

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
SISTEMINĖS ANALIZĖS KATEDRA

Vaiva Leščiauskiene

**Kvadratūrinių formulių liekamųjų narių įverčiai ir
jų analizė**

Magistro darbas

Darbo vadovas
doc. dr. Kostas Plukas

Kaunas, 2006

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
SISTEMINĖS ANALIZĖS KATEDRA

**Kvadratūrinių formulių liekamųjų narių įverčiai ir
jų analizė**

Magistro darbas

Kalbos konsultantė	Vadovas
Lietuvių k. katedros lekt. dr. I.Mickienė	doc. dr. K.Plukas
2006-05-	2006-05-
Recenzentas	Atliko
doc. dr. E.Mačikėnas	IFM-0/1 gr. stud. V.Leščiauskiene
2006-05-	2006-05-

Kaunas, 2006

Turinys

1. ĮVADAS.....	4
2. SKAITINIO INTEGRVIMO UŽDAVINYS IR JO SPRENDIMO ALGORITMAI...6	6
2.1 SKAITINIO INTEGRVIMO UŽDAVINIO APRAŠYMAS.....	6
2.2. KVADRATŪRINIŲ FORMULIŲ TIPAI.....	7
2.2.1. <i>Niutono ir Koteso kvadratūrinės formulės</i>	7
2.2.2. <i>Gauso kvadratūrinės formulės</i>	10
2.3. INTEGRVIMO STRATEGIJOS	12
2.3.1. <i>Vienodo žingsnio visame integravimo intervale strategija</i>	13
2.3.2. <i>Adaptyviojo integravimo strategija</i>	14
3. KVADRATŪRINIŲ FORMULIŲ PAKLAIDOS ĮVERTIS.....	17
3.1. KVADRATŪRINĖS FORMULĖS IR NULINĖS TAISYKLĖS.....	18
3.2. LOKALIOS PAKLAIDOS ĮVERTINIMO ALGORITMAS	20
4. EKSPERIMENTINIS TYRIMAS	25
5. IŠVADOS	41
6. LITERATŪROS SĄRAŠAS.....	42
7. ERROR ESTIMATES OF QUADRATURE FORMULAS AND THEIR ANALYSIS	43

1. ĮVADAS

Dažniausiai naudojamos skaitinės procedūros apima baigtinių integralų aproksimavimą. Integralų aproksimavimo uždavinys formuluojamas taip: duotas integralas $I[f] = \int_a^b f(x)dx$, čia a ir b – integravimo rėžiai, o $f(x)$ – pointegralinė funkcija, pateikta arba analiziškai, arba reikšmių lentele, reikia apskaičiuoti $I[f]$ reikšmę ε tikslumu.

Visų skaitinio integravimo metodų esmė yra ta, kad pointegralinė funkcija $f(x)$ keičiama aproksimuojančiąja funkcija $F(x)$ ir laikoma, jog $I[f] = \int_a^b f(x)dx \approx \int_a^b F(x)dx$. Paprastai parenkama tokia aproksimuojančioji funkcija $F(x)$, kad integralas $\int_a^b F(x)dx$ būtų lengvai integruojamas analiziškai.

Istoriškai plačiausiai naudojama aproksimuojančioji funkcija $F(x)$ yra n -osios eilės interpoliacinis polinomas, einantis per taškus $(x_i; f(x_i))$, $i = \overline{0, n}$. Keičiant n reikšmę bei įvairiai parenkant taškus x_i , gaunamos skirtingos skaitinio integravimo formulės, kurios vadinamos kvadratūrinėmis formulėmis.

Kai pointegralinė funkcija keičiama interpoliaciniu polinomu, kvadratūrinės formulės skirstomos į dvi stambias klases:

- Niutono ir Koteso kvadratūrines formules.
- Gauso kvadratūrines formules.

Integruojant skaitiniu būdu, susiduriama su dviejų rūšių paklaidomis:

- Integravimo metodo paklaidomis.
- Apvalinimo paklaidomis.

Apvalinimo paklaidos susijusios su pointegralinės funkcijos reikšmių apskaičiavimu. Kuo daugiau šios funkcijos reikšmių naudojama integruojant, tuo didesnė apvalinimo paklaida.

Metodo, kuriuo būtų galima iš anksto apskaičiuoti konkrečios kvadratūrinės formulės ir pointegralinės funkcijos optimalų integravimo žingsnį, nėra. Todėl reikia rinktis tokią integravimo strategiją, kuri leistų norimą integravimo tikslumą pasiekti esant mažiausiam pointegralinės funkcijos reikšmių skaičiui.

Šiame darbe aptariamos dvi integravimo strategijos:

- Vienodo žingsnio visame integravimo intervale strategija.
- Adaptyviojo integravimo strategija.

Darbo tikslai:

1. Išnagrinėti žinomų kvadratūrinių formulių liekamųjų narių įverčio metodus.
2. Iširti K.Pluko pasiūlytą metodą [6].
3. Patyrinėti žinomus metodus bei palyginti su K.Pluko pasiūlytu metodu [6].
4. Palyginti K.Pluko pasiūlyto metodo rezultatus nenormuojant ir normuojant paklaidos įvertį.

2. SKAITINIO INTEGRAVIMO UŽDAVINYS IR JO SPRENDIMO ALGORITMAI

2.1 Skaitinio integravimo uždavinio aprašymas

Viena iš dažniausiai naudojamų skaitinių procedūrų yra baigtinių integralų aproksimavimas. Turime integralą $I[f] = \int_a^b f(x)dx$, kur a ir b – integravimo rėžiai, o $f(x)$ – pointegralinė funkcija, pateikta arba analiziškai, arba reikšmių lentele. Reikia apskaičiuoti integralo įverčio \hat{Q} reikšmę ε tikslumu.

Visų skaitinio integravimo metodų esmė yra ta, kad pointegralinė funkcija $f(x)$ keičiama aproksimuojančiąja funkcija $F(x)$ ir laikoma, jog $I[f] = \int_a^b f(x)dx \approx \int_a^b F(x)dx$. Paprastai parenkama tokia aproksimuojančioji funkcija $F(x)$, kad integralas $\int_a^b F(x)dx$ būtų lengvai integruojamas analiziškai.

Dažniausiai naudojama aproksimuojančioji funkcija $F(x)$ yra n -osios eilės interpoliacinis polinomas, einantis per taškus $(x_i; f(x_i))$, $i = \overline{0, n}$. Keičiant n reikšmę bei įvairiai parenkant taškus x_i , gaunamos skirtingos skaitinio integravimo formulės, kurios vadinamos kvadratūrinėmis formulėmis.

Kai pointegralinė funkcija keičiama interpoliaciniu polinomu, kvadratūrinės formulės skirstomos į dvi stambias klases:

- 1) Niutono ir Koteso kvadratūrinės formulės.
- 2) Gauso kvadratūrinės formulės.

Jei $F(x)$ yra n -osios eilės interpoliacinis polinomas, einantis per taškus $(x_i; f(x_i))$, kur $i = \overline{0, n}$, tai gauta kvadratūrinė formulė tiksliai integruoja visus polinomas iki n -tosios eilės imtinai. Todėl labai patogu kvadratūrinės formulės išvesti R.V.Hemingo siūlomą metodu:

$$\int_a^b f(x)dx \cong \sum_{i=0}^n w_i f(x_i),$$

kur w_i - koeficientai, kurie apskaičiuojami taip, kad formulė tiksliai integruotų bazinių funkcijų $1, x, x^2, \dots, x^n$ sistemą.

2.2. Kvadratūrinių formulių tipai

2.2.1. Niutono ir Koteso kvadratūrinės formulės

Niutono ir Koteso kvadratūrinės formulės yra interpoliacinio tipo, kai pointegralinė funkcija $f(x)$ keičiama interpoliaciniu polinomu, kurį perteikia iš anksto fiksuoti vienodai vienas nuo kito nutolę taškai x_i .

Jeigu reikia apskaičiuoti integralą $I[f] = \int_a^b f(x)dx$, tada n -tosios eilės Niutono ir Koteso kvadratūrinė formulė išvedama taip:

1) integravimo intervalas $[a; b]$ taškais $a = x_0, x_1, \dots, x_{n-1}, x_n = b$ dalijamas į n lygių dalių, kur $x_i = x_0 + ih$, $i = \overline{0, n}$, o $h = (b - a)/n$;

2) pointegralinė funkcija $f(x)$ keičiama n -tosios eilės interpoliaciniu polinomu $F(x)$, einančiu per taškus $(x_i; y_i)$, $i = \overline{0, n}$; čia $y_i = f(x_i)$;

$$3) I[f] = \int_a^b f(x)dx \approx \int_a^b F(x)dx = \sum_{i=0}^n w_i y_i. \quad (1)$$

(1) formulė turi tiksliai integruoti funkcijas $1, x, x^2, \dots, x^k, \dots, x^n$. Tada simboliu m_k pažymėjus $\int_a^b x^k dx$, iš (1) formulės gaunama lygčių sistema

$$\begin{cases} w_0 + w_1 + \dots + w_n = m_0, \\ w_0 x_0 + w_1 x_1 + \dots + w_n x_n = m_1, \\ \dots \\ w_0 x_0^k + w_1 x_1^k + \dots + w_n x_n^k = m_k, \\ \dots \\ w_0 x_0^n + w_1 x_1^n + \dots + w_n x_n^n = m_n. \end{cases} \quad (2)$$

(2) sistemos matricos A determinantas yra Vandermondo determinantas, t.y.

$$\det(A) = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_0 & x_1 & \dots & x_n \\ \dots & \dots & \dots & \dots \\ x_0^n & x_1^n & \dots & x_n^n \end{bmatrix}.$$

Jis nelygus nuliui, jei $x_i \neq x_j$. Vadinasi, (2) lygčių sistema yra suderinta ir turi vienintelį sprendinį.

Tarkim turime polinomą

$$\omega_i(x) = (x - x_0) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n). \quad (3)$$

Padauginus ir sutraukus panašiuosius narius, gauname:

$$\omega_i(x) = \sum_{k=0}^n c_k x^k, \quad c_n = 1.$$

Kadangi $\omega_i(x_j) = 0$, kai $j \neq i$, ir $\omega_i(x_i) \neq 0$, tai matricos A^{-1} i -toji eilutė bus $\frac{1}{\omega_i(x_i)}(c_0, c_1, \dots, c_n)$. Padauginus A^{-1} i -tąją eilutę iš A j -tojo stulpelio, gausime:

$$\frac{\sum_{k=0}^n c_k x_j^k}{\omega_i(x_i)} = \frac{\omega_i(x_j)}{\omega_i(x_i)} = \begin{cases} 1, & \text{kai } j = i, \\ 0, & \text{kai } j \neq i. \end{cases}$$

Vadinasi,

$$w_i = \frac{1}{\omega_i(x_i)} \sum_{k=0}^n c_k m_k. \quad (4)$$

Iš (4) formulės galima gauti koeficiento w_i analizinę išraišką, priklausančią nuo taškų x_i ir momentų m_k . Ši formulė patogi, kai taškai x_i nesudaro aritmetinės progresijos.

Toliau pateikta Niutono ir Koteso kvadratūrinių formulių ir jų liekamųjų narių lentelė (1 lentelė). Joje $x \in [a; b]$, o k_8, k_9 ir k_{10} – koeficientai.

Niutono ir Koteso kvadratūrinių formulių bei jų liekamųjų narių lentelė integralui

$$\int_a^b f(x) dx = Ah(B_0f(a) + B_1f(a+h) + \dots + B_i f(a+ih) + \dots + B_n f(b)) + K_n \text{ apskaičiuoti}$$

n	A	B_0	B_1	B_2	B_3	B_4	B_5	K_n
1	1/2	1	1					$-(1/12)h^3 f^{(2)}(x)$
2	1/3	1	4	1				$-(1/90)h^5 f^{(4)}(x)$
3	3/8	1	3	3	1			$-(3/80)h^5 f^{(4)}(x)$
4	2/45	7	32	12	32	7		$-(8/945)h^7 f^{(6)}(x)$
5	5/288	19	75	50	50	75	19	$-(275/12096)h^7 f^{(6)}(x)$
6	1/140	41	216	27	272	27	216	$-(9/1400)h^9 f^{(8)}(x)$
7	7/17280	751	3577	1323	2989	2989	1323	$-(8183/518400)h^9 f^{(8)}(x)$
8	4/14175	989	5888	-928	10496	-4540	10496	$k_8 h^{11} f^{(10)}(x)$
9	9/89600	2857	15741	1080	19344	5778	5778	$k_9 h^{11} f^{(10)}(x)$
10	5/299376	16067	106300	-48525	272400	-260550	427368	$k_{10} h^{13} f^{(12)}(x)$

Išanalizavus 1 lentelės duomenis galima daryti tokias išvadas:

1. Niutono ir Koteso kvadratūrinių formulių koeficientai w_i yra simetriški centro atžvilgiu, ir jų suma lygi n .

2. Lyginės eilės Niutono ir Koteso kvadratūrinės formulės tiksliai integruoja visus polinomus iki $(n+1)$ -osios eilės imtinai, tuo tarpu nelyginės eilės kvadratūrinės formulės - tik iki n -osios eilės imtinai. Ši lyginės eilės formulių savybė atsiranda dėl to, kad Niutono ir Koteso kvadratūrinėse formulėse naudojami vienodai nutolę vienas nuo kito taškai.

3. Lyginės eilės Niutono ir Koteso kvadratūrinės formulės liekamasis narys yra proporcingas h^{k+3} (čia k - lyginis skaičius) ir į proporcingumo koeficientą kaip daugiklis įeina $(k+2)$ -osios eilės pointegralinės funkcijos išvestinė. Nelyginės eilės kvadratūrinės formulės liekamasis narys proporcingas h^{l+2} (čia l - nelyginis skaičius) ir proporcingumo koeficientas turi daugiklį, lygų pointegralinės funkcijos $(l+2)$ -ajai išvestinei.

2.2.2. Gauso kvadratūrinės formulės

Niutono ir Koteso kvadratūrinėse formulėse taškai x_i , apibrėžiantys pointegralinę funkciją aproksimuojantį polinomą, yra nutolę vienodai vienas nuo kito ir iš anksto fiksuoti, o kvadratūrinė formulė turi išraišką

$$\int_a^b f(x) dx = \sum_{i=1}^n w_i f(x_i). \quad (5)$$

Reikia rasti tokius taškus x_i ir svorinius koeficientus w_i , su kuriais (5) formulė tiksliai integruotų kiek galima aukštesnės eilės polinomus.

Kadangi turime $2n$ laisvai kintamų parametru, tai galime reikalauti, kad (5) formulė tiksliai integruotų visus polinomus iki $(2n-1)$ -osios eilės imtinai. Nagrinėjamas uždavinys nebus susiaurintas, jeigu vietoj integravimo rėžių a ir b paimsime rėžius -1 ir 1 , taigi galima nagrinėti formulę

$$\int_{-1}^1 f(x) dx = \sum_{i=1}^n w_i f(x_i). \quad (6)$$

Integralą $\int_a^b f(x) dx$ nesunku paversti integralu $\int_{-1}^1 \varphi(t) dt$, taikant keitinį

$$x = \frac{1}{2}(b-a)t + \frac{1}{2}(b+a). \quad (7)$$

Tada

$$\int_a^b f(x) dx = \frac{1}{2}(b-a) \int_{-1}^1 f\left(\frac{1}{2}(b-a)t + \frac{1}{2}(b+a)\right) dt. \quad (8)$$

1 teorema. (6) kvadratūrinė formulė tiksliai integruoja visus polinomus iki $(2n-1)$ -osios eilės imtinai tada ir tik tada, kai:

1) daugianaris $\omega(x) = (x-x_1)(x-x_2)\dots(x-x_n)$ yra ortogonalus bet kuriam polinomui $q(x)$, kurio laipsnis $k \leq n-1$, t.y.

$$\int_{-1}^1 \omega(x) q(x) dx = 0;$$

2) ta formulė yra interpoliacinio tipo kvadratūrinė formulė, t.y. kai

$$w_i = \int_{-1}^1 \frac{\omega(x)}{(x-x_i)\omega'(x_i)} dx.$$

2 teorema. Egzistuoja vienintelis polinomas $\omega(x)$, tenkinantis 1 teoremos pirmąją sąlygą.

3 teorema. Jei $\omega(x)$ yra n -tojo laipsnio polinomas, tenkinantis 1 teoremos pirmąją sąlygą, tai intervale $[-1; 1]$ jis turi n tikrųjų šaknų.

Polinomas, tenkinantis 1 teoremos pirmąją sąlygą, yra Ležandro polinomas $P_n(x)$, kurį patogiau reikšti rekurenčiomis formulėmis:

$$\begin{aligned} P_0(x) &= 1, \\ P_1(x) &= x, \\ P_n(x) &= \frac{1}{n}((2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x)), \quad n \geq 2. \end{aligned} \tag{9}$$

Gauso kvadratūrinės formulės taškas x_i ir svorinius koeficientus w_i galima apskaičiuoti dviem metodais.

Pirmasis metodas. Taškai x_i sutampa su Ležandro polinomo šaknimis, o svoriniai koeficientai w_i apskaičiuojami pagal formulę

$$w_i = \frac{2}{(1-x_i^2)(P_n'(x_i))^2}. \tag{10}$$

Antrasis metodas (Hemingo metodas). Šis metodas remiasi tuo, kad (6) formulė turi tiksliai integruoti visus polinomus iki $(2n-1)$ -osios eilės imtinai. Tada iš (6) formulės gaunama lygčių sistema

$$\begin{cases} w_1 + w_2 + \dots + w_n = m_0, \\ w_1x_1 + w_2x_2 + \dots + w_nx_n = m_1, \\ w_1x_1^2 + w_2x_2^2 + \dots + w_nx_n^2 = m_2, \\ \dots \\ w_1x_1^{2n-1} + w_2x_2^{2n-1} + \dots + w_nx_n^{2n-1} = m_{2n-1}; \end{cases}, \tag{11}$$

čia $m_k = \int_{-1}^1 x^k dx$.

R.V.Hemingas pasiūlė originalų šios sistemos sprendimo metodą. Tarkime, kad žinome taškus x_i , tenkinančius (11) sistemą. Apibrėžkime polinomą

$$\pi(x) = \prod_{i=1}^n (x - x_i) = \sum_{k=0}^n c_k x^k, \text{ čia } c_n = 1.$$

Aišku, kad $\pi(x_i) = 0, i = \overline{1, n}$.

Padauginkime (11) sistemos pirmąją lygtį iš c_0 , antrąją – iš c_1 ir t.t., $(n+1)$ -ąją – iš c_n ir šias lygtis sudėkime. Atsižvelgdami į tai, kad $\pi(x_i) = 0$, gauname:

$$\sum_{k=0}^n c_k m_k = 0.$$

Procesą pakartokime pradėdami antrąja (11) sistemos lygtimi. Gausime:

$$\sum_{k=0}^n c_k m_{k+1} = 0.$$

Tai pakartoję n kartų, turėsime tiesinių lygčių sistemą

$$\sum_{k=0}^n c_k m_{k+j} = 0, j = \overline{0, n-1}. \quad (12)$$

Jeigu jos determinantas nelygus nuliui, tai, ją išsprendę, gausime polinomą, kurio šaknys yra Gauso kvadratūrinės formulės taškai. Turėdami juos ir paėmę (11) sistemos n pirmųjų lygčių, iš gautos tiesinių lygčių sistemos apskaičiuosime svorinius koeficientus w_i .

Kai (12) lygčių sistemos determinantas lygus nuliui, galimas vienas iš dviejų atvejų: arba (12) lygčių sistema yra nesuderinta ir kartu nėra tokios kvadratūrinės formulės, arba matricos rangas žemesnis už n . Šiuo atveju prie (12) lygčių sistemos prijungiamos papildomos lygtys, kurios gaunamos iš (11) sistemos, polinomo laipsnį imant aukštesnį nei $2n-1$. Tai reiškia, kad gaunama tikslesnė kvadratūrinė formulė.

2.3. Integravimo strategijos

Integruojant skaitiniu būdu, susiduriama su dviejų rūšių paklaidomis:

- 1) Integravimo metodo paklaidomis.
- 2) Apvalinimo paklaidomis.

Apvalinimo paklaidos susijusios su pointegralinės funkcijos reikšmių apskaičiavimu. Kuo daugiau šios funkcijos reikšmių naudojama integruojant, tuo didesnė apvalinimo paklaida.

Kaip matyti iš 1 lentelės, integravimo metodo paklaida yra proporcinga h^p , čia h – integravimo žingsnio ilgis, o p – teigiamasis sveikasis skaičius. Tuo tarpu apvalinimo paklaida proporcinga $1/h$. Tai reiškia, kad mažinant integravimo žingsnį (didinant pointegralinės funkcijos reikšmių skaičių), metodo paklaida mažėja, o apvalinimo paklaida didėja. Kiekvienai kvadratūrinei formulei ir pointegralinei funkcijai galima parinkti tokią optimalią integravimo žingsnio reikšmę, su kuria metodo ir apvalinimo paklaidos yra tos pačios eilės. Kai integravimo žingsnio ilgis pasidaro mažesnis už šią reikšmę, apvalinimo paklaida viršija metodo paklaidą ir toliau mažinti integravimo žingsnį nėra prasmės.

Metodo, kuriuo būtų galima iš anksto apskaičiuoti konkrečios kvadratūrinės formulės ir pointegralinės funkcijos optimalų integravimo žingsnį, nėra. Todėl reikia rinktis tokią integravimo strategiją, kuri leistų norimą integravimo tikslumą pasiekti esant mažiausiam pointegralinės funkcijos reikšmių skaičiui.

2.3.1. Vienodo žingsnio visame integravimo intervale strategija

Šią strategiją nagrinėsime imdami 4-osios eilės Niutono ir Koteso kvadratūrinę formulę.

Algoritmas $\int_a^b f(x)dx$ apskaičiuoti, kai 4-osios eilės Niutono ir Koteso formulė taikoma k

kartų:

- Inicializuojami $\hat{Q} = 0$, $x = a$, $h = (b - a)/4k$, $y_0 = f(x)$, $sk = 1$.

- Kol $1 \leq i \leq k$:

- Apskaičiuojami $y_4 = f(x + 4h)$,

$$\hat{Q} = \hat{Q} + 7y_0 + 32f(x + h) + 12f(x + 2h) + 32f(x + 3h) + 7y_4.$$

- Atnaujinami $y_0 = y_4$, $x = x + 4h$, $sk = sk + 4$.

- Atnaujinamas $\hat{Q} = \frac{2h\hat{Q}}{45}$.

Algoritmas integralui $\int_a^b f(x)dx$ apskaičiuoti *eps* tikslumu (*max* – didžiausias leistinas pointegralinės funkcijos reikšmių skaičius, *err* – integralo santykinė paklaida):

- Inicializuoti $p = 6$, $k = 1$, $z = 1 + eps$.
- Pagal aukščiau aprašytą algoritmą randami \hat{Q}_h ir sk .
- Kol $|z| > eps$ ir $sk \leq \max$:
 - Atnaujinti $k = 2k$.
 - Pagal aukščiau aprašytą algoritmą randami $\hat{Q}_{h/2}$ ir sk .
 - Apskaičiuojamas liekamasis narys $z = (\hat{Q}_{h/2} - \hat{Q}_h) / (2^p - 1)$.
 - Atnaujinamas $\hat{Q}_h = \hat{Q}_{h/2}$.
- Jeigu $sk \leq \max$:
 - Apskaičiuojama $\hat{Q} = \hat{Q}_{h/2} + z$, $err = |z / \hat{Q}|$.

2.3.2. Adaptyviojo integravimo strategija

Tarkime, kad norime apskaičiuoti $R = \int_a^b f(x)dx$ reikšmę ε tikslumu, integravimui taikydami kvadratūrinę formulę, kurios liekamasis narys proporcingas h^p , čia h – integravimo žingsnis. Adaptyviojo integravimo strategijos idėja labai paprasta: intervalas $[a; b]$ dalijamas į trumpesnius intervalus ten, kur pointegralinė funkcija kinta sparčiau, ir į ilgesnius intervalus ten, kur ji kinta lėčiau. Kiekvienam intervalui taikoma pasirinkta kvadratūrinė formulė, o intervalas imamas tokio ilgio, kad jame integravimo paklaida būtų ne didesnė už $\frac{H_i}{b-a} \varepsilon$, čia H_i – intervalo ilgis. Tada integravimo paklaida visame intervale $[a; b]$ bus ne didesnė kaip

$$\sum_i \frac{H_i}{b-a} \varepsilon = \frac{\varepsilon}{b-a} \sum_i H_i = \varepsilon.$$

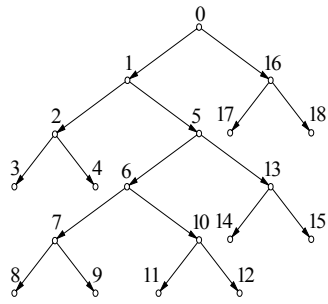
Plačiausiai taikoma adaptyviojo integravimo strategija vykdoma taip:

- Apskaičiuojamas integravimo intervalo ilgis $H = b - a$.

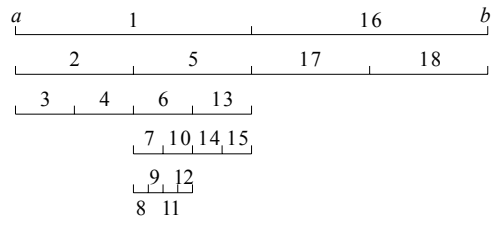
- Kvadratinė formulė taikoma intervale $[a; b]$ ir gaunama integralo reikšmė \hat{Q}_h , čia h – integravimo žingsnis.
- Apskaičiuojama $\hat{Q}_{h/2} = \hat{Q}_{h/2}^k + \hat{Q}_{h/2}^d$, kur $\hat{Q}_{h/2}^k$ ir $\hat{Q}_{h/2}^d$ - integralo \hat{Q} reikšmės atitinkamai integravimo intervalo kairiajame ir dešiniajame pusintervalyje, kai integravimo žingsnis $h/2$.
- Jei $\left| \frac{\hat{Q}_{h/2} - \hat{Q}_h}{2^p - 1} \right| > \frac{H}{b-a} \varepsilon$, tai nagrinėjame intervale integravimo tikslumas yra nepakankamas. Tada dešiniojo pusintervalio integralo reikšmę $\hat{Q}_{h/2}^d$, integravimo abscises bei jas atitinkančias pointegralinės funkcijos reikšmes išimame; integravimo intervalą imame lygų kairiajam pusintervaliui, t.y. $\hat{Q}_h = \hat{Q}_{h/2}^k$, $H = H/2$, ir grįžtame į 3 punktą.
- Jei $\left| \frac{\hat{Q}_{h/2} - \hat{Q}_h}{2^p - 1} \right| \leq \frac{H}{b-a} \varepsilon$, tai nagrinėjame intervale, kurio ilgis H , integravimo tikslumas yra pasiektas ir $\hat{Q}_H = \hat{Q}_{h/2} + \frac{\hat{Q}_{h/2} - \hat{Q}_h}{2^p - 1}$.
- Jei yra, nagrinėjame dešinįjį pusintervalį, t.y. $\hat{Q}_h = \hat{Q}_{h/2}^d$, surenkame išimintas jo integravimo abscises bei tas abscises atitinkančias pointegralinės funkcijos reikšmes, integravimo intervalą imame lygų dešiniajam pusintervaliui, apskaičiuojame intervalo integravimo ilgį H ir grįžtame į 3 punktą.
- Jei nagrinėtų dešiniųjų pusintervalių nėra, tai integralo \hat{Q} reikšmė *eps* tikslumu lygi visų \hat{Q}_H sumai.

Skaičiavimas laikantis adaptinio integravimo strategijos vykdomas pagal binarinę medį. Vaizdžiai ši strategija parodyta [1](#) paveiksle.

Nesunku pastebėti, kad adaptinio integravimo strategijos organizavimas yra sudėtingesnis nei vienodo žingsnio visame integravimo intervale strategijos, tačiau, taikant adaptinio integravimo strategiją, norimas integravimo tikslumas pasiekiamas esant gerokai mažesniai pointegralinės funkcijos reikšmių skaičiui.



a)



b)

1 pav. Binarinis medis (a) ir jį atitinkantis integravimo intervalo skaidymas į subintervalus (b).

3. KVADRATŪRINIŲ FORMULIŲ PAKLAIDOS ĮVERTIS

Automatinės kvadratūros algoritmuose paklaidos įvertis nulemia, ar gražinti šiuo metu gautas aproksimacijas ir nutraukti skaičiavimus, ar tęsti skaičiavimus ieškant tikslesnio rezultato. Šiuo atveju tiek efektyvumas, tiek patikimumas labai priklauso nuo paklaidos įvertinimo procedūros. Daugelyje adaptyvių algoritmų lokali paklaidos įvertis yra skaičiuojamas imant skirtumo tarp dviejų kvadratūrų aproksimacijų absoliučią vertę, tačiau tokie skaičiavimai gali būti labai nepatikimi. Vienas iš sėkmingiausių algoritmų yra QAG, kuris naudoja Gauso-Kronrodo formulę kaip pagrindinę formulę ir kurio sukūrimui buvo panaudoti eksperimentų duomenys.

Prieš dešimt metų Berntsenas ir Espelidas pristatė naują paklaidos įvertinimo metodą, naudojamą adaptyviosios kvadratūros algoritmuose. Šis metodas buvo sukurtas naudojant nulinių taisyklių seką ir gali būti taikomas kartu su daug skirtingų pagrindinių kvadratūrinių formulių.

Pagrindiniai adaptyvios kvadratūros algoritmo elementai:

- Intervalų rinkinys saugomas tam tikroje duomenų struktūroje. Tam tikru momentu gali būti M intervalų.
- Kvadratinė formulė Q , kuri pateikia lokalų integralo įvertį kiekvienam rinkinio intervalui.
- Lokali paklaidos įvertinimo procedūra, kuri randa paklaidą kiekvienoje lokaloje aproksimacijoje, kurią pateikia Q . Šis paklaidos įvertis paprastai yra gaunamas panaudojus vieną ar keletą nulinių formulių.
- Kito intervalo, kurį reikia apdoroti, parinkimo strategija. Globaliai adaptyviame algoritme tai būtų intervalas su didžiausiu paklaidos įverčiu.
- Tokio intervalo padalinimo strategija. Dažniausiai intervalas yra dalinamas pusiau.

Tarkime, kad lokalių integralų ir paklaidų įverčiai yra atitinkamai \hat{Q}_k ir \hat{E}_k . Tada globalūs įverčiai gali būti paskaičiuoti pagal formules:

$$\hat{Q} = \sum_{k=1}^M \hat{Q}_k$$

$$\hat{E} = \sum_{k=1}^M \hat{E}_k$$

Ieškant metodo paklaidos yra lyginami:

- Dviejų ar daugiau skirtingų integravimo formulių (su skirtingais tikslumais) rezultatai.
- Vienos integravimo formulės, pritaikytos visam intervalui, o po to atskiroms jo dalims, rezultatai.
- Keleto atsitiktinių formulių (su panašiais tikslumais) rezultatai.

Vienas iš būdų, kaip gauti paklaidos įvertį yra palyginti dvi kvadratūrinės formules Q_{n1} ir Q_{n2} . Yra laikoma, kad Q_{n2} duoda tikslesnius rezultatus negu Q_{n1} . Ši prielaida

$$|\hat{Q}_{n2} - I[f]| \ll |\hat{Q}_{n1} - I[f]|$$

reiškia, kad įvertis

$$\hat{Q}_{n1} - I[f] \approx \hat{Q}_{n1} - \hat{Q}_{n2}$$

turėtų būti teisingas daugumai f . \hat{Q}_{n2} euristiškai yra laikoma tikrąja verte lyginant su mažiau tikslia \hat{Q}_{n1} . $\hat{Q}_{n1} - \hat{Q}_{n2}$ tikrai pateikia informaciją apie mažiau tikslios vertės \hat{Q}_{n1} paklaidą. Tikslesnė vertė \hat{Q}_{n2} , kuriai rasti reikalingi didesni skaičiavimai, yra naudojama tik paklaidos įverčio radimui. Taip pat nėra garantijos, kad \hat{Q}_{n2} yra tikrai tikslesnė negu \hat{Q}_{n1} konkrečiam integralui. Todėl yra tikimybė, kad integravimo algoritmas pateiks bendrą paklaidą, kuri bus gerokai didesnė negu maksimalus leistinas nuokrypis.

Paprasčiausias būdas gauti du skirtingus įverčius \hat{Q}_{n1} ir \hat{Q}_{n2} yra naudoti sudėtinės formules. Q_n yra taikoma intervalui $[a, b]$, o tada dviem subintervalams $[a, (a+b)/2]$ ir $[(a+b)/2, b]$. Abiejų subintervalų rezultatų sudėjimas duoda formulę Q_{2n} , kuri yra laikoma tikslesne negu Q_n .

3.1. Kvadratūrinės formulės ir nulinės taisyklės

Integralas, kurį reikia apskaičiuoti, apibrėžiamas taip:

$$I[f] = \int_a^b f(x) dx.$$

Tarkim turim $2n+1$ taškų x_i , $i = -n, \dots, n$ intervale $[a, b]$, ir kvadratūrinę formulę paremtą tais taškais

$$Q[f] = \sum_{i=-n}^n w_i f(x_i).$$

x_i ir w_i , $i = -n, -n+1, \dots, n$, yra atitinkamai formulės mazgai ir svoriai. Ši formulė lengvai gali būti pritaikoma bet kuriam lokaliai intervalui, kurį pateikia adaptyvus algoritmas.

Kvadratinė formulė $Q[f]$ yra d laipsnio, jeigu tiksliai integruoja visus polinomus, kurių laipsnis yra mažesnis arba lygus d , ir tiksliai neintegruoja $f(x) = x^{d+1}$. Interpoliacinė kvadratinė formulė, paremta $2n+1$ skirtingais taškais, yra bent jau $2n$ laipsnio. Kvadratinė taisyklė, paremta $2n+1$ skirtingais taškais ir esanti $d \geq 2n$ laipsnio, yra unikali ir dėl to turi būti interpoliacinė.

4 teorema. Interpoliacinė kvadratinė formulė, paremta $n+1$ skirtingų taškų, yra bent jau n laipsnio. Kvadratinė formulė, paremta $n+1$ skirtingų taškų, yra $d \geq n$ laipsnio, yra unikali ir turi būti interpoliuojanti. Kvadratinė formulė, paremta $n+1$ skirtingų taškų, gali būti daugiausiai $2n+1$ laipsnio.

Formulė

$$N[f] = \sum_{i=0}^n u_i f(x_i)$$

yra nulinė taisyklė, jeigu ji turi bent vieną nenulinį svorį ir

$$\sum_{i=0}^n u_i = 0.$$

Nulinė formulė yra d laipsnio, jeigu integruoja iki nulio visus polinomus, kurių laipsnis yra mažesnis arba lygus d , ir neintegruoja $f(x) = x^{d+1}$.

Tarkim, kad Q ir N yra atitinkamai $d_Q \geq 0$ ir d_N laipsnių. Tada

$$Q_\lambda[f] = Q[f] + \lambda N[f] = \sum_{i=0}^n (w_i + \lambda u_i) f(x_i)$$

yra kvadratinė formulė, kurios laipsnis yra $d \geq \min(d_Q, d_N)$. Tada nulinę taisyklę galima užrašyti kaip skirtumą tarp dviejų kvadratinėlių formulių, kurių laipsniai yra didesni arba lygūs nuliui:

$$\lambda N[f] = Q_\lambda[f] - Q[f].$$

Tarkime turime $2n+1$ fiksuotą tašką ir kaip adaptyvaus algoritmo kvadratinė formulė yra pasirenkama unikali interpoliacinė formulė Q , kurios laipsnis yra d . Nesunkiai galima

sukonstruoti nulinių taisyklių, paremtų tais $2n+1$ taškais, seką N_1, N_2, \dots . Tarkime, kad $f[z_0, z_1, \dots, z_m]$ yra funkcijos f skirtuminis santykis, paremtas skirtingų taškų rinkiniu $\{z_0, z_1, \dots, z_m\}$, kuris yra $2n+1$ taškų poaibis:

$$f[z_0, z_1, \dots, z_m] = \sum_{j=0}^m f(z_j) / \prod_{i=0, i \neq j}^m (z_j - z_i). \quad (13)$$

Jeigu f yra pakankamai glodi, tada

$$f[z_0, z_1, \dots, z_m] = f^{(m)}(\xi_m) / m!, \quad (14)$$

kai $\xi_m \in [\min z_i, \max z_i]$. (13) formulė parodo, kad skirtuminis santykis yra linijinė funkcijos verčių kombinacija, o (14) formulė parodo, kad ši linijinė funkcijos verčių kombinacija grąžina nulines vertes visiems polinomams iki $m-1$ laipsnio ir vertę 1, kai $f(x) = x^m$. Todėl skirtuminis santykis, pateiktas (13) formulėje, yra nulinė $m-1$ laipsnio taisyklė.

3.2. Lokalios paklaidos įvertinimo algoritmas

Pagrindinės adaptyvios kvadratūros klaidų įvertinimo problemos yra tokios:

- *Asimptotinė/neasimptotinė*. Klasikinis lokalios paklaidos apskaičiavimo būdas yra paremtas tuo, kad asimptotiškai $|I - Q| < |N[f]|$, kai f yra pakankamai glodi, o h pakankamai mažas. $|N[f]|$ kaip lokalios paklaidos įverčio naudojimas adaptyviame algoritme gali duoti nepatikimus rezultatus, todėl svarbu patikrinti, ar tam tikrame intervale susidarys situacija, kuriai bus galima taikyti asimptotinę teoriją.
- *Fazės faktorius*. Fazės faktoriaus poveikis gali sugadinti $|N[f]|$ kaip realios paklaidos įvertį turint funkciją $f_\lambda(x)$, kuri priklauso nuo parametro λ . Tada $|N[f_\lambda]|$ gali būti nulis su keliomis parametro λ reikšmėmis, kai tuo tarpu reali paklaida su šiomis λ reikšmėmis nėra nulis.
- *Apvalinimo triukšmai*. Lokalios paklaidos skaičiavimas yra sukonzentruotas į tai, kad lokali apvalinimo paklaida būtų kiek galima patikimesnė ir ekonomiškesnė. Apvalinimo paklaidos gali daryti įtaką tiek lokalaus integralo įverčiui, tiek visų nulinių taisyklių įvertinimui.

Tarkime, kad turime $2n + 1$ skirtingų taškų rinkinį ir kad buvo sukurta $2n$ eilės ortonormuotų nulinių taisyklių seka. Duotam daliniam intervalui, kurio ilgis yra h , ir funkcijai f galima suskaičiuoti $2n$ lokalių produktų

$$E_j = |N_j[f]|, j = 1, 2, \dots, 2n.$$

Asimptotiškai

$$E_j = O(h^{2n+2-j}), j = 1, 2, \dots, 2n. \quad (15)$$

Lokali paklaida būtų

$$E_0 = |Q[f] - I[f]|,$$

jos asimptotinė išraiška:

$$E_0 = O(h^{d+2}), \quad (16)$$

kur $d \geq 2n$ yra taisyklės tikslumo laipsnis.

Tai reiškia, kad kai h yra pakankamai mažas ir f yra pakankamai glodi, galima tikėtis, kad

$$E_0 \ll E_1 \ll E_2 \ll \dots \ll E_{2n}. \quad (17)$$

Normuojantis daugiklis yra

$$r_j = \frac{E_j}{E_{j+1}}, j = 1, 2, \dots, K,$$

kai $K < 2n$ ir $r = \max_{j=1,2,\dots,K} r_j$.

$r = O(h)$ asimptotiškai, todėl galima tikėtis $r < 1$, kai h bus pakankamai mažas ir f bus pakankamai glodi. Tam, kad įsitikintume, ar h bus pakankamai mažas ir f bus pakankamai glodi, reikia patikrinti, ar $r < r_{kritisė}$ euristinei reikšmei $r_{kritisė} < 1$. Jeigu ši sąlyga yra patenkinama, galima taikyti optimistinę paklaidos įvertį

$$\hat{E} = cr^\alpha E_2. \quad (18)$$

Kadangi $r_0 = E_0 / E_2$ yra $O(h^{d+2-2n})$ eilės, galima parinkti α reikšmę iš intervalo $1 \leq \alpha \leq (d + 2 - 2n)$. $r < r_{kritisė}$ sąlyga vadinama stipria asimptotinio elgesio sąlyga, o $r_{kritisė} \leq r \leq 1$ - stipria asimptotinio elgesio sąlyga. Jeigu yra tenkinama pastaroji sąlyga,

$$\hat{E} = c'rE_2, \quad (19)$$

kur $c' = Cr_{kritinė}^{\alpha-1}$.

Neasimptotiniu atveju yra naudojama tokia paklaidos įvertinimo formulė:

$$\hat{E} = C \max_{j=1,2,\dots,K+1} E_j.$$

Konstanta C turėtų būti parenkama tokia, kad nebūtų situacijų, kuriose $E_0 > \max E_j$.

Pasirinkus konstantą C ir norint, kad gautūsi kiek galima glodesnė paklaidą apskaičiuojanti funkcija, nustatoma

$$c' = Cr_{kritinė}^{\alpha-1} = C,$$

t.y. $c = Cr_{kritinė}^{1-\alpha}$.

Tam, kad patikrintume, ar buvo pasiektas triukšmo lygis, atliekamas triukšmo testas. Iš pradžių visam intervalui $[a, b]$ nustatome $isabs = \sum_{i=-n}^n |w_i f(x_i)|$, o triukšmo lygis yra apibrėžiamas kaip $triukšmas = 50 * e * isabs$.

Lokaliuos paklaidos algoritmas su triukšmo testu būtų toks:

- Apskaičiuoti: $E_j = |N_j[f]|$, $j = 1, 2, \dots, K + 1$; $r_j = \frac{E_j}{E_{j+1}}$, $j = 1, 2, \dots, K$; $r = \max_{j=1,2,\dots,K} r_j$.
- Jeigu $r > 1$, tada $\hat{E} = C \max_{j=1,2,\dots,K} r_j$,
 - priešingu atveju, jeigu $r_{kritinė} \leq r$, tada $\hat{E} = CrE_2$,
 - priešingu atveju $\hat{E} = Cr_{kritinė}^{1-\alpha} r^\alpha E_2$.
- Jeigu $E_1 < triukšmas$ ir $E_2 < triukšmas$, tada $\hat{E} = 0$.

Šį algoritmą O.Espelidas panaudojo Ganderio ir Gautschi algoritmų modifikavimui. Ganderio ir Gautschi algoritmai ir jų modifikacijos gali būti aprašytos taip:

- *adaptsim.* Šį algoritmą sukūrė Ganderis ir Gautschi. Jis yra paremtas penkių taškų uždara Niutono-Koteso kvadratūrine formule. Algoritmas naudoja dalinimą pusiau lokaliai adaptyviai strategijai, pasinaudodamas rekursija. Visi funkcijos įverčiai, išskyrus penkis pirmus, apskaičiuotus pradžioje, prisideda prie galutinio įverčio.
- *modsim.* Šis algoritmas yra paremtas ta pačia kvadratūrine formule, kaip ir *adaptsim*, tačiau naudoja lokaliuos paklaidos algoritmą, pritaikytą šiai penkių taškų formulei. Po

to yra naudojama ta pati adaptyvi strategija kaip ir `adaptsim`, tačiau yra taikoma devynių taškų uždara Niutono-Koteso formulė tam, kad būtų gautas pradinis integralo įvertis, kuris būtų naudojamas santykinės paklaidos įvertinimui. Pirmajame žingsnyje yra taikomas dalinimas į keturis intervalus, o tai reiškia, kad visos apskaičiuotos funkcijos reikšmės prisideda prie galutinio įverčio.

- `adaptlob`. Šį algoritmą sukūrė Ganderis ir Gautschi. Jis yra paremtas septynių taškų Lobato-Kronrodo formule. Šis algoritmas naudoja lokalią adaptyvią strategiją su intervalo dalinimu į šešias dalis kiekviename žingsnyje. Taigi visos apskaičiuotos funkcijos reikšmės prisideda prie galutinio įverčio.
- `modlob`. Tai yra `adaptlob` algoritmo modifikacija. Šis algoritmas naudoja lokalios paklaidos algoritmą, pritaikytą septynių taškų Lobato-Kronrodo formulei tam, kad būtų padidintas patikimumas. Dalinimas į šešis intervalus yra pakeistas dalinimu pusiau. Algoritmas nenaudoja visų funkcijos reikšmių galutinio įverčio skaičiavimui.
- `coteda`. Naujas algoritmas, kuris naudojasi tiek penkių taškų uždara Niutono-Koteso formule, tiek devynių taškų uždara Niutono-Koteso formule. Algoritmas gali būti stabdomas, jeigu penkių taškų įvertis yra pakankamai geras arba jeigu devynių taškų įvertis yra pakankamai geras. Devynių taškų įverčiui reikalingi papildomi keturi taškai.

Tarkime, kad kvadratūrinės formulės metodo paklaida proporcinga h^p ir proporcingumo koeficientas turi daugiklį, lygų pointegralinės funkcijos p -osios eilės išvestinei. Simboliais \hat{Q}_h ir $\hat{Q}_{h/2}$ pažymėkime integralo $I[f]$ reikšmes, kai naudojamas atitinkamai integravimo žingsnis h ir $h/2$. Tada, laikydami, kad pointegralinės funkcijos p -osios eilės išvestinė integravimo intervale yra pastovi, galime užrašyti:

$$\begin{cases} I[f] = \hat{Q}_h + Ch^p, \\ I[f] = \hat{Q}_{h/2} + C(h/2)^p. \end{cases}$$

Iš šios lygčių sistemos apskaičiuojame $C(h/2)^p$:

$$C(h/2)^p = \frac{\hat{Q}_{h/2} - \hat{Q}_h}{2^p - 1}. \quad (20)$$

Remdamiesi (20) formule, gauname:

$$I[f] = \hat{Q}_{h/2} + \frac{\hat{Q}_{h/2} - \hat{Q}_h}{2^p - 1}. \quad (21)$$

Tarkime, kad Q yra p -osios eilės kvadratinė formulė, sukonstruota naudojant interpoliavimo mazgus $x_0 = a, x_1, x_2, \dots, x_n = b$, o Q_1 – $(p-1)$ -osios eilės kvadratinė formulė, sukonstruota naudojant tuos pačius interpoliavimo mazgus, išskyrus mazgą x_k , $k \in [0, n]$, kuris paprastai parenkamas taip, kad formulės Q_1 paklaida būtų mažiausia. Kvadratinė formulė Q_1 lengvai apskaičiuojama neapibrėžtinių koeficientų metodu, o jos paklaida – remiantis interpoliacinio polinomo liekamuoju nariu ir apibendrintąja vidurinių reikšmių teorema.

Tarkime, kad Δ ir Δ_1 yra kvadratinė formulė Q ir Q_1 paklaidos, o \hat{Q} ir \hat{Q}_1 integralo $I[f] = \int_a^b f(x) dx$ reikšmės, apskaičiuotos pagal kvadratinėse formules Q ir Q_1 .

Tada $I[f] = \hat{Q} + \Delta = \hat{Q}_1 + \Delta_1$. Iš pastarosios lygybės gauname:

$$|\hat{Q} - \hat{Q}_1| = |\Delta_1 - \Delta| \quad (22)$$

Jei $f^{(p)}(x) = \text{const}$, kai $x \in [a, b]$, tai $|\Delta_1| = |\hat{Q} - \hat{Q}_1|$. Vadinas, dydį

$$\delta_p = |\hat{Q} - \hat{Q}_1| \quad (23)$$

galima laikyti kvadratinės formulės Q paklaidos įverčiu.

Šis įvertis turi tokius privalumus:

- Išraiška pilnai sutampa su (1) formule tik su kitais tiesinio darinio koeficientais, todėl nereikia papildomai skaičiuoti $\hat{Q}_{h/2}$.
- Šis įvertis yra „pesimistiškesnis“, t.y., paprastai įvertis yra didesnis už kvadratinės formulės Q paklaidos modulio reikšmę, nes žemesnės eilės kvadratinės formulės paklaidos modulis paprastai yra didesnis nei aukštesnės eilės kvadratinės formulės paklaidos modulis, todėl naudodami šį įvertį beveik visada apskaičiuosime integralo reikšmę norimu tikslumu.

4. EKSPERIMENTINIS TYRIMAS

Buvo panaudota adaptivityviojo integravimo strategiją realizuojanti procedūra `rc8` [6]. Paklaidos įvertis buvo skaičiuojamas pagal formulę (žr. 2 lentelę):

$$\delta_8 = \left| \frac{928}{99225} h(y_0 - 8y_1 + 28y_2 - 56y_3 + 70y_4 + \dots + y_8) \right|$$

2 lentelė.

Niutono ir Koteso kvadratūros paklaidos įverčio formulė. [6]

p	w_0	w_1	w_2	w_3	w_4	w_5	Δ
4	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$	$\frac{64}{45}$	$\frac{14}{45}$		$\frac{-128}{21} \frac{f^{(6)}(\xi)}{6!} h^7$
3	$\frac{2}{9}$	$\frac{16}{9}$	0	$\frac{16}{9}$	$\frac{2}{9}$		$\frac{32}{15} \frac{f^{(4)}(\xi)}{4!} h^5$
δ_4	$\frac{4}{45}$	$\frac{-16}{45}$	$\frac{24}{45}$	$\frac{-16}{45}$	$\frac{4}{45}$		
6	$\frac{41}{140}$	$\frac{216}{140}$	$\frac{27}{140}$	$\frac{272}{140}$	$\frac{27}{140}$	$\frac{216}{140}$	$\frac{-1296}{5} \frac{f^{(8)}(\xi)}{8!} h^9$
5	$\frac{14}{50}$	$\frac{81}{50}$	0	$\frac{110}{50}$	0	$\frac{81}{50}$	$\frac{324}{35} \frac{f^{(6)}(\xi)}{6!} h^7$
δ_6	$\frac{9}{700}$	$\frac{-54}{700}$	$\frac{135}{700}$	$\frac{-180}{700}$	$\frac{135}{700}$	$\frac{-54}{700}$	
8	$\frac{3956}{14175}$	$\frac{23552}{14175}$	$\frac{-3712}{14175}$	$\frac{41984}{14175}$	$\frac{-18160}{14175}$	$\frac{41984}{14175}$	$\frac{-606208}{33} \frac{f^{(10)}(\xi)}{10!} h^{11}$
7	$\frac{1908}{6615}$	$\frac{10496}{6615}$	0	$\frac{16128}{6615}$	$\frac{-4144}{6615}$	$\frac{16128}{6615}$	$\frac{-118784}{315} \frac{f^{(8)}(\xi)}{8!} h^9$
δ_8	$\frac{-928}{99225}$	$\frac{7424}{99225}$	$\frac{-25984}{99225}$	$\frac{51968}{99225}$	$\frac{-64960}{99225}$	$\frac{51968}{99225}$	
10	$\frac{80335}{299376}$	$\frac{531500}{299376}$	$\frac{-242625}{299376}$	$\frac{1362000}{299376}$	$\frac{-1302750}{299376}$	$\frac{2136840}{299376}$	$\frac{-538540000}{273} \frac{f^{(12)}(\xi)}{12!} h^{13}$
9	$\frac{11690}{40824}$	$\frac{65125}{40824}$	0	$\frac{97500}{40824}$	$\frac{-23250}{40824}$	$\frac{106110}{40824}$	$\frac{-6470000}{99} \frac{f^{(10)}(\xi)}{10!} h^{11}$
δ_{10}	$\frac{-16175}{898128}$	$\frac{161750}{898128}$	$\frac{-727875}{898128}$	$\frac{1941000}{898128}$	$\frac{-3396750}{898128}$	$\frac{4076100}{898128}$	

Tikroji integralo reikšmė buvo apskaičiuota naudojantis programiniu paketu MathCad. 3 lentelėje pateikti atlikto eksperimento rezultatai: apskaičiuota integralo reikšmė, tikroji integralo reikšmė bei gauta paklaida. Didžiausia gauta paklaida buvo 10^{-4} eilės, mažiausia – 10^{-13} .

Integralo reikšmės, apskaičiuotos naudojantis paklaidos įverčiu, tikros integralo reikšmės bei gauta paklaida.

f(x)	Intervalas [a,b]	Apskaičiuota reikšmė	Tikra reikšmė	Paklaida
$\frac{1}{1+x}$	[0;1]	0.693147180662528	0.693147180559945	-0.000000000102583
e^x	[0;1]	1.718281828460020	1.718281828459050	-0.000000000000970
\sqrt{x}	[0;1]	0.6666666638128286	0.666666666666667	0.000000028538381
$\frac{1}{x^4 + x^2 + 0.9}$	[-1;1]	1.582232964961290	1.582232963729670	-0.000000001231620
$\sqrt{x^3}$	[0;1]	0.400000006223167	0.400000000000000	-0.000000006223167
$\frac{1}{1+x^4}$	[0;1]	0.866972987348355	0.866972987339911	-0.00000000008444
$\frac{2}{2 + \sin(10\pi x)}$	[0;1]	1.154700538494800	1.154700538379250	-0.00000000115550
$\frac{1}{1+e^x}$	[0;1]	0.379885493053485	0.379885493041722	-0.00000000011763
$\frac{\sin(100\pi x)}{\pi x}$	[0,1;1]	0.009098637586074	0.009098637539167	-0.00000000046907
$25e^{-25x}$	[0;10]	1.000000001961170	1.000000000000050	-0.000000001961120
$50\left(\frac{\sin(50\pi x)}{50\pi x}\right)^2$	[0,01;1]	0.112139303958666	0.112006721913048	-0.000132582045618
$\sqrt{50}e^{-50\pi x^2}$	[0;10]	0.500000003262837	0.500000000000000	-0.000000003262837
$\frac{50}{\pi(2500x^2 + 1)}$	[0;10]	0.499363380867984	0.499363380516716	-0.000000000351268
$\frac{1}{x^2 + 1.005}$	[-1;1]	1.564396445030990	1.564396444069050	-0.000000000961940
$4\pi^2 x \sin(20\pi x) \cos(2\pi x)$	[0;1]	-0.634665182501960	-0.634665182543398	-0.00000000041438
$\frac{1}{1 + (230x - 30)^2}$	[0;1]	0.013492485651404	0.013492473531959	-0.000000012119445

Buvo surasti funkcijų neapibrėžtinai integralai, apskaičiuotos jų reikšmės daliniuose intervaluose ir palygintos su daliniuose intervaluose apskaičiuotomis integralo reikšmėmis. Gauti rezultatai (4 lentelė) parodė, kad visų nagrinėtų funkcijų visuose daliniuose intervaluose skirtumas tarp tikrosios integralo reikšmės ir apskaičiuotos reikšmės neviršijo nurodytos galimos skaičiavimo paklaidos (0.000001).

Daliniai intervalai, integralo reikšmės daliniuose intervaluose, apskaičiuotos naudojantis paklaidos įverčiu, ir tikros integralo reikšmės daliniuose intervaluose.

f(x)	Intervalas $[\alpha, \beta]$	Apskaičiuota reikšmė	Tikra reikšmė	
e^x	[0.0000000000000000E+0000; 1.0000000000000000E+0000]	1.7182818284600219E+0000	1.7182818284590452E+0000	
\sqrt{x}	[0.0000000000000000E+0000; 4.8828125000000000E-0004]	7.1648163765960099E-0006	7.1930621458592481E-0006	
	[4.8828125000000000E-0004; 9.7656250000000000E-0001]	1.3151989931701402E-0005	1.3151989937474085E-0005	
	[9.7656250000000000E-0001; 1.9531250000000000E-0003]	3.7199445067213038E-0005	3.7199445083540651E-0005	
	[1.9531250000000000E-0003; 3.9062500000000000E-0003]	1.0521591945361122E-0004	1.0521591949979268E-0004	
	[3.9062500000000000E-0003; 7.8125000000000000E-0003]	2.9759556053770430E-0004	2.9759556066832521E-0004	
	[7.8125000000000000E-0003; 1.5625000000000000E-0002]	8.4172735562888973E-0004	8.4172735599834146E-0004	
	[1.5625000000000000E-0002; 3.1250000000000000E-0002]	2.3807644843016344E-0003	2.3807644853466017E-0003	
	[3.1250000000000000E-0002; 6.2500000000000000E-0002]	6.7338188450311178E-0003	6.7338188479867316E-0003	
	[6.2500000000000000E-0002; 1.2500000000000000E-0001]	1.9046115874413075E-0002	1.9046115882772814E-0002	
	[1.2500000000000000E-0001; 2.5000000000000000E-0001]	5.3870550760248943E-0002	5.3870550783893853E-0002	
	[2.5000000000000000E-0001; 5.0000000000000000E-0001]	1.5236892699530460E-0001	1.5236892706218251E-0001	
	[5.0000000000000000E-0001; 1.0000000000000000E+0000]	4.3096440608199154E-0001	4.3096440627115083E-0001	
	$\sqrt{x^3}$	[0.0000000000000000E+0000; 3.1250000000000000E-0002]	6.9059590731872503E-0005	6.9053396600248782E-0005
		[3.1250000000000000E-0002; 6.2500000000000000E-0002]	3.2157160342309804E-0004	3.2157160339975122E-0004
[6.2500000000000000E-0002; 1.2500000000000000E-0001]		1.8190836913400306E-0003	1.8190836912079610E-0003	
[1.2500000000000000E-0001; 2.5000000000000000E-0001]		1.0290291309539137E-0002	1.0290291308792039E-0002	
[2.5000000000000000E-0001; 5.0000000000000000E-0001]		5.8210678122880979E-0002	5.8210678118654752E-0002	
[5.0000000000000000E-0001; 1.0000000000000000E+0000]		3.2928932190525240E-0001	3.2928932188134525E-0001	
$\frac{1}{1+e^x}$		[0.0000000000000000E+0000; 1.0000000000000000E+0000]	3.7988549305348535E-0001	3.7988549304172248E-0001
$25e^{-25x}$	[0.0000000000000000E+0000; 3.9062500000000000E-0002]	6.2339654928939958E-0001	6.2339654928911959E-0001	
	[3.9062500000000000E-0002; 7.8125000000000000E-0002]	2.3477329162364332E-0001	2.3477329162353788E-0001	
	[7.8125000000000000E-0002; 1.5625000000000000E-0001]	1.2171436511125566E-0001	1.2171436506060164E-0001	
	[1.5625000000000000E-0001; 2.3437500000000000E-0001]	1.7262767766944284E-0002	1.7262767759760016E-0002	
	[2.3437500000000000E-0001; 3.1250000000000000E-0001]	2.4483810986735566E-0003	2.4483810976546106E-0003	
	[3.1250000000000000E-0001; 4.6875000000000000E-0001]	3.9650552869333316E-0004	3.9650541044618167E-0004	
	[4.6875000000000000E-0001; 6.2500000000000000E-0001]	7.9760235456590901E-0006	7.9760211670237464E-0006	
	[6.2500000000000000E-0001; 1.2500000000000000E+0000]	1.6551899013371109E-0007	1.6373768624904259E-0007	
	[1.2500000000000000E+0000; 2.5000000000000000E+0000]	3.2719093271113355E-0014	2.6810038677817313E-0014	
	[2.5000000000000000E+0000; 5.0000000000000000E+0000]	1.5709524487608272E-0027	7.1877817390609886E-0028	
	[5.0000000000000000E+0000; 1.0000000000000000E+0001]	2.2529078424862725E-0054	5.1664206328378609E-0055	
	$\frac{1}{x^2 + 1.005}$	[-1.0000000000000000E+0000; -5.0000000000000000E-0001]	3.2069968247417376E-0001	3.2069968254045156E-0001
		[-5.0000000000000000E-0001;	4.6149854004132170E-0001	4.6149853949407333E-0001

	[0.0000000000000000E+0000]		
	[0.0000000000000000E+0000; 5.0000000000000000E-0001]	4.6149854004132170E-0001	4.6149853949407333E-0001
	[5.0000000000000000E-0001; 1.0000000000000000E+0000]	3.2069968247417376E-0001	3.2069968254045156E-0001
$\frac{1}{1 + (230x - 30)^2}$	[0.0000000000000000E+0000 3.1250000000000000E-0002]	4.5593805567070676E-0005	4.5593805564664871E-0005
	[3.1250000000000000E-0002; 6.2500000000000000E-0002]	8.7414181113636839E-0005	8.7414181002669154E-0005
	[6.2500000000000000E-0002; 7.8125000000000000E-0002]	8.2667120854886643E-0005	8.2667120852406365E-0005
	[7.8125000000000000E-0002; 9.3750000000000000E-0002]	1.5235630221652880E-0004	1.5235630213155070E-0004
	[9.3750000000000000E-0002; 1.0156250000000000E-0001]	1.3694334379460255E-0004	1.3694334379337404E-0004
	[1.0156250000000000E-0001; 1.0937500000000000E-0001]	2.3533077518871914E-0004	2.3533077516649845E-0004
	[1.0937500000000000E-0001; 1.1718750000000000E-0001]	4.9363836451036493E-0004	4.9363836407029990E-0004
	[1.1718750000000000E-0001; 1.2109375000000000E-0001]	5.1523152781409753E-0004	5.1523152784566885E-0004
	[1.2109375000000000E-0001; 1.2304687500000000E-0001]	4.1866795107768518E-0004	4.1866795107792326E-0004
	[1.2304687500000000E-0001; 1.2500000000000000E-0001]	6.2093900391007581E-0004	6.2093900390931601E-0004
	[1.2500000000000000E-0001; 1.2695312500000000E-0001]	9.6016631266202099E-0004	9.6016631264171324E-0004
	[1.2695312500000000E-0001; 1.2890625000000000E-0001]	1.4658739555464569E-0003	1.4658739557692366E-0003
	[1.2890625000000000E-0001; 1.3085937500000000E-0001]	1.8931030272564683E-0003	1.8931030274958659E-0003
	[1.3085937500000000E-0001; 1.3281250000000000E-0001]	1.7525793035866158E-0003	1.7525793028698715E-0003
	[1.3281250000000000E-0001; 1.3476562500000000E-0001]	1.2304365459572940E-0003	1.2304365460254680E-0003
	[1.3476562500000000E-0001; 1.3671875000000000E-0001]	7.9170252278277212E-0004	7.9170252277371288E-0004
	[1.3671875000000000E-0001; 1.4062500000000000E-0001]	8.7818280862944742E-0004	8.7818280915197734E-0004
	[1.4062500000000000E-0001; 1.4453125000000000E-0001]	4.5263211976107592E-0004	4.5263211977315255E-0004
	[1.4453125000000000E-0001; 1.4843750000000000E-0001]	2.7045381463443355E-0004	2.7045381463419010E-0004
	[1.4843750000000000E-0001; 1.5625000000000000E-0001]	3.0485459314537135E-0004	3.0485459306677476E-0004
	[1.5625000000000000E-0001; 1.6406250000000000E-0001]	1.6641538651968154E-0004	1.6641538651610278E-0004
	[1.6406250000000000E-0001; 1.7187500000000000E-0001]	1.0453929538139113E-0004	1.0453929538112090E-0004
	[1.7187500000000000E-0001; 1.8750000000000000E-0001]	1.2387836647947511E-0004	1.2387836645352424E-0004
	[1.8750000000000000E-0001; 2.1875000000000000E-0001]	1.1675002379002006E-0004	1.1675002319187028E-0004
	[2.1875000000000000E-0001; 2.5000000000000000E-0001]	5.5841012729566078E-0005	5.5841012721592388E-0005
	[2.5000000000000000E-0001; 3.1250000000000000E-0001]	5.4224182887737266E-0005	5.4224182682211582E-0005
	[3.1250000000000000E-0001; 3.7500000000000000E-0001]	2.6522407631241025E-0005	2.6522407628220689E-0005
	[3.7500000000000000E-0001; 5.0000000000000000E-0001]	2.6138009205103602E-0005	2.6138009121141287E-0005
	[5.0000000000000000E-0001; 7.5000000000000000E-0001]	2.0637976618781802E-0005	2.0637976008020764E-0005
	[7.5000000000000000E-0001; 1.0000000000000000E+0000]	8.7716101516344717E-0006	8.7716101476332618E-0006

Daliniuose intervaluose apskaičiuotos realios paklaidos buvo palygintos su paklaidų įverčiais. 5 lentelėje pateikti rezultatai parodo, kokios yra realios paklaidos ir paklaidų įverčiai daliniuose intervaluose. Taip pat išskirti atvejai (patamsintos eilutės), kai paklaidų įverčiai yra per daug

optimistiniai, t.y. kai reali paklaida yra didesnė negu jos įvertis, tačiau tokie atvejai yra pavieniai ir tarp jų nėra kažkokio dėsningumo.

5 lentelė.

Daliniai intervalai, realios dalinio intervalo paklaidos ir dalinių intervalų paklaidų įverčiai.

f(x)	Intervalas $[\alpha, \beta]$	Reali paklaida	Paklaidos įvertis	
e^x	[0.0000000000000000E+0000; 1.0000000000000000E+0000]	9.7667783149207876E-0013	1.1548508080817116E-0010	
\sqrt{x}	[0.0000000000000000E+0000; 4.8828125000000000E-0004]	2.8245769263238142E-0008	1.6435573807922811E-0009	
	[4.8828125000000000E-0004; 9.7656250000000000E-0000]	5.7726832190784243E-0015	2.2105270149159132E-0014	
	[9.7656250000000000E-0001; 1.9531250000000000E-0003]	1.6327613797091525E-0014	6.2523145689860935E-0014	
	[1.9531250000000000E-0003; 3.9062500000000000E-0003]	4.6181465752627394E-0014	1.7684216119327306E-0013	
	[3.9062500000000000E-0003; 7.8125000000000000E-0003]	1.3062091037673220E-0013	5.0018516551888748E-0013	
	[7.8125000000000000E-0003; 1.5625000000000000E-0002]	3.6945172602101915E-0013	1.4147372895461845E-0012	
	[1.5625000000000000E-0002; 3.1250000000000000E-0002]	1.0449672830138576E-0012	4.0014813241510998E-0012	
	[3.1250000000000000E-0002; 6.2500000000000000E-0002]	2.9556138081681532E-0012	1.1317898316369476E-0011	
	[6.2500000000000000E-0002; 1.2500000000000000E-0001]	8.3597382641108610E-0012	3.2011850593208799E-0011	
	[1.2500000000000000E-0001; 2.5000000000000000E-0001]	2.3644910465345226E-0011	9.0543186530955805E-0011	
	[2.5000000000000000E-0001; 5.0000000000000000E-0001]	6.6877906112886888E-0011	2.5609480474567039E-0010	
	[5.0000000000000000E-0001; 1.0000000000000000E+0000]	1.8915928372276181E-0010	7.2434549224764644E-0010	
	$\sqrt{x^3}$	[0.0000000000000000E+0000; 3.1250000000000000E-0002]	6.1941316237212814E-0009	8.2066192800986430E-0010
		[3.1250000000000000E-0002; 6.2500000000000000E-0002]	2.3346824711707444E-0014	1.1804245600734222E-0013
		[6.2500000000000000E-0002; 1.2500000000000000E-0001]	1.3206958455308347E-0013	6.6774896888324461E-0013
[1.2500000000000000E-0001; 2.5000000000000000E-0001]		7.4709839077463820E-0013	3.7773585922349510E-0012	
[2.5000000000000000E-0001; 5.0000000000000000E-0001]		4.2262267056986710E-0012	2.1367967004263827E-0011	
[5.0000000000000000E-0001; 1.0000000000000000E+0000]		2.3907148504788422E-0011	1.2087547495151843E-0010	
$\frac{1}{1+e^x}$	[0.0000000000000000E+0000; 1.0000000000000000E+0000]	1.1762877211987308E-0011	1.5906243896588628E-0010	
$25e^{-25x}$	[0.0000000000000000E+0000; 3.9062500000000000E-0002]	2.7999006108406221E-0013	3.4714940048528474E-0011	
	[3.9062500000000000E-0002; 7.8125000000000000E-0002]	1.0544522069735884E-0013	1.3073766213181093E-0011	
	[7.8125000000000000E-0002; 1.5625000000000000E-0001]	5.0654026166694482E-0011	1.5702347932940358E-0009	
	[1.5625000000000000E-0001; 2.3437500000000000E-0001]	7.1842685931024350E-0012	2.2270665053793346E-0010	
	[2.3437500000000000E-0001; 3.1250000000000000E-0001]	1.0189459580180627E-0012	3.1586519675518654E-0011	
	[3.1250000000000000E-0001; 4.6875000000000000E-0001]	1.1824715149131069E-0010	9.1674079195908720E-0010	
	[4.6875000000000000E-0001; 6.2500000000000000E-0001]	2.3786353436513223E-0012	1.8440968946960048E-0011	
	[6.2500000000000000E-0001; 1.2500000000000000E+0000]	1.7813038846684996E-0009	8.7981550368672921E-0010	
	[1.2500000000000000E+0000; 2.5000000000000000E+0000]	5.9090545932960413E-0015	8.3249647847047684E-0016	
	[2.5000000000000000E+0000; 5.0000000000000000E+0000]	8.5217427485472836E-0028	5.2348665417633098E-0029	
	[5.0000000000000000E+0000; 1.0000000000000000E+0001]	1.7362657792024864E-0054	7.5498111410842954E-0056	

$\frac{1}{x^2 + 1.005}$	[-1.0000000000000000E+0000; -5.0000000000000000E-0001]	6.6277794044016305E-0011	5.9505890897404304E-0010
	[-5.0000000000000000E-0001; 0.0000000000000000E+0000]	5.4724836804059124E-0010	2.0290625355517326E-0009
	[0.0000000000000000E+0000; 5.0000000000000000E-0001]	5.4724836804059124E-0010	2.0290625355517326E-0009
	[5.0000000000000000E-0001; 1.0000000000000000E+0000]	6.6277794044016305E-0011	5.9505890897404304E-0010
$\frac{1}{1 + (230x - 30)^2}$	[0.0000000000000000E+0000 3.1250000000000000E-0002]	2.4058048565189277E-0015	3.4690345954906189E-0014
	[3.1250000000000000E-0002; 6.2500000000000000E-0002]	1.1096768535456022E-0013	8.4334645313597580E-0013
	[6.2500000000000000E-0002; 7.8125000000000000E-0002]	2.4802779387703265E-0015	4.0525840507829899E-0014
	[7.8125000000000000E-0002; 9.3750000000000000E-0002]	8.4978103977254851E-0014	7.8002868615677370E-0013
	[9.3750000000000000E-0002; 1.0156250000000000E-0001]	1.2285080787450062E-0015	2.6731263531373206E-0014
	[1.0156250000000000E-0001; 1.0937500000000000E-0001]	2.2220692224576034E-0014	3.2049550109162994E-0013
	[1.0937500000000000E-0001; 1.1718750000000000E-0001]	4.4006503022128208E-0013	6.4113049206068987E-0012
	[1.1718750000000000E-0001; 1.2109375000000000E-0001]	3.1571317482628227E-0014	9.5178360790308014E-0014
	[1.2109375000000000E-0001; 1.2304687500000000E-0001]	2.3808359729776511E-0016	8.0670472892280559E-0015
	[1.2304687500000000E-0001; 1.2500000000000000E-0001]	7.5980898835695991E-0016	3.9580266957366400E-0014
	[1.2500000000000000E-0001; 1.2695312500000000E-0001]	2.0307748000473688E-0014	2.3636816283987498E-0013
	[1.2695312500000000E-0001; 1.2890625000000000E-0001]	2.2277978367758977E-0013	1.6546842255033208E-0013
	[1.2890625000000000E-0001; 1.3085937500000000E-0001]	2.3939757473312700E-0013	3.4178753789121234E-0012
	[1.3085937500000000E-0001; 1.3281250000000000E-0001]	7.1674431088270771E-0013	5.0465339543767764E-0012
	[1.3281250000000000E-0001; 1.3476562500000000E-0001]	6.8174061282165596E-0014	9.9743209965387391E-0013
	[1.3476562500000000E-0001; 1.3671875000000000E-0001]	9.0592402464289884E-0015	1.0382695854739200E-0014
	[1.3671875000000000E-0001; 1.4062500000000000E-0001]	5.2252992095194707E-0013	4.9449793452981679E-0012
	[1.4062500000000000E-0001; 1.4453125000000000E-0001]	1.2076629102564772E-0014	5.2921013483188087E-0015
	[1.4453125000000000E-0001; 1.4843750000000000E-0001]	2.4344341002975351E-0016	1.8504160274392157E-0014
	[1.4843750000000000E-0001; 1.5625000000000000E-0001]	7.8596584090071656E-0014	9.8642533226409172E-0013
	[1.5625000000000000E-0001; 1.6406250000000000E-0001]	3.5787640844490783E-0015	6.6386017068924241E-0014
	[1.6406250000000000E-0001; 1.7187500000000000E-0001]	2.7023095933556877E-0016	7.4303027243337576E-0015
	[1.7187500000000000E-0001; 1.8750000000000000E-0001]	2.5950874592124031E-0014	2.8873894018929264E-0013
	[1.8750000000000000E-0001; 2.1875000000000000E-0001]	5.9814978199283516E-0013	3.4293383275663406E-0012
	[2.1875000000000000E-0001; 2.5000000000000000E-0001]	7.9736902724708350E-0015	9.4115075687000551E-0014
	[2.5000000000000000E-0001; 3.1250000000000000E-0001]	2.0552568445200683E-0013	1.2429781665172868E-0012
	[3.1250000000000000E-0001; 3.7500000000000000E-0001]	3.0203360311751381E-0015	3.7178449510514623E-0014
	[3.7500000000000000E-0001; 5.0000000000000000E-0001]	8.3962314257095300E-0014	5.2429379353465834E-0013
	[5.0000000000000000E-0001; 7.5000000000000000E-0001]	6.1076103790997655E-0013	2.4168723335514640E-0012
	[7.5000000000000000E-0001; 1.0000000000000000E+0000]	4.0012099167027052E-0015	3.7149176510839357E-0014

6-24 lentelėse pateikti procedūros rc8 skaičiavimo rezultatai – iteracijų, kurios buvo reikalingos surasti integralo reikšmę, skaičius. adaptsim, modsim, adaptlob, modlob, coteda procedūrų rezultatai paimti iš [2]. rc8norm procedūra atitinką rc8 procedūrą, tik paklaidos įvertis yra normuojamas tokiu būdu:

$$norm(a) = \sqrt{a_1^2 + \dots + a_9^2}, \quad norm(b) = \sqrt{b_1^2 + \dots + b_9^2},$$

kur a_1, \dots, a_9 - 8 eilės Niutono-Koteso paklaidos įverčio formulės koeficientai (žr. 2 lentelę), b_1, \dots, b_9 - 7 eilės nulinės taisyklės koeficientai (žr. 2 lentelę).

$$K = \frac{norm(a)}{norm(b)}$$

$$\delta_8 = \left| \frac{928}{99225} h(y_0 - 8y_1 + 28y_2 - 56y_3 + 70y_4 + \dots + y_8) \right| / K$$

Atlikus rc8 ir rc8norm procedūrų palyginimą su adaptsim, modsim, adaptlob, modlob, coteda, gauti rezultatai parodė, kad rc8 ir rc8norm procedūros pateikia rezultatus visais atvejais, netgi kai to nepadaro geriausiai rezultatus parodžiusi coteda procedūra. Tačiau daugeliu atvejų rc8 ir rc8norm procedūrų pateikti rezultatai yra didesni negu coteda procedūros. Lyginant rc8 ir rc8norm procedūras galima pastebėti, kad rc8norm pateikti rezultatai blogiausiu atveju yra tokie pat kaip rc8 procedūros, tačiau dauguma atveju jie yra geresni.

6 lentelė.

$$\text{Integralas } \int_0^1 e^x dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	10	17	18	17	9	9	9
10^{-3}	10	17	18	17	9	9	9
10^{-4}	10	17	18	17	9	9	9
10^{-5}	14	17	18	17	9	9	9
10^{-6}	22	17	18	17	9	9	9
10^{-7}	22	29	18	17	9	9	9
10^{-8}	38	33	18	17	9	9	9
10^{-9}	58	49	18	17	17	17	17
10^{-10}	70	65	18	17	17	17	17
10^{-11}	134	81	18	17	17	33	25
10^{-12}	218	129	18	27	33	33	33

7 lentelė.

Integralas $\int_0^1 f(x)dx$, kur $f = 1$, jeigu $x > 0.3$, kitais atvejais – $f = 0$.

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-10	33	48	57	29	57	41
10^{-2}	-18	49	78	87	45	89	57
10^{-3}	-34	65	108	127	61	121	89
10^{-4}	-42	73	168	157	69	137	121
10^{-5}	-58	89	198	187	85	169	137
10^{-6}	-74	105	198	227	101	201	169
10^{-7}	-82	113	258	257	109	217	201
10^{-8}	-98	129	288	287	125	249	217
10^{-9}	-114	145	318	327	141	265	249
10^{-10}	-122	153	378	357	149	297	281
10^{-11}	-138	169	408	387	165	329	297
10^{-12}	-154	185	438	427	181	345	329

8 lentelė.

Integralas $\int_0^1 \sqrt{x}dx$.

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	-10	25	18	37	21	25	17
10^{-3}	-18	37	48	57	29	49	33
10^{-4}	-26	45	78	87	41	65	49
10^{-5}	-38	57	78	107	57	81	65
10^{-6}	-54	73	108	127	73	97	89
10^{-7}	-82	101	168	147	89	129	105
10^{-8}	-126	141	228	187	113	161	137
10^{-9}	-190	201	288	237	161	209	185
10^{-10}	-290	265	438	277	193	265	233
10^{-11}	-498	365	558	347	241	329	297
10^{-12}	-750	505	708	437	297	417	369

9 lentelė.

$$\text{Integralas } \int_{-1}^1 \frac{1}{x^4 + x^2 + 0.9} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	37	9	17	9
10^{-2}	14	17	18	37	17	17	17
10^{-3}	14	41	18	37	17	17	17
10^{-4}	22	41	18	37	33	33	17
10^{-5}	38	49	48	57	33	33	33
10^{-6}	46	89	48	77	33	33	33
10^{-7}	70	129	48	77	49	65	65
10^{-8}	118	177	48	117	65	65	65
10^{-9}	174	249	108	197	81	113	65
10^{-10}	254	337	168	237	113	129	129
10^{-11}	486	457	168	277	129	129	129
10^{-12}	662	649	228	337	129	241	177

10 lentelė.

$$\text{Integralas } \int_0^1 \sqrt{x^3} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	10	17	18	17	9	9	9
10^{-3}	10	17	18	17	9	17	9
10^{-4}	18	21	48	37	17	33	25
10^{-5}	22	29	48	47	25	41	33
10^{-6}	30	37	48	57	37	49	41
10^{-7}	54	49	78	77	45	73	57
10^{-8}	70	69	108	87	57	89	73
10^{-9}	106	93	138	107	77	105	89
10^{-10}	190	137	198	147	93	153	129
10^{-11}	266	177	258	177	121	185	161
10^{-12}	414	261	318	207	157	233	193

11 lentelė.

$$\text{Integralas } \int_0^1 \frac{1}{1+x^4} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	21	18	17	9	9	9
10^{-2}	10	21	18	17	9	9	9
10^{-3}	14	25	18	17	17	9	9
10^{-4}	14	29	18	17	17	17	17
10^{-5}	22	45	18	27	17	17	17
10^{-6}	34	65	18	67	33	33	17
10^{-7}	46	93	48	67	33	33	33
10^{-8}	66	133	48	87	41	57	33
10^{-9}	126	193	48	97	49	65	57
10^{-10}	162	261	48	97	65	65	65
10^{-11}	254	357	108	147	89	113	73
10^{-12}	482	489	168	197	105	129	121

12 lentelė.

$$\text{Integralas } \int_0^1 \frac{2}{2 + \sin(10\pi x)} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	89	48	97	65	9	9
10^{-2}	38	113	138	177	89	65	65
10^{-3}	-54	193	198	277	129	113	81
10^{-4}	-94	281	198	317	177	161	129
10^{-5}	134	369	288	397	225	193	193
10^{-6}	198	497	498	557	257	273	209
10^{-7}	-358	761	648	617	337	369	321
10^{-8}	542	969	858	757	401	401	369
10^{-9}	-750	1449	1188	1057	497	593	481
10^{-10}	1438	1889	1698	1297	609	737	657
10^{-11}	2134	2753	2538	1657	769	865	753
10^{-12}	2982	3657	3258	1997	929	1377	1105

13 lentelė.

$$\text{Integralas } \int_0^1 \frac{1}{1+x} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	10	17	18	17	9	9	9
10^{-3}	10	17	18	17	9	9	9
10^{-4}	14	17	18	17	9	9	9
10^{-5}	18	17	18	17	9	17	9
10^{-6}	26	25	18	17	17	17	17
10^{-7}	34	33	18	17	17	25	17
10^{-8}	54	45	48	27	25	33	25
10^{-9}	82	61	48	27	25	41	33
10^{-10}	118	81	48	37	33	49	41
10^{-11}	202	113	48	47	41	65	57
10^{-12}	302	161	48	57	49	81	65

14 lentelė.

$$\text{Integralas } \int_0^1 \frac{1}{1+e^x} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	10	17	18	17	9	9	9
10^{-3}	10	17	18	17	9	9	9
10^{-4}	10	17	18	17	9	9	9
10^{-5}	14	17	18	17	9	9	9
10^{-6}	14	21	18	17	9	9	9
10^{-7}	22	29	18	17	9	17	9
10^{-8}	34	33	18	17	9	17	17
10^{-9}	38	53	18	27	9	17	17
10^{-10}	66	69	18	37	17	33	17
10^{-11}	122	97	48	47	17	33	33
10^{-12}	134	133	48	57	25	33	33

15 lentelė.

$$\text{Integralas } \int_{0.1}^1 \frac{\sin(100\pi x)}{\pi x} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-22	669	-198	717	513	265	17
10^{-2}	-218	857	678	867	513	473	305
10^{-3}	-470	1273	768	1327	633	513	489
10^{-4}	-658	1857	1038	1567	961	585	513
10^{-5}	1014	2601	1368	1847	1025	985	825
10^{-6}	-1706	3581	2328	2357	1089	1017	1009
10^{-7}	2694	5213	3138	3127	1817	1465	1041
10^{-8}	4030	6945	4398	3847	2017	2033	1913
10^{-9}	6886	10157	5748	4787	2049	2129	2041
10^{-10}	10790	13649	6918	6157	3273	3593	2297
10^{-11}	16126	19593	12468	7597	3977	4049	3953
10^{-12}	27614	26337	15288	9307	4081	4785	4073

16 lentelė.

$$\text{Integralas } \int_0^{10} \sqrt{50} e^{-50\pi^2 x} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-14	37	48	77	33	57	49
10^{-2}	-26	49	78	87	37	57	57
10^{-3}	-34	61	78	87	45	65	65
10^{-4}	-38	65	108	97	49	65	65
10^{-5}	42	89	138	117	53	73	65
10^{-6}	54	105	168	127	69	97	89
10^{-7}	-70	165	168	157	93	105	97
10^{-8}	-102	197	228	167	97	137	113
10^{-9}	-134	293	288	197	125	177	153
10^{-10}	234	361	468	247	153	185	177
10^{-11}	342	557	528	357	173	233	217
10^{-12}	-458	697	678	397	193	289	281

17 lentelė.

$$\text{Integralas } \int_0^{10} 25e^{-25x} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-14	37	78	57	29	49	49
10^{-2}	-26	37	78	67	33	57	49
10^{-3}	-34	53	78	67	41	57	57
10^{-4}	38	65	108	77	49	65	57
10^{-5}	-42	77	138	87	53	73	65
10^{-6}	-50	101	168	97	69	89	81
10^{-7}	-74	141	168	117	89	97	89
10^{-8}	-98	181	198	137	97	121	113
10^{-9}	-138	257	288	167	129	161	129
10^{-10}	-230	333	498	187	157	185	177
10^{-11}	-354	481	618	237	189	241	209
10^{-12}	-498	629	708	287	225	321	257

18 lentelė.

$$\text{Integralas } \int_0^{10} \frac{50}{\pi(2500x^2 + 1)} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-14	37	78	77	33	65	49
10^{-2}	-26	49	108	87	41	73	65
10^{-3}	-34	49	108	107	41	81	73
10^{-4}	-42	73	108	107	65	81	81
10^{-5}	-46	105	168	117	81	89	81
10^{-6}	-62	137	168	117	97	121	105
10^{-7}	-90	193	258	177	137	169	137
10^{-8}	-146	265	378	217	169	209	169
10^{-9}	-226	373	528	287	193	273	225
10^{-10}	346	509	648	347	249	353	281
10^{-11}	578	721	1008	427	305	433	369
10^{-12}	-870	993	1308	547	377	601	481

19 lentelė.

$$\text{Integralas } \int_{0.01}^1 50 \left(\frac{\sin(50\pi x)}{50\pi x} \right)^2 dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-10	69	78	67	41	73	65
10^{-2}	-26	197	78	227	161	137	73
10^{-3}	-34	485	-168	557	-261	209	185
10^{-4}	-42	-689	-498	947	-297	281	249
10^{-5}	-90	1237	918	1197	-393	353	297
10^{-6}	-238	1709	1008	1587	905	833	641
10^{-7}	-534	2505	1458	1917	1065	1065	937
10^{-8}	990	3529	2718	2347	1185	1241	1105
10^{-9}	-1438	4905	3588	3007	1625	1689	1321
10^{-10}	-2282	6817	4518	3877	2017	2257	2033
10^{-11}	-3962	9557	6318	4877	2273	2705	2337
10^{-12}	5798	13433	8568	6187	2849	3881	2905

20 lentelė.

$$\text{Integralas } \int_0^{\pi} \cos(\cos(x) + 3\sin(x) + 2\cos(2x) + 3\cos(3x)) dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	30	85	48	97	49	41	33
10^{-2}	34	117	78	137	57	65	57
10^{-3}	58	153	168	147	73	73	65
10^{-4}	102	245	198	227	113	97	89
10^{-5}	170	305	228	267	137	129	105
10^{-6}	218	425	258	417	177	137	129
10^{-7}	410	625	498	497	209	225	185
10^{-8}	658	869	618	587	257	257	233
10^{-9}	894	1201	768	697	329	353	281
10^{-10}	1626	1661	1128	837	385	425	369
10^{-11}	2614	2353	1368	1127	473	569	497
10^{-12}	3566	3253	2118	1437	569	753	633

21 lentelė.

Integralas $\int_0^1 f(x)dx$, kur $f = \ln(x)$, jeigu $x > 10^{-15}$, kitais atvejais $f = 0$.

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-10	49	48	77	45	41	25
10^{-2}	-26	65	78	117	57	81	57
10^{-3}	-46	81	108	147	73	105	89
10^{-4}	-58	97	138	187	97	137	113
10^{-5}	-86	125	198	217	125	169	145
10^{-6}	-134	169	258	257	149	209	177
10^{-7}	-194	225	348	307	193	257	217
10^{-8}	-286	301	468	387	253	337	281
10^{-9}	498	417	648	447	305	417	345
10^{-10}	-730	565	858	557	385	521	425
10^{-11}	-1098	785	1188	677	489	689	561
10^{-12}	-1914	1085	1758	827	601	857	705

22 lentelė.

Integralas $\int_{-1}^1 \frac{1}{1.005 + x^2} dx$.

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	10	17	18	17	9	9	9
10^{-2}	14	17	18	17	9	17	17
10^{-3}	14	25	18	17	17	17	17
10^{-4}	22	41	18	37	17	33	17
10^{-5}	38	65	18	37	33	33	33
10^{-6}	38	81	48	57	33	33	33
10^{-7}	70	121	48	57	49	65	33
10^{-8}	118	161	48	117	49	65	65
10^{-9}	142	217	48	137	65	65	65
10^{-10}	254	321	48	177	81	129	97
10^{-11}	318	457	168	197	113	129	129
10^{-12}	542	601	168	277	129	177	129

23 lentelė.

$$\text{Integralas } \int_0^1 4\pi^2 x \sin(20\pi x) \cos(2\pi x) dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	-10	185	168	187	129	105	89
10^{-2}	-10	229	198	257	129	121	113
10^{-3}	-10	297	288	367	177	193	129
10^{-4}	-10	465	588	447	257	257	225
10^{-5}	-10	689	678	607	257	257	257
10^{-6}	-10	981	1008	787	289	481	345
10^{-7}	-10	1417	1128	927	465	505	497
10^{-8}	-10	1913	1848	1097	505	537	505
10^{-9}	-10	2777	2928	1437	529	993	809
10^{-10}	-10	3761	3498	1857	833	1017	993
10^{-11}	-10	5425	5058	2327	985	1433	1017
10^{-12}	-10	7369	6078	2907	1033	1985	1833

24 lentelė.

$$\text{Integralas } \int_0^1 \frac{1}{1+(230x-30)^2} dx.$$

Leistinas nuokrypis	adaptsim	modsim	adaptlob	modlob	coteda	rc8	rc8norm
10^{-1}	50	57	108	117	49	81	81
10^{-2}	58	61	108	127	49	97	97
10^{-3}	94	81	168	167	73	105	97
10^{-4}	142	129	198	217	97	153	121
10^{-5}	198	153	288	247	121	209	169
10^{-6}	354	209	348	307	157	241	217
10^{-7}	530	317	468	377	209	313	265
10^{-8}	782	425	738	477	241	393	353
10^{-9}	1402	597	1038	567	305	513	433
10^{-10}	2118	817	1218	757	369	681	561
10^{-11}	3110	1165	1608	907	457	897	737
10^{-12}	5606	1601	2538	1157	545	1161	945

5. IŠVADOS

1. K.Pluko pasiūlytu metodu buvo apskaičiuoti [3](#) lentelėje nurodytų funkcijų integralai bei palyginti su tikromis integralų reikšmėmis. Didžiausia gauta paklaida buvo 10^{-4} eilės, mažiausia – 10^{-13} . Toks pat eksperimentas buvo atliktas ir daliniuose intervaluose. Gauti rezultatai ([4](#) lentelė) parodė, kad visų nagrinėtų funkcijų visuose daliniuose intervaluose skirtumas tarp tikrosios integralo reikšmės ir apskaičiuotos reikšmės neviršijo nurodytos galimos skaičiavimo paklaidos ε (0.000001).
2. Klasikinis lokalių paklaidos apskaičiavimo būdas yra paremtas tuo, kad asimptotiškai $|I - Q| < |N[f]|$, kai f yra pakankamai glodi, o h pakankamai mažas. K.Pluko pasiūlytu metodu buvo apskaičiuoti integralo ir paklaidos įverčiai daliniuose intervaluose. [5](#) lentelėje pateikti rezultatai parodo, kokios yra realios paklaidos ir paklaidų įverčiai daliniuose intervaluose. 1/8 visų atvejų buvo gautas per daug optimistinis paklaidos įvertis, tačiau tokių įverčių atsiradimo dėsningumas nepastebėtas.
3. Tariant `rc8` ir `rc8norm` procedūras gauti rezultatai parodė, kad `rc8` ir `rc8norm` procedūros pateikia rezultatus visais atvejais, netgi kai to nepadaro geriausiai rezultatus parodžiusi `coteda` procedūra. Tačiau daugeliu atvejų `rc8` ir `rc8norm` procedūrų pateikti rezultatai yra didesni negu `coteda` procedūros. `rc8` procedūros rezultatai beveik visada buvo prastesni negu `rc8norm`.

6. LITERATŪROS SĄRAŠAS

1. ESPELID, T.O. A test of QUADPACK and Four Doubly Adaptive Quadrature Routines. Department of Informatics, University of Bergen, Bergen, Norway, 2004. Report #281 ISSN 0333-3590.
2. ESPELID, T.O. Doubly Adaptive Quadrature Routines based on Newton-Cote Rules. Department of Informatics, University of Bergen, Bergen, Norway, 2002. Report #229. ISSN 0333-3590.
3. ESPELID, T.O. Extended Doubly Adaptive Quadrature Routines. Department of Informatics, University of Bergen, Bergen, Norway, 2004. Report #266. ISSN 0333-3590.
4. GANDER, W.; GAUTSCHI, W. Adaptive Quadrature – Revisited. Institut für Wissenschaftliches Rechnen ETH Zurich, 1998. Research Report #306.
5. PLUKAS, K. Skaitiniai metodai ir algoritmai. Kaunas: Naujasis lankas, 2001, 421-493p. ISBN 9955-03-061-5.
6. PLUKAS, K.; PLUKIENĖ, D. Niutono ir Koteso kvadratūrinių formulių paklaidos įvertinimas. Lietuvos matematikos rinkinys / Matematikos ir informatikos institutas, Lietuvos matematikų draugija, Vilniaus universitetas, 2004, T. 44, spec. nr., 660–664p. ISSN 0132-2818.
7. UEBERHUBER, C.W. Numerical Computation 2. Methods, Software, and Analysis. Berlin Heidelberg New York: Springer-Verlag, 1997, 150-157p. ISBN 3-540-62057-5.

7. ERROR ESTIMATES OF QUADRATURE FORMULAS AND THEIR ANALYSIS

Summary

In this paper the problems of finding error estimates of quadrature formulas are discussed. A method proposed by K.Plukas was tested. One of the most important tests was the one determining the error estimates that are too optimistic. The results have shown that there are 1/8 of such error estimates and that there is no visible pattern when they occur. The second very important test was the one that shows how many iterations are needed to get the estimate of integral. After comparing the results to the ones produced by method of T.O.Espelid it was obvious that method of K.Plukas produced results even when method of T.O.Espelid was not able to. Comparison of these results have also shown that method of K.Plukas is not always as effective as method of T.O.Coteda, i.e. in many cases method of K.Plukas produced the result after more iterations than method of T.O.Coteda.