**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**Informatikos fakultetas**

**Skaitiniai metodai ir algoritmai (P170B115)**

***4 laboratorinis darbas***

17 variantas

**Dėstytojas**:

lekt. Andrius Kriščiūnas

**Darbą atliko**:

IFF – 8/13 Mykolas Paulauskas

KAUNAS, 2020

Turinys

[Užduotis 3](#_Toc58522349)

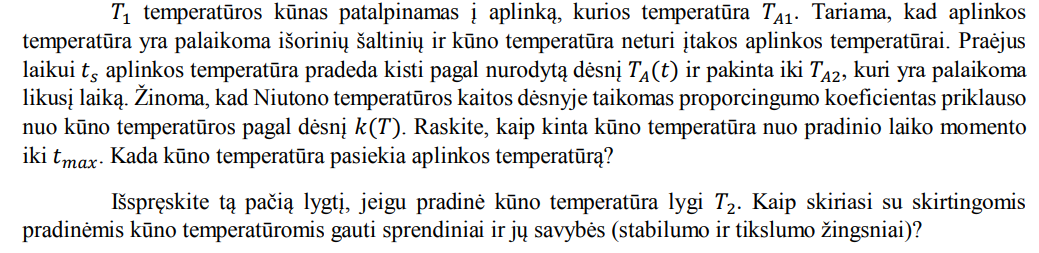
[Diferencialinė lygtis 3](#_Toc58522350)

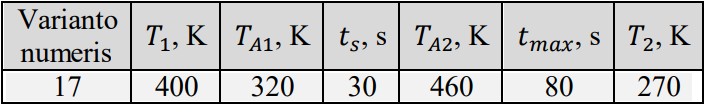
[Eulerio metodas 4](#_Toc58522351)

[IV eilės Rungės ir Kutos metodas 5](#_Toc58522352)

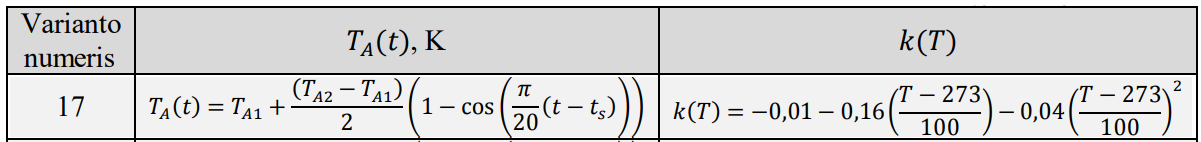
[Rezultatai 7](#_Toc58522353)

# Užduotis





pav. 1 Duomenys



pav. 2 Formulės

# Diferencialinė lygtis

Pagal duotą Niutono temperatūros kaitos dėsnį mes žinome, kad temperatūros kitimo greitis yra proporcingas kūno ir jo aplinkos temperatūrų skirtimui, kadangi greitis iš principo yra išvestinė mes galim susidaryti diferencialinę lygtį : 𝑑𝑇𝑑𝑡 = 𝑘 ∙ (𝑇−𝑇A).

Pirmame etape mes tokią diferencialinę lygtį ir naudosime, tačiau atėjus kambario šildymo etapui, 𝑇A bus skaičiuojamas pagal duotą kambario temperatūros kitimo formulę.

Kode realizuotos funkcijos atrodo taip:

Konstantos skaičiavimas:

def get\_k(T):

    return -0.01 - 0.16 \* (T - 273) / 100 - 0.04 \* ((T - 273) / 100) \*\* 2

Kambario temperatūros didinimo funkcija:

if t >= ts and np.abs(TA - TA2) > margin:

            TA = TA1 + (TA2 - TA1) / 2 \* (1 - np.cos(np.pi / 20 \* (t - ts)))

# Eulerio metodas

def euler\_method(T1, TA1, ts, TA2, tmax, t0, dt):

    print("Euler method calculating...")

    x\_over\_time = [t0]

    y\_subject\_temp = [T1]

    y\_env\_temp = [TA1]

    TA = TA1

    T = T1

    margin = 1e-2

    subject\_env\_temp\_is\_equal = False

    for t in np.arange(t0, tmax, dt):

        # Praėjus duotam laikui temperatūros kitimo dėsnis keičiamas į TA(t)

        if t >= ts and np.abs(TA - TA2) > margin:

            TA = TA1 + (TA2 - TA1) / 2 \* (1 - np.cos(np.pi / 20 \* (t - ts)))

        # Eulerio formule. 18 skaidre

        k = get\_k(T)

        dTdt = k \* (T - TA)

        T = T + dt \* dTdt

        if np.abs(T - TA) < margin and not subject\_env\_temp\_is\_equal:

            print(f"Kūno ir aplinkos temperatūros tapo vienodos praėjus: {t}s, temperatūra = {T}T")

            subject\_env\_temp\_is\_equal = True

            plt.scatter(t, T, color="red", zorder=3, label='Kūno temperatūra pasiekia aplinkos temperatūrą')

        x\_over\_time.append(t)

        y\_subject\_temp.append(T)

        y\_env\_temp.append(TA)

    plt.plot(x\_over\_time, y\_subject\_temp, zorder=1, label='Kūno temperatūra')

    plt.plot(x\_over\_time, y\_env\_temp, zorder=2, label='Aplinkos temperatūra')

    plt.xlabel('Laikas, s')

    plt.ylabel('Temperatūra, K')

    plt.title(f"Uždavinys variantams 16-20, žingsnis: {dt}")

    plt.legend()

    plt.show()

# IV eilės Rungės ir Kutos metodas

def runge\_kutta\_method(T1, TA1, ts, TA2, tmax, t0, dt):

    print("Runge-Kutta method calculating...")

    x\_over\_time = [t0]

    y\_subject\_temp = [T1]

    y\_env\_temp = [TA1]

    TA = TA1

    T = T1

    margin = 1e-2

    subject\_env\_temp\_is\_equal = False

    for t in np.arange(t0, tmax, dt):

        # Praėjus duotam laikui temperatūros kitimo dėsnis keičiamas į TA(t)

        if t >= ts and np.abs(TA - TA2) > margin:

            TA = TA1 + (TA2 - TA1) / 2 \* (1 - np.cos(np.pi / 20 \* (t - ts)))

        k = get\_k(T)

        dTdt = k \* (T - TA)

        f1 = dTdt

        # Pirmas Runge skaiciavimo etapas

        T1 = T + dt / 2 \* f1

        k = get\_k(T1)

        dTdt = k \* (T1 - TA)

        f2 = dTdt

        # Antras Runge skaiciavimo etapas

        T2 = T + dt / 2 \* f2

        k = get\_k(T2)

        dTdt = k \* (T2 - TA)

        f3 = dTdt

        # Trecias Runge skaiciavimo etapas

        T3 = T + dt \* f3

        k = get\_k(T3)

        dTdt = k \* (T3 - TA)

        f4 = dTdt

        # Apjungimas. 30 skaidre

        T = T + dt / 6 \* (f1 + 2 \* f2 + 2 \* f3 + f4)

        if np.abs(T - TA) < margin and not subject\_env\_temp\_is\_equal:

            print(f"Kūno ir aplinkos temperatūros tapo vienodos praėjus: {t}s, temperatūra = {T}T")

            subject\_env\_temp\_is\_equal = True

            plt.scatter(t, T, color="red", zorder=3, label='Kūno temperatūra pasiekia aplinkos temperatūrą')

        x\_over\_time.append(t)

        y\_subject\_temp.append(T)

        y\_env\_temp.append(TA)

    plt.plot(x\_over\_time, y\_subject\_temp, zorder=1, label='Kūno temperatūra')

    plt.plot(x\_over\_time, y\_env\_temp, zorder=2, label='Aplinkos temperatūra')

    plt.xlabel('Laikas, s')

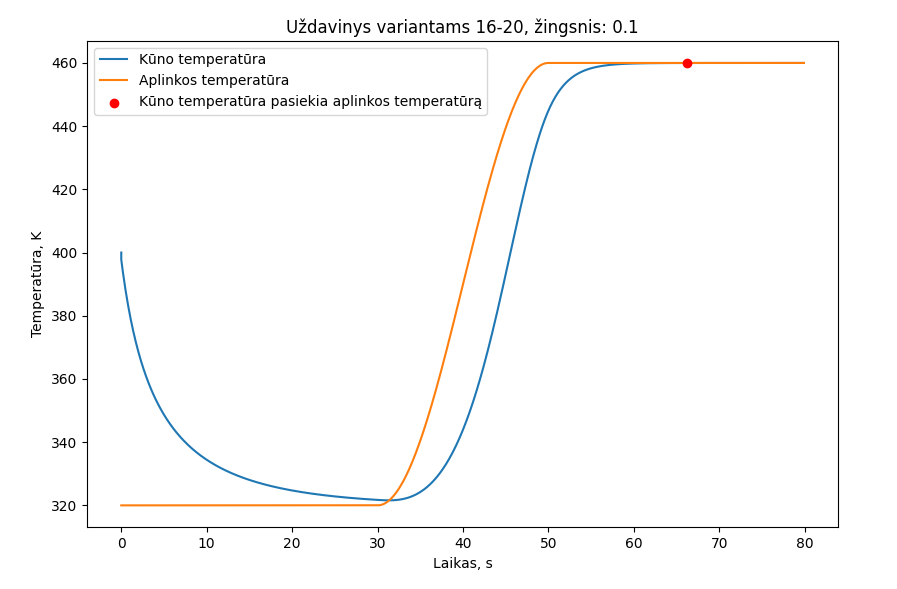
    plt.ylabel('Temperatūra, K')

    plt.title(f"Uždavinys variantams 16-20, žingsnis: {dt}")

    plt.legend()

    plt.show()

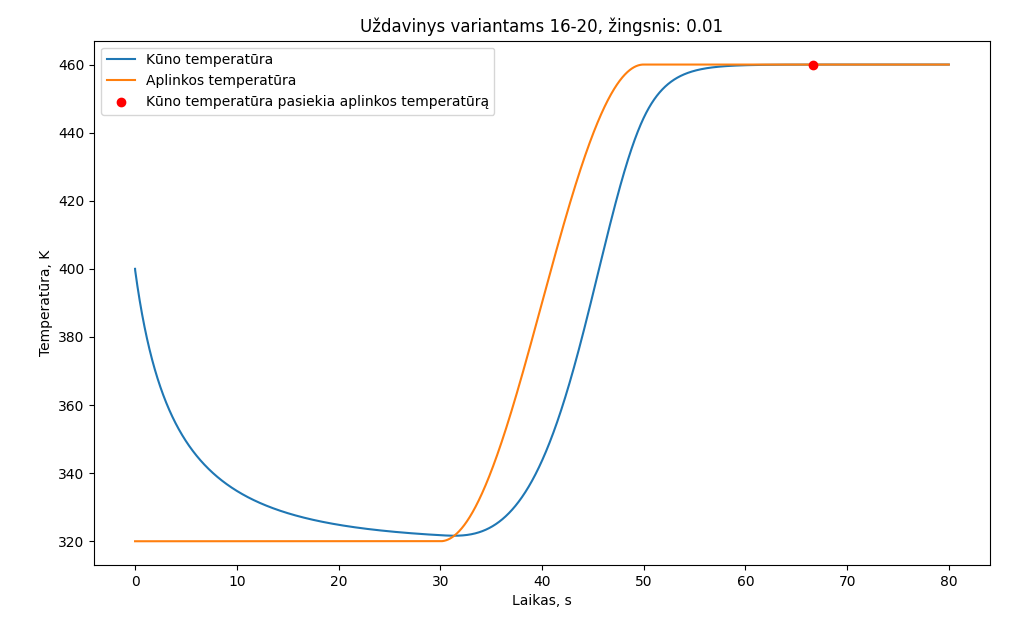
# Rezultatai pirmos dalies



pav. Eulerio metodo rezultatai, kai T = T1 ir žingsnis = 0.1

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 66.2s,

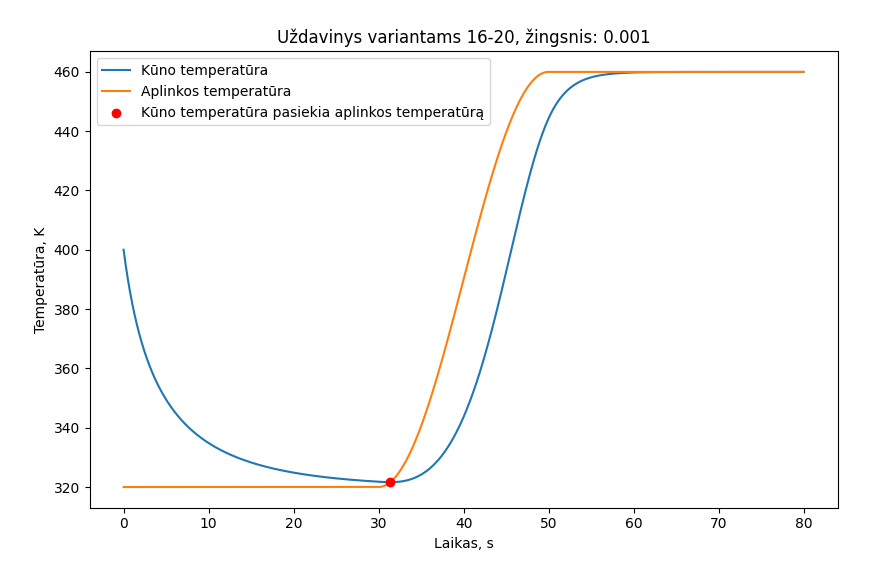
temperatūra = 459.9815221341439T



pav. Eulerio metodo rezultatai, kai T = T1 ir žingsnis = 0.01

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 66.6s,

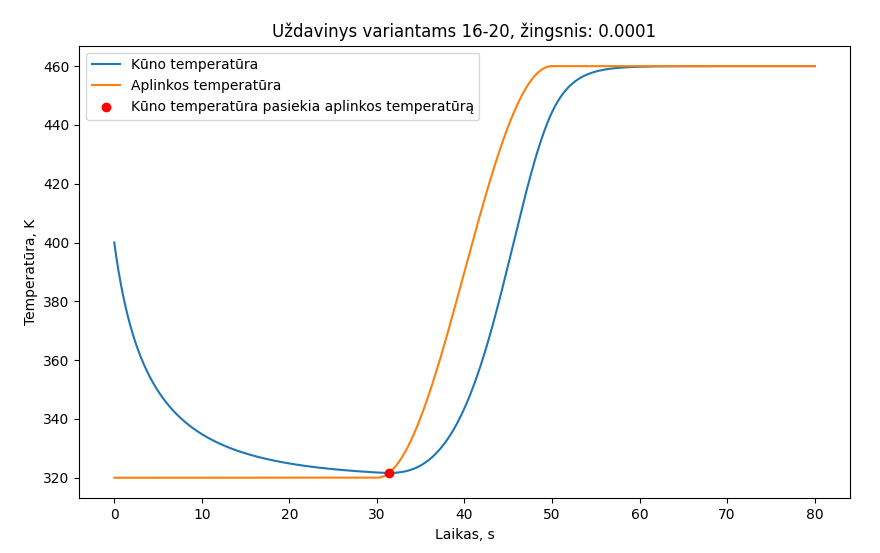
temperatūra = 459.9813943146658T



pav. Eulerio metodo rezultatai, kai T = T1 ir žingsnis = 0.001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 31.373s,

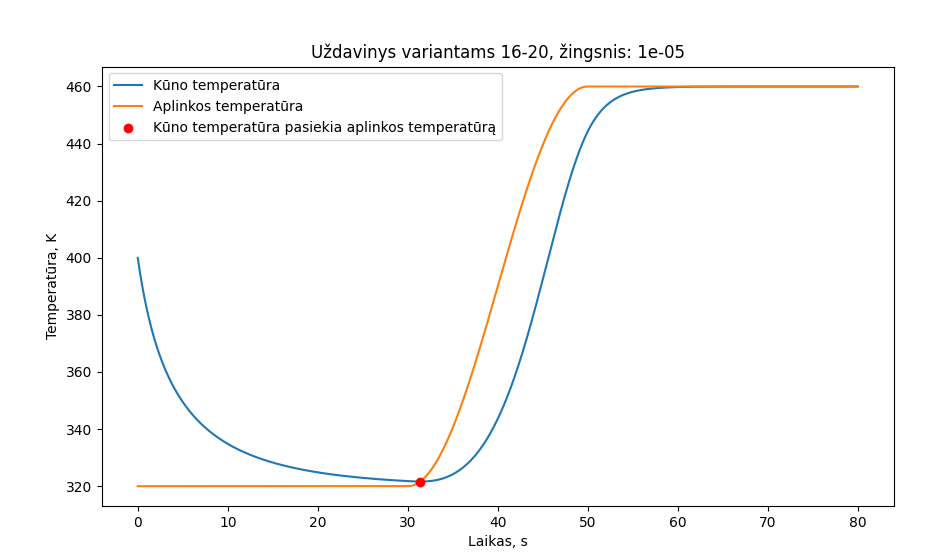
temperatūra = 321.6296871510557T



pav. Eulerio metodo rezultatai, kai T = T1 ir žingsnis = 0.0001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai

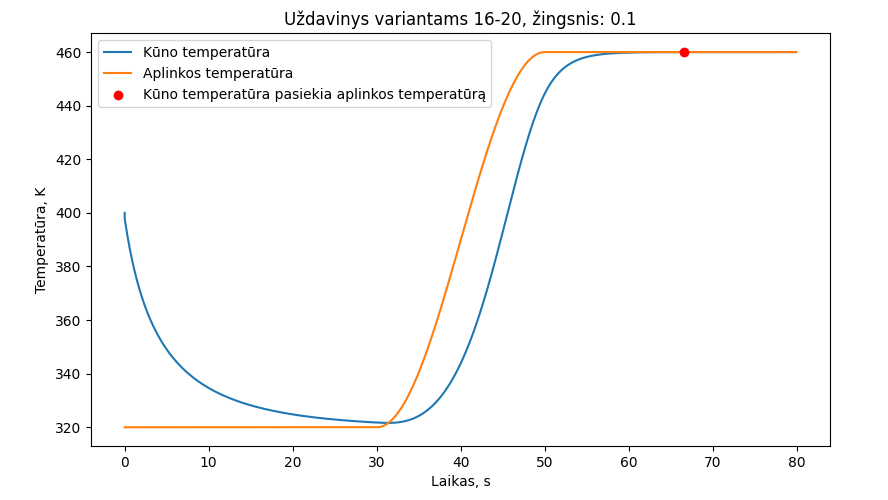
laikas = 31.372400000000003s, temperatūra = 321.6300937796677T



pav. Eulerio metodo rezultatai, kai T = T1 ir žingsnis = 1e-05

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 31.37235s,

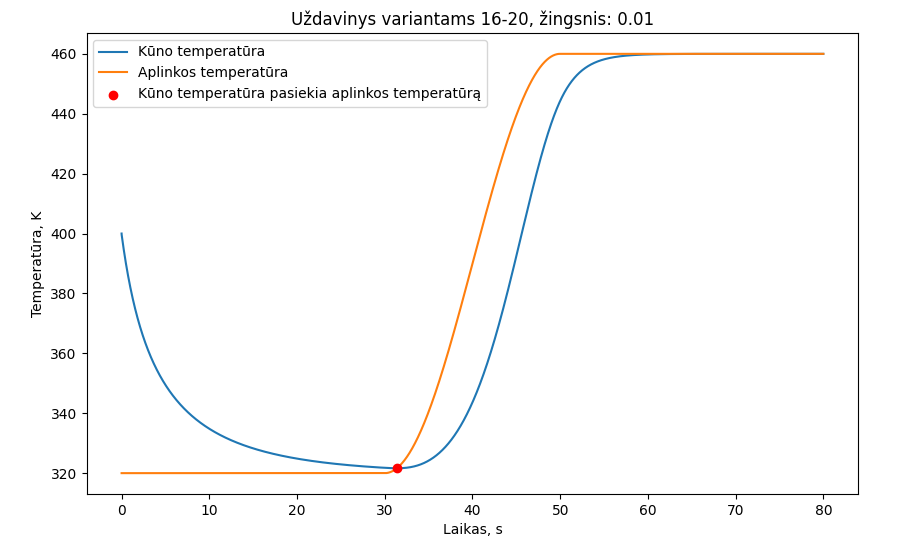
temperatūra = 321.6301344528633T



pav. IV eilės Rungės ir Kutos metodo rezultatai, kai T = T1 ir žingsnis = 0.1

Kūno temperatūra pasiekė aplinkos temperatūrą, kai

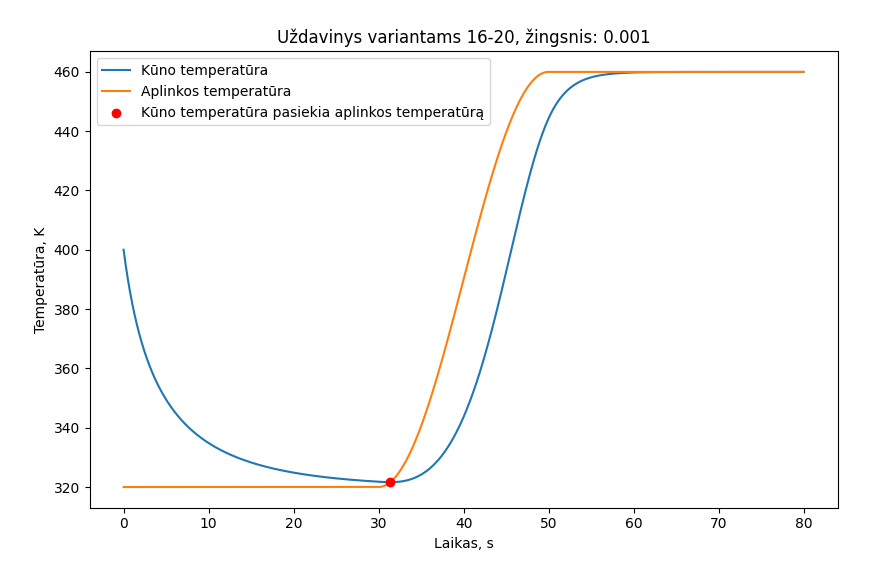
laikas = 66.60000000000001s, temperatūra = 459.98140531911633T



pav. IV eilės Rungės ir Kutos metodo rezultatai, kai T = T1 ir žingsnis = 0.01

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 31.38s,

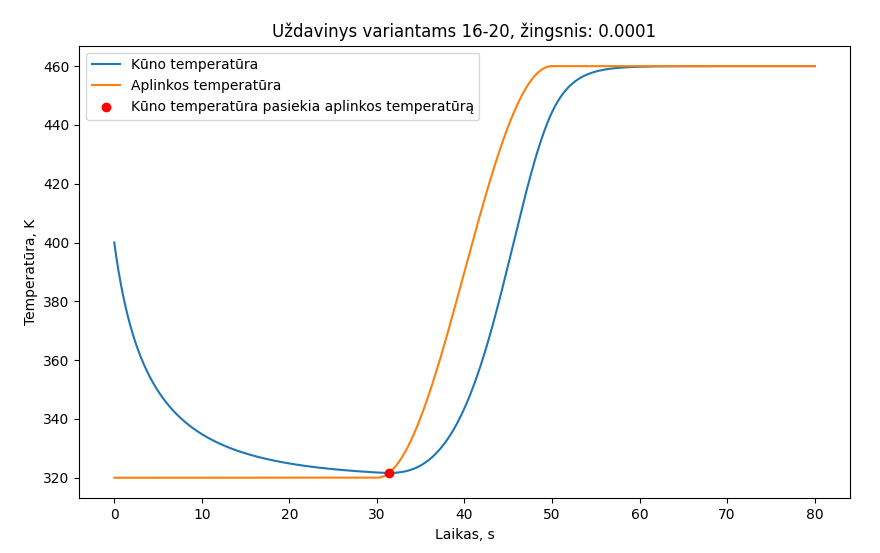
temperatūra = 321.62938631838955T



pav. 10 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T1 ir žingsnis = 0.001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 31.373s,

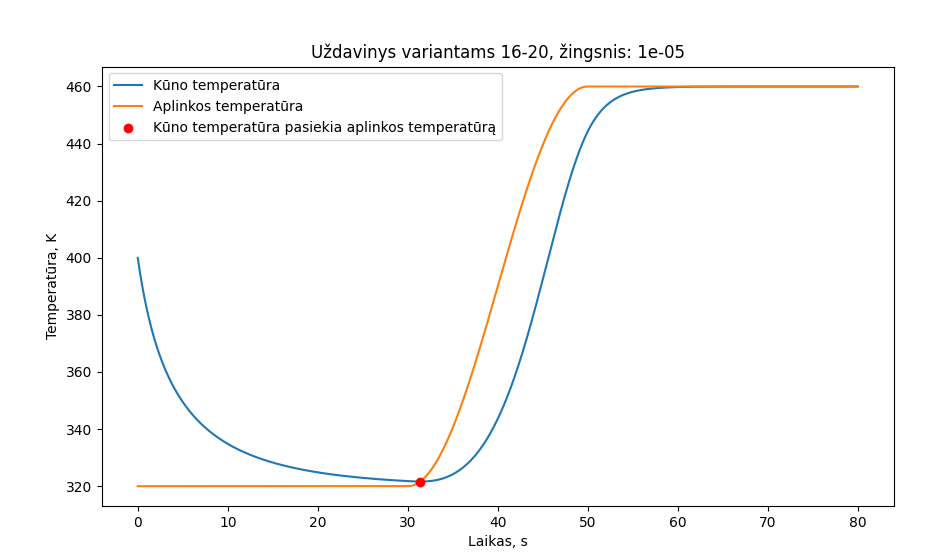
temperatūra = 321.6300621195401T



pav. 11 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T1 ir žingsnis = 0.0001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai

laikas = 31.372400000000003s, temperatūra = 321.63013128270245T



pav. 12 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T1 ir žingsnis = 1e-05

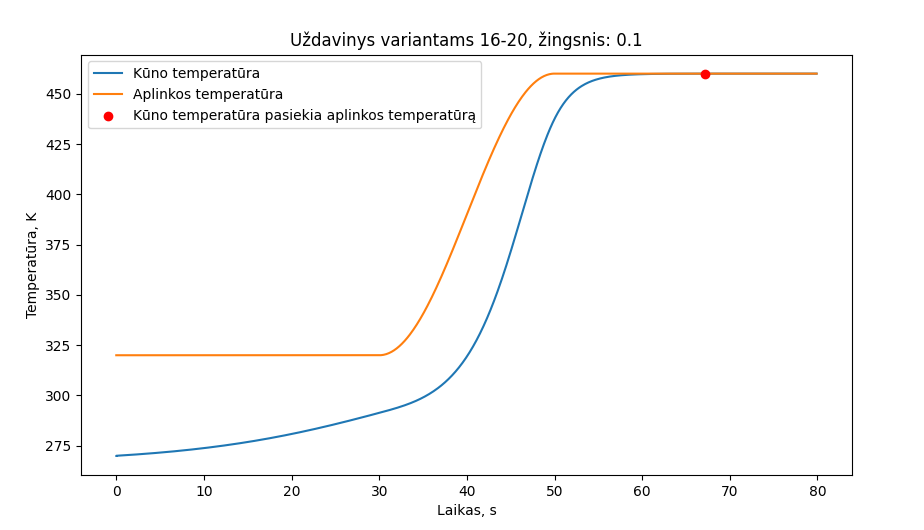
Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 31.37235s,

temperatūra = 321.63013820321015T

## Pirmos dalies išvados

Iš gautų rezultatų matome, kad IV eilės Rungės ir Kutos metodas greičiau nustato kada kūnas pasiekia aplinkos temperatūra ir kad laikas, kada tai įvyksta, yra ~ 31.37 sekundę.

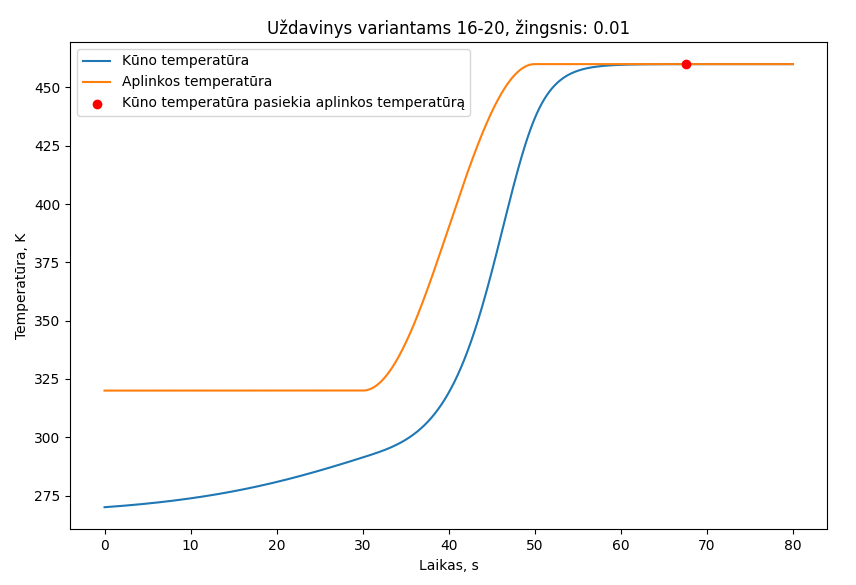
# Rezultatai antros dalies



pav. 13 Eulerio metodo rezultatai, kai T = T2 ir žingsnis = 0.1

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.2s,

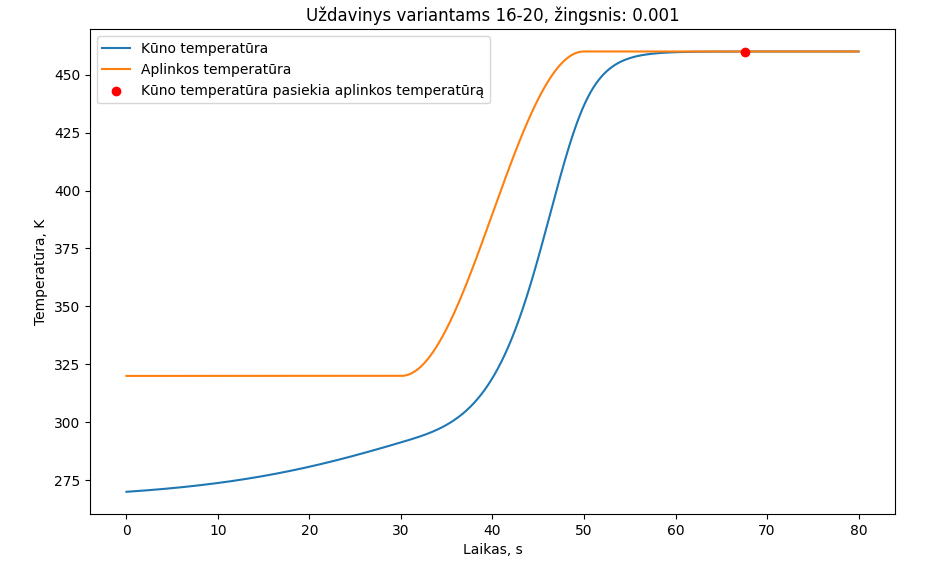
temperatūra = 459.98145812496705T



pav. 14 Eulerio metodo rezultatai, kai T = T2 ir žingsnis = 0.01

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.6s,

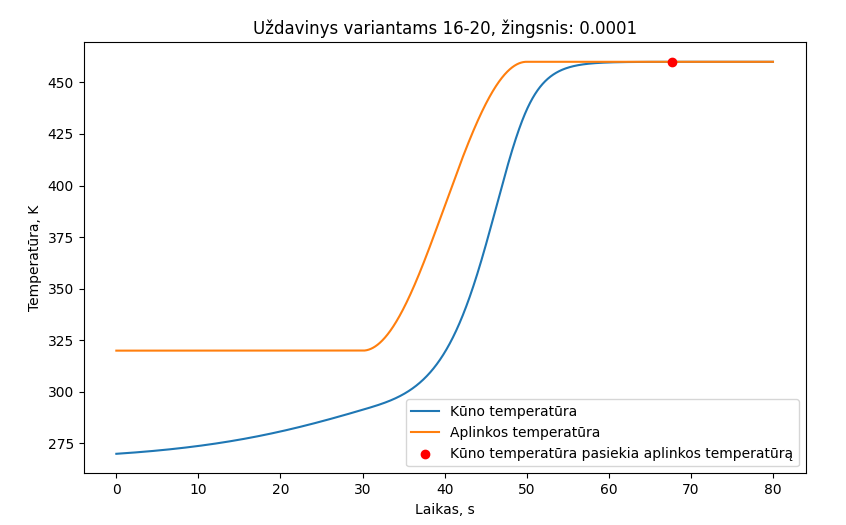
temperatūra = 459.98137535695435T



pav. 15 Eulerio metodo rezultatai, kai T = T2 ir žingsnis = 0.001

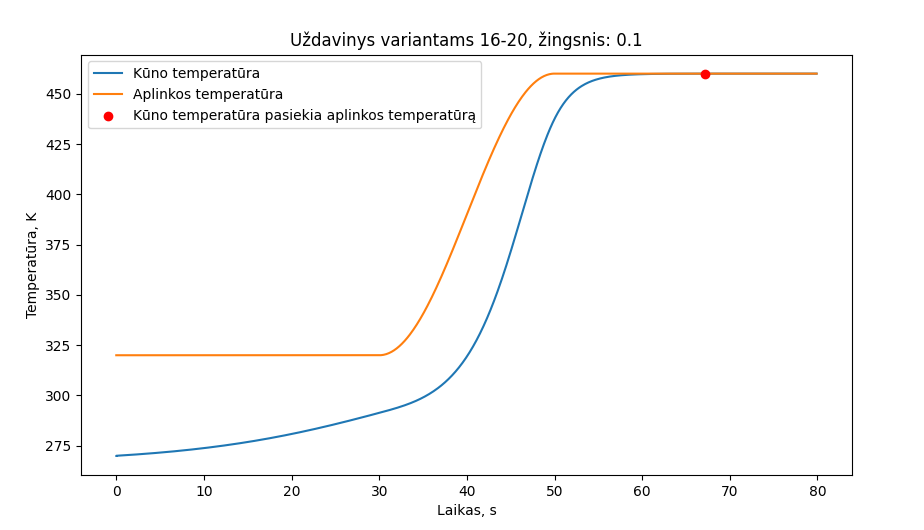
Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.64s,

temperatūra = 459.98011680866875T



pav. 16 Eulerio metodo rezultatai, kai T = T2 ir žingsnis = 0.0001

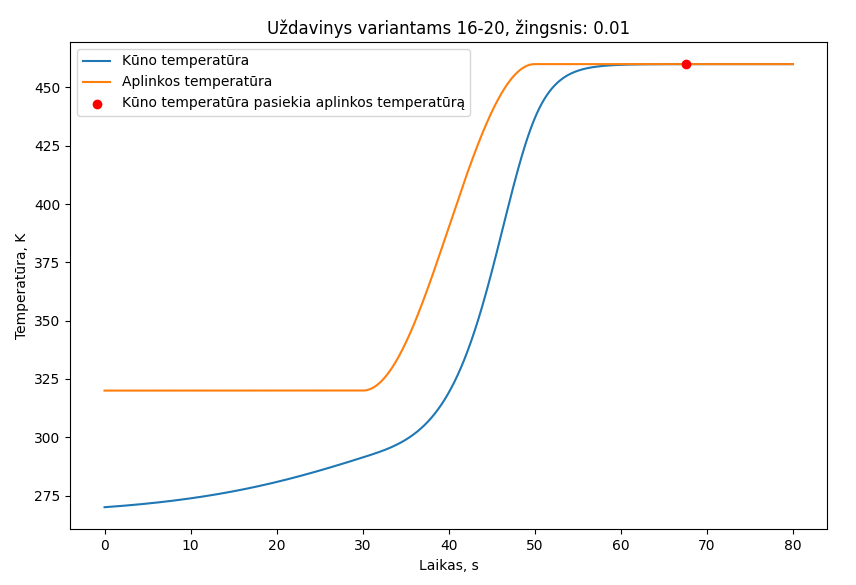
Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.6434s, temperatūra = 459.98000220971284T



pav. 17 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T2 ir žingsnis = 0.1

Kūno temperatūra pasiekė aplinkos temperatūrą, kai

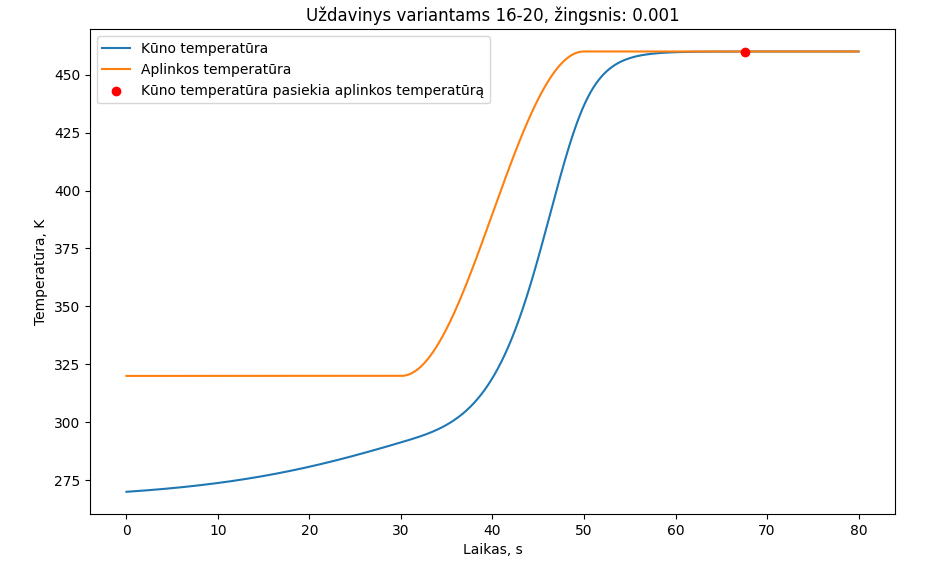
laikas = 67.60000000000001s, temperatūra = 459.9814094993552T



pav. 18 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T2 ir žingsnis = 0.01

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.64s,

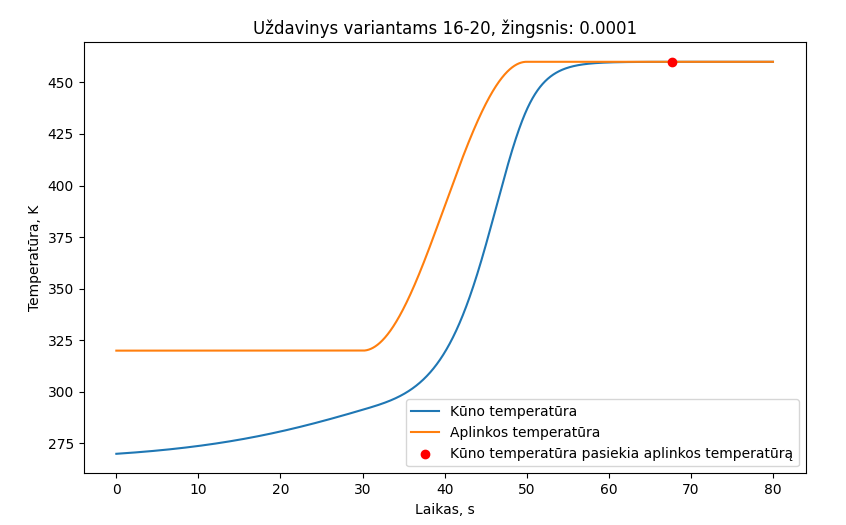
temperatūra = 459.9813716365492T



pav. 19 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T2 ir žingsnis = 0.001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.644,

temperatūra = 459.9801164477151T



pav. 20 IV eilės Rungės ir Kutos metodo rezultatai, kai T = T2 ir žingsnis = 0.0001

Kūno temperatūra pasiekė aplinkos temperatūrą, kai laikas = 67.6438,

temperatūra = 459.9800021737776T

## Antros dallies išvados

Iš gautų rezultatų matome, kad IV eilės Rungės ir Kutos metodas greičiau nustato kada kūnas pasiekia aplinkos temperatūra ir kad laikas, kada tai įvyksta, yra ~ 67.64 sekundę. Žingsnių kiekis yra mažesnis, lyginant su pirmąja temperatūra, kad galėtume nustatyti, ties kuria reikšme *nusistovi* gautas rezultatas.