BENDRASIS PROGRAMAVIMO DOKUMENTAS	EUROPOS SĄJUNGA Europos socialinis fondas	ŠVIETIMO IR MOKSLO MINISTERIJA	
KURKIME ATEITĮ DRAUGE!			

2004-2006 m. Bendrojo programavimo dokumento 2 prioriteto „Žmogiškųjų išteklių plėtra“ 4 priemonė „Mokymosi visą gyvenimą sąlygų plėtra“

Projekto sutarties numeris: **ESF/2004/2.4.0-K01-160/SUT-261**

Projekto pavadinimas: **Inovatyvūs mokymosi metodai ir naujausios technologijos gamtos mokslų bakalaurų rengimui**

FIZ 313 KOMPIUTERINĖ FIZIKA

Laboratorinis darbas

FIZIKOS DIFERENCIALINIŲ LYGČIŲ SPRENDIMAS RUNGĖS KUTOS METODU

1. Darbo tikslas:

1. Išmokti taikyti Oilerio bei Rungės Kuto metodus fizikos diferencialinių lygčių sprendimui.

2. Darbo užduotнs:

1. Rasti parašiutininko leidimosi greičio priklausomybę nuo laiko Oilerio ir Rungės Kuto metodais.

2. Nustatyti trumpiausią saugaus parašiutininko nusileidimo trukmę.

3. Bendroji teorija

Oilerio metodas

Nagrinėdami realius mus supančius reiškinius dažnai norime juos ne tik suprasti kokybiškai, bet ir išsiaiškinti kiekybinius dėsningumus – gauti tam tikrų dydžių skaitines vertes. Tam neretai kokybinio modelio nepakanka – dar reikia sugalvoti, kokiu būdu analitines lygtis paversti skaitinio išvedimo rezultatais. Statiniais (pusiausvyriniais) atvejais tai paprasta. Pavyzdžiui, į dujų būsenos lygtį įstatę dominančius skaičius suskaičiuojame atsakymą, ir kompiuterio šiam darbui net nereikia, nebent norėtume gauti didelę skaičių lentelę. Šiuo atveju nereikia ir programuoti, pakanka pasinaudoti kokia nors populiaria skaičiuokle.

Visai kitokia situacija yra su realiomis situacijomis, kai fizikiniai (mechaniniai, elektriniai ar net biologiniai) dydžiai keičiasi proceso eigoje. Tada šių dydžių priklausomybes galima atvaizduoti diferencialinių lygčių pagalba. Palyginti nedaug diferencialinių lygčių galima išspręsti analiziškai, realioms situacijoms reikia taikyti skaitinius metodus. Pats paprasčiausias metodas yra Oilerio metodas.

Oilerio metodui išsiaiškinti pasinaudosime ore krentančio kūno judėjimo tyrimu. Tarkime, parašiutininkas iššoka iš lėktuvo ir 24 sekundes neišskleidžia parašiuto. Reikia rasti parašiutininko greičio priklausomybę nuo laiko.

Parašiutininko skrydį apsprendžia dvi jėgos: gravitacija ir oro pasipriešinimas. Iš pradžių tarsime, kad oro pasipriešinimas yra proporcingas greičio kvadratui βv^2. Antrasis Niutono dėsnis atrodo taip:

$$m\mathrm{d}v/\mathrm{d}t = mg - \beta v^2 \tag{1}$$

kur m yra parašiutininko masė, g – laisvojo kritimo pagreitis, v – jo greitis. Padalinę iš m gauname:

$$\mathrm{d}v/\mathrm{d}t = g - kv^2, \tag{2}$$

kur $k = \beta/m$. Koeficientas k priklauso nuo įvairių sunkiai apibrėžiamų parametrų, pavyzdžiui, judančio kūno formos, todėl jį paprasčiau nustatyti eksperimentiškai. Po kurio laiko krintančio kūno greitis pasiekia maksimalią vertę, t.y., dv = 0. Šiuo atveju diferencialinę lygtį galima išspręsti analitiškai

$$v_t = \sqrt{g/k}\ , \tag{3}$$

$$v(t) = \frac{a\left(C - e^{-2akt}\right)}{C + e^{-2akt}}\ , \tag{4}$$

kur koeficientai lygūs $a = v_t$ ir $C = [a + v(0)]/[a - v(0)]$.

Sprendžiant lygtį Oilerio metodu skaitiškai remiamasi tokia idėja. Tam tikru momentu greičio pokytį Δv per Δt apsprendžia jėga $(g - kv^2)$. Kuo didesnis Δt, tuo ilgiau veikia ši jėga, tuo didesnis bus pokytis:

$$\Delta v = (g - kv^2)\,\Delta t, \tag{5}$$

kur $\Delta v = v(t + \Delta t) - v(t)$.

Reikia neužmiršti, kad krintančio kūno greitis v keičiasi, todėl narys kv^2 yra nepastovus. Tai reiškia, kad greičio pokytis Δv įvairiais laiko momentais yra skirtingas, todėl jeigu Δt bus didelis. Tačiau jei imsime labai mažą Δt, tai galima daryti prielaidą, kad greitis per tokį mažą laiką pasikeičia nežymiai, todėl galima teigti, kad greičio pokytis proporcingas laiko padidėjimui Δt ir kūną veikiančiai jėgai, t.y., $g - kv^2$. Δt paprastai vadinamas žingsniu: $h = dt$.

Tada skaitiniam sprendimui paruošta lygtis atrodo taip:

$$v(t+h) = -v(t) + h(g - kv^2(t)), \tag{6}$$

t.y. kūno greitis laiko momentu t + h yra lygus greičiui laiko momentu t plius priedas $h(g - kv^2(t))$, kurį sąlygoja kūną veikiančios jėgos. Jeigu kūnas juda laiką T, o žingsnio trukmė h, tai greitį skaičiuosime n kartų, kur $T/h = n$. Perrašius (6) lygtį į rekurentinę formą, gauname

$$v_{i+1} = -v_i + h(g - kv_i^{\,2}), \tag{7}$$

kur kiekvieno žingsnio i laiko momentą t galime rasti naudodami sąryšį $t = h\cdot i$.

Oilerio metodo matematika. Kad pateiktas metodas būtų lengviau suprantamas, Oilerio metodą trumpai užrašysime matematikoje įprasta simbolika. Tarkime, kad skaitiškai tiriame funkciją $y(x)$. Bendru atveju funkcijos prieaugis, kai $\Delta x = h$, lygus $f(x, y)$, t.y.,

$$\frac{y(x+h)-y(x)}{h} = f(x,y) \tag{8}$$

Tada funkcijos reikšmė

$$y_{n+1} = y_n + hf(x_n, y_n) \tag{9}$$

kur y_n atitinka $y(x_n)$; $x_n = x = nh$, o $y_0 = y(0)$.

4-os eilės Runge-Kutto metodas

Oilerio metodas (OM) tinka tada, kai ieškoma funkcija kinta lėtai. Tiriant realius fizikinius procesus dažnai tokai sąlyga neišpildoma. Įvairaus pobūdžio fizikos diferencialinių lygčių sprendimų teorija gana stipriai išvystyta, sukurti skaitinio integravimo algoritmai, todėl fizikui dažnai nereikia gilintis į matematinius subtilumus. Fortrano kalbai pritaikytose bibliotekose, tame tarpe ir MSIMSL, yra ne viena paprogramė, kurią galima lanksčiai pritaikyti konkretaus uždavinio sprendimui. Beveik visi šie algoritmai pagrįsti Runge-Kutta metodu. Nesigilindami į matematinį šio algoritmo pagrįstumą ir tikslumą, išsiaiškinsime ketvirtos eilės Runge-Kutta (RK) metodą, kurio turėtų pakakti praktiškai visiems atvejams, su kurias susiduria fizikai.

Tarkime, kad DL, kurią reikia spręsti, yra

$$dy / dx = f(x, y) \tag{10}$$

kai duota pradinė sąlyga $y(0)$.

Kito žingsnio funkcijos reikšmė $y(x+h)$ surandama pagal formulę:

$$y^* = y + (k_1 + 2k_2 + 2k_3 + k_4)/6 \tag{11}$$

kur koeficientai k_i lygūs:

$$
\begin{aligned}
k_1 &= hf(x, y) \\
k_2 &= hf(x+0.5h, y+0.5k_1) \\
k_3 &= hf(x+0.5h, y+0.5k_2) \\
k_4 &= hf(x+h, y+k_3)
\end{aligned}
\qquad (12)
$$

4. Tyrimo metodika

Aprašytam Oilerio metodui išbandyti naudokite su paruoštą programa, kuri pateikiama priede. Šį programa skaitiniu būdu sprendžia parašiutininko leidimosi uždavinį analiziniu atveju, tai yra, kai oro pasipriešinimo koeficientas yra 2, $\sim\beta v^2$. Šioje programoje integruotas ir analizinio sprendinio skaičiavimas, ir skaitinis įvertinimas naudojant Oilerio metodą. Oilerio metodo tikslumas priklauso nuo pasirinkto žingsnio, todėl ištirkite, kaip skaičiavimo rezultatai priklauso nuo žingsnio h vertės. Lygindami su analiziniu atsakymu, mažinkite žingsnį tol, kol tikslumas bus pakankamas.

Modeliavimo metu reikia atsižvelgti į tai, kad tam tikru momentu išskleidžiamas parašiutas. Skaitiniame uždavinio sprendime tai atsispindi oro pasipriešinimo koeficiento pasikeitime; pilnai išskleidus parašiutą jis lygus maždaug $k \sim 0{,}3$. Pakeiskite programą taip, kad joje būtų galima tam tikru momentu „išskleisti“ parašiutą. Išnagrinėkite parašiutininko kritimo greičio priklausomybę laike, kai jis iš pradžių krenta laisvai, o po to išskleidžia parašiutą. Nustatykite stacionarų laisvo kritimo greitį su parašiutu ir be parašiuto. Įvertinkite, per kokį trumpiausią laiką parašiutininkas gali saugiai nusileisti ant žemės, jei jis iššoka iš lėktuvo, skrendančiame 1000 m aukštyje. Suraskite maksimalią laisvo kritimo (be parašiuto) trukmę iki to momento, kad parašiutą išskleidus, parašiutininko greitis suspėtų sumažėti.

Reali oro pasipriešinimo priklausomybė nuo greičio yra artima $\sim kv^{1,8}$. Šis atvejis jau neišsprendžiamas analiziškai, tačiau parašyta programą lygiai taip pat galima naudoti ir šiam atvejui. Patikrinkite, kaip pasikeičia v(t), kai oro pasipriešinimas $\sim kv^{1,8}$ ir kai $\sim kv^2$. Gautas priklausomybes nubraižykite.

Programos su parašiutininko antroji versija padaryta naudojant 4-os eilės Runge-Kutta metodą. Išsiaiškinkite ją ir palyginkite, rezultatų tikslumą esant tam pačiam žingsniui h Oilerio metodu ir RK metodu .

5. Tyrimo eiga

1. Laboratorinis darbas atliekamas kompiuterių klasėje.

2. Ištirkite parašiutininko nusileidimo greičio priklausomybę nuo laiko naudodami Oilerio metodą. Nubraižykite analizinio sprendinio kreivę bei Oilerio metodo kreives, gautas esant skirtingiems žingsniams. Paaiškinkite rezultatų netikslumų priežastis.

3. Papildykite programą taip, kad ji skaičiuotų parašiutininko greičio priklausomybę išskleidus parašiutą. Ištirkite parašiutininko kritimo greičio priklausomybę laike, kai jis iš pradžių krenta

laisvai, o po to išskleidžia parašiutą. Nustatykite stacionarų laisvo kritimo greitį su parašiutu ir be parašiuto. Pavaizduokite gautas priklausomybes grafiškai bei trumpai paaiškinkite gautus rezultatus.

4. Papildykite programą taip, kad ji skaičiuotų parašiutininko aukštį virš žemės paviršiaus.

5. Įvertinkite, per kokį trumpiausią laiką parašiutininkas gali saugiai nusileisti ant žemės, jei jis iššoka iš lėktuvo, skrendančiame 1000 m aukštyje. Suraskite maksimalią laisvo kritimo (be parašiuto) trukmę iki to momento, kad parašiutą išskleidus, parašiutininko greitis suspėtų sumažėti.

6. Nubraižykite parašiutininko aukščio virš žemės paviršiaus ir greičio priklausomybes nuo laiko ir paaiškinkite gautus rezultatus.

7. Papildykite programą taip, kad ji spręstų 5 užduotį Rungės kuto metodu ir leistų palyginti Oilerio ir Rungės Kuto metodo tikslumą. Nubraižykite parašiutininko, iššokusio iš lėktuvo parašiuto, greičio priklausomybes nuo laiko abiem metodais, esant skirtingam žingsniui.

8. Atliktą darbą pateikti parašytą tekstiniu redaktoriumi. Atsiskaitymo dokumente turi būti užduoties sąlyga bei atsakymai į 2, 3, 5, 6, 7 tyrimo eigos klausimus. Visi rezultatai turi turėti aiškiai apibrėžtą išvadą bei trumpą paaiškinimą.

6. Kontroliniai kiausimai

1. Parašiutininko kritimo užduoties formulavimas, analizinio ir skaitinio sprendimo metodai.

2. Oilerio ir Rungės Kuto metodai diferencialinių lygčių sprendimui.

3. Fizikinių procesų, aprašomų pirmos eilės diferencialinėmis lygtimis su pastoviais koeficientais, pavyzdžiai.

4. Mokėti atsakyti į visus klausimus, susijusius su naudotos programos programavimo technika.

7. Literatūra

1. A. Kanapickas. Paskaitų konspektas. 4 skyrius: vidinės funkcijos.

2. Tamašauskas A. , Vosylius J. Fizika, T. 1. - Vilnius: Mokslas, 1989. – 195 p.

3. Fortran for scientists and engineers. Chapter 16.1: introduction to numerical methods – equations.

Priedas Nr. 1. Užduoties sprendimo pavyzdys Oilerio metodu.

```
PROGRAM Para
IMPLICIT NONE
REAL, PARAMETER :: g = 9.8
REAL K                  ! Pasipriesinimo koeficientas
REAL H                  ! zingsnis dt
REAL T                  ! laikas
REAL T0                 ! Pradinis laikas
REAL Tend               ! Galutinis laikas
REAL V                  ! greitis
REAL V0                 ! pradinis greitis
! REAL X                !

INTEGER I, N
10  FORMAT( 1x, A )
WRITE( *, 10, ADVANCE = 'NO' ) 'Iveskite pasipriesinimo koeficienta'
WRITE( *, 10, ADVANCE = 'NO' )  '(~ 0.004 zmogui, 0.3 parasiutui): '
READ*, K
WRITE( *, 10, ADVANCE = 'NO' )  'zingsni ( <= 1): '
READ*, H
WRITE( *, 10, ADVANCE = 'NO' )  'Pradini laika (~0): '
READ*, T0
WRITE( *, 10, ADVANCE = 'NO' )  'Pradini greiti: '
READ*, V0
WRITE( *, 10, ADVANCE = 'NO' )  'Galutini laika: '
READ*, Tend

N = INT((Tend - T0) / H) + 1
T = T0
V = V0
PRINT "(3A10)", "Time", "Euler", "Exact"

DO I = 1, N
  PRINT "(3F10.2)", T, V, Vexact(T, V0, G, K)
  V = V + H * (g - K * V * V)
  T = T + H
END DO

CONTAINS
  FUNCTION Vexact(T, V0, G, K)
    REAL Vexact
    REAL, INTENT(IN) :: g, K, T, V0
    REAL A, C
    A = SQRT( g / K )
    C = (A + V0) / (A - V0)
    Vexact = A * (C - EXP(-2*A*K*T))/(C + EXP(-2*A*K*T))
  END FUNCTION Vexact
END PROGRAM Para
```

Priedas Nr. 2. Paprogramės greičio skaičiavimui analziniu, Oilerio ir Rungės Kuto metodais

```
FUNCTION Vexact(T, V0, G, K)
   REAL Vexact
   REAL, INTENT(IN) :: g, K, T, V0
   REAL A, C
   A = SQRT( g / K )
   C = (A + V0) / (A - V0)
   Vexact = A * (C - EXP(-2*A*K*T))/(C + EXP(-2*A*K*T))
  END FUNCTION Vexact

  FUNCTION dcRunge(conc, K, dt)
   REAL Vrunge
   REAL, INTENT(IN) :: g, K, V, dt
   REAL k1, k2, k3, k4                              ! k1 = h * f(x        , y)
             ! k2 = h * f(x + 0.5*h, y + 0.5*k1)
             ! k3 = h * f(x + 0.5*h, y + 0.5*k2)
                              ! k4 = h * f(x + h    , y + k3)
   k1 = dt * (g - K * V * V)
   k2 = dt * (g - K * (V + 0.5 * k1) * (V + 0.5 * k1))
   k3 = dt * (g - K * (V + 0.5 * k2) * (V + 0.5 * k2))
   k4 = dt * (g - K * (V + k3) * (V + k3))

   Vrunge = V + (k1 + 2 * k2 + 2 * k3 + k4)/6
  END FUNCTION Vrunge
```