podziału przedziału całkowania na podprzedziały oraz metod wyboru warunków początkowych na początku podprzedziałów. Jako rozwinięcie koncepcji zaproponowanej metody spekulacyjnej przedstawiono algorytm zmodyfikowanej metody spekulacyjnej, wymagającej mniejszej liczby węzłów obliczeniowych. Dla zmodyfikowanej metody spekulacyjnej przedstawiono metodę wyboru warunków początkowych opartą na analizie rozwiązań przybliżonych oraz na przybliżeniu wykresu zbieżności funkcją wykładniczą. Działanie powyższych algorytmów sprawdzono na trzech praktycznych przykładach, wybranych ze względu na interesującą dynamikę zmiennych stanu (analiza dynamiki silnika asynchronicznego pierścieniowego), konieczność zastosowania w obliczeniach bardzo małego kroku całkowania (linearyzacja globalna nieliniowych równań opisujących model silnika prądu stałego) i dużą liczbę równań (stany nieustalone w linii długiej). W związku z tak przyjętym zakresem pracy, rozprawę podzielono na dziewięć rozdziałów.

Pierwszych pięć rozdziałów są to rozdziały wstępne. Przedstawiono w nich metody analizy stanów nieustalonych, sekwencyjne i równoległe metody numeryczne analizy równań stanu oraz ogólną charakterystykę systemów klaster. W rozdziale pierwszym przedstawiono uzasadnienie tematu rozprawy.

W rozdziale drugim podano ogólną charakterystykę modelowania i analizy stanów nieustalonych układów elektrycznych za pomocą równań różniczkowych

zwyczajnych. Przedstawiono podstawowe definicje związane z tą problematyką, a następnie opisano trzy podstawowe metody analizy stanów nieustalonych: metodę klasyczną, metodę operatorową i metodę zmiennych stanu. W podsumowaniu tego rozdziału zwrócono uwagę na możliwość zastosowania każdej z tych metod do analizy układów liniowych lub nieliniowych, niskiego lub wysokiego rzędu. Na tej podstawie jako najbardziej uniwersalną metodę przyjęto metodę zmiennych stanu, gdyż umożliwia ona analizę układów liniowych jak i nieliniowych dowolnego rzędu.

Trzeci rozdział pracy poświęcono metodom numerycznym rozwiązywania równań stanu, czyli metodom całkowania zagadnień początkowych równań różniczkowych zwyczajnych. Opisano podstawowe pojęcia związane z tą problematyką. Następnie przedstawiono metody jednokrokowe, wielokrokowe i metody predyktor-korektor, zwracając szczególną uwagę na metody jednokrokowe i algorytmy wyboru długości kroku całkowania w tych metodach, gdyż są one stosowane w metodzie spekulacyjnej.

W rozdziale czwartym przedstawiono problematykę równoległych algorytmów całkowania zagadnień początkowych. Opisano sytuacje, kiedy zastosowanie algorytmów równoległych jest opłacalne i umożliwia otrzymanie wydajnego algorytmu. Następnie podano istniejące sposoby zrównoleglania metod analizy zagadnień początkowych i klasyfikację tych metod. Wyróżniono wśród nich równoległość układu, równoległość metody i równoległość czasu (kroków). W podsumowaniu tego rozdziału zwrócono uwagę na wybór metody ze względu na docelową architekturę równoległą oraz rozmiar analizowanego problemu. Ponieważ obecnie większość obliczeń równoległych wykonywana jest przy zastosowaniu systemów klaster, które zazwyczaj nie mają szybkich sieci komputerowych, to oceniono iż najbardziej odpowiednie w takim przypadku jest zastosowanie równoległości czasu.

W rozdziale piątym przedstawiono opis równoległego systemu komputerowego (klastra), na którym wykonano wszystkie obliczenia. Podano ogólną charakterystykę systemów klaster i ich podział. Następnie opisano konstrukcję klastra KETIM2

znajdującego się w Katedrze Elektrotechniki Teoretycznej i Metrologii oraz sposób kompilacji i uruchamiania programów na tym klastrze.

Z przedstawionych powyżej rozważań dotyczących metod analizy stanów nieustalonych i metod numerycznych rozwiązywania równań stanu wyniknęły cele, tezy i oryginalne elementy rozprawy przedstawione w rozdziale szóstym.

W rozdziale siódmym przedstawiono metodę spekulacyjną analizy stanów nieustalonych. Opisano algorytm metody spekulacyjnej ze stałym krokiem całkowania wykorzystujący równomierny podział przedziału czasu analizy stanu nieustalonego na podprzedziały i wybór warunków początkowych oparty na analizie rozwiązań przybliżonych. Następnie przedstawiono wykorzystanie metody spekulacyjnej ze stałym krokiem całkowania do analizy dynamiki silnika asynchronicznego pierścieniowego. W wyniku przeprowadzonych obliczeń otrzymano skrócenie czasu analizy stanu nieustalonego, co **potwierdziło tezę rozprawy**. Następnie przedstawiono metodę spekulacyjną ze zmiennym krokiem całkowania. Zastosowanie zmiennego kroku całkowania wymagało opracowania metody podziału całkowitego przedziału czasu analizy stanu nieustalonego na podprzedziały oraz nowej metody wyboru warunków początkowych. Metody te zostały oparte na analizie rozwiązań przybliżonych. Jako przykład zastosowania metody spekulacyjnej ze zmiennym krokiem całkowania przedstawiono ten sam przykład, co poprzednio. W wyniku przeprowadzonych obliczeń otrzymano skrócenie ich czasu, co ponownie **potwierdziło słuszność tezy rozprawy**.

Rozdział ósmy poświęcono zmodyfikowanej metodzie spekulacyjnej. W algorytmie metody zastosowano podział przedziału czasu analizy stanu nieustalonego na równomierne podprzedziały oraz przedstawiono dwie metody wyznaczania warunków początkowych oparte na analizie rozwiązań przybliżonych i na przybliżeniu wykresu zbieżności funkcją wykładniczą. W zmodyfikowanej metodzie spekulacyjnej ograniczono liczbę stosowanych procesorów poprzez wybór w każdym podprzedziale tylko jednego warunku początkowego dla każdej zmiennej stanu. Jako przykład zastosowania zmodyfikowanej metody spekulacyjnej przedstawiono analizę

dynamiki nieliniowego modelu silnika prądu stałego zasilanego z generatora fotowoltaicznego oraz analizę stanów nieustalonych w linii długiej. W obu przykładach otrzymano skrócenie czasu analizy stanu nieustalonego, co **potwierdziło tezę rozprawy**.

W pracy przedstawiono metodę spekulacyjną oraz zmodyfikowaną metodę spekulacyjną analizy stanów nieustalonych. Do wykonania obliczeń w metodzie spekulacyjnej konieczne jest zastosowanie systemu komputerowego o dużej liczbie węzłów. Przyspieszenia obliczeń uzyskiwane w tej metodzie są bardzo dobre. Zmodyfikowana metoda spekulacyjna wymaga znacznie mniejszej liczby węzłów, ale mniejsze są także otrzymywane przyspieszenia. Jest to spowodowane tym, że wykonanie sekwencyjnej części algorytmu zmodyfikowanej metody spekulacyjnej zajmuje proporcjonalnie więcej czasu niż w metodzie spekulacyjnej.

Zdaniem autora, do najważniejszych osiągnięć własnych można zaliczyć:

- Opracowanie algorytmu metody spekulacyjnej opartej na dekompozycji czasu analizy stanu nieustalonego. Zastosowanie powyższego algorytmu oraz komputerów równoległych lub klastrów umożliwia skrócenie czasu analizy, a tym samym obniżenie kosztów obliczeń. W opracowanym algorytmie, procesory wykonujące równolegle obliczenia przesyłają informacje między sobą tylko przed rozpoczęciem obliczeń (dane początkowe) oraz po ich zakończeniu (wyniki obliczeń). Brak komunikacji między procesorami podczas obliczeń jest niewątpliwą zaletą opracowanego algorytmu, gdyż jego zastosowanie będzie wydajne także w systemach równoległych nie posiadających szybkiej komunikacji między procesorami.
- Opracowanie algorytmu zmodyfikowanej metody spekulacyjnej, wymagającej znacznie mniejszej liczby węzłów obliczeniowych niż metoda spekulacyjna. Ograniczenie liczby węzłów obliczeniowych wykorzystywanych w obliczeniach osiągnięto dzięki zmniejszeniu liczby warunków początkowych w każdym podprzedziale podziału czasu analizy stanu nieustalonego.

- Opracowanie metody podziału czasu analizy stanu nieustalonego na podprzedziały zapewniającej porównywalny czas obliczeń w każdym podprzedziale, w przypadku, gdy do obliczeń stosowana jest sekwencyjna metoda numeryczna ze zmiennym krokiem całkowania. Podział ten został oparty na wynikach obliczeń z dużą tolerancją błędu oraz dodatkowo zmodyfikowany w ten sposób, aby porównywalny czas obliczeń występował także przy zwiększonej dokładności obliczeń.
- Opracowanie metody wyznaczania wartości zmiennych stanu na początku każdego podprzedziału na podstawie analizy rozwiązań przybliżonych. W przypadku metody spekulacyjnej wykorzystującej sekwencyjną metodę numeryczną ze stałym krokiem całkowania, wyznaczenie tych wartości odbywa się na podstawie wyników obliczeń wykonanych z dwoma krokami całkowania znajdującymi się na granicy lub w pobliżu granicy stabilności zastosowanej sekwencyjnej metody numerycznej. Natomiast w przypadku metody spekulacyjnej wykorzystującej sekwencyjną metodę numeryczną ze zmiennym krokiem całkowania - na podstawie wyników obliczeń wykorzystywanych do kontroli długości tego kroku.
- Opracowanie metody wyznaczania wartości zmiennych stanu na początku każdego podprzedziału na podstawie przybliżenia wykresu zbieżności rozwiązania za pomocą funkcji wykładniczej. Wykonanie kilku obliczeń z dużymi krokami całkowania pozwala na wyznaczenie parametrów funkcji wykładniczej i na tej podstawie określenie poszukiwanych wartości dla małego kroku całkowania.
- Opracowanie programów komputerowych implementujących algorytm metody spekulacyjnej i algorytm zmodyfikowanej metody spekulacyjnej. Programy zostały napisane w języku C++, zaś komunikacja między procesorami została zrealizowana przy wykorzystaniu standardu komunikacji MPI. Takie podejście do oprogramowania umożliwia przenoszenie kodu programu pomiędzy różnymi systemami równoległymi bez jego zmian.

W rozprawie został rozwiązany problem badawczy polegający na opracowaniu równoległej metody numerycznej rozwiązywania równań stanu. Problematyka przedstawiona w pracy została opracowana pod kątem praktycznego zastosowania algorytmu metody w systemach klaster, będących obecnie najczęściej stosowanymi systemami obliczeń równoległych. Uzyskane w pracy wyniki przyczynią się do rozwoju równoległych metod numerycznych rozwiązywania równań stanu.

Z przedstawionego przeglądu wykonanych prac wynika, że zastosowanie równoległej metody spekulacyjnej do analizy stanów nieustalonych układów elektrycznych umożliwia przyspieszenie otrzymania rozwiązania.

Zatem teza pracy:

Algorytm spekulacyjny umożliwia skrócenie czasu analizy stanów nieustalonych w układach elektrycznych

zdaniem autora została udowodniona.

Literatura

[1] Aniserowicz K.: Projektowanie układów elektronicznych wspomagane komputerem. Wydawnictwa Politechniki Białostockiej, Białystok, 1994.

[2] Baker M., Buyya R.: Cluster Computing at a Glance. High Performance Cluster Computing Architectures and Systems (Vol. 1), Prentice Hall, New York, 1999.

[3] Baker M., Buyya R.: Cluster Computing: The Commodity Supercomputing. International Journal of Software Practice and Experience, 29 (6), Wiley And Sons, 1999, str. 551-576.

[4] Baker M. (ed.): Cluster Computing. White Paper. University of Portsmouth, U.K., 2000.

[5] Barlaud M., de Fornel B., Gauvrit M., Requier J.P.: Commande optimale d'un système générateur photovoltaique convertisseur statique-récepteur. Revue Phys. Appl. 19, Commision des Publications Françaises de Physique, Paris, 1984, str. 905-915.

[6] Baron B., Krych J.: Zastosowanie metody różnic skończonych i metody Fehlberga do badania stanów nieustalonych w linii długiej. ZKwE'2001, VI Konferencja Naukowo-Techniczna Zastosowania Komputerów w Elektrotechnice, Poznań/Kiekrz, 23-25 kwietnia 2001, str. 29-32.

[7] Baron B., Marcol A., Pawlikowski S.: Metody numeryczne w Delphi 4. Helion, Gliwice, 1999.

[8] Baron B., Pasierbek A., Maciążek M.: Algorytmy numeryczne w Delphi. Księga eksperta. Helion, Gliwice, 2006.

[9] Baron B., Piątek Ł.: Metody numeryczne w C++ Builder. Helion, Gliwice, 2004.

[10] Ben-Ari M.: Podstawy programowania współbieżnego i rozproszonego. WNT, Warszawa, 1996.

[11] Bolkowski S.: Teoria obwodów elektrycznych. WNT, Warszawa, 2005.

[12] Bolkowski S.: Stany nieustalone w obwodach elektrycznych. WNT, Warszawa, 1976.

[13] Borowik B.E.: Programowanie równoległe w zastosowaniach. Mikom, Warszawa, 2001.

[14] Burns G., Daoud R., Vaigl J.: LAM: An Open Cluster Environment for MPI. Proceedings of Supercomputing Symposium ’94, University of Toronto, 1994, str. 379-386.

[15] Burrage K.: Parallel Methods for Initial Value Problems. Applied Numerical Mathematics 11, 1993, str. 5-25.

[16] Burrage K.: Parallel and Sequential Methods for Ordinary Differential Equations. Clarendon Press, Oxford, 1995.

[17] Burrage K.: Parallel Methods for Systems of Ordinary Differential Equations. SIAM, 1995.

[18] Butcher J.C.: Numerical Methods for Ordinary Differential Equations in the 20th Century. Journal of Computational and Applied Mathematics, Vol. 125, 2000, str. 1-29.

[19] Cholewicki T.: Elektrotechnika teoretyczna. Tom II. WNT, Warszawa, 1971.

[20] Chua L.O., Lin P.-M.: Komputerowa analiza układów elektronicznych. Algorytmy i metody obliczeniowe, WNT, Warszawa, 1981.

[21] Collatz L.: Metody numeryczne rozwiązywania równań różniczkowych. PWN, Warszawa, 1960.

[22] Conte S.D., de Boor C.: Elementary Numerical Analysis. An Algorithmic Approach. McGraw-Hill, 1980.

[23] Cunningham W.J.: Analiza układów nieliniowych. WNT, Warszawa, 1962.

[24] Czban W.: Nowoczesne metody numeryczne. Wydawnictwo Uniwersytetu Rzeszowskiego, Rzeszów, 2003.

[25] Dahlquist G., Björck Å.: Metody numeryczne, PWN, Warszawa, 1983.

[26] Fehlberg E.: Klassische Runge-Kutta-Formeln fünfter und siebenter Ordung mit Schrittweiten-Kontrolle. Computing 4, 1969, str. 93-106.

[27] Fehlberg E.: Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme. Computing 6, 1970, str. 61-71.

[28] Forenc J., Jordan A., Tudruj M.: Speculative Parallel Processing Applied to Modelling of Initial Problems in Electrical Circuits. PARELEC'2000, International Conference on Parallel Computing in Electrical Engineering, IEEE Computer Society, Los Alamitos, 2000, str. 192-196.

[29] Forenc J., Jordan A., Tudruj M.: A Survey of Speculative Methods for Transient State Analysis. PARELEC'2002, International Conference on Parallel Computing in Electrical Engineering, IEEE Computer Society, Los Alamitos, 2002, str. 353-358.

[30] Forenc J., Jordan A., Tudruj M.: Parallel Modeling of Transient States Analysis in Electrical Circuits. EuroPVM/MPI 2005, Special Session: Current Trends in Numerical Simulation for Parallel Engineering Environments New Directions and Work-in-Progress, September 18-21, 2005, Sorrento Peninsula, Italy, Lectures Notes in Computer Sciences, Vol. 3666, Springer-Verlag, 2005, str. 512-519.

[31] Forenc J., Jordan A.: Speculative Analysis of Transient States. SCI'2001/ISAS, World Multiconference on Systemics, Cybernetics and Informatics, Vol. XV, Industrial Systems, Orlando, 2001, str. 327-331.

[32] Forenc J., Jordan A.: The Modified Speculative Method for the Transient States Analysis. PARELEC'2004, International Conference on Parallel Computing in Electrical Engineering, IEEE Computer Society, Los Alamitos, 2004, str. 189-193.

[33] Forenc J., Walendziuk W., Jordan A.: Application of the Runge-Kutta-Fehlberg Method in the Speculative Method of Transient States Analysis. CIPC2003,

Concurrent Information Processing and Computing, NATO Advanced Research Workshop, Pre-proceedings, "Al.I.Cuza" University Press, Iasi, 2003, str. 97-106.

[34] Forenc J., Walendziuk W.: Application of the Speculative Method to Analysis of the Dynamics of an Asynchronous Slip-Ring Motor. XIV IEEE-SPIE Symposium on Photonics and Web Engineering, SPIE 5775, 2005, str. 414-421.

[35] Forenc J., Walendziuk W.: Zastosowanie metody spekulacyjnej do analizy dynamiki silnika asynchronicznego. Elektronika, 2006 (zaakceptowany do publikacji).

[36] Forenc J.: Analiza spekulacyjna stanów nieustalonych. OWD'2001, III Ogólnopolskie Warsztaty Doktoranckie, Istebna-Zaolzie, 2001, str. 276-281.

[37] Forenc J.: The Speculative Method of Transient State Analysis with a Variable Integration Step. PARELEC'2002, International Conference on Parallel Computing in Electrical Engineering, IEEE Computer Society, Los Alamitos, 2002, str. 364-368.

[38] Forenc J.: Metoda spekulacyjna analizy stanów nieustalonych ze zmiennym krokiem całkowania. OWD'2002, IV Ogólnopolskie Warsztaty Doktoranckie, Archiwum Konferencji PTETiS, vol. 14, Warszawa, 2002, str. 361-366.

[39] Forenc J.: Zastosowanie metody spekulacyjnej do analizy stanów nieustalonych w linii długiej. IC-SPETO 2003, Międzynarodowa Konferencja z Podstaw Elektrotechniki i Teorii Obwodów, vol. 2, Gliwice, 2003, str. 431-434.

[40] Forenc J.: A New Approach to the Speculative Method for the Transient States Analysis. XIV IEEE-SPIE Symposium on Photonics and Web Engineering, SPIE 5775, 2005, str. 406-413.

[41] Forenc J.: Nowe podejście do metody spekulacyjnej analizy stanów nieustalonych. Elektronika, 2006 (zaakceptowany do publikacji).

[42] Forenc J.: Nowa metoda wyznaczania warunków początkowych w metodzie spekulacyjnej analizy stanów nieustalonych. ZKwE'2005, X Konferencja Zastosowania Komputerów w Elektrotechnice, Poznań, 2005, str. 123-124.

[43] Fortuna Z., Macukow B., Wąsowski J.: Metody numeryczne, WNT, Warszawa, 2005.

[44] Gear C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Englewood Cliffs, Prentice-Hall, Inc., New York, 1971.

[45] Gear C.W.: Massive Parallelism Across Space in ODEs. Applied Numerical Mathematics 11, 1993, str. 27-43.

[46] Gear C.W., Xuhai Xu: Parallelism Across Time in ODEs. Applied Numerical Methematics 11, 1993, str. 45-68.

[47] Hoffman J.D.: Numerical Methods for Engineers and Scientists. Second Edition. Revised and Expanded. Marcel Dekker, Inc., 2001.

[48] Jackson K.R., Norsett S.P.: The Potential for Parallelism in Runge-Kutta Method. Part 1. RK Formulas in Standard Form. SIAM Journal on Numerical Analysis, Volume 32, Issue 1, 1995, str. 49-82.

[49] Jackson K.R.: A Survey of Parallel Numerical Methods for Initial Value Problems for Ordinary Differential Equations. IEEE Transactions on Magnetics, Vol. 27, No. 5, September 1991, str. 3792-3797.

[50] Jakóbiec T.: Modelowanie procesów przejściowych w silnikach indukcyjnych. http://www.silniki.cad.net.pl

[51] Jankowscy J. i M.: Przegląd metod i algorytmów numerycznych. Część 1. WNT, Warszawa, 1981.

[52] Jordan A., Benmouna M., Bensenane A., Borucki A.: Optimal Linearization of Non-Linear State Equations. Rairo - Automatique, Systems Analysis and Control, 1987, str. 263-271.

[53] Jordan A., Benmouna M.: Optimal Linearization Method. Applications to some Nonlinear Physical Systems. Wydawnictwa Politechniki Białostockiej, Białystok, 1989.

[54] Jordan A., Bycul R., Forenc J., Myszkowski P.: Systemy wieloprocesorowe w automatyce i pomiarach. Pomiary Automatyka Kontrola, 10/2004, str. 16-20.

[55] Jordan A., Forenc J., Butryło B., Tudruj. M.: Concurrent Numerical Methods - Współbieżne metody numeryczne. IC - SPETO'2000, XXIII Międzynarodowa Konferencja z Podstaw Elektrotechniki i Teorii Obwodów, Gliwice, 2000, str. 27-37.

[56] Jordan A., Forenc J., Tudruj M.: Speculative Parallel Processing Applied to Modelling of Initial Problems. COMPEL, The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 24, No. 1, 2005, str. 127-144.

[57] Jordan A., Nowacki J.P.: Global Linearization of Non-Linear State Equations. International Journal of Applied Electromagnetics and Mechanics, Volume 19, Number 4/2004, str. 637-642.

[58] Kaczorek T., Jordan A., Forenc J.: Global Linearization of a Non-Linear Model of a DC Drive System. The Second Grant Conference: Numerical Methods in Modelling of Electric Devices, Warszawa, 2002, str. 7-16.

[59] Kaczorek T., Dzieliński A., Dąbrowski W., Łopatka R.: Podstawy Teorii Sterowania. WNT, Warszawa, 2005.

[60] Kaczorek T.: Algebra Liego. Seminarium Katedry Elektrotechniki Teoretycznej i Metrologii, Politechnika Białostocka, Białystok, 2003 (nie publikowane)

[61] Kaczorek T.: Teoria sterowania i systemów. Wydawnictwo Naukowe PWN, Warszawa, 1999.

[62] Kaczorek T.: Wektory i macierze w automatyce i elektrotechnice. WNT, Warszawa, 1998.

[63] Kaczorek T.: Linear Control Systems, vol. 1 and 2, J. Wiley, 1993.

[64] Karbowski A., Niewiadomska-Szynkiewicz E.: Obliczenia równoległe i rozproszone. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 2001.

[65] Kącki E., Małolepszy A., Romanowicz A.: Metody numeryczne dla inżynierów. Wydawnictwo Politechniki Łódzkiej, Łódź, 2000.

[66] Kącki E.: Równania różniczkowe cząstkowe w zagadnieniach fizyki i techniki. WNT, Warszawa, 1995.

[67] Klamka J. i in.: Metody numeryczne. Politechnika Śląska, Gliwice, 1998.

[68] Kozielski S., Szczerbiński Z.: Komputery równoległe. Architektura, elementy programowania. WNT, Warszawa, 1993.

[69] Krakowski M.: Podstawy elektrotechniki. Politechnika Łódzka, Łódź, 1970.

[70] Krupka J. Morawski R.Z., Opalski L.J.: Metody numeryczne. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 1997.

[71] Krupowicz A.: Metody numeryczne zagadnień początkowych równań różniczkowych zwyczajnych. PWN, Warszawa, 1986.

[72] Lal K., Rak T.: Linux a technologie klastrowe. Mikom, Warszawa, 2005.

[73] LAM/MPI User's Guide Version 7.1.1, The LAM/MPI Team, Open Systems Lab, Indiana University, 2004.

[74] Machowski J., Bernas S.: Stany nieustalone i stabilność systemu elektroenergetycznego. WNT, Warszawa, 1989.

[75] Majchrzak E., Mochnacki B.: Metody numeryczne. Podstawy teoretyczne, aspekty praktyczne i algorytmy. Wydawnictwo Politechniki Śląskiej, Gliwice, 2004.

[76] Matthew N, Stones R.: Zaawansowanie programowanie w systemie Linux. Helion, Gliwice, 2002.

[77] MPI Forum - http://www.mpi-forum.org

[78] Message Passing Interface Forum, MPI: A Message-Passing Interface Standard, Report No. CS-94-230, University of Tennessee, Knoxville, May 5, 1994.

[79] Nivergelt J.: Parallel Methods for Integrating Ordinary Differential Equations. Communications of the ACM, Volume 7, Issue 12, 1964, str. 731-733.

[80] Nowak L.: Modele polowe przetworników elektromechanicznych w stanach nieustalonych. Wydawnictwo Politechniki Poznańskiej, Poznań, 1999.

[81] Osiowski J., Szabatin J.: Podstawy teorii obwodów. Tom II. WNT, Warszawa, 1998.

[82] Osiowski J., Szabatin J.: Podstawy teorii obwodów. Tom III. WNT, Warszawa, 1995.

[83] Palczewski A.: Równania różniczkowe zwyczajne. Teoria i metody numeryczne z wykorzystaniem komputerowego systemu obliczeń symbolicznych. WNT, Warszawa, 1999.

[84] Pao Y.C.: Engineering Analysis. Interactive Methods and Programs with Fortran, QuickBASIC, MATLAB and Mathematica. CRC Press, 2001.

[85] Petcu D.: Parallelism in Solving Ordinary Differential Equations. Mathematical Monographs 64, Timisoara University Press, 1998.

[86] Press W.H., Flannery B.P., Teukolsky S.A., Vetterling W.T.: Numerical Recipes in C: The Art Of Scientific Computing. Cambridge University Press, 1993.

[87] PVM: Parallel Virtual Machine - http://www.csm.ornl.gov/pvm

[88] Quarteroni A., Sacco R., Saleri F.: Numerical Mathematics. Springer-Verlag, New York, 2000.

[89] Ralston A. Wstęp do analizy numerycznej. PWN, Warszawa, 1983.

[90] Rosłoniec S.: Metody matematyczne w projektowaniu układów elektronicznych o parametrach rozłożonych. WNT, Warszawa, 1988.

[91] Shampine L.F., Allen R.C., Pruess S.: Fundamentals of Numerical Computing. John Wiley & Sons, Inc., 1997.

[92] Shampine L.F., Gordon M.K.: Computer Solution of Ordinary Differential Equations. Freeman, 1975.

[93] Simonsen H.H.: Extrapolation Methods for ODE's: Continuous Approximations, a Parallel Approach. Ph. D. Thesis, Mathematics Sciences Division, Norwegian Institute of Technology, Trondheim, Norway, 1980.

[94] Squyres J.M., Lumsdaine A.: A Component Architecture for LAM/MPI. 10th European PVM/MPI Users' Group Meeting, LNCS 2840, 2003, str. 379-387.

[95] Stoer J., Bulirsch R.: Wstęp do metod numerycznych. Tom II. PWN, Warszawa, 1980.

[96] Stoer J., Bulirsch R.: Wstęp do analizy numerycznej. PWN, Warszawa, 1987.

[97] Straszecka E. (red.): Laboratorium metod numerycznych. Wydawnictwo Politechniki Śląskiej, Gliwice, 2002.

[98] Szatkowski A., Cichosz J.: Metody numeryczne. Podstawy teoretyczne. Wydawnictwo Politechniki Gdańskiej, Gdańsk, 2002.

[99] TOP500 Supercomputer Sites - http://www.top500.org

[100] Walczak J., Pasko M.: Elementy dynamiki liniowych obwodów elektrycznych. Wydawnictwo Politechniki Śląskiej, Gliwice, 2001.

[101] Wąsowski J.: Podstawy elektrotechniki. Stany nieustalone w obwodach elektrycznych. Politechnika Częstochowska, Częstochowa, 1976.

[102] Weiss Z., Gruźlewski T.: Programowanie współbieżne i rozproszone w przykładach i zadaniach. WNT, Warszawa, 1993.

[103] Wojtkiewicz A.: Teoria obwodów. Część II. Wydawnictwa Politechniki Warszawskiej, Warszawa, 1991.