



$$\begin{aligned} \sum_{j=1}^m y_j^{(i)} (a_{1j} - K_{1ji}^{(i)} A_i) &= f_1^{(i)} + \sum_{j=1}^m \sum_{l=1}^{i-1} K_{1ji}^{(l)} y_j^{(l)} A_l, \\ \sum_{j=1}^m y_j^{(i)} (a_{2j} - K_{2ji}^{(i)} A_i) &= f_2^{(i)} + \sum_{j=1}^m \sum_{l=1}^{i-1} K_{2ji}^{(l)} y_j^{(l)} A_l, \\ &\dots\dots\dots, \\ \sum_{j=1}^m y_j^{(i)} (a_{mj} - K_{mji}^{(i)} A_i) &= f_m^{(i)} + \sum_{j=1}^m \sum_{l=1}^{i-1} K_{mji}^{(l)} y_j^{(l)} A_l \end{aligned} \quad (2)$$

или в компактной форме

$$\sum_{j=1}^m y_j^{(i)} (a_{rj} - K_{rji}^{(i)} A_i) = f_r^{(i)} + \sum_{j=1}^m \sum_{l=1}^{i-1} K_{rji}^{(l)} y_j^{(l)} A_i, \quad (3)$$

где  $K_{rji}^{(l)} = K_{rj}(x_i, x_l)$ ,  $y_j^{(l)} = y_j(x_l)$ ,  $f_r^{(i)} = f_r(x_i)$ ,  $x_i$  – фиксированные абсциссы (узлы) отрезка  $[a, b]$ , причем  $x_1 = a$ ,  $x_n = b$ ,  $A_i$  – числовые коэффициенты или весовые множители, обычно  $A_i \geq 0$  и  $\sum_{i=1}^n A_i = b - a$ .

Числовые коэффициенты  $A_i$  зависят от применяемой квадратурной формулы. Задача выбора квадратурной формулы для решения уравнений Вольтерры не является однозначной, причем в литературе нет завершенных, готовых для практики рекомендаций. Причина этого состоит в особенностях задачи вычисления интеграла с переменными границами.

При решении интегральных уравнений на ЭВМ достаточно широко применяются формулы прямоугольников и трапеций, являющиеся формулами замкнутого вида, что объясняется простотой расчетных выражений. Однако при решении систем интегральных уравнений точность и скорость сходимости этих формул может быть недостаточна.

**Комбинации квадратурных формул.** Один из методов повышения точности решения СИУ состоит в применении квадратурных формул высокого порядка точности. Однако их применение для решения уравнений Вольтерры вызывает трудности, поскольку приходится вычислять интегралы на дискретном множестве точек с переменным количеством узлов, а все квадратурные формулы рассчитаны на фиксированное количество узлов. В связи с этим предлагается использование комбинаций квадратурных формул, основанных на совместном использовании некоторой общей формулы с несколькими простыми формулами в конце интегрирования.

На основе многочисленных вычислительных экспериментов предлагаются такие комбинации квадратурных формул:

- 1) формул трапеций и Симпсона;
- 2) формул Симпсона и Ньютона “трех-восьмых”;
- 3) составной 11-точечной формулы Ньютона-Котеса и простых формул Ньютона-Котеса более низкого порядка.

Рассмотрим стратегию применения предлагаемых комбинаций квадратурных формул.

1. Комбинация формул Симпсона и трапеций применяется по такому правилу: при непарном числе узлов используется составная формула Симпсона, при парном числе узлов формула Симпсона дополняется формулой трапеций.

В результате получают формулы вида.

$$\int_{x_1}^{x_i} f(x_i) dx = \begin{cases} \sum_{k=1}^M \left( \frac{h}{3} f(x_{2k-1}) + \frac{4}{3} hf(x_{2k}) + f(x_{2k+1}) \right), \\ M = (i-1)/2, \text{ при } i - \text{непарном;} \\ \sum_{k=1}^M \left( \frac{h}{3} f(x_{2k-1}) + \frac{4}{3} hf(x_{2k}) + f(x_{2k+1}) \right) + \\ + \frac{h}{2} f(x_{i-1}) + \frac{h}{2} f(x_i), M = i/2 - 1, \text{ при } i - \text{парном.} \end{cases} \quad (4)$$

2. Возможны различные схемы применения комбинаций формул Симпсона и Ньютона “трех-восьмых” (расчет осуществляется по четырем точкам). Целесообразен следующий сценарий, при количестве узлов:

$6k$  – применяют формулу “трех-восьмых” на интервале  $[x_1, x_{6k-2}]$  и формулу Симпсона на интервале  $[x_{6k-2}, x_{6k}]$ ;

$6k + 1$  – применяют формулу “трех-восьмых”;

$6k + 2$  – формулу Симпсона на интервале  $[x_1, x_{6k-1}]$  совмещают с формулой “трех-восьмых” на интервале  $[x_{6k-1}, x_{6k+2}]$ ;

$6k + 3$  – формулу “трех-восьмых” на интервале  $[x_1, x_{6k+1}]$  и формулу Симпсона на интервале  $[x_{6k+1}, x_{6k+3}]$ ;

$6k + 4$  – применяют формулу “трех-восьмых”;

$6k + 5$  – используют формулу Симпсона на всем интервале  $[x_1, x_{6k+5}]$ .

В результате получаем формулы:

$$\int_{x_1}^{x_i} f(x_i) dx = \frac{3}{8} \sum_{l=1}^M (h_l f(x_l) + 3h_{l+1} f(x_{l+1}) + 3h_{l+2} f(x_{l+2}) + h_{l+3} f(x_{l+3})) + \\ + \frac{h_{i-2}}{3} f(x_{i-2}) + \frac{4}{3} h_{i-1} f(x_{i-1}) + \frac{h_i}{3} f(x_i), M = (i-3)/3, \text{ при } i = 6k;$$

$$\int_{x_1}^{x_i} f(x_i)dx = \frac{3}{8} \sum_{l=1}^M (h_l f(x_l) + 3h_{l+1}f(x_{l+1}) + 3h_{l+2}f(x_{l+2}) + h_{l+3}f(x_{l+3})),$$

$$M = (i-1)/3, \quad \text{при } i = 6k + 1;$$

$$\int_{x_1}^{x_i} f(x_i)dx = \frac{1}{3} \sum_{l=1}^M (h_l f(x_l) + 4h_{l+1}f(x_{l+1}) + h_{l+2}f(x_{l+2})) + \frac{3}{8} [h_{i-3}f(x_{i-3}) +$$

$$3h_{i-2}f(x_{i-2}) + 3h_{i-1}f(x_{i-1}) + h_i f(x_i)], M = (i-4)/2, \quad \text{при } i = 6k + 2;$$

$$\int_{x_1}^{x_i} f(x_i)dx = \frac{3}{8} \sum_{l=1}^M (h_l f(x_l) + 3h_{l+1}f(x_{l+1}) + 3h_{l+2}f(x_{l+2}) + h_{l+3}f(x_{l+3})) +$$

$$+ \frac{h_{i-2}}{3} f(x_{i-2}) + \frac{4}{3} h_{i-1}f(x_{i-1}) + \frac{h_i}{3} f(x_i), M = (i-3)/3, \quad \text{при } i = 6k + 3;$$

$$\int_{x_1}^{x_i} f(x_i)dx = \frac{3}{8} \sum_{l=1}^M (h_l f(x_l) + 3h_{l+1}f(x_{l+1}) + 3h_{l+2}f(x_{l+2}) + h_{l+3}f(x_{l+3})),$$

$$M = (i-1)/3, \quad \text{при } i = 6k + 4;$$

$$\int_{x_1}^{x_i} f(x_i)dx = \frac{1}{3} \sum_{l=1}^M (h_l f(x_l) + 4h_{l+1}f(x_{l+1}) + h_{l+2}f(x_{l+2})), \quad (5)$$

$$M = (i-1)/2, \quad \text{при } i = 6k + 5.$$

3. Комбинация составной 11-точечной формулы Ньютона-Котеса и простых формул Ньютона-Котеса более низкого порядка получается из нижеизложенного правила.

Возможно составление комбинации формул Ньютона-Котеса по такому правилу: в узлах с  $x_i$  при  $i = m(n-1) + 1$  расчет производить по  $n$ -точечной составной квадратурной формуле Ньютона-Котеса, а в узлах с  $i = m(n-1) + k$ ,  $k = 1, \dots, n-1$  использовать комбинацию  $n$ -точечной составной квадратурной формулы и  $k$ -точечной простой квадратурной формулы (в конце области интегрирования). Так, при  $n = 3m$  получаем комбинацию таких квадратурных формул – трехточечной составной формулы Симпсона и двухточечной формулы трапеций; при  $n = 4m$  – комбинацию 4-х точечной формулы “трех-восьмых”, 3-х точечной формулы Симпсона и 2-х точечной формулы трапеций; и т.д. Соответствующие формулы получаются громоздкими. Однако легко можно найти квадратурные коэффициенты комбинированной формулы, которые вычисляются следующим образом: если  $A_i$ ,  $i = \overline{1, n}$  – квадратурные коэффициенты  $n$ -точечного составного пра-

вила,  $B_j, j = \overline{1, k}$  – квадратурные коэффициенты  $k$ -точечного простого правила, то квадратурными коэффициентами комбинированной формулы будут числа  $C_l, l = \overline{1, n+k-1}$ , где

$$C_l = \begin{cases} A_l, & l < n, \\ A_n + B_1, & l = n, \\ B_{l-n+1}, & l > n. \end{cases} \quad (6)$$

Так можно составить правила для произвольного  $n$ , однако повышение порядка  $n$  не гарантирует повышения точности вычислений.

**Алгоритмы на основе комбинированных формул.** Блок-схема решения системы интегральных уравнений вида (1) представлена на рис. 1.

Внешний модуль **Koef** по количеству узлов дискретизации вычисляет квадратурные коэффициенты соответствующего квадратного правила, основанного на описанных выше комбинированных формулах, и возвращает их в виде вектора  $A$  в основную программу. Внешний модуль **СЛАУ** решает систему линейных алгебраических уравнений с левой частью  $L$  и правой частью  $R$ , рассчитанных и переданных из основной программы, и возвращает решение в виде вектора  $u$  основной программе.

Команды **size** и **zeros** являются командами Matlab. Команда **size** возвращает длину вектора. Команда **zeros** служит для выделения памяти под массив, размерность которого указывается в аргументах команды.

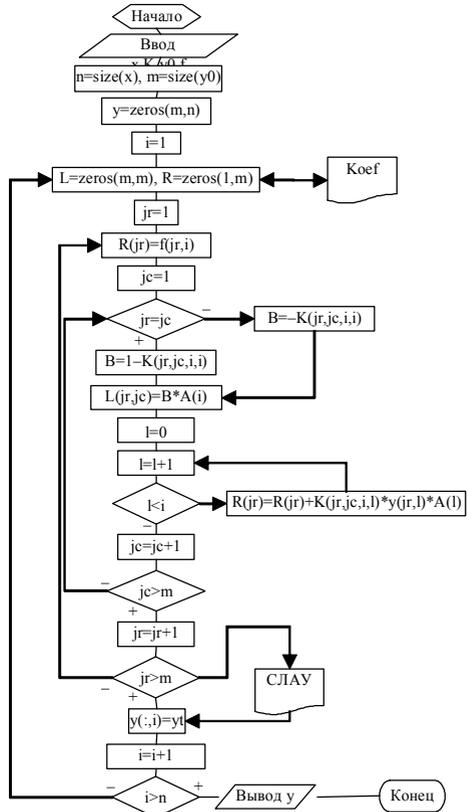


Рис. 1. Блок-схема решения системы интегральных уравнений вида (1) с помощью метода квадратур

**Модельний приклад.** Описані алгоритми досліджувалися при розв'язанні модельного прикладу. Розглянемо систему інтегральних рівнянь Вольтерри другого роду виду

$$\left\{ \begin{aligned} y_1(x) - \int_0^x (x-s)y_1(s)ds - \int_0^x (x+s)y_2(s)ds &= \\ &= 2(1-x)\sin x - \cos x - x + 1; \\ y_2(x) - \int_0^x (x-2s)y_1(s)ds - \int_0^x (2x-s)y_2(s)ds &= \\ &= (2-x)\sin x + (2-x)\cos x - x - 1; \end{aligned} \right. \quad (7)$$

її точне рішення

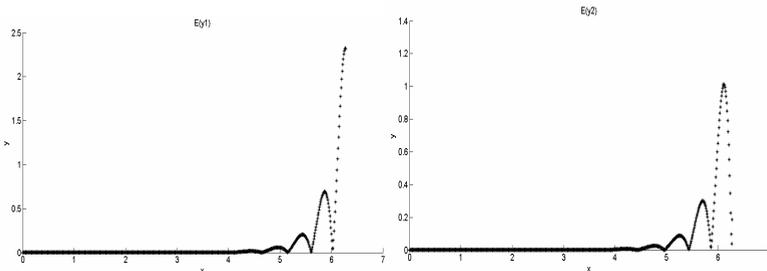
$$y_1^T(x) = \sin x, y_2^T(x) = \cos x. \quad (8)$$

Задамо наступні початкові дані: крок  $h = 0,01$ , кінці відрізка інтегрування  $a = 0, b = 2\pi$ . В даному прикладі відносна похибка формули трапецій досягає 230% (в точці  $x = 2\pi$ ).

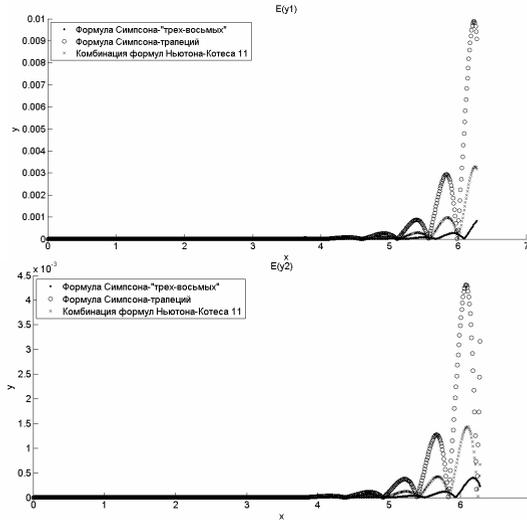
Абсолютна похибка рішення системи (7) методом трапецій представлена на *рис. 2*.

Абсолютна похибка рішення системи (7) методом квадратур з використанням комбінованих квадратурних формул (4), (5) і з допомогою комбінації складової 11-точкової формули Ньютона-Котеса і простих формул Ньютона-Котеса більш низького порядку при тих же початкових даних представлена на *рис. 3*.

Як видно з графіків, запропоновані комбінації квадратурних формул показали при розв'язанні модельного прикладу результати, значно перевищуючі по точності точність формули трапецій. Найкращу точність дали формули на основі комбінації формул Симпсона і "трьох-восьмих". Їх відносна похибка в даному прикладі не перевищує 0,085% (в точці  $x = 2\pi$ ).



*Рис. 2. Абсолютна похибка рішення системи (7) методом квадратур з використанням формули трапецій*



**Рис. 3.** Абсолютная погрешность решения системы (7) методом квадратур с использованием комбинированных формул

**Выводы.** Метод квадратур применен для решения систем интегральных уравнений Вольтерры. Предложены и реализованы алгоритмы, основанные на применении комбинаций квадратурных формул Ньютона-Котеса. Алгоритмы обеспечивают достаточно высокую точность результатов решения.

### Список использованной литературы:

1. Верлань А. Ф., Сизиков В. С. Интегральные уравнения: методы, алгоритмы, программы. – К.: Наукова думка, 1986. – 542 с.
2. Горошко И. О., Тихоход В. А. Компьютерная реализация решения систем интегральных уравнений Вольтерры при исследовании многосвязных динамических объектов // Электронное моделирование. – 2007. – Т. 29. – №3. – С.101-105.

Combined quadrature algorithms for the systems of Volterra integral equations solving in multiply connected dynamic systems simulation are proposed. The algorithms are implemented in the Matlab system.

**Key words:** multicoherent dynamic system, integral model, method of kvadratuv, quadrature formulas, combined quadrature algorithms.

Отримано: 05.06.2008