

Uvod u simulaciju sistema u energetskej elektronici

Predrag Pejović

13. mart 2011

1 Diskretizacija

Pretpostavimo da je potrebno da numerički rešimo (simuliramo) diferencijalnu jednačinu oblika

$$\frac{dx}{dt} = f(x, t)$$

za koju je dat odgovarajući početni uslov. Jednačina je skalarna (po jednoj promenljivoj), prvog reda, u normalnoj formi. Da bi gornja jednačina, data u kontinualnom vremenu, bila simulirana pomoću digitalnog računara, potrebno je preći na diskretizovano vreme

$$t_n = n \Delta t$$

pošto digitalni računar može da da podatke o odzivu samo u diskretnim trenucima vremena. Prelazak na diskretizovano vreme zahteva zamenu polazne diferencijalne jednačine diferencnom jednačinom. Zamena diferencijalne jednačine diferencnom se zove diskretizacija.

Razmatraćemo tri metoda diskretizacije. Kod svih metoda koji će ovde biti razmatrani, izvod će biti **aproksimiran** sa

$$\frac{dx}{dt} = \frac{x_{n+1} - x_n}{\Delta t}$$

Ovo je samo jedan od mogućih načina aproksimacije izvoda.

2 *Forward-Euler*

Prvi metod koji će biti razmatran je *forward-Euler*. Kod njega se funkcija tokom razmatranog vremenskog koraka aproksimira svojom vrednošću na početku koraka

$$\frac{x_{n+1} - x_n}{\Delta t} = f(x_n, t_n) = f_n.$$

Rešenje ovako dobijene diferencne jednačine u diskretnim tačkama vremena se **neće** poklapati sa odbircima rešenja polazne diferencijalne jednačine. U cilju poređenja ta dva rešenja u dugom periodu vremena (ne na nivou jednog vremenskog koraka simulacije, takva analiza se zove „analiza greške numeričke integracije“) razmatraćemo oblast stabilnosti rešenja polazne diferencijalne jednačine i korespodentne diferencne jednačine ako je polazna diferencijalna jednačina linearna, homogena, sa konstantnim koeficijentom

$$\frac{dx}{dt} = \lambda x.$$

Ova jednačina ima rešenje

$$x(t) = x(0) e^{\lambda t}.$$

Ako dopustimo da je λ kompleksan broj, sa

$$\Re(\lambda) = \lambda_r$$

i

$$\Im(\lambda) = \lambda_i$$

tj. ako je

$$\lambda = \lambda_r + j \lambda_i$$

tada je

$$x(t) = x(0) e^{\lambda_r t} (\cos(\lambda_i t) + j \sin(\lambda_i t))$$

pa je rešenje stabilno za

$$\lambda_r < 0.$$

Primenom diskretizacije po *forward-Euler* metodu dobija se

$$\frac{x_{n+1} - x_n}{\Delta t} = \lambda x_n$$

odnosno

$$x_{n+1} = x_n + \lambda \Delta t x_n = (1 + \lambda \Delta t) x_n.$$

Sukcesivna primena dobijene diferencne jednačine daje

$$\begin{aligned} x_1 &= (1 + \lambda \Delta t) x_0 \\ x_2 &= (1 + \lambda \Delta t) x_1 = (1 + \lambda \Delta t)^2 x_0 \\ x_3 &= (1 + \lambda \Delta t) x_2 = (1 + \lambda \Delta t)^3 x_0 \\ &\dots \\ x_n &= (1 + \lambda \Delta t) x_{n-1} = (1 + \lambda \Delta t)^n x_0 \\ &\dots \end{aligned}$$

odakle se rešenje dobijene diferencne jednačine u zatvorenoj formi identifikuje kao

$$x_n = (1 + \lambda \Delta t)^n x_0.$$

Rešenje dobijene diferencne jednačine je stabilno ako

$$\lim_{n \rightarrow \infty} (1 + \lambda \Delta t)^n = 0$$

što je zadovoljeno kada je

$$|1 + \lambda \Delta t| < 1.$$

Kako je

$$\Re(\lambda) = \lambda_r$$

i

$$\Im(\lambda) = \lambda_i$$

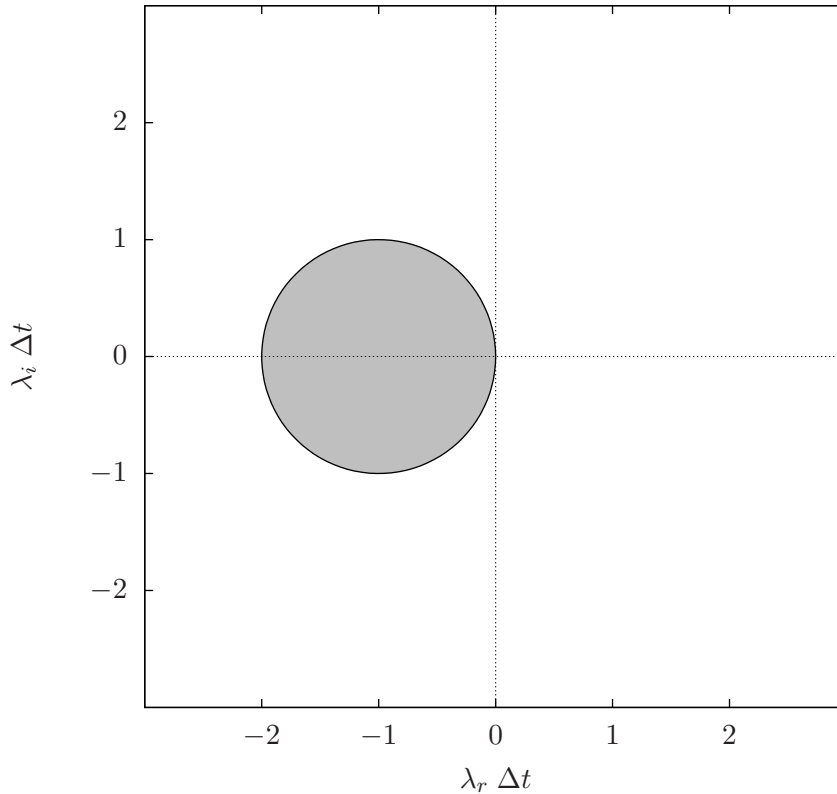
dobija se

$$|1 + \lambda_r \Delta t + j \lambda_i \Delta t| < 1$$

što u razvijenoj formi ima oblik

$$(1 + \lambda_r \Delta t)^2 + (\lambda_i \Delta t)^2 < 1$$

Dobijena oblast u λ ravni predstavlja unutrašnjost kruga sa centrom u $\lambda = -\frac{1}{\Delta t}$ poluprečnika $\frac{1}{\Delta t}$, kako je prikazano na slici 1.



Slika 1: Oblast stabilnosti *forward-Euler* algoritma

3 *Backward-Euler*

Drugi metod koji će biti razmatran je *backward-Euler*. Kod njega se funkcija tokom razmatranog vremenskog koraka aproksimira svojom vrednošću na kraju koraka

$$\frac{x_{n+1} - x_n}{\Delta t} = f(x_{n+1}, t_{n+1}) = f_{n+1}.$$

Vrednost x_{n+1} je nepoznata, a javlja se sa leve strane jednačine, u aproksimaciji izvoda, kao i sa desne strane jednačine, kao argument funkcije $f(x, t)$. Ova osobina svrstava *backward-Euler* metod u implicitne metode, za razliku od metoda kod kojih se nepoznata vrednost x_{n+1} javlja samo sa leve strane, nikako kao argument funkcije $f(x, t)$, koji su eksplicitni. Prethodno razmatran *forward-Euler* metod je eksplicitan. Implicitni metodi mogu da budu teški za implementaciju zbog potrebe da se rešava algebarska jednačina po x_{n+1} , što je u nekim slučajevima težak numerički problem, sklon problemima konvergencije. Ovde ćemo analizirati linearne jednačine, pa određivanje x_{n+1} po implicitnim metodima neće biti problem.

U cilju poređenja sa ostalim algoritmima, razmatraćemo oblast stabilnosti rešenja korespondentne diferencne jednačine dobijene po *backward-Euler* metodu ako je polazna diferencijalna jednačina linearna, homogena, sa konstantnim koeficijentom, ista kao i u slučaju razmatranja oblasti stabilnosti *forward-Euler* metoda

$$\frac{dx}{dt} = \lambda x.$$

Primenom diskretizacije po *backward-Euler* metodu dobija se

$$\frac{x_{n+1} - x_n}{\Delta t} = \lambda x_{n+1}$$

odnosno

$$x_{n+1} = x_n + \Delta t \lambda x_{n+1}$$

odakle se zbog linearnosti jednačine x_{n+1} lako određuje kao

$$x_{n+1} = \frac{x_n}{1 - \lambda \Delta t}$$

Sukscesivna primena dobijene diferencne jednačine daje

$$\begin{aligned}x_1 &= (1 - \lambda \Delta t)^{-1} x_0 \\x_2 &= (1 - \lambda \Delta t)^{-1} x_1 = (1 - \lambda \Delta t)^{-2} x_0 \\x_3 &= (1 - \lambda \Delta t)^{-1} x_2 = (1 - \lambda \Delta t)^{-3} x_0 \\&\dots \\x_n &= (1 - \lambda \Delta t)^{-1} x_{n-1} = (1 - \lambda \Delta t)^{-n} x_0 \\&\dots\end{aligned}$$

odakle se rešenje dobijene diferencne jednačine u zatvorenoj formi identifikuje kao

$$x_n = (1 - \lambda \Delta t)^{-n} x_0.$$

Rešenje dobijene diferencne jednačine je stabilno ako

$$\lim_{n \rightarrow \infty} (1 - \lambda \Delta t)^{-n} = 0$$

što je zadovoljeno kada je

$$|1 - \lambda \Delta t| > 1.$$

Pod istim pretpostavkama kao i u slučaju analize *forward-Euler* metoda dobija se

$$|1 - \lambda_r \Delta t - j \lambda_i \Delta t| > 1$$

što u razvijenoj formi ima oblik

$$(1 - \lambda_r \Delta t)^2 + (\lambda_i \Delta t)^2 > 1$$

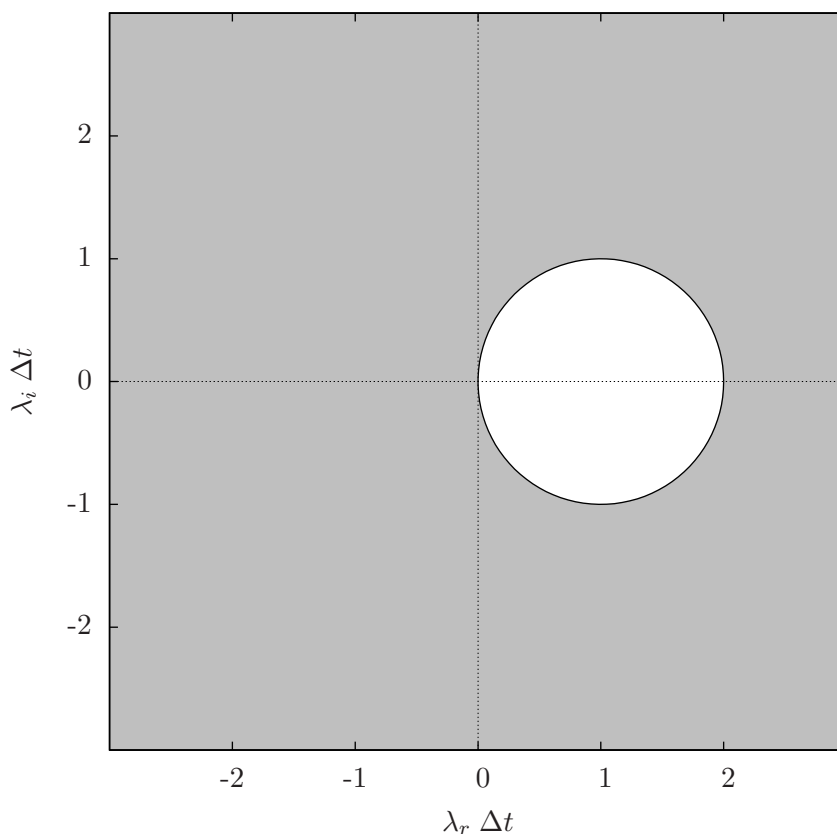
Dobijena oblast u λ ravni predstavlja spoljašnjost kruga sa centrom u $\lambda = \frac{1}{\Delta t}$ poluprečnika $\frac{1}{\Delta t}$, kako je prikazano na slici 2. Dobijena oblast stabilnosti je znatno veća nego u slučaju *forward-Euler* metoda, ali se i dalje ne poklapa sa oblašću stabilnosti polazne diferencijalne jednačine. Eventualne numeričke greške nastale usled konačne dužine reči *backward-Euler* metod će prigušiti znatno bolje nego *forward-Euler* metod, ali će osim toga prigušiti i stvarno rešenje, što će biti ilustrovano u primeru koji će biti prikazan.

4 Trapezno pravilo

Treći metod koji će biti razmatran je trapezno pravilo. Kod trapeznog pravila se funkcija tokom vremenskog koraka aproksimira aritmetičkom sredinom svojih vrednosti na početku i na kraju posmatranog vremenskog koraka

$$\frac{x_{n+1} - x_n}{\Delta t} = \frac{1}{2} (f(x_n, t_n) + f(x_{n+1}, t_{n+1})) = \frac{1}{2} (f_n + f_{n+1}).$$

Vrednost x_{n+1} je nepoznata, a javlja u aproksimaciji izvoda i kao argument funkcije $f(x, t)$. Ova osobina svrstava trapezno pravilo, isto kao i *backward-Euler* metod, u implicitne metode.



Slika 2: Oblast stabilnosti *backward-Euler* algoritma

Ako je funkcija $f(x, t)$ linearna, primena implicitnih metoda ne uzrokuje veće teškoće. Međutim, ako je $f(x, t)$ nelinearna funkcija, u svakom vremenskom koraku simulacije neophodno je rešavanje nelinearne jednačine ili sistema nelinearnih jednačina. Ovaj zahtev predstavlja dodatno računsko opterećenje, a može biti i uzrok problema ako iterativni postupak za rešavanje nelinearnih jednačina nema zagarantovanu konvergenciju.

U cilju poređenja sa ostalim algoritmima, razmatraćemo oblast stabilnosti rešenja korepodentne diferencne jednačine dobijene po trapeznom pravilu ako je polazna diferencijalna jednačina linearna, homogena, sa konstantnim koeficijentom, ista kao i u slučaju razmatranja oblasti stabilnosti Ojlerovih metoda

$$\frac{dx}{dt} = \lambda x.$$

Primenom diskretizacije po trapeznom pravilu dobija se

$$\frac{x_{n+1} - x_n}{\Delta t} = \frac{1}{2} \lambda (x_n + x_{n+1})$$

odnosno

$$x_{n+1} \left(1 - \frac{\lambda \Delta t}{2} \right) = x_n \left(1 + \frac{\lambda \Delta t}{2} \right)$$

odakle se zbog linearnosti jednačine x_{n+1} lako određuje kao

$$x_{n+1} = \frac{1 + \frac{\lambda \Delta t}{2}}{1 - \frac{\lambda \Delta t}{2}} x_n$$

Sukcesivna primena dobijene diferencne jednačine daje rešenje u zatvorenoj formi

$$x_n = \left(\frac{1 + \frac{\lambda \Delta t}{2}}{1 - \frac{\lambda \Delta t}{2}} \right)^n x_0.$$

Rešenje je stabilno ako

$$\lim_{n \rightarrow \infty} \left(\frac{1 + \frac{\lambda \Delta t}{2}}{1 - \frac{\lambda \Delta t}{2}} \right)^n = 0$$

što je zadovoljeno kada je

$$\left| \frac{1 + \frac{\lambda \Delta t}{2}}{1 - \frac{\lambda \Delta t}{2}} \right| < 1.$$

Pod pretpostavkom da λ može biti kompleksan broj, kao i u prethodnim slučajevima, dobija se

$$\frac{\left| 1 + \frac{\lambda_r \Delta t}{2} + j \frac{\lambda_i \Delta t}{2} \right|}{\left| 1 - \frac{\lambda_r \Delta t}{2} - j \frac{\lambda_i \Delta t}{2} \right|} < 1$$

što se svodi na

$$\left| 1 + \frac{\lambda_r \Delta t}{2} + j \frac{\lambda_i \Delta t}{2} \right| < \left| 1 - \frac{\lambda_r \Delta t}{2} - j \frac{\lambda_i \Delta t}{2} \right|.$$

U razvijenoj formi gornja jednačina ima oblik

$$(1 + \lambda_r \Delta t)^2 + (\lambda_i \Delta t)^2 < (1 - \lambda_r \Delta t)^2 + (\lambda_i \Delta t)^2$$

što se svodi na

$$(1 + \lambda_r \Delta t)^2 < (1 - \lambda_r \Delta t)^2.$$

Razvojem kvadrata binoma dobija se

$$1 + 2 \lambda_r \Delta t + (\lambda_r \Delta t)^2 < 1 - 2 \lambda_r \Delta t + (\lambda_r \Delta t)^2$$

što se svodi na

$$\lambda_r < 0$$

jer $\Delta t > 0$. Oblast stabilnosti dobijene diferencne jednačine se poklapa sa oblašću stabilnosti polazne diferencijalne jednačine. Oblast stabilnosti je prikazana na slici 3.

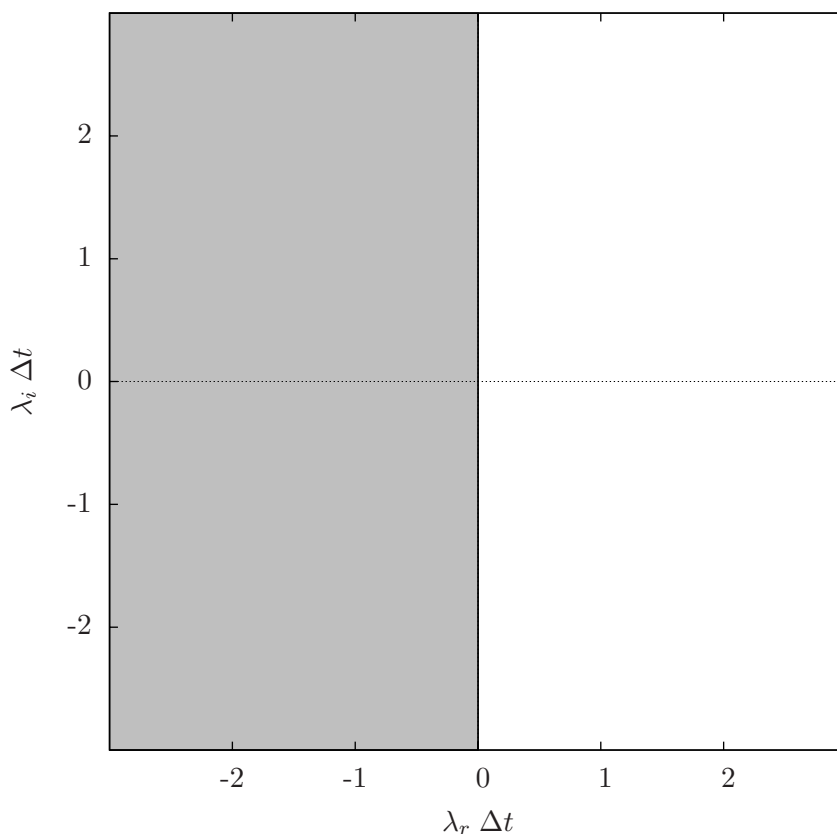
5 Primer

U cilju ilustrovanja numeričkog rešavanja diferencijalnih jednačina, razmatraćemo kao primer paralelno oscilatorno kolo prikazano na slici 4. Kolo sa slike 4 je autonomno pošto nema nezavisne generatore. Autonomna kola imaju samo sopstveni odziv, pošto zbog odsustva nezavisnih izvora nemaju prinudni odziv. Diferencijalne jednačine koje opisuju autonomna kola su homogene.

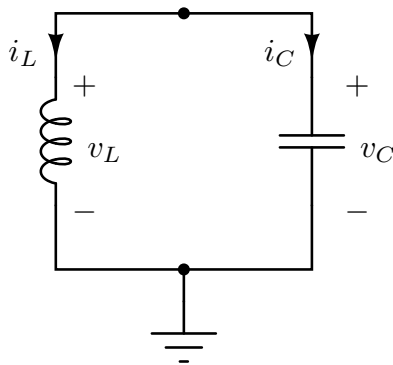
U analizi kola sa slike 4 smatraćemo da su vrednosti elemenata $L = 1 \mu\text{H}$ i $C = 1 \mu\text{F}$, što daje rezonantnu kružnu frekvenciju $\omega_0 = \frac{1}{\sqrt{LC}} = 1 \text{ M} \frac{\text{rad}}{\text{s}}$ i karakterističnu otpornost oscilatornog kola $R_0 = \sqrt{\frac{L}{C}} = 1 \Omega$. Simulacijom će biti određen odziv kola za početne uslove $v_C(0) = 10 \text{ V}$ i $i_L(0) = 0$.

Pri zadatim uslovima odziv razmatranog kola se analitički dobija kao

$$v_C(t) = 10 \text{ V} \cos \frac{t}{1 \mu\text{s}}$$



Slika 3: Oblast stabilnosti trapeznog pravila



Slika 4: Primer, LC oscilatorno kolo

i

$$i_L(t) = 10 \text{ A} \sin \frac{t}{1 \mu\text{s}}.$$

Sistem jednačina stanja kola sa slike 4 je

$$\frac{d}{dt} \begin{bmatrix} v_C \\ i_L \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{C} \\ \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} v_C \\ i_L \end{bmatrix}.$$

Vremenski korak za simulacije čiji će rezultati biti prikazani u ovom tekstu izabran je kao $\Delta t = 0.2 \mu\text{s}$. Izbor je motivisan ciljem da se jasno ilustruju osobine predstavljenih algoritama, kao i da se algoritmi uporede. U prilogu je dat izvorni kod Octave programa za simulaciju, a čitaocima se preporučuje da sami probaju nekoliko većih i manjih koraka od izabranog kako bi stekli osećaj za uticaj izbora vremenskog koraka na rezultat simulacije.

Sopstvene frekvencije kola su $\lambda_{1,2} = \pm \frac{j}{\sqrt{LC}} = \pm j M_s^{\text{rad}}$, što daje proizvode $\lambda_{1,2} \Delta t = \pm 0.2 j$. Sopstvene frekvencije kola i izabrani vremenski korak daju proizvode koji su u oblasti nestabilnosti *forward-Euler* algoritma, stabilnosti *backward-Euler* algoritma, i na granici stabilnosti za trapezno pravilo diskretizacije. Simulirani sistem je granično stabilan.

Jednačine stanja autonomnog sistema imaju oblik

$$\frac{d}{dt} \mathbf{x} = \mathbf{A} \mathbf{x}.$$

Generalizacijom prethodnih izvođenja koja su se odnosila na skalarne funkcije na višedimenzioni slučaj, po *forward-Euler* diskretizacionom pravilu se dobija diferencna jednačina

$$\mathbf{x}_{n+1} = (\mathbf{I} + \Delta t \mathbf{A}) \mathbf{x}_n$$

po *backward-Euler* diskretizacionom pravilu se dobija

$$\mathbf{x}_{n+1} = (\mathbf{I} - \Delta t \mathbf{A})^{-1} \mathbf{x}_n$$

dok diskretizacija po trapeznom pravilu daje

$$\mathbf{x}_{n+1} = \left(\mathbf{I} - \frac{\Delta t}{2} \mathbf{A} \right)^{-1} \left(\mathbf{I} + \frac{\Delta t}{2} \mathbf{A} \right) \mathbf{x}_n$$

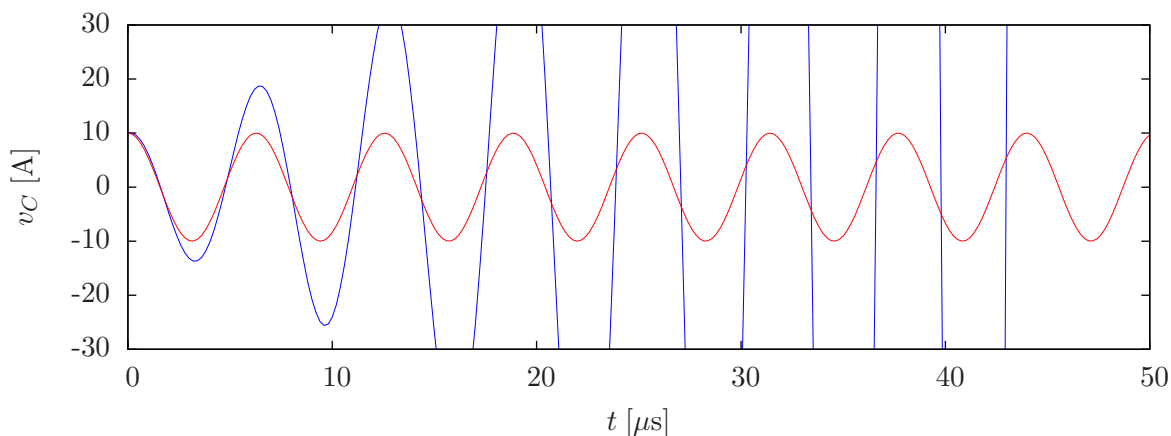
gde je \mathbf{I} jedinična matrica dimenzije 2×2 .

U sva tri slučaja promenljive stanja na kraju vremenskog koraka se dobijaju množenjem vektora promenljivih stanja na početku koraka sa matricom prelaza \mathbf{T}

$$\mathbf{x}_{n+1} = \mathbf{T} \mathbf{x}_n$$

pri čemu je \mathbf{T} konstantna matrica zavisna od izabranog algoritma diskretizacije.

Rezultati simulacije po *forward-Euler* diskretizacionom pravilu su prikazani na slikama 5 i 6. Rezultat simulacije ukazuje na nestabilan sistem, iako je originalni sistem granično stabilan.

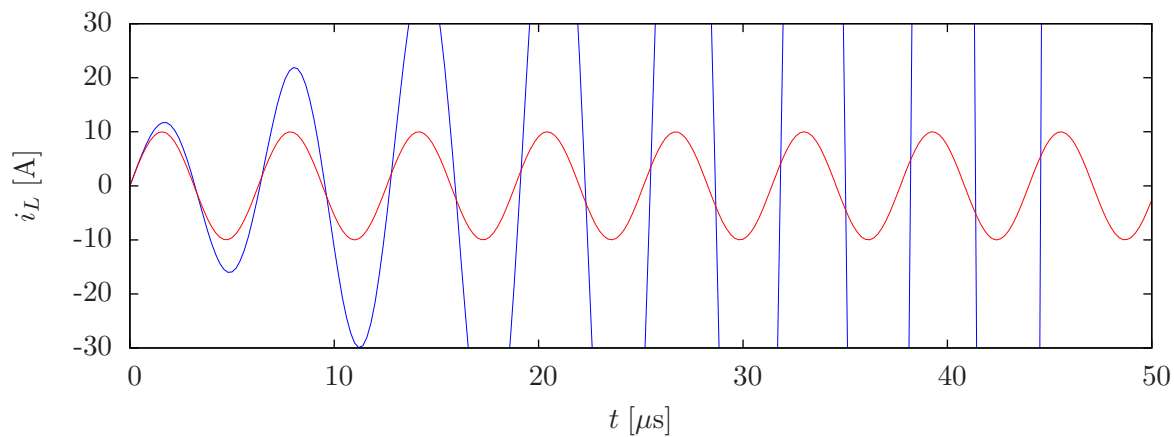


Slika 5: Dijagram v_C , *forward-Euler*

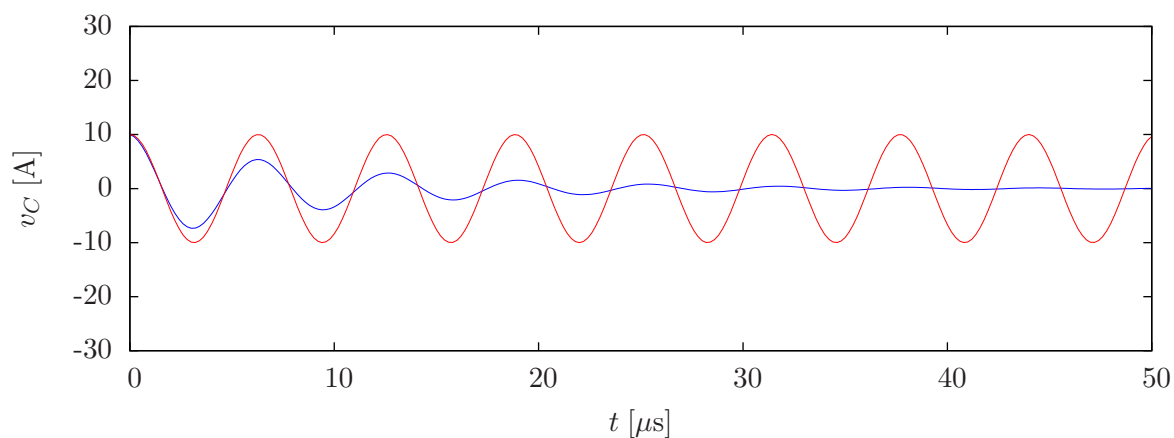
Rezultati simulacije po *backward-Euler* diskretizacionom pravilu su prikazani na slikama 7 i 8. Rezultat simulacije ukazuje na stabilan sistem, iako je originalni sistem granično stabilan.

Rezultati simulacije po trapeznom pravilu su prikazani na slikama 9 i 10. Rezultat simulacije ukazuje na granično stabilan sistem, kakav je i originalni sistem. Ipak, greška simulacije se uočava u faznom odstupanju rezultata simulacije u odnosu na egzaktno rešenje.

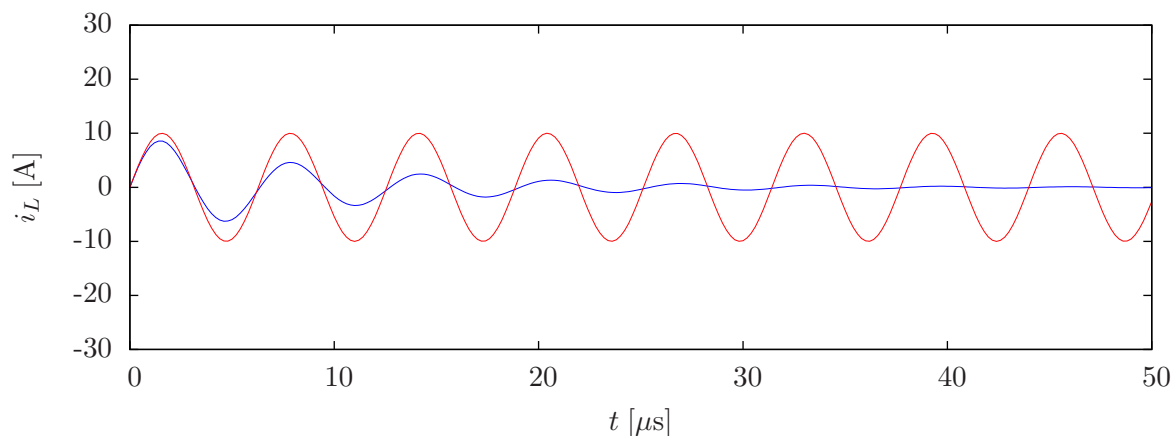
Na osnovu prikazanih rezultata simulacije može se zaključiti da je trapezno pravilo dalo najbolje rezultate u pogledu tačnosti i saglasnosti stabilnosti sa stabilnošću rešenja originalne



Slika 6: Dijagram i_L , *forward-Euler*



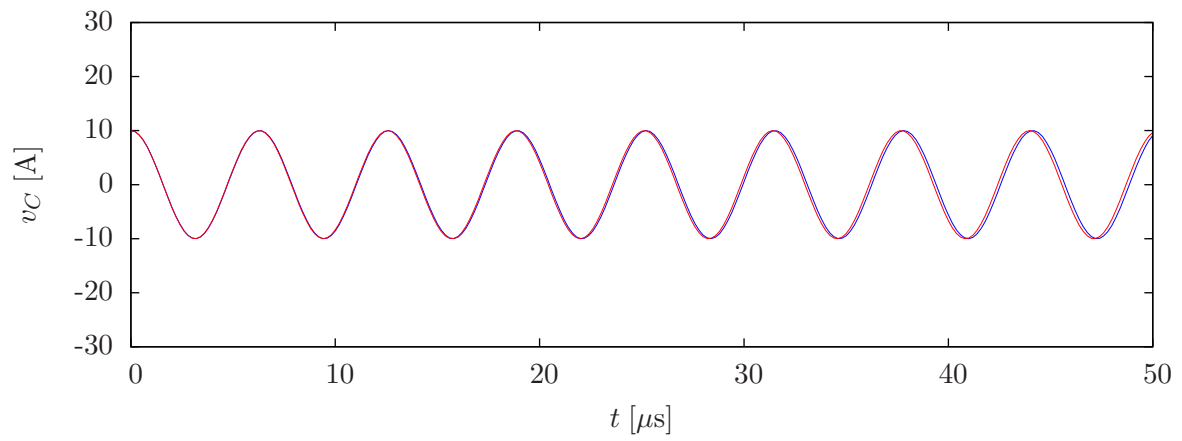
Slika 7: Dijagram v_C , *backward-Euler*



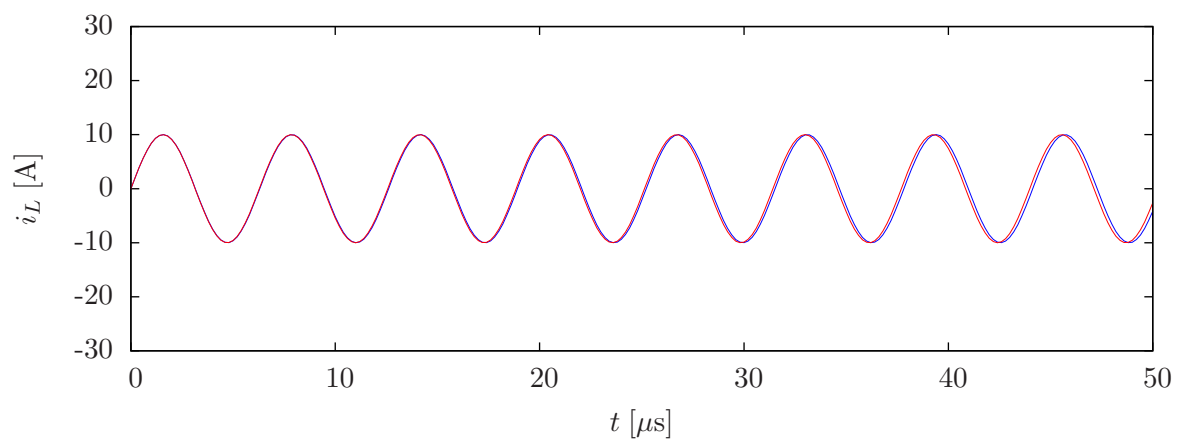
Slika 8: Dijagram i_L , *backward-Euler*

diferencijalne jednačine. Ovde valja naglasiti da najveći deo računskog opterećenja predstavlja množenje matrice prelaza \mathbf{T} prethodnim vektorom stanja u svakom koraku simulacije, a to opterećenje je isto za sva tri razmatrana metoda, pa trapezno pravilo daje najbolji rezultat simulacije pri praktično istom računarskom opterećenju za sva tri poređena algoritma.

Čitaocima se savetuje, ukoliko su zainteresovani, da prouče grešku diskretizacije na nivou jednog koraka, kao i adaptivne algoritme za izbor vremenskog koraka.



Slika 9: Dijagram v_C , trapezno pravilo



Slika 10: Dijagram i_L , trapezno pravilo