Implementation of a Regularization-Based Numerical Method for Solving the First-Kind Volterra Integral Equation Using Python and Numpy

A.O. Ryspaev*[1], M.A. Temirov[2]

[1,2]*Institute of Computer Technologies and Artificial Intelligence, Bishkek, Kyrgyz Republic*
[1,2]*E-mail: ryspaev.amantur@yandex.ru, m.temirov2015@gmail.com*

Abstract

In this paper, the problem of solving the Volterra integral equation of the first kind, which belongs to the class of ill-posed problems, is considered. Based on the method of regularization and quadrature approximation, a stable numerical algorithm was created. A modern software implementation using the Python programming language and the NumPy library is presented, which provides efficient numerical processing and computational stability of the algorithm. The stability, computational complexity, and accuracy of the algorithm have been analyzed. The results of numerical experiments confirm the effectiveness of the proposed method in solving inverse problems of mathematical physics.

Keywords: Volterra equation of the first kind, regularization, Python, NumPy, numerical methods, ill-posed problem, integral equations, inverse problems, computational experiment.
2020 MSC: 45D05, 65R20, 65J20.

1. Introduction

Volterra integral equations of the first kind are widely used in modeling of heat conduction processes, geophysical exploration, signal processing, and solving of inverse problems. A distinctive feature of these equations is their Hadamard's ill-posedness, as a result of which small errors in the initial data can lead to significant deviations in the solution. To ensure the stability of the computational process, regularization methods are applied, which allow transforming the original problem into a stable form. Despite a significant body of research dedicated to first-kind integral equations, the problem of developing efficient computational algorithms remains highly relevant [1, 2, 6].

2. Regularization-numerical algorithm

The purpose of this work is to develop a regularization-numerical algorithm for solving the Volterra integral equation of the first kind and to investigate its stability based on the difference approximations.

Furthermore, a numerical method to solve this equation was developed based on the constructed systematic algorithms, with the resulting difference (grid) schemes being stable

*Corresponding author: ryspaev.amantur@yandex.ru

[4]. Let us consider the Volterra integral equation of the first kind:

$$\int_0^x K(x,t)z(t)dt = f(x), \tag{2.1}$$

where $K(x,t)$ – is the known kernel of the integral operator, $f(x)$ is a given function, and $(z(x))$ represents the solution to be found. Since the problem is ill-posed, a regularization parameter $\varepsilon > 0$ is introduced to construct a stable computational algorithm.
where $K\,|_{t=x} = 0, \quad K_t(t,t) \geq \alpha > 0\,; \quad f(0) = 0$. Consequently, we have:

$$\begin{cases} \int\limits_0^x K_t(t,t)\psi(t)dt + \int\limits_0^x K_0(x,t)\psi(t)dt = f(x), \\ \int\limits_0^x z(t)dt = \psi(x), \quad \psi(0) = 0, \end{cases} \tag{2.2}$$

for simplicity: $z(0) = 0, K_0 \equiv K_t(x,t) - K_t(t,t)$, (if $z(0) = q$, then taking into account $z = V + q$, we will receive $V(0) = 0$. Therefore, without loss of generality, we require $z(0) = 0$).

In the papers [7-8] a systemic algorithm for regularization of the system was developed, and regularizability of (2.2) in a space with a uniform metric was proven, i.e.:
$(\psi_\varepsilon, z_\delta) \xrightarrow[\varepsilon\to 0(\delta\to 0)]{} (\psi, z), \quad (\varepsilon, \delta-$ are small parameters).

Therefore, based on system (2.2) and taking into account the parameters ε, δ, the following numerical solution method is provided in this paper:

$$\begin{cases} \varepsilon\psi_i + \sum\limits_{j=1}^{i} K_{0j}\psi_j h + \sum\limits_{j=1}^{i} N_{ij}\psi_j h = -f_i, \\ \delta z_i + \sum\limits_{j=1}^{i} z_j h = \psi_{\varepsilon i}, \quad i = \overline{1,n}. \end{cases}$$

$$\varepsilon\psi_{i-1} + \sum_{j=1}^{i} K_{0j-1}\psi_{j-1}h + \sum_{j=1}^{i} N_{ij-1}\psi_{j-1}h = \tilde{f}(s_i) \quad (i = \overline{1,n}) \tag{2.3}$$

and denote as $\tilde{\psi}^{\varepsilon,h} = \left\{\tilde{\psi}_{i-1}^{\varepsilon,h}\right\}, \quad (i = \overline{1,n})$ the solution of (2.3), and we introduce the vector $\tilde{\beta}^{\varepsilon,h} = \left\{\tilde{\beta}_{i-1}^{\varepsilon,h}\right\} \equiv \left\{\psi(s_{i-1}) - \tilde{\psi}_{i-1}^{\varepsilon,h}\right\}, \quad i = \overline{1,n.}$ We need the estimation of $\left\|\tilde{\beta}^{\varepsilon,h}\right\|_{C_n}$. For this we have:

$$\varepsilon\psi(s_{i-1}) + \sum_{j=1}^{i} K_{0j-1}\psi(s_{j-1})h + \sum_{j=1}^{i} N_{i\,j-1}\psi(s_{j-1})h = -f_i - \tilde{f}(s_i) - R_i + \varepsilon\psi(s_{i-1}) \tag{2.4}$$

where

$$R_i = \int_0^{x_i} N(x_i,s)\psi(s)ds - h\sum_{j=1}^{i} N_{i,j-1}\psi(s_{j-1}) \tag{2.5}$$

is the remainder of the quadrature of the right rectangles. Therefore, subtracting (2.3) from (2.4), we have:

$$[\varepsilon + hK_{0i\,j-1} + hN_{i,j-1}]\tilde{\beta}^{\varepsilon,h}_{i-1} + h\sum_{j=1}^{i-1} K_{0i\,j-1}\tilde{\beta}^{\varepsilon,h}_{i-1} + h\sum_{j=1}^{i-1} N_{i,j-1}\tilde{\beta}^{\varepsilon,h}_{j-1} =$$

$$= f(s_i) - \tilde{f}(s_i) - R_i + \varepsilon\psi(s_{i-1}). \tag{2.6}$$

Finding the difference between adjacent rows of the triangular system of linear algebraic equations (SLAE) (2.6) and proceeding to the estimate, we obtain:

$$\left|\tilde{\beta}^{\varepsilon,h}_{i-1}\right| \le \frac{\varepsilon}{\varepsilon+hK_1+hK_2}\left|\tilde{\beta}^{\varepsilon,h}_{i-2}\right| + \frac{h^2L_1}{\varepsilon+hK_1+hK_2}\sum_{j=1}^{i-1}\left|\tilde{\beta}^{\varepsilon,h}_{j-1}\right| + \frac{2\tilde{\delta}+|R_i-R_{i-1}|+\varepsilon|\psi(s_{i-1})-\varphi(s_{i-2})|}{\varepsilon+hK_1+hK_2} + \\ + \frac{h^2L_2}{\varepsilon+hK_1+hK_2}\sum_{j=1}^{i-1}\left|\tilde{\beta}^{\varepsilon,h}_{j-1}\right|, \tag{2.7}$$

$L_1 = \max\limits_{[0,b]\times R}|K'_x(x,s)|$, $L_2 = \max\limits_{[0,b]\times R'}|N'_x(x,s)|$, $K_1 = \min\limits_{[0,b]\times R}|K(s,s)|$, $K_2 = \min\limits_{[0,b]\times R}|N(s,s)|$, because $|R_i - R_{i-1}| \le ch^2$, $c = const$ [1].
Since $|\psi(s_{i-1}) - \psi(s_{i-2})| \le \Phi_1 h$, $\Phi_1 = \max|\psi'(s)|$, from (7) the inequality follows:

$$\left|\tilde{\beta}^{\varepsilon,h}_{i-1}\right| \le \frac{\varepsilon}{\varepsilon+hK_1+hK_2}\left|\tilde{\beta}^{\varepsilon,h}_{i-2}\right| + \frac{h^2L_1}{\varepsilon+hK_1+hK_2}\sum_{j=1}^{i-1}\left|\tilde{\beta}^{\varepsilon,h}_{j-1}\right| + \frac{h^2L_2}{\varepsilon+hK_1+hK_2}\sum_{j=1}^{i-1}\left|\tilde{\beta}^{\varepsilon,h}_{j-1}\right| + \\ + \frac{2\tilde{\delta}+Ch^2+\varepsilon h\Phi_1}{\varepsilon+hK_1+hK_2}. \tag{2.8}$$

It follows directly from (2.6) that

$$\left|\tilde{\beta}^{\varepsilon,h}_{0}\right| \le \frac{2\tilde{\delta} + ch^2 + \varepsilon\,|\psi(s_1)|}{\varepsilon + hK_1 + hK_2} \tag{2.9}$$

If $\psi(s_0) = 0$, then

$$\left|\tilde{\beta}^{\varepsilon,h}_{0}\right| \le \frac{2\tilde{\delta} + Ch^2 + \varepsilon h\Phi}{\varepsilon + hK_1 + hK_2}, \quad \Phi = \|\psi(s)\|_C \tag{2.9'}$$

We denote by μ_0 and μ_1 the right sides of inequalities (2.9) and (2.9′) respectively. Let's find an estimate of $\left|\tilde{\beta}^{\varepsilon,h}_{1}\right|$. Putting (2.8) in $i = 2$ we find:
$\left|\tilde{\beta}^{\varepsilon,h}_{1}\right| \le \mu_0\left(1 + \frac{\varepsilon+h^2L_1+h^2L_2}{\varepsilon+hK_1+hK_2}\right)$ if $\psi(s_0) = 0$, and $\left|\tilde{\beta}^{\varepsilon,h}_{1}\right| \le \mu_0 + \frac{\varepsilon+h^2L_1+h^2L_2}{\varepsilon+hK_1+hK_2}\mu_1$, otherwise, since $\psi(t_0) = 0$.
Let us consider the difference equation

$$\mu_i = \frac{\varepsilon + h^2L_1 + h^2L_2}{\varepsilon + hK_1 + hK_2}\mu_{i-1} + \frac{\varepsilon + h^2L_1 + h^2L_2}{\varepsilon + hK_1 + hK_2}\sum_{j=1}^{i-2}\mu_j + \mu_0, \quad i = \overline{1,n},\ \mu_1 = \mu_0. \tag{2.10}$$

It is obvious that

$$\left|\tilde{\beta}^{\varepsilon,h}_{i-1}\right| \le \mu_i, \quad i = \overline{1,n}. \tag{2.11}$$

Finding the difference $\mu_i - \mu_{i-1}$, and reducing (2.10) to the Cauchy problem for a homogeneous difference equation of the second order, we obtain:

$$\mu_i - \left(\frac{1 + \varepsilon + h^2L_1 + h^2L_2}{\varepsilon + hK_1 + hK_2}\right)\mu_{i-1} + \frac{\varepsilon}{\varepsilon + hK_1 + hK_2}\mu_{i-2} = 0, \quad i = \overline{3,n}, \tag{2.12}$$

$$\mu_1 = \mu_0 \quad \mu_2 = \left(1 + \frac{\varepsilon + h^2L_1 + h^2L_2}{\varepsilon + hK_1 + hK_2}\right)\mu_0. \tag{2.13}$$

The solution (2.12), (2.13) is well known and has the form:

$$\mu_i = l_1\lambda_1^{i-1} + l_2\lambda_2^{i-1}, \quad i = \overline{1,n}, \tag{2.14}$$

where λ_1, λ_2 - are the roots of the characteristic equation
$\lambda^2 - \left(1 + \frac{\varepsilon+h^2L_1+h^2L_2}{\varepsilon+hK_1+hK_2}\right)\lambda + \frac{\varepsilon}{\varepsilon+hK_1+hK_2} = 0$, and $l_1 = \frac{\mu_1\lambda_2-\mu_2}{\lambda_2-\lambda_1}, \quad l_2 = \frac{\mu_2-\mu_1\lambda_1}{\lambda_2-\lambda_1}$.
Evaluation gives us the following:

$$\lambda_{1,2} = \frac{2\varepsilon + h(K_1+K_2) \pm [h(K_1+K_2) + h^2(L_1+L_2)\eta(\varepsilon,h)]}{2(\varepsilon + h(K_1+K_2))}, \tag{2.15}$$

here

$$\eta(\varepsilon,h) = \sqrt{1 + 4\varepsilon(L_1+L_2)[(K_1+K_2) + h((L_1+L_2))]}, \tag{2.16}$$

$$l_1 = \frac{\mu_0}{2}\left(1 + \frac{2\varepsilon + h^2(L_1+L_2)}{[h(K_1+K_2) + h^2(L_1+L_2)\eta(\varepsilon,h)]} + \frac{1}{\eta(\varepsilon,h)}\right), \tag{2.17}$$

$$l_2 = \frac{\mu_0}{2}\left(1 - \frac{2\varepsilon + h^2(L_1+L_2)}{[h(K_1+K_2) + h^2(L_1+L_2)\eta(\varepsilon,h)]} - \frac{1}{\eta(\varepsilon,h)}\right). \tag{2.18}$$

Formulas (2.15)-(2.18) completely determine the solution to the problem (2.13), (2.14). Since $\lambda_2 \geq 0 \quad l_2 \leq 0 \quad \forall\, \alpha \geq 0$, from (2.14), we have

$$\mu_i \leq l_1\lambda_1^{i-1}, \quad i = \overline{1,n.} \tag{2.19}$$

It is obvious that $\eta(\varepsilon,h) = 1 + \mathrm{O}(\varepsilon), \quad \forall\, h > 0$, and it follows from (2.15) $\lambda_1 = 1 + \mathrm{O}(h)$, and (2.17) gives $l_1 \leq \mu_0 = const$. Thus from (2.19), we get:
$\mu_i \leq \mu_0 = const,\ i = \overline{1,n}$.
Then taking into account that $\mu_0 = \frac{2\tilde{\delta}+Ch^2+\varepsilon h\Phi}{\varepsilon+h(K_1+K_2)}$, we arrive to inequality $\mu_i \leq \frac{c_1\tilde{\delta}+c_2h^2+c_3\varepsilon h}{\varepsilon+h(K_1+K_2)}$, $i = \overline{1,n}, \quad c_1, c_2, c_3 = const$. By virtue of (11) we obtain

$$\left|\tilde{\beta}_{i-1}^{\varepsilon,h}\right| \leq \frac{c_1\tilde{\delta} + c_2h^2 + c_3\varepsilon h}{\varepsilon + h(K_1+K_2)}, \quad i = \overline{1,n}. \tag{2.20}$$

Thus, the desired error estimate of the regularized framework of the approximate solution of the equation is obtained. We call such dependencies quasi-optimally consistent $h(\tilde{\delta})$ and $\varepsilon(\tilde{\delta})$, which maximize the rate at which the right-hand sides of (2.18) tend to zero when $\tilde{\delta} \to 0$.

It is easy to see that in the case $\psi(0) = 0$: $h_{k.o.s.}(\tilde{\delta}) \approx \tilde{\delta}, \quad \varepsilon_{k.o.s.}(\tilde{\delta}) \approx \tilde{\delta}^{\frac{1}{2}}$, at the same time $\left\|\tilde{\beta}^{\varepsilon_{k.o.s.},h_{k.o.s.}}\right\|_{C_{n\ k.o.s.}} = \mathrm{O}(\tilde{\delta}^{\frac{1}{2}})$

Theorem 2.1. *For ε-regularized framework of an approximate solution of equation (2.1) satisfying SLAE (2.3), if with $\varepsilon \to 0$, the following assessment is valid:* $\left|\psi(s_{i-1}) - \psi_{i-1}^{\varepsilon(\tilde{\delta}),h(\tilde{\delta})}\right| = \mathrm{O}(\tilde{\delta}^{\frac{1}{2}}),\ i = \overline{1,n}$.

Let's consider the equation $\delta z_i + \sum\limits_{j=1}^{i} z_i h = \psi_{\varepsilon i}$, and it is given that $z(0) = 0$ and $z(t)$ - the exact solution of equation (1), then the following estimate is valid [2]: $\left\|z(t) - \tilde{z}^{\varepsilon,\delta}(t)\right\|_{C_n} \leq d_1\delta + d_2\frac{\varepsilon}{\delta} \leq \mathrm{O}(\varepsilon^{\frac{1}{2}})$, where $d_1, d_2 = const$, $\delta(\varepsilon) \approx \varepsilon^{\frac{1}{2}}$.

3. Develop a regularization-numerical algorithm

A regularization-numerical algorithm can also be introduced based on the direct system (2.2):

$$\begin{cases} \sum\limits_{j=1}^{i} K_{0j}\psi_j h + \sum\limits_{j=1}^{i} N_{ij}\psi_j h = f_i + K_0, \\ \sum\limits_{j=1}^{i} z_j h = \psi_i, \quad i = \overline{1,n}. \end{cases} \tag{3.1}$$

Let us denote by η_i and R_i, respectively, the error vector of the method and the error of the formula of right rectangles on the segment $[0, x_i)$ $\eta_i = \psi_i - \psi_{\varepsilon,i}$,
$R_i = \int\limits_0^{x_i} N(x_{i,j}, s)\psi(s)ds - h\sum\limits_{j=1}^{i} N_{i,j}\psi_j$. By applying the right rectangle formula and introducing the substitution $\eta_i = \psi_i^T - \psi_i$, we get:
$h\sum\limits_{j=1}^{i}(K_{0i,j}\eta_j + N_{ij}(\eta_i + \psi_i^T)) = R_i^* - \sum\limits_{j=1}^{n}[K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j})]$,
where $\eta_0 = z_{\varepsilon,\eta} - z(0) = z_{\xi,\eta}$, $R_i^* = R_i + R_i'$, $R_i' = \int\limits_0^{x_0} K^0(x_i, s)\psi(s)ds - h\sum\limits_{i=1}^{n} K_{ij}^0\psi_i$.
The following estimates take place:
$|\eta_i| \le \frac{h}{h|K_{0i,j}|}\sum\limits_{j=1}^{i-1}[|K_{0i,j} - K_{0i-1,j}|\,|\eta_j| + \left|N_{i,j}(\eta_j + \psi_j^T) - N_{i-1,j}(\eta_j + y_j^T) + N_{i-1,j}(y_j^T)\right|] +$
$+\frac{1}{h|K_{0i,j}|}[\left|R_i^* - R_{i-1}^*\right| + h\sum\limits_{j=1}^{n}\left|K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j}) - K_{i-1,j}^0(z_j) + K_{i-1,j}^0(z_{\varepsilon,j})\right|]$.
Let $L_m = \max\limits_{[0,b]\times R'}\left|\frac{\partial^2}{\partial x\partial s}N(x,s)\right|$, $L_k = \max\limits_{[0,b]}|K_x(x,s)|$.
Then we get $|\eta_i| \le \frac{L_k+L_m}{|K_{i,i}|}h\sum\limits_{j=1}^{n}|\eta_j| + \frac{L_k+L_m}{|K_{i,i}|}h\sum\limits_{j=1}^{n}\left|K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j})\right| + \frac{1}{h|K_{i,i}|}\left|R_i^* - R_{i-1}^*\right|$.
Assessing: $\left|R_i^* - R_{i-1}^*\right| \le \frac{h^2 b}{2}\max\limits_{[0,b]\times R'}|(N_x'(x,s)\psi(s))_s'| + \frac{h^2}{2}\max\limits_{[0,b]\times R'}|(N(x,s)\psi(s))_s'| \le ch^2$
we have: $|\eta_i| \le c_1 h\sum\limits_{j=1}^{i-1}|\eta_i| + c_1 h\sum\limits_{j=1}^{n}\left|K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j})\right| + c_2 h \le c_1 h\sum\limits_{j=1}^{i-1}|\eta_i| + Ph$,
where $c = \frac{h^2 b}{2}\max\limits_{[0,b]\times R'}|(N_x'(x,s)\psi(s))_s'| + \max\limits_{[0,b]\times R'}|N(x,s)\psi(s))_s'|$ $c_2 = \frac{c}{|K_{i,j}|}$, $c_1 = \frac{L_k+L_m}{|K_{i,i}|}$,
$p = c_1\sum\limits_{j=1}^{n}\left|K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j})\right| + c_2$, $0 < c_2 = const$.
Arising in the study of numerical methods, they are c, m and i, usually interconnected. More precisely $c = c_1 h^\lambda$, $m = m_1 h$, $ih \le d$, where h is a grid step;c_1, λ, m_1, d - are independent on h constants; $u_i \le c_1^* h^\lambda (1+h)^{dh-1} \le c_1^* h^\lambda e^{m_1 d}$, u_i - the error modulus of the grid solution at the i-th grid node, we obtain:$|\eta_i| \le hPe^{c_1 b}$ where $c_1^* = P = c_1 b\sum\limits_{j=1}^{n}\left|K_{i,j}^0(z_j) - K_{i,j}^0(z_{\varepsilon,j})\right| + c_2, \lambda = 1$, $m_1 = c_1$, $d = b$. The convergence of order h, of this quadrature method has been proved.

4. Main Results

Implementing the Algorithm with NumPy.The NumPy library provides efficient work with arrays and matrices, significantly reducing computation time compared to traditional

loop implementations. The algorithm's code is provided below.

```
import numpy as np

def solve_volterra(n=100,
epsilon=0.005):

h = 1.0 / n
x = np.linspace(0, 1, n)
z = np.zeros(n)

def kernel(xi, tj):
return 1 + xi - tj

def f(x):
return x**2 / 2 + x**3 / 6 + epsilon*x

for i in range(n):
s = 0.0

for j in range(i):
s += kernel(x[i], x[j]) * z[j]

z[i] = (f(x[i]) - h*s) /
(epsilon + h*kernel(x[i], x[i]))

return x, z
```

The algorithm implements the proposed regularized scheme and calculates an approximate solution at the grid nodes.
A model solution was used to test the algorithm $z(x) = 1$.
The calculations were performed with different values of grid parameters and regularization.

n	h	Error
33	0.0303	0.0410
66	0.0151	0.0204
128	0.0078	0.0098
256	0.0039	0.0047

Table 1 – Dependence of error on the number of nodes

A stable convergence of the numerical solution is observed with decreasing grid step.

5. Conclusion

The obtained results confirm the effectiveness of regularization. A software implementation of the regularization-numerical method for solving the Volterra integral equation of the

ε	Error
10^{-1}	0.092
10^{-2}	0.041
10^{-3}	0.018
10^{-4}	0.011

Table 2 – Effect of the regularization parameter

first kind was developed using Python and the NumPy library. An analysis of the algorithm's stability, convergence, and computational complexity was conducted. The results of computational experiments demonstrated the effectiveness of the proposed approach and its applicability to a wide class of inverse problems in mathematical physics.

6. Acknowledgments

We express our sincere gratitude to the reviewer for their careful review of our manuscript and valuable comments. Your recommendations helped us improve the quality of the article, clarify certain aspects of the study, and enhance the scientific significance of the presented results. Thank you for your time, professional expertise, and constructive suggestions. We highly value your contribution to improving our work and hope that the changes made will be fully incorporated in accordance with your recommendations.

References

[1] A.S. Apartsin, *Non-classical Volterra equations of the first kind: theory and numerical methods*, Novosibirsk: Nauka. Sib. Publishing House of the Russian Academy of Sciences, (1999).

[2] A.M. Denisov, *On an approximate solution of the Volterra equation of the first kind, associated with an inverse problem for the heat equation*, Vest. Moscow State University. Comput. math. and cybern, **(3)** (1980), 49–52.

[3] V.I. Dmitriev, *Inverse problems of electromagnetic methods of geophysics*, Incorrect problems of natural science, (1987), 54–76.

[4] S.K. Godunov, V.S. Rebyansky, Difference schemes, Moscow, Nauka, (1977).

[5] T.D. Omurov, A.O. Ryspaev, *Inverse problems of Bohn-Mahony type in an unbounded domain*, Studies of Integro-Differential Equations of the National Academy of Sciences of the Kyrgyz Republic, (2009).

[6] T.D. Omurov, T.T. Karakeev, *Regularization and numerical methods for solving inverse and nonlocal boundary value problems*. Vest. Kyrgyz National University, (2006).

[7] T.D. Omurov, A.O. Ryspaev, *Multidimensional inverse problem with Goursat type conditions*, Differential Equations and Control Processes, **(4)** (2016).

[8] A.O. Ryspaev, *Regularization method for two-dimensional Volterra-Fredholm integral equations of the first kind*, Problems of Modern Science Education, **8**(40) (2016), 6–9.