УДК 004.942

ЗАСТОСУВАННЯ ПРОЦЕСУ ЕЙТКЕНА ПРИ РОЗРАХУНКАХ ІНТЕГРАЛІВ З ОСОБЛИВОСТЯМИ ЧИСЕЛЬНИМИ МЕТОДАМИ

Кисельова Г.О.,
Кисельов В.Б.
Черкаський державний технологічний університет

Рассматривается численный метод приближенного вычисления интегралов с особенностями с применением процесса Эйткена для конструирования адаптивных алгоритмов по формулам Ньютона-Котеса.

Ключевые слова: *интеграл, тест, метод, матлаб.*

In the article the numerical method of integral approximate calculation with peculiarities, with the use of Aitken process for adoptive algorithms construction by Newton-Cotes formulas, is considered.

Key words: *integral, test, method, maths lab.*

Вступ. Істотну частину підінтегральних функцій, що реально зустрічаються, складають функції з особливостями, причому особливість може міститися або у функції, або в її похідній. Для обчислення інтегралів з особливостями застосовують ряд способів, які умовно можуть бути поділені на чисельно-аналітичні та чисельні. При розрахунках чисельно-аналітичними способами основна задача полягає в усунені особливості, що досягається заміною змінної інтегрування, або виділенням особливих точок за допомогою представлення підінтегральної функції у вигляді комбінації двох функцій, одна з яких має всі особливості і може бути обчислена аналітично, а друга не має особливих точок і обчислюється чисельними методами [1, 2, 3, 5, 6]. Основою чисельних розрахунків інтегралів з особливостями є оптимальний розподіл вузлів інтегрування, що досягається будовою спеціальних алгоритмів, які враховують особливості данної функції, або використанням стандартних програм з автоматичним вибором кроку [1, 2, 8, 9].

Недоліком чисельно-аналітичних та чисельних способів з використанням спеціальних алгоритмів, які б враховували особливості данної функції є необхідність попереднього дослідження підінтегральної функції, що вимагає досить високої кваліфікації дослідника в області математики. Найбільш простим і універсальним способом обчислення значної кількості інтегралів з особливостями, який не потребує попередньої підготовки є використання стандартних програм з автоматичним вибором кроку, але похибка при використанні таких програм може бути досить великою.

Мета. Довести ефективність застосування процесу Ейткена в програмах з автоматичним вибором кроку на основі формул Ньютона-Котеса при розрахунках інтегралів з особливостями чисельними методами.

Постановка проблеми. В більшості програм з автоматичним вибором кроку необхідно визначати величину похибки обчислень. В алгоритмах на основі формул Ньютона-Котеса для практичного обчислення похибки застосовується формула Рунге [1, 2, 3, 5, 8], яка заснована на виділенні головного члена похибки за результатами розрахунків інтегралу з двома різними кроками

$$O_h = \frac{I_h - I_k}{\left(\frac{k}{h}\right)^p - 1},$$

де I_h – наближене значення інтеграла, обчислене з кроком h;

O_h – похибка для інтеграла з кроком h;

I_k – наближене значення інтеграла, обчислене з кроком k;

p – порядок точності методу. Величина p у формулі Рунге визначається теоретично, так для методу трапецій $p = 2$, а для методу Сімпсона $p = 4$, але якщо підінтегральна функція має особливість то реальний порядок точності може значно відрізнятися від теоретичного. Визначити реальний порядок точності можливо застосувавши процес Ейткена [2], який заснований на визначенні уточненого значення інтеграла за результатами розрахунків з трьома різними кроками.

Шляхи вирішення проблеми. Застосовуючи процес Ейткена для інтегралів з постійним відношенням кроків $h_1 = 2 \cdot h_2 = 4 \cdot h_3$, обмежившись головним членом похибки, можна записати

$$I = I_k + C \cdot h_k^p, \quad k = 1,\ 2,\ 3.$$

Це система трьох рівнянь для визначення невідомих I, C, p:

$$\begin{cases} I = I_1 + C \cdot h_1^p \\ I = I_2 + C \cdot h_2^p \\ I = I_3 + C \cdot h_3^p \end{cases} \Rightarrow \begin{cases} I = I_1 + C \cdot \left(4 \cdot h_3\right)^p \\ I = I_2 + C \cdot \left(2 \cdot h_3\right)^p \\ I = I_3 + C \cdot h_3^p \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} I = I_1 + \left(2^p\right)^2 C \cdot h_3^p \\ I = I_2 + 2^p \cdot C \cdot h_3^p \\ I = I_3 + C \cdot h_3^p \end{cases};$$

провівши відповідні спрощення отримаємо:

$$\Rightarrow \begin{cases} I = \dfrac{I_2^2 - I_1 \cdot I_3}{2 \cdot I_2 - I_1 - I_3} \\ p = \dfrac{\ln\left(\dfrac{I_1 - I_2}{I_2 - I_3}\right)}{\ln 2} \\ C = \dfrac{\left(I_3 - I_2\right)^2}{\left(2 \cdot I_2 - I_1 - I_3\right) \cdot h_3^{\ln\left(\frac{I_1 - I_2}{I_2 - I_3}\right) / \ln 2}} \end{cases};$$

звідки можемо отримати величину похибки O_3 для I_3

$$O_3 = C \cdot h_3^p = \frac{\left(I_3 - I_2\right)^2}{2 \cdot I_2 - I_1 - I_3}$$

та уточнене значення інтеграла

$$I = I_3 + O_3 = I_3 + \frac{\left(I_3 - I_2\right)^2}{2 \cdot I_2 - I_1 - I_3}.$$

Отримані результати. Використовуючи отримані формули розроблено програми quadeyt та quadrom, в основу яких покладений алгоритм adaptsim [8] з застосуванням формули Сімпсона (аналогічний алгоритм використаний при будові програм quad та quadl в МАТЛАБ 7).

Відмінність програм quadeyt та quadrom полягає в тому, що в quadeyt для визначення похибки та уточненого значення інтеграла застосована формула Ейткена, а в quadrom — формула Ромберга, яка є похідною від формули Рунге при $p = 4$. Обидві програми розроблені для порівняння ефективності використання двох різних формул, тому мають максимально спрощений алгоритм достатній для виконання поставленої задачі, але дані розробки можуть бути покладені в основу більш складних та універсальних алгоритмів. Відповідні m-файли, розроблені для системи МАТЛАБ 7 приведені нижче.

Програма quadeyt.m.

```
function [I] = ...
    quadeyt(F,a,b,tol,varargin)
f=fcnchk(F);
h=(b-a)/8;
x=[a,a+h,a+2*h,a+3*h,...
    (a+b)/2,b-3*h,b-2*h,b-h,b];
y=f(x,varargin{:});
if ~isfinite(y(1))
   y(1)=f(a+eps,varargin{:});
end
if ~isfinite(y(9))
   y(9)=f(b-eps,varargin{:});
end
I1=h*4/3*(y(1)+4*y(5)+y(9));
I2=h*2/3*(y(1)+4*y(3)+2*y(5)+...
    4*y(7)+y(9));
I3=h/3*(y(1)+4*y(2)+2*y(3)+...
    4*y(4)+2*y(5)+4*y(6)+...
    2*y(7)+4*y(8)+y(9));
O3=(I3-I2)^2/(2*I2-I3-I1);
if abs(O3)<=tol
    I=I3+O3;
    return
end
[I(1)]=adapteyt(f,x(1),...
    x(2),x(3),x(4),x(5),...
    y(1),y(2),y(3),y(4),...
    y(5),tol,varargin{:});
[I(2)]=adapteyt(f,x(5),...
    x(6),x(7),x(8),x(9),...
    y(5),y(6),y(7),y(8),...
    y(9),tol,varargin{:});
I=sum(I);
% --------------------------
function [I] = ...
    adapteyt(f,a,d,c,e,b,...
    fa,fd,fc,fe,fb,tol,varargin)
h=(b-a)/8;
x=[(a+h),(d+h),(c+h),(b-h)];
y=f(x,varargin{:});
I1=h*4/3*(fa+4*fc+fb);
I2=h*2/3*(fa+4*fd+2*fc+4*fe+fb);
I3=h/3*(fa+4*y(1)+2*fd+4*y(2)+...
    2*fc+4*y(3)+2*fe+4*y(4)+fb);
O3=(I3-I2)^2/(2*I2-I3-I1);
if abs(O3)<=tol,
   I=I3+O3;
   return
else
   [Iac]=adapteyt(f,a,x(1),d,...
       x(2),c,fa,y(1),fd,y(2),...
       fc,tol,varargin{:});
   [Icb]=adapteyt(f,c,x(3),e,...
       x(4),b,fc,y(3),fe,y(4),...
       fb,tol,varargin{:});
   I=Iac+Icb;
end
```

Програма quadrom.m.

```
function [I] = ...
    quadrom(F,a,b,tol,varargin)
f=fcnchk(F);
h=(b-a)/8;
x=[a,a+h,a+2*h,a+3*h,...
    (a+b)/2,b-3*h,b-2*h,b-h,b];
y=f(x,varargin{:});
if ~isfinite(y(1))
```

```
   y(1)=f(a+eps,varargin{:});
end
if ~isfinite(y(9))
   y(9)=f(b-eps,varargin{:});
end
I2=h*2/3*(y(1)+4*y(3)+2*y(5)+...
    4*y(7)+y(9));
I3=h/3*(y(1)+4*y(2)+2*y(3)+...
    4*y(4)+2*y(5)+4*y(6)+...
    2*y(7)+4*y(8)+y(9));
O3=(I3-I2)/15;
if abs(O3)<=tol
    I=I3+O3;
    return
end
[I(1)]=adaptrom(f,x(1),x(2),...
   x(3),x(4),x(5),y(1),y(2),y(3),...
    y(4),y(5),tol,varargin{:});
[I(2)]=adaptrom(f,x(5),x(6),...
   x(7),x(8),x(9),y(5),y(6),y(7),...
    y(8),y(9),tol,varargin{:});
I=sum(I);
% --------------------------------
function [I] = ...
adaptrom(f,a,d,c,e,b,fa,fd,fc,...
fe,fb,tol,varargin)
h=(b-a)/8;
x=[(a+h),(d+h),(c+h),(b-h)];
y=f(x,varargin{:});
I2=h*2/3*(fa+4*fd+2*fc+4*fe+fb);
I3=h/3*(fa+4*y(1)+2*fd+4*y(2)+...
    2*fc+4*y(3)+2*fe+4*y(4)+fb);
O3=(I3-I2)/15;
if abs(O3)<=tol,
   I=I3+O3;
   return
else
    [Iac] = ...
adaptrom(f,a,x(1),d,x(2),c,fa,...
y(1),fd,y(2),fc,tol,varargin{:});
   [Icb] = ...
adaptrom(f,c,x(3),e,x(4),b,fc,...
y(3),fe,y(4),fb,tol,varargin{:});
   I=Iac+Icb;
end
```

Звернення до програм quadeyt та quadrom аналогічне використовуваному в МАТЛАБ 7 зверненню до програм quad [4]. Основною відмінністю є необхідність задавати величину похибки обчислень та відсутність можливості виводу додаткової інформації про хід розрахунків.

$$I = \mathrm{quadeyt}(f, a, b, tol),$$

$$I = \mathrm{quadrom}(f, a, b, tol),$$

де I – значення інтеграла;

f – підінтегральна функція;

a, b, – межі інтегрування;

tol – величина припустимої похибки обчислень.

Тест програм. Для порівняння ефективності використання функцій quadeyt та quadrom розраховуємо за їх допомогою кілька прикладів [6]. При розрахунках даних прикладів програмою quad можна відмітити [6], що отримані результати не відповідають заданій точності. Наводимо ці приклади.

Приклад 1: $\int_0^{\frac{p}{2}} \ln(\sin x)\,dx = -\frac{p}{2}\ln 2.$

Приклад 2: $\int_0^1 \sqrt{1-x^2}\,dx = \frac{p}{4}.$

Приклад 3: $\int_0^1 \frac{1}{(1+x)\sqrt{x}}dx = \frac{p}{2}.$

Приклад 4: $\int_0^1 \frac{\ln x}{1-x^2}dx = -\frac{p^2}{8}.$

Приклад 5: $\int_0^1 \sqrt{x} = \frac{2}{3}.$

Особливості даних інтегралів полягають в тому, що на проміжку інтегрування функція або її похідна обертається в нескінченність, тому порядок методу Сімпсона $ð \neq 4$.

Основною метою даного тесту є порівняння якості результатів, отриманих при розрахунках програмами quadeyt та quadrom, тому головним параметром, за яким визначається ефективність алгоритму є відповідність отриманого результату заданій точності.

Абсолютна похибка розрахунків Δ_{abs} визначається як модуль різниці між точним значенням інтеграла $I_т$ та наближеним $I_н$ розрахованим із заданою точністю tol

$$\Delta_{abs} = |I_ð - I_i|.$$

Результат розрахунку вважається задовільним, якщо абсолютна похибка менша за задану точність $\Delta_{abs} < tol.$

Для порівняння ефективності алгоритмів при різній заданій точності, необхідно визначити відповідність отриманого результату заданій точності, тобто коефіцієнт відповідності $k_в$, який визначає відсоток абсолютної похибки Δ_{abs} від заданої точності tol

$$k_â = \frac{tol - \Delta_{abs}}{tol} \cdot 100\%.$$

Результат вважається задовільним, якщо коефіцієнт відповідності $k_â \geq 0$.

Отримані результати розрахунків двома програми quadeyt та quadrom прикладів 1–5 зведені до відповідних таблиць 1–5. Перший стовпчик показує задану похибку *tol*, другий – абсолютну похибку розрахунків Δ_{abs} для програми quadeyt, третій – коефіцієнт відповідності $k_в$ для програми quadeyt, четвертий – абсолютну похибку розрахунків Δ_{abs} для програми quadrom, п'ятий – коефіцієнт відповідності $k_в$ для програми quadrom.

Таблиця 1

Похибки розрахунків для прикладу 1

r=abs(-pi/2*log(2)-quadeyt('log(sin(x))',0,pi/2))				
	quadeyt		quadrom	
tol	Δ_{abs}	$k_в$	Δ_{abs}	$k_в$
1e-1	1.4394 e-004	99.8561	0.9497	-849.7
1e-2	1.9617 e-005	99.8038	0.1108	-1008
1e-3	3.1271 e-006	99.6873	0.0129	-1190
1e-4	9.2923 e-007	99.0708	7.2091 e-004	-620.91
1e-5	6.1223 e-007	93.8777	8.2573 e-005	-725.73
1e-6	1.5825 e-007	84.175	9.4439 e-006	-844.39
1e-7	2.5451 e-008	74.549	1.0559 e-006	-955.9
1e-8	3.5370 e-009	64.630	1.1671 e-007	-1067.1
1e-9	3.3724 e-010	66.276	1.2707 e-008	-1170.7
1e-10	4.2693 e-011	57.307	6.4052 e-010	-540.52
1e-11	4.4824 e-012	55.176	6.4970 e-011	-549.7
1e-12	4.0101 e-013	59.899	6.2712 e-012	-527.12
1e-13	4.9516 e-014	50.484	5.4423 e-013	-444.23
1e-14	6.2172 e-015	37.828	9.5257 e-014	-852.57
1e-15	1.3323 e-015	-33.230	4.4409 e-015	-344.09
1e-16	6.6613 e-016	-566.130	2.2204 e-016	-122.04

Для порівняння програм, величина коефіцієнта відповідності $k_в$ зображена на рисунках 1–5 у вигляді стовпчикових діаграм, де по вісі абсцис відкладена величина заданої точності *tol*, по вісі ординат – величина коефіцієнта відповідності $k_в$. Стовпчики, які на діаграмах рис. 1–5 білого кольору, визначають коефіцієнт відповідності для програми quadeyt, а стовпчики на діаграмах темного кольору, визначають коефіцієнт відповідності для програми quadrom. Відповідність результату розрахунків заданій точності можна визначити по величині стовпця – чим менший стовпчик, тим краща відповідність. Якщо стовпчик знаходиться вище вісі абсцис – це означає, що отримана точність вища за задану і результат можна вважати задовільним, якщо стовпчик знаходиться нижче вісі абсцис – це означає, що отримана точність менша заданої і результат вважається незадовільним.

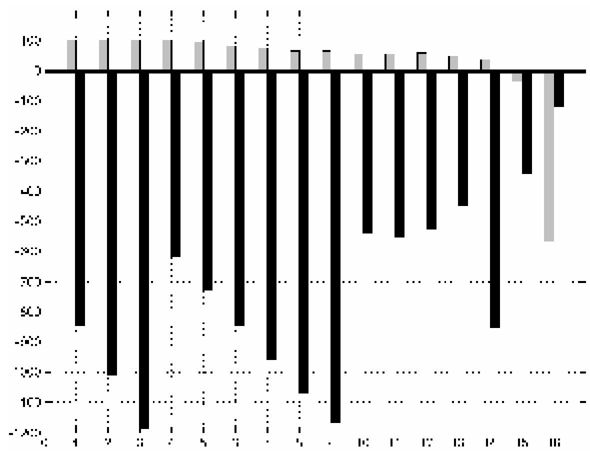

Рис. 1. Зміна коефіцієнта відповідності $k_в$ програм quadeyt та quadrom для прикладу 1

Таблиця 2

Похибки розрахунків для прикладу 2

r=abs(pi/4-quadeyt('sqrt(1-x.^2)',0,1))				
	quadeyt		quadrom	
tol	Δ_{abs}	$k_в$	Δ_{abs}	$k_в$
1	2	3	4	5
1e-1	4.8528 e-005	99.9515	0.0045	95.5
1e-2	4.8528 e-005	99.5147	0.0045	55
1e-3	1.8923 e-006	99.8108	0.0045	-350
1e-4	4.4001 e-007	99.56	5.5754 e-004	-457.54
1e-5	9.8101 e-008	99.019	6.9653 e-005	-596.53
1e-6	4.5676 e-008	95.4324	3.0943 e-006	-209.43
1e-7	1.5509 e-008	84.491	3.9071 e-007	-290.71
1e-8	1.0705 e-009	89.295	4.8928 e-008	-389.28

Продовження табл. 2

1	2	3	4	5
1e-9	1.4405 e-010	85.595	6.0613 e-009	-506.13
1e-10	1.8857 e-011	81.143	2.7249 e-010	-172.49
1e-11	2.5135 e-012	74.865	3.4110 e-011	-241.1
1e-12	3.0154 e-013	69.846	4.2597 e-012	-325.97
1e-13	1.8652 e-014	81.348	5.3024 e-013	-430.24
1e-14	1.8874 e-015	81.126	6.5614 e-014	-556.14
1e-15	2.2204 e-016	77.796	2.9976 e-015	-199.76
1e-16	0	100.0	4.4409 e-016	-344.09

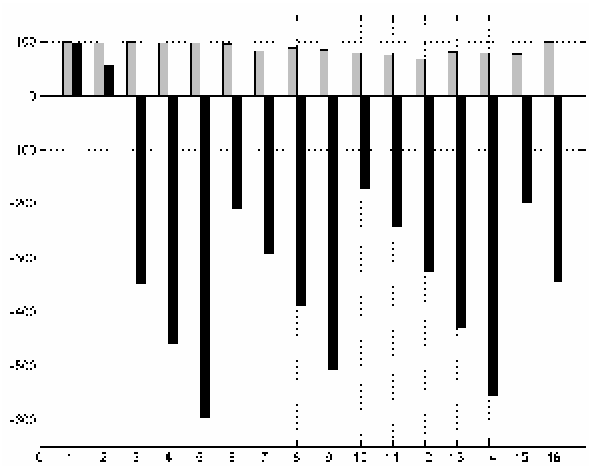

Рис. 2. Зміна коефіцієнта відповідності $k_в$ програм quadeyt та quadrom для прикладу 2

Таблиця 3

Похибки розрахунків для прикладу 3

r=abs(pi/2-quadeyt('1./(1+x)./sqrt(x)',0,1))				
	quadeyt		quadrom	
tol	Δ_{abs}	$k_в$	Δ_{abs}	$k_в$
1	2	3	4	5
1e-1	2.7518 e-005	99.972	1.2441	-1144.1
1e-2	7.7025 e-006	99.923	0.0777	-677
1e-3	3.4326 e-006	99.657	0.0097	-870
1e-4	1.9230 e-006	98.077	0.0012	-1100
1e-5	1.3893 e-006	86.107	7.4076 e-005	-640.76
1e-6	6.2288 e-007	37.712	8.8955 e-006	-789.55

Продовження табл. 3

1	2	3	4	5
1e-7	1.0463 e-007	-4.63	9.3050 e-007	-830.5
1e-8	2.6471 e-008	-164.71	4.8725 e-008	-387.25
1e-9	1.7724 e-008	-1672.4	1.7557 e-008	-1655.7
1e-10	9.9572 e-011	0.428	1.6476 e-008	-16376
1e-11	9.3390 e-012	6.610	2.7973 e-010	-2697.3
1e-12	1.1080 e-012	-10.8	2.4685 e-011	-2368.5
1e-13	1.0569 e-013	-5.69	3.1020 e-012	-3002
1e-14	9.7700 e-015	2.3	2.7400 e-013	-2640
1e-15	1.3323 e-015	-33.23	3.4195 e-014	-3319.5
1e-16	0	100	3.1086 e-015	-3008.6

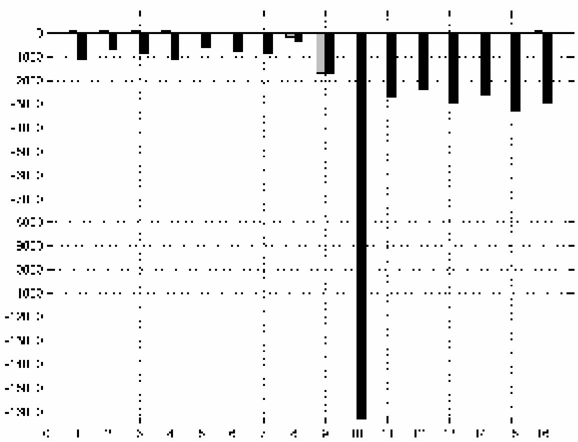

Рис. 3. Зміна коефіцієнта відповідності $k_в$ програм quadeyt та quadrom для прикладу 3

Таблиця 4

Похибки розрахунків для прикладу 4

r=abs(-pi^2/8-quadeyt('log(x)./(1-x.^2)',0,1))				
	quadeyt		quadrom	
tol	Δ_{abs}	$k_в$	Δ_{abs}	$k_в$
1	2	3	4	5
1e-1	1.8116 e-004	99.8188	1.2186	-1118.6
1e-2	2.4432 e-005	99.7557	0.0694	-594
1e-3	2.0209 e-006	99.7979	0.0080	-700
1e-4	6.0104 e-007	99.399	9.2691 e-004	-826.91

Продовження табл. 4

1	2	3	4	5
1e-5	4.1045 e-007	95.8955	1.0613 e-004	-961.3
1e-6	1.9440 e-007	80.56	1.2091 e-005	-1109.1
1e-7	3.1383 e-008	68.617	6.6013 e-007	-560.13
1e-8	2.9513 e-009	70.487	7.3020 e-008	-630.2
1e-9	4.0714 e-010	59.286	7.8612 e-009	-686.12
1e-10	4.2854 e-011	57.146	8.2914 e-010	-729.14
1e-11	3.8867 e-012	61.133	8.4772 e-011	-747.72
1e-12	4.7873 e-013	52.127	8.1668 e-012	-716.68
1e-13	5.0626 e-014	49.374	7.2298 e-013	-622.98
1e-14	5.7732 e-015	42.268	5.3513 e-014	-435.13
1e-15	8.8818 e-016	11.182	6.8834 e-015	-588.34
1e-16	4.4409 e-016	-344.09	2.2204 e-016	-122.04

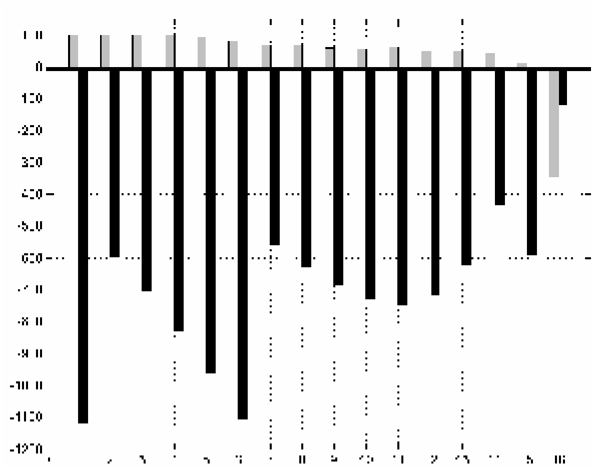

Рис. 4. Зміна коефіцієнта відповідності $k_в$ програм quadeyt та quadrom для прикладу 4

Таблиця 5

Похибки розрахунків для прикладу 5

r=abs(2/3-quadet('sqrt(x)',0,1))				
	quadeyt		quadrom	
tol	Δ_{abs}	$k_в$	Δ_{abs}	$k_в$
1	2	3	4	5
1e-1	2.0560 e-005	99.9794	0.0032	96.8
1e-2	2.0560 e-005	99.7944	0.0032	68.0
1e-3	2.5951 e-006	99.7405	0.0032	-220
1e-4	3.4945 e-007	99.6506	3.9383 e-004	-293.83
1e-5	6.8746 e-008	99.3125	4.9238 e-005	-392.38
1e-6	3.3657 e-008	96.6343	6.1649 e-006	-516.49
1e-7	1.0737 e-008	89.263	2.7618 e-007	-176.18
1e-8	1.8667 e-009	81.333	3.4702 e-008	-247.02
1e-9	2.6007 e-010	73.993	4.3465 e-009	-334.65
1e-10	1.6587 e-011	83.413	5.4208 e-010	-442.08
1e-11	1.8275 e-012	81.725	6.7043 e-011	-570.43
1e-12	2.2904 e-013	77.096	3.0170 e-012	-201.7
1e-13	2.7867 e-014	72.133	3.7692 e-013	-276.92
1e-14	2.9976 e-015	70.024	4.7073 e-014	-370.73
1e-15	1.1102 e-016	88.8980	5.8842 e-015	-488.42
1e-16	1.1102 e-016	-11.02	7.7716 e-016	-677.16

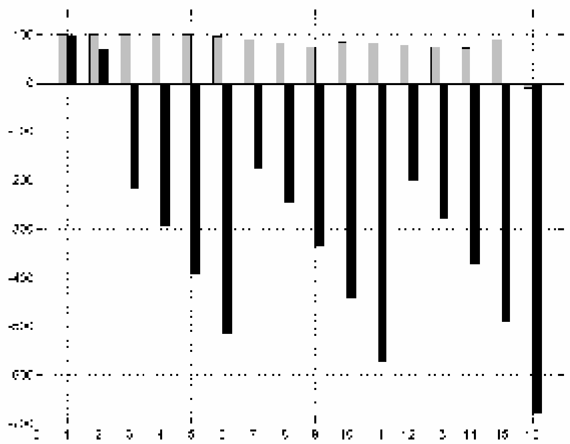

Рис. 5. Зміна коефіцієнта відповідності $k_в$ програм quadeyt та quadrom для прикладу 5

Висновки. Результати порівняння показують, що середня величина коефіцієнта відповідності програми quadeyt для прикладу 1 — 27.7037, для прикладу 2 — 88.6746, для прикладу 3 — –78.7921, для прикладу 4 — 43.9223, для прикладу 5 — 81.3731, а програми quadrom для прикладу 1 — –738.2938, для прикладу 2 — –307.4319, для прикладу 3 —

-2594.2, для прикладу 4 — -696.7744, для прикладу 5 — -315.1994. Таким чином загальна середня величина коефіцієнта відповідності для всіх прикладів при використанні програми quadeyt складає 32.5763, а при використанні програми quadrom — -930.3799. Найбільш важливим в даному порівнянні є не величина коефіцієнта відповідності, а його знак, тобто додатна величина для програми quadeyt вказує на повну відповідність отриманих результатів, а від'ємна для програми quadrom — невідповідність переважної більшості розрахунків заданій точності. Зважаючи на те, що величина машинної точності самої програми МАТЛАБ порядка 10^{-16}, при порівнянні програм результати останньої строки таблиць (для $tol = 10^{-16}$) можна не враховувати, однак це практично не впливає на кінцевий результат.

Порівнюючи результати тестів можна зробити висновок про високу стабільність та якість розрахунків, проведених за допомогою програми quadeyt, в якій для визначення похибки та уточненого значення інтеграла застосована формула Ейткена. Аналіз данних, отриманих при розрахунках програмою quadrom, в якій застосована формула Ромберга, показує повну невідповідність отриманих результатів заданій точності. Таким чином, формула Ромберга може бути застосована при розрахунках інтегралів з особливостями лише за умови контролю точності отриманих результатів, або після попереднього усунення особливості [6]. Застосування процесу Ейткена дає можливість будувати алгоритми на основі формул Ньютона-Котеса для розрахунків інтегралів з особливостями чисельними методами з заданою точністю, однак при складанні програм необхідно враховувати, що знаменник формули при певних умовах може дорівнювати нулю.

ЛІТЕРАТУРА

1. Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. – М.: Наука, 2003.
2. Калиткин Н.Н. Численные методы. – М.: Наука, 1978.
3. Гусак А.А., Гусак Г.М., Бричикова Е.А.. Справочник по высшей математике. – М.: ТетраСистемс, 1999. – 640 с.
4. Клетков Ю.Л., Клетков А.Ю., Шульц М.М., MATLAB 7: программирование, численные методы. – С.Пб.: БХВ-Петербург, 2005. – 752 с.
5. Манжиров А.В., Полянин А.Д. Методы решения интегральных уравнений: Справочник. М.: Факториал, 1999. – 272 с.
6. Киселева А.А., Киселев В.Б., Корнеев А.М., Протасов С.Ю. Численно-аналитический способ вычисления интегралов с особенностями. Вісник ЧДТУ, №3, 2008. – С. 64–69.
7. Shampine L.F., Allen R.C., Jr. and S. Pruess, Fundamentals of Numerical Computing, Wiley, New York, 1997.
8. W. Gander and W. Gautschi Adaptive Quadrature – Revisited, BIT Vol. 40, №. 1, March 2000. P. 84–101. TR-Report 306, Departement Informatik, 1998.
9. Shampine L.F., "Vectorized Adaptive Quadrature in Matlab", Journal of Computational and Applied Mathematics 211, 2008. – P. 131–140.

Кисельова Г.О., старший викладач кафедри електротехнічних систем Черкаського державного технологічного університету.

Кисельов В.Б., асистент кафедри електротехнічних систем Черкаського державного технологічного університету.