

Maciej Bargielski

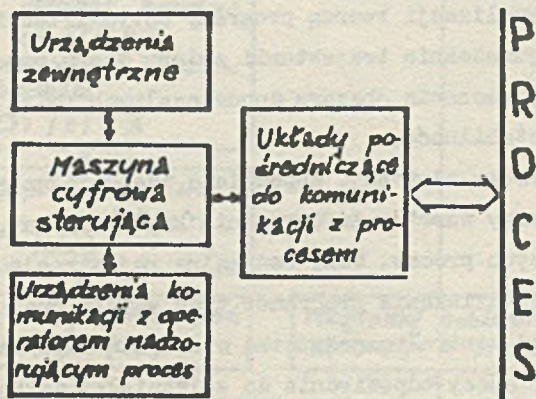
Instytut Kompleksowych Systemów Sterowania

NUMERYCZNE WYZNACZANIE OBSZARÓW STABILNOŚCI  
W SYSTEMACH AUTOMATYKI KOMPLEKSOWEJ

Streszczenie. W pracy przedstawiono różne struktury systemów automatyki kompleksowej. Dokładniej omówiono strukturę programową, wskazując na miejsce problemu wyznaczania obszarów stabilności w tej strukturze. Następnie podano twierdzenie umożliwiające wyznaczenie dla pewnej klasy równań różniczkowych pełnych obszarów stabilności. Twierdzenie zilustrowano algorytmem postępowania oraz przykładami zarówno dla równań deterministycznych, jak i z parametrami stochastycznymi.

1. Struktura systemu automatyki kompleksowej

Współcześnie ukształtowała się struktura systemu automatyki kompleksowej [1, 6] przedstawiona schematycznie na rys. 1. Można ją w sposób ogólny podzielić na strukturę urządzeniową i programową.



Rys. 1. Ogólna struktura systemu automatyki kompleksowej

W strukturze urządzeniowej rolę centralnej jednostki sterującej spełnia zwykle minikomputer zapewniający przepływ i przetwarzanie informacji pomiędzy poszczególnymi elementami omawianej struktury. Minikomputer ten komunikuje się z procesem poprzez urządzenia pośredniczące zapewniające pobieranie informacji o procesie i jej zamianę na postać akceptowalną przez maszynę cyfrową, a także umożliwiające czynne oddziaływanie na proces. Operator procesu komunikuje się z nim za pośrednictwem jednostki sterującej poprzez urządzenia komunikacji z operatorem, zaś urządzenia zewnętrzne umożliwiają gromadzenie i wydawanie informacji o procesie w postaci raportów utrwalanych na urządzeniach typu drukarka wierszowa, dalekopis, perforator czy pamięć masowa.

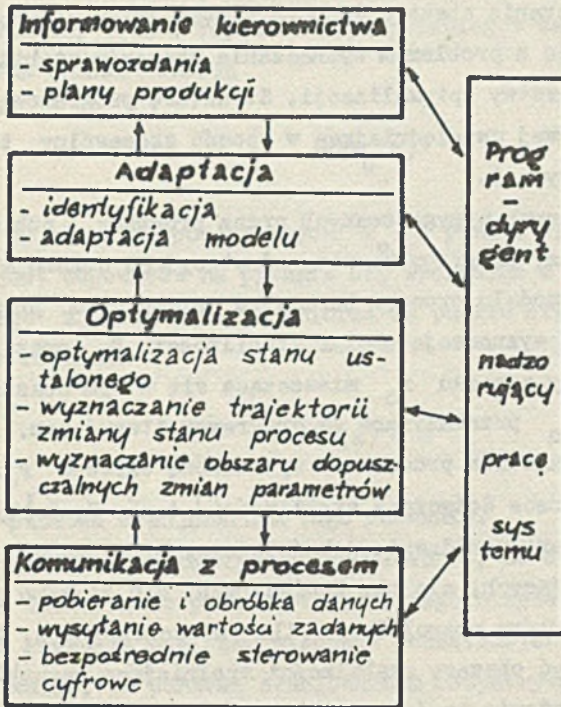
Struktury programowe, w przeciwieństwie do urządzeniowych, są sprawą umowną - chodzi tu bowiem o pewne pogrupowanie programów. Można je więc dzielić zarówno od strony jednostki centralnej, jak i od strony procesu. Struktura ta, z punktu widzenia jednostki centralnej, prowadzi do podziału na programy użytkowe, pomocnicze, zarządzające czy szeregujące. Od strony zaś procesu, co dalej jest istotne, odpowiedni podział przedstawia rys. 2.

Przedstawiona tu struktura programowa [1, 2, 6] jest strukturą wielowarstwową. Warstwa najniższa - komunikacji z procesem zapewnia pobieranie i obróbkę danych o procesie, wyznaczanie i wydawanie wartości zadanych regulatorów konwencjonalnych zainstalowanych w procesie, czy wreszcie bezpośrednie sterowanie cyfrowe.

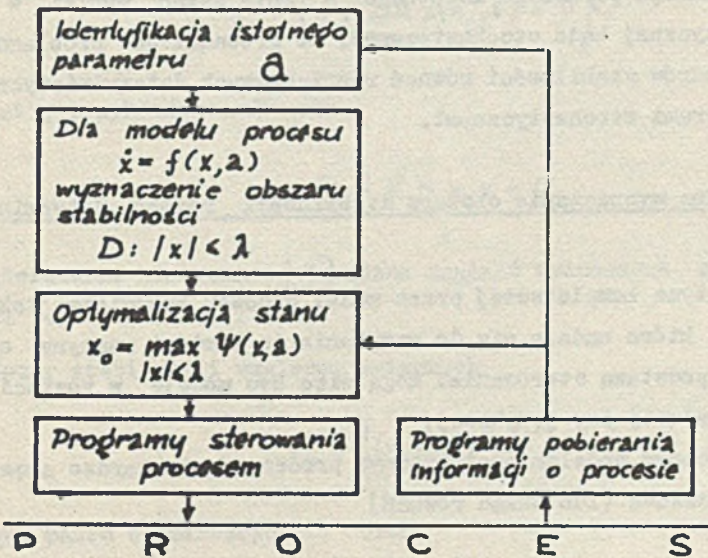
Warstwę optymalizacji tworzą programy optymalizacji stanu ustalonego, programy wyznaczania trajektorii zmiany stanu procesu, czy wreszcie programy wyznaczania obszaru dopuszczalnych zmian parametrów np. ze względu na stabilność.

Programy warstwy adaptacji zapewniają identyfikację i adaptację modelu, zaś programy warstwy informowania kierownictwa zbieranie danych charakteryzujących proces - tzn. centralną rejestrację danych.

Właściwe współdziałanie programów tych warstw oraz realizację zadań stawianych przed centralną jednostką sterującą zapewnia program nadzorczy uaktywniający odpowiednie do zaistniałej sytuacji programy.



Rys. 2. Struktura programowa systemu automatyki kompleksowej



Rys. 3. Szczegółowa struktura programowa

Programy wyznaczania obszaru dopuszczalnych zmian parametrów wiążące się bezpośrednio z problemem wyznaczania obszarów stabilności należą do programów warstwy optymalizacji. Strukturę programową systemu automatyki kompleksowej uwzględniającą w sposób szczególny tę problematykę przedstawia rys. 3.

Na podstawie danych przygotowanych przez programy pobierania informacji o procesie, programy warstwy adaptacyjnej dokonują identyfikacji istotnego w modelu procesu parametru  $a$ , po czym programy warstwy optymalizacji wyznaczają obszar stabilności  $D$  oraz wyznaczają optymalne parametry procesu  $x_0$  mieszczące się w tym obszarze ograniczeń. Parametry  $x_0$  przekazywane są programom sterującym, realizującym nastawianie parametrów procesu na wyznaczoną wartość  $x_0$ .

Dotychczasowe prace dotyczące stabilności [4, 6, 7, 8] były wszystkie prowadzone z punktu widzenia jakościowego, prowadzącego zawsze do warunków wystarczających, ale nie koniecznych. Nie starano się opracować takich metod, które pozwoliłyby dla poszczególnych, konkretnych przypadków wyznaczyć obszary stabilności spełniające warunki konieczne i wystarczające. Zadanie to jest możliwe do zrealizowania na drodze wykorzystania maszyn matematycznych i metod numerycznych.

Identyfikacja parametru istotnego  $a$  może dawać wyniki w postaci deterministycznej bądź stochastycznej, co prowadzi do problemu wyznaczania obszarów stabilności równań różniczkowych deterministycznych jak i z parametrami stochastycznymi.

## 2. Numeryczne wyznaczanie obszaru stabilności równań deterministycznych

W automatyce kompleksowej przez model procesu rozumiemy taką formę jego opisu, która nadaje się do wczytania do pamięci maszyny cyfrowej i może być podstawą sterowania. Mogą więc być modele w postaci analitycznej, grafowej czy językowej.

Matematycznym modelem analitycznym procesu jest bardzo często równanie różniczkowe (lub układ równań)

$$\dot{x} = f(x, a)$$

gdzie  $a$  jest stałą wyznaczoną w procesie identyfikacji. Rozwiązanie tego równania, będące reprezentacją przebiegów zachodzących w procesie, można opisać funkcją

$$x(x_0, t)$$

zależną od warunków początkowych  $x_0$ . Z punktu widzenia sterowania istotna jest odpowiedź na pytanie czy wychodząc z danego warunku początkowego  $x_0$  rozwiązanie zmierza do punktu stabilnego, a więc rozwiązania równania

$$f(x, a) = 0$$

zwanego punktem stacjonarnym tego równania.

Zbiór punktów posiadających tę własność, że startujące z nich trajektorie dochodzą do punktu stacjonarnego nazywamy obszarem stabilności tego równania względem warunków początkowych [8].

Przyjmując, że punktem stacjonarnym rozpatrywanego równania jest początek układu współrzędnych, analitycznie można ten obszar zapisać

$$D_0 = \left\{ x_0 : \lim_{t \rightarrow \infty} x(x_0, t) = 0 \right\}.$$

Rozważmy przykładowo równanie

$$\ddot{x} + (1 - \dot{x}^2)\dot{x} + x = 0$$

W literaturze klasycznej [7] można znaleźć wyznaczone obszary stabilności tego równania. I tak:

\* obszar stabilności względem zmiennych

$$D_2 = \left\{ (x, \dot{x}) : \dot{x}^2 \leq 1 \right\}$$

będący pasem o szerokości 1 oraz

\* obszar stabilności względem warunków początkowych, będący maksymalną kulą wpisaną w obszar  $D_2$ :

$$D_1 = \left\{ (x, \dot{x}) : x^2 + \dot{x}^2 \leq 1 \right\}.$$

(przyjmując przestrzeń metryczną euklidesową).

Wszyscy autorzy podkreślają, że jest to obszar wystarczający, jednak nie konieczny i nie było metody pozwalającej wyznaczyć obszary konieczne.

Dla wyznaczenia obszarów spełniających warunki konieczne i wystarczające jakościowa teoria równań różniczkowych jest nieprzydatna. Problem ten można jednak rozwiązać na innej drodze, wykorzystując te możliwości, jakie stwarzają maszyny matematyczne i metody numeryczne.

Prezentowana tu metoda wykorzystuje pewne właściwości trajektorii równań sprzężonych. Na przykład równaniem sprzężonym do przykładowego jest

$$\ddot{x} - (1 - \dot{x}^2)\dot{x} + x = 0$$

charakteryzujące się identycznym kształtem trajektorii, a jedynie kierunek ruchu po nich dla  $t \rightarrow \infty$  dla równań podstawowych i sprzężonych jest przeciwny [5]. Podstawą do wykorzystania tej własności równań sprzężonych do wyznaczania obszarów stabilności równań różniczkowych zwyczajnych, jest następujące

**T w i e r d z e n i e**

Niech prawa strona układu

$$\dot{x} = f(x)$$

spełnia warunki twierdzenia o istnieniu i jednoznaczności

$$\forall x \in \Omega \subset \mathbb{R}^2: f(x) \in C, f'_x(x) \in C$$

i niech jedynym skończonym jego punktem stacjonarnym będzie początek układu współrzędnych. Jeśli jest on punktem równowagi stabilnej, a rozwiązanie  $y = y(y_0, t)$  równania sprzężonego

$$\dot{y} = -f(y)$$

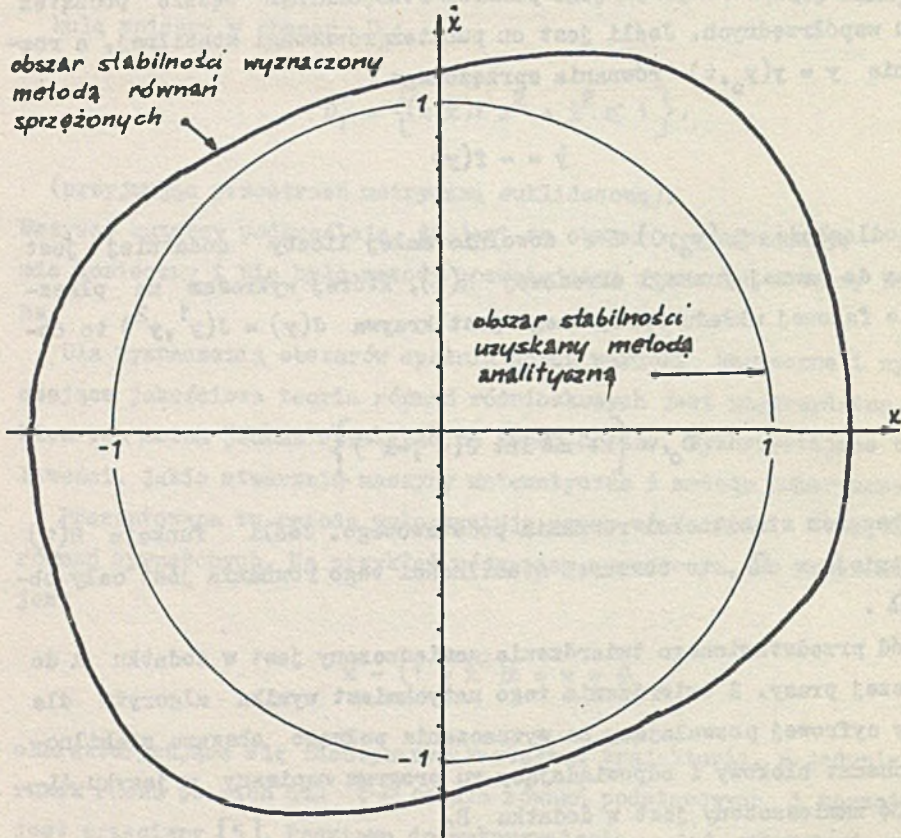
gdzie odległość  $\rho(y_0, 0) \leq \epsilon$  dowolnie małej liczby dodatniej jest zbieżne do pewnej funkcji okresowej  $H(t)$ , której wykresem na płaszczyźnie fazowej układu sprzężonego jest krzywa  $J(y) = J(y^1, y^2)$  to obszar

$$D_0 = \left\{ x: x \in \text{int } J(x^1, -x^2) \right\}$$

jest obszarem stabilności równania podstawowego. Jeśli funkcja  $H(t)$  nie istnieje w  $\Omega$ , to obszarem stabilności tego równania jest cały obszar  $\Omega$ .

Dowód przedstawionego twierdzenia zamieszczony jest w dodatku A do niniejszej pracy. Z twierdzenia tego natychmiast wynika algorytm dla maszyny cyfrowej pozwalający na wyznaczenie pełnego obszaru stabilności. Schemat blokowy i odpowiadający mu program napisany w języku ALGOL-1204 zamieszczony jest w dodatku B.

Przy pomocy tego programu wyznaczono obszar stabilności równania przykładowego przedstawiony na rys. 4. Cienką linią przedstawiono na tym rysunku obszar stabilności otrzymany metodą analityczną. Porównując oba te obszary można stwierdzić znaczne, bo wynoszące ok. 42%, powiększenie pola powierzchni obszaru uzyskanego prezentowaną metodą w stosunku do uzyskanego metodą analityczną.



Rys. 4. Porównanie obszarów stabilności uzyskanych różnymi metodami

### 3. Numeryczne wyznaczanie obszaru stabilności równań z parametrami stochastycznymi

Nie wszystkie procesy prowadzą w sensie modeli do równań różniczkowych o współczynnikach stałych. Są i takie, których modele prowadzą do równań różniczkowych o współczynnikach zmiennych, które w procesie identyfikacji mogą być identyfikowane tylko z dokładnością do ich niektórych parametrów stochastycznych.

Wprowadźmy pojęcie obszaru stabilności procesu stochastycznego  $x_t$  względem prawdopodobieństwa [1, 2] zdefiniowanego jako

$$S_e = \left\{ \check{x} : P\left\{x(\check{x}, t) \rightarrow 0 \left\{ \leq e, \quad 0 \leq e \leq 1 \right\} \right\} \right\}$$



(przyjęto tu, podobnie jak i poprzednio, że początek układu współrzędnych jest punktem stacjonarnym stabilnym). Jest to więc taki obszar, że dla warunków początkowych z jego wnętrza proces  $x_t$  osiągnie swój punkt równowagi z prawdopodobieństwem nie mniejszym niż  $e$ . Zadaniem do rozwiązania jest tu wyznaczenie dla danego, konkretnego procesu obszaru stabilności  $S_p$ , w którym proces jest stabilny w podanym sensie z zadanyam prawdopodobieństwem  $p_{\text{zad}}$ . Rozwiązanie postawionego problemu można otrzymać dwiema drogami:

1° Znalezienie rozkładu  $Q(x)$  prawdopodobieństwa przejścia procesu z punktu  $x$  do początku układu współrzędnych:  $x \rightarrow 0$  dla wszystkich interesujących wartości  $x$ , a następnie określenie obszaru

$$S_p = \{x: Q(x) \leq p\}$$

2° Bezpośrednie znalezienie obszaru  $S_p$  jedynie na podstawie równań opisujących proces.

Dla procesów, których modele dadzą się przedstawić w postaci pewnej klasy równań różniczkowych z parametrami stochastycznymi można zastosować metodę 1°. O tej klasie równań zakłada się dodatkowo (oprócz założeń twierdzenia), że spełnia hipotezę Ajzermana, tzn. ciągłą i jednoznaczną zależność rozwiązań od wartości parametru stochastycznego  $u$ :

$$\forall u_1 < u_2 \text{ (lub } u_1 > u_2), \forall t: \rho(x(x_0, t; u_1), 0) < \rho(x(x_0, t; u_2), 0)$$

Dla przedstawienia sposobu postępowania zmierzającego do wyznaczenia obszaru stabilności w wyżej podanym sensie rozpatrzmy, dla ustalenia uwagi, równanie

$$\ddot{x} + (1 - \dot{x}^2)\dot{x} + (1 + u)x = 0$$

gdzie  $u$  jest zmienną losową o rozkładzie równomiernym na przedziale  $[a, b] = [1, 3]$ , a zadane prawdopodobieństwo  $p_{\text{zad}} = 0,65$ .

Algorytm jest następujący:

- (i) wyznaczenie obszarów stabilności  $D_a$  i  $D_b$  równania, w którym parametr  $u$  przyjmuje stałe wartości  $a$  oraz  $b$ . Jest to problem wyznaczania obszarów stabilności równań różniczkowych zwyczajnych o stałych współczynnikach, który można rozwiązać przedstawioną wyżej metodą równań sprzężonych.
- (ii) wyznaczenie obszaru  $T$  będącego pierścieniem

$$T = (D_a \setminus D_b) \cup (D_b \setminus D_a)$$

wewnątrz którego rozwiązania równania wyjściowego są stabilne lub nie, w zależności od wartości parametru  $u$ . Granica więc obszaru stabilności względem prawdopodobieństwa należy do  $T$  i wewnątrz niego należy jej szukać. Oznaczając bowiem przez  $Q(x, \dot{x})$  prawdopodobieństwo dojścia trajektorii rozpatrywanego równania do początku układu współrzędnych zachodzi:

$$\forall x, \dot{x} \in D_a \cap D_b: \quad Q(x, \dot{x}) = 1$$

$$\forall x, \dot{x} \in \mathbb{R}^2 \setminus (D_a \cup D_b): \quad Q(x, \dot{x}) = 0$$

$$\forall x, \dot{x} \in T: \quad 0 < Q(x, \dot{x}) < 1$$

zaś na brzegu  $\partial D$  obszaru stabilności

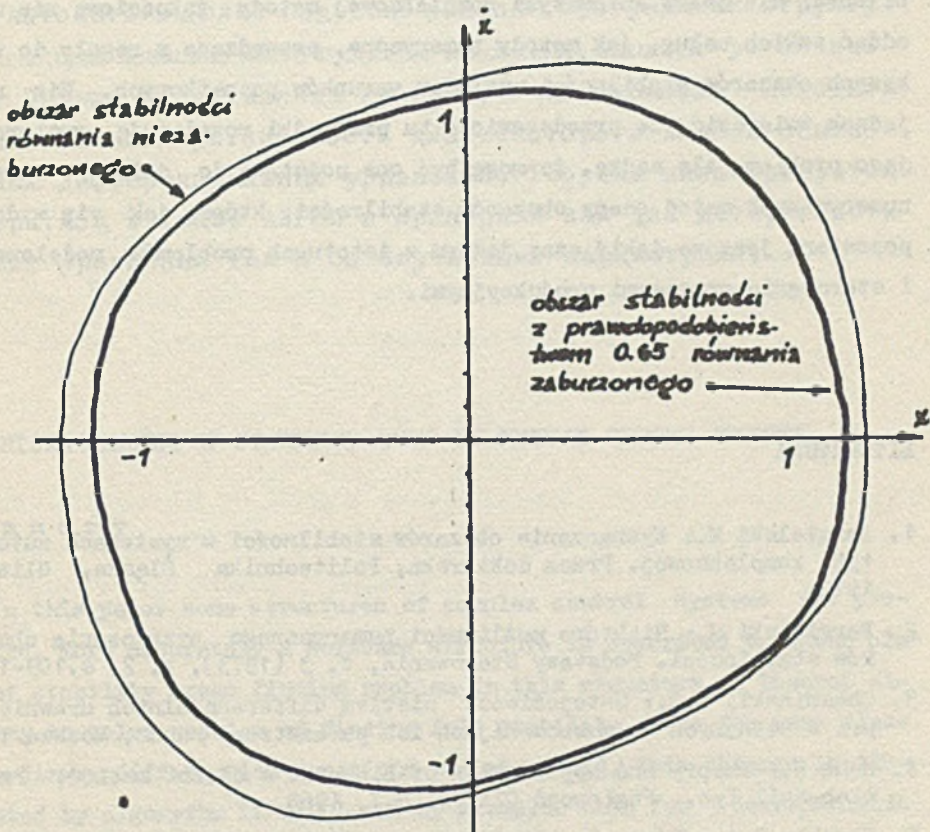
$$\forall x, \dot{x} \in \partial D \quad Q(x, \dot{x}) = p_{\text{zad}}$$

Ponieważ  $Q(x)$  jako prawdopodobieństwo dojścia do początku układu współrzędnych jest niemalejącą funkcją  $x$ , więc

$$\forall x_1, x_2, g(x_1, 0) < g(x_2, 0): \quad Q(x_1) > Q(x_2)$$

co określa algorytm poszukiwania brzegu  $\partial D$ . Szukany obszar stabilności jest oczywiście jego wnętrzem.

Posługując się maszyną cyfrową do realizacji przedstawionego algorytmu uzyskano dla omawianego, przykładowego równania obszar stabilności przedstawiony na rys. 5, na którym cienką kreską przedstawiono obszar stabilności równania niezaburzonego, tzn. równania, w którym  $u=0$ . Jak widać, wprowadzenie zaburzeń losowych zmniejsza obszar stabilności.



Rys. 5. Porównanie obszaru stabilności z rys. 4 z obszarem stabilności z prawdopodobieństwem  $p_{\text{zad}} = 0,65$  równania zaburzonego

W rozpatrywanym przykładzie parametr stochastyczny  $u$  przyjmował wartości w ograniczonym przedziale. Praktycznie tylko z takimi parametrami mamy do czynienia. Niemniej jednak, rozpatrując parametry z nieskończonego przedziału (np. o rozkładzie normalnym) problem wyznacza-

nia obszaru stabilności można rozwiązać przedstawioną metodą przez obcięcie brzegów rozkładu tzw. ogonów w taki sposób, aby odrzucone wartości były przyjmowane z dostatecznie małym prawdopodobieństwem [1].

#### 4. Wnioski i uwagi końcowe

Podsumowując, można by stwierdzić, że w wyznaczaniu obszarów stabilności dla celów automatyki kompleksowej metody jakościowe nie mogą oddać takich usług, jak metody numeryczne, prowadzące z reguły do większych obszarów stabilności względem warunków początkowych. Nie mogę jednak twierdzić, że przedstawione tu przypadki rozwiązują wystarczająco problem, ale sądzę, że mogą być one podstawą do dalszego rozwoju numerycznych metod oceny obszarów stabilności, które, jak się wydaje, pozostaną jeszcze jakiś czas jednym z istotnych problemów modelowania i sterowania procesami produkcyjnymi.

#### LITERATURA

1. Bargielski M.: Wyznaczanie obszarów stabilności w systemach automatyki kompleksowej. Praca doktorska, Politechnika Śląska, Gliwice 1972.
2. Bargielski M.: Niektóre możliwości numerycznego wyznaczania obszarów stabilności. Podstawy Sterowania, t. 3 (1973), z. 2, s.109-116.
3. Chasminskij R.Z.: Ustojcziwost sistem differencjalnych urawnienij pri szuczajnych wozmusczenijach ich parametrov. Nauka, Moskwa 1969.
4. Hahn W.: Theory and application of Liapunov's direct method, Prentice-Hall Inc., Englewood Cliffs, N.J. 1963.
5. Struble R.A.: Równania różniczkowe nieliniowe PWN, Warszawa 1965.
6. Węgrzyn S.: Podstawy automatyki. PWN, Warszawa 1972.
7. Węgrzyn S., Gille J.C.: O pewnym wystarczającym warunku stabilności, Arch. Elektr., t. XIII, zes. 1, 1964.
8. Węgrzyn S., Gille J.C., Vidal P., Palusiński O.: Wprowadzenie do analizy stabilności w przestrzeniach metrycznych, PWN, Warszawa 1970.

ЧИСЛЕННОЕ НАХОЖДЕНИЕ ОБЛАСТЕЙ УСТОЙЧИВОСТИ  
В СИСТЕМАХ КОМПЛЕКСНОЙ АВТОМАТИКИ

## Р е з ю м е

В работе представлены разные структуры систем комплексной автоматике. Более подробно описана программная структура, в которой показано место проблемы нахождения областей устойчивости. Далее представлена теорема, позволяющая находить полные области устойчивости для некоторого класса обыкновенных дифференциальных уравнений. Теорема иллюстрируется алгоритмом в языке АЛГОЛ и примерами как для детерминистических уравнений так и со случайными параметрами.

## NUMERICAL FINDING OF STABILITY AREAS IN COMPLEX CONTROL SYSTEMS

## S u m m a r y

In this paper some structures of complex control systems are presented. More accurately a software structure is described to show: place of stability areas finding problem in this structure. A Theorem allowing to solve problem of finding full stability areas for some class of ordinary differential equations is introduced. This theorem is illustrated by algorithm in ALGOL and by examples both for deterministic and with stochastic parameters equations.

Dodatek A**T w i e r d z e n i e**

Niech prawa strona układu (1):

$$\dot{x} = f(x), \quad x \in \mathbb{R}^2 \quad (1)$$

spełnia warunki twierdzenia o istnieniu i jednoznaczności:

$$\forall x \in \Omega \subset \mathbb{R}^2: f(x) \in C, \quad f_x(x) \in C \quad (2)$$

i niech jedynym skończonym jego punktem stacjonarnym będzie początek układu współrzędnych. Jeśli jest on punktem równowagi stabilnej, a rozwiązanie  $y = y(y_0, t)$  równania sprzężonego

$$\dot{y} = -f(y) \quad (3)$$

gdzie odległość  $\rho(y_0, 0) \leq \varepsilon$  dowolnie małej liczby dodatniej jest zbliżne do pewnej funkcji okresowej  $H(t)$ , której wykresem na płaszczyźnie fazowej układu sprzężonego jest krzywa  $J(y) = J(y^1, y^2)$ , to obszar

$$D_0 = \left\{ x: x \in \text{int } J(x^1, -x^2) \right\} \quad (4)$$

jest obszarem stabilności równania podstawowego. Jeśli funkcja  $H(t)$  nie istnieje w  $\Omega$ , to obszarem stabilności tego równania jest cały obszar  $\Omega$ .

**D o w ó d**

Jak wynika z założeń, prawe strony spełniają warunki twierdzenia o istnieniu i jednoznaczności rozwiązań układu równań różniczkowych, a więc przez każdy punkt obszaru jego określoności  $\Omega$  przechodzi jedna i tylko jedna trajektoria.

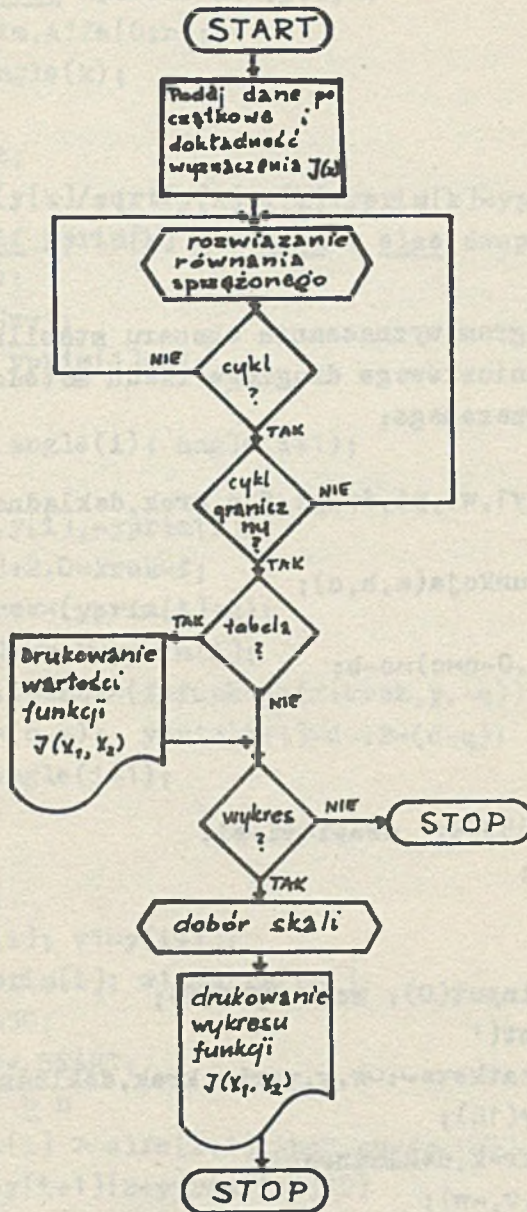
Przebieg trajektorii układu sprzężonego (3) jest geometrycznie identyczny z przebiegiem trajektorii układu (1), a jedynie kierunek ruchu

po nich dla  $t \rightarrow \infty$  jest przeciwny [5]. Jedyne skończony punkt stacjonarny układu (1):  $\dot{x}=0$  z założenia stabilny, jest niestabilnym punktem stacjonarnym układu sprzężonego (3). Mogą zajść dwa przypadki:

- a) niestabilna trajektoria układu (3) rozwija się do pewnej funkcji  $H(t)$ ,  $H(t+T) = H(t)$ , gdzie  $T$  jest okresem tej funkcji, której wykres na płaszczyźnie fazowej tego układu jest krzywa  $J(y^1, y^2)$  dla  $t \rightarrow \infty$ ,
- b) trajektoria układu (3) wychodzi z obszaru  $\Omega$  dla  $t \rightarrow \infty$ .

Oznacza to (z uwagi na jednoznaczność rozwiązań), że wszystkie trajektorie układu sprzężonego (3) zaczynające się we wnętrzu krzywej  $J(y^1, y^2)$  rozwijają się do tej krzywej. Tak więc wszystkie trajektorie układu (1) zaczynające się w zbiorze (4) nawijają się na jedyny punkt stacjonarny w nim się znajdujący, tj. na początek układu współrzędnych.

W przypadku b) trajektorie układu (3) rozwijają się poza granicę  $\Omega$ , a więc trajektorie układu (1) zaczynające się w dowolnym punkcie obszaru określoności równania  $\Omega$  nawijają się na punkt  $x = 0$ , co kończy dowód.



Rys. 6. Schemat blokowy programu wyznaczania obszaru stabilności metodą równań sprzężonych



Dodatek B

```

begin
comment program wyznaczania obszaru stabilności równania
    różniczkowego drugiego rzędu metoda równania
    sprzężonego;
integer n;
real u, v, w, v1, w1, pi, dwapi, f, q, krok, dokładność;
?
procedure funkcja(a, b, c);
real a, b, c;
funkcja := -(1.0 - c * c) * c - b;
?

n := 100;
pi := 3.14159265359; dwapi := pi + pi;
setinput(1);
if key(10)
then begin
    setinput(0); setoutput(0);
    print('
warunki początkowe: u, y, yprim, krok, dokładność?')
    end key(10);
read(u, v, w, krok, dokładność);
f := funkcja(u, v, -w);

q := w + krok * f; v1 := v + .5 * krok * (q + w);
w1 := w + .5 * krok * (f + funkcja(u + krok, v + krok * w, -q));
START: begin

```

```

integer i; real x,alfastart,p,c,d;
array y,yprim,alfa[0:n];
procedure angle(k);
integer k;
begin real t;
  t:=arccos(y[k]/sqrt(y[k]*y[k]+yprim[k]*yprim[k]));
  alfa[k]:=if yprim[k] >= 0 then t else dwapi-t;
end angle(k);
y[0]=v; y[1]:=v1;
yprim[0]:=w; yprim[1]:=w1;
x=u+krok;
STARTCYKL:i=0; angle(i); angle(i+1);
CYKL: i=i+1;
  f=funkcja(x,y[i],-yprim[i]);
  q=yprim[i-1]+2.0*krok*f;
  c=y[i]+.5*krok*(yprim[i]+q);
  p=y[i-1]+2.0*krok*yprim[i];
  d=yprim[i]+.5*krok*(f+funkcja(x+krok,p,-q));
  y[i+1]=c-.2*(c-p); yprim[i+1]=d-.2*(d-q);
  x=x+krok; angle(i+1);
  if i+1 >= n
  then begin
    u=x;
    v=y[i]; v1=y[i+1];
    w=yprim[i]; w1=yprim[i+1];
    n=n+50;
    go to START;
  end i+1 >= n
  else if alfa[i] > alfa[i+1] then go to CYKL;
  if abs(sqrt(y[i+1]2+yprim[i+1]2)
    -sqrt(y[0]2+yprim[0]2)) <= dokladnosc
then
  begin
    integer k,my,myprim;
    setinput(0); setoutput(0);
    print('??Czy=drukowac=tabelle=cyklu=granicznego?');

```

```

    outchar(68);
    print('jesli tak, napisz dowolna liczbe dodatnia,');
    print('jesli nie, to liczbe ujemna:');
    if inreal < 0 then go to DRUKWYKRESU;
DRUKTABELI: setoutput(1);
    print('
NUMERYCZNE WYZNACZANIE OBSZARU STABILNOSCI GLOBALNEJ
: parametry obszaru na płaszczyźnie fazowej
??
format('
123-0.12345610+12-0.12345610+12');
    for k=0 step 1 until i+1 do print(k,y[k],-yprim[k]);
DRUKWYKRESU: setinput(0); setoutput(0);
    print('Czy drukować wykresu cyklu granicznego?');
    outchar(68);
    print('jesli tak, napisz dowolna liczbe dodatnia,');
    print('jesli nie, to liczbe ujemna:');
    if inreal < 0 then go to KONIEC;
    print('
Podaj cyfry dziesiątek i punktów wykresu: ');
    read(my); line(1); space(37);
    print('wosi yprim:'); read(myprim);
    begin
    integer l;
    integer array phasesourf[0:10*myprim,0:10*my],
        kolumna[0:10*myprim];
    real ymin,ymax,yprimmin,yprimmax,skoky,skokyprim;
    for k=0 step 1 until 10*myprim do
        for l=0 step 1 until 10*my do phasesourf[k,l]=0;
    ymin:=ymax:=y[0]; yprimmin:=yprimmax:=yprim[0];
    for k=1 step 1 until i+1 do
    begin
        if y[k] < ymin then ymin:=y[k];
        if y[k] > ymax then ymax:=y[k];

```

```

    if -yprim[k] < yprimmin then yprimmin=-yprim[k];
    if -yprim[k] > yprimmax then yprimmax=-yprim[k];
end k;
skoky=(ymax-ymin)/(10*my);
skokyprim=(yprimmax-yprimmin)/(10*myprim);

for k=0 step 1 until i+1 do
    phasesurf[(yprimmax+yprim[k])/skokyprim,
               (y[k]-ymin)/skoky]=38;
    phasesurf[yprimmax/skokyprim,-ymin/skoky]=69;
    setoutput(1);
    print('
??=GRAFI CZNE WYZNACZANIE OBSZARU STABILNOSCI GLOBALNEJ=
? Parametry rysunku: ');
    format('
? yprimmin=-0.12310+12 yprimmax=-0.12310+12?
? yprimmin=-0.12310+12 yprimmax=-0.12310+12?
? yprimmin=-0.12310+12 yprimmax=-0.12310+12?
? yprimmin=-0.12310+12 yprimmax=-0.12310+12? ');
    print(ymin,ymax,skoky);
    format('
? yprimmin=-0.12310+12 yprimmax=-0.12310+12?
? yprimmin=-0.12310+12 yprimmax=-0.12310+12? ');
    print(yprimmin,yprimmax,skokyprim);
    for l=0 step 1 until my-1 do
        begin
            format('1,,,ψ,,,'); print(l);
        end l;
    outchar(if my=10 then 65 else my); line(1);
    kolumna[0]=16;
    kolumna[10*myprim]=if myprim=10 then 65 else myprim;
    for k=1 step 1 until 10*myprim-1 do kolumna[k]=27;
    for k=1 step 1 until myprim-1 do kolumna[10*k]=k;
    for k=5 step 10 until 10*myprim-5 do kolumna[k]=32;
    for k=0 step 1 until 10*myprim do

```

```
begin
  outchar(kolumna[k]);
  for l=0 step 1 until 10*my do
    outchar(phasesourf[k,l]);
  outchar(kolumna[k]); line(1);
  end k;
space(1);
for l=0 step 1 until my-1 do
  begin
    format('1,,,ψ,,,'); print(1);
  end l;
outchar(if my=10 then 65 else my); line(3);

  end DRUKWYKRESU;
KONIEC:stop;
  end
else copy(2,y[i],y[0]);
  copy(2,yprim[i],yprim[0]);
  copy(2,alfa[i],alfa[0]);
  go to STARTCYKL;
end START;
end
?
```