VILNIAUS UNIVERSITETAS
FIZIKOS FAKULTETAS
KVANTINĖS ELEKTRONIKOS KATEDRA
BIOFOTONIKOS LABORATORIJA

Laboratorinis darbas (BSEM)

Biologinių sistemų evoliucijos modeliavimas

Darbą paruošė: K. Ikamas
R.Rotomskis

VILNIUS
1995

1. Darbo tikslas

Susipažinti su dinaminių sistemų ir fraktalų teorijomis; ištirti konkrečią dinaminę sistemą.

2. Darbo užduotys

1. Susipažinti su dinaminių sistemų teorija.
2. Susipažinti su fraktalų teorija.
3. Peržiūrėti demonstracines programinių paketų "Fractint" ir "PHASER" programas.
4. Susipažinti su programa "Evoliucija" .
5. Rasti regresijos ir katastrofos sąlygas atsitiktinei kolonijai.
6. Išmatuoti α *ir p* parametrus regresijos, katastrofos ir pusiausvyrinių sąlygų atvejais.

3. Teorinė dalis

Gamtoje sunku rasti sistemą, kuri nekistų laike. Net ir kalnai, iš pažiūros nekintantys, iš tikrųjų, veikiami erozijos procesų ir tektoninių jėgų, kinta. Tiriant biologinius procesus dažnai svarbiau yra žinoti, kaip kinta laike tam tikri sistemos parametrai, o ne statinis jos vaizdas. Statinis sistemos vaizdas teikia mažai informacijos apie tos sistemos elgseną arba evoliuciją laike. Todėl dinaminių sistemų teorija užima svarbią vietą biologinių, fizikinių, cheminių ir kitų reiškinių tyrime.

Dinaminę sistemą – sistemą, kintančią laike – galima aprašyti tam tikru kintamųjų ir parametrų rinkiniu. Kintamieji kinta laike, tuo tarpu parametrai išlieka pastovūs (bendru atveju ir parametrai gali kisti laike). Kintamaisiais gali būti, pavyzdžiui, individų skaičius, membranos elektrinis potencialas, medžiagos koncentracija, o parametrais – maisto ištekliai, elektrinis membranos laidumas, temperatūra ir pan. Tarkime, sistema aprašoma n kintamaisiais x_i (i = 1,2,...,n) ir m parametrais a_j (j = 1,2,...,m). Tada tokios sistemos kitimą laike galima aprašyti pirmos eilės diferencialinių lygčių sistema:

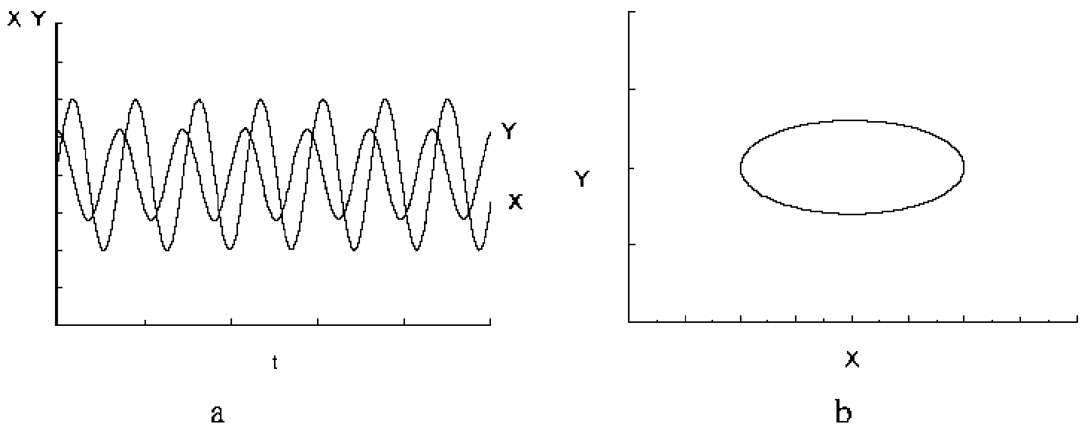
$$\begin{cases} \dfrac{dx_1}{dt} = f_1(x_1, x_2, ..., x_n, a_1, a_2, ..., a_m) \\ \dfrac{dx_2}{dt} = f_2(x_1, x_2, ..., x_n, a_1, a_2, ..., a_m) \\ \vdots \\ \dfrac{dx_n}{dt} = f_n(x_1, x_2, ..., x_n, a_1, a_2, ..., a_m) \end{cases} \quad (1).$$

Kintamasis x_i kinta laike $x_i = x_i(t)$ dėl sąveikos su kitais sistemos kintamaisiais. $\frac{dx_i}{dt}$ yra kintamojo kitimo greitis. Funkcijos $f_1,...,f_n$ priklauso ir nuo parametrų $a_1,...,a_n$.

Kai kurių paprastų dinaminių sistemų atveju (1) lygčių sistema išsprendžiama lengvai, kitu atveju galima gauti tik dalinį sprendinį. Pažymėtina, kad pirmas atvejis realioms sistemoms pasitaiko retai. Bendru atveju sprendinys turi pavidalą

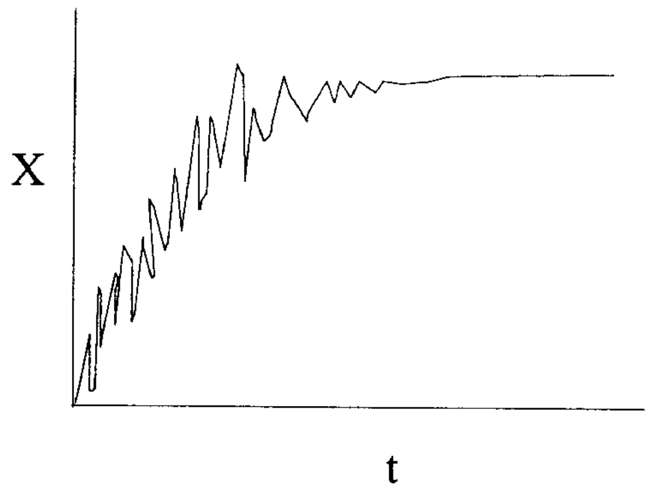
$$x_i = g_i(t, x_1^0, x_2^0, ..., x_n^0, a_1, a_2, ..., a_m) \qquad (2),$$

čia x_i^0 – kintamojo x_i reikšmė pradiniu laiko momentu (t = 0) ir vadinama pradinėmis sąlygomis. Turėdami tikslų sprendinį ir pradines sąlygas, galime išsamiai nupasakoti dinaminės sistemos evoliuciją laike.



1pav. Dinaminės sistemos laikinė diagrama (a) ir fazinė erdvė (b).

Dinaminių sistemų tyrimui patogu įvesti fazinę erdvę. Jei sistemą aprašo du kintamieji x_1 ir x_2, kurie kinta laike $(x_1 = x_1(t)\ ir\ x_2 = x_2(t))$, tai sistemos būsena tam tikru laiko momentu aprašoma tašku (x_1, x_2) fazinėje erdvėje. Šiuo atveju fazinė erdvė bus plokštuma, kurios x-ašyje atidedama x_1 reikšmė laiko momentu t, o y-ašyje – x_2 reikšmė tuo pačiu laiko momentu (l pav.). Bendru atveju, jei sistemą aprašo n kintamųjų, tai ją galima apibūdinti n-mate fazine erdve.



2 pav. Pereinamieji reiškiniai dinaminėje sistemoje.

Fazinė erdvė patogi kokybiniam dinaminės sistemos aprašymui. Šioje erdvėje sistemos evoliucija pateikiama geometriniu pavidalu. Atraktoriai – tai geometrinės struktūros, charakterizuojančios sistemos elgseną fazinėje erdvėje, kai laikas artėja į begalybę. Pradžioje sistema nėra pusiausvyroje, joje vyksta pereinamieji procesai (2 pav.). Tik praėjus tam tikram laikui (griežčiau kalbant, kai $t \to \infty$) pradeda nusistovėti pusiausvyra (jeigu ji būdinga nagrinėjamai sistemai). Atraktorius ir yra ta pusiausvyra, į kurią stengiasi patekti sistema.

Yra keturios atraktorių rūšys:

1. Stabilus taškas arba pusiausvyros padėtis. Sistemos kintamieji nekinta laiko atžvilgiu, t.y. jų išvestinės $\frac{dx_i}{dt} = 0$. Fazinėje erdvėje turėsime tašką. Realus modelis – paprasčiausia pirmos eilės cheminė reakcija:

$$A \xrightarrow{k} B$$

čia A, B – pradinė medžiaga ir produktas; *k* - reakcijos spartos konstanta. Tokios sistemos diferencialinių lygčių sistema

$$\begin{cases} \frac{d[A]}{dt} = -k[A] \\ \frac{d[B]}{dt} = k[A] \end{cases} \qquad (3),$$

o sprendiniai

$$[A] = [A_0]e^{-kt}$$
$$[B] = [A_0](1 - e^{-kt})$$

čia $[A]$ ir $[B]$ – pradinės medžiagos ir produkto koncentracijos; $[A_0]$ – medžiagos *A* koncentracija reakcijos pradžioje (laiko momentu $t = 0$).

Kartais sistema gali turėti kelias pusiausvyrąsias būsenas. Tokiu atveju, į kurią būseną sistema pateks, priklauso nuo pradinių sąlygų.

2. Ciklas. Jei sistemoje nusistovi periodiniai svyravimai, fazinėje erdvėje turėsime uždarą kreivę, kuri vadinama ciklu. Pavyzdys: Voltera aukos ir plėšrūno modelis. Šį modelį aprašo dif. lygčių sistema:

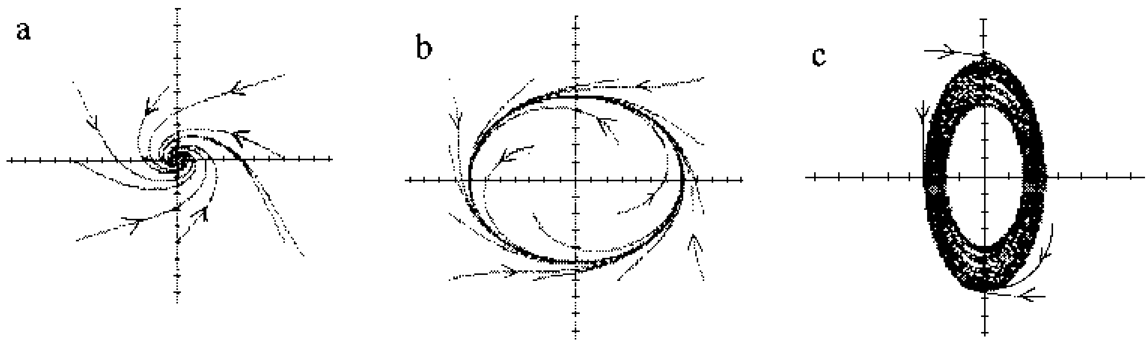
$$\begin{cases} \frac{dx}{dt} = \alpha_1 \cdot x - \beta_1 \cdot x \cdot y \\ \frac{dy}{dt} = \alpha_2 \cdot x \cdot y - \beta_2 \cdot y \end{cases} \qquad (4),$$

kur x – aukų skaičius, y – plėšrūnų skaičius, α_i – dauginimosi koeficientai, β_i – mirimo ir žuvimo koeficientai.

3. Toro pavidalo atraktorius būdingas sistemai, kurioje nusistovi kvaziperiodiniai svyravimai. Tokią elgseną atitinka du ar daugiau nepriklausomi svyravimai – osciliacijos. Lygčių sistema, aprašanti osciliacijas mechanikoje, elektrinėse schemose ir pan., gali būti tokio pavidalo:

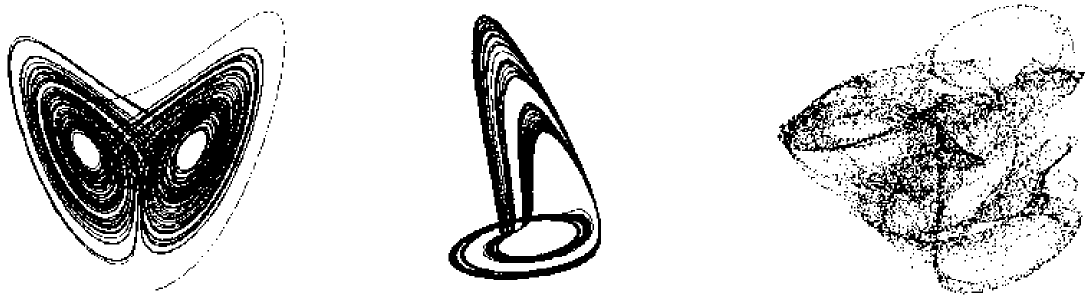
$$\begin{cases} \frac{dx_1}{dt} = x_2 \\ \frac{dx_2}{dt} = -\frac{c}{m}x_2 - \frac{k}{m}x_1 + \frac{f}{m}\cos(a \cdot x_3) \\ \frac{dx_3}{dt} = const. \end{cases} \qquad (5),$$

čia f – sistemą veikianti jėga; *k, m, c* – parametrai. Daugiamačiu atveju turėsime n-mačius torus.



3 pav. Sistemos atraktoriai: a - stabilus taškas; b - ciklas; c- toras.

4. Pirmieji trys atraktoriai daugiau ar mažiau sudėtingi, tačiau juos jungia viena bendra savybė: sistemų, kurioms būdingi tokie atraktoriai, elgsena yra prognozuojama, t.y. žinant dėsnius, pagal kuriuos sistema kinta ir dabartinę jos būseną, galima nustatyti sistemos būseną bet kuriuo laiko momentu. Tačiau ne visos sistemos prognozuojamos. Neprognozuojamų sistemų elgsena yra atsitiktinė. Tokias sistemas fazinėje erdvėje atitinka *chaotinis* atraktorius arba keistasis atraktorius (4 pav.). Pačios sistemos vadinamos chaotinėmis sistemomis. Chaotinės sistemos pavyzdžiu gali būti orai, turbulentinis tekėjimas. Gamtoje dažniausiai sutinkamos chaotinės sistemos, tuo tarpu kitos sistemos yra išimtis ir dažniausiai tik supaprastinti realių sistemų modeliai.

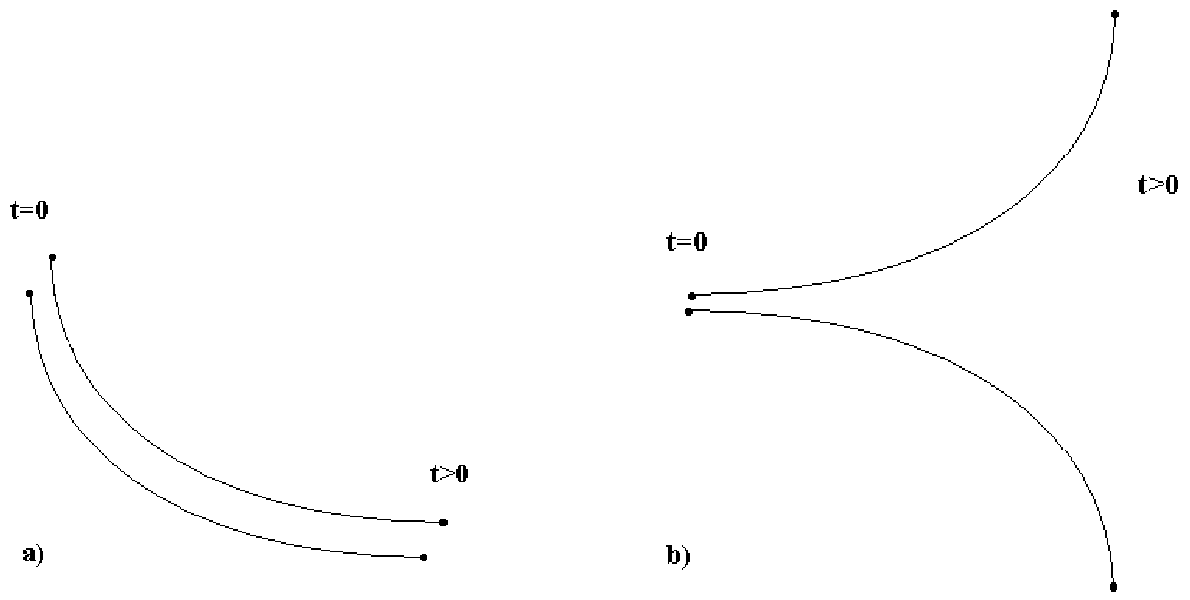


4 pav. Chaotiniai atraktoriai

Svarbi chaotinių sistemų savybė yra jų jautrumas pradinėms sąlygoms. Jei fazinėje erdvėje padėsime du artimus taškus, kurie atitinka dvi pradines sąlygas, ir paseksime, kaip evoliucionuoja sistema, tai pastebėsime, kad chaotinių sistemų atveju šie du taškai labai .greitai nutols vienas nuo kito. Paprastų, nechaotinių stabilių sistemų atvejais dvi artimos trajektorijos išlieka artimos kiek norime laiko (5 pav.). Kitaip sakant, atstumas tarp dviejų pradinių matuojamų dydžių (pradinė paklaida) chaotinėje sistemoje po tam tikro laiko tampa palyginamas su pačiais dydžiais. Dėl šios savybės chaotinės sistemos yra sunkiai prognozuojamos.

Dar viena chaotinių sistemų savybė – periodinių taškų egzistavimas. Tai yra, galima parinkti tokias pradines sąlygas, kad dinaminėje sistemoje po kurio laiko nusistovi pusiausvyra. Trečia savybė – maišymasis – susijusi su atskirų sistemos kintamųjų elgsena fazinėje erdvėje. Jei pasirinksime tam tikrą reikšmių, kurias gali įgyti vienas iš kintamųjų, intervalą ir paseksime, kaip kintamasis laikui bėgant padengia tą intervalą, tai pastebėsime, kad chaotinėje sistemoje (laikui artėjant į begalybę) kintamasis padengs intervalą kiek norima tankiai. Kitaip sakant, chaotinės sistemos būsena niekada nesikartoja (išimtis – periodiniai taškai).Visos šios savybės yra kokybinės chaoso tam tikroje sistemoje charakteristikos. Ne visada konkrečiai sistemai visos šios sąlygos būdingos.

Kiekybinės dinaminių sistemų charakteristikos yra Liapunovo rodikliai ir sistemos atraktoriaus matavimas (dimensija). Liapunovo rodikliai kiekybiškai nusako jautrumą



5 pav. Paprastos (a) ir chaotinės (b) sistemos jautrumas pradinėms sąlygoms.

pradinėms sąlygoms (tiksliau, pirmasis Liapunovo rodiklis). Paprastai, priklausomai nuo to, kokio matavimo yra fazinė erdvė (d), tiek yra Liapunovo rodiklių. Pagrindinis yra pirmasis rodiklis, kuris įvertina, kaip sparčiai didėja pradinė paklaida tarp pradinių taškų. Jo išraiška vienmačiu diskretiniu atveju

$$l_1 = \lim_{n\to\infty} \lim_{E_0\to 0} \frac{1}{n} \sum_{k=1}^{n} \log\left|\frac{E_k}{E_{k-1}}\right| \qquad (6),$$

$$E_k = x_1 - x_2$$

čia E_0 – pradinė paklaida; E_k - paklaida k žingsnyje; E_{k-1} – paklaida ankstesniame žingsnyje; n – žingsnių skaičius. Jei k-ajame žingsnyje paklaida buvo E_k, tai kitame žingsnyje paklaida vidutiniškai bus lygi $E_{k+1} = E_k \cdot e^{\lambda l}$. Aukštesnės eilės rodikliai tenkina sąlygą

$$\lambda_1 > \lambda_2 > ... > \lambda_d \qquad (7)$$

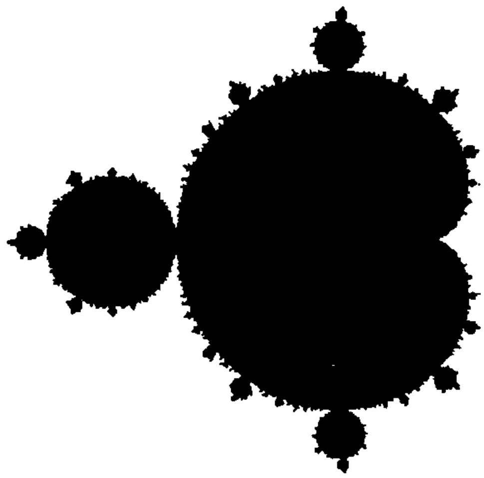
Geometrinė Liapunovo rodiklių prasmė yra tokia: jei n-matėje fazinėje erdvėje pasirinksime pradinį tūrį, tai laikui bėgant šis tūris kis pagal eksponentinį dėsnį:

$$V_{k+1} \approx V_k \cdot \exp(\lambda_1 + \lambda_2 + ... + \lambda_d) \qquad (8)$$

Yra parodyta kad chaotinei sistemai $\lambda_l > 0$. Taigi, pirmas Liapunovo rodiklis nusako dinaminės sistemos pobūdį.

Kitas kiekybinis parametras – atraktoriaus matas arba dimensija. Paprastas sistemas (fazinėje erdvėje) atitinka sveiko skaičiaus atraktoriaus dimensija, tuo tarpu chaotinių sistemų atraktoriai dažnai turi trupmenines dimensijas. Bendrai dimensiją galima traktuoti kaip sudėtingumo matą. Chaotiniai atraktoriai dažnai yra fraktalinės struktūros – struktūros, pasižyminčios trupmenine dimensija.

Pirmasis fraktalo sąvoką apie 1980 metus (lot. "fractus", nuo veiksmažodžio „fragere“ – laužyti, kurti nereguliarius raštus) įvedė Mandelbrotas (Mandelbrot), IBM firmos matematikas. Iki jo jau buvo žinomos fraktalinės struktūros (dar vadinamos matematiniais „monstrais“), tokios kaip Kantoro (Cantor) aibė, Peano (Pean) kreivė, Kocho (Koch) kreivė, Sierpinskio (Sierpinski) trikampis ir kilimas, Žiulia (Julia) aibės. Tačiau tik Mandelbrotas įžvelgė jose daugiau negu matematikų išmones ir išvystė naują mokslo šaką – fraktalinę geometriją. Per nepilnus 20 metų fraktalų sąvoka pradėta naudoti daugelyje mokslo šakų: fizikoje, biologijoje, chemijoje, sociologijoje, urbanistikoje... Kaip jau buvo minėta, chaotinių sistemų atraktoriai yra fraktalai. Todėl fraktalinė geometrija iš esmės yra chaoso geometrija.

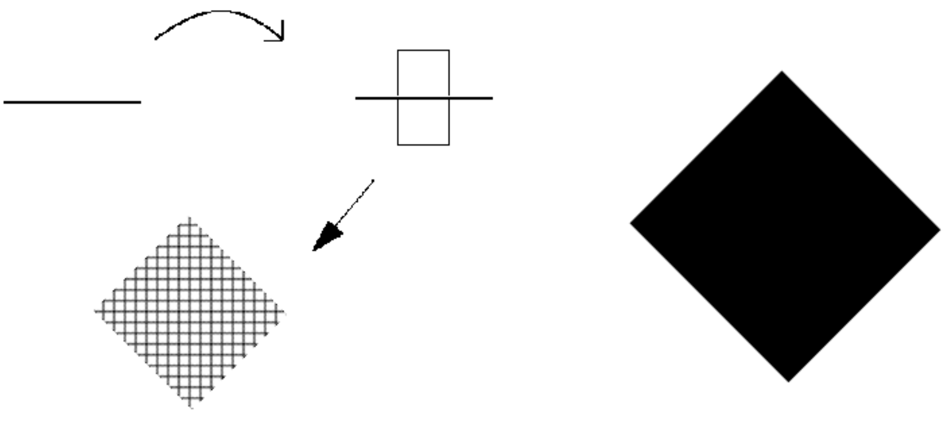


6 pav. Mandelbroto fraktalas.

Fraktalas – tai savipanašumu (mastelio simetrija) pasižymintis objektas. Savipanašumas – tai tokia struktūros savybė, kai objektą galima išskaidyti į mažas dalis, kurios yra sumažintos visos struktūros kopijos. Pavyzdžiui, medžio šaka yra panaši į viso augalo sumažintą kopiją, dėl to medžiai yra natūralūs fraktalai. Gamtoje natūralių fraktalų yra labai daug: pradedant augalais, žmogaus organais (inkstai, plaučiai, kraujotakos ir nervų sistema) ir baigiant kalnais, debesimis, kranto linijomis. Kartais struktūra nėra griežtai savipanaši. Pavyzdžiui, Brauno dalelės trajektorija yra fraktalas, tačiau ji tėra iš dalies savipanaši. Tokiu atveju fraktalas bus objektas, kuriame didinant išryškėja vis daugiau detalių. Matematiškai fraktalu vadinama struktūra, kurios Hausdorfo dimensija yra griežtai didesnė už topologinę dimensiją:

$$D_H > D_T \qquad (9)$$

Topologinė dimensija D_T – tai klasikinis figūrų matiškumas, priskiriantis: taškui – 0 matavimą, kreivei – l, plokštumai (kvadratui) – 2, kubui – 3 ir t.t. Tarpinių;, trupmeninių matavimų negali būti. Ilgą laiką tebuvo tik tokie matavimai. Tačiau atsirado geometrinės struktūros, kurių dimensija buvo aiškiai didesnė nei topologinė. Pvz., plokštumą užpildanti kreivė (Peano kreivė, 7 pav.); nors kreivės topologinė dimensija yra l, tačiau bendros figūros matavimas yra 2. Dėl to buvo įvesta nauja dimensijos rūšis – Hausdorfo dimensija D_H.



7pav. Peano kreivė: pirmi trys sudarymo žingsniai ir galutinė figūra.

Praktinis D_H matavimas pagal klasikinį apibrėžimą yra sudėtingas, todėl yra matuojamos gimininingos dimensijos:

1. *Savipanašumo dimensija.* Jei objektas pasižymi savipanašumu, tai jo savipanašumo dimensija bus lygi

$$D_S = \frac{\log a}{\log 1/s} \qquad (10),$$

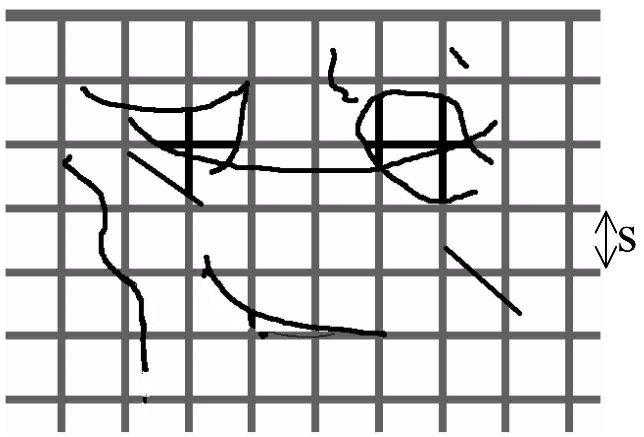
kur *a*- dalių, į kurias galima suskaldyti figūrą skaičius; 1/*s* - didinimo faktorius (kiek reikia padidinti dalį, kad ji taptų visos figūros dydžiu). Kaip pavyzdį galima paskaičiuoti kvadrato dimensiją. Kvadratą galima padalinti į keturis mažesnius kvadratus, kurie yra du kartus sumažintos visos figūros kopijos. Todėl $D_S = \log 4/\log 2 = 2$. Kaip matome, paprastų figūrų fraktalinė dimensija (Hausdorfo dimensija) sutampa su jų topologine dimensija.

2. *Padengimo dimensija.* Figūra padengiama kvadratais (jei figūra yra plokštumoje; n-mačiu atveju padengiama n-mačiais kubais). Kvadrato kraštinės ilgis yra s_k. Suskaičiuojamas kvadratų, į kuriuos patenka figūra, skaičius N_k. Skaičiavimai pakartojami su kitu kvadrato kraštinės ilgiu $s_{k+1} < s_k$. Tada padengimo dimensija

$$D_b = \frac{\log N_{k+1} - \log N_k}{\log 1/s_{k+1} - \log 1/s_k} \qquad (11)$$

Jei padengiama ne kvadratais, o bet kokios formos elementariais n-mačiais tūrio vienetais, ir tų tūrio vienetų spindulys artėja prie nulio, tai padengimo dimensija D_b virsta Hausdorfo dimensija:

$$D_H = \lim_{s \to 0} \frac{\log N(s)}{\log 1/s} \qquad (12)$$



8 pav. Padengimo dimensijos skaičiavimas.

3. Dar vienas būdas dimensijos matavimui yra *laipsninis dėsnis.* Tarkime, matuojamas dydis yra *u.* Jei l/*s* yra matavimo tikslumas, tai fraktalinei struktūrai galioja laipsninis dėsnis

$$u \sim (1/s)^d \qquad (13)$$

Atidėjus grafike x ašyje matavimo tikslumo logaritmą $\ln 1/s$, o y ašyje – matuojamo dydžio logaritmą $\ln u$ gausime tiesę, kurios polinkis yra lygus *d.* Fraktalinė dimensija randama iš paprasto sąryšio:

$$D_C = 1 + d \qquad (14)$$

Šiuo būdu buvo išmatuota Didžiosios Britanijos salos kranto ilgio dimensija, kuri lygi 1.31. Iš tikrųjų, matuodami skirtingo tikslumo liniuotėmis, gausime skirtingo ilgio kranto liniją. Taip atsitinka todėl, kad matuodami tiksliau, įskaitome vis smulkesnius kranto nelygumus, iškyšulius ir įlankas, – dėl to matuojamas ilgis padidėja.

Be šių trijų fraktalinių dimensijų yra dar kitos (informacinė, entropijos ir pan.). Dažnai išmatuotos tam pačiam objektui įvairios dimensijos sutampa, tačiau kartais ir skiriasi. Nežiūrint to, fraktalinė dimensija yra svarbus nagrinėjamo objekto sudėtingumo matas.

Objektas	Topologinė dimensija	Fraktalinė dimensija
Kreivė	1	1
Kvadratas	2	2
Kubas	3	3
Kocho kreivė	1	1.26
Peano kreivė	1	2
Sierpinskio trikampis	1	1.89
Brauno judėjimas	1	2
Cinko elektrolizė*	1	1.71
Anglijos kranto linija	1	1.31

*Cinko elektrolizė- elektrolizės metu susidariusio cinko sluoksnio struktūra turi fraktalinių savybių.

Yra keletas matematinių metodų fraktalams gauti:

1. *Iteracinių funkcijų sistema* (IFS). Imama bet kokios formos pradinė figūra ir jai pritaikomas afininių transformacijų rinkinys. Afininė transformacija w susideda iš elementarių posūkio, atspindžio, sumažinimo ir slinkties transformacijų. Jei $P = (x,y)$ yra taškas plokštumoje, tai pritaikius šiam taškui afininę transformaciją, gausime kitą tašką, kurio koordinatės $P = (u,v)$ skaičiuojamos pagal sekančias lygtis:

$$\begin{aligned} u &= ax + by + e \\ v &= cx + dy + f \end{aligned} \qquad (15)$$

čia $a,...,f$ – transformacijos koeficientai.
Pritaikę afininių transformacijų rinkinį figūrai (taškų aibei), gausime kitą figūrą (kitą taškų aibę). Jei dabar pritaikysime gautam vaizdui tas pačias transformacijas, turėsime dar kitą figūrą. Jei šį procesą kartosime, po tam tikro laiko gausime vaizdą, kuris nesikeis, pritaikius jam transformacijas. Šis vaizdas vadinamas atraktoriumi ir paprastai yra fraktalinė struktūra.

Galimas IFS modifikavimas – kelios iteracinių funkcijų sistemos, kurios tarpusavyje surištos. Naudojant tokias sudėtingas sistemas, galima gauti iš dalies savipanašias figūras, tuo tarpu pavienė IFS visada duos griežtai savipanašų galutinį vaizdą; dalinai savipanašaus objekto tik kai kurios dalys yra sumažintos viso objekto kopijos, tuo tarpu likusios šia savybe nepasižymi.

IFS bandoma panaudoti vaizdų kodavimui. Jei pavyktų rasti transformacijas, kurių dėka galima gauti vaizdą, tai būtų galima žymiai sutaupyta vietos vaizdo saugojimui atmintyje (net iki 1000 kartų).

2. Praktiniam IFS pritaikymui yra svarbus vienas parametras – po kokio transformacijų pritaikymo skaičiaus bus gautas atraktorius. Kartais šis skaičius gali būti labai didelis ir praktikoje neįgyvendinimas (tada vaizdo atkodavimui prireiktų labai daug laiko).

Todėl yra naudojamas dar vienas būdas fraktalui gauti – *atsitiktinių iteracinių funkcijų sistema.* Šio metodo esmė yra ta, kad taikomos ne visos transformacijos iš karto, bet tik viena iš jų, be to, veikiamas pavienis taškas. Jei yra n afininių transformacijų w_i, tai joms parenkamos atitinkamos pasirodymo tikimybės p_i ($\sum p_i = 1$).

9 pav. Barnslėjaus (M.Barnsley) paparčio lapas.

Kokia ir kaip dažnai transformacija bus taikoma priklauso nuo jos tikimybės. Tokiu būdu gauta taškų aibė z_0, $z_1 = w_{s1}(z_0)$, $z_2 = w_{s2}(z_1)$, $z_3 = w_{s3}(z_1)$,... (kur $s1, s2, s3$ yra atsitiktiniai skaičiai iš intervalo 1...n) sudarys tą patį atraktorių, kuris gaunamas pritaikius IFS. Teisingai parinkus tikimybes, laikas vaizdo atkodavimui žymiai sumažėja. Pvz., 9 pav. pavaizduotą fraktalą galima užkoduoti 4 afininėm transformacijom. Tačiau naudojant IFS jo atkodavimui greičiausiu kompiuteriu prireiktų milijardų metų. Tuo tarpu pritaikant atsitiktinių iteracinių funkcijų sistemą, vaizdas gaunamas per kelias sekundes. Atskira problema yra tinkamų transformacijų parinkimas vaizdo kodavimui.

3. L-sistemos. L-sistemas pirmasis panaudojo Lindenmajeris (Lindenmayer) augalų ir dumblių augimo procesams aprašyti. Kiekviena ląstelė turi savo pažymėjimą – raidę, kuri gali reikšti ląstelės amžių, dydį ar kitokią laike kintančią charakteristiką. Po to įvedamos perrašymo taisyklės kiekvienai raidei. Pvz., ką tik gimusi ląstelė pažymima A raide, o suaugusi – B. Ląstelė subręsta per vieną laiko tarpsnį ir kitu tarpsniu pasidalina į dvi ląsteles. Tokiu atveju perrašymo taisyklės yra tokios: A $\rightarrow$ B ir B $\rightarrow$ AA. Jei pradėtume nuo vienos suaugusios ląstelės, gautume tokį vaizdą:

0 ciklas	B
1 ciklas	AA
2 ciklas	BB
3 ciklas	AAAA
4 ciklas	BBBB
5 ciklas	AAAAAAAA
ir t.t.	

Jei parinktume grafinį gautos sekos pavaizdavimą, galėtume stebėti, kaip kinta erdvėje ląstelių kolonija (paprastai kompiuteriniame L-sistemų modeliavime grafiniam simbolių vaizdavimui naudojama vėžlinė grafika). Tokiu būdu galima vizualiai stebėti fraktalų vystymąsi arba augimą.

4. Ląsteliniai automatai. Ląstelinių automatų teorija nagrinėja savireprodukuojančių mašinų egzistavimo galimybes. Paprastai šios hipotetinės mašinos egzistuoja begaliniame lauke, suskirstytame į elementarius laukelius, su tam tikromis taisyklėmis ir laukelio būsenų skaičiumi. Evoliucijos (kuri prasideda nuo tam tikro pradinio laukelių būsenų pasiskirstymo) metu generuojamos figūros dažnai yra fraktalinės struktūros.

L-sistema:	Krūmas
Pradinis simbolis	F
Perrašymo taisyklė	F $\rightarrow$ FF+[+F-F-F]-[-F+F+F]
	[$\rightarrow$ [
	] $\rightarrow$]
	+ $\rightarrow$ +
	- $\rightarrow$ -

10 pav. Fraktalas gautas naudojantis L-sistema.

Vienas iš ląstelinių automatų pavyzdžių – šiame darbe tiriama sistema „Evoliucija“, kuri aprašoma kvadratiniu lauku, dviem laukelio būsenomis (yra / nėra ląstelės) ir „genetinėmis taisyklėmis“.

Šiais trimis metodais gaunami matematiniai, dirbtiniai fraktalai. Praktinis jų taikymas natūralių procesų ir objektų aprašymui kol kas yra keblus, nors tam tikrų pasiekimų jau yra. Pagrindinė problema yra natūralių fraktalinių struktūrų baigtinumas bei atsitiktinumas. Dėl šių priežasčių natūralūs fraktalai nėra griežtai savipanašūs.

4. Tyrimo metodika

1. "Fractint 18.2" programinis paketas. "Fractint" - tai programa, kurioje sutelkti pagrindiniai fraktalai, gaunami kompiuteriu. Programa sudaro galimybes sukurti ir savus fraktalus. Laboratoriniame darbe panaudota "Fractint" demonstracinė dalis, kurios metu susipažįstama su klasikiniais fraktalais. Programos vykdymo metu būtina atkreipti dėmesį į Mandelbroto fraktalo („mandel“) savipanašumą – po tam tikro didinimų skaičiaus gaunama pradinė figūra.

Demonstracijos metu parodomos programos galimybės, todėl be pačių fraktalų vaizdavimo ekrane dar atliekama eilė papildomų veiksmų: vaizdo padidinimas, pasukimas, fraktalą aprašančios lygties parametrų keitimas, ekrano spalvų keitimas ir t.t.

2. Diferencialinių lygčių sprendimo programinis paketas "PHASER". Programa padeda spręsti diferencialines lygtis ir stebėti jų laikines diagramas bei fazinius portretus. Laboratoriniame darbe naudojama parodomoji programos dalis, kurioje supažindinama su diferencialinėmis lygtimis, jų sprendinių vaizdavimu erdvėje ir visais atraktorių tipais. T.y. programa sudaro galimybes matematiškai modeliuoti dinamines sistemas.

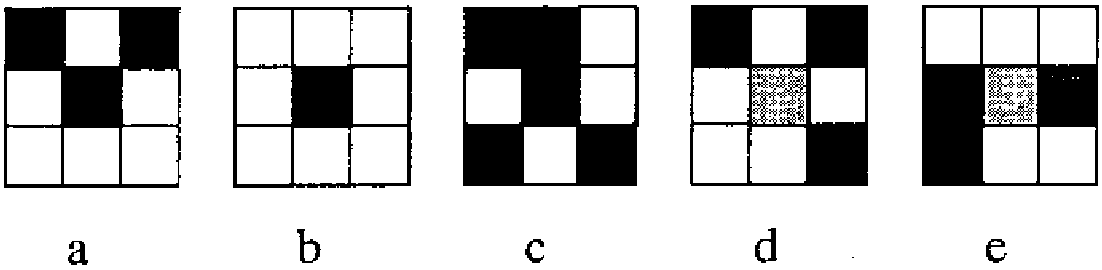
3. Dinaminės sistemos „Evoliucija“ tyrimas. Laboratoriniame darbe tiriamos sistemos „Evoliucija“ prototipas yra matematiko Konuėjaus (Conway) sugalvotas matematinis žaidimas „Gyvenimas“. Atsirandančios situacijos žaidimo metu yra panašios į realius procesus, vykstančius gimstant, vystantis ir mirštant gyvoms organizmų kolonijoms. Todėl, „Gyvenimas“ ir „Evoliucija“ priklauso taip vadinamai „modeliuojančių žaidimų“ kategorijai, kuri apima žaidimus, imituojančius procesus, vykstančius realiame gyvenime. „Gyvenimas“

vyksta lauke, suskirstytame į kvadratinius laukelius. Pagrindinė idėja yra ta, kad pradėjus nuo tam tikro ląstelių išsidėstymo lauke, sekama, kaip pradinė kolonija evoliucionuoja laike. Naujų ląstelių gimimą ir senų išmirimą bei išlikimą lemia taip vadinamos „genetinės taisyklės“ (toliau – taisyklės). Kiekvieną lauko langelį supa aštuoni gretimi langeliai: keturi turi bendras kraštines, o keturi – bendras viršūnes. Tada Konuėjaus taisyklės kaimynų išsidėstymui yra tokios:

a) *Išgyvenimas.* Kiekviena gyva ląstelė, kuri turi du ar tris kaimynus, išgyvena ir pereina į kitą kartą.

b) *Mirtis.* Kiekviena ląstelė, turinti daugiau nei tris kaimynus, žūva dėl ląstelių pertekliaus. Kiekviena ląstelė, kuri liečiasi tik su vienu arba nė su vienu kaimynu, žūva dėl vienatvės.

c) *Gimimas.* Jei ląstelių, su kuriomis ribojasi tuščias laukelis, skaičius yra tiksliai lygus trims, tai tame laukelyje sekančioje kartoje atsiranda naujas "organizmas".



11 pav. Konuėjaus taisyklės aštuonių kaimynų konfigūracijai:a) atveju vidurinė ląstelė išliks gyva sekančiame cikle, b) ir c)- žus, d) ir e) atveju kitoje kartoje viduriniame langelyje gims nauja ląstelė.

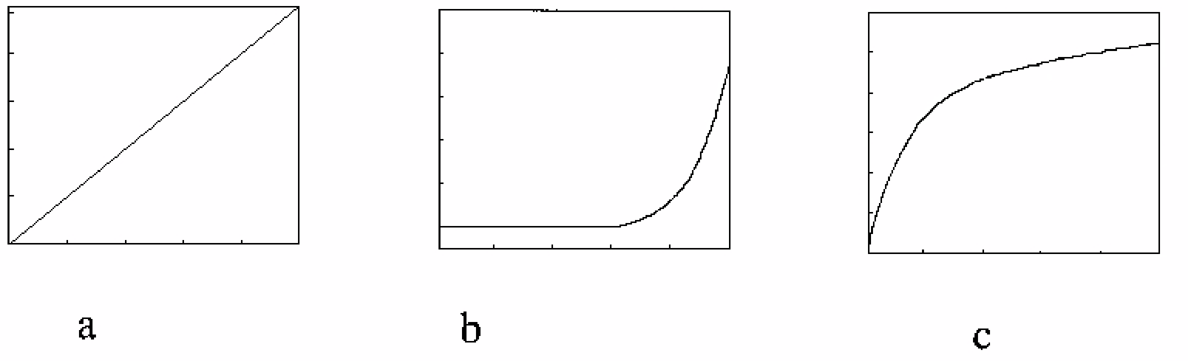
Svarbiausia taisyklė yra ta, kad visi organizmai miršta ir gimsta vienu metu. Visi kartu organizmai sudaro vieną kartą, arba vieną ciklą pradinės konfigūracijos evoliucijoje. Tokios sistemos realus modelis galėtų būti bakterijų kolonija Petri lėkštelėje, kurioje palaikoma pastovi maisto terpė.

Kaip matome, taisyklės yra paprastos, todėl šį žaidimą nesunku sumodeliuoti kompiuterio pagalba. Kuriant kompiuterinį žaidimo variantą buvo įvesta galimybė keisti taisykles – kaimynų skaičių, kuris lemia ląstelės mirimą ir gimimą. Kadangi evoliucijai aprašyti užtenka tik dviejų taisyklių, tai buvo paliktos *Mirties* ir *Gimimo* taisyklės. Be to, įvesta galimybė įskaityti ir tolimesnius ląstelės kaimynus – nutolusius per vieną, du ir daugiau langelių (sąlyga "Kaimynai" dialoginiame lange „Evoliucijos sąlygos“). Tokiu būdu tapo įmanoma keisti dinaminės sistemos, kokia galima pavadinti žaidimą „Gyvenimas“, parametrus.

Vykstant evoliucijai kartais pradinė kolonija palaipsniui išmiršta, t.y. visos ląstelės žūna, bet tai atsitinka ne iš karto. Tokia sistemos elgsena vadinama regresija. Kitu atveju, kolonija gali pereiti į stabilią konfigūraciją (taip vadinamus „ramaus gyvenimo mėgėjus“) arba pereina į svyravimų režimą. Tai pusiausvyrinis arba ciklinis vystymasis. Trečiuoju, katastrofos, atveju pradinė konfigūracija plečiasi be galo. Realiai kolonija plėstis negali, nes plėtimąsi riboja lauko, kuriame vyksta evoliucija, dydis. Eksperimentiškai parodyta, kad Konuėjaus pasiūlytos taisyklės yra pusiausvyra tarp regresijos ir katastrofos. T.y. pagal šias taisykles besiplečianti kolonija labai dažnai pereina į stabilų arba ciklinį režimą. *Šiame laboratoriniame darbe reikia rasti taisykles, atitinkančias kitus du atvejus.*

Pradinės konfigūracijos evoliucijos pobūdis atsispindi laikinėje diagramoje. Šio grafiko abscisių ašyje atidedamas laikas (ciklai), o ordinačių ašyje – ląstelių skaičius. Grafikas evoliucijos metu visą laiką matomas dešiniajame apatiniame ekrano kampe. Jei nustatytos sąlygos, prie kurių sistemoje vyks katastrofa, laikinėje diagramoje matysime pastovų kilimą – ląstelių skaičiaus didėjimą iki tam tikros ribos. Riba, kaip jau buvo minėta, nulemiama

baigtiniu lauko dydžiu, kuris nustatomas dialoginiame lange „Evoliucijos sąlygos“. Grafiko pobūdis gali būti įvairus: tiesė, eksponentė arba logaritminė funkcija.
Kita kraštinė sistemos elgsena – tai regresija (greitas išnykimas). Grafike turėsime krintančią kreivę, kurios pobūdis vėlgi priklauso nuo sąlygų parinkimo. Pažymėtina, kad regresijos atveju nebūtinai žūna visos ląstelės. Bendrai regresija charakterizuojama staigiu organizmų mažėjimu.

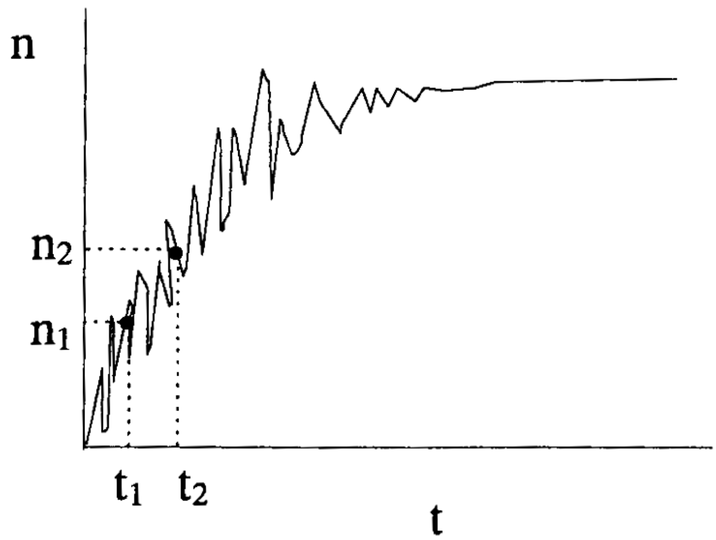


12 pav. Kreivių pavyzdžiai: a-tiesė, b-eksponentė, c-logaritmas.

Paskutinis atvejis— pusiausvyrinės sąlygos. Šiuo atveju sistemos elgesys jau pradeda priklausyti nuo pradinės kolonijos erdvinio išsidėstymo. Prie vienų konfigūracijų sistema išnyksta iš karto, prie kitų – gana ilgą laiką evoliucionuoja, kol pereina prie stabilaus režimo. Pirmaisiais dviem atvejais ši priklausomybė yra silpna. Dėl šios priežasties dinaminės sistemos elgsenos tyrimui geriausia naudoti atsitiktinę, daug ląstelių turinčią pradinę konfigūraciją.
Kiekybiniam elgesio įvertinimui reikia paskaičiuoti kreivės polinkį

$$\alpha = \frac{n_2 - n_1}{t_2 - t_1} \qquad (16),$$

kur n_1 ir n_2 – ląstelių skaičius laiko tarpu t_1 ir t_2 atitinkamai. Parametras α skaičiuojamas tokiu būdu (žr. 13 pav.):
1.Grafike pasirenkami du kreivės taškai. Atstumas tarp taškų turi būti nedidelis (apie dešimtadalį grafiko). Jeigu grafikas nėra lygus (raižytas, kaip parodyta 13 pav.), tai grafikas aproksimuojamas kreive ir taškai atidedami ant šitos kreivės.
2. Randami juos atitinkantys taškai x ašyje t_1 ir t_2 ir y ašyje – n_1 ir n_2.
3. Pagal formulę randame α.



13 pav. α parametro skaičiavimas.

Iš tikrųjų, skaičiuojant α, pradžioje reikia grafiką reguliarizuoti ir tik po to atidėti taškus. Tik tada gaunamas korektiškas rezultatas. Bet mūsų atveju tikslumas nevaidina didelės reikšmės, todėl α skaičiuojamas pagal aukščiau pateiktą planą.
Išvestinę α geriausia skaičiuoti grafiko pradžioje, kai dar nepasireiškia įsisotinimas. Iš tikrųjų α yra ląstelių dauginimosi greitis – kiek ląstelių gimsta (jei $\alpha>0$) arba miršta ($\alpha<0$) per vieną ciklą. $\alpha=0$ atvejis atitinka stabilią būseną. Absoliutinis α dydis rodo katastrofos arba regresijos spartą.
Pastaba: programoje yra galimybė gauti grafiką α (ciklas), kuriame atidėtas ląstelių skaičiaus n+1 cikle ir n cikle skirtumas: $\alpha = n(t+1) - n(t)$.

Antras svarbus evoliucijos parametras yra nusistovėjęs ląstelių tankis

$$\rho = \frac{N}{S} \qquad (17),$$

kur *N* yra ląstelių skaičius plote *S*. Evoliucijos metu programa automatiškai suskaičiuoja ρ reikšmę – kiekvieną ciklą, todėl šio parametro patiems skaičiuoti nereikia. Užtenka išsikviesti $\rho(t)$ grafiką. Žodis „nusistovėjęs“ reiškia, kad vertė imama pasiekus įsisotinimo laipsnį. Parametras ρ rodo parinktų taisyklių palankumą kolonijos vystymuisi, t.y. santykio tarp *mirties ir gimimo taisyklių* įtaką evoliucijai. Kuo ρ didesnis, tuo palankesnės sąlygos ląstelių gimimui ir tuo mažesnė tikimybė, kad mirs senos ląstelės.

Programa sudaro galimybę stebėti ir sistemos fazinius portretus. Pirmojoje fazinėje erdvėje x ašyje atidėtas ląstelių skaičius n-tuoju laiko momentu, o y ašyje – ląstelių skaičius $n+1$ laiko momentu. Ši erdvė iš dalies atspindi evoliucijos proceso kryptingumą: jei esti katastrofos arba regresijos sąlygos, tai ląstelių skaičius pastoviai didės arba mažės, ir fazinėje erdvėje turėsime beveik tvarkingą, sukoncentruotą tam tikroje erdvės dalyje trajektoriją, kuri artėja link stabilaus atraktoriaus. Prie pusiausvyrinių taisyklių (pvz., Konuėjaus), fazinė trajektorija tampa chaotine. Antroji fazinė erdvė pagal savybes panaši į pirmąją. Jos ašyse atidedami mirusių ir gimusių ląstelių skaičiai. Abu faziniai portretai teikia informaciją apie sistemos elgseną.

5. Darbo eiga

Laboratorinis darbas atliekamas kompiuteriu. Visų programų archyvai yra įrašyti viename diskelyje* arba internete kartu su laboratorinio darbo aprašu. Jei dirbama su IBM AT 486 DX ar analogišku kompiuteriu**:** atsiradus DOS-o komandinei eilutei „C:\“ įstatomas diskelis su laboratoriniu darbu, surenkama komanda b:\prad ir nuspaudžiamas „Enter“ klavišas.

Tolimesni veiksmai:
1. Kietajame diske (diskas C) sukuriamas naujas katalogas "LAB", kuriame įrašomos visos išarchyvuotos programos.
2. Išarchyvuojamas failas "fractint.arj". Išarchyvuoti failai patalpinami kietajame diske. Paleidžiama parodomoji programinio paketo "Fractint 18.2" dalis. Demonstracinė dalis dirba visiškai savarankiškai, todėl jos darbo metu spausti klavišų nepatartina, nes tai gali nutraukti demonstraciją. Generuojamo fraktalo pavadinimas parodomas dialoginiame lange „Select Fractal Type“. Pasibaigus demonstracijai, programa „Fractint“ baigia darbą ir visi paketo failai yra pašalinami iš kieto disko.
3. Išarchyvuojamas failas „phaser.arj“ ir paleidžiamas programinis paketas „PHASER“. Atsiradus ekrano apačioje eilutei „Make a choice:“ surenkama klavišų kombinacija „u“, „r“, „demo“ ir nuspaudžiamas „Enter“ klavišas. Paleidžiama demonstracinė programos dalis. Jos metu kiekviename žingsnyje reikia spausti tarpo klavišą (ilgas, baltas klavišas). „Enter“

* kartu su mokomąja priemone: R.Rotomskis Biofotonika. Teorija ir praktiniai darbai. VU KEK, Vilnius, 1997.

klavišo paspaudimas nutraukia demonstraciją. Demonstruojamos lygties pavadinimas užrašomas „SETUP“ lange, po žodžio „Equation“.
4. Į darbo žurnalą perrašomi lygčių pavadinimai, formulės ir nustatoma, kokio tipo atraktorius būdingas lygčiai.
Pastaba: formulėje ženklas „ ' “ reiškia išvestinę.
5. Pasibaigus demonstracijai, ekrano apačioje vėl atsiranda eilutė „Make a choice:“. Jei būtina pakartotinai peržiūrėti demonstracinę dalį, surenkama „rdemo“ ir nuspaudžiamas „Enter“ klavišas. Programos darbas baigiamas nuspaudus klavišus „q“ ir „y“. Išėjus iš programos, visi paketo failai automatiškai pašalinami iš kietojo disko.
6. Išarchyvuojamas failas „evol.arj“ ir paleidžiamas programinis paketas „Evoliucija“. Ištiriama dinaminė sistema „Evoliucija“. Išėjus iš programos, visi failai išvalomi, o darbo pradžioje sukurtas katalogas kietajame diske panaikinamas. Dinaminės sistemos tyrimo eiga:
1. Parenkamas mažiausias mastelis. Tam nuspaudžiame klavišą „F5“. Visas laukas, apibrėžtas raudona linija, turi tilpti ekrane.
2. Įsitikinama, kad evoliucijos sąlygos atitinka Konuėjaus taisykles. Nuspaudus „F8“ klavišą, atidaromas dialoginis langas „Evoliucijos sąlygos“. Laukuose „Gimimo sąlygos“ ir „Mirimo sąlygos“ turi būti įrašytos reikšmės „3“ ir „0,1,4-8“, atitinkamai. Jei taip nėra, laukų reikšmės ištaisomos. Po lango laukus judama „Tab“ (žemyn) ir „Shift“+„Tab“ (aukštyn) klavišais. Pabaigus įvedimą, atsistojama ant lauko „Ok“ ir nuspaudžiamas „Enter“ klavišas. Jei viskas buvo įvesta gerai, langas užsidaro ir atsimenamos laukų reikšmės. Jei buvo įvestos neteisingos reikšmės (pvz., ne to ženklo arba vietoj skaičių įrašytos raidės), langas neužsidaro, o kursorius pereina į neteisingą reikšmę turintį laukelį. Jei norima išeiti iš lango be reikšmių įsiminimo, spaudžiamas „Esc“ klavišas.
Pastaba: visų laukelių paskirtį lange „Evoliucijos sąlygos“ galima sužinoti pasirinkus laukelį „Pagalba“ ir nuspaudus „Enter“ (iš pagalbos išeinama paspaudus „Esc“ klavišą). Darbo metu keičiamos tik „Gimimo“ ir „Mirimo sąlygos“ laukelių reikšmės.
3. Sukuriama atsitiktinė konfigūracija: kartu nuspaudžiami klavišai „Alt“ ir „r“. Atsiradus mėlynam stačiakampiui, kursoriaus valdymo klavišais (pažymėti rodyklėmis) išplečiamas stačiakampis tiek, kad jis užimtų apie ketvirtadalį lauko. Jei reikia sumažinti stačiakampį, naudojamasi tais pačiais valdymo klavišais, tik papildomai nuspaudus „Shift“ klavišą. Nuspaudus „Enter“ klavišą, apibrėžtame stačiakampio lauke atsitiktinai išmėtomos ląstelės su ląstelės buvimo laukelyje tikimybe 0,5. Todėl pradinis ląstelių tankis turi būti apie 0,5.
4. Gauta figūra išsaugojama. Tam nuspaudžiamas „F2“ klavišas ir įvedamas figūros pavadinimas (pvz., „random“). Įvestas pavadinimas užsirašomas.
5. Išsaugojus figūrą nuspaudžiamas „F9“ klavišas ir paleidžiamas kolonijos vystymasis. Dešiniajame apatiniame ekrano kampe stebima laikinė evoliucijos diagrama – N(ciklas) grafikas, kur N – ląstelių skaičius. Nusistovėjus pusiausvyrinei būsenai – grafike horizontali tiesė, arba ciklui – pjūklinis grafikas, vystymasis sustabdomas nuspaudus klavišą „Esc“ arba pakartotinai „F9“.
6. Sustabdžius vystymąsi, pereinama į grafikų režimą. Tam nuspaudžiama „Alt“+„g“ klavišų kombinacija. Ekrane turi pasirodyti pilnas N(ciklas) grafikas. Kitų grafikų pavaizdavimui reikia nuspausti vieną iš skaitinių klavišų. Klavišą "1" atitinka grafikas N(ciklas), klavišą "2"- fazinis portretas $N_{ciklas+1}(N_{ciklas})$, kur N_{ciklas} – ląstelių skaičius po n ciklo, o $N_{ciklas+1}$ – ląstelių skaičius po n+1 ciklo; klavišą „3“- fazinis portretas $N_{mir}(N_{gim})$, kur N_{mir} , N_{gim} – mirusių ir gimusių ląstelių skaičius n cikle; klavišą „4“- grafikas ρ(ciklas), kur ρ- ląstelių tankis; klavišą „5“- grafikas α(ciklas). Iš grafiko N(ciklas) apskaičiuojama α reikšmė pagal (16) formulę. Iš grafiko ρ(ciklas) randama ρ reikšmė nusistovėjus pusiausvyrai. Darbo žurnale užrašomas ir fazinių portretų (2 ir 3 grafikai) pobūdis.
Iš grafikų režimo išeinama nuspaudus „Esc“ klavišą.
Pastaba: Naudojant „Tab“ klavišą, galima keisti taškų vaizdavimą grafike nuo pavienių taškų iki linijom sujungtų taškų ir atvirkščiai.

7. Išvalomas laukas. Tam paspaudžiamas „F4“ klavišas ir pasirodžius dialoginiam langui, pereinama į langelį „Taip“ (paspaudus klavišą „Tab“) ir paspaudžiamas „Enter“.
8. Punktai 3-7 pakartojami du kartus su skirtingomis atsitiktinėmis pradinėmis konfigūracijomis. Gautos α ir ρ reikšmės suvidurkinamos.
9. Pakeičiamos evoliucijos taisyklės (žr. 4 punktą). Pradžioje patogu pakeisti vieną iš taisyklių – *gimimo* arba *mirimo.* Įvedamos reikšmės turi būti iš intervalo nuo 0 iki 8 (tiek gali būti kaimynų). Jeigu sąlygoje yra daugiau kaip vienas skaičius, tai įvedami skaičiai skiriami kableliu. Skaičių intervalas nurodomas per brūkšnį. Pvz., Konuėjaus *mirimo* sąlyga yra „0,1,4–8“. Tai reiškia, kad evoliucijos eigoje ląstelės, kurios turi nulį, vieną, keturis, penkis, šešis, septynis arba aštuonis kaimynus, žūna. *Gimimo* sąlyga yra tik vienas skaičius – „3“ (tai atitinka 3 kaimynus).
10. Į išvalytą lauką įterpiama pirma iš išsaugotų figūrų. Tam nuspaudžiamas „F3“ klavišas ir įvedamas figūros pavadinimas.
11. Pradinė konfigūracija paleidžiama vystytis („F9“ klavišas) ir stebima laikinė diagrama. Nusistovėjus pusiausvyrai (įsisotinimui katastrofos atveju, išnykimui regresijos atveju) evoliucija sustabdoma ir ištiriami grafikai (6 punktas). Randamos α ir ρ reikšmės.
12. Punktai 10 ir 11 pakartojami su visomis pradinėmis išsaugotomis figūromis. Rezultatai įrašomi į lentelę:

Sąlygos	α	$<\alpha>$	ρ	$<\rho>$	Faziniai portretai
Gim: 3 Mir: 0,1,4-8	...	...	...	...	...

13. Surandamos trys regresijos ir trys katastrofos atvejus atitinkančios sąlygos. Kiekvienai sąlygai kartojami 9 – 12 punktai.
Pastaba: Darbo metu galima išsaugoti gaunamus vaizdus ir grafikus. Kolonijos vaizdas išsaugojamas vienu metu spaudžiant „Alt“ ir „p“ klavišus. Pasirodžiusiame lange įvedamas failo vardas. Sukurtas failas turi praplėtimą „*.plt“ ir yra tame pačiame kataloge kaip ir programa „Evoliucija“. Grafiko duomenys išsaugojami grafikų režime nuspaudus „s“ raidę. Sukurtas failas turi praplėtimą „*.dat“.

Pilnas aprašymas, kaip naudotis programa „Evoliucija“, yra programos pagalboje. Pagalba iškviečiama spaudžiant klavišą „F1“.

6. Kontroliniai klausimai

1. Kas yra dinaminė sistema ?
2. Kas yra fazinė erdvė? Jos reikšmė tiriant dinamines sistemas.
3. Atraktoriaus sąvoka. Atraktorių rūšys.
4. Chaotinių sistemų savybės.
5. Dinaminių sistemų kiekybinės charakteristikos.
6. Kas yra fraktalas?
7. Fraktalinės dimensijos sąvokos. Dimensijų rūšys.
8. Fraktalų sudarymo metodai.

7. Literatūra

1. Peitgen HO, Jurgens H, Saupe D. Chaos and fractals: New frontiers of science. Stringer-Verlag, 1992.
2. М. Гарднер. Крестики – нолики. М., Мир, 1988.
3. Г.М. Заславский, Р.З. Сагдеев. Введение в нелинейную физику: От маятника до турбулентности и хаоса. М., Наука, 1988.
4. М. Гарднер. От мозаик Пенроуза к надежным цифрам. М., Мир, 1993.
5. В мире науки. М., Мир, 1987, Nr. 2, 3, 9; 1988, Nr. l, 5; 1989, Nr. 2, 4, 9; 1990, Nr. 4, 6, 7, 10; 1991, Nr. 11; 1993, Nr. 2-3.