P.4201/85

PL ISSN 0209-1593

1 1985

prace naukowo-badawcze

Instytutu Maszyn Matematycznych

rok XXVII

I n s t y t u t M a s z y n M a t e m a t y c z n y c h

P.4201/85

p r a c e n a u k o w o - b a d a w c z e
I n s t y t u t u
M a s z y n
M a t e m a t y c z n y c h

Wojciech Mokrzycki: Całkowitoliczbowe dyskretyzacje krzywych algebraicznych w komputerowym obrazowaniu informacji - s.3

W a r s z a w a 1985

53. AA.203

Copyright © 1965 - by Instytut Maszyn Matematycznych
Poland
Wszelkie prawa zastrzeżone

KOMITET REDAKCYJNY

mgr inż. Zdzisław GROCHOWSKI
mgr Aleksander KAMIŃSKI
dr inż. Bronisław PIWOWAR /redaktor naczelny/
mgr inż. Jerzy SŁAWIŃSKI
prof. dr inż. Maciej STOLARSKI
doc. dr inż. Zdzisław WRZESZCZ

Opracowanie redakcyjne: mgr Hanna DROZDOWSKA

Adres redakcji: Instytut Maszyn Matematycznych
Branżowy Ośrodek INTE
02-078 Warszawa, ul.Krzywickiego 34
tel. 28-37-29 lub 21-84-41 w. 244

PK 244/88

Prace naukowo-badawcze Instytutu Maszyn Matematycznych

Całkowitoliczbowe dyskretyzacje krzywych algebraicznych w komputerowym obrazowaniu informacji

Wojciech MOKRZYCKI

Wykaz oznaczeń
1. Wstęp
1.1. Uzasadnienie podjęcia tematu
1.2. Podstawowe pojęcia i definicje
1.3. Układ treści rozprawy
2. Strukturalne, warunkowe i strukturalno-warunkowe dyskretyzacje
2.1. Dyskretyzacje strukturalne
2.2. Dyskretyzacje warunkowe na siatkach G_{4s} i G_{8s}
2.3. Dyskretyzacje strukturalno-warunkowe na siatkach G_{8s}
2.4. Dyskretyzacje warunkowe na siatkach G_{ks}
2.5. Problemy badawcze w dyskretyzacji krzywych
2.6. Problem badawczy rozprawy
3. Algorytmy dyskretyzacji krzywych algebraicznych na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$
3.1. Sformułowanie zagadnienia dyskretyzacji
3.2. Analiza zagadnienia
3.2.1. Dyskretyzacja na siatce $G_{4s}(h^2)$
3.2.2. Dyskretyzacja na siatce $G_{8s}(h^2)$
3.2.3. Dane początkowe dyskretyzacji
3.2.4. Izolowanie punktów osobliwych krzywej oraz wybór początkowego i końcowego węzła generowanego segmentu
3.3. Algorytmy dyskretyzacji krzywych
3.3.1. Algorytm dyskretyzacji na siatce $G_{4s}(h^2)$
3.3.2. Algorytm dyskretyzacji na siatce $G_{8s}(h^2)$
3.3.3. Algorytmy dla krzywych 2-stopnia
4. Analiza złożoności obliczeniowej algorytmów
4.1. Całkowitoliczbowość algorytmów
4.2. Złożoność obliczeniowa algorytmów
4.2.1. Złożoność obliczeniowa wyznaczania jednego węzła ciągu dla $h \neq 1$
4.2.2. Złożoność obliczeniowa wyznaczania jednego węzła ciągu dla $h=1$
4.2.3. Złożoność obliczeniowa wyznaczania jednego węzła ciągu na podstawie równania ogólnego krzywej
4.3. Porównanie złożoności obliczeniowej algorytmów dla generacji krzywej dyskretnej
4.4. Złożoność obliczeniowa dyskretyzacji krzywych
5. Analiza błędów dyskretyzacji
5.1. Szczytowe błędy dyskretyzacji
5.1.1. Bezwzględne szczytowe błędy dyskretyzacji
5.1.2. Względne szczytowe błędy dyskretyzacji
5.2. Średnioglobalna ocena jakości dyskretyzacji
5.3. Podstawowe rezultaty oceny jakości aproksymacji

6. Zakończenie
6.1. Podstawowe rezultaty pracy
6.2. Przewidywane zastosowania
6.3. Określenie kierunku dalszych badań
Literatura
Dodatek: Komputerowa implementacja algorytmów

Wykaz oznaczeń

a, b	- półosie krzywej 2- stopnia
a_{ij}	- współczynniki równania uwikłanego krzywej
$\mathfrak{A}$	- zbiór algorytmów dyskretyzacji
A	- element zbioru $\mathfrak{A}$
A_i, B_i	- nazwy podokresów w kodzie prostej dyskretnej
B	- zmienna logiczna
BSWG	- bezpośrednio spójne węzły siatki
$c_i, k_i, m, m_i, n_i, p_i, q_i, r_i$	- wskaźniki pomocnicze, na ogół elementy ciągów liczb całkowitych
C, D	- zbiory parzystych i nieparzystych kierunków elementarnych ruchów /wektorów/ wzdłuż linii siatki $G_{8s}(h^2)$
C_i, D_i	- elementy zbiorów C i D
$C_i^{n_i} D_i$	- podokres w kodzie prostej dyskretnej
d, d_i, d_{ij}	- odległości między punktami /stałe bądź zmienne/
e	- parametr krzywej 2-stopnia
E(x)	- funkcja entier
$f_{O(1)k}$, $g_{O(1)k}$	- sekwencje liczb naturalnych ze wskaźnikami zmieniającymi się od 0 do k
F(x,y) = 0	- równanie funkcji unikłanej opisujące krzywą dyskretyzowaną K
$F_i(x,y)$, $F(P_i)$	- wartości funkcji F(x,y) w punktach o numerach "i" oraz "P_i"
$G(h^2)$	- siatka dyskretyzacyjna /jednorodna kwadratowa z modułem h: G_{4s} - 4 spójna, G_{8s} - 8-spójna, G_{ks} - k-spójna/
GLG	- główne linie siatki dyskretyzacyjnej
h	- moduł siatki dyskretyzacyjnej
H	- krzywa dyskretna powstała w wyniku dyskretyzacji krzywej K
i,j,k,l,s,r,p,q	- wskaźniki /liczby całkowite/
K	- element zbioru $\mathfrak{K}$ /krzywa podlegająca dyskretyzacji/
$\mathfrak{K}$	- zbiór /klasa/ krzywych dyskretyzowanych
l_i	- długość cięciwy wyznaczonej przez punkty dwóch kolejnych przecięć krzywej z łamaną
$l^H_{4s,8s,ro}$	- długość krzywej dyskretnej generowanej za pomocą algorytmu dyskretyzacji na siatce, odpowiednio: $G_{4s}(h^2)$, $G_{8s}(h^2)$ oraz wprost z równania /3.1/
l^K	- długość krzywej dyskretyzowanej K
L, L(i), $L(P_i/P_j)$	- linia, linia o numerze "i", linia wyznaczona przez punkty P_i oraz P_j
L(A,K)	- liczba operacji niezbędnych do przeprowadzenia dyskretyzacji krzywej K za pomocą algorytmu A
L^o_j	- liczba operacji wyznaczania pochodnych cząstkowych według jednej zmiennej
L^o_m	- liczba operacji wyznaczania pochodnych cząstkowych mieszanych
L^o_Σ	- liczba operacji wyznaczania wszystkich pochodnych cząstkowych
L^o_ε	- liczba operacji wyznaczania operatorów $L^{(\cdot)}F(x,y)$ oraz $L^{(\cdot)}F_{rxsy}(x,y)$ dla algorytmów z $h \neq 1$
L^o_1	- liczba operacji wyznaczania operatorów $L^{(\cdot)}F(x,y)$ oraz $L^{(\cdot)}F_{rxsy}(x,y)$ dla algorytmów z $h = 1$

L_r^o - liczba operacji wykonywanych w pętli algorytmu dyskretyzacji krzywej na podstawie jej równania ogólnego

L_ϵ^p - liczba operacji wykonywanych w pętli algorytmu dyskretyzacji na siatce z modułem $h \neq 1$

L_1^p - liczba operacji wykonywanych w pętli algorytmu dyskretyzacji na siatce z modułem $h = 1$

$L^xF(x,y)$, $L^yF(x,y)$ | $L^PF(x,y)$, $L^RF(x,y)$ - operatory kierunkowe funkcji $F(x,y)$ odpowiednio x-owy, y-owy, przekątny oraz różnicowy

$L^xF_{rxsy}(x,y)$ | $L^yF_{rxsy}(x,y)$ | $L^PF_{rxsy}(x,y)$ | $L^RF_{rxsy}(x,y)$ - operatory kierunkowe pochodnych mieszanych /r+s/-rzędu $F_{rxsy}(x,y)$ funkcji $P(x,y)$ odpowiednio: x-owy, y-owy, przekątny i różnicowy

LV1,LV1',LV2,LV2' - lokalizacja 1- i 2-stopnia wektora stycznej $\vec{V}$

τ_i - długość łuku krzywej wyznaczonego przez punkty dwóch kolejnych przecięć krzywej z łamaną

M - zbiór węzłów siatki dyskretyzacyjnej

m - rozmiar liniowy obszaru dyskretnego

n - stopień krzywej /liczba naturalna/

p_o, q_o - liczby całkowite p i q zredukowane do liczb względem siebie pierwszych

P, P(i), P(x,y) - punkt, punkt o numerze "i", punkt o współrzędnych $<x,y>$

(r+s) PLG - (r+s)- przekątne linie siatki dyskretyzacyjnej

Q - nachylenie linii na płaszczyźnie dyskretnej

Q_h - funkcja siatkowa: zależność między odległościami węzłów ciągu od krzywej w ramach elementarnego pola siatki

r=1(1)s - zmiana wskaźnika r od wartości 1 do s co 1

res - reszta z całkowitoliczbowego dzielenia liczb

R - promień krzywizny krzywej /stały bądź zmienny/

RSG - rząd spójności siatki dyskretyzacyjnej

$R_{t,i}(\cdot)$ - wyrażenie /funkcja/ pomocnicze

$sgn(n_i)$ - znak liczby n_i

S_i - pole powierzchni zawarte między krzywą i jej aproksymantą dla dwóch kolejnych wzajemnych przecięć

t - parametr

T, T_p, T_{ok} - pasmo tolerancji dyskretyzacji krzywej /T_p - dla prostej, T_{ok} - dla okręgu/

$\vec{V}^+$, $\vec{V}^-$ - wektory kierunkowe stycznej

W_x, W_y - wymiary powierzchni obrazowania mierzone liczbą znaków

WKD - warunek końca dyskretyzacji

$<x_i,y_i>$ - współrzędne bieżące i-tego węzła ciągu aproksymującego

$<x_p,y_p>$, $<x_k,y_k>$ - początkowy i końcowy punkt segmentu dyskretyzowanej linii

$Z(A)$ - złożoność obliczeniowa algorytmu A

Z^* - złożoność obliczeniowa dyskretyzacji krzywych K klasy $\mathcal{K}$ ze względu na zbiór algorytmów A

$+,-,\cdot,:,\vee,\wedge$ - operacje arytmetycznego dodawania, odejmowania, mnożenia i dzielenia oraz logiczne - alternatywy i koniunkcji

α, β, δ - kąty nachylenia stycznych do krzywej w dwóch kolejnych węzłach ciągu aproksymującego oraz kąt środkowy

δ, δ^i, δ_i - funkcje pomocnicze

Δ - nachylenie całkowitoliczbowe linii dyskretnej

$\Delta F(x,y)$ - przyrost wartości funkcji $F(x,y)$ w trakcie ruchu z węzła $<x_i,y_i>$ do węzła $<x_{i+1},y_{i+1}>$

Δt - przyrost parametru t

Δx, Δy - przyrosty wzdłuż współrzędnych x oraz y

$*$ - operacja całkowitoliczbowego dzielenia

φ_1, φ_2 - odchylenie kątów nachylenia stycznych /od kąta środkowego/

ψ - kąt nachylenia głównych linii siatki w stosunku do osi układu współrzędnych

θ - kąt nachylenia stycznej w stosunku do głównych linii siatki

$\frac{\partial F}{\partial x}$, F_x; $\frac{\partial F}{\partial y}$, F_y - pochodne cząstkowe I-rzędu funkcji F(x,y) względem zmiennych x oraz y odpowiednio

$\frac{\partial^r F}{\partial x^r}$, F_{rx}; $\frac{\partial^s F}{\partial y^s}$, F_{sy} - pochodne cząstkowe r- i s-rzędu względem zmiennych odpowiednio x i y

$\frac{\partial^{r+s} F}{\partial x^r \partial y^s}$, F_{rxsy} - pochodna cząstkowa mieszana (r+s)- rzędu funkcji F(x,y) względem zmiennych x/r-rzędu/ i y/s-rzędu/

ε_i, ε_{i+1} - kąty środkowe krzywej K wyznaczone przez trzy kolejne punkty przecinania się krzywej z łamaną

ϱ - bezwzględny błąd szczytowy dyskretyzacji krzywej mierzony prostopadle do krzywej

ϱ_F^H - błąd szczytowy dyskretyzacji krzywej K mierzony wartością $|F(x,y)|$

ϱp_i - moduł pola przylegający do krzywej

ϱ_{sp} - wartość średnia modułu pola przylegającego do jednostkowego odcinka krzywej

ϱw_i - względny szczytowy błąd aproksymacji

$\varrho w_{śr}$ - wartość średnia względnego szczytowego błędu aproksymacji

$\lfloor x \rfloor$ - wartość liczby x zaokrąglona do najbliższej liczby całkowitej, nie większej od x

$[x]$ - wartość liczby x zaokrąglona do najbliższej liczby całkowitej,nie mniejszej od x

$[x]$ - wartość liczby x zaokrąglona do najbliższej liczby całkowitej

■ - koniec dowodu.

1. WSTĘP

1.1. Uzasadnienie podjęcia tematu

W wielu różnych dziedzinach nauki i techniki, a więc także mikroelektronice i informatyce, wynikają potrzeby doskonalszych bądź nowych rozwiązań w dziedzinie metod numerycznych, m.in. w zakresie dyskretyzacji i aproksymacji krzywych. Potrzeby te odnoszą się zwłaszcza do tych dziedzin zastosowań, w których zachodzące procesy opisują się prościej w przestrzeni dyskretnej, ze zdefiniowanymi węzłami całkowitoliczbowej siatki współrzędnych, np. w monitorach graficznych, kreślakach, dyskretyzatorach, cyfrowym sterowaniu obrabiarkami, mapach topograficznych, itp. Terminale obrazowe, w tym graficzne w komputerowych systemach użytkowych wielkich i małych, a nawet w mikrokomputerach osobistych stają się bowiem coraz powszechniejsze, a ich znaczenie ciągle rośnie. Zwiększają się też wymagania dotyczące jakości zobrazowania: żąda się ergonomicznej formy przedstawiania opracowanych wyników, tzn. dostosowanej do cech psychofizycznych człowieka.

Dyskretyzacji stawia się więc szczególne, jakościowo nowe wymagania, jak np. bardzo duża szybkość wykonywania algorytmów, używanie jedynie najprostszych operacji /typu sumowania, przesuwu wartości binarnych w rejestrach, prostych operacji logicznych/, działania jedynie w arytmetyce całkowitoliczbowej, lub w najgorszym wypadku stołoprzecinkowej, wyeliminowanie kumulacji błędów, itp. Wymagań tych nie spełniają dobrze znane klasyczne metody dyskretyzacji /aproksymacji/ krzywych, w których punkty aproksymujące oblicza się za pomocą podstawowych funkcji trygonometrycznych lub rozwinięć

w szeregi potęgowe. Ich stosowanie jest przyczyną kumulujących się w procesie obliczeń błędów systematycznych, prowadzących do "degeneracji" krzywych, zwłaszcza przy długotrwałych obliczeniach.

W związku z tymi potrzebami opracowano kilka metod tzw. całkowitoliczbowych aproksymacji /dyskretyzacji/, które aproksymują krzywe ciągiem punktów mających współrzędne całkowitliczbowe i ze swojej natury działania nie wnoszą błędów systematycznych. Odrzucone zostało założenie typowe dla klasycznych metod dyskretyzacji, że punkty aproksymujące powinny leżeć na krzywej. W metodach całkowitoliczbowych punkty aproksymujące nie leżą na krzywej z założenia. Mają one jednak takie współrzędne całkowitoliczbowe, których odchylenie od punktów zadanej krzywej jest z góry ograniczone i nie przekracza odległości między sąsiednimi węzłami siatki, czyli nie przekracza jednej jednostki. Poza tym odchylenie to jest uwzględniane podczas wyznaczania kolejnego punktu ciągu. Istnieje bowiem zależność między warunkami wyboru kolejnego punktu ciągu i odchyleniem poprzedniego punktu od krzywej zadanej.

Krzywa dyskretyzowana opisana jest zwykle w kartezjańskim układzie współrzędnych. W tym układzie zapis większości krzywych oraz różne przekształcenia krzywych są tradycyjnie stosowane. Poza tym wyrażają się one w prosty sposób i realizują za pomocą stosunkowo prostych operacji. W tym wypadkach dla celów dyskretyzacji wprowadza się siatki prostokątne, najczęściej kwadratowe jednorodne i na nich przeprowadza się dyskretyzację krzywych.

Występują jednak zastosowania /np. rastrowe monitory graficzne z radialnym bądź koncentrycznym rozwinięciem obrazu/, w których zapis krzywych i realizowanych na nich operacji wyraża się prościej w innych układach współrzędnych, np. w układzie trójkątnym czy biegunowym. Wprowadza się wtedy odpowiednie siatki: trójkątne, sześciokątne, koncentryczne, spiralne, itp. i dyskretyzację krzywych przeprowadza się na tych właśnie siatkach.

W niektórych z zastosowań stawia się wymaganie układowej realizacji dyskretyzacji, co oznacza, że algorytmy powinny prowadzić do łatwych i prostych rozwiązań technicznych. Wymaganie to wyklucza stosowanie funkcji trygonometrycznych, wykładniczych, pierwiastkowań, a nawet dzieleń i mnożeń przez dowolne liczby rzeczywiste, czy podobnego typu operacje.

Ich realizacja techniczna jest zwykle złożona i trudna, wymaga bowiem skomplikowanych rozwiązań układowych i z natury rzeczy wprowadza błędy systematyczne /zaokrąglanie do wartości całkowitych/. Wymaganie możliwie prostej realizacji układowej algorytmu narzuca na niego ograniczenie stosowania jedynie prostych operacji, tzn, sumowania liczb całkowitych lub w najgorszym wypadku stałoprzecinkowych, mnożenia i dzielenia przez całkowite potęgi dwójki, proste operacje logiczne typu sumy i iloczynu na argumentach boolowskich, operacje porównywania argumentów stałoprzecinkowych lub inne.

Niektóre z metod są ukierunkowane na ściśle określone zastosowania, w których występują zwiększone wymagania w stosunku do określonego parametru dyskretyzacji, jak np. zwiększona dokładność czy "gładkość" w sterowaniu obrabiarkami lub duża szybkość w monitorach graficznych. W niektórych z zastosowań pewne cechy mogą być pierwszoplanowe /kosztem innych/, np. dokładność i "gładkość" kosztem szybkości /w sterowaniu obrabiarkami/ lub też szybkość generacji kosztem dokładności /monitory graficzne/.

W dyskretyzacji złożonych krzywych wyższych stopni może wyniknąć problem niedostatecznej pojemności pamięci operacyjnej komputera, czy też niedostatecznego przedziału liczbowego dla różnych reprezentacji liczbowych. Powstać więc mogą dodatkowe utrudnienia, czy nawet nieprzewidziane wcześniej błędy /np. wynikłe ze sposobu reprezentacji liczb w komputerze/.

1.2. Podstawowe pojęcia i definicje

Stosowany w teorii aproksymacji termin "dyskretyzacja" oznacza odwzorowanie zbioru funkcji rzeczywistych lub zespolonych w zbiór n-wymiarowych ciągów liczb rzeczywistych, zespolonych bądź też w zbiór nieskończonych ciągów liczbowych.

W teorii sygnałów termin "dyskretyzacja" dotyczy konwersji sygnałów ciągłych na dyskretne i obejmuje dwie operacje, tzw. próbkowanie oraz kwantowanie /porcjowanie/. Próbkowanie polega na pomiarze /analogowych/ wartości ciągłego sygnału w określonych odstępach czasowych. Jest więc przekształceniem ciągłego przebiegu sygnału w odpowiadający mu dyskretny zbiór próbek. Kwantowanie /porcjowanie/ polega na zamianie analogowej wartości próbki najbliższym tej próbce poziomem wartości sporcjowanej /skwantowanej/.

W niniejszej rozprawie termin "dyskretyzacja linii" rozumiem z jednej strony jako jedną z metod aproksymacji linii, ale przede wszystkim jako odwzorowanie analitycznie zadanych, ciągłych krzywych określonych równością $F(x,y) = 0$, w skończony zbiór dyskretnych wartości. Zbiór ten jest dyskretnym obrazem linii ciągłej /analitycznej/ i aproksymuje /w sensie pewnej metryki, np. metryki przestrzeni euklidesowej/ linię zadaną.

Motywacją do dyskretyzacji linii jest potrzeba stosowania technik i metod cyfrowych w różnych procesach technicznych /opisujących się za pomocą dyskretyzowanych krzywych/, a zwłaszcza w obrazowaniu informacji za pomocą graficznych urządzeń komputerowych.

Ze szczególnym zainteresowaniem traktuję dyskretyzację linii na dwuwymiarowych siatkach. Polega ona na zdefiniowaniu, na obrazującej linię płaszczyźnie, dwuwymiarowej siatki i wyborze węzłów tej siatki leżących wzdłuż linii i tworzących ciąg, który przybliża linię zadaną. Ciąg ten nazywam linią dyskretną. Kolejne węzły linii dyskretnej łączy się zwykle aproksymantami, tj. elementarnymi wektorami skierowanymi wzdłuż linii siatkowych bądź skierowanymi segmentami wybranych krzywych.

Standaryzacja aproksymant /do zbioru np. 4- bądź 8-elementowego/ i ich ponumerowanie /kolejnymi liczbami ze względu, np. na wzrastanie ich kąta nachylenia/ umożliwia zapisanie linii dyskretnej w tzw. "łańcuchowo-kodowej" postaci. Jest to ciąg liczb-numerów aproksymant, których łańcuch tworzy linię dyskretną.

Znaną techniką dyskretyzacji linii, prowadzącą wprost do łańcuchowo-kodowej reprezentacji linii jest wybieranie, dla każdego przecięcia krzywej z linią siatki, najbliższego temu przecięciu węzła i utworzenie ciągu z numerów aproksymant łączących kolejno wybrane węzły.

W dalszych rozważaniach nad odwzorowaniem krzywych ciągłych na krzywe dyskretne używam /w wielu wypadkach zamiennie/ trzech następujących terminów/:

- aproksymacja krzywej analitycznej /zadanej/ za pomocą krzywej dyskretnej,
- dyskretyzacja krzywej analitycznej /zadanej/ do postaci krzywej dyskretnej,
- generacja krzywej dyskretnej na podstawie /zadanych/ parametrów krzywej analitycznej.

Pojęcia "aproksymacja krzywej" używam w rozważaniach nad przybliżaniem krzywej ciągłej - krzywą dyskretną szczególnie wtedy, gdy oceniam jakość tego przybliżenia.

Pojęć "dyskretyzacja krzywej" oraz "generacja krzywej" używano w trakcie rozważań nad zbiorem elementów wykorzystywanych do tworzenia krzywej dyskretnej, nad generacją tej linii i złożonością tej generacji. Drugiego z tych terminów używam szczególnie w trakcie omawiania algorytmów generacji krzywej dyskretnej /generacji ciągu sporcjowanych wartości współrzędnych kolejnych punktów-węzłów bądź standaryzowanych aproksymant tworzących krzywą dyskretną/ i w trakcie oceny złożoności tej generacji.

W rozważaniach uwzględniam więc trzy parametry dyskretyzacji:

- formę /jednorodną kwadratową siatkę na płaszczyźnie/,
- rozmiar /moduł siatki/,
- aproksymantę użytą do reprezentacji porcji /odcinek prostej bądź standardowy segment krzywej/.

W niniejszej rozprawie rozważania ograniczam jedynie do dyskretyzacji krzywych algebraicznych płaskich na jednorodnych kwadratowych siatkach za pomocą prostoliniowych aproksymant, innymi słowy, rozważania jest aproksymacja, w której aproksymantami mogą być jedynie odcinki linii siatki, na której jest ta aproksymacja dokonywana.

Niech w układzie Oxy będzie zadana kwadratowa jednorodna siatka $G(h^2)$ za pomocą linii:

- pionowych $x_i = h \cdot i$, $-\infty < y < +\infty$ oraz
- poziomych i ukośnych $y_i = h \cdot i + (p/q) \cdot x$, $-\infty < x < +\infty$

gdzie:

h - moduł siatki, $i = -\left\lfloor \frac{m-1}{2} \right\rfloor (1) \left\lfloor \frac{m}{2} \right\rfloor$,

m - rozmiar liniowy siatki, /h oraz m - liczby naturalne/,

p i q - liczby całkowite.

Pomocnicze definicje dotyczące struktury siatki $G(h^2)$.

Definicja 1. Węzłami siatki nazwano punkty przecinania się głównych linii siatkowych.

Definicja 2. Rzędem spójności siatki RSG nazwano liczbę półprostych siatkowych wychodzących z dowolnego węzła siatki.

Definicja 3. Bezpośrednio spójnymi węzłami siatki BSWG nazwano takie dwa węzły, które linia siatkowa łączy bezpośrednio /bez pośrednictwa innych węzłów siatki/.

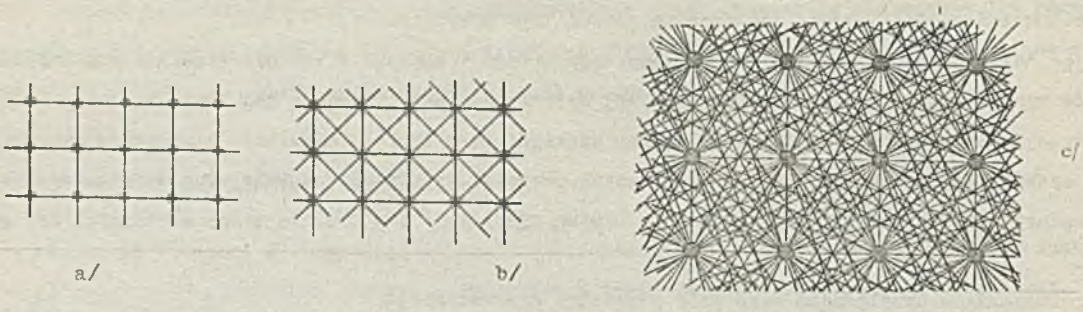
Definicja 4. Głównymi liniami siatki GLG nazwano te linie siatki, wzdłuż których odległość między bezpośrednio spójnymi węzłami równa jest h, 2-przekątnymi 2PLG - gdy ta odległość jest równa $\sqrt{2}\,h$, (r+s)-przekątnymi (r+s)PLG - gdy jest ona równa $\sqrt{r^2+s^2}\,h$ /r i s - liczby naturalne/.

Przyjmując za p i q różne liczby całkowite /ale $q \neq 0$/, otrzymuje się różne rodzaje siatek:

1/ p=0; jednorodna kwadratowa siatka 4-spójna $G_{4s}(h^2)$ z linii:
   - pionowych $x_i = h \cdot i$, $-\infty < y < +\infty$, oraz
   - poziomych $y_i = h \cdot i$, $-\infty < x < +\infty$;

2/ $p/q = \pm 1$; jednorodna kwadratowa siatka 8-spójna $G_{8s}(h^2)$ z linii:
   - pionowych $x_i = h \cdot i$, $-\infty < y < +\infty$, oraz
   - poziomych i ukośnych - $y_i = h \cdot i \pm x$, $-\infty < x < +\infty$;

3/ p,q - dowolne liczby całkowite, $q \neq 0$; otrzymano siatkę k-spójną $G_{ks}(h^2)/k=2^l$, l - liczba naturalna/.



Rys.1. Jednorodne siatki kwadratowe: a/ 4-spójna, b/ 8-spójna, c/ 32-spójna

Aproksymanty na siatce mogą mieć długości obejmujące wszystkie możliwe odległości między węzłami w całej przestrzeni dyskretnej. W praktyce najczęściej kilka-kilkanaście jednostek.

1.3. Układ treści rozprawy

Wyniki analizy rozważanych zagadnień autor starał się uogólnić do poziomu twierdzeń i w takiej postaci /sześciu twierdzeń z dowodami/ zawarł w tekście rozprawy.

Rozprawa składa się z sześciu rozdziałów oraz dodatku.

Po wstępie, w rozdziale 2 omówiłem znane z literatury algorytmy całkowitoliczbowej dyskretyzacji /aproksymacji/ krzywych. Algorytmy te podzieliłem na grupy według metodyki konstruowania ciągów aproksymujących, typu kryterium wyboru węzłów siatki na ciąg aproksymujący oraz rodzaju siatki aproksymacyjnej. W każdej z grup algorytmy przedstawiłem w kolejności ich publikacji. Stan dotychczasowych badań zaprezentowałem więc jako zespół następujących zagadnień szczegółowych:

- rodzaju siatki, na której jest dokonywana dyskretyzacja,
- metodyk konstruowania ciągów aproksymujących /podejście warunkowe, strukturalne i strukturalno-warunkowe/,
- kryteriów wyznaczania węzłów ciągu aproksymującego.

Omawiając zależności aproksymacji od rodzaju siatki skupiłem uwagę przede wszystkim na siatkach $G_{4s}(h^2)$, $G_{8s}(h^2)$ i $G_{ks}(h^2)$.

Analizując różne metodyki dyskretyzowania krzywych rozróżniam algorytmy dyskretyzacji oparte na podejściu strukturalnym, warunkowym oraz strukturalno-warunkowym.

Spośród wyznaczania całkowitoliczbowych punktów ciągu aproksymującego stosowane są różne: minimum odległości liniowej węzła od krzywej bądź minimum wartości funkcji $F(x,y)$ opisującej krzywą zadaną równaniem $F(x,y) = 0$ w tych węzłach, bądź przemienności znaków wartości funkcji $F(x,y)$ w kolejnych węzłach ciągu aproksymującego; niekiedy też wyznacza się całkowitoliczbowe pierwiastki równania $F(x,y) = 0$, będące współrzędnymi punktów, które mają obydwie cechy, tj. są całkowitoliczbowe i leżą na krzywej. Stosowane są też inne kryteria.

Podstawowe treści rozprawy, obejmujące analizę algorytmów całkowitoliczbowej dyskretyzacji krzywych algebraicznych płaskich na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$, ocenę złożoności obliczeniowej

algorytmów dyskretyzacyjnych oraz szacowanie szczytowych i polowych błędów aproksymacji za pomocą tych algorytmów, zawarte są w rozdziałach: 3, 4 i 5.

W rozdziale 3 przeprowadziłem analizę dyskretyzacji krzywych algebraicznych płaskich na jednorodnych siatkach kwadratowych i, na podstawie metod warunkowych, zsyntetyzowałem dwa algorytmy dyskretyzacji tych krzywych na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$. Metryką generacyjną w tych algorytmach jest moduł wartości funkcji $F(x,y)$.

W rozdziale 4 oszacowałem złożoność obliczeniową i wynikającą z niej szybkość realizacji opracowanych w rozdziale 3 algorytmów oraz wyprowadziłem wzory na liczbę tzw. "operacji głównych", wykonywanych w trakcie dyskretyzacji krzywych. Liczba ta określa się wielomianem względem stopnia krzywej. Stopień wielomianu zależy natomiast od zastosowanej metody dyskretyzacji oraz od wielkości modułu siatki h. Złożoność dyskretyzacji krzywych algebraicznych za pomocą opisywanych metod okazała się więc złożonością wielomianową.

W rozdziale 5 przeprowadziłem lokalną i globalną ocenę jakości aproksymacji krzywych algebraicznych zadanych równaniem $F(x,y) = 0$ na jednorodnych siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$. Dokładność aproksymacji wzrasta wraz ze wzrostem rzędu spójności siatki, na której jest dokonywana dyskretyzacja.

W rozdziale 6 podsumowałem uzyskane w rozprawie wyniki i wskazałem na potrzebę dalszych badań w dziedzinie całkowitoliczbowej aproksymacji /aproksymacji na siatkach/.

W dodatku omówiłem implementację algorytmów na komputerach IRIS 80 oraz MERA 400 i przedstawiłem rysunki zdyskretyzowanych krzywych 3-, 4- i 6-stopnia na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$, z różnymi wartościami h.

2. STRUKTURALNE, WARUNKOWE I STRUKTURALNO-WARUNKOWE DYSKRETYZACJE

Podział na dyskretyzację strukturalną, warunkową oraz strukturalno-warunkową wynika z zastosowanej metodyki konstrukcji linii dyskretnej /wyboru węzłów aproksymujących zadaną linię/.

W podejściu strukturalnym zakłada się występowanie określonej struktury linii dyskretnej, tj. występowanie w linii dyskretnej związków między długością i nachyleniem tworzących ją segmentów linii siatkowych oraz kolejnością ich występowania, zależnie bezpośrednio i jednoznacznie od parametrów linii zadanej /termin "struktura" używany jest tutaj w sensie potocznym/.

W podejściu warunkowym zakłada się, że o wyborze punktu-węzła na węzeł aproksymujący decyduje wynik testowania pewnego warunku /zwykle arytmetycznego/, na podstawie którego wybiera się z sąsiadujących z krzywą węzłów kolejny węzeł aproksymujący. Warunek ten pozwala wybrać węzeł najbliższy krzywej zadanej według określonej /zwykle liniowej/ metryki.

Każda z metod dyskretyzacji ma swoje wady i zalety.

Metody strukturalne są bardziej skomplikowane, wymagają większej liczby złożonych operacji oraz długiego czasu realizacji, a także większych obszarów pamięci komputera. W zamian za to dają przejrzysty obraz struktury linii dyskretnej i występujących w niej uwarunkowań. Z literatury znanych jest jednak zaledwie kilka /sześć/ algorytmów dyskretyzacji opartych na podejściu strukturalnym i wszystkie dotyczą dyskretyzacji linii prostej na siatce $G_{8s}(h^2)$.

Metody warunkowe, opisywane w literaturze wcześniej i częściej są metodami prostszymi, szybszymi, używającymi prostych operacji arytmetycznych, wymagającymi /w ich komputerowej programowej realizacji/ niewielkich pojemności pamięci operacyjnej komputera oraz są doskonalsze. Z literatury znanych jest kilkanaście różnych podejść warunkowych do dyskretyzacji linii, a dotyczą one tak dyskretyzacji prostej, jak i krzywych drugiego i wyższych stopni. Niestety, metody warunkowych dyskretyzacji nie uwidaczniają struktury linii dyskretnej.

Znane są również algorytmy, w syntezie których zastosowano zarówno metody strukturalne, jak i warunkowe.

W podejściu strukturalno-warunkowym w pewnym stadium działania są wykonywane operacje charakterystyczne dla podejścia strukturalnego, a w stadium kolejnym - operacje typowe dla podejścia warunkowego. Algorytmy tego typu łączą w sobie zalety obydwu metod dyskretyzacji: prostotę obliczeniową metody warunkowej oraz przejrzystość struktury linii dyskretnej metody strukturalnej.

W dalszej części rozdziału zaprezentowałem znane z dostępnej literatury algorytmy dyskretyzacji krzywych: strukturalne, warunkowe oraz strukturalno-warunkowe. Omówiłem również algorytmy dyskretyzacji krzywych algebraicznych na jednorodnych kwadratowych siatkach $G_{ks}(h^2)$.

Na zakończenie rozdziału sformułowałem problemy badawcze w całkowitoliczbowej dyskretyzacji krzywych oraz problem badawczy rozprawy.

2.1. Dyskretyzacje strukturalne

Dyskretyzacje strukturalne opierają się na założeniu, że linia dyskretna ma ściśle określoną strukturę, tj. tworzące ją odcinki linii siatkowych o różnych długościach i nachyleniach są uszeregowane według ściśle określonej zasady, a typy tych odcinków, ich długości, nachylenia i kolejności występowania są całkowicie zdeterminowane parametrami linii zadanej.

Z literatury znana jest jedynie struktura prostych zdyskretyzowanych na siatce $G_{8s}(h^2)$, opisana przez Freemana w 1970 r. i sformułowana w postaci trzech podstawowych zasad:

1/ w łańcuchu prostej dyskretnej występują co najwyżej dwie różne aproksymanty,

2/ jeżeli w łańcuchu prostej dyskretnej występują dwie różne aproksymanty, to mają one kierunki sąsiednie /tzn. różniące się o kąt $\pm\pi/4$/,

3/ jeżeli w łańcuchu prostej dyskretnej występują dwie różne aproksymanty, to są one w łańcuchu prostej tak jednolicie uszeregowane, jak jest to tylko możliwe.

Nie są jednak znane struktury krzywych dyskretnych.

Metody strukturalnej dyskretyzacji są stosunkowo złożone, wymagają wykonania znacznej liczby operacji oraz obszaru pamięci komputera na jednostkę długości linii dyskretyzowanej. Liczba ta jednak zależy w niewielkim tylko stopniu od długości dyskretyzowanej linii. Zależy również od kąta nachylenia linii. Proces dyskretyzacji sprowadza się do sprecyzowania okresów i podokresów w strukturze linii dyskretnej i ewentualnie do wypośrodkowania linii dyskretnej w stosunku do linii zadanej, w celu minimalizacji błędów szczytowych dyskretyzacji.

Dotychczas opisanych zostało zaledwie kilka algorytmów strukturalnych i wszystkie dotyczą dyskretyzacji linii prostej.

Metzger R.A. w 1969 r. jako pierwszy zaproponował algorytm strukturalnej dyskretyzacji prostej. Centralnym problemem w jego metodzie jest problem określenia wartości Δ - liczby /całkowitej/ elementarnych przyrostów wzdłuż osi x, przypadających na jednostkowy przyrost wzdłuż osi y /dla prostej o kącie nachylenia $0 \leqslant \alpha \leqslant 45^{\circ}$/. Jeżeli Δ wyznaczona jest według wzoru /punkty $<x_p,y_p>$ i $<x_k,y_k>$ są punktami zadającymi prostą/:

$$\Delta = [Q] = \left[\left|x_k-x_p\right| \div \left|y_k-y_p\right|\right] ,$$

wtedy błąd ϱ dyskretyzacji w punkcie końcowym odcinka jest równy resztce (res) z całkowitoliczbowego wzajemnego dzielenia wartości przyrostów Δx i Δy określających dyskretyzowany odcinek prostej. Gdy $\Delta = [Q]$, wtedy błąd ϱ dyskretyzacji w punkcie końcowym dyskretyzowanego odcinka jest równy:

$$\varrho = \begin{cases} res & gdy \quad res \leqslant \Delta y/2 \\ \Delta y - res & gdy \quad res > \Delta y/2 \end{cases}$$

W ogólności, gdy przyjmuje się wartość $\Delta = [Q]$, wtedy po to, aby w punktach końcowych dyskretyzowanego odcinka był błąd zerowy, należy kod prostej złożyć z (res) przyrostów równych (Q+1) oraz z(Δy-res) przyrostów Δ równych Q. Otrzymujemy wtedy następującą zależność między parametrami dyskretnego odcinka linii:

$$\Delta x = (Q+1)\cdot(res) + (\Delta y - res)\cdot Q = Q\cdot\Delta y + res .$$

Metzger uważa jednak ten algorytm dyskretyzacji za zbyt złożony i proponuje stosowanie znacznie prostszych jego zdaniem algorytmów warunkowych.

R.Brons w 1974 roku proponuje algorytm tzw. równoległej generacji prostej. Wychodząc z trzech podstawowych stwierdzeń Freemana dotyczących ciągów kodowych reprezentujących linie proste na siatce $G_{8s}(h^2)$, a szczególnie na "jednolitości uszeregowania, jak to tylko możliwe", podaje poniższy rekurencyjny algorytm konstrukcji okresów i podokresów ciągu symboli reprezentujących prostą o nachyleniu p/q, $0<p<q$, p, q - całkowite:

1/ zredukują /gdy to możliwe/ p/q do p_o/q_o; p_o, q_o - liczby względem siebie pierwsze,
2/ oblicz $r_o = q_o - p_o$; r_o jest symbolem zera w okresie, p_o - jest symbolem jedynki,
3/ oblicz $c_i = \max(p_i, r_i)$; nazywaj symbolami C_i symbole występujące c_i razy,
4/ oblicz $m_i = \min(p_i, r_i)$; nazywaj symbolami D_i symbole występujące m_i razy,
5/ gdy $m_i = 1$ generuj okres $C_i^{c_i} D_i$ i zatrzymaj program,
6/ oblicz $n_i = E(c_i/m_i)$,
7/ oblicz $p_{i+1} = c_i - m_i n_i$,
8/ oblicz $r_{i+1} = m_i - p_{i+1}$,
9/ generuj r_{i+1} podokresów $C_i^{n_i} D_i$; nazywaj je B_{i+1},
10/ generuj p_{i+1} podokresów $C_i^{n_{i+1}} D_i$; nazywaj je A_{i+1},
11/ przejdź do kroku 3.

Algorytm powyższy jest podobny do /znanego z teorii liczb/ algorytmu Euklidesa znajdowania największego wspólnego dzielnika dwóch liczb.

Odmienną metodę strukturalnej dyskretyzacji prostej prezentuje Stamopoulos w 1974 r. Do dyskretyzacji prostej stosuje on komórkowy procesor obrazu oraz algorytmy "pocieniania". Zasada zdziałania algorytmu dyskretyzacji opartego na tej metodzie jest następująca: na punktach końcowych dyskretyzowanego odcinka buduje się równoległobok za pomocą linii, poprowadzonych w dwóch sąsiednich /na danej siatce/ kierunkach, ograniczających obustronnie kierunek, określony przez odcinek linii dyskretyzowanej. Tak zbudowany równoległobok rozważa się jako obraz dyskretny, w którym każdy węzeł siatki jest "czarną" komórką tego obrazu. Z kolei stosuje się algorytm "pocieniania", który przekształca "ciemny" obraz płaszczyzny dyskretnej w dyskretną linię /złożoną z węzłów siatki/.

Jakość dyskretyzacji odcinka zależy praktycznie od doskonałości działania algorytmu "pocieniania". Autor podaje dwa różniące się między sobą algorytmy "pocieniania", stosowane w proponowanym algorytmie dyskretyzacji. Pierwszy z nich eliminuje "czarne" punkty, które mają czterech "białych" sąsiadów i czterech "czarnych" /na siatce G_{8s}/. Drugi eliminuje te "czarne" punkty, które mają czterech "białych" sąsiadów oraz trzech lub czterech "czarnych". Obydwa algorytmy są rotacyjne i asymetryczne, działają kolejno w jednym, a następnie w innym kierunku, usuwając selektywnie punkty spełniające warunek "pocieniania".

Algorytm pierwszy generuje linię dyskretną podobną jak algorytmy warunkowe na siatce G_{4s}. Algorytm drugi - podobną jak algorytmy warunkowe na siatce G_{8s}. Obydwie linie dyskretne są jednak mniej doskonałe niż linie generowane przez warunkowe algorytmy dyskretyzacji na siatkach odpowiednio G_{4s} i G_{8s}.

Podobnego typu algorytmy zaproponował Stamopoulos również dla siatki 6-spójnej /heksagonalnej/.

Arcelli i Massarotti, wykorzystując uzyskane wyniki badań teoretycznych nad regularnymi dyskretnymi odcinkami, wykazują w 1978 r., że algorytm prostej, zaproponowany przez Bronsa w 1974 r. rzeczywiście generuje prosty dyskretny odcinek, tj. dyskretyzuje segment prostej, a jej dyskretna postać spełnia tzw. zależność cięciwy.

Proces generacji linii za pomocą algorytmu Bronsa jest równoległy, przez co należy rozumieć, że w każdym stadium generacji dyskretnych odcinków /tworzących generowany segment/ pozostają one niezależne /tj. nie połączone/. Segment otrzymuje się w końcowym stadium generacji.

Arcelli i Massarotti twierdzą, że algorytm Bronsa konstruuje, zarówno regularne, jak nieregularne odcinki. Proponują zatem modyfikację tego algorytmu w taki sposób, aby w każdym stadium generacji otrzymywać jedynie regularne odcinki. Modyfikacja wymaga wprowadzenia funkcji pomocniczych:

$$\delta_i \stackrel{df}{=} \begin{cases} 1/(p_i - r_i) & \text{dla} \quad p_i > r_i \\ -1/(p_i - r_i) & \text{dla} \quad p_i < r_i \end{cases},$$

parametru t_i, a także dodatkowych operacji wewnątrz algorytmu:

3a/ oblicz δ_i;

3b/ gdy i=0, przyjmij $t_i=1$, dla pozostałych i przyjmuj $t_i = t_{i-1} \cdot \delta_i$;

8a/ gdy $r_{i+1} = p_{i+1}$ generuj: podokres $A_{i+1}B_{i+1}$, gdy $t_i=1$ lub podokres $B_{i+1}A_{i+1}$, gdy $t_i=-1$ i zatrzymaj program. Natomiast operacje wyszczególnione w punktach 7/ i 8/ algorytmu Bronsa należy zastąpić poniższymi:

7/ generuj r_{i+1} podokresów $C_1^{n_i}D_1$, gdy $t_i=1$ lub podokresów $D_1C_1^{n_i}$ gdy $t_i=-1$, nazywaj je B_{i+1};

8/ generuj p_{i+1} podokresów $C_1^{n_{i+1}}D_1$ gdy $t_i=1$ lub podokresów $D_1C_1^{n_{i+1}}$ gdy $t_i=-1$, nazywaj je A_{i+1}.

Zmodyfikowany w ten sposób algorytm Bronsa generuje regularne i dyskretne odcinki w każdym stadium generacji.

Arcelli i Massarotti proponują również inną modyfikację algorytmu Bronsa. Twierdzą, że algorytm wyjściowy należy odwrócić i podzielić na dwie części. Część I powinna identyfikować hierarchiczną strukturę generowanej linii oraz generować sekwencje liczb naturalnych $f_0,f_1,\ldots,f_k,g_0,g_1,\ldots,g_k$, które w zapisie łańcucha kodowego są wykładnikami /odpowiednikami wielkości r_{i+1},p_{i+1}/. Gdy tylko znane są kody wyjściowe /"0" i "1" dla pierwszej "ósemki" płaszczyzny/, możliwa jest generacja wymaganej linii. Część II powinna odpowiadać z grubsza krokom 7/ i 8/ algorytmu Bronsa i generować łańcuch kodów w zwartej formie, tzn. w postaci ciągu par: podokres i liczba jego powtórzeń.

Rothstein i Weiman w 1976 r. proponują odmienny od poprzednich algorytm strukturalnej dyskretyzacji prostej, oparty na poniższej zasadzie: prosta przecinająca linie siatkowe przecina każdą parę przeciwległych bądź sąsiednich boków elementarnych pól siatkowych /komórek/. Generując kod takiej prostej /leżącej w I "ósemce"/ wstawiamy "0" na odpowiedniej pozycji kodu wówczas, gdy prosta przecina przeciwległe boki odpowiadającej tej pozycji komórki lub wstawiamy "1", gdy przecina ona sąsiednie boki tej komórki. Innymi słowy: wstawiamy "1" na pozycji j kodu linii, gdy kodowana linia przecina poziomą linię siatki na odcinku między j-tą (j+1)-pionowymi liniami siatki.

Z tej zasady wynika w sposób oczywisty maksymalnie możliwa jednolitość uszeregowania "1" w kodzie linii.

Otrzymany tym sposobem kod linii $y=(p_o/q_o)\cdot x$ jest periodyczny, gdy q_o i p_o są względem siebie pierwsze. Długość okresu jest wtedy q_o, p_o jest liczbą "1" w okresie. Ciąg kq_o,kp_o, $k=1,2,3,\ldots$ wyznacza te węzły siatki, przez które przechodzi zadana linia.

Zasadniczą różnicą między kodem prostej otrzymanym powyższym sposobem i kodem prostej generowanym za pomocą algorytmu Bronsa jest przesunięcie w fazie okresów tych kodów. Cykliczne permutacje grup kodowych w opisywanym kodzie zależą od nachylenia linii. Zależność ta jednak jest dość złożona.

W 1979 r. Coderberg opisał algorytm równoległej generacji linii prostej, w którym problemem centralnym jest jak najbardziej jednorodne uszeregowanie /rozstawienie/ elementów w kodowej reprezentacji linii oraz problem centrowania linii.

Rozstawienie elementów we freemanowskiej reprezentacji linii odbywa się według następującej zasady:

niech

$$C\in[0,2,4,6],\quad D\in[1,3,5,7],\quad |C-D|_{\mathrm{mod}8}=1$$

$$q_o \stackrel{\mathrm{def}}{=} \Delta x-\Delta y,\quad p_o \stackrel{\mathrm{def}}{=} \Delta y,\quad \text{dla } q_o \geqslant p_o$$

Ciąg kodów $C^{q_o}D^{p_o}$ tworzy tzw. 0-aproksymację.

Przyjmując liniową zależność między q_o i p_o:

$$q_o = m_o p_o + n_o$$

i zapisując powyższą zależność w postaci

$$q_o = m_o(p_o - |n_o|) + (m_o + k_o) \cdot |n_o| \quad , \quad k_o = \mathrm{sgn}(n_o),$$

otrzymuje się kodowe /permutacyjne/ grupy pierwotne

$$(C^{m_o} D)^{p_o - |n_o|} \quad (C^{m_o + k} D)^{|n_o|}$$

tworzące tzw. 1-aproksymację; aby dojść do idealnej prostej, grupy te powinny zmieniać się tak często, jak to jest tylko możliwe. W związku z tym kolejny, 1-stopień /poziom/ rozstawiania osiąga się przez poniższe czynności:

$$q_1 \stackrel{\text{def}}{=} p_o - |n_o| \stackrel{\text{def}}{=} m_1 p_1 + n_1$$

$$p_1 \stackrel{\text{def}}{=} |n_o| .$$

Otrzymuje się w ten sposób $p_1 - |n_1|$ dodatkowych grup permutacyjnych postaci

$$(C^{m_o} D)^{m_1} \quad C^{m_o + k_o} D$$

oraz $|n_1|$ grup postaci

$$(C^{m_o} D)^{m_1 + k_1} \quad C^{m_o + k_o} D \ .$$

Ogólna postać grup kodowych reprezentujących linię jest następująca:

$$\left\{ \left[\ldots \left\{ \left[(C^{m_o} D)^{m_1} C^{m_o + k_o} D \right]^{m_2} (\ldots)^{m_1 + k_1} (\ldots) \right\}^{m_3} \left\{ \ldots \right\} \ldots \right]^{m_t} \times \left[\ldots \left\{ \ldots \right\}^{m_3 + k_3} \left\{ \ldots \right\} \ldots \right] \right\}^{p_t} \qquad /2/$$

Parametry i-tego poziomu rozstawienia wyznacza się na podstawie zależności:

$$q_i = m_i p_i + n_i$$

$$p_i - |n_i| \geqslant |n_i|$$

$$q_i = p_{i-1} - |n_{i-1}|$$

$$p_i = n_{i-1}$$

$$k_i = \mathrm{sgn}(n_1).$$

Algorytm kończy działanie, gdy $n_t = 0 . p_t > 1$ oznacza $(p_t - 1)$ przecięć dyskretnej i zadanej linii.

Otrzymana za pomocą tego algorytmu linia dyskretna nie zawsze jest scentrowana /wypośrodkowana/ na linii zadanej, chociaż rozpoczyna się i kończy w punktach zadanych. Idealne wypośrodkowanie powinno doprowadzać linię dyskretną do postaci, w której odpowiadająca jej linia średnia arytmetyczna pokrywa się z linią zadaną.

Coderberg zastosował metodę centrowania wynikającą wprost z hierarchicznego zapisu linii /2/ i polegającą na przestawieniu kolejności występowania poszczególnych członów równania linii /2/. Centrowanie segmentów linii dyskretnej odbywa się na każdym poziomie rozstawiania w sposób poniższy.

Gdy $n_o = 0$, wówczas równanie /2/ ma postać

$$(C^{m_o} D)^{p_o}$$

i rozwiązanie problemu centrowania osiąga się sprowadzając grupę

$C^{m_o} D$ do postaci $C^{m_o/2} D C^{m_o/2}$,

gdy m_o jest parzyste oraz do postaci /równoważnych/

$$C^{(m_o+1)/2} D C^{(m_o-1)/2} \quad \text{lub} \quad C^{(m_o-1)/2} D C^{(m_o+1)/2}$$

gdy m_o jest liczbą nieparzystą.

Gdy $n_o \neq 0$, wówczas, zależnie od parzystości bądź nieparzystości m_o, dzieli się na połowę bądź to m_o bądź $(m_o + k_o)$.

Wprowadzając oznaczenia:

$$C_1 = C^{m_o} D$$

$$D_1 = C^{m_o + k_o} D$$

$$C_2 = C_1^{m_o} D_1$$

$$D_2 = C_1^{m_o + k_o} D_1$$

można nadać regule centrowania linii następującą treść: każdy poziom rozstawienia zawiera C_1 grupy oraz D_1 grupę; gdy m_o jest parzyste, wówczas generacja linii rozpoczyna się w środku grupy C_1; gdy m_o jest nieparzyste - w środku grupy D_1. Na następnym poziomie sytuacja jest następująca: gdy m_o jest nieparzyste i punkt początkowy znajduje się w grupie D_1, wówczas zależnie od parzystości bądź nieparzystości m_1 grupa D_1 umiejscowiona jest w środku grupy C_2 bądź D_2. W obydwu wypadkach po D_1 następuje

$$C_1^{m_1} D_1$$

Gdy m_0 jest parzyste, wówczas są dwa przypadki, zależne od m_1. Gdy m_1 jest nieparzyste, to punkt środkowy jest w C_2 i następuje podział liczby (m_1-1), gdy m_1 jest parzyste - punkt środkowy jest w D_2 i następuje podział liczby (m_1+k_1-1).

Powyższa zasada obowiązuje dla każdego poziomu. Reguła używana na danym poziomie jest określona przez wybór punktu środkowego na poziomie o 2 niższym.

Metoda centrowania zależna jest więc od parzystości m_1 oraz miejsca położenia punktu środkowego na poprzednim poziomie.

Generacja linii opisaną tu metodą realizowana jest więc w trzech poniższych etapach:

- iteracyjne wyliczanie stałych m_i, k_i, p_t,
- permutacja członów równania /2/,
- generacja przyrostów dla kolejnych grup permutacyjnych.

2.2. Dyskretyzacje warunkowe na siatkach G_{4s} i G_{8s}

Dyskretyzacje warunkowe oparte są na założeniu, że na punkty aproksymujące krzywą wybierane są te węzły siatki, które spełniają pewien warunek /zwykle arytmetyczny/. Warunek ten wybiera z reguły węzły najbliższe krzywej w sensie określonej metodyki, np. wybierany jest punkt:

1/ bliższy linii zadanej w sensie odległości liniowej, mierzonej prostopadle do linii bądź wzdłuż osi układu, bądź

2/ leżący po przeciwnej stronie linii zadanej niż poprzednio wybrany punkt, bądź

3/ z najmniejszym modułem wartości funkcji $F(x,y)$, bądź

4/ z modułem wartości funkcji $F(x,y)$ leżącym w pewnym /określonym dla każdej z krzywych/ zakresie /podanie tolerancji/, bądź

5/ w którym pewna funkcja kontrolna jest większa, bądź mniejsza od zera,

itp.

W literaturze opisane są liczne różne algorytmy warunkowej dyskretyzacji krzywych na siatkach G_{4s} i G_{8s}. Dotyczą one z reguły dyskretyzacji prostej bądź też prostej i okręgu, czy krzywych algebraicznych 2-stopnia, a nawet wyższych stopni. Różnią się między sobą postacią warunku decyzyjnego, rodzajem i liczbą stosowanych operacji arytmetycznych oraz szybkością działania i zajętością pamięci operacyjnej. Istotną rolę w algorytmach warunkowych odgrywa postać warunku decyzyjnego, która w zasadzie określa złożoność oraz szybkość działania algorytmu.

Najliczniej i najwcześniej opisane zostały algorytmy dyskretyzacji linii, w których węzeł aproksymujący wybierany jest według zasady minimalizacji jego odległości liniowej /euklidesowej, bądź mierzonej wzdłuż osi układu x lub y/.

Stockton w 1963 r. jako pierwszy opisał aproksymację linii prostej na siatce $G_{8s}(h^2)$, w której węzły aproksymujące są wybierane na zasadzie minimum odległości, mierzonej wzdłuż osi x bądź y od prostej zadanej. W algorytmie zastosował on arytmetykę całkowitoliczbową oraz tylko proste operacje typu sumowań i przesunięć w rejestrze binarnym.

Bresenham w 1965 r. zaprezentował algorytm całkowitoliczbowej aproksymacji prostej oparty na podobnej zasadzie co algorytm Stocktona /minimum odległości węzła od prostej - odległości mierzonej

wzdłuż jednej z osi x bądź y/. Algorytm Bresenhama jest znacznie szybszy /o ok. 50%/ i oszczędniejszy pamięciowo /jest zakodowany zwięźlej - wymaga około półtora raza mniej miejsca w pamięci komputera/. Podobnie jak algorytm Stocktona używa jedynie prostych operacji typu sumowania i porównywania modułów liczb.

Pitteway w 1968 r. rozszerzył algorytm Bresenhama na krzywe 2-stopnia zachowując sposób minimalizacji błędu liniowego, mierzonego wzdłuż osi układu. Rzeczywisty błąd aproksymacji krzywych 2-stopnia w tym algorytmie jest większy od błędu aproksymacji prostej, gdyż w algorytmie Pitteway pomiar odległości węzłów siatki od krzywej został zamieniony pomiarem odległości węzłów od stycznej, wystawionej w bieżącym węźle siatki. Jeśli więc moduł siatki jest dostatecznie mały /w stosunku do krzywizny krzywej/, to i błąd popełniony z powodu powyższej zamiany jest mały.

Algorytm jest jednak złożony. W każdej "ósemce" płaszczyzny obowiązują różne wzory, a zależności sprawdzające przejścia do sąsiednich "ósemek" są skomplikowane, chociaż używane są jedynie proste operacje sumowań i komparacji modułów liczb całkowitych.

Botting i Pitteway w 1968 r. podali zależności rozszerzające algorytm Pitteway na krzywe 3-stopnia. Nie podali jednak żadnych danych na temat złożoności implementacji tego algorytmu. Sądzić jednak należy, że jest ona znacznie większa niż w wypadku algorytmu dla prostej czy krzywych 2-stopnia.

Metzger w 1969 r. zaproponował warunkowo algorytmy dyskretyzacji prostej, okręgu i paraboli /dla każdej z tych linii oddzielny algorytm/, w których punkty dyskretyzujące wybierane są według kryterium minimum odległości liniowej od linii, mierzonej prostopadle do linii. Dla przypadku linii prostej odpowiednie wyrażenia warunkujące wybór punktu są identyczne, jak proponowane przez Bresenhama w 1965 r. Dla okręgu wzór podany przez Metzgera na wartość, której znak określa wybór punktu, jest dość złożony, wymaga potęgowania wartości współrzędnych punktu bieżącego, a poza tym nie jest to wzór ogólny. W każdej z "ósemek" płaszczyzny obowiązuje inny. Dla paraboli obowiązujące wzory są jeszcze bardziej złożone, wymagają pierwiastkowania liczb rzeczywistych i również zależne są od "ósemek" płaszczyzny, w której jest położona dyskretyzowana linia.

W sumie algorytmy proponowane przez Metzgera są złożone, nieuniwersalne, określone nie tylko oddzielnie dla każdej z wymienionych linii, ale również są różne dla każdej z "ósemek" płaszczyzny. Błędy aproksymacji za pomocą tych algorytmów są jednak małe - nie przekraczają $\frac{h}{2}$.

Freeman w 1970 r. dał dwa ogólne schematy aproksymacji /dyskretyzacji/ dowolnych krzywych płaskich na siatkach kwadratowych. Jeden z nich, zwany "schematem dyskretyzacji metodą przecięć siatką" polega na rejestrowaniu punktów przecięcia się krzywej z nałożoną na płaszczyznę siatką i zastępowaniu tych punktów najbliżej położonymi węzłami siatki /najbliżej w sensie liniowych odległości mierzonych wzdłuż osi układu współrzędnych/. Ten schemat dyskretyzacji odpowiada dyskretyzacji na siatce G_{8s}.

Według innego schematu dyskretyzacji krzywych zaproponowanego przez Freemana w 1970 r. zwanego "dyskretyzacją typu kwadratowa skrzynka" i odpowiadającego dyskretyzacji na siatce G_{4s}, określony węzeł siatki wybierany jest na punkt ciągu aproksymującego wówczas, gdy krzywa przechodzi przez kwadrat równy elementarnemu polu siatki /komórce siatki/, umieszczony symetrycznie wokół danego węzła siatki.

Horn w 1976 r. opisał algorytmy dyskretyzacji prostej i okręgu działające na zasadzie przecięć siatką opisanej przez Freemana w 1970 r.

W wypadku prostej Horn twierdzi, że należy wyznaczyć te punkty siatki, dla których wielkość $\delta = px - q(y + \frac{1}{2})$ jest najmniejsza z liczb dodatnich /minimalizacja odległości liniowej wzdłuż jednej z osi układu/. Obliczanie wartości δ $(-q \leqslant \delta \leqslant p)$ jest jednak istotą algorytmu Bresenhama z 1965 r. generacji prostej.

W wypadku okręgu wzór na δ /dla pierwszej "ósemki" płaszczyzny/ ma postać:

$$\delta = (x - \tfrac{1}{2})^2 + y^2 - R^2 \ .$$

Dla okręgu zakres zmian δ jest $-2x \leqslant \delta \leqslant 2y$.

Ten sposób wyznaczania ciągu punktów aproksymujących jest równoważny z wyliczaniem punktów skrajnych dla każdego z poziomych rzędów okręgu według następującego wzoru:

$$x = \left\lfloor (R^2 - y^2)^{\frac{1}{2}} + \tfrac{1}{2} \right\rfloor .$$

Złożoność algorytmów jest duża i spowodowana odmiennością wzorów w każdej z "ósemek" płaszczyzny, na podstawie których wyznacza się współrzędne węzłów aproksymujących.

Badler w 1977 r. zmodyfikował algorytm Horna z 1976 r. do postaci, w której generuje on w stosunkowo prosty i szybki sposób /używając jedynie operacji sumowania i binarnego przesunięcia w rejestrze/ dysk kołowy oraz pole obszaru prostokątnego. Jak twierdzi Badler, algorytm jest wystarczająco szybki do generacji dysku kołowego w czasie rzeczywistym dla regeneracji obrazu na rastrowym monitorze graficznym, wymagającym odnawiania obrazu z częstotliwością 50 Hz.

Kulpa w 1979 r., rozważając dyskretyzację całkowitoliczbową okręgu zaproponowaną przez Horna z 1976 r. wykazał, że przy stosowaniu jako kryterium dyskretyzacji minimum odległości liniowej wzdłuż jednej z osi układu, w dyskretyzacji okręgu mającego parametry całkowitoliczbowe nie występuje dwuznaczność w wyborze punktu - węzła na węzeł ciągu aproksymującego.

Stosunkowo liczbą grupę stanowią algorytmy, w których punkt aproksymujący jest wybierany na podstawie wyniku testowania kombinacji znaków wartości $F(x,y)$, $\Delta F(x,y)$, F_x, F_y według zasady, aby wartość $F(x,y)$ nie wzrastała /wzrastała możliwie najmniej/.

Partridge w 1968 r., w aproksymacji krzywej 2-stopnia używał wartości funkcji $F(x,y)$ jako kryterium odchylenia punktu. Do wyznaczania wartości funkcji w kolejnych punktach ciągu aproksymującego Partridge wykorzystał rozwinięcie funkcji $F(x,y)$ w szereg Taylora. Następny ruch wybierany jest w takim kierunku, aby wartość $F(x,y)$ była możliwie bliska zeru. Kierunek ruchu wzdłuż krzywej nie zmienia się, jeśli wartość funkcji w węźle bieżącym oraz przyrost wartości funkcji, powstający w trakcie przejścia do następnego punktu, mają przeciwne znaki.

Zaproponowany algorytm jest prosty i przejrzysty.

Danielsson w 1970 r. opisał podobny algorytm, w których znak wartości funkcji $F(x,y)$ jest wykorzystywany do określenia kierunku następnego kroku.

W odróżnieniu od podejścia Partridge w podejściu Danielssona przemieszczenia mogą odbywać się jedynie w czterech kierunkach /aproksymacja na siatce G_{4s}/. Kierunek następnego ruchu zależy

od znaków wartości $F(x,y)$ i jej pochodnych cząstkowych F_x i F_y w punkcie bieżącym. Wybieranie ruchu następuje według zasady, aby nie powiększać bezwzględnej wartości $F(x,y)$.

Algorytm generuje mało "gładkie" krzywe i niejednoznacznie zatraca symetrię krzywych.

Bresenham w 1977 r. zaprezentował algorytm całkowitoliczbowej dyskretyzacji okręgu, w którym wybór kolejnego punktu na węzeł ciągu aproksymującego dokonywany jest na podstawie znaków F_i - wartości funkcji, $F(x,y)$ w punkcie $\langle x_i+1,\ y_i-1\rangle$ oraz znaków pewnych wartości δ_i /wzory dotyczą pierwszej "ósemki" płaszczyzny/:

$$\delta_i = \begin{cases} 2F_i + 2y_i - 1 & \text{dla } F_i < 0 \\ 2F_i - 2x_i - 1 & \text{dla } F_i > 0 \end{cases}$$

Algorytm jest jednak dość złożony, gdyż występuje w nim dużo operacji pomocniczych, polegających na przesuwaniu okręgu do środka układu, określeniu początkowej "ósemki" płaszczyzny i śledzeniu przechodzenia punktu bieżącego do kolejnych "ósemek" płaszczyzny.

Bresenham udowodnił jednak, że wybrane za pomocą jego algorytmu węzły na ciąg aproksymujący są wybrane najlepiej według trzech różnych kryteriów dokładności:

- minimum kwadratu odległości od okręgu zadanego,
- minimum modułu wartości funkcji w punkcie - węźle oraz
- minimum odległości liniowej między okręgiem i punktem - węzłem /gdy tylko środek i promień okręgu są całkowitoliczbowe/.

Między tymi kryteriami istnieje, dla przypadku okręgu, następująca zależność.
Wyrażenia na δ i δ' można przepisać w postaci:

$$\delta = 2\left\{(x_i + 1)^2 + (y_i - \tfrac{1}{2})^2 - (R^2 - \tfrac{1}{4})\right\}$$

$$\delta' = 2\left\{(x_i + \tfrac{1}{2})^2 + (y_i - 1)^2 - (R^2 - \tfrac{1}{4})\right\},$$

co jest podwojoną resztą wartości funkcji $F(x,y)$ okręgu o promieniu

$$(R^2 - \tfrac{1}{4})^{\frac{1}{2}}$$

wyznaczoną w punktach

$$\langle x_i+1,\ y_i-\tfrac{1}{2}\rangle \quad \text{i} \quad \langle x_i+\tfrac{1}{2},\ y_i-1\rangle .$$

Formuła decyzyjna ma dla tego przypadku postać:

- jeśli $\delta \leq 0$, to ruch wzdłuż x,
- jeśli $\delta' > 0$, to ruch wzdłuż $-y$.

W przeciwnym razie ruch po przekątnej $\langle x, -y\rangle$.

Algorytm Pitteway z 1968 r., stosując tę samą formułę decyzyjną, używa wartości promienia R, a nie

$$(R^2 - \tfrac{1}{4})^{\frac{1}{2}} .$$

Stosunkowo liczną grupę stanowią algorytmy, w których na węzeł aproksymujący wybierany jest taki węzeł z otoczenia krzywej, dla którego $|F(x,y)|$ jest minimalny.

Jordan i inni w 1973 r. opisali metodę całkowitoliczbowej dyskretyzacji krzywych 2-stopnia na siatce G_{8s}, w której moduł wartości funkcji $F(x,y)$ przyjęty został jako miara oddalenia węzła od krzywej. Kolejnym punktem ciągu aproksymującego jest ten z trzech węzłów sąsiednich, wyznaczonych przez kierunek ruchu oraz kierunki różniące się o $\pm\frac{\pi}{4}$, w którym moduł wartości funkcji ma wartość najmniejszą. W każdym kroku aproksymacji należy więc obliczyć moduły wartości funkcji w trzech sąsiednich węzłach i wybrać na kolejny punkt ciągu aproksymującego węzeł z najmniejszym modułem $F(x,y)$.

McLain w 1974 r., rozważając dyskretyzację krzywych zadanych równaniem $F(x,y) = 0$ na siatce kwadratowej, krzywą w elementarnym polu siatki przybliżył odcinkiem prostej, łączącym rogi tego obszaru w taki sposób, aby zminimalizować bezwzględną wartość $F(x,y)$.

A.Ziemkiewicz w 1975 r. opisał metodę całkowitoliczbowej aproksymacji okręgu na siatce G_{8s} całkowicie podobną do metody Jordana i innych z 1973 r. W tej metodzie aproksymuje się okrąg również tymi węzłami, w których wartość $|F(x,y)|$ jest minimalna. Zagadnienie dyskretyzacji okręgu sprowadził on do wyprowadzenia rekurencyjnej zależności, służącej do wyznaczania wartości błędu w węzłach z otoczenia punktu bieżącego. Obliczanie zmiennych występujących w zależności rekurencyjnej jest jednak złożone, sprowadza się bowiem do działania na macierzach $[3x3]$ oraz uzależnione jest od numeru "ósemki" płaszczyzny, na której położony jest dyskretyzowany okrąg.

A.Ziemkiewicz wykazał również, że aproksymacja okręgu według kryterium $\min|F(x,y)|$ daje mniejszy od $\frac{1}{2}$ błąd dyskretyzacji, tj. $|R^2 - d_i^2| < (R - \frac{1}{4})$, a stąd $|d_i - R| < \frac{1}{2}$ /co jest prawdziwe pod warunkiem, że promień okręgu i jego środek mają współrzędne całkowite/, i jest równoważna kryterium minimum odległości euklidesowej.

W sumie algorytm A.Ziemkiewicza jest mniej atrakcyjny od algorytmu Jordana.

Rubin w 1976 r. zaproponował ulepszenie algorytmu Jordana i in. z 1973 r. twierdząc, że kierunek ruchu wzdłuż krzywej należy wybierać tylko raz na początku algorytmu, a następnie przyjmować ten, w którym został wykonany ostatni ruch. Przy tym założeniu można będzie wyeliminować ograniczenie na funkcję $F(x,y)$, że musi ona być algebraiczna i różniczkowalna. Funkcja ta nie może jednak zbyt szybko zmieniać się, tzn. nie więcej niż o $\frac{\pi}{4}$ w jednostce rastra.

Belser w 1976 r. wysunął wniosek, że algorytm Jordana z 1973 r. nie śledzi poprawnie wszystkich krzywych 2-stopnia i że przyjęcie $|F(x,y)|$ jako miary odległości punktu od krzywej jest dobre jedynie dla prostej i okręgu. Ponadto zarzuca Jordanowi, że nieprecyzyjnie formułuje warunki, przy których aproksymacja jest całkowicie poprawna. Termin "dobrze zachowująca się funkcja" jest terminem nic nie mówiącym. Natomiast algorytm Jordana niepoprawnie dyskretyzuje niektóre krzywe np. krzywą $y = 2x^2$.

Ramot w 1976 r. zaproponował rozszerzenie algorytmu Jordana na krzywe 3-stopnia.

Analizując działanie algorytmu śledzenia krzywej zadanej równaniem $F(x,y) = 0$ na siatce $G_{8s}(h^2)$, Sutcliffe w 1976 r. również zauważył, że kierunek ruchu wzdłuż krzywej należy określić jedynie raz, na początku działania algorytmu. Kierunki kolejnych ruchów wzdłuż krzywej nie powinny różnić się od siebie więcej niż o $\pm\frac{\pi}{4}$. Rozważył również występujące w algorytmie zjawisko wybierania lokalnych minimów funkcji bez sprawdzania jej znaku i zjawisko "ześlizgiwania się z krzywej" oraz jej "degeneracji".

Sugeruje on, aby w trakcie śledzenia krzywej w rejonach, w których może zaistnieć zjawisko "ześlizgiwania się z krzywej" /tj. wyboru węzłów siatki z lokalnym minimum $|F(x,y)|$/ obliczyć wartości funkcji $F(x,y)$ i kontrolować jej znaki w trzech sąsiednich węzłach wyznaczonych przez kierunek ruchu bieżącego i kierunki różniące się od niego o $\pm\frac{\pi}{4}$. Gdy wartość funkcji przyjmuje różne znaki, śledzenie przebiega prawidłowo. Gdy znaki są jednakowe, wówczas należy włączyć do algorytmu śledzenia dodatkowe operacje, realizujące rozwiązywanie równania kwadratowego, otrzymanego z funkcji $F(x,y)$ przez przyrównanie jednej ze współrzędnych do zera i zmiany drugiej w zakresie ± 1. Jest to jednak zabieg bardzo czasochłonny, stąd zalecany tylko w nadzwyczajnych wypadkach. Inną, zalecaną przez Sutcliffe operacją jest zmiana kierunku śledzenia krzywej.

Chung w 1977 r. zanalizował algorytm Jordana z 1973 r. i stwierdził, że jakiekolwiek porównania algorytmów mają sens tylko wtedy, gdy robi się je według jednolitych, ściśle określonych kryteriów. Według niego w aproksymacji np. okręgu takimi kryteriami mogą być:

1/ symetria i jednolitość aproksymacji, co zapewnia estetykę figur i funkcjonalność algorytmów;

2/ dokładność aproksymacji, mierzona miarami:

a/ $|R - d_{ij}|$, d_{ij} - odległość węzła $<x_i,y_j>$ od środka okręgu aproksymowanego

$$/ -\left\lfloor\frac{m-1}{2}\right\rfloor \leqslant x_i,\ y_j \leqslant \left\lfloor\frac{m}{2}\right\rfloor / ,$$

b/ moduł wartości funkcji $F(x_i,y_j)$.

c/ moduł z różnicy: pole okręgu aproksymowanego oraz pole okręgu dyskretnego;

3/ złożoność obliczeniowa, prędkość aproksymacji oraz ekonomia.

Zdaniem autora generacja okręgu za pomocą algorytmu Jordana jest równoważna generacji okręgu metodą łańcucha resztowego, opisaną przez Freemana w 1969 r., minimalizującą odległość liniową węzła od okręgu wyjściowego.

Podobne do algorytmów Jordana są również algorytmy opisane przez autora rozprawy w 1977 r. /algorytmy dyskretyzacji krzywych 2-stopnia na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$/. Podstawowa różnica w stosunku do algorytmu Jordana polega na wprowadzeniu do algorytmu zależności typu $>$, $<$ między pochodnymi I-rzędu funkcji $F(x,y)$ w węźle bieżącym, co umożliwiło wyznaczanie kolejnego węzła ciągu aproksymującego jako jednego z dwóch sąsiednich węzłów, a nie trzech, jak to jest w algorytmie Jordana. Ten zabieg uprościł w pewnym stopniu algorytm oraz przyspieszył go.

We wspomnianej publikacji z 1977 r. przeprowadzono również globalną i lokalną ocenę dokładności dyskretyzacji krzywych 2-stopnia na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$. Zgodnie z tą oceną kres górny błędu szczytowego dyskretyzacji dowolnego okręgu metodą minimalizacji modułu wartości funkcji $F(x,y)$ przy założeniu, że $R_{min} \geqslant h$, nie przekracza h oraz $h\frac{3-\sqrt{3}}{2}$ odpowiednio dla

$$G_{4s}(h^2) \quad \text{i} \quad G_{8s}(h^2) .$$

S. Paszkowski w 1978 r., rozważając dyskretyzację okręgu na siatce $G_{8s}(h^2)$, również wykorzystał moduł wartości funkcji do wyboru węzłów na ciąg aproksymujący. Postawił jednak warunek, aby wykorzystane w algorytmie parametry miały wartości jak najmniejsze i całkowite.

Prezentowany przez niego algorytm jest optymalny w tym sensie, że w swojej podstawowej, cyklicznie powtarzalnej części działa wyłącznie na liczbach całkowitych i na jak najmniejszych wartościach, nie przekraczających wartości promienia okręgu /dokładniej $R+\frac{1}{2}$/. Udowadnia poza tym, że tej granicznej wartości nie da się obniżyć dla dowolnych okręgów /o parametrach całkowitoliczbowych/.

Inny od poprzednich algorytm dyskretyzacji prostej i okręgu na siatce G_{8s} zaproponowali Suenaga i in. w 1979 r. Do wyznaczania kolejnych węzłów aproksymujących wykorzystali oni również wartość modułu wartości funkcji, z tą jednakże różnicą w stosunku do algorytmów wcześniej opisanych, że nie wybierali węzła z najmniejszym modułem $F(x,y)$ lecz węzeł, dla którego $|F(x,y)| \leqslant T$, gdzie T jest pasmem tolerancji dyskretyzacji krzywych, ustalanym indywidualnie dla każdego typu krzywej. Dla prostej $T = T_p = \frac{h}{2}$ /h - moduł siatki/. Jest ona równe wartości funkcji opisującej płaszczyznę $z = ay - bx$ w punkcie, którego rzut na płaszczyznę xy jest węzłem odległym o $\frac{h}{2}$ od krawędzi przecinania się płaszczyzn: z i xy. Dla okręgu $T = T_{ok} = R$ i jest ono równe wartości funkcji opisującej paraboloidę $z = x^2 + y^2 - R^2$ w punkcie, którego rzut na płaszczyznę xy /tworzącej na przecięciu z paraboloidą okrąg/ jest odległy od aproksymowanego okręgu nie więcej niż $\frac{h}{2}$. Autorzy twierdzą również, że wtedy pasmo tolerancji wokół okręgu ma szerokość większą od modułu siatki.

Błędy dyskretyzacji tego algorytmu są identyczne lub niewiele większe od błędów algorytmu Jordana. Niższe niż w algorytmie Jordana są jednak wymagania dotyczące zajętości pamięci. Złożoność obliczeniowa dyskretyzacji, tj. liczba operacji potrzebnych do wyznaczenia jednego węzła, jest znacznie mniejsza niż w algorytmie Jordana, mniejsza niż w innych algorytmach z wyjątkiem algorytmu Danielssona. Zdaniem autorów jedynym mankamentem metody jest konieczność określenia a priori typu generowanej linii oraz określenie wartości T - pasma tolerancji dyskretyzacji.

2.3. Dyskretyzacje strukturalno-warunkowe na siatkach G_{8s}

W dyskretyzacjach strukturalno-warunkowych występują na pewnym etapie realizacji algorytmu operacje typowe dla podejścia strukturalnego, a następnie operacje typowe dla podejścia warunkowego, np. początkowy podział dyskretyzowanego odcinka linii na podokresy z zastosowaniem podejścia strukturalnego, a następnie wybór kolejnych węzłów dyskretyzujących dla jednego z podokresów, z zastosowaniem podejścia warunkowego i proste przeniesienie kolejności wybranych punktów na pozostałe podokresy.

Tego typu dyskretyzacje łączą w sobie zalety obydwu metod: prostotę obliczeniową algorytmów warunkowych oraz przejrzystość struktury linii dyskretnych algorytmów strukturalnych. Dotyczą jednak tylko linii prostych.

Z literatury znanych jest zaledwie kilka publikacji na temat tego typu dyskretyzacji.

Reggiori w 1972 r. opisał algorytm dyskretyzacji prostej, który wykorzystuje, zarówno podejście warunkowe, jak również podejście strukturalne. W metodzie Reggiori podejście strukturalne zastosowane jest jeden raz. W wyniku powstają dwa podokresy. Dla każdego z nich stosowane jest podejście warunkowe. Zastosowana tu metoda dyskretyzacji prowadzi do bardziej wydajnego algorytmu niż metoda warunkowa i może być stosowana bardzo prosto, w odróżnieniu od algorytmu strukturalnej generacji, który jest złożony.

Bongiovanni G. i in. w 1976 r. przedstawili algorytm dyskretyzacji /generacji/ prostej, w którym generacja linii dyskretnej odbywa się w dwóch etapach. W pierwszym etapie określone są /na podstawie podanego przez autorów równania prostej dyskretnej/ punkty-węzły /elementarne ruchy wzdłuż linii siatkowych/, tworzące w strukturze prostej dyskretnej okres. Drugi etap generacji polega na cyklicznym powtarzaniu wyznaczonych w pierwszym etapie ruchów /tworzących okres/, aż do stanu, w którym zostanie osiągnięty punkt końcowy dyskretyzowanego segmentu prostej.

Wygenerowana w ten sposób prosta dyskretna pokrywa się z prostą dyskretną wygenerowaną za pomocą algorytmów warunkowych, dobierających na punkty aproksymujące węzły siatki z minimalną odległością liniową od prostej zadanej.

Kolejność czynności w ostatnim algorytmie jest odwrotna niż w algorytmie Reggiori. Tutaj najpierw stosuje się podejście warunkowe, tj. wyznacza się tworzące okres w strukturze prostej dyskretnej węzły aproksymujące, które są całkowitoliczbowymi rozwiązaniami równania dyskretnego prostej, a następnie stosuje się podejście strukturalne, tzn. znalezioną w pierwszym etapie grupę kodową powtarza się aż do osiągnięcia punktu końcowego segmentu.

2.4. Dyskretyzacje warunkowe na siatkach C_{ko}

Dyskretyzacje na siatkach C_{ks} są stosowane wtedy, gdy nie ma ograniczenia maksymalnej odległości między kolejnymi punktami ciągu aproksymującego. Łączy się je wtedy aproksymantami, zwykle prostoliniowymi /np. w monitorach graficznych ze sprzętowymi generatorami wektorów/, uzyskując bardziej "gładką" łamaną aproksymującą oraz znacznie zmniejszą liczbę punktów aproksymujących. Co się tego, w dyskretyzacji na siatkach C_{ko} wprowadza się warunek uzyskiwania maksymalnie możliwych odległości między sąsiednimi punktami aproksymującymi, przy założonej dokładności aproksymacji.

Tego typu dyskretyzacje, podobnie jak i inne dyskretyzacje całkowitoliczbowe, są mało znane i stosunkowo rzadko w literaturze opisywane.

W pierwszej ze znanych publikacji z 1972 r. Lappalainen i Ojala zaproponowali prostą i szybką oraz dokładną metodę dyskretyzacji okręgu, opracowaną specjalnie do generowania testu-szablonu, pozwalającego rozpoznawać niedoskonałości obrazu telewizyjnego.

Zasada działania tego algorytmu jest następująca: jedna ze współrzędnych /x bądź y/ ciągu punktów dyskretyzujących okrąg zmienia się na każdym kroku dyskretyzacji o jedną jednostkę. Przyrost wzdłuż drugiej ze współrzędnych jest zmienny, ale również jest całkowitoliczbowy. Kwadrat odległości kolejnych punktów ciągu od środka okręgu wzdłuż drugiej ze współrzędnych jest liniową funkcją promienia okręgu oraz numeru punktu ciągu i będzie liczbą całkowitą, gdy podwojony promień okręgu $2R$ będzie liczbą całkowitą. Ma on następującą postać:

$$\delta_{i+1} = \delta_i + 2R + 1 - 2i$$

$$\delta_1 = 0, \quad i = 1(1)2R \ .$$

Powyższy wzór rekurencyjny jest zupełnie prosty i błędów nie kumuluje. Wyznacza się na podstawie δ_i drugiej współrzędnej punktu ciągu, który jest równy pierwiastkowi kwadratowemu z liczby

całkowitej δ_i, autorzy wykonują w sposób bardzo naturalny i ekonomiczny, polegający na generacji sekwencji k^2+k, $k=1,2,3,\ldots$ i porównywaniu każdej wartości $k_i^2 + k_i$ z δ_i. Jako wynik tego porównania jest brane to k_i, dla którego $k_i^2 + k_i \geqslant \delta_i$. Kwadrat wielkości k otrzymywano metodą Brauna za pomocą następujących par rekurencyjnych:

$$k_{i+1}^2 = f_{i+1} = f_i + g_{i+1}, \quad f_1 = 1,$$

$$g_{i+1} = g_i + 2, \quad g_1 = 1$$

Metoda powyższa jest bardzo prosta. W każdej pętli generacyjnej jest wymaganych zaledwie kilka operacji sumowania liczb całkowitych oraz ich porównywania. Z tych też względów jest ona bardzo szybka. Jej dokładność dyskretyzacji jest określona dokładnością wyznaczania pierwiastka kwadratowego z δ_i i jest nie gorsza niż jedna jednostka.

Oryginalną metodę całkowitoliczbowej dyskretyzacji okręgu zaproponował E. Denert w pracy z 1973 r. Punktem wyjścia w tej metodzie jest spostrzeżenie, że istnieje całkowitoliczbowe rozwiązanie równania okręgu $x^2+ y^2 + R^2$. Trzy liczby x_i, y_i, R_i spełniające to równanie są zwane trójką Pitagorasa. Jedna taka trójka daje osiem punktów na okręgu, które łącznie z czterema punktami na obu osiach dają 12 punktów aproksymujących okrąg. Ponieważ jednak trójki Pitagorasa istnieją jedynie dla nielicznych okręgów z promieniem całkowitoliczbowym, a aproksymacja dwunastobokami jest niewystarczająca, zwłaszcza dla większych okręgów, więc w tym wypadku wyznacza się również trójki dla okręgów o promieniach mniejszych. Przez punkty te i środek okręgu prowadzi się linie aż do przecięcia z okręgiem zadanym. Przemieszczając się wzdłuż tych linii według podanego algorytmu, osiąga się dodatkowe punkty w pobliżu okręgu zadanego.

Metoda Denerta jest dość złożona. W szczególności jest złożone wyznaczanie dodatkowych punktów na okręgu. Poza tym podział kątowy dokonywany przez wyznaczone na okręgu punkty jest ekstremalnie nierówny, co jest przyczyną dużych błędów dyskretyzacji /rzędu kilku jednostek/. W sumie metoda ta jest mało atrakcyjna ze względu na jej złożoność i małą dokładność.

Powyższy algorytm Denerta ulepszył M.L.V.Pitteway w publikacji z 1974 r. proponując, aby kolejne punkty aproksymujące okrąg były wybierane według ciągu

$$\left\langle (R-i)^2 \cdot \left[2i^2 R^2 - i^4\right]^{\frac{1}{2}} \right\rangle / \ [\ldots] \ - \text{najbliższa liczba całkowita, } i = 0,1,2,3,\ldots/.$$

Postępując tak otrzymuje się całkowitoliczbową aproksymację okręgu odcinkami linii prostej o mniej więcej jednakowej długości. Liczba odcinków aproksymujących jest proporcjonalna do $\sqrt{R}$.

Ostatni algorytm, w porównaniu z algorytmem Denerta, ma tę przewagę, że punkty aproksymujące okrąg rozłożone są na nim równomiernie. Liczba tych punktów jest /dla większych okręgów/ znacznie mniejsza niż w algorytmie Denerta - przy zachowaniu tej samej dokładności aproksymacji. Złożoność obliczeniowa algorytmu jest taka sama, jak i algorytmu Denerta. W sumie jednak algorytm jest stosunkowo złożony, wymaga bowiem oprócz sumowania również potęgowania oraz pierwiastkowania liczb.

Zasadniczo odmienną metodę od dotychczasowych całkowitoliczbowych metod dyskretyzacji krzywych przedstawił G.M.Chaikin w 1974 r. W tej metodzie niezamkniętą krzywą bez punktów przegięcia zadaje się

określonymi całkowitoliczbowymi punktami P1, P2, P3 i P4 wybranymi w następujący sposób: punkty P1 i P4 są odpowiednio punktem początkowym i końcowym krzywej. Przez punkty te prowadzi się styczne do krzywej L12 i L34; punkty P2 i P3 stanowią punkty przecinania się linii L12 i L34 z linią L23 poprowadzoną w ten sposób, że jest ona styczna do krzywej, a punkt styczności dzieli odcinek L23 na połowy. Algorytm generacji krzywej działa następująco: znajduje się środki odcinków L12 i L23; nowym punktem P4 staje się punkt środkowy odcinka L23 /punkt styczności z krzywą/:

$$P4: = \frac{1}{2}(P2 + P3).$$

Nowym punktem P2 jest środek odcinka L12, a nowym punktem P3 jest środek odcinka wyznaczonego przez poprzedni punkt P2 i punkt styczności linii L23 z krzywą /czyli nowy punkt P4/:

$$P3: = \frac{1}{2}(P2 + P4),$$

$$P2: = \frac{1}{2}(P2 + P1).$$

Przed wyznaczeniem nowych punktów P2, P3 i P4 poprzednie współrzędne punktów P2 i P4 zapamiętuje się w specjalnym stosie. Z kolei znajduje się środki nowo wyznaczonych odcinków L12 i L23 oraz wyznacza się nowe punkty, kolejno, P1, P3, P2, identycznie jak za pierwszym razem. Operację tę powtarza się tak długo, aż odległość pomiędzy sąsiednimi punktami P1, P2, P3 i P4 stanie się nie większa od zadanej. Wówczas to łączy się kolejno ze sobą punkty P1, P2, P3 i P4, a punkt P4 staje się nowym punktem P1. Ostatnio umieszczone w stosie punkty P2 i P4 pobiera się jako aktualne i cykl podziału odcinków, łączenia ich środków oraz sprawdzania odległości pomiędzy sąsiednimi punktami powtarza się, aż do osiągnięcia punktu P4 zadanego pierwotnie.

Algorytm Chaikina jest stosunkowo prosty. Używane są bowiem tylko operacje sumowania liczb całkowitych, dzielenia przez 2 oraz porównywania liczb. Istnieje jednak potrzeba używanie pamięci do tworzenia stosu, której pojemność zależy od rozmiarów generowanej krzywej. Dla pola obrazowania 1024x1024 jednostek są wymagane cztery dziesięciolokacyjowe stosy. Prędkość generacji jest zmienna i ma charakter narastający. Na początku generacji jest wymagana większa liczba podziałów. W trakcie generacji wykorzystuje się te punkty ze stosu, które wyznaczono w poprzednich cyklach generacji. Do wyznaczenia n punktów jest wymaganych n iteracji. Istotną zaletą algorytmu jest to, że daje się on łatwo rozszerzyć na krzywe trójwymiarowe. Autor algorytmu nie podał jednak żadnych danych na temat dokładności generacji krzywej. Sądzić jednak należy, że nie jest ona zbyt wysoka. /błędy rzędu kilku-kilkunastu jednostek/.

G.F.Riesenfeld w 1975 r. ulepszył algorytm Chaikina do postaci, w której może on generować również krzywe wielomianowe. Wykazał ponadto, że krzywe generowane przez ten algorytm są parametrycznymi krzywymi B-sklejanymi 2-stopnia. Pozostaje jednak nie rozwiązany problem dokładności generacji zdefiniowanych w podobny sposób krzywych.

V.V. Athani w 1975 r., a następnie E.A.Barnhaw w 1980 r. rozwinęli algorytm generacji prostej na siatce G_{8n}, wprowadzając 8 dodatkowych kierunków ruchów. Uzyskali w ten sposób algorytm dyskretyzacji prostej na 16-spójnej siatce. Relacje w algorytmie zawierają zależność między znakami

i wielkością ilorazu wartości Δx, Δy – rzutów generowanego odcinka prostej. Odległość między kolejnymi węzłami ciągu aproksymującego może więc przybierać trzy różne wartości: 1, $\sqrt{2}$ oraz $\sqrt{5}$.

W porównaniu ze swoim pierwowzorem algorytm pozwala aproksymować proste za pomocą mniejszej liczby przemieszczeń oraz z mniejszym o połowę maksymalnym odchyleniem $\varrho \leq \frac{1}{4}$.

W publikacji z 1976 r. Horn opisał metodę wielomianowej całkowitoliczbowej aproksymacji okręgu i elipsy, która daje mniej więcej jednakowo odległe punkty na obwodzie /bardzo blisko linii/ i stosuje jedynie stałoprzecinkową arytmetykę. Koncepcja tej metody oparta jest na użyciu równań różniczkowych do opisu okręgu:

$$\frac{dx}{d\alpha} = -y, \quad \frac{dy}{d\alpha} = x, \quad 0 \leq \alpha \leq 2\pi ,$$

i odpowiadających im równań różnicowych:

$$x_{i+1} = x_i - y_i/t , \quad y_{i+1} = y_i + x_i/t , \quad t \text{ – parametr,}$$

na podstawie których generowana jest elipsa, a nie okrąg, zbliżająca się kształtem do okręgu, w miarę wzrostu wartości parametru t.

Horn twierdzi, że zmiana tego układu równań do postaci:

$$x_{i+1} = x_i - y_i/t , \quad y_{i+1} = x_{i+1}/t + y_i$$

i zastosowanie arytmetyki stałoprzecinkowej nie tylko nie pogarsza dokładności aproksymacji lecz na odwrót, dla t z ustalonego zakresu, punkty aproksymujące tezą w pewnym ograniczonym obszarze, jeszcze bliżej upodziewanej elipsy. Operacje algorytmu są rewersyjne, gdyż zachodzi:

$$y_i = y_{i+1} - x_{i+1}/t ,$$

$$x_i = x_{i+1} + y_i/t .$$

a więc pierwszy powtórzony punkt pokrywa się z punktem początkowym.

Niezwykle istotny jest jednak wybór t. Dla małych t aproksymacja elipsy jest całkiem ekscentryczna. Dla dużej liczby kroków zniekształcenie z powodu zaokrągleń jest tak duże, że elipsa staje się ośmiobokiem /przy $t = \max(|x_p|, |y_p|)$, $\langle x_p, y_p \rangle$ – współrzędne punktu początkowego/. Dla dobrze określonych wartości t punkty przylegają całkowicie do rozważanego okręgu.

Dobrą aproksymacją x' i y' są wyrażenia:

$$x' = (x+y)\left(1 - \frac{1}{4t}\right)$$

$$y' = (x-y)\left(1 + \frac{1}{4t}\right) .$$

Jeśli t jest potęgą 2 wymagane są jedynie operacje sumowania i przesuwania w rejestrze.

Autor twierdził również, że wzory powyższe zapewniają generację jednakowo odległych punktów na okręgu.

Tych pozytywnych opinii nie potwierdzili Maxwell i Baker w publikacji z 1979 r. Ich zdaniem uzyskanie dokładności jednej j.r. generacji krzywych 2-stopnia w obszarze 1024x1024 j.r. wymaga zastosowania dokładniejszego wzoru kwadraturowego /z członem 2-rzędu/ oraz dostatecznie małego przyrostu parametru Δt, nie przekraczającego wartości 2^{-5}. Metoda aproksymacji wykorzystująca równania różnicowe kumuluje bowiem błędy już z definicji.

D.C. Sutcliffe w 1976 r. opisał algorytm podobny do algorytmu Athaniego z 1974 r., który do aproksymacji dowolnej krzywej zadanej równaniem $F(x,y) = 0$ stosował prostoliniowe odcinki o długości $\sqrt{2}$ i 2 jednostki, oraz wykorzystał własność funkcji $F(x,y)$, która przyjmuje różne znaki po obu stronach linii. Podobnego typu algorytm opisał również Earnshaw w 1980 r.

Obszar, na którym ma być dyskretyzowana krzywa zadana równaniem $F(x,y) = 0$ dzielony jest na prostokąty /komórki/, których boki są równe podwojonej długości kroku kreślaka. Proces wyznaczania krzywej dyskretnej odbywa się w ten sposób, że znajduje się bok komórki /linię bazową/, który jest przecinany przez krzywą zadaną i do środka tego boku przemieszcza się głowicę kreślaka. Następnie odbywa się śledzenie położenia krzywej, polegające na wyznaczaniu wartości funkcji w obu rogach komórki przeciwległych do linii bazowej i wybranie na następny węzeł aproksymujący środka tego boku komórki, który przecinany jest przez krzywą. Przecięty bok komórki staje się linią bazową /podstawą/ dla następnej komórki poddawanej badaniu i proces powtarza się, aż do przekroczenia granicy rejonu, w którym krzywa jest aproksymowana.

Autor publikacji przeznaczył również sporo miejsca problemowi tzw. zdegenerowanych komórek, tj. takich, w których wszystkie cztery boki są przecinane przez krzywą. Sutcliffe sugeruje, żeby powyższy przypadek rozstrzygać opierając się na zasadzie minimalnej zmiany kierunku przemieszczenia. Polega ona na wybraniu tego boku, którego osiągnięcie wymaga jak najmniejszej zmiany kierunku ruchu /w stosunku do ruchu poprzedniego/.

Odmienny od poprzednich algorytm całkowitoliczbowej aproksymacji krzywych 2-stopnia zaproponował autor niniejszej rozprawy w publikacji z 1977 r. oraz w publikacji z 1978 r. /algorytm aproksymacji na siatce n-spójnej/. Algorytm ten wyznacza ciąg całkowitoliczbowych punktów aproksymujących /dyskretyzujących/ dowolną z krzywych 2-stopnia.

Odległość pomiędzy punktami aproksymującymi nie jest w góry ograniczona. Jedna ze współrzędnych x bądź y kolejnego punktu ciągu różni się od tejże współrzędnej punktu poprzedniego co najwyżej o jedną jednostkę. Druga ze współrzędnych - o całkowitą liczbę jednostek zależną od parametrów geometrycznych krzywej /lokalnej krzywizny/ oraz położenia krzywej względem GLG.

Wyznaczanie kolejnego punktu aproksymującego krzywą odbywa się za pomocą dwóch typów elementarnych ruchów /ruchów do punktów sąsiednich, odległych o jedną lub $\sqrt{2}$ jednostek/: jednego tzw. "ruchu od krzywej" oraz szeregu tzw. "ruchów wzdłuż krzywej". Kierunek następnego elementarnego ruchu nie ulega zmianie, gdy moduł wartości funkcji w osiąganym węźle jest mniejszy niż w punkcie poprzednim. W przeciwnym razie następuje modyfikacja kierunku ruchu /o kąt $\pm\frac{\pi}{2}$ zgodnie z przyjętą zasadą/, aby uzyskać zmniejszenie się modułu wartości funkcji lub przecięcie krzywej.

Ruch "od krzywej" jest zawsze ruchem do sąsiedniego węzła wzdłuż przekątnej /odległego o $\sqrt{2}$ jednostek/. Ruchy "wzdłuż krzywej" są ruchami wzdłuż tej ze współrzędnych x bądź y, która tworzy

z wektorem kierunkowym stycznej, wystawionym w punkcie bieżącym, mniejszy kąt. Ruchy "wzdłuż krzywej" wykonywane są aż do przecięcia krzywej. Na kolejny węzeł aproksymujący wybierany jest ten z dwóch sąsiednich punktów, w którym moduł wartości funkcji jest mniejszy.

Ten ostatni algorytm ma przewagę nad algorytmami poprzednio opisanymi przede wszystkim ze względu na uniwersalność /aproksymuje dowolne krzywe 2-stopnia/ i dokładność /błędy nie większe niż $\sqrt{2}/2$ jednostki/. Zapewnia również jednoznaczność aproksymacji krzywych niezależnie od ich kierunku obiegu. Ma on jednak istotne wady wynikające z założenia, że niezależnie od położenia krzywej względem układu współrzędnych ruch "od krzywej" odbywa się zawsze wzdłuż przekątnej, natomiast ruchy "wzdłuż krzywej" - zawsze wzdłuż kierunku jednej z osi x bądź y. Powoduje to, że łamana aproksymująca odcinek krzywej przebiegający równolegle /w przybliżeniu/ do przekątnej układu współrzędnych jest mało "gładka" i złożona jest z dużej liczby krótkich odcinków.

W publikacji z 1979 r. Ward opisał metodę zgrubnej aproksymacji krzywych, niepodobną do żadnej z metod poprzednio opisanych, a przeznaczoną do obrazowania krzywej zadanej równaniem $F(x,y) = 0$ na urządzeniu znakowym /np. drukarce/.

Koncepcja metody jest następująca: obszar, w którym krzywa ma być wykreślona dzieli się na mniejsze podobszary:

$$x_i \le x \le x_{i+1}, \quad y_j \le y \le y_{j+1}, \quad i=0(1)k, \quad j=0(1)l,$$

mające położenie $<i,j>$ oraz wymiary identyczne jak wymiary pola znaku. Następnie bada się położenie krzywej w kolejnych podobszarach i w tych podobszarach przybliża się ją odpowiednimi znakami graficznymi. Problemem jest wybranie z małego, stałego alfabetu takiego symbolu graficznego, który będąc wykreślony w analizowanym podobszarze, będzie w nim najściślej przylegał do krzywej.

Najprostsze /1-rzędu/ rozwiązanie zagadnienia dopasowania wymaga jedynie sześciu znaków graficznych i próbkowania wartości funkcji F w czterech rogach badanego podobszaru. Gdy badanie funkcji w całym obszarze obrazowania odbywa się według metody rozwinięcia rastrowego, wtedy liczba próbek funkcji w całym obszarze obrazowania wynosi $(W_x+1)(W_y+1)$.

Znak graficzny przybliżający krzywą w danym obszarze jest wybierany na podstawie kombinacji położenia dodatnich i ujemnych znaków wartości funkcji $F(x,y)$ w rogach badanego podobszaru.

Dokładność odwzorowania krzywej opisanym sposobem jest ograniczona. Poza tym pojawiają się lokalne dwuznaczności, np. gdy przecinają się gałęzie krzywej /występują punkty rozgałęzienia krzywej/.

Lepsze dopasowanie znaku do położenia krzywej można uzyskać wykonując próbkowanie funkcji F w większej liczbie punktów obszaru znaku i rozszerzając liczebność zbioru znaków graficznych. Dopasowanie 2-rzędu wymaga próbkowania funkcji F również w środkowych punktach obrzeża pola znaku oraz 24 odpowiednio dobranych znaków graficznych. Znak wartości każdej próbki F określa, które z brzegów pola znaku są przecinane przez krzywą; określane są także punkty skrajne znaku dopasowującego.

W ogóle, dopasowanie r-rzędu wymaga operowania zbiorem $6r^2$ znaków graficznych i próbkowania funkcji F r razy na każdej z czterech granic pola znaku.

Interesujący algorytm dyskretyzacji parametrycznych krzywych giętkich 3-stopnia zaproponowali S.Moss i A.Lindgurd w 1979 r. Algorytm ten, opracowany do wykreślania znaków alfanumerycznych za pomocą kreślaka posługuje się jedynie arytmetyką całkowitoliczbową.

Koncepcja algorytmu jest następująca: kształty znaku alfanumerycznego zadaje się na pomocą niewielkiej liczby k węzłów siatki, tzw. punktów charakterystycznych:

$$P_i = \left\{ P_x(t_i),\ P_y(t_i) \right\} = \langle x_i, y_i \rangle,\quad t_i = (i-1)\Delta t,\quad \Delta t = \text{const.},\quad i = 1,2,3,\ldots,k$$

Przez punkty te prowadzi się parametryczną krzywą giętką P(t) 3-stopnia, nadającą znakom płynnie zmieniające się, przyjemne dla oka kształty. P_i jest wartością funkcji w punkcie charakterystycznym, a Δt jest przyrostem parametru t przy wyznaczaniu punktów na krzywej giętkiej.

Na krzywej giętkiej wyznacza się z kolei punkty:

$$T_j = \left\{ 2\Delta t^3 P_x(j),\ 2\Delta t^3 P_y(j) \right\},\quad j = t_j = 1(1)\ k \cdot \Delta t,$$

które sprowadzone do najbliższych węzłów siatki dają dyskretną formę tej krzywej.

Dyskretyzacja krzywej giętkiej wykonywana jest następująco: wyliczona jest różnica między punktem osiągniętym w poprzednim ruchu i punktem na krzywej giętkiej obliczonym przy przyroście parametru t o stałą wartość Δt. Ruch następuje wówczas, gdy ta różnica przekracza moduł siatki. Jeśli Δt jest duże, wtedy wyznaczyć należy więcej niż jeden punkt na krzywej giętkiej zanim nastąpi ruch do kolejnego punktu. Jeśli Δt jest małe - kolejne punkty na krzywej giętkiej odległe są od siebie o kilka modułów siatki. Górna granica liczby kroków, jakie należy wykonać zanim osiągnie się następny punkt na krzywej giętkiej, określa się wyrażeniem $4d/\Delta t/d$ - maksymalna odległość między sąsiednimi punktami charakterystycznymi/.

Maksymalne odchylenie między łamaną aproksymującą i krzywą giętką nie przekroczy modułu siatki, gdy zachowany zostanie zależność $\Delta t > \sqrt{d}$. Maksymalna liczba kroków pomiędzy kolejnymi węzłami na krzywej giętkiej wyniesie wtedy $4\sqrt{d}$.

Dyskretyzację krzywej giętkiej można więc przeprowadzić za pomocą dwóch różnych schematów:

- schemat pojedynczego kroku, gdy $\Delta t > 4d$ oraz
- schemat linii prostej, gdy $\Delta t > \sqrt{d}$.

Metoda powyższa jest złożona, a poza tym nie zapewnia symetrii krzywym dyskretnym, będącym postaciami symetrycznych krzywych giętkich. Zasługuje jednak na szczególną uwagę, gdyż jest jedyną znaną w literaturze całkowitoliczbową metodą dyskretyzacji parametrycznych krzywych giętkich na dwuwymiarowej siatce, z oszacowanym błędem aproksymacji.

Ogólne podejście do problemu dyskretyzacji krzywych sformułowali H. Freeman i J.A.Saghri w 1970 r. Opisali oni metodę dyskretyzacji według tzw. schematu uogólnionego łańcucha kodowego. Łańcuch kodowy jest ciągiem kodów przemieszczeń do kolejnych punktów-węzłów aproksymujących krzywą. Punkty aproksymujące wyznacza się po nałożeniu na krzywą siatki i wybraniu węzłów tej siatki najściślej przylegających do krzywej. Odległość między węzłami aproksymującymi nie może być większa niż założona.

Dyskretyzacja według schematu uogólnionego łańcucha kodowego realizuje się w dwóch etapach:

- w etapie pierwszym wybiera się /dla pierwszego węzła $\langle x_p, y_p \rangle$, będącego punktem początkowym segmentu aproksymowanego i pewnej dopuszczalnej długości odcinka aproksymującego krzywą/ zbiór szablonów;

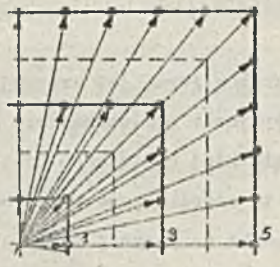
- w etapie drugim wybiera się taki szablon, który w całości zawiera pewien, możliwie najdłuższy odcinek aproksymowanej krzywej. Węzeł siatki, którego szablon został wybrany staje się kolejnym węzłem aproksymującym.

Każdy z dwóch etapów zawiera kilka czynności.

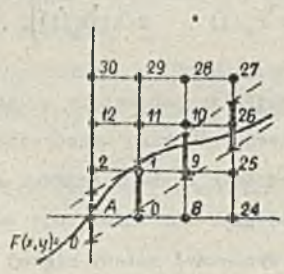
Określenie zbioru szablonów wykonywane jest w następujący sposób:

1/ tworzy się koncentryczne względem węzła $<x_p, y_p>$ kwadratowe pierścienie /rys.2/;

2/ dla każdego węzła na każdym z pierścieni wyznacza się szablon za pomocą dwóch równoległych linii, od punktu $<x_p, y_p>$, przechodzących przez punkty środkowe ogniw pierścienia, przylegających do danego węzła /rys.3/.



Rys. 2. Położenie pierścieni kodowych i wektorów przemieszczeń dla kodu przemieszczeń 1-3-5



Rys 3. Szablon dla pierścienia 3 węzła 26, zawierający krzywą F(x,y) = 0

Wybranie szablonu zawierającego określony odcinek aproksymujący krzywą rozpoczyna się od szablonów zewnętrznego pierścienia i kontynuuje się je dla pierścieni coraz to mniejszych.

Algorytm ten zapewnia błąd aproksymacji nie większy niż $\frac{h}{2}$ i produkuje możliwie najdłuższe z dozwolonych odcinki aproksymujące.

Zaproponowany schemat aproksymacji jest uniwersalny. Obejmuje on bowiem aproksymację dowolnych krzywych analitycznych. Ma jednak podstawową wadę: nie nadaje się do realizacji wprost, a może jedynie służyć jako podstawa do opracowania szczegółowych algorytmów dyskretyzacji, obejmujących dokładniej sprecyzowane klasy krzywych, uwzględniających sposoby ich przedstawiania.

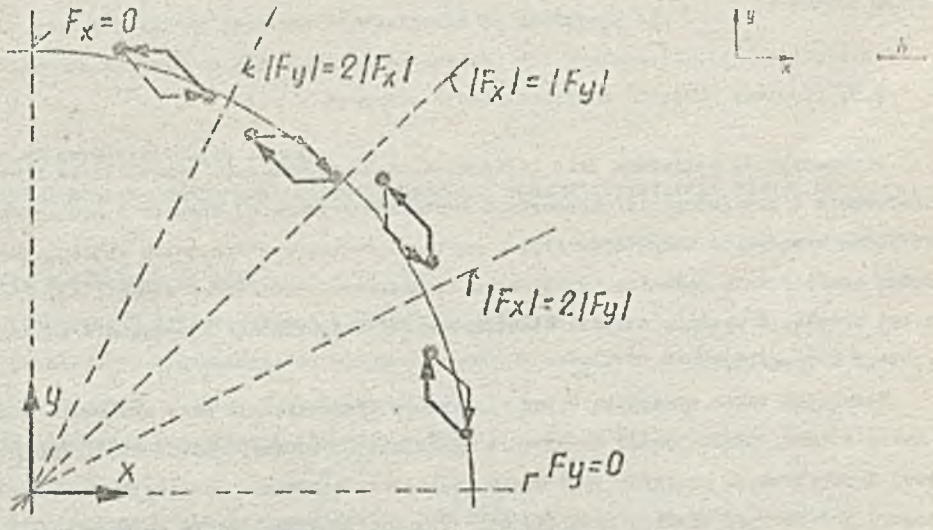
W pracy z 1982 r. autor niniejszej rozprawy zaproponował dwa inne algorytmy dyskretyzacji krzywych, wolne od wad algorytmów wcześniej opisanych oraz bardziej uniwersalne - dotyczą bowiem dyskretyzacji dowolnych algebraicznych krzywych płaskich.

Pierwszy z tych algorytmów w swojej koncepcji działania jest podobny do algorytmu opisanego przez autora w 1978 r. /algorytm aproksymacji na siatce n-spójnej/, jest jednak doskonalszy, chociaż nieco bardziej złożony. Dyskretyzację krzywej sprowadza się w nim do śledzenia przebiegu krzywej i wyznaczenia węzła aproksymującego, gdy krzywa ta jest przecinana przez ruch śledzący, odbywający się według wybranego wcześniej kierunku ruchu wzdłuż linii siatki $G_{OS}(h^2)$. Na węzeł aproksymujący wybierany jest wtedy węzeł najbliższy punktowi przecięcia według kryterium $\min |F(x,y)|$. Śledzenie krzywej odbywa się również za pomocą dwóch typów ruchów: ruchu "od krzywej" oraz sprzężonym z nim ruchów "wzdłuż krzywej" /rys.4/. W odróżnieniu od algorytmu, który opisał autor w 1978 r., każdy z typów ruchów śledzących może odbywać się zarówno wzdłuż GLG, jak 2PLG.

Modyfikacja kierunku ruchu odbywa się natomiast o kąt $\pm\frac{\pi}{4}$ /a nie $\pm\frac{\pi}{2}$ / i dla jednego odcinka łamanej co najwyżej jednokrotnie. Łamana aproksymująca, generowana przez ten algorytm jest bardziej "gładka" niż w algorytmie z 1978 r. Odnosi się to szczególnie do łamanej aproksymującej segmenty położone ukośnie w stosunku do GLG.

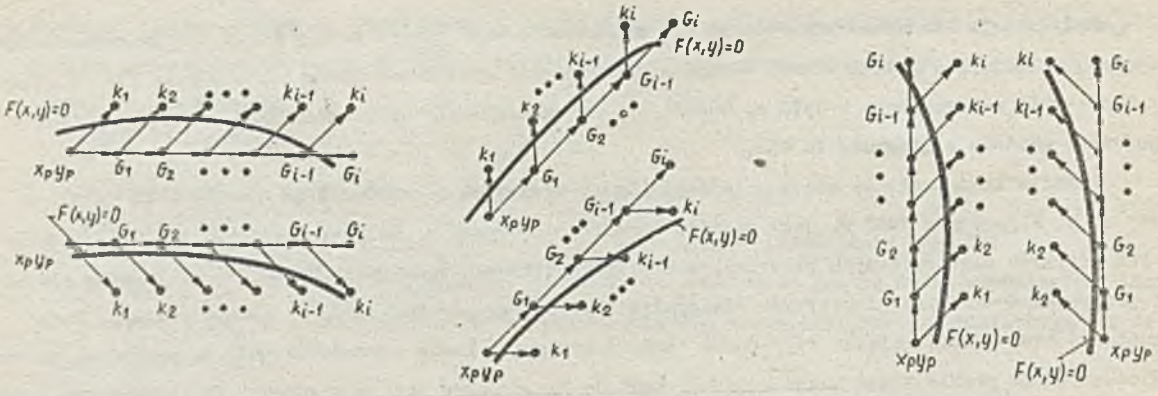
Ponadto łamana składa się z mniejszej liczby dłuższych odcinków. Błąd aproksymacji jest mniejszy od h. Algorytm z 1982 r. jest ponadto bardziej uniwersalny, dotyczy bowiem dyskretyzacji dowolnych krzywych algebraicznych płaskich, a nie tylko krzywych 2-stopnia. Ma on jednak istotną wadę. W przypadku dyskretyzacji krzywych wielogałęziowych, przy pewnym kierunku obiegu krzywej, może nastąpić tzw. "ześlizgnięcie się" ruchu "wzdłuż krzywej", które doprowadza bądź do przejścia ruchów śledzących na przylegającą gałąź krzywej, bądź do izolowanego punktu krzywej i zapętlenia się procesu dyskretyzacji. W takim wypadku dyskretyzację należy przerwać i powtórzyć ją przy zmienionym kierunku obiegu krzywej.

Drugi z zaproponowanych w 1982 r. algorytmów jest zasadniczo różny, z powodu odmiennej koncepcji ruchów śledzących przebieg krzywej. Śledzenie przebiegu krzywej odbywa się za pomocą dwóch ruchów: "głównego" i "kontrolnego", wybieranych początkowo tak, że ich kierunki pokrywając się z liniami siatki $G_{88}(h^2)$ różnią się o kąt $\frac{\pi}{4}$ i prowadzą do węzłów siatki leżących po przeciwnych stronach krzywej /rys.5/.



Rys. 4. Ilustracja ruchów śledzących krzywą wg sposobu I

Ruchy śledzące odbywają się wzdłuż kierunku "ruchu głównego" do sąsiednich węzłów siatki i wykonywane są dopóty, dopóki dwa węzły wyznaczone przez ruchy: "główny" i "kontrolny" znajdą się po tej samej stronie krzywej. Stan, w którym węzły te znajdują się już po tej samej stronie krzywej, oznacza, że śledzona krzywa przekroczyła "pasmo tolerancji" dyskretyzacji i należy wyznaczyć punkt końcowy odcinka aproksymującego. Zostaje nim węzeł siatki najbliższy /w sensie min $|F(x,y)|$ / punktowi przecięcia.



Rys. 3. Ilustracja ruchów śledzących krzywą wg sposobu II

Ostatni algorytm również jest przeznaczony do dyskretyzacji krzywych algebraicznych dowolnego stopnia. Ma on jednak przewagę nad algorytmem poprzednim z powodu niewystępowania tzw. "ześlizgiwania się" ruchu śledzącego krzywą, aproksymuje krzywe nieco mniejszą liczbą, ale dłuższych odcinków i chociaż szczytowe błędy aproksymacji są nieco większe, to jednak nie przekraczają one wielkością modułu siatki h.

2.5. Problemy badawcze w dyskretyzacji krzywych

Dyskretyzacja rozważana jako integralna część zagadnienia wprowadzania i wyprowadzania /również generowania i przetwarzania/ konturów i kształtów /rysunków, obrysów i powierzchni granicznych/ na graficzne urządzenia komputerów /lub w postaci programów sterujących obróbką skrawaniem/ jest rozkładem analitycznie zadanych krzywych na standardowe elementarne części; aby była pożyteczna, powinna być prosta, w wysokim stopniu standaryzowana, uniwersalnie stosowana i powinna prowadzić do prostego kodowania.

Stawiając takie wymagania wobec algorytmów dyskretyzacji oraz wymaganie całkowitoliczbowości, a także i inne, sformułowane we wstępie niniejszej rozprawy, należy stwierdzić, że w całkowitoliczbowej dyskretyzacji krzywych jest wiele problemów, które nie zostały dotychczas w dostatecznym stopniu przebadane, chociaż istnieje duże i pilne zapotrzebowanie na pozytywne wyniki takich badań.

Z zamieszczonego w tym rozdziale przeglądu rozwiązań literaturowych algorytmów wynika następujący stan badań w dziedzinie dyskretyzacji krzywych.

W warunkowej dyskretyzacji krzywych na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$ jest nie przebadana /w zasadzie całkowicie/ dyskretyzacja krzywych algebraicznych stopni wyższych od drugiego, a także dyskretyzacja krzywych innych niż algebraiczne.

W warunkowej dyskretyzacji na siatkach G_{ks} opracowano kilka algorytmów. Jednak każdy z nich obarczony jest dość istotnymi ujemnymi cechami, jak np. operowanie w algorytmach bardzo dużymi liczbami, zależnymi wprost proporcjonalnie od wymiarów geometrycznych i stopnia krzywej. Liczby te,

z powodu ich zakresu zmienności, wymuszają stosowanie /w komputerowej realizacji algorytmów/ zmiennoprzecinkowej reprezentacji liczb, która jest przyczyną powstawania systematycznych błędów obliczeń.

W dyskretyzacji strukturalnej i strukturalno-warunkowej opracowano jedynie algorytmy dyskretyzacji prostej. Dyskretyzacje strukturalna i strukturalno-warunkowa krzywych nie jest sygnalizowana w literaturze w ogóle. Sądzić zatem należy, że jest ona całkowicie nieprzebadana.

Zupełnie nie jest przebadany problem całkowitoliczbowej dyskretyzacji krzywych przestrzennych. Badania tego zagadnienia są szczególnie ważne i pilne, zarówno w grafice komputerowej /zobrazowanie przestrzenne/, jak i w numerycznym sterowaniu obrabiarkami, w których to zagadnieniach istotnym zadaniem jest proste /w sterowaniu obrabiarkami również precyzyjne/ wyznaczanie przestrzennej drogi ruchu /elementu kreślącego obraz lub narzędzia skrawającego/. Autor niniejszej rozprawy zamierza podjąć badania nad całkowitoliczbową dyskretyzacją krzywych przestrzennych.

Wymaga przebadania również zagadnienie całkowitoliczbowej dyskretyzacji powierzchni, które wydają się być istotne nie tylko w grafice komputerowej i numerycznym sterowaniu obrabiarkami, ale w ogóle w komputerowo wspomaganym projektowaniu i wytwarzaniu.

2.6. Problem badawczy rozprawy

Rozważane w dalszej części rozprawy problemy sformułowane są następująco.

Od użytkowników napływają zadania, a każde z nich ma postać, jak w sformułowaniu /podrozdział 3.1/. Zadaniem badawczym jest:

1/ opracowanie metody rozwiązywania takich zadań,

2/ skonstruowanie algorytmu komputerowego realizującego metodę i charakteryzującego się szybkością i prostotą.

Rozwiązaniem zadania badawczego są m.in.:

- metoda przyrostowego "kroczenia" wzdłuż krzywej i śledzenia jej przebiegu,
- dwa nowe algorytmy dyskretyzacji algebraicznych krzywych płaskich dowolnego stopnia na siatkach $G_{4s}(h^2)$ i $G_{8o}(h^2)$,
- ocena złożoności obliczeniowej dyskretyzacji krzywych w funkcji rodzaju siatki, wielkości jej modułu oraz stopnia krzywej,
- oszacowanie szczytowych i polowych błędów aproksymacji dokonywanych za pomocą skonstruowanych algorytmów dyskretyzacyjnych.

3. ALGORYTMY DYSKRETYZACJI KRZYWYCH ALGEBRAICZNYCH NA SIATKACH $G_{4s}(h^2)$ I $G_{8s}(h^2)$

3.1. Sformułowanie zadania dyskretyzacji

Niech w prostokątnym układzie współrzędnych Oxy będzie zadana krzywa algebraiczna K n-stopnia

$$
\begin{aligned}
K:\ F(x,y) = {} & a_{n,0}x^n + a_{n-1,1}x^{n-1}y + \ldots + a_{1,n-1}xy^{n-1} + a_{0,n}y^n + \\
& + a_{n-1,0}x^{n-1} + \ldots + a_{1,n-2}xy^{n-2} + a_{0,n-1}y^{n-1} + \\
& \quad \vdots \qquad\quad \vdots \qquad\quad \vdots \\
& + \ldots + a_{1,n-r}xy^{n-r} + a_{0,1}y^r + \\
& \quad \vdots \qquad\quad \vdots \\
& + a_{1,0}x + a_{0,1}y + \\
& + a_{0,0} = \\
= {} & \sum_{k=0}^{n} \sum_{i=0}^{n-k} a_{n-i-k,i}\, x^{n-i-k} y^i = 0
\end{aligned}
\qquad /3/
$$

/n - liczba naturalna, macierz $[a_{i,j}]$ - rzeczywista i niezerowa/, i niech w tym układzie będzie zadana również kwadratowa siatka $G(h^2)$, określona za pomocą wzorów /1/.

Dla krzywej K i siatki $G(h^2)$ poszukuje się takiej krzywej dyskretnej H - ciągu: $\{<x_p,y_p>, \ldots, <x_i,y_i>, \ldots, <x_k,y_k>\}$ BSWG siatki $G(h^2)$, aby "odległość" między krzywą K i aproksymującą ją krzywą H definiowana jako

$$
\delta_F^H = \max_{p \leq i \leq k} |F(x_i,y_i)| \qquad /4/
$$

była minimalna.

Dodatkowo pożądane jest, aby metoda dyskretyzacji nie powodowała błędów systematycznych.

Ze względu na przewidywane zastosowanie algorytmów /grafika komputerowa, numeryczne sterowanie obrabiarek/ podstawowymi miernikami jakości algorytmu są szybkość i prostota. W grafice komputerowej właśnie te cechy algorytmu określają jego przydatność.

Chung /1977 r./, wypowiadając się na temat wielkości kryterialnych, jakie należy stosować w algorytmach dyskretyzacji przeznaczonych dla grafiki komputerowej wymienił m.in. $|F(x,y)|$. Wydaje się więc, że ze względu na szybkość i prostotę algorytmu dyskretyzacyjnego najwłaściwsze będzie zastosowanie w nim przybliżenia jednostajnego, polegającego na uzyskaniu najmniejszego maksimum "odległości" między krzywymi K i H, a jako miarę "odległości" węzła $<x_i,y_i>$ krzywej H od krzywej K przyjęcie wartości $|F(x_i,y_i)|$. Jest bowiem oczywiste, że jeżeli krzywa dyskretna H będzie w każdym swoim punkcie - węźle $<x_i,y_i>$ spełniać warunek

$$|F(x_i,y_i)| = \min_{<x_j,y_j>\in M} |F(x,y)| \qquad /4a/$$

to będzie spełniać również warunek minimalności /4/.

Z przyjętego założenia o jednostajności przybliżenia wynika w naturalny sposób metoda /opisana i uzasadniona w następnych podrozdziałach/, jaką przyjęto do dyskretyzacji krzywej K. Istota tej metody polega na przyrostowym "kroczeniu" wzdłuż krzywej i śledzeniu jej przebiegu, dokładniej - "kroczeniu" wzdłuż linii siatki dyskretyzacyjnej $G(h^2)$ w kierunku wektora stycznej i wybieraniu tych BSWG, które są "najbliższe" krzywej w sensie $|F(x,y)|$.

Należy podkreślić, że znajomości wartości $F(x_i,y_i)$ wymaga sama metoda śledzenia krzywej. Przyjęcie $|F(x,y)|$ za "metrykę" generacyjną zapewnia dodatkowe wykorzystywanie obliczanej wartości $|F(x,y)|$.

3.2 Analiza zagadnienia

Niech krzywa K będzie w rozważanym obszarze ciągła, za wyjątkiem co najwyżej skończonej liczby punktów /osobliwych, patrz punkt 3.2.4/, i niech $F(x,y)$ ma równocześnie nierówne zeru pochodne cząstkowe 1-rzędu. Określmy kierunek ruchu wzdłuż stycznej do krzywej za pomocą wektorów $\vec{V}^+$ i $\vec{V}^-$, których kierunki będą zgodne z kierunkami ruchów po krzywej; kierunek wektora $\vec{V}^+$ nazywamy kierunkiem dodatnim ruchu po krzywej, natomiast kierunek wektora $\vec{V}^-$ - kierunkiem ujemnym. Wektory te określamy następująco:

$$\vec{V}^+ = \left(-\frac{\partial F}{\partial y}, \frac{\partial F}{\partial x}\right)$$
$$\vec{V}^- = \left(\frac{\partial F}{\partial y}, -\frac{\partial F}{\partial x}\right) . \qquad /5/$$

Wprowadźmy również /dla wygody zapisu/ zmienną logiczną B oznaczającą zwrot kierunku, jak poniżej:

$$B = \begin{cases} 1, \text{ jeśli przyjęto wektor } \vec{V}^+ \\ 0, \text{ jeśli przyjęto wektor } \vec{V}^- \end{cases} \qquad /5a/$$

Przyjmijmy, że punkty $<x_p,y_p>$ i $<x_k,y_k>$ leżą w węzłach siatki: na krzywej lub w najbliższym jej sąsiedztwie. Tak więc węzły te oraz wartość zmiennej B jednoznacznie wyznaczają dyskretyzowany segment krzywej. Należy określić ciągu BSWG: $\{<x_p,y_p>,\ldots,<x_i,y_i>,\ldots,<x_k,y_k>\}$ spełniający podane wcześniej warunki.

Jeden z możliwych sposobów wyznaczania ciągu H aproksymującego krzywą K polega na wyliczaniu, dla każdego węzła z otoczenia krzywej K, wartości funkcji $F(x_i,y_i)$ przez wykonywanie wszystkich operacji potęgowania, mnożenia i dodawania występujących w równaniu /3/. Zakładając warunek bezpośredniej spójności węzłów krzywej H należy zmieniać, poczynając od punktu $<x_p,y_p>$ współrzędne x_i i y_i kolejnych węzłów siatki o moduł h w taki sposób, aby był zachowany założony kierunek

obiegu krzywej określony przez B oraz, aby punkty te leżały w bezpośredniej bliskości /w sensie min $|F(x,y)|$/ dyskretyzowanej krzywej K. Metoda ta wymaga jednak wykonania dużej liczby operacji, zwłaszcza potęgowania dla każdego z węzłów ciągu. Z tego powodu jest mało efektywna. Przeanalizujmy więc metodę wydajniejszą z punktu widzenia złożoności obliczeniowej, wykorzystującą rozwinięcie funkcji $F(x,y)$ w szereg Taylora.

Wprowadźmy operatory kierunkowe funkcji $F(x,y)$:

$$L^x F(x_i,y_i) \stackrel{df}{=} F(x_i+\Delta x_i, y_i) - F(x_i,y_i),$$

$$L^y F(x_i,y_i) \stackrel{df}{=} F(x_i, y_i+\Delta y_i) - F(x_i,y_i),$$

$$L^p F(x_i,y_i) \stackrel{df}{=} F(x_i+\Delta x_i, y_i+\Delta y_i) - F(x_i,y_i),$$

gdzie:

$$|\Delta x_i| = |\Delta y_i| = h .$$

Wprowadźmy również /dla wygody zapisu/ oznaczenia dla pochodnych cząstkowych funkcji $F(x,y)$:

$$F_{rx} \stackrel{df}{=} \frac{\partial^r F(x,y)}{\partial x^r} ,$$

$$F_{sy} \stackrel{df}{=} \frac{\partial^s F(x,y)}{\partial y^s} , \qquad /6/$$

$$F_{rxsy} \stackrel{df}{=} \frac{\partial^{r+s} F(x,y)}{\partial x^r \partial y^s} ,$$

r,s - rząd pochodnych względem zmiennych, odpowiednio x i y .

Korzystając z rozwinięcia w szereg Taylora funkcji $F(x,y)$ w otoczeniu punktu $<x,y>$:

$$F(x+\Delta x, y+\Delta y) = \sum_{i=0}^{n} \frac{1}{i!} \left(\Delta x \frac{\partial}{\partial x} + \Delta y \frac{\partial}{\partial y} \right)^i F(x,y)$$

/n - stopień krzywej/ i uwzględniając wyrażenia /6/, otrzymujemy wzory:

$$L^x F(x,y) = \sum_{i=1}^{n} \frac{(\Delta x)^i}{i!} F_{ix}(x,y) ,$$

$$L^y F(x,y) = \sum_{i=1}^{n} \frac{(\Delta y)^i}{i!} F_{iy}(x,y) , \qquad /7/$$

$$L^p F(x,y) = \sum_{k=1}^{n} \sum_{i=0}^{k} \frac{(\Delta x)^{k-i} (\Delta y)^i}{i!\,(k-i)!} F_{(k-i)xiy}(x,y) .$$

oraz zależność

$$L^{P}F(x,y) = L^{X}F(x,y) + L^{Y}F(x,y) + L^{R}F(x,y) \qquad /8/$$

gdzie:

$$L^{R}F(x,y) = \sum_{k=1}^{n-1} \sum_{i=1}^{k} (\Delta x)^{k-i+1} (\Delta y)^{i} \frac{F_{(k-i+1)x,iy}}{(k-i+1)!\ i!}$$

i jest znacznie prostszy do wyznaczenia niż operator $L^{P}F(x,y)$ /obliczenie operatora $L^{R}F(x,y)$ wymaga mniejszej liczby operacji sumowania niż przy obliczaniu operatora $L^{P}F(x,y)$ o taką liczbę operacji sumowania, jaką należy wykonać w trakcie wyznaczania operatorów $L^{X}F(x,y)$ oraz $L^{Y}F(x,y)$ łącznie/.

Stąd na podstawie wyrażeń /6/ można wyznaczyć wartości funkcji $F(x,y)$ w punktach $\langle x_i+\Delta x_i, y_i\rangle$, $\langle x_i, y_i+\Delta y\rangle$ oraz $\langle x_i+\Delta x_i, y_i+\Delta y_i\rangle$ pod warunkiem, że znane są pochodne cząstkowe rzędu $1+n$ w punkcie $\langle x_i, y_i\rangle$. Te ostatnie można wyznaczyć w podobny sposób, korzystając z rozwinięcia w szereg Taylora pochodnych cząstkowych w punkcie poprzednim oraz z poniższych wzorów, analogicznych do wzorów /6/ i /8/:

$$L^{X}F_{rxsy}(x_{i-1},y_{i-1}) \stackrel{df}{=} F_{rxsy}(x_{i-1}+\Delta x_{i-1}, y_{i-1}) + F_{rxsy}(x_{i-1},y_{i-1}),$$

$$L^{Y}F_{rxsy}(x_{i-1},y_{i-1}) \stackrel{df}{=} F_{rxsy}(x_{i-1}, y_{i-1}+\Delta y_{i-1}) + F_{rxsy}(x_{i-1},y_{i-1}),$$

$$L^{P}F_{rxsy}(x_{i-1},y_{i-1}) \stackrel{df}{=} F_{rxsy}(x_{i-1}+\Delta x_{i-1}, y_{i-1}+\Delta y_{i-1}) + F_{rxsy}(x_{i-1},y_{i-1}),$$

oraz

$$L^{X}F_{rxsy} = \sum_{k=1}^{n-r-s} F_{(r+k)xsy} \frac{(\Delta x)^{k}}{k!},$$

$$L^{Y}F_{rxsy} = \sum_{k=1}^{n-r-s} F_{rx(s+k)y} \frac{(\Delta y)^{k}}{k!}, \qquad /9/$$

$$L^{P}F_{rxsy} = \sum_{k=1}^{n-r-s} \sum_{i=0}^{k} F_{(r+k-i)x(s+i)y} \frac{(\Delta x)^{k-i}(\Delta y)^{i}}{(k-i)!\ i!} \qquad /9'/$$

i należność

$$L^{P}F_{rxsy} = L^{X}F_{rxsy} + L^{Y}F_{rxsy} + L^{R}F_{rxsy},$$

gdzie

$$L^{R}F_{rxsy} = \sum_{k=1}^{n-r-s-1} \sum_{i=0}^{k} F_{(r+k-i+1)x(s+i)y} \frac{(\Delta x)^{k-i+1}(\Delta y)^{i}}{(k-i+1)!\ i!}$$

dla $r+s = 0(1)n-1$, $r,s = 0(1)n-1$.

Wartości pochodnych cząstkowych w punkcie $\langle x_p, y_p \rangle$ wyznacza się, gdy znane są współczynniki równania uwikłanego krzywej $a_{i,j}$ /i,j = 0(1)n/, z następującego wzoru:

$$F_{rxsy} = \sum_{k=0}^{n-r-s} \sum_{i=s}^{n-r-k} \frac{(n-k-i)!\ i!}{(n-k-i-r)!\ (i-s)!}\, a_{n-k-i,i} \cdot x^{n-k-i-r} \cdot y^{i-s} \qquad /10/$$

dla $r,s = 0(1)n$, $r+s = 1(1)n$.

Wzory /10/ można sprowadzić do postaci wygodniejszej w komputerowej realizacji algorytmu. Dla pochodnych względem jednej zmiennej przedstawiają się one następująco:

$$F_{rx} = \sum_{i=0}^{n-r} \sum_{i=0}^{n-r-k} \prod_{p=n-k-r-i+1}^{n-k-i} (p) \cdot a_{n-k-i,i} \cdot x^{n-k-r-i} \cdot y^{i},$$

$$F_{sy} = \sum_{k=0}^{n-s} \sum_{i=s}^{n-k} \prod_{q=i-s+1}^{i} (q) \cdot a_{n-k-i,i} \cdot x^{n-k-i} \cdot y^{i-s} \qquad /11/$$

gdzie: $r,s = 1(1)n$,

a dla pochodnych mieszanych:

$$F_{rxsy} = \sum_{k=0}^{n-r-s} \sum_{i=s}^{n-k-r} \prod_{p=n-k-r-i+1}^{n-k-i} (p) \cdot \prod_{q=i-s+1}^{i} (q) \cdot a_{n-k-i,i} \cdot x^{n-k-r-i} \cdot y^{i-s} \qquad /11a/$$

gdzie: $r,s = 1(1)n$, $r+s = 2(1)n$.

Określimy teraz zależności pozwalające wyeliminować większość z możliwych na danej siatce ruchów i pozostawiające przemieszczenia wzdłuż tych linii siatkowych, których kierunek jest najbardziej zbliżony do kierunku wektora $\vec{V}$. Unikniemy w ten sposób konieczności wyznaczania operatorów funkcji dla większości BSWG. W tym celu należy zlokalizować położenie wektora $\vec{V}$ w jednym z czterech przedziałów o wielkości

$$\frac{\pi}{2} : \left[0, \frac{\pi}{2}\right), \left[\frac{\pi}{2}, \pi\right), \left[\pi, \frac{3}{2}\pi\right), \left[\frac{3}{2}\pi, 2\pi\right)$$

/lokalizacja w przedziale $\frac{\pi}{2}$/ dla ruchów wzdłuż linii siatki $G_{4s}(h^2)$ oraz do jednego z ośmiu przedziałów kątowych o wielkości

$$\frac{\pi}{4} : \left[0, \frac{\pi}{4}\right), \ldots, \left[\frac{7}{4}\pi, 2\pi\right)$$

/lokalizacja w przedziale $\frac{\pi}{4}$/ dla ruchów wzdłuż linii siatki $G_{8s}(h^2)$.

3.2.1. Dyskretyzacja na siatce $G_{4s}(h^2)$

Z przyjętej definicji wektorów $\vec{v}^+$ i $\vec{v}^-$ wynika, że znak przemieszczenia Δx_i z punktu $\langle x_i, y_i \rangle$ do punktu $\langle x_{i+1}, y_{i+1} \rangle$ powinien być przeciwny do znaku pochodnej cząstkowej F_y funkcji $F(x,y)$ w punkcie $\langle x_i, y_i \rangle$, a znak przemieszczenia Δy_i powinien być zgodny ze znakiem pochodnej cząstkowej F_x w tym punkcie. Zależności te pozwalają wyprowadzić wyrażenia określające dwa kierunki ruchów wzdłuż GLG określające przedział kątowy $\frac{\pi}{2}$, w którym zawiera się wektor $\vec{V}$. Kierunki te, określone przez wektory V^x i V^y są więc najbliższe do przyjętego kierunku wektora $\vec{V}$. Określenie kierunków ruchu wzdłuż GLG, między którymi zawiera się wektor $\vec{V}$ nazwiemy lokalizacją 1-stopnia wektora $\vec{V}$ /LV1/.

$$LV1: \left\{ V^x, V^y \right\}. \qquad /12/$$

Składowe tych wektorów są:

$$V^x = V^x(\Delta x, 0), \qquad V^y = V^y(0, \Delta y), \qquad /12a/$$

gdzie:

$$\Delta x = \begin{cases} +h \text{ jeżeli } (F_y \geq 0) \wedge \bar{B} \vee (F_y < 0) \wedge B = 1 \\ -h \text{ w przeciwnym wypadku} \end{cases} \qquad /12b/$$

$$\Delta y = \begin{cases} +h \text{ jeżeli } (F_x \geq 0) \wedge B \vee (F_x < 0) \wedge \bar{B} = 1 \\ -h \text{ w przeciwnym wypadku.} \end{cases}$$

Na siatce $G_{4s}(h^2)$ wyrażenia /12/ eliminują 2 z 4 możliwych na tej siatce kierunków ruchu. Wyznaczenie kolejnego punktu ciągu H wymaga wybrania jednego z dwóch pozostałych węzłów - bliższego krzywej K /w sensie $\min|F(x,y)|$/. W tym celu należy wyznaczyć wartości L^xF i L^yF za pomocą wzorów /7/. Na kolejny węzeł ciągu H należy wybrać zaś ten z punktów $\langle x_i + \Delta x_i, y_i \rangle$ bądź $\langle x_i, y_i + \Delta y_i \rangle$, w którym jest mniejszy $|F(x,y)|$. A więc kolejny punkt ciągu H osiąga się wykonując elementarny ruch:

$$ER1 = \begin{cases} \langle \Delta x, 0 \rangle \text{ jeżeli } |F(x_i + \Delta x, y_i)| < |F(x_i, y_i + \Delta y)| = 1 \\ \langle 0, \Delta y \rangle \text{ jeżeli } |F(x_i, y_i + \Delta y)| < |F(x_i + \Delta x, y_i)| = 1 \end{cases} \qquad /13/$$

Dla przypadku równości modułów:

$$|F(x_i + \Delta x, y_i)| = |F(x_i, y_i + \Delta y)|$$

należy przyjąć dodatkową zasadę, rozstrzygającą o wyborze punktu na kolejny węzeł ciągu. Niech nią będzie położenie punktu po określonej stronie krzywej, np. po tej stronie krzywej, po której zachodzi $F(x_i, y_i) > 0$, co, jak się okaże, jest warunkiem niezbędnym do jednoznaczności dyskretyzacji.

Tak więc mówiąc ogólnie, algorytm wyznacza ten kolejny węzeł, w którym osiąga się minimum bezwzględnej wartości lewej strony w jednym z poniższych wyrażeń:

$$F(x_i + \Delta x_i, y_i) = L^x F(x_i, y_i) + F(x_i, y_i)$$

$$F(x_i, y_i + \Delta y_i) = L^y F(x_i, y_i) + F(x_i, y_i) \qquad /14/$$

dla Δx_i, Δy_i wyznaczonych za pomocą relacji /12/.

Dla uaktualnienia wartości pochodnych cząstkowych w nowo wyznaczonym punkcie można posłużyć się wzorami /9/ i /10/.

3.2.2. Dyskretyzacja na siatce $G_{8s}(h^2)$

W wypadku dyskretyzacji na siatce $G_{8s}(h^2)$ wyrażenia /12/ rozszerzymy o człony $(F_y=0)\wedge(F_x<0)$ i $(F_x=0)\wedge(F_y<0)$, za pomocą których dokładniej lokalizuje się nachylenie wektora $\vec{V}$, gdy jedna z pochodnych cząstkowych 1-rzędu jest równa zeru /a nie jest to punkt przegięcia krzywej/. Ma to istotne znaczenie zwłaszcza w dyskretyzacji krzywych o promieniach krzywizn niewiele większych od h. Relacje /12/ nie zapewniają wówczas wyboru kierunku ruchu doprowadzającego do węzła w najmniejszym modułem $F(x,y)$, jak również powodują uzależnienie kształtu łamanej od kierunku obiegu krzywej określonego przez B.

Lokalizacji wektora $\vec{V}$ w przedziale $\frac{\pi}{2}$ będziemy więc dokonywać w sposób zmodyfikowany - za pomocą LV1'.

$$LV1': \left\{ V^{x'}, V^{y'} \right\} \qquad /15/$$

$$V^{x'} = V^{x'}(\Delta x', 0),$$

$$V^{y'} = V^{y'}(0, \Delta y'),$$

gdzie

$$x' = \begin{cases} +h \text{ jeżeli } (F_y>0)\wedge\bar{B} \vee (F_y<0)\wedge B \vee (F_y=0)\wedge(F_x<0) = 1 \\ -h \text{ w przeciwnym wypadku.} \end{cases}$$

$$y' = \begin{cases} +h \text{ jeżeli } (F_x<0)\wedge\bar{B} \vee (F_x>0)\wedge B \vee (F_x=0)\wedge(F_y<0) = 1 \\ -h \text{ w przeciwnym wypadku.} \end{cases} \qquad /15a/$$

Zawężmy teraz obszar kątowy zawierania się wektora $\vec{V}$ z $\frac{\pi}{2}$ do $\frac{\pi}{4}$, nazywając to lokalizacją 2-stopnia wektora $\vec{V}$ /LV2/. Ograniczające ten obszar kierunki ruchu, jeden wzdłuż linii GLG, drugi - wzdłuż 2PLG, określimy za pomocą wektorów V^x i V^{xx}.

Oczywiste są następujące zależności:

- jeżeli $|F_x| > |F_y|$, to mniejszy z kątów między wektorem $\vec{V}$ i osią X jest mniejszy od $\frac{\pi}{4}$,
- jeżeli $|F_x| < |F_y|$, to mniejszy z kątów między wektorem $\vec{V}$ i osią Y jest mniejszy od $\frac{\pi}{4}$.

Uwzględniając w rozważaniach również kierunek obiegu krzywej określony przez B, lokalizację LV2 definiujemy następująco: LV2 : $\left\{ V^{*}, V^{**} \right\}$, /16/

wyrażenia na składowe wektorów V^{*} i V^{**} są:

$$V^{*} = \begin{cases} V^{*}(\Delta x,0) & \text{jeżeli } \left(|F_x| < |F_y|\right) \vee \left(F_x = F_y\right) \wedge B \vee \left(F_x = -F_y\right) \wedge \bar{B} = 1 \\ V^{*}(0,\Delta y) & \text{jeżeli } \left(|F_x| > |F_y|\right) \vee \left(F_x = -F_y\right) \wedge B \vee \left(F_x = F_y\right) \wedge \bar{B} = 1 \end{cases}$$

$$V^{**} = V^{**}(\Delta x, \Delta y) ,$$

gdzie Δx, Δy są określone za pomocą wzorów /15a/.

Przeprowadzając dyskretyzację na siatce $O_{8s}(h^2)$ zastosujemy kolejno lokalizację LV1' /relacje /15// oraz lokalizację LV2 /relacje /16//.

Lokalizacje te eliminują 6 z 8 możliwych na danej siatce, kierunków ruchu. Następnie zastosujemy kryterium mniejszego modułu funkcji i na kolejny węzeł ciągu aproksymującego wybierzemy ten z pozostałych dwóch punktów, w którym funkcja $F(x,y)$ przyjmuje wartość bezwzględnie mniejszą. Przypadek graniczny równości modułów funkcji, nie uwzględniony w relacjach /16/, rozstrzygniemy na korzyść punktów leżących zawsze po tej samej stronie krzywej, dokładniej, po tej stronie krzywej, dla której $F(x,y) > 0$.

Należy również rozstrzygnąć przypadek graniczny równości funkcji $F(x,y)$ w dwóch punktach leżących po tej samej stronie krzywej. Może on zaistnieć w dyskretyzacji krzywych o bardzo dużych krzywiznach, gdy kąt nachylenia stycznej na odcinku łuku krzywej, odpowiadającym długością modułowi siatki h, zmienia się o wielkość przekraczającą $\frac{\pi}{4}$. Przypadek ten rozstrzygniemy na korzyść ruchu wzdłuż linii 2PLG, jako przybliżającego dłuższy odcinek krzywej.

Wprowadzając oznaczenia dla wyrażeń logicznych:

$$I \stackrel{df}{=} \left(|F_x| > |F_y|\right) \vee \left(F_x = F_y\right) \wedge \bar{B} \vee \left(F_x = -F_y\right) \wedge B,$$

$$N \stackrel{df}{=} \left[|F(x+\Delta x, y+\Delta y)| < |F(x, y+\Delta y)|\right] \vee \left[|F(x+\Delta x, y+\Delta y)| = |F(x, y+\Delta y)|\right] \wedge \left[F(x+\Delta x, y+\Delta y) \geq 0\right] ,$$

/17/

$$U \stackrel{df}{=} \left[|F(x+\Delta x, y+\Delta y)| < |F(x+\Delta x, y)|\right] \vee \left[|F(x+\Delta x, y+\Delta y)| = |F(x+\Delta x, y)|\right] \wedge \left[F(x+\Delta x, y+\Delta y) \geq 0\right] ,$$

całościowy schemat określenia elementarnego ruchu ER2, prowadzącego do kolejnego węzła ciągu H, zapisujemy następująco:

$$ER2 := \begin{cases} <\Delta x, 0> & \text{jeżeli } \bar{I} \wedge \bar{U} = 1 \\ <0, \Delta y> & \text{jeżeli } I \wedge \bar{N} = 1 \\ <\Delta x, \Delta y> & \text{jeżeli } I \wedge N \vee \bar{I} \wedge U = 1 \end{cases} \qquad /18/$$

Wartości funkcji $F(x,y)$ w nowo wyznaczonych węzłach wylicza się na podstawie wzorów /6/ i /8/, wartości pochodnych cząstkowych - ze wzorów /9/ i /10/.

3.2.3. Dane początkowe dyskretyzacji

Przeanalizowane wyżej sposoby dyskretyzacji wymagają, jako danych początkowych, określenia współrzędnych punktu początkowego $<x_p,y_p>$ i końcowego $<x_k,y_k>$ generowanego segmentu krzywej, wartości funkcji w punkcie początkowym $F(x_p,y_p)$, kierunku obiegu krzywej określonego przez B oraz wartości pochodnych cząstkowych rzędu $1 \div n$ w punkcie początkowym. Punkty początkowy i końcowy powinny bezwzględnie pokrywać się z węzłami siatki /powinny mieć współrzędne całkowitoliczbowe/. Innym sposobem zadania krzywej, wygodniejszym z punktu widzenia realizacji komputerowej algorytmów, jest określenie współczynników $a_{i,j}$ równania /3/ krzywej, w miejsce pochodnych cząstkowych i wartości funkcji w węźle początkowym. Wartość $F(x_p,y_p)$, niezbędną do wyznaczenia ciągu aproksymującego, można wtedy wyznaczyć wprost z równania /3/, natomiast wartości pochodnych cząstkowych w węźle początkowym - z równań /10/ ewentualnie /11/.

Węzły początkowy $<x_p,y_p>$ i końcowy $<x_k,y_k>$/a szczególnie ten drugi/ powinny leżeć możliwie najbliżej dyskretyzowanego segmentu krzywej w sensie minimalnej odległości /od krzywej/ mierzonej w metryce generacyjnej, tj. $|F(x,y)|$.

Niezbędny w lokalizacjach LV1' i LV2 kierunek obiegu krzywej określony przez B można zastąpić określeniem $\Delta x_p, \Delta y_p$ - elementarnymi przyrostami wzdłuż linii siatkowych, z węzła początkowego $<x_p,y_p>$ do drugiego punktu ciągu. Na podstawie pierwszego elementarnego ruchu można bowiem określić kierunek obiegu krzywej określony przez B. Takie przyjęcie kierunku ruchu wzdłuż krzywej wydaje się być jednak mniej wygodne, ze względu na konieczność wyznaczenia /poza algorytmem generacyjnym/ wartości funkcji w węzłach siatki przewidywanych na drugi węzeł ciągu aproksymującego.

Długość dyskretyzowanego odcinka krzywej może być również podana kilkoma sposobami, np. przez określenie Δx, Δy - zmian współrzędnych między punktem początkowym i końcowym segmentu, przez określenie punktu końcowego $<x_k,y_k>$ segmentu lub w jeszcze inny sposób. Gdy punkt końcowy dyskretyzowanego segmentu krzywej jest określony za pomocą współrzędnych $<x_k,y_k>$ węzła końcowego, warunek końca dyskretyzacji WKD ma postać:

$$(|x_i - x_k| < h) \wedge (|y_i - y_k| < h) = 1 . \qquad /19/$$

3.2.4. Izolowanie punktów osobliwych krzywej oraz wybór początkowego i końcowego węzła generowanego segmentu

Niezdegenerowane postaci krzywych n-stopnia, dla $n>2$, mogą mieć punkty osobliwe, w szczególności punkty rozgałęzienia ostrza krzywej, punkty samostyczności lub punkty odosobnione. W punktach tych nie jest spełniony warunek regularności krzywej, tj. skończoność pochodnych cząstkowych oraz zależność $F_x^2 + F_y^2 > 0$. Z tego względu w punkcie osobliwym nie można przeprowadzić lokalizacji

/1- i 2-stopnia/ nachylenia wektora $\overrightarrow{V}$. Nie można tym samym określić kolejnego punktu ciągu aproksymującego krzywą. Punkty osobliwe należy więc izolować.

W dyskretyzacji krzywych zawierających punkty osobliwe i leżące blisko siebie różne gałęzie krzywej opisanej jednym równaniem /3/ występują więc następujące ograniczenia:

1/ dyskretyzowany segment krzywej nie może zawierać punktów osobliwych między punktami początkowym i końcowym; punkt początkowy segmentu $<x_p,y_p>$ również nie może być /ze względu na niemożliwość lokalizacji LV1' i LV2 /punktem osobliwym;

2/ na punkty początkowe $<x_p,y_p>$ nie można wybierać węzłów siatki leżących w bezpośredniej bliskości przylegającej do rozważanej gałęzi krzywej /na odległość mniejszą niż h/, tzn. w tych rejonach krzywej, w których przez elementarne pole siatki przechodzi więcej niż jedna gałąź krzywej. Rozpoczynając z takiego węzła dyskretyzację otrzymuje się linię łamaną z defektem, w kształcie ostrej piły, z dużymi zniekształceniami. Może również zostać wygenerowana niewłaściwa gałąź krzywej. Relacje /15/ lokalizacji LV1' wyznaczają w takim punkcie ruch prostopadły do krzywej. Punkt ten jest bowiem punktem lokalnym tych gałęzi krzywej, które przechodzą przez przylegające do niego elementarne pole siatki. Dla krzywej przedstawionej na rys.6 na punkt początkowy nie powinien zostać wybrany żaden z punktów krzywej /w pobliżu krzywej/ bliższy początkowi układu niż punkt P(4,5).

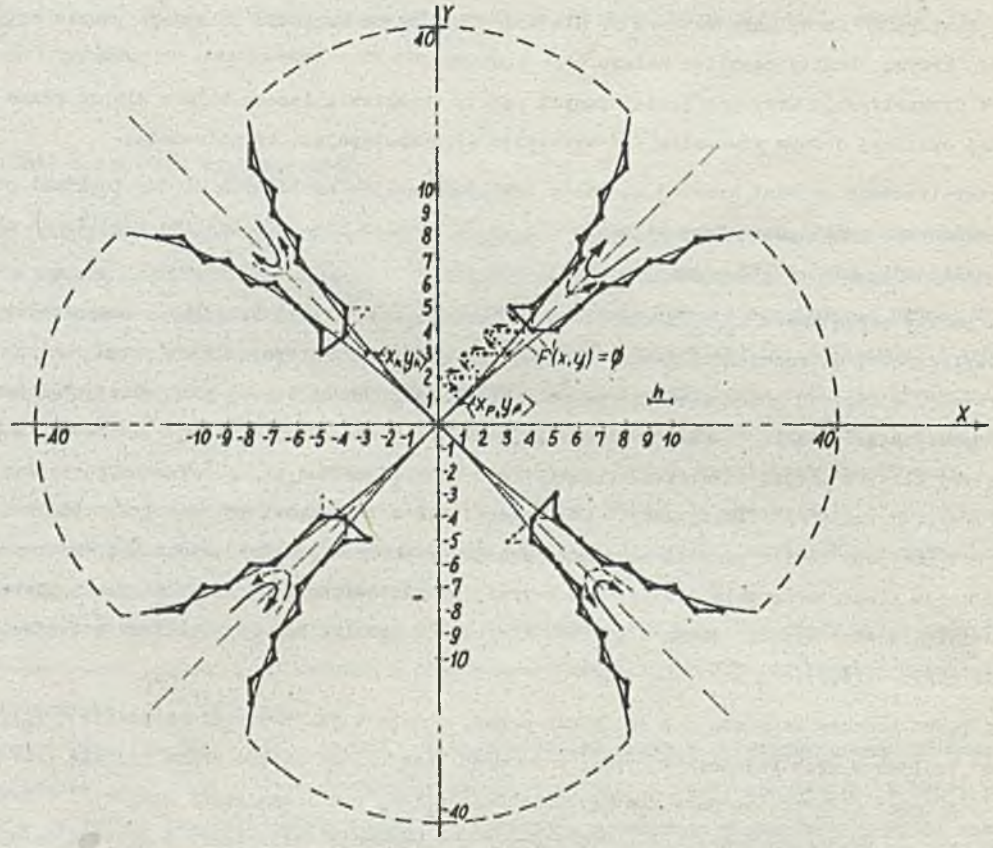
3/ na punkt końcowy segmentu $<x_k,y_k>$ nie można, podobnie jak w przypadku punktu $<x_p,y_p>$, wybierać punktów w tych rejonach krzywej, w których leżą blisko siebie różne gałęzie krzywej /tzn. gdzie przez elementarne pole siatki przechodzi więcej niż jedna gałąź krzywej/. W przedstawionym na rys. 6 przypadku punkt końcowy $<x_k,y_k>$ nie zostanie znaleziony, a generacja zapętli się: nastąpi wygenerowanie wszystkich czterech pętli krzywej, a następnie wielokrotne powtarzanie generacji całej krzywej. Punkty końcowe generowanych segmentów mogą natomiast być punktami osobliwymi. Podstawowym warunkiem jest umieszczenie ich w takim miejscu segmentu, w którym inne gałęzie krzywej są oddalone co najmniej o odległość równą h.

3.3. Algorytmy dyskretyzacji krzywych

W algorytmach dyskretyzacyjnych do obliczania składowych wektora $\overrightarrow{V}$ są wykorzystywane wartości pochodnych cząstkowych funkcji F(x,y) w bieżącym węźle ciągu H, który w ogólności nie leży na krzywej K. Popełniany z tego powodu błąd w określeniu wektora $\overrightarrow{V}$ jest jednak niewielki w stosunku do przedziałów kątowych $\frac{\pi}{4}$ i $\frac{\pi}{2}$, w których jest on lokalizowany. Poza tym dokładne położenie wektora $\overrightarrow{V}$ nie jest niezbędne do wyznaczenia kolejnego punktu ciągu H. Ostatecznym kryterium rozstrzygającym o wyborze węzła $<x_i,y_i>$ na punkt ciągu H jest nim $|F(x_i,y_i)|$.

Przeprowadzone w tym rozdziale rozważania pozwalają skonstruować algorytmy dyskretyzacji krzywych algebraicznych n-stopnia płaskich na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$.

Po zapamiętaniu danych początkowych wyznacza się wartości pochodnych cząstkowych w punkcie $<x_p,y_p>$, a następnie przeprowadza się lokalizację LV1'. Następny krok algorytmu zależy od RSG siatki, na której jest dokonywana dyskretyzacja.

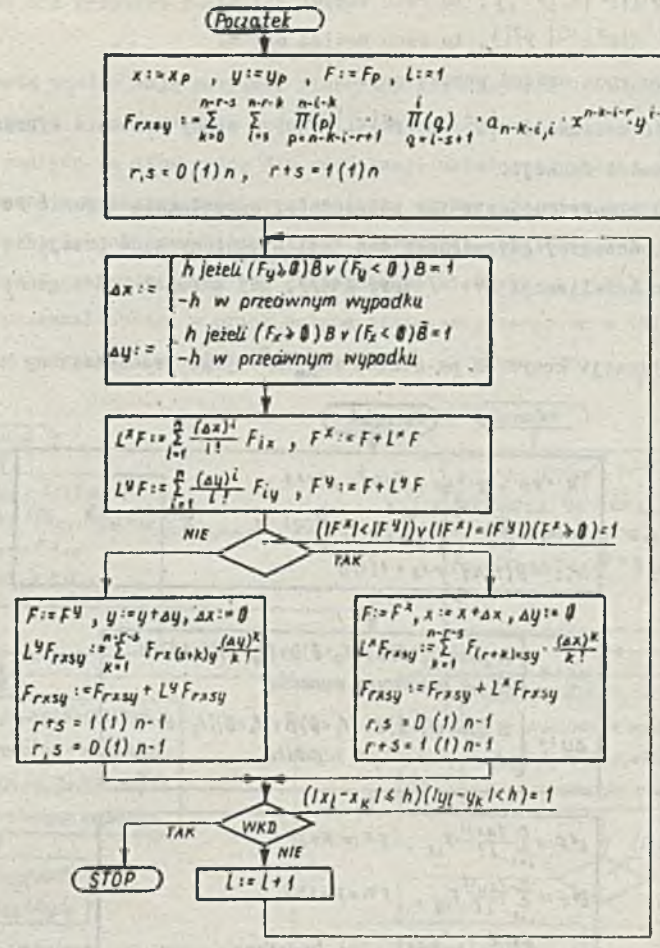


Rys.6. Ilustracja do zasady wyboru punktów: początkowego $<x_p, y_p>$ i końcowego $<x_k, y_k>$ w przypadku aproksymacji wielogałęziowej krzywej algebraicznej /koniczyny czterolistnej/

3.3.1. Algorytm dyskretyzacji na siatce $G_{4s}(h^2)$

Algorytm dyskretyzacji na siatce $G_{4s}(h^2)$ jest przedstawiony na rys.7,

Po lokalizacji LV1' odbywa się wyznaczenie wartości F^x i F_y w węzłach przewidywanych na kolejny punkt ciągu. W tym celu wyznacza się operatory L^xF i L^yF /wzory /8//. Zostaje wybrany - zgodnie z wyrażeniem /13/ na elementarny ruch ER1 - ten węzeł, w którym $|F(x,y)|$ jest mniejszy /lewa bądź prawa gałąź algorytmu/. Końcowymi operacjami algorytmu są: uaktualnienie pochodnych cząstkowych rzędu $1 \div n-1$ dla nowo wybranego punktu /wzory /10// oraz sprawdzenie warunku WKD. Nie spełniony warunek WKD powoduje przejście do lokalizacji LV1' w nowo wyznaczonym punkcie ciągu.



Rys. 7. Algorytm dyskretyzacji krzywej n-stopnia na siatce $G_{4s}(h^2)$

3.3.2. Algorytm dyskretyzacji na siatce $G_{8s}(h^2)$

W wypadku siatki $G_{8s}(h^2)$ następnym krokiem algorytmu /po lokalizacji LV1/, wzory /3.13// jest lokalizacja LV2 /według wyrażeń /16//, po czym następuje wyznaczenie wartości funkcji w dwóch węzłach siatki /wzory /6/ - /8//, przewidywanych do wyboru na kolejny węzeł ciągu. Całościowy schemat określenia elementarnego ruchu ER2 na siatce $G_{8s}(h^2)$ opisany jest za pomocą wyrażeń /17/ i /18/ /wybór lewej, środkowej bądź prawej gałęzi algorytmu/.

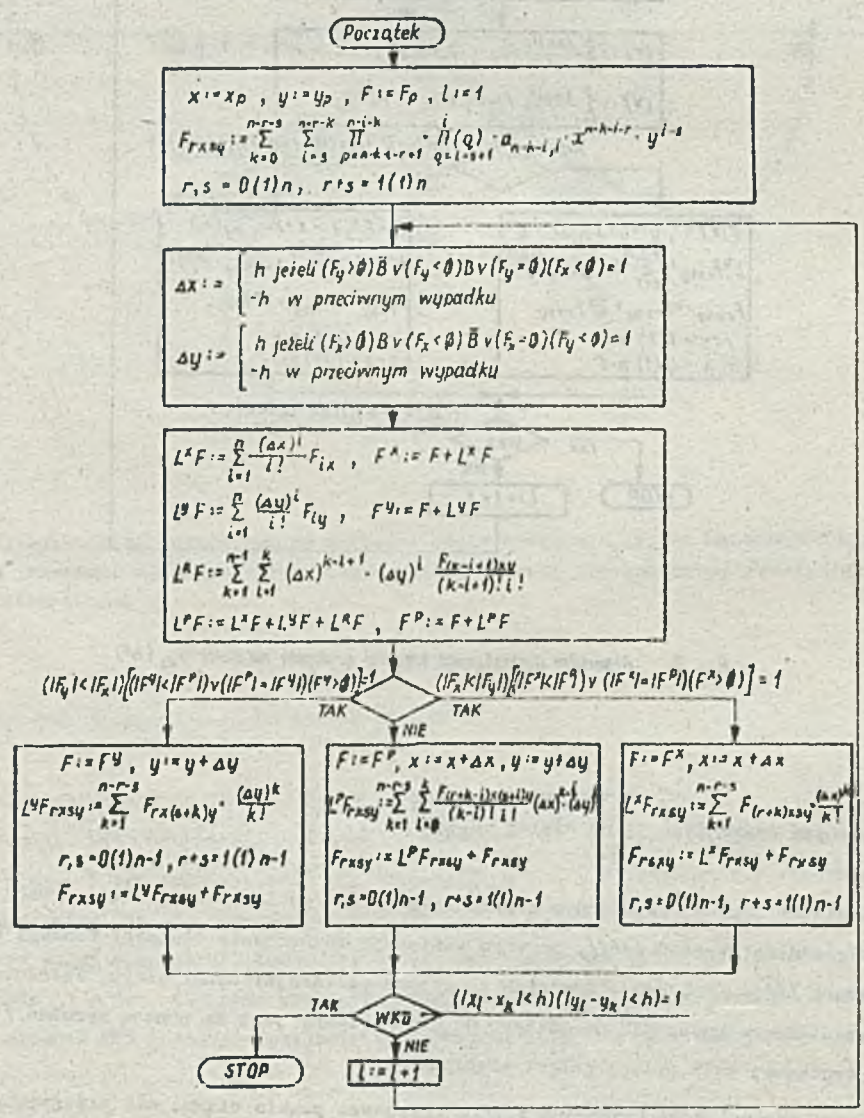
Prostszym od wyrażeń /18/ kryterium wyboru kolejnego punktu ciągu, nie wymagającym przeprowadzania lokalizacji LV2 wektora $\overrightarrow{V}$, jest następujące kryterium:

- jeżeli $(|F^y| < |F^p|) \wedge (|F^y| < |F^x|)$, to ruch wzdłuż osi y,
- jeżeli $(|F^x| < |F^p|) \wedge (|F^x| < |F^y|)$, to ruch wzdłuż osi x, /20/
- w przeciwnym wypadku ruch wzdłuż przekątnej.

Może jednak zajść zależność $|F^x| = |F^y| < |F^p|$, a wtedy zostanie wybrany węzeł siatki z największym modułem wartości funkcji.

Operacją końcową algorytmu jest, jak poprzednio, sprawdzenie warunku końca generacji WKD /relacja /19// i jej zakończenie, gdy warunek ten jest spełniony bądź przejście do początkowych operacji algorytmu, tj. do lokalizacji LV1' /wzory /15//, gdy warunek końca generacji nie zostanie spełniony.

Algorytm dyskretyzacji krzywych na siatce $G_{8s}(h^2)$ jest przedstawiony na rys.8.

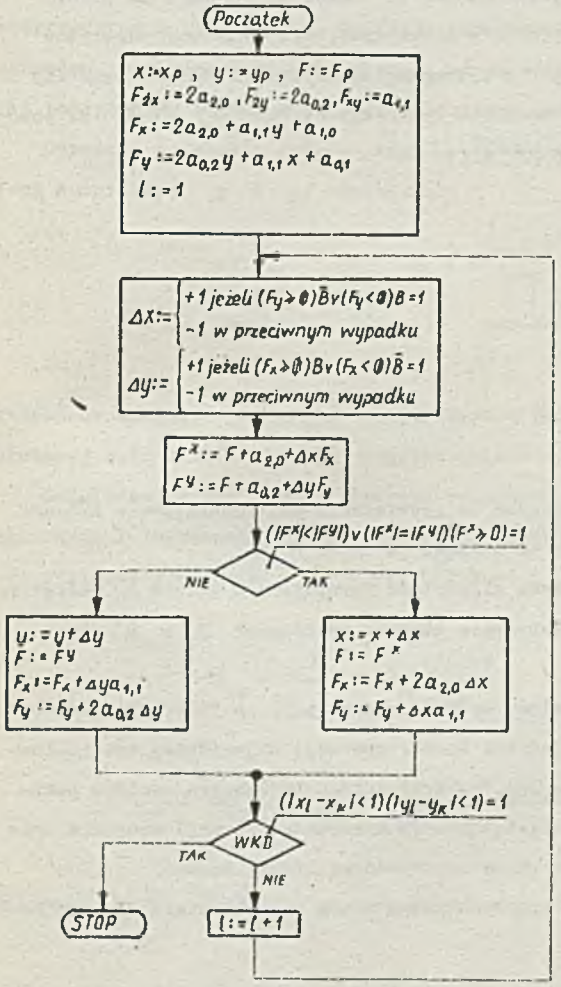


Rys. 8. Algorytm dyskretyzacji krzywej n-stopnia na siatce $G_{8s}(h^2)$

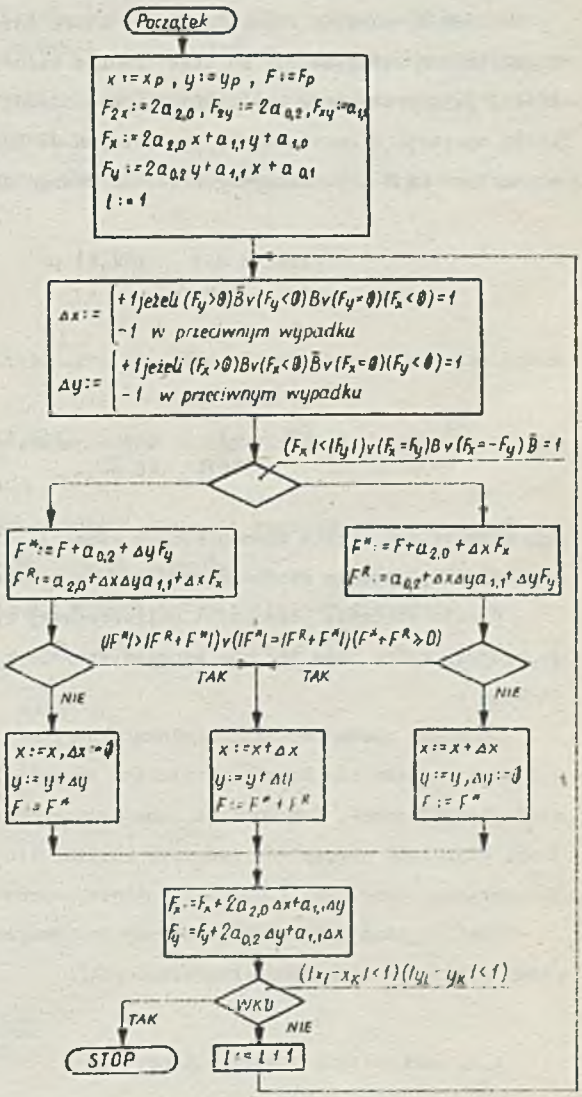
3.3.3. Algorytmy dla krzywych 2-stopnia /n=2/

Szczególnie prostą postać mają powyższe algorytmy dla krzywych 2-stopnia. Zawierają bowiem jedynie operacje sumowania i porównywania modułów liczb oraz proste operacje logiczne na argumentach boolowskich. Z tego względu są atrakcyjne dla realizacji układowej. Układy generatorów są wtedy bardzo proste, a generacja może być realizowana bardzo szybko. Są więc szczególnie przydatne w takich zastosowaniach, jak na przykład monitory i drukarki graficzne.

Algorytmy układowej generacji krzywych 2-stopnia na siatkach $G_{4s}(h^2)$, $G_{8s}(h^2)$ oraz $G_{ks}(h^2)$ zostały szczegółowo przeanalizowane w pracy autora niniejszej rozprawy z 1977 r. W tej rozprawie algorytmy te /dla przypadku h=1/ przedstawione są na rys. 9 i rys. 10.



Rys. 9.
Algorytm dyskretyzacji krzywej 2 stopnia na siatce $G_{4s}(1^2)$



Rys. 10.
Algorytm dyskretyzacji krzywej 2-stopnia na siatce $G_{8s}(1^2)$

4. ANALIZA ZŁOŻONOSCI OBLICZENIOWEJ ALGORYTMÓW

W wielu zastosowaniach algorytmów dyskretyzacji parametrem podstawowym jest ich złożoność obliczeniowa, identyfikowana potocznie z szybkością ich realizacji. Szybkość realizacji algorytmu określa jego przydatność, bowiem spośród różnych algorytmów stosowany jest ten, który dla określonej techniki realizacji /np. komputerowej, układowej bądź programowej/ jest realizowany najszybciej. Ocena szybkości /złożoności obliczeniowej/ realizacji algorytmu jest zagadnieniem równie ważnym jak jego opracowanie. W dalszym ciągu zajmiemy się oceną złożoności obliczeniowej algorytmów przedstawionych w rozdziale 3. Będziemy próbować znaleźć odpowiedź na pytanie: jak szybko można otrzymać rozwiązanie zadania stosując dany algorytm.

Niech $\mathcal{K}$ oznacza klasę krzywych, które można dyskretyzować za pomocą algorytmu A ze zbioru algorytmów $\mathcal{A}$. Zakładamy, że czas trwania każdej operacji w algorytmie, dla dowolnego algorytmu $A \in \mathcal{A}$, jest krotnością czasu trwania określonej operacji elementarnej. Symbolem $L(K,A)$ oznaczmy liczbę operacji elementarnych, potrzebnych do przeprowadzenia dyskretyzacji krzywej $K \in \mathcal{K}$ za pomocą algorytmu $A \in \mathcal{A}$. Poszukujemy więc dla danego algorytmu A wartości

$$Z(A) = \max_{K \in \mathcal{K}} L(K,A) , \qquad /21/$$

która określa jego złożoność obliczeniową. Liczba

$$Z^{*} = \min_{A \in \mathcal{A}} \max_{K \in \mathcal{K}} L(K,A) \qquad /22/$$

tj. najmniejsza liczba elementarnych operacji niezbędnych do rozwiązania najtrudniejszego zadania w klasie $\mathcal{K}$, określa złożoność obliczeniową klasy $\mathcal{K}$ ze względu na zbiór $\mathcal{A}$.

W celu zbadania złożoności obliczeniowej opisanych algorytmów zgrupujemy w klasie $\mathcal{K}$ wszystkie te krzywe, które mają ten sam stopień krzywizny n. Następnie zbadamy zależności Z i Z^{*} jako funkcję n.

Szacując złożoność obliczeniową algorytmów będziemy opierać się na pojęciu "operacji głównej": będziemy oceniać nie liczbą wszystkich operacji, a jedynie liczbą operacji najbardziej czasochłonnych /komputerowo/. Sumowanie i inne proste operacje /np. na argumentach logicznych/ celowo pomijamy. Złożoność algorytmów będziemy oceniać liczbą niezbędnych do wykonania operacji mnożenia oraz potęgowania, dzielenia i operacji silnia, sprowadzonych do odpowiedniej liczby mnożeń.

Analizę złożoności obliczeniowej dyskretyzacji krzywych autor uważa za całkowicie oryginalną /brak podobnych opracowań literaturowych/.

4.1. Całkowitoliczbowość algorytmów

Całkowitoliczbowość algorytmów oznacza możliwość wykonywania w tych algorytmach wszystkich obliczeń w liczbach całkowitych. Pozwala to wyeliminować z komputerowej realizacji algorytmów błędy reprezentacji liczb oraz uzyskać dużą szybkość obliczeń.

Rozważmy algorytmy dyskretyzacji z rozdziału 3. Występują w nich operacje sumowania, mnożenia i dzielenia liczb, a także porównanie i testowanie znaków liczb. Szczególną klasę równań /3/ dyskretyzowanych krzywych są równania o współczynnikach $a_{i,j}$ całkowitych. Wówczas można udowodnić następujące

● Twierdzenie 1

Wszystkie operacje algorytmów dyskretyzacji /opisanych w rozdz.3.1/ można sprowadzić do operacji na liczbach całkowitych, gdy współczynniki $a_{i,j}$ równania /3/ krzywej są liczbami całkowitymi.

Dowód

Operacja podnoszenia do potęgi i operacja silni sprowadzają się do mnożenia liczb. Z przyjętego wcześniej założenia, że moduł siatki h jest liczbą naturalną, wynika całkowitoliczbowość współrzędnych węzłów siatki. Operacje potęgowania $x^{n-k-r-i}$ oraz y^{i-s} /występujące we wzorze na pochodne cząstkowe w punkcie $<x_p,y_p>$/ są więc operacjami potęgowania liczb całkowitych. Wykładniki potęgowe są bowiem z założenia liczbami naturalnymi.

Operacje dzielenia $\frac{i!}{(i-s)!}$ oraz $\frac{(n-k-i)!}{(n-k-i-r)!}$ można sprowadzić do równoważnych operacji mnożenia liczb naturalnych p i q, odpowiednio

$$\prod_{q=i-s+1}^{i}(q) \quad \text{oraz} \quad \prod_{p=n-k-r-i+1}^{n-k-i}(p) \ .$$

Wyznaczone wartości pochodnych cząstkowych w punkcie początkowym są więc liczbami całkowitymi /przy założonej całkowitoliczbowości współczynników równania /3//.

Lokalizacja LV1' wymaga jedynie wykonania kilku prostych operacji logicznych na argumentach logicznych i testowania znaków wartości pochodnych cząstkowych 1-rzędu.

Obliczanie wartości $L^xF(x,y)$, $L^yF(x,y)$ i $L^RF(x,y)$ wymaga wyznaczenia sum z wyrażeń

$$F_{rx}\frac{(\Delta x)^r}{r!} \ , \quad F_{sy}\frac{(\Delta y)^s}{s!} \qquad \text{oraz}$$

$$F_{(r-i+1)xiy}\cdot\frac{(\Delta x)^{r-i+1}(\Delta y)^i}{(r-i+1)!\ \ i!}$$

Ponieważ $|\Delta x| = |\Delta y| = h$, pozostaje wykazać, że ilorazy

$$\frac{F_{rx}}{r!} \ , \quad \frac{F_{sy}}{s!} \quad \text{oraz} \quad \frac{F_{(r-i+1)xiy}}{(r-i+1)!\ \ i!}$$

są liczbami całkowitymi, a wówczas wyznaczenie powyższych wartości będzie wymagać wykonania operacji mnożenia i sumowania liczb całkowitych i da wynik całkowitoliczbowy.

Wyrażenie $\frac{F_{rx}}{r!}$ można przedstawić w postaci:

$$\frac{F_{rx}}{r!} = \sum_{k=0}^{n-r} \sum_{i=0}^{n-k-r} \frac{(n-i-k)!}{(n-i-k-r)!\ r!} \cdot a_{n-i-k,i} \cdot x^{n-i-k-r} \cdot y^{i} = \sum_{k=0}^{n-i} \sum_{i=0}^{n-k-r} \binom{n-i-k}{r} \cdot R_{t,i}(x) ,$$

gdzie: $R_{t,i}(x) = a_{n-i-k,i} \cdot x^{n-i-k-r} \cdot y^{i}$, wskaźnik $t=n-i-k$, $r \leqslant n-i-k$, $t+i = 1(1)n$.

$\binom{n-i-k}{r}$ jako r-elementowa kombinacja z $(n-i-k)$ elementów jest z definicji całkowite.

$\frac{F_{rx}}{r!}$ jest więc liczbą całkowitą, gdy $R_{t,i}(x)$ jest liczbą całkowitą. Podobnie można przedstawić wyrażenie:

$$\frac{F_{sy}}{s!} = \sum_{k=0}^{n-s} \sum_{i=s}^{n-k} \frac{i!}{(i-s)!\ s!} \cdot a_{n-i-k-i} \cdot x^{n-i-k} \cdot y^{i-s} = \sum_{k=0}^{n-s} \sum_{i=s}^{n-k} \binom{i}{s} \cdot R_{t,i}(y)$$

gdzie: $R_{t,i}(y) = a_{n-i-k,i} \cdot x^{n-i-k} \cdot y^{i-s}$, $i \geqslant s$, $t+i = 1(1)n$,

przyjmujące również wartości całkowite. Wyrażenie:

$$\frac{F_{(r-s+1)xsy}}{(r-s+1)!\ s!} = \sum_{k=0}^{n-r+s-1} \sum_{i=s}^{n-i-r+s-1} \frac{(n-i-k)!\ i!}{(r-s+1)!\ s!\ (n-i-k-r+s-1)!} \cdot a_{n-i-k,i} \cdot x^{n-i-k-r+s-1} \cdot y^{i-s} =$$

$$= \sum_{k=0}^{n-r+s-1} \sum_{i=s}^{n-k-r+s-1} \binom{i}{s} \cdot \binom{n-i-k}{r-s+1} \cdot R_{t,i}(x,y) ,$$

gdzie: $R_{t,i}(x,y) = a_{n-i-k,i} \cdot x^{n-i-k-r+s-1} \cdot y^{i-s}$, $i \geqslant s$, $n-i-k \geqslant r-s+1$

/przy całkowitych $a_{n-i-k,i}$/ daje również wartość całkowitą.

Operacje wyznaczenia modułów wartości liczbowych $L^{x}F$, $L^{y}F$, $L^{p}F$ sprowadzają się więc do prostych operacji sumowania liczb całkowitych.

W identyczny sposób wykażemy, że wyniki operacji:

$$\frac{F_{(r+p)xsy}}{p!} , \quad \frac{F_{rx(s+q)y}}{q!} \quad i \quad \frac{F_{(r+p-q)x(s+q)y}}{(p-q)!\ q!}$$

są liczbami całkowitymi.

Wyrażenie pierwsze daje się przedstawić w postaci:

$$\frac{F_{(r+p)xsy}}{p!} = \sum_{k=0}^{n-s-r-p} \sum_{i=s}^{n-k-r-p} \frac{(n-i-k)!\ i!}{(n-i-k-r-p)!\ (i-s)!} \cdot a_{n-i-k,i} \cdot x^{n-i-k-r-p} \cdot y^{i-s} =$$

$$= \sum_{r=0}^{n-s-r-p} \sum_{i=s}^{n-k-r-p} \binom{n-i-k}{r+p} \cdot \binom{i}{s} \cdot \binom{r+p}{p} \cdot r!\ s! \cdot R_{t,i}(x) ,$$

gdzie: $R_{t,i}(x) = a_{n-i-k-i} \cdot x^{n-i-k-r-p} \cdot y^{i-s}$, $n-i-k \geqslant r+p$, $i \geqslant s$,

tj. jako suma iloczynów liczb całkowitych.

Natomiast wyrażenie drugie daje się przedstawić w postaci:

$$\frac{F_{rs(s+q)y}}{q!} = \sum_{k=0}^{n-r-s-q} \sum_{i=s+q}^{n-r-k} \frac{(n-i-k)!\ i!}{q!(n-i-k-r)!\ (i-s-q)!} \cdot a_{n-i-k,i} \cdot x^{n-i-k-r} \cdot y^{i-s-q} =$$

$$= \sum_{k=0}^{n-r-s-q} \sum_{i=s+q}^{n-r-k} \binom{n-i-k}{r} \cdot \binom{i}{s+q} \cdot \binom{s+q}{q} \cdot r!s! \cdot R_{t,i}(y) ,$$

gdzie: $R_{t,i}(y) = a_{n-i-k-i} \cdot x^{n-i-k-r} \cdot y^{i-s-q}$, $n-i-k \geqslant r$, $i \geqslant s+q$,

$t+i = 1(1)n-r-s-q$, $r+s+q \leqslant n$,

jest więc również sumą iloczynów liczb całkowitych.

Wyrażeniu trzecie zaś daje się przedstawić w następującej postaci /$p \geqslant q$, $r \geqslant q$/:

$$\frac{F_{(r+p+q)x\ (s+q)y}}{(p-q)!\ q!} = \sum_{k=0}^{n-r-p-s} \sum_{i=s+q}^{n-k-r-p+q} \frac{(n-i-k)!\ i!}{(p-q)!\ q!(n-i-k-r-p+q)!(i-s-q)!} \cdot$$

$$\cdot a_{n-i-k,i} \cdot x^{n-i-k-r-p+q} \cdot y^{i-s-q} =$$

$$= \sum_{k=0}^{n-r-p-s} \sum_{i=s+q}^{n-k-r-p+q} \binom{r+p-q}{p-q} \cdot \binom{r}{q} \cdot \binom{i}{s+q} \cdot \binom{n-i-k}{r+p-q} \cdot (r-q)! \cdot (s+q)! \cdot R_{t,i}(x,y),$$

gdzie: $R_{t,i}(x,y) = a_{n-i-k,i} \cdot x^{n-i-k-r-p+q} \cdot y^{i-s-q}$.

Tak więc zostało wykazane, że wszystkie z występujących w algorytmach operacje: sumowania, mnożenia, dzielenia, potęgowania oraz operacje silnia są - przy założonej całkowitoliczbowości współczynników $a_{i,j}$ równania /3/ - operacjami na liczbach całkowitych i dają wyniki całkowitoliczbowe. ■

4.2. Złożoność obliczeniowa algorytmów

W literaturze specjalistycznej, omawiającej problematykę złożoności obliczeniowej algorytmów /np.[44]/ rozróżnia się dwie klasy złożoności obliczeniowej algorytmów: wykładniczą i wielomianową. W świetle tej klasyfikacji omawiane w niniejszej rozprawie algorytmy cechuje złożoność wielomianowa.

● Twierdzenie 2

Złożoność obliczeniowa algorytmów dyskretyzacji z rozdz.3 jest złożonością wielomianową, tzn. wzrasta wielomianowo wraz ze wzrostem stopnia dyskretyzowanej krzywej.

Dowód

Dla dowodu wyprowadzimy wzory na złożoność obliczeniową poszczególnych algorytmów dla modułów siatki $h=1$ oraz $h\neq 1$.

4.2.1. Złożoność wyznaczenia jednego węzła ciągu dla $h \neq 1$

Operacje wyznaczania pochodnych cząstkowych w punkcie początkowym $<x_p,y_p>$ wymagają wykonania następujących mnożeń:

● w celu wyznaczenia pochodnych względem jednej zmiennej dla zmiennych x oraz y /według wzoru /11// należy wykonać mnożenia, których liczba określa się wyrażeniem:

$$L_j^o(\sum) = 2\sum_{r=1}^{n}\sum_{k=0}^{n-r}\left[\sum_{i=0}^{n-r-k}(n-k)-1\right] = \frac{n^4}{4}+\frac{5}{6}n^3-\frac{n^2}{4}-\frac{5}{6}n \quad ,$$

● w celu wyznaczenia pochodnych cząstkowych mieszanych - mnożenia, których liczba określa się wyrażeniem:

$$L_m^o(\sum) = \sum_{s=1}^{n-1}\sum_{r=1}^{n-s}\sum_{k=0}^{n-r-s}\left[\sum_{i=s}^{n-r-k}(n-k)-2\right] = \frac{n^5}{30}+\frac{n^4}{12}-\frac{n^3}{3}-\frac{n^2}{12}+\frac{3}{10}n \quad .$$

Wyznaczenie wszystkich pochodnych cząstkowych w punkcie początkowym wymaga więc łącznie wykonania następującej liczby mnożeń, określającej się wielomianem 5-stopnia względem stopnia krzywej:

$$L_\Sigma^o = L_j^o(\sum)+L_m^o(\sum) = \frac{n^5}{30}+\frac{n^4}{3}+\frac{n^3}{2}-\frac{n^2}{3}-\frac{8}{15}n \qquad /23/$$

/dla $n = 2,3,4,5$, $L_\Sigma^o = 8,44,144,784$/.

Operacja wyznaczenia operatora L^RP wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_\xi^o\ (L^RP) = \sum_{k=1}^{n-1}\sum_{i=1}^{k} 2k = \frac{1}{3}(2n^3-3n^2+n) \, .$$

Każda z operacji wyznaczenia operatorów L^xF i L^yF wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_c^o\left(L^xF,\ L^yF\right) = \sum_{k=1}^{n} (2k-1) = n^2 ,$$

co daje razem:

$$L_c^o\left(L^xF + L^yF\right) = 2n^2$$

a łącznie z wyznaczeniem operatora L^RF:

$$L_c^o\left(\sum LF\right) = \frac{2}{3}n^3 + n^2 + \frac{4}{3}n \qquad /24/$$

/dla n=3, $L_c^o\left(\sum LF\right) = 28$/.

Operacje wyznaczenia operatorów L^xF_{rxsy} i L^yF_{rxsy}, dla $r,s = 0(1)n-1$ wymagają wykonania następującej liczby mnożeń:

- operacje wyznaczenia operatorów L^xF_{rx}:

$$L_c^o\left(\sum L^xF_{rx}\right) = \sum_{r=1}^{n-1}\ \sum_{k=1}^{n-r} (2k-2) = \frac{n^3}{3} - \frac{n^2}{2} + \frac{n}{6} ,$$

- tyleż samo mnożeń należy wykonać w trakcie wyznaczenia operatora L^xF_{sy}.

Łączna liczba mnożeń wynosi więc:

$$L_c^o\left(\sum L^xF_{rx} + \sum L^xF_{sy}\right) = \frac{2}{3}n^3 - n^2 + \frac{n}{3} .$$

Każda z operacji wyznaczenia operatorów L^xF_{rxsy} i L^yF_{rxsy}, dla $r,s > 0$, wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_c^o\left(L^xF_{rxsy},\ L^yF_{rxsy}\right) = \sum_{r=1}^{n-2}\ \sum_{s=1}^{n-r-1}\ \sum_{k=1}^{n-r-s} (2k-1) = \frac{n^4}{12} - \frac{n^3}{3} + \frac{5}{12}n^2 - \frac{n}{6} .$$

Stąd wyznaczenie operatorów L^xF_{rxsy} i L^yF_{rxsy} wszystkich pochodnych cząstkowych /r,s=0(1)n-1/ wymaga wykonania łącznie mnożeń:

$$L_c^o\left(\sum L^xF_{rxsy}, \sum L^yF_{rxsy}\right) = \frac{1}{12}\left(n^4 + 4n^3 - 7n^2 + 2n\right) \qquad /25/$$

/dla n=3, $L_c^o\left(\sum L^xF_{rxsy},\ \sum L^yF_{rxsy}\right) = 11$/.

Wyznaczenie operatora $L^{p}F_{rxsy}$ wymaga wykonania mnożeń w liczbie:

- dla pochodnych cząstkowych względem jednej zmiennej:

$$L_c^o\left(\Sigma L^{p}F_{rx},\ \Sigma L^{p}F_{sy}\right) = \sum_{r=1}^{n-1}\cdot\sum_{k=1}^{n-r}\left[\sum_{i=0}^{k} 2(k-1)+2\right] = \frac{n^4-n^2}{6} \qquad /$$

dla każdej ze zmiennych x i y /dla n=3, $L_c^o\left(\Sigma L^{p}F_{rx},\ \Sigma L^{p}F_{sy}\right) = 12$/,

- dla pochodnych cząstkowych względem obydwu zmiennych:

$$L_c^o\left(\Sigma L^{p}F_{rxsy}\right) = \sum_{r=1}^{n-2}\ \sum_{s=1}^{n-r-1}\ \sum_{k=1}^{n-r-s}\left[\sum_{i=0}^{k} 2(k-1)+2\right] = \frac{n^5}{30}+\frac{n^4}{12}+\frac{n^2}{12}-\frac{n}{5}\ .$$

Łączna liczba niezbędnych do wykonania mnożeń, w trakcie obliczenia operatora $L^{p}F_{rxsy}$, każdej z pochodnych cząstkowych jest równa:

$$L_c^o\left(\Sigma L^{p}F_{rxsy}\right) = \frac{n^5}{30}+\frac{n^4}{4}-\frac{n^2}{4}-\frac{n}{30} \qquad /26/$$

/dla n=3 $L_c^o\left(\Sigma L^{p}F_{rxsy}\right) = 26$/.

W pętli algorytmu są więc wykonywane mnożenia:

$$L_c^p\left(P_n^{\ 1}\right) = L_c^o(\Sigma LP)+L_c^o\left(\Sigma L^{x}F_{rxsy},\ \Sigma L^{y}F_{rxsy}\right) = \frac{n^4}{12}+n^3+\frac{5}{12}n^2+\frac{n}{2} \qquad /27/$$

/dla n=2,3,4,6 , $L_c^p\left(P_n^{\ 1}\right) = 12, 39, 94, 342$/,

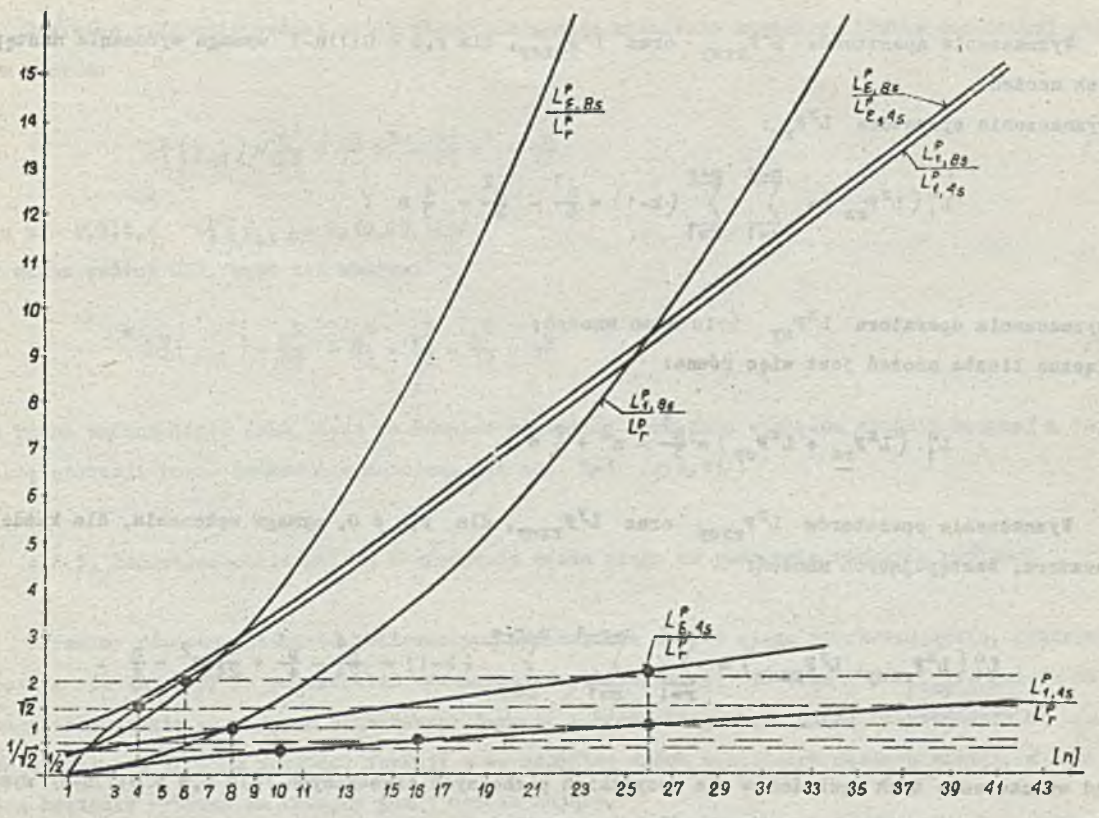
bądź

$$L_c^p\left(P_n^{\ 2}\right) = L_c^o(\Sigma LP)+L_c^o\left(\Sigma L^{p}F_{rxsy}\right) = \frac{n^5}{30}+\frac{n^4}{4}+\frac{2}{3}n^3+\frac{3}{4}n^2+\frac{3}{10}n \qquad /28/$$

/dla n=2,3,4,6 $L_c^p\left(P_n^{\ 2}\right) = 14, 54, 154, 756$/, zależnie od tego, czy odcinek łamanej jest wyznaczony wzdłuż GLO czy 2PLO. Złożoność przebiegu jednokrotnego pętli algorytmu jest więc określona wielomianem 5-stopnia względem n, tj. względem stopnia krzywej /rys.11/.

4.2.2. Złożoność obliczeniowa wyznaczania jednego węzła ciągu dla h=1

Załóżmy, że h=1, tj. że siatka ma wymiar jednostkowy. Jest to przypadek często występujący w praktyce. Co więcej, przypadki, w których h≠1, można w prosty sposób sprowadzić do poprzedniego przez np. odpowiednie skalowanie układu współrzędnych. Złożoność obliczeniowa algorytmów zmniejszy się wtedy w sposób istotny, chociaż nie zmieni się złożoność wyznaczania pochodnych w punkcie początkowym.



Rys. 11. Stosunki złożoności obliczeniowych wyznaczania jednego węzła ciągu aproksymującego dla różnych metod aproksymacji w funkcji stopnia krzywej n

Wyznaczenie operatora $L^R F$ wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_1^o(L^R F) = \sum_{k=1}^{n-1} \sum_{i=1}^{k} (k-1) = \frac{n^3 - 3n^2 + 2n}{3} .$$

Wyznaczenie operatorów $L^X F$ i $L^Y F$ wymaga wykonania, dla każdego operatora mnożeń w liczbie:

$$L_1^o(L^X F, L^Y F) = \sum_{k=1}^{n} (k-1) = \frac{n^2 - n}{2}$$

czyli w sumie:

$$L_1^o(L^X F + L^Y F) = n^2 - n .$$

a łącznie z wyznaczeniem operatora $L^R F$:

$$L_1(\Sigma LF) = \frac{n^3 - n}{3} \qquad /29/$$

/dla n=3 $L_1^o(\Sigma LF) = 8$/.

Wyznaczenie operatorów $L^x F_{rxsy}$ oraz $L^y F_{rxsy}$, dla $r,s = 0(1)n-1$ wymaga wykonania następujących mnożeń:

- wyznaczenia operatora $L^x F_{rx}$:

$$L_1^o(L^x F_{rx}) = \sum_{r=1}^{n-1} \sum_{k=1}^{n-r} (k-1) = \frac{n^3}{6} - \frac{n^2}{2} - \frac{4}{3} n \ ;$$

- wyznaczenia operatora $L^x F_{sy}$ tylu samo mnożeń;

Łączna liczba mnożeń jest więc równa:

$$L_1^o \left(L^x F_{rx} + L^x F_{sy} \right) = \frac{n^3}{3} - n^2 + \frac{2}{3} n \ .$$

Wyznaczenie operatorów $L^x F_{rxsy}$ oraz $L^y F_{rxsy}$, dla $r,s \neq 0$, wymaga wykonania, dla każdego operatora, następujących mnożeń:

$$L_1^o \left(L^x F_{rxsy}, L^y F_{rxsy} \right) = \sum_{r=1}^{n-2} \sum_{s=1}^{n-r-1} \sum_{k=1}^{n-r-s} (k-1) = \frac{n^4}{24} - \frac{n^3}{4} + \frac{11}{24} n^2 - \frac{n}{4} \ .$$

Stąd wyznaczenie tych operatorów dla wszystkich pochodnych cząstkowych /dla $r,s = 0(1)n-1$/ wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_1^o \left(\Sigma L^x F_{rxsy}, \Sigma L^y F_{rxsy} \right) = \frac{n^4}{24} + \frac{n^3}{12} - \frac{13}{24} n^2 + \frac{5}{12} n \qquad /30/$$

/dla $n=3$ $L_1^o \left(\Sigma L^x F_{rxsy}, \Sigma L^y F_{rxsy} \right) = 2$/.

Wyznaczenie operatora $L^p F_{rxsy}$ wymaga wykonania mnożeń w liczbie:

- dla pochodnych cząstkowych względem jednej zmiennej, kolejno x i y:

$$L_1^o \left(\Sigma L^p F_{rx}, \Sigma L^p F_{sy} \right) = \sum_{r=1}^{n-1} \sum_{k=1}^{n-r} \left[\sum_{i=0}^{k} (k-2) + 2 \right] = \frac{n^4}{12} - \frac{n^3}{6} - \frac{n^2}{12} + \frac{n}{6} \ .$$

- dla pochodnych cząstkowych względem obydwu zmiennych:

$$L_1^o \left(L^p F_{rxsy} \right) = \sum_{r=1}^{n-2} \sum_{s=1}^{n-r-1} \sum_{k=1}^{n-r-s} \left[\sum_{i=0}^{k} (k-2) + 2 \right] = \frac{n^5}{60} - \frac{n^4}{12} + \frac{n^3}{12} + \frac{n^2}{12} - \frac{n}{10} n$$

Łączna liczba niezbędnych do wykonania mnożeń w trakcie wyznaczania operatora $L^p F_{rxsy}$ wynosi:

$$L_1^o \left(\Sigma L^p F_{rxsy} \right) = \frac{n^5}{60} + \frac{n^4}{12} - \frac{n^3}{4} - \frac{n^2}{12} + \frac{7}{30} n \qquad /31/$$

/dla $n=3$ $L_1^o \left(\Sigma L^p F_{rxsy} \right) = 4$/.

Tak więc wykonanie jednej pętli algorytmu wymaga wykonania mnożeń w liczbie określonej poniższym wzorem:

$$L_1^p(p_{a1}) = \frac{n^4}{24} + \frac{5}{12}n^3 - \frac{13}{24}n^2 + \frac{n}{12} \qquad /32/$$

/dla n = 2,3,4,6 $L_1^p(p_{a1}) = 2,10,29,125$/
dla ruchu wzdłuż GLG, bądź też wzorem:

$$L_1^p(p_{a2}) = \frac{n^5}{60} + \frac{n^4}{12} + \frac{n^3}{12} - \frac{n^2}{12} - \frac{n}{10} \qquad /33/$$

dla ruchu wzdłuż linii 2PLG. Jest to również wielomian 5-stopnia względem stopnia krzywej n, dający liczbę operacji ponad dwukrotnie mniejszą niż gdy $h \neq 1$ /rys.11/.

4.2.3. Złożoność obliczeniowa wyznaczenia węzła ciągu na podstawie równania ogólnego

Określmy również złożoność obliczeniową wyznaczania węzłów ciągu aproksymującego, generowanego wprost z równania /3/ /z pominięciem wyznaczania operatorów $L^{(\cdot)}F$ oraz $L^{(\cdot)}F_{rxsy}$/. Metoda ta wymaga wyznaczenia pochodnych cząstkowych I-rzędu w bieżącym punkcie ciągu /niezbędnych do wyznaczenia wektorów $\overrightarrow{V}$/ oraz wartości funkcji w co najmniej dwóch sąsiednich węzłach siatki, z których jeden zostanie wybrany na kolejny punkt aproksymujący.

Wyznaczenie pochodnej I-rzędu /według wzoru /11// wymaga wykonania mnożeń, których liczba określa się wyrażeniem:

$$L_r^o(F_x,F_y) = \sum_{k=0}^{n-1}\left[\sum_{i=0}^{n-k-1}(n-k) - 1\right] = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5}{6}n \qquad /34/$$

Natomiast obliczanie wartości funkcji wprost z równania /3/ wymaga

$$\sum_{i=1}^{n}(i+1)\,i = \frac{n^3}{3} + n^2 + \frac{2}{3}n \qquad /35/$$

mnożeń. Wyznaczenie pochodnych F_x i F_y oraz wartości funkcji $F(x,y)$ w dwu węzłach wymaga więc wykonania mnożeń w liczbie określonej następującym wzorem:

$$L_r^p = \frac{4}{3}n^3 + 3n^2 - \frac{1}{3}n \qquad /36/$$

/dla n=2,3,4,6 L_r^p = 22, 62, 132, 394/, /rys.11/.

A więc złożoności obliczeniowe poszczególnych algorytmów określają się wielomianami względem stopni dyskretyzowanych krzywych, co należało wykazać. ■

4.3. Porównanie złożoności obliczeniowej algorytmów dla generacji krzywej dyskretnej

Wyprowadzone wyżej wzory na złożoność obliczeniową wyznaczenia jednego węzła ciągu aproksymującego za pomocą różnych algorytmów /na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$, z $h=1$, $h\neq1$ oraz wprost z równania uwikłanego krzywej /3.1// pozwalają na następujące stwierdzenia:

a/ najprostszym algorytmem jest:

- dla $n \leqslant 26$ - algorytm dyskretyzacji na siatce $G_{4s}(h^2)$ z $h=1$,
- dla $n > 26$ - algorytm dyskretyzacji wprost z równania uwikłanego /3/;

b/ najbardziej złożonym algorytmem jest:

- dla $n < 3$ - algorytm dyskretyzacji wprost z równania uwikłanego /3/;
- dla $n \geqslant 4$ - algorytm dyskretyzacji na siatce $G_{8c}(h^2)$ z $h\neq1$.

Z najmniejszej złożoności wyznaczenia jednego węzła ciągu określonym sposobem nie wynika jednak najmniejsza złożoność generacji krzywej tymże sposobem. Odległość między kolejnymi węzłami ciągu aproksymującego, wyznaczonego za pomocą różnych algorytmów dyskretyzacji jest bowiem różna. Algorytm dyskretyzacji na siatce $G_{4s}(h^2)$ zapewnia, że odległość ta jest stała i równa h. W algorytmie dyskretyzacji na siatce $G_{8c}(h^2)$ lub algorytmie generacji wprost z równania /3/ odległość ta może być równa h lub $\sqrt{2}h$. Jeden punkt ciągu może więc aproksymować dłuższy niż w przypadku poprzednim odcinek krzywej.

Przeprowadźmy wzajemne porównanie rozważanych metod dyskretyzacji, obliczając stosunki złożoności obliczeniowej wyznaczania jednego węzła ciągu aproksymującego w funkcji stopnia krzywej n /rys.11/. Interesować nas będą przebiegi zależności oraz wartości n, dla których stosunki złożoności:

a/ $\dfrac{L^p_{c,8s}}{L^p_r}$ i $\dfrac{L^p_{1,8s}}{L^p_r}$ są równe 1,

b/ $\dfrac{L^p_{c,4s}}{L^p_r}$ i $\dfrac{L^p_{1,4s}}{L^p_r}$ są równe $\dfrac{1}{\sqrt{2}}$,

c $\dfrac{L^p_{c,8s}}{L^p_{c,4s}}$ i $\dfrac{L^p_{1,8s}}{L^p_{1,4s}}$ są równe $\sqrt{2}$.

Na podstawie analizy tych zależności formułujemy następująco stwierdzenie o najszybszej generacji krzywych dyskretnych stopnia n przez poszczególne algorytmy:

Najszybszą generację krzywych dyskretnych, aproksymujących krzywe algebraiczne stopnia n można osiągnąć:

- dla $n \leqslant 3$ - algorytmem dyskretyzacji na siatce $G_{8s}(1^2)$,
- dla $4 \leqslant n \leqslant 16$ - algorytmem dyskretyzacji na siatce $G_{4s}(1^2)$,
- dla $n > 17$ - algorytmem dyskretyzacji wprost z równania /3/.

Tabela 1. Złożoność obliczeniowa wyznaczenia jednego węzła ciągu aproksymującego krzywą /mierzona liczbą wykonywanych mnożeń/ w funkcji stopnia krzywej n, dla różnych algorytmów aproksymacji krzywych oraz dla różnych modułów siatki /h=1, b≠1/

n	$L^p_{\epsilon,4s}$	$L^p_{\epsilon,8s}$	$L^p_{1,4s}$	$L^p_{1,8s}$	L^p_r
1	2	2	0	0	4
2	12	14	2	2	22
3	39	54	10	12	62
4	94	154	29	42	132
5	190	364	65	112	240
6	342	756	125	252	394
7	567	1428	217	504	602
8	884	2508	350	924	872
9	1324	4158	534	1584	1212
10	1880	6578	780	2574	1630
11	2607	10010	1100	4004	2134
12	3522	14742	1507	6006	2732
13	4654	21112	2015	8736	3432
14	6034	29512	2639	12376	4242
15	7695	40392	3395	17136	5170
16	9672	54264	4300	23256	6224
17	12002	71706	5372	31008	7412
18	14724	93366	6630	40698	8742
19	17879	119966	8094	52668	10222
20	21510	152306	9785	67298	11860
21	25662	191268	11725	85008	13664
22	30382	237920	13937	106260	15642
23	35719	293020	16445	131560	17802
24	41724	358020	19274	161460	20152
25	48450	434069	22450	196560	22700

4.4. Złożoność obliczeniowa dyskretyzacji krzywych

W poprzednich podrozdziałach przeanalizowaliśmy złożoności obliczeniowe wyznaczenia jednego węzła ciągu aproksymującego oraz generacji krzywej dyskretnej H, za pomocą poszczególnych algorytmów dyskretyzacyjnych.

Ze złożoności tych nie wynika jednak wprost złożoność obliczeniowa dyskretyzacji krzywej K. Różne są bowiem odległości między kolejnymi węzłami w krzywych dyskretnych, jak i długości tych krzywych i zależą od algorytmu, który te krzywe generował. Odległość między kolejnymi węzłami ciągu w algorytmie $G_{4s}(h^2)$ jak już wspomniano, jest stała i wynosi h, zaś w algorytmach $G_{8s}(h^2)$ oraz w algorytmie generującym na podstawie równania ogólnego krzywej, odległość ta może być równa h bądź $\sqrt{2}h$, w zależności od położenia krzywej względem linii siatki dyskretyzacyjnej. Długość l^H_{4s} krzywej dyskretnej, generowanej za pomocą algorytmu na siatce $G_{4s}(h^2)$ jest natomiast zawsze większa od długości l^H_{8s} i l^H_{ro} krzywych generowanych za pomocą dwóch pozostałych algorytmów. Z odległości

między kolejnymi węzłami krzywych dyskretnych wynika poniższa zależność na długości tych krzywych:

$$l^H_{8s,ro} \leqslant l^H_{4s} \leqslant 2\, l^H_{8s,ro} \ .$$

W praktyce $l^H_{4s} \simeq 1{,}26\ l^H_{8s,ro}$.

W odniesieniu do długości l^K krzywej dyskretyzowanej K można podać inną praktyczną zależność:

$$l^H_{8s} \simeq 1{,}12\ l^K \ .$$

Istotną rolę w ocenie złożoności obliczeniowej dyskretyzacji krzywych K mają następujące własności algorytmów dyskretyzacyjnych:

gdy segment krzywej K przebiega równolegle /w przybliżeniu/ do 2PLG, wówczas ten sam segment krzywej K aproksymowany jest różną liczbą węzłów oddalonych od siebie na różną odległość, np. segment krzywej o długości ok. 3h aproksymowany jest przez algorytm $G_{4s}(h^2)$ ciągiem pięciu węzłów odległych od siebie o h, natomiast dwa pozostałe algorytmy aproksymują ten segment ciągiem trzech węzłów, odległych od siebie o $\sqrt{2}\ h$.

Złożoność obliczeniowa wyznaczenia jednego węzła krzywej H jest, jak wiadomo, dla każdego algorytmu różna. Chcąc zatem porównać złożoności obliczeniowe dyskretyzacji krzywych za pomocą poszczególnych algorytmów porównamy jeszcze raz złożoności obliczeniowe wyznaczenia jednego węzła ciągu aproksymującego. Interesować nas jednak będą inne wartości liczbowe stosunków tych złożoności, a mianowicie stosunki odniesione do złożoności obliczeniowych dyskretyzacji jednostkowego odcinka krzywej K. A więc interesują nas przedziały liczbowe wartości n stopnia krzywej przy następujących wartościach liczbowych poszczególnych stosunków złożoności:

$$\frac{L^p_{1,8s}}{L^p_{1,4s}} \quad \text{i} \quad \frac{L^p_{c,8s}}{L^p_{c,4s}} \qquad \text{równych } 2$$

$$\frac{L^p_{1,8s}}{L^p_r} \quad \text{i} \quad \frac{L^p_{c,8s}}{L^p_r} \qquad \text{równych } 1$$

$$\frac{L^p_{1,4s}}{L^p_r} \quad \text{i} \quad \frac{L^p_{c,4s}}{L^p_r} \qquad \text{równych } \frac{1}{2} \ .$$

Dla wymienionych wartości złożoności algorytmy, które porównujemy, są pod względem tych złożoności równoważne.

Z powyższych porównań wynikają przedziały wartości stopnia krzywej n, dla których dyskretyzacja krzywej K, za pomocą określonego algorytmu jest najszybsza:

- dla $n \leqslant 5$ - algorytmem dyskretyzacji na siatce $G_{8s}(l^2)$,

- dla $6 \leq n < 10$ - algorytmem dyskretyzacji na siatce $G_{4s}(1^2)$,
- dla $n > 11$ - algorytmem dyskretyzacji wprost z równania /3/.

Złożoność obliczeniową dyskretyzacji określonej krzywej, będącej funkcją długości tej krzywej, można wyliczyć z poniższych wzorów.

W dyskretyzacji na siatce $G_{4s}(1^2)$ złożoność wyraża się wzorem:

$$L_{1,4s} = L^p_{\Sigma} + m_{xy} L^p_i (p_{a1}) .$$

gdzie m_{xy} jest pomniejszoną o 1 liczbą punktów ciągu aproksymującego krzywą. W aproksymacji krzywej odcinkami linii siatkowych liczba ta jest długością linii łamanej, aproksymującej daną krzywą.

Dla siatki $G_{8s}(1^2)$ złożoność obliczeniowa dyskretyzacji krzywej wyraża się innym wzorem:

$$L_{1,8s} = L^p_{\Sigma} + m_{xy} L^p (p_{a1}) + m_p L^p_i (p_{a2}) .$$

przy czym $m_{xy} + m_p$ jest pomniejszoną o 1 liczbą punktów ciągu aproksymującego krzywą:

- m_{xy} jest liczbą punktów wyznaczonych w pętli p_{a1} algorytmu,
- natomiast m_p jest liczbą punktów wyznaczonych w pętli p_{a2} tegoż algorytmu.

Liczba $m_{xy} + \sqrt{2} \cdot m_p$ określa długość linii łamanej, aproksymującej krzywą zadaną.

Przeprowadzona wyżej ocena złożoności obliczeniowej algorytmów dyskretyzacyjnych dotyczy algorytmów dyskretyzacji dowolnych krzywych algebraicznych stopnia n. Stąd też, dla uproszczonych postaci algorytmów, prawdziwe są równości:

$$Z_{4s}(1^2) = L_{1,4s}, \qquad Z_{8s}(1^2) = L_{1,8s} \qquad \text{oraz} \qquad Z(rh) = L^p_r ,$$

gdzie $Z_{4n}(1^2)$ i $Z_{8s}(1^2)$ są złożonościami obliczeniowymi dyskretyzacji krzywych algebraicznych stopnia n ze względu na przedstawione algorytmy dyskretyzacji na siatkach

$$G_{4s}(1^2) \quad \text{i} \quad G_{8s}(1^2) .$$

W praktyce występują krzywe, dla których niektóre ze współczynników $a_{i,j}$ równania /3/ są zerowe. Zerowe wartości będą miały wówczas również i niektóre z pochodnych cząstkowych. Sprawdzając zatem wartości tych współczynników można w przypadku ich zerowej wartości zmniejszyć liczbę operacji w algorytmach.

5. ANALIZA BŁĘDÓW DYSKRETYZACJI

Głównym celem każdej aproksymacji jest osiągnięcie pewnego określonego stopnia dokładności przybliżenia według zadanego kryterium. Tak więc podjęcie aproksymacji powinno być uwarunkowane założeniem, że skuteczność tego przybliżenia da się oszacować.

W teorii aproksymacji znanych jest wiele różnych miar jakości aproksymacji funkcji $F(t)$ funkcją $f(t,p)/p$ - parametr aproksymacji/. Wyróżnić można klasę miar skrzyżowych, charakteryzujących aproksymację lokalnie, wykazujących ekstremalne wartości błędu i argumentu, przy których on występuje, oraz klasę miar globalnych, charakteryzujących aproksymację w przedziale $[a,b]$ globalnie, bez wyróżnienia argumentu t, dla którego to przybliżenie jest najgorsze lub najlepsze.

Ocena błędów aproksymacji ściśle zależy od wyboru miary, w której ocena ta jest dokonywana. Miara natomiast powinna być adekwatna do problemu, w ramach którego aproksymacja jest wykonywana.

Na etapie syntezy algorytmów zastosowaliśmy przybliżenia jednostajne, polegające na minimalizacji lokalnej odległości między krzywą zadaną i jej przybliżeniem /łamaną/. W tym rozdziale wybierzemy metrykę do oceny tego przybliżenia, mając na uwadze przeznaczenie omawianych metod. Sądzimy, że powinna to być inna metryka niż generacyjna, która została wybrana przede wszystkim ze względu na prostotę /i szybkość/ działania algorytmów.

Zakładając zastosowanie algorytmów przede wszystkim w grafice komputerowej i numerycznym sterowaniu obrabiarkami, w których to dziedzinach stawia się wymagania aproksymowania kształtów figur geometrycznych, czyli aproksymacji w zbiorach sądzimy, że wybrana do oceny jakości aproksymacji metryka powinna uwzględniać cechy percepcji wzrokowej człowieka, tzn. powinna być odpowiednia do "metryki" systemu wzrokowego człowieka. Należy więc uwzględnić fakty znane z dziedziny psychofizjologii systemu wzrokowego człowieka. Wiadomo, że w procesie oceny wzrokowej kształtów figur geometrycznych istnieją dwa stadia:

1/ ocena lokalnych odchyłek od "regularności" przylegających części figury, co identyfikuje się z pomiarem względnych skrzyżowych błędów, mierzonych w metryce zbiorów;

2/ ocena globalna figury uwarunkowana kontekstem, wzorcami, regularnością kształtów i zniekształceń.

Żadna ze znanych metod oceny aproksymacji nie uwzględnia jednak w pełni czynników wpływających na ocenę kształtów przez system wzrokowy człowieka.

W literaturze fachowej zagadnienie doboru metod oceny algorytmów dyskretyzacji krzywych do zastosowań w grafice komputerowej nie jest prawie w ogóle omawiane. W jedynej publikacji poruszającej te problemy, Chuang /1977/r./ uważa, że algorytmy dyskretyzacyjne powinny minimalizować jedną z następujących wielkości:

a/ ϱ - odległość liniową węzła od krzywej,

b/ $|F(x,y)|$ - moduł wartości funkcji opisującej krzywą zadaną równaniem $F(x,y) = 0$,

c/ |pole wyjściowe minus pole wynikowe| aproksymowanej figury.

W niniejszej rozprawie ocenę aproksymacji wykonamy w dwóch "metrykach": w metryce przestrzeni euklidesowej /miara lokalna/ oraz w "metryce polowej" /miara globalna/, "nieczułej" na lokalne zniekształcenia figur, a oceniającej jakość aproksymacji wielkością modułu pola między krzywą aproksymowaną i aproksymującą.

5.1. Szczytowe błędy dyskretyzacji

5.1.1. Bezwzględne szczytowe błędy dyskretyzacji

Za pomocą omówionych w poprzednim rozdziale algorytmów, na podstawie równania $F(x,y)=0$ opisującego krzywą, można generować ciąg bezpośrednio spójnych węzłów siatki, w których moduł wartości funkcji $F(x,y)$ osiąga lokalne minimum. To minimum gwarantuje nie większą odległość niż h każdego z punktów ciągu od zadanej krzywej. Spośród węzłów ciągu aproksymującego krzywą najbardziej oddalone od niej będą te węzły, w których linia łamana /łącząca sąsiednie punkty ciągu/ zmienia kierunek /przeguby linii łamanej/, co odbywa się zawsze w kierunku krzywej, tak aby osiągnąć węzeł z możliwie najmniejszym modułem $F(x,y)$. Stąd też wyznaczenie szczytowych błędów dyskretyzacji sprowadza się do wyznaczenia odległości przegubów linii łamanej od krzywej.

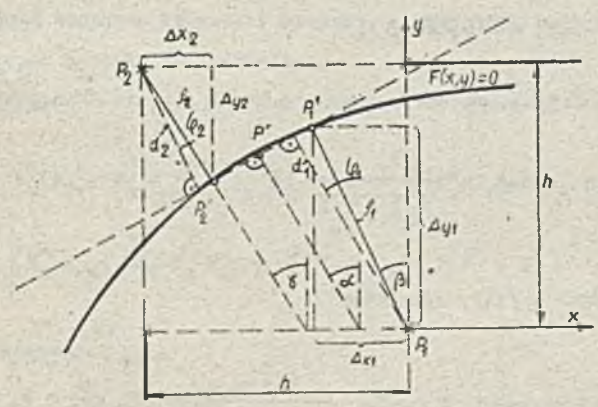
Spośród wszystkich możliwych wzajemnych położeń siatki i krzywej największe odchylenie linii łamanej i krzywej /czyli największy bezwzględny szczytowy błąd aproksymacji krzywej łamaną/ wystąpi wówczas, gdy w dwóch węzłach siatki przewidywanych na kolejny węzeł ciągu, wartości funkcji różnić się będą tylko znakami, tzn. węzły te będą położone po przeciwnych stronach krzywej, a moduły wartości funkcji będą równe.

Rozpatrzmy teraz łuk krzywej K zadanej równaniem /3/, dla którego to łuku kąt nachylenia stycznej $0 \leqslant \alpha \leqslant \frac{\pi}{2}$ oraz aproksymującą ten łuk krzywą dyskretną H - ciąg węzłów siatki, wygenerowany jest przez algorytm, którego metryką generacyjną jest $|F(x,y)|$. Niech w dwóch węzłach siatki, leżących po przeciwnych stronach krzywej K moduły wartości funkcji będą sobie równe. Oznaczny te węzły przez P_1 i P_2, a ich odległości od krzywej przez ϱ_1 i ϱ_2 /rys.12/. Wyznaczmy bezwzględny szczytowy błąd ϱ_1, gdy zachodzi równość:

$$|F(P_1)| = |F(P_2)| \qquad /37/$$

lub jej równoważna

$$F(P_1) + F(P_2) = 0 \qquad /38/$$



Rys.12. Ilustracja wyznaczania odległości węzłów od krzywej

Twierdzenie 3

W dyskretyzacji krzywych K zadanych równaniem /3/ /włącznie z założeniami/ przeprowadzonej na siatce $G(h^2)$ w "metryce" $|F(x,y)|$ oszacowanie ϱ bezwzględnego szczytowego błędu tej dyskretyzacji, mierzonego w metryce euklidesowej opisuje się równaniem algebraicznym /zmiennej ϱ/ stopnia nie wyższego niż stopień dyskretyzowanej krzywej.

Dowód

Dla dowodu wyprowadzimy równanie zmiennej ϱ, z którego można wyznaczyć wartość ϱ/oszacowania bezwzględnego szczytowego błędu dyskretyzacji/ ogólnie znanymi metodami. Stopień tego równania będzie nie wyższy niż stopień równania dyskretyzowanej krzywej /3/.

Poprowadźmy prostopadłe do krzywej K z punktów P_1 i P_2 i oznaczmy punkty przecięć prostopadłych i krzywej odpowiednio P_1' i P_2', jak na rys. 12. Poprowadźmy również linię równoległą do prostej $\overline{P_1', P_2'}$ i styczną do krzywej w punkcie P'. Kąt nachylenia stycznej w punkcie P' oznaczmy przez α.

Wyznaczmy następnie wartości funkcji $F(x,y)$ w punktach P_1 i P_2 posługując się rozwinięciem tej funkcji w szereg Taylora w otoczeniu punktów P_1' i P_2':

$$F(P_1) = F(P_1') + \sum_{i=1}^{n} \frac{1}{i!} \left(\Delta x_1 \frac{\partial}{\partial x} + \Delta y_1 \frac{\partial}{\partial y} \right)^i F(P_1') \qquad /39/$$

$$F(P_2) = F(P_2') + \sum_{i=1}^{n} \frac{1}{i!} \left(\Delta x_2 \frac{\partial}{\partial x} + \Delta y_2 \frac{\partial}{\partial y} \right)^i F(P_2')$$

Ponieważ zgodnie z definicją krzywej zachodzi:

$$|F(P_1')| = |F(P_2')| \overset{df}{=} 0 \qquad /40/$$

oraz prawdziwe są, dla kąta nachylenia stycznej do krzywej $0 < \alpha < \frac{\pi}{2}$, poniższe związki

$$\Delta x_1 = \varrho_1 \sin\beta \ , \quad \Delta y_1 = -\varrho_1 \cos\beta \ ,$$
$$\Delta x_2 = -\varrho_2 \sin\delta \ , \quad \Delta y_2 = \varrho_2 \cos\delta \ , \qquad /41/$$

gdzie $\sin\alpha = -\dfrac{F_y}{\sqrt{F_x^2 + F_y^2}}(P')$, $\cos\alpha = \dfrac{F_x}{\sqrt{F_x^2 + F_y^2}}(P')$

otrzymujemy z /38/, na podstawie /39/ + /41/, równanie:

$$\sum_{i=1}^{n} \frac{1}{i!} \left[\frac{(-\varrho_1)^i}{(F_x^2 + F_y^2)^{i/2}} \left(F_x \frac{\partial}{\partial x} + F_y \frac{\partial}{\partial y} \right)^i F(P_1') + \frac{\varrho_2^i}{(F_x^2 + F_y^2)^{i/2}} \left(F_x \frac{\partial}{\partial x} + F_y \frac{\partial}{\partial y} \right)^i F(P_2') \right] = 0.$$

Łatwo wykazać również istnienie poniższego związku

$$\varrho_1 \cos \beta_1 + \varrho_2 \cos \beta_2 = Q_h, \quad \beta_1 = \beta - \alpha, \quad \beta_2 = \alpha - \delta, \qquad /43/$$

gdzie $Q_h = Q_h(\alpha, \gamma)$ jest funkcją rodzaju siatki kwadratowej, kąta nachylenia stycznej do krzywej α i kąta γ nachylenia głównych linii siatki do osi układu współrzędnych.

Dokonajmy następującego uproszczenia: linearyzujmy krzywą, na przestrzeni ograniczonej przez elementarne pole siatki, odcinkiem stycznej wystawiony w punkcie P'. Dla krzywych o dostatecznie dużych - w stosunku do modułu siatki h - promieniach krzywizn $/R \gg h/$ kąty β i δ różnią się od kąta α nieznacznie. Linearyzacja ta jest równoważna założeniu

$$\beta_1 \approx \beta_2 \approx 0$$

co oznacza, że $\beta \approx \alpha \approx \delta$, a wartości pochodnych cząstkowych funkcji F(x,y) w punktach P_1', P_2' i P' można przyjąć za równe.

Równania /42/ i /43/ przyjmą wtedy odpowiednio postać:

$$\sum_{i=1}^{n} \frac{(-1)^i}{i!\left(F_x^2 + F_y^2\right)^{i/2}} \left[\varrho_1^i + (-\varrho_2)^i \right] \left[F_x \frac{\partial}{\partial x} + F_y \frac{\partial}{\partial y} \right]^i F(P') = 0 \qquad /44/$$

oraz

$$\varrho_1 + \varrho_2 = Q_h, \quad \varrho_1 = d_1, \quad \varrho_2 = d_2 . \qquad /45/$$

Przyjmując Q_h jako znane, otrzymujemy ostatecznie równanie algebraiczne n-stopnia, z którego można wyznaczyć ϱ_1 - kres górny szczytowego błędu bezwzględnego

$$\sum_{i=1}^{n} \frac{(-1)^i}{i!\left(F_x^2 + F_y^2\right)^{i/2}} \left[2\,\varrho_1^i + \sum_{k=1}^{i} \binom{i}{k} \varrho_1^{i-k} (-Q_h)^k + (-Q_h)^i \right] \cdot \left(F_x \frac{\partial}{\partial x} + F_y \frac{\partial}{\partial y} \right)^i F(P') = 0 \qquad /46/$$

gdzie n jest stopniem krzywizny krzywej, a wszystkie pochodne cząstkowe wyznacza się w punkcie P'.

A więc mamy jak w tezie twierdzenia 3. ∎

Dla krzywych 2-stopnia /n=2/ otrzymujemy równanie:

$$2\varrho_1^2 F_x^2 (F_{2x} + 2F_x F_y F_{xy} + F_y^2 F_{2y}) - 2\varrho_1 \left[(F_x^2 F_{2x} + 2F_y F_{2y}) Q_h + 2(F_x^2 + F_y^2)^{\frac{3}{2}} \right] -$$
$$- Q_h \left[(F_x^2 F_{2x} + 2F_x F_y F_{xy} + F_y^2 F_{2y}) Q_h + 2(F_x^2 + F_y^2)^{\frac{3}{2}} \right] = 0 , \qquad /47/$$

którego rozwiązaniem jest:

$$\varrho_1 = \frac{R}{\left(\frac{h}{\sqrt{2}}\right)^{\frac{2}{3}} + t\left(\frac{Rh}{2}\right)^{\frac{2}{3}} - 1} + \frac{Q_h}{2} - \sqrt{\left[\frac{R^2}{\left(\frac{R}{2}\right)^{\frac{2}{3}} + t\left(\frac{Rh}{2}\right)^{\frac{2}{3}} - 1} \right]^2 + \frac{Q_h^2}{4}} \qquad /48/$$

gdzie R, a, b i e oznaczają odpowiednio: promień krzywizny, półosie dużą i małą oraz parametr krzywej 2-stopnia, natomiast

$$t = \begin{cases} 1 & \text{dla prostej okręgu i elipsy} \\ 0 & \text{dla paraboli} \\ -1 & \text{dla hiperboli.} \end{cases}$$

Uwzględniając, że dla siatki $G_{4s}(h^2)$ zachodzi związek:

$$d_1 + d_2 = h(\sin\theta + \cos\theta) = \frac{h\sqrt{2}}{2}\cos\left(\frac{\pi}{4} - \theta\right), \quad 0 \leqslant \theta \leqslant \frac{\pi}{2} \qquad /49/$$

a z przyjętych uproszczeń wynika $d_1 = \varrho_1$ i $d_2 = \varrho_2$, po rozwinięciu wyrażenia $Q_h = Q_h(\alpha, \Psi)$ i podstawieniu go do wzoru /48/, otrzymujemy uogólniony wzór szacujący ϱ_1 dla krzywych 2-stopnia, aproksymowanych na siatce $G_{4s}(h^2)$

$$\varrho_1 = \frac{R}{\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1} + \frac{h\sqrt{2}}{2}\cos\left(\frac{\pi}{4} - \theta\right) + \sqrt{R^2\left[\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1\right]^{-2} + \frac{h^2\cos^2\left(\frac{\pi}{4} - \theta\right)}{2}} \qquad /50/$$

Podobnie dla siatki $G_{8s}(h^2)$, prawdziwe są związki:

$$\begin{aligned} d_1 + d_2 &= h\cos\theta, \quad 0 \leqslant \theta \leqslant \tfrac{\pi}{4}, \\ d_1 + d_2 &= h\sin\theta, \quad \tfrac{\pi}{4} \leqslant \theta \leqslant \tfrac{\pi}{2}, \end{aligned} \qquad /51/$$

które, ze względu na uproszczenie $d_1 = \varrho_1$ i $d_2 = \varrho_2$, pozwalają zapisać wzór /48/ w postaciach:

$$\varrho_1 = \frac{R}{\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1} + \frac{h\cos\theta}{2} - \sqrt{\frac{R^2}{\left[\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1\right]^2} + \frac{h^2\cos^2\theta}{4}} \qquad /52/$$

dla $0 \leqslant \theta \leqslant \frac{\pi}{4}$, oraz

$$\varrho_1 = \frac{R}{\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1} + \frac{h^2\sin\theta}{2} - \sqrt{\frac{R^2}{\left[\left(\frac{R}{e}\right)^{\frac{2}{3}} + t\left(\frac{Rb}{a^2}\right)^{\frac{2}{3}} - 1\right]^2} + \frac{h^2\sin^2\theta}{4}} \qquad /53/$$

dla $\frac{\pi}{4} \leqslant \theta \leqslant \frac{\pi}{2}$, gdzie $\theta = \alpha - \Psi$ jest kątem nachylenia stycznej, a Ψ - kątem nachylenia GLG.

Z rozważań zawartych w [63] wynika, że dla krzywych 2-stopnia:

1/ błąd szczytowy ϱ zależy od trzech wielkości: R, RSG, θ;

2/ $\exists_{\theta_{ex}} \left\{ \max \varrho_{8s}(\theta_{ex}) = \min \varrho_{4s}(\theta_{ex}) \right\}$.

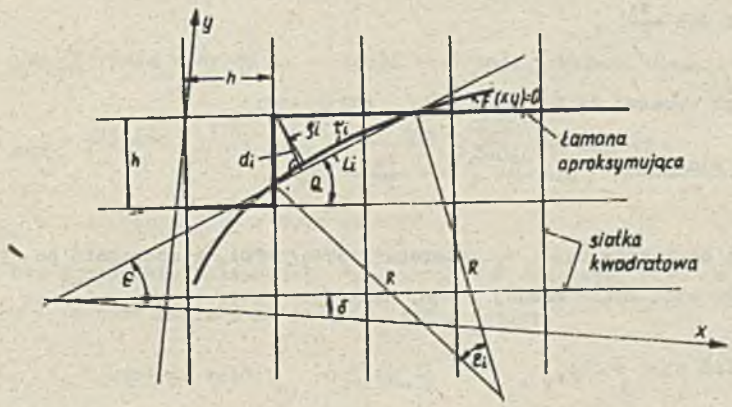
tj. dla pewnego kąta nachylenia stycznej θ_{ex} błąd ϱ osiąga swoją wartość ekstremalną: maksymalną dla dyskretyzacji na siatce $G_{8s}(h^2)$, minimalną dla dyskretyzacji na siatce $G_{4s}(h^2)$, przy czym wartości te są sobie równe.

Na podstawie analizy błędów szczytowych dyskretyzacji prostych i krzywych 2-stopnia wnioskujemy, że dla dowolnego n ϱ również będzie zależało od trzech wielkości: R, RSG, θ.

5.1.2. *Względne szczytowe błędy dyskretyzacji*

Zdefiniujmy względny szczytowy błąd dyskretyzacji jako stosunek odległości ϱ_i przegubu łamanej H od krzywej K do długości τ_i łuku, wyznaczonego przez dwa najbliższe punkty przecięcia się krzywej z łamaną, jak na rys. 13:

$$\varrho_{w_i} \stackrel{df}{=} \frac{\varrho_i}{\tau_i} \qquad /54/$$



Rys.13. Określenie względnego szczytowego błędu dyskretyzacji

● Twierdzenie 4

Względny szczytowy błąd ϱ_{w_i} dyskretyzacji krzywych K, przy założeniu $R \geq 10h$, jest równy:

1/ dla siatki $G_{4s}(h^2)$

$$\varrho_{w_i} < 1.04 \sin\theta \cos\theta h, \quad 0 \leq \theta \leq \frac{\pi}{2}, \qquad /55a/$$

2/ dla siatki $G_{8s}(h^2)$

$$\varrho_{w_i} < 1.06(\sin\theta \cos\theta - \sin^2\theta)h, \quad 0 \leq \theta \leq \frac{\pi}{4}. \qquad /55b/$$

Dowód

$$\varrho_{w_i} \stackrel{df}{=} \frac{\varrho_i}{\tau_i} = \frac{d_i + \Delta\varrho_i}{\tau_i},$$

gdzie $\tau_i = \varepsilon_i R_i$, a ε_i jest kątem środkowym łuku τ_i krzywej K.

Z założenia $R \gg h$ wynika, że w elementarnym polu siatki jest $R \gg \Delta R$, tzn. zmiana promienia krzywizny krzywej jest do pominięcia /$\lim_{(h/R)\to 0} (\Delta R/R) = 0$/.

Wówczas $\Delta\varrho_i$ wyznaczymy ze wzoru na strzałkę ugięcia łuku τ_i okręgu o kącie środkowym ε_i i promieniu R.

Dla siatki $G_{4s}(h^2)$ mamy:

$$\Delta\varrho_i = R\left(1 - \cos\frac{\varepsilon_i}{2}\right) \cdot 2 \sin\theta \cos\theta$$

$$\varrho w_i = \frac{d_i + 2 \cdot R \sin\theta \cos\theta\left(1 - \cos\frac{\varepsilon_i}{2}\right)}{\varepsilon_i R_i}$$

$$\frac{d_i}{l_i} = \sin\theta \cos\theta, \quad 0 \leq \theta \leq \frac{\pi}{2} .$$

natomiast l_i określa się wzorem:

$$l_i = 2 R \sin\frac{\varepsilon_i}{2} .$$

Po podstawieniu ostatnich wyrażeń do wzoru na ϱw_i otrzymamy:

$$\varrho w_i = \frac{2 \sin\theta \cos\theta}{\varepsilon_i}\left(\sin\frac{\varepsilon_i}{2} - \cos\frac{\varepsilon_i}{2} + 1\right) h$$

Po zastąpieniu sinusów i cosinusów kąta ε_i szeregami potęgowymi, a następnie po wyciągnięciu pierwszego wyrazu szeregu spod znaku każdej z sum, otrzymamy wzor:

$$\varrho w_i = h \sin\theta \cos\theta + \Delta\varrho w_i ,$$

gdzie $\Delta\varrho w_i = \sin\theta \cos\theta\, h \sum_{n=1}^{\infty} \frac{(-1)^n}{(2n)!}\left(\frac{\varepsilon_i}{2}\right)^{2n-1}\left(\frac{\varepsilon_i}{2(2n+1)} - 1\right)$.

Występujący we wzorze na $\Delta\varrho w_i$ szereg jest szeregiem przemiennym. Jak wiadomo, suma szeregu przemiennego jest mniejsza od jego pierwszego wyrazu i ma znak taki, jak ten wyraz. Możemy więc napisać:

$$\Delta\varrho w_i < \delta_i h \cdot \sin\theta \cos\theta ,$$

oraz

$$\Delta\varrho w_i > 0 ,$$

gdzie

$$\delta_i = \frac{\varepsilon_i}{4}\left(1 - \frac{\varepsilon_i}{6}\right).$$

Funkcja δ_i osiąga maksimum w punkcie $\varepsilon_i = 3$. A więc δ_i wzrasta, gdy ε_i wzrasta, co zachodzi w przedziale $0 \leq \varepsilon_i \leq 3$, δ_i maleje, gdy ε_i wzrasta, co zachodzi w przedziale $3 < \varepsilon_i < \infty$, oraz $\delta_i(3) = \frac{3}{8}$.

Jeżeli $\varepsilon_{max} < 3$ /tzn. znajduje się w przedziale, w którym dla zwiększającego się ε_i, δ_i maleje, wówczas ε_i możemy wyznaczyć jako różnicę szczytowych wartości błędów ρ w dyskretyzacji prostej oraz okręgu /wzór /50//:

$$\Delta R_{max} = \rho_1(0) - \rho_1(P) = R - \sqrt{R^2 - \frac{h^2}{2}\cos^2\left(\frac{\pi}{4} - \theta\right)}.$$

Z drugiej strony

$$\Delta R_i = R\left(1 - \cos\frac{\varepsilon_i}{2}\right).$$

Przyrównując do siebie te dwie wielkości otrzymujemy

$$\varepsilon_{max} = 2 \arccos \sqrt{1 - \frac{h^2\cos^2\left(\frac{\pi}{4} - \theta\right)}{2R^2}}$$

a następnie

$$\delta_{max} = \delta\left(\varepsilon_{max}\right) = \frac{1}{2}\arccos\sqrt{1 - \frac{h^2\cos^2\left(\frac{\pi}{4} - \theta\right)}{2R^2}} \cdot \left(1 - \frac{1}{3}\arccos\sqrt{1 - \frac{h^2\cos^2\left(\frac{\pi}{4} - \theta\right)}{2R^2}}\right)$$

Dla $R = 10h$ i $\theta = \frac{\pi}{4}$ /dla których to wartości występuje maksimum strzałki ugięcia/ otrzymamy:

$$\delta_{max} = \frac{\arccos}{2}\sqrt{\frac{199}{200}} \cdot \left(1 - \frac{\arccos}{3}\sqrt{\frac{199}{200}}\right) \approx 0.0342,$$

przy czym $\varepsilon_{max} = 2 \arccos\sqrt{\frac{199}{200}} \approx 0.14 \text{ rd} \approx 8^\circ$.

W interesującym nas przedziale zmienności ε_i funkcja δ_i maleje, gdy ε_i maleje. Dlatego też dla siatki $G_{4s}(h^2)$, możemy napisać nierówność, jak w tezie:

$$\rho_{w_i} < 1.04h\sin\theta\cos\theta, \qquad 0 \leqslant \theta \leqslant \frac{\pi}{2}.$$

Dla siatki $G_{8s}(h^2)$ obowiązuje inna zależność:

$$d_i = l_i(\sin\theta\cos\theta - \sin^2\theta), \qquad 0 \leqslant \theta \leqslant \frac{\pi}{4}.$$

Wzór na ρ_{w_i} ma więc postać:

$$\rho_{w_i} = \frac{2R\sin\frac{\varepsilon_i}{2}(\sin\theta\cos\theta - \sin^2\theta) + R\left(1-\cos\frac{\varepsilon_i}{2}\right)\cdot 2(\sqrt{2}+1)(\sin\theta\cos\theta - \sin^2\theta)}{\varepsilon_i R} =$$

$$= \frac{2(\sin\theta\cos\theta - \sin^2\theta)}{\varepsilon_i}h\left[\sin\frac{\varepsilon_i}{2} - (\sqrt{2}+1)\cos\frac{\varepsilon_i}{2} + \sqrt{2} + 1\right]h.$$

Zamieniając w ostatnim wyrażeniu sinusy i kosinusy kąta ε_i rozwinięciami potęgowymi, a następnie wydzielając spod znaków sum pierwsze wyrazy, otrzymamy następujące zależności:

$$\rho_{w_i} = h(\sin\theta\cos\theta - \sin^2\theta)h + \rho_{w_i}(\theta, \varepsilon_i),$$

gdzie:

$$\Delta\zeta_{w_i} = (\sin\theta\cos\theta - \sin^2\theta)h\sum_{n=1}^{\infty}\frac{(-1)^n\left(\frac{\varepsilon_i}{2}\right)^{2n-1}}{(2n)!}\left[\frac{\varepsilon_i}{2(2n+1)} - (\sqrt{2}+1)\right].$$

Pierwszy wyraz tego /przemiennego/ szeregu jest równy:

$$\delta_i = \frac{\varepsilon_i}{4}\left(\sqrt{2}+1-\frac{\varepsilon_i}{6}\right),$$

a jego wartość osiąga maksimum w punkcie $\varepsilon_i = 3(\sqrt{2}+1)$. Wartość δ_i zwiększa się, gdy ε_i zwiększa się w przedziale $0 \leqslant \varepsilon_i \leqslant 3(\sqrt{2}+1)$; δ_i maleje, gdy ε_i zwiększa się w przedziale $3(\sqrt{2}+1) < \varepsilon_i < \infty$, oraz $\delta_i(3\sqrt{2}+3) = \frac{3}{8}(2\sqrt{2}+3)$.

Jeżeli $\varepsilon_{max} = 3(\sqrt{2}+1)$, tzn. znajduje się w przedziale, w którym ε_i zwiększa się, gdy δ_i się zwiększa, wówczas ε_{max} możemy wyznaczyć jako różnicę błędów ϱ w dyskretyzacji prostej P i okręgu O /wyrażenie /52//:

$$\Delta R_{max} = \varrho_1(O) - \varrho_1(P) = R - \sqrt{R^2 - \frac{h^2\cos^2\theta}{4}}$$

Przyrównując ΔR_{max} określone powyższą zależnością z ΔR_i ze wzoru:

$$\Delta R_i = R\left(1 - \cos\frac{\varepsilon_i}{2}\right),$$

otrzymujemy wyrażenie na ε_{max}:

$$\varepsilon_{max} = 2\arccos\sqrt{1 - \frac{h^2\cos^2\theta}{4R^2}},$$

oraz wyrażenie na δ_{max}:

$$\delta_{max} = \delta_i(\varepsilon_{max}) = \frac{1}{2}\arccos\sqrt{1-\frac{h^2\cos^2\theta}{4R^2}}\cdot\left(\sqrt{2}+1-\frac{1}{3}\arccos\sqrt{1-\frac{h^2\cos^2\theta}{4R^2}}\right).$$

Wielkości te, dla $R = 10h$ i $\theta = \frac{\pi}{8}$ /dla których to wartości występuje maksimum strzałki ugięcia/ przyjmują następujące wartości:

$$\delta_{max} = \frac{1}{2}\arccos\sqrt{1-\frac{2+\sqrt{2}}{1600}}\left(\sqrt{2}+1-\frac{1}{3}\arccos\sqrt{1-\frac{2+\sqrt{2}}{1600}}\right) \approx 0.0599$$

$$\varepsilon_{max} = 2\arccos\sqrt{1-\frac{2+\sqrt{2}}{1600}} \approx 0.1\ \text{rd} \approx 6^\circ.$$

Ponieważ w interesującym nas przedziale zmienności ε_i wartość δ_i maleje, gdy ε_i maleje, dlatego też możemy, dla siatki $G_{8s}(h^2)$ napisać nierówność, jak w tezie twierdzenia:

$$\varrho_{w_1} < h1.06(\sin\theta\cos\theta - \sin^2\theta)h, \quad 0 \leqslant \theta \leqslant \frac{\pi}{4}.$$

■

Wyznaczone za pomocą wzorów /54/ i /55/ błędy mają swoje kresy górne:

$$\sup(\varrho_{w_1}) \approx 0.52h \quad \text{dla} \quad \theta = \frac{\pi}{4} \qquad /56/$$

w przypadku siatki $G_{4s}(h^2)$, oraz

$$\sup\left(\varrho_{w_1}\right) \approx 0.2195h \quad \text{dla} \quad \theta = \frac{\pi}{8} \qquad /57/$$

w przypadku siatki $G_{8s}(h^2)$.

Z udowodnionych zależności wynikają następujące właściwości błędu ϱ_{w_1} :

1/ $\varrho_{w_1} = \varrho_{w_1}(R, RSG, \theta)$.

2/ $\varrho_{w_1,4s} > \varrho_{w_1,8s}$.

3/ $\sup\left(\varrho_{w_1,4s}\right) > 2\sup\left(\varrho_{w_1,8s}\right)$.

Ponadto dla dyskretyzacji krzywych 2-stopnia obowiązuje poniższe zależność na wartość średnią $\varrho_{w_{śr}}$ błędu ϱ_{w_1} , powstałą z jego uśrednienia wzdłuż łuku krzywej K /wyprowadzona w pracy [63]/:

$$\varrho_{w_{śr},4s} > 2\varrho_{w_{śr},8s} .$$

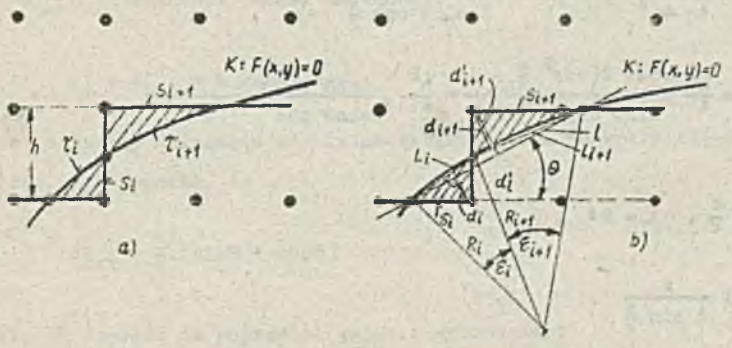
Ponieważ $\varrho_{w_{śr}}$ dyskretyzacji krzywych 2-stopnia zależy przede wszystkim od rzędu spójności siatki wielokątnej, to dla dowolnego n

$$\varrho_{w_{śr},4s} > \varrho_{w_{śr},8s} .$$

7.2. Średniogłobalna ocena jakości dyskretyzacji

Wprowadźmy miarę polową jakości dyskretyzacji krzywych na siatkach kwadratowych, będącą modułem pola między krzywą i aproksymującą ją łamaną przypadającym na jednostkowy odcinek krzywej /rys. 14a/:

$$\varrho_{P_i} \stackrel{df}{=} \frac{S_i}{\tau_i} .$$



Rys. 14. Względny błąd polowy dla siatki $G_{4s}(h^2)$

Twierdzenie 5

Dla dyskretyzacji krzywych K na siatce $G_{4s}(h^2)$, przy założeniu $R \geq 10h$, istnieje oszacowanie:

$$\frac{0.26\,h}{\cos\theta - 0.07\sin\theta} > \varrho_{P_i} > 0.2\,h\cos\theta, \quad 0 \leq \theta \leq \frac{\pi}{4} . \qquad /58/$$

Dowód

Rozważmy pola S_i oraz S_{i+1} z rys. 14 sprzężone ze sobą w ten sposób, że zmiana położenia krzywej K pociąga za sobą zwiększenie się jednego pola oraz równoczesne zmniejszanie się drugiego. Suma tych pól sprzężonych osiąga wartość minimalną, gdy ograniczające je segmenty krzywej mają jednakową długość. Wówczas jakiekolwiek przesunięcie równoległe krzywej K /nie powodujące jednak zmiany łamanej/ powoduje większy przyrost w polu zwiększonym niż ubytek w polu zmniejszonym /tj. powoduje zwiększenie się sumy pól sprzężonych/, przy stałej wartości sumy $\tau_i + \tau_{i+1}$ ograniczających te pola łuków.

Oznaczmy więc $\tau_i = \tau_{i+1} = \tau$. Zatem dla siatki $G_{45}(h^2)$ mamy:

$$S_i = \frac{l_i^2}{2} \sin\left(\theta + \frac{\varepsilon_{i+1}}{2}\right) \cos\left(\theta + \frac{\varepsilon_{i+1}}{2}\right) + \frac{1}{2}\left(\varepsilon_{i+1} - \sin \varepsilon_{i+1}\right) R^2 ,$$

$$S_{i+1} = \frac{l_{i+1}^2}{2} \sin\left(\theta - \frac{\varepsilon_i}{2}\right) \cos\left(\theta - \frac{\varepsilon_i}{2}\right) - \frac{1}{2}\left(\varepsilon_i - \sin \varepsilon_i\right) R^2 .$$

Ponieważ zmiana promienia krzywizny krzywej na odcinku leżącym w elementarnym polu siatki, ze względu na założenie $R \gg h$, jest $R \gg \Delta R$ /$\lim_{(h/R)\to 0}\left(\Delta R/R\right) = 0$/ i jest do pominięcia, więc $R_i = R_{i+1} = R$.

Stąd, ze względu na równość $\tau_i = \tau_{i+1} = \tau$, różnica między kątami ε_i oraz ε_{i+1} jest również do pominięcia, czyli $\varepsilon_i = \varepsilon_{i+1} = \varepsilon$, a więc i $l_i = l_{i+1} = l$. Zachodzą również związki:

$$l_i = \frac{a_i}{\sin\left(\theta + \frac{\varepsilon}{2}\right)} , \qquad l_{i+1} = \frac{h - a_i}{\sin\left(\theta - \frac{\varepsilon}{2}\right)} ,$$

więc, z uwagi na $l_i = l_{i+1} = l$, otrzymujemy:

$$a_i = \frac{h}{2}\left(1 + \frac{\operatorname{tg}\frac{\varepsilon}{2}}{\operatorname{tg}\theta}\right) , \qquad l = \frac{h}{2 \sin\theta \cos\frac{\varepsilon}{2}} ,$$

$$S_i + S_{i+1} = \frac{h^2}{4} \frac{\cos\theta\, 2(\cos^2\frac{\varepsilon}{2} - 1)}{\sin\theta \cos^2 \varepsilon/2} = \frac{h^2}{4} \frac{\cos\theta \cos\varepsilon}{\sin\theta \cos^2 \varepsilon/2} .$$

Ponieważ

$$l = 2R \sin\frac{\varepsilon}{2} , \qquad \tau = R\varepsilon ,$$

stąd

$$\tau = l \frac{\tau}{l} = l \frac{\varepsilon}{2 \sin\frac{\varepsilon}{2}} .$$

Natomiast

$$\varrho_{p_i} = \frac{S_i + S_{i+1}}{\tau_i + \tau_{i+1}} = \frac{S_i + S_{i+1}}{2l\frac{\tau}{l}} = \frac{h}{2} \frac{\cos\theta \cos\varepsilon \sin\frac{\varepsilon}{2}}{\varepsilon \cos \varepsilon/2} .$$

Zastępując sinusy i kosinusy kąta ε szeregami potęgowymi, oraz redukując licznik i mianownik przez czynnik $\frac{\varepsilon}{2}$, otrzymujemy zależności:

$$\varrho_{p_i} = \frac{h\cos\theta}{4} \cdot \frac{\sum_{n=0}^{\infty} \frac{(-1)^n (\varepsilon)^{2n}}{(2n)!} \cdot \sum_{n=0}^{\infty} \frac{(-1)^n \left(\frac{\varepsilon}{2}\right)^{2n}}{(2n+1)!}}{\sum_{n=0}^{\infty} \frac{(-1)^n \left(\frac{\varepsilon}{2}\right)^{2n}}{(2n)!}}.$$

Występujące w powyższym wyrażeniu szeregi potęgowe są szeregami przemiennymi, a jak wiadomo, w takich szeregach błąd sumy powstały z odrzucenia mniejszych wyrazów szeregu jest mniejszy bezwzględnie od pierwszego odrzuconego wyrazu i ma znak taki sam, jak ten wyraz. Przyjmijmy więc jako wartość sum szeregów tyle ich wyrazów, aby całe wyrażenie miało wartość zaniżoną. Znak równości w wyrażeniu na ϱ_{p_i} będziemy mogli wówczas zastąpić znakiem $>$. A więc, weźmy po dwa wyrazy z szeregów w liczniku /trzecie wyrazy obydwu szeregów mają znaki dodatnie, a więc przyjęte wartości sum szeregów są zaniżone/ oraz trzy wyrazy z szeregu mianownika /suma w mianowniku wzięta jest z nadmiarem/. Otrzymam wówczas nierówność na ϱ_{p_i} :

$$\varrho_{p_i} > \frac{h\cos\theta}{4} \cdot \frac{\left(1-\frac{\varepsilon}{2}\right)^2 \left(1-\frac{(\varepsilon/2)^2}{6}\right)}{1-\frac{1}{2}\left(\frac{\varepsilon}{2}\right)^2+\frac{1}{24}\left(\frac{\varepsilon}{2}\right)^4} = \frac{h\cos\theta}{4}\left(\frac{1-\frac{13}{24}\varepsilon^2+\frac{\varepsilon^4}{48}}{1-\frac{\varepsilon^2}{8}+\frac{\varepsilon^4}{384}}\right),$$

a po podstawieniu za ε wartości $\varepsilon_{max} \approx 0.14$, wyznaczonej w dowodzie twierdzenia 5.2, otrzymamy dolną granicę błędu polowego w aproksymacji na siatce $G_{4\theta}(h^2)$ w postaci:

$$\varrho_{p_i} > 0.89\, h\cos\theta .$$

Błąd maksymalny będzie wówczas, gdy odległość krzywej K i łamanej H będzie maksymalna. Dla siatki $G_{4\theta}(h^2)$ zachodzi jednak zależność:

$$d'_i + d'_{i+1} = h\cos\theta .$$

Natomiast z zasady działania algorytmu wybierającego na węzły ciągu aproksymacyjnego węzły z $\min \left|F(x_i,y_i)\right|$ wynika, że

$$d'_{max} = \frac{h}{2}(\sin\theta + \cos\theta) .$$

Stąd też, ze względu na poprzedni związek wyznaczamy

$$d'_{min} = \frac{h}{2}(\cos\theta - \sin\theta) .$$

A więc, jako błąd maksymalny wyznaczać będziemy błąd powstający w wypadku maksymalnej różnicy odległości kolejnych przegubów łamanej i krzywej. Jeżeli przyjmiemy, że

$$d_i = d_{max}, \text{ wówczas } d_{i+1} = d_{min}, \text{ oraz } b_{i+1} = \frac{d'_{max}}{\sin\theta} = \frac{1}{2}(\operatorname{ctg}\theta - 1),$$

$$a_i = \frac{d_{max}}{\sin\theta} = \frac{1}{2}\,\mathrm{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)\left(1 - \mathrm{ctg}\theta\right),$$

$$b_i = \frac{d_{max}}{\sin\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)} = \frac{1}{2}\left(1 + \mathrm{ctg}\theta\right),$$

oraz, z dokładnością do różnicy odcinków pól koła przylegających do obszarów pól błędu:

$$S_i + S_{i+1} = \frac{h^2}{8}\,\mathrm{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)\frac{1 + 2\sin\theta\cos\theta}{\sin^2\theta} + \frac{h^2\sin\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)}{8\sin^2\theta\cos\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)} - \frac{h^2}{4} =$$

$$= h^2\,\frac{\cos\frac{\varepsilon_{i+1}}{2} + \sin\frac{\varepsilon_{i+1}}{2}}{4\sin\theta\cos\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)}.$$

Wyrażenie na błąd połowy ϱ_{P_i} ma natomiast postać:

$$\varrho_{P_i} = \frac{\sin\frac{1}{2}\left(\varepsilon_i + \varepsilon_{i+1}\right)\left(\sin\frac{1}{2}\varepsilon_{i+1} + \cos\frac{1}{2}\varepsilon_{i+1}\right)h}{2\left(\varepsilon_i + \varepsilon_{i+1}\right)\cos\theta\cos\frac{1}{2}\varepsilon_{i+1} - \sin\theta\sin\frac{1}{2}\varepsilon_{i+1}}.$$

Dzieląc licznik i mianownik przez $\cos\frac{1}{2}\varepsilon_{i+1}$ a następnie zamieniając $\sin\left(\varepsilon_i + \varepsilon_{i+1}\right)$ pierwszym wyrazem szeregu potęgowego, a $\mathrm{tg}\frac{1}{2}\varepsilon_{i+1}$ pierwszymi trzema wyrazami szeregu po to, aby wartość całego wyrażenia przyjęta była z nadmiarem, otrzymujemy nierówność na ϱ_{P_i}:

$$\varrho_{P_i} \leq h\,\frac{1 + \frac{\varepsilon_{i+1}}{2} + \frac{\varepsilon_{i+1}^3}{24} + \frac{\varepsilon_{i+1}^5}{240}}{4\left(\cos\theta - \frac{\varepsilon_{i+1}}{2}\sin\theta\right)},$$

którego wartość zwiększa się, gdy ε się zwiększa /dla $0 \leq \theta \leq \frac{\pi}{4}$/, i z tego powodu, po zamianie ε_{i+1} na ε_{max} /wyznaczone dla siatki $G_{4\sqrt{3}}(h^2)$ w dowodzie Twierdzenia 4, możemy napisać nierówność jak w tezie:

$$\varrho_{P_i} \leq h\,\frac{0.268}{\cos\theta - 0.073\sin\theta}, \qquad 0 \leq \theta \leq \frac{\pi}{4}.$$

■

● Twierdzenie 6

Dla dyskretyzacji krzywych K na siatce $G_{3n}(h^2)$, przy założeniu $R \geq 10h$, istnieje oszacowanie:

$$\varrho_{P_i} > \left[0.2495\left(\cos\theta - \sin\theta\right) - 25 \cdot 10^{-4}\cos\theta\left(1 + \mathrm{ctg}\theta\right)\right]h$$

$$\varrho_{P_i} < \cos\theta\,\frac{1.05 + 0.95\,\mathrm{tg}\theta}{3.8 - 4.2\,\mathrm{tg}\theta}\,h, \qquad 0 \leq \theta \leq \mathrm{arctg}\frac{1}{2} \qquad /59/$$

Dowód

Rozważmy błąd minimalny /dolną granicę błędu/, który, podobnie jak w przypadku siatki $G_{4s}(h^2)$, wystąpuje wówczas, gdy przylegające do sprzężonych pól błędu łuki krzywej K mają jednakowe długości, czyli $\tilde{l}_i = \tilde{l}_{i+1} = \tilde{l}$ /rys.14/. Z założenia $R \gg h$ wynika, że $R \gg \Delta R$, tzn. że zmiana promienia krzywizny krzywej w elementarnym polu siatki jest do pominięcia. Wynikają stąd równości:

$$l_i = l_{i+1} = l \quad \text{oraz} \quad \varepsilon_i = \varepsilon_{i+1} = \varepsilon .$$

Dla siatki $G_{2s}(h^2)$ zachodzą związki:

$$\frac{a_i}{l} = \sin\theta\left(+\frac{\varepsilon}{2}\right)\cos\left(\theta + \frac{\varepsilon}{2}\right) - \sin^2\left(\theta + \frac{\varepsilon}{2}\right) ,$$

$$\frac{a_{i+1}}{l} = \sin\left(\theta - \frac{\varepsilon}{2}\right)\cos\left(\theta - \frac{\varepsilon}{2}\right) - \sin^2\left(\theta - \frac{\varepsilon}{2}\right), \qquad 0 \leq \theta < \frac{\pi}{4} .$$

Mamy zatem

$$l_i = \frac{a_i}{\sin\left(\theta + \frac{\varepsilon}{2}\right)} ,$$

a dla $0 \leq \theta < \operatorname{arc\,tg}\frac{1}{2}$

$$l_{i+1} = \frac{h - a_i}{\sin\left(\theta - \frac{\varepsilon}{2}\right)} .$$

Z równości $l_i = l_{i+1}$ wynika zatem

$$a = a_i = \frac{h}{2}\left(1 + \operatorname{ctg}\theta \operatorname{tg}\frac{\varepsilon}{2}\right) ;$$

a stąd

$$l = \frac{h}{2\sin\theta\cos\frac{\varepsilon}{2}} .$$

Ponieważ dla $0 \leq \theta < \operatorname{arc\,tg}\frac{1}{2}$ zachodzi $0 \leq a < h$, dlatego też we wzorze na a otrzymujemy nierówność $\theta > \frac{\varepsilon}{2}$ warunkującą wystąpienie pola minimalnego. Mamy zatem:

$$S_{\Sigma_i} = S_i + S_{i+1} = \frac{1}{2}(d_i + d_{i+1}) = \frac{(\sin\theta + \cos\theta)(\cos\theta\cos\varepsilon - \cos^2\frac{\varepsilon}{2})}{4\sin^2\theta\cos^2\frac{\varepsilon}{2}} \cdot h^2$$

stąd

$$\delta_{\Sigma_i} = \frac{\delta S_{\Sigma_i}}{S_{\Sigma_i}} = \frac{\delta S_{\Sigma_i}}{l\,\frac{\varepsilon}{\sin\frac{\varepsilon}{2}}} ,$$

gdzie $\tilde{l}_{\Sigma_i} = \tilde{l}_i + \tilde{l}_{i+1} = \dfrac{l\,\varepsilon}{\sin\varepsilon/2}$.

A więc

$$\varrho_{p_i} = \frac{h \sin\frac{\epsilon}{2}}{2\epsilon}\left[\cos\frac{\epsilon}{2}(\cos\theta - \sin\theta) - \frac{\sin^2\frac{\epsilon}{2}}{\cos\frac{\epsilon}{2}}\cos\theta(1 + \operatorname{ctg}\theta)\right].$$

Zamieniając sinusy i kosinusy kąta ϵ szeregami potęgowymi: taką liczbą wyrazów szeregu, aby wartość całego wyrażenia na ϱ_{p_i} była z nadmiarem, a więc wartości $\cos\frac{\epsilon}{2}$ - dwa wyrazy rozwinięcia /z niedomiarem/, wartości $\sin\frac{\epsilon}{2}$: w pierwszej części wyrażenia - 2 wyrazy /z niedomiarem/, w drugiej - 1 wyraz /z nadmiarem/, możemy napisać nierówność:

$$\varrho_{p_i} > \frac{\left(\frac{\epsilon}{2} - \frac{\epsilon^3}{48}\right)}{2\epsilon}\left[\left(1 - \frac{\epsilon^2}{8}\right)(\cos\theta - \sin\theta) - \frac{\frac{\epsilon^2}{2}}{1 - \frac{\epsilon^2}{8}}\cos\theta(1 + \operatorname{ctg}\theta)\right],$$

a po podstawieniu za ϵ wartości ϵ_{max}, wyznaczonej w dowodzie Twierdzenia 4 dla siatki $G_{8s}(h^2)$ i wynoszącej $\epsilon_{max} \approx 0.1$, otrzymamy wyrażenie na dolną granicę błędu polowego w aproksymacji na siatce $G_{8s}(h^2)$:

$$\varrho_{p_i} > \left[0.2495(\cos\theta - \sin\theta) - 35 \cdot 10^{-4}\cos\theta(1 + \operatorname{ctg}\theta)\right] \cdot h .$$

Błąd maksymalny powstanie wówczas, gdy wystąpuje maksimum odległości przegubu łamanej i krzywej czyli maksymalna wartość d_i bądź d_{i+1}. A więc jako błąd maksymalny wyznaczać będziemy błąd powstający w wypadku maksymalnej różnicy odległości kolejnych przegubów łamanej od krzywej.

Dla elementarnego pola siatki $G_{8s}(h^2)$ zachodzi zależność:

$$d_1' + d_2' = h \cdot \cos\theta \quad , \quad 0 \le \theta \le \frac{\pi}{4} .$$

Natomiast z zasady działania algorytmu wybierającego na wyjściu ciągu aproksymującego węzły w z nim $|F(x_i, y_i)|$ wynika, że

$$d'_{max} = \frac{h\cos\theta}{2} .$$

Również suma odległości sąsiednich przegubów łamanej od cięciwy spełnia zależność:

$$d'_{i+1} = \frac{h}{2}\cos\theta - h\sin\theta .$$

Mamy więc:

$$b_{i+1} = \frac{d'_{min}}{\sin\theta} = b_i\left(\frac{\operatorname{ctg}\theta}{2} - 1\right),$$

$$b_1 = \frac{d'_{max}}{\sin\theta} = \frac{h}{2}\operatorname{ctg}\theta .$$

$$a_i = b_i \frac{\sin\left(\theta + \frac{\epsilon_{i+1}}{2}\right)}{\cos\left(\theta + \frac{\epsilon_{i+1}}{2}\right) - \sin\left(\theta + \frac{\epsilon_{i+1}}{2}\right)} = \frac{h\cdot\operatorname{ctg}\theta}{2}\cdot\frac{\sin\left(\theta + \frac{\epsilon_{i+1}}{2}\right)}{\cos\left(\theta + \frac{\epsilon_{i+1}}{2}\right) - \sin\left(\theta + \frac{\epsilon_{i+1}}{2}\right)} .$$

Suma pól zawartych między łamaną i krzywą, z dokładnością do różnicy odcinków pól koła, przylegających do obszarów pól błędu, jest równa:

$$S_i + S_{i+1} = \frac{1}{2}\left[a_i(b_i - b_{i+1}) + h \cdot b_i\right] = \frac{h^2 \operatorname{ctg}\theta}{4} \cdot \frac{1 + \operatorname{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)}{1 - \operatorname{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)} .$$

Wyrażenie na błąd polowy ma natomiast postać:

$$\varrho_{p_i} = \frac{S_{\Sigma_i}}{\frac{\varepsilon_i + \varepsilon_{i+1}}{\sin\frac{1}{2}(\varepsilon_i + \varepsilon_{i+1})}} = \frac{h \cos\theta \cdot \sin\frac{1}{2}(\varepsilon_i + \varepsilon_{i+1})\left(1 + \operatorname{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)\right)}{2(\varepsilon_i + \varepsilon_{i+1})\left(1 - \operatorname{tg}\left(\theta + \frac{\varepsilon_{i+1}}{2}\right)\right)} =$$

$$= \frac{h \sin\frac{1}{2}(\varepsilon_i + \varepsilon_{i+1})\cos\theta}{2(\varepsilon_i + \varepsilon_{i+1})} \cdot \frac{\cos\theta \sin\left(\frac{\pi}{4} + \frac{\varepsilon_{i+1}}{2}\right) + \sin\theta \cos\left(\frac{\pi}{4} + \frac{\varepsilon_{i+1}}{2}\right)}{\cos\theta \cos\left(\frac{\pi}{4} + \frac{\varepsilon_{i+1}}{2}\right) - \sin\theta \sin\left(\frac{\pi}{4} + \frac{\varepsilon_{i+1}}{2}\right)} .$$

Po zamianie sinusów i kosinusów kąta ε pierwszymi wyrazami ich rozwinięć potęgowych po to, aby wartości sinusów i kosinusów wziąć z nadmiarem, otrzymamy wyrażenie:

$$\varrho_{p_i} < \frac{\cos\theta}{4} \cdot \frac{\left(1 + \frac{\varepsilon_{i+1}}{2}\right) + \left(1 - \frac{\varepsilon_{i+1}}{2}\right) \operatorname{tg}\theta}{\left(1 - \frac{\varepsilon_{i+1}}{2}\right) - \left(1 + \frac{\varepsilon_{i+1}}{2}\right) \operatorname{tg}\theta} \cdot h$$

które, po przyjęciu dla kąta ε wartości $\varepsilon_{max} \approx 0.1$, wyznaczonej w dowodzie Twierdzenia 4, przyjmuje postać:

$$\varrho_{p_i} < \cos\theta \cdot \frac{1.05 + 0.95 \operatorname{tg}\theta}{3.8 - 4.2 \operatorname{tg}\theta} \cdot h \qquad 0 \leqslant \theta \leqslant \operatorname{arc\,tg}\frac{1}{2} ,$$

a więc, jak w tezie. ■

Z udowodnionych twierdzeń 5 i 6 wynika, że ϱ_{p_i} zależy od trzech wielkości:

- promienia krzywizny krzywej R,
- rzędu spójności siatki RSG,
- kąta θ położenia krzywej w stosunku do linii siatki,

a wartość ϱ_{p_i} jest odwrotnie proporcjonalna do R i RSG oraz wprost proporcjonalna do θ.

Uśredniając błąd ϱ_{p_i} wzdłuż łuku krzywej zgodnie ze wzorem

$$\varrho_{sp} \overset{df}{=} \frac{\int \varrho_{p_i}\, ds}{\int ds} \qquad /60/$$

gdzie całki występujące we wzorze są całkami krzywoliniowymi, branymi wzdłuż łuku krzywej K, otrzymujemy nowy rodzaj błędu, który zależy od typów i parametrów krzywych. Wartość błędu ϱ_{sp}, jak łatwo zauważyć, będzie zależała od R, RSG i θ, które dla każdej krzywej są określone.

Z analizy wzorów na ϱ_{sp} dla krzywych 2-stopnia [63] wynika, że

$$\inf_{\substack{k \in \mathcal{K} \\ \omega \in [0,2\pi]}} \varrho_{sp,4s} = \sup_{\substack{k \in \mathcal{K} \\ \omega \in [0,2\pi]}} \varrho_{sp,8s} = 0.25h$$

5.3. Podstawowe wyniki oceny dokładności dyskretyzacji

Z przeprowadzonej analizy wyrażeń określających błędy dyskretyzacji krzywych algebraicznych n-stopnia na siatkach

$$G_{4s}(h^2) \quad i \quad G_{8s}(h^2)$$

wynika, że każdy z błędów: ϱ, ϱ_{w_1}, $\varrho_{w_{\delta r}}$, ϱ_{p_1}, ϱ_{sp}, zależy /w różnym stopniu/ od trzech podstawowych parametrów:

- R - promienia krzywizny krzywej /parametrów krzywej/,
- RSG - rzędu spójności siatki,
- θ - położenia krzywej w stosunku do linii siatki.

Z analizy tej wynika następujący generalny wniosek: niezależnie od typu błędu, każdy z rozważanych błędów jest większy dla siatki $G_{4s}(h^2)$ niż dla siatki $G_{8s}(h^2)$.

Aproksymacja na siatce $G_{8s}(h^2)$ jest więc dokładniejsza.

Nasuwa się więc wniosek, że w dyskretyzacjach na siatkach $G(h^2)$ dokładność dyskretyzacji wzrasta wraz ze wzrostem rzędu spójności siatki RSG, na której jest dokonywana dyskretyzacja.

6. ZAKOŃCZENIE

Celem prowadzonych badań było rozwinięcie teorii i uzyskanie wyników aplikacyjnych w dziedzinie całkowitoliczbowej aproksymacji krzywych algebraicznych płaskich. W świecie badania tego typu są prowadzone od kilkunastu lat; uzyskano wiele pozytywnych wyników w zakresie algorytmów dyskretyzacji prostej i krzywych 2-stopnia. Badania niniejsze rozwijają i udoskonalają algorytmy całkowitoliczbowej aproksymacji prostej i krzywych 2-stopnia na całą klasę krzywych algebraicznych płaskich. W ramach tych badań przeprowadzono również ocenę dokładności aproksymacji w mierze lokalnej i globalnej oraz ocenę złożoności obliczeniowej algorytmów w ich realizacji komputerowej.

Różnorodność w zbiorze możliwych rozwiązań tak postawionego zadania została ograniczona nałożeniem na algorytmy aproksymacji istotnych ograniczeń:

- działanie jedynie na liczbach całkowitych,
- niewystępowanie zjawiska kumulacji błędów,
- stosowanie jedynie prostych operacji algebraicznych /arytmetycznych całkowitoliczbowych lub elementarnych operacji boolowskich/,
- prostota obliczeniowa /duża szybkość działania w realizacji komputerowej/.

6.1. Podstawowe wyniki badań

Omówione w pracy badania są analizą zagadnienia dyskretyzacji krzywych algebraicznych płaskich na jednorodnych siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$ oraz syntezą algorytmów dyskretyzacji tych krzywych metodą podejścia warunkowego, w której moduł wartości funkcji $F(x_i,y_i)$ został przyjęty za miarę odległości punktu $<x_i,y_i>$ od krzywej /metryka aproksymacyjna/.

Zaprezentowane algorytmy są rozwinięciem i uogólnieniem znanych z literatury algorytmów dyskretyzacji prostej i krzywych 2-stopnia, a szczególnie algorytmów, które autor niniejszej rozprawy opisał w pracy z 1977 r. /algorytmy aproksymacji krzywych 2-stopnia na siatkach $G_{4s}(h^2)$ i $G_{8s}(h^2)$/, na całą klasę krzywych algebraicznych płaskich, w tym i wielogałęziowych.

W ramach prowadzonych badań oszacowano również szczytowe i polowe błędy algorytmów. W szczególności wyprowadzone zostało rownanie /46/, z którego można wyznaczyć kres górny bezwzględnego szczytowego błędu aproksymacji jako funkcję stopnia krzywej. Oszacowane względne szczytowe błędy aproksymacji /wzory /54/ i /55// są funkcjami rzędu spójności siatki i w zasadzie nie zależą od stopnia krzywej /promienia krzywizny i parametrów geometrycznych krzywej/.

Przeprowadzona przez autora globalna ocena jakości aproksymacji doprowadziła do wzorów na moduł pola przylegającego do jednostkowego odcinka krzywej, oraz na jego wartość średnią, powstałą z uśrednienia modułu pola wzdłuż konturów krzywych.

Wyniki oceny szczytowych i polowych błędów aproksymacji krzywych algebraicznych płaskich stanowią uogólnienie oszacowań błędów dyskretyzacji krzywych algebraicznych płaskich 2-stopnia /wykonanych i opisanych przez autora rozprawy w 1977 r./ na całą klasę krzywych algebraicznych płaskich.

Oszacowana została także złożoność obliczeniowa dyskretyzacji jako funkcja stopnia krzywej, rodzaju siatki i wielkości jej modułu oraz sposobu wyznaczenia wartości funkcji w węzłach siatki.

Szczególnie duży wpływ na szybkość dyskretyzacji ma stopień krzywej oraz wielkość modułu siatki h.

Tak jednoznaczne i precyzyjne metody oceny złożoności obliczeniowej /szybkość działania/ do oceny algorytmów dyskretyzacji krzywych nie były dotychczas stosowane /brak podobnych opracowań w literaturze/. Autor sądzi jednak, że dokładne wyznaczenie szybkości działania algorytmów dyskretyzacyjnych jest niezbędne do właściwej oceny ich możliwości aplikacyjnych.

Wyniki przeprowadzonych badań mają aspekty, zarówno poznawcze, jak też aplikacyjne.

Osiągnięte i sformułowane w postaci sześciu twierdzeń z dowodami wyniki są przyczynkiem do teorii aproksymacji krzywych algebraicznych. Stanowią zasadniczą treść teorii dyskretyzacji krzywych, są rozwinięciem teorii aproksymacji krzywych w nowym kierunku, który jest nazywany "aproksymacją całkowitoliczbową" bądź też "aproksymacją na dwuwymiarowych siatkach". Stworzone zostały ulepszone i uproszczone metody tej aproksymacji oraz ich ocena pod względem dokładności, jak też złożoncści obliczeniowej algorytmów.

Aspekty aplikacyjne przeprowadzonych badań są nie mniej znaczące, jak aspekty poznawcze. Opracowane algorytmy upraszczają proces aproksymacji krzywych algebraicznych oraz stwarzają nowe możliwości. Mają bowiem cechę, której brakuje algorytmom znanym wcześniej: nie kumulują błędów. Pozwalają także na rozwiązywanie zadań aproksymacji krzywych łatwiej i szybciej. Dotyczy to

w szczególności dyskretyzacji krzywych wielosłęziowych, oceny dokładności i złożoności obliczeniowej algorytmów, pozwalającej stosować je optymalnie, zarówno ze względu na dokładność, jak i wymagane środki sprzętowe oraz czas ich wykonywania.

6.2. Przewidywane zastosowania

Zaproponowane algorytmy dyskretyzacji zostały zsyntetyzowane pod kątem potrzeb grafiki komputerowej i numerycznego sterowania obrabiarkami. Autor sądzi jednak, że znajdą zastosowanie w wielu różnych dziedzinach nauki i techniki: w grafice komputerowej, w szczególności w monitorach graficznych, kreślakach czy drukarkach graficznych - do konstrukcji układowych bądź programowych interpolatorów i generatorów krzywych i funkcji; w koderach rysunków /dyskretyzatorach/ - do dyskretyzacji konturów, obrysów i dokumentacji rysunkowej wprowadzanej do komputerów; w mapach konturowych i projektowaniu inżynierskim - w zadaniach dopasowania kształtów /modelowanie profilowanych, opływowych kształtów, zwłaszcza w przemyśle okrętowym i lotniczym, czy nawet w architekturze/, a także doborze optymalnych wzorników /szkrobników/, czy też w układaniu zabaw mozaikowych.

Algorytmy to - wydaje się - znajdą zastosowanie również w modelowaniu i przetwarzaniu rysunków liniowych i obrazów, w graficznych procesorach obrazów /procesory kwantujące sekwencyjne dane i rozkazy graficzne na postać komórkową, procesory kompresji obrazu/, do opisu obrazów w zadaniach rozpoznawania szablonów, czy do analizy pisma odręcznego /grafologia/.

Ponadto algorytmy i zastosowane w nich metody mogą znaleźć także zastosowanie w konstrukcji interpolatorów dla obrabiarek sterowanych numerycznie i w technologicznych programach generacji krzywych dla celów interpolowania profilowanych powierzchni, numerycznym sterowaniu procesami technologicznymi, czy w konstrukcji specjalnych jednostek decyzyjno-sterujących do sterowania ruchem błażące, zwłaszcza procesami szybkozmiennymi.

Algorytmy i zastosowane w nich metody mogłyby także znaleźć zastosowanie w zupełnie odmiennych dziedzinach nauki i techniki: w fizyce cząstek wysokiej energii do identyfikacji torów cząstek elementarnych, w teorii obliczeń współbieżnych czy lingwistyce matematycznej /modelowanie języków formalnych/, a nawet w modelach mechanizmów neuropsychologicznych i teorii sieci nerwowych.

Zastosowana metodyka oceny błędów aproksymacji może być też wykorzystana np. przy określaniu tolerancji obróbki w programach technologicznych sterowania obróbką detali, gdzie wymagana jest wysoka /ściśle określona/ tolerancja obróbki profilu, czy też w cyfrowym sterowaniu innymi procesami technologicznymi, w metrologii i systemach automatycznego sterowania do oceny dyskretyzacji przebiegów ciągłych itp.

Przy wyborze algorytmów do określonego zastosowania należy brać pod uwagę przede wszystkim dwa czynniki: złożoność obliczeniową /mierzoną liczbą wykonywanych operacji przypadających na jednostkę długości krzywej/ oraz efektywną jakość /ocenianą wizualnie lub mierzoną w jednej z zastosowanych w rozprawie metryk/.

6.3. Określenie kierunku dalszych badań

Uzyskane przez różnych badaczy wyniki, jak również i te, które prezentowane są w niniejszej rozprawie, nie wyczerpują listy nierozwiązanych problemów w dyskretyzacji na siatkach w ogóle, a w szczególności na siatkach kwadratowych. Wiele problemów czeka jeszcze na pilne rozwiązanie. Dotyczy to szczególnie zagadnienia operowania na wielkich liczbach w warunkowej dyskretyzacji krzywych algebraicznych na siatkach kwadratowych, w których $|F(x,y)|$ jest metryką generacyjną.

Za celowe należy uznać podjęcie badań nad strukturalnymi dyskretyzacjami krzywych na jednorodnych siatkach kwadratowych, a także nad dyskretyzacjami krzywych na siatkach kwadratowych niejednorodnych oraz na siatkach innych niż kwadratowe /trójkątne, sześciokątne, koncentryczne/.

Pilnym do rozwiązania problemem jest też dyskretyzacja na siatkach przestrzennych /szczególnie dla potrzeb grafiki komputerowej i numerycznego sterowania obrabiarkami/, przede wszystkim na jednorodnych siatkach przestrzennych, ale również i na siatkach krzywoliniowych /koncentrycznych/.

Na rozwiązanie czeka również zagadnienie całkowitliczbowej aproksymacji powierzchni zarówno tych najprostszych /2-stopnia algebraicznych/, jak też tych bardziej złożonych, których niezbędność w grafice komputerowej i numerycznym sterowaniu obrabiarkami jest coraz bardziej oczywista.

LITERATURA

1963

[1] Stockton F.G.: Algorithm 162, XY move plotting. Communications of the ACM 1963 vol. 6 nr 2 s.161

1965

[2] Bresenham J.E.: Algorithm for computer control of a digital plotter. IBM System Journal 1965 vol.4 nr 1 s. 25-30

1967

[3] Pitteway M.L.V.: Algorithm for drawing ellipses or hyperbolas with a digital plotter. The Computer Journal 1967 vol.10 nr 3 s. 282-289

1968

[4] Botting R.J., Pitteway M,L.V.: Algorithm for drawing elipses or hyperbolas with a digital plotter. The Computer Journal 1968 vol. 11 nr 2 s.120

[5] Morse S.P.: A mathematical model for the anelisis of contour line data. Journal of the ACM 1968 vol.15 nr 2 s.205-220

[6] Montanari U.: A metrod for obtaining skeletons using a quasi-euclidean distance. Journal of the ACM 1968 vol. 15 nr 4 s. 600-624

[7] Partridge M.F.: Algorithm for drawing ellipses of hyperbolas with a digital plotter. The Computer Journal 1968 vol.11 nr 2 s. 119-120

1969

[8] Freeman H.: A review of relevant problems in the processing of line-drawing data. Automatic Interpretation and Classification of Images. New York: Grasselli A. 1969

[9] Metzger R.A.: Computer generated graphic segments in a raster display. Proceeding of the 1969 Spring Joint Computer Conference

[10] Rosenfeld.: Picture processing by Computer. London: Academic Press 1969

1970

[11] Froeman H.: Boundary encoding and processing. Picture Processing and Psychopictorics. London: Lipkin B.S. and Rosenfeld A.1970

[12] Danielsson P.E.: Incremental curve generation. IEEE Transactions on Computers 1970 vol. C-19 nr 9 s. 783-793

[13] Montanari U.: A note on minimal lenght polygonal approximation to a digitizet contour. Communications of the ACM 1970 vol. 13 nr 1 s. 41-47

[14] Montanari U.: On limit properties in digitization schemes. Journal of the ACM 1970 vol. 17 nr 2 s. 348-360

[15] Rosenfeld A.: Connectivity in digital pictures. Journal of the ACM 1970 vol. 17 nr 1 s.146-160

[16] Poljakov V.G.: Estestvennaja diskretnaja model' risunka, nekotoryje asimptotičeskie ee svojstva i predskazanie choda sledjaščej razviortki. Problemy peredači informacji 1970 t. IV vyp.3

1972

[17] Lappalainen P., Ojala L.: A nowel digital method generating a circle test pattern for television. The Radio and Electronic Engineer 1972 vol. 42 nr 1 s. 21-27

[18] Reggiori G.B.: Digital computer transformations for irregular line drawings. Technical Report 403-22 New York University april 1972

1973

[19] Jordan B.W., Holm B.D., Lennon W.J.: An improved algorithm for the generation of nonparametric curves. IEEE Transactions Computers 1973 vol. C-22 nr 12 s. 1052-1060

[20] Denert E.: A method for computing points of a circle using only integers. Computer Graphics and Image processing 1973 vol. 2 nr 1 s. 83-91

[21] Newman W.M., Sproull R.F.: Principles of Interactive Computer Graphics. New York: McGraw-Hill 1973 ss. 607

[22] Rosenfeld A., Johnston E.: Angle detection on digital curves. IEEE Transactions on Computers 1973 vol. C-22 nr 9 s. 875-878

[23] Rosenfeld A.: Arcs and curves in digital pictures. Journal of the ACM 1973 vol.20 nr 1 s.81-87

1974

[24] Athani V.V.: The 16-vector algorithm for computer controlled digital X-Y plotter. IEEE Transactions on Computers 1974 vol. C-24 nr 8 s.831-835

[25] Bornhill i Riesenfeld, Ed.: Computer Aided Geometric Desing. New York: Academic Press 1974

[26] Brons R.: Linquistic methods for the description of a straight line on a grid. Computer Graphics and Image Processing 1974 vol. 3 nr 1 s. 48-62

[27] Chaikin G.M.: Algorithm for high-speed curve generations. Computer Graphics and Image Processing 1974 vol. 3 nr 4 s. 346-349

[28] Freeman H.: Computer processing of line-drawing images. Computing Surveys 1974 vol.6 nr 1 s. 57-97

[29] McLain D.H.: Drawing contours from arbitrary data points. The Computer Journal 1974 vol. 17 nr 4 s. 318-324

[30] Pitteway M.L.V., Botting R.J.: "Integer circles, etc." - Thee move extension of Bresenham's algorithm. Computer Graphics and Image Processing 1974 vol. 3 nr 2 s. 260-261

[31] Pitteway M.L.V.: Integer circles, etc. - Some fourther thinghts. Computer Graphics and Image Processing 1974 vol. 3 nr 3 s. 262-265

[32] Reumann K., Witkam A.P.M.: Optimizing curve segmentation in computer graphics. International Computing Symposium 1973 Konstanz: Günter A. i in. North-Holland Publ. Co. 1974 s.467-472

[33] Rosenfeld A.: A note on perimetr in digital pictures. Information and Control 1974 vol. 24 nr 4 s. 384-388

[34] Rosenfeld A.: Adjacency in digital pictures. Information and Control 1974 vol. 26 nr 1 s.24-33

[35] Stomopoulos C.D.: Parallel algorithms for joining two point by a straight-line segment. IEEE Transactions on Computers 1974 vol. C-22 nr 6 s. 642-646

1975

[36] Arcelli C., Massarotti A.: Regular arcs in digital contours. Computer Graphics and Image Processing 1975 vol. 4 nr 2 s. 339-360

[37] Bongiovanni G., Luccio F., Zorat A.: The discrete equation of the straight line. IEEE Transactions on Computers 1975 vol. C-24 nr 3 s. 310-313

[38] Komeswara R.C.F.: Comment on a method for computing points of a circle using only integers. Computer Graphics and Image Processing 1975 vol. 4 nr 1 s. 79

[39] Klaasman H.: Some aspects of the accuracy of the approximated position of a straight line on a square grid. Computer Graphics and Image Processing 1975 vol. 4 nr 3 s. 225-235

[40] Mokrzycki W.: Przyrostowa generacja krzywych drugiego stopnia w systemach obrazowania informacji, Biuletyn WAT 1975 nr 9/277 s. 73-77

[41] Riesenfeld R.F.: On Chaikin's algorithm. Computer Graphics and Image Processing 1975 vol. 4 nr 4 s. 304-310

[42] Rosenfeld A.: A converse do the Jordan curve theorem for digital curves. Information and Control 1975 vol. 29 nr 2 s. 292-293

[43] Rosenfeld A., Weszka J.: An improved method of angle detection on digital curves. IEEE Transactions on Computers 1975 vol. C-24 nr 9 s. 940-941

[44] Walukiewicz S.: Prawie liniowe zadanie programowania dyskretnego. Prace Instytutu Organizacji i Kierowania 1975 z.23

[45] Ziemkiewicz A.: Wybrane procedury graficzne w systemach komputerowych. Prace IBM 1975 z.2 s.43-62

1976

[46] Bolser K.: Comment on "An improved algorithm for the generation on nonparametric curves". IEEE Transactions on Computers 1976 vol. C-25 nr 1 ss.103

[47] Horn W.K.P.: Circle generators for display devices. Computer Graphics and Image Processing 1976 vol. 5 nr 2 s. 280-228

[48] Morrin T.H.: Chain-link compression of arbitrary black-white images. Computer Graphics and Image Processing 1976 vol. 5 nr 2 s. 172-189

[49] Ramot J.: Nonparametric curves. IEEE Transactions on Computers 1976 vol. C-25 nr 1 s.103-104

[50] Rogers D., Adams J.: Mathematical Elements for Computer Graphics. New York: McGraw-Hill 1976 ss. 239

[51] Rothstein J., Weiman C.: Parallel and sequential specification of a context sensitive language for straight lines on grids. Computer Graphics and Image processing 1976 vol. 5 nr 1 s.106-124

[52] Rubin F.: Generation of nonparametric curves. IEEE Transactions on Computers 1976 vol. C-25 nr 1 s. 103

[53] Sutcliffe D.C.: An algorithm for drawing the curve $f(x,y)= 0$, The Computer Journal 1976 vol. 19 nr 3 s. 246-249

[54] Sutcliffe D.C.: A remark on a contouring algorithm. The Computer Journal 1976, vol. 19 nr 4 s. 333-335

1977

[55] Badler N.I.: Disk generators for a raster display device. Computer Graphics and Image Processing 1977 vol. 6 nr 4 s. 589-593

[56] Bresenham J.E.: A linear algorithm for incremental digital display of circular arcs. Communications of the ACM 1977 vol. 20 nr 2 s. 100-106

[57] Chung W.L.: On circle generation algorithms. Computer Graphics and Image Processing 1977 vol.6 nr 1 s.196-198

[58] Freeman H., Davis L.: A corner-finding algorithm for chain-coded curves. IEEE Transactions on Computers 1977 vol. C-26 nr 3 s. 297-303

[59] Gaafar M.: Convexity verification, block-chords, and digital straight line. Computer Graphics and Image Processing 1977 vol. 6 nr 4 s. 361-370

[60] Mokrzycki W.: Algorytmy i ocena dokładności układowych generacji krzywych 2-stopnia na siatkach kwadratowych. Biuletyn Informacyjny OSK 1977 R.XVI nr 5-6 s.89-109

1978

[61] Arcelli C., Massarotti A.: On thr parallel generation of straight digital lines. Computer Graphics and Image Processing 1978 vol. 7 nr 1 s. 67-83

[62] Freeman H.: Shape description via the use of critical points. Pattern Recognition 1978 vol.11 nr 3 s. 159 - 166

[63] Groen F.C.A., Verbeek P.W.: Freeman - code probabilities of object bondary quantized contours. Computer Graphics and Image Processing 1978 vol. 7 nr 4 s. 391-402

[64] Mokrzycki W.: Analiza cyfrowych metod generacji układowej krzywych drugiego stopnia. Prace naukowo-badawcze IBM 1978 z.2

[65] Paszkowski S.: Optymalny algorytm kreślenia okręgu. Raport nr N-44, Instytut Informatyki Uniwersytetu Wrocławskiego, 1978

[66] Pratt W.K.: Digital Image Processing, A Wiley-Interscience Publication John Wiley and Sons, New York /Chichester/Brisbane/Toronto 1978

[67] Rosenfeld A.: Geodesics in digital pictures. Information and Control 1978 vol. 36 nr 1 s.74-84

[68] Sankar P., Sharma C.: A parallel procedure for the detection of dominant points on the digital curve. Computer Graphics and Image Processing 1978 vol. 7 nr 3 s. 403-412

[69] Sankar P.V.: Grid intersect quantization schemes for solid object digitalization. Computer Graphics and Image Processing 1978 vol. 8 nr 1 s. 25-42

[70] Ward A.S.: Real time plotting of approximate contour maps. Communications of ACM 1978 vol. 21 nr 9 s. 788-790

1979

[71] Coderberg R.L.T.: A new method for vector generation. Computer Graphics and Image Processing 1979 vol. 9 nr 2 s. 183-195

[72] Freeman H.: Algorithm for generating a digital straight line on a triangular grid. IEEE Transactions on Computers 1979 vol. C-28 nr 2 s. 150-152

[73] Horn B.K.P.: Comments on: "A note on the paper by B.K.P. Horn" by Zenon Kulpa. Computer Graphics and Image Processing 1979 vol. 9 nr 1 s. 104

[74] Kulpa Z.: A note on the paper by B.K.P. Horn: "Circle generators for display devices". Computer Graphics and Image Processing 1979 vol. 9 nr 1 s. 102-103

[75] Lindner R.: A processor for real time TV raster scan conversion. Computer and Graphics 1979 vol. 4 nr 1 s. 23-28

[76] Maxwell P.C., Baker P.W.: The generation of polygons representing circles, elipses and hyperbolas. Computer Graphics and Image Processing 1979 vol. 10 nr 1 s. 84-93

[77] Moss R., Lindgand A.: Parametric spline curves in integer arithmetic designed for use in microcomputer controlled plotters. Computers and Graphics 1979 vol. 4 nr 1 s. 51-61

[78] Weiman C.F.R., Chaikin C.: Logaritmic spiral grids for image processing and display. Computer Graphics and Image Processing 1979 vol. 11 nr 2 s.197-226

1980

[79] Beetem J.: Vector graphics for raster display. BYTE Publications inc. October 1980 s. 286-293

[80] Chow W.W.: Interlocking shapes in art and engineering. Computer Aided Design 1980 vol. 12 nr 1 s. 29-34

[81] Coneignoux Ph., Guedj R.: Computer Generation of Colored Planar Patterns on TV-like Rasters. Proceeding of the IEEE 1980 vol. 68 nr 7 s. 753-944

[82] Danielsson P.E.: Euclidean distance mapping. Computer Graphics and Image Processing 1980 vol. 14 nr 3 s. 227-248

[83] Earnshaw R.A.: Line trocking for incremental plotters. The Computer Journal 1980 vol. 23 nr 1 s. 46-52

[84] Frank A.J., Daniels J.D., Unangst D.R.: Progressive image transmission using a growth-geometry coding. Proceedings of the IEEE 1980 vol. 68 nr 7 s. 897-909

[85] Freeman H., Saghri J.A.: Comparative analysis of line-drawing modeling schemes. Computer Graphics and Image Processing 1980 vol. 12 nr 2 s. 203-223

1981

[86] Böhm W.: Generating the Bézier points of B-spline curves and surfaces. Computer Aided Design 1981 t. 13 nr 6 s. 365-366

[87] Charkavarty I.: A sinle-pass, chain generating algorithm for region boundaries. Computer Graphics and Image Processing 1981 t. 15 nr 2 s. 182-193

[88] Gossling T.H.: Bulge, shear and squash: a representation for the general conic arc. Computer Aided Design 1981 t. 13 nr 2 s. 81-84

[89] Oliviero A., Scarpetta G.: A new aproach to contour coding. Computer Graphics and Image Processing 1981 t. 15 nr 1 s. 87-92

[90] Sebok T.J., Roemer L.E., Malindrok G.S. jr: An algorithm for line intersection identification. Pattern Recognition 1981 t. 13 nr 2 s. 159-166

[91] Wang C.Y.: Shape classification of the parametric cubic curves and parametric B-spline cubic curve. Computer Aided Design 1981 t.13 nr 4 s. 199-206

1982

[92] Aleliunas R., Rosenberg A.L.: On embedding rectangular grids in square grids. IEEE Transaction on Computers 1982 t. C-31 nr 9 s. 907-913

[93] Havada K., Nakomae E.: Application of the Bézier curve to data interpolation. Computer Aided Design 1982 t. 14 nr 1 s. 55-59

[94] Kim Ch. E.: On cellular Straight line segments. Computer Graphics and Image Processing 1982 t. 18 nr 4 s. 369-381

[95] Lee H.C., Fu K.S.: Using the FFT to determine digital straight line chain codes Computer Graphics and Image Processing 1982 t. 18 nr 3 s. 359 - 368

[96] Mokrzycki W.: Dyskretyzacje krzywych algebraicznych na jednorodnych kwadratowych siatkach k-spójnych. Prace naukowo-badawcze IMM 1982 nr 2 s. 3-41

[97] Sharpe R.J., Thorne R.W.: Numerical method for extracting an are lenght parametrization from parametric curves. Computer Aided Design 1982 t. 14 nr 2 s. 79-81

1983

[98] Hölzle G.E.: Knot placement for piecevise polynomial approximation of curves. Computer Aided Design 1983 t. 15 nr 5 s. 295-296

[99] Kulpa Z.: More about areas and perimeters of quantized objects. Computer Vision, Graphics and Image Processing 1983 t. 22 nr 3 s. 268-276

[100] Mokrzycki W.: Całkowitoliczbowe dyskretyzacje krzywych algebraicznych na jednorodnych siatkach kwadratowych. Prace naukowo-badawcze IMM 1983 nr 3 s. 2-92

[101] Sarvarayndu G.P.R., Settli J.K.: Walsh descriptors for polygonal curves. Pattern Recognition 1983 t. 16 nr 3 s. 327-336

1984

[102] Doros M.: On some properties of the generation of discrete circular arcs on a square grid. Computer Vision, Graphics, and Image Processing 1984 t. 28 nr 4 s. 377-383

[103] Edelsbumner H., Overmars H.M., Seidel R.: Some methods of computational geometry applied to computer graphics. Computer Vision, Graphics, and Image Processing 1984 t.28 nr 1 s. 92-108

[104] Lee R.B., Fredricks D.A.: Intersection of parametric surfaces on a plane. IEEE Computer Graphics and Applications 1984 t.4 nr 8 s. 48-51

[105] Lester L.W., Sandor J.: Computer graphics on a hexagonal grid. Computer and Graphics 1984 t. 8 nr 4 s. 401 - 409

[106] Mokrzycki W.: O pewnej metodzie dyskretyzacji algebraicznych krzywych przestrzennych na jednorodnych siatkach prostokątnych. Prace naukowo-badawcze IMM 1984 nr 1 s. 3-40

[107] Nakamura A., Aizava K.: Digital circles. Computer Vision, Graphics and Image Processing 1984 t. 26 nr 2 s. 242-255

[108] Piogl L.: Defining C^1 curves containing conic segments. Computer and Graphics 1984 t.8 nr 2 s. 177-182

[109] Van Wyk Ch. J.: Clipping to the boundary of a circular arc poligon. Computer Vision, Graphics and Image Processing 1984 t. 25 nr 4 s. 383-392

Dodatek

Implementacja algorytmów

W celu zweryfikowania opracowanych w rozdz.3 algorytmów dyskretyzacji krzywych algebraicznych, tzn. sprawdzenia poprawności ich działania, porównania uzyskanych wyników z założeniami na dyskretyzację i ujawnienia innych ich cech niż te, które były przewidziane na etapie ich syntezy, a także do wizualnej oceny krzywych dyskretnych generowanych przez te algorytmy, zaimplementowano je na komputerach IRIS 80, MERA 400 oraz SM-4 w języku FORTRAN, z wyprowadzeniem wyników na drukarkę, rastrowy monitor graficzny oraz kreślak.

Wiele różnych przebiegów programów realizujących obydwa algorytmy / z różnymi typami, stopniami i kierunkami obiegów krzywych oraz wielkością modułu siatki h/ świadczy o poprawności ich konstrukcji.

Jako ilustrację tych badań zamieszczamy dyskretne postaci 26 różnych typów krzywych /3-, 4- i 6-stopnia/, z których każda jest generowana za pomocą każdego z algorytmów z modułem siatki h=1[mm] oraz za pomocą bądź to algorytmu na siatce $G_{4s}(h^2)$ bądź $G_{8s}(h^2)$ z modułem siatki h=0.25[mm]. Moduł siatki h= 1[mm] został tak dobrany, aby /dla określonych parametrów geometrycznych krzywych/ zachować kształty geometryczne tych krzywych i równocześnie uwidocznić istotną różnicę w "gładkości" łamanych generowanych za pomocą każdego z algorytmów. Moduł h=0.25[mm] zapewnia dostateczną dla szerokiej klasy zastosowań dokładność zobrazowania, szczególnie na monitorach graficznych, niezależnie /w zasadzie/ od rzędu spójności siatki, chociaż pozwala /jeszcze/ dostrzec różnice

w "gładkości" łamanych generowanych przez każdy z algorytmów. Na przedstawionych rysunkach widać istotną różnicę w "gładkości" wygenerowanych łamanych na korzyść krzywych wygenerowanych na siatce o wyższym rzędzie spójności. Stwierdzić można również wyraźną zależność między "gładkością" krzywej, a promieniem krzywizny tej krzywej oraz wzajemnym położeniem siatki i krzywej /szczególnie w krzywych generowanych na siatce $G_{4a}(h^2)$/. "Gładkość" krzywych jest większa w tych rejonach krzywej, w których jest mniejsza krzywizna krzywej oraz kąt /liczony mod. 90^0/ między styczną do krzywej i głównymi liniami siatki.

W wyniku omawianej komputerowej realizacji algorytmów uwidoczniła się pewna negatywna strona zasady wykonywania operacji algorytmów na liczbach całkowitych. Dla krzywych wyższych stopni i określonych wymiarów geometrycznych krzywych, w niektórych rejonach tych krzywych - wartości pochodnych cząstkowych przyjmują duże wartości liczbowe. Wymaga to długiego słowa rozkazowego w komputerze. W przeciwnym razie następuje przepełnienie rejestrów i przekłamania w obliczeniach, doprowadzające w rezultacie do "degeneracji " krzywej. W komputerze MERA 400 zakres liczb całkowitych /$\pm$ 32767/ był przekraczany w trakcie generacji krzywych stopnia $n \geq 4$ i wymiarów geometrycznych krzywych spoza zakresu liczbowego - 40 * 40 jednostek. Jeśli istnieje obawa, że opisany stan może zaistnieć, należy wówczas wykonywać obliczenia w reprezentacji zmiennoprzecinkowej. Co prawda, ta reprezentacja wprowadza z natury swojej pewne błędy /błędy reprezentacji/. Poza tym eliminuje się /w znacznym stopniu/ w ten sposób podstawowe zalety stosowanej metody dyskretyzacji, tj. szybkość obliczeń i niewystępowanie kumulacji błędów.

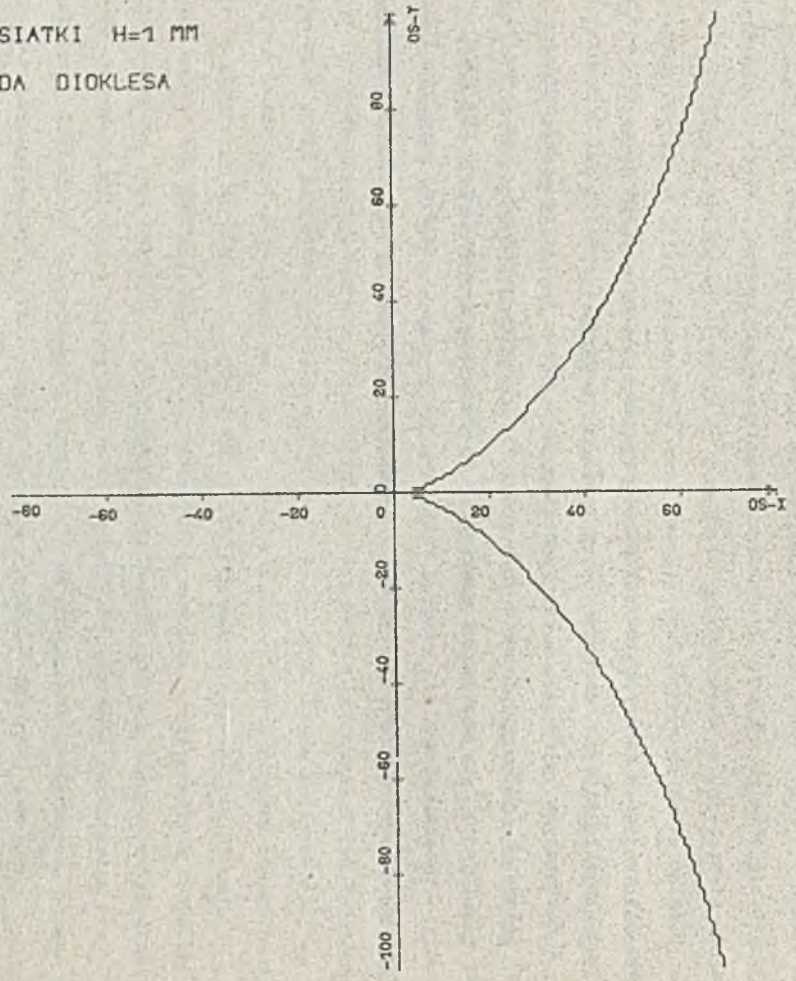
Negatywną cechą metody będącej podstawą proponowanych algorytmów jest więc operowanie na dużych liczbach całkowitych /dużych w stosunku do zakresu liczb całkowitych minikomputerów/, zwiększających się wraz ze stopniem i wymiarami geometrycznymi krzywej. Błędy reprezentacji są jednak z reguły niewielkie, a "degeneracja" następuje po wyznaczeniu kilkuset punktów ciągu aproksymującego. Dzieląc zatem dyskretyzowany segment krzywej na pewną liczbę krótszych podsegmentów i wyznaczając dla każdego z nich /poprawnie/ dane początkowe, można uzyskać aproksymację zadowalającą, zarówno z punktu widzenia dokładności, jak również szybkości.

W testowanych przykładach krzywych stopnia $n \leq 6$ i wymiarach geometrycznych krzywych rzędu $\pm 10^4$ jednostek /jedna jednostka = 1h - moduł siatki/ zakres liczbowy pojedynczej precyzji wynoszący $\pm 0.17 \cdot 10^{39}$ w zupełności wystarczał do poprawnego wykonywania niezbędnych w algorytmach operacji. Nie stwierdzało się również "degeneracji" krzywych z powodu stosowania arytmetyki zmiennoprzecinkowej /pojedynczej precyzji/, mimo wnoszonych przez tę reprezentację błędów.

APROKSYMACJA KRZYWYCH 3-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM

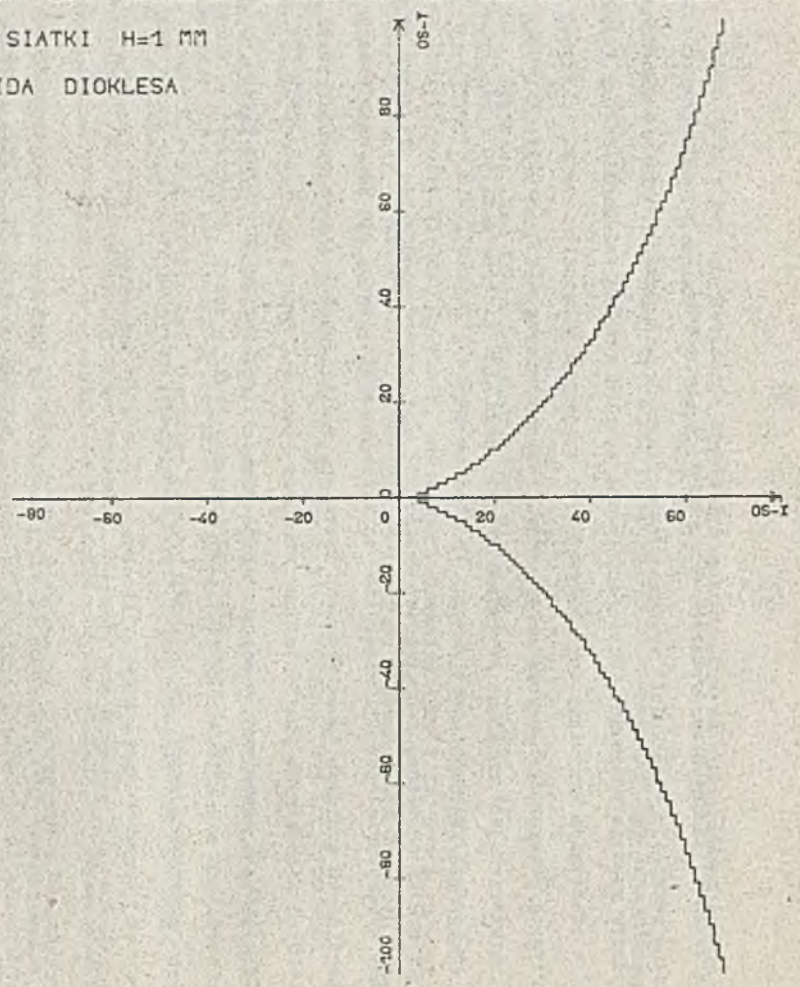
CISSOIDA DIOKLESA



APROKSYMACJA KRZYWYCH 3-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

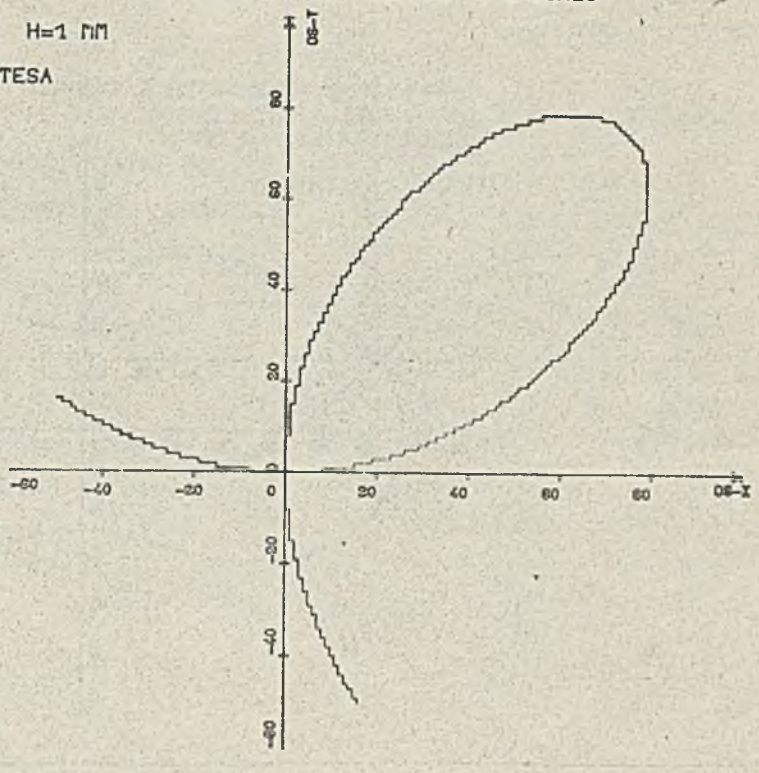
CISSOIDA DIOKLESA



APROKSYMACJA KRZYWYCH 3-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

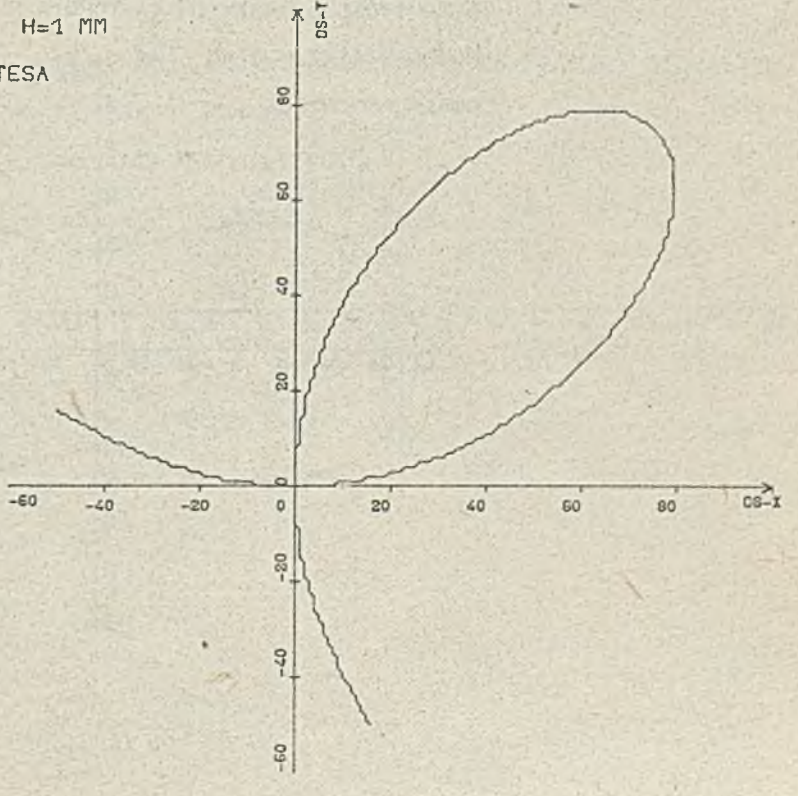
LISC DESCARTESA



APROKSYMACJA KRZYWYCH 3-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM

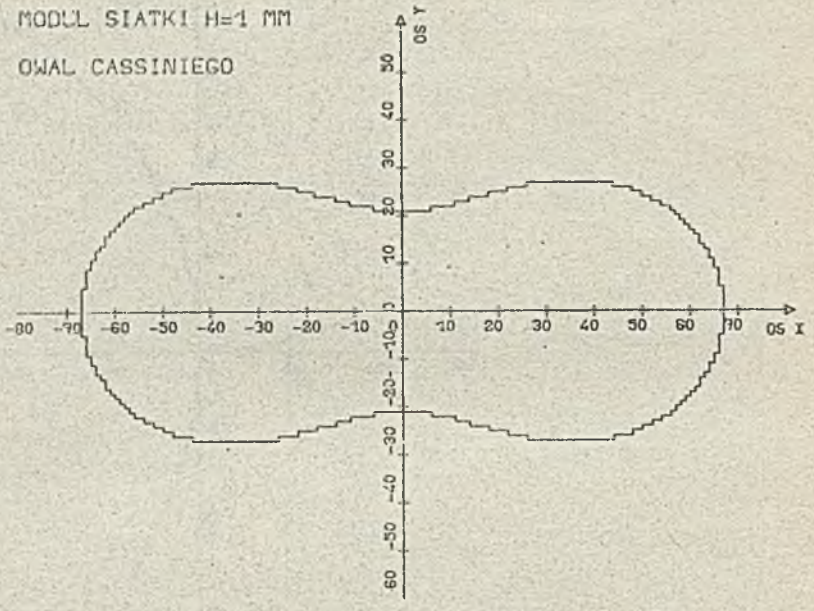
LISC DESCARTESA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

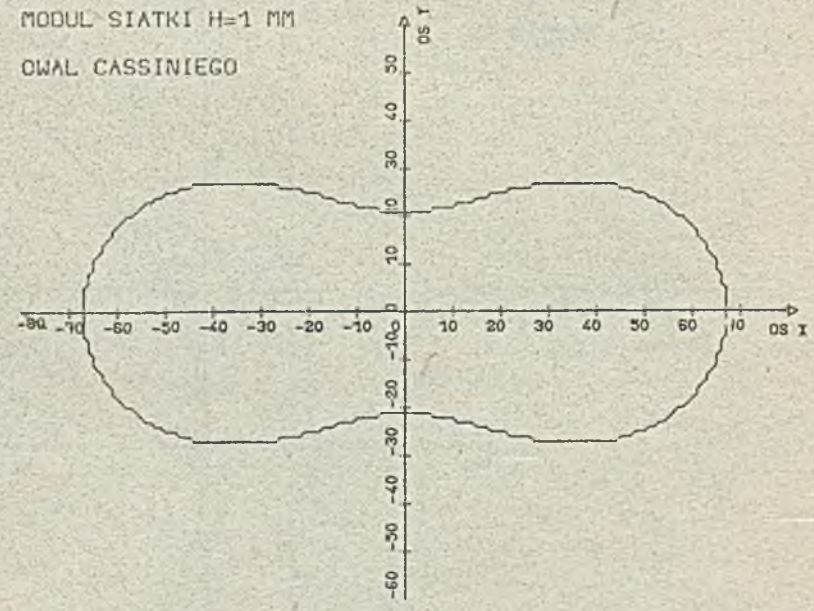
OWAL CASSINIEGO



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM

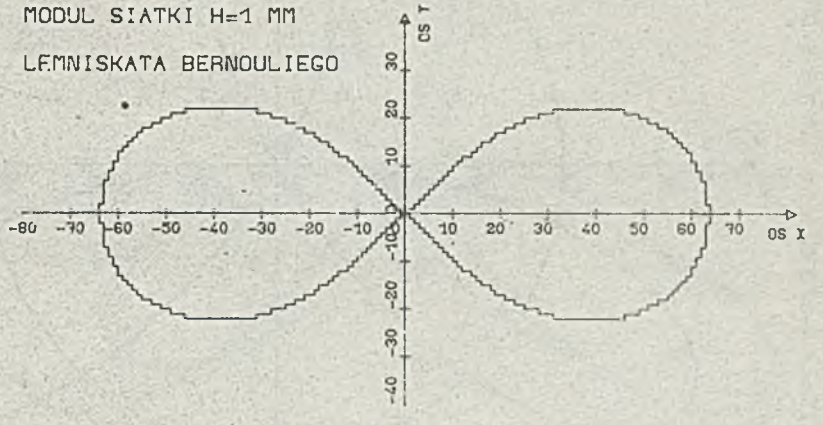
OWAL CASSINIEGO



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

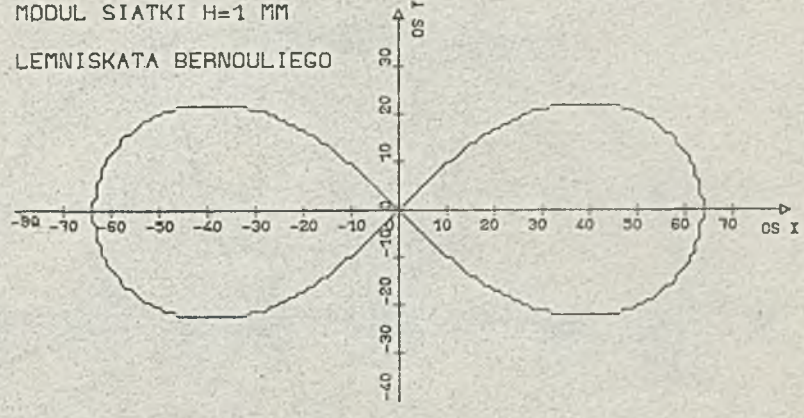
LEMNISKATA BERNOULIEGO



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 8-SPOJNEJ

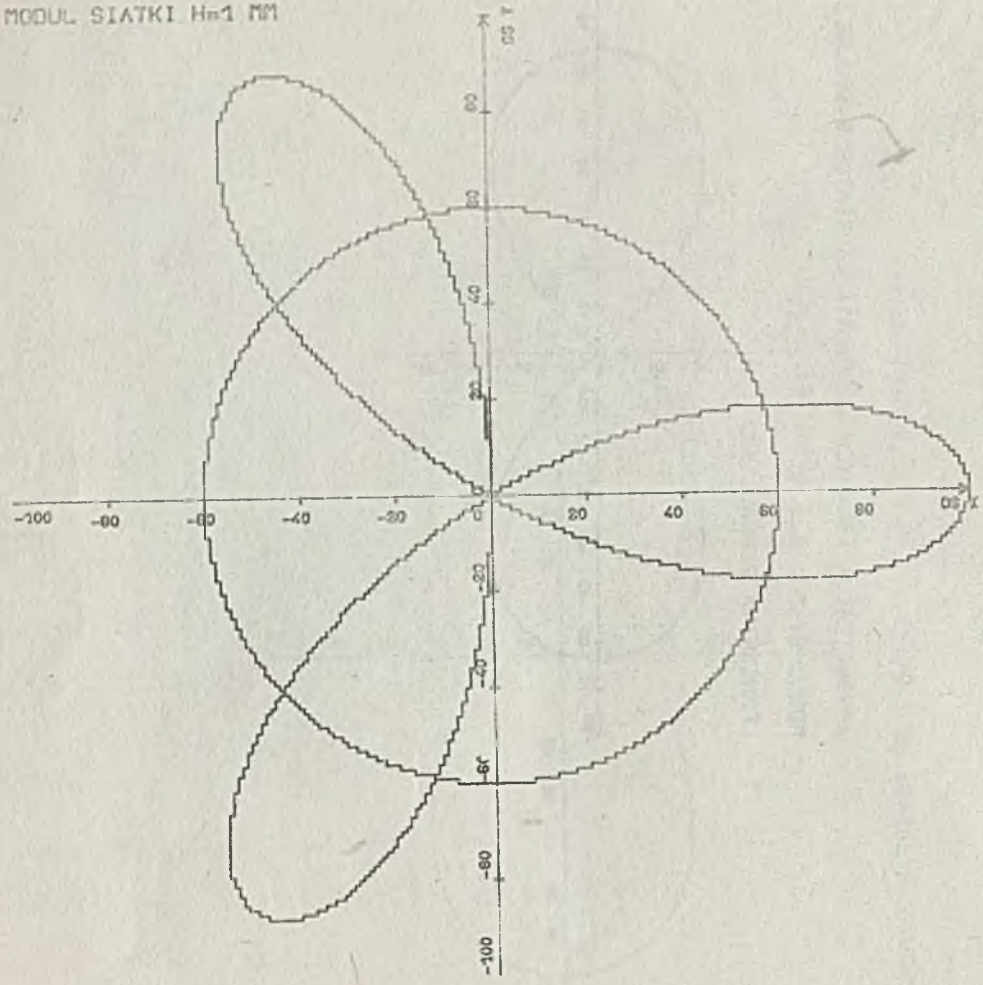
MODUL SIATKI H=1 MM

LEMNISKATA BERNOULIEGO



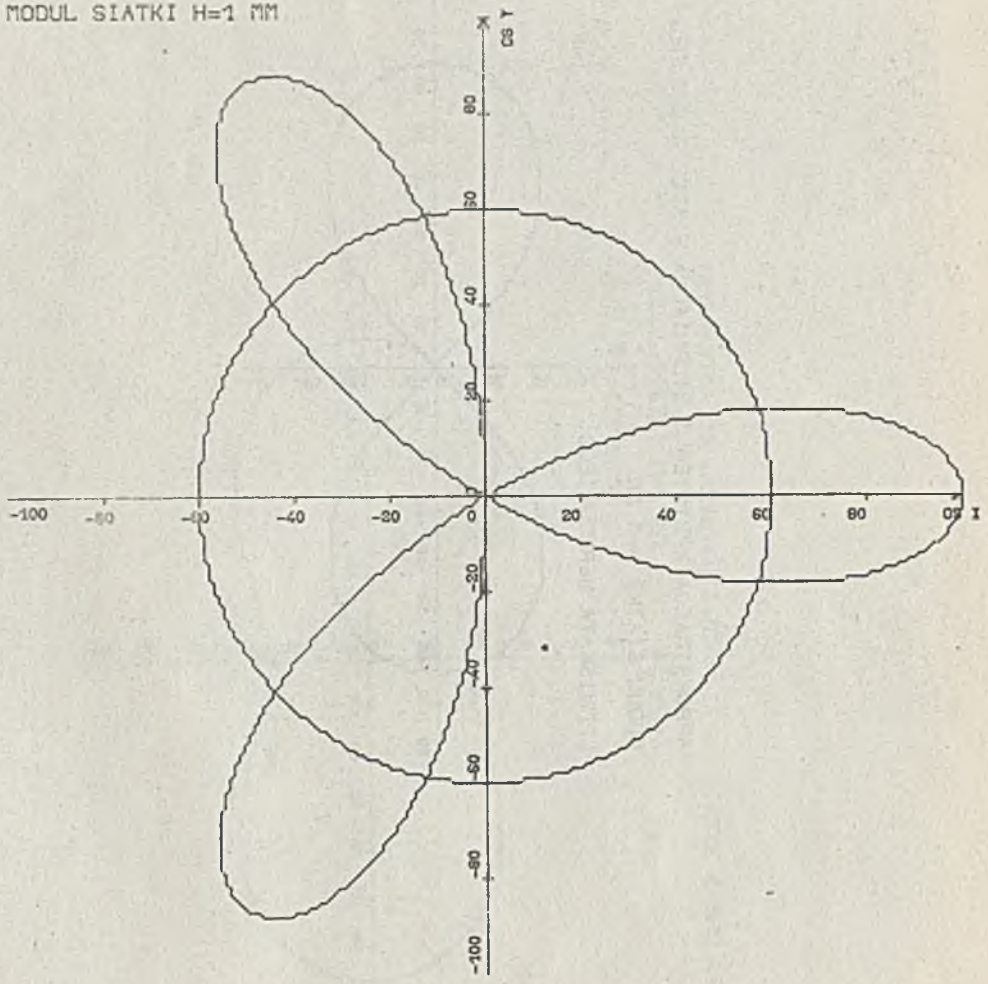
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM



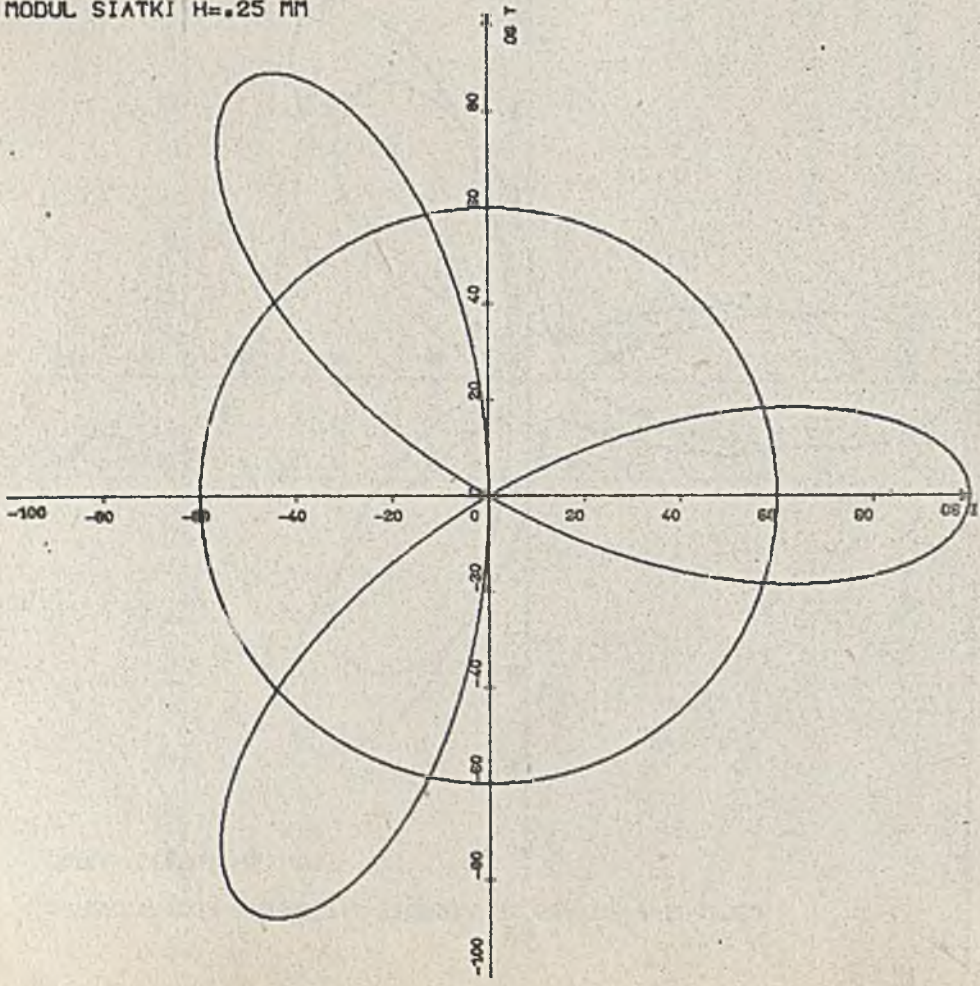
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM



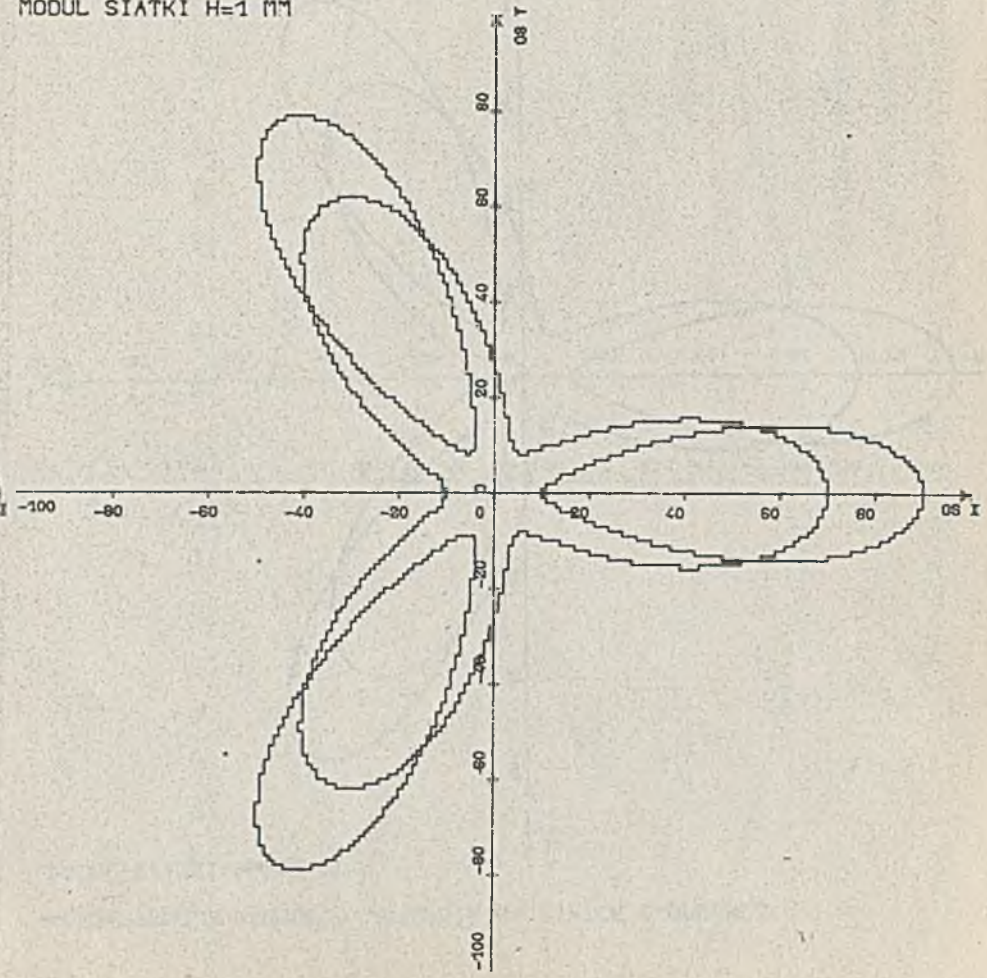
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=.25 MM



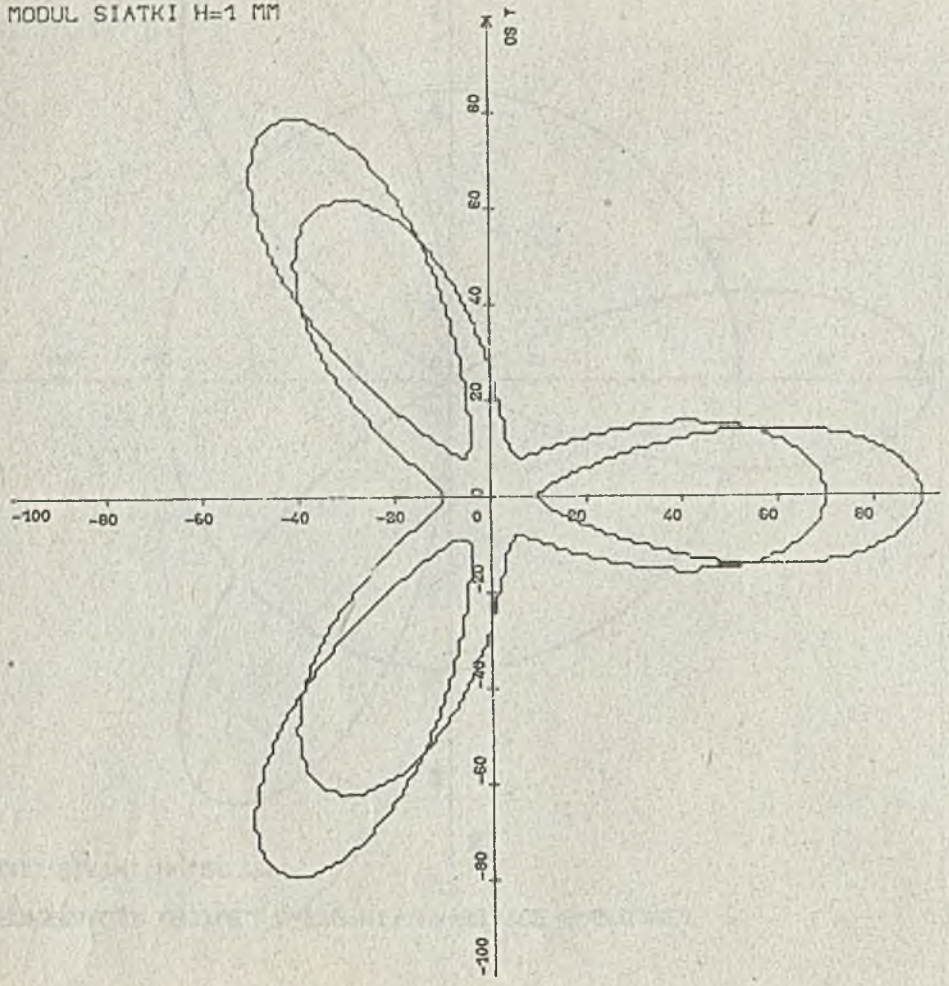
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM



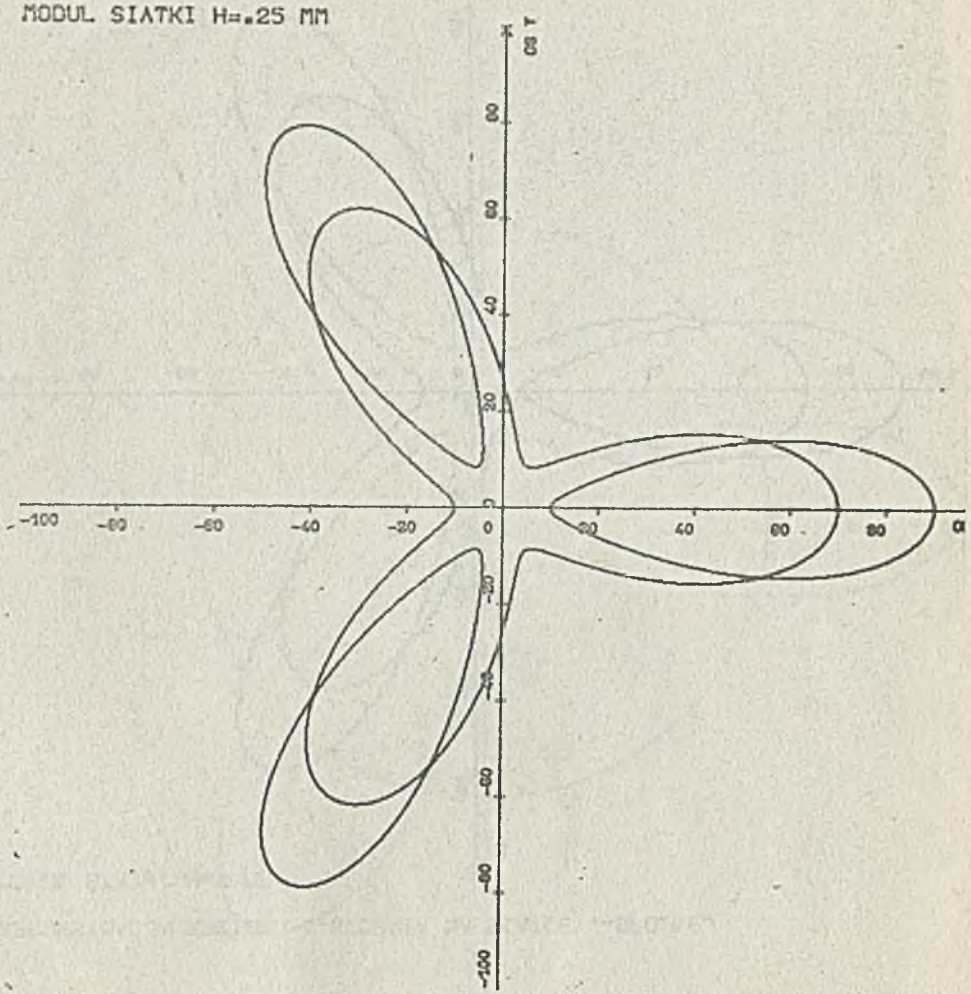
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM



APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

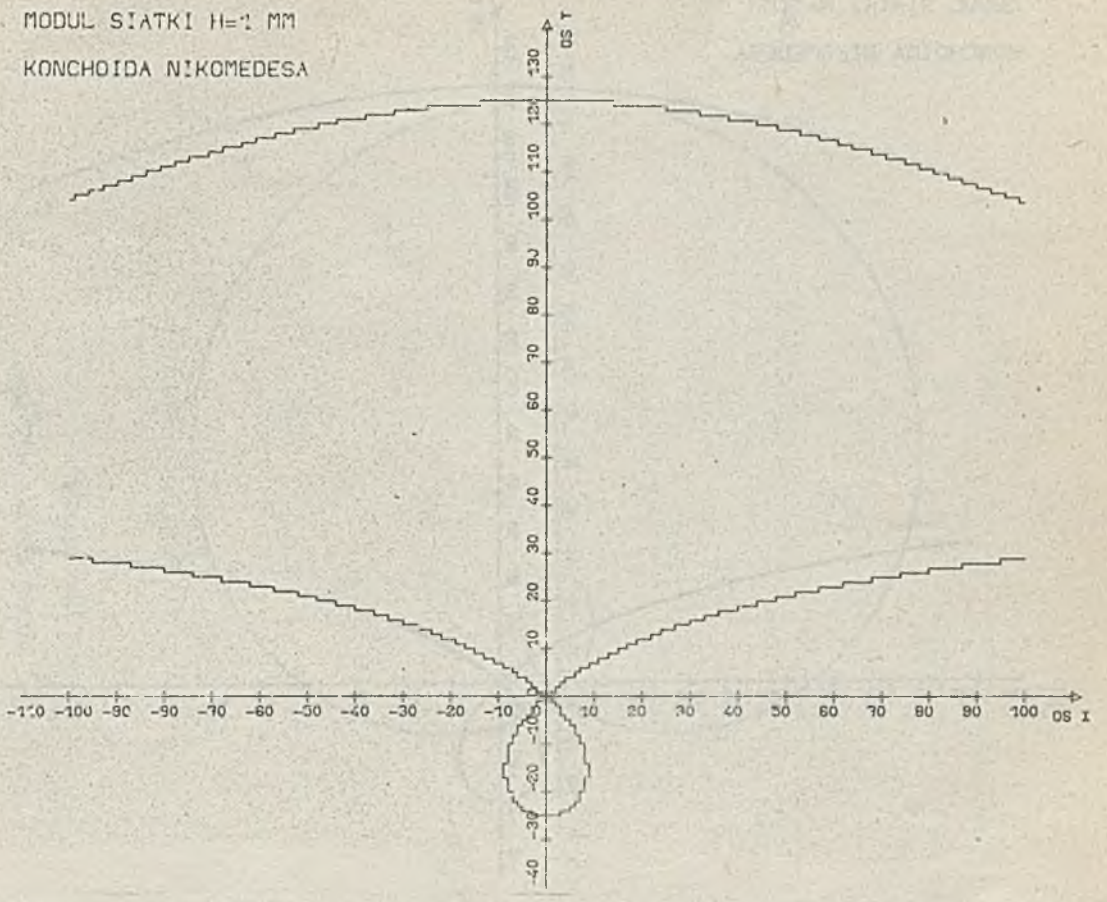
MODUL SIATKI H=.25 MM



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

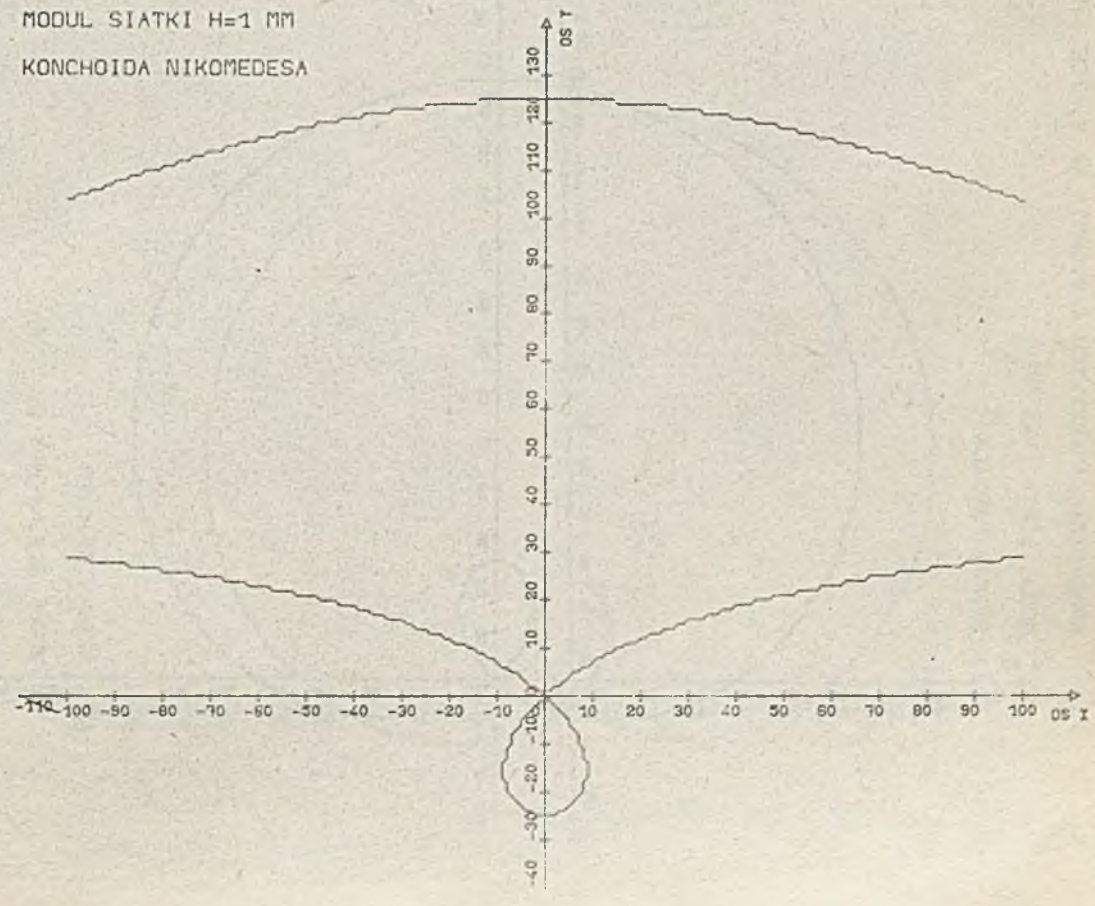
KONCHOIDA NIKOMEDESA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM

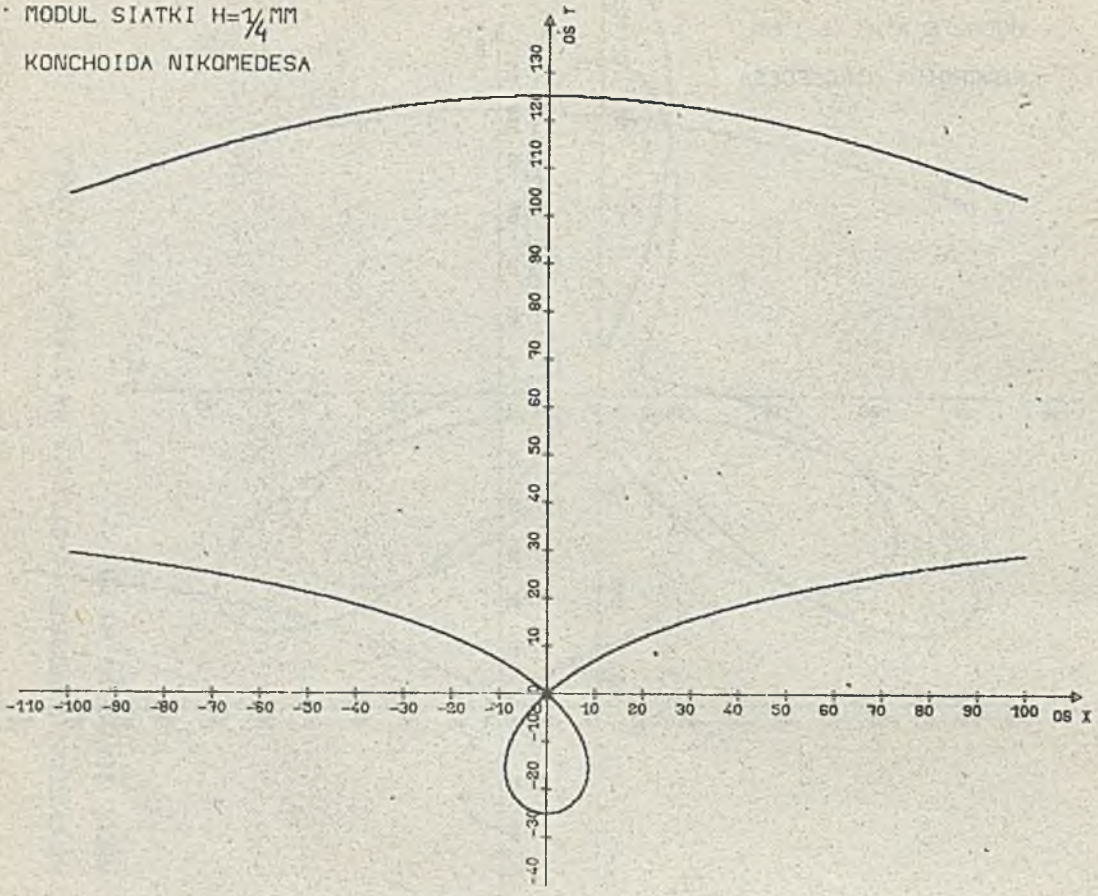
KONCHOIDA NIKOMEDESA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1/4 MM

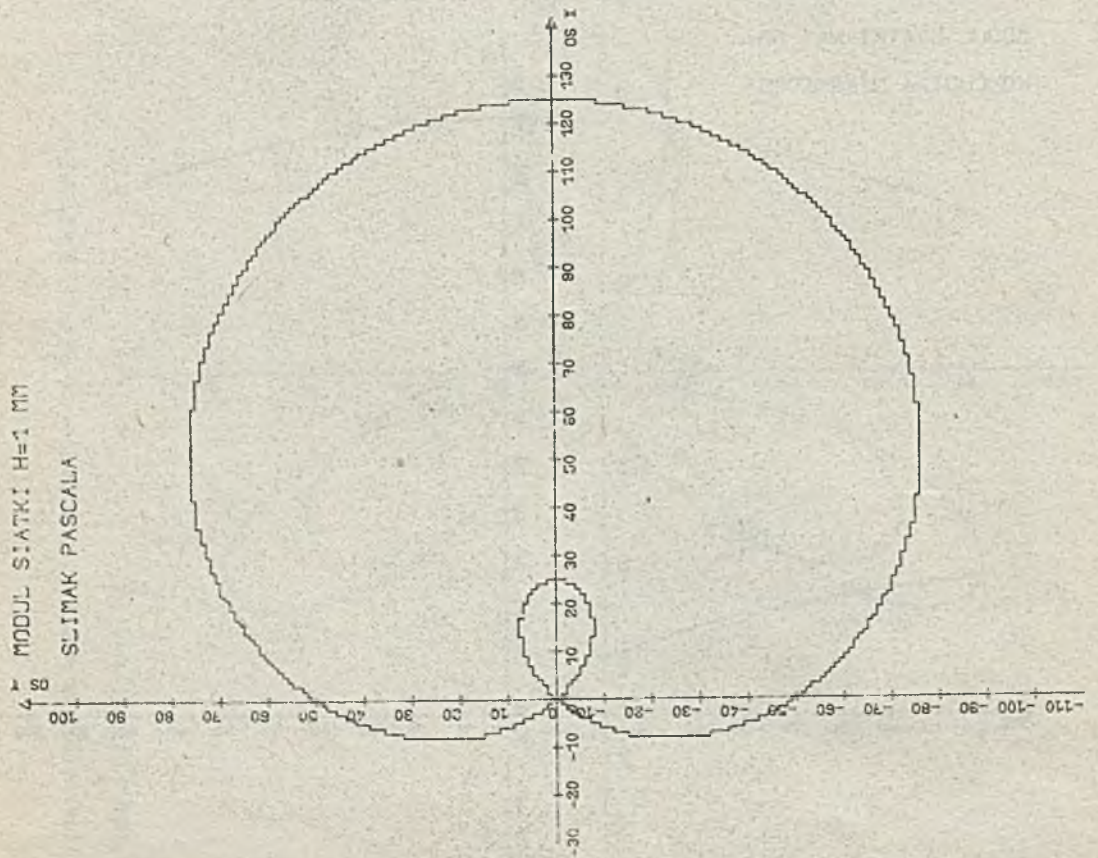
KONCHOIDA NIKOMEDESA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM

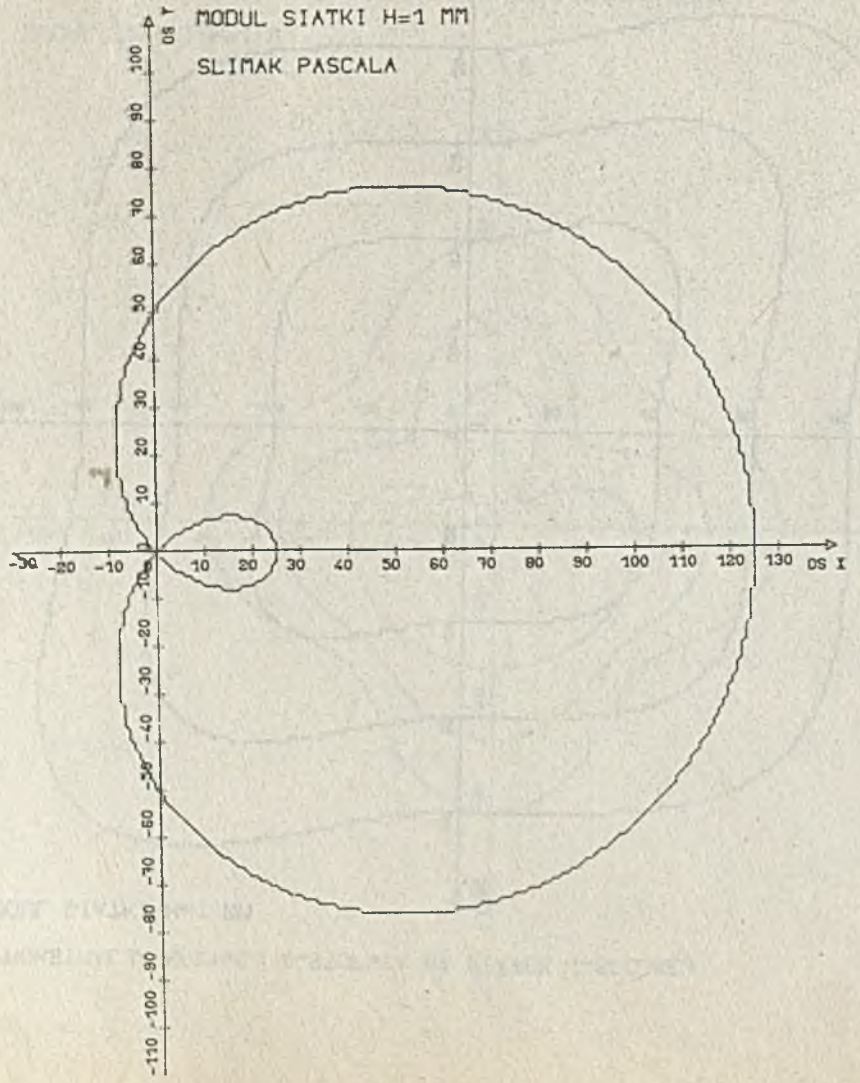
SLIMAK PASCALA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM

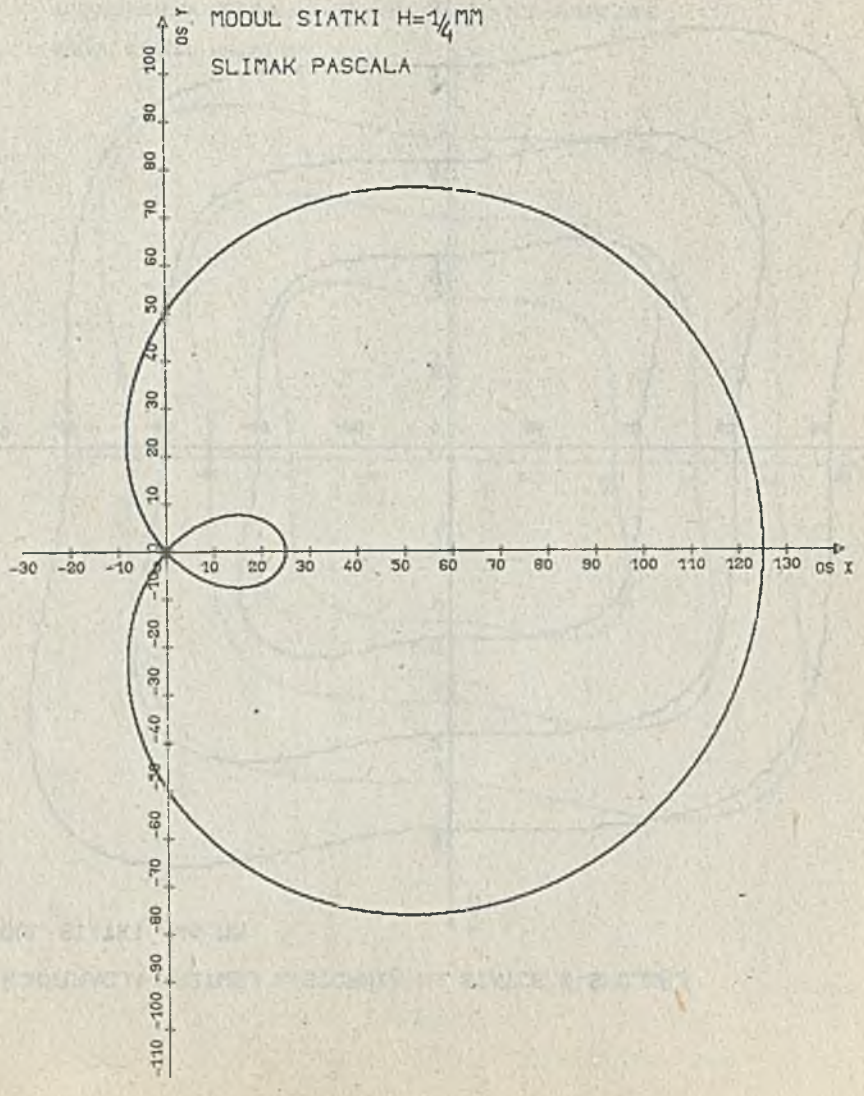
SLIMAK PASCALA



APROKSYMACJA KRZYWYCH 4-STOPNIA NA SIATCE 4-SPOJNEJ

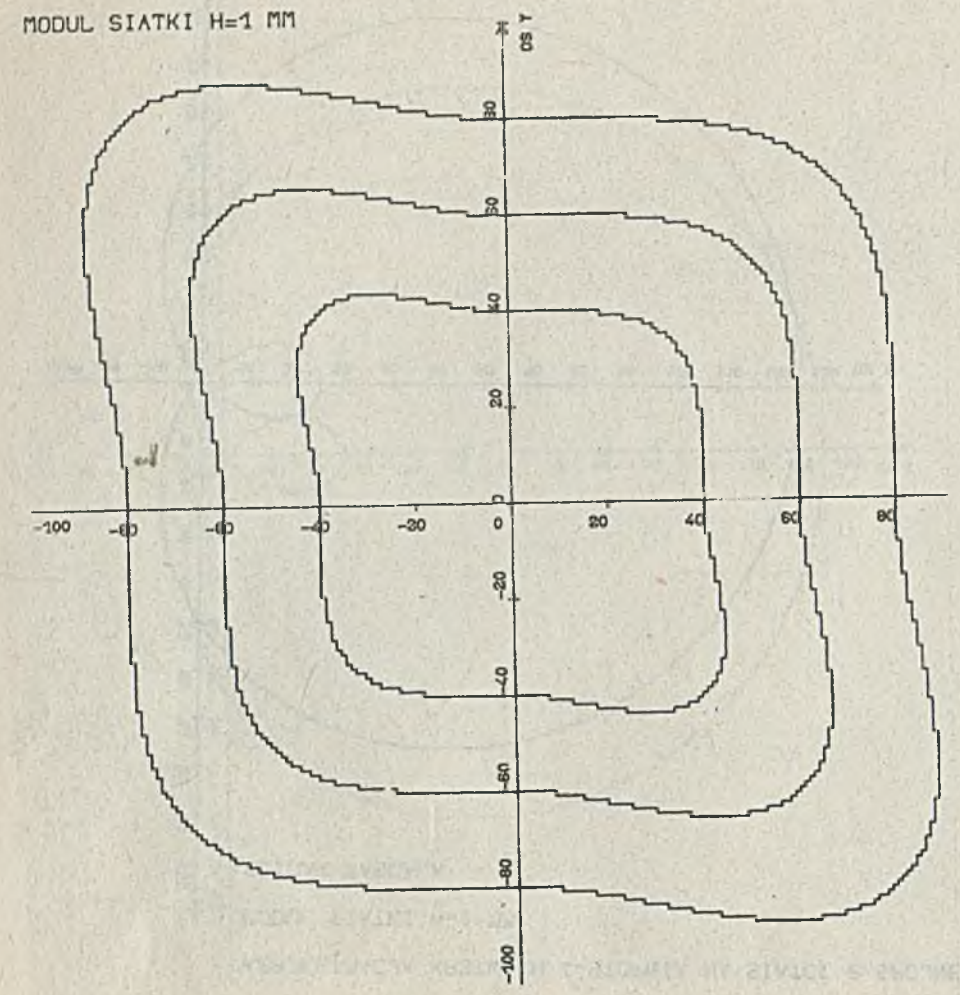
MODUL SIATKI H=1/4 MM

SLIMAK PASCALA



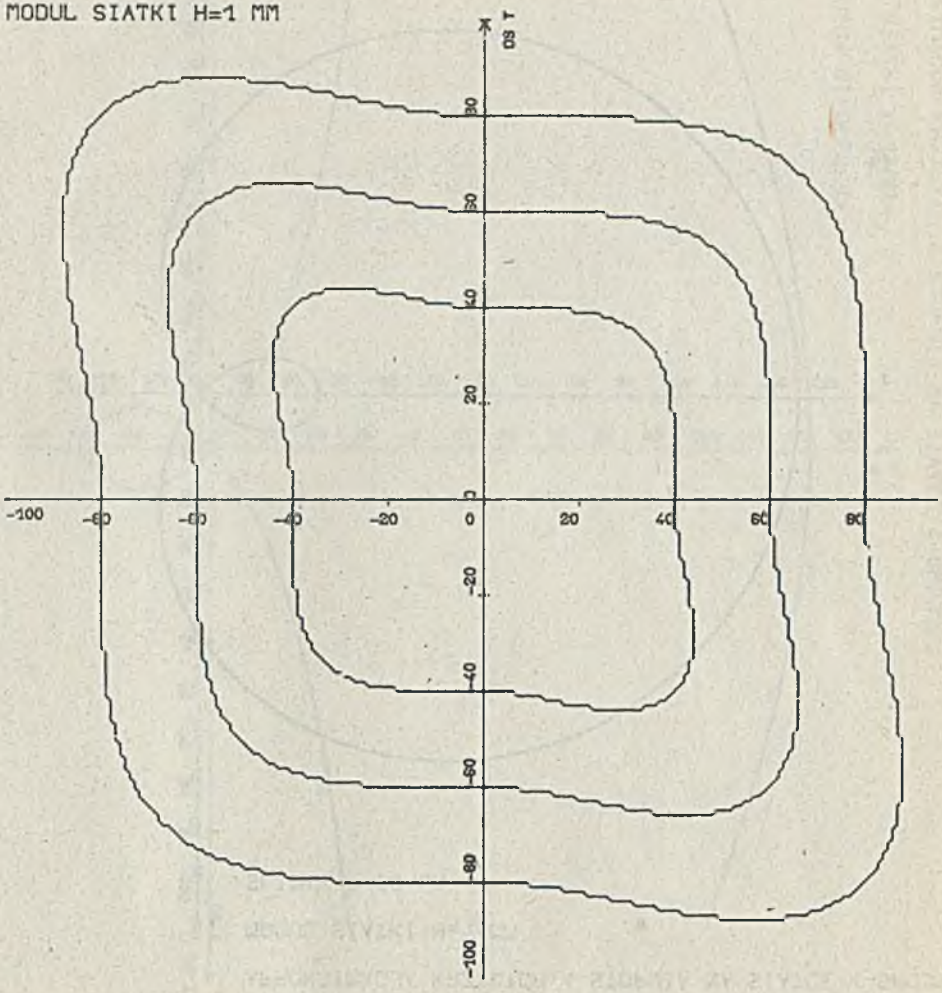
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM



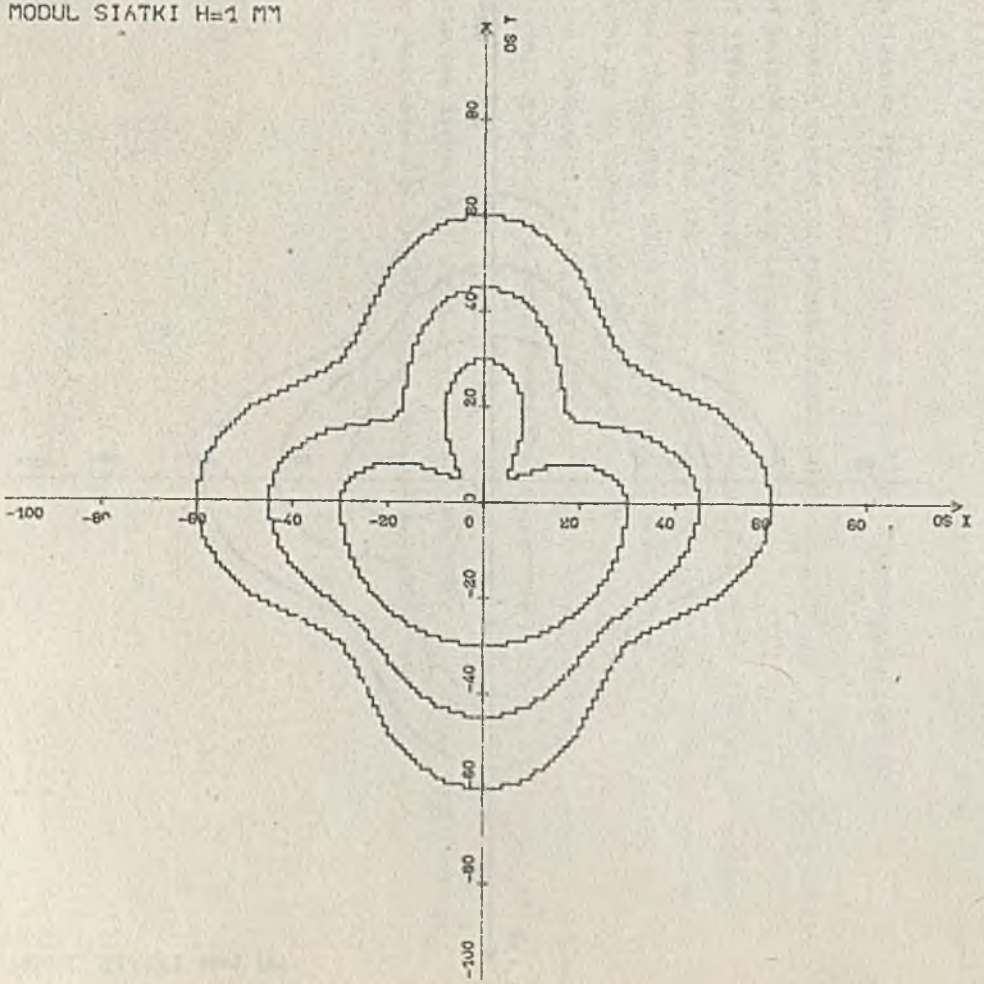
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM



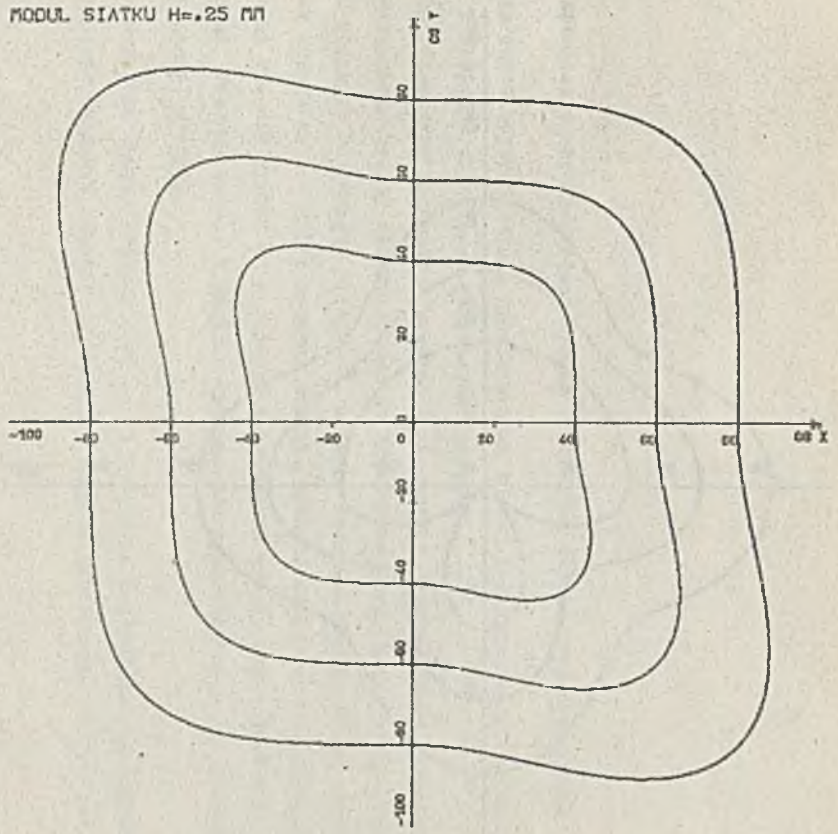
APROKSYMACJA KRZYWEJ 6-STOPNIA NA SIATCE 4-SPOJNEJ

MODUL SIATKI H=1 MM



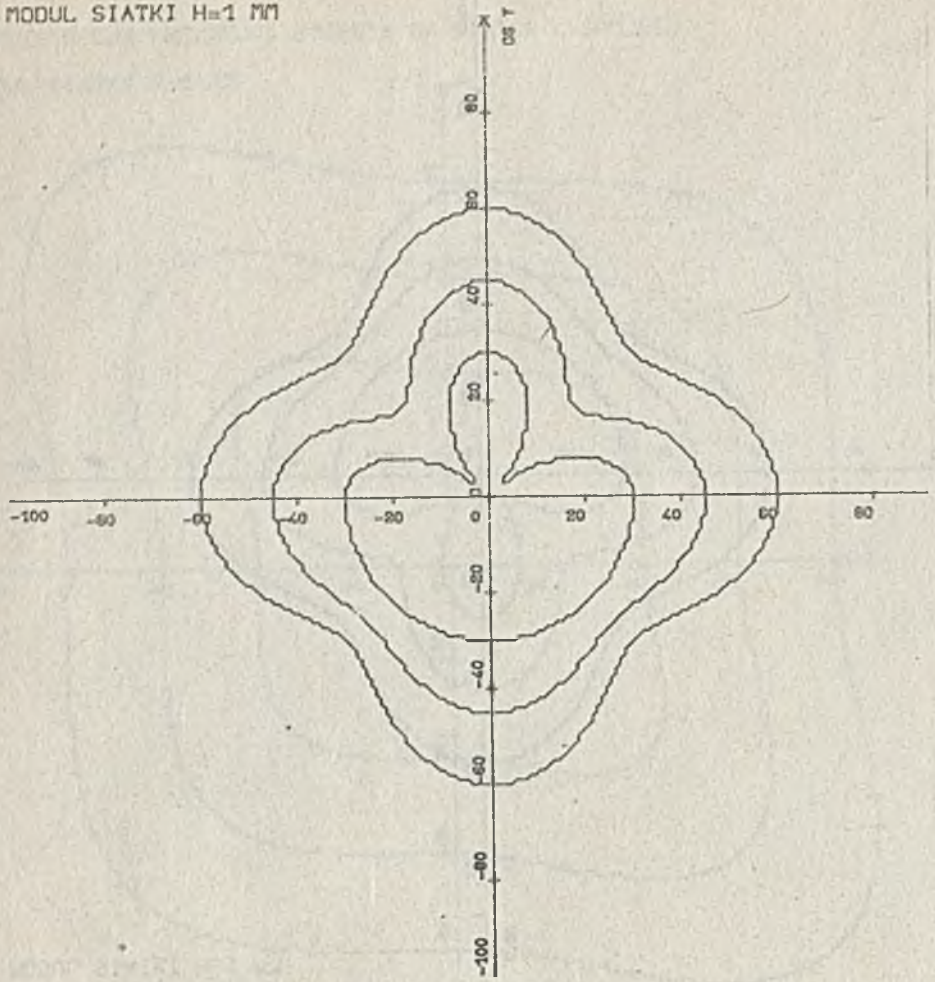
APROKSYMACJA KRZYWEJ 4-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKU H=.25 MM



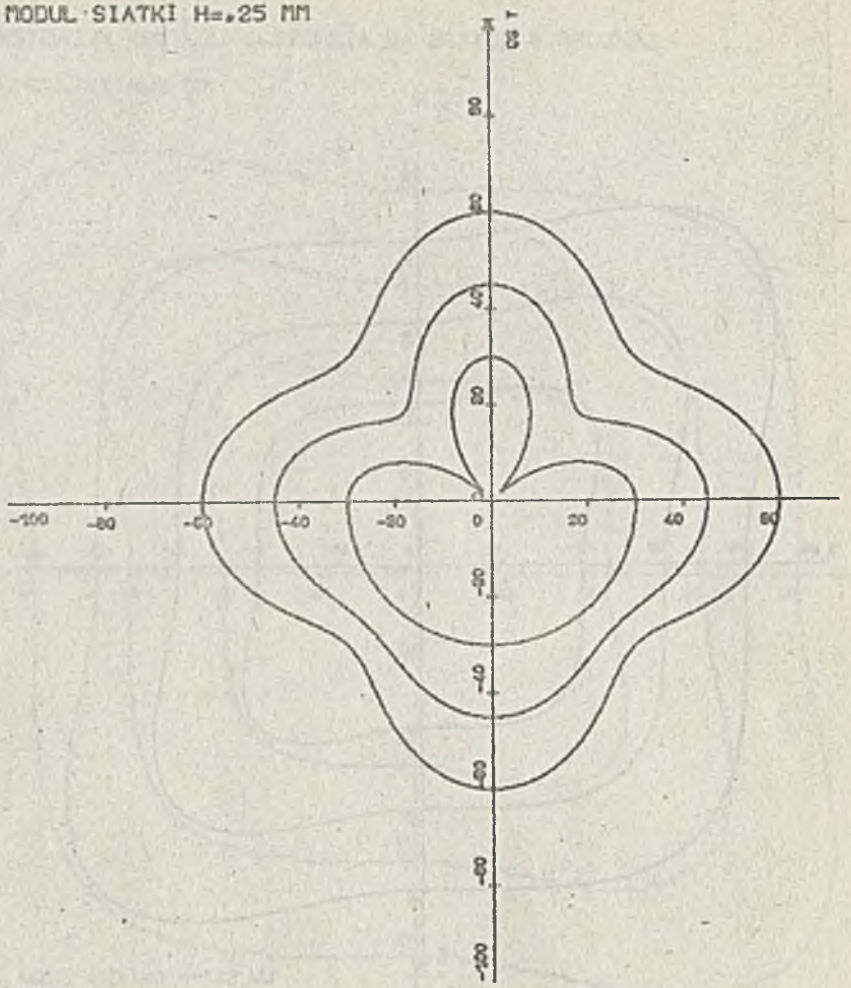
APROKSYMACJA KRZYWEJ 6-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=1 MM



APROKSYMACJA KRZYWEJ 6-STOPNIA NA SIATCE 8-SPOJNEJ

MODUL SIATKI H=.25 MM



Р е з ю м е

Дискретизация алгебраических кривых в машинном изображении информации

В диссертации анализируется методы дискретизации кривых предназначенные к применению в машинной графике и в цифровом управлении фрезерными станками.

Сделано обзор литературы касающейся методов и алгоритмов дискретизации (целочисленной аппроксимации) и сформулировано задачу дискретизации на однородных квадратных сетках алгебраических кривых n-степени заданных ревенством $F(x,y)=0$, а также представлено сделаные автором алгоритмы дискретизации этих кривых на 4- и 8-связанных сетках. Сделано оценку вычислительной сложности алгоритмов в функции степени кривой n и порядка связанности сетки. Переведено тоже анализ качества дискретизации: пиковых ошибок в метрике евклидового пространства и глобальную оценку в полевой (интегральной) метрике.

В приложении включено графики кривых 3-, 4- и 6-степени выполненные графопостроителем, полученные в результате машинной реализации представляемых в диссертации алгоритмов.

S u m m a r y

Discretization of algebraic curves in computer displaing information

In this dissertation methods of curves discretization adjusted for computer graphics and digital control of machine tools are analyzed.

The review of literature concerning methods and algorithms of discretization /integer approximation/ has been made and the problem of curves $F(x,y)=0$ discretization on homogeneous square grids has been formulated. Algorithms of curves discretization on grids 4- and 8-connective elaborated by the author have been presented.

The computation conplexity of each algorithm /as the function of n-degree curves/ is evaluated. Analysis of maximal errors of the curve discretization on square grids in Euclidean space metrics as well as global evaluation of discretization quality in polar metrics has been made.

In the appendix the diagrams of the 3-, 4- and 6-degree curve plotted has been shown as the result of computer realization of the algorithms proposed in this work.

U W A G A !

Instytut Maszyn Matematycznych przekaże nieodpłatnie do eksploatacji sprzętowo niezależny system programowania grafiki komputerów PSG zaimplementowany na minikomputerze MERA400.

Bliższe informacje: Pracownia Grafiki Komputerowej, tel. 21-84-41 w. 271, 388, 428.

Informacja o cenach i warunkach prenumeraty na 1986 r.
- dla czasopism Instytutu Maszyn Matematycznych

● Cena prenumeraty rocznej

Techniki Komputerowe - Biuletyn Informacyjny	1560.-	dwum.
Przegląd Dokumentacyjny - Nauki i Techniki Komputerowe	1260.-	dwum.
Informacja Ekspresowa - Nauki i Techniki Komputerowe	2400.-	mies.
Prace naukowo-badawcze Instytutu Maszyn Matematycznych	660.-	3x w roku

● Warunki prenumeraty

1/ dla osób prawnych - instytucji i zakładów pracy:
- instytucje i zakłady pracy zlokalizowane w miastach wojewódzkich i pozostałych miastach, w których znajdują się siedziby oddziałów RSW "Prasa-Książka-Ruch" zamawiają prenumeratę w tych oddziałach;
- instytucje i zakłady pracy zlokalizowane w miejscowościach, gdzie nie ma oddziałów RSW "Prasa-Książka-Ruch" i na terenach wiejskich opłacają prenumeratę w urzędach pocztowych i u doręczycieli;

2/ dla osób fizycznych - prenumeratorów indywidualnych:
- osoby fizyczne zamieszkałe na wsi i w miejscowościach, gdzie nie ma oddziałów RSW "Prasa-Książka-Ruch" opłacają prenumeratę w urzędach pocztowych i u doręczycieli;
- osoby fizyczne zamieszkałe w miastach - siedzibach oddziałów RSW "Prasa-Książka-Ruch" opłacają prenumeratę wyłącznie w urzędach pocztowych nadawczo-oddawczych właściwych dla miejsca zamieszkania prenumeratora. Wpłaty dokonują używając "blankietu wpłaty" na rachunek bankowy miejscowego oddziału RSW "Prasa-Książka-Ruch";

3/ Prenumeratę ze zleceniem wysyłki za granicę przyjmuje RSW "Prasa-Książka-Ruch", Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto NBP XV Oddział w Warszawie nr 1153-201045-139-11. Prenumerata ze zleceniem wysyłki za granicę pocztą zwykłą jest droższa od prenumeraty krajowej o 50% dla zleceniodawców indywidualnych i o 100% dla zlecających instytucji i zakładów pracy.

● Terminy przyjmowania prenumeraty na kraj i za granicę:

- do dnia 10 listopada na I kwartał, I półrocze roku następnego oraz na cały rok następny,
- do dnia 1-każdego miesiąca poprzedzającego okres prenumeraty roku bieżącego.

Zamówienia na prenumeratę "Prac naukowo-badawczych Instytutu Maszyn Matematycznych przyjmuje Dział Sprzedaży Wysyłkowej Ośrodka Rozpowszechniania Wydawnictw Naukowych PAN, Warszawa, Pałac Kultury i Nauki, tel. tel.20-02-11 w.2516. Egzemplarze pojedyncze Prac są do nabycia w księgarni ORWN PAN, Warszawa, Pałac Kultury i Nauki, tel.20-02-11 w.2109.

P.4201/85

Cena zł. 220,-