



VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS  
KOMPIUTERIJOS KATEDRA

Metodinė medžiaga studentams

**Reikalavimai ir rekomendacijos rašto darbams**

Vilnius  
2015

# Turinys

<b>1. Iliustracijos, lentelės ir pseudokodas</b>	<b>3</b>
1.1. Reikalavimai iliustracijoms ir lentelėms . . . . .	3
1.1.1. Reikalavimai iliustracijų aprašymui . . . . .	3
1.1.2. Reikalavimai iliustracijų apipavidalinimui . . . . .	3
1.1.3. Kita informacija apie iliustracijas ir lenteles . . . . .	4
1.2. Iliustracijų pavyzdžiai . . . . .	4
1.2.1. Pirmas iliustracijos pavyzdys . . . . .	4
1.2.2. Antras iliustracijos pavyzdys . . . . .	5
1.2.3. Trečias iliustracijos pavyzdys . . . . .	6
1.2.4. Ketvirtas iliustracijos pavyzdys . . . . .	7
1.3. Iliustracijos ir lentelės pavyzdys . . . . .	8
1.4. Lentelės pavyzdys . . . . .	11
1.5. Pseudokodo pavyzdys . . . . .	12
<b>2. Šablonas</b>	<b>14</b>
2.1. Darbų tipai . . . . .	14
2.2. Dokumento struktūra . . . . .	14
2.3. Literatūros šaltiniai . . . . .	15
2.4. Pseudo-kodas ir programinio kodo dalys . . . . .	15
2.5. Paveikslėlių paruošimas . . . . .	15
2.6. Sudėtingesnės lentelės . . . . .	17
<b>Literatūros šaltiniai</b>	<b>18</b>

# 1. Iliustracijos, lentelės ir pseudokodas

## 1.1. Reikalavimai iliustracijoms ir lentelėms

### 1.1.1. Reikalavimai iliustracijų aprašymui

Visos darbe naudojamos iliustracijos turi būti sunumeruotos ir privalo turėti aprašymą (angliškai “caption”, trumpą tekstą iliustracijos apačioje), kuriame būtų paminėta kas šioje iliustracijoje (ar kiekvienoje ją sudarančioje dalyje, jeigu iliustraciją sudaro, pavyzdžiui, keletas skirtingų grafikų) pavaizduota. Jeigu tai turi prasmę, iliustracijos aprašyme pateikite ir papildomą informaciją, pavyzdžiui:

- modelio pavadinimą arba numerį (jeigu darbe analizuojate daugiau nei vieną modelį);
- algoritmo, kurio veikimo rezultatai atspindėti iliustracijoje, pavadinimą arba numerį (jeigu darbe analizuojate daugiau nei vieną algoritmą konkrečiam modeliui spręsti);
- apdorojamų duomenų pavadinimą arba numerį (jeigu darbe analizuojate daug skirtingų duomenų);
- išvardinkite konkrečias skaitines visų parametrų (tiek modelio, tiek algoritmo) vertes, su kuriomis buvo gautas iliustracijoje matomas rezultatas. Jeigu konkretus parametras nėra bedimensinis, nepamirškite nurodyti ir jo dimensiją (pavyzdžiui, sekundės, metai, metrai, centimetrai, gramai, gigaflopai ir pan., naudodami visuotinai priimtus trumpinius). Jeigu darbe analizuojate tik vieną parametrų rinkinį, jį nurodykite ne iliustracijos aprašyme, bet pagrindiniame darbo tekste;
- kitą svarbią su iliustracija susijusią informaciją.

Visos iliustracijos privalo būti cituojamos darbe. Darbo tekste pateikite trumpą iliustracijoje gauto rezultato interpretaciją – kokias išvadas galima padaryti iš šios iliustracijos, ją galbūt palyginant su kitomis iliustracijomis, nurodant lyginamų iliustracijų numerius.

### 1.1.2. Reikalavimai iliustracijų apipavidalinimui

Atkreipkite dėmesį į tinkamą iliustracijų apipavidalinimą:

- iliustracija neturi būti pernelyg sumažinta ar išdidinta. Esant galimybei, iliustraciją sukonstruokite iš dviejų dalių (iliustracijos kairėje ir iliustracijos dešinėje), išnaudodami visą puslapio plotį. Toliau esančiuose pavyzdžiuose pademonstruota kokio dydžio turėtų būti iliustracijos;
- grafikų kreivės turi būti pakankamai storos, iliustracija neturi būti perkrauta kreivių gausa (esant reikalui, vietoje vienos iliustracijos galima pateikti dvi ar daugiau);
- ašių kintamieji, gradacija bei kita iliustracijoje pateikiama informacija (tekstas, rodyklės ir pan.) turi būti aiškūs, nesmulkūs. Atkreipkite dėmesį, kad grafinė iliustracijos byla maketavimo metu (ją įdedant į darbo tekstą) dažnai ženkliai sumažinama. Peržiūrint iliustraciją darbo tekste, visa joje esanti tekstinė informacija turi būti atvaizduojama nemažesniu nei pagrindiniame darbo tekste naudojamu šrifto dydžiu, grafikų kreivės bei kita grafinė informacija privalo būti optimalaus storio, lengvai išžiūrima ir atskiriama.
- turi būti paminėti ašių kintamųjų pavadinimai ir, jeigu atitinkamas kintamasis nėra bedimensinis – dimensijos (pavyzdžiui, sekundės, metai ir pan., naudojant visuotinai priimtus trumpinius).

### 1.1.3. Kita informacija apie iliustracijas ir lenteles

Siekdami geriausios įmanomos iliustracijų grafinės kokybės, visada jas kurkite tik *vektoriniame* formate (o ne rastriniame, taigi JPG formato niekada nenaudokite – jis pasižymi pikselizacija bei kitais artefaktais). Dažnas klausimas – kurį iš vektorinės grafikos formatų (iliustracijų grafinių byloms) pasirinkti: PostScript (PS), Encapsulated PostScript (EPS) ar Portable Document Format (PDF)? Rekomenduojame naudoti PDF formatą, tokio pasirinkimo priežastys nurodytos, pavyzdžiui, čia: <http://www.adobe.com/print/features/psvspdf>, taip pat <http://tex.stackexchange.com/questions/2092/which-figure-type-to-use-pdf-or-eps>

Jeigu darbą planuojate atspausdinti (prieš įrišdami) nespaltvotą spausdintuvu, nenaudokite spalvotų iliustracijų. Atspausdinę darbą patikrinkite, ar visos iliustracijos ant popieriaus atrodo taip pat gerai kaip monitoriaus ekrane.

Dažna klaida kai, vaizduojant kreivės priklausomybę nuo kintamojo (pavyzdžiui, valiutų kurso priklausomybę nuo laiko), iliustracijos abscisių ašyje atidedamos ne kintamojo reikšmės (pavyzdžiui, laiko reikšmės: einamųjų metų mėnesiai), bet kreivės diskrečių reikšmių masyvo indeksai (dažniausiai neperteikiantys jokios svarbios informacijos).

Taip pat atkreipkite dėmesį, kad modeliuojant fizikinius, cheminius ir panašius reiškinius dimensijos turi būti suderintos – skaičiavimų metu visų parametrų ir ieškomų dydžių skaitinės vertės turi būti apibrėžtos vieningoje sistemoje (pavyzdžiui, SI sistemoje – tuomet 2  $\mu\text{m}$  skaičiavimuose įvedama kaip  $2 \times 10^{-6}$  m, nes SI sistemos vienetas yra metras, o ne mikrometras).

Jeigu reikia, tyrimų rezultatus galite pateikti ir kaip skaičių lenteles. Lentelės taip pat turi būti numeruojamos, turėti savo aprašymus ir cituojamos darbe. Pagal mokslinėje literatūroje nustovėjusias taisykles, lentelės aprašymas patalpinamas lentelės viršuje (skirtingai nuo iliustracijos aprašymo, kurio vieta po iliustracija).

Toliau pateikti tinkamai apipavidalintų iliustracijų ir lentelės (bei jų aprašymo darbo tekste) pavyzdžiai. 1.5 skyrelyje rasite pseudokodo pavyzdį.

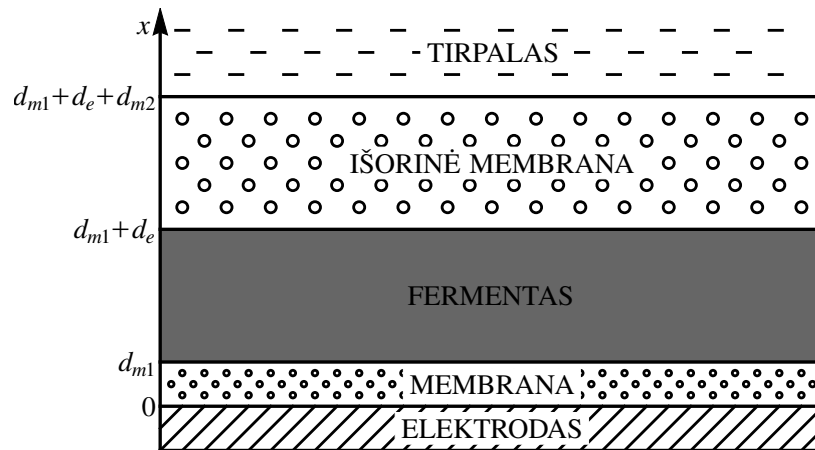
## 1.2. Iliustracijų pavyzdžiai

### 1.2.1. Pirmas iliustracijos pavyzdys

Modeliuosime elektrocheminį biojutiklį (žr. 1 pav.), sudarytą iš elektrodo, elektrodą gaubiančios membranos (kurios storis  $d_{m1} \geq 0$ ), fermento sluoksnio (kurio storis  $d_e > 0$ ) ir išorinės membranos (kurios storis  $d_{m2} > 0$ ). Atskiru atveju, elektrodą gaubiančios membranos (dar kartais vadinamos diskriminacine membrana) gali nebūti (jeigu  $d_{m1} = 0$ ). Biojutiklis panardintas į tirpalą, kuriame palaikoma pastovi substrato (tam tikros cheminės medžiagos, pavyzdžiui, gliukozės, kurios koncentraciją matuoja biojutiklis) koncentracija. Substratui (kuris nėra elektrochemiškai aktyvus) reaguojant su fermentu (fermento sluoksnyje) substratas virsta produktu (kita chemine medžiaga, pavyzdžiui, vandenilio peroksidu). Taip pat reakciją gali įtakoti, joje dalyvauti ar jos eigoje susidaryti kiti cheminiai junginiai. Produktas yra elektrochemiškai aktyvi medžiaga, taigi generuoja (ant elektrodo) elektros srovę, kurią matuodami vertiname substrato koncentraciją tirpale.

Biojutiklį sudarančių sluoksnių (membranų, fermento sluoksnio) storiai (taigi ir paties biojutiklio gabaritai) matuojami mikronais (sinonimas mikrometrai, žymėjimas  $\mu\text{m}$ ). Šie storiai gali būti lygūs nuo kelių iki keliolikos mikronų. Palyginimui – tokie objektai kaip popieriaus lapo storis, žmogaus plauko diametras gali svyruoti nuo 50 iki 100 mikronų. Taigi, biojutiklis yra išties “mažas” prietaisas.

Matematinis kintamasis  $x$  žymi koordinatę biojutiklyje (atstumą iki elektrodo), o  $t \geq 0$  yra laikas.



1 pav. Elektrocheminio biojutiklio sandaros schema. Rodykle pažymėta kintamojo  $x$  ašis.

Išorinė membrana dažniausiai būna pagaminta iš polimerinio pluošto. Labai dažnai tai būna celofano plėvelės. Kartais naudojamos tankios polimerinės plėvelės, kuriose dirbtinai padaryta skylių. Pavyzdžiui, argono branduoliais prašaudyta teflono plėvelė – po elektroniniu mikroskopu tokia plėvelė atrodo kaip medžioklinio šautuvo šratais sušaudytas faneros lakštas.

### 1.2.2. Antras iliustracijos pavyzdys

Kraštinės sąlygos substrato koncentracijai  $S$  modeliavimo srities  $0 \leq x \leq d_{m1} + d_e + d_{m2}$  kraštuose formuluojamos atsižvelgiant į tai, kad į elektrodą substratas įsiskverbti negali, o tirpale (į kurį panardintas biojutiklis) yra palaikoma pastovi substrato koncentracija  $S_0$ :

$$\left. \frac{\partial S}{\partial x} \right|_{x=0} = 0, \quad S(x = d_{m1} + d_e + d_{m2}, t) = S_0, \quad t > 0. \quad (1.1)$$

Be to, darome prielaidą, kad ant elektrodo produktas labai greitai reaguoja. Todėl produkto koncentracija  $P(x = 0, t)$  (ant briaunos su elektrodu) visą laiką bus lygi nuliui.

Modeliuodami nagrinėsime du skirtingus atvejus.

**Difunduojančio į tirpalą produkto atvejis.** Tarkime, išorinės membranos sienelė leidžia produktui difunduoti į tirpalą. Tuomet produkto koncentracija  $P$  intervalo  $0 \leq x \leq d_{m1} + d_e + d_{m2}$  kraštuose tenkina sąlygas

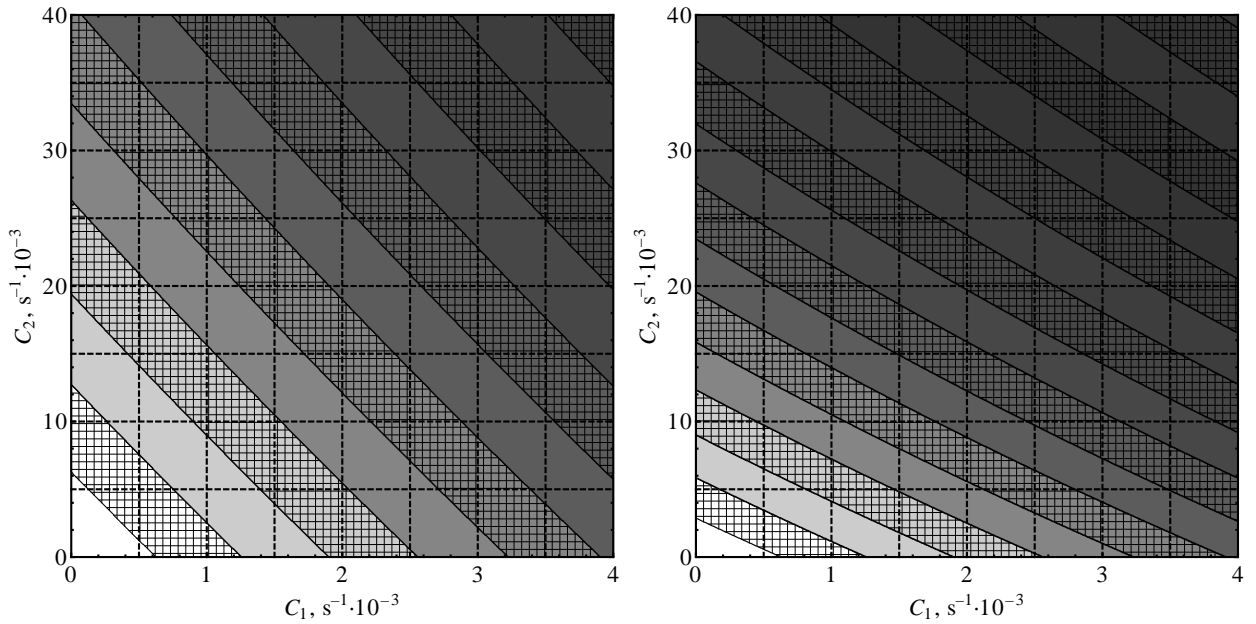
$$P(x = 0, t) = 0, \quad P(x = d_{m1} + d_e + d_{m2}, t) = 0, \quad t > 0. \quad (1.2)$$

**Atvejis kai produktas negali patekti į tirpalą.** Jeigu išorinės membranos sienelė nepralaidi produktui, kraštinėse sąlygose (1.2) antroji sąlyga turi būti pakeista reikalavimu, kad srities krašte funkcijos išvestinė turi būti lygi nuliui (nėra produkto ištekėjimo). Taigi, tokiu atveju produkto koncentracijai  $P$  formuluojame tokias kraštines sąlygas:

$$P(x = 0, t) = 0, \quad \left. \frac{\partial P}{\partial x} \right|_{x=d_{m1}+d_e+d_{m2}} = 0, \quad t > 0. \quad (1.3)$$

Jeigu  $C_1 > 0$  ir/arba  $C_2 > 0$ , tai dėl degradacijos biojutiklio atsakas (nusistovėjęs laike srovės tankis)  $I$  sumažėja (palyginus su atveju be degradacijos  $C_1 = 0 \text{ s}^{-1}$ ,  $C_2 = 0 \text{ s}^{-1}$ ). Ištirsime šio sumažėjimo priklausomybę nuo parametrų  $C_1$  ir  $C_2$ .

Iliustracijoje 2 pav. grafiškai išskirtos parametų erdvės ( $C_1, C_2$ ) dalys, kurioje biojutiklio atsakas sumažėja nuo 0% iki 1% (balta sritis), nuo 1% iki 2% (balta užbrūkšniuota sritis), nuo 2% iki 3% (šviesiai pilka sritis), nuo 3% iki 4% (šviesiai pilka užbrūkšniuota sritis), ir taip toliau – vis tamsėjant srities fonui.



2 pav. Biojutiklio atsako  $I$  degradacija (sumažėjimas) koordinatų plokštumoje ( $C_1, C_2$ ). Iliustracija kairėje: kraštinės sąlygos (1.2) (difunduojančio į tirpalą produkto atvejis). Iliustracija dešinėje: kraštinės sąlygos (1.3) (atvejis kai produktas negali patekti į tirpalą). Modelio ir algoritmo parametrai:  $N = 100$ ,  $\tau = 0.01$  s,  $d_{m1} = 2 \mu\text{m}$ ,  $d_e = 9 \mu\text{m}$ ,  $d_{m2} = 10 \mu\text{m}$ ,  $D_{S_{m1}} = 6 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_{P_{m1}} = 5 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_{S_e} = 22 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_{P_e} = 20 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_{S_{m2}} = 7 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_{P_{m2}} = 6 \mu\text{m}^2 \text{s}^{-1}$ ,  $V_{max} = 0.3 \text{ mmol m}^{-3} \text{s}^{-1}$ ,  $K_M = 0.23 \text{ mol m}^{-3}$ ,  $S_0 = 0.07 \text{ mol m}^{-3}$ .

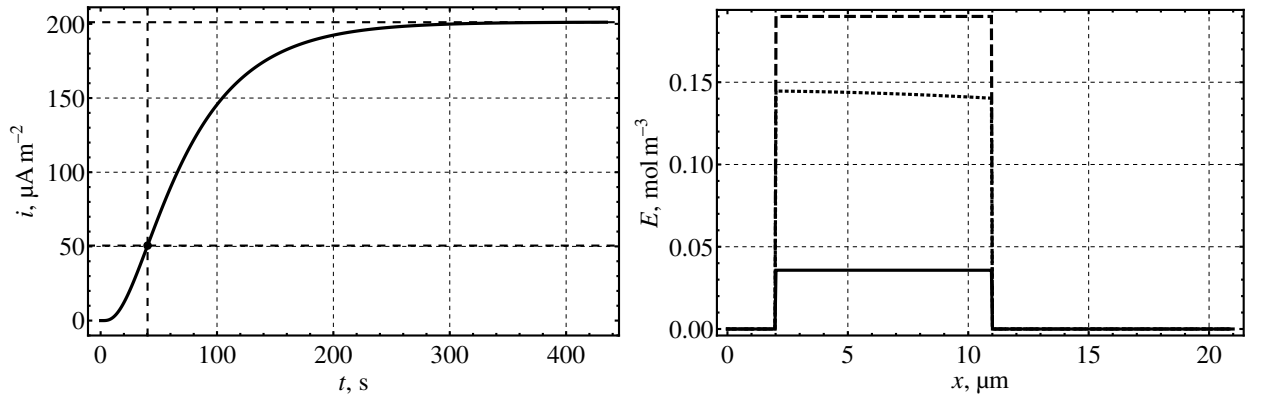
Matome, kad skaičiuojant su 2 pav. nurodytais parametrais rezultatai gana *ženkliai priklauso nuo to su kuriomis kraštinėmis sąlygomis apibrėžtas matematinis modelis*: kraštinėmis sąlygomis (1.2) ar (1.3).

Taip pat pastebėjime (žr. 2 pav.), kad aiškinantis kokios (skirtingos) parametų  $C_1 > 0$  ir  $C_2 > 0$  vertės gali sąlygoti kažkokį vieną ir tą patį (fiksotą)  $I$  sumažėjimą, gaunamas *beveik tiesinis* ryšys tarp parametų  $C_1$  ir  $C_2$ . Tiesa, iliustracijoje matomos  $I$  (vienodo) sumažėjimo izolinijos nėra absoliučiai idealios tiesės.

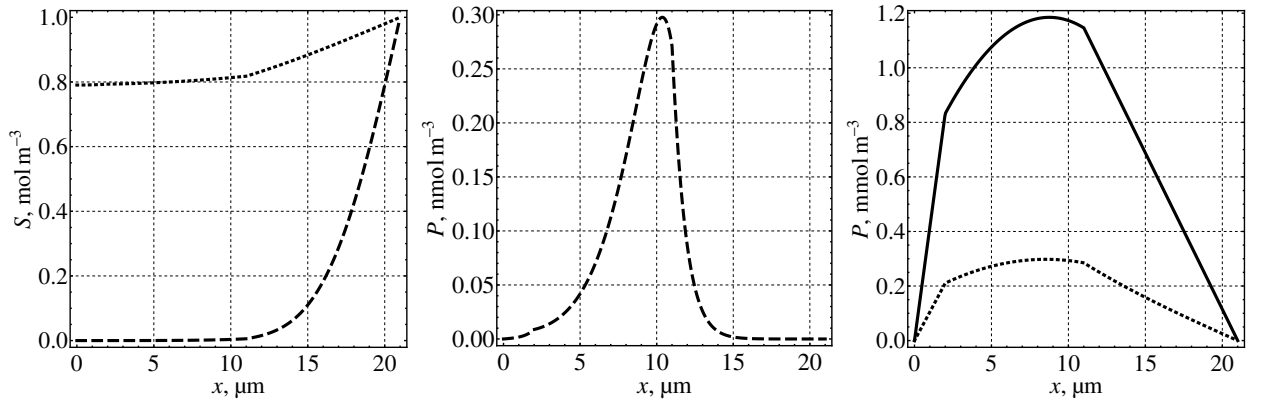
Verta pastebėti ir tai, kad parametro  $C_1$  įtaka degradacijai *keletą kartų didesnė* už parametro  $C_2$  įtaką. Tiesa, kiek būtent kartų – priklauso ir nuo kraštinių sąlygų pasirinkimo. Skaičiuojant su kraštinėmis sąlygomis (1.2), matyti (žr. 2 pav. iliustraciją kairėje), kad parametų pasirinkimai  $C_1 = 4 \text{ m(s}^{-1})$ ,  $C_2 = 0 \text{ m(s}^{-1})$  ir  $C_1 = 0 \text{ m(s}^{-1})$ ,  $C_2 = 40 \text{ m(s}^{-1})$  sukelia apytiksliai vienodą nusistovėjusio srovės tankio  $I$  sumažėjimą, taigi parametro  $C_1$  įtaka apie 10 kartų didesnė už parametro  $C_2$  įtaką. Tačiau kraštinių sąlygų (1.3) pasirinkimas rodo (žr. 2 pav. iliustraciją dešinėje), kad  $C_1$  įtaka tik apie 5 kartus didesnė už  $C_2$  įtaką.

### 1.2.3. Trečias iliustracijos pavyzdys

Skaičiavimai vykdyti su matematinio modelio ir algoritmo parametrais: kraštinės sąlygos (1.2) (difunduojančio į tirpalą produkto atvejis),  $N = 1000$ ,  $h = 0.021 \mu\text{m}$ ,  $\tau_{max} = 0.01$  s,  $d_{m1} = 2 \mu\text{m}$ ,



3 pav. Iliustracija kairėje: srovės tankio  $i(t)$  priklausomybė nuo laiko  $t$  (punkturu paryškinti laiko momentas  $t = t_{slopeMAX}$  ir biojutiklio atsakas  $I$ ). Iliustracija dešinėje: fermento koncentracijos  $E(x, t)$  priklausomybės nuo biojutiklio koordinatės  $x$ , laiko momentu  $t = 1$  s (brūkšninė kreivė), laiko momentu  $t = t_{slopeMAX}$  (taškinė kreivė) ir laiko momentu  $t = t_I$  (ištiesinė kreivė).



4 pav. Iliustracija kairėje: substrato koncentracijos  $S(x, t)$  priklausomybės nuo biojutiklio koordinatės  $x$ , laiko momentu  $t = 1$  s (brūkšninė kreivė) ir laiko momentu  $t = t_{slopeMAX}$  (taškinė kreivė). Iliustracija centre: produkto koncentracijos  $P(x, t)$  priklausomybė nuo biojutiklio koordinatės  $x$ , laiko momentu  $t = 1$  s (brūkšninė kreivė). Iliustracija dešinėje: produkto koncentracijos  $P(x, t)$  priklausomybės nuo biojutiklio koordinatės  $x$ , laiko momentu  $t = t_{slopeMAX}$  (taškinė kreivė) ir laiko momentu  $t = t_I$  (ištiesinė kreivė).

$d_e = 9 \mu\text{m}$ ,  $d_{m2} = 10 \mu\text{m}$ ,  $C_1 = 0 \text{ s}^{-1}$ ,  $C_2 = 0 \text{ s}^{-1}$ ,  $S_0 = 1 \text{ mol m}^{-3}$ ,  $E_0 = 0.19 \text{ mol m}^{-3}$ ,  $k_{+1} = 0.015 \text{ m}^3 \text{ mol}^{-1} \text{ s}^{-1}$ ,  $k_{-1} = 0.0015 \text{ s}^{-1}$ ,  $k_{+3} = 0.002 \text{ s}^{-1}$ ,  $k_{-3} = 0.002 \text{ m}^3 \text{ mol}^{-1} \text{ s}^{-1}$ ,  $n_e = 1$ .

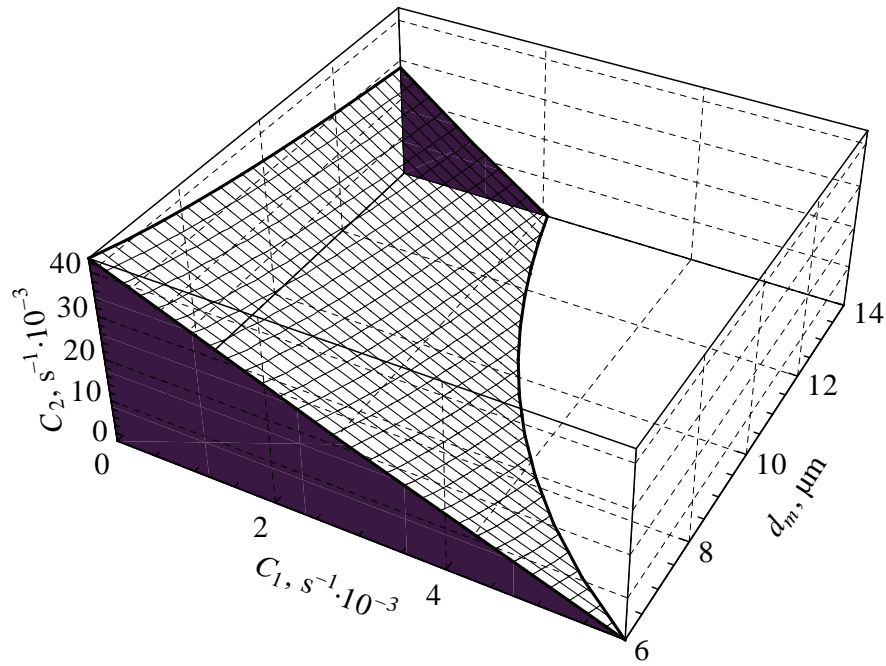
3 pav. ir 4 pav. pateikiami rezultatai, kai difuzijos koeficientai yra lygūs:  $D_{S_{m1}} = 6 \mu\text{m}^2 \text{ s}^{-1}$ ,  $D_{P_{m1}} = 5 \mu\text{m}^2 \text{ s}^{-1}$ ,  $D_{S_e} = 22 \mu\text{m}^2 \text{ s}^{-1}$ ,  $D_{P_e} = 20 \mu\text{m}^2 \text{ s}^{-1}$ ,  $D_{S_{m2}} = 7 \mu\text{m}^2 \text{ s}^{-1}$ ,  $D_{P_{m2}} = 6 \mu\text{m}^2 \text{ s}^{-1}$ .

Ivertinta biojutiklio atsako reikšmė  $I = 201.1 \mu\text{A m}^{-2}$ , pasiekus laiko momentą  $t_I = 434.6$  s. Srovės tankis intensyviausiai augo laiko momentu  $t_{slopeMAX} = 40.3$  s.

#### 1.2.4. Ketvirtas iliustracijos pavyzdys

Ištirsime biojutiklio atsako  $I$  penkių procentų degradacijos *izoliniją* (vienodo  $I$  santykinio sumažėjimo kreivę) koordinatų plokštumoje ( $C_1, C_2$ ). Izolinijai priklausantys taškai nusako tokias parametrų  $C_1$  ir  $C_2$  vertes, su kuriomis nusistovėjęs srovės tankis  $I$  sumažėja tiksliai 5% (palyginus su  $I$  reikšme, gaunama atveju be degradacijos  $C_1 = 0 \text{ s}^{-1}$ ,  $C_2 = 0 \text{ s}^{-1}$ ).

Ši (ir kitos) izolinija matoma 2 pav. Šiame skyrelyje papildomai panagrinėsime minėtos izo-



5 pav. Biojutiklio atsako  $I$  penkių procentų degradacijos izolinijos (apibrėžtos koordinačių plokštumoje  $(C_1, C_2)$ ) priklausomybė (paviršius) nuo membranos storio  $d_m$ . Modelio ir algoritmo parametrai:  $N = 100$ ,  $\tau = 0.01$  s,  $d_e = 9$   $\mu\text{m}$ ,  $D_{S_e} = 22$   $\mu\text{m}^2 \text{s}^{-1}$ ,  $D_{P_e} = 20$   $\mu\text{m}^2 \text{s}^{-1}$ ,  $D_{S_m} = 7$   $\mu\text{m}^2 \text{s}^{-1}$ ,  $D_{P_m} = 6$   $\mu\text{m}^2 \text{s}^{-1}$ ,  $V_{max} = 0.3$   $\text{mmol m}^{-3} \text{s}^{-1}$ ,  $K_M = 0.23$   $\text{mol m}^{-3}$ ,  $S_0 = 0.07$   $\text{mol m}^{-3}$ .

linijos (apibrėžtos koordinačių plokštumoje  $(C_1, C_2)$ ) priklausomybę nuo membranos storio  $d_m$  intervale  $d_m \in [6, 14]$   $\mu\text{m}$ .

Skaičiavimų rezultatas pavaizduotas 5 pav. Matyti, kad sritis po izolinija (5 pav. paryškinta pilkos spalvos fonu) mažėja, didinant  $d_m$ . Su parametrais  $C_1 \geq 0$ ,  $C_2 \geq 0$  iš šios srities (esančios po izolinija)  $I$  degradacija neviršija 5%. Pastebėkime, kad visiems  $d_m \in [6, 14]$   $\mu\text{m}$  izolinija apytiksliai išlaiko tiesės pavidalą, tačiau siekiant ją aproksimuoti dar tiksliau reikėtų naudoti parabolinę kreivę.

Užrašysime izolinijos aproksimacijos tiesę analitines priklausomybes nuo parametro  $d_m$ , galiojančias intervale  $d_m \in [6, 14]$   $\mu\text{m}$ , su 5 pav. nurodytais matematinio modelio parametrais. Remdamiesi skaičiavimo rezultatais, pateiktais 5 pav., mažiausių kvadratų (angliškai “least squares”) metodu išvedame tokias analitines išraiškas:

$$C_2 = K(d_m) C_1 + M(d_m), \quad C_1 \geq 0, \quad C_2 \geq 0, \quad (1.4)$$

$$K(d_m) = -0.791771 d_m / \mu\text{m} - 2.48837, \quad (1.5)$$

$$M(d_m) = \left( -0.583423 d_m / \mu\text{m} + 28.3346 + \frac{110.039}{d_m / \mu\text{m}} \right) / 1000. \quad (1.6)$$

Kiekvienam  $d_m$ , izolinija koordinačių plokštumoje  $(C_1, C_2)$  eina per taškus

$$(C_1 = 0, C_2 = M(d_m)) \text{ s}^{-1}, \quad \left( C_1 = -\frac{M(d_m)}{K(d_m)}, C_2 = 0 \right) \text{ s}^{-1}.$$

### 1.3. Ilustracijos ir lentelės pavyzdys

Siekiant dar labiau optimizuoti Cooley–Tukey greitosios Furje transformacijos algoritmą, programuojant galima atsisakyti rekursijos, o duomenis (signalų reikšmes  $f_j$ ) iš karto išdėstyti tokia tvarka, kokia jie būtų sumuojami paskutiniame (giliausiam) rekursijos lygyje.



Taip pat verta iš anksto suskaičiuoti ir laikyti masyvuose nepriklausančias nuo apdorojamo signalo reikšmių  $f_j$  kompleksines konstantas  $W_m^k$ , tik tokiems sveikiesiems  $m$  ir  $k$ , kuriems šių konstantų iš tiesų prireiks skaičiuojant.

Pasiaiškinsime kokia tvarka Cooley–Tukey algoritme apdorojamos signalo reikšmės  $f_j$  bei kurie koeficientai  $W_m^k$  reikalingi.

Algoritmo veikimo principą galima perprasti nagrinėjant nedaug reikšmių turinčių duomenų atvejį. Tarkime,  $N = 8 = 2^3$ . Tuomet duomenys (skaitmeninio signalo reikšmės) yra skaičiai

$$f = (f_0, f_1, f_2, f_3, f_4, f_5, f_6, f_7).$$

Mūsų tikslas – apskaičiuoti dydžius

$$C_k = f_0 + f_1 W_8^k + f_2 W_8^{2k} + f_3 W_8^{3k} + f_4 W_8^{4k} + f_5 W_8^{5k} + f_6 W_8^{6k} + f_7 W_8^{7k},$$

$$k = 0, 1, 2, 3, 4, 5, 6, 7.$$

Pirmajame Cooley–Tukey algoritmo rekursijos lygyje, kiekvienam fiksuotam  $k = 0, 1, 2, 3$  gausime dvi sumas  $A_k$  ir  $B_k$  iš keturių dėmenų:

$$C_k = A_k + W_8^k B_k, \quad C_{k+4} = A_k - W_8^k B_k, \quad k = 0, 1, 2, 3,$$

$$A_k = \left[ f_0 + f_2 W_4^k + f_4 W_4^{2k} + f_6 W_4^{3k} \right], \quad B_k = \left[ f_1 + f_3 W_4^k + f_5 W_4^{2k} + f_7 W_4^{3k} \right],$$

arba

$$C_k = \left[ f_0 + f_2 W_4^k + f_4 W_4^{2k} + f_6 W_4^{3k} \right] + W_8^k \left[ f_1 + f_3 W_4^k + f_5 W_4^{2k} + f_7 W_4^{3k} \right].$$

Antrajame rekursijos lygyje kiekvienam fiksuotam  $k = 0, 1$  turėsime jau keturias sumas  $AA_k$ ,  $AB_k$ ,  $BA_k$ ,  $BB_k$ , kiekvieną iš dviejų dėmenų:

$$A_k = AA_k + W_4^k AB_k, \quad A_{k+2} = AA_k - W_4^k AB_k,$$

$$B_k = BA_k + W_4^k BB_k, \quad B_{k+2} = BA_k - W_4^k BB_k, \quad k = 0, 1,$$

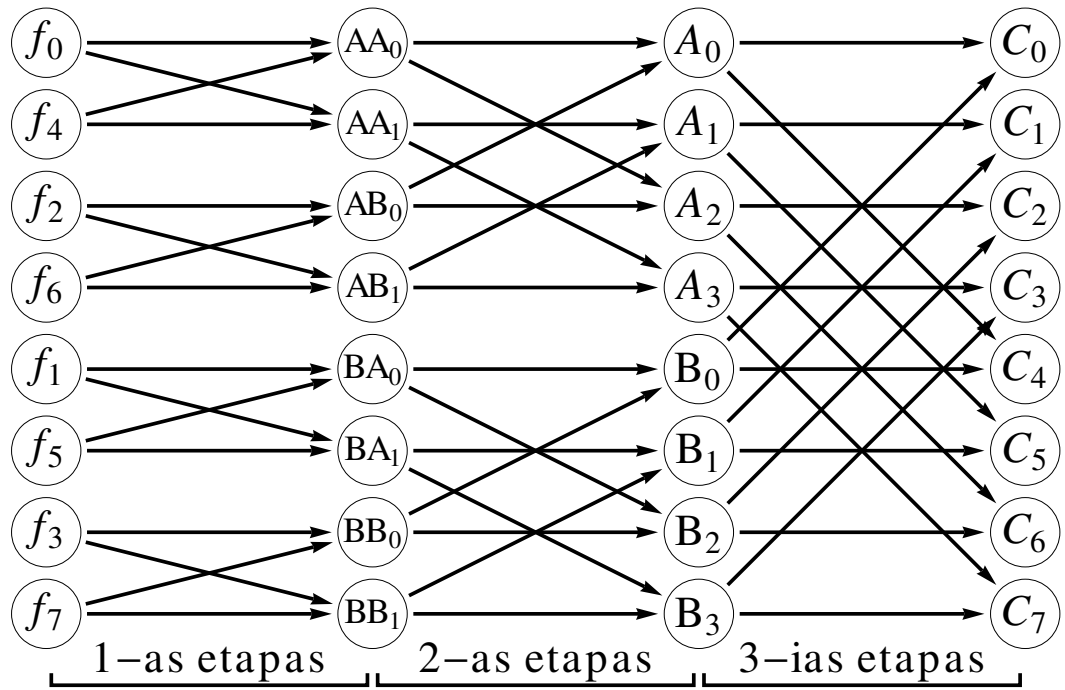
$$AA_k = \left( f_0 + f_4 W_2^k \right), \quad AB_k = \left( f_2 + f_6 W_2^k \right),$$

$$BA_k = \left( f_1 + f_5 W_2^k \right), \quad BB_k = \left( f_3 + f_7 W_2^k \right),$$

arba

$$C_k = \left[ \left( f_0 + f_4 W_2^k \right) + W_4^k \left( f_2 + f_6 W_2^k \right) \right] + W_8^k \left[ \left( f_1 + f_5 W_2^k \right) + W_4^k \left( f_3 + f_7 W_2^k \right) \right].$$

Trečiajame rekursijos lygyje formaliai būtų apibrėžtos aštuonios “sumos”, kiekviena sudaryta iš vienintelio dėmens – signalo reikšmės. Kokia tvarka išsidėsčiusios šios reikšmės ir iš kokių koeficientų juos reikės padauginti sumuojant, matyti jau iš antrojo rekursijos lygio formulių. Todėl iš karto galime pereiti prie 1-ojo algoritmo etapo, kuriame prasideda realūs skaičiavimai.



6 pav. Cooley–Tukey greitosios Furje transformacijos algoritme vykdomų aritmetinių operacijų schema 8 taškų ( $N = 8$ ) atveju. Rodyklės rodo kurie dydžiai iš praeito algoritmo etapo dalyvauja skaičiuojant sekančio etapo dydžius.

1-as etapas. Naudodami signalo reikšmes  $f_j$  ir koeficientus  $W_2^k$ ,  $k = 0, 1$ , apskaičiuojame tarpinius dydžius  $AA_k$ ,  $AB_k$ ,  $BA_k$ ,  $BB_k$ ,  $k = 0, 1$ . Skaičiavimo schema pateikta 6 pav. Šioje schemoje rodyklės rodo kurie dydžiai dalyvauja atliekant aritmetines operacijas. Pavyzdžiui,

$$AA_0 = f_0 + f_4 W_2^0,$$

taigi iš  $f_0$  ir iš  $f_4$  eina po rodyklę į  $AA_0$ . Rodyklė taip pat reiškia vieną aritmetinę operaciją:  $f_4$  dauginamas iš konstantos  $W_2^0$ ; rezultatas sudedamas su  $f_0$ .

Beje, 1-ajame etape prireikia tik vienos konstantos  $W_2^0 = 1$  (kadangi  $W_2^1 = -W_2^0 = -1$ ). Kadangi visi daugikliai lygūs vienetui, šiame etape sandaugų skaičiavimo galima išvengti.

1 lentelė. Cooley–Tukey algoritmo optimalaus duomenų  $f_j$  išdėstymo tvarka pagal indeksus  $j$  su atvirkščiai užrašytais dvejetainiais bitais 8 taškų ( $N = 8$ ) atveju.

Indeksai $j$ tradicine tvarka	Dvejetainiai indekso $j$ bitai	Indekso $j$ bitai atvirkščiai	Indeksai $j$ atvirkščių bitų tvarka
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

Taip pat, matome kokia tvarka reikėtų išdėstyti duomenis, kad jie būtų greičiausiai pasiekiami:

$$f_0, f_4, f_2, f_6, f_1, f_5, f_3, f_7.$$

Žiūrint į šią išdėstymo tvarką, sunkoka suvokti kaip ją būtų galima apibendrinti kai  $N = 16$ ,  $N = 32$  ir taip toliau. Viskas paaiškėja duomenų indeksus išreiškus ne dešimtainėje, bet dvejetainėje skaičiavimo sistemoje, žr. 1 lentelę. Tereikia reikšmių  $f_j$  indeksų dvejetainius bitus užrašyti atvirkštine (veidrodine) tvarka.

*2-as etapas.* Dabar jau naudojame praeitame – 1-ajame etape apskaičiuotus dydžius  $AA_k$ ,  $AB_k$ ,  $BA_k$ ,  $BB_k$ ,  $k = 0, 1$  ir suskaičiuojame šio etapo tarpinius dydžius  $A_k$ ,  $B_k$ ,  $k = 0, 1, 2, 3$ . Skaičiavimo schema matoma 6 pav.

2-ajame etape reikalingos dvi konstantos  $W_4^k$ ,  $k = 0, 1$ .

*3-ias etapas.* Turėdami praeitame (2-ajame) etape apskaičiuotus dydžius  $A_k$ ,  $B_k$ ,  $k = 0, 1, 2, 3$ , randame koeficientus  $C_k$ ,  $k = 0, 1, 2, 3, 4, 5, 6, 7$ . Skaičiavimo schema pateikta 6 pav.

3-iajame etape naudojame konstantas  $W_8^k$ ,  $k = 0, 1, 2, 3$ .

3-iasis etapas yra paskutinis, kadangi turėjome apdoroti  $N = 2^3$  ilgio duomenų seką.

Visose algoritmo sandaugose vienas iš dauginamųjų yra kompleksinė konstanta  $W_m^k$ . Naudojamos tik konstantos (šiems indeksams  $m$  ir  $k$ ):

$$W_m^k = e^{-i \frac{2\pi}{m} k} = \cos \frac{2\pi k}{m} - i \sin \frac{2\pi k}{m}, \quad m = 2, 4, 8, 16, \dots, N, \quad k = 0, 1, \dots, \frac{m}{2} - 1.$$

Beje, kai  $k = 0$ ,  $W_m^0 = 1$ .

#### *Cooley–Tukey algoritmo sudėtingumas*

Įvertinkime kiek aritmetinių operacijų iš viso daroma Cooley–Tukey algoritme. Atimtį ir sudėtį laikysime ekvivalenčiomis kompiuterio procesoriaus laiko požiūriu operacijomis, o sandaugas suskaičiuosime atskirai.

Kaip matome iš nagrinėto pavyzdžio, reikalingi  $n = \log_2 N$  etapai. Kiekviename etape skaičiuojama  $N$  sudėčių ir  $N/2$  sandaugų. Sandaugų yra dvigubai mažiau nei sudėčių, nes kiekviename etape antroji pusė dydžių skaičiuojama analogiškai kaip pirmoji pusė, tik pakeitus sudėtį į atimtį. Pavyzdžiui, mūsų nagrinėtame pavyzdyje su  $N = 8$ , 2-ajame etape skaičiavome dydžius  $A_k$  pagal formules

$$A_k = AA_k + W_4^k AB_k, \quad A_{k+2} = AA_k - W_4^k AB_k, \quad k = 0, 1,$$

taigi skaičiuojant  $A_{k+2}$  nebereikia skaičiuoti sandaugos  $W_4^k AB_k$  (ji jau suskaičiuota, randant  $A_k$ ).

Vadinasi, iš viso reikia atlikti  $(N/2) \log_2 N$  kompleksinių sandaugų ir  $N \log_2 N$  kompleksinių sudėčių. Sandaugų kiekį galima dar šiek tiek sumažinti, pasinaudojus tuo, kad  $W_m^0 = 1$  bei kitais specialiais konstantų  $W_m^k$  atvejais. Pavyzdžiui, kaip matėme 1-ajame algoritmo etape galima apylamai išvengti dauginimo veiksmo.

Taigi, Cooley–Tukey algoritmo sudėtingumas yra  $O(N \log_2 N)$ .

## **1.4. Lentelės pavyzdys**

2 lentelėje ištirta koks yra defektų tankio  $N_{def}$  įverčio santykinis tikslumas.

2 lentelė. Defektų tankio  $N_{def}$  įverčių santykinės paklaidos.

Defekto spindulys $r_0$ , nm	Dažnis $f_{min}$ , Hz	Tikslus $N_{def}$ , $\mu\text{m}^{-2}$	Įvertintas $N_{def}$ , $\mu\text{m}^{-2}$	Santykinė paklaida
1	516	$2.89 \cdot 10^1$	$3.02 \cdot 10^1$	4.5%
10	1090	$2.89 \cdot 10^1$	$3.81 \cdot 10^1$	32.1%
1	3.3	$2.89 \cdot 10^{-1}$	$2.72 \cdot 10^{-1}$	5.7%
10	5.02	$2.89 \cdot 10^{-1}$	$2.57 \cdot 10^{-1}$	11.0%
100	9.78	$2.89 \cdot 10^{-1}$	$3.01 \cdot 10^{-1}$	4.3%
1	0.0242	$2.89 \cdot 10^{-3}$	$2.83 \cdot 10^{-3}$	2.0%
10	0.0328	$2.89 \cdot 10^{-3}$	$2.39 \cdot 10^{-3}$	17.2%
100	0.0502	$2.89 \cdot 10^{-3}$	$2.26 \cdot 10^{-3}$	21.7%
1000	0.0976	$2.89 \cdot 10^{-3}$	$2.62 \cdot 10^{-3}$	9.3%

## 1.5. Pseudokodo pavyzdys

Kartais kyla poreikis sugeneruoti tam tikromis statistinėmis savybėmis pasižymintį signalą kompiuteriu. Tokie signalai vadinami *sintetinės kilmės* signalais arba signalų *simuliacijomis*, kadangi jie nėra jokio realiame gyvenime vykstančio proceso eksperimentiniai duomenys, o tik jų matematinė imitacija.

Sintetinės kilmės signalus galima lengvai ir greitai generuoti, tuo tarpų realių eksperimentinių duomenų surinkimas reikalauja nepalyginamai daugiau laiko ir kitų išteklių. Tokie sintetiniai signalai gali būti ir yra naudojami akademiniams tikslams, matematinių modelių palyginimui su eksperimentiniais rezultatais, kompiuterinių algoritmų realizacijų testavimui. Pavyzdžiui, labai patogu kompiuterinės simuliacijos būdu imituoti įvairius triukšmus, pasižyminčius reikiamomis savybėmis.

Sintetiniai signalai gaunami iš tam tikros matematinės formulės arba išsprendus duotą matematinį modelį (pavyzdžiui, rekurentinius sąryšius, diferencialinę lygtį ir pan.). Matematinė formulė arba modelis gali būti determinuoti (be atsitiktinai kintančių dydžių). Tokiu atveju tiesiog užprogramuojame šią formulę arba algoritmą duotam modeliui spręsti.

Tačiau dažnai reikia programuoti signalų simuliacijas, kuriose yra vienas ar daugiau atsitiktinai kintantys dydžiai. Pavyzdžiui, simuliuojant triukšmą, atsitiktinai klaidžiojančios dalelės judėjimą, miesto transporto atvykimo laikus ir t. t.

Laikysime, kad žinome kaip kompiuteriu gauti pseudoatsitiktinių skaičių su vienoda tikimybe įgyjantį bet kurį intervalo  $[0, 1)$  tašką (dar vadinamą atsitiktiniu dydžiu, tolygiai pasiskirsčiusiu intervale  $[0, 1)$ ). Tokius skaičius žymėsime  $u_1$ ,  $u_2$ , ir taip toliau. Visos programavimo kalbos ir aplinkos turi priemones tokiems skaičiams gauti.

Reikėtų pasakyti, kad dauguma programavimo kalbų ir aplinkų instrumentų šiuos pseudoatsitiktinius skaičius generuoja gana primityviai, todėl jų atsitiktinumas yra paviršutiniškas. Todėl rimtuose taikymuose reikėtų naudoti specialias programų bibliotekas, taikančias specializuotus algoritmus (pavyzdžiui, “Mersenne twister” metodą) kokybiškų pseudoatsitiktinių skaičių  $u_1$ ,  $u_2$ , ir t. t. gavimui.

Puasono atsitiktinis dydis  $X \sim P(\lambda)$  arba atsitiktinis dydis iš Puasono pasiskirstymo (angliškai

“Poisson distribution”) su parametru  $\lambda$  naudojamas aptarnavimo teorijos modeliuose, pavyzdžiui atsitiktinai generuojant laiką, kurį klientui reikės laukti kirpykloje, poliklinikoje, autobuso stotelėje ir pan., kai žinoma statistiškai vidutinė laukimo trukmė  $\lambda > 0$ . Taigi, parametras  $\lambda > 0$  nusako Puasono atsitiktinio dydžio vidurkį, taip pat ir dispersiją (šiam pasiskirstyme vidurkis ir dispersija sutampa). Reikia pažymėti, kad Puasono atsitiktinis dydis įgyja tik natūrines reikšmes.

*Metodas Puasono atsitiktinio dydžio generavimui.* Tarkime, kad  $u_1, u_2, \dots \in [0, 1)$  yra tarpusavyje nepriklausomi, pseudoatsitiktiniai skaičiai, tolygiai pasiskirstę intervale  $[0, 1)$ . Skaičiuojame sandaugas:  $u_1, u_1u_2, u_1u_2u_3$  ir t. t., kol sandauga tampa mažesne už  $e^{-\lambda}$ . Tuomet dauginamųjų skaičius  $n$  priešpaskutinėje sandaugoje kurioje  $e^{-\lambda}$  vis dar buvo viršijamas:  $u_1u_2 \cdots u_n \geq e^{-\lambda}$  ir yra pseudoatsitiktinis skaičius iš Puasono pasiskirstymo su parametru  $\lambda > 0$ . Šio algoritmo pseudokodas:

```

 $a = e^{-\lambda};$ 
 $r = 1;$ 
 $n = -1;$ 
while  $r > a$  do
    sugeneruojame pseudoatsitiktinį  $u \in [0, 1);$ 
     $r = r * u;$ 
     $n = n + 1;$ 
end while
return  $n;$ 

```

Atsitiktinių dydžių iš įvairių pasiskirstymų kompiuterinis generavimas taip pat įmanomas naudojantis tam tikslui skirtais kompiuteriniais įrankiais.

## 2. Šablonas

Darbų rašymams rekomenduojama naudoti katedros L<sup>A</sup>T<sub>E</sub>X šabloną.

### 2.1. Darbų tipai

Kompiuterijos katedros kuriojamų studijų programų studentai studijų metu turi rašyti skirtingų tipų rašto darbus. Rašto darbo tipas nurodomas rašto darbo ataskaitos viršelyje. Darbų tipai su vertimais į anglų kalbą pateikiami 3 lentelėje.

3 lentelė. Rašto darbų tipai

Tipas	Vertimas	Pakopa
Kursinis darbas	Semester Project	BSc
Problemų sprendimu grįstas projektas	Problem-Based Project	BSc
Problemų sprendimu grįstas projektas II d.	Problem-Based Project. Part II	BSc
Bakalauro baigiamasis darbas	Final Bachelor Thesis	BSc
Mokslo tiriamasis darbas I/II	Scientific Research I/II	MSc
Mokslo tiriamojo darbo projektas	Scientific Research Project	MSc
Magistro baigiamasis darbas	Final Master Thesis	MSc
<i>Dalyko ataskaita</i>	<i>Subject Related Report</i>	BSc
Objektinis programavimas. Mini-projektas	Object-Oriented Programming. Mini-Project	
Objektinis-programavimas. Recenzija	Object-Oriented Programming. Peer-Review	

Skirtingi dalykai turi skirtingus rašto darbus, pavyzdžiui, mini-projektą, recenziją, projekto dokumentaciją, reikalavimų aprašą ar pan. Taigi dalykų rašto darbų tipai gali būti skirtingi.

Darbai gali būti rašomi lietuvių arba anglų kalba. Šablone įjungiamoji speciali žymė nurodo kalbą. Šablone daroma prielaida, kad vieną darbą gali atlikti iki penkių studentų.

### 2.2. Dokumento struktūra

Rašto darbuose Santrauka (ir Summary), Įvadas, Išvados, Literatūros šaltiniai, Sutartinių terminų sąrašas yra nenumenuojami, bet vis tiek turi būti įtraukti į darbo turinį. Šablone yra sukurta komanda *sectionWithoutNumber*, skirta nenumenuojamiems skyriams sukurti kartu su standartine žyme, pavyzdžiui, *sec:intro*.

Kai kuriuose darbuose, pavyzdžiui, projektų ataskaitose, reikalinga pratarmė, kurioje paaiškinama, kurį semestrą atliktas darbas, padėkojama už suteiktus duomenis ar pagalbą (skirtą laiką). Padėka išreiškiama išoriniams žmonėms iš kitų Vilniaus universiteto padalinių ar įmonių. Patarmėje taip pat galima paaiškinti, kurios darbo dalys yra iš ankstesnio darbo etapo, jei darbas yra tęstinis. Kai darbą atlieka studentų grupė, studentų parašai gali būti pratarmės puslapyje. Šablone yra nustatoma speciali žymė *signaturesOnTitlePage*, kuri parašo vietas sugeneruoja arba ant viršelio, arba pratarmės puslapyje.

Dažniausiai santraukoje būna 100–200 žodžių. Žodžiams skaičiuoti galima panaudoti Unix komandą *wc -w failoVardas* [6]. Šablone santrauka, santrauka anglų kalba, pratarmė, įvadas, išvados yra atskiruose L<sup>A</sup>T<sub>E</sub>X failuose.

## 2.3. Literatūros šaltiniai

Visi literatūros šaltinių sąraše esantys elementai turi būti referuojami (cituojami) tekste. Naudojant  $\LaTeX$ , bibliografinius šaltinius valdo įrankis  $\BibTeX$ . Šaltiniai būna skirtingų tipų. Pavyzdžiui, [1] ir [7] šaltiniai yra knygos, [8] šaltinis yra straipsnis žurnale, o [9] yra straipsnis konferencijoje. Beje, mokslinių straipsnių, knygų ir pan.  $\BibTeX$  šaltinių paruoštukus *bib* formatu galima rasti kompiuterių mokslo bibliografijoje *dblp* [3].

Rašto darbuose reikia naudoti ir referuoti tik patikimus šaltinius. Taip pat šaltinių sąraše gali būti duomenų šaltinis, pavyzdžiui, statistiniai duomenys gali būti imami iš Europos komisijos duomenų bazės [4]. Informatikos terminų vertimus į lietuvių kalbą galima pasitikslinti Kompiuterinių terminų žodyne [2].

## 2.4. Pseudo-kodas ir programinio kodo dalys

Algoritmus rekomenduojama pateikti pseudo-kodu. Šablone pagal nutylėjimą yra įtraukti paketai *algorithm* ir *algorithmic*. Naudojant šiuos paketus pseudo-kode galima turėti žymes kodo eilutėms. Pavyzdžiui, 1 algoritme 1 eilutė pažymi pseudo-kodo sąlygą. Pagal nutylėjimą įvesties ir išvesties parametrai žymimi lietuviškais terminais Įvestis ir Išvestis, bet juos galima pakeisti. Algoritmus galima aprašyti ir naudojant kitą paketą *algorithm2e*.

---

### 1 algoritmas. Algoritmo pavyzdys

---

**Įvestis:**  $a, b, c \in \mathbb{N}$

**Išvestis:**  $x : \{\text{true}, \text{false}\} \times \mathbb{N}$

```
1: if  $a < b$  and  $b < c$  then  
2:    $a \leftarrow c$   
3:   return ( $\text{false}, a$ )  
4: end if  
5: return ( $\text{true}, a$ )
```

---

Jeigu prireikia nurodyti programų išeities kodo dalis, galima naudotis *listings* paketu, pavyzdys yra pateiktas 1 išeities kode. Šiame dokumente įjungta Java programavimo kalba kaip parametras, nors paketas palaiko nemažą aibę skirtingų paradigmu kalbų.

### 1 išeities kodas. Java metodo pateikimo pavyzdys

```
1 boolean method(String a, String b){  
2   if (a.contains(b)) return true;  
3   return false;  
4 }
```

## 2.5. Paveikslėlių paruošimas

Turint duomenų, tinkamai juos paruošti kaip paveikslėlius (grafikus) galima tarpplatforminiu įrankiu Gnuplot [5]. Pavyzdžiui, statistinių duomenų blokas, pateiktas 4 lentelėje, yra atvaizduotas 7 pav, o paveikslėlis buvo sugeneruotas Gnuplot įrankiu ir panaudojant kodą iš 2 išeities kodo.

1 pavyzdyje yra pateikiamas duomenų blokas ir jo pateikimas grafiku, kuris generuojamas Gnuplot įrankiu.

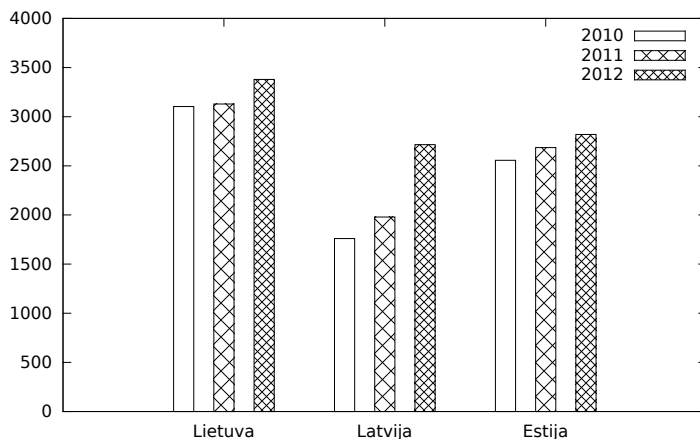
## 2 išeities kodas. Gnuplot kodo pavyzdys

```
1 # saugoti pvz. pavadinimu skriptas.gpi
2 # vykdyti gnuplot skriptas.gpi
3
4 set terminal pdf monochrome
5 set output 'EUdataLT.pdf'
6 set style data histogram
7 set samples 11
8 set boxwidth 0.5
9 set style fill pattern border
10 set style histogram cluster gap 1
11 set yrange [0:4000]
12
13 plot 'dataLT.dat' using 2:xtic(1) title columnheader(2),
14      for [i=3:4] '' using i title columnheader(i)
```

**1 pavyzdys.** Čia pradedamas pavyzdys. Pavyzdžio pabaigą rodo kvadratėlis. Nors paveikslėliai ir lentelė priklauso pavyzdžiui, jie tekste gali „plaukioti.“

4 lentelė. Judrūs studentai pagal kilmę [4]

	2010	2011	2012
Lietuva	3103	3129	3379
Latvija	1760	1979	2716
Estija	2556	2686	2819



7 pav. Judrių studentų skaičiai. Grafiko y ašyje yra nurodyti studentų skaičiai.

□

Šablone galima paruošti ir apibrėžimą. Pavyzdžiui, 1 apibrėžime apibrėžiamas kelias.

**1 apibrėžimas.** Keliu  $k$  laikysime taškų seką  $\langle t_1, \dots, t_n \rangle$ , kai  $t_i \in \mathbb{R} \times \mathbb{R}$ .



## 2.6. Sudėtingesnės lentelės

Sudėtingesnėms lentelėms aprašyti gali reikėti papildomų paketų: ilgoms lentelėms *longtable*, stulpelių/eilučių jungimui *multirow*, teksto pasukimui *rotating*. Sudėtingesnės lentelės pavyzdys pateiktas 5 lentelėje.

5 lentelė. Sudėtingesnės lentelės pavyzdys

Module	ECTS	Student workload	Contact Hours	Private Work Hours	Key Programme Competences										
					Generic Competences					Specific Competences					
					GC1	GC2	GC3		SC1	SC2	SC3				
					Learning Outcomes										
					GC1-1	GC2-1	GC2-2	GC3-1	GC3-2	SC1-1	SC1-2	SC2-1	SC2-2	SC3-1	
I YEAR	60	1600	358	1242											
1 SEMESTER	30	800	221	579											
Management	5	125	50	75		X			X	X				X	

## Literatūros šaltiniai

- [1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 3rd edition, 2009.
- [2] Valentina Dagienė, Gintautas Grigas, and Tatjana Jevsikova. Anglų–lietuvių kalbų kompiuterijos žodynas. Vilniaus universitetas. Matematikos ir informatikos institutas, 2012. <http://ims.mii.lt/ALK%C5%BD/>.
- [3] dblp. Computer Science Bibliography. <http://dblp.uni-trier.de/db/>.
- [4] European Commission. Eurostat. Browse / search database, 2014. [http://epp.eurostat.ec.europa.eu/portal/page/portal/statistics/search\\_database](http://epp.eurostat.ec.europa.eu/portal/page/portal/statistics/search_database).
- [5] Gnuplot homepage, November 2014. <http://www.gnuplot.info/>.
- [6] O'Reilly Media, Inc. *Linux in a Nutshell. Linux Command Directory*, 5th edition edition, 2014. <http://www.linuxdevcenter.com/cmd/>.
- [7] Philippe Rigaux, Michel Scholl, and Agnes Voisard. *Spatial Databases with Application to GIS*. Morgan Kaufmann Publishers, 2002.
- [8] Darius Sidlauskas, Simonas Saltenis, and Christian S. Jensen. Processing of extreme moving-object update and query workloads in main memory. *VLDB J.*, 23(5):817–841, 2014.
- [9] Laurynas Speicys, Christian S. Jensen, and Augustas Kligys. Computational data modeling for network-constrained moving objects. In *ACM-GIS 2003, Proceedings of the Eleventh ACM International Symposium on Advances in Geographic Information Systems, New Orleans, Louisiana, USA, November 7-8, 2003*, pages 118–125, 2003.