Kauno technologijos universitetas

INFORMATIKOS FAKULTETAS
SISTEMINĖS ANALIZĖS KATEDRA

Giedrė Sudniutė

FRAKTALINIŲ INTERPOLIACINIŲ FUNKCIJŲ PRAKTINIO PANAUDOJIMO ANALIZĖ

Magistro darbas

Vadovas
doc. dr. J. Valantinas

KAUNAS, 2005

SUMMARY

The present thesis introduces fractal interpolation functions; reveals advantages of fractal interpolation of real world objects and presents organisational procedures of fractal interpolation process.

The work briefly presents the context needed for introduction of fractal notion. Also, detailed descriptions of algorithms of fractal generation (i.e. determined, casual iteration, “runtime”) are indicated, and types of fractals are commented on. Since the objects of the research are theoretical and practical aspects of fractal interpolation function analysis, special attention is paid to geometric fractals got by applying systems of iteration (affined) functions.

The notion of fractal interpolation function is provided in the work. The author shows that it is possible to get functions for various types of data. Such functions (by implication of Hausdorf dimension) are “close” to the data. Moreover, it is possible to ensure congruity of fractal interpolation graph dimension with fractal dimension of experimental data (graph). The algorithm of casual iteration is used for the analysis of interpolation functions, since it is marked by quickness and good quality of fractal synthesised images. The author tries to analyse and solve the problem of selection of interpolation points (general case). The following methods are proposed: the method of equal selection of interpolation points aimed at interactive usage and collage. Also, a new attitude on selection of oblique affined transformations that make IFS is suggested in the work.

INTERPO tool intended for fractal interpolation research was used for performance of practical usage experiment of fractal interpolation function. The experiment was applied to three different objects: landscape, currency market trade object and stock market trade object. The results of the experiment revealed that INTERPO is the tool by means of which it is possible to generate FIF for each type of data analysis.

TURINYS

I. ĮVADAS 7
II. FRAKTALINĖ GEOMETRIJA — NAUJAS MATEMATINIO MODELIAVIMO METODAS 10
2.1. Fraktalai, fraktalų sintezė 10
2.1.1. Iteruotųjų funkcijų sistemos ir jų atraktoriai (fraktalai) 10
2.1.2. Fraktalų sintezės algoritmai 12
2.1.3. Fraktalų tipai 16
2.2. Fraktalinės dimensijos sąvoka 17
2.3. Fraktalinės interpoliacinės funkcijos, jų specifika 21
2.4. Kai kurie fraktalų taikymo pavyzdžiai 24
III. FRAKTALINIŲ INTERPOLIACINIŲ FUNKCIJŲ ANALIZĖ: TEORINIAI IR PRAKTINIAI ASPEKTAI 28
3.1. INTERPO—fraktalinio interpoliavimo įrankis 33
3.2. Interpoliavimo taškų parinkimo kriterijai ir jų nustatymas 36
3.2.1. Interaktyviam naudojimui orientuoto tolygaus interpoliavimo taškų išdėstymas 37
3.2.2. Koliažo organizavimas 44
3.3. Interpoliuojamo objekto fraktalinės dimensijos įvertinimas 47
3.4. Eksperimento rezultatų analizė 49
IV. IŠVADOS 67
V. LITERATŪRA 68
PRIEDAS1 INTERPO ALGORITMAS C KALBA 81
PRIEDAS 2 FRACDIM TAIKYMO PAVYZDŽIAI 86
PRIEDAS 3 GREITOS FURJE ATRANKOS ALGORITMAS C KALBA 88
PRIEDAS 4 VB INTERPO ALGORITMAS VISUAL BASIC 94
PRIEDAS 5 (CD) PAGRINDINĖ DARBO DALIS 106

Lentelių sąrašas

3.1 lentelė Koordinatinio duomenų masyvo turinys 50
3.2 lentelė Naujas duomenų rinkinys 51
3.3 lentelė Trečias duomenų rinkinys 51
3.4 lentelė 3.3 lentelės rinkinio tęsinys 51
3.5 lentelė Skaitmenizuotą vaizdą nusakantis duomenų rinkinys 51
3.6 lentelė 3.5 lentelės rinkinio tęsinys 51
3.7 lentelė Gauti skaičiavimų rezultatai 54
3.8 lentelė Gauti skaičiavimų rezultatai 60
3.9 lentelė Gauti skaičiavimų rezultatai 61
3.10 lentelė Gauti skaičiavimų rezultatai 62
3.11 lentelė Gauti skaičiavimų rezultatai 64
3.12 lentelė Duomenys apie prekybą akcijomis 65

Paveikslėlių sąrašas

2.1 pav. Determinuotojo IFS atraktoriaus sintezės algoritmo veikimo rezulltatas 13
2.2 pav. Iteruotųjų funkcijų sistemų atraktoriai (fraktaliniai vaizdai), gauti taikant atsitiktinių iteracijų algoritmą 14
2.3 pav. Mandelbroto ir Žulija aibės; $\{C; z^2\text{-}\lambda\}$ (centre—Mandelbroto (parametro reikšmių λ, kurias atitinkančios Žulija aibės yra jungios) aibė) 16
2.4 pav. Van Kocho kreivė (fraktalas) 18
2.5 pav. Sierpinskio trikampis 20
2.6 pav. Dalimis tiesinės interpoliacinės funkcijos, einančios per interpoliavimo taškus, grafikas; šis grafikas sutampa su IFS $\{R^2; \omega_1, \omega_2, \ldots \omega_N\}$ atraktoriumi 22
3.1 pav. Interpoliavimo priemones struktūra (modulių diagrama) 28
3.2 pav. Interpoliavimo priemonės tikslų modelis 29
3.3 pav. Interpoliavimo priemones funkcijų modelis 30
3.4 pav. Interpoliavimo priemonės įvykių modelis 30
3.5 pav. Interpoliavimo priemonės būsenų modelis 31
3.6 pav. Duomenų analizės proceso sąveikų modelis 31
3.7 pav. FIF kreivės formavimo proceso metodų modelis 32
3.8 pav. Interpoliavimo priemonės panaudojimo atvejų modelis 32

3.9 pav. Interpoliavimo priemonės problemų modelis 33
3.10 pav. Pradinis INTERPO ekranas – jame siūloma pasirinkti monitoriaus tipą 34
3.11 pav. Duomenų aibės (Sample.fip), kai *D*=1, interpoliavimo rezultatas 35
3.12 pav. Duomenų aibės (Avg100.fip), kai *D*=1,3 interpoliavimo rezultatas 35
3.13 pav. Duomenų aibės pateikimas 35
3.14 pav. FIF grafikas; Pvz2a.fip, kai *D*=1 36
3.15 pav. FIF grafikas; Pvz2a.fip, kai *D*=1.61 36
3.16 pav. Diskretizuotos kreivės pvz. 38
3.17 pav. Pirminis intervalo dalijimas pusiau 39
3.18 pav. Naujai dalijamos srities pasirinkimas 39
3.19 pav. Modulinė priemonės struktūra 40
3.20 pav. VB Interpo priemonės pagrindinio ekrano maketas 41
3.21 pav. VB Interpo pagrindinis ekranas (darbinė sritis) 42
3.22 pav. Duomenų nuskaitymas VB Interpo 42
3.23 pav. Pradinis skaičiavimų ir duomenų spausdinimo etapas 43
3.24 pav. Koliažo organizavimo ypatumai 46
3.25 pav. Neries upės vagos žemėlapyje skenuotas vaizdas 49
3.26 pav. Bendras vaizdas 52
3.27 pav. Paklaida 52
3.28 pav. Gauta FIF kreivė 53
3.29 pav. Gauta FIF kreivė 53
3.30 pav. Gauta FIF kreivė 54
3.31 pav. Gauta FIF kreivė 54
3.32 pav. Gauta FIF kreivė 55
3.33 pav. Gauta FIF kreivė 56
3.34 pav. Upės vagos vaizdas žemėlapyje 56
3.35 pav. Gauta FIF kreivė, kai *D*=1.03 56
3.36 pav. Gauta FIF kreivė, kai *D*= 1.2 56
3.37 pav. Gauta FIF kreivė, kai *D*= 1.3 56
3.38 pav. Neries upės vaga žemėlapyje 57
3.39 pav. Gauta FIF kreivė, kai *D* =1.0 57
3.40 pav. Gauta FIF kreivė, kai *D*= 1.1 57
3.41 pav. Neries vagos vaizdas žemėlapyje 57
3.42 pav. Gauta FIF kreivė, kai *D*= 1.0 57
3.43 pav. Gauta FIF kreivė, kai *D*= 1.1 58

3.44 pav. Gauta FIF kreivė, kai *D*= 1.2 …….58
3.45 pav. Gauta FIF kreivė, kai *D*= 1.3 …….58
3.46 pav. JAV dolerio kurso kaitos lentelė ……..….59
3.47 pav. JAV dolerio kurso kaitos diagrama ……...59
3.48 pav. Gauta FIF kreivė ……...…….59
3.49 pav. Gauta FIF kreivė ……...…….60
3.50 pav. Gauta FIF kreivė ……...…….60
3.51 pav. Gauta FIF kreivė ……...…….61
3.52 pav. Gauta FIF kreivė ……...…….61
3.53 pav. Gauta FIF kreivė ……...…….62
3.54 pav. LB pateikta grafinė diagrama ……...…….62
3.55 pav. Gauta FIF kreivė ……...…….63
3.56 pav. Gauta FIF kreivė ……...…….64
3.57 pav. Biržos svetainėje pateikta grafinė diagrama …….…….64
3.58 pav. MS Excel diagrama pusei metų (duomenys fiksuoti kasdien) …….…….65
3.59 pav. Interpo rezultatas, kai duomenys skaitomi iš Nafta10t.fip (76 taškai, skalė 10x1, *D*=1.1) ..66
3.60 pav. Gauta FIF kreivė *D*=1.5 …….…66
3.61 pav. Gauta FIF kreivė *D*=1.6 …….66
3.62 pav. Gauta FIF kreivė *D*=1.7 …….66
3.63 pav. MS Excel pusmečio rezultatas ……..…….66

I. ĮVADAS

Sąvokos „ fraktalas“ ir „ fraktalinė geometrija“, atsirado aštuntojo dešimtmečio pabaigoje, o nuo devintojo dešimtmečio vidurio jau ilgam įsitvirtino matematikų ir programuotojų žodyne. Terminas „ fraktalas“ paimtas iš lotynų kalbos: „ fractus“ reiškia nereguliarų fragmentą. Beje, šį terminą 1975 m. pasiūlė amerikiečių matematikas Benua Mandelbrotas (Benoit Mandelbrot) nereguliarioms, bet panašioms į save struktūroms, pažymėti. Fraktalinės geometrijos gimimas siejamas su 1977 m. pasirodžiusia B. Mandelbroto knyga „ Fraktalinė gamtos geometrija“ [7]. Knygoje panaudoti ir kitų mokslininkų, dirbusių šioje srityje 1875—1925 metų laikotarpiu (Kantoro (Georg Cantor), Peano (Giuseppe Peano), Hilberto (David Hilbert), Kocho (Helge von Koch), Sierpinskio (Waclaw Sierpinski), Žulia (Gaston Julia), Hausdorfo (Felix Hausdorff), Bezicovičiaus (Abram Besicovitch), Fatu (Pierre Fatou), Puankare) mokslinių tyrimų rezultatai. Svarbu tai, kad tik pastaruoju metu B. Mandelbroto dėka visi tie darbai buvo apjungti į vientisą sistemą.

Tarp kitko, savo knygoje B. B. Mandelbrotas rašė: „ Kodėl (Euklido) geometrija dažnai vadinama šalta ir sausa? Viena iš priežasčių—tai negalėjimas atvaizduoti debesų, kalnų, krantų ar medžio formų. Debesys—ne rutuliai, kalnai—ne kūgiai, krantai—ne apskritimai, ir nei medžio žievė glodi, nei žaibas tiesėmis sklinda… daugybė gamtos formų yra tokios netaisyklingos ir padrikos, kad, palyginus su Euklido geometrija, gamta demonstruoja ne tik aukštesnį, bet ir visiškai kitokį sudėtingumo laipsnį“.

Viena iš pagrindinių fraktalų savybių —atsikartojimas savyje. Pačiu paprasčiausiu atveju, nedidelė fraktalo dalis kaupia informaciją apie visą fraktalą. B. Mandelbrotas fraktalą apibrėžė tokiais žodžiais: „ Fraktalas— tai struktūra, sudaryta iš dalių, kurios kokia tai prasme primena visumą“ [4].

Fraktalų įvairovę patogu pateikti, atsižvelgiant į visuotinai priimtą jų klasifikaciją, būtent: geometriniai fraktalai, algebriniai fraktalai, stochastiniai fraktalai.

Geometriniai fraktalai vaizdžiausi. Dvimačiu atveju, jie gaunami kokios tai laužtės (arba paviršiaus, trimačiu atveju), vadinamos generatoriumi, pagalba. Kiekviename algoritmo žingsnyje, laužtę sudaranti atkarpa keičiama laužte—generatoriumi (atitinkamai parinkus mastelį). Daugkartinio šios procedūros taikymo rezultatas—geometrinis fraktalas.

Kompiuterinėje grafikoje geometriniai fraktalai yra būtina priemonė medžių, krūmų, kalnų (kranto) linijos ir kitiems vaizdams sukurti. Dvimačiai geometriniai fraktalai naudojami erdvinėms tekstūroms (objektų paviršiaus piešiniams) išgauti.

Algebriniai fraktalai—tai pati didžiausia fraktalų grupė. Jie generuojami (gaunami) netiesinių iteracinių procesų *n*—matėse erdvėse pagalba. Geriausiai išanalizuoti yra dvimačiai procesai.

Interpretuojant tokį netiesinį iteracinį procesą kaip diskrečiąją dinaminę sistemą, galime naudotis šių sistemų teorijos terminija: fazinis portretas, nusistovėjęs procesas, atraktorius ir panašiai.

Žinoma, jog netiesinės dinaminės sistemos turi kelias stabilias būsenas (padėtis). Būsena, kurioje atsidūrė dinaminė sistema po baigtinio iteracijų skaičiaus, priklauso nuo jos pradinės padėties. Todėl kiekviena stabili būsena (dar vadinama atraktoriumi) turi fiksuotą aibę pradinių padėčių, iš kurių sistema būtinai pereina į analizuojamą galutinę stabilią būseną. Tokiu būdu, fazinė sistemos erdvė gali būti suskaidyta į atraktorių pritraukimo zonas. Jei fazinė erdvė—dvimatė erdvė, tai, parinkdami pritraukimo zonoms skirtingas spalvas, galime nesunkiai gauti spalvinį fazinį šios sistemos (iteracinio proceso) portretą. Keičiant spalvos parinkimo algoritmą, gaunami sudėtingi fraktaliniai paveikslai su nuostabiais įvairiaspalviais raštais.

Dar vieną žinomą fraktalų klasę sudaro stochastiniai fraktalai. Jie gaunami, iteracinio proceso metu atsitiktinai keičiant vienus ar kitus proceso parametrus. Generuojami objektai labai panašūs į gamtinius objektus—nesimetrinius medžius, išraižytas kranto linijas ir kita. Dvimačiai stochastiniai fraktalai, paprastai, naudojami modeliuojant vietovės reljefą bei jūros paviršių. [21]

Fraktalų tyrimai—tai nauja pažinimo sritis, kuri apjungia matematiką, teorinę fiziką, meną ir kompiuterines technologijas. Tai naujo tipo geometrijos atsiradimas. Atsiradimas geometrijos, kuri aprašo mus supantį pasaulį ir kurią tyrinėti galime ne tik remiantis vadovėliais, bet ir analizuojant procesus, vykstančius gamtoje bei visoje neaprėpiamoje Visatoje.

Daugumos ženklių mokslo apie fraktalus laimėjimų nebūtų buvę įmanoma pasiekti be skaičiavimo matematikos priemonių, kurios tiesiogiai remiasi šiuolaikinių kompiuterių taikymu. Milžiniški kompiuterių skaičiavimo pajėgumai leido pakankamai išsamiai apžvelgti įvairias fraktalines struktūras bei suprasti jų atsiradimo priežastis. Kita vertus, gana dažnai teorinis šių struktūrų modeliavimas aplenkdavo eksperimentinius metodus, kuriais tiriami realūs sudėtingos formos gamtiniai objektai.

Fraktalai bene labiausiai įtakojo kompiuterinės grafikos raidą. Kompiuterinės grafikos pagalba galima sukurti (monitoriaus ekrane) begalinę fraktalinių formų ir peizažų įvairovę, patalpinančią žiūrovą į nuostabią virtualiąją erdvę. Kompiuterių įsiveržimas į meno pasaulį daugeliu atvejų pakeitė grožio ir harmonijos, tapybos išraiškingumo ir aplinkinio pasaulio atkūrimo tikslumo supratimą.

Fraktalų grožį parodė ir visą pasaulį aplankiusi fraktalinių vaizdų paroda, surengta Chainco—Oto Peitgeno ir Peterio Richterio vadovaujamos Bremeno matematikų grupės (jos eksponatai vėliau buvo pateikti ir Peitgeno bei Richterio knygoje „ Fraktalų grožis “, [11]).

Fraktalinės geometrijos srityje taip pat daug yra nuveikę Klifordas A. Pikoveris (Clifford A. Pickover), Džeimsas Gleikas (James Gleick), G. O. Peitgenas (H. O. Peitgen) bei kiti. Šie mokslininkai stengėsi išplėsti naująją geometriją taip, kad ją būtų galima taikyti nuo vertybinių popierių biržos

prognozių iki naujų atradimų teorinės fizikos srityje. JAV fraktalinių modelių tyrimu užsiima specialus centras NCSA (Nacionalinis papildymų superkompiuteriams centras).

Reikia pabrėžti, jog geriausi ir ryškiausi rezultatai yra gauti, taikant fraktalinės geometrijos aparatą realaus pasaulio objektų modeliavimui. To priežastis—fraktalai aprašo realų pasaulį (dažnai) geriau nei tradicinė fizika ar matematika. Iš tikrųjų, pakanka žvilgterti į keletą svarbesnių fraktalinės geometrijos praktinio panaudojimo sričių.

Kompiuterinės sistemos. Kompiuterių moksle fraktalai efektyviausiai panaudojami duomenų suspaudimui. Esmė tame, kad realųjį pasaulį fraktalinė geometrija aprašo ypač gerai. Duomenų (vaizdų) suspaudimo efektas didesnis nei taikant įprastinius metodus (JPEG, gif ir pan.). Kitas fraktalinio suspaudimo privalumas yra tas, kad, išdidinus vaizdą, nepasireiškia pikselizacijos efektas, t. y. po išdidinimo vaizdas dažnai atrodo netgi geriau.

Skysčių mechanika. Srautų turbulentiškumo tyrimai —dar viena niša fraktalų taikymui. Turbulentiniai srautai chaotiški, todėl juos sunku tiksliai modeliuoti. Gal būt vienintelė išeitis yra perėjimas prie fraktalinio jų pateikimo. Tai, savo ruožtu, labai palengvina inžinierių bei fizikų darbą, įsisavinant ir interpretuojant sudėtingų srautų dinamiką. Beje, fraktalų pagalba taip pat galima sumodeliuoti ugnies „ liežuvius“.

Telekomunikacijos. Duomenų perdavimui per atstumą pastaruoju metu pradedamos naudoti antenos, turinčios fraktalines formas. Jos yra mažesnės ir lengvesnės, savo kokybe pralenkiančios įprastines antenas.

Paviršių fizika. Fraktalai naudojami paviršių kreivumui aprašyti. Nelygus paviršius, paprastai, charakterizuojamas dviejų skirtingų fraktalų deriniu.

Medicina. Biosensorinių paveikslų gavimas, širdies ritmo modeliavimas.

Biologija. Chaotiškų (populiacijos) procesų modeliavimas.

Šiame darbe pristatomos ir analizuojamos fraktalinės interpoliacinės funkcijos, atskleidžiami realaus pasaulio objektų fraktalinio interpoliavimo privalumai, pateikiamos paties interpoliacinio proceso organizavimo procedūros.

II. FRAKTALINĖ GEOMETRIJA — NAUJAS MATEMATINIO MODELIAVIMO METODAS

Fraktalinė geometrija — tai šiuolaikinės matematikos šaka, apjungianti klasikinę (Euklido) geometriją, topologiją, mato bei dinaminių sistemų teorijas. Benua Mandelbrotas — fraktalų geometrijos pradininkas — šią geometriją pavadino gamtos geometrija [7]. Būtent jisai surado nišą įvairiems „patologiniams“ (klasikinei matematinei analizei „nepaklūstantiems“) objektams bei reiškiniams (Kantoro dulkės, Peano kreivės, Vejerštraso funkcijos ir kt.). Būtent B. Mandelbroto dėka fraktalinė geometrija tapo taikomuoju mokslu. Fraktalų panaudojimo sritys nuolat plečiasi. Jeigu prieš dešimt metų didžiausias dėmesys buvo skiriamas realaus pasaulio objektų (debesys, kalnai, jūros paviršius ir pan.) modeliavimui, tai dabar fraktalinės geometrijos, kaip matematinio įrankio, taikymo sričių spektras kur kas platesnis. Tai — duomenų suglaudinimas (kompiuterinė grafika), techninė kainų analizės teorija (ekonomika), kietųjų kūnų paviršių analizė ir sintezė (fizika), biosensorinių tarpusavio poveikių tyrimas (medicina) ir t. t.

Žemiau glaustai pristatomas kontekstas, reikalingas fraktalo sąvokos įvedimui, pateikiami pakankamai detalūs fraktalų sintezės (generavimo) algoritmų aprašai bei komentuojami fraktalų tipai.

2.1. Fraktalai, fraktalų sintezė

Kadangi tiriamojo darbo objektas yra teoriniai ir praktiniai fraktalinių interpoliacinių funkcijų analizės aspektai, tai pagrindinį dėmesį toliau skirsime geometriniams fraktalams, gaunamiems taikant iteruotąsias (afiniųjų) funkcijų sistemas.

2.1.1. Iteruotųjų funkcijų sistemos ir jų atraktoriai (fraktalai)

Imkime metrinę erdvę (X, d) ir joje apibrėžtą transformaciją $f: X \to X$. Pastebėsime, jog $f(S)=\{f(x) \mid x \in S\}$, kai $S \subset X$. Be to, transformacija f yra apgręžiama, kai ji yra abipusiškai vienareikšmė ir $f(X)=X$. Šiuo atveju, galima apibrėžti atvirkštinę transformaciją $f^{-1}: X \to X$ tokią, kad $f^{-1}(y)=x$, ir $x \in X$ yra vienintelis taškas, su kuriuo $f(x)=y$.

Transformacijos $f: X \to X$ iteracijomis pirmyn vadinamos transformacijos $f^{0n}: X \to X$, apibrėžiamos lygybėmis:

$$f^{00}(x) = x,\ f^{01}(x) = f(x),\ f^{0(n+1)}(x) = f(f^{0n}(x)),\ \text{su visais } n = 0, 1, 2, \ldots$$

Jeigu f yra apgręžiama, tai transformacijos f iteracijomis atgal vadinamos transformacijos $f^{0(-m)}$: $X \to X$, apibrėžiamos lygybėmis:

$$f^{0(-1)}(x) = f^{-1}(x), f^{0(-m)}(x) = (f^{0m})^{-1}(x), \text{ su visais } m = 1, 2, 3, \ldots$$

Praktiniuose taikymuose, svarbiausia akcentuoti ryšį tarp transformacijas apibūdinančių formulių ir jų poveikyje atsirandančių geometrinių pasikeitimų (ištempimų, poslinkių, lenkimų ir pan.) erdvėse. Be to, svarbu tai, kaip transformacijos veikia ne atskirus erdvės X taškus, o jos poaibius.

Kalbant apie geometrinius fraktalus, jų prigimtį, paprastai, imamos Euklido metrinės erdvės — $(\mathbb{R}, d)$, $(\mathbb{R}^2, d)$ arba $(\mathbb{R}^3, d)$ —ir joje veikiančios afiniosios transformacijos.

Bendru atveju, afinioji transformacija, veikianti (tarkime, dvimatėje) Euklido erdvėje $(\mathbb{R}^2, d)$, užrašoma taip — ω: $\mathbb{R}^2 \to \mathbb{R}^2$, ir apibrėžiama lygybe — $\omega(x)=\omega(x_1, x_2)=(ax_1 + bx_2 + e, cx_1 + dx_2 + f)$, su visais $x=(x_1, x_2)\in\mathbb{R}^2$; čia a, b, c, d, e, f yra realieji skaičiai (afiniosios transformacijos ω parametrai). Afiniosios transformacijos turi daug svarbių geometrinių ir algebrinių savybių. Jų pagalba galima realizuoti posūkio, atspindžio, panašumo, pražulniąsias ir kitas transformacijas Euklido erdvėje $(\mathbb{R}^2, d)$ [14].

Jeigu su visais $x, y\in \mathbb{R}^2$ teisinga nelygybė $d(\omega(x), \omega(y))\leq s\cdot d(x,y)$, $0\leq s<1$, tai afinioji transformacija ω: $\mathbb{R}^2\to \mathbb{R}^2$ vadinama suspaudžiančiąja, o skaičius s — afiniosios transformacijos suspaudimo koeficientu.

Imkime afiniųjų suspaudžiančiųjų transformacijų ω_i: $\mathbb{R}^2\to\mathbb{R}^2$, $i= 1, 2, \ldots, N$, rinkinuką; atskirų afiniųjų transformacijų suspaudimo koeficientus pažymėkime s_i, $i= 1, 2, \ldots, N$. Tada, Euklido erdvė $(\mathbb{R}^2, d)$ su joje veikiančių suspaudžiančiųjų afiniųjų transformacijų rinkiniu vadinama iteruotųjų funkcijų sistema, ir žymima — IFS$\{\mathbb{R}^2; \omega_1, \omega_2, \ldots, \omega_N\}$. IFS suspaudimo koeficientu laikomas skaičius $s= \max\{s_1, s_2, \ldots, s_N\}$.

Apibrėžkime dar vieną metrinę erdvę $(\acute{H}(\mathbb{R}^2), h)$ tokiu būdu: $\acute{H}(\mathbb{R}^2)$— visų netuščių uždarųjų aibės $\mathbb{R}^2$ poaibių aibė; h — metrika, nusakanti atstumą tarp bet kurių dviejų aibės $\acute{H}(\mathbb{R}^2)$ elementų (aibės $\mathbb{R}^2$ poaibių), būtent:

$$h(A, B)= \max\{d(A, B), d(B, A)\};$$

čia $d(A, B)= \max\limits_{x\in A}\{\min\limits_{y\in B}\{d(x,y)\}\}$; $d(B, A)= \max\limits_{y\in B}\{\min\limits_{x\in A}\{d(x,y)\}\}$.

Įvestoji erdvė dažnai vadinama fraktaline erdve.

Perkelkime IFS sudarančias afiniąsias transformacijas į erdvę $(\acute{H}(\mathbb{R}^2), h)$. Tada ω_i: $\acute{H}(\mathbb{R}^2) \to \acute{H}(\mathbb{R}^2)$, apibrėžiama lygybe $\omega_i(B)= \{\omega_i(y) \mid y\in B\}$, $\forall B\in \acute{H}(\mathbb{R}^2)$, yra suspaudžiančioji transformacija erdvėje $(\acute{H}(\mathbb{R}^2), h)$, ir jos suspaudimo koeficientas s_i, $i= 1, 2, \ldots, N$.

Pagaliau, apibrėžkime dar vieną transformaciją fraktalinėje erdvėje $(\acute{H}(\mathbb{R}^2), h)$, būtent:

$$W: \acute{H}(\mathrm{R}^2) \to \acute{H}(\mathrm{R}^2); \text{ čia } W(B)=\omega_1(B)\cup \omega_2(B)\cup \ldots \cup\omega_N(B)= \bigcup_{i=1}^{N}\omega_i(B) \text{ , } \forall B\in\acute{H}(\mathbb{R}^2).$$

Galima įsitikinti, jog pastaroji transformacija irgi yra suspaudžiančioji, t.y.

$h(W(B), W(C)) \leq s \cdot h(B, C)$, su visais $B, C \in \acute{H}(\mathbb{R}^2)$; be to, $s = \max\{s_1, s_2, \ldots, s_N\}$.

Vienintelis nejudamasis transformacijos $W: \acute{H}(\mathbb{R}^2) \to \acute{H}(\mathbb{R}^2)$ taškas A ($A \in \acute{H}(\mathbb{R}^2)$) toks, kad $A = W(A) = \bigcup_{i=1}^{N} \omega_i(A) = \lim_{n\to\infty} W^{0n}(B)$, $\forall B \in \acute{H}(\mathbb{R}^2)$, vadinamas IFS atraktoriumi (arba fraktalu) [14]. Daug iteruotųjų funkcijų sistemų bei jų atraktorių pavyzdžių galima rasti literatūroje [14].

2.1.2. Fraktalų sintezės algoritmai

Yra žinomi ir plačiau praktikoje taikomi keli IFS atraktorių sintezės (generavimo) algoritmai. Tai — determinuotasis algoritmas, atsitiktinių iteracijų algoritmas bei „pabėgimo laiko“ algoritmas.

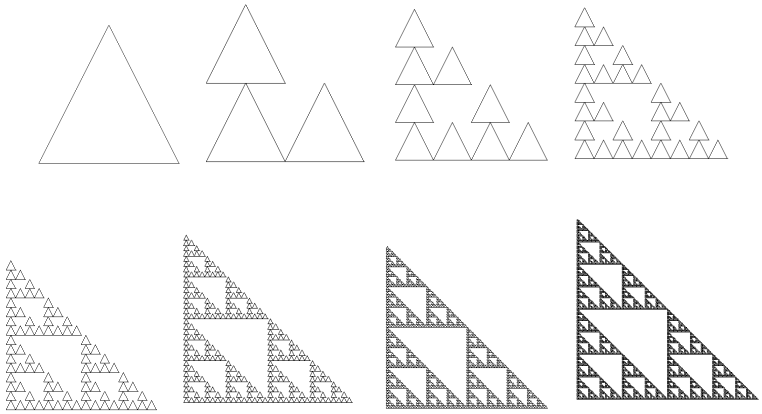
Žemiau pateikiami detalesni šių algoritmų aprašai.

Determinuotojo fraktalų (IFS atraktorių) generavimo algoritmo „veikimas“ tiesiogiai remiasi IFS atraktoriaus apibrėžimu.

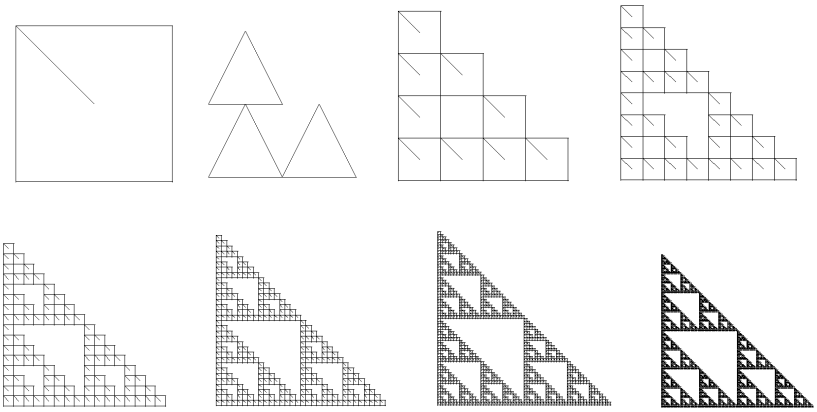
Tarkime, kad $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots, \omega_N\}$ yra iteruotųjų funkcijų sistema (IFS); čia ω_i ($i = 1, 2, \ldots, N$) yra suspaudžiančiosios afiniosios transformacijos. Parenkame (pradinę) uždarąją aibę $A_0 \subset \mathrm{R}^2$ ($A_0 \in \acute{H}(\mathbb{R}^2)$,) ir nuosekliai formuojame aibių seką $\{A_n\}$, būtent:

$A_n = W^{0n}(A_0) = W(A_{n-1}) = \bigcup_{i=1}^{N} \omega_i(A_{n-1})$, su visais $n = 1, 2, \ldots$ Įrodoma, jog tai (Koši) seka, kuri konverguoja į IFS atraktorių A. Kai n yra pakankamai didelis skaičius, aibė A_n vizualiai yra „artima“ aibei (fraktalui, IFS atraktoriui) A.

Kaip matome, visiškai nesvarbu, kokia imama pradinė uždaroji aibė A_0. Iteracinės procedūros rezultatas yra vienas ir tas pats — IFS atraktorius A. Kitaip tariant, aibė A pilnai nusakoma afiniųjų transformacijų ω_i, $i = 1, 2, \ldots, N$, veikiančių fraktalinėje erdvėje ($\acute{H}(\mathbb{R}^2)$, h), išraiškomis (2.1 pav.)



arba



2.1 pav. Determinuotojo IFS atraktoriaus sintezės algoritmo veikimo rezultatas

Pateiksime atsitiktinių iteracijų algoritmo aprašą. Vėlgi, tarkime, kad $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots, \omega_N\}$ yra iteruotųjų funkcijų sistema. Kiekvienai suspaudžiančiąjai afiniąjai transformacijai ω_i (i= 1, 2, …, N) yra priskiriamas teigiamas skaičius (tikimybė) p_i taip, kad $p_1+ p_2+ \ldots + p_n=1$. Jeigu $\omega_i=\omega_i(x)=\omega_i(x_1,x_2)=(a_ix_1+b_ix_2+e_i,\ c_ix_1+d_ix_2+f_i)$, i= 1, 2, …, N, tai apytikslės tikimybių p_i reikšmės randamos iš formulės

$$p_i \cong \frac{|a_i d_i - b_i c_i|}{\sum_{j=1}^{N} |a_j d_j - b_j c_j|}.$$

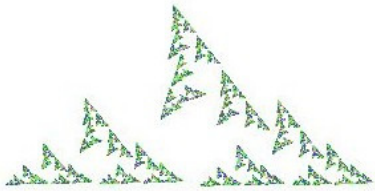
Jeigu su kuria nors reikšme i, $a_id_i-b_ic_i= 0$, tai p_i yra pakankamai mažas teigiamas skaičius (tarkim, p_i= 0,001).

Parenkamas pradinis taškas x_0 ($x_0\in \mathbb{R}^2$). Konstruojama erdvės $\mathbb{R}^2$ taškų seka $\{x_n\}_{n=0}^{\infty}$; čia $x_n\in \{\omega_1(x_{n-1}),\omega_2(x_{n-1}), \ldots, \omega_N(x_{n-1})\}$, su visais n= 1, 2, …, ir įvykio, jog x_n įgys reikšmę $\omega_i(x_{n-1})$, tikimybė yra lygi p_i, t. y. $P\{x_n= \omega_i(x_{n-1})\}= p_i$, i= 1, 2, …, N.

Atsitiktinių iteracijų algoritmo „veikimas“ yra grindžiamas fraktalo (IFS atraktoriaus) taškams būdinga chaotiška judesio dinamika. Keletas IFS kodų bei juos atitinkančių fraktalinių vaizdų parodyti 2.2 pav.

(a) IFS { X; ω_1, ω_2 , ω_3, ω_4} atraktorius, gautas taikant atsitiktinių iteracijų algoritmą; IFS kodas:

{ 0.50 0.00 0.00 0.50 -0.49 0.16 0.39
0.50 0.00 0.00 0.50 0.48 -0.16 0.39
-0.02 -0.28 0.17 0.14 -0.37 0.36 0.07
-0.21 -0.43 0.14 -0.18 -0.82 -0.10 0.15 }

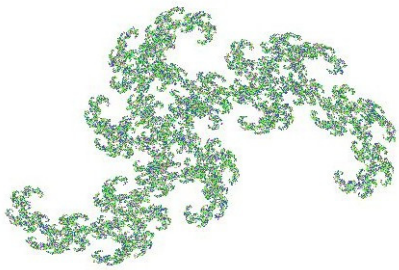


(b) IFS { X; ω_1, ω_2 , ω_3} atraktorius, gautas taikant atsitiktinių iteracijų algoritmą; IFS kodas:

{0.48 0.01 -0.01 0.47 0.53 -0.34 0.35

0.47 0.01 -0.01 0.47 -0.55 -0.28 0.35

-0.25 0.34 -0.36 -0.26 0.03 -0.06 0.30}



(c) IFS { X; ω_1, ω_2 } atraktorius, gautas taikant atsitiktinių iteracijų algoritmą; IFS kodas:

{ -0.58 -0.50 0.50 -0.58 0.46 -0.24 0.61

0.52 -0.45 0.45 0.32 -0.85 0.43 0.39 }

2.2 pav. Iteruotųjų funkcijų sistemų atraktoriai (fraktaliniai vaizdai), gauti taikant atsitiktinių iteracijų algoritmą

„Pabėgimo laiko“ algoritmas žinomas daugiau kaip teorinis rezultatas (idėja), jo praktinė realizacija iki šiol dar kelia tam tikrų problemų.

Pateikiame pačią idėją (algoritmą).

Tarkime, kad IFS{R^2; ω_1, ω_2, …, ω_N} atraktorius yra aibė A (akivaizdu, $A=\omega_1(A)\cup\omega_2(A)\cup\ldots\cup\omega_N(A)$). Tarkime, taipogi, jog visos afiniosios transformacijos yra abipusiškai vienareikšmės, t. y. apgręžiamos.

Duotąjai IFS konstruojama dinaminė poslinkių sistema {A; S}, kai poslinkio transformacija S: $A\to A$ apibrėžiama (kiekvienam taškui $a\in A$) taip: $S(a)=\omega_i^{-1}(a)$, jeigu $a\in\omega_i(A)$ ($i\in$ {1, 2, …, N}) ir $a\notin\bigcup_{j=1(j\neq i)}^{N}\omega_j(A)$; sakoma, jog taškas a yra transformacijos ω_i^{-1} veikimo zonoje; $S(a)=\omega_t^{-1}(a)$, jeigu $a\in\bigcap_{s=1}^{r}\omega_{i_s}(A)$; čia $t\in$ {i_1, i_2, …, i_r}$\subset$ {1, 2, …, N}; sakoma, jog taškas a yra bet kurios iš transformacijų ω_t^{-1} ($t=i_1, i_2, ..., i_r$) veikimo zonoje.

Su duotąja IFS susijusi dinaminė poslinkių sistema {A; S} praplečiama į visą erdvę $\mathbb{R}^2$, t. y. konstruojama nauja dinaminė sistema {R^2; $\hat{S}$ }. Naujoji poslinkio transformacija $\hat{S}$: $\mathbb{R}^2\to\mathrm{R}^2$

apibrėžiama lygybe $\hat{S}(x)=\omega_i^{-1}(x)$, kai taškas $x\in \mathbb{R}^2$ patenka į i-tosios atvirkštinės afiniosios transformacijos ω_i^{-1} ($i\in\{1, 2, \ldots, N\}$) veikimo zoną. Aišku, kad $\hat{S}$ sutampa su S, kai taškas $x\in A$.

Toliau, tarkime, kad IFS atraktorius A patenka į stačiakampį R, t. y. $A\subset R\subset\mathbb{R}^2$. Kiekvienam stačiakampio R taškui x konstruojama orbita $\{\hat{S}^{0n}(x)\}_{n=1}^{\tau}$; čia τ—iš anksto apibrėžtas iteracijų skaičius.

Fiksuokime skritulį su centru stačiakampio įstrižainių susikirtimo taške (pažymėkime O) ir spinduliu R; beje, skritulys talpina savyje stačiakampį R.

Jeigu $d(\hat{S}^{0\tau}, O)\le R$, t. y. po τ iteracijų taško $x\in R$ orbita nepalieka skritulio, daroma išvada, jog taškas x priklauso IFS atraktoriui, t. y. $x\in A$; priešingu atveju ($d(\hat{S}^{0\tau}, O)> R$), $x\notin A$.

„Pabėgimo laiko“ algoritmo „veikimas“ remiasi tuo, jog IFS atraktorius (aibė A) yra transformacijos $\hat{S}$: $\acute{H}(\mathbb{R}^2)\to\acute{H}(\mathbb{R}^2)$ atstumiantysis nejudamasis taškas. Kitaip tariant, taškų, esančių arčiau (metrikos h prasme) atraktoriaus A, orbitos ilgiau „užsibūna“ skritulyje, negu orbitos taškų, labiau nutolusių nuo atraktoriaus.

Pagrindinė kliūtis, dėl kurios šis algoritmas sunkokai pritaikomas praktikoje, — tai nebuvimas afiniųjų transformacijų veikimo zonų atskyrimo kriterijaus.

Banacho teorema apie suspaudžiančiuosius atvaizdžius teigia, kad „bet kuriam pertvarkiui egzistuoja „nejudamasis taškas x_{nej}“, kuris šiuo pertvarkiu atvaizduojamas pats į save: $x_{nej}=f(x_{nej})$. Taškas x_{nej} vadinamas atraktoriumi“ [14].

M. Barnsley apibendrino (1985 m.) Banacho teoremą keliems pertvarkiams. Tai Koliažo teorema.

Teorema / Koliažo /. Tarkime, kad (X, d) yra pilnoji metrinė erdvė, $T\in\acute{H}(X)$ yra duotoji aibė ir $\varepsilon\ge 0$. Parinkime IFS (arba IFS su kondensacija) $\{X; (\omega_0), \omega_1, \omega_2, \ldots, \omega_N\}$, kurios suspaudimo koeficientas s ($0\le s<1$), taip, kad

$$h\left(T,\bigcup_{i=1}^{N}\omega_i(T)\right)\le\varepsilon.$$

Tada $h(T, A)\le\dfrac{1}{1-s}\cdot h\left(T,\bigcup_{\substack{i+1\\(i=0)}}^{N}\omega_i(T)\right)\le\dfrac{\varepsilon}{1-s}$; čia A yra IFS atraktorius.

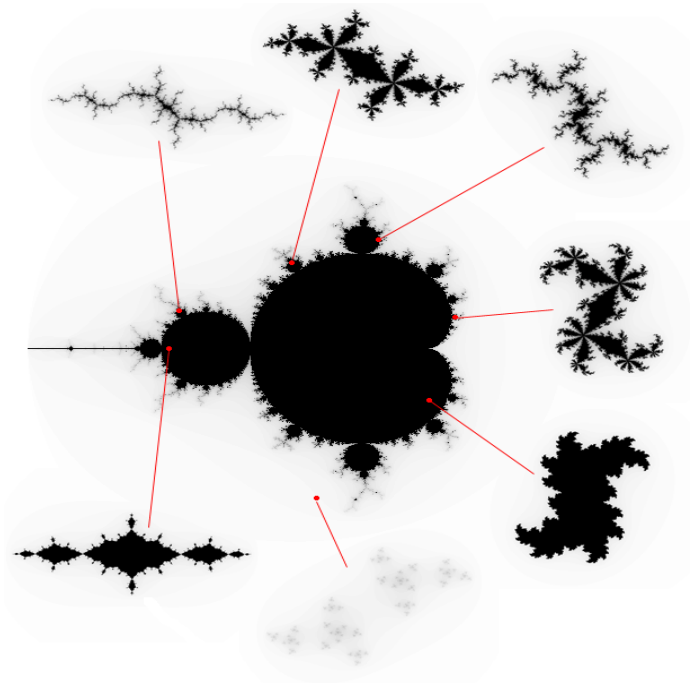
Šiame darbe, fraktalinių interpoliacinių funkcijų analizei mes naudojome atsitiktinių iteracijų algoritmą. Būtent šis algoritmas pasižymi spartumu bei gera sintezuojamų fraktalinių vaizdų kokybe

2.1.3. Fraktalų tipai

Fraktalinėje geometrijoje išskiriami šie fraktalų tipai — geometriniai fraktalai, algebriniai fraktalai, stochastiniai fraktalai.

Geometriniai fraktalai siejami su ankstyvuoju fraktalinės geometrijos (kaip mokslo šakos) vystymosi periodu. Faktiškai, tai iteruotųjų funkcijų sistemų, charakterizuojamų afiniųjų transformacijų, veikiančių įvairaus matavimo Euklido erdvėse, rinkiniais, atraktorių aibė. Šio tipo fraktalai yra vaizdžiausi ir, galima sakyti, geriausiai išanalizuoti.

Algebriniai fraktalai priklauso pačiai didžiausiai fraktalų grupei. Jie siejami su netiesinėmis dinaminėmis sistemomis, veikiančiomis kompleksinėje plokštumoje. Šio tipo fraktalų sintezei gana efektyviai panaudojamas anksčiau aptartas „pabėgimo laiko“ algoritmas. Vienas ryškiausių šio tipo fraktalų yra Mandelbroto aibė ir su ja susijusios Žulija aibės (2.3 pav.). Pastarosios generuojamos, tiriant netiesinės parametrizuotos sistemos $\{\mathbb{C}; f_\lambda\}$, $f_\lambda(z) = z^2 - \lambda$, dinamiką. Čia pat pastebėsime, jog algebriniai fraktalai nėra pilnai ištirti, vis dar slepia savyje daug paslapčių.



2.3 pav. Mandelbroto ir Žulija aibės; $\{C; z^2-\lambda\}$ (centre—Mandelbroto (parametro reikšmių λ, kurias atitinkančios Žulija aibės yra jungios) aibė)

Stochastinius fraktalus, tam tikra prasme, galima būtų priskirti geometriniams fraktalams. Skirtumas tas, jog vietoj įprastinių iteruotųjų funkcijų sistemų imamos parametrizuotos sistemos. Iteracinio proceso metu keičiamos parametrų reikšmės, ir tai daroma atsitiktinai (atsitiktine tvarka). Rezultatas — gaunami fraktalai labai panašūs į realaus pasaulio objektus („asimetriniai“ medžiai,

karpytos pakrantės linijos ir pan.). Dvimačiai stochastiniai fraktalai naudojami, modeliuojant vietovės reljefą, jūros paviršių ir kt.

2.2. Fraktalinės dimensijos sąvoka

Skaičiai, charakterizuojantys fraktalus, paprastai, vadinami fraktalinėmis dimensijomis. Tai gana svarbios kiekybinės fraktalų charakteristikos. Jos leidžia įvertinti subjektyvų požiūrį apie tai, kaip „tirštai“ (tankiai) realaus pasaulio objektas (fraktalas) užpildo erdvę, kurios poaibis jis pats yra. Fraktalinė dimensija—tai objektyvi priemonė fraktalų palyginimui.

Istoriškai, svarbu apibrėžti Hausdorfo ir Bezicovičiaus matą bei dimensiją, ir kartu atskleisti su jų apskaičiavimu susijusius sunkumus [14].

Imkime poaibį $A\subset\mathbb{R}^n$; čia $(\mathbb{R}^n, d)$ —n-matė Euklido erdvė. Tegu $|A|=\mathrm{diam}(A)=\sup\{d(x,y)|\ x, y\in A\}$ žymi poaibio A diametrą.

Fiksuokime teigiamą skaičių $\varepsilon>0$. Imkime skaičią poaibių A_i ($|A_i| \le\varepsilon$) aibę $\{A_i\}$ ir tokią, kad ji padengtų poaibį A, t.y. $A\subset\bigcup_{i=1}^{\infty}A_i$; rinkinys (aibė) $\{A_i\}$ vadinamas poaibio A ε-denginiu.

Toliau, tarkime, kad $p>0$, ir apibrėžkime dydį $H_\varepsilon^p(A)$ tokiu būdu:

$$H_\varepsilon^p(A)=\inf\{\sum_{i=1}^{\infty}|A_i|^p \mid \{A_i\} \text{ yra poaibio } A\ \varepsilon\text{-denginys}\}.$$

Faktiškai, tai minimizavimo uždavinys, kai stengiamasi minimizuoti dengiančiųjų aibių diametrų, pakeltų p—tuoju laipsniu, sumą (pagal visus įmanomus poaibio A ε-denginius). Kita vertus, dydis $H_\varepsilon^p(A)$, kaip funkcija, yra nemažėjantis, kai ε artėja prie nulio. Pereidami prie ribos, kai $\varepsilon\to0$, apibrėšime Hausdorfo ir Bezicovičiaus matą (poaibiui A) tokiu būdu

$$H^p(A)=\lim_{\varepsilon\to0}H_\varepsilon^p(A).$$

Dydžio $H^p(A)$ priklausomybės nuo parametro p grafikas rodo, jog yra kritinė p reikšmė, kai $H^p(A)$ „šoka“ nuo ∞ prie 0 [14]. Tą kritinį tašką atitinkanti p reikšmė vadinama Hausdorfo ir Bezicovičiaus dimensija (poaibiui A), t.y.

$$D_H(A)=\sup\{p \mid H^p(A)=\infty\}=\inf\{p \mid H^p(A)=0\}.$$

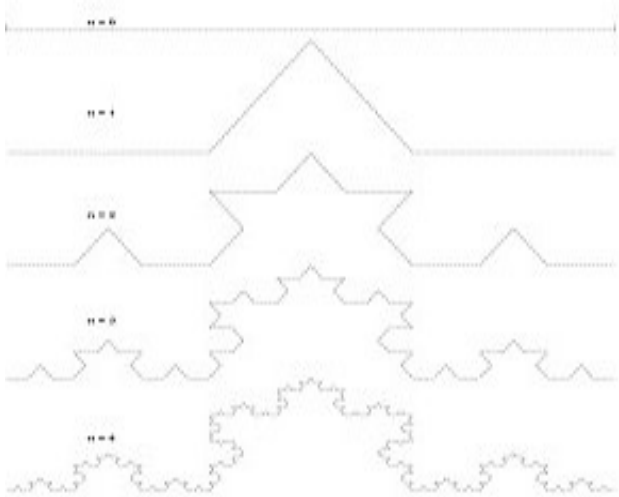
Pastebėsime, jog šią dimensiją apskaičiuoti netgi „paprastoms“ aibėms (fraktalams) yra pakankamai sudėtinga. Kaip pavyzdį, panagrinėkime Kocho kreivę [14]. Po k iteracijų kreivę sudaro 4^k segmentai, kurių kiekvieno ilgis 3^{-k}. Todėl kreivę galima padengti (tiksliai) 4^k aibėmis, kurių diametrai lygūs 3^{-k}. Tokiu būdu,

$$H_\varepsilon^p(A)=H_{3^{-k}}^p(A)=4^k(3^{-k})^p.$$

Pertvarkykime šį rezultatą taip:

$H_{\varepsilon}^{p}(A)=H_{3^{-k}}^{p}(A)=3^{k(D-p)}$; čia D=ln4/ln3. Pereidami prie ribos, kai dengiančiųjų aibių diametrai artėja prie nulio (t.y. $k\to\infty$), gauname

$$H^{p}=\lim_{k\to\infty}H_{3^{-k}}^{p}=\begin{cases}\infty, p<D\\ 1, p=D\\ 0, p>D\end{cases}.$$



2.4 pav. Van Kocho kreivė (fraktalas)

Taigi, galima teigti, jog Van Kocho kreivės Hausdorfo ir Bezicovičiaus dimensija lygi D=ln4/ln3≅1.262.

Vėliau, B. Mandelbrotas šią dimensiją pavadino fraktaline dimensija. Faktiškai, tai nauja erdvės (poaibių) matavimo priemonė.

Daugelyje šaltinių, fraktalinės dimensijos sąvokos įvedimui naudojamas šiek tiek supaprastintas požiūris (interpretacija).

Imkime metrinę erdvę ($\mathbb{R}^n$, d). Tegul $A\in$ Ή($\mathbb{R}^n$). Tarkime, kad $\varepsilon>0$ ir $N(A, \varepsilon)$ žymi mažiausią uždarųjų rutulių $B(x, \varepsilon)=\{y \in \mathbb{R}^n \mid d(x, y)\le\varepsilon\}$, sudarančių baigtinį poaibio A denginį, skaičių, t.y. $N(A, \varepsilon)$—mažiausias sveikasis skaičius toks, kad $A\subset \bigcup_{i=1}^{N(A,\varepsilon)} B(x_i,\varepsilon)$;

čia $\{x_1, x_2, \ldots, x_{N(A,\varepsilon)}\}\subset \mathbb{R}^n$.

Fraktalinės dimensijos sąvoka įvedama, remiantis intuityvia idėja, jog aibė A ($A\subset \mathbb{R}^n$) turi fraktalinę dimensiją D, jeigu

$$N(A,\varepsilon)\approx C\varepsilon^{-D}; \qquad \textbf{(*)}$$

čia: C — tam tikra konstanta; „≈ “ reiškia, jog $\lim_{\varepsilon\to 0}(\ln f(\varepsilon)/\ln g(\varepsilon))=1$, kai $f(\varepsilon)\approx g(\varepsilon)$.

Dabar, išsprendę **(*)** „lygybę“ (D atžvilgiu), gauname

$$D \approx \frac{\ln N(A,\varepsilon) - \ln C}{\ln \frac{1}{\varepsilon}}.$$

Perėję prie ribos ($\varepsilon \to 0$), turime:

$$D = \lim_{\varepsilon \to 0} \frac{\ln N(A,\varepsilon)}{\ln(\frac{1}{\varepsilon})}.$$

Ši riba vadinama poaibio A ($A \subset \mathbb{R}^n$) fraktaline dimensija ir žymima $D(A) = D$.

Žemiau pateikiame keletą svarbesnių teiginių, skirtų teoriniam ir eksperimentiniam fraktalinės Euklido erdvės poaibių dimensijos nustatymui.

1 teorema. Tegul $A \subset$ ΄H($\mathbb{R}^m$); be to, $\varepsilon_n = Cr^n$ ($C>0$, $0<r<1$), $n=1, 2, \ldots$ Poaibis A turi fraktalinę dimensiją D, apibrėžiamą formule:

$$D(A) = D = \lim_{n \to \infty} \left\{ \frac{\ln N_n(A)}{\ln(\frac{1}{\varepsilon_n})} \right\}.$$

Ši teorema leidžia tolydųjį kintamąjį ε pakeisti diskrečiuoju kintamuoju ε_n ; supaprastėja skaičiavimo procedūra. Teoremos (kaip ir visų kitų teiginių) įrodymą galima rasti knygoje [14].

2 teorema. Tarkime, kad $A \subset$ ΄H($\mathbb{R}^m$); be to, erdvė $\mathbb{R}^m$ padengta nesusikertančiais m—mačiais „kubiukais“ (dėžutėmis), kurių briaunos ilgis lygus $1/2^n$; tegul $N_n(A)$ žymi „kubiukų“, persidengiančių su poaibiu A, skaičių. Tada skaičius

$$D = \lim_{n \to \infty} \left\{ \frac{\ln N_n(A)}{\ln(2^n)} \right\}$$

vadinamas poaibio A fraktaline dimensija.

Būtent pastarasis teiginys sudaro (dažniausiai) pagrindą eksperimentiniam realaus pasaulio (fraktalinio) objekto dimensijos nustatymui (dimensijos įverčio gavimui).

Kaip pavyzdį, panagrinėkime fraktalą A (2.5 pav. Sierpinskio trikampis). Nesunku pastebėti, jog:

$N_1(A)=3$, kai $n=1$;

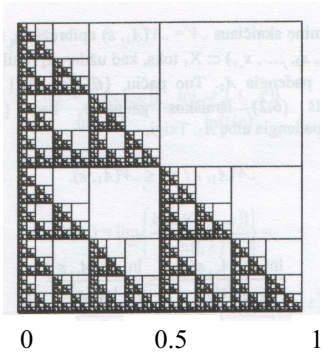
$N_2(A)=9$, kai $n=2$, t.y. kai $\varepsilon_2 = \frac{1}{2^2} = \frac{1}{4}$;

$N_3(A) = 3^3 = 27$, kai $n = 3$ ir t.t.

Pagaliau, $N_n(A) = 3^n$, kai $\varepsilon_n = \frac{1}{2^n}$

Taigi, fraktalinė Sierpinskio trikampio dimensija lygi:

$$D = \lim_{n \to \infty} \frac{\ln 3^n}{\ln 2^n} = \frac{\ln 3}{\ln 2} \cong 1.585.$$



2.5 pav. Sierpinskio trikampis

3 teorema. Jeigu $A, B \in$ H($\mathbb{R}^n$) ir $A \subset B$, tai $D(A) \leq D(B)$; be to, $D(A) \geq 0$, $D(B) \leq n$.

4 teorema. Jeigu $A, B \in$ H($\mathbb{R}^n$) ir $D(A)$, $D(B)$ bei $D(A \cup B)$ žymi atitinkamai aibių A, B ir $A \cup B$ fraktalines dimensijas, tai $D(A \cup B) = D(A)$, kai $D(B) < D(A)$.

5 teorema. Tarkime, kad $A \subset \mathbb{R}^n$ sutampa su visiškai nejungios IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots \omega_N\}$, sudarytos iš afiniųjų panšumo transformacijų, atraktoriumi. Tada, poaibio A fraktalinė dimensija D yra lygties

$$\sum_{i=1}^{N} s_i^D = 1$$

sprendinys; čia s_i yra i—tosios afiniosios transformacijos suspaudimo koeficientas, i=1, 2, …, N.

Pavydžiui, Kocho kreivės atveju (2.4 pav.), pastarojo teiginio taikymas leidžia gauti tokį rezultatą. Kadangi Kocho kreivė yra IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \omega_3, \omega_4\}$ atraktorius, be to, s_i=1/3, su visais i=1, 2, 3, 4, tai $\sum_{i=1}^{4}\left(\frac{1}{3}\right)^D = 1$;

iš čia $4 \cdot \left(\frac{1}{3}\right)^D = 1$;

$$\left(\frac{1}{3}\right)^D = \frac{1}{4};$$

$$D = \frac{\ln 4}{\ln 3} \cong 1.262$$

6 teorema. Tarkime, kad $D(A)$ ir $D_H(A)$ žymi kokio nors aprėžto poaibio $A \subset \mathbb{R}^n$ atitinkamai fraktalinę bei Hausdorfo ir Bezicovičiaus dimensijas. Tada

$$0 \leq D_H(A) \leq D(A) \leq n.$$

Galima būtų teigti, jog $D_H(A)$ „šiek tiek subtiliau“ charakterizuoja poaibį A, negu $D(A)$.

Yra ir daugiau požiūrių bei interpretacijų, pateikiant fraktalines Euklido erdvės poaibių dimensijas. Tai — koreliacijos dimensija, informacijos dimensija, Liapunovo dimensija ir panašiai. Plačiau apie jas čia nekalbėsime.

2.3. Fraktalinės interpoliacinės funkcijos, jų specifika

Elementariosios funkcijos (sinusas, kosinusas, daugianariai ir kt.) sudaro tradicinio eksperimentinių duomenų analizės metodo pagrindą. Tarkime, kad eksperimento metu matuojamos tam tikros realiosios funkcijos $F(x)$ reikšmės. Eksperimento rezultatas — duomenų rinkinys $\{(x_i, F_i) \mid i=0, 1, \ldots, N\}$; čia $F_i= F(x_i)$, $i=0, 1, \ldots, N$, ir $x_0< x_{1<} \ldots< x_N$. Šie duomenys pavaizduojami grafiškai (erdvėje $\mathbb{R}^2$) ir analizuojami, t. y. stengiamasi parinkti galimai žemesnio lygio daugianarį (atskiru atveju, „laužtę“), kuris „neblogai modeliuotų“ duomenis segmente $[x_0, x_N]$, t.y. kurio grafikas „eitų“ per taškus (x_i, F_i), $i=0, 1, \ldots, N$.

Tačiau, grafikos sistemoms kartais keliami šiek tiek didesni reikalavimai, būtent: galimybė modeliuoti realaus pasaulio (fizinius) objektus, tokius kaip debesys, kalnų masyvo profilis, kabantys stalaktitai ir panašiai. Šiems fiziniams objektams (sistemoms) būdinga tai, kad kiekvienos sistemos dalies struktūra dažnai „atkartoja“ visos sistemos struktūrą. Tai galima aiškinti tuo, jog sistemą įvairiuose jos lygiuose veikiančios ir formuojančios jėgos yra panašios. Natūralu, jog Euklido geometrijos ir elementariųjų funkcijų čia jau nebepakanka.

Pateiksime fraktalinės interpoliacinės funkcijos sąvoką, parodysime, jog minėto tipo duomenims galima parinkti funkcijas, kurios (Hausdorfo metrikos prasme) būtų „artimos“ tiems duomenims; be to, įmanoma užtikrinti, kad fraktalinės interpoliacinės funkcijos grafiko dimensija sutaptų su eksperimento duomenų (grafiko) fraktaline dimensija.

Tarkime, kad taškų rinkinys $\{(x_i, F_i)\in \mathbb{R}^2 \mid i=0, 1, \ldots, N\}$ sudaro duomenų aibę; čia $x_0< x_{1<} \ldots< x_N$. Tolydžioji funkcija f: $[x_0, x_N]\to \mathbb{R}$ tokia, kad $f(x_i)=F_i$, $i=0, 1, \ldots, N$, vadinama interpoliacine funkcija, atitinkančia duomenų aibę. Taškai (x_i, F_i), $i=0, 1, \ldots, N$, vadinami interpoliavimo taškais.

Pasirodo, jog erdvėje $\mathbb{R}^2$ galima sukonstruoti IFS, kurios atraktorius (aibė A) sutaptų su tolydžiosios duomenis interpoliuojančios funkcijos f: $[x_0, x_N] \to \mathbb{R}$ grafiku. Tam pakanka imti pražulniąsias (nėra posūkio y—ašies atžvilgiu) afiniąsias transformacijas, t.y.

$$\omega=\omega(x,y)=\begin{pmatrix} r_1\cos\Theta^\circ - r_2\sin 0^\circ \\ r_1\sin\Theta^\circ r_2\cos 0^\circ \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}+\begin{pmatrix} e \\ f \end{pmatrix}=\begin{pmatrix} a & 0 \\ b & c \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}+\begin{pmatrix} e \\ f \end{pmatrix}, \text{ kai } (x, y)\in \mathbb{R}^2.$$

Kita vertus, afinioji transformacija

$$\omega_i=\omega_i(x, y)=\begin{pmatrix} a_i & 0 \\ c_i & d_i \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}+\begin{pmatrix} e_i \\ f_i \end{pmatrix}, i\in\{1, 2, \ldots, N\},$$

turi atvaizduoti duomenų aibę (nuo x_0 iki x_N) į „juostą“ (nuo x_{i-1} iki x_i), t.y. turi būti tenkinamos sąlygos:

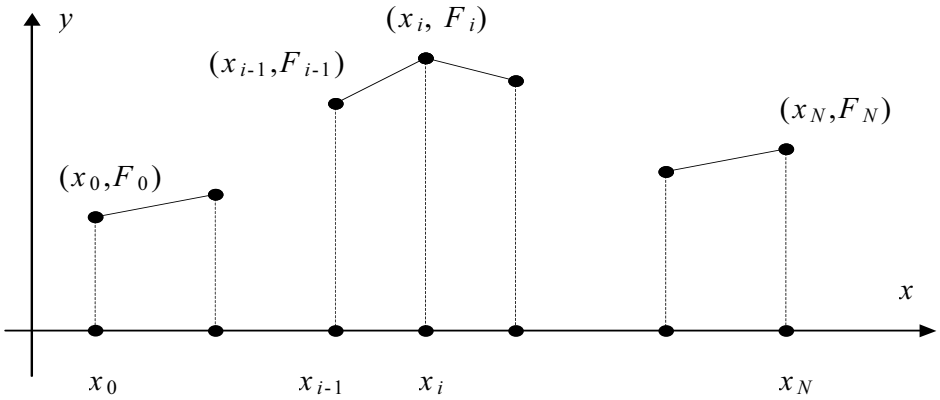
$$\begin{cases} a_i x_0 + e_i = x_{i-1} \\ a_i x_N + e_i = x_i \\ c_i x_0 + d_i F_0 = f_i = F_{i-1} \\ c_i x_N + d_i F_N + f_i = F_i \end{cases}$$

Nesunku pastebėti, jog šioje tiesinių algebrinių lygčių sistemoje vienas kintamasis (tarsime, kad tai vertikalųjį mastelį keičiantis parametras d_i) yra laisvai pasirenkamas. Tada

$$a_i = \frac{x_i - x_{i-1}}{x_N - x_0},\ c_i = \frac{F_i - F_{i-1}}{x_N - x_0} - d_i \frac{F_n - F_0}{x_N - x_0}$$

$$e_i = \frac{x_N x_{i-1} - x_0 x_i}{x_N - x_0},\ f_i = \frac{x_N F_{i-1} - x_0 F_i}{x_N - x_0} - d_i \frac{x_N F_0 - x_0 F_N}{x_N - x_0}. \qquad (*)$$

Jeigu parinktume d_i=0, su visais i=1, 2, …, N, tai tokios IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots \omega_N\}$, atraktorius A sutaptų su laužte (dalimis tiesine interpoliacine funkcija, kurios grafikas „eina“ per interpoliavimo taškus (2.6 pav.))



2.6 pav. Dalimis tiesinės interpoliacinės funkcijos, einančios per interpoliavimo taškus, grafikas; šis grafikas sutampa su IFS $\{\mathbf{R}^2; \omega_1, \omega_2, \ldots \omega_N\}$ atraktoriumi

Jeigu parametrų d_i (i=1, 2, …, N) reikšmės nėra nulinės, tai IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots, \omega_N\}$atraktorius išlieka duomenis $\{(x_i, F_i) \in \mathbb{R}^2 \mid i=0, 1, \ldots, N\}$ interpoliuojančiąja funkcija, tačiau tai jau nėra dalimis tiesinė funkcija. To priežastys akivaizdžios — IFS sudaro pražulniosios afiniosios transformacijos, t.y. bet kokia tiesė (atkarpa), lygiagreti ašiai O_y, atvaizduojama (ω_i, $i \in \{1, 2, \ldots, N\}$, pagalba) į tiesę (atkarpą), lygiagrečią ašiai O_y. Be to, abiejų atkarpų (prieš ir po vaizdavimo) ilgių santykis išreiškiamas skaičiumi $|d_i|$. Nesunku pastebėti, jog parametrai d_i (i=1, 2, …, N) įtakoja IFS atraktoriaus (fraktalinės interpoliacinės funkcijos) fraktalinę dimensiją.

Pateikiame vieną (šia prasme labai svarbų) teiginį.

1 teorema. Tarkime, kad IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots \omega_N\}$ yra susieta su duomenų aibe $\{(x_i, F_i) \in \mathbb{R}^2 \mid i=0, 1, \ldots, N\}$; be to,

$$\omega_i(x, y)=\begin{pmatrix} a_i & 0 \\ c_i & d_i \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}+\begin{pmatrix} e_i \\ f_i \end{pmatrix};$$

čia koeficientai a_i, c_i, e_i ir f_i apibrėžiami (* 21 psl.) išraiškomis, su visais $i=1, 2, \ldots, N$; $0 \le d_i < 1$, $i=1, 2, \ldots, N$. Tarkim, kad A yra fraktalinės interpoliacinės funkcijos grafikas (IFS atraktorius).

Jeigu $\sum_{i=1}^{N} |d_i| > 1$ ir interpoliavimo taškai nepriklauso vienai tiesei, tai A turi fraktalinę dimensiją D, kuri yra vienintelis realusis lygties $\sum_{i=1}^{N} |d_i| \cdot a_i^{D-1} = 1$ sprendinys; priešingu atveju, $D(A) = 1$.

Šios teoremos ir kitų įdomių teiginių įrodymus galima rasti knygoje [14].

Pastebėsime, jog tuo atveju, kai interpoliavimo taškai yra pasiskirstę segmente $[x_0, x_N]$ tolygiai, t.y.

$x_i = x_0 + \frac{i}{N}(x_N - x_0)$, $i=0, 1, \ldots, N$, koeficientų a_i ($i=1, 2, \ldots, N$) reikšmės lygios $1/N$. Gauname

$$\sum_{i=1}^{N} |d_i| = N^{D-1}.$$

Pastarąją lygtį spręsdami D atžvilgiu, randame, jog:

$$D=1 + \frac{\ln(\sum_{i=1}^{N} | d_i |)}{\ln N}.$$

Vadinasi, manipuliuojant parametrų d_i ($i=0, 1, \ldots, N$) reikšmėmis, dydžiui $\sum_{i=1}^{N} |d_i|$ galima priskirti bet kurią reikšmę iš intervalo $[1, N]$, tuo pačiu fraktalinės interpoliacinės funkcijos (IFS atraktoriaus A) fraktalinę dimensiją prilyginti bet kuriai reikšmei iš intervalo [1,2]. Tai labai svarbus momentas, kadangi atsiranda galimybė „priderinti“ fraktalines interpoliacinių funkcijų dimensijas prie duomenų aibės (eksperimento rezultatų) fraktalinės dimensijos.

Įdomu ir tai, kad fraktalinė dimensija nepriklauso nuo reikšmių $\{F_i \mid i=0, 1, \ldots, N\}$, išskyrus tą faktą, jog interpoliavimo taškai neturi priklausyti vienai tiesei. Taigi, galima kalbėti apie fraktalinių interpoliacinių funkcijų, turinčių tą pačią dimensiją D, rinkinį. Tam pakanka išpildyti sąlygą—

$$\sum_{i=1}^{N} |d_i| = N^{D-1}.$$

Nors fraktalinės interpoliacinės funkcijos (FIF) buvo formalizuotos daugiau nei prieš dešimtmetį, tačiau tik pastaraisiais metais imta plačiau jas taikyti kompiuterinėje grafikoje. FIF taikomos kalbos signalų interpoliavimui, modeliuojant kalno vaizdą, seisminius duomenis,

elektrokardiogramas. Modeliuojant įvairius paviršius taikomos dvi sudėtinės FIF dalys: FIS (fraktalinės interpoliacinės sistemos) ir FIV (fraktaliniai interpoliaciniai paviršiai).

Be įprastinės FIF dar taikomos ir RFIF (Recurrent Fractal Interpolation Functions) ir PFIF (Partitioned Fractal Interpolation Functions). Rekurentinės FIF — FIF apibendrinimas, sudarantis lankstesnes sąlygas „grubių“ kreivių formavimui. RFIF leidžia naudoti atitinkamos srities dalių sąjungą, o PFIF supaprastina šį apribojimą ir leidžia naudoti laisvai pasirenkamą srities dalį.

Tad reikalingi metodai, skirti fraktalinių funkcijų, interpoliuojančių duotą duomenų rinkinį, formavimui. Tuo pačiu, FIF sudaro galimybes naudoti naujus eksperimentinių duomenų aproksimavimo metodus. W. O. Kochranas (W. O. Cochran), J. C. Hart Patrikas (J. C. Hart Patrik) bei J. Flainas (J. Flynn) savo straipsnyje apie „grubių“ kreivių aproksimavimą, naudojant fraktalines funkcijas, siūlo aproksimavimo problemoms spręsti taikyti Hugho (Hough) transformacijų, vaizdų suspaudimo bei mažiausiųjų kvadratų metodus [2].

Darbe bandoma analizuoti ir spręsti interpoliavimo taškų parinkimo (bendru atveju) problemą. Taipogi, siūlomas naujas požiūris į pražulniųjų afiniųjų transformacijų, sudarančių IFS, parinkimą.

2.4. Kai kurie fraktalų taikymo pavyzdžiai

Nors fraktalų teorija taikoma tik keletą dešimtmečių, tačiau ja paremti tyrimai (eksperimentai) leido mokslininkams paaiškinti galaktikų evoliuciją, gardelės vystymąsi, kalnodaros ir debesų susidarymo procesus, biržos kainų kaitos procesus ir netgi visuomenės bei šeimos vystymąsi, t.y. vis daugėja sričių, kuriose yra taikomi fraktalinės geometrijos teiginiai ir metodai.

Kaip jau buvo minėta įvade, fraktalai pirmiausiai buvo pritaikyti kompiuterinėje grafikoje (vaizdų suspaudimas, vaizdų modeliavimas ir pan.), kompiuterinėse technologijose (duomenų suspaudimas), fizikoje, medicinoje bei ekonomikoje.

Fraktalai daugelį pirmiausiai patraukia vaizdo neįprastumu ir spalvingumu. Jie— nuostabios matematinės dailės sritis—nesudėtingų formulių ir algoritmų pagalba gaunami nepaprastai gražūs, bet sudėtingi paveikslai, kuriuose galime įžvelgti lapus, medžius ar gėles.

Tad, kaip jau minėta, fraktalai pirmiausiai taikomi kompiuterinėje grafikoje: vaizdų fraktalinis suspaudimas, landšaftų, medžių, krūmų bei kitų augalų modelių kūrimas, fraktalinių tekstūrų generavimas. Fraktalinė geometrija nepakeičiama generuojant dirbtinius debesis, kalnus, jūros paviršius.

1991 m. M. Barnsley pateikė fraktalinio vaizdų suspaudimo algoritmą, kurio esmė ta, kad IFS parametrų pagalba vaizdas saugomas daug kompaktiškiau. Be to, galima išskirti keletą fraktalinio vaizdų suspaudimo privalumų: suspaustas vaizdas užima nedaug vietos ir jo atkūrimas užtrunka trumpiau, nei taikant kitus metodus; aukšta atkurto vaizdo kokybė; video suspaudimo galimybė. Savo

ruožtu, fraktalinio vaizdų suspaudimo algoritmas leidžia (kaip ir JPEG atveju) užduoti suspaudimo laipsnį (suspaudimas su praradimu). Algoritmo esmė: paieška didelių vaizdo fragmentų, kurie būtų panašūs į mažesnius fragmentus. Faile saugomos tik rastų atitikmenų poros. Tuo principu remdamiesi „Iterated“ darbuotojai sukūrė naują vaizdų formatą *.string, kuriuo įrašytas vaizdas, palyginus su nesuspaustu vaizdu, užima 15—20 % mažiau vietos.

Fraktalų panašumas į kalnus, gėles, medžius taikomas kai kuriuose grafiniuose redaktoriuose (fraktaliniai debesys— 3D Studio MAX, fraktaliniai kalnai— World Builder ir kt.).

Be to, eksperimentais yra įrodyta, kad norint kompiuteryje pavaizduoti trimatę gamtą, būtina kurti trimates fraktalines figūras, nes fraktalai pateikia didžiausią realių detalių kiekį, esant mažiausiam kompiuterio atmintyje saugomų duomenų kiekiui. Vietoj atmintyje saugomų visų kalno detalių galima aprašyti objektą fraktaline lygtimi, arba saugoti tik atraktorių, nusakantį visą objektą (analogiškai, trimačiams upės, jūros pakrančių vingių vaizdams gauti ir kt.).

Fraktalų panašumo savybė itin naudinga, formuojant naujas tekstūras. Juk kiek bedidintumėm, fraktalinis vaizdas nėra „stambinamas“, praktiškai, nuolat atstatomas pradinis vaizdas. Be to, fraktalo „neribotumas“ tiek į plotį, tiek į gylį, leidžia, labai sumažinus ar padidinus fraktalą, gauti, praktiškai, naujus paveikslėlius.

Fraktalo sąvokos atsiradimas fizikoje susijęs su kranto linijos ilgio nustatymo uždaviniu. Šį ilgį matuojant žemėlapiuose, buvo fiksuota įdomi situacija – kuo didesnis žemėlapio mastelis, tuo ilgesnė linija.

Pirmuoju šio tipo uždaviniu tapo anglų fiziko Ričardsono bandymas išmatuoti Anglijos pakrančių ilgį. Naudodamas detalų žemėlapį, jis aproksimavo pakrantės liniją kreive L_b, sudaryta iš atskirų stygų, kurių ilgis b ir kurių viršūnės— krante.

Fraktalų struktūra tokia sudėtinga, kad įtakoja fizikinius procesus, vykstančius fraktaluose, juos traktuojant kaip nešėjus. Fraktalai kitaip išsklaido spinduliavimą, kitaip svyruoja ir skamba, kitaip praleidžia elektrą, fraktaluose kitaip vyksta objekto difuzija. Atsiranda nauja pažinimo sritis —fraktalų fizika. Fraktalai tampa patogiais modeliais, kažkuo panašiais į integruojamas klasikinės mechanikos užduotis, procesų, anksčiau laikytų nesutvarkytais, terpėse aprašymui. Lyginant su ankstesnėmis metodikomis, paprastai, pagrįstomis mažų detalių nutrynimu, fraktalinė fizika atsižvelgia į terpės saviafininę struktūrą.

Fraktalinės geometrijos taikymas (IFS, ar FIF) — tai naujas požiūris į procesų analizę bei prognozes. Kaip rašo W. Paulsonas „mes galime eikvoti kompiuterio laiką, analizuodami 1 000 000 taškų, kai pakaktų 10 000, geram vaizdui gauti. Kitaip sakant, dirbtumėm 2 valandas vietoj 20 minučių“ [9]. JAV žemdirbystės departamente (US Department of Agriculture) tyrinėjami fraktalų taikymai fotografijoms (darytoms iš oro), siekiant išanalizuoti augalininkystės rūšių plėtros galimybes. M. Barnsley vienas pirmųjų pasiūlė fraktalinę geometriją taikyti iš palydovų gautų vaizdų analizei,

upių pakrančių ir jų baseinų struktūros analizei. Pastaruoju metu fraktalai vis plačiau taikomi modeliuojant socialinių arba civilizacinių procesų modelius.

XX a. 80—ųjų pradžioje analitikas Džonas Merfis sukūrė techninės kainų analizės teoriją, kurios tikslas – kainų raidos prognozavimas. Nors buvo manoma, kad kainų kaita —chaotiškas darinys, matematikas B. Mandelbrotas sukūrė universalų techninės kainų analizės metodą, kurio pagalba galima prognozuoti įvairių prekių ir prekybos kainas, kadangi ir kaitos taisyklės, ir kainų grafiko struktūra— nekinta.

Techninė analizė taip pat pagrįsta fraktalų geometrijos teiginiais. Apskritai, egzistuoja trys techninės analizės metodų grupės: grafiniai metodai, filtravimo metodai ir ciklų teorija . Tipinis techninės analizės fraktalo pavyzdys yra Elioto bangos.

B. Viljamsas knygoje „Trading Chaos“ [15], savo ruožtu, aprašė visas procedūras ir fraktalų panaudojimo būdus prekybos tyrimams.

Fraktalai, pritaikyti labai sudėtingoje antenų konstravimo srityje, pagerino antenų matricų kokybę. Be to, antenoms suteikus fraktalo formą, pagerėjo antenų parametrai. Pavyzdžiui, antenoms suteikus Kocho kreivės ar Serpinskio trikampio formą, antenų plotas sumažėjo 6 kartus; buvo sukurta elektrinė talpa bei indukcija. To pasekoje, derinant anteną ar didinant jos priimamų dažnių juostos plotį, nebereikia išorinių komponentų. Fraktalų antenos sėkmingai naudojamos Motorola mobiliuosiuose telefonuose.

Nuo 1980—ųjų svarbiausia fraktalinės geometrijos taikymo sritimi tampa biologija ir medicina. Šį procesą įtakojo tas faktas, kad fraktalinėmis savybėmis pasižymi praktiškai visos gyvo organizmo sistemos: kraujo indai ir kapiliarai, bronchai ir plaučių bronchiolės (suformuoja „medį“ su daugybe išsišakojimų), kvėpavimo takai (jų struktūra nekinta, keičiant mastelį), limfmazgiai, nervų sistema, virškinamasis traktas (plonosios žarnos stambios ir smulkios detalės panašios) ir kita. Smegenų, širdies funkcijų dinamika taip pat fraktalinė. Pastaraisiais metais daugiausia analizuoti buvo kvėpavimo takai. Čia savo indėlį įnešė įvairių šalių mokslininkai: E. Uebelas, D. Gomesas, O. Raabe, Uestas, Goldbergeris, V. Bchargava, T. Nelsonas ir kiti. Širdies arterijų ir venų fraktalinę struktūrą tyrinėja Dž. Bassingtueitas, Ch. Fon Beekas (pagrindinis tyrimų tikslas —kraujotakos anomalijų paaiškinimas). Neuronus, kaip fraktalines struktūras (dendritai išsišakoja į vis plonesnes gijas) pastaraisiais metais yra tyrinėję G. Majeris—Kresas, P. Repas, A. Bablojanis bei kiti. Svarbiausieji šios srities pasiekimai nuolat aptariami specializuotose tarptautinėse konferencijose (pirmoji įvyko 1993 m. Budapešte), kuriose pateikiama fraktalinė kardiologijos ir encefalografijos fraktalinių procesų apdorojimo metodų analizė. Taip pat yra leidžiamas specializuotas žurnalas „Fraktalai“.

Apskritai, pastaraisiais dešimtmečiais fraktalai vis dažniau taikomi chemijos, biologijos moksluose, kalbotyros moksluose bei muzikoje. Ivairių fraktalinių dydžių analizė ir fraktalinių objektų modeliavimas, seniai nebe fizikų ir programuotojų prerogatyva, randa pačias netikėčiausias taikymo

sritis. Tiriama tekstų ir hipertekstų, muzikinių kūrinių, biržos kotiruočių, smegenų procesų dinamikos, jausmų atraktorių, komunikavimo socialinėse sistemose darinių fraktalinės struktūros, globalių informacinių tinklų dinaminiai procesai“ [6].

Fraktalinės geometrijos elementai taikomi trijose pagrindinėse muzikos srityse:
kompozicijoje—Č. Dodžas, G. Nelsonas, B. Evansas bei kiti novatoriai sėkmingai naudoja grafinio vaizdo konvertavimą (fraktalų pervedimą į garsinę formą);
garso sintezėje—iš fraktalų generuojami mini—takeliai, kuriuos sintezatorius verčia į garsą;
analiziniuose tyrimuose—pavyzdžiui, R. Voss bei D. Clarke taiko fraktalinės geometrijos elementus skambesio spektrinei analizei.

Fraktalinės geometrijos kalba būtina, taipogi, tiriant spinduliavimo elementariose terpėse sklidimą ir sugėrimą, ryškios turbulencijos charakterizavimui, kietųjų kūnų paviršių savybių modeliavimui, žaibo ir elektros smūgio aprašymui, objektų senėjimo procesų analizėje, difuzijos ir agregacijos įtakos kūnams tyrimui, kvantinėje mechanikoje— banginių funkcijų geometrinių struktūrų Andersono metalo— dielektriko perėjimo taške aprašymui. Astrofizikoje— aprašant Visatos galaktikų klasterizacijos procesus; kartografijoje— tiriant pakrančių linijų formas, upių tinklą; biologijoje— kraujotakos sistemos struktūros analizei arba sudėtingų paviršių ląstelinių membranų apžvalgai. Be to, tiriamos fraktalinės tekstų ir hipertekstų struktūros, muzikos kūriniai, biržos kotiruotės, smegenų procesų požymių dinamikoje, auklėjimo atraktoriai, socialinių sistemų komunikacijos struktūros, globalių informacinių tinklų dinaminiai procesai.

Apskritai, fraktalams atsirado vieta įvairiuose moksluose. Pavyzdžiui, susiformavo fraktalinių įtrūkimų teorija, fraktalinių paviršių trinties teorija, medžio—polimerų pavidalo kompozitų fraktalinė mechanika, fraktalinė optika bei kitos sritys. Fraktalų pagalba mokslininkai aiškina galaktikų evoliuciją, kalnodaros procesus, debesų susidarymą, biržos kainų pokyčius, visuomenės ir šeimos vystymąsi.

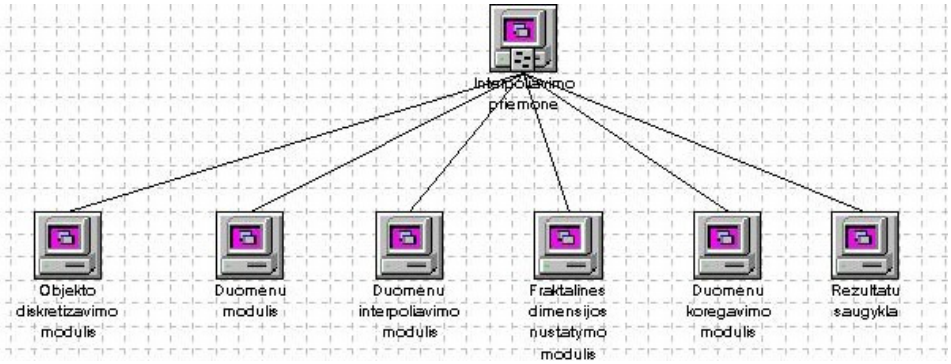
Filosofai, savo ruožtu, iškėlė teoriją apie fraktalinę sąmonę. Lygindami sudėtingų signalų encefalogramų ar širdies ritmo fraktalines dimensijas, medikai gali diagnozuoti kardiologines ligas pradinėje jų stadijoje. Meteorologai pagal radaro ekrane, esančio debesyse, kylančių srautų greičio vaizdo fraktalinę dimensiją gali jūreiviams ir lakūnams pateikti štormų prognozes, bei kita.

III. FRAKTALINIŲ INTERPOLIACINIŲ FUNKCIJŲ ANALIZĖ: TEORINIAI IR PRAKTINIAI ASPEKTAI

Plečiantis fraktalinių interpoliacinių funkcijų (FIF) praktinio panaudojimo sferai reikalinga priemonė, kurios pagalba būtų galima analizuoti (interpoliuoti) ne vien duomenis apie geografinius landšafto objektus, bet ir ekonominius, statistinius, medicininius ir kt. pobūdžio duomenis. Atskirų tipų duomenų analizei priemonių yra sukurta nemažai (Fractan, Market Way Pro, Investor's Dream, SKO bei kt.), bet vienos universalios priemonės, kurią būtų galima taikyti įvairiarūšei informacijai analizuoti, naudoti neteko. Tam tikru universalumu pasižymi fraktalinio interpoliavimo įrankis INTERPO [8].

Šiame darbe analizuojami priemonės, skirtos FIF generavimui, taikymo metodai bei apžvelgiama keleto konkrečių nevienarūšių objektų FIF kreivių generavimo eksperimentų rezultatai. Eksperimento metu pastebėtos darbo su esamu Interpo varijantu problemos privertė pabandyti priemonę adaptuoti . Interpoliavimo priemonę galėtų sudaryti sekantys moduliai:

* Objekto diskretizavimas;
* Duomenų koregavimas arba atranka;
* Duomenų rinkinys;
* Objekto fraktalinės dimensijos apskaičiavimas;
* Duomenų interpoliavimas;
* Rezultatų saugykla.

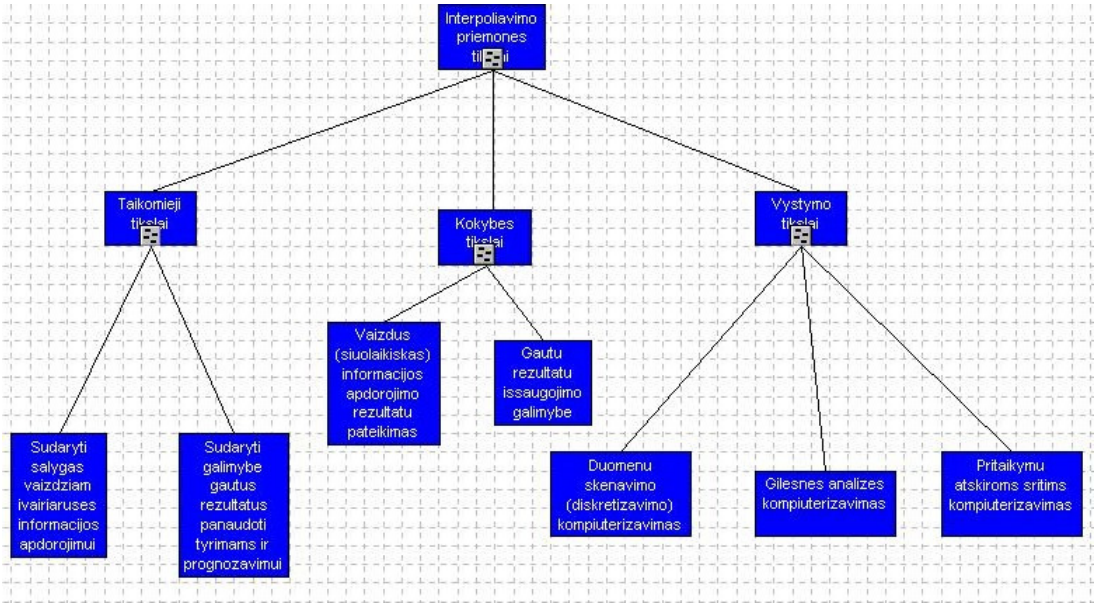


3.1 pav. Interpoliavimo priemonės struktūra (modulių diagrama)

Šis modulių rinkinys leistų pilnai realizuoti interpoliavimo priemonei keliamus reikalavimus (tikslus):

* Galimybė apdoroti įvairiarūšę informaciją (landšafto objektus, statistinius, finansinius, biržų prekybos, medicininius duomenis);
* Galimybė duomenis taikyti gilesnei analizei (ekstrapoliavimui, prognozėms ir pan.);
* Galimybė vizualiai pateikti duomenų apdorojimo (interpoliavimo) rezultatus;

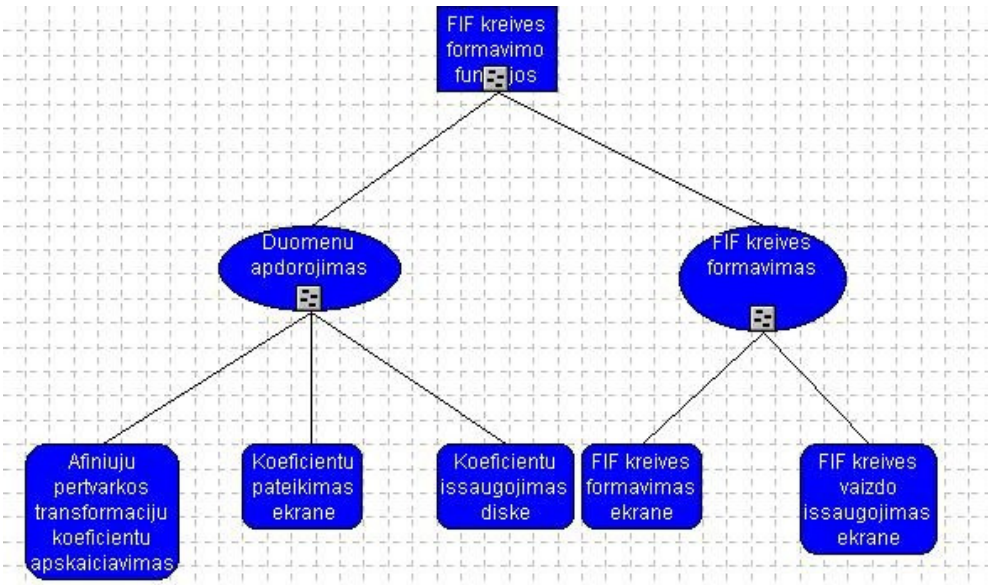
* Galimybė kompiuterizuoti duomenų koregavimo procesą;
* Galimybė išsaugoti gautus rezultatus (tiek skaičiavimų, tiek ir grafinius) informacinėje laikmenoje.



3.2 pav. Interpoliavimo priemonės tikslų modelis

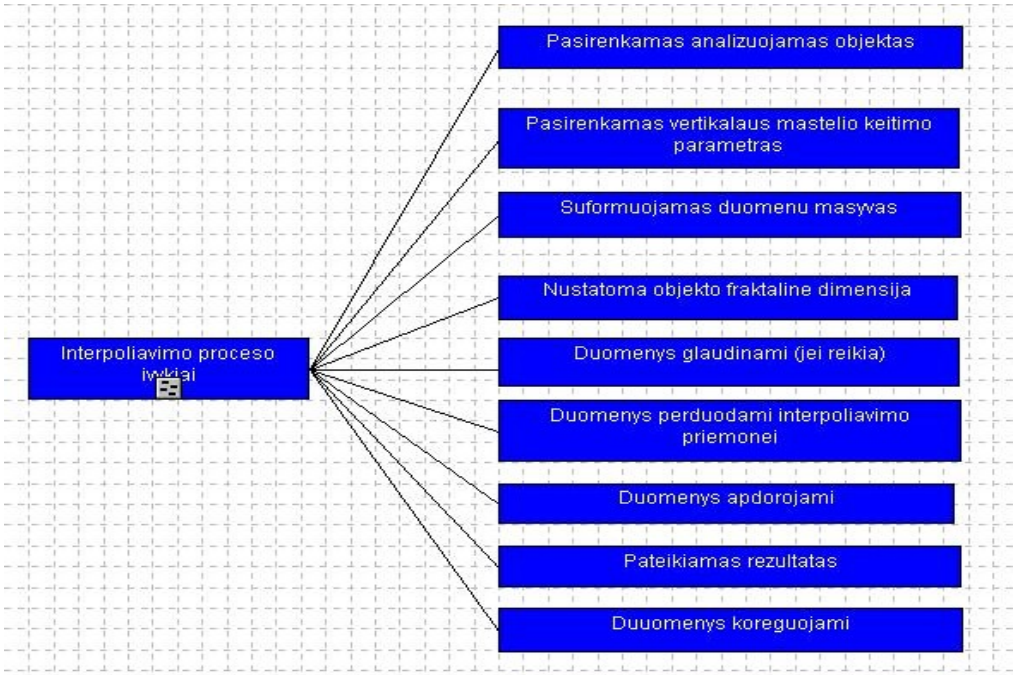
Šiuos tikslus būtų galima pasiekti, taikant priemonę, atliekančią tokias funkcijas:

* Duomenų apdorojimo:
    * Afiniųjų transformacijų koeficientų (parametrų) apskaičiavimo;
    * Koeficientų reikšmių pateikimo ekrane;
    * Koeficientų reikšmių saugojimo informacinėje laikmenoje;
* FIF kreivės formavimo:
    * FIF kreivės formavimo kompiuterio ekrane;
    * FIF kreivės vaizdo išsaugojimo informacinėje laikmenoje.



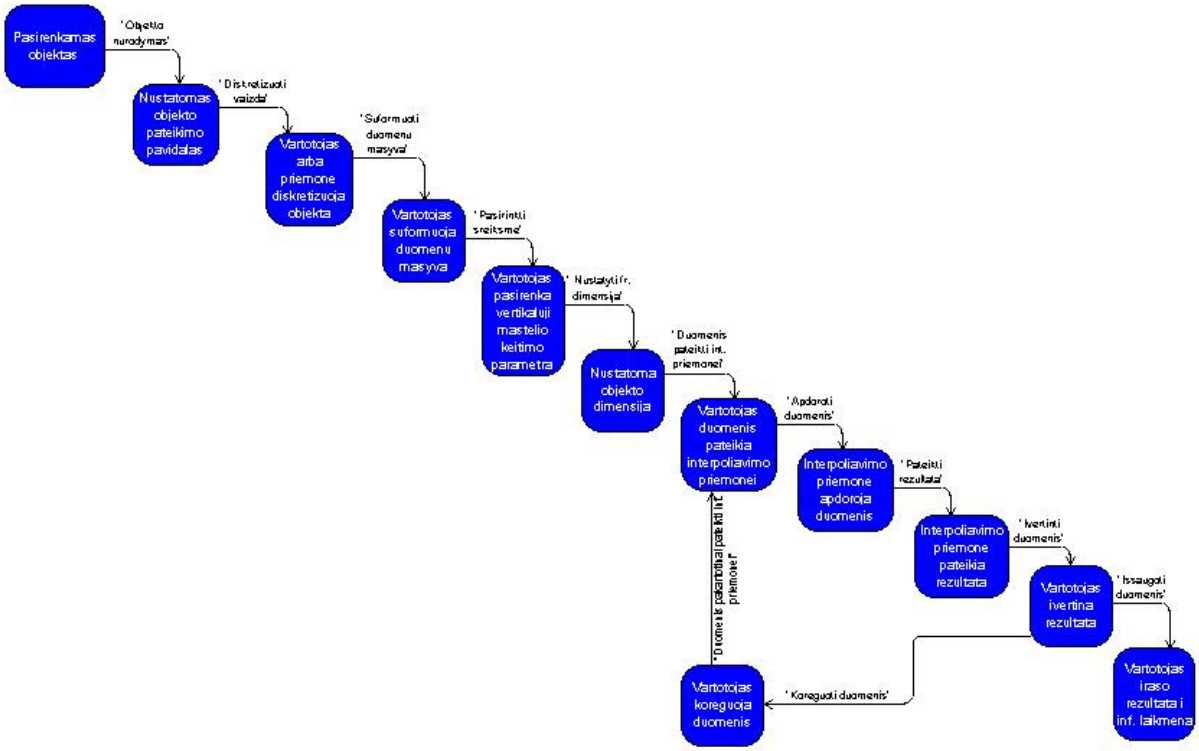
3.3 pav. Interpoliavimo priemonės funkcijų modelis

Procesus, vykstančius informacijos apdorojimo metu nusako priemonės įvykių modelis:



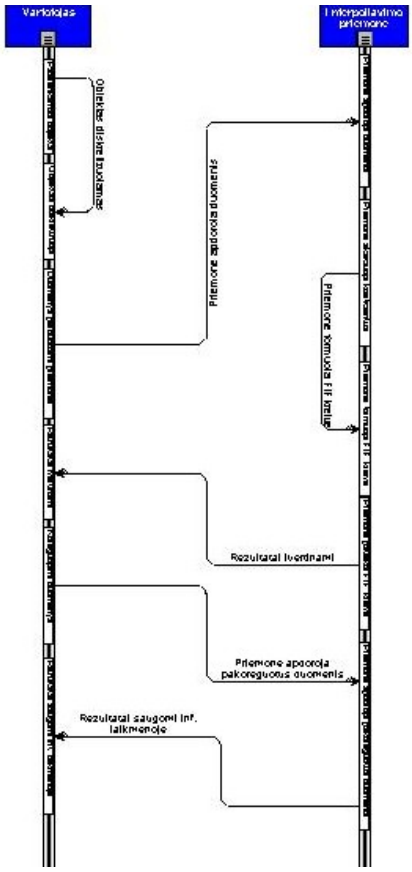
3.4 pav. Interpoliavimo priemonės įvykių modelis

Būsenų modelis nusako priemonės taikymo darbų (priemonės būsenų) spektrą:



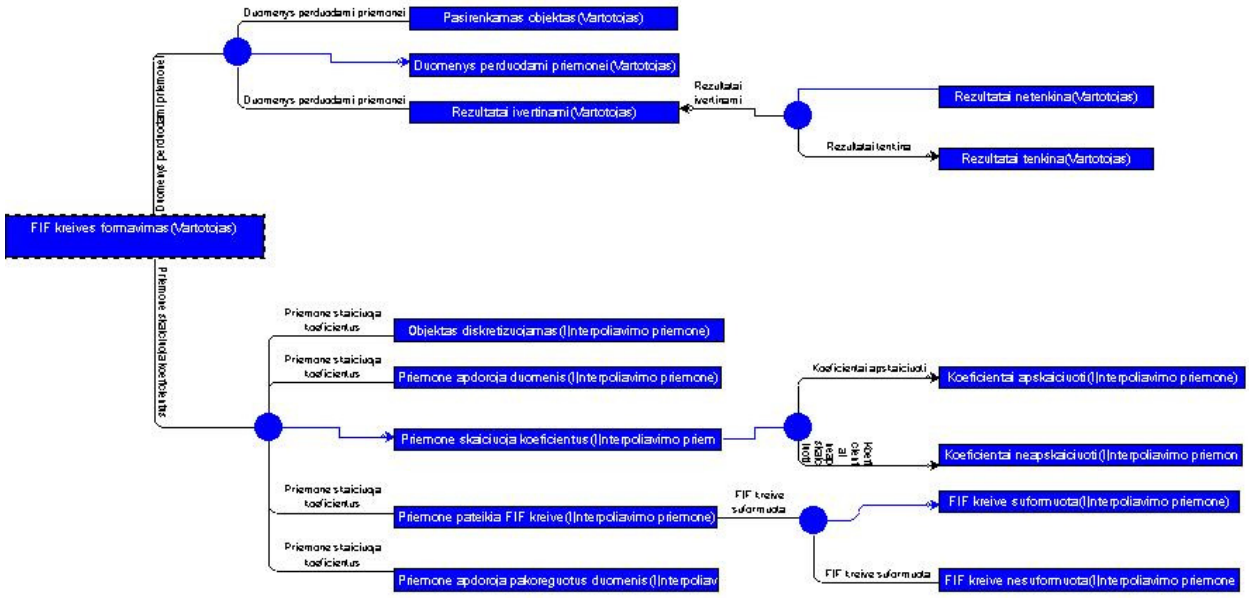
3.5 pav. Interpoliavimo priemonės būsenų modelis

Sąveikų modelis nusako veiksmų pasiskirstymo tarp priemonės ir vartotojo seką:



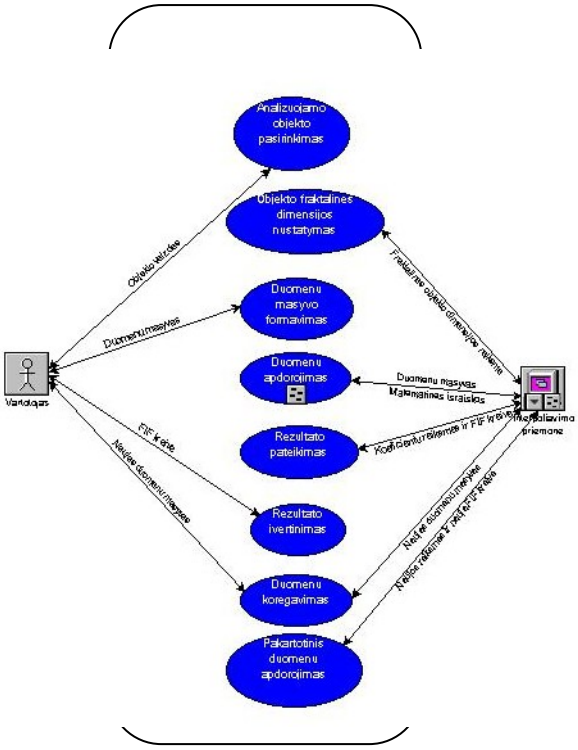
3.6 pav. Duomenų analizės proceso sąveikų modelis

Metodų modelis nusako FIF kreivės formavimo procese taikomų metodų rinkinį:



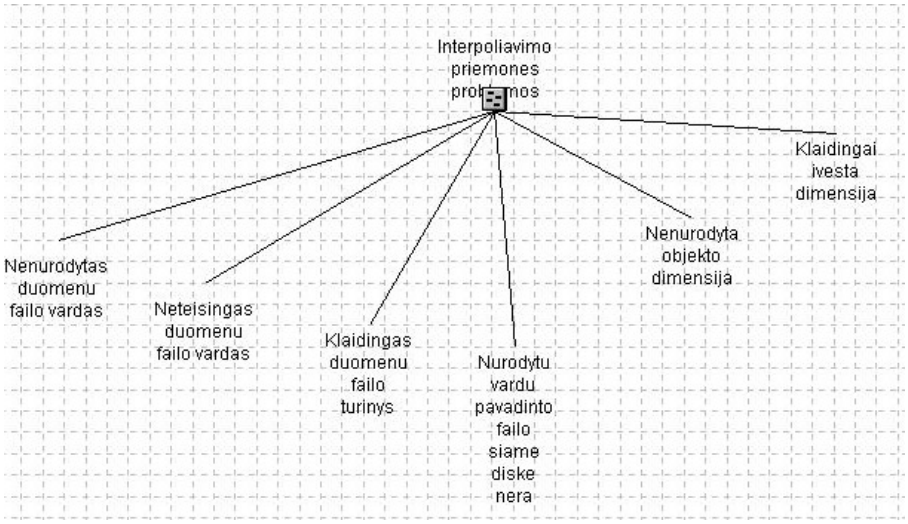
3.7 pav. FIF kreivės formavimo proceso metodų modelis

Panaudojimo atvejų modelis nusako veikėjų (vartotojo ir priemonės) veiksmų eigą ir informacijos cirkuliaciją tarp jų:



3.8 pav. Interpoliavimo priemonės panaudojimo atvejų modelis

Klaidų modelis nurodo pagrindines problemas, su kuriomis gali susidurti vartotojas:



3.9 pav. Interpoliavimo priemonės problemų modelis

Pateiktą interpoliavimo priemonės schemą dalinai tenkina jau sukurta priemonė INTERPO, kurios trumpą aprašą pateikiame žemiau.

3.1. INTERPO—fraktalinio interpoliavimo įrankis

Fraktalinės interpoliacinės funkcijos (FIF)—tai naujas eksperimentinių duomenų perteikimo įrankis. Fraktalinės interpoliacinės funkcijos gali būti naudojamos tokių eksperimentinių duomenų, kaip laiko funkcijos (elektroencefalogramos ir pan.), modeliavimui. FIF grafo fraktalinė dimensija „atitinka“ duomenis tam tikrame intervale [1].

FIF generavimas rankiniu būdu yra sudėtingas, o remiantis tiesinės interpoliacijos principais, FIF paiešką galima kompiuterizuoti. Viena iš tokių priemonių ir yra D. Oliverio sukurta priemonė Fractal Vision [8].

Produkto pavadinimas: **Fractal Vision**

Autorius: **Dick Oliver**:

Priemonės sudėtis:

Fractal Vision – fraktalų generavimui skirta priemonė **(fv.exe);**

Interpo – fraktaliniam interpoliavimui skirta priemonė **(interpo.exe);**

Life – žaidimas **(life.exe).**

Produkto anotacija:

Produktas sukurtas 1990 m. balandžio mėn.; pradinė jo paskirtis – fraktalų generavimas.

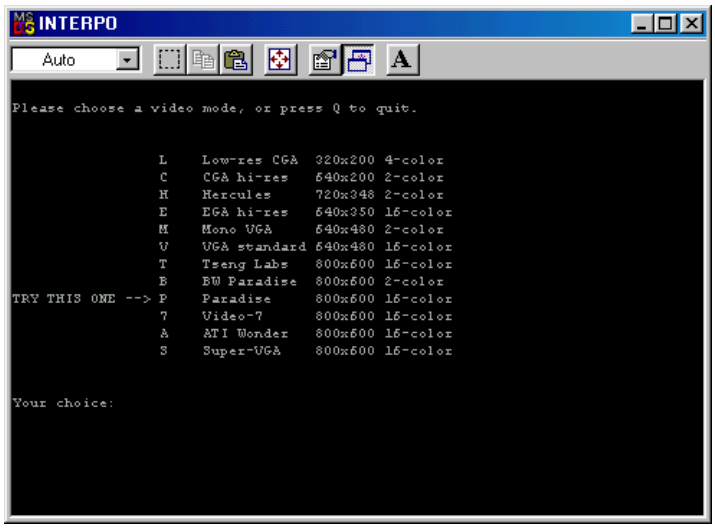
INTERPO – *Fractal Vision* sudėtinė dalis, skirta fraktalų interpoliacijos tyrimams. Priemonei pritaikyti reikalingas pradinis *fip* tipo failas, kuriame pateikiami duomenys apie interpoliavimo taškus. Vykdant programą, pirmiausiai nurodomas vaizduoklio tipas, po to – nurodomas duomenų failas bei dimensija. Priemonė ekrane pateikia:

- Interpoliuojamą duomenų aibę bei kitus parametrus;
- interpoliavimo taškais apribotoje srityje gautą fraktalinę kreivę.

„Interpo skaito tekstinį (ASCII) failą, kuriame saugomos *x*, *y* reikšmės, spausdina gautus rezultatus ir pateikia ekrane grafinį vaizdą. Taip gaunamos fraktalinės interpoliacinės funkcijos, einančios per duomenų faile nurodytus taškus, vaizdas. Kreivės išlinkimai priklauso nuo skaitinės vertikalųjį mastelį keičiančio parametro reikšmės“ [8].

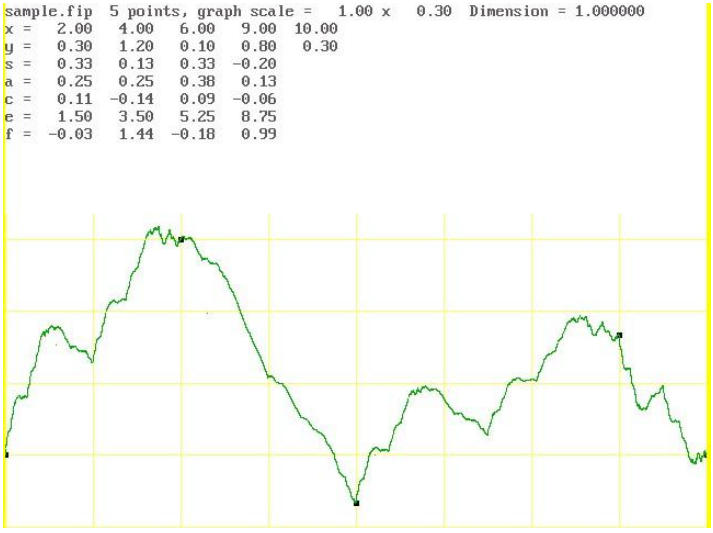
Kadangi pradinė Interpo.c versija sukurta rinkiniui, sudarytam iš mažiau nei 100 elementų, tai norėdami ją taikyti didesniam duomenų kiekiui turime keisti pradinę programą. Tad duomenų rinkinį turėtų nusakyti taškai, geriausiai atspindintys kreivės svyravimus.

INTERPO pertvarko bet kokį duomenų, esančių tarp dviejų nuoseklių intervalo galų, rinkinį. Pavyzdžiui, jei užduosime 100 taškų, tai kreivę gausime po 99 pertvarkių. FIF atitikimas realiai situacijai bus tuo tikslesnis, kuo tinkamiau bus parinkti interpoliavimo taškai. Kaip juos parinkti? „Idealiausia, kad tai būtų kraštiniai dalių, panašių į visumą taškai. Jei vizualiai nustatyti panašumą sunku, galima parinkti kas 10—tą, kas 50—tą ar 500—tąjį tašką, priklausomai nuo to, kiek duomenų turime apskritai.“ [8]

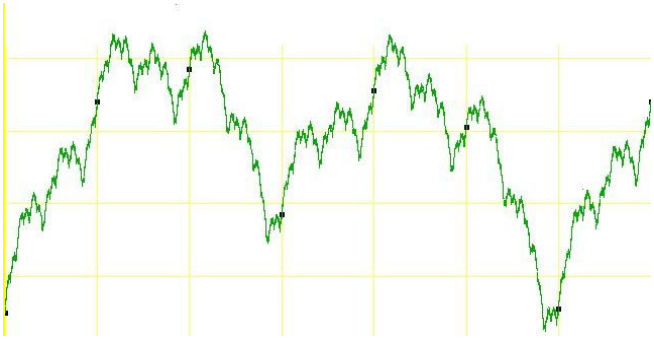


3.10 pav. Pradinis INTERPO ekranas – jame siūloma pasirinkti monitoriaus tipą

Toliau nurodome reikiamos bylos pavadinimą ir pasirenkame interpoliuojamo objekto fraktalinės dimensijos *D* reikšmę (nuo 1 iki 2).

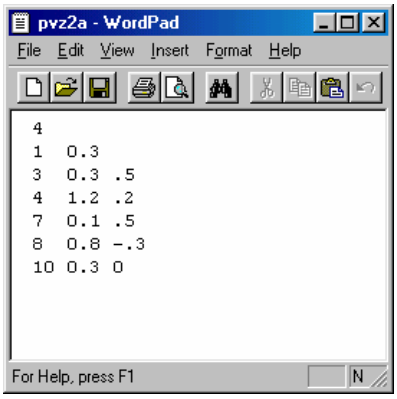


3.11 pav. Duomenų aibės (Sample.fip), kai *D*=1, interpoliavimo rezultatas



3.12 pav. Duomenų aibės (Avg100.fip), kai *D*=1,3, interpoliavimo rezultatas

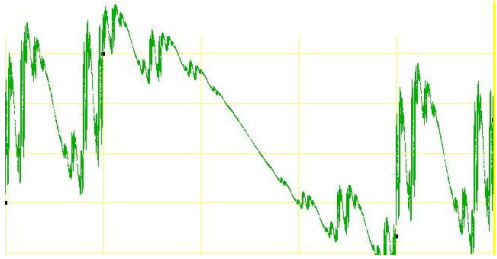
Norėdami sudaryti savo fraktalą, turime suformuoti naują **fip** tipo bylą. Tą padaryti galime su paprasčiausia teksto redagavimo priemone (Notepad, ar WordPad):



3.13 pav. Duomenų aibės pateikimas

Šiame dokumente nurodome interpoliavimo taškų kiekį, skalę (tinklelį), kintamųjų *x* ir *y* bei vertikalųjį mastelį keičiančio parametro *s* reikšmes. Pagal pateiktus duomenis, papildomai nurodę dimensiją, gausime atitinkamą fraktalinę kreivę (FIF grafiką):

3.14 pav. FIF grafikas; Pvz2a.fip, kai *D*=1



3.15 pav. FIF grafikas; Pvz2a.fip, kai *D*=1.61

Parametras *s* apsprendžia kreivės „grubumą“. Pavyzdžiui, jei pirmojo interpoliavimo intervalo parametras s= 0,5, o visų kitų—0,2, tai kraštinė kairioji kreivės dalis bus aukštesnė ir „grubesnė“ nei likusioji dalis. Neigiama parametro s reikšmė nusako atvirkščią kreivės linkį horizontalios ašies atžvilgiu. „Jei parametro s reikšmės nežinome, pradiniam eksperimentui renkamės 0,5, o vėliau, įvertinę gautą rezultatą vizualiai, reikšmes tiksliname. Apskritai s reikšmės priklauso intervalui (-1; 1)“ [8].

Tinklelį parenkame priklausomai nuo aprašomų duomenų kaitos intervalų. Pavyzdžiui, jei *x* ir *y* reikšmės yra nutolusios (kinta) per vieną vienetą, tinklelis 1x1, jei *x* yra tarp 300 ir 500, o *y* tarp 0,01 ir 0,02, tai tinklelis gali būti 50x0,001. Šių duomenų reikia FIF tinkleliui nurodyti.

Taigi, INTERPO duomenų failas turėtų atrodyti taip:

- bendras interpoliavimo taškų skaičius
- tinklelis
- *x*, *y*, vertikalųjį mastelį keičiantis parametras *s*.

Paskutinėje duomenų eilutėje esantis parametras nebereikalingas, nes už paskutinio interpoliavimo taško kreivės nėra, bet kadangi programoje nurodyti trys skaitomi duomenys, tai ir paskutinėje duomenų failo eilutėje šis parametras įrašomas.

Vykdant INTERPO, pasirinkta fraktalinės dimensijos reikšmė apsprendžia FIF kreivės vertikalų „grubumą“. Fraktalinei dimensijai nustatyti galime panaudoti papildomą priemonę –*Fracdim*. Ši priemonė leidžia stebėti FIF kreivės priklausomybę nuo dimensijos *D* kaitos. *D* reikšmes galime užduoti savarankiškai (pasirinkdami iš tam tikro intervalo, arba taikydami atsitiktine tvarka parinktas vertes). *Fracdim* ypač naudinga tais atvejais, kai nustatyti dimensiją programiškai neįmanoma (pavyzdžiui, analizuojamas skenuotas vaizdas), o matematiniai skaičiavimai —sudėtingi ir reikalaujantys daug laiko.

3.2. Interpoliavimo taškų parinkimo kriterijai ir jų nustatymas

Interpoliavimo taškų parinkimui šiame darbe siūlomi šie metodai:

- Interaktyviam naudojimui orientuotas tolygaus interpoliavimo taškų parinkimo metodas;
- Koliažas.

3.2.1. Interaktyviam naudojimui orientuoto tolygaus interpoliavimo taškų išdėstymas

Objekto fraktalinės analizės procesą galima suskaidyti į keletą etapų:

1) Nustatoma objekto fraktalinė dimensija ;
2) Suformuojamas objektą nusakantis duomenų masyvas (duomenų aibė). Jei duomenų aibę sudaro daugiau nei 100 elementų—ją reikia sumažinti, taikant duomenų atrankos metodus (INTERPO leidžia apdoroti tik iki 100 elementų);
3) Nustatomos vertikalųjį mastelį keičiančių parametrų s reikšmės;
4) Stebima fraktalinės kreivės priklausomybė nuo fraktalinės dimensijos (priemonė Fracdim);
5) Formuojama fraktalinė interpoliacinė kreivė (su priemone *INTERPO*).

Šiuos etapus apžvelgsime detaliau.

1. Grafinio *.pcx tipo vaizdo fraktalinės dimensijos nustatymui galime pasinaudoti Fractal Vision arba rinktis pradinę atsitiktinę reikšmę, kurią vėliau bus galima koreguoti (plačiau ši dalis bus aptarta 3.2.3. skyrelyje).

2. Objekto fraktalinis interpoliavimas priklauso nuo objektą nusakančių duomenų parinkimo metodų.

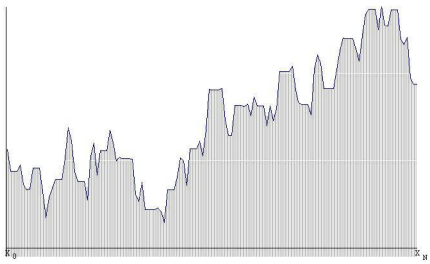
Grafinio objekto vaizdą geriausiai nusako koordinatinis duomenų masyvas (x_i, y_i), i=0, 1, ..., N. Šį masyvą galima suformuoti tiek naudojant *vizualinį metodą* (analizuojant geografinius atlasus ar žemėlapius, kai rezultatas priklauso nuo vietovės mastelio), tiek ir naudojant *programines priemones*. Pastarasis būdas leidžia objektą nusakyti tikslesniu duomenų rinkiniu, nei elementarus vizualinis būdas. Tačiau ir čia daug kas priklauso nuo grafinio objekto vaizdo pateikimo bei analizės priemonės. Galime naudoti skenuotą vaizdą, jį analizuodami kokio nors grafinio redaktoriaus pagalba. Tada rezultatas priklausys nuo pradinio vaizdo mastelio, skenerio dpi, ekrano raiškos bei priemonės — grafinio redaktoriaus —tinklelio tikslumo. Galutinį duomenų masyvą (duomenų aibę) galime suformuoti, pasirinkdami, tarkime, patį mažiausią (1x1) tinklelį ir įtraukdami jo vertikalės ir pradinį objektą nusakančios kreivės susikirtimo taškus.

Dar optimalesnį rezultatą gautume, panaudoję skaitmeninio vaizdo vektorizavimą specializuotomis priemonėmis (eksperimentiniai duomenys (3.4 skyrelis) buvo gauti su *GIS (Arcview and Arcgis)*(*.shp formato duomenys; pilnai upė vektorizuota 1:200 000 masteliu; vektorizacija atlikta LKS 94 koordinačių sistemoje). Duomenų tikslumas, paprastai, priklauso nuo pradinio vaizdo mastelio.

Tačiau, šiuo būdu suformuotą duomenų aibę, gali būti sudėtinga panaudoti fraktaliniam interpoliavimui (eksperimento metu, pavyzdžiui, buvo gautas iš 1459 elementų sudarytas duomenų masyvas) su INTERPO, nes gautos aibės elementų skaičius didesnis nei 100 elementų.

Šią problemą išspręsime, taikydami *duomenų atrankos* metodus.

Tarkime, turime fraktalinę kreivę, nusakytą duomenų rinkiniu (x_i, y_i) (i=0, 1, … N), atvaizduojančią fraktalinę funkciją f: $y_i=f(x_i)$, kai $i\in$ [0; N] ir f: $\mathbb{R}^2 \to \mathbb{R}^2$. Reikia parinkti rinkinį $f(x_j')$, kad $0<j<N$, $f(x_j') = f(x_i)$ ir f: $\mathbb{R}^2 \to \mathbb{R}^2$.



3.16 pav. Diskretizuotos kreivės pvz.

Kadangi duomenų masyve gali būti ne daugiau kaip 100 elementų, tai pats paprasčiausias (elementariausias) jų atrankos metodas būtų *atsitiktinės duomenų atrankos* metodas, kai iš turimo duomenų rinkinio atsitiktine tvarka būtų išrenkama 100 elementų (pavyzdžiui, 99 pirmieji elementai ir paskutinysis ar pirmasis ir 99 paskutinieji elementai).

Tačiau tikimybė, kad optimaliausią, t.y. geriausiai atspindintį visumą, duomenų rinkinį pavyks gauti iš karto, labai maža, jei vizualiai negalime nustatyti interpoliavimo taškų. Tokiu atveju teks atranką pakartoti. Tad nuoseklus duomenų aibės formavimo eksperimentas gali tęstis gan ilgai (tokių derinių C_N^{100}).

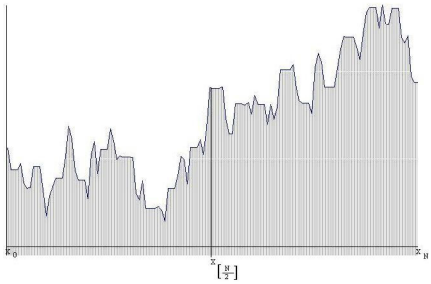
Kitas, sąlyginai, atsitiktinės atrankos metodas—*atranka pagal pasirinktą požymį* (statistinių duomenų, biržos duomenų atranka pagal savaitės ar mėnesio dieną ir pan.)

Tačiau efektyviausia bus duomenų atranka, pritaikant kokį nors *matematinį duomenų atrankos* metodą.

Paprasčiausias iš jų —*aritmetinės progresijos principu paremtas* metodas, kai $x_j=x_{j-1}+h$, j=1, 2, ..., N. Tokiu būdu, priklausomai nuo pasirinkto žingsnio h, gausime duomenų rinkinį, kuriame bus $\left[\frac{N}{h}\right]$ elementų. Taigi, galutinis rezultatas priklauso nuo pasirinkto žingsnio dydžio. Interpoliuojant su INTERPO, žingsniui h galioja vienintelė sąlyga—gautų elementų (taškų) skaičius negali būti didesnis nei 100. Dažniausiai tokį duomenų rinkinį $\{x_0, x_{0+h}, \dots x_n \mid n\leq 100<N\}$ sudaro kas 5—tas, 10—tas, 50—tas, 100—tasis ir pan. elementai. Statistiniai, biržos duomenys gali būti atrenkami ir imant kas 7—tą (savaitė), 30—tą (mėnuo) ar 365—tą (metai) elementą.

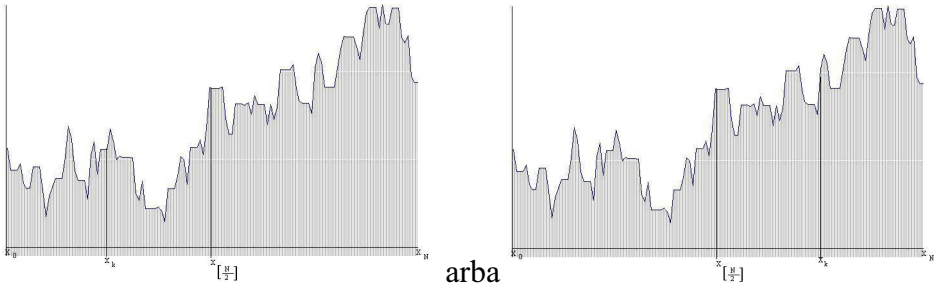
Pagrindinė problema—kaip nustatyti žingsnį h? Atsitiktinai parinktas h nevisada gali sudaryti sąlygas (net vizualiai vertinant) interpoliavimo metu gauti siektiną rezultatą (kaip ir, apskritai, atsitiktinės atrankos metodu gautas duomenų rinkinys).

Dar vienas iš elementarių matematinių metodų yra *intervalo dalijimo pusiau metodas*. Intervalas [0; N] dalijamas pusiau, ko pasekoje gaunami trys taškai, kuriems jau galima taikyti interpoliavimą: $\{x_0, x_{\left[\frac{N}{2}\right]}, x_N\}$.



3.17 pav. Pirminis intervalo dalijimas pusiau

Gautas interpoliavimo rezultatas greičiausiai (net vizualiai vertinant) nebus siektinas rezultatas. Todėl algoritmą reikėtų taikyti toliau, pasirinkus vieną iš gautų intervalų:



a) b)

3.18 pav. Naujai dalijamos srities pasirinkimas

Dalijimo pusiau procedūra tęsiama tol, kol bus gautas pakankamai geras vaizdas. Jį įvertinti dažniausiai galime vizualiai. Statistinio rezultato tikslumą (patikimumą) galime įvertinti ir skaičiuodami statistinį patikimumo įvertį μ ($\mu = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_{ti}}{y_i}\right| \cdot 100\%$). Jei šis įvertis neviršyja 10%, teigiama, kad gauti duomenys yra pakankamai patikimi.

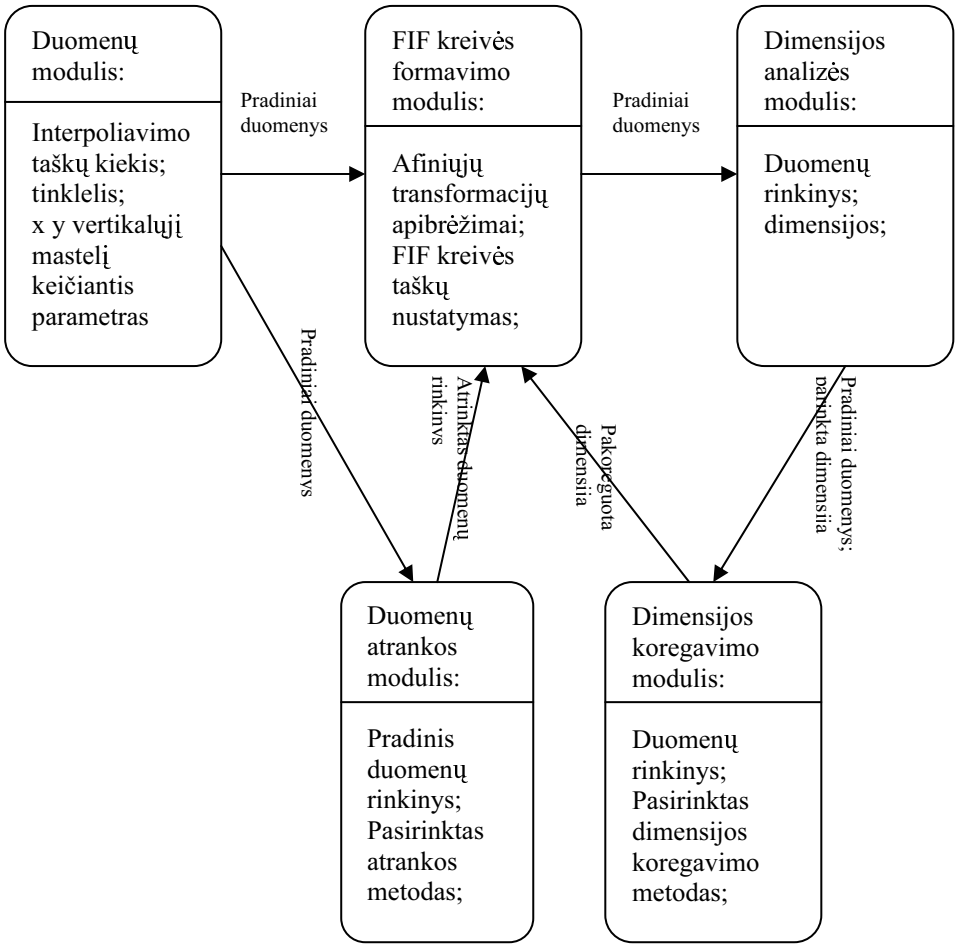
Dar optimalesnius rezultatus gausime, duomenų atrankai pritaikę sudėtingesnius matematinius metodus—*polinominės atrankos, splainų, bangų, greitą Furje atranką, pabėgimo laiko algoritmą* (aprašytas 2.1.2 skyrelyje) bei kitus metodus.

3. Vertikalųjį mastelį keičiančių parametrų reikšmių nustatymas. Pradiniame etape, paprastai, pasirenkama ta pati reikšmė visiems parametrams, pavyzdžiui— s_i =0,5 (i=1, 2, ..., N). Vėliau, dažniausiai vertinant vizualiai, s_i reikšmės koreguojamos, turint omeny tai, kad $s_i \in$ (-1; 1) ir neigiamos s_i reikšmės panaudojimo rezultatas— atitinkamos kreivės dalies atspindys x ašies atžvilgiu. Be to, didinant parametrą, kreivės lankas („danteliai“) ilgės, mažinant— trumpės.

Eksperimento metu (3.4 skyrelis) pastebėta, kad šis parametras privalomai turi būti nurodytas tik [N/2] eilutėje (jo nenurodžius duomenų failas nenuskaitomas ir programa INTERPO nevykdoma).

Šiuo etapu praktiškai baigiamas duomenų paruošimas ir jie perduodami INTERPO. INTERPO, kaip jau buvo minėta, pirmiausiai nurodomas vaizduoklio tipas bei duomenų failo pavadinimas (būtinai turi būti nurodytas ir tipas *.fip), įvedama dimensija ir gaunamas rezultatas. Toliau, duomenys gali būti koreguojami ir eksperimentas kartojamas iš naujo tol, kol bus gautas optimaliausias rezultatas —FIF kreivė —vizualiai panaši į analizuojamą objektą.

Siūlome tokį interaktyviam naudojimui orientuotos duomenų analizės (FIF pagalba) modelį:



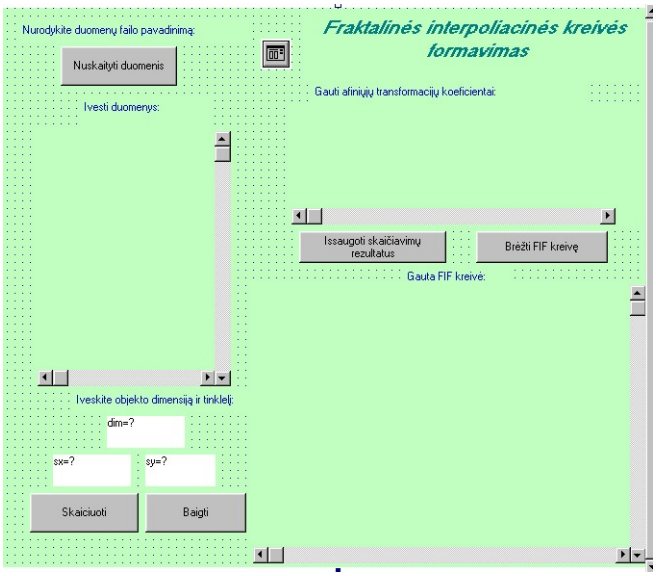
3.19 pav. Modulinė priemonės struktūra

Šiame darbe analizuojamas eksperimentas atliktas, kai duomenų modulis— *fip* tipo failas; FIF kreivės formavimo modulis—priemonė INTERPO (priedas1—šiek-tiek patobulintas algoritmas C

kalba); dimensijos analizės modulis—priemonė *fracdim* (priedas2—fracdim.bat pavyzdys); duomenų atrankos modulis—priemonė *FFT* (Greita Furje atranka)(priedas3—algoritmas C kalba); dimensijos koregavimo modulis kol kas nerealizuotas programiškai.

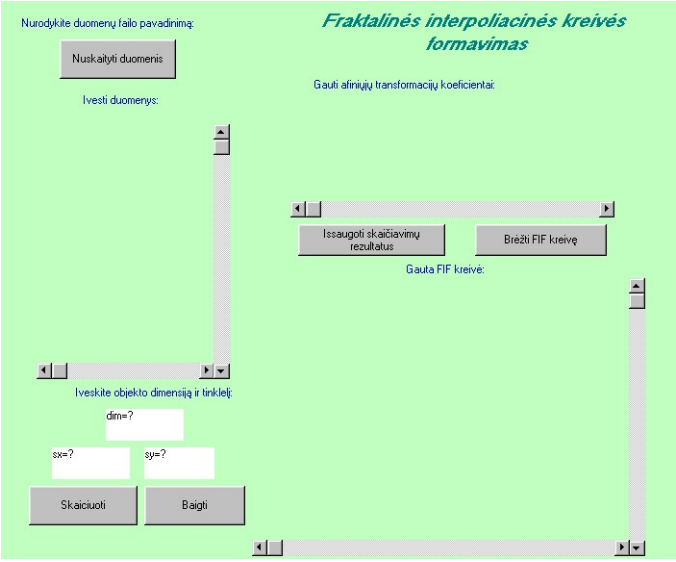
Pastebėtos INTERPO taikymo problemos buvo pabandytos spręsti, sukuriant naują priemonę (Visual Basic pagalba), paremtą tais pačiais, kaip ir INTERPO, duomenų analizės metodais.

Pagrindinis priemonės langas suskaidytas į kelias sritis: duomenų įvedimo, skaičiavimų rezultatų spausdinimo, FIF kreivės braižymo sritis. Priemonės darbas (perėjimas prie naujo etapo) vykdomas mygtukų („Nuskaityti duomenis“, „Skaiciuoti“, „Issaugoti koeficientu reiksmes“ bei „Brezti FIF kreive“) pagalba.



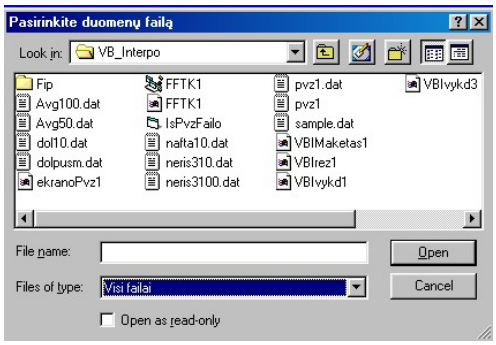
3.20 pav. VB Interpo priemonės pagrindinio ekrano maketas

Darbui su priemone šiame etape reikalingas duomenų rinkinys (failas, išsaugotas kokioje nors informacinėje laikmenoje). Šiame faile turėtų būti surašyti duomenys, nusakantys objektą. Konkrečiai, koordinatinės interpoliavimo taškų išraiškos bei vertikalaus mastelio keitimo parametro reikšmė (reikšmės). Lyginant duomenų failą su INTERPO, praktiškai randame tik du pakeitimus: duomenų faile nebereikia nurodyti taškų kiekio bei koordinatinio tinklelio reikšmių. Vertikalaus mastelio keitimo parametro (parametrų) reikšmės gali būti parenkamos analogiškai INTERPO duomenų rinkiniui. Duomenų failas gali būti suformuojamas bet kokiu teksto redaktoriumi. Jam siūloma suteikti tipą *.dat.



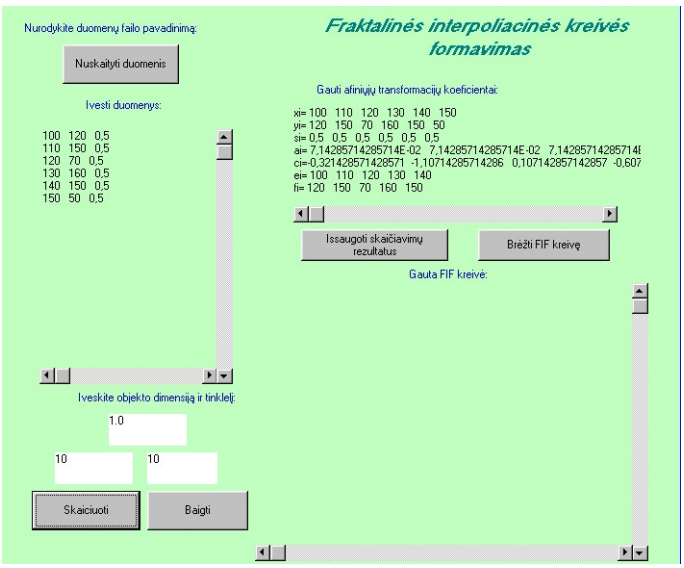
3.21 pav. VB Interpo pagrindinis ekranas (darbinė sritis)

Įvykdžius programą, pirmiausiai reikia pasirinkti duomenų rinkinį, saugomą informacinėje laikmenoje. Tam skirtas failo paieškos diske įrankis, aktyvuojamas mygtuko „Nuskaityti duomenis“ pagalba. Tai naujas įrankis, kurio senojoje INTERPO versijoje nebuvo. Pasirinkimas vykdomas įprasta failo paieškos tvarka, turint omenyje, kad rodomi bus visi (arba tik duomenų (jų tipas—dat)) failai. Ieškoti, kaip įprasta, galima bet kokioje informacinėje laikmenoje ir bet kuriame jos kataloge:



3.22 pav. Duomenų nuskaitymas VB Interpo

Nurodžius duomenų failą, duomenys pateikiami ekrane (tam skirta speciali sritis „Ivesti duomenys“); įvedama klaviatūros pagalba fraktalinė objekto dimensija ir nurodomas tinklelis (Interpo jis buvo nurodomas duomenų faile). Nuspaudus mygtuką „Skaiciuoti“ atitinkamoje ekrano srityje pateikiamos afiniųjų transformacijų koeficientų reikšmės bei kiti rezultatų masyvai:



3.23 pav. Pradinis skaičiavimų ir duomenų spausdinimo etapas

Toliau galima rinktis vieną iš dviejų: arba išsaugoti gautus skaičiavimo rezultatus (to INTERPO nebuvo numatyta) arba formuoti FIF kreivę. Suformavus FIF kreivę, toliau reikėtų vėl rinktis—ar koreguoti duomenis (tam turėtų būti skirtas papildomas priemonės modulis (įrankis „Koreguoti duomenis“), koreguojantis (atrenkantis) duomenis pagal tam tikrą algoritmą (aptarta anksčiau)) ar išsaugoti gautą kreivę informacinėje laikmenoje. Šiems elementams nusakyti šiame darbe pateiktas algoritmas, užrašytas Visual Basic (priedas 4).

Pradiniame darbo etape panaudojus dar vieną (objekto diskretizavimo) modulį (priemonės įrankį), duomenys galėtų būti įvedami, ne rankiniu būdu renkantis failą, o tiesiogiai, t.y. tą padarytų priemonė (ši problema dalinai galėjo būti sprendžiama kartu su INTERPO, panaudojus Fractal Vision priemonę). Analogiškai, dimensija taip pat galėtų būti netik įvedama rankiniu būdu, bet ir apskaičiuojama programiškai (priemonės įrankis, skirtas fraktalinės dimensijos skaičiavimui). Pastarieji elementai kol kas nerealizuoti ir šioje naujai formuojamoje priemonėje. Tai galėtų būti šio darbo tęsinys (čia pateikta metodika leistų operatyviau suformuoti papildomus modulius). Be to, priemonė galėtų būti papildyta ir naujais interpoliavimo bei fraktalinės dimensijos įvertinimo įrankiais, kurių veikimas būtų pagrįstas 3.2.2. bei 3.3. skyreliuose pateikta metodika.

3.2.2. Koliažo organizavimas

Tarkime, kad duomenų aibė

$$S=\{(x_j, F_j)|\, j=0, 1, \ldots, M\}$$

gauta diskretizuojant (skenuojant) realaus pasaulio objektą arba užrašant jau atlikto eksperimento rezultatus. Tarkime, taipogi, jog taškai x_j (j=0, 1, …, M) yra išsidėstę tolygiai, t.y. vienodu atstumu vienas nuo kito; be to, skaičius M yra pakankamai didelis.

Dėl įvairių priežasčių (3.1 skyrelis), interpoliavimo procedūra dažnai organizuojama ne visai duomenų aibei S, o tik tam tikram jos poaibiui S' (pavyzdžiui, imant kas antrą, kas trečią (ar panašiai) aibės S elementą), t.y.

$$S'=\{(x_j, F_j)|\, j=\lambda i,\ i=0, 1, ..., N\};$$

čia $\lambda\in\{2, 3, \ldots\}$.

Įveskime pažymėjimus:

$$x_j = x_{\lambda i} = x_i^{'}\ ,\ F_j = F_{\lambda i} = F_i^{'}\ ,$$

su visais i =0, 1, …, N. Tada, interpoliavimui imama duomenų aibė bus tokia

$$S'=\{(x_i^{'},\ F_i^{'})|\ i=0, 1, ..., N\}.$$

Fraktalinės interpoliacinės funkcijos (IFS atraktoriaus) generavimui pasinaudosime (lokaliąja) Koliažo teorema [14]. Ši teorema siejama su atveju, kai nagrinėjamam realaus pasaulio objektui (diskretizuotam jo analogui—duomenų aibei S) koliažas konstruojamas ne iš paties objekto sumažintų kopijų (2.1.2. skyrelis), bet iš to objekto fragmentų sumažintų kopijų.

Šiame kontekste (nagrinėjamas objektas—duomenų aibė S) objekto fragmentus sudarys intervalai $[x_{j-1}^{'},\ x_{j+1}^{'}]$, $j\in\{1, 2, \ldots, N-1\}$, o jų sumažintų kopijų analogų bus ieškoma intervalų $[x_{i-1}^{'}, x_i^{'}]$, i=1, 2, ..., N, aibėje. Intervalus $[x_{i-1}^{'}, x_i^{'}]$ (i=1, 2, ..., N) vadinsime reikšmių srities intervalais, o intervalus $[x_{j-1}^{'},\ x_{j+1}^{'}]$ ($j\in\{1, 2, \ldots, N-1\}$)—apibrėžimo srities intervalais.

Fraktalinės interpoliacinės funkcijos (IFS $\{\mathbb{R}^2;\ \omega_1, \omega_2, \ldots\omega_N\}$ atraktoriaus) duomenų aibei S' generavimas suvedamas į geriausių (VKP prasme) porų „reikšmių srities intervalas—apibrėžimo srities intervalas“ paiešką.

Fiksuokime reikšmių srities intervalą $[x_{i-1}^{'}, x_i^{'}]$, $i\in$1, 2, ..., N. Šiam intervalui randamas apibrėžimo srities intervalas $[x_{j-1}^{'},\ x_{j+1}^{'}]$, $j\in\{1, 2, \ldots, N-1\}$, kuriame apibrėžtos reikšmės $\{F_{\lambda(j-1)}, F_{\lambda(j-1)+1}, \ldots F_{\lambda(j+1)}\}$ geriausiai atvaizduojamos (konkrečios afiniosios transformacijos, tarkime, ω_i pagalba) į reikšmes $\{F_{\lambda(i-1)}, F_{\lambda(i-1)+1}, \ldots F_{\lambda i}\}$, apibrėžtas intervale $[x_{i-1}^{'}, x_i^{'}]$. Tokios afiniosios transformacijos (atvaizdžio ω_i) nustatymui ir realizavimui, visų pirma, dukart turi būti mažinamas su intervalu $[x_{j-1}^{'}$,

x'_{j+1}] susietų reikšmių kiekis. Tai galima atlikti įvairiais būdais. Žemiau siūlomas variantas, kai dvi iš eilės einančios reikšmės keičiamos tų reikšmių aritmetiniu vidurkiu.

Naują (sumažintą) reikšmių, susietų su intervalu [x'_{j-1}, x'_{j+1}], rinkinuką pažymėkime taip— $\{\tilde{F}_{\lambda(j-1)}, \tilde{F}_{\lambda(j-1)+1}, \ldots \tilde{F}_{\lambda j}\}$. Jeigu λ yra nelyginis skaičius, tai pastarosios reikšmės apskaičiuojamos tokiu būdu:

$\tilde{F}_{\lambda(j-1)} = F_{\lambda(j-1)} = F'_{j-1}$, $\tilde{F}_{\lambda j} = F_{\lambda(j+1)} = F'_{j+1}$; $\tilde{F}_{\lambda(j-1)+s} = \frac{1}{2}(F_{\lambda(j-1)+2s-1} + F_{\lambda(j-1)+2s})$, kai s=1, 2, …, (λ-1)/2,

ir

$$\tilde{F}_{\lambda(j-1)+s} = \frac{1}{2}(F_{\lambda(j-1)+2s} + F_{\lambda(j-1)+2s+1}), \text{ kai } s = (\lambda+1)/2, (\lambda+1)/2+1, \ldots, \lambda-1;$$

$$\tilde{F}_{\lambda(j-1)+(\lambda-1)/2+1} = F_{\lambda(j-1)+(\lambda-1)/2+1}.$$

Jeigu λ yra lyginis skaičius, tai siūlome naudotis tokiomis išraiškomis:

$\tilde{F}_{\lambda(j-1)} = F_{\lambda(j-1)} = F'_{j-1}$, $\tilde{F}_{\lambda j} = F_{\lambda(j+1)} = F'_{j+1}$; $\tilde{F}_{\lambda(j-1)+s} = \frac{1}{2}(F_{\lambda(j-1)+2s-1} + F_{\lambda(j-1)+2s})$, kai s=1, 2, …, λ/2-1, ir

$$\tilde{F}_{\lambda(j-1)+s} = \frac{1}{2}(F_{\lambda(j-1)+2s} + F_{\lambda(j-1)+2s+1}), \text{ kai } s = \lambda/2+1, \lambda/2+2, \ldots, \lambda-1;$$

$$\tilde{F}_{\lambda(j-1)+\lambda/2} = \frac{1}{3}(F_{\lambda j-1} + F_{\lambda j} + F_{\lambda j+1}).$$

Toliau, konstruojama suspaudžiančioji (pražulnioji) afinioji transformacija

$$\omega_i = \omega_i(x, y) = \begin{pmatrix} a_i & 0 \\ c_i & d_i \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e_i \\ f_i \end{pmatrix},$$

„pervedanti“ tašką [x'_{j-1}, F'_{j-1}] į tašką [x'_{i-1}, F'_{i-1}], ir tašką [x'_{j+1}, F'_{j+1}]—į tašką [x'_i, F'_i]. Pražulniąją transformaciją ω_i charakterizuojantys koeficientai randami pagal formules, pateiktas 2.3. skyrelyje, būtent:

$$a_i = \frac{x'_i - x'_{i-1}}{x'_{j+1} - x'_{j-1}}, \; c_i = \frac{F'_i - F'_{i-1}}{x'_{j+1} - x'_{j-1}} - d_i \frac{F'_{j+1} - F'_{j-1}}{x'_{j+1} - x'_{j-1}},$$

$$e_i = \frac{x'_{j+1}x'_{i-1} - x'_{j-1}x'_i}{x'_{j+1} - x'_{j-1}}, \; f_i = \frac{x'_{j+1}F'_{i-1} - x'_{j-1}F'_i}{x'_{j+1} - x'_{j-1}} - d_i \frac{x'_{j+1}F'_{j-1} - x'_{j-1}F'_{j+1}}{x'_{j+1} - x'_{j-1}}.$$

Koeficiento d_i reikšmė apskaičiuojama, taikant mažiausiųjų kvadratų metodą, būtent:

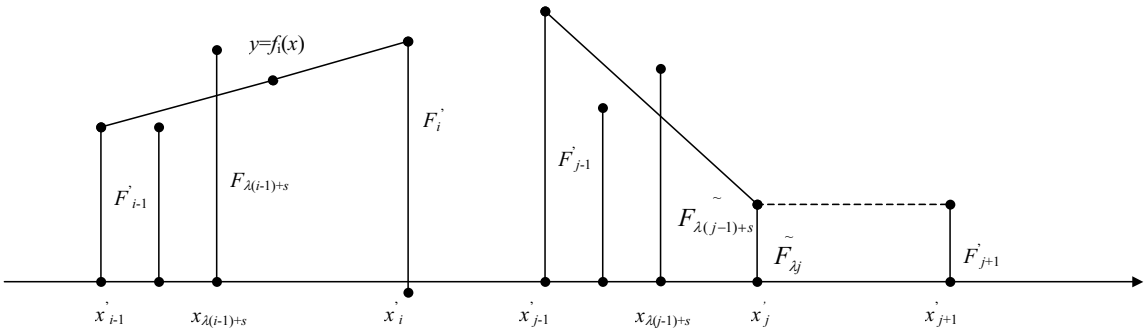
$$\sum_{s=1}^{\lambda-1} (F_{\lambda(i-1)+s} - d_i \tilde{F}_{\lambda(j-1)+s})^2 \to min. \qquad \textbf{(*)}$$

Tai paprasčiausias minimizavimo uždavinys, todėl, prilyginę kairėje pusėje esančios išraiškos išvestinę (kintamojo d_i atžvilgiu) nuliui, nesunkiai randame

$$d_i = \sum_{s=1}^{\lambda-1} F_{\lambda(i-1)+s} \cdot \tilde{F}_{\lambda(j-1)+s} \Bigg/ \sum_{s=1}^{\lambda-1} \left(\tilde{F}_{\lambda(j-1)+s} \right)^2$$

koks bebūtų $j \in \{1, 2, \ldots, N\}$.

Būtina pabrėžti, jog konkrečiam reikšmių srities intervalui $[x'_{i-1}, x'_i]$ ($i \in \{1, 2, \ldots, N\}$) fiksuojamas (kaip geriausias!) tas apibrėžimo srities intervalas (fragmentas) $[x'_{j-1}, x'_{j+1}]$, kuris suteikia minimumą (VKP prasme) **(*)** išraiškai (3.24 pav.).



3.24 pav. Koliažo organizavimo ypatumai

Kartais, siekiant padidinti gerų (VKP prasme) porų „reikšmių srities intervalas—apibrėžimo srities intervalas“ aptikimo tikimybę, fragmentams (intervalams) $[x'_{j-1}, x'_{j+1}]$ papildomai taikoma „atspindžio“ transformacija, t.y. reikalaujama, kad afinioji transformacija ω_i ($i \in \{1, 2, \ldots, N\}$) pervestų tašką (x'_{j-1}, F'_{j-1}) į tašką (x'_i, F'_i) ir tašką (x'_{j+1}, F'_{j+1})—į tašką (x'_{i-1}, F'_{i-1}). Aišku, tokiu atveju atitinkamai turi būti koreguojamos afiniosios transformacijos koeficientų apskaičiavimui skirtos išraiškos.

Aprašytąją Koliažo organizavimo procedūrą galima nesunkiai apibendrinti bet kokiam nepersidengiančiųjų vienodo ilgio reikšmių srities intervalų skaičiui $r \in \{2, 3, \ldots, N\}$ (aukščiau išnagrinėtas atvejis, kai $r=N$) ir bet kokiam apibrėžimo srities intervalų skaičiui $p \in \{1, 2, \ldots, N(1-2/r)+1\}$ (aukščiau išnagrinėtas atvejis, kai $p=N-1$). Toks apibendrinimas kartais leidžia gauti jeigu ne „geresnę“ fraktalinę interpoliacinę funkciją, tai bent paprastesnę (su mažesniu afiniųjų transformacijų skaičiumi) IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots \omega_r\}$.

3.3. Interpoliuojamo objekto fraktalinės dimensijos įvertinimas

Imkime pirminę duomenų aibę

$$S=\{(x_j, F_j)|\, j=0, 1, \ldots, M\}.$$

Tarkime, kad interpoliavimui naudojamas jos poaibis

$$S'=\{(x_i^{'}, F_i^{'})|\, i=0, 1, ..., N\};$$

čia $x_i^{'}=x_{\lambda i}=x_j$, $F_i^{'}=F_{\lambda i}=F_j$, su visais $i=0, 1, \ldots, N$; $\lambda\in\{2, 3, \ldots\}$; $N<M$.

Naudojant tolygų interpoliavimo taškų pasiskirstymą bei anksčiau paminėtą fraktalinių interpoliacinių funkcijų generavimo (IFS pagalba) principą (2.3. skyrelis), belieka apibrėžti generavimo procese taikomų (tuo pačiu, ir INTERPO procedūroje) afiniųjų transformacijų parametrų d_i (i=1, 2, ..., N) reikšmes. Pastebėsime dar kartą, jog šie parametrai keičia vertikalųjį mastelį vaizdavimo metu ir įtakoja galutinį interpoliavimo rezultatą—fraktalinę interpoliacinės funkcijos dimensiją.

Tarkime, kad duomenų aibę S' interpoliuojančioji funkcija sutampa su IFS $\{\mathbb{R}^2; \omega_1, \omega_2, \ldots \omega_N\}$ atraktoriumi A. Tada afiniųjų (pražulniųjų) transformacijų išraiškos yra tokios (2.3. skyrelis):

$$\omega_i(x, y)=\begin{pmatrix} a_i & 0 \\ c_i & d_i \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}+\begin{pmatrix} e_i \\ f_i \end{pmatrix};$$

čia koeficientai a_i, c_i, e_i, f_i apskaičiuojami pagal (*) formules, pateiktas 2.3. skyrelyje. Vertikalųjį mastelį keičiančiųjų koeficientų d_i reikšmių nustatymui siūlome tokią procedūrą.

Pirmiausia, užrašoma tiesės atkarpos, jungiančios taškus $(x_{i-1}^{'}, F_{i-1}^{'})$ ir $(x_i^{'}, F_i^{'})$ ($i\in\{1, 2, ..., N\}$), lygtis (30 pav., 3.2.2. skyrelis):

$$y=f_i(x)=\frac{F_i^{'}-F_{i-1}^{'}}{x_i^{'}-x_{i-1}^{'}}(x-x_{i-1}^{'})+F_{i-1}^{'}.$$

Kadangi į intervalą $[x_{i-1}^{'}, x_i^{'}]$ patenka, bendru atveju, daugiau nei du taškai x_j, $j\in\{0, 1, ..., M\}$, apskaičiuojamas reikšmių

$$\{F_{i-1}^{'}=F_{\lambda(i-1)}, F_{\lambda(i-1)+1}, ..., F_{\lambda(i-1)+\lambda}=F_{\lambda i}=F_i^{'}\}$$

vidurkis m_i bei jų išsibarstymo matas (dispersija) σ_i^2 tiesės $y=f_i(x)$ atžvilgiu. Tam panaudojamos išraiškos

$$m_i=\frac{1}{\lambda+1}\sum_{s=0}^{\lambda}(F_{\lambda(i-1)+s}-y_s),\ \sigma_i^2=\frac{1}{\lambda+1}\sum_{s=0}^{\lambda}(F_{\lambda(i-1)+s}-y_s)^2-m_i^2; \qquad \textbf{(**)}$$

čia $y_s=f_i(x_{\lambda(i-1)+s})$, s=0, 1, ..., λ.

Kadangi, esant fiksuotai (iš anksto apibrėžtai) fraktalinės interpoliacinės funkcijos (IFS atraktoriaus A) dimensijai (tarkime, kad $D(A)=D$) turi būti išpildoma sąlyga (2.3. skyrelis):

$$\sum_{i=1}^{N} |d_i| = N^{D-1},$$

tai koeficientų d_i (i=1, 2, ..., N) reikšmės („grubumo“ faktoriai), užtikrinančios lygybę $D(A)=D$, apskaičiuojamos taip:

$$d_i = \mathrm{sign}(m_i) \cdot k \cdot \sigma_i^2,$$

su visais i=1, 2, ..., N; čia

$$\mathrm{sign}(m_i) = \begin{cases} 1, m_i > 0, \\ 0, m_i = 0, \\ -1, m_i < 0; \end{cases}$$

ir $k = N^{D-1} / \sum_{i=1}^{N} \sigma_i^2$.

Be abejo, pastarosios išraiškos (reikšmių d_i radimui) netinka tuo atveju, kai fraktalinių interpoliacinių funkcijų radimui organizuojamas vienoks ar kitoks koliažas, aprašytas 3.2.2. skyrelyje. Pats koliažo organizavimo procesas leidžia teigti, jog fraktalinė IFS atraktoriaus (interpoliacinės funkcijos) dimensija bus artima nagrinėjamo realaus pasaulio objekto (diskretizuoto jo analogo—duomenų aibės S) fraktalinei dimensijai.

Panagrinėkime pavyzdį, pateiktą 3.1 skyrelyje.

Turime duomenų aibę, sudarytą iš penkių interpoliavimo taškų: $S'=\{(x_i', F_i') \in \{(3, 0.3), (4, 1.2), (7, 0.1), (8, 0.8), (10, 0.3)\} | i=0, 1, 2, 3, 4\}$ Tarkime, kad vertikalaus mastelio keitimo parametro reikšmės yra: $s_i \in \{0.5, 0.2, 0.5, —0.3, 0\}$. Šiuos duomenis perdavę INTERPO, gausime 3.1 skyrelyje pateiktą interpoliavimo rezultatą (3.11 pav.). Pritaikykime duotai duomenų aibei šiame skyrelyje išdėstytą metodiką ir patikrinkime, ar patikslintos parametro s reikšmės, gautos interpoliuojant su INTERPO ir d reikšmės, gautos taikant naująją metodiką sutampa.

Diskretizuokime kiekvieną duotą intervalą, raskime nuokrypius, apskaičiuokime vidurkius m_i bei jų išsibarstymo matą (dispersiją) σ_i^2 tiesės $y=f_i(x)$ atžvilgiu, kai $i \in \{1, 2, 3, 4\}$. Tam panaudokime anksčiau pateiktas išraiškas **(**)**. Gausime, kad

m_1=0,255555555; m_2=0,111111111; m_3=0,315384615; m_4=—0,06;

σ_1=0,26034165; σ_2=0,105409255; σ_3=0,260915547; σ_4=0,260768096;

σ_{1p}=0,245452467; σ_{2p} =0,099380799; σ_{3p}=0,250679549; σ_{4p}=0,233238075.

Tada d reikšmės atitinkamai lygios: d_1=0,29; d_2=0,118; d_3=0,29; d_4=—0,29.

Analogiškai, s reikšmės, gautos interpoliuojant su INTERPO, yra lygios: s_1=0,33; s_2=0,13; s_3=0,33; s_4=—0,20. Skirtumai nežymūs ir gauti tik dėl netikslių nuokrypių matavimų rezultatų.

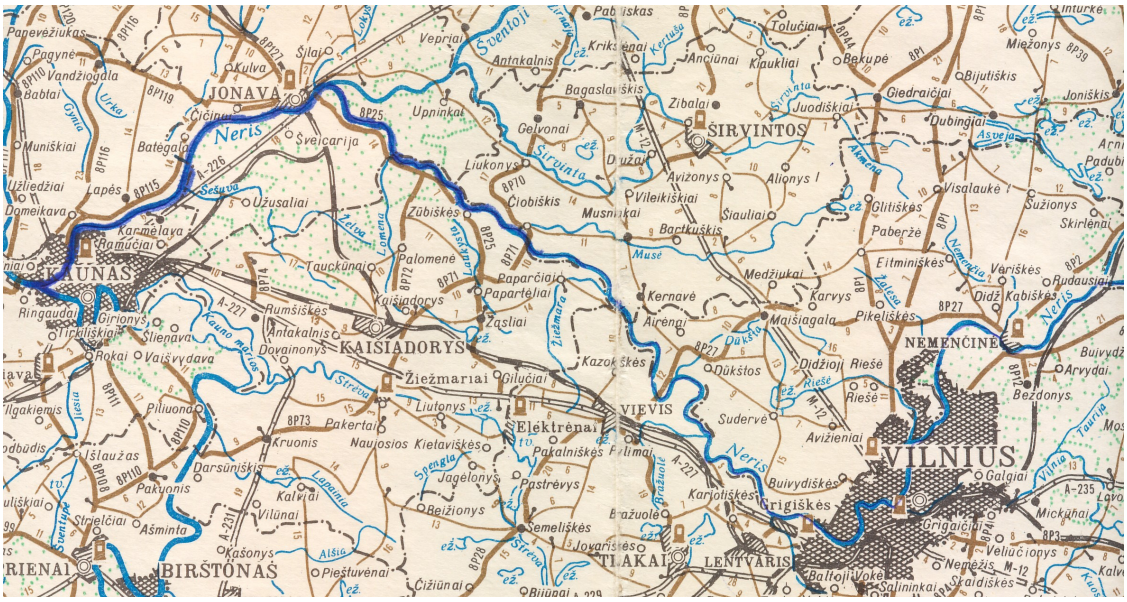
3.4. Eksperimento rezultatų analizė

Fraktalinių interpoliacinių funkcijų praktinio panaudojimo eksperimentas buvo atliktas naudojant fraktalų interpoliacijos tyrimams skirtą priemonę INTERPO. Atliktuose eksperimentuose buvo naudotas koliažas, kai „reikšmių srities intervalų“ skaičius lygus [M/λ], o „apibrėžimo srities intervalų“ skaičius lygus r=1. Eksperimentas taikytas trims skirtingiems objektams:

* Landšafto objektui;
* Valiutų biržos prekybos objektui;
* Akcijų biržos prekybos objektui.

1. FIF generavimas INTERPO pagalba landšafto objektui.

Pirmasis eksperimentas buvo atliktas, panaudojant konkretų geografinio landšafto objektą – Neries upės vagą tarp Kauno ir Vilniaus:



3.25 pav. Neries upės vagos žemėlapyje skenuotas vaizdas

Tyrimo metu buvo panaudoti keturi plokštuminių koordinačių rinkiniai (priemonės):

- Elementarus rankinis metodas: ant objekto buvo uždėtas 0,5x0,5 cm tankio tinklelis; jo pagalba buvo nustatytos atitinkamų objekto taškų koordinatės (buvo imta vertikalių tinklelio dedamųjų ir objekto linijos sankirtos taškai); gautos koordinatės surašytos į masyvą:

3.1 lentelė

Koordinatinio duomenų masyvo turinys.

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
y	5.0	6.1	6.35	7.75	8.0	8.3	7.8	7.1	6.4	5.5	5.3	3.9	2.5	1.25	0.6	1.1	2.3	4.25	3.8	4.8	4.2	4.2

Kuris pateiktas tekstiniame fip tipo dokumente:

22

1.0 1.0

0 5.0 0.5

1.0 6.1 0.5

2.0 6.35 0.5

3.0 7.75 0.5

4.0 8.0 0.5

5.0 8.3 0.5

6.0 7.8 0.5

7.0 7.1 0.5

8.0 6.4 0.5

9.0 5.5 0.5

10.0 5.3 0.5

11.0 3.9 0.5

12.0 2.5 0.5

13.0 1.25 0.5

14.0 0.6 0.5

15.0 1.1 0.5

16.0 2.3 0.5

17.0 4.25 0.5

18.0 3.8 0.5

19.0 4.8 0.5

20.0 4.2 0.5

21.0 4.2;

- Taškų koordinatės nustatytos grafinio redaktoriaus Paint Shop Pro8 pagalba, panaudojus analogišką tinklelį:

3.2 lentelė

Naujas duomenų rinkinys

x	12	39	74	115	135	161	212	223	235	246	269	282	288	295	305	327	379	415	451
y	3	11	59	112	70	94	134	173	186	192	202	201	202	203	219	215	247	208	200

- Taškų koordinatės nustatytos grafinio redaktoriaus Arc Soft PhotoStudio2000 pagalba, panaudojus 100.0x100.0 tinklelį:

3.3 lentelė

Trečias duomenų rinkinys

x	100	200	300	400	500	600	700	800	900	1000	1100	1200	1300	1400
y	612	732	768	906	959	974	1029	950	860	768	716	670	566	379

3.4 lentelė

3.3 lentelės rinkinio tęsinys

x	1500	1600	1700	1800	1900	2000	2100	2200	2300
y	240	164	118	76	170	508	502	495	602

- Taškų koordinatės nustatytos, panaudojus specialią programinę įrangą (buvo naudota GIS (Arcview and Arcgis) *.shp formato duomenys; upė vektorizuota 1:200 000 masteliu; vektorizacija atlikta LKS 94 koordinačių sistemoje.):

3.5 lentelė

Skaitmenizuotą vaizdą nusakantis duomenų rinkinys

x	491849.27	499779.22	507109.17	514511.42	522896.78	528405.09	533906.17	539877.12	545277.00
y	6084850.49	6091211.79	6098433.32	6102763.34	6104772.94	6100103.16	6097124.91	6092932.23	6088313.06

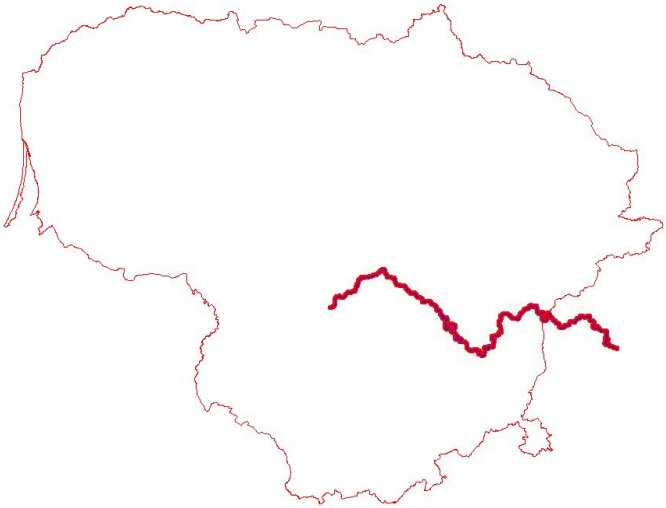
3.6 lentelė

3.5 lentelės rinkinio tęsinys

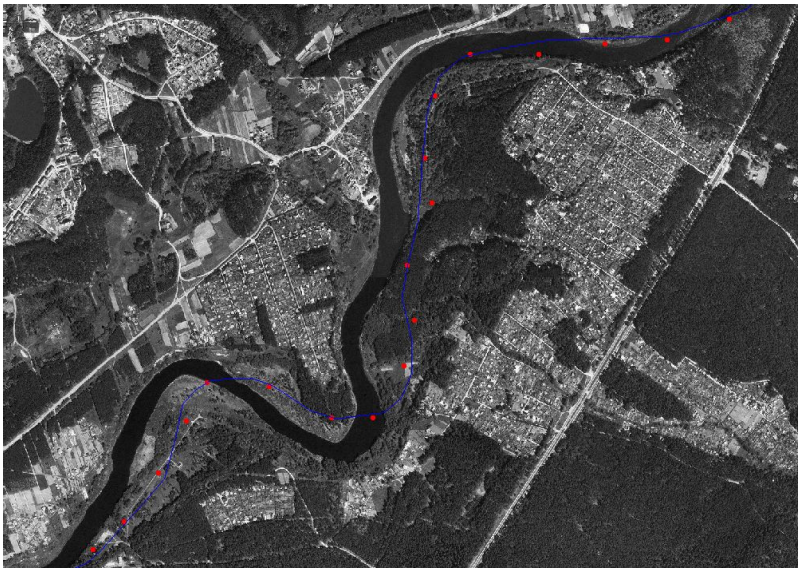
x	552050.35	555361.12	557096.02	559206.82	560479.08	563399.49	569059.61
y	6085363.73	6080036.13	6075518.16	6075669.97	6071780.90	6066937.63	6064378.65

(imtas kas 50-tas taškas)

Pirmaisiais trimis atvejais naudotas Lietuvos Respublikos žemėlapio 1: 600 000 skenuotas vaizdas. Ketvirtuoju atveju— vektorizuotas 1:200 000 vaizdas, kurį pateikė geologijos institutas:



3.26 pav. Bendras vaizdas



3.27 pav. Paklaida

Užduotis: apdoroti gautas reikšmių lenteles, sudarant optimaliausią interpoliavimo taškų masyvą, nusakantį generuojamą FIF.

Turimi koordinačių masyvai surašomi į fip tipo dokumentus (rankiniu būdu pasirinktoms koordinatėms -Nerisr1, su Photo Shop8 – Nerisps, su PhotoStudio2000 - NerisAps, o iš vektorizuoto vaizdo -Neris3_10 (kas dešimtas taškas); Neris3_50 (kas penkiasdešimtas taškas); upe3_100 (kas šimtasis taškas), t.y. taikytas nuoseklios atrankos metodas), juose nurodant taškų koordinates bei pasirinktą vertikalųjį mastelį keičiantį parametrą (0.5). Tyrimo eigoje buvo keičiamos dimensijos ir stebimi gauti rezultatai.

Pirmuoju atveju nagrinėjame 22 taškus, nusakančius upės vagą:

Nerisr1.fip:

```
22
1.0 1.0
0 5.0 0.5
1.0 6.1 0.5
2.0 6.35 0.5
3.0 7.75 0.5
4.0 8.0 0.5
5.0 8.3 0.5
6.0 7.8 0.5
7.0 7.1 0.5
8.0 6.4 0.5
9.0 5.5 0.5
10.0 5.3 0.5
11.0 3.9 0.5
12.0 2.5 0.5
13.0 1.25 0.5
14.0 0.6 0.5
15.0 1.1 0.5
16.0 2.3 0.5
17.0 4.25 0.5
18.0 3.8 0.5
19.0 4.8 0.5
20.0 4.2 0.5
21.0 4.2
```

Taškų sk.=22 ; Skalė=1.0x1.0; *D*=1.1

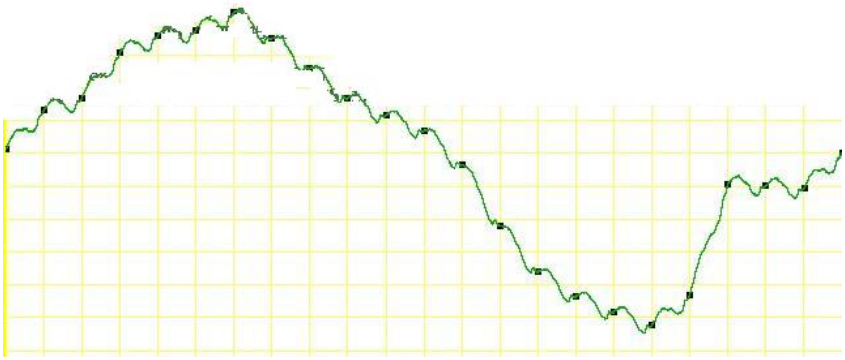
3.28 pav. Gauta FIF kreivė

Gauta kreivė vizualiai lyg ir panaši į nagrinėjamą objektą, tačiau (net ir keičiant dimensiją) tai nėra norimas rezultatas. Tolimesniam tyrimui tenka pasirinkti duomenis iš likusių rinkinių.

NerisAps.fip:

```
23
100 100
100 612 0.5
200 732 0.5
300 768 0.5
400 906 0.5
500 959 0.5
600 974 0.5
700 1029 0.5
800 950 0.5
900 860 0.5
1000 768 0.5
1100 716 0.5
1200 670 0.5
1300 566 0.5
1400 379 0.5
1500 240 0.5
1600 164 0.5
1700 118 0.5
1800 76 0.5
1900 170 0.5
2000 508 0.5
2100 502 0.5
2200 495 0.5
2300 602
```

Taškų sk.=23 ; Skalė=100x100; *D*=1.1



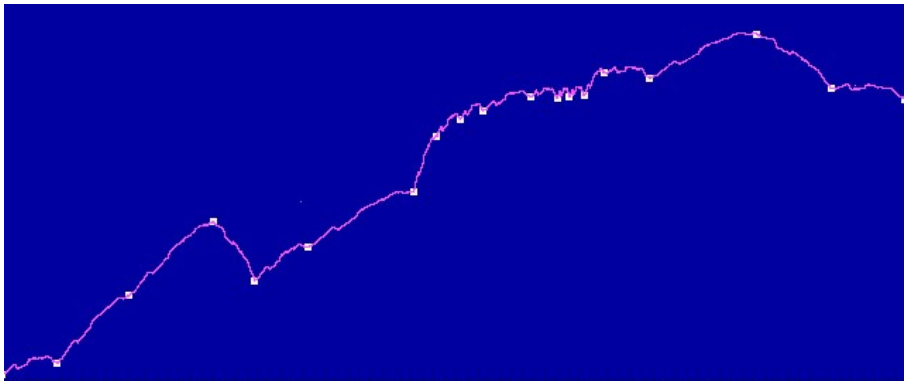
3.29 pav. Gauta FIF kreivė

Gauta kreivė (geriausiu atveju) šiek tiek panaši į kairiąją upės dalį, tad tenka išbandyti ir kitą rinkinį:

Neris2.fip:

19
1 1
12 3 0.5
39 11 0.5
74 59 0.5
115 112 0.5
135 70 0.5
161 94 0.5
212 134 0.5
223 173 0.5
235 186 0.5
246 192 0.5
269 202 0.5
282 201 0.5
288 202 0.5
295 203 0.5
305 219 0.5
327 215 0.5
379 247 0.5
415 208 0.5
451 200 0.5

Taškų sk.=5 ; Skalė=1x1; *D*=1.03



3.30 pav. Gauta FIF kreivė

Bei paskutinįjį rinkinį:

n609a.fip:

9

1 1

491849.27 6084850.49 0.5

492976.96 6085262.53 0.5

493996.21 6086166.12 0.5

494307.05 6087149.23 0.5

494415.48 6088009.45 0.5

494422.71 6088920.28 0.5

494610.66 6090192.54 0.5

495217.87 6091045.53 0.5

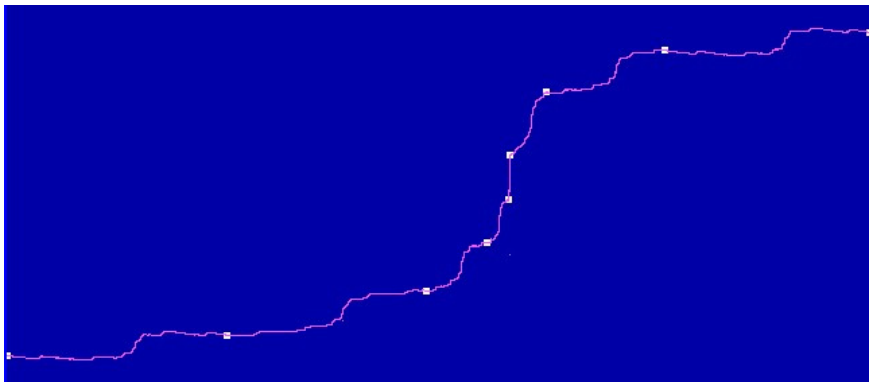
496258.81 6091414.20 0.5

Taškų sk.=9 ; Skalė=1x1; *D*=1.03

3.7 lentelė

Gauti skaičiavimų rezultatai

x	491849.28	492976.97	493996.22	494307.06	494415.47	494422.72	494610.66	495217.88	496258.81
y	6084850.50	6085262.50	6086166.00	6087149.00	6088009.50	6088920.50	6090192.50	6091045.50	6091414.00
s	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	
a	0.26	0.23	0.07	0.02	0.00	0.04	0.14	0.24	
c	-0.10	0.01	0.02	-0.00	0.01	0.09	-0.00	-0.11	
e	366064.41	379287.47	459324.00	482215.19	493606.78	473459.75	426880.06	379109.31	
f	5326732.50	5272321.50	5264357.50	5279004.50	5274232.00	5234876.50	5282884.50	5337779.50	



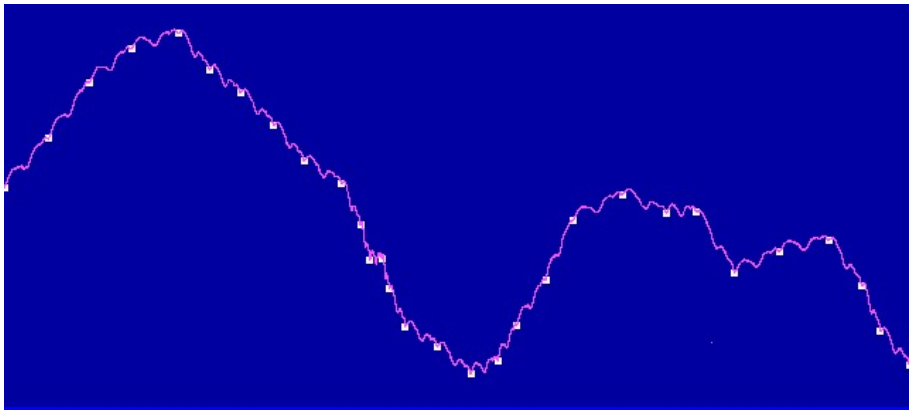
3.31 pav. Gauta FIF kreivė

Pastaroji kreivė artimiausia analizuojamai, tačiau optimalesniam rezultatui gauti reikia keisti dimensiją bei duomenų parinkimo metodą. Tad tolimesniam tyrimui pasirinktas vaizdo (objekto) vektorizuotas duomenų rinkinys. Pirmiausia, analizuojamas kas 50 įrašas:

Neris3_50.fip:

```
30
50 1
491849,27 6084850,49 0,5
499779,22 6091211,79 0,5
507109,17 6098433,32 0,5
514511,42 6102763,34 0,5
522896,78 6104772,94 0,5
528405,09 6100103,16 0,5
533906,17 6097124,91 0,5
539877,12 6092932,23 0,5
545277 6088313,06 0,5
552050,35 6085363,73 0,5
555361,12 6080036,13 0,5
557096,02 6075518,16 0,5
559206,82 6075669,97 0,5
560479,08 6071780,9 0,5
563399,49 6066937,63 0,5
569059,61 6064378,65 0,5
575072,81 6060750,89 0,5
579901,23 6062531,76 0,5
583299,94 6067071,73 0,5
588416,82 6072916,01 0,5
593220,16 6080603,87 0,5
602024,2 6083914,79 0,5
610038,13 6081569,55 0,5
615192,63 6081820,38 0,5
622153,09 6073844,07 0,5
630129,4 6076590,63 0,5
638933,43 6078108,13 0,5
644852,96 6072226,23 0,5
648126,25 6066344,33 0,5
653330,92 6061929,77 0,5
```

Taškų sk.=30 ; Skalė=50x1; *D*=1.1



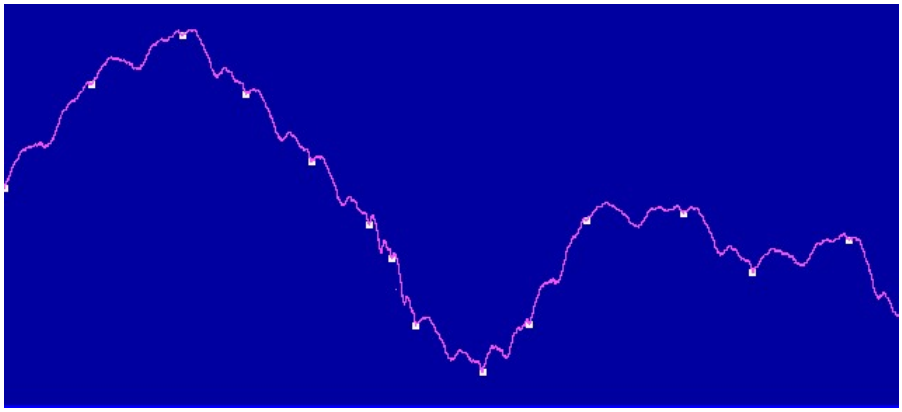
3.32 pav. Gauta FIF kreivė

Toliau išbandoma kas 100 elementas:

Neris3_100.fip:

```
15
100 1
491849.27 6084850.49 0.5
507109.17 6098433.32 0.5
522896.78 6104772.94 0.5
533906.17 6097124.91 0.5
545277 6088313.06 0.5
555361.12 6080036.13 0.5
559206.82 6075669.97 0.5
563399.49 6066937.63 0.5
575072.81 6060750.89 0.5
583299.94 6067071.73 0.5
593220.16 6080603.87 0.5
610038.13 6081569.55 0.5
622153.09 6073844.07 0.5
638933.43 6078108.13 0.5
648126.25 6066344.33 0.5
```

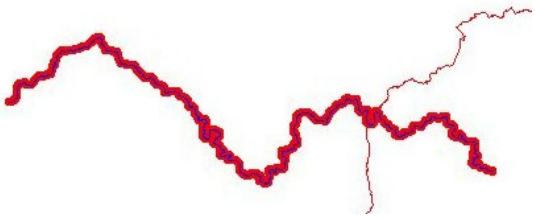
Taškų sk.=15 ; Skalė=100x1; *D*=1.1



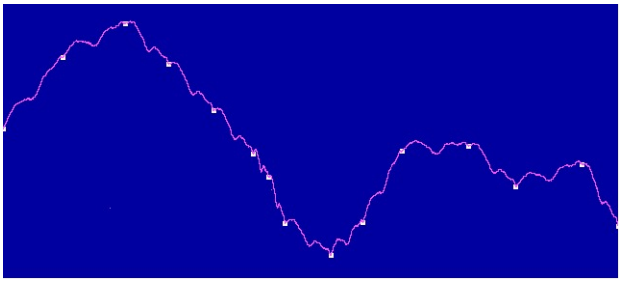
3.33 pav. Gauta FIF kreivė

Ankstesniuoju atveju gautas detalesnis vaizdas, bet, vizualiai vertinant, geresnė FIF (panašesnis vaizdas) gautas šiame pavyzdyje.

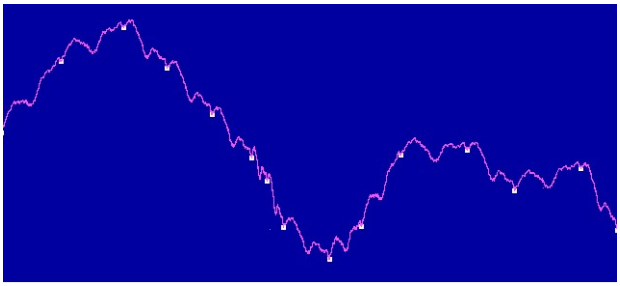
Paanalizuokime kreivės (FIF grafiko) priklausomybę nuo dimensijos *D* pasikeitimo:



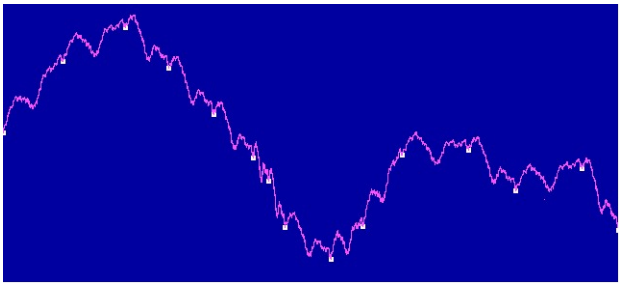
3.34 pav. Upės vagos vaizdas žemėlapyje



3.35 pav. Gauta FIF kreivė, kai *D*=1.03



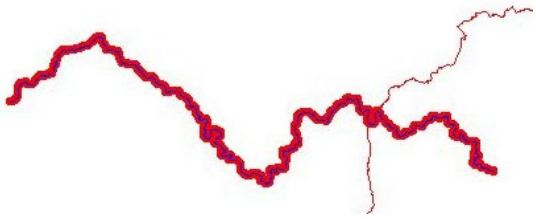
3.36 pav. Gauta FIF kreivė, kai *D*= 1.2



3.37 pav. Gauta FIF kreivė, kai *D*= 1.3

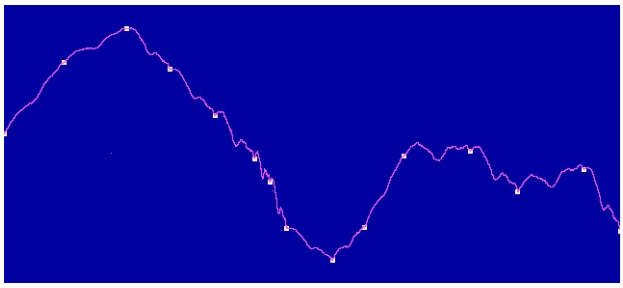
Kaip matome, artimiausias rezultatas yra gautas, kai buvo pasirinkta *D*=1.03.

Toliau, pabandykime šiek tiek pakeisti vertikalųjį mastelį nusakančio parametro reikšmes:

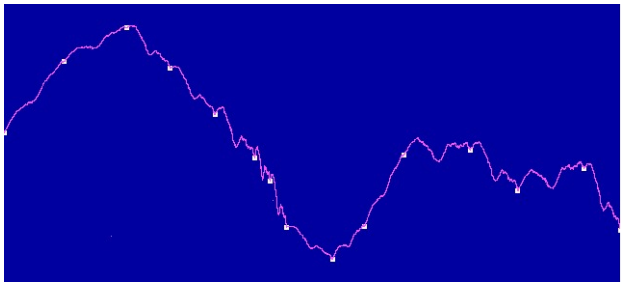


3.38 pav. Neries upės vaga žemėlapyje

15
100 1
491849.27 6084850.49 0.2
507109.17 6098433.32 0.2
522896.78 6104772.94 0.3
533906.17 6097124.91 0.3
545277 6088313.06 0.5
555361.12 6080036.13 0.5
559206.82 6075669.97 0.5
563399.49 6066937.63 0.2
575072.81 6060750.89 0.2
583299.94 6067071.73 0.2
593220.16 6080603.87 0.5
610038.13 6081569.55 0.5
622153.09 6073844.07 0.5
638933.43 6078108.13 0.5
648126.25 6066344.33 0.5

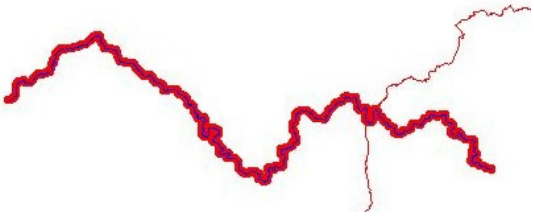


3.39 pav. Gauta FIF kreivė, kai *D* =1.0

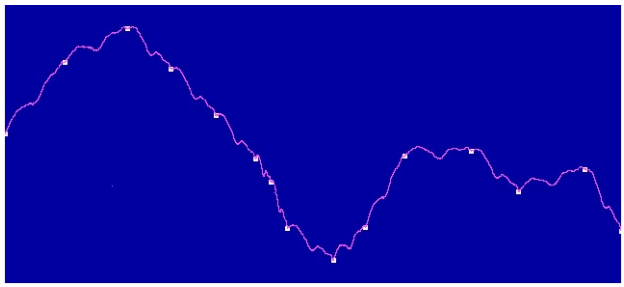


3.40 pav. Gauta FIF kreivė, kai *D*= 1.1

Toliau dar sykį pakeiskime vertikaliojo mastelio parametrą:



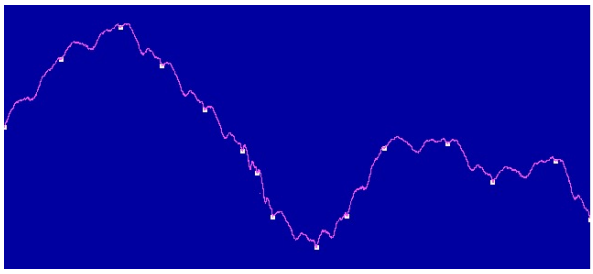
3.41 pav. Neries vagos vaizdas žemėlapyje.



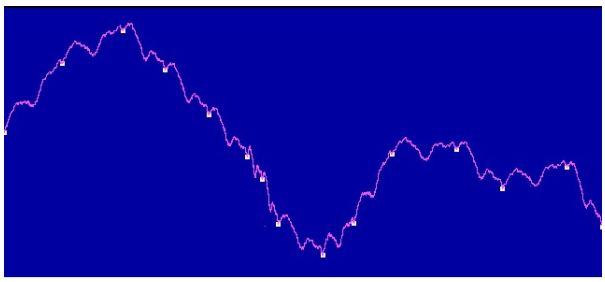
3.42 pav. Gauta FIF kreivė, kai *D*= 1.0

Bei stebėkime fraktalinės dimensijos pasikeitimą:

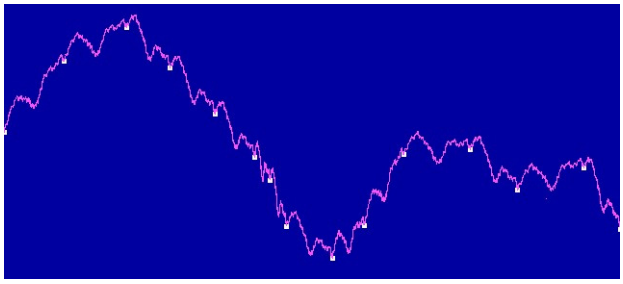
15
100 1
491849.27 6084850.49 0.7
507109.17 6098433.32 0.7
522896.78 6104772.94 0.7
533906.17 6097124.91 0.7
545277 6088313.06 0.7
555361.12 6080036.13 0.7
559206.82 6075669.97 0.7
563399.49 6066937.63 0.7
575072.81 6060750.89 0.7
583299.94 6067071.73 0.7
593220.16 6080603.87 0.7
610038.13 6081569.55 0.7
622153.09 6073844.07 0.7
638933.43 6078108.13 0.7
648126.25 6066344.33 0.7



3.43 pav. Gauta FIF kreivė, kai *D*= 1.1



3.44 pav. Gauta FIF kreivė, kai *D*= 1.2



3.45 pav. Gauta FIF kreivė, kai *D*= 1.3

Šis eksperimentas patvirtino faktą, kad landšafto objektui gali būti taikomi fraktalų interpoliavimo principai. Be to, interpoliavimas gali būti taikomas naudojant įvairiais metodais parinktus duomenis, nusakančius nagrinėjamą objektą. Tačiau, naudojant detaliau objektą nusakančius duomenis, gali iškilti duomenų atrankos problema, kurios sprendimas, pateiktas ankstesniuose šio darbo skyreliuose, šiuo eksperimentu buvo patvirtintas praktiškai.

Analogiškai buvo atliktas ir antrasis eksperimentas.

2. *Valiutų biržos duomenų generavimas Interpo pagalba*

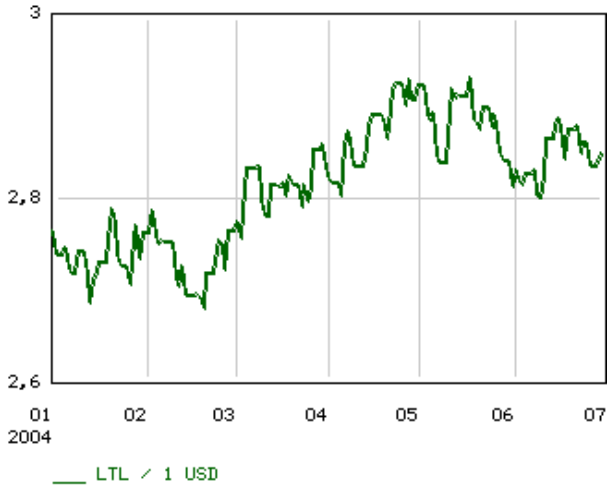
Sekančiam tyrimui pasirinktas dolerio kurso kitimo tyrimas. Buvo analizuojami dolerio kurso pokyčiai per 2004 01 01 – 2004 07 01 laikotarpį. Tyrimui panaudoti Lietuvos banko duomenys (http://www.lbank.lt/exchange/history.asp?Lang=L&Cid=USD&Y=2004 &M=1&D=1&id=77461&ord=1&dir=ASC)

JAV doleris (USD)

litais už valiutos vienetą

2004-06-29	2,8463
2004-06-28	2,8405
2004-06-27	2,8332
2004-06-26	2,8332
2004-06-25	2,8332
2004-06-24	2,8590
2004-06-23	2,8590
2004-06-22	2,8476
2004-06-21	2,8778
2004-06-20	2,8731

3.46 pav. JAV dolerio kurso kaitos lentelė



3.47 pav. JAV dolerio kurso kaitos diagrama

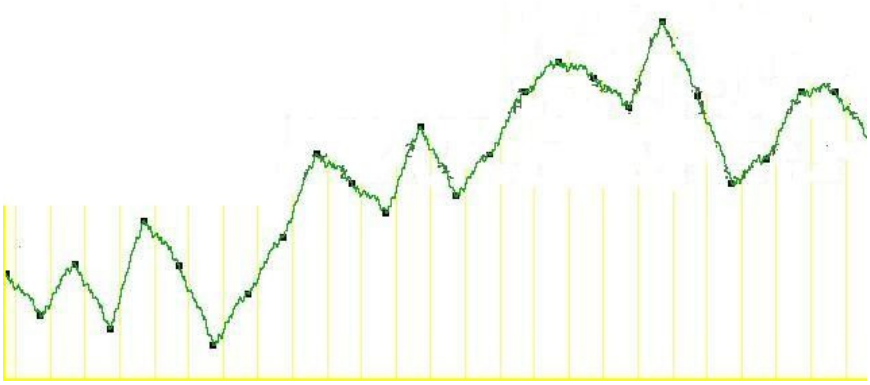
Remiantis šiais duomenimis, vėlgi, pirmiausiai buvo suformuoti duomenų rinkiniai (doleris, doleris1-kiekvieno pirmadienio duomenys, doleris7 – kiekvieno sekmadienio, doleris_pmd-kiekvieno mėnesio pirmos dienos duomenys.)

INTERPO pritaikome „pirmadienių“ rezultatų rinkiniui:

Doleris1.fip:

26
7 1
5 2.7451 0.5
12 2.7146 0.5
19 2.7527 0.5
26 2.7048 0.5
33 2.7842 0.5
40 2.7511 0.5
47 2.6936 0.5
54 2.7311 0.5
61 2.7721 0.5
68 2.8324 0.5
75 2.8112 0.5
82 2.7896 0.5
89 2.852 0.5
96 2.8019 0.5
103 2.8324 0.5
110 2.8777 0.5
117 2.8994 0.5
124 2.8883 0.5
131 2.8658 0.5
138 2.9293 0.5
145 2.8752 0.5
152 2.8114 0.5
159 2.8289 0.5
166 2.8777 0.5
173 2.8778 0.5
180 2.8405 0.5

Dienų sk.=26; Skalė=7x1; *D*=1.1



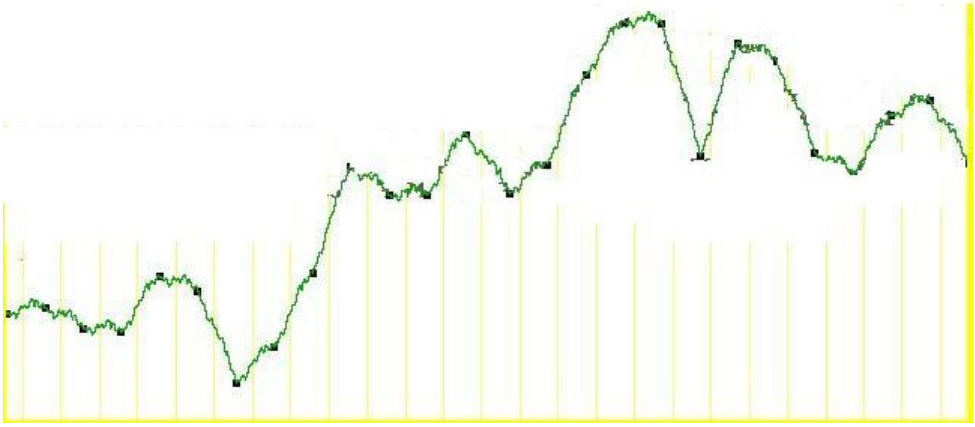
3.48 pav. Gauta FIF kreivė

Imame „sekmadienių“ duomenis:

Doleris7.fip:

26
7 1
4 2.7376 0.5
11 2.7412 0.5
18 2.7281 0.5
25 2.7258 0.5
32 2.7607 0.5
39 2.752 0.5
46 2.6936 0.5
53 2.7166 0.5
60 2.7635 0.5
67 2.8307 0.5
74 2.8129 0.5
81 2.8125 0.5
88 2.852 0.5
96 2.8141 0.5
103 2.8324 0.5
110 2.889 0.5
117 2.9231 0.5
124 2.9218 0.5
131 2.8375 0.5
138 2.9097 0.5
145 2.8975 0.5
152 2.8396 0.5
159 2.8256 0.5
166 2.8636 0.5
173 2.8731 0.5
180 2.8332 0.5

Dienų sk.=26; Skalė=7x1; *D*=1.1



3.49 pav. Gauta FIF kreivė

Bei pirmųjų mėnesių dienų duomenis:

Doleris_pmd.fip:

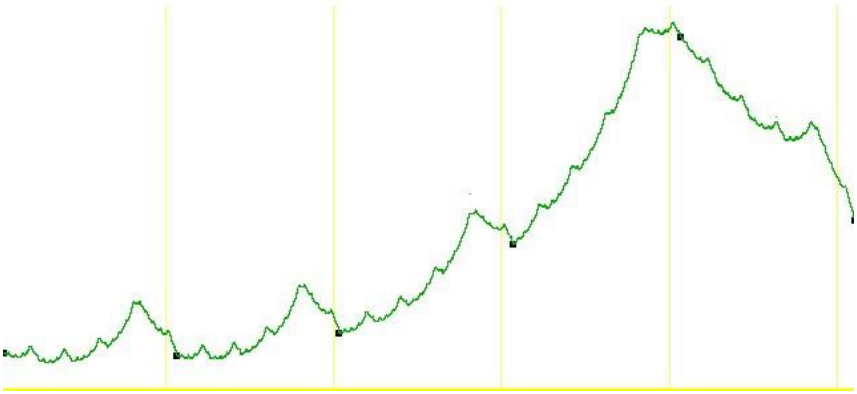
6
30 1
1 2.7621 0.5
32 2.7607 0.5
61 2.7721 0.5
92 2.8169 0.5
122 2.9218 0.5
153 2.8291 0.5

Dienų sk.=6; Skalė=30x1; *D*=1.1

3.8 lentelė

Gauti skaičiavimų rezultatai

x	1.00	32.00	61.00	92.00	122.00	153.00
y	2.76	2.76	2.77	2.82	2.92	2.83
s	0.23	0.23	0.23	0.23	0.23	
a	0.20	0.19	0.20	0.20	0.20	
c	-0.00	-0.00	0.00	0.00	-0.00	
e	0.80	31.81	60.80	91.80	121.80	
f	2.11	2.11	2.12	2.17	2.27	



3.50 pav. Gauta FIF kreivė

Pakeičiame dimensiją:

Doleris_pmd.fip:

6

30 1

1 2.7621 0.5

32 2.7607 0.5

61 2.7721 0.5

92 2.8169 0.5

122 2.9218 0.5

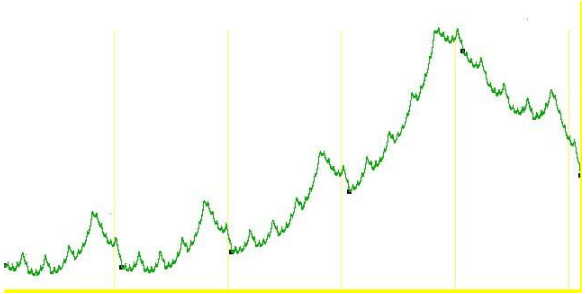
153 2.8291 0.5

Dienų sk.=6; Skalė=30x1; *D*=1.3

3.9 lentelė

Gauti skaičiavimų rezultatai

x	1.00	32.00	61.00	92.00	122.00	153.00
y	2.76	2.76	2.77	2.82	2.92	2.83
s	0.32	0.32	0.32	0.32	0.32	
a	0.20	0.19	0.20	0.20	0.20	
c	-0.00	-0.00	0.00	0.00	-0.00	
e	0.80	31.81	60.80	91.80	121.80	
f	1.87	1.87	1.88	1.92	2.03	



3.51 pav. Gauta FIF kreivė

Iš gautų rezultatų galima daryti išvadą, kad nei pirmųjų mėnesių dienų, nei pirmadienių bei sekmadienių rezultatai nepateikia reikiamo rezultato. Artimiausią rezultatą gauname pasirinkę sekmadienių duomenis.

Pakeiskime interpoliavimo duomenų parinkimo taisyklę ir pasirinkime kas10-tą bei kas 20-tą dieną. Kas 10-ta diena:

Dol10.fip:

19

10 1

1 2.7621 0.5

11 2.7412 0.5

21 2.7719 0.5

31 2.7607 0.5

41 2.7109 0.5

51 2.7166 0.5

61 2.7721 0.5

71 2.7785 0.5

81 2.8125 0.5

91 2.8307 0.5

101 2.8324 0.5

111 2.8635 0.5

121 2.9218 0.5

131 2.8658 0.5

141 2.8739 0.5

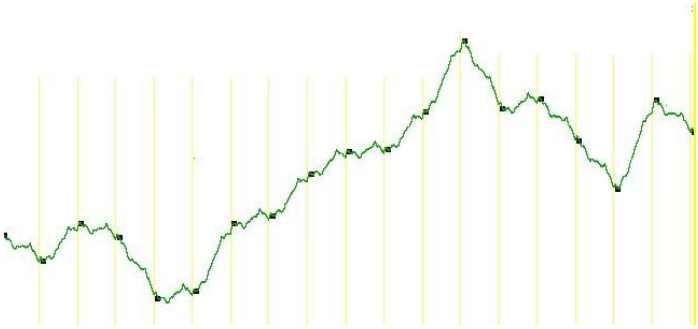
151 2.8396 0.5

161 2.7998 0.5

171 2.8731 0.5

181 2.8463 0.5

Dienų sk.=19; Skalė=10x1; *D*=1.1



3.52 pav. Gauta FIF kreivė

Kas 20-ta diena:

Dol20.fip: Dienų sk.=10; Skalė=20x1; *D*=1.1

```
10
20 1
1       2.7621 0.5
21      2.7719 0.5
41      2.7109 0.5
61      2.7721 0.5
81      2.8125 0.5
101     2.8324 0.5
121     2.9218 0.5
141     2.8739 0.5
161     2.7998 0.5
181     2.8463 0.5
```

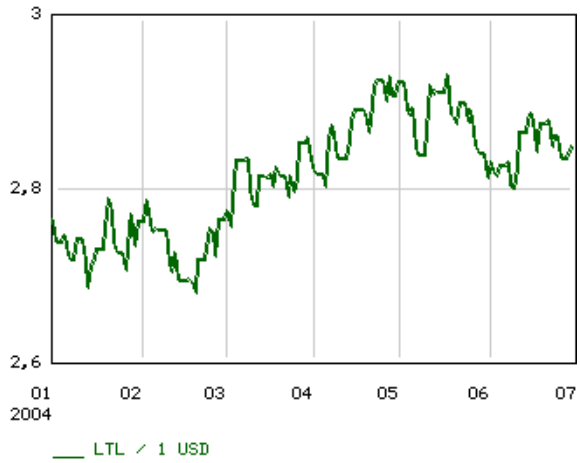
3.10 lentelė

Gauti skaičiavimų rezultatai

x	1.00	21.00	41.00	61.00	81.00	101.00	121.00	141.00	161.00	181.00
y	2.76	2.77	2.71	2.77	2.81	2.83	2.92	2.87	2.80	2.85
s	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	
a	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	
c	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	
e	0.89	20.89	40.89	60.89	80.89	100.89	120.89	140.89	160.89	
f	2.38	2.39	2.33	2.39	2.43	2.45	2.54	2.49	2.42	

3.53 pav. Gauta FIF kreivė

Gautus rezultatus galime palyginti su LB pateikta grafine diagrama:



3.54 pav. LB pateikta grafinė diagrama

Ši eksperimento dalis parodė, kad, pasirinkę tinkamiausią duomenų atrankos metodą, interpoliuoti galime ir informaciją apie bet kokios valiutos kurso pakitimus per pasirinktą laikotarpį.

Toliau buvo įvykdytas paskutinysis eksperimentas.

3. Akcijų biržos duomenų generavimas Interpo pagalba

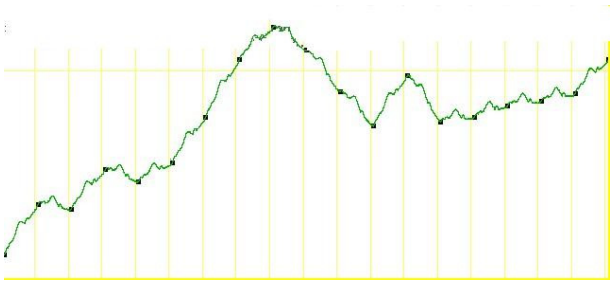
Prekybos AB "Mažeikių Nafta" akcijomis tyrimas

Tyrimui naudosime Vilniaus Valstybinių popierių biržos duomenis. Analizuosime kas 10-tos bei kas 20-tos dienos duomenis. Analizės laikotarpis 2004 01 01 – 2004 09 20.

Kas 10- tos dienos:

Nafta10.fip: Dienų sk.=19; Skalė=10x1; *D*=1.1

```
19
10 1
1       2.20 0.5
11      2.42 0.5
21      2.40 0.5
31      2.57 0.5
41      2.52 0.5
51      2.60 0.5
61      2.80 0.5
71      3.05 0.5
81      3.19 0.5
91      3.09 0.5
101     2.91 0.5
111     2.76 0.5
121     2.98 0.5
131     2.78 0.5
141     2.80 0.5
151     2.85 0.5
161     2.87 0.5
171     2.90 0.5
181     3.05 0.5
```



3.55 pav. Gauta FIF kreivė

Kas 20-tos dienos:

Nafta20.fip: Dienų sk.=10; Skalė=20x1; *D*=1.1

10

20 1

1 2.20 0.5

21 2.40 0.5

41 2.52 0.5

61 2.80 0.5

81 3.19 0.5

101 2.91 0.5

121 2.98 0.5

141 2.80 0.5

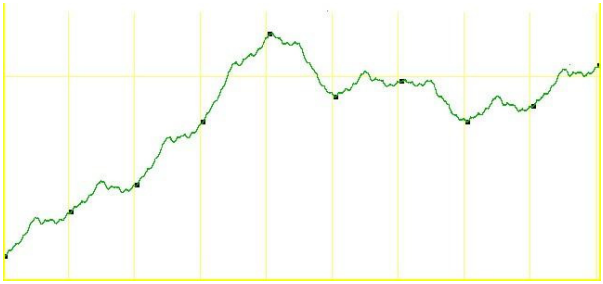
161 2.87 0.5

181 3.05 0.5

3.11 lentelė

Gauti skaičiavimų rezultatai

x	1.00	21.00	41.00	61.00	81.00	101.00	121.00	141.00	161.00	181.00
y	2.20	2.40	2.52	2.80	3.19	2.91	2.98	2.80	2.87	3.05
s	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.14	
a	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	
c	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	
e	0.89	20.89	10.89	60.89	80.89	100.89	120.89	140.89	160.89	
f	1.90	2.10	2.21	2.49	2.89	2.61	2.68	2.50	2.57	

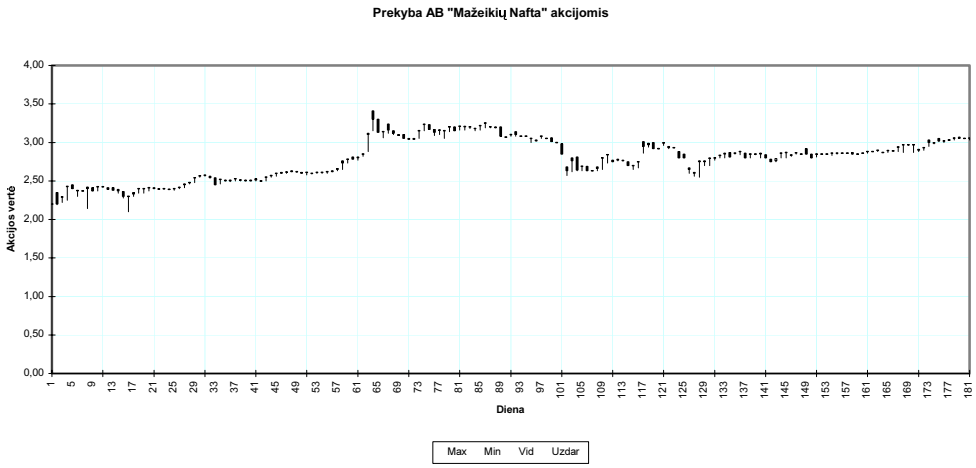


3.56 pav. Gauta FIF kreivė

Gautus rezultatus vėl galime palyginti su biržos svetainėje pateikiama grafine diagrama:

3.57 pav. Biržos svetainėje pateikta grafinė diagrama

Trendui:



3.58 pav. MS Excel diagrama pusei metų (duomenys fiksuoti kasdien)

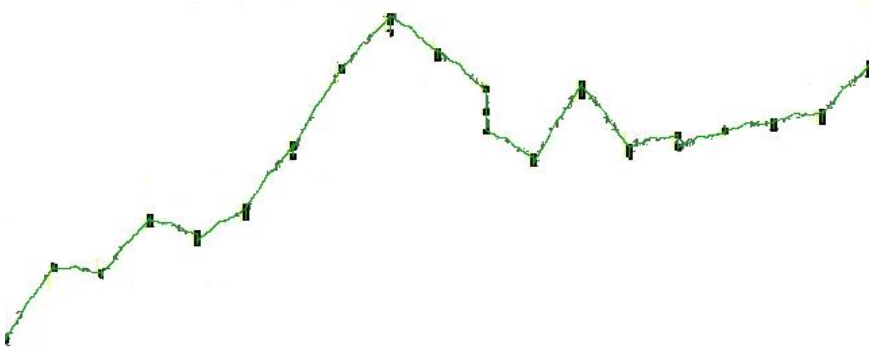
Atliekant šią eksperimento dalį buvo panaudota išplėstinė informacija:

3.12 lentelė

Duomenys apie prekybą akcijomis

	Max	Min	Vid	Uzdar
1	2,20	2,20	2,20	2,20
11	2,43	2,42	2,42	2,42
21	2,41	2,40	2,40	2,40
31	2,58	2,56	2,57	2,57
41	2,53	2,50	2,52	2,51
51	2,61	2,58	2,60	2,61
61	2,81	2,77	2,80	2,80
71	3,05	3,04	3,05	3,04
81	3,21	3,16	3,19	3,21
91	3,10	3,08	3,09	3,10
101	2,98	2,85	2,91	2,85
111	2,77	2,75	2,76	2,75
121	3,00	2,96	2,98	2,99
131	2,80	2,77	2,78	2,80
141	2,84	2,80	2,80	2,80
151	2,85	2,81	2,85	2,85
161	2,88	2,86	2,87	2,88
171	2,91	2,88	2,90	2,90
181	3,06	3,03	3,05	3,05

MS Excel prekybos AB “Mažeikių Nafta” akcijomis duomenys, kai imama kas 10-ta diena



3.59 pav. Interpo rezultatas, kai duomenys skaitomi iš Nafta10t.fip (76 taškai, skalė 10x1, *D*=1.1)

Toliau keičiama dimensija:

3.60 pav. Gauta FIF kreivė *D*=1.5

3.61 pav. Gauta FIF kreivė *D*=1.6

3.62 pav. Gauta FIF kreivė *D*=1.7

3.63 pav. MS Excel pusmečio rezultatas

Palyginę pastarojo eksperimento rezultatus su MS Excel pagalba gautomis diagramomis ar su diagramomis, pateiktomis Internet'e, galima teigti, kad ir šio tipo (informacijai apie prekybą akcijomis) galima taikyti fraktalinio interpoliavimo principus ir priemones. Be to, šiomis priemonėmis galima gauti rezultatus, labai panašius į realius —tereikia tik tinkamai parinkti interpoliavimo taškus (metodai aprašyti ankstesniuose skyreliuose) bei vertikalųjį mastelį keičiančio parametro reikšmes.

Apibendrinant visus tris eksperimentus galima teigti, kad fraktalinės interpoliacinės funkcijos gali būti taikomos tiek landšafto duomenų, tiek valiutų ar akcijų biržų duomenų analizei bei kitai analogiško pobūdžio informacijai (statistiniams, triukšmo, medicininiams duomenims) apdoroti.

IV. IŠVADOS

1. Atlikus teorinės informacijos analizę buvo suformuluoti nauji fraktalinių interpoliacinių funkcijų praktinio pritaikymo metodai.

2. Išanalizavus fraktalų generavimo bei taikymo priemonių veiklos principus buvo nustatyti reikalavimai keliami fraktalinio interpoliavimo priemonei.

3. Eksperimentų rezultatai parodė, kad INTERPO—priemonė, kurios pagalba galima generuoti FIF įvairių tipų duomenims, jų analizei. Iškilusios problemos (vaizdumo, rezultatų išsaugojimo galimybių stoka) leidžia daryti išvadą, kad sėkmingesniam priemonės taikymui ją būtina atnaujinti, priderinant prie naujų kompiuterių techninių parametrų ir naudojamų OS (būtina pakeisti vartotojo terpę bei išplėsti duomenų sąrašą).

4. Teorinės informacijos apžvalga bei gauti eksperimentų rezultatai rodo, kad suformuoti interpoliavimo taškų masyvai galėtų būti naudojami ekstrapoliavimui (kreivės-fraktalo perspektyvos generavimui (prekybinės prognozės, ligos eigos prognozės, meteorologinės, biržų prekybos bei kt. prognozės)). Suformuoti duomenų masyvai leistų sutaupyti kompiuterio resursus – kreivės (fraktalo) aprašui būtų naudojami mažesni masyvai (pvz. vektorizuotas landšafto objektas, pateikiamas daugiau nei 1400 įrašų masyvu, gali būti nusakytas 100 ar bent 700 įrašų masyvu). Tarkime, gautos kreivės, nusakančios širdies ritmogramą, leistų stebėti tarpinius (šiaip nefiksuojamus duomenis) ir laiku diagnozuoti širdies ritmo sutrikimus, kurie susirgimo pradiniame etape šiaip jau būna gan reti ir ne visada pavyksta juos užfiksuoti įprastais metodais.

5. Pasiūlyta metodika analizuojamo pradinio objekto fraktalinės dimensijos fiksavimui bei FIF kreivės dimensijos priderinamumui.

6. Koliažo rankinis (interaktyvus) taikymas—imlus laiko sąnaudoms, nėra tikslus (taikant LIFS), todėl sėkmingas taikymas įmanomas tik automatizavus šį procesą.

V. LITERATŪRA

1. M. BARNSLEY, Fractals Everywhere, Second edition, Academic Press Professional, Cambridge 1993.
2. W. O. COCHRAN, J. C. HART PATRICK, J. FLYNN, On Approximating Rough CurvesWith Fractal Functions School of EECS Washington State University Pullman, WA
3. FractalVision: Put Fractals to Work For You Authors/ Dick Oliver: Publisher; 1992.— 485 p.
4. J. FEDER, Fractals Plenum Press New York 1988
5. N. KOSINOV, Fraktal protona—novy predstavitel v semeistve fraktalov.
6. N. KOSINOV, Protono ir deitrono fraktalai – mikropasaulio fraktalų atstovai)
7. B. B. MANDELBROT, The Fractal Gheometry of Nature. 1977
8. D. OLIVER, Fractal Vision, Sams Publishing (A division of Prentice Hall Computer Publishing, Carmel, Indiana). 1992.
9. W. PAULSON, Introduction to Fractals and IFS; 1998
10. H. PEITGEN, P. RICHTER, Krasota fraktalov. Obrazy kompleksnych dinamičeskich sistem.— M.:Mir, 1993.
11. H. PEITGEN, P. RICHTER, The Beauty of Fractals, Springer—Verlag, Berlin, New York. 1986.
12. P.TANNENBAUMAS, R.ARNOLDAS, Kelionės į šiuolaikinę matematiką; V.: TEV, 1995.
13. V. TARASENKO, Ossobennosti vvedenija poniatija fraktala.
14. J. VALANTINAS, Fraktalinė geometrija.— K.: Technologija, 1999.—p.10— 69,121— 153
15. B. VILJAMS, Trading Chaos.
16. M. MIRILOVIČ, Iterovanie funkcie komplexnej premennej. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://alife.tuke.sk/projekty/fracint/komplex.html >
17. Dvie fraktalnych zadački. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://arbuz.ferghana.ru/y_kv_f.html >
18. V. Orlov, A. Danilov Krasivaja žizn kompleksnich čisel; Ž. Hard'n'Soft №9 2002, 90 p. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://arbuz.ferghana.ru/x_complex.html >
19. A. Sundučkov Fraklaly, sintez izobraženij. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://cast.h1.ru/Articles/fract_and_synthes.shtml >
20. T. Stilson 3D Strange Attractors and Similar Objects.— [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://ccrma—www.stanford.edu/~stilti/images/chaotic_attractors/nav.html >
21. A. Šabaršin Vvedenije v fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://codenet.ru/progr/fract/fractr0.php#frac >
22. Paul Bourke Fractal Landscapes, How to create the IFS Fern — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://codenet.ru/progr/fract/fractr1.php>

23. B. Pobedria O fraktalach v mechanike. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://composite.msu.ru/win/s_work/1/fr.htm >
24. Fraktal (techninė analizė). — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://darkflame.dax.ru/?finances&t=ta&p=fractal>
25. Fraktalnij analiz bioritmov organizma.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://dyn.ru/products.php?id=1>
26. P. Bourke Fraktaly. 2002.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://dzenzel.boom.ru/fractals/index.htm >
27. Poisk grafičeskoj informaciji— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://elvisti.kiev.ua/skl/articles/pictures/pic.html>
28. Postrojenije fraktalov protona i deitrona; 2002 04 13; SciTecLibrary.ru. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://excluziv.fromru.com/stat/Cosinov/fractal—03.doc>
29. Surchalova, Sumina Rekursiji i fraktaly. Samara. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://fio.samara.ru/works/18/sarativ/index.htm >
30. Multifraktalnaja progulka bdol Uoll Strit, Multifraktaly i rynok.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <.http://forex—kiev.narod.ru/ta/multi_fr.html >
31. Fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <.http://forex—trader.by.ru/fractals.htm >
32. Dostupno o fraktalach. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fract.narod.ru>
33. O fraktalach i ich algoritmach.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fract.narod.ru/about.htm >
34. A. Beda Čto takoje fraktal? 2003. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractal.boom.ru/ >
35. Osnovnaja charakteristika fraktalov — samopodobije. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://fractals.chat.ru/index.htm >
36. J. P. Louvet Les Fractales.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractals.iut.u—bordeaux1.fr/jpl/jpl01.html >
37. Rol fraktalov v sovremennom mire.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractals.narod.ru/intro.htm>
38. A. Kravčenko, D. Mechoncev 3D Fractals.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractals.nsu.ru/ >
39. Vselennaja fraktalov. Statji o fraktalach. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractals.nsu.ru/fractals.chat.ru/index.htm >

40. J. P. Louvet Dimension fractale.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fractals.iut.u—bordeaux1.fr/jpl/dimension.html >
41. V. Kelasev I vnov fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fundmanager.bip.ru/Kelasev/8.htm >
42. Fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fxr2002.narod.ru/Fr/page4.html >
43. 8—Ispolzovanije fraktalov i ryčaga.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://fx—trader.narod.ru/Tradxaos8.htm >
44. Fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://graphics.cs.msu.su/courses/cg99/assigns/assign1/read_fr.htm >
45. Magija fraktalov.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://grs.narod.ru/litera/ell/fraktal/fraktal2.htm>
46. J. DeAlmo Applications of Fractal Geometry An Honors Project at the University of Rhode Island. 2002.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://hypatia.math.uri.edu/~kulenm/honprsp02/index.html >
47. J. DeAlmoFractal Interpolation. — [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://hypatia.math.uri.edu/~kulenm/honprsp02/interpolation.html >
48. Paul N. Lee Fractal Census. 2003.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://home.att.net/~Paul.N.Lee/Census_T.html >
49. A. Šabaršin Vvedenije vo fraktaly.— [žiūrėta 2003—10—15]. Prieiga per Internetą: <http://home.ural.ru/~shabun/fractals/fractals.htm>
50. E. Skliarevskij, M. Memetov Fraktaly, biomorfy. — [žiūrėta 2003—10—15]. Prieiga per Internetą: < http://home.ural.ru/~shabun/fractals/index.htm >
51. Hard Wisdom Čisla kak fraktaly. 1998 11 24. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://hw.cracks.ru/articles/fractals_rus.html >
52. V. Syčev Teoretičeskaja osnova.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://impb.psn.ru/~sychyov/soft.shtml#models >
53. A Multifractal Walk Down Wall Street. 1999. Scientific American. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://investo.ru/digest/fractal.html >
54. Sposoby postrojenija fraktalov.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://janetta.nm.ru/theory.html >
55. Čto takoje fraktal? — [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://karev.narod.ru/fraktal.htm >
56. Fraktalų galerijos.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://www.lifesmith.com/gallery.html >

57. L. Vepstas Renormalizing the Mandelbrot Escape (A simplified presentation for the Fractal Art FAQ). 1997. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://linas.org/art—gallery/escape/escape.html>
58. L. Vepstas Smooth Escape Iteration Counts (Derivation of same results via Spectral Analysis). 1997. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://linas.org/art—gallery/escape/math.html >
59. L. Vepstas Douady—Hubbard Potential. 2000. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://linas.org/art—gallery/escape/ray.html>
60. A. Tišin Fraktaly i cikly v procese poznanija. Biškek. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://lpur.tsu.ru/Public/a0101/a010201.shtm >
61. R. Suchomlinov, Birobidžan Fraktaly i cikly v evoliuciji živych sistem. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://lpur.tsu.ru/Public/a0101/a011801.shtm >
62. FRACTALINA —Created by Noah Goodmann, and modified by Adrian Vajiac — For comments and suggestions write to Robert L. Devaney .— [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://math.bu.edu/DYSYS/applets/fractalina.html>
63. Cynthia Lanius — The Anti—Snowflake.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://math.rice.edu/~lanius/frac/anti.html>
64. Cynthia Lanius — The Sierpinski Triangle.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://math.rice.edu/~lanius/fractals/ >
65. © 1999 CRC Press LLC, © 1999—2004 Wolfram Research, — [žiūrėta 2004—03—18]. Prieiga per Internetą:< http://mathworld.wolfram.com/Fractal.html >
66. Fractal — Eric W. Weisstein. "Fractal." From *MathWorld*——A Wolfram Web Resource.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <.http://mathworld.wolfram.com/Fractal.html >
67. The Mandelbrot and Julia sets Anatomy.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://mitpress.mit.edu/books/FLAOH/cbnhtml/ >
68. Poniatije fraktala.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://multifractals.narod.ru/RF/IntroRF.htm >
69. A. Dmitrijev Chaos, fraktaly i informacija. Nauka i žizn. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://nauka.relis.ru/cgi/nauka.pl?07+0105+07105044+HTML >
70. V. Tarasenko Osobennosti vvedenija poniatija fraktala. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://neuromania.narod.ru/fracdef.html >
71. Fraktalnyje magnity izmeniat sovremennoje predstavlenije o vysokych technologijach. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://news.finestreet.ru/cgi—bin/news/view.cgi?id=751&cat_id=1&page=1>

72. N. Meščaninov Fraktaly, čast 2. 2001 01.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://nsft.narod.ru/Fractals/fract2.htm >
73. O. Voronin Fraktaly. Ukraina, g. Chmelnickij.— [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://oco.newmail.ru/fractals.htm >
74. Geometrija prirody. Fraktaly. — [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://opensystems.ru/pcworld/2000/09/014.htm >
75. Udivitelnyj mir fraktalov. 2003 No16. — [žiūrėta 2004—03—18]. Prieiga per Internetą: <geekinfo.nethttp://pi.zen.ru/arhiv/2003/016/fractals.shtml >
76. Algoritm fraktalnovo sžatija realnych izobraženij; Osnovnyje poniatija i obščeje opisanije algoritma.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://public.tsu.ru/~shab/fcomp/3—2.htm>
77. Litow 3D fraktal. 2000 08 07.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://resource.h1.ru/cgi—bin/resource.pl?do=open&what=/3/0/0.h >
78. J. Danilov Krasota fraktalov. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://re—tech.narod.ru/inf/sinergy/fractal_b.htm>
79. A. Šabaršin Mir fraktalov. 1996—1999 Rossija. Ekaterinburg. — [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://robots.ural.net/fractals/ >
80. Fraktaly.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://rusproject.narod.ru/article/fractals.htm >
81. B. Rassel Fraktaly vokrug nas.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://sakva.narod.ru/fractals.htm#History >
82. Pismo D. Samoilovu o fraktalach.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://saturn.spaceports.com/~speakrus/f014.htm >
83. Fraktalai.— [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://siauliai.mok.lt/aurelijus/html/meniu.htm >
84. J. Danilov, Kurčatov Krasota fraktalov. — [žiūrėta 2004—03—18]. Prieiga per Internetą: <http://sky.kuban.ru/socio_etno/iphrRAS/~mifs/danilov.htm >
85. Fractals— [žiūrėta 2004—03—18]. Prieiga per Internetą: < http://slovesnov.narod.ru/fractals/>
86. J. C. Sprott Sprott's Software.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://sprott.physics.wisc.edu/software.htm >
87. E. Arinštein Material seminarov po fraktalam. Tiumen. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://stratum.pstu.ac.ru/lectures/Kgrafic/html/DOPOLNIT/cterra/c40.htm>
88. A. Šabaldin Sžatije izobraženij, osnovannoje na teoriji fraktalov. Tomsk. — [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://subscribe.ru/archive/inet.webbuild.yahoo/200012/23145532.text>

89. D. Šepovalov Fraktaly.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://titbit.narod.ru/tech.htm >
90. Fraktaly.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://tmn.fio.ru/works/02x/306/fractals/various.html>
91. M. Čekulajev Magija fraktalov. Valiutny spekuliant. 2000 No8. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://traders.kiev.ua/ta/magia_fr.html>
92. Fraktaly. Poliot mysly.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://untitled.agava.ru/thinks/fractal/addon.shtml>
93. O fraktalach živych i razumnych.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://untitled.agava.ru/thinks/fractal/index_fr.shtml>
94. Ilja „Voyager"Fraktaly. Poliot mysly(1). — [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://untitled.agava.ru/thinks/fractal/index_fr.shtml >
95. I. Andrijanov Kto že otkryl fraktal Mandelbrota? VIVOS VOCO! 1997 No 10.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <.http://vivovoco.rsl.ru/VV/PAPERS/NATURE/MANDEL.HTM>
96. Geometrija prirody fraktalna.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.314159.ru/kosinov/kosinov1.htm >
97. G. Rizničenko Populiacionnaja dinamika.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.625—net.ru/archive/0497/worldbuilder.htm >
98. I. Borovikov World Builder – strojitel mirov. 625—net. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.625—net.ru/archive/0497/worldbuilder.htm >
99. Fraktaly.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.aip.mk.ua/cplusp/all_96/6n96y/6n96y1a.htm >
100. Fractalvision: Put Fractals to Work for You/Book and Disk by Dick Oliver— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.amazon.com/exec/obidos/ASIN/0672302489/categoricalgeome>
101. Programirovanije— Fraktaly.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.arbuz.narod.ru/ >
102. Ellementary Functions at a Glance (Use Tools for Graphic Design). .— [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.artlandia.com/products/artlandia/at_a_glance/ >
103. Eric W. WeissteinFractals. 2003.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.astro.virginia.edu/~eww6n/books/Fractals.html>
104. Modelirovanije fraktalov. / elektronnyj učebnik/ — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.bsu.ru/library/berson/index.html >

105. Gražūs fraktalai.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.baco.nu/fractais/ >

106. O fraktalach.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.chat.ru/~fractals/math.htm >

107. H. Peitgen, R. Voss NSF Pattern Exploration Directions and Software. Florida. 2000.— [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.cevis.uni—bremen.de/fractals/>

108. Fraktalai: pradžiamokslis. Gyvos Lisažu figūros — virtualus osciloskopas. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.best.com/~ejad/java/fractals/ >

109. Krasota fraktalov.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.bsu.ru/library/berson/vvedenie.html >

110. Paul Bourke Fraktalnyje množestva (Delfi). — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.codenet.ru/progr/fract/fdelphi.php >

111. Primenenije fraktalov pri postrojeniji mirov.— [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.codenet.ru/progr/fract/fractr0.php >

112. Primenenije fraktalnych metodov dlia obrabotki radiolokacionnych izobraženij. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.cplire.ru/joined/win/lection6/ >

113. Opisanije raboty program Fract 16 i Fract 256.— [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.davidov.ru/mirea.i—am.ru/study/fract.html >

114. Fraktaly i vozroždenije teoriji iteracij Benua Mandelbrota. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.delphaq.newmail.ru/Base/Graphics/fractal.htm >

115. Kakaja polza ot fraktalov? 2002 12 01.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.dig.ru/dw/dw_3857.htm >

116. Poniatije atraktorov, Fraktalnaja geometrija.— [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.d—trade.ru/?a=17&chap=32>

117. Supplemental Information – Martin/Eal – Engineering Arts Laboratory. — [žiūrėta 2004—04—05]. Prieiga per Internetą: < http://www.ealnet.com/ealsoft/fractal >

118. Programy risovanija fraktalov. — [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.eclectasy.com/fractovia/ >

119. Terry's Fractal Mirage.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.eclectasy.com/twg/ >

120. A. Našlėnienė, A. Našlėnas Fraktalai (Vaizdų albumas; Šiek tiek istorijos; Fraktalai realiame pasaulyje; Fraktalinė dimensija bei kt.). — [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.elnet.lt/vartiklis/fractals/ >

121. Fraktalai.— [žiūrėta 2004—04—05]. Prieiga per Internetą: <http://www.elnet.lt/vartiklis/fractals/album.htm >

122. P. Tisnovský 1 Fraktály. —Nelineární dynamické systémy. 2003.— [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.fit.vutbr.cz/~tisnovpa/fractals.html >

123. A. Morozov Vvedenije v teoriju fraktalov. — [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.fizmatkniga.ru/product_info.php?products_id=1414&PHPSESSID >

124. E. Mogilevskij Fraktaly na solnce. —[žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.fizmatlit.ru/index.shtml?section=1202&&article=1251 >

125. IFS, Linenyj fraktal.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.fract.narod.ru/about.htm >

126. Fraktaly v akustikie. —[žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.fractal.ru/FractalAcoustics_ru.htm >

127. Fraktalnyje galereji.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.fractalus.com/galleries/ >

128. Fractals and other mysteries. 2004.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.geocities.com/SoHo/Lofts/5601/ >

129. Mir fraktalov.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.geocities.com/SoHo/Studios/6648/fractalgallery.htm >

130. I. Tugoj Fraktaly i teorija chaosa.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.ghcube.com/fractals >

131. A. Berliant Kartografičeskaja generalizacija i teorija fraktalov. M. 1998.— [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.gisa.ru/406.html?action=print >

132. H. Joyce, A. Beardon Some Fractal History (Pirmasis įvadas į fraktalus — Straipsnio A First Introduction to Fractals vertimas). — [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.goshen.edu/~kevin/fractint/history.html >

133. A. Belko Analiz generaciji fraktalnych struktur polučennych metodom Vittena—Sandera. Grodno.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.grsu.by/~ftf/data/nir/belko/doclad.html >

134. Fractals (Introduction; My very first Fractals). — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.harald—zappe.de/ >

135. Special Fractals.— [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.harald—zappe.de/fractspec/specials.html >

136. S. Krylov 2.13 Fraktalnaja geometrija prirody. — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.icape.ru/condens/kurs/kurs2_13.htm >

137. Fraktalinio scenovaizdžio kūrimo algoritmai. — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.ik.ku.lt/lessons/konspekt/graphics/dvyl.htm >

138. Programa postrojenija fraktalov. —[žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.informika.ru/text/inftech/edu/edujava/mathematics/Chaos/>
139. Nemnogo o fraktalach.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.ioffe.org/register/?doc=physica3/fractal.tex >
140. Krasota fraktalov.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.iph.ras.ru:8101/~mifs/rus/danilov.htm >
141. 19. Rekursiji i fraktaly. — [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.karelia.ru/psu/Chairs/IMO/pascal/theory/part2_12.html >
142. Julija Set.— [žiūrėta 2004—05—05]. Prieiga per Internetą: <http://www.kv.by/index1997192001.htm>
143. Fraktalnyj paporotnik i affinnyje preobrazovanija. — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.kv.minsk.by/index2002491201.htm >
144. N. Kosinov Fraktal protona — novyj predstavitel v semje fraktalov. — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.laboratory.ru/articl/hypo/ax160r.htm >
145. Andy Burbanks Iterated Function Systems. — [žiūrėta 2004—05—05]. Prieiga per Internetą: < http://www.lboro.ac.uk/departments/ma/gallery/ifs/index.html>
146. Andy Burbanks Quaternionic Fractals. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.lboro.ac.uk/departments/ma/gallery/quat/intro.html >
147. Pan TWG's Gallery. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.legendsmagazine.net/pan/rayn/twg/>
148. Fraktaly.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.looking.ru/video/?sgid=19 >
149. Andy BurbanksSelf—referential Sentences. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.lut.ac.uk/departments/ma/gallery/selfref/first >
150. Self-referential Sentences Produced by Andy Burbanks — [žiūrėta 2004—05—11]. Prieiga per Internetą: < self—documenting pair http://www.lut.ac.uk/departments/ma/gallery/selfref/pair >
151. M. Urbanski Fractals in the Plane — the Ergodic Theory Methods by F. Przytycki and M. Urbanski. Texas. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.math.unt.edu/~urbanski/book1.html >
152. S. Hinsley Definition of Polymer, P—System, Metapolymer and Pseudopolymer, etc. 2000, 2001.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.meden.demon.co.uk/Fractals/polymer.html >
153. S. Hinsley pseudo—terdragon. 2002. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.meden.demon.co.uk/Fractals/pterdragon.html >

154. Techniques for designing IFSs. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.meden.demon.co.uk/Fractals/Techniques/affine.html >
155. MetaStock , Analitika+.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.metastock.ru/>
156. Fractal Galleries.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.mysticfractal.com/galleries.html >
157. Fractal Links.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.mysticfractal.com/links.htm >
158. Fractal Programs. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.mysticfractal.com/programs.html>
159. Ssylki na samyje fraktalnyje saity ruskovo interneta .— [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.mosreg.fio.ru/potoky/potok2/pr13/fractal.htm >
160. N. Petrova Malenkije modeli bolšovo mira. 1997. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.museum.ru/museum/cga/education/dendrites/index.html >
161. Fraktaly— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.nsft.narod.ru/Fractals/fract.htm>
162. Vselennaja fraktalov, O promeneniji fraktalov. Nižnevartovsk. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.nsu.ru/fractals/fractals.chat.ru/appl.htm>
163. Big list of links to fractal pages.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.nsu.ru/fractals/fractals.chat.ru/galaxy.htm >
164. Links to fractal galleries.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.nsu.ru/fractals/fractals.chat.ru/gall >
165. Nepreryvnyje fraktalnyje krivyje. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.nsu.ru/fractals/fractals.chat.ru/zippers2.htm>
166. J. Černov Geometričeski samopodobnyje fraktaly. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.nsu.ru/mmf/tvims/chernova/fract/fr.html >
167. E. Demido A virtual investigation with interactive pictures (Part 1. The Mandelbrot and Julia sets Anatomy Contents). V. 2003. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.people.nnov.ru/fractal/MSet/Contents.htm >
168. Paitgen Ch., Richter P. Krasota fraktalov— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.philsci.univ.kiev.ua/biblio/Pajt.html>
169. N. Dolbilin Čto takoje fraktal? — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.pms.ru/learning/seminars/est04.html >
170. For more on interpolating sin() and cos().— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.rdrop.com/~cary/html/arithmetic >

171. VLSI: designing electronic chips. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.rdrop.com/~cary/html/arithmetic >

172. R. Kronover Fraktaly i chaos. M. 2000.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.rechka.ru/fract/fractals.html >

173. Fraktaly iz knigi Džef Prouza „Kak rabotajet kompiuternaja grafika“. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.sch93.chel.su/Users/Chekanova/frabout.html >

174. S. Miler, A. Diudni Zanimatelnyj kompiuter. Scientific American. 1988 No7. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.sciam.com/>

175. N. Kosinov Fraktaly protona — predstaviteli fraktalov mikromira. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.sciteclibrary.ru/rus/catalog/pages/3459.html >

176. A. Šabanin Mir fraktalov. 1996—2003. Rosija. E. — [žiūrėta 2004—05—11]. Prieiga per Internetą: < http://www.shaos.ru/fractals/ >

177. A. Šabaršin Vvedenije vo fraktaly. — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.shaos.ru/fractals/intro/fractals.htm >

178. Fraktalnaja logika. M. IFRAN 1997.— [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.synergetic.ru/sections/fractal/ >

179. Fraktalai. (Apibrėžimas ir savybės). — [žiūrėta 2004—05—11]. Prieiga per Internetą: <http://www.spauda.lt/fractals/apibr.htm >

180. Fraktalai. (Fraktalinė dimensija). — [žiūrėta 2004—05—11]. Prieiga per Internetą: <.http://www.spauda.lt/fractals/dimens.htm >

181. Fraktalai (Fraktalų istorijos epizodai). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <.http://www.spauda.lt/fractals/istor.htm >

182. Fraktalai. (Kai kurių fraktalų konstravimas Koch snaigė). — [žiūrėta 2004—05—12]. Prieiga per Internetą: < http://www.spauda.lt/fractals/konstr.htm >

183. Fraktalai. (Vaizdų albumas). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.spauda.lt/fractals/virsus.htm >

184. Magija fraktalov, ili naskolko prav Bill Uiljams. Valiutnyj spekuliant. 2000 No8 (10). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.spekulant.ru/view/index.html?nid=10&tid=973790553 >

185. *Iterated.com*.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.stingray.nm.ru/Download/Studies/ImgProcs/Fractals.rar >

186. Mark McClure Mathematica Graphics. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.unca.edu/~mcmcclur/mathematicaGraphics/index.html >

187. Mark McClure Directed—Graph Iterated Function Systems. Asheville. — [žiūrėta 2004—05—12]. Prieiga per Internetą: < http://www.unca.edu/~mcmcclur/professional/DigraphFractals/ >

188. A. Vlasova Fraktaly. — [žiūrėta 2004—05—12]. Prieiga per Internetą: < http://www.uni—dubna.ru/kafedr/mazny/page11.htm >

189. B. Viljams Torgovyj chaos. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.universalinternetlibrary.ru/book/Vilyams/000.shtml >

190. Fraktaly, sintez izobraženij.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.visti.net/cplusp/all_96/6n96y/6n96y1a.htm >

191. V. Tarasenko Varijaciji na temy Maršala Makluena, Timoti Liri i Benua Mandelbrota. 1997.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <http://www.zhurnal.ru/nepogoda/variacii.htm>

192. Programy dlia modelirovanija L—fraktalov, napisannyje na Q—basic i na C. — [žiūrėta 2004—05—12]. Prieiga per Internetą: < inf.1september.ru/2000/6/fract/main.htm >

193. Vselennaja fraktalov.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.chat.ru/~fractals>

194. Mokomosios programos kompaktinėse plokštelėse: matematika (... Fraktalai. Pristatomi įvairiausi fraktalai (pateikiama daug straipsnių ir animacijos). Bibliografija.). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.emokykla.lt/mokymas/mokymopr/cd/matm.htm>

195. Kompiuterinė Grafika.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.ik.ku.lt/lessons/konspekt/graphics/dalis11.htm >

196. Komunikacinė (postmodernioji) meno samprata.— [žiūrėta 2004—05—12]. Prieiga per Internetą: < www.infovi.vu.lt/postmenas.html >

197. Mathematica taikymo pavyzdžiai.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.itpa.lt/mathematica/usageexlt.html >

198. Editorial.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.maf.vu.lt/ljmm/nrich/editor.htm >

199. G. Bareikis Fraktalai. VU MF. 2002.— [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.maf.vu.lt/ttsk/bylos/bg/bg.html >

200. 7.1 Parametrinės aibės žemėlapis.— [žiūrėta 2004—05—12]. Prieiga per Internetą:<www.maf.vu.lt/ttsk/bylos/bg/pas/a7fr.pdf >

201. A. Kolesnikov Fraktaly. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.pilabs.org.ua/wisdom/articles/fractals_rus.html >

202. A. Bogomolnyj zanimatelnyje statji po vsem pazdielam matematiki s java—appletami. Ajova.— [žiūrėta 2004—05—12]. Prieiga per Internetą: < www.rechka.ru >

203. Fraktalai. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/apibr.htm>

204. Fraktalai. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/desine.htm >

205. Fraktalai. (Mandelbroto aibė. Mandel4 fraktalas. Mandellambda fraktalas. Lambda fraktalas. Barnsleym3 fraktalas. Julia4 fraktalas. Kompleksmarksjul fraktalas.).— [žiūrėta 2004—05—12]. Prieiga per Internetą: < www.spauda.lt/fractals/kaire.htm >

206. Fraktalai. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/kasyra.htm >

207. Fraktalai. — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/man05.htm >

208. Fraktalai (Paskalio programa Mandelbroto aibei gauti). — [žiūrėta 2004—05—12]. Prieiga per Internetą: < www.spauda.lt/fractals/paskal.htm >

209. Fraktalai. (Butterfly fraktalas). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/butter.htm >

210. Fraktalai (Kompleksmarksjul fraktalas). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/cmjul.htm>

211. Fraktalai (Kompleksbasin fraktalas). — [žiūrėta 2004—05—12]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/cobasin.htm>

212. Fraktalai. (Eye fraktalas). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/eye.htm >

213. Fraktalai (Mandelcloud fraktalas). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/mcloud.htm >

214. Fraktalai (Mandel4 fraktalas.). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/mandel4.htm >

215. Fraktalai (Papartis). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/fern.htm >

216. Fraktalai. (Sierpinskio nėrinys). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/sierp.htm >

217. Fraktalai. (Kompleksniuton fraktalas). — [žiūrėta 2004—05—20]. Prieiga per Internetą: <www.spauda.lt/fractals/vaizdai/conewton.htm >

PRIEDAS 1

INTERPO ALGORITMAS C KALBA:

```
/* INTERPO.C
 *
 * Ši programa skirta fraktalinės interpoliacinės funkcijos (FIF)
 * kreivės generavimui, kai duomenys skaitomi iš ASCII failo
 * duomenų failo pavyzdys: failas SAMPLE.FIP
 *
 */
#include <stdio.h>      /* Įvedimo/išvedimo standartas  */
#include <stdlib.h>
#include <string.h>     /*  strcpy */
#include <math.h>       /* mat f-jos */
#define MAIN            /* Pagrindinė. Dydžio D skaičiavimui */
#include "fv.h"         /* Instrukcija */
#define drawclr 7       /* Kreivės linijos spalva */
#define pointclr 11     /* duotų taškų spalva */
#define gridclr 2       /* grafo linijos spalva */
#define NPOINTS 100     /* Maksimalus duomenų kiekis */
float   x[NPOINTS], y[NPOINTS],     /* duomenų failo duomenys */
        scale[NPOINTS],             /* y mastelio keitimo parametras */
        a[NPOINTS - 1], b, c[NPOINTS - 1],     /* IFS kodas */
        e[NPOINTS - 1], f[NPOINTS - 1];
void main(int nargs, char **arg)     /* komandinės eilutės paruošimas*/
{   int vidmode, i, j;          /* monitoriaus tipo nurodymas */
    float x1, y1, x2, y2,         /* brėžti kreivei */
          xscale, yscale,     /* monitoriaus taškų skalė */
          maxx, maxy, minx, miny,    /* max ir min x bei y */
          dim, sum, ratio,     /* dimensijos skaičiavimui */
          xgrid, ygrid;    /* grafo formavimas pagal įvestus duomenis */
    int xoffset, yoffset, yadjust,    /* ekrano pozicionavimas */
        npoints;          /* duomenų taško eilė duomenų faile */
    FILE *diskfile;           /* failo vardui */
    char filename[20];
```

```
vidmode = pickmode(detectmode()); /* video moduliui */
if (nargs > 1) strcpy(filename, arg[1]);
else     /* jei failo vardas kom eil nenurodytas */
{   printf("\n\n");
    printf("Programa skirta fraktalinei kreivei generuoti"
           " pagal duomenis\n");
    printf("iš tekstinio failo: taško x,y.  Duomenų įvedimo "
           "pavyzdys yra faile SAMPLE.FIP.");
    printf("\n\nĮveskite duomenų tipo pavadinimą: ");
    scanf("%s", filename);
}
if (nargs > 2) dim = atof(arg[2]);
else    /* jei dimensija nenurodyta */
{   printf("\n\nPasirinkite fraktalinės dimensijos "
           "reikšmę (iš intervalo [1;2]: ");
    scanf("%f", &dim);
}
if ((dim > 2.0) || (dim < 1.0))   /* dimensija nurodyta tinkamai? */
{   printf("Fraktalinė dimensija-skaičius  "
           "iš intervalo [1;2] (pvz. 1.2).");
    exit(0);
}
if ((diskfile = fopen(filename, "r")) == NULL)
{   printf("Klaidingai nurodėte failo vardą %s.", filename);
    exit(0);
}
fscanf(diskfile, "%d", &npoints);          /* duomenų nuskaitymas */
fscanf(diskfile, "%f %f", &xgrid, &ygrid);
for (i = 0; (i < npoints) && (i < NPOINTS); i++)
   fscanf(diskfile, "%f %f %f", x + i, y + i, scale + i);
fclose(diskfile);           /* kol nėra klaidų!! */
if (setvidmode(vidmode, 1) != 1)
{   printf("\nPakeiskite monitoriaus tipą.\n");
    exit(0);
}
```

```
/* vaizdo formavimas */
yadjust = scrny / 20;          /* tinklelis */
maxx = x[0]; minx = x[0];
maxy = y[0]; miny = y[0];
for (i = 1; i < npoints; i++)  /* plačiausia/siauriausia */
{   if (x[i] > maxx) maxx = x[i];
    if (y[i] > maxy) maxy = y[i];
    if (x[i] < minx) minx = x[i];
    if (y[i] < miny) miny = y[i];
}
xscale = scrnx / (maxx - minx);     /* skaič. x, y */
yscale = (scrny / (maxy - miny)) / 2;
xoffset = - (int) (minx * xscale);   /* perkėlimas */
yoffset = scrny - yadjust + (int) (miny * yscale);
/* Skaičiuojami transformacijų
   koeficientai ir vertik. mastelį keič. parametro suma */
b = x[npoints - 1] - x[0]; /* b reikšmės */
sum = 0.0;                 /* v.m.k. parametro sum  */
for (i = 0; i < npoints - 1; i++) sum += fabs(scale[i]);
ratio = pow(npoints - 1, dim - 1) / sum;
for (i = 0; i < npoints - 1; i++)  /* nustatomas IFS kodas */
{   scale[i] *= ratio;
    a[i] = (x[i + 1] - x[i]) / b;
    e[i] = (x[npoints - 1] * x[i] - x[0] * x[i + 1]) / b;
    c[i] = (y[i + 1] - y[i] - scale[i] *
            (y[npoints - 1] - y[0])) / b;
    f[i] = (x[npoints - 1] * y[i] -
            x[0] * y[i + 1] -
            scale[i] *
            (x[npoints - 1] * y[0] -
            x[0] * y[npoints - 1])) / b;
}

/* Rezultatų spausdinimas */
printf("%s  %d points, "
```

```
      "graph scale = %6.2f x %6.2f  Dimension = %f",
      filename, npoints, xgrid, ygrid, dim);
printf("\nx =");
for (i = 0; i < npoints; i++) printf(" %6.2f", x[i]);
printf("\ny =");
for (i = 0; i < npoints; i++) printf(" %6.2f", y[i]);
printf("\ns =");
for (i = 0; i < npoints - 1; i++) printf(" %6.2f",
                                scale[i]);
printf("\na =");
for (i = 0; i < npoints - 1; i++) printf(" %6.2f", a[i]);
printf("\nc =");
for (i = 0; i < npoints - 1; i++) printf(" %6.2f", c[i]);
printf("\ne =");
for (i = 0; i < npoints - 1; i++) printf(" %6.2f", e[i]);
printf("\nf =");
for (i = 0; i < npoints - 1; i++) printf(" %6.2f", f[i]);
      getch();
settextmode();
      /* brėžiama kreivė */
setcolor(gridclr);
xgrid = xgrid * xscale;  ygrid *= yscale;
for (x1 = (float) xoffset; x1 < scrnx; x1 += xgrid)
   drawline((int) x1, scrny / 2 - yadjust * 2,
           (int) x1, scrny - 1);
for (y1 = (float) yoffset;
    y1 >= (float) scrny / 2 - yadjust * 2;
    y1 -= ygrid)
   drawline(0, (int) y1, scrnx - 1, (int) y1);
/* tinklelis */
for (i = 0; i < npoints; i++)
{   x1 = x[i] * xscale + xoffset;
   y1 = yoffset - y[i] * yscale;
   fillrect((int) x1 - 2, (int) y1 - 2,
           (int) x1 + 2, (int) y1 + 2, pointclr);
```

```
    }
    /* kreivė pateikiama ekrane*/
    x1 = (float) xoffset;              /* pradžia */
    y1 = (float) yoffset;
    while(!kbhit())            /* ciklas */
    {   j = rand() % (npoints - 1);       /* chaosas */
        x2 = a[j] * x1 + e[j];      /* atraktorius */
        y2 = c[j] * x1 + scale[j] * y1 + f[j];
        x1 = x2, y1 = y2;
        putpixel((int) (x1 * xscale + xoffset),
                (int) (yoffset - y1 * yscale), drawclr);
    }
    /* pabaiga */
    getch();
    settextmode();
    closedown();
}
```

PRIEDAS 2

FRACDIM TAIKYMO PAVYZDŽIAI

Priemonė kreipiasi į Interpo; Interpo apskaičiuoja afiniųjų transformacijų koeficientų reikšmes ir per duotus interpoliavimo taškus brėžia FIF kreivę .

Pirmasis pavyzdys, kai dimensijos pateikiamos iš tam tikro intervalo:

```
***************
@echo off
if exist %1 goto start
echo  sis bat tipo failas papildo INTERPO.EXE programa
echo ir leidzia stebeti FIF kreives priklausomybe
echo nuo pasirinktu D reiksmiu intervale [1;2]
echo.
echo FRACDIM DATAPTS.FIP
echo.
goto end
:start
interpo %1 1.0
interpo %1 1.1
interpo %1 1.2
interpo %1 1.3
interpo %1 1.4
interpo %1 1.5
interpo %1 1.6
interpo %1 1.7
interpo %1 1.8
interpo %1 1.9
:end
```

Antrasis pavyzdys, kai dimensijos nurodomos laisvai (atsitiktine tvarka) Fracdim1:

```
@echo off
if exist %1 goto start
echo  sis bat tipo failas papildo INTERPO.EXE programa
echo ir leidzia stebeti FIF kreives priklausomybe
echo nuo pasirinktu D reiksmiu
echo.
echo FRACDIM DATAPTS.FIP
echo.
goto end
:start
interpo %1 1.001
interpo %1 1.010
interpo %1 1.011
interpo %1 1.012
interpo %1 1.013
interpo %1 1.014
interpo %1 1.015
interpo %1 1.110
interpo %1 1.111
interpo %1 1.112
interpo %1 1.115
interpo %1 1.105
interpo %1 1.117
interpo %1 1.120
interpo %1 1.131
interpo %1 1.141
interpo %1 1.151
interpo %1 1.210
interpo %1 1.311
interpo %1 1.412
:end
```

PRIEDAS 3

GREITA FURJE PAIEŠKA

Pagrindiniai principai

Tiesioginė ir atvirkštinė Furje paieškos gali būti užduotos sekančiai:

$$H_n \equiv \sum_{k=0}^{N-1} h_k \exp\left(\frac{kn}{N} 2\pi i\right) \qquad n \in [-\frac{N}{2}, \frac{N}{2}]$$

$$H_n \equiv \sum_{k=0}^{N-1} h_k \exp\left(\frac{kn}{N} 2\pi i\right) \qquad n \in [-\frac{N}{2}, \frac{N}{2}]$$

Kadangi formulės skiriasi tik daugikliu ir eksponentės ženklu, tai galima taikyti vieną ir tą patį algoritmą abiem paieškoms. GFP pagrindinė idėja yra ta, kad lyginiam *N* galima išskirti FP viso masyvo padalytą iš paieškos elementų su lyginiais ir elementų su nelyginiais numeriais sumos:

$$H_n \equiv \sum_{k=0}^{N-1} h_k \exp\left(\frac{kn}{N} 2\pi i\right) \qquad n \in [-\frac{N}{2}, \frac{N}{2}]$$

$$H_n \equiv \sum_{k=0}^{N-1} h_k \exp\left(\frac{kn}{N} 2\pi i\right) \qquad n \in [-\frac{N}{2}, \frac{N}{2}]$$

Jei *N* dalus iš 4, galima taikyti rekursiją. Jei *N* yra dvejeto laipsnis, tai skaičiavimai susiveda į vieno skaičiaus DFP. (bit-reversing – elementų perstatymui masyve)

FFT ALGORITMAS C KALBA:

```
/*
  FFT skaiciavimo algoritmas C++
*/
void FFT(int dir, long m, complex <double> x[])
{
  long i, i1, i2,j, k, l, l1, l2, n;
  complex <double> tx, t1, u, c;

  /*Nustatomas tasko nr */
  n = 1;
  for(i = 0; i < m; i++)
    n <<= 1;

  /* bito anuliavimas */
  i2 = n >> 1;
  j = 0;
  for (i = 0; i < n-1 ; i++)
  {
    if (i < j)
      swap(x[i], x[j]);
    k = i2;
    while (k <= j)
          {
      j -= k;
      k >>= 1;
    }
    j += k;
  }

  /* skaiciuojama FFT */
  c.real(-1.0);
  c.imag(0.0);
  l2 = 1;
  for (l = 0; l < m; l++)
```

```
  {
    l1 = l2;
    l2 <<= 1;
    u.real(1.0);
    u.imag(0.0);
    for (j = 0; j < l1; j++)
          {
      for (i = j; i < n; i += l2)
                  {
        i1 = i + l1;
        t1 = u * x[i1];
        x[i1] = x[i] - t1;
        x[i] += t1;
      }
      u = u * c;
    }
    c.imag(sqrt((1.0 - c.real()) / 2.0));
    if (dir == 1)
      c.imag(-c.imag());
    c.real(sqrt((1.0 + c.real()) / 2.0));
  }
  /* transform pirmyn skaiciavimas */
  if (dir == 1)
  {
    for (i = 0; i < n; i++)
      x[i] /= n;
  }
  return;
}
```

FFT algoritmas C kalba2:

Aprašas:

nn-f-jos reikšmių kiekis;

a –masyvas (0; 2*nn-1)

InverseFFT – False-tiesioginė; True-atvirkštinė GFP;

Naujas **a** – pertvarkos rezultatas;

```
/**********************************************
FFT skaiciavimas2
**********************************************/
#include "ap.h"
void fastfouriertransform(ap::real_1d_array& a, int nn, bool inversefft);
/*****************************************************************
FFT
*****************************************************************/
void fastfouriertransform(ap::real_1d_array& a, int nn, bool inversefft)
{
   int ii;
   int jj;
   int n;
   int mmax;
   int m;
   int j;
   int istep;
   int i;
   int isign;
   double wtemp;
   double wr;
   double wpr;
   double wpi;
   double wi;
   double theta;
   double tempr;
   double tempi;

   if( inversefft )
```

```
{
    isign = -1;
}
else
{
    isign = 1;
}
n = 2*nn;
j = 1;
for(ii = 1; ii <= nn; ii++)
{
    i = 2*ii-1;
    if( j>i )
    {
        tempr = a(j-1);
        tempi = a(j);
        a(j-1) = a(i-1);
        a(j) = a(i);
        a(i-1) = tempr;
        a(i) = tempi;
    }
    m = n/2;
    while(m>=2&&j>m)
    {
        j = j-m;
        m = m/2;
    }
    j = j+m;
}
mmax = 2;
while(n>mmax)
{
    istep = 2*mmax;
    theta = double(2*ap::pi())/double(isign*mmax);
    wpr = -2.0*ap::sqr(sin(0.5*theta));
```

```
        wpi = sin(theta);
        wr = 1.0;
        wi = 0.0;
        for(ii = 1; ii <= mmax/2; ii++)
        {
            m = 2*ii-1;
            for(jj = 0; jj <= (n-m)/istep; jj++)
            {
                i = m+jj*istep;
                j = i+mmax;
                tempr = wr*a(j-1)-wi*a(j);
                tempi = wr*a(j)+wi*a(j-1);
                a(j-1) = a(i-1)-tempr;
                a(j) = a(i)-tempi;
                a(i-1) = a(i-1)+tempr;
                a(i) = a(i)+tempi;
            }
            wtemp = wr;
            wr = wr*wpr-wi*wpi+wr;
            wi = wi*wpr+wtemp*wpi+wi;
        }
        mmax = istep;
    }
    if( inversefft )
    {
        for(i = 1; i <= 2*nn; i++)
        {
            a(i-1) = double(a(i-1))/double(nn);
        }
    }
}
```

PRIEDAS 4

VB INTERPO ALGORITMAS VISUAL BASIC

Duomenų masyve pateikiamos analizuojamo objekto interpoliavimo taškų koordinatės. Nuskaičius duomenis, yra suformuojami trys masyvai: $X(100)$ (interpoliavimo taškų x koordinačių reikšmių masyvas), $Y(100)$ (interpoliavimo taškų y koordinačių reikšmių masyvas), $S(100)$ (interpoliavimo taškų vertikalųjį mastelį keičiančio parametro reikšmių masyvas). Įvedama fraktalinė objekto dimensija fdim bei koordinatinis tinklelis s_x x s_y. Įvesti duomenys panaudojami afiniųjų transformacijų koeficientams apskaičiuoti:

$$a_i = \frac{x_i - x_{i-1}}{x_N - x_0},\ c_i = \frac{F_i - F_{i-1}}{x_N - x_0} - d_i \frac{F_n - F_0}{x_N - x_0}$$

$$e_i = \frac{x_N x_{i-1} - x_0 x_i}{x_N - x_0},\ f_i = \frac{x_N F_{i-1} - x_0 F_i}{x_N - x_0} - d_i \frac{x_N F_0 - x_0 F_N}{x_N - x_0}.$$

Gautos koeficientų reikšmės surašomos į rezultatų masyvus: $a(100)$, $c(100)$, $e(100)$, $f(100)$.

FIF kreivė pradedama formuoti nuo koordinatinių ašių. Tam pirmiausiai apskaičiuojamos didžiausios ir mažiausios duomenų masyvų $X(100)$ bei $Y(100)$ reikšmės: *minx*, *maxx*, *miny*, *maxy*. Gautos afiniųjų pertvarkių koeficientų reikšmės leidžia aprašyti FIF kreivės taškų koordinates $X1$ ir $Y1$. Per šiomis koordinatėmis nusakomus taškus brėžiama FIF kreivė.

```
Rem ********************************************************
Rem *                G. Sudniute                           *
Rem *                                                      *
Rem *  Fraktaliniu interpoliaciniu funkciju kreiviu generavimo priemone   *
Rem *                                                      *
Rem *                Kaunas, 2005                          *
Rem ********************************************************

Rem Pagrindine skaiciavimu dalis
Private Sub Command1_Click()
Rem min max elementu paieska
maxx = X(0)
minx = X(0)
maxy = Y(0)
miny = Y(0)
For i = 1 To m
```

```
If X(i) > maxx Then maxx = X(i)
If Y(i) > maxy Then maxy = Y(i)
If X(i) < minx Then minx = X(i)
If Y(i) < miny Then miny = Y(i)
Next i
xs = sx / (maxx - minx)
ys = (sy / (maxy - miny)) / 2
xo = -(Int(minx * xs))
yo = sy - ya + (Int(miny * ys))

Rem afiniuju transformaciju koeficientu apskaiciavimas:
B = X(m - 1) - X(0)
Sum = 0
For i = 0 To m - 1
Sum = Sum + Abs(s(i))
Next i
Rem pos= (n-1,fdim-1)/sum
For i = 1 To m - 1
Rem s(i)=pos
a(i) = (X(i + 1) - X(i)) / B
e(i) = (X(m - 1) * X(i) - X(0) * X(i + 1)) / B
c(i) = (Y(i + 1) - Y(i) - s(i) * (Y(m - 1) - Y(0))) / B
f(i) = (X(m - 1) * Y(i) - X(0) * Y(i + 1) - s(i) * (X(m - 1) * Y(0) - X(0) * Y(m - 1))) / B
Next i

Rem rezultatu spausdinimas
Picture2.Print "xi=";
For i = 1 To m
Picture2.Print X(i); " ";
Next i
Picture2.Print
Picture2.Print "yi=";
For i = 1 To m
Picture2.Print Y(i); " ";
Next i
```

```
Picture2.Print
Picture2.Print "si=";
For i = 1 To m
Picture2.Print s(i); " ";
Next i
Picture2.Print
Picture2.Print "ai=";
For i = 1 To m
Picture2.Print a(i); " ";
Next i
Picture2.Print
Picture2.Print "ci=";
For i = 1 To m
Picture2.Print c(i); " ";
Next i
Picture2.Print
Picture2.Print "ei=";
For i = 1 To m
Picture2.Print e(i); " ";
Next i
Picture2.Print
Picture2.Print "fi=";
For i = 1 To m
Picture2.Print f(i); " ";
Next i
End Sub

Rem Duomenu nuskaitymo dalis
Private Sub Command5_Click()
Dim X(100), Y(100), s(100) As Double
Dim a(100), c(100), e(100), f(100) As Double
Dialog.Filter = "Visi failai|*.*|Duomenys|*.dat"
Dialog.DialogTitle = "Pasirinkite duomenų failą"
Dialog.Action = 1
dkelias = Dialog.InitDir
```

```
dfailas = Dialog.FileName
duomenys = dkelias + "\" + dfailas
Picture1.Print "Jus pasirinkote:"; duomenys

Open CommonDialog1.FileName For Input As #1

m = 0
Do Until EOF(1)
m = m + 1
Input #1, X(m), Y(m), s(m)
Loop
Close #1

Rem duomenu spausdinimas
For i = 1 To m
Picture1.Print X(i); " "; Y(i); " "; s(i)
Next i

fdim = Text1.Text
sx = Text2.Text
sy = Text3.Text
Rem duomenu apdorojimas

End Sub

Rem Rezultatu issaugojimo dalis
Private Sub Command3_Click()
Dialog.Filter = "Visi failai|*.*|Rezultatai|*.rez"
Dialog.DialogTitle = "Išsaugokite gautus skaičiavimų rezultatus"
Dialog.Action = 2
rkelias = Dialog.InitDir
rfailas = Dialog.FileName
rezultatai = rkelias + "\" + rfailas
Picture2.Print "Jus pasirinkote:"; rezultatai
Open CommonDialog2.FileName For Output As #2
```

```
Rem rezultatu irasymas i faila
Print #2, "xi=";
For i = 1 To m
Print #2, X(i); " ";
Next i
Print #2,
Print #2, "yi=";
For i = 1 To m
Print #2, Y(i); " ";
Next i
Print #2,
Print #2, "si=";
For i = 1 To m
Print #2, s(i); " ";
Next i
Print #2,
Print #2, "ai=";
For i = 1 To m
Print #2, a(i); " ";
Next i
Print #2,
Print #2, "ci=";
For i = 1 To m
Print #2, c(i); " ";
Next i
Print #2,
Print #2, "ei=";
For i = 1 To m
Print #2, e(i); " ";
Next i
Print #2,
Print #2, "fi=";
For i = 1 To m
Print #2, f(i); " ";
Next i
```

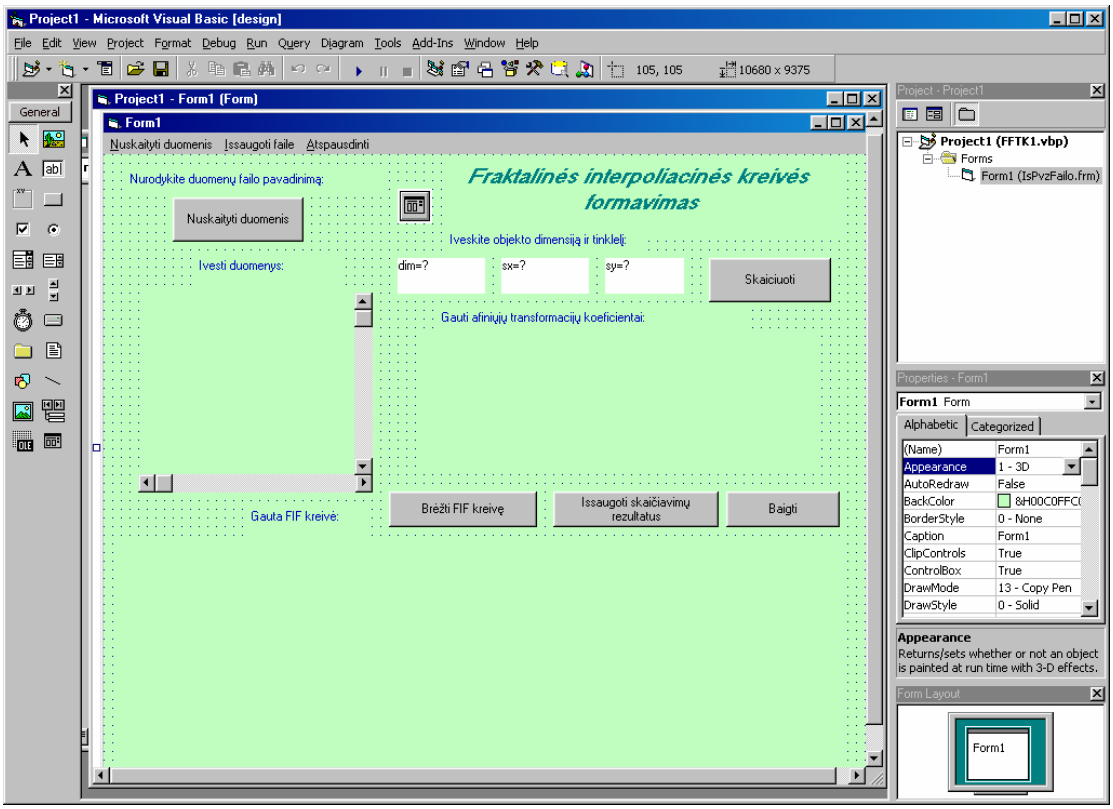
```
Close #2

End Sub

Rem FIF kreives formavimo dalis
Private Sub Command4_Click()
ScaleMode = vbPixels
Rem grafo liniju braizymas:
xg = xg * xs
yg = yg * ys
For X1 = xo To sx Step xg
Picture3.Line (Int(X1), sy / 2 - ya * 2)-(Int(X1), sy - 1), 7
Next X1
For Y1 = yo To sy / 2 - ya * 2 Step -yg
Picture3.Line (0, Int(Y1))-(sx - 1, Int(Y1))
Next Y1
For i = 0 To m
X1 = X(i) * xs + xo
Y1 = yo - Y(i) * ys
Picture3.Line (Int(X1) - 2, Int(Y1) - 2)-(Int(X1) + 2, Int(Y1) + 2), 3, B
Next i
X1 = xo
Y1 = yo
Rem while !kbhit...
For j = 1 To m
j = Rnd(m - 1)
X2 = a(j) * X1 + e(j)
Y2 = c(j) * X1 + s(j) * Y1 + f(j)
X1 = X2
Y1 = Y2
print3.PSet (Int(X1 * xs + xo), Int(yo - Y1 * ys)), 11
Rem wend
Next j
Picture3.Cls
```

```
For i = 1 To m
Picture3.PSet (X(i), Y(i)), 7
Next i
End Sub

Rem Darbo pabaigos dalis
Private Sub Command2_Click()
End
End Sub

Rem Duomenu spausdinimo meniu dalis
Private Sub menDuomenis_Click()
Rem duomenu spausdinimas
For i = 1 To m
Picture1.Print X(i); " "; Y(i); " "; s(i)
Next i
End Sub

Rem FIF kreives formavimo meniu dalis
Private Sub menFIFkreive_Click()
ScaleMode = vbPixels
Rem grafo liniju braizymas:
xg = xg * xs
yg = yg * ys
For X1 = xo To sx Step xg
Picture3.Line (Int(X1), sy / 2 - ya * 2)-(Int(X1), sy - 1), 7
Next X1
For Y1 = yo To sy / 2 - ya * 2 Step -yg
Picture3.Line (0, Int(Y1))-(sx - 1, Int(Y1))
Next Y1
For i = 0 To m
X1 = X(i) * xs + xo
Y1 = yo - Y(i) * ys
Picture3.Line (Int(X1) - 2, Int(Y1) - 2)-(Int(X1) + 2, Int(Y1) + 2), 3, B
Next i
```

```
X1 = xo
Y1 = yo
Rem while !kbhit...
For j = 1 To m
j = Rnd(m - 1)
X2 = a(j) * X1 + e(j)
Y2 = c(j) * X1 + s(j) * Y1 + f(j)
X1 = X2
Y1 = Y2
print3.PSet (Int(X1 * xs + xo), Int(yo - Y1 * ys)), 11
Rem wend
Next j
Picture3.Cls
For i = 1 To m
Picture3.PSet (X(i), Y(i)), 7
Next i
End Sub

Rem Kreives irasymo i grafini faila meniu dalis
Private Sub menKreiveFaile_Click()
Rem issaugoti kreives vaizda JPEG ar GIF formate
End Sub

Rem Private Sub HScroll1_Change()
Rem Picture1.Print HScroll1.CausesValidation
Rem End Sub

Rem Duomenu nuskaitymo meniu dalis
Private Sub menNuskaitytiDuomenis_Click()

Dim X(100), Y(100), s(100) As Double
Dim a(100), c(100), e(100), f(100) As Double

Meniu.Filter = "Visi failai|*.*|Duomenys|*.dat"
Meniu.DialogTitle = "Pasirinkite duomenų failą"
```

```
Meniu.Action = 1
dkelias = Meniu.InitDir
dfailas = Meniu.FileName
duomenys = dkelias + "\" + dfailas
Picture1.Print "Jus pasirinkote:"; duomenys
Open CommonDialog1.FileName For Input As #1

m = 0
Do Until EOF(1)
m = m + 1
Input #1, X(m), Y(m), s(m)
Loop
Close #1
End Sub

Rem Rezultatu skaiciavimo meniu dalis
Private Sub menRezultataiFaile_Click()
Dialog.Filter = "Visi failai|*.*|Rezultatai|*.rez"
Dialog.DialogTitle = "Išsaugokite gautus skaičiavimų rezultatus"
Dialog.Action = 2
rkelias = Dialog.InitDir
rfailas = Dialog.FileName
rezultatai = rkelias + "\" + rfailas
Picture2.Print "Jus pasirinkote:"; rezultatai
Open CommonDialog2.FileName For Output As #2
Rem rezultatu irasymas i faila
Print #2, "xi=";
For i = 1 To m
Print #2, X(i); " ";
Next i
Print #2,
Print #2, "yi=";
For i = 1 To m
Print #2, Y(i); " ";
Next i
```

```
Print #2,
Print #2, "si=";
For i = 1 To m
Print #2, s(i); " ";
Next i
Print #2,
Print #2, "ai=";
For i = 1 To m
Print #2, a(i); " ";
Next i
Print #2,
Print #2, "ci=";
For i = 1 To m
Print #2, c(i); " ";
Next i
Print #2,
Print #2, "ei=";
For i = 1 To m
Print #2, e(i); " ";
Next i
Print #2,
Print #2, "fi=";
For i = 1 To m
Print #2, f(i); " ";
Next i

Close #2
End Sub

Rem Skaiciavimu rezultatu spausdinimo meniu dalis
Private Sub menSkaiciavimuRezultatus_Click()
Rem rezultatu spausdinimas
Picture2.Print "xi=";
For i = 1 To m
Picture2.Print X(i); " ";
```

```
Next i
Picture2.Print
Picture2.Print "yi=";
For i = 1 To m
Picture2.Print Y(i); " ";
Next i
Picture2.Print
Picture2.Print "si=";
For i = 1 To m
Picture2.Print s(i); " ";
Next i
Picture2.Print
Picture2.Print "ai=";
For i = 1 To m
Picture2.Print a(i); " ";
Next i
Picture2.Print
Picture2.Print "ci=";
For i = 1 To m
Picture2.Print c(i); " ";
Next i
Picture2.Print
Picture2.Print "ei=";
For i = 1 To m
Picture2.Print e(i); " ";
Next i
Picture2.Print
Picture2.Print "fi=";
For i = 1 To m
Picture2.Print f(i); " ";
Next i

End Sub
```

VB Interpo maketas: