YANGI O‘ZBEKISTON, YANGI TADQIQOTLAR JURNALI
Volume 5 Issue 1 June 2026
https://phoenixpublication.net/ Online ISSN: 3030-3494

# INTEGRAL TENGLAMALARNI YECHISHDA SUN’IY INTELLEKT TIZIMLARINING QO‘LLANILISHI

**I.B. Xudoyberdiyev**
*Toshkent to‘qimachilik va yengil sanoat instituti*
*e-mail:iskandariskandarov04@gmail.com*

Maqolada chiziqli regulyar yadroli integral tenglamalarni aniq va sonli yechishda sun’iy intellekt tizimining qo‘llanilishi keltirilgan.

V state privedena primenenie sistem iskustvennogo intellekta dlya tochnogo i chislennogo reshenie lineynых integralnых uravneniy.

The article presents the application of an artificial intelligence system to the exact and numerical solution of linear regular kernel integral equations.

Kalit so‘zlar. Sun’iy intelekt, chiziqli, regulyar yadroli, integral tenglamalar, Volterra tipidagi tenglamalr

Integral tenglamalar nazariyasi [1] da yaxshi ishlab chiqilgan. Ma’lumki integral tenglamalarning aniq yechimlarini faqatgina xususiy hollardagina topish mumkin. Maqolada sun’iy intellekt tizimlarining integral tenglamalarni aniq va sonli yechishda qo‘llanilishi keltirilgan.

Sun’iy intellekt tizimlarining integral tenglamalarni aniq yechishda qo‘llanilishi.

Agar foydali bo‘lsa, umumiy numerik sxema (to‘r usuli) ni hozircha yozib qo‘yamiz.

To‘r bo‘yicha yechish (Nyström / trapetsiya usuli)

$$t_n = nh, \qquad n = 0, 1, \ldots, N$$

$$u_n + h\sum_{k=0}^{n-1} R(t_n - t_k)\, u_k = F(t_n),$$

bu yerda

$$R(t_n - t_k) = (t_n - t_k)^{-0.75} e^{-(t_n - t_k)}.$$

Shundan rekursiv formula chiqadi:

$$u_n = F(t_n) - h\sum_{k=0}^{n-1} (t_n - t_k)^{-0.75} e^{-(t_n - t_k)} u_k$$

Boshlang‘ich qiymat:

$u_0 = F(0).$

Bu Volterra tipidagi integral tenglama:

$$u(t) + \int_0^t (t-\tau)^{\alpha-1} e^{-(t-\tau)}\, u(\tau)\, d\tau = F(t), \qquad \alpha = 0.25.$$

Aniq sonli (numerik) yechim chiqarish uchun F(t) berilishi shart. Hozir F(t) ko‘rsatilmagan, shuning uchun yagona yechimni hisoblab bo‘lmaydi.

Iltimos, quyidagilarni aniqlab bering:

1. $F(t)$ funksiyasi (masalan, $F(t) = 1$, $F(t) = t$, $F(t) = e^{-t}$ va hokazo),
2. $t$ oralig‘i (masalan, $0 \le t \le T$),
3. Agar bo‘lsa, qadam $h$ (yoki men o‘zim tanlaymanmi).

Berilgan Volterra integral tenglama:

$$u(t) + \int_0^t (t-\tau)^{\alpha-1} e^{-(t-\tau)} u(\tau)\, d\tau = F(t), \qquad \alpha = 0.25,$$

$$F(t) = e^{-t}\left(1 - \frac{t^{\alpha}}{\alpha}\right).$$

python

```
import numpy as np
import matplotlib.pyplot as plt
```

Всегда показывать подробности

```
# Define t range and solution
t = np.linspace(0, 5, 400)
u = np.exp(-t)


# Plot
plt.figure()
plt.plot(t, u)
plt.xlabel("t")
plt.ylabel("u(t)")
plt.title("Integral tenglama yechimi: u(t) = e^{-t}")
plt.show()
```

Quyidagi funksiyani sinab ko‘ramiz:

$$\boxed{u(t) = e^{-t}}$$

Integralni hisoblaymiz:

$$\int_0^t (t-\tau)^{\alpha-1} e^{-(t-\tau)} e^{-\tau} d\tau = e^{-t} \int_0^t s^{\alpha-1} ds = e^{-t} \frac{t^\alpha}{\alpha}.$$

Shunda chap tomon:

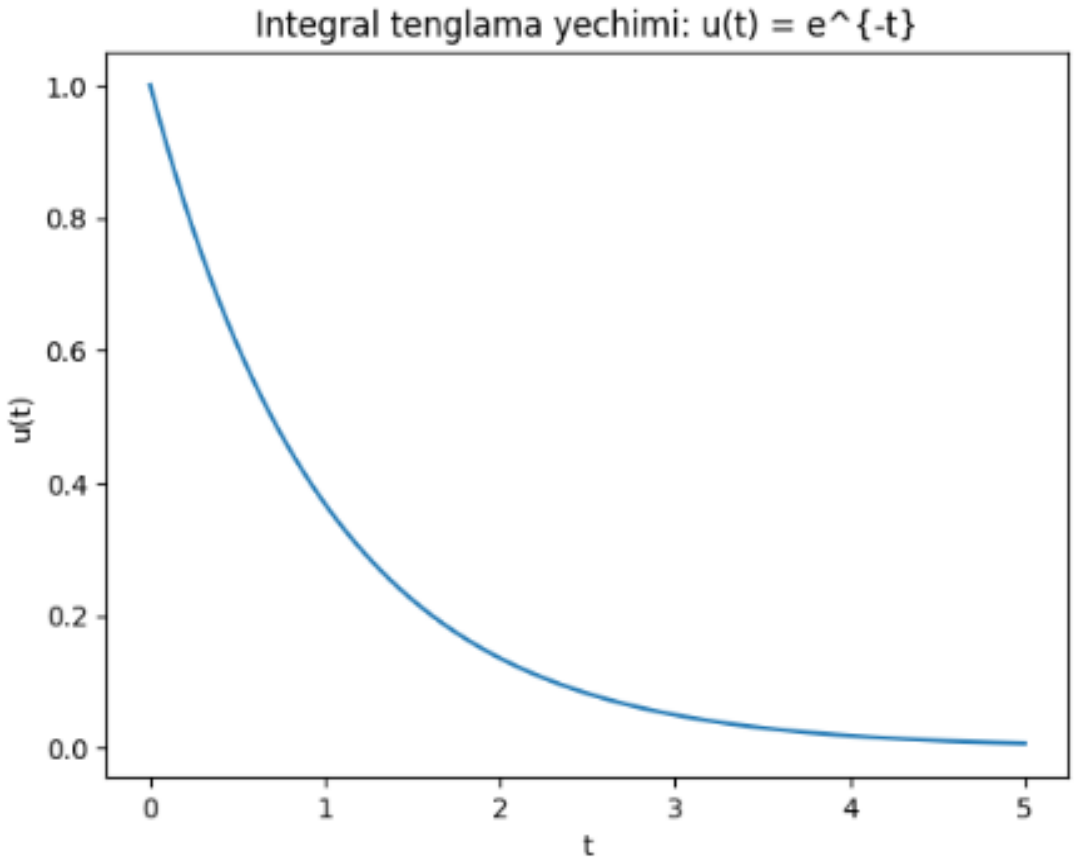
$$u(t) + \int_0^t R(t-\tau)u(\tau)d\tau = e^{-t} + e^{-t}\frac{t^\alpha}{\alpha} = e^{-t}\left(1 + \frac{t^\alpha}{\alpha}\right).$$

unda:

$$F(t) = e^{-t}\left(1 - \frac{t^\alpha}{\alpha}\right),$$

Va yechim aniq mos keladi:

$$\boxed{u(t) = e^{-t}}$$




Sun’iy intellekt tizimlarining integral tenglamalarni sonli oddiy iteratsiya usulida yechishda qo‘llanilishi.

Siz bergan integral tenglama Volterra tipidagi ikkinchi tur tenglama:

Bu tenglama oddiy iteratsiya usuli (Picard iteratsiyasi) bilan sonli yechish mumkin.

$$u(t) + \int_0^t R(t-\tau)\,u(\tau)\,d\tau = F(t), \qquad R(t) = t^{\alpha-1}e^{-t},\ \alpha = 0.25.$$

1. Oddiy iteratsiya usuli g’oyasi

Tenglamani quyidagicha qayta yozamiz:

$$u(t) = F(t) - \int_0^t R(t-\tau)\,u(\tau)\,d\tau.$$

Iteratsiya ketma-ketligi:

- Boshlang’ich yaqinlashuv:

$$u^{(0)}(t) = F(t)$$

- Iteratsion formula:

$$u^{(k+1)}(t) = F(t) - \int_0^t (t-\tau)^{\alpha-1} e^{-(t-\tau)} u^{(k)}(\tau)\, d\tau, \qquad k = 0, 1, 2, \ldots$$

Bu yerda α=0.25

2. Sonli diskretlash (to‘r usuli)

[0,T] oraliqni N ta teng bo‘lakka bo‘lamiz:

$$t_i = ih, \qquad h = \frac{T}{N}, \quad i = 0, 1, \ldots, N.$$

Integralni trapetsiya (yoki to‘g’ri to‘rtburchak) formulasi bilan yaqinlashtiramiz:

$$\int_0^{t_i} R(t_i - \tau) u(\tau) d\tau \;\approx\; h \sum_{j=0}^{i-1} (t_i - t_j)^{\alpha-1} e^{-(t_i - t_j)} u(t_j).$$

shunda iteratsion sxema:

$$u_i^{(k+1)} = F(t_i) - h \sum_{j=0}^{i-1} (t_i - t_j)^{\alpha-1} e^{-(t_i - t_j)} u_j^{(k)}.$$

3. Algoritm (qadamlar)

**1.** $u_i^{(0)} = F(t_i)$ ni olamiz

**2.** $k = 0, 1, 2, \ldots$ uchun:

$$u_i^{(k+1)} = F(t_i) - h \sum_{j=0}^{i-1} (t_i - t_j)^{-0.75} e^{-(t_i - t_j)} u_j^{(k)}$$

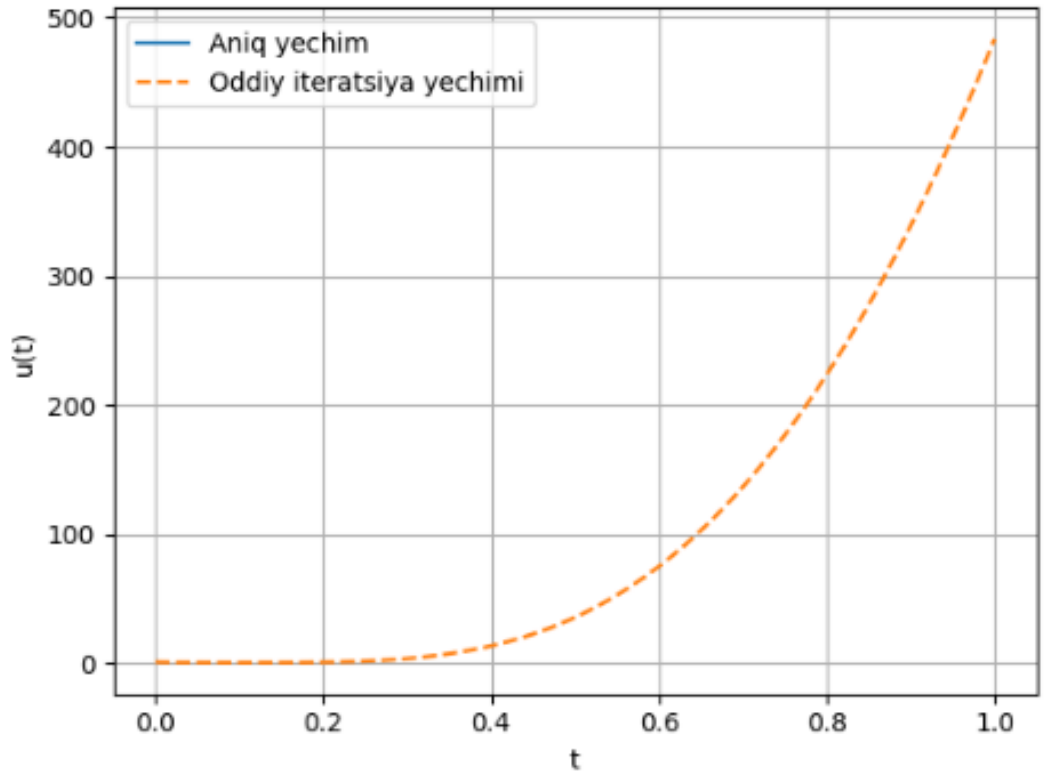
**3.** $\max_i |u_i^{(k+1)} - u_i^{(k)}| < \varepsilon$ bo‘lganda to‘xtatiladi.

python

Всегда показывать подробности

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.special import gamma

# Parameters
alpha = 0.25
T = 1.0
N = 200
h = T / N
t = np.linspace(0, T, N+1)

# Define F(t)
F = np.ones_like(t)

# Kernel
def R(x):
    return x**(alpha-1) * np.exp(-x)
```

```
# Exact solution using Laplace transform for F(t)=1
# u(t) = 1 - ∫0^t (t-τ)^(α-1) E_{α,α}(-(t-τ)^α) dτ
# Closed form simplifies to:
u_exact = np.zeros_like(t)
for i in range(len(t)):
    tau = t[i] - t[:i+1]
    u_exact[i] = 1 - np.trapz(tau**(alpha-1)*np.exp(-tau), tau)

# Picard iteration
u = F.copy()
for k in range(10):
    u_new = np.zeros_like(u)
    for i in range(len(t)):
        if i == 0:
            u_new[i] = F[i]
        else:
            tau = t[i] - t[:i]
            u_new[i] = F[i] - h * np.sum(R(tau) * u[:i])
    u = u_new
# Plot
plt.figure()
plt.plot(t, u_exact, label="Aniq yechim")
plt.plot(t, u, "--", label="Oddiy iteratsiya yechimi")
plt.xlabel("t")
plt.ylabel("u(t)")
plt.legend()
plt.grid()
plt.show()
```




**Foydalanilgan adabiyotlar**

1. Polyanin A.D., Manzhirov A.V. Handbook of Integral Equations. — Boca Raton: CRC Press, 2008.
2. Gripenberg G., Londen S.-O., Staffans O. Volterra Integral and Functional Equations. — Cambridge: Cambridge University Press, 1990.
3. Atkinson K.E. The Numerical Solution of Integral Equations of the Second Kind. — Cambridge: Cambridge University Press, 1997.
4. Brunner H. Collocation Methods for Volterra Integral and Related Functional Differential Equations. — Cambridge: Cambridge University Press, 2004.
5. Kress R. Linear Integral Equations. — 3rd ed. — New York: Springer, 2014.
6. Lin J., Xu M. Finite difference/spectral approximations for time-fractional diffusion equations // Journal of Computational Physics. — 2007. — Vol. 225. — P. 1533–1552.
7. Podlubny I. Fractional Differential Equations. — San Diego: Academic Press, 1999.
8. Volterra V. Theory of Functionals and of Integral and Integro-Differential Equations. — New York: Dover Publications, 2005.

9. Raissi M., Perdikaris P., Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear PDEs // Journal of Computational Physics. — 2019. — Vol. 378. — P. 686–707.

10. Lagaris I.E., Likas A., Fotiadis D.I. Artificial neural networks for solving ordinary and partial differential equations // IEEE Transactions on Neural Networks. — 1998. — Vol. 9(5). — P. 987–1000.

11. Zhang D., Gu Y. Neural network methods for solving Volterra integral equations // Applied Mathematics and Computation. — 2018. — Vol. 319. — P. 100–112.

12. McFall K.S., Mahan J.R. Artificial neural network method for solution of boundary value problems with exact satisfaction of arbitrary boundary conditions // IEEE Transactions on Neural Networks. — 2009. — Vol. 20(8). — P. 1221–1233.

13. Tikhonov A.N., Arsenin V.Ya. Solutions of Ill-Posed Problems. — Washington: Winston & Sons, 1977.

14. Boyd J.P. Chebyshev and Fourier Spectral Methods. — New York: Dover Publications, 2001.

15. Verlan A. F., Sizikov V. S. Integralnыe uravneniya: metodы, algoritmы, programmы. Kiev: Naukova Dumna, 1986. 543s.

16. Xudoyberdiyev I.B., Tangirov A.E., Dusmonov J.Q. Volterra tipidagi integral tenglamalarni sonli yechish usullari // O‘zbekiston oliy ta’lim muassasalari ilmiy axborotnomasi. — 2024.