



Soviet-era science, translated into English

F. M. FILLER

1965

SovietRxiv

View the original and related papers at <https://sovietrxiv.org/items/ru-196501.95242>

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.

Abstract

Full Text

F. M. FILLER

ON THE NUMERICAL SOLUTION OF FREDHOLM INTEGRAL EQUATIONS OF THE SECOND KIND

(Presented by Academician I. G. Petrovskii on 21 X 1964)

MATHEMATICS

1°. A new method is proposed for reducing the numerical solution of a Fredholm integral equation of the second kind

$$f(x) + \int_a^b K(x, y)f(y) dy = \varphi(x) \quad (1)$$

to the solution of a system of linear algebraic equations. In comparison with the known methods, it has an advantage in speed and, when working on a computer, in the amount of memory required.

2°. Usually equation (1) is replaced by a system of linear equations

$$f(x_i) + \sum_{j=1}^n A_j K(x_i, y_j)f(y_j) = \varphi(x_i), \quad i = 1, 2, \dots, n. \quad (2)$$

Here the integral with respect to y is replaced by a sum in which y_j are the nodes and A_j the weights of some quadrature formula. We shall henceforth denote such a formula by $\{y_j, A_j, n\}$. The points x_i at which the function $f(x)$ is sought coincide with the integration points y_j in the quadrature formula. The number of subdivision points of the interval $[a, b]$ required in this case to attain a prescribed accuracy is dictated both by the smoothness of the function $f(x)$ and by the smoothness of the kernel $K(x, y)$. In practice, many cases occur in which the function $f(x)$ is substantially smoother than the kernel $K(x, y)$, which, for example, may have singularities. In this situation, if n points are necessary to describe the function $f(x)$ with the prescribed accuracy, while k points are required to describe $K(x, y)$, then one has to solve a system of linear equations of order $N = \max(n, k)$. To solve such a system, cN^3 operations are necessary, and when this problem is solved on a computer, N^2 memory cells are required to store the matrix.

3°. In the proposed method, the number n of points x_i , and correspondingly the number of equations, does not depend on the numbers k_i of integration points with respect to y for different x_i . Let n points x_1, x_2, \dots, x_n be chosen on the interval $[a, b]$ of the x -axis. For each x_i , replace the integral in equation (1) by an integral sum according to some quadrature formula, and for different x_i the quadrature formulas may be different. We replace equation (1) by the system of linear equations

$$f(x_i) + \sum_{j=1}^{k_i} A_{ij} K(x_i, y_{ij}) \tilde{f}(y_{ij}) = \varphi(x_i). \quad (3)$$

The values $\tilde{f}(y_{ij})$ entering the left-hand side will be expressed in terms of $f(x_i)$ by some interpolation formula*

$$f(\xi) = \sum_{s=1}^n C_s(\xi) f(x_s). \quad (4)$$

* Generally speaking, the interpolation formulas could be taken different for different x_i , but in fact we do not do this.

and substitute into system (3). Then we obtain the system of equations

$$f(x_i) + \sum_{j=1}^{k_i} A_{ij} K(x_i, y_{ij}) \sum_{s=1}^n C_s(y_{ij}) f(x_s) = \varphi(x_i), \quad (5)$$

the solution of which gives $f(x_1), f(x_2), \dots, f(x_n)$.

4°. The following algorithm is proposed for finding the points x_1, x_2, \dots, x_n and the corresponding quadrature formulas $\{y_{ij}, A_{ij}; k_i\}$, necessary for attaining a prescribed accuracy.

Let some $\varepsilon > 0$ be given. We choose initial points x_1, x_2, \dots, x_n . If there is not sufficient information about the function sought, then it is reasonable to choose these points equally spaced and few in number, for example 3: $x_1 = a$, $x_2 = (a + b)/2$, $x_3 = b$. If there is no special information concerning the kernel, then for the selected points x_i we take identical quadrature formulas (as always, it is advantageous to take a sufficiently accurate quadrature formula, for example Gauss' formula). We choose some interpolation method (4), for example quadratic interpolation, and form system (5). Solving this system, we obtain initial values of the function sought

$$f^0(x_1), f^0(x_2), \dots, f^0(x_n). \quad (6)$$

Then for each x_i we compute $\int_a^b K(x_i, y) \tilde{f}(y) dy$ with accuracy $\varepsilon/2$, applying the quadrature formula $\{y_{ij}, A_{ij}; k_i\}$ required for this purpose, its own for each x_i . Using these quadrature formulas, we again form system (5) and check whether it is satisfied, to the prescribed accuracy, by the solution (6). If it is, then we do not solve system (5), but as its solution

$$f(x_1), f(x_2), \dots, f(x_n) \quad (7)$$

we take the values $f^0(x_i)$. If, however, the numbers $f^0(x_i)$ do not satisfy the new system, then we solve it and obtain the refined values (7)*.

5°. After this we proceed to check whether the selected points x_1, x_2, \dots, x_n are sufficient to describe the function sought $f(x)$.

We add to them one more point $\bar{x} = (x_1 + x_2)/2$. We compute $\int_a^b K(\bar{x}, y) \tilde{f}(y) dy$ with accuracy $\varepsilon/2$ and take

$$f(\bar{x}) = \varphi(\bar{x}) - \int_a^b K(\bar{x}, y) \tilde{f}(y) dy.$$

We form a system of $n + 1$ equations for the points $x_1, \bar{x}, x_2, \dots, x_n$ and check whether it is satisfied, to the prescribed accuracy, by the solution

$$f(x_1), f(\bar{x}), f(x_2), \dots, f(x_n). \quad (8)$$

If not, we include \bar{x} among the points x_i for which the function $f(x)$ is sought, and solve the resulting system of equations; after this we again check whether the point $\bar{x} = (x_1 + x_2)/2$ is needed, where the former point \bar{x} now already plays the role of x_2 , and so on.

* In fact, in the program in which this method is implemented, one more check is carried out: from the values (7), for each x_i the integral $\int_a^b K(x_i, y) \tilde{f}(y) dy$ is again computed with the initial accuracy standard $\varepsilon/2$, and using the quadrature formulas needed for this, system (5) is again formed. If it happens that the solution (7) of this system is not satisfied with accuracy ε , then the accuracy standard for computing the integral $\int_a^b K(x_i, y) \tilde{f}(y) du$ is reduced by a factor of four and the process is repeated.

If, however, the solution (8) satisfies the system of equations that has been formed, then we shall not include the point \bar{x} among the points x_i , and shall proceed to check whether it is necessary to include the point $(x_2 + x_3)/2$, and so on, up to the last interval x_{n-1}, x_n . As a result we obtain the final set of points x_1, x_2, \dots, x_n and the corresponding values of the required function

$$f(x_1), f(x_2), \dots, f(x_n), \quad (9)$$

which we shall take as the solution of the given equation with the prescribed accuracy.

6°. From the description of the algorithm it is clear that, after solving the equation, one can obtain the value of the required function at any point $a \leq \xi \leq b$ from the equation itself:

$$f(\xi) = \varphi(\xi) - \int_a^b K(\xi, y) \tilde{f}(y) dy. \quad (10)$$

Solving equation (1) by the proposed method requires cn^3 operations, while in solving the problem on a computer the necessary number of memory cells for storing the matrix of the system is n^2 . In the case when $f(x)$ is a smoother function than $K(x, y)$, this gives savings both in time and in the required memory.

At the Institute of Theoretical and Experimental Physics, the method described has been implemented as program B-19 of the B-61 standard-program library. As initial x_i , this program selects, on the interval $[a, b]$, at the user's request, from 3 to 40 equally spaced values of x ; as the initial quadrature formula, the 5-point Gauss formula is used; interpolation is quadratic. At output, the program, at the user's request, prints the final values x_i and $f(x_i)$, after which it becomes a program for computing the value of a function of one variable, i.e., when called with argument α it returns the value $f(\alpha)$. This is convenient for subsequent work with the function $f(x)$.

7°. In addition, the method described has also been implemented in another variant: a set of points x_i is determined such that, when an additional point $(x_i + x_{i+1})/2$ is added to it, the value of the required function at the old points changes by no more than ε . This requires certain changes in the algorithm. Namely, the question of whether it is necessary to add a new point \bar{x} to the already existing set of points x_i is decided differently.

Suppose the points x_i and the corresponding values $f(x_i)$ are given; denote them by

$$f^{\text{st}}(x_1), f^{\text{st}}(x_2), \dots, f^{\text{st}}(x_n). \quad (11)$$

Add a new point \bar{x} . For x_i and \bar{x} we form the system (5), and denote its solution by

$$f^{\text{new}}(x_1), f^{\text{new}}(x_2), \dots, f^{\text{new}}(x_n). \quad (12)$$

Now we check the fulfillment of the inequality

$$\max_i |f^{\text{new}}(x_i) - f^{\text{st}}(x_i)| \leq \varepsilon. \quad (13)$$

The point \bar{x} is included in the case when this inequality is not satisfied. Such an algorithm corresponds approximately to an estimate of the closeness of the solution in the C metric, whereas the main algorithm corresponds approximately to the L metric. Other accuracy criteria are also possible and easily implementable in a program.

Received
21 I 1964

Note: Figure translations are in progress. See original paper for figures.

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.