

Sveučilište u Zagrebu
Građevinski fakultet

Kolegij: Primjenjena matematika
**ODREĐIVANJE DINAMIČKOG ODZIVA MEHANIČKOG
SUSTAVA METODOM RUNGE-KUTTA**

Seminarski rad

Student: Marija Nikolić

Mentor: prof.dr.sc. Vera Čuljak

Zagreb, 2011.god.

Sadržaj:

1. Uvod.....	2
2. Numeričke metode za rješavanje običnih diferencijalnih jednažbi prvog reda.....	3
2.1. Eulerova metoda.....	3
2.1.1. Procjena pogreške.....	5
2.2. Runge-Kutta metoda.....	5
2.2.1. Izvod Runge-Kutta metode drugog reda iz Taylorove formule.....	7
2.2.2. Izvod Runge-Kutta metode četvrtog reda preko Simsonove formule.....	10
2.3. Rješavanje sustava i diferencijalnih jednažbi višeg reda.....	13
3. Primjena Runge-Kutta metode na rješavanje diferencijalnih jednažbi u dinamici konstrukcija.....	15
3.1. Sustav s jednim dinamičkim stupnjem slobode gibanja.....	15
3.2. Sustav s više stupnjeva slobode gibanja.....	17
3.3. Kod za rješavanje sustava diferencijalnih jednažbi metodom Runge-Kutta.....	19
3.4. Numerički primjeri iz područja dinamike konstrukcija.....	24
4. Zaključak.....	30
5. Literatura.....	31

1. Uvod

Matematička formulacija fizikalnih modela za analizu njihovog dinamičkog odziva dana je diferencijalnom jednačbom ili sustavom diferencijalnih jednačbi. Vrlo često nije moguće odrediti analitička rješenja tih jednačbi te je tada potrebno nekim od numeričkih postupaka odrediti aprosimacijska rješenja. Numeričke metode danas imaju značajnu prednost u odnosu na analitičke, prije svega jer se algoritmi tih metoda mogu programirati pomoću računala. Što bolja aproksimacija rješenja te jednostavnost programiranja i brzina proračuna uz što manje računalne memorije i vremena osnovni su kriterij pri odabiru numeričke metode. U okviru ovog rada izvesti će se algoritmi za numeričko rješavanje običnih diferencijalnih jednačbi Eulerovom metodom i Runge-Kutta metodom drugog i četvrtog stupnja. Objasniti će se primjena tih metoda na rješavanje običnih diferencijalnih jednačbi višeg reda kao i njihova primjena na rješavanje sustava diferencijalnih jednačbi. Eulerova te obje Runge-Kutta metode primjenjene su na rješavanje obične diferencijalne jednačbe prvog reda radi usporedbe točnosti s egzaktnim rješenjem, dok su problemi dinamike konstrukcija riješeni primjenom Runge-Kutta metode četvrtog reda.

2. Numeričke metode za rješavanje običnih diferencijalnih jednadžbi prvog reda

Obična diferencijalna jednadžba (ODJ) prvog reda ima oblik $F(x, y, y') = 0$ i često se može napisati u eksplicitnom obliku $y' = f(x, y)$. Za tako definiranu diferencijalnu jednadžbu možemo postaviti problem početne vrijednosti:

$$y' = f(x, y), \quad y(x_0) = y_0, \quad (1)$$

gdje su x_0 i y_0 zadane početne vrijednosti i pretpostavljamo da problem za dane početne uvjete ima jedinstveno rješenje na intervalu $a < x < b$ koji sadrži x_0 .

Numeričke metode koriste se za izračunavanje aproksimacijskih numeričkih vrijednosti rješenja $y(x)$ unutar zadanog intervala u jednako razmaknutim točkama na x -osi:

$$\begin{aligned} x_1 &= x_0 + h, \\ x_2 &= x_0 + 2h, \\ x_3 &= x_0 + 3h, \\ &\dots \end{aligned} \quad (2)$$

gdje je veličina koraka h nekakva odabrana vrijednost, npr. $0,2$ ili $0,1$ itd. Takve metode se nazivaju korak po korak i koriste iste formule u svakom koraku, mogu se izvesti iz Taylorovog reda koji razvija funkciju $y(x)$ u red potencija u okolini točke $x + h$:

$$y(x+h) = \sum_{n=0}^{\infty} h^n \cdot \frac{y^{(n)}(x)}{n!}, \quad (3)$$

$$y(x+h) = y(x) + h \cdot y'(x) + \frac{h^2}{2!} \cdot y''(x) + \frac{h^3}{3!} \cdot y'''(x) + \dots \quad (4)$$

2.1. Eulerova metoda

Kod odabira male vrijednosti koraka h , potencije broja h , npr. $h^2, h^3 \dots$ su jako male veličine. Ako u Taylorovom redu uzmemo samo prvi član razvoja dobiti ćemo vrlo grubu aproksimaciju:

$$\begin{aligned} y(x+h) &\approx y(x) + h \cdot y'(x) \\ &= y(x) + h \cdot f(x, y), \end{aligned} \quad (5)$$

gdje je funkcija $f(x, y)$ dana diferencijalnom jednadžbom, odnosno izrazom (1) na temelju čega slijedi iteracijski proces.

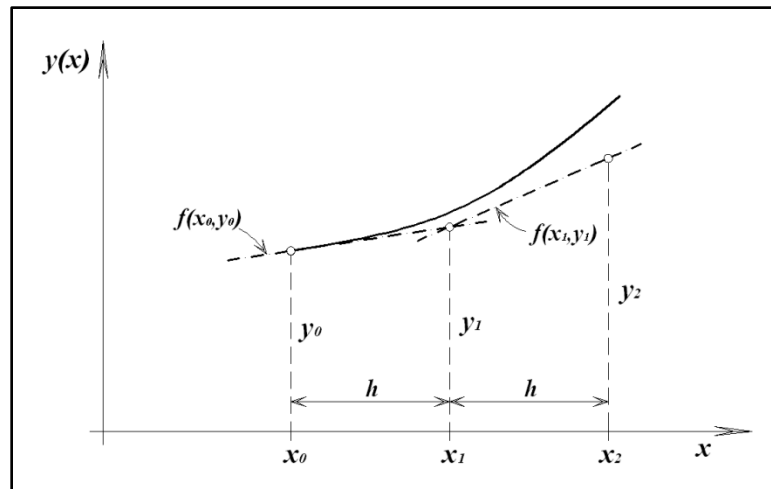
Poznavajući vrijednosti na početku intervala x_0 i y_0 u prvom koraku određujemo vrijednost y_1 :

$$y_1 = y_0 + h \cdot f(x_0, y_0), \quad (6)$$

koja aproksimira vrijednost funkcije $y(x)$ u točki $x_0 + h$. U drugom koraku računamo:

$$y_2 = y_1 + h \cdot f(x_1, y_1), \quad (7)$$

koja aproksimira vrijednost funkcije $y(x)$ u točki $x_0 + 2h$, itd.



Slika 1. Grafička interpretacija Eulerove metode

Stoga za neki unutarnji korak proračuna možemo općenito pisati:

$$y_{n+1} = y_n + h \cdot f(x_n, y_n). \quad (8)$$

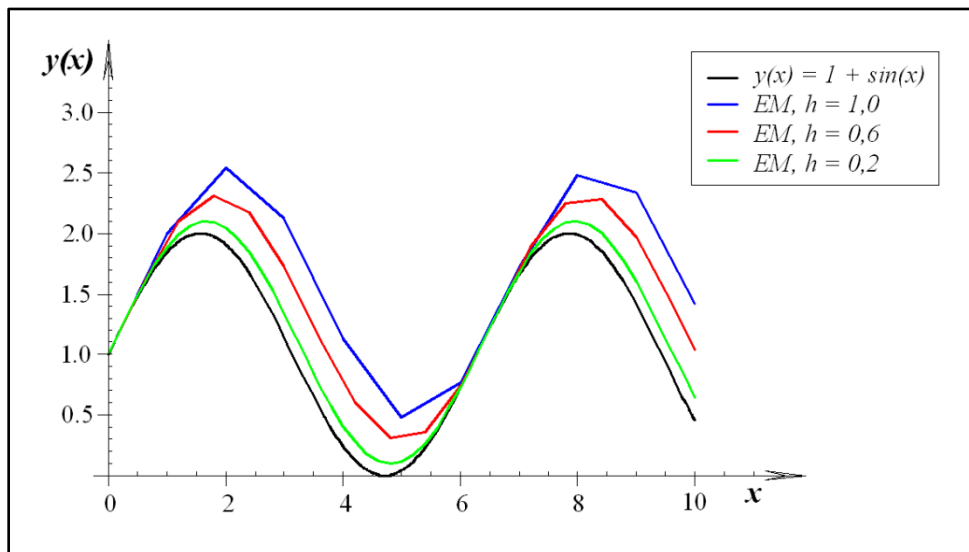
Ovaj postupak se naziva **Eulerova metoda** ili **Euler-Cauchy-eva metoda**. Geometrijski predstavlja aproksimaciju krivulje $y(x)$ poligonom čija je prva stranica tangenta krivulje u x_0 . Na slici 1. prikazana je grafička interpretacija Eulerove metode. Ova metoda ima vrlo grubu aproksimaciju i gotovo se nikad ne koristi u praksi, no budući je vrlo jednostavna jako dobro objašnjava princip metoda koje se temelje na aproksimaciji funkcije u okolini neke točke Taylorovim redom.

Primjer 1. Primjeni Eulerovu metodu za rješavanje problema početne vrijednosti:

$$y' = \cos(x), \quad y(0) = 1,$$

uz primjenu veličine koraka $h=1,0$, $h=0,6$ i $h=0,2$.

Na slici 2. grafički su prikazana rješenja dobivena Eulerovim postupkom i egzaktno rješenje zadane diferencijalne jednadžbe $y(x) = 1 + \sin(x)$.



Slika 2. Grafički prikaz rješenja za primjer 1.

2.1.1. Procjena pogreške

Taylorova formula s ostatkom ima oblik:

$$y(x+h) = y(x) + h \cdot y'(x) + \frac{1}{2} \cdot h^2 \cdot y''(\xi), \quad (9)$$

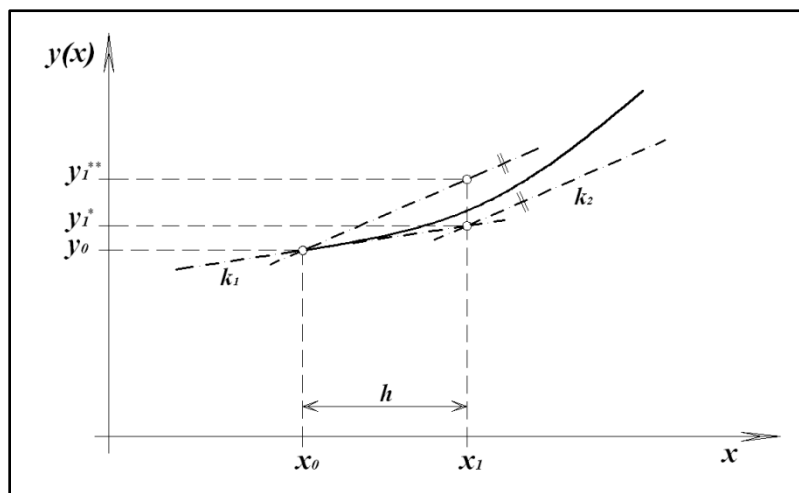
gdje je $x \leq \xi \leq x + h$. Ovaj izraz pokazuje da je u Eulerovoj metodi greška odsijecanja u svakom koraku ili lokalna greška odsijecanja proporcionalna s h^2 , pišemo $O(h^2)$, gdje O upućuje na red (order) pogreške. Pošto se aproksimacija rješenja ODJ određuju na nekom intervalu $x_0 < x < x_0 + N \cdot h$, broj koraka N koji je potreban kako bi se prešao zadani interval je proporcionalan s $1/h$. Tako da je totalna greška ili globalna pogreška proračuna proporcionalna s $h^2(1/h) = h^1$. Radi ovog razloga, Eulerova metoda se naziva **metoda prvog reda**. Uz to postoje i dodatne pogreške zaokruživanja u ovoj i drugim metodama koje mogu utjecati na točnost vrijednosti y_1, y_2 , itd. sve više i više kako se broj koraka N povećava.

2.2. Runge-Kutta metoda

Očito je da postoji kompromis između točnosti i složenosti proračuna koji značajno ovisi o odabranoj vrijednosti koraka h . Smanjivanjem koraka h proračun dulje traje, ali je točniji. No ako se korak h previše smanji, tada zaokruživanje koje se vrši pri kompjuterskom računanju (jer se realni broj u računalu mora prikazati konačnim brojem znamenki) počinje se akumulirati sve više i to može

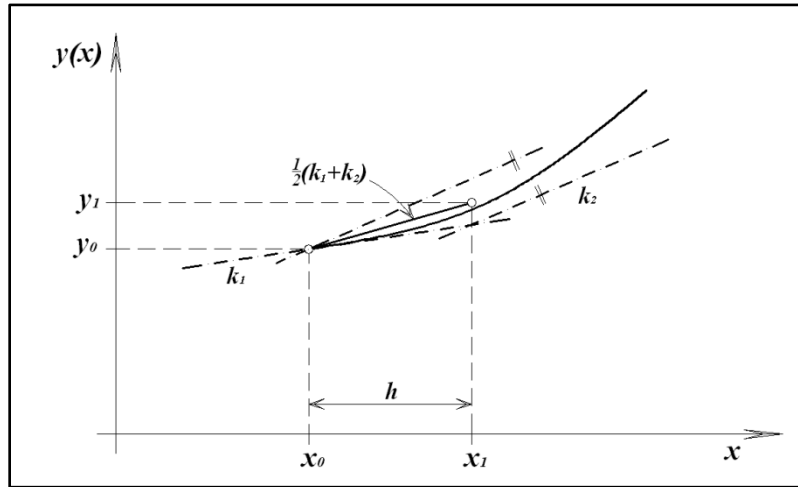
uzrokovati značajne pogreške. Stoga je za mnoge sustave višeg reda jako teško učiniti Eulerovu aproksimaciju efektivnom. Radi ovog razloga razvijene su više točne i razrađenije metode njihova proračuna. Ovdje će se opisati jedna takva metoda koju su razvili dva matematičara, Runge i Kutta početkom 20-tog stoljeća po kojima je metoda nazvana. Bitno je naglasiti da danas postoji cijela familija Runge-Kutta metoda koje se temelje na jednoj osnovnoj ideji koja će se pobliže objasniti u nastavku.

Runge-Kutta metoda može se najjednostavnije shvatiti kao pobljšanje Eulerove metode. Osnovna ideja metode prikazana je grafički na slici 3. i slici 4., a temelji se na određivanju što povoljnijeg nagiba tangente kojim će se prijeći interval duljine h . Kako bi se odredila aproksimacijska vrijednost funkcije $y(x)$ na kraju koraka h prvo se s poznatom derivacijom funkcije $y(x)$ u x_0 (odnosno nagibom tangente koju ćemo označiti s k_1) prijeđe interval za veličinu koraka h i tako odredi prva procjena vrijednosti funkcije y_1^* . Ako pogledamo sliku 3. možemo vidjeti da tangenta k_1 u ovom primjeru daje podcijenjenu vrijednost funkcije u točki x_0+h . Kada je poznata vrijednost y_1^* može se odrediti približna vrijednost za derivaciju (vrijednost nagiba tangente) funkcije $y(x)$ u točki $x_0 + h$ (koja će se označiti s k_2).



Slika 3.

Ako se sada promatrani interval prijeđe koristeći nagib k_2 , dobit ćemo drugu procjenu vrijednosti funkcije y_2^{**} u $x + h$. U ovom slučaju za promatrani primjer vrijednost y_2^{**} daje veću vrijednost funkcije od stvarne. Iz ovog obrazloženja intuitivno se može zaključiti da će se odabirom srednje vrijednosti nagiba tangente dobiti bolja procjena za vrijednost $y(x_0+h)$ – ovo je osnovna ideja Runge-Kutta metode. Stoga odredimo srednju vrijednost dviju krajnjih derivacija $k = (k_1+k_2)/2$ da bi dobili konačnu procjenu vrijednosti $y(x_0+h)$ na kraju koraka koju ćemo označiti s y_1 što možemo vidjeti na slici 4. Ovo je metoda drugog reda jer rezultat odgovara razvoju Taylorovog redu na dva člana.



Slika 4. Osnovna ideja Runge-Kutta metode

2.2.1. Izvod Runge-Kutta metode drugog reda iz Taylorove formule

Runge-Kutta metode su zamišljene da imaju aproksimacijske greške višeg stupnja, ali i da za njih nije potrebno računati parcijalne derivacije funkcije $f(x,y)$, što je u fizikalnim simulacijama neizvedivo, zbog toga što se ne zna unaprijed buduće stanje sustava.

Dakle, u ovoj metodi kako bi dobili veću točnost uzimamo u obzir više članova Taylorovog reda:

$$y(x+h) = y(x) + h \cdot y'(x) + \frac{h^2}{2!} \cdot y''(x) + \underbrace{\frac{h^3}{3!} \cdot y'''(x) + \dots}_{O(h^3)}, \quad (10)$$

$$y(x+h) = y(x) + h \cdot y'(x) + \frac{h^2}{2!} \cdot y''(x) + O(h^3), \quad (11)$$

gdje su:

$$\begin{aligned} y'(x) &= f(x, y(x)) = f(x, y), \\ y''(x) &= f_x(x, y(x)) + y'(x) \cdot f_y(x, y(x)) = f_x(x, y) + f(x, y) \cdot f_y(x, y). \end{aligned} \quad (12)$$

Za računanje izraza (11) potrebno je znati parcijalne derivacije prvog reda funkcije $f(x,y)$, što se želi izbjeći primjenom ove metode. Novom aproksimacijom mogu se zamijeniti derivacije funkcije $f(x,y)$ s novom vrijednošću, jer želimo dobiti izraz u obliku:

$$y(x+h) = y(x) + h \cdot a \cdot f(x+b, y+c) + \bar{O}(h^3). \quad (13)$$

Da bi izveli Runge-Kutta metodu potrebno je proširiti Taylorov teorem na funkciju dvije varijable:

$$f(x+a, y+b) = f(x, y) + \sum_{n=1}^{\infty} \frac{1}{n!} \left(\frac{\partial}{\partial x} \cdot a + \frac{\partial}{\partial y} \cdot b \right)^n f(x, y). \quad (14)$$

Budući nam trebaju samo prve parcijalne derivacije iz prethodnog izraza uzimamo u obzir samo prve članove razvoja reda:

$$f(x+b, y+c) = f(x, y) + b \cdot f_x(x, y) + c \cdot f_y(x, y) + O(h^2). \quad (15)$$

Uvrštavanjem gornjeg izraza u jednadžbu (13) dobivamo:

$$y(x+h) = y(x) + h \cdot a \cdot f(x, y) + h \cdot a \cdot b \cdot f_x(x, y) + h \cdot a \cdot c \cdot f_y(x, y) + \underbrace{h \cdot a \cdot O(h^2) + \bar{O}(h^3)}_{O(h^3)}. \quad (16)$$

Ako gornju jednadžbu izjednačimo s jednadžbom (10) i uzmemo u obzir izraze (12) možemo odrediti vrijednosti koeficijenata a , b i c :

$$a = 1; \quad b = \frac{h}{2}; \quad c = \frac{h}{2} \cdot f(x, y). \quad (17)$$

Uvodimo oznaku za prirast u početnoj točki intervala:

$$k_1 = h \cdot f(x, y). \quad (18)$$

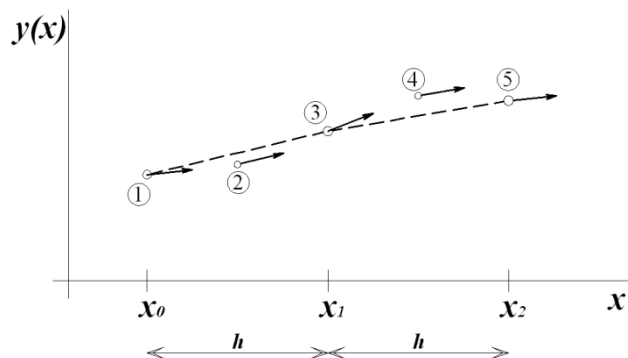
Kada određene vrijednosti koeficijenata uvrstimo u izraz (16) dobivamo aproksimaciju za parcijalne derivacije i uvedemo novu oznaku k_2 :

$$k_2 = h \cdot f(x+b, y+c) = h \cdot f\left(x + \frac{h}{2}, y + \frac{1}{2}k_1\right). \quad (19)$$

Novu vrijednost y možemo izračunati iz izraza:

$$y(x+h) = y(x) + k_2 + O(h^3). \quad (20)$$

Za razliku od Eulerove metode koja koristi derivaciju samo na početku intervala, Runge-Kutta koristi istu derivaciju ali samo za pomoćni korak kako bi se došlo do srednje točke intervala. Zatim u srednjoj točki računa derivaciju (nagib tangente) koju dalje koristi za određivanje "pravog" koraka kojim prelazi cijeli interval i određuje vrijednost funkcije na kraju intervala. Ovaj način proračuna naziva se i metoda srednje točke. Točnost drugog reda osigurana je korištenjem početne derivacije na početku svakog koraka kako bi se odredila srednja točka intervala, te se derivacijom (nagibom tangente) u srednjoj točki prelazi cijela duljina intervala. Shema metode prikazana je na slici 5.



Slika 5. Metoda srednje točke – Runge-Kutta drugog reda

Sada možemo napisati algoritam za računalno izračunavanje aproksimacijskih vrijednosti funkcije $y(x)$ koja je rješenje problema početne vrijednosti $y' = f(x, y)$, $y(x_0) = y_0$ u jednako razmaknutim točkama $x_1 = x_0 + h$, $x_2 = x_0 + 2h$, \dots , $x_N = x_0 + N \cdot h$, gdje je $f(x, y)$ takva funkcija da problem ima jedinstveno rješenje na intervalu $[x_0, x_N]$.

ULAZNE VRIJEDNOSTI: Funkcija $f(x, y)$, početne vrijednosti x_0 , y_0 , veličina koraka h i broj koraka N .

IZLAZNE VRIJEDNOSTI: Aproksimacijske vrijednosti y_{n+1} rješenja $y(x_{n+1})$ u $x_{n+1} = x_0 + (n+1)h$, gdje je $n = 0, 1, \dots, N-1$.

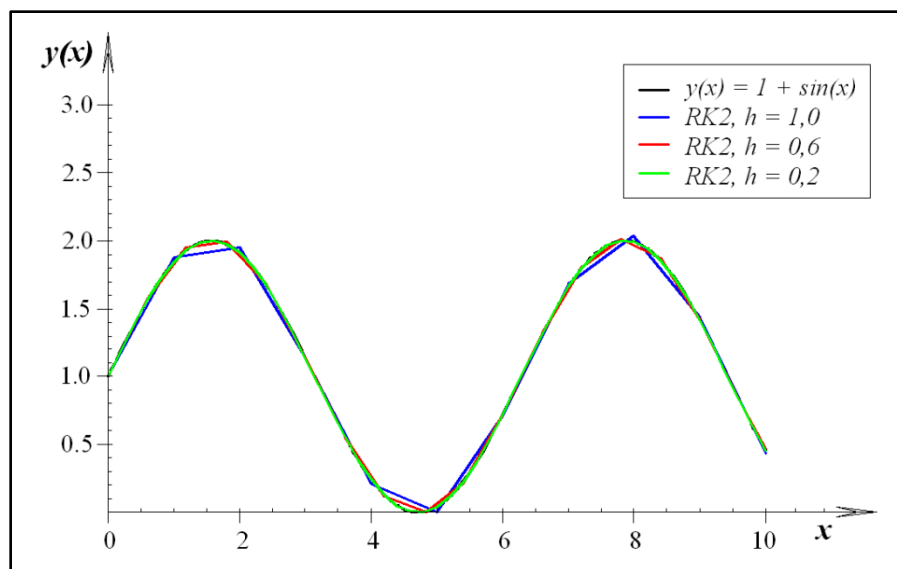
```

For  $n = 0, 1, \dots, N-1$  do :
     $k_1 = h \cdot f(x_n, y_n)$ 
     $k_2 = h \cdot f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$ 
     $x_{n+1} = x_n + h$ 
     $y_{n+1} = y_n + k_2$ 
    OUTPUT  $x_{n+1}, y_{n+1}$ 
End

```

Primjer 2. Primjeni Runge Kutta metodu drugog reda za rješavanje problema početne vrijednosti iz primjera 1. uz primjenu veličine koraka $h=1,0$, $h=0,6$ i $h=0,2$.

Na slici 6. grafički su prikazana rješenja dobivena Runge-Kutta metodom drugog reda (RK2) i egzaktno rješenje zadane diferencijalne jednadžbe $y(x) = 1 + \sin(x)$.



Slika 6.

Iz primjera možemo zaključiti da RK2 metoda djeluje tako da povećava točnost korištenjem bolje procjene nagiba tangente unutar intervala.

2.2.2. Izvod Runge-Kutta metode četvrtog reda preko Simsonove formule

Postoje i Runge-Kutta metode višeg reda, od kojih je najviše u uporabi Runge-Kutta metoda četvrtog reda. Ovakve metode višeg reda uzimaju u obzir više članova iz aproksimacije Taylorovim redom uzimanjem težinskog prosjeka od nekoliko aproksimacija derivacije unutar koraka. Radi jednostavnijeg prikaza izvoda ovdje će se umjesto razvoja Taylorovog reda do četvrtog člana početi od Picardove metode uzastopnih aproksimacija prema kojoj se aproksimacijska rješenja obične diferencijalne jednadžbe prvog reda $y' = f(x, y)$ uz početni uvjet $y(x_0) = y_0$ mogu odrediti iz sljedećim izrazom:

$$y(x) = y_0 + \int_{x_0}^x f(x, y) dx. \quad (21)$$

Odnosno za određivanje vrijednosti funkcije na kraju intervala $[x_0, x_1]$:

$$y(x_1) = y_1 = y_0 + \int_{x_0}^{x_1} f(x, y) dx. \quad (22)$$

Općenito možemo pisati:

$$y(x_n + h) = y_{n+1} = y_n + \int_{x_n}^{x_n+h} f(x, y) dx. \quad (23)$$

Za približno izračunavanje integrala u izrazu (23) koristimo Simsonovu formulu za funkciju dviju varijabli:

$$\int_{x_n}^{x_n+h} f(x, y) dx = \frac{h}{2} \cdot \frac{1}{3} \cdot \left(f(x_n, y(x_n)) + 4f(x_n + \frac{1}{2}h, y(x_n + \frac{1}{2}h)) + f(x_n + h, y(x_n + h)) \right) + O(h^5), \quad (24)$$

$$\int_{x_n}^{x_n+h} f(x, y) dx = \frac{h}{6} \cdot \left(f_{(x_n)} + 4f_{(x_n+\frac{1}{2}h)} + f_{(x_n+h)} \right) + O(h^5). \quad (25)$$

Simsonova formula na intervalu $[x_n, x_n+h]$ vrijednosti od $f(x, y)$ zamjenjuje polinomom drugog stupnja, koji funkciju $f(x, y)$ interpolira u čvorovima $x_n, x_n + \frac{1}{2}h, x_n + h$. Lokalna pogreška Simsonove formule je petog reda, tj. $O(h^5)$. Za primjenu metode Runge-Kutta točnost će se još više poboljšati tako da se srednji član razdvaja na dva dijela kako je prikazano u izrazu (26).

$$\int_{x_n}^{x_n+h} f(x, y) dx = \frac{h}{6} \cdot \left(f_{(x_n)} + 2f_{(x_n+\frac{1}{2}h)}^{(1)} + 2f_{(x_n+\frac{1}{2}h)}^{(2)} + f_{(x_n+h)} \right) + O(h^5). \quad (26)$$

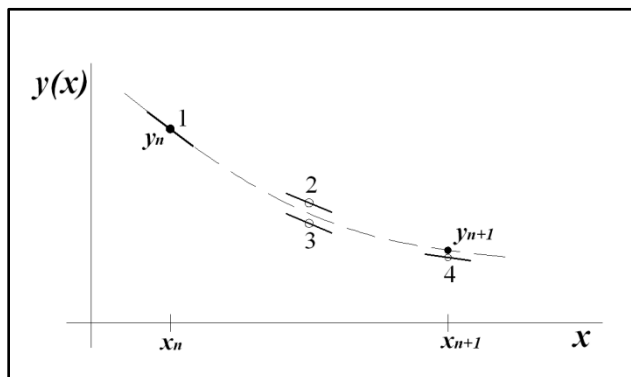
Derivacija u sredini intervala integracije računa se dva puta: prvo za vrijednost funkcije određene uz prirast na početku intervala integracije a zatim za vrijednost funkcije određene tako da prirast iz srednje točke intervala pridružimo početku intervala. Sukladno tome uvodimo sljedeće oznake:

$$\begin{aligned} k_1 &= f_{(x_n)} = h \cdot f(x_n, y_n) \\ k_2 &= f_{(x_n+\frac{1}{2}h)}^{(1)} = h \cdot f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1) \\ k_3 &= f_{(x_n+\frac{1}{2}h)}^{(2)} = h \cdot f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2) \\ k_4 &= f_{(x_n+h)} = h \cdot f(x_n + h, y_n + k_3) \end{aligned} \quad (27)$$

Uvrštavanjem u jednadžbu (23) dobivamo konačnu aproksimacijsku vrijednost za približno izračunavanje vrijednosti funkcije $y(x)$ u točki x_n :

$$y(x_n + h) = y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + O(h^5) \quad (28)$$

Visoka točnost metode, iako se koriste samo prve derivacije funkcije, osigurana je četverostrukim računanjem prve derivacije što je shematski prikazano na slici 7. Na taj način postignuta točnost koja odgovara razvoju Taylorovog reda do h^4 (lokalna pogreška reda veličine h^5).



Slika 7. Grafički prikaz Runge-Kutta metode četvrtog stupnja

Algoritam za računalno izračunavanje aproksimacijskih vrijednosti funkcije $y(x)$ po RK4 nije složeniji niti kompliciraniji u odnosu na RK2, jer u svakom koraku računa samo dva dodatna koeficijenta, dakle vrijeme računanja je nešto duže ali je točnost približnih vrijednosti višestruko bolja.

ULAZNE VRIJEDNOSTI: Funkcija $f(x,y)$, početne vrijednosti x_0, y_0 , veličina koraka h i broj koraka N .

IZLAZNE VRIJEDNOSTI: Aproksimacijske vrijednosti y_{n+1} rješenja $y(x_{n+1})$ u $x_{n+1}=x_0+(n+1)h$, gdje je $n = 0, 1, \dots, N-1$.

For $n = 0, 1, \dots, N-1$ do :

$$k_1 = h \cdot f(x_n, y_n)$$

$$k_2 = h \cdot f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$$

$$k_3 = h \cdot f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2)$$

$$k_4 = h \cdot f(x_n + h, y_n + k_3)$$

$$x_{n+1} = x_n + h$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

OUTPUT x_{n+1}, y_{n+1}

End

2.3. Rješavanje sustava i diferencijalnih jednadžbi višeg reda

Problem početne vrijednosti sustava ODJ prvog reda ima oblik:

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}) \qquad \mathbf{y}(x_0) = \mathbf{y}_0 \qquad (29)$$

Ako raspišemo komponente imamo:

$$\begin{aligned} y'_1 &= f_1(x, y_1, \dots, y_m) & y_1(x_0) &= y_{10} \\ y'_2 &= f_2(x, y_1, \dots, y_m) & y_2(x_0) &= y_{20} \\ &\vdots & &\vdots \\ y'_m &= f_m(x, y_1, \dots, y_m) & y_m(x_0) &= y_{m0} \end{aligned} \qquad (30)$$

Pretpostavljamo da za zadanu funkciju $\mathbf{f}(x, \mathbf{y})$ sustav ima jedinstveno rješenje $\mathbf{y}(x)$ na nekom otvorenom x -intervalu koji sadržava x_0 . Kada imamo sustav diferencijalnih jednadžbi algoritam za provedbu Runge-Kutta metode samo se proširuje u tom pogledu da umjesto skalara pri računanju koeficijenata za svaki korak imamo vektore m -tog reda:

ULAZNE VRIJEDNOSTI: Vektor funkcija \mathbf{f} , vektori početnih vrijednosti x_0, \mathbf{y}_0 , veličina koraka h i broj koraka N .

IZLAZNE VRIJEDNOSTI: Aproksimacijske vrijednosti \mathbf{y}_{n+1} rješenja $\mathbf{y}(x_{n+1})$ u $x_{n+1} = x_0 + (n+1)h$, gdje je $n = 0, 1, \dots, N-1$.

For $n = 0, 1, \dots, N-1$ do :

$$\mathbf{k}_1 = h \cdot \mathbf{f}(x_n, \mathbf{y}_n)$$

$$\mathbf{k}_2 = h \cdot \mathbf{f}(x_n + \frac{1}{2}h, \mathbf{y}_n + \frac{1}{2}\mathbf{k}_1)$$

$$\mathbf{k}_3 = h \cdot \mathbf{f}(x_n + \frac{1}{2}h, \mathbf{y}_n + \frac{1}{2}\mathbf{k}_2)$$

$$\mathbf{k}_4 = h \cdot \mathbf{f}(x_n + h, \mathbf{y}_n + \mathbf{k}_3)$$

$$x_{n+1} = x_n + h$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)$$

OUTPUT $x_{n+1}, \mathbf{y}_{n+1}$

End

Prije objašnjenja metoda za rješavanje sustava jednažbi bitno je uzeti u obzir da izraz (29) uključuje problem početne vrijednosti za jednu ODJ m -tog reda:

$$y^{(m)} = f(x, y, y', y'', \dots, y^{(m-1)}) \quad (31)$$

Početni uvjeti:

$$y(x_0) = K_1, y'(x_0) = K_2, \dots, y^{(m-1)}(x_0) = K_m \quad (32)$$

Ova dva izraza vrlo lako možemo povezati ako uzmemo da je:

$$y_1 = y, \quad y_2 = y', \quad y_3 = y'', \quad \dots, \quad y_m = y^{(m-1)} \quad (33)$$

Tako oblikujemo sustav jednažbi:

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= y_3 \\ &\vdots \\ y_{m-1}' &= y_m \\ y_m' &= f(x, y_1, \dots, y_m) \end{aligned} \quad (34)$$

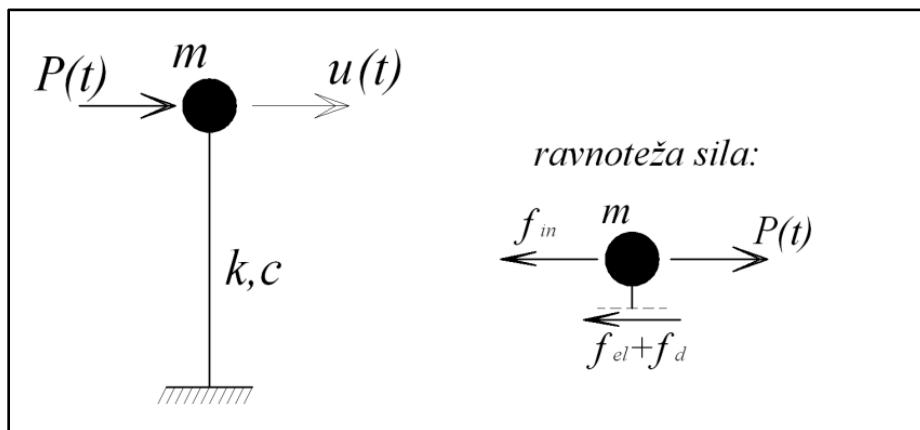
S početnim uvjetima:

$$\begin{aligned} y_1(x_0) &= K_1 \\ y_2(x_0) &= K_2 \\ &\vdots \\ y_m(x_0) &= K_m \end{aligned} \quad (35)$$

3. Primjena Runge-Kutta metode na rješavanje diferencijalnih jednačini u dinamici konstrukcija

3.1. Sustav s jednim dinamičkim stupnjem slobode gibanja

Sustav s jednim dinamičkim stupnjem slobode gibanja najjednostavniji je model konstrukcije za koji je moguće provesti dinamičku analizu. Da bismo pobudili sustav, moramo ga izvesti iz ravnotežnog položaja i zatim pustiti u gibanje. Sustav će slobodno oscilirati konstantnom amplitudom i takve će oscilacije teorijski težiti u beskonačnost. Naravno, da takvo ponašanje modela nije realno i intuitivno je jasno da će realni sustav podvrgnut slobodnim oscilacijama nakon izvjesnog vremena stati. Budući da se amplitude oscilacija smanjuju, očito je da sustav troši energiju. Utrošak energije modela u građevinarstvu najčešće opisujemo viskoznom prigušivačem¹ koji je proporcionalan s brzinom gibanja.



Slika 8. Matematički model dinamičkog sustava s jednim stupnjem slobode gibanja

Matematički model sustava (slika 8.) koji je opterećen dinamičkom silom $P(t)$ možemo opisati D'Alembertovim principom dinamičke ravnoteže sila:

$$f_{in} + f_d + f_{el} = P(t) \quad (36)$$

Vanjska sila $P(t)$ dinamičkog je karaktera, tj. varira s vremenom, a budući da je pomak sustava u uzrokovan silom $P(t)$ tada je i pomak $u(t)$ funkcija vremena. Iz jednačine dinamičke ravnoteže vidljivo je da se vanjskoj sili $P(t)$ opiru unutarnje sile sustava: elastična sila f_{el} , sila prigušenja f_d i sila inercije f_{in} . U bilo kojem trenutku masa je u dinamičkoj ravnoteži pod djelovanjem ovih sila.

¹ Otpor gibanju unutar gustih tekućina naziva se viskozno prigušenje (npr. gibanje klipa u motornom ulju). Kod malih brzina sila otpora proporcionalna je brzini.

Za linearni sustav elastična sila je: $f_{el} = k \cdot u(t)$, (37)

k - krutost sustava na poprečni pomak.

$u(t)$ - relativni pomak

Sila prigušenja iznosi: $f_d = c \cdot \dot{u}(t)$, (38)

c - koeficijent viskoznog prigušenja

$\dot{u}(t)$ - relativna brzina

Sila inercije koja djeluje na masu u gibanju je: $f_{in} = m \cdot \ddot{u}(t)$, (39)

Vidimo da je elastična sila proporcionalna pomaku, sila prigušenja proporcionalna je brzini dok je inercijalna sila po iznosu jednaka produktu mase i ubrzanja te mase. Uzimajući u obzir ove ovisnosti od izraza (36) formiramo diferencijalnu jednadžbu drugog reda:

$$m\ddot{u}(t) + c\dot{u}(t) + ku(t) = P(t). \quad (40)$$

Izraz (40) predstavlja diferencijalnu jednadžbu dinamičke ravnoteže sustava s jednim dinamičkim stupnjem slobode gibanja opterećenog silom $P(t)$. Diferencijalnu jednadžbu zapisati ćemo u eksplicitnom obliku:

$$\ddot{u}(t) = \frac{P(t)}{m} - \frac{ku(t)}{m} - \frac{c\dot{u}(t)}{m} \quad (41)$$

Da bi jednadžba (41) bila pogodna za rješavanje Runge-Kutta metodom moramo prvo smanjiti red diferencijalne jednadžbe. Napraviti ćemo sljedeću supstituciju:

$$\dot{u}(t) = v(t) \quad (42)$$

Uzimajući u obzir uvedenu supstituciju jednadžba (41) sada ima oblik:

$$\dot{v}(t) = \frac{P(t)}{m} - \frac{ku(t)}{m} - \frac{cv(t)}{m} \quad (43)$$

Na ovaj način smo oblikovali sustav diferencijalnih jednadžbi prvog reda, odnosno:

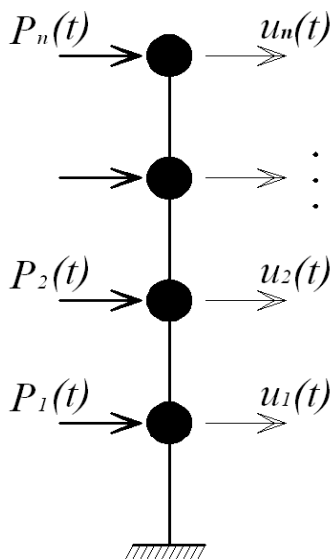
$$\begin{aligned} \dot{u}(t) &= f_1(t, u, v) = v(t) \\ \dot{v}(t) &= f_2(t, u, v) = \frac{P(t)}{m} - \frac{ku(t)}{m} - \frac{cv(t)}{m} \end{aligned} \quad (44)$$

Okako formiran sustav jednadžbi je sada pogodan za rješavanje Runge-Kutta metodom.

3.2. Sustav s više stupnjeva slobode gibanja

Prije objašnjenja koda za rješavanje diferencijalnih jednačbi napravljenog u programu *Wolfram Mathematica 8.0.* obraditi će se formiranje sustava jednačbi kojima se opisuje dinamičko gibanje fizikalnog sustava s više stupnjeva slobode gibanja.

Na slici 9. prikazan je sustav s proizvoljnim brojem n dinamičkih stupnjeva slobode gibanja.



Slika 9. Matematički model dinamičkog sustava s više stupnjeva slobode gibanja

Slično kao i kod sustava s jednim dinamičkim stupnjem slobode gibanja, sustave s više stupnjeva slobode opisujemo matričnom diferencijalnom jednačbom drugog reda:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{P}(t) \quad (45)$$

Gdje su vektor pomaka \mathbf{u} , njegove derivacije te vektor sile $\mathbf{P}(t)$ funkcije vremena, dok su kvadratne matrice \mathbf{M} , \mathbf{C} i \mathbf{K} reda n i određuju se na temelju fizikalnih svojstava sustava. Pošto je za diskretne sustave matrica mase \mathbf{M} dijagonalna, matričnu jednačbu (45) možemo napisati u eksplicitnom obliku:

$$\ddot{\mathbf{u}} = \mathbf{M}^{-1}\mathbf{P}(t) - \mathbf{M}^{-1}\mathbf{K}\mathbf{u} - \mathbf{M}^{-1}\mathbf{C}\dot{\mathbf{u}} \quad (46)$$

Sljedeći korak koji moramo napraviti kako bi gornji sustav jednačbi riješili metodom Runge-Kutta je snižavanje reda sustava diferencijalnih jednačbi što će se napraviti nizom supstitucija napisanih vektorskom jednačbom:

$$\dot{\mathbf{u}} = \mathbf{v} \quad (47)$$

Što nam daje n diferencijalnih jednačbi prvog reda te uvrštavanjem uvedenih supstitucija u sustav (49) imamo:

$$\dot{\mathbf{v}} = \mathbf{M}^{-1}\mathbf{P}(t) - \mathbf{M}^{-1}\mathbf{K}\mathbf{u} - \mathbf{M}^{-1}\mathbf{C}\mathbf{v} \quad (48)$$

Odnosno sada imamo dva sustava jednadžbi:

$$\dot{\mathbf{u}} = \mathbf{f}_1(t, \mathbf{u}, \mathbf{v}) = \mathbf{v}, \quad (49)$$

$$\dot{\mathbf{v}} = \mathbf{f}_2(t, \mathbf{u}, \mathbf{v}) = \mathbf{M}^{-1}\mathbf{P}(t) - \mathbf{M}^{-1}\mathbf{K}\mathbf{u} - \mathbf{M}^{-1}\mathbf{C}\mathbf{v} \quad (50)$$

Ovim supstitucijama smo sustav od n diferencijalnih jednadžbi drugog reda sveli na sustav od $2n$ diferencijalnih jednadžbi prvog reda s $2n+1$ varijabli koji sada možemo riješiti metodom Runge-Kutta kako je to opisano u *poglavlju 2.3*.

U sljedećem poglavlju dan je kod za metodu Runge-Kutta četvrtog reda u programu *Wolfram Mathematica 8.0*. kojim se rješava sustav diferencijalnih jednadžbi drugog reda. Kod je proceduralno programiran tako da je za njegovo razumijevanje dovoljno biti upoznat s prethodnim izlaganjem jer su u zagradama dana dodatna objašnjenja za svaki proceduralni korak. Bitno je naglasiti da je ovaj kod korišten za sve numeričke primjere koji će se pokazati u nastavku jer rješava sve sustave diferencijalnih jednadžbi drugog reda bez obzira na broj dinamičkih stupnjeva slobode n . Za prikaz koda korištene su ulazne vrijednosti iz primjera 4.

3.3. Kod za rješavanje sustava diferencijalnih jednačbi metodom Runge-Kutta

ulaze vrijednosti:

$$\mathbf{M} = \begin{pmatrix} 1.25 & 0 & 0 \\ 0 & 7.5 & 0 \\ 0 & 0 & 2.5 \end{pmatrix}; \quad \mathbf{c} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix};$$
$$\mathbf{K} = \begin{pmatrix} \frac{3515625}{704} & -\frac{234375}{32} & -\frac{703125}{176} \\ -\frac{234375}{32} & \frac{234375}{16} & 0 \\ -\frac{703125}{176} & 0 & \frac{1171875}{44} \end{pmatrix}; \quad \mathbf{pt} = \begin{pmatrix} 0 \\ 0 \\ 20 \end{pmatrix};$$

```
(*ulazne vrijednosti matricne diferencijalne  
jednadzbe drugog reda*)
```

```
n = Length[M]
```

```
(*ocitavanje broja dinamičkih stupnjeva slobode  
iz matrice masa*)
```

```
3
```

formiranje varijabli i funkcija ovisno o stupnjevima slobode sustava :

```
ss = Table[i, {i, n}];
```

```
ss = Map[ToString[#] &, ss];
```

```
(*formiranje broja stupnjeva slobode "ss" = "n"*)
```

```
u = Table["u", {n}];
```

```
(*formiranje varijabli u(t) (pomaka svake mase) *)
```

```
v = Table["v", {n}];
```

```
(*formiranje varijabli v(t) (brzine svake mase)*)
```

```
u = Table[StringJoin[u[[i]], ss[[i]]], {i, n}];
```

```
v = Table[StringJoin[v[[i]], ss[[i]]], {i, n}];
```

```
var = Join[u, v];
```

```
var = Prepend[var, "t"];
```

```
(*formiranje liste varijabli t,u,v *)
```

```
var = Table[StringJoin[var[[i]], "_"], {i, 2 n + 1}];
```

```
bf = Table[i, {i, 2 n}];
```

```
(*formiranje broja funkcija za rješavanje  
sustava jednačbi*)
```

```
bf = Map[ToString[#] &, bf];
```

```
f = Table["f", {2 n}];
```

```
f = Table[StringJoin[f[[i]], bf[[i]]], {i, 2 n}];
```

```

f = Map[ToExpression[#] &, f]
(*pretvorba teksta u varijable*)
u = Map[ToExpression[#] &, u]
(*pretvorba teksta u varijable*)
v = Map[ToExpression[#] &, v]
(*pretvorba teksta u varijable*)
var = Map[ToExpression[#] &, var]
(*pretvorba teksta u varijable *)

d = Flatten[Inverse[M].pt - Inverse[M].c.v -
  Inverse[M].K.u];
(*određivanje produkta - desna strana izraza (48)*)

Do[f[[i]][Apply[Sequence, var]] :=
  (v[[i]] // Evaluate), {i, 1, n]];
(*pridruživanje vrijednosti funkcijama -
  izraz (49)*)
Do[f[[i]][Apply[Sequence, var]] :=
  (d[[i - n]] // Evaluate), {i, n + 1, 2 n]];
(*pridruživanje vrijednosti funkcijama -
  izraz (50)*)

{f1, f2, f3, f4, f5, f6}

{u1, u2, u3}

{v1, v2, v3}

{t_, u1_, u2_, u3_, v1_, v2_, v3_}

```

početni uvjeti:

```

t[0] = 0;
Do[u[[i]][0] = 0, {i, 1, n]];
(*uvršavanje početnih uvjeta*)
Do[v[[i]][0] = 0, {i, 1, n]];

```

vremenski korak:

```

h = 0.005; (*odabir vremenskog koraka*)

RKu1 = ConstantArray[0, {3000, 2}];
RKu2 = ConstantArray[0, {3000, 2}];
RKu3 = ConstantArray[0, {3000, 2}];
(*formiranje listi u koje će se zapisivati
rješenja za svaki korak*)
RKv1 = ConstantArray[0, {3000, 2}];
RKv2 = ConstantArray[0, {3000, 2}];
RKv3 = ConstantArray[0, {3000, 2}];

```

Runge - Kutta metoda četvrtog stupnja :

```

Do[t[j + 1] = t[j] + h;

pom1 = Table[u[[i]][j], {i, 1, n}];
pom2 = Table[v[[i]][j], {i, 1, n}];
pom = Join[pom1, pom2];
pom = Prepend[pom, t[j]];
(*pridruživanje vrijednosti funkcijama f*)
k1 = Table[0, {i, 2 n}];
(*definiranje vektora koeficijenta k1*)
Do[k1[[i]] = h * f[[i]][Apply[Sequence, pom]],
  {i, 1, 2 n}]; (*računanje koeficijenta k1 za
pojedini korak*)

pom1 = Table[u[[i]][j] +  $\frac{1}{2}$  k1[[i]], {i, 1, n}];
pom2 = Table[v[[i]][j] +  $\frac{1}{2}$  k1[[n + i]], {i, 1, n}];
pom = Join[pom1, pom2];
pom = Prepend[pom, t[j] +  $\frac{1}{2}$  h];
(*pridruživanje vrijednosti funkcijama f*)
k2 = Table[0, {i, 2 n}];
(*definiranje vektora koeficijenta k2*)
Do[k2[[i]] = h * f[[i]][Apply[Sequence, pom]],

```

```

    {i, 1, 2 n}]; (*računanje koeficijenta k2 za
    pojedini korak*)

    pom1 = Table[u[[i]][j] +  $\frac{1}{2}$  k2[[i]], {i, 1, n}];
    pom2 = Table[v[[i]][j] +  $\frac{1}{2}$  k2[[n + i]], {i, 1, n}];
    pom = Join[pom1, pom2];
    pom = Prepend[pom, t[j] +  $\frac{1}{2}$  h];

    (*pridruživanje vrijednosti funkcijama f*)
    k3 = Table[0, {i, 2 n}];
    (*definiranje vektora koeficijenta k3*)
    Do[k3[[i]] = h * f[[i]][Apply[Sequence, pom]],
    {i, 1, 2 n}]; (*računanje koeficijenta k3 za
    pojedini korak*)

    pom1 = Table[u[[i]][j] + k3[[i]], {i, 1, n}];
    pom2 = Table[v[[i]][j] + k3[[n + i]], {i, 1, n}];
    pom = Join[pom1, pom2];
    pom = Prepend[pom, t[j] + h];

    (*pridruživanje vrijednosti funkcijama f*)
    k4 = Table[0, {i, 2 n}];
    (*definiranje vektora koeficijenta k4*)
    Do[k4[[i]] = h * f[[i]][Apply[Sequence, pom]],
    {i, 1, 2 n}]; (*računanje koeficijenta k4 za
    pojedini korak*)

    Do[u[[i]][j + 1] =
    u[[i]][j] +
     $\frac{1}{6}$  (k1[[i]] + 2 k2[[i]] + 2 k3[[i]] + k4[[i]]),
    {i, 1, n}]; (*određivanje vrijednosti
    vektora pomaka na kraju koraka*)
    Do[v[[i]][j + 1] =
    v[[i]][j] +

```

```


$$\frac{1}{6} (k1[[n+i]] + 2 k2[[n+i]] + 2 k3[[n+i]] + k4[[n+i]]), \{i, 1, n\}];$$

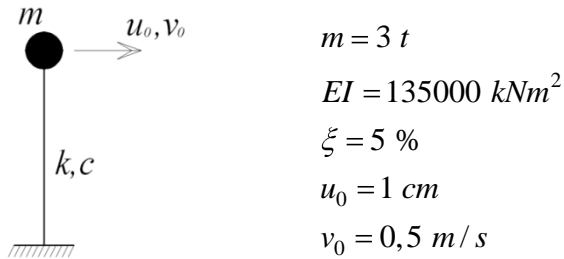
(*određivanje vrijednosti vektora brzine na kraju koraka*)

RKu1[[j+1]] = List[t[j], u[[1]][j]];
RKu2[[j+1]] = List[t[j], u[[2]][j]];
RKu3[[j+1]] = List[t[j], u[[3]][j]];
RKv1[[j+1]] = List[t[j], v[[1]][j]];
RKv2[[j+1]] = List[t[j], v[[2]][j]];
RKv3[[j+1]] = List[t[j], v[[3]][j]]; , {j, 0, 2999} ]
(*upisivanje određenih vrijednosti u liste rješenja*)

```


3.4. Numerički primjeri iz područja dinamike konstrukcija

Primjer 3. Potrebno je odrediti zakon slobodnih oscilacija prikazanog sustava s jednim dinamičkim stupnjem slobode gibanja koje će nastati ako se masi m zadaju početni uvjeti.



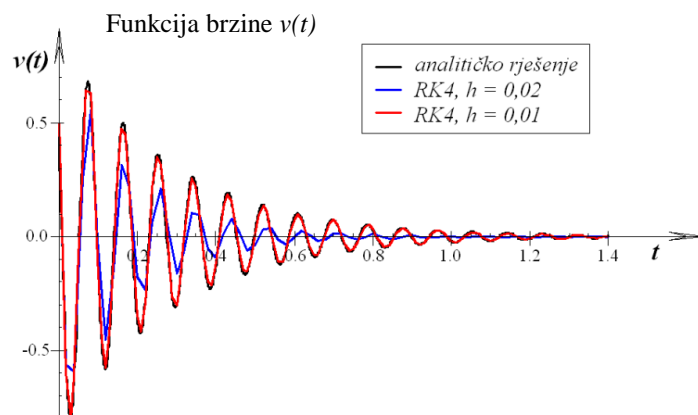
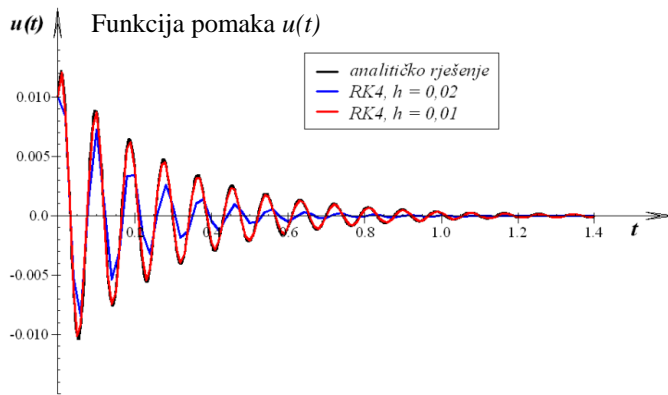
Analitičko rješenje linearno elastičnog sustava je:

$$k = \frac{3EI}{L^3} = 15000 \text{ kN/m} \quad \omega_d = \omega \sqrt{1 - \xi^2}$$

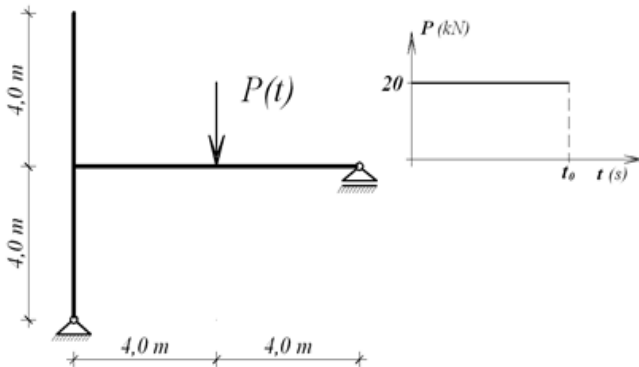
$$\omega = \sqrt{\frac{k}{m}} = 70,71 \text{ r/s} \quad \xi = \frac{c}{c_{kr}} \rightarrow c = \xi \cdot c_{kr} = \xi \cdot 2m\omega$$

$$T = \frac{2\pi}{\omega} = 0,089 \text{ s} \quad x(t) = \left[x_0 \cos(\omega_d t) + \frac{\dot{x}_0 + x_0 \xi \omega}{\omega_d} \sin(\omega_d t) \right] \cdot e^{-\xi \omega t}$$

Grafički prikaz rješenja dobivenog metodom RK4 i analitičkog rješenja:



Primjer 4. Potrebno je odrediti zakon oscilacija vrha stupa prikazanog sustava koje će nastati uslijed djelovanja sile $P(t)$ uz zanemarenje prigušenja konstrukcije.



Fizikalne karakteristike sustava:

$b/h = 50/50$ - poprečni presjek stupa i grede

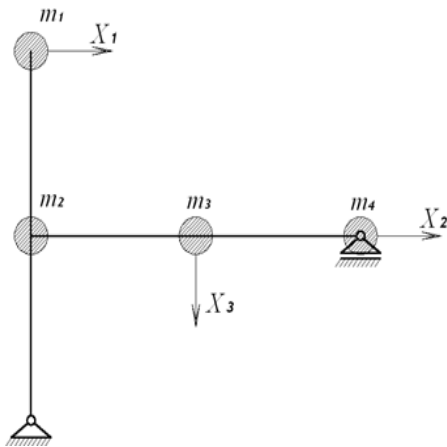
C25/30 - beton

$$E = 3 \cdot 10^7 \text{ kN} / \text{m}^2$$

$$I = \frac{b \cdot h^3}{12} = \frac{1}{192} \text{ m}^4$$

$$m = 2,5 \text{ t} / \text{m}^3 \rightarrow m = 0,625 \text{ t} / \text{m}$$

Da bi ovaj kontinuirani sustav proračunali potrebno je prvo formirati diskretni matematički model. Uzimajući u obzir dominantne dinamičke pomake ovog sustava u ravni koncentriramo mase u točke i oblikujemo matricu mase te iz elastičnih svojstava sustava matricu krutosti.



$$m_1 = 2 \cdot 0,625 = 1,25 \text{ t}$$

$$m_2 = 2 + 2 + 2 \cdot 0,625 = 3,75 \text{ t}$$

$$m_3 = 4 \cdot 0,625 = 2,5 \text{ t}$$

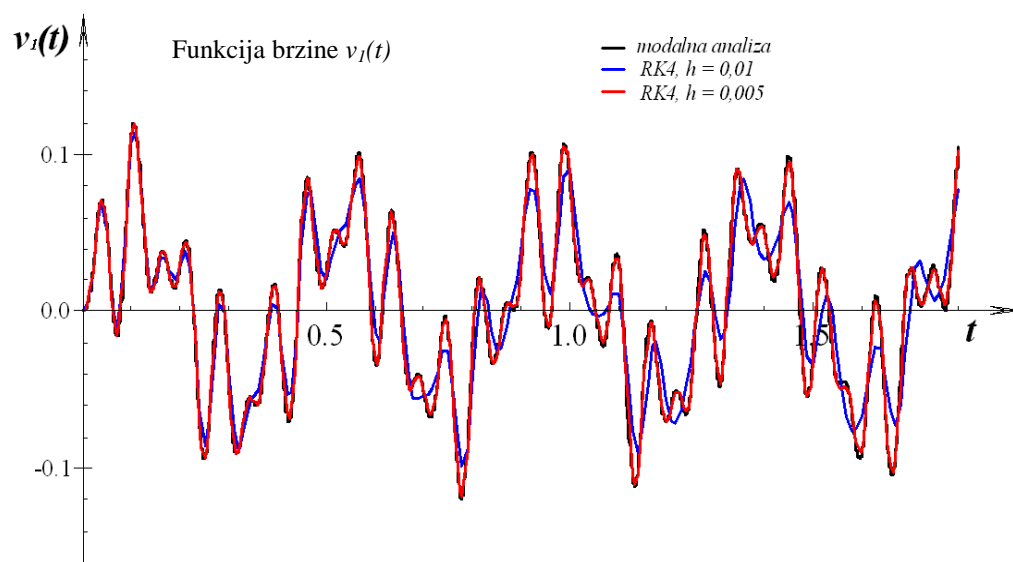
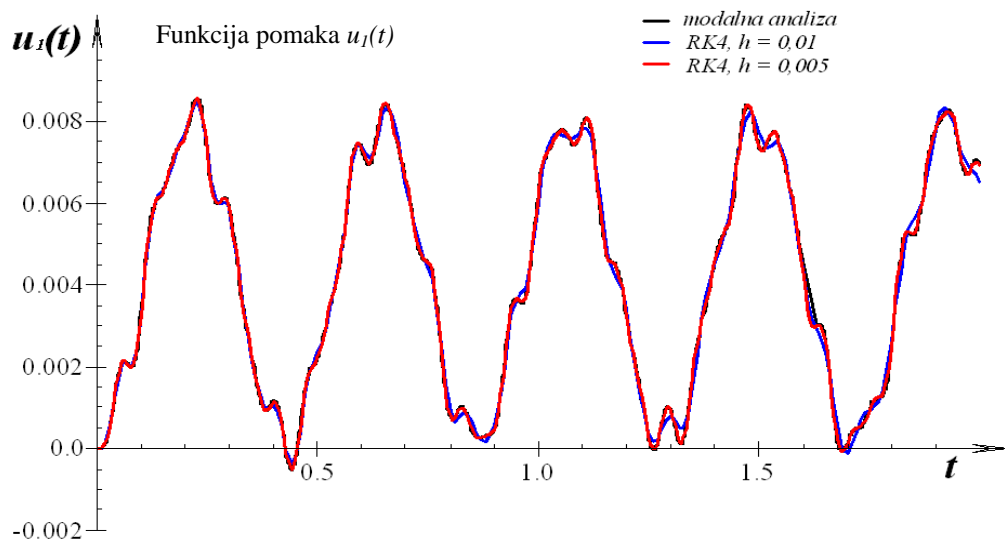
$$m_4 = 2 \cdot 0,625 = 1,25 \text{ t}$$

$$M = \begin{bmatrix} 1,25 & 0 & 0 \\ 0 & 7,5 & 0 \\ 0 & 0 & 2,5 \end{bmatrix}$$

$$K = \begin{bmatrix} \frac{45EI}{1408} & -\frac{3EI}{64} & -\frac{9EI}{352} \\ -\frac{3EI}{64} & \frac{3EI}{32} & 0 \\ -\frac{9EI}{352} & 0 & \frac{15EI}{88} \end{bmatrix}$$

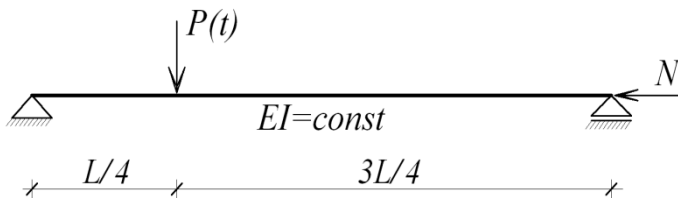
Proračun ovog primjera dan je detaljnim opisom matematičkog koda na stranici 19. Radi usporedbe na grafičkoj ilustraciji rješenja dano je i rješenje odziva sustava napravljeno modalnom analizom.

Grafički prikaz rješenja, odnosno pomaka i brzine mase m_1 u intervalu $0 \text{ s} \leq t \leq 2 \text{ s}$:



Primjer 5. Potrebno je odrediti zakon oscilacija u sredini raspona betonske grede koje će nastati uslijed djelovanja sile $P(t)$ za različite vrijednosti uzdužne sile N . Ako je sila $P(t)$:

- a) $P(t) = 10$ [kN]
 b) $P(t) = 10 \cdot \sin(400 \cdot t)$ [kN]

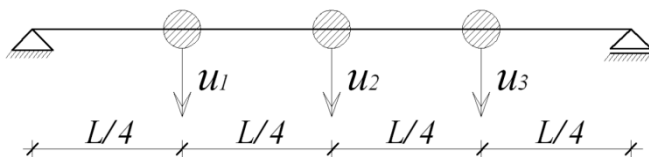


$$L = 4 \text{ m}$$

$$\bar{m} = 0,5 \text{ t/m'}$$

$$EI = 156000 \text{ kNm}^2$$

Prvo oblikujemo matematički diskretan model, sustav s tri stupnja slobode gibanja:



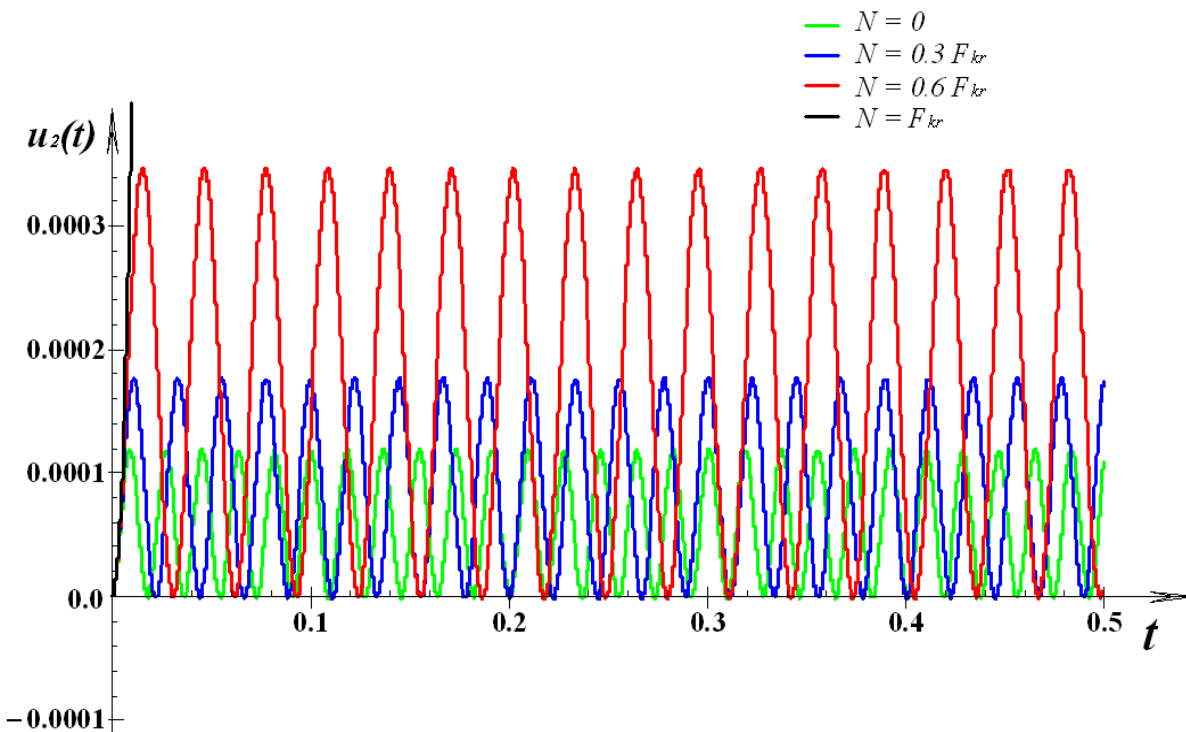
Cilj ovog zadatka je promotriti ponašanje sustava pod dinamičkim opterećenjem za različite vrijednosti uzdužne tlačne sile. Pošto linijski elementi ovisno o vrijednosti uzdužne sile mijenjaju svoja konstrukcijska svojstva, tj. vlačna sila povećava krutost na savijanje dok tlačna sila smanjuje krutost sustava. Kod djelovanja tlačne uzdužne sile javlja se problem stabilnosti elementa. Prikazani konstrukcijski sustav izgubit će stabilnost ukoliko je vrijednost uzdužne tlačne sile jednaka vrijednosti kritične sile izvijanja. Za slobodno oslonjenu gredu poznata je vrijednost krizične sile i ona iznosi:

$$F_{kr} = \frac{\pi^2 EI}{L^2}$$

Ovdje ćemo promotriti što se dešava s dinamičkim odzivom prikazanog sustava ako je uzdužna sila N manja ili jednaka kritičnoj vrijednosti F_{kr} .

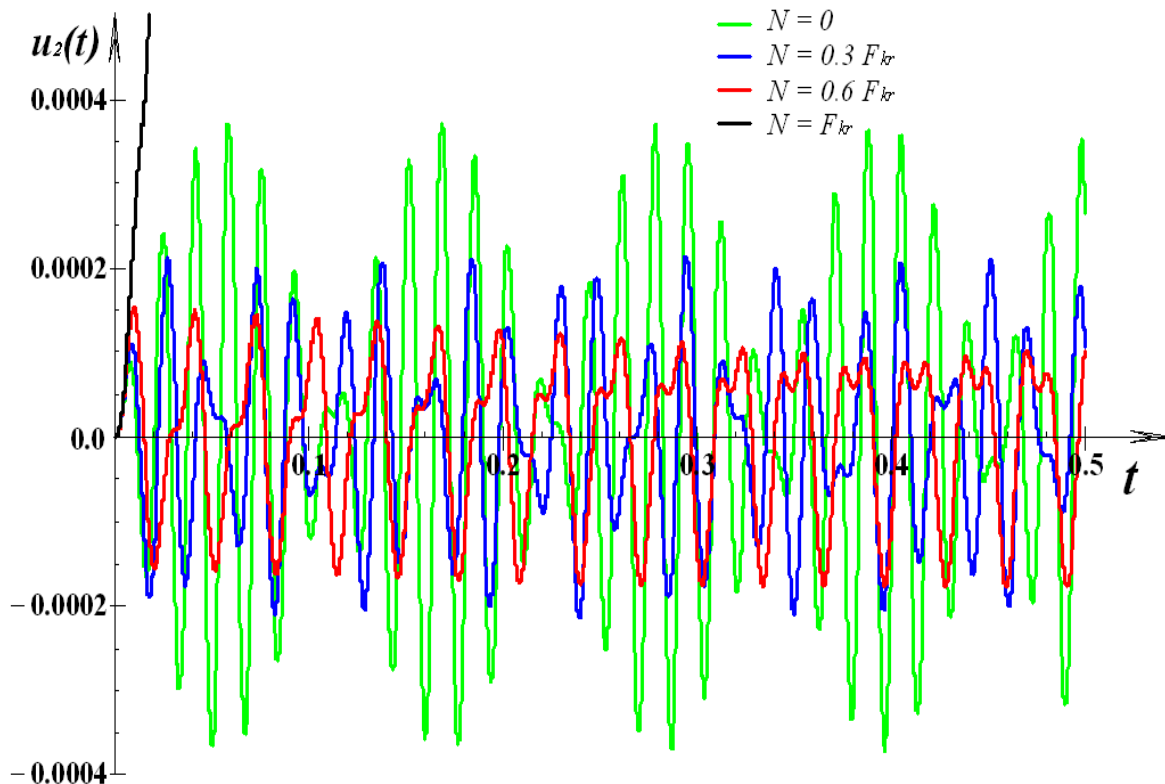
Vremenski korak za metodu Runge-Kutta je za sve jednak i iznosi $h=0,0001$. Ovako mali korak za provedbu metode RK je odabran iz razloga velike krutosti sustava (mali periodi oscilacija).

a) $P(t) = 10 \text{ [kN]}$



Na gornjoj slici je prikazana funkcija pomaka točke u sredini grede. Vidimo da kad je uzdužna sila jednaka nuli da je sustav najkrući, odnosno ima najmanje pomake i najveću frekvenciju oscilacija. Povećajem uzdužne sile sustav postaje mekši, odnosno pomaci su mu sve veći dok periodi postaju sve duži. Kada uzdužna sila dosegne vrijednost kritične sile izvijanja sustav gubi stabilnost. Dinamički karakter ovog problema je da u trenutku gubitka stabilnosti periodi konstrukcije postaju beskonačno veliki odnosno frekvencija je jednaka nuli. Svojstvo da gibanje sustava više nema oscilirajući karakter vidimo i rješavanjem sustava po RK4 metodi u kojoj pomaci postaju beskonačno veliki što nas upućuje na to da je dosegnuta kritična sila izvijanja.

a) $P(t) = 10 \cdot \sin(400 \cdot t)$ [kN]



Prikazani dijagrami daju odziv sustava (točke u sredini raspona grede) na djelovanje harmonijske sile koja ima jednaku veličinu amplitude kao i sila u prethodnom primjeru. Pošto je frekvencija sile približno jednaka prvoj frekvenciji sustava dolazi do odziva koji nazivamo udaranje (eng. beating). Kako frekvencije sustava i sile međusobno dolaze u fazu uočava se značajno povećanje pomaka kod odziva sustava kojemu je vrijednost uzdužne tlačne sile jednaka nuli (zeleno krivulja). Povećanjem vrijednosti uzdužne sile dolazi do promjene fizikalnih svojstava sustava, kao i u prethodnom primjeru sustav postaje mekši te mu se također mijenjaju dinamička svojstva kao što je frekvencija sustava. Radi razdvajanja osnovne frekvencije sustava i frekvencije sile vrijednosti pomaka postaju manje. No, u trenutku kada vrijednost uzdužne sile dosegne vrijednost kritične sile izvijanja sustav će izgubiti stabilnost.

4. Zaključak

U ovome seminaru objašnjena je teorijska podloga Runge-Kutta metode koja se pokazala vrlo pogodnom za rješavanje problema koji se javljaju u dinamici konstrukcija. Najveća prednost Runge-Kutta metode je u tome da je potpuno nebitno da li je funkcija $f(x,y)$ u izrazu (1) linearna ili nelinearna. Odnosno nije potrebno koristiti neku od metoda za rješavanje nelinearnih jednadžbi jer se direktno iz funkcije $f(x,y)$ računaju koeficijenti za aproksimaciju funkcije $y(x)$.

Kod koji je napisan u programu *Wolfram Mathematica 8.0.* se može dodatno prilagoditi rješavanju dinamičkih sustava, npr. veličinu koraka h trebalo bi odabirati ovisno o veličini osnovnog perioda sustava kako bi se izbjegla nerealna rješenja što se može dogoditi ukoliko je korak veći ili je približno jednak osnovnim periodima oscilacija sustava.

Za kraj se još može spomenuti da danas postoji cijela familija Runge-Kutta metoda čiji je jednostavan algoritam posebno pogodan za provedbu proračuna uz pomoć računala.

5. Literatura

1. Clough, R.W., Penzien, J. (2003), Dynamics of Structures-3rd Edition, McGraw-Hill Book Co., New York
2. Chopra, A.K.(2001), Dynamics of Structures: Theory and Applications to Earthquake Engineering, Prentice-Hall
3. Bronstein, I.N., Semendjajew, K.A., Musiol G., Mühlig H. (2004), Matematički priručnik, Golden marketing – Tehnička knjiga, Zagreb
4. Kreyszig, E., Advanced Engineering Mathematics (2006), John Wiley and Sons, Singapore
5. Raduka, V., Dinamika konstrukcija i postresno inženjerstvo, Bilješke s predavanja