

ИНТЕРПОЛАЦИЯ

Нека стойностите на функцията $f(x)$, дефинирана в интервала $[a, b]$, са известни само за краен брой точки x_0, x_1, \dots, x_n в $[a, b]$. Стойностите

$y_i = f(x_i)$ могат например да са резултат от някакви измервания, като липсва каквато и да е допълнителна информация за функционалната зависимост f между двете множества x_0, x_1, \dots, x_n и y_0, y_1, \dots, y_n .

Например, да предположим, че обемът V на затворен в цилиндър газ се мени с налягането P по следния начин

$P \text{ cm}^2$	60	80	100	120	140	160
$V \text{ cm}^3$	80.0	09.2	60.0	52.0	45.0	18.6

Колко ще бъде V при $P = 93$?

На този въпрос можем да отговорим, ако намерим нова функция $F(x)$, която:

(а) е близка в някакъв смисъл до $f(x)$

(б) има някои "хубави свойства" - например лесно се изчислява, гладка е или е интегрируема.

Един от разумните критерии за "близост" е съвпадение на стойностите на $f(x)$ и $F(x)$ за дадено множество от точки, т.е.

$$f(x_i) = F(x_i), \quad i = 0, 1, \dots, n.$$

В този случай говорим за **интерполация**. Разбира се, много е важно и как ще изпълним второто изискване за $F(x)$. В следните параграфи ще разгледаме случаите, когато $F(x)$ е полином, кубичен сплайн или рационална функция.

В глава 6 ще разгледаме друг критерий за "близост".

4.1 ПОЛИНОМИ НА ЛАГРАНЖ

Първо, да разгледаме следната проста задача. В равнината са дадени две точки

$$(x_0, y_0), (x_1, y_1)$$

Искаме да намерим полином $P(x)$ такъв, че

$$P(x_0) = y_0, P(x_1) = y_1$$

Очевидно правата, минаваща през дадените точки:

$$y - y_0 = \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) \Leftrightarrow y = \frac{(x - x_1)}{(x_0 - x_1)}y_0 + \frac{(x - x_0)}{(x_1 - x_0)}y_1,$$

е решение на задачата. Така че решение съществува и то е единствено, ако поставим ограничението степента на $P(x)$ да е най-много 1. Последната забележка ни подсказва да формулираме общата интерполационна задача, както следва.

Нека имаме данните

x	x_0	x_1	\dots	x_n
y	y_0	y_1	\dots	y_n

Искаме да намерим полином $P_n(x)$ от степен най-много n такъв, че

$$P_n(x_0) = y_0, P_n(x_1) = y_1, P_n(x_n) = y_n,$$

Решението на задачата е *единствено*, но може да бъде записано в различни форми. Най-лесната за запомняне се нарича **полином на Лагранж** и имитира решението в случая $n = 1$, разгледан по-горе.

Интерполационен полином на Лагранж

$$(4.1) \quad L(x) = \sum_{k=0}^n y_k L_{nk}(x),$$

където:

$$\begin{aligned} L_{nk}(x) &= \frac{(x - x_0)(x - x_1) \dots (x - x_{k-1})(x - x_{k+1}) \dots (x - x_n)}{(x_k - x_0)(x_k - x_1) \dots (x_k - x_{k-1})(x_k - x_{k+1}) \dots (x_k - x_n)} \\ &= \prod_{i=0, i \neq k}^n \frac{(x - x_i)}{(x_k - x_i)} \end{aligned}$$

Очевидно $L_{n0}(x), L_{n1}(x), \dots, L_{nn}(x)$ са полиноми от степен n и имат свойството

$$L_{nk}(x_i) = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases}$$

което е достатъчно, за да се разбере, че $L(x)$ решава задачата при $n + 1$ точки.

Да се върнем към задачата за интерполация на функция $f(x)$ такава, че

$$y_i = f(x_i), \quad i = 0, 1, \dots, n.$$

При допълнителни предположения можем да оценим грешката на апроксимиране на $f(x)$ с $L(x)$.

Теорема 5. /5/ Нека $x_0, x_1, \dots, x_n \in [a, b]$ са различни числа и $f \in C^{n+1}[a, b]$. Тогава за всяко $x \in [a, b]$ съществува число $\xi(x) \in (a, b)$ такава, че

$$(4.2) \quad f(x) = L(x) + \frac{f^{(n+1)}(\xi(x))}{(n+1)!} (x - x_0)(x - x_1) \dots (x - x_n),$$

където $L(x)$ е интерполационният полином на Лагранж (4.1).

Горната формула повече или по-малко е от теоретичен интерес, защото в практиката обикновено нямаме информация за производните на $f(x)$. Но ние все пак можем да я използваме.

Пример 1 Нека са дадени данните

x	2	2.5	4
$f(x)$	0.5	0.4	0.25

Да определим приближено $f(3)$.

1) Коефициентите-полиноми са

$$\begin{aligned} L_{20} &= \frac{(x - 2.5)(x - 4)}{(2 - 2.5)(2 - 4)} = (x - 6.5)x + 10 \\ L_{21} &= \frac{(x - 2)(x - 4)}{(2.5 - 2)(2.5 - 4)} = -\frac{1}{3}((4x - 24)x + 32) \\ L_{22} &= \frac{(x - 2)(x - 2.5)}{(4 - 2)(4 - 2.5)} = \frac{1}{3}((x - 4.5)x + 5) \end{aligned}$$

2) Полиномът на Лагранж е

$$L_2(x) = 0,5L_{20}(x) + 0,4L_{21}(x) + 0,25L_{22}(x) = (0,05x - 0,425)x + 1,15$$

3) Числото $f(3)$ е

$$f(3) \approx L_2(3) = 0.325$$

4) Грешката от приближение е

$$|f(3) - L_2(3)| \leq \frac{1}{6} |(3 - 2,5)(3 - 4)| \max_{2 \leq x \leq 4} |f'''(x)|$$

Ако предположим, че $f(x) = 1/x$ (което се съгласува с данните) то

$$|f'''(x)| = \left| \frac{6}{x^4} \right| \leq \frac{6}{2^4} = \frac{3}{8}, 2 \leq x \leq 4$$

$$\text{и } |f(3) - L_2(3)| \leq \frac{1}{12} \cdot \frac{3}{8} = 0,03125.$$

Както може да се види, полиномът на Лагранж се основава на ясна концепция — коефициентите-полиноми $L_{nk}(x)$ са лесни за определяне и зависят само от стойностите във възлите. Обаче изчисляването на $L_{nk}(x)$ е "скъпо" (изисква много изчисления) и преходът от $L_n(x)$, към $L_{n+1}(x)$ (т.е. при прибавяне на нова точка) е трудно. В следващия параграф ще покажем как този недостатък може да се отстрани.

4.2 РАЗДЕЛЕНИ РАЗЛИКИ

Нека $L_n(x)$ е полиномът на Лагранж, построен по данните

$$(x_0, f(x_0)), (x_1, f(x_1)), \dots (x_n, f(x_n)).$$

Възможно е да го представим в следната форма

(4.3)

$$L_n(x) = c_0 + c_1(x - x_0) + \dots + c_n(x - x_0)(x - x_1) \dots (x - x_{n-1}).$$

Опитвайки се да определим c_i , полагаме по-горе $x = x_0$ и получаваме $c_0 = f(x_0)$. Аналогично за $x = x_1$ намираме (4.3) $L_n(x) = f(x_0) + c_1(x - x_0)$ и като имаме предвид, че $L_n(x_1) = f(x_1)$, получаваме

$$c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Продължавайки по същия начин, можем да изразим всички c_i , индуктивно с помощта на следното полезно означение.

Определение 1. Нулевата **разделена разлика** по отношение на x_i , е

$$f[x_i] = f(x_i)$$

Първата разделена разлика по отношение на x_i , и x_{i+1} е

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}$$

а k -та разделена разлика по отношение на $x, x_{i+1}, \dots, x_{i+k}$ е

$$f[x_i, x_{i+1}, \dots, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \dots, x_{i+k}] - f[x_i, x_{i+1}, \dots, x_{i+k-1}]}{x_{i+k} - x_i}$$

където членовете в числителя са $(k - 1)$ -те разделени разлики.

Тогава коефициентите в (4.3) са всъщност разделени разлики и единственият интерполационен полином $L_n(x)$ може да се запише като:

(4.4)

$$L_n(x) = f[x_0] + \sum_{k=1}^n f[x_0, x_1, \dots, x_k](x - x_0) \dots (x - x_{k-1})$$

Изразът е познат като *интерполационна формула на Нютон с разделени разлики*.

Пресмятането на s , е лесно, ако се използва таблица. За $n=3$ имаме

x	$f(x)$	1p.p.	2p.p.	3
x_0	$f[x_0]$			
		$f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}$		
x_1	$f[x_1]$			
			$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$	
		$f[x_1, x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1}$		Δ
x_2	$f[x_2]$			
			$f[x_1, x_2, x_3] = \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1}$	
		$f[x_2, x_3] = \frac{f[x_3] - f[x_2]}{x_3 - x_2}$		
x_3	$f[x_3]$			

където Δ е третата разделена разлика и е изчислена като разлика на числата в предния стълб — един ред нагоре и един надолу, върху разликата на

съответните x_i , в случая $x_3 - x_0$. Същото правило се прилага, за да се изчислят числата в останалите стълбове (след втория). Коефициентите c_i са числата по горния диагонал на таблицата.

Пример 2. Ще намерим полинома от най-ниска степен, който апроксимира данните, дадени в първите два стълба на следната таблица:

x	$f(x)$	1 р.р.	2 р.р.	3 р.р.	4 р.р.
4	<u>93</u>				
		<u>83</u>			
6	259		<u>18</u>		
		155		<u>1</u>	
8	569		24		<u>0</u>
		251		1	
10	1071		30		
		371			
12	1873				

Коефициентите в (4.4) са подчертаните числа. Тъй като 4-та разлика е нула (третите разлики са равни помежду си), данните могат да се приблизят с кубична функция. Интерполационната формула на Нютон дава

$$L_3(x) = 93 + 83(x - 4) + 18(x - 4)(x - 6) + (x - 4)(x - 6)(x - 8),$$

което се редуца до $L_3(x) = x^3 + 7x + 1$.

Очевидно изчисляването на $L_n(x)$ във формата (4.4) е по-икономично, отколкото използването на оригиналната формула на Лагранж. Също така, ако добавим нов възел x_{n+1} със стойност $f(x_{n+1})$, то таблицата няма да се промени освен с прибавянето на един нов ред. Следователно $L_{n+1}(x)$ ще се получи от $L_n(x)$ като добавим в (4.4) събираемото $f[x_0, x_1, \dots, x_n](x - x_0)(x - x_{n-1})(x - x_n)$.

Идеята да се използват разделени разлики при интерполиране може да се развие по-нататък. Ние обаче няма да го правим, тъй като подходът, който ще дискутираме сега, се е доказал като по-успешен в приложенията.

4.3 КУБИЧЕН СПЛАИН

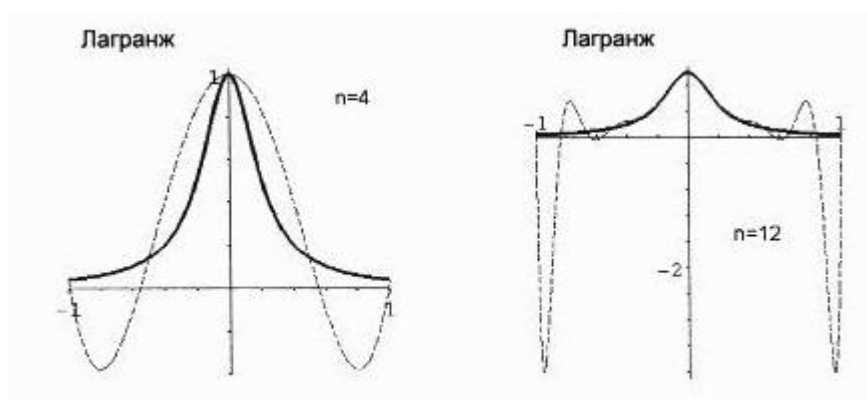
Основен недостатък на разгледаните формули е, че, когато броят на възлите става голям, полиномът на Лагранж започва да осцилира. Разбира се, това противоречи на здравия разум, който подсказва, че трябва да можем да приближим една крива по-добре, ако тя е известна в повече точки. Да разгледаме следния пример.

Пример 3. В началото на XX век немският математик Рунге интерполирал функцията

$$f(x) = \frac{1}{1 + 25x^2}$$

в интервала $[-1,1]$ като използвал разположени през равно разстояние възли x_i .

С други думи $x_i = -1 + ih$, където $h = 2/n$, $i = 0, 1, \dots, n$ и полиномът на Лагранж се строи по данните $(x_i, f(x_i))$. Резултатите за $n = 4$ и $n = 12$ са показани по-долу.



Фиг. 4.1

Както се вижда, когато степента на интерполиращия полином нарасне, отклоненията му от оригиналната крива (от която са взети данните за възлите), може да нарасне неимоверно. Едно от решенията на този проблем е да се раздели интервала за променливата x на подинтервали, където данните да се апроксимират с полиноми от ниска степен, които добре (т.е. гладко) се слепват (срещат) в общите краища на подинтервалите. Това се нарича **частично полиномиална интерполация**. Най-често срещаната интерполация от този тип използва кубични полиноми (във всеки подинтервал) и се нарича **интерполация с кубични сплайни**.

Определение 2. За дадена функция $f(x)$, $x \in [a, b]$ и множество от **възли** $a = x_0 < x_1 < \dots < x_n$, **кубичен сплайн**, интерполиращ f , е такава функция S , че:

- (а) S е кубичен полином $S_i(x)$ върху всеки $[x_j, x_{j+1}]$, $j = 0, 1, \dots, n-1$;
- (б) $S(x_j) = f(x_j)$, $j = 0, 1, \dots, n$;
- (в) $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$, $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$
 $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$, $j = 0, 1, \dots, n-2$.

Ако $S''(x_0) = S''(x_n) = 0$, то сплайнът S се нарича **натурален** и графиката му наподобява формата, която заема дълга гъвкава пръчка, огъната така, че да минава през точките $(x_i, f(x_i)), i = 0, 1, \dots, n$.

Удобно е да предположим, че

$$S_j(x) = a_j + b_j(x - x_j)^2 + d_j(x - x_j)^3, j = 0, 1, \dots, n - 1.$$

Тогава от условие (б) получаваме (полагайки $x = x_j$)

$$a_j = f(x_j), j = 0, 1, \dots, n - 1.$$

Като използваме последното условие в (в) и означението

$$c_n = \frac{S''(x_n)}{2}, h_j = x_{j+1} - x_j, \text{ лесно намиране}$$

$$(4.5) \quad d_j = \frac{c_{j+1} - c_j}{3h_j}, j = 0, 1, \dots, n - 1.$$

Сега оставащите две условия в (в) (за S_j и S''_j) ще дадат 2 отношения, в които участват променливите b_j , и c_j :

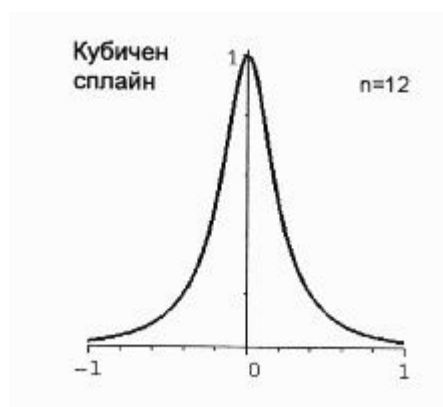
$$a_{j+1} = a_j + b_j h_j + \frac{h_j^2(2c_j + c_{j+1})}{3}$$

$$b_{j+1} = b_j + h_j(c_j + c_{j+1})$$

Решавайки първото уравнение относно b_j , и замествайки с този израз в (4.6), получаваме следната линейна система от уравнения за c_j

$$h_{j-1}c_{j-1} + 2(h_{j-1} + h_j)c_j + h_j c_{j+1} = \frac{3}{h_j}(a_{j+1} - a_j) - \frac{3}{h_{j-1}}(a_j - a_{j-1}),$$

където $j=1, 2, \dots, n$. Тази система има матрица със строго доминиращ диагонал, така че системата има единствено решение c_0, c_1, \dots, c_n . Като намерим c_j веднага можем да определим b_j , и d_j , от (4.6) и (4.5).



Фиг. 4.2

Линейната система за c_j има специалния **тридиагонал** вид. а за решаването на такива системи има бързи алгоритми, които не са описани в глава 3. Един от тях обаче е реализиран и обяснен в *работна страница 4* от приложеното CD.

Нека сега видим на фиг. 4.2 каква е графиката на натуралния кубичен сплайн, интерполиращ функцията на Рунге $f(x) = 1/(1+25x^2)$, сравнен с графиката на f , за $n = 12...$ Осцилациите изчезнаха.

4.4 АПРОКСИМАЦИЯ НА ПАДЕ

Тук ще разгледаме интерполация (или по-добре да се каже, *апроксимация*) с рационални функции. Една рационална функция $r(x)$ от степен $N = n + m$ има формата

$$r(x) = \frac{p(x)}{q(x)} = \frac{p_0 + p_1x + \dots + p_nx^n}{q_0 + q_1x + \dots + q_mx^m}.$$

Ще предположим, че $q_0 = 1$, което е еквивалентно на изискването $r(x)$ да е определена за $x = 0$.

Причината да се използва частно на два полинома вместо един полином отново е желанието да се избягнат ефекти на осцилиране. В добавка, рационалните функции приближават добре функции, имащи точки на прекъсване в или близо до разглеждания интервал, в които клонят към безкрайност. Това е модерна техника на интерполиране. често комбинирана с други подходи, вж. Глава 6.2.

В рационалната функция $r(x)$ от степен N има за определяне $N + 1$ параметъра - коефициентите p_i и q_i , $i = 0, 1, \dots, n$, $j = 1, 2, \dots, m$. При апроксимацията на **Паде** на функцията $f(x)$ за тази цел се използват условията

$$(4.7) \quad f^{(k)}(0) =, k = 0, 1, \dots, N.$$

Когато $n = N$ и $m = 0$, горните условия определят полинома на Маклорен от степен N за f .

От условие (4.7) следва, че нулата е $N + 1$ кратен корен на функцията $(f - r)(x) = f(x) - r(x)$. Ако редът на Маклорен за f е $f(x) = \sum_{i=0}^{\infty} a_i x^i$, то

$$(f - r)(x) = \frac{\sum_{i=0}^{\infty} a_i x^i - \sum_{i=0}^{\infty} q_i x^i \sum_{i=0}^{\infty} p_i x^i}{q(x)}$$

Тогава (4.7) дава, че редът в числителя по-горе няма да съдържа членове от ред N или по-малък. За удобство определяме $p_{n+1} = \dots = p_n = 0$ и

$q_{n+1} = \dots = q_n = 0$. Приравнявайки на нула коефициентите пред x^0, x^1, \dots, x^N в споменатия числител

$$(a_0 + a_1x + \dots)(1 + q_1x + \dots q_mx^m) - (p_0 + p_1x + \dots q_nx^n)$$

получаваме линейна система от $N + 1$ уравнения

$$\sum_{i=0}^k a_i q_{k-i} = p_k, k = 0, 1, \dots, N,$$

откъдето можем да определим константите ($N + 1$ на брой) в апроксимацията на Паде $r(x)$.

Пример 4. Да намерим апроксимацията на Паде за e^x от степен 5 с $n = 2$ и $m = 3$. Тогава коефициентите на $x^k, k = 0, 1, \dots, 5$ в

$$\left(1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \dots\right)(1 + q_1x + q_2x^2 + q_3x^3) - (p_0 + p_1x + p_2x^2)$$

трябва да са 0. Така получаваме системата

$$\begin{aligned} x^5: 1/120 + (1/24)q_1 + (1/6)q_2 + (1/2)q_3 &= 0 \\ x^4: 1/24 + (1/6)q_1 + (1/2)q_2 + q_3 &= 0 \\ x^3: (1/6)q_1 + q_2 + q_3 &= 0 \\ x^2: 1/2 + q_1 + q_2 &= p_2 \\ x^1: 1 + q_1 &= p_1 \\ x^0: 1 &= p_0 \end{aligned}$$

където степента на x в началото на уравнението показва от кой коефициент пред x е получено то. Системата има решение

$$p_0 = 1, p_1 = \frac{2}{5}, p_2 = \frac{1}{20}, q_1 = \frac{3}{5}, q_2 = \frac{3}{20}, q_3 = -\frac{1}{60}$$

така, че апроксимацията на Паде е

$$r(x) = \frac{1 + (2/5)x + (1/20)x^2}{1 - (3/5)x + (3/20)x^2 - (1/60)x^3}.$$

Тъй като разпределението на грешката при апроксимиране с рационални функции не е равномерно в разглеждания интервал, то се

комбинираща понякога с други апроксимации, да речем с тригонометрични полиноми. По този начин се конструира рационалната апроксимация на Чебишев, където x^k в апроксимацията на Паде е заместено с тригонометричен полином.

4.5 ЗАДАЧИ

1. Намерете стойността в $x = 0,5; 1,5; 3$ на полинома на Лагранж, построен по следните данни

а)

x	0	1	2	3
y	0	1	8	27

б)

x	0,99	1,4	3,6	8,4
y	1	3,07	5,5	8,1

2. Нека $f(x) = e^x$, $0 \leq x \leq 2$. Използвайки данните

x	0.0	0.5	1.0	2.0
$f(x)$	1.00000	1.64872	2.71828	7.38906

изчислете следното:

- а) Апроксимирайте $f(0,25)$ с линейна функция, като $x_0 = 0$ и $x_1 = 0,5$;
 - б) Апроксимирайте $f(0,75)$ с линейна функция, като $x_0 = 0,5$ и $x_1 = 1$;
 - в) Апроксимирайте $f(0,25)$ и $f(0,75)$, като използвате полином на Лагранж от втора степен, като $x_0 = 0$, $x_1 = 1$ и $x_2 = 2$.
- Кои приближения са по-добри и защо?

Отг. а) 1.32436; б) 2.18350; в) 1.15277, 2.01191.

3. Населението на САЩ от 1930 до 1980 се е променяло, както следва

Година	1930	1940	1950	1960	1970	1980
Насел. в хил.	123203	131669	150697	179323	203212	226505

Намерете полинома на Лагранж от степен 5, който интерполира данните и го използвайте, за да оцените големината на популацията в следните години: 1920, 1965 и 2000.

4. Използвайте таблицата от предната задача и разделени разлики, за да намерите приближено населението в 1965 година.

5. Можем да решим приближено уравнението

(4.8)
$$f(x) = 0, x \in [a, b],$$

като използваме интерполиране. Да предположим, че обратната функция $x = f^{-1}(y)$ съществува и е единствена (напр. ако $f(x)$ е монотонна). Тогава ако са известни стойностите $y_i = f(x_i)$, $i = 0, 1, \dots, n$ на $f(x)$ в $n + 1$ точки, то и $f^{-1}(y)$ е известна за $n + 1$ стойности:

$$x_i = f^{-1}(y_i), i = 0, 1, \dots, n$$

и можем да я интерполираме с полинома на Лагранж $L_n(y)$ от степен n . Това се нарича **обратна интерполация** тъй като $f^{-1}(y) \approx L_n(y)$. Решението x^* на (4.8) приблизително е $f^{-1}(0) \approx L_n(0)$.

С помощта на обратното интерполиране може да се уточни корена x^* , получен с някои от методите за приближено решаване на (4.8). Ако да речем x_0, x_1, \dots, x_n са намерени по метода на простата итерация и $L_n(y)$ е построен, както е описано по-горе, то $n + 1$ -то приближение може да се вземе, като

$$x_{n+1} = L_n(0).$$

След това можем да повторим описаната процедура с x_1, x_2, \dots, x_{n+1} и т.н. Можем да продължаваме докато разликата между две последователни приближения не стане по-малка по абсолютна стойност от някое отнапред зададено "малко" число ε .

Напишете програма за метода на обратното интерполиране и намерете приближените стойности на реалните корени на

а) $4x^3 + 6x^2 + 12x - 45 = 0$

б) $x^3 - 3x^2 + 9x + 38 = 0$

с три точки. т.е. $n = 2$ и $\varepsilon = 10^{-4}$. Сравнете с точните стойности на корените, които съответно са $x = 1,5$ и $x = -2$.

6. Използвайте натурален кубичен сплайн, за да апроксимирате

а) $f(2,5)$. ако

x	2.2	2.4	2.6
$f(x)$	0,5207843	0,5104147	0,4813306

б) $f(3)$, ако

x	1	2	4	5
$f(x)$	0	0	2	2

7. Намерете апроксимацията на Паде от степен 5 за $f(x) = e^{-x}$ с:

а) $n = 3$ и $m = 2$; б) $n = 1$ и $m = 4$.

Сравнете двата резултата за $x_i = 0,2i$, $i = 1, 2, \dots, 5$ с тези получени от полинома на Маклорен от степен не по-висока от 5-та.

8. Намерете апроксимацията на Паде от степен 6 с $n = m = 3$ за $f(x) = \sin(x)$. Сравнете резултата в $x_i = 0,1i, i = 0,1, \dots, 5$ с точните стойности и тези на полинома на Маклорен от степен не по-висока от 6-та.

9. Химичен експеримент дава следните данни

x	-1.000	-0.960	-0.860	-0.790	0.220	0.500	0.930
y	-1.000	-0.151	0.894	0.986	0.895	0.500	-0.306

Да се оценят стойностите на $y(x)$ за x между -1.000 и 1.000. като:

- а) се нанесат данните и се приближат интуитивно с гладка крива;
- б) се намери и построи графиката на полинома на Лагранж;
- в) се построи натуралния кубичен сплайн. интерполиращ данните.

Доколко кривите от а) и б) отразяват тенденцията на изменение на данните?

10. Нека $x_i = i - 11, i = 1, 2, \dots, 21$ и

$$y_i = \begin{cases} 1 & \text{за } i = 11 \\ 0 & \text{за } i \neq 11 \end{cases}$$

Начертайте графиките на полинома на Лагранж и на натуралния кубичен сплайн, интерполиращи данните. Сравнете двете графики.

11. По-долу имаме извадка от равномерно табулирани данни за термодвойка:

$T^{\circ}\text{F}$	0	20	40	60	80	100
MV	-0.670	-0.254	0.171	0.609	1.057	1.517

Да се намери стойността за $T = 55^{\circ}\text{F}$. като се използва полиномът на Лагранж. Каква ще е тази стойност, ако интерполирате с кубичен сплайн?

ЧИСЛЕНО ИНТЕГРИРАНЕ

Един определен интеграл

$$I \stackrel{\text{def}}{=} \int_a^b f(x) dx$$

може да бъде изчислен аналитично с помощта на формулата на Нютон-Лайбниц. Със сигурност обаче случаите, когато това е възможно, са по-малко от случаите, когато не е. Например в често използваната функция на грешката

$$\text{erf}(b) = \frac{2}{\sqrt{\pi}} \int_0^b f(x) dx$$

интегралът не може да бъде изчислен като неопределен.

От друга страна, функцията $f(x)$ може да е зададена само в краен брой точки в интервала $[a, b]$ (табулирана), което веднага поставя въпроса, как да изчислим I приближено.

Два са основните начина да се направи това: а) да се замени $f(x)$ с лесно интегрируема функция, напр. полином и да се пресметне интегралът от него; б) да се сметне

$$\int_a^b f(x) dx \approx \sum_{i=0}^n A_i f(x_i)$$

където $x_i \in [a, b]$, $A_i \in \mathbf{R}$, $i = 0, 1, \dots, n$ трябва да се зададат.

Между двата подхода има връзка. Ако интегрираме полинома, интерполиращ функцията $f(x)$ в x_0, x_1, \dots, x_n , ще получим формула от типа (5.1). Когато мрежата x_i $i = 0, 1, \dots, n$ в $[a, b]$ е равномерна, т.е. $x_{i+1} - x_i = h$, $i = 0, 1, \dots, n-1$, формулите (5.1) са известни като формули на Нютон-Коутс.

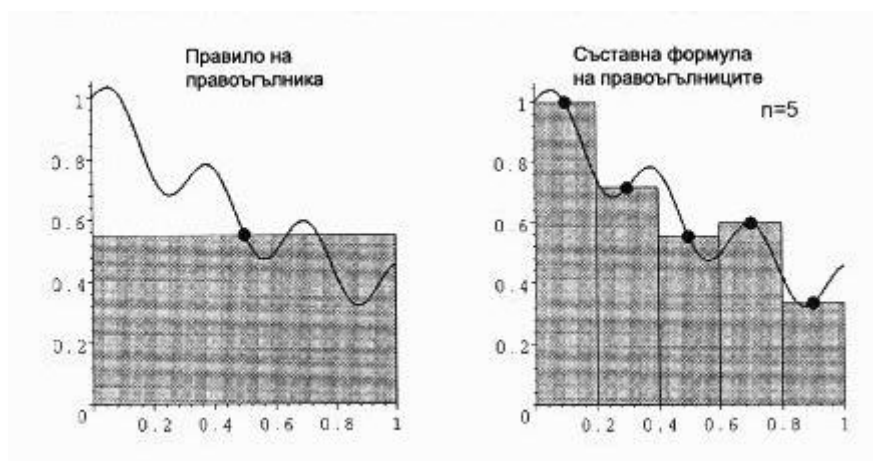
Ще разгледаме някои от тях, които освен това имат просто геометрично тълкуване.

5.1 ФОРМУЛА НА ПРАВОЪГЪЛНИЦИТЕ

Да разгледаме полинома на Лагранж $L_0(x)$ от степен $m = 0$, построен по единствения възел $x_0 = (a + b)/2 \in [a, b]$ и съответната стойност на $f(x)$. Очевидно $L_0(x) = f(a + b)/2$ и след интегриране над $[a, b]$ ще получим

$$\int_a^b f(x) dx \approx (b - a) f\left(\frac{a + b}{2}\right).$$

Името на правилото се изяснява от фигурата (отляво)



Фиг.5.1

тъй като площта, зададена от интеграла от $f(x)$, заместваме с площта на съответния правоъгълник.

Очевиден недостатък на горната формула е, че грешката, т.е. разликата между истинската и приближената стойност на интеграла, може да е твърде голяма. В някаква степен нещата могат да се оправят ако разделим $[a, b]$ на n равни интервала чрез точките $x_i = a + ih$, $i = 0, 1, \dots, n$, където $h = (b - a)/n$. Събирайки полученото във всеки от n -те интервала, получаваме *съставната формула*

$$\int_a^b f(x) dx = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \approx \sum_{i=0}^{n-1} h f\left(\frac{x_{i+1} + x_i}{2}\right)$$

След като изразим събираемостта, съдържащо грешката (това е изпуснато тук, но може да се направи с помощта на реда на Тейлър), получаваме *формулата на правоъгълниците*

$$\int_a^b f(x)dx = h \left[f\left(\frac{x_0 + x_1}{2}\right) + f\left(\frac{x_1 + x_2}{2}\right) + \dots + f\left(\frac{x_{n-1} + x_n}{2}\right) \right] + \frac{(b-a)h^2}{24} f''(\xi).$$

Грешката е изразена с втората производна на $f(x)$, изчислена в някоя точка $\xi \in (a, b)$ и с $(b-a)h^2 = O(1/n^2)$, което е по-важното. Този израз показва сходимост от втори ред по отношение на броя на подинтервалите n . Формулата е точна за полиноми от степен 0.

Пример 1. Да приложим за интеграла

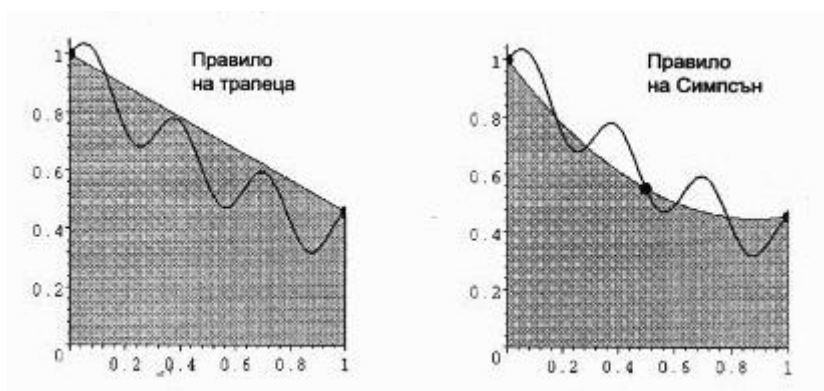
$$I \stackrel{\text{def}}{=} \int_a^b \frac{dx}{1+x^2} = \frac{\pi}{4} = 0,7853981634$$

правилото на правоъгълниците с $n = 4$, т.е. с $h = 1/4$. Резултатът е

$$I \approx \frac{1}{4} \left(\frac{64}{65} + \frac{64}{73} + \frac{64}{89} + \frac{64}{113} \right) \approx 0,7867001296$$

5.2 ФОРМУЛА НА ТРАПЕЦИТЕ

Тук ще използваме интерполационен полином, построен по точките $x_0 = a$ и $x_1 = b$. Резултатът от интегрирането е илюстриран на фиг. 5.2, в ляво.



Фиг.5.1

$$\int_a^b f(x)dx \approx \frac{(b-a)}{2} (f(x_0) + f(x_1)).$$

Съответната съставна формула е

$$\int_a^b f(x)dx = \sum_{i=1}^{n-1} \int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{h}{2}(y_1 + y_2) + \dots + \frac{h}{2}(y_{n-1} + y_n),$$

където $x_i = a + ih$, $h = (b - a)/n$, $y_i = f(x_i)$, $i = 0, 1, \dots, n$. Опростявайки (отново без да извеждаме израза за грешката), получаваме *формулата на трапеците*

$$\int_a^b f(x)dx = \frac{h}{2}(y_0 + 2y_1 + 2y_2 + \dots + 2y_{n-1} + y_n) - \frac{(b-a)h^2}{12} f''(\xi).$$

Формулата е точна за полиноми от степен, не по-голяма от 1.

Пример 2. Ще изчислим приближено същия интеграл като в предния параграф

$$I \stackrel{\text{def}}{=} \int_0^1 \frac{dx}{1+x^2} = 0,7853981634$$

по формулата на трапеците с $h = 0,25$. Резултатът

$$I = \frac{0,25}{2} [1 + 0,5 + 2(0,9411764706 + 0,8 + 0,64)] \approx 0,7827941176$$

5.3 ФОРМУЛА НА СИМПСЪН

Следвайки схемата от предишните параграфи, да построим интерполяционен полином от степен 2 по точките $x_0 = a$, $x_1 = (a + b)/2$ и $x_2 = b$. Резултатът от неговото интегриране ще бъде, вж. фиг. 5.2 ОТНОВО

$$\int_a^b f(x)dx \approx \frac{(b-a)}{6} (f(x_0) + 4f(x_1) + f(x_2)).$$

Съставната формула за $2n$ интервала или *формулата на Симпсън* е

$$\int_a^b f(x)dx = \frac{h}{3} \left(y_0 + 4 \sum_{i=1}^n y_{2i} + 2 \sum_{i=1}^{n-1} y_{2i+1} + y_{2n} \right) - \frac{(b-a)h^4}{180} f^{(4)}(\xi).$$

Тя се използва широко и може да се запомни чрез мнемоничното правило

$$\frac{h}{3} (\text{първа} + 4 * \text{нечетни} + 2 * \text{четни} + \text{последна}).$$

Пример 3. Прилагаме формулата на Симпсън за същия интеграл

$$I \stackrel{\text{def}}{=} \int_0^1 \frac{dx}{1+x^2} = 0,7853981634$$

с $h=0,25$. Резултатът е

$$I \approx \frac{0,25}{3} [1 + 0,5 + 4(0,9411764706 + 0,64) + 2,0,8] \approx 0,7853921569$$

5.4 ИНТЕГРИРАНЕ ПО РОМБЕРГ

Да разгледаме *екстраполацията на Ричардсън*. Нека математическата формула M е приближена с помощта на числената процедура N , като е използвана стъпка h :

$$M = N(h) + ch^n + O(h^m), \quad \text{където } m > n.$$

С друга стъпка ah , $a \neq 1$, $a > 0$

$$M = N(ah) + c(ah)^n + O(a^m h^m)$$

Елиминирайки членовете с h^n получаваме

$$M = \frac{a^n N(h) - N(ah)}{a^n - 1} + O(h^m).$$

Сега грешката в M е от ред $m > n$.

Ще приложим екстраполацията на Ричардсън за формулата на трапеците. Първо, използвайки стъпка $h_1 = (b - a)/m$ (m интервала), получаваме от формулата на трапеците

(5.2)

$$\int_a^b f(x)dx = T_{11} + O(h_1^2) = \frac{h_1}{2} \left[f(a) + f(b) + 2 \sum_{i=1}^{m-1} f(a + ih_i) \right] + O(h_1^2).$$

Формулата на трапеците със стъпка $h_2 = h_1/2$ ($2m$ интервала) дава

(5.3)

$$\int_a^b f(x)dx = T_{21} + O((h_1/2)^2)$$

$$= \frac{h_2}{2} \left[f(a) + f(b) + 2 \sum_{i=1}^{2m-1} f(a + ih_2) \right] + O((h_1/2)^2).$$

Тогава, от екстраполацията на Ричардсън. числото

$$(5.4) \quad T_{22} = \frac{4T_{21} - T_{11}}{3}$$

приближава интеграла I с точност h_2^3 или по-добра. Всъщност грешката е $O(h_2^4)$. защото подробното развитие на формулата (5.4) ще покаже, че T_{22} е формулата на Симпсън за $2m$ интервала.

С $h_3 = h_2/2 = h_1/2^2$ формулата на трапеците дава T_{31} , след което от екстраполацията на Ричардсън получаваме

$$(5.5) \quad T_{32} = \frac{4T_{31} - T_{21}}{3}$$

имаща грешка $O(h_3^4)$ (както в (5.4)).

Прилагайки екстраполационната процедура към (5.4) и (5.5), намираме

$$T_{33} = \frac{4^2 T_{32} - T_{22}}{4^2 - 1},$$

която е от ред h_3^5 или по-добра, във всъщност е от ред h_3^6 . Става ясно, че можем да построим по този начин таблицата

$$\begin{array}{cccc} T_{11} & & & \\ T_{21} & T_{22} & & \\ T_{31} & T_{32} & T_{33} & \\ \vdots & \vdots & \vdots & \ddots \\ T_{n1} & T_{n2} & T_{n3} & \dots & T_{mn} \end{array}$$

от приближения на I , като използваме следната зависимост между стълбовете

$$(5.6) \quad T_{ij} = \frac{4^{j-1} T_{ij-1} - T_{i-1j-1}}{4^{j-1} - 1}$$

при всеки $i = 2, 3, \dots, n$ и $j = 2, \dots, i$.

При интегрирането по Ромберг $h_l = b - a$. Означаваме приближенията с R_{ij} . Те могат да се подредят в таблица както по-горе, като зависимостта

между стълбовете е като в (5.6). Ако разгледаме по-подробно (5.2) и (5.3), ще видим, че

$$(5.7) \quad R_{ij} = \frac{1}{2} \left[R_{i-1,1} + h_{i-1} \sum_{k=1}^{2^{i-2}} f(a + (k-1/2)h_{i-1}) \right], \\ i=2,3,\dots,n.$$

Така, че преходът от стъпка h_{i-1} към $h_i = h_{i-1}/2$, т.е. от $R_{i-1,1}$ към R_{i1} , може да се извърши, като изчислим f само в новите точки.

В заключение, приближаването на I по метода на Ромберг се осъществява по следния начин. Нужно е да се съхраняват само два реда от таблицата. Ако знаем един цял неин ред (напр. R_{1l}), можем да изчислим елемента в следващия ред, лежащ в първия стълб (т.е. R_{2l}), по правилото на трапеците, като използваме (5.7). Останалите от елементите във втори ред (т.е. R_{22}) се изчисляват чрез (5.6), където T_{ij} се заместват с R_{ij} . Елементите от диагонала R_{nn} клонят по-бързо към I , отколкото R_{n1} при $n \rightarrow \infty$.

Пример 4. Да приложим алгоритъма на Ромберг към нашия тестов интеграл

$$I \stackrel{\text{def}}{=} \int_0^1 \frac{4dx}{1+x^2} = \pi.$$

Намираме за R_{ij} следната таблица

3
3,1 3,133333
3,131177 3,141567 3,142117
3,138989 3,141593 3,141595 3,141586
3,140942 3,141593 3,141593 3,141593 3,141593

Забележете, че диагоналят клони по-бързо от първия стълб (който е изчислен по формулата на трапеците чрез разполовяване на стъпката) към I

ЗАДАЧИ

1. Покажете, че формулата на Симпсън е точна за полиноми от степен, не по-голяма от 3.

Упътване: Използвайте правилото в $[-h, h]$ за произволна кубична функция и покажете, че резултатът съвпада с точния, който е

$$\int_{-h}^h (ax^3 + bx^2 + cx + d)dx = \frac{h}{3}(2bh^2 + 6d).$$

2.Докажете, че

$$\int_0^{\infty} e^{-x} f(x) dx \approx \frac{2 + \sqrt{2}}{2} f(2 - \sqrt{2}) + \frac{2 - \sqrt{2}}{4} f(2 + \sqrt{2})$$

е точна за полиноми от степен 3.

3.Приложете правилото на Симпсън с $h = 0,02$ и $h = 0,01$ за интеграла:

$$a) \int_0^1 \frac{dx}{1+x^2}; \quad б) \int_0^1 e^{-x} dx$$

Оценете грешката, като използвате производни, и я сравнете с грешката, получена от екстраполацията на Ричардсън

4.Изчислете следните интеграли, като използвате съставното правило на правоъгълниците със стъпки $h = 0,1$ и $h = 0,01$:

$$a) \int_0^7 \sqrt{1 + \cos x} \cos x dx; \quad e) \int_0^7 \cos(x + 10) \sin \sqrt{1 + \cos x} dx;$$

$$б) \int_0^4 2 + \ln |1 + \sin(x^3)| dx; \quad ж) \int_{10}^{25} -|1 - \ln(1 + \sin x)| \cos x dx;$$

$$в) \int_0^{15} 1 - \ln |1 + \sin x| dx; \quad з) \int_0^{20} [\ln(x^4 + 5)] / \sin x dx;$$

$$г) \int_{0,01}^{0,05} \sin^{20}(\ln x^7) dx; \quad и) \int_{-1}^3 \sum_{n=1}^{30} (-1)^n |x - \frac{n}{30}| dx;$$

$$д) \int_2^{20} [\ln(x^4 + 5)] / \sin x dx;$$

Направете предположение за реда на грешката на апроксимиране според резултата.

5.Изчислете интегралите от предишната задача, като използвате съставното правило на трапеците и съставното правило на Симпсън със стъпка $h = 0,1, 0,01$ и $0,001$.

6.Използвайте съставното правило на трапеците с дадените стойности за n , за да изчислите следните определени интеграли:

$$a) \int_1^3 \frac{dx}{x}, n = 4; \quad г) \int_0^1 \sin \pi x dx, n = 6;$$

$$б) \int_0^2 x^3 dx, n = 4; \quad д) \int_0^1 x \sin x dx, n = 8;$$

в) $\int_0^3 x\sqrt{1+x^2}dx, n = 6;$ е) $\int_0^1 x^2 e^x dx, n = 8;$

Сравнете приближените с точните резултати

7.Използвайте интегриране по Ромберг, за да изчислите R_{33} за интегралите от предишната задача, и сравнете с получените там резултати.

8.Намерете приближение на площта, оградена от графиката на функцията

$$y = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x/\sigma)^2/2}$$

и x -оста върху интервала $[-\sigma, \sigma]$, като използвате

а) съставната формула на трапеците

б) съставната формула на Симпсън

с $n = 8$.

9.Намерете дължината на обиколката на елипса с уравнение $4x^2 + 9y^2 = 36$ с точност 10^{-6} .

10.Кола прави пълна обиколка на писта за 84s. Скоростта на колата за всеки 12 секунден интервал се измерва с помощта на радар в m/s и е дадена в следната таблица

Време	0	12	24	36	48	60	72	84
Скорост	37.8	45.1	44.8	36.9	30.2	23.8	32.9	37.5

Каква е дължината на пистата?

11. Частица с маса m движеща се във флуид, изпитва съпротивление R , което е функция на скоростта v . Връзката между съпротивлението R , скоростта v и времето t е

$$t = \int_{v(t_0)}^{v(t)} \frac{m}{R(u)} du$$

Да предположим, че $R(v) = -v\sqrt{v}$ за един определен флуид, където R се измерва в $[N]$ и v - в $[m/s]$. Ако $m = 10 \text{ kg}$ и $v(0) = 10 \text{ m/sec.}$, определете приближено времето, необходимо на частицата да се забави до $v = 5 \text{ m/sec.}$

12. Често срещана задача е да се реши интегралното уравнение

$$f(x) + \int_a^b K(t, x)u(t)dt = y(x),$$

където f и K са дадени, а се търси y .

Ако приближим интеграла с квадратурна формула, т.е

$$\int_a^b K(t, x)y(t)dt \approx \sum_{i=1}^n \alpha_i K(x, t_i)y(t_i),$$

то интегралното уравнение се превръща в система линейни алгебрични уравнения

$$f(t_j) + \sum_{i=1}^n \alpha_i K(t_j, t_i)y(t_i) = y(t_j), j = 1, 2, \dots, n.$$

Решението $y(t_i)$ е приближение на $y(t)$ в точките $t_i, i = 1, 2, \dots, n$. Използвайки формулата на Симпсън, определете приближено решението на

$$\frac{4x^3 + 5x^2 - 2x + 5}{8(1+x)^2} + \int_0^1 \left(\frac{1}{1+t} - x \right) y(t)dt = y(x).$$

Апроксимирайте със сплайн $y(t)$ по точките $(t_i, y(t_i))$. Сравнете получения сплайн с точното решение $y(x) = 1/(1+x)^2$ за различни точки в интервала $[0, 1]$.

АПРОКСИМАЦИЯ

Апроксимирането (на функция) включва по-широк клас от задачи от интерполирането. Отново се търси "по-проста" функция $F(x)$, която е "близо" до дадена функция $f(x)$. Когато $f(x)$ е табулирана, т.е. зададена за крайно множество стойности на аргумента x , използваме дискретен тип критерии. Ако искаме по-простата функция $F(x)$ да удовлетворява

$$f(x_i) = F(x_i), i = 0, 1, \dots, m,$$

просто стигаме до задачата за интерполация. Ако избираме $F(x)$ да удовлетворява

(6.1)

$$\sum_{i=0}^m (f(x_i) - F(x_i))^2 \rightarrow \min,$$

то решаваме задачата за апроксимиране на $f(x)$ по (дискретния) метод на най-малките квадрати.

Добри критерии за близост също са

$$\sum_{i=0}^m |f(x_i) - F(x_i)| \rightarrow \min \quad \text{или}$$

(6.2)

$$\max_{i=0, \dots, m} |f(x_i) - F(x_i)| \rightarrow \min,$$

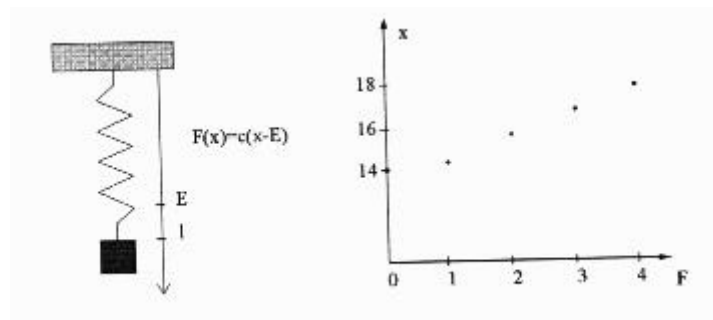
но (6.1) има някои хубави свойства, които ще дискутираме по-късно

Когато е полином от степен $n < m$, т.е. $F(x) = a_n x^n + \dots + a_1 x$, задачата (6.1) е да се намерят коефициентите $a_i, i = 0, \dots, n$ и тя винаги има единствено решение. За да разберем, защо по принцип степента n е по-малка от броя на данните m , нека разгледаме следния пример.

Към пружина, направена от хомогенен материал и имаща начална дължина 14 *cm*, последователно са приложени сили от 1, 2, 3 и 4 *kg*. Дължината нараства съответно до 14.3, 15.7, 17 и 18 *cm*.

По закона на Хук дължината на пружината зависи линейно от силата, т.е. $F(x) = c(x - E)$, където $F(x)$ представлява силата, необходима да се разтегли пружината с x единици, E е дължината, когато няма приложена сила, и c е константата на пружината.

Да предположим, че искаме да определим c . От фигурата може да се види, че данните не лежат на права линия.



Фиг. 6.1

Можем да изберем случайно две точки и да намерим от линията, минаваща през тях, стойността на c . Но така в зависимост от избора ще получаваме различни стойности за c . Означава ли това, че законът на Хук не е верен? Не, просто измерванията са неточни и е по-добре да използваме (6.1).

6.1 МЕТОД НА НАЙ-МАЛКИТЕ КВАДРАТИ - ДИСКРЕТЕН СЛУЧАЙ

Да формулираме задачата. За функцията $f(x)$, дефинирана върху $[a, b]$ и зададена чрез таблицата

x	x_0	x_1	\dots	x_m
y	y_0	y_1	\dots	y_m

където $y_i = f(x_i)$, намерете полином $F(x) = P_n(x) = \sum_{k=0}^n a_k x^k$, за който

$$E(a_0, a_1, \dots, a_n) = \sum_{i=0}^m (y_i - P_n(x_i))^2$$

има минимална стойност.

Коефициентите a_i ще разглеждаме като неизвестни и трябва да ги определим по такъв начин, че грешката на метода на най-малките квадрати $E(a_0, a_1, \dots, a_n)$ да има минимум. Тогава $\partial E / \partial a_k = 0$, $k = 0, \dots, n$ или

$$\frac{dE}{da_k} \equiv \sum_{i=0}^m \frac{d}{da_k} (y_i - P_n(x_i))^2 \equiv -2 \sum_{i=0}^m (y_i - P_n(x_i)) x_i^k = 0$$

Следователно

$$\sum_{i=0}^m P_n(x_i) x_i^k = \sum_{i=0}^m y_i x_i^k, k = 0, \dots, n,$$

като наричаме тези уравнения нормални. По-подробно, имаме

$$(6.3) \quad \begin{aligned} a_0 \sum_{i=0}^m x_i^0 + a_1 \sum_{i=0}^m x_i^1 + \dots + a_n \sum_{i=0}^m x_i^n &= \sum_{i=0}^m y_i x_i^0 \\ a_0 \sum_{i=0}^m x_i^0 + a_1 \sum_{i=0}^m x_i^2 + \dots + a_n \sum_{i=0}^m x_i^{n+1} &= \sum_{i=0}^m y_i x_i^1 \\ \vdots \\ a_0 \sum_{i=0}^m x_i^0 + a_1 \sum_{i=0}^m x_i^{n+1} + \dots + a_n \sum_{i=0}^m x_i^{2n} &= \sum_{i=0}^m y_i x_i^n \end{aligned}$$

Пример 1. В примера от началото $n = 1$ и $m = 3$. Тогава системата за коефициентите a_0 и a_1 в $F(x) = a_1 x + a_0 \equiv cx - cE$ е

$$\begin{aligned} a_0 \sum_{i=0}^3 x_i^0 + a_1 \sum_{i=0}^3 x_i^1 &= \sum_{i=0}^3 y_i x_i^0 \\ a_0 \sum_{i=0}^3 x_i^1 + a_1 \sum_{i=0}^3 x_i^2 &= \sum_{i=0}^3 y_i x_i^1 \end{aligned}$$

и тя се свежда до

$$\begin{aligned} 4a_0 + (x_0 + x_1 + x_2 + x_3)a_1 &= \sum_{i=0}^3 y_i \\ (x_0 + x_1 + x_2 + x_3)a_0 + (x_0^2 + x_1^2 + x_2^2 + x_3^2)a_1 &= \sum_{i=0}^3 y_i x_i \end{aligned}$$

Замествайки със стойностите за x_i и y_i получаваме

$$\begin{aligned} 4a_0 + 10a_1 &= 65 \\ 10a_0 + 30a_1 &= 168.7 \end{aligned}$$

Решението е $a_0 = 13.15$ и $a_1 = 1.24$. Тогава от $c = a_1$ намираме стойността на константата на пружината.

Детерминантата на матрицата от коефициентите на системата (6.3) винаги е различна от нула. Може да се докаже, че съответното единствено решение на (6.3) наистина дава минимум на $E(a_0, a_1, \dots, a_n)$. С цел да се изравнят мерните единици обикновено се използва

$$(6.4) \quad \Delta = \sqrt{E(a_0, a_1, \dots, a_n)/n}$$

за оценка на грешката.

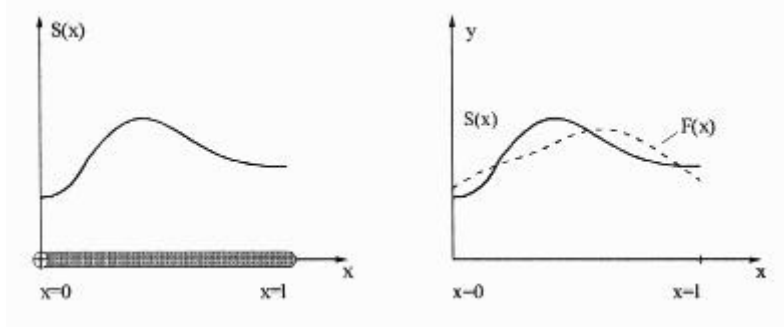
Недостатък на метода е, че (6.3) е лошо обусловена (вж. Глава 3) за големи m и са необходими специални мерки, за да бъде решена правилно. Ако използваме като критерий за близост (6.2), то не е така просто да се намерят a_1 или система подобна на (6.3), защото $|\cdot|$ който се използва и при двата критерия не е диференцируем в 0. За тези случаи, на базата на теорията на негладкия анализ са разработени други методи, вж. [6].

6.2 МЕТОД НА НАЙ-МАЛКИТЕ КВАДРАТИ -НЕПРЕКЪСНАТ СЛУЧАЙ

Сега ще разгледаме задачата за апроксимиране на функция вместо вече разгледаното приближаване на набор от данни (или табулирана функция). Да предположим, че $f(x)$ е непрекъсната в $[a, b]$ и се търси "по-проста" непрекъсната функция $F(x)$, удовлетворяваща

$$(6.5) \quad \int_a^b (f(x) - F(x))^2 dx \rightarrow \min$$

Такава задача възниква, ако разгледаме разпределението на напрежението $S(x)$ (при опън) в хомогенен прът по оста x от $x = 0$ до $x = l$. вж. 6.2.



Фиг. 6.2

Енергията в пръта, породена от опъването, е пропорционална на интеграла

$$\int_0^1 [S(x)]^2 dx$$

Близостта на апроксимацията $F(x)$ до $S(x)$ може да се оцени с помощта на енергията от разликата на двете разпределения. А именно, тя е пропорционална на

$$\int_0^1 [S(x) - F(x)]^2 dx$$

което е критерият (6.5) на метода на най-малките квадрати.

Когато в (6.5) $F(x) = P_n(x) = \sum_{k=0}^n a_k x^k$ е полином, можем да действаме като в предишния параграф и да намерим следната система от **нормални уравнения**

(6.6)

$$\sum_{i=0}^n a_i \int_a^b x^{i+k} dx = \int_a^b x^k f(x) dx, k = 0, \dots, n.$$

Но ние можем да поставим въпроса по-общо и да потърсим

$$F(x) = c_0 \psi_0(x) + c_1 \psi_1(x) + \dots c_n \psi_n(x),$$

където ψ_i са дадени функции, имащи някои хубави свойства. Тогава системата от нормални уравнения за $c_i, i = 0, 1, \dots, n$ ще бъде

(6-7)

$$\sum_{i=0}^n c_i \int_a^b \psi_i(x) \psi_k(x) dx = \int_a^b f(x) \psi_k(x) dx, \\ k = 0, 1, \dots, n.$$

Лесно е да се види, че ако $\psi_i(x) = x^i$ т.е. $F(x)$ е полином, (6.7) се превръща в (6.6).

Изчисляването на c_i се опростява много, ако ψ_i удовлетворява

$$\int_a^b \psi_i(x) \psi_k(x) dx = \begin{cases} 0, & i \neq k \\ 1, & i = k \end{cases}$$

Тогава от (6.7)

$$c_k = \int_a^b \psi_i(x) \psi_k(x) dx, k = 0, 1, \dots, n.$$

Един такъв избор на $\psi_k, k = 0, 1, \dots, 2n$, когато $[a, b] = [-\pi, \pi]$, е

$$\psi_0(x) = \frac{1}{\sqrt{2\pi}}, \psi_k(x) = \frac{1}{\sqrt{\pi}} \cos kx, \psi_{k+n}(x) = \frac{1}{\sqrt{\pi}} \sin kx, \\ k = 1, 2, \dots, n.$$

Означавайки

$$a_0 \equiv c_0 = \frac{1}{\sqrt{2\pi}} \int_a^b f(x) dx, \\ (6.8)$$

$$a_k \equiv c_k = \frac{1}{\sqrt{\pi}} \int_{-\pi}^{\pi} f(x) \cos kx dx, \\ b_k \equiv c_{n+k} = \frac{1}{\sqrt{\pi}} \int_{-\pi}^{\pi} f(x) \sin kx dx, k = 1, 2, \dots, n,$$

получаваме *средноквадратичното приближение*

$$F(x) = \frac{c_0}{2} + \sum_{k=1}^n (a_k \cos kx + b_k \sin kx),$$

което всъщност е редът на Фурие за f , отрязан след n -ия член. Този тип апроксимация се нарича *тригонометричен полином*.

6.3 БЪРЗО ПРЕОБРАЗУВАНИЕ НА ФУРИЕ

Съществува дискретен аналог на апроксимираното с тригонометрични полиноми, което току-що споменахме. Той е удобен при приближаване на голямо количество данни, x -координатите, на които са разположени през равни разстояния.

По-точно, нека са дадени данните $\{(x_j, y_j)\}_{j=0}^{2m-1}$. където

$$x_j = -\pi + \frac{1}{m}\pi, j = 0, 1, \dots, 2m-1$$

са разположени равномерно в $[-\pi, \pi]$. Тогава тригонометричният полином

$$S_m(x) = \frac{a_0 + a_m \cos mx}{2} + \sum_{k=1}^{m-1} (a_k \cos kx + b_k \sin kx),$$

който апроксимира данните по метода на най-малките квадрати

$$\sum_{j=1}^{2m-1} (y_j - S_m(x_j))^2 dx \rightarrow \min$$

има следните коефициенти
(6.9)

$$a_k = \frac{1}{m} \sum_{j=1}^{2m-1} y_j \cos kx_j, b_k = \frac{1}{m} \sum_{j=1}^{2m-1} y_j \sin kx_j$$

за $k = 0, 1, \dots, m$. Има очевидно сходство между тези a_k, b_k и a_k, b_k изчислени в (6.8).

Алгоритъмът, известен като **Бързо преобразуване на Фурие (БПФ)** (Fast Fourier Transform (FFT) на английски), е създаден, за да намали изчисленията необходими за намиране на коефициентите в (6.9). Вместо около $(2m)^2$ умножения и още толкова събирания, които са нужни в (6.9), БПФ се справя само с $O(m \log_2 m)$ на брой такива операции. Когато количеството на данните е голямо, разликата между тези две оценки е съществена. Например, ако $m = 2^{10}$ то $(2m)^2 \approx 4000000$, докато $O(m \log_2 m) \approx 2^{10} \times 10 \approx 10000$.

Основното в БПФ е, че първо се изчисляват комплексните коефициенти c_k във формулата

$$\phi(x) = \frac{1}{m} \sum_{k=0}^{2m-1} c_k e^{ikx} (i^2 = -1)$$

А именно

$$c_k = \frac{1}{m} \sum_{j=0}^{2m-1} y_j e^{\pi i k j / m}, k = 0, 1, \dots, 2m-1.$$

Връзката с (6.9) е

$$\frac{1}{m} c_k e^{-ixk} = a_k + ib_k.$$

Когато $m = 2^p$, p -положително и цяло, алгоритъмът е по-лесен за реализиране. За подробности, *вж. работна страница 6* от СО, придружаващ книгата.

6.4 ЗАДАЧИ

1. Намерете линейната функция, приближаваща по метода на най-малките квадрати следните данни:

а)

x	0	1	2	3	4
y	0	1	-2	3	-4

б)

x	0	1	2	3	4
y	0	1	4	9	16

в)

x	0	-1		1	2	-2
y	-3	-7	9	-1	1	2
				2		-8

г)

x	.25	.7	.81	1.13	1.23	.44	1.5	1.52	1.77	1.9
y	.1	.37	.5	.1	.97	.96	.9	.65	.06	1.19

2. Намерете полинома от степен 2, приближаващ данните от задача 1. Кой полином, от степен 1 или от степен 2, дава по-малка грешка на приближение (6.4)?

3. Изчислете полинома на Лагранж от втора степен за $f(x) = e^x$, като използвате възли $x_0 = 0$, $x_1 = 0.5$, $x_2 = 1$. Сравнете с полинома от втора степен, апроксимиращ по метода на най-малките квадрати данните

x	0	0.25	0.5	0.75	1.00
y	1.0000	1.2840	1.6487	2.1170	2.7183

с цел да определите кой от тях приближава по-добре $f(x) = e^x$ върху $[0,1]$.

4. Таблицата, която следва, съдържа точките от контролни и от крайния изпит на 30 студенти по приложна математика. Определете правата, апроскимираща данните по метода на най-малките квадрати, и използвайте нейното уравнение, за да намерите броя точки от домашни, който предполага съответно оценки 6 (90%) и 3 (60%) на крайния изпит.

Домашни	Изпит	Домашни	Изпит
302	45	323	82
325	72	337	99
285	54	337	71
339	54	304	62
334	79	319	66
322	65	234	51
331	99	337	53
279	63	351	100
316	65	339	67
343	83	314	42
290	74	344	79
326	76	185	59
347	99	343	83
233	57	340	75
354	45	316	45

5. По данните

x	4.0	4.2	4.5	4.7	5.1	5.5	5.9	6.3	6.8	7.1
y	102	113	130	142	167	195	224	256	299	326

а) Постройте по метода на най-малките квадрати приближенията от степен 1,2 и 3. Изчислете грешката;

б) Постройте по метода на най-малките квадрати приближение от вида be^{ax} и намерете грешката;

в) Постройте приближение от вида bx^a по метода на най-малките квадрати и намерете грешката.

6. Намерете полинома от степен 1, приближаващ по метода на най-малките квадрати $f(x)$ (т.е. средноквадратичното приближение) върху дадения интервал:

а) $f(x) = x^2 - 2x + 3, [0,1]$; г) $f(x) = x^3 - 1, [0,2]$;

б) $f(x) = 1/x, [1,3]$; д) $f(x) = e^{-x}, [0,1]$;

в) $f(x) = \cos \pi x, [0,1]$; е) $f(x) = \ln x, [1,2]$.

7. Пререшете горната задача за апроксимиращ полином от степен 2.

8. Намерете тригонометричния полином, апроксимиращ

$$f(x) = x \text{ върху } [-\pi, \pi], \text{ с } n = 2.$$

9. Решете преопределените системи

а)	$x + y = 1$	б)	$x + y = 3$
	$y + z = 1$		$2x - y = 0.2$
	$x + z = 1$		$x + 3y = 7$
	$x + y + z = 1$		$3x + y = 5$

Отг. а) $x = y = z = 3/7$; б) $x = 1618/155, y = 302/155$.

ОБИКНОВЕНИ ДИФЕРЕНЦИАЛНИ УРАВНЕНИЯ

Обикновените диференциални уравнения често се използват за описание на практически задачи. За да посочим поне една - спомнете си уравнението, описващо движението на маса закрепена за пружина. В много случаи не е възможно да се намери аналитично решение така, че се налага то да се търси числено, т.е. приближено. Ще разгледаме процедури за числено решаване на класическата начална задача, наричана още **задача на Коши**:

$$(7-1) \quad y' = f(x,y) \text{ C } y(x_0) = y_0$$

Често тези процедури могат да се модифицират за решаване на системи от уравнения или уравнения от по-висок ред (включващи производни на неизвестната функция $y(x)$ от ред по-голям от първи). Ще видим как може да се направи това.

Можем да разгледаме решението на (7.1) като крива $y(x)$, започваща в точката (x_0, y_0) . Разделяме интервала $[a, b]$ на n равни части с точките $a = x_0 < x_1 < \dots < x_n = b$, където $x_i = x_0 + ih, i = 1, 2, \dots, N$ и $h = (b - a)/N$. След което, оценяваме нарастването Δy_0 в y , когато x нараства от x_0 до $x_1 = x_0 + h$ и h е стъпката. Ще имаме

$$(7.2) \quad y(x_1) = y(x_0 + h) \approx y_0 + \Delta y_0.$$

Можем да вземем $(x_1, y_0 + \Delta y_0)$ като нова начална точка и да повторим процеса. По този начин генерираме редица y_0, y_1, \dots, y_N от приближени стойности за решението в мрежата от точки x_0, x_1, \dots, x_N . Сега можем да оценим $y(x)$ за всяко $x \in [a, b]$, като използваме интерполация.

7.1 МЕТОД НА ОЙЛЕР

От дефиницията за производна $y' \approx (\Delta y / \Delta x)$ ако Δx е малко. Тогава

$$(7.3) \quad y(x_1) = y(x_0) + \Delta y_0 \approx y_0 + h y'_0,$$

където $h = \Delta x$ и $y'_0 = y'(x_0)$ Изобщо, за да преминем от $y(x_i)$ към $y(x_{i+1})$, използваме

$$(7.4) \quad y(x_{i+1}) \approx y(x_i) + h y'_i$$

От (7.1), $y'_i \equiv y'(x_i) = f(x_i, y_i)$ и получаваме

Формула на Ойлер:

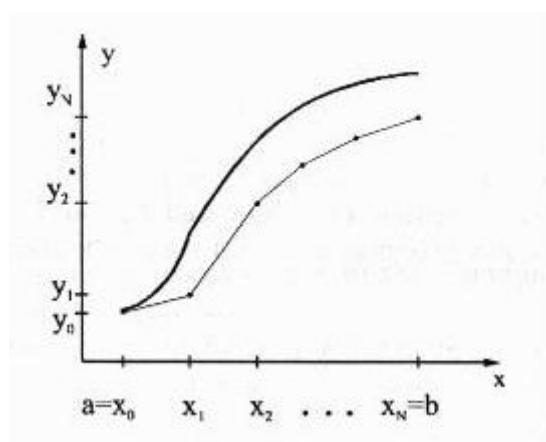
$$(7-5) \quad y_{i+1} = y_i + h f(x_i, y_i), i = 0, 1, \dots$$

Тук и по-нататък означаваме с y_i *приближение* на точната стойност $y(x_i)$ на аналитичното решение $y(x)$ за $x = x_i$.

Методът на Ойлер е най-простият метод за приближено решаване на диференциални уравнения. Може да се интерпретира геометрично като генериращ начупена линия с върхове $(x_0, y_0), (x_1, y_1), \dots, (x_N, y_N)$ вместо интегралната крива $y(x)$, като правата свързваща (x_i, y_i) и (x_{i+1}, y_{i+1}) има наклон (ъглов коефициент) $f(x_i, y_i) = y'(x_i)$. Фиг. 7.1.

За да оценим разликата $|y_i - y(x_i)|$, достатъчно е да забележим, че (7.4) е получена от развитието в ред на Тейлър около $x = x_i$

$$(7.6) \quad y(x_{i+1}) = y(x_i) + \frac{h}{1!} y'(x_i) + \frac{h^2}{2!} y''(x_i) + \dots + \frac{h^n}{n!} y^{(n)}(x_i) + \frac{h^{(n+1)}}{(n+1)!} y^{(n+1)}(\xi_i), x_i < \xi_i < x_{i+1}$$



Фиг. 7.1

отрязано след втория член Тогава, при определени условия за $f(x, y)$.

$$(7.7) \quad d_i \stackrel{\text{def}}{=} |y_{i+1} - y(x_{i+1}; y_i)| \leq Ch^2, \quad i = 1, 2, \dots, N,$$

където $y(x_{i+1}; y_i)$ е стойността на $y(x; y_0)$ за $x = x_{i+1}$. Тук с $y(x; y_0)$ сме означили аналитичното решение на уравнението (7.1) върху интервала $[x_0, b]$ и с начално условие $y(x_0) = y_0$. Да забележим, че $y(x; y_i)$ съвсем спокойно може да се различава от $y(x)$, защото по принцип интегралната крива на (7.1), започваща от (x_0, y_0) , не е нужно да минава през (x_i, y_i) . Поради това $y(x) \neq y(x; x_i)$ в точката x_i а оттам и върху целия интервал $[x_i, b]$.

Определение 1. Числото $d_i \stackrel{\text{def}}{=} |y_{i+1} - y(x_{i+1}; y_i)|$ се нарича **локална грешка (в x_i)** на метода Ако

$$d_i \leq Ch^{m+1}, \quad i = 1, 2, \dots, N,$$

където числото C не зависи от стъпката h , казваме, че методът е **от ред m** .

Оценката в (7.7) ни позволява да кажем, че **локалната грешка** на метода на Ойлер е $O(h^2)$. Тогава, според горното определение, методът на Ойлер е **от ред 1**. Защо обаче намаляваме степента в израза за локалната грешка с 1?

На всяка стъпка грешката от изчисление или идваща от метода се предава на следващата стъпка чрез (7.5). Ето защо се интересуваме от натрупаната или **глобалната грешка**

$$\max_{1 \leq i \leq N} |y_i - y(x_i)|.$$

За метода на Ойлер тя е $O(h)$, което може да се докаже, ако се допусне, че f е липшицова по y и $|\partial f / \partial x|$ е ограничена; вж. [2]. Но това е разумно да се приеме и без доказателство, като имаме пред вид, че $N = (b - a)/h$. Тогава от (7.7) $|y_N - y(x_N)| \leq N \cdot Ch^2 = C_1 h$.

Определение 2. Метод, за който **глобалната грешка** е $O(h^m)$, m - положително число, се нарича **метод от ред m** .

Както видяхме, методът на Ойлер е от първи ред така, че той не е много точен. Именно за това и не се използва в практиката.

7.2 МЕТОД НА РУНГЕ-КУТА

Зависимостта (7.3), от която получихме метода на Ойлер, е редът на Тейлър, отрязан след члена, съдържащ първата производна. Естествено можем да се опитаме да използваме по-голяма част от сумата в Тейлървия ред с цел да конструираме по-прецизен метод. Но това е свързано с тежки изчисления. Например, ако отрежем след третия член в (7.6), трябва да изчислим $y''(x_i)$ във всеки възел x_i .

От (7.1) имаме

(7.8)

$$\begin{aligned} y''(x_i) &= (y'(x_i))' = \frac{d}{dx} f(x_i, y(x_i)) \\ &= \left[\frac{df}{dx}(x, y) + \frac{df}{dy}(x, y) \cdot f(x, y) \right]_{x=x_i, y=y_i} \end{aligned}$$

и се налага да пресмятаме частните производни на f , което поглъща много време.

Методите на **Рунге-Кута** предлагат друго, много елегантно решение на задачата за конструиране на методи от ред $m > 1$. При тях се търсят функции k_i , за които

(7.9)

$$y(x_{i+1}) - y(x_i) = \sum_{p=1}^m w_p k_p + O(h^{m+1}).$$

Тук k_p са равни на f , изчислена в специално подбрани точки, и числата w_p са избрани така, че дясната страна на горното равенство съвпада с развитието в ред на Тейлър на $y(x_{i+1}) - y(x_i)$ до членове от ред h^m включително. Общата форма на функциите k_p е

$$k_p = hf \left(x_i + \alpha_p h, y_i + \sum_{q=1}^{p-1} \beta_{pq} k_q \right).$$

Тогава (предполага се, че $\alpha_1 = 0$)

$$\begin{aligned} k_1 &= hf(x_i, y_i) \\ k_2 &= hf(x_i + \alpha_2 h, y_i + \beta_{21} k_1) \\ k_3 &= hf(x_i + \alpha_3 h, y_i + \beta_{31} k_1 + \beta_{32} k_2) \end{aligned}$$

Нека да изведем формулите на Рунге-Кута от **втори ред** Като комбинираме (7.6) и (7.8). намираме

(7-10)

$$y(x_{i+1}) - y(x_i) = hf_i + \frac{h^2}{2} \left(\frac{df_i}{dx} + \frac{df_i}{dy} f_i \right) + \dots,$$

където f_i означава $f_i(x_i, y_i)$ и т.н. От (7 9) получаваме

$$(7.11) \quad y_{i+1} - y_i = w_1 k_1 + w_2 k_2$$

като

$$k_1 = hf_i$$

$$\begin{aligned} k_2 &= hf(x_i + \alpha h, y_i + \beta k_1) \\ &= h \left(f(x_i, y_i) + \alpha h \frac{df(x_i, y_i)}{dx} + \beta k_1 \frac{df(x_i, y_i)}{dy} + \dots \right) \\ &= h \left(f_i + \alpha h \frac{df_i}{dx} + \beta k_1 \frac{df_i}{dy} + \dots \right) \end{aligned}$$

Изразът за k_2 е получен чрез развитие в ред на Тейлър Слагайки k_1 и k_2 в (7.11), получаваме (до h^2)

(7.12)

$$y_{i+1} - y_i = w_1 hf_i + w_2 h \left(f_i + h\alpha \frac{df_i}{dx} + h\beta \frac{df_i}{dy} f_i \right).$$

Като приравним коефициентите в уравнения (7.12) и (7.10). имаме

$$\begin{aligned} hf_i: w_1 + w_2 &= 1 \\ h^2 \frac{df_i}{dx}: \alpha w_2 &= \frac{1}{2} \\ h^2 \frac{df_i}{dy} f_i: \beta w_2 &= \frac{1}{2} \end{aligned}$$

От горната система с три уравнения за неизвестните α, β, w_1 и w_2 следва, че $\alpha = \beta$ Можем да изберем w_1 и w_2 като коя да е двойка числа със сума 1, след това да определим α и β от второто и третото уравнения.

Изброените по-долу варианти са членове на *фамилията* формули на Рунге-Кута от втори ред:

а) $\alpha = 1/2, w_1 = 0, w_2 = 1$ (Метод на правоъгълниците)

$$y_{y+1} = y_i + hf\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}f(x_i, y_i)\right), i = 0, 1, \dots$$

б) $\alpha = 1, w_1 = 1/2, w_2 = 1/2$ (Модифициран метод на Ойлер)

$$y_{y+1} = y_i + \frac{h}{2}(f(x_i, y_i) + f(x_i + h, y_i + hf(x_i, y_i))), i = 0, 1, \dots$$

Като използваме подобни на горните аргументи, получаваме фамилията методи на **Рунге-Кута от четвърти ред**. Тъй като процедурата по извеждане на формулите е твърде дълга, просто ще представим най-популярната от тях:

Формула на Рунге-Кута от четвърти ред: Първо изчисляваме

$$k_1 = hf(x_i, y_i)$$

$$k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right)$$

$$k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_2}{2}\right)$$

$$k_4 = hf(x_i + h, y_i + k_3)$$

Тогава

(7.13)

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + k_2 + k_3 + k_4), i = 0, 1, \dots$$

Методите, които обсъдихме до този момент имат един сериозен недостатък - трудно е да се оцени грешката и тя да се контролира, като избираме стъпката h . Едно възможно разрешение е да се приложи формулата на Рунге-Кута със стъпка h . а после още веднъж - със стъпка $h/2$.

Така ще имаме приближени решения y_i^h и $y_i^{h/2}, i = 0, 1, \dots$

Чрез екстраполацията на Ричардсън (параграф 5.4), оценяваме локалната грешка по следния начин

$$|y(x_i) - y_i| \approx \frac{16}{15} \left| y_i^h - y_i^{\frac{h}{2}} \right|.$$

Ако се окаже, че оценката за локалната грешка е твърде голяма, вземаме два пъти по-малка стъпка и правим същото като по-горе, но този път с $h/2$ (резултатът вече е известен) и $h/4$. Можем да продължим този процес на разполовяване на стъпката, докато локалната грешка влезе в желаните граници. За съжаление този подход изисква твърде много изчисления.

Пример 1. Нека приложим разгледаните методи към

$$y' = (4x^3 - 18x^2 + 22x - 6)y, y(0) = 1$$

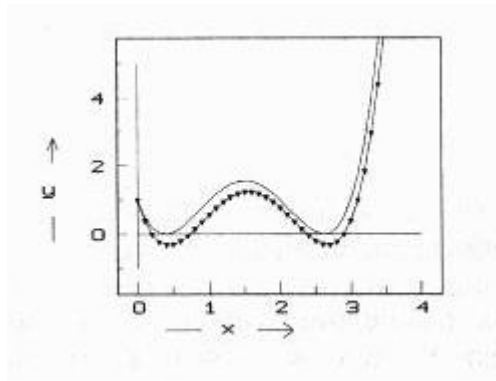
с различни стъпки h и да сравним резултатите с аналитичното решение (проверете!)

$$y(x) = \exp[x(x-1)(x-2)(x-3)].$$

Резултатите са обобщени в долната таблица.

Метод	h	$x = 1$	$x = 2$	$x = 3$
Ойлер	0.1	0.58	0.78	0.4
	0.01	0.9598	0.9788	0.94
	0.001	0.995998	0.997998	0.994
Рунге-Кута	0.1	1.0	1.0	1.0
	0.01	1.0	1.0	1.0
	0.001	1.0	1.0	1.0
Точно	1.0	1.0	1.0	1.0

Графичното им представяне при $h = 0.1$ е на **фиг. 7.2**. Приближението по Ойлер е нарисувано с линия, свързваща триъгълничета. Приближението по метода на Рунге-Кута не може да се отличи от точното решение, което е нарисувано с обикновена линия.



Фиг. 7.2

7.3 СИСТЕМИ ОТ ОДУ

Методите, които разгледахме по-горе, могат да се приложат за системи от обикновени диференциални уравнения (системи от ОДУ) с минимални промени. Да предположим, че се търси решението на следната задача на Коши

$$\begin{aligned}
 (7.14) \quad & y_1' = f_1(x, y_1, \dots, y_n), y_1(x_0) = y_{10} \\
 & y_2' = f_2(x, y_1, \dots, y_n), y_2(x_0) = y_{20} \\
 & \vdots \\
 & y_n' = f_n(x, y_1, \dots, y_n), y_n(x_0) = y_{n0}
 \end{aligned}$$

Да означим с y_{ki} приближение на $y_k(x_i)$, където отново $x_i = x_0 + ih$. По метода на **Ойлер** приближенията y_{ik} се получават като

$$\begin{aligned}
 (7.15) \quad & y_{k,i+1} = y_{ki} + hf_k(x, y_{1i}, \dots, y_{ni}), \\
 & k = 1, 2, \dots, n, i = 0, 1, \dots, N.
 \end{aligned}$$

Много по-удобни са векторните означения, като при тях векторите се представят с получерни букви. Като ги използваме, можем да препишем системата (7.14) като

$$y' = f(x, y), y(x_0) = y_0$$

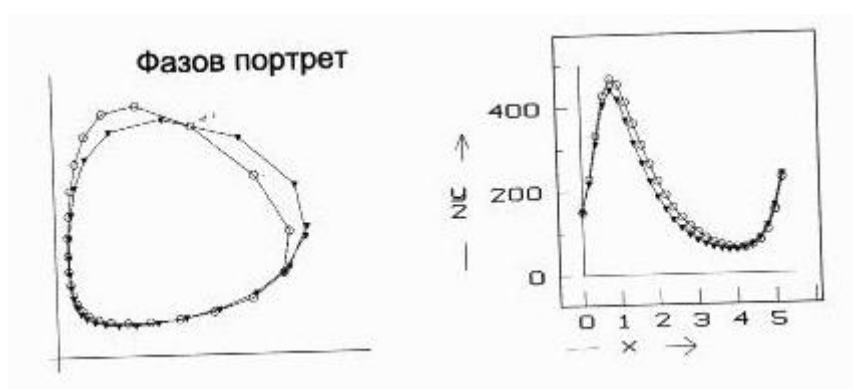
където $y \stackrel{\text{def}}{=} (y_1, y_2, \dots, y_n)$ и т.н. Очевидно това означение формално съвпада с (7.1). Използвайки същата уговорка, можем да препишем (7.15) като

$$y_{i+1} = y_i + hf(x_i, y_i), i = 0, 1, \dots,$$

която е идентична с формулата на Ойлер (7.5) за едно уравнение.

Не е нужно да споменаваме, че формулите на Рунге-Кута за системи от ОДУ са същите, ако използваме векторни означения. По-добре да разгледаме един пример, демонстриращ как системите от ОДУ могат да се използват като средство за моделиране.

Пример 2. Да разгледаме екологична система, състояща се от два вида - жертва (да кажем заек) и хищник (да кажем лисица). Един добър модел за тази популация се дава от системата на Лотка-Волтера от две ОДУ.



Фиг. 7.3

Предполагаме, че броят на зайците в момента x е $y_1(x)$ и те разполагат с неограничен запас от храна, а броят на лисиците, хранещи се със зайци, е $y_2(x)$. Естествено е да допуснем, че раждаемостта на жертвите (зайците) е право пропорционална на броя на живите жертви в момента x , т.е. е $k_1 y_1(x)$. Смъртността на жертвите се предполага, че е $k_2 y_2(x) y_1(x)$, т.е. зависи от броя на живите жертви и хищници едновременно. Аналогично приемаме, че раждаемостта на хищниците (лисиците) е $k_3 y_1(x) y_2(x)$, докато тяхната смъртност е $k_4 y_2(x)$. Всички $k_i > 0$

В нашия модел, производните представят изменението в популацията, така че

$$\begin{aligned} y_1'(x) &= k_1 y_1(x) - k_2 y_1(x) y_2(x) \\ y_2'(x) &= k_3 y_1(x) y_2(x) - k_4 y_2(x) \end{aligned}$$

Сега ще предположим, че началната популация на жертвите (зайците) и хищниците (лисиците) съответно е

$$y_1^0 = 300, \quad y_2^0 = 150$$

и $k_1 = 2, k_2 = k_3 = 0.01, k_4 = 1$. На **фиг. 7.3** (отдясно) са показани графиките на решението на системата за $x \in [0, 5]$, намерено по методите на Ойлер и Рунге-Кута от четвърти ред със стъпка $h = 0.02$.

Фазовият портрет, т.е. графиката в равнината (y_1, y_2) , е показан на фиг. 7.3 (отляво) като първо с начупена линия, свързваща триъгълничета, е представено решението по метода на Ойлер. На дясната графика, където y_2 е нарисувано като функция на x , може по-ясно да се види разликата в точността на двата метода.

Забележете периодичното поведение на състоянието на популацията

7.4 УСТОЙЧИВОСТ И ТВЪРДИ УРАВНЕНИЯ

Грубо казано, един метод е **неустойчив**, ако малки грешки (в началните условия, локални, от изчисления) в някоя стъпка предизвикват големи разлики на приближеното от точното решение в оставащата част от разглеждания интервал. В тази връзка нека припомним обсъждането от параграф 1.3. **Устойчивостта** на един метод е обратното свойство. Понякога обаче неустойчивостта на решението на една задача на Коши е вродена и не зависи от числения метод.

За да разберем за какво става дума, да разгледаме решението за $x \in [0,5]$ на следните две уравнения

$$(7.16) \quad y' = y \text{ и } y' = -y$$

с начално условие $y(0) = 0$, а после с начално условие $y(0) = 1, 2$ и 3 . Интегралните криви на първото уравнение увеличават разликата при $x = 0$ между решението $y = 0$ и другите решения, които започват от изместените от нулата начални условия. В резултат глобалната грешка е по-голяма от сумата от локалните грешки и $y \equiv 0$ е неустойчиво решение на $y' = y$. Обратно $y \equiv 0$ е устойчиво решение на $y' = -y$.

Друг капан обаче е свързан с второто уравнение в (7.16). Да предположим, че коефициентът пред y е по-голям, а именно -100 . Той се нарича *времеконстанта* понеже управлява скоростта на намаляване на решението. Нека разгледаме следната модификация на уравнението

$$y' = -100y + 100, y(0) = 2.$$

Точното решение е

$$(7.17) \quad y(x) = e^{-100x} + 1$$

и то клони бързо намалявайки към $y = 1$.

По метода на Ойлер намираме следното *диференчно* уравнение

$$(7.18) \quad y_{n+1} = y_n + h(-100y_n + 100).$$

Означавайки $\alpha = 1 - 100h$ и $b = 100h$, можем да го препишем като

$$\begin{aligned} y_{n+1} &= \alpha y_n + b = \alpha(\alpha y_{n-1} + b) + b = \alpha^2 y_{n-1} + (\alpha + 1)b \\ &= \alpha^3 y_{n-2} + (\alpha^2 + \alpha + 1)b = \dots \\ &= \alpha^{n+1} y(0) + (\alpha^n + \dots + \alpha + 1)b = 2\alpha^{n+1} + \frac{\alpha^{n+1} - 1}{\alpha - 1} b \end{aligned}$$

Откъдето решението на (7.18) е

$$(7.19) \quad y_n = (1 - 100h)^n + 1,$$

Сравнявайки (7.17) и (7.19), виждаме, че изразът e^{-100x} в точното решение е "малък" за x далеч от 0, докато съответният му израз $(1 - 100h)^{n+1}$ в решението на диференчното уравнение по-принцип не е. Наистина, ако $h \geq 0.1$, то $e^{-100x} \leq 0.45 \times 10^{-5}$ и е ясно, че $y(x)$ бързо клони към $y = 1$ когато x нараства. Но, ако $h = 0.1$, то $1 = 10 \times h$ и приближеното решение (7.19) за $x = 1$ е $y_{10} = (-9)^{10} + 1$. Като цяло при $|1 - 100h| > 1$ приближението y_n ще нараства с n , като се отдалечава все повече и повече от "вярната" стойност $y = 1$. Оттук получаваме следното ограничение за стъпката. $-1 < 1 - 100h < 1$ или $h < 2/100 = 0.02$.

Горната ситуация е типична за така наречените твърди уравнения. За такива задачи можем да използваме успешно метода на Ойлер само ако грижливо регулираме големината на стъпката. Или, и това е по-доброто решение, можем да сменим метода с по-добър. Една система от твърди уравнения е разгледана в задача 7г.

7.5 МНОГОСТЪПКОВИ МЕТОДИ

Методът на Ойлер и разгледаните досега методи са *едностъпкови* - за да намерят y_{i+1} , те използват само информацията в предишната точка (x_i, y_i) . Когато решението бързо се променя, както при твърдите уравнения, това може да доведе до загуба на точност. Освен това, $f(x, y)$ трябва да се изчислява многократно (особено при методите на Рунге-Кута), а това увеличава броя на машинните операции.

Този проблем може да се разреши като се използват **многостъпкови** методи. Един от тях използва следната

Формула на Адамс-Башфорт:

(7.20)

$$y_{i+1} = y_i + \frac{h}{24}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}),$$

наречена **предиктор** и формулата на **Адамс-Мултън**

$$(7.21) \quad y_{i+1} = y_i + \frac{h}{24}(9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})$$

наречена **коректор**

Тук отново с fk сме означили $f(x_k, y_k)$ - Както може да се забележи (7.20) е *явна* - ако f ни е известна в предходните 4 точки, можем да продължим с изчисляването на (x_{i+1}, y_{i+1}) . Но (7.21) е *неявна* тъй като f_{i+1} съдържа y_{i+1} т.е. уравнението не е решено спрямо y_{i+1} .

Начинът, по който се използват (7.20). (7.21), се нарича **предикторно-коректорна** схема и се описва така

Стъпка 0. Изчисляваме f_0, f_1, f_2, f_3 , като използваме едностъпков метод, напр. Рунге-Кута, за да подготвим използването на (7.20). Полагаме $i = 3$;

Стъпка 1. "Предсказваме" (predict - предсказвам, оттук **предиктор**) решението за x_{i+1} , като намираме приближението y_{i+1}^p от (7.20);

Стъпки 2. "Коригираме" (correct - поправям) приближението за x_{i+1} , използвайки (7.21) така

$$y_{i+1} = y_i + \frac{h}{24}(9f(x_{i+1}, y_{i+1}^p) + 19f_{i-1} + f_{i-2})$$

Полагаме $i = i + 1$ и отиваме на *стъпка 1*.

Забележете, че f_i, f_{i-1}, f_{i-2} вече са изчислени при предишното изпълнение на цикъла и единственото нещо, което се променя в *стъпка 2*, е приближената стойност на $y(x_{i+1})$.

Като страничен продукт от предикторно-коректорната схема получаваме следната оценка за локалната грешка

$$d_i \approx K|y_{i+1}^p - y_{i+1}|, K = \text{const.}$$

В параграф 7.2, подчертахме какъв сериозен проблем е това при едностъпковите методи.

7.6 ЗАДАЧИ

1. Намерете приближено решение по метода на Ойлер със стъпки $h = 0.5, 0.1$ и 0.01 на следните начални задачи в дадения интервал. Нарисувайте резултата.

а) $y' = \sin x + e^{-x}$, $y(0) = 0$, $x \in [0, 1]$;

б)

$$y' = \frac{1}{2}(y + y^2), \quad y(1) = -2, \quad x \in [1, 3];$$

в) $y' = 1 + 0.2y \sin x - 0.25y^2$, $y(0) = 0$, $x \in [0, 10]$, $(0 \leq y \leq 3)$;

г)

$$y' = \frac{2 - x^2 - y^2}{2 + x^2 + xy}, \quad y(0) = 0, \quad x \in [0, 1];$$

д) $y' = 1 + y \sin x - 2y^2$, $y(0) = 0$, $x \in [0, 10]$, $(-1 \leq y \leq 1)$;

е) $y' = (y + \sqrt{x^2 + y^2})/x$, $xy(1) = 0$, $x \in [1, 1.5]$;

ж) $y' = \cos(2x + y) - 1.5(x - y)$, $y(0) = 0$, $x \in [0, 1]$;

з)

$$y' = -\frac{xy}{(1 + x^2)} - 1/(1 + x^2), \quad y(0) = 0, \quad x \in [0, 0.5];$$

и)

$$y' = -\frac{y}{x} - \frac{4}{13}y^2 \ln x, \quad y(1) = 1, \quad x \in [1, 10];$$

2. Приложете метода на Рунге-Кута от четвърти ред към уравненията от предната задача, като използвате същите стъпки. Сравнете графично резултатите с точното решение.

3. За задачата на Коши

$$y' = \frac{2}{x}y + x^2 e^x, \quad y(1) = 0, \quad 1 \leq x \leq 2,$$

имаща точно решение $y(x) = x^2(e^x - e)$:

а) Приложете метода на Ойлер с $h = 0.1$, за да намерите приближено решение и го сравнете с точното решение y :

б) Използвайте метода на Рунге-Кута от четвърти ред, за да направите същите изчисления, както в а);

в) По данните намерени в а) и б) постройте частично линейна функция, с която да апроксимирате $y(1.05)$, $y(1.56)$ и $y(1.98)$. Сравнете с точните стойности.

4. Във верига с променливо напрежение U и съпротивление R , индуктивност L и кондензатор с капацитет C , свързани паралелно, токът i удовлетворява диференциалното уравнение

$$\frac{di}{dt} = C \frac{d^2U}{dt^2} + \frac{1}{R} \frac{dU}{dt} + \frac{1}{L} U.$$

Да предположим, че $C = 0.3$ фарада, $R = 1.4$ ома, $L = 1.7$ хенри и волтажът е даден с

$$U(t) = e^{-0.06\pi t} \sin(2t - \pi).$$

Ако $i(0) = 0$, намерете силата на тока i в моментите от времето $t = 0.1j$, $j = 0, 1, \dots, 100$, като използвате метода на Ойлер.

5. Нека $P(t)$ е броят на индивидите в една популация в момента t , измерван в години. Ако средната (годишна) раждаемост b е постоянна, а средната смъртност d е пропорционална на големината на популацията (което отразява ефекта на пренаселеността) то нарастването на големината на популацията се задава с **логистичното уравнение**

$$\frac{dP(t)}{dt} = bP(t) - k[P(t)]^2,$$

където $d = kP(t)$. Нека $P(0) = 50976$, $b = 2.9 \times 10^{-2}$ и $k = 1.4 \times 10^{-7}$. Намерете големината на популацията след 5 години като използвате някой числен метод.

6. Начертайте решението на системата от пример 2. като използвате други начални условия и стъпки, например $y_1^0 = 15$, $y_2^0 = 22$ и $h = 0.01$.

7. Апроксимирайте решенията на следните системи от диференциални уравнения от първи ред. Използвайте метода на Ойлер с $h = 0.5, 0.1$ и 0.01 . Сравнете с точните решения, където те са известни (за намирането им може да използвате MAPLE).

а) $y_1' = 3y_1 + 2y_2, y_1(0) = 0,$
 $y_2 = 4y_1 + y_2, 2(0) = 1, x \in [0,1]$

Точно решение:

$$y_1(x) = \frac{1}{3}(e^{5x} - e^{-x}) \text{ и } y_2(x) = \frac{1}{3}(e^{5x} - 2e^{-x})$$

б) $y_1' = -4y_1 - 2y_2 + \cos x + 4\sin x, y_1(0) = 0$
 $y_2 = 3y_1 + y_2 + 3\sin x, y_2(0) = -1, x \in [0,2]$

Точно решение:

$$y_1(x) = 2e^{-x} - 2e^{-2x} + \sin x \text{ и } y_2(x) = -3e^{-x} - 2e^{-2x}$$

в) $y_1' = y_2, y_1(0) = 0,$
 $y_2 = 0.07xy_2 - xy_1^2, y_2(0) = 0.125, x \in [0,3]$

г) $y_1' = 998y_1 + 1998y_2, y_1(0) = 0,$
 $y_2 = -999y_1 - 1999y_2, y_2(0) = -1, x \in [0,5]$

Точно решение:

$$y_1(x) = 4e^{-x} - 3e^{-1000x}, y_2(x) = -2e^{-x} + 3e^{-1000x}$$

8. Решете предишната задача, като използвате метода на Рунге-Кута от четвърти ред.

9. Намерете приближени решения на следните диференциални уравнения от втори ред. Използвайте метода на Ойлер с $h = 0.5, 0.1$ и 0.01 . Сравнете с точните решения.

а) $y'' + 2y' + y = e^x, y(0) = 1, y'(0) = -1, x \in [0,2]$

б) $x^2y'' - 2xy' + 2y = x^3 \ln x, y(1) = 1, y'(1) = 0, x \in [1,2]$

Точно решение:

$$y(x) = \frac{7}{4}x + \frac{x^3}{2}\ln x - \frac{3}{4}x^3.$$

10. Да разгледаме екологична система, състояща се от два вида, които се хранят с една и съща храна. Да означим броя на живите индивиди в момента t с $x_1(t)$ и $x_2(t)$. Предполагаме освен това, че коефициентът на раждаемост на индивидите е пропорционален на броя на живите индивиди в този момент. Смъртността зависи от размера на популацията и на двата вида.

Нека популацията на една конкретна двойка видове се описва от системата

$$x_1'(t) = x_1(t)[4 - 0.0003x_1(t) - 0.0004x_2(t)]$$

$$x_2'(t) = x_2(t)[2 - 0.0002x_1(t) - 0.0001x_2(t)]$$

Нека началният брой на индивидите от всеки вид е 10 000. Намерете решението на системата за $t \in [0,4]$, като използвате някой числен метод. Има ли устойчиво решение на този популационен модел?

8

ЛИТЕРАТУРА

- [1] И. Гинчев, К. Ганев, Ч. Маринов, С. Савов, Р. Цанев, *Числени методи-ръководство за лабораторни упражнения*, 1989, ВМЕИ-Варна.
- [2] Б. Сендов, В. Попов, *Числени методи*. Първа и втора част, 1976, "Наука и изкуство", София
- [3] Дж. Форсайт, М. Малкълм, К. Молър, *Компютърни методи за математически пресмятания*. Второ издание, 1986. "Наука и изкуство", София.
- [4] Anton H. and C. Rorres, *Elementary Linear Algebra— Applications Version*, Seventh Edition, 1994, John Wiley & Sons, Inc., New York.
- [5] Burden R. and J. Faires. *Numerical Analysis*, Fourth Edition. 1989, PWS-KENT Publishing Co., Boston.
- [6] Clarke F., Yu. Ledyayev, R. Stern, P. Wolenski *Nonsmooth Analysis and Control Theory*, 1998, Springer-Verlag, New York.
- [7] J. Mathews, *Numerical Methods for Mathematics, Science and Engineering* Second Edition, 1992, Prentice-Hall. Englewood Cliffs