Numeričko rješavanje diferencijalnih jednadžbi

7. dio kolegija Numeričke metode financijske matematike

Nela Bosner

Primjer

Opcija je:

- *pravo kupnje i prodaje riskantne imovine (najčešće dionice) za prije određenu cijenu unutar određenog perioda,*
- *financijski instrument koji dozvoljava "klađenje" da li će vrijednost te imovine rasti ili padati,*
- *sporazum između dvije strane oko trgovine imovinom u nekom određenom trenutku u budućnosti.*
  - *Jedna strana (često banka) određuje uvjete ugovora za opciju i prodaje opciju.*
  - *Druga strana kupuje opciju i plaća njenu tržišnu vrijednost koja se naziva* ***premija****.*

Problem je kako izračunati vrijednost opcije .

Primjer (nastavak)

- *Opcija ima ograničeno vrijeme trajanja.*
- *Datum dospijeća T određuje trenutak istjecanja opcije, i nakon tog datuma opcija je bezvrijedna ($t > T$).*
- *Postoje dvije osnovne vrste opcija:*
  - ***call** opcija omogućuje drugoj strani da kupi imovinu za dogovorenu cijenu E datuma T,*
  - ***put** opcija omogućuje drugoj strani da proda imovinu za dogovorenu cijenu E datuma T.*
- *Druga strana nije obavezna kupiti ili prodati imovinu u zadanom vremenu.*
- *U trenutku t ona može:*
  - *prodati opciju po trenutnoj vrijednosti za $t < T$,*
  - *zadržati opciju i ne učiniti ništa,*
  - *kupiti ili prodati imovinu u trenutku dospijeća ($t \leq T$),*
  - *dozvoliti da opcija istekne i postane bezvrijedna ($t \geq T$).*

Primjer (nastavak)

- *Postoje još i podjela opcija po vremenu u kojem se može kupiti ili prodati imovina:*
  - *Kod* ***Europskih opcija*** *to se može učiniti jedino na datum dospijeća T.*
  - *Kod* ***Američkih opcija*** *to se može učiniti u bilo koje vrijeme do datuma dospijeća $t \leq T$.*
- *Vrijednost opcije V ovisi o:*
  - *trenutnoj cijeni imovine $S = S(t) = S_t$,*
  - *o vremenu t, preciznije o preostalom vremenu do dospijeća $T - t$*
- *Zbog toga vrijednost opcije označavamo kao funkciju $V(S, t)$.*

Primjer (nastavak)

- *Osim toga $V(S,t)$ ovisi još i o:*
  - *dogovorenoj cijeni E,*
  - *datumu dospijeća T,*
  - *kamatnoj stopi r — primjenjuje sa na neriskantne investicije,*
  - *volatilnosti σ cijene S_t (standardna devijacija fluktuacije od S_t) — mjeri nesigurnost imovine.*
- *Pretpostavit ćemo još:*
  - $0 \leq t \leq T$*, gdje vrijeme* $t = 0$ *označava “danas”,*
  - *cijena S_t je stohastički proces,*
  - *kamatna stopa r i volatilnost σ su konstantni za* $0 \leq t \leq T$ *(iako na realnom tržištu variraju u vremenu).*

Primjer (nastavak)

- *Naš cilj je izračunati plohu*

$$V(S,t), \quad \text{za } S > 0,\ 0 \le t \le T.$$

- *Koristit ćemo matematički model koji aproksimira i idealizira kompleksnu realnost financijskog svijeta.*
- *$V(S,t)$ se može dobiti kao rješenje **Black–Scholesove diferencijalne jednadžbe***

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

za koju još moramo definirati terminalni uvjet za $t = T$ i rubne uvjete za $S = 0$ i $S \to \infty$.

Primjer (nastavak)

- ***terminalni uvjet***
  - *call opcija:*
$$V(S,T)=\max\{S-E,0\}$$
  - *put opcija:*
$$V(S,T)=\max\{E-S,0\}$$
- ***rubni uvjeti***
  - *call opcija:*
$$\begin{aligned} V(0,t)&=0,\\ V(S,t)&\to S \qquad \text{za } S\to\infty \end{aligned}$$
  - *put opcija:*
$$\begin{aligned} V(0,t)&=Ee^{-r(T-t)},\\ V(S,t)&\to 0 \qquad \text{za } S\to\infty \end{aligned}$$

Primjer (nastavak)

- *Radi jednostavnijeg rješavanja uvodimo transformacije varijable*

$$
\begin{aligned}
S &= Ee^x \\
t &= T - \frac{2\tau}{\sigma^2} \\
q &= \frac{2r}{\sigma^2}
\end{aligned}
$$

- *Ovime dobivamo*

$$V(S,t) = V\left(Ee^x, T - \frac{2\tau}{\sigma^2}\right) = v(x,\tau).$$

Primjer (nastavak)

- *Dalje uvodimo još*

$$v(x,\tau) = Ee^{-\frac{1}{2}(q-1)x-\frac{1}{4}(q+1)^2\tau}y(x,\tau),$$

gdje se može pokazati da je $y(x,\tau)$ rješenje difuzijske jednadžbe

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

za $0 \leq \tau \leq \frac{1}{2}\sigma^2 T$ i $x \in \mathbb{R}$.

- *Terminalni uvjet sada prelazi u* ***inicijalni uvjet***
  - *call opcija:*

    $$y(x,0) = \max\{e^{\frac{1}{2}(q+1)x} - e^{\frac{1}{2}(q-1)x}, 0\}$$

  - *put opcija*

    $$y(x,0) = \max\{e^{\frac{1}{2}(q-1)x} - e^{\frac{1}{2}(q+1)x}, 0\}$$

Primjer (nastavak)

- *I rubne uvjete moramo transformirati, i to tako da*
  - *rubni uvjet za* $S = 0$ *prelazi u rubni uvjet za* $x \to -\infty$
  - *rubni uvjet za* $S \to \infty$ *prelazi u rubni uvjet za* $x \to \infty$
- ***Rubni uvjeti*** *za* $y(x, \tau)$ *su sada oblika*
  - *call opcija:*

$$\begin{aligned} y(x,\tau) &= 0 && \text{za } x \to -\infty, \\ y(x,\tau) &= e^{\frac{1}{2}(q+1)x + \frac{1}{4}(q+1)^2\tau} && \text{za } x \to \infty \end{aligned}$$

  - *put opcija:*

$$\begin{aligned} y(x,\tau) &= e^{\frac{1}{2}(q-1)x + \frac{1}{4}(q-1)^2\tau} && \text{za } x \to -\infty, \\ y(x,\tau) &= 0 && \text{za } x \to \infty \end{aligned}$$

Primjer (nastavak)

- *Za difuzijsku jednadžbu postoji eksplicitno analitičko rješenje dano sa*

$$y(x,\tau) = \frac{1}{2\sqrt{\pi\tau}} \int_{-\infty}^{\infty} y_0(s) e^{-\frac{(x-s)^2}{4\tau}} ds,$$

  gdje je

$$y_0(x) = y(x,0),$$

  inicijalni uvjet.
- *Za call opciju tada ispada da je njena vrijednost dana sa*

$$V(S,t) = \frac{S}{\sqrt{2\pi}} \int_{-\infty}^{d_1} e^{-\frac{s^2}{2}} ds - \frac{Ee^{-r(T-t)}}{\sqrt{2\pi}} \int_{-\infty}^{d_2} e^{-\frac{s^2}{2}} ds,$$

Primjer (nastavak)

gdje su

$$d_1 = \frac{\log\left(\frac{S}{E}\right) + \left(r + \frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{(T-t)}},$$

$$d_2 = \frac{\log\left(\frac{S}{E}\right) + \left(r - \frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{(T-t)}},$$

a za put opciju je

$$V(S,t) = \frac{Ee^{-r(T-t)}}{\sqrt{2\pi}} \int_{-\infty}^{-d_2} e^{-\frac{s^2}{2}}\,ds - \frac{S}{\sqrt{2\pi}} \int_{-\infty}^{-d_1} e^{-\frac{s^2}{2}}\,ds.$$

Numeričko rješavanje paraboličke parcijalne diferencijalne jednadžbe

- Black–Scholesova i difuzijska jednadžba spadaju u paraboličke parcijalne diferencijalne jednadžbe.
- Difuzijska jednadžba je jednostavnija za rješavanje pa se najčešće najprije računa njeno rješenje $y(x,\tau)$, a onda se iz njega transformacijom varijabli dobiva rješanje Black–Scholesove jednadžbe $V(S,t)$:

$$V(S,t)=E^{\frac{1}{2}(q+1)}S^{-\frac{1}{2}(q-1)}e^{-\frac{1}{8}(q+1)^2\sigma^2(T-t)}y\left(\log\left(\frac{S}{E}\right),\frac{\sigma^2}{2}(T-t)\right),$$

gdje je $q=\frac{2r}{\sigma^2}$.

- Obzirom da egzaktno rješenje nije jednostavno za izračunati, računa se aproksimativno numeričko rješenje.
- Za numeričko rješavanje parcijalne diferencijalne jednadžbe koristit ćemo **metodu konačnih razlika** koja parcijalne derivacije aproksimira pomoću Taylorovog reda funkcije.

Metoda konačnih razlika

- Parcijalnu derivaciju $\partial y / \partial \tau$ možemo definirati kao

$$\frac{\partial y}{\partial \tau}(x,\tau) = \lim_{\delta\tau \to 0} \frac{y(x,\tau+\delta\tau) - y(x,\tau)}{\delta\tau}.$$

- Umjesto računanja limesa kada $\delta\tau \to 0$, uzet ćemo $\delta\tau > 0$ koji je vrlo mali i dobit ćemo aproksimaciju

$$\frac{\partial y}{\partial \tau}(x,\tau) = \frac{y(x,\tau+\delta\tau) - y(x,\tau)}{\delta\tau} + \mathcal{O}(\delta\tau).$$

- Ovime smo dobili **konačnu razliku** od $\partial y / \partial \tau$, a konačnu razliku ovoga oblika posebno ćemo još zvati i **konačna razlika unaprijed**.
- Izraz $\mathcal{O}(\delta\tau)$ dolazi iz Taylorovog reda, i što je $\delta\tau$ manji dobit ćemo točniju aproksimaciju.

- Alternativno možemo definirati

$$\frac{\partial y}{\partial \tau}(x,\tau) = \lim_{\delta\tau \to 0} \frac{y(x,\tau) - y(x,\tau-\delta\tau)}{\delta\tau},$$

tako da je aproksimacija dana sa

$$\frac{\partial y}{\partial \tau}(x,\tau) = \frac{y(x,\tau) - y(x,\tau-\delta\tau)}{\delta\tau} + \mathcal{O}(\delta\tau).$$

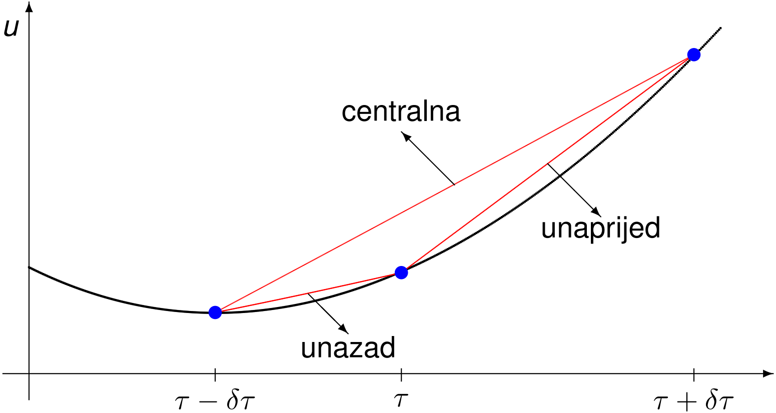
- Konačnu razliku ovoga oblika zovemo **konačna razlika unazad**.
- Također možemo primijetiti da je

$$\frac{\partial y}{\partial \tau}(x,\tau) = \lim_{\delta\tau \to 0} \frac{y(x,\tau+\delta\tau) - y(x,\tau-\delta\tau)}{2\delta\tau},$$

i definirati **centralnu konačnu razliku**

$$\frac{\partial y}{\partial \tau}(x,\tau) = \frac{y(x,\tau+\delta\tau) - y(x,\tau-\delta\tau)}{2\delta\tau} + \mathcal{O}\left((\delta\tau)^2\right).$$

- Vidimo da je centralna konačna razlika točnija.



Slika: Konačna razlika unazad, unaprijed, i centralna.

- Kada se primijenjuju na difuzijsku jednadžbu, konačne razlike unaprijed i unazad koje aproksimiraju $\partial y/\partial \tau$ vode do **eksplicitne** odnosno **implicitne** metode konačnih razlika.
- Centralna konačna razlika gornjeg oblika po varijabli τ se ne koriste u praksi jer daje nestabilne metode.
- Centralna konačna razlika oblika

$$\frac{\partial y}{\partial \tau}(x,\tau) = \frac{y(x,\tau+\delta\tau/2) - y(x,\tau-\delta\tau/2)}{\delta\tau} + \mathcal{O}\left((\delta\tau)^2\right)$$

pojavljuje se u **Crank–Nicolsonovoj shemi za konačne razlike**.

- Parcijalne derivacije po varijabli x možemo definirati na analogan način:
  - **konačna razlika unaprijed**

$$\frac{\partial y}{\partial x}(x,\tau) = \frac{y(x+\delta x,\tau) - y(x,\tau)}{\delta x} + \mathcal{O}(\delta x)$$

  - **konačna razlika unazad**

$$\frac{\partial y}{\partial x}(x,\tau) = \frac{y(x,\tau) - y(x-\delta x,\tau)}{\delta x} + \mathcal{O}(\delta x)$$

  - **centralna konačna razlika**

$$\frac{\partial y}{\partial x}(x,\tau) = \frac{y(x+\delta x,\tau) - y(x-\delta x,\tau)}{2\delta x} + \mathcal{O}\left((\delta x)^2\right)$$

- Za drugu parcijalnu derivaciju $\partial^2 y/\partial x^2$ možemo definirati simetričnu konačnu razliku kao
  - konačnu razliku unaprijed od konačnih razlika unazad
  - konačnu razliku unazad od konačnih razlika unaprijed

- U oba slučaja dobivamo **simetričnu centralnu konačnu razliku**

$$\frac{\partial^2 y}{\partial x^2}(x,\tau) = \frac{\frac{y(x+\delta x,\tau)-y(x,\tau)}{\delta x} - \frac{y(x,\tau)-y(x-\delta x,\tau)}{\delta x}}{\delta x} + \mathcal{O}\left((\delta x)^2\right)$$

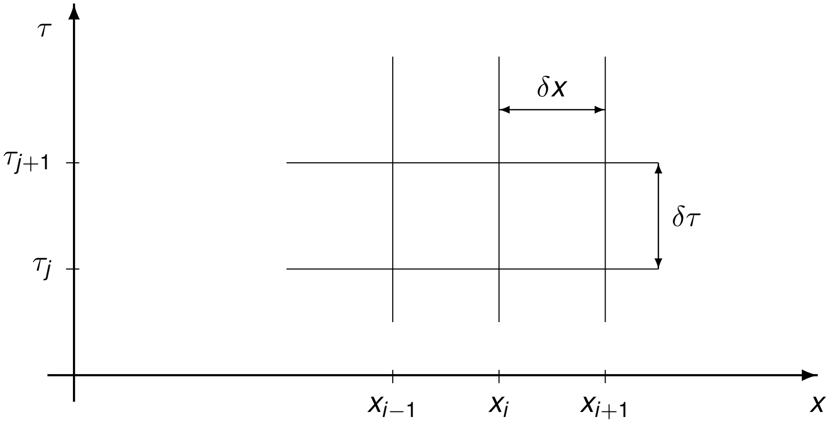
$$= \frac{y(x+\delta x,\tau) - 2y(x,\tau) + y(x-\delta x,\tau)}{(\delta x)^2} + \mathcal{O}\left((\delta x)^2\right).$$

- Kako bismo mogli primijeniti metodu konačnih razlika na difuzijsku jednadžbu moramo podijeliti
  - x os na ekvidistantne čvorove sa razmakom od δx
  - τ os na ekvidistantne čvorove sa razmakom od $\delta\tau$
- Ovime na (x,τ) ravnini definiramo mrežu, pri čemu čvorovi mreže imaju oblik

$$(x_i, \tau_j) = (i\delta x, j\delta\tau)$$

- U tom slučaju računat ćemo aproksimativno rješenje samo u čvorovima mreže, i pišemo

$$y_{i,j} \approx y(i\delta x, j\delta\tau).$$



Slika: Oznake na mreži.

Exsplicitna metoda konačnih razlika

- Razmatramo opći oblik transformiranog Black–Scholesovog modela za Europsku opciju, odnosno difuzijsku jednadžbu

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

sa rubnim i inicijalnim uvjetima

$$\begin{aligned} \lim_{x\to-\infty} y(x,\tau) &= y_{-\infty}(x,\tau), \\ \lim_{x\to\infty} y(x,\tau) &= y_{\infty}(x,\tau), \\ y(x,0) &= y_0(x). \end{aligned}$$

- Želimo naći aproksimaciju rješenja u čvorovima mreže koristeći
  - konačnu razliku unaprijed za $\partial y/\partial \tau$,
  - simetričnu centralnu konačnu razliku za $\partial^2 y/\partial x^2$.

- Pri tome se difuzijska jednadžba transformira u

$$\frac{y_{i,j+1} - y_{i,j}}{\delta\tau} + \mathcal{O}(\delta\tau) = \frac{y_{i+1,j} - 2y_{i,j} + y_{i-1,j}}{(\delta x)^2} + \mathcal{O}\left((\delta x)^2\right).$$

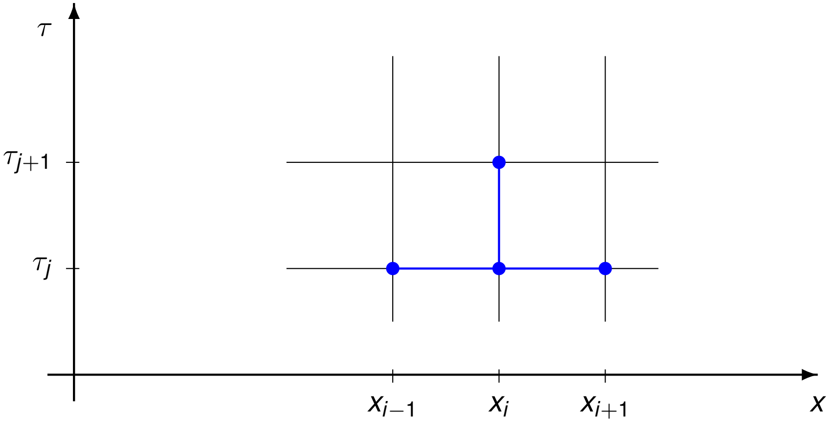
- Zanemarujući izraze $\mathcal{O}(\delta\tau)$ i $\mathcal{O}\left((\delta x)^2\right)$ dobivamo diferencijsku jednadžbu

$$y_{i,j+1} = \lambda y_{i-1,j} + (1 - 2\lambda) y_{i,j} + \lambda y_{i+1,j},$$

gdje je

$$\lambda = \frac{\delta\tau}{(\delta x)^2}.$$

- Ako u vremenskom koraku j znamo $y_{i,j}$ za sve vrijednosti od i, tada $y_{i,j+1}$ možemo izračunati **eksplicitno**.



Slika: $y_{i,j+1}$ ovisi samo o $y_{i-1,j}$, $y_{i,j}$ i $y_{i+1,j}$.

- Ako izaberemo ekvidistantne točke x_i, ne možemo riješiti problem za sve $-\infty < x_i < \infty$ bez da izvršimo beskonačno mnogo koraka.
- Zbog toga uzimamo konačan i dovoljno velik broj koraka po x-u.
- Riješit ćemo problem za

$$n_{min}\delta x = x_{n_{min}} \leq x_i \leq x_{n_{max}} = n_{max}\delta x,$$

gdje su $-n_{min}$ i n_{max} veliki prirodni brojevi.

- Dalje, segment $\left[0, \frac{\sigma^2 T}{2}\right]$ dijelimo na m jednaka podintervala, tako da je

$$\delta\tau = \frac{\sigma^2 T}{2m}$$

- Sada možemo riješiti diferencijsku jednadžbu za

$$n_{min} < i < n_{max}, \quad 0 < j \leq m.$$

- Rubne uvjete koristimo za određivanje $y_{n_{min},j}$ i $y_{n_{max},j}$:

$$
\begin{aligned}
y_{n_{min},j} &= y_{-\infty}(n_{min}\delta x, j\delta\tau), & 0 < j \leq m,\\
y_{n_{max},j} &= y_{\infty}(n_{max}\delta x, j\delta\tau), & 0 < j \leq m.
\end{aligned}
$$

- Za pokretanje ove iterativne metode koristimo inicijalni uvjet

$$y_{i,0} = y_0(i\delta x), \quad n_{min} \leq i \leq n_{max}.$$

- Iterativna metoda završava za $j = m$ i rješenjem

$$y_{i,m}, \quad n_{min} \leq i \leq n_{max},$$

što predstavlja aproksimaciju rješenja za $y(i\delta x, \frac{\sigma^2 T}{2})$, $n_{min} \leq i \leq n_{max}$.

- Transformacijom varijabli dobivamo mrežu na S koordinati

$$S_i = Ee^{i\delta x}, \quad n_{min} \leq i \leq n_{max},$$

i transformacijom $y_{i,m}$ u $V_{i,m}$ dobivamo aproksimativno rješenje za $V(S_i, 0)$, jer je za $\tau = \frac{\sigma^2 T}{2}$, $t = T - \frac{2\tau}{\sigma^2} = 0$.
- Ovimo smo dobili današnju aproksimativnu vrijednost opcije za diskretne vrijednosti cijene imovine.

Algoritam (Eksplicitna metoda konačnih razlika za difuzijsku jednadžbu)

$\lambda = \frac{\delta\tau}{(\delta x)^2}$;

for $i = n_{min} : n_{max}$

  $y_{stari}(i) = y_0(i \cdot \delta x)$;

end

for $j = 1 : m$

  $y_{novi}(n_{min}) = y_{-\infty}(n_{min} \cdot \delta x, j \cdot \delta\tau)$;

  $y_{novi}(n_{max}) = y_{\infty}(n_{max} \cdot \delta x, j \cdot \delta\tau)$;

  for $i = (n_{min} + 1) : (n_{max} - 1)$

    $y_{novi}(i) = \lambda \cdot y_{stari}(i-1) + (1 - 2 \cdot \lambda) \cdot y_{stari}(i) +$
    $+ \lambda \cdot y_{stari}(i+1)$;

  end

  $y_{stari} = y_{novi}$;

end

$y = y_{stari}$;

Zadaci

Zadatak

Napišite M-file funkciju `ex_mkr_difuzijska()` *koja implementira eksplicitnu metodu konačnih razlika za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre*

- *korak* δx *po* x *koordinati i korak* $\delta\tau$ *po* τ *koordinati*
- *maksimalan broj koraka* m *po* τ *koordinati*
- *minimalan* n_{min} *i maksimalan* n_{max} *indeks čvorova po* x *koordinati*
- *pokazivač na funkciju inicijalnog uvjeta* y_0
- *pokazivače na funkcije rubnih uvjeta* $y_{-\infty}$ *i* y_{∞}.

Funkcija neka vraća

- *aproksimativno rješenje* y *u čvorovima sa* x *koordinatama* $n_{min}\delta x : \delta x : n_{max}\delta x$ *i* τ *koordinatom* $\tau = m\delta\tau = \frac{1}{2}\sigma^2 T$.

Zadatak

Svoju funkciju `ex_mkr_difuzijska()` *isprobajte na sljedećem primjeru.*

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

$$y(x,0) = \sin(\pi x), \quad y(0,\tau) = y(1,\tau) = 0.$$

Egzaktno rješnje glasi

$$y(x,\tau) = e^{-\pi^2 \tau} \sin(\pi x).$$

Uzmite sljedeće parametre:

- $\delta x = 0.1$
- $n_{min} = 0$ *i* $n_{max} = 10$

Zadatak (nastavak)

Numeričko rješenje pomoću eksplicitne metode konačnih razlika nađite za dvije različite situacije:

1. *kada je* $\lambda = 0.05$
   - $\delta\tau = 0.0005$
   - $m = 1000$
2. *kada je* $\lambda = 1$
   - $\delta\tau = 0.01$
   - $m = 50$

U oba slučaja $\tau_m = 0.5$. *Dobivena rješenja usporedite sa egzaktnim rješenjem. Što možete zaključiti?*

Zadatak

Napišite M-file funkciju `ex_mkr_black_scholes()` *koja numerički rješava Black–Scholesovu jednadžbu pomoću eksplicitne metode konačnih razlika za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre*

- *dogovorenu cijenu E na datumu dospijeća*
- *vrijeme do dospijeća T izraženo u godinama*
- *kamatnu stopu r*
- *volatilnost σ cijene imovine*
- *maksimalan broj koraka m po τ koordinati za rješavanje difuzijske jednadžbe*
- *minimalan n_{min} i maksimalan n_{max} indeks čvorova po x koordinati za rješavanje difuzijske jednadžbe*
- *parametar $\lambda = \frac{\delta\tau}{(\delta x)^2}$*

Zadatak (nastavak)

- *string vrsta koji označava da li se radi o "call" ili o "put" opciji.*

Funkcija treba odrediti

$$\delta\tau = \frac{\sigma^2 T}{2m}, \quad \delta x = \sqrt{\frac{\delta\tau}{\lambda}}.$$

Funkcija neka vraća

- *polje* V_0 *duljine* $n_{max} - n_{min} + 1$ *koje sadrži aproksimativna rješenja* $V(S_i, 0)$
- *polje* S *duljine* $n_{max} - n_{min} + 1$ *koje sadrži mrežu* $S_i = Ee^{i\delta x}$, $n_{min} \leq i \leq n_{max}$.

Zadatak (nastavak)

Uputa: *Izlazne vrijednosti ove funkcije u MATLAB-u izračunajte iz izlaznih vrijednosti funkcije* `ex_mkr_difuzijska()` *na sljedeći način:*

$$\begin{aligned} V_0 &= E * exp(-0.25 * (q+1)^2 * tau) * \\ &\quad exp(-0.5 * (q-1) * x). * y; \\ S &= E * exp(x); \end{aligned}$$

pri čemu je x vektor čvorova mreže za difuzijsku jednadžbu, a $tau = m \cdot \delta\tau = \frac{\sigma^2 T}{2}$.

Zadatak

Svoju funkciju `ex_mkr_black_scholes()` *isprobajte na primjeru Black–Scholesovog modela za put opciju sa sljedećim parametrima.*

- $E = 10$
- $T = 0.5$
- $r = 0.05$
- $\sigma = 0.20$
- $m = 100$
- $n_{min} = -1000$ i $n_{max} = 1000$

Zadatak (nastavak)

Numeričko rješenje pomoću eksplicitne metode konačnih razlika za difuzijsku jednadžbu nađite za tri različite situacije:

1. $\lambda = 0.25$
2. $\lambda = 0.5$
3. $\lambda = 0.55$

Dobivena rješenja usporedite sa rješenjem dobivenim MATLAB-ovom funkcijom `blsprice()`*. Što možete zaključiti?*

Stabilnost eksplicitne metode konačnih razlika

- Stabilnost numeričke metode je u bliskoj vezi sa numeričkom greškom.
- Metoda konačnih razlika je stabilna ako greška učinjena u jednom koraku metode ne utječe na povečanje greške u koracima koji slijede.
- Kod **neutralno stabilne** metode greška ostaje konstantna u svim koracima.
- Ako greške opadaju i po mogućnosti se prigušuju, kažemo da je numerička metoda **stabilna**.
- Ako, s druge strane, greška raste sa povećanjem broja koraka, aproksimativno rješenje divergira, i kažemo da je numerička metoda **nestabilna**.
- Za probleme koji ovise o vremenu stabilnost garantira da će numerička metoda dati ograničeno rješenje, kadgod je egzaktno rješenje diferencijalne jednadžbe ograničeno.

- Za daljnju analizu korisno je sve vrijednosti $y_{i,j}$ za fiksni vremenski korak j organizirati u vektor

$$y^{(j)} = \begin{bmatrix} y_{n_{min}+1,j} & \cdots & y_{n_{max}-1,j} \end{bmatrix}^T.$$

- Za matrični oblik diferencijske jednadžbe definiramo $(n_{max} - n_{min} - 1) \times (n_{max} - n_{min} - 1)$ matricu A

$$A = \begin{bmatrix} 1-2\lambda & \lambda & 0 \cdots & 0 & \\ \lambda & 1-2\lambda & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \lambda \\ 0 & \cdots & 0 & \lambda & 1-2\lambda \end{bmatrix}.$$

- Tada eksplicitnu metodu možemo zapisati kao
$$y^{(j+1)} = Ay^{(j)} + y_r^{(j)},$$
gdje je
$$y_r^{(j)} = \lambda[\ y_{n_{min},j} \quad 0 \quad \cdots \quad 0 \quad y_{n_{max},j}\ ]^T.$$
- Pretpostavimo sada da našu numeričku metodu izvršavamo na računalu u aritmetici konačne preciznosti.
- U tom slučaju ćemo u svakom koraku j umjesto $y^{(j+1)}$ izračunati aproksimativnu vrijednost $\tilde{y}^{(j+1)}$ koja sadrži i greške zaokruživanja.
- Neka je
$$\tilde{y}^{(j+1)} = A\tilde{y}^{(j)} + \tilde{y}_r^{(j)} + f^{(j+1)},$$
gdje $f^{(j+1)}$ sadrži greške zaokruživanja koje su se dogodile kod računanja $A\tilde{y}^{(j)} + \tilde{y}_r^{(j)}$.

- Definirajmo ukupnu grešku

$$e^{(j)} = \tilde{y}^{(j)} - y^{(j)}, \quad j \geq 0.$$

- Možemo zaključiti da vrijedi sljedeće:

$$e^{(j+1)} = Ae^{(j)} + (\tilde{y}_r^{(j)} - y_r^{(j)}) + f^{(j+1)}.$$

- Usredotočimo se sada na utjecaj grešaka zaokruživanja $e^{(0)}$ kod računanja inicijalnog uvjeta (za $j = 0$).
- Zbog jednostavnosti, pretpostavimo da su svi daljnji koraci ($j > 0$) izračunati egzaktno, tj. da je

$$\tilde{y}_r^{(j)} = y_r^{(j)}, \quad f^{(j)} = 0, \quad j > 0.$$

- Tada imamo
$$e^{(j+1)} = Ae^{(j)} = A^{j+1}e^{(0)}.$$
- Da bi metoda bila stabilna, greška $e^{(0)}$ mora biti prigušena, a za to treba biti
$$\lim_{j\to\infty} A^j e^{(0)} = 0,$$
što nam je poznata situacija iz iterativnih metoda za sustave linearnih jednadžbi.
- Od tamo znamo da će metoda biti stabilna ako i samo ako je
$$\rho(A) < 1,$$
gdje je $\rho(A)$ spektralni radijus matrice A.
- Dakle, da bi metoda bila stabilna zahtijevamo da za sve svojstvene vrijednosti $\mu_1(A), \ldots, \mu_{n_{max}-n_{min}-1}(A)$ od A vrijedi
$$|\mu_k(A)| < 1, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Slijedeći korak je računanje svojstvenih vrijednosti od A.
- U tu svrhu, matricu pišemo kao

$$A = I - \lambda \cdot \underbrace{\begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ 0 & & & -1 & 2 \end{bmatrix}}_{=G}.$$

- Preostaje nam sada samo naći svojstvene vrijednosti $\mu_k(G)$ od matrice G, jer su tada

$$\mu_k(A) = 1 - \lambda \cdot \mu_k(G), \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Matrica G pojavljuje se često u primjeni i pomno je proučavana, pa je pokazano da su njene svojstvene vrijednosti dane sa

$$\begin{aligned}\mu_k(G) &= 2 - 2\cos\left(\frac{k\pi}{n_{max} - n_{min}}\right)\\ &= 4\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right), \quad k = 1, \ldots, n_{max} - n_{min} - 1.\end{aligned}$$

- Dalje vrijedi

$$\mu_k(A) = 1 - 4\lambda\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right), \; k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Da bi uvjet stabilnosti $|\mu_k(A)| < 1$ bio zadovoljen, mora vrijediti

$$\left|1 - 4\lambda\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right)\right| < 1, \; k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Budući da je $\lambda > 0$ po definiciji slijedi da je uvijek $1 - 4\lambda \sin^2\left(\frac{k\pi}{2(n_{max}-n_{min})}\right) < 1$.
- S druge strane, uvjet

$$-1 < 1 - 4\lambda \sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right)$$

se može pojednostavniti na

$$\lambda \sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right) < \frac{1}{2}.$$

- Dakle, gornji uvjet stabilnosti ekvivalentan je dvijema jednadžbama

$$\lambda > 0$$

$$\lambda \sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right) < \frac{1}{2}.$$

- Najveći izraz sa sinusom je

$$\sin\left(\frac{(n_{max} - n_{min} - 1)\pi}{2(n_{max} - n_{min})}\right) < 1,$$

a ako povećavamo dimenziju matrice A $d = n_{max} - n_{min} - 1$ tada vrijedi

$$\lim_{d\to\infty} \sin\left(\frac{d\pi}{2(d+1)}\right) = 1.$$

- Ovime smo dobili konačan uvjet.

Teorem

Za

$$0 < \lambda \leq \frac{1}{2}$$

eksplicitna metoda konačnih razlika za difuzijsku jednadžbu je stabilna.

- Iz prethodnih zadataka vidjeli smo da aproksimacija znatno odstupa od egzaktnog rješenja difuzijske jednadžbe za $\lambda > \frac{1}{2}$.
- Ovaj kriterij stabilnosti daje uvjet na veličinu koraka:

$$0 < \delta\tau \leq \frac{(\delta x)^2}{2}.$$

- Kao posljedica uvjeta stabilnosti, parametri m, n_{min} i n_{max} ne mogu biti izabrani nezavisno jedan od drugoga.
- Ako moramo izračunati rješenje sa velikom točnošću, tada δx mora biti malen, što daje kvadratnu ogradu za $\delta\tau$ koji mora biti još manji.
- Zato nam je praktičnije naći numeričku metodu koja je bezuvjetno stabilna.

Potpuna implicitna metoda konačnih razlika

- Implicitne metode se koriste kako bi se izbjegla ograničenja vezana uz stabilnost eksplicitne metode.
- Ove metode nam omogućuju da koristimo mreže u x koordinati sa velikim brojem čvorova, bez da moramo uzeti jako mali $\delta\tau$.
- Jedna od implicitnih metoda je i potpuna implicitna metoda konačnih razlika, koja računa aproksimaciju rješenja difuzijske jednadžbe u čvorovima mreže koristeći
  - konačnu razliku unazad za $\partial y/\partial\tau$,
  - simetričnu centralnu konačnu razliku za $\partial^2 y/\partial x^2$.

- Pri tome se difuzijska jednadžba transformira u

$$\frac{y_{i,j+1} - y_{i,j}}{\delta\tau} + \mathcal{O}(\delta\tau) = \frac{y_{i+1,j+1} - 2y_{i,j+1} + y_{i-1,j+1}}{(\delta x)^2} + \mathcal{O}\left((\delta x)^2\right).$$

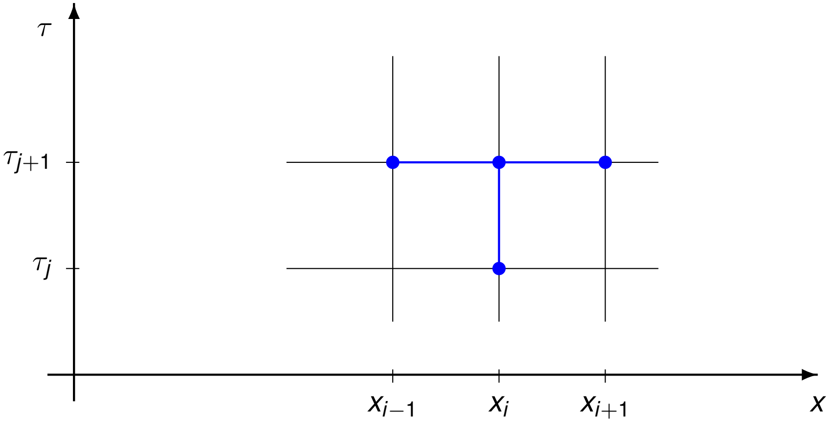
- Zanemarujući izraze $\mathcal{O}(\delta\tau)$ i $\mathcal{O}\left((\delta x)^2\right)$ dobivamo diferencijsku jednadžbu

$$-\lambda y_{i-1,j+1} + (1+2\lambda) y_{i,j+1} - \lambda y_{i+1,j+1} = y_{i,j},$$

gdje je opet

$$\lambda = \frac{\delta\tau}{(\delta x)^2}.$$

- U potpunoj implicitnoj metodi $y_{i-1,j+1}$, $y_{i,j+1}$ i $y_{i+1,j+1}$ implicitno ovise o $y_{i,j}$: nove vrijednosti se ne mogu razdvojiti i eksplicitno izračunati iz starih vrijednosti.
- Radi se o simultanom rješavanju jednadžbi, odnosno o rješavanju sustava linearnih jednadžbi.



Slika: $y_{i-1,j+1}$, $y_{i,j+1}$ i $y_{i+1,j+1}$ ovise o $y_{i,j}$.

- Kao i kod eksplicitne metode uzimamo konačan i dovoljno velik broj koraka po x-u.
- Riješit ćemo problem za

$$n_{min}\delta x = x_{n_{min}} \leq x_i \leq x_{n_{max}} = n_{max}\delta x,$$

gdje su $-n_{min}$ i n_{max} veliki prirodni brojevi.
- Dalje, segment $\left[0, \frac{\sigma^2 T}{2}\right]$ dijelimo na m jednaka podintervala, tako da je

$$\delta\tau = \frac{\sigma^2 T}{2m}$$

- Sada možemo riješiti diferencijsku jednadžbu za

$$n_{min} < i < n_{max}, \quad 0 < j \leq m.$$

- Rubne uvjete koristimo za određivanje $y_{n_{min},j}$ i $y_{n_{max},j}$:

$$\begin{aligned} y_{n_{min},j} =& y_{-\infty}(n_{min}\delta x, j\delta\tau), \qquad & 0 < j \leq m, \\ y_{n_{max},j} =& y_{\infty}(n_{max}\delta x, j\delta\tau), \qquad & 0 < j \leq m. \end{aligned}$$

- Za pokretanje ove iterativne metode koristimo inicijalni uvjet

$$y_{i,0} = y_0(i\delta x), \quad n_{min} \leq i \leq n_{max}.$$

- Iterativna metoda završava za $j = m$ i rješenjem

$$y_{i,m}, \quad n_{min} \leq i \leq n_{max},$$

što predstavlja aproksimaciju rješenja za $y(i\delta x, \frac{\sigma^2 T}{2})$, $n_{min} \leq i \leq n_{max}$.

- S obzirom da moramo rješavati sustave, sada nam i u fazi računanja treba matrični oblik diferencijske jednadžbe.
- Definiramo $(n_{max} - n_{min} - 1) \times (n_{max} - n_{min} - 1)$ matricu A

$$A = \begin{bmatrix} 1+2\lambda & -\lambda & 0 \cdots & 0 & \\ -\lambda & 1+2\lambda & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\lambda \\ 0 & \cdots & 0 & -\lambda & 1+2\lambda \end{bmatrix},$$

i vektor desne strane sustava

$$b = y^{(j)} + y_r^{(j+1)},$$

gdje su

$$\begin{aligned} y^{(j)} &= \begin{bmatrix} y_{n_{min}+1,j} & \cdots & y_{n_{max}-1,j} \end{bmatrix}^T, \\ y_r^{(j+1)} &= \lambda \begin{bmatrix} y_{n_{min},j+1} & 0 & \cdots & 0 & y_{n_{max},j+1} \end{bmatrix}^T. \end{aligned}$$

- Sada potpunu implicitnu metodu možemo napisati u matričnom obliku kao
$$Ay^{(j+1)} = y^{(j)} + y_r^{(j+1)} = b^{(j)}.$$
- Vektor $y_r^{(j+1)}$ se pojavljuje zbog rubnih uvjeta, npr. iz prve jednadžbe slijedi
$$(1+2\lambda)y_{n_{min}+1,j+1} - \lambda y_{n_{min}+2,j+1} = y_{n_{min}+1,j} + \lambda y_{n_{min},j+1}.$$
- Pokazat ćemo da je matrica A regularna pa se korak implicitne metode može napisati eksplicitno kao
$$y^{(j+1)} = A^{-1}\left(y^{(j)} + y_r^{(j+1)}\right).$$

Algoritam (Potpuna implicitna metoda konačnih razlika za difuzijsku jednadžbu)

definiraj rješavač s matricom A; $\lambda = \frac{\delta\tau}{(\delta x)^2}$;

$$
\begin{array}{l}
\textbf{for } i = n_{min} : n_{max} \\
\quad y(i) = y_0(i \cdot \delta x); \\
\textbf{end} \\
\textbf{for } j = 1 : m \\
\quad \textbf{for } i = (n_{min}+1) : (n_{max}-1) \\
\quad\quad b(i) = y(i); \\
\quad \textbf{end} \\
\quad y(n_{min}) = y_{-\infty}(n_{min} \cdot \delta x, j \cdot \delta\tau); \\
\quad y(n_{max}) = y_{\infty}(n_{max} \cdot \delta x, j \cdot \delta\tau); \\
\quad b(n_{min}+1) = b(n_{min}+1) + \lambda \cdot y(n_{min}); \\
\quad b(n_{max}-1) = b(n_{max}-1) + \lambda \cdot y(n_{max}); \\
\quad \textit{riješi sustav } Ay(n_{min}+1 : n_{max}-1) = b; \\
\textbf{end}
\end{array}
$$

Stabilnost potpune implicitne metode konačnih razlika

- Kao i kod eksplicitne metode pretpostavimo da našu numeričku metodu izvršavamo na računalu u aritmetici konačne preciznosti.
- U tom slučaju ćemo u svakom koraku j umjesto $y^{(j+1)}$ izračunati aproksimativnu vrijednost $\tilde{y}^{(j+1)}$ koja sadrži i greške zaokruživanja.
- Neka je

$$\tilde{y}^{(j+1)} = A^{-1}\left(\tilde{y}^{(j)} + \tilde{y}_r^{(j+1)}\right) + f^{(j+1)},$$

gdje $f^{(j+1)}$ sadrži greške zaokruživanja koje su se dogodile kod računanja $A^{-1}\left(\tilde{y}^{(j)} + \tilde{y}_r^{(j+1)}\right)$.

- Definirajmo ukupnu grešku

$$e^{(j)} = \tilde{y}^{(j)} - y^{(j)}, \quad j \geq 0.$$

- Možemo zaključiti da vrijedi sljedeće:

$$e^{(j+1)} = A^{-1}e^{(j)} + A^{-1}(\tilde{y}_r^{(j+1)} - y_r^{(j+1)}) + f^{(j+1)}.$$

- Usredotočimo se sada na utjecaj grešaka zaokruživanja $e^{(0)}$ kod računanja inicijalnog uvjeta (za $j = 0$).
- Zbog jednostavnosti, pretpostavimo da su svi daljnji koraci ($j > 0$) izračunati egzaktno, tj. da je

$$\tilde{y}_r^{(j)} = y_r^{(j)}, \quad f^{(j)} = 0, \quad j > 0.$$

- Tada imamo

$$e^{(j+1)} = A^{-1}e^{(j)} = A^{-(j+1)}e^{(0)}.$$

- Da bi metoda bila stabilna, greška $e^{(0)}$ mora biti prigušena, a za to treba biti

$$\lim_{j\to\infty} A^{-j}e^{(0)} = 0.$$

- Znamo da će metoda biti stabilna ako i samo ako je
$$\rho(A^{-1}) < 1,$$
- Dakle, da bi metoda bila stabilna zahtijevamo da za sve svojstvene vrijednosti $\mu_1(A^{-1}), \ldots, \mu_{n_{max}-n_{min}-1}(A^{-1})$ od A^{-1} vrijedi
$$|\mu_k(A^{-1})| < 1, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$
- Dalje vrijedi da je $\mu_k(A^{-1}) = (\mu_k(A))^{-1}$, a svojstvene vrijednosti matrice A dobit ćemo iz rastava
$$A = I + \lambda \cdot \underbrace{\begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ 0 & & & -1 & 2 \end{bmatrix}}_{=G}.$$

- Budući da znamo svojstvene vrijednosti $\mu_k(G)$ od matrice G, tada je

$$\mu_k(A^{-1}) = \frac{1}{1 + \lambda \cdot \mu_k(G)}, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Kako je

$$\mu_k(G) = 4\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right), \; k = 1, \ldots, n_{max} - n_{min} - 1,$$

imamo

$$\mu_k(A) = 1+4\lambda\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right), \; k = 1, \ldots, n_{max}-n_{min}-1.$$

- Jer je $\lambda > 0$, a $\sin\left(\frac{k\pi}{2(n_{max}-n_{min})}\right) \neq 0$ za $k = 1, \ldots, n_{max} - n_{min} - 1$, onda je

$$\mu_k(A) > 1$$

$$0 < \mu_k(A^{-1}) < 1, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Dakle, za bilo koji $\lambda > 0$ je $0 < \mu_k(A^{-1}) < 1$, što znači da je potpuna implicitna metoda konačnih razlika bezuvjetno stabilna.
- S druge strane, vidimo da su sve svojstvene vrijednosti matrice A pozitivne, što znači da je matrica pozitivno definitna.
- Zbog toga za rješavanje sustava $Ay^{(j+1)} = b^{(j)}$ možemo koristiti metode
    - faktorizaciju Choleskog
    - Gauss–Seidelovu i SOR metodu
    - metodu konjugiranih gradijenata

  koje su specijalno prilagođene za tridijagonalnu matricu.
- Kod efikasno implementirane eksplicitne i implicitne metode broj operacija je istog reda veličine, pa rješavanje sustava kod implicitne metode ne predstavlja preveliki dodatni trošak u odnosu na eksplicitnu.

Zadaci

Zadatak

Napišite M-file funkciju `im_mkr_difuzijska()` koja implementira potpunu metodu konačnih razlika za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre

- *korak δx po x koordinati i korak $\delta\tau$ po τ koordinati*
- *maksimalan broj koraka m po τ koordinati*
- *minimalan n_{min} i maksimalan n_{max} indeks čvorova po x koordinati*
- *pokazivač na funkciju inicijalnog uvjeta y_0*
- *pokazivače na funkcije rubnih uvjeta $y_{-\infty}$ i y_{∞}.*

Funkcija neka vraća

- *aproksimativno rješenje y u čvorovima sa x koordinatama $n_{min}\delta x : \delta x : n_{max}\delta x$ i τ koordinatom $\tau = m\delta\tau = \frac{1}{2}\sigma^2 T$.*

Zadatak (nastavak)

Za rješavanje sustava $Ay^{(j+1)} = b^{(j)}$ napišite posebnu funkciju koja implementira Gauss–Seidelovu metodu posebno napravljenu za tridijagonalnu matricu A.

- *Kao kriterij zaustavljanja uzmite*

$$\|y^{(j+1,k)} - y^{(j+1,k-1)}\|_2 \leq 10^{-8},$$

  gdje je $y^{(j+1,k)}$ aproksimacija rješenja sustava u k-tom koraku Gauss–Seidelove metode.
- *Za početnu iteraciju Gauss–Seidelove metode uzmite*

$$y^{(j+1,0)} = y^{(j)}.$$

Zadatak

Svoju funkciju `im_mkr_difuzijska()` *isprobajte na sljedećem primjeru.*

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

$$y(x,0) = \sin(\pi x), \quad y(0,\tau) = y(1,\tau) = 0.$$

Egzaktno rješnje glasi

$$y(x,\tau) = e^{-\pi^2 \tau} \sin(\pi x).$$

Uzmite sljedeće parametre:

- $\delta x = 0.1$
- $n_{min} = 0$ *i* $n_{max} = 10$

Zadatak (nastavak)

Numeričko rješenje pomoću potpune implicitne metode konačnih razlika nađite za dvije različite situacije:

1. *kada je* $\lambda = 0.05$
   - $\delta\tau = 0.0005$
   - $m = 1000$
2. *kada je* $\lambda = 1$
   - $\delta\tau = 0.01$
   - $m = 50$

U oba slučaja $\tau_m = 0.5$. *Dobivena rješenja usporedite sa egzaktnim rješenjem. Što možete zaključiti?*

Zadatak

Napišite M-file funkciju `im_mkr_black_scholes()` *koja numerički rješava Black–Scholesovu jednadžbu pomoću potpune implicitne metode konačnih razlika za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre*

- *dogovorenu cijenu E na datumu dospijeća*
- *vrijeme do dospijeća T izraženo u godinama*
- *kamatnu stopu r*
- *volatilnost σ cijene imovine*
- *maksimalan broj koraka m po τ koordinati za rješavanje difuzijske jednadžbe*
- *minimalan n_{min} i maksimalan n_{max} indeks čvorova po x koordinati za rješavanje difuzijske jednadžbe*
- *parametar $\lambda = \frac{\delta\tau}{(\delta x)^2}$*

Zadatak (nastavak)

- *string vrsta koji označava da li se radi o "call" ili o "put" opciji.*

Funkcija treba odrediti

$$\delta\tau = \frac{\sigma^2 T}{2m}, \quad \delta x = \sqrt{\frac{\delta\tau}{\lambda}}.$$

Funkcija neka vraća

- *polje* V_0 *duljine* $n_{max} - n_{min} + 1$ *koje sadrži aproksimativna rješenja* $V(S_i, 0)$
- *polje* S *duljine* $n_{max} - n_{min} + 1$ *koje sadrži mrežu* $S_i = Ee^{i\delta x}$, $n_{min} \leq i \leq n_{max}$.

Napomena: *Ova funkcija je istovjetna funkciji* `ex_mkr_black_scholes()`, *samo što difuzijsku jednadžbu rješava pomoću potpune implicitne metode konačnih razlika umjesto eksplicitne.*

Zadatak

Svoju funkciju `im_mkr_black_scholes()` *isprobajte na primjeru Black–Scholesovog modela za put opciju sa sljedećim parametrima.*

- $E = 10$
- $T = 0.5$
- $r = 0.05$
- $\sigma = 0.20$
- $m = 100$
- $n_{min} = -1000$ i $n_{max} = 1000$

Zadatak (nastavak)

Numeričko rješenje pomoću potpune implicitne metode konačnih razlika za difuzijsku jednadžbu nađite za tri različite situacije:

1. $\lambda = 0.25$
2. $\lambda = 0.5$
3. $\lambda = 0.55$

Dobivena rješenja usporedite sa rješenjem dobivenim MATLAB-ovom funkcijom `blsprice()`*. Što možete zaključiti?*

Crank–Nicolsonova metoda

- Crank–Nicolsonova metoda je također implicitna metoda koja nema problema sa stabilnošću, ali ima grešku diskretizacije derivacije $\partial y / \partial \tau$ reda veličine $\mathcal{O}\left((\delta\tau)^2\right)$.
- Crank–Nicolsonova metoda računa aproksimaciju rješenja difuzijske jednadžbe u čvorovima mreže tako da uzima srednju vrijednost diferencijskih jednadžbi eksplicitne i potpuno implicitne metode.

- Dakle, ako koristimo konaču razliku unaprijed za $\partial y/\partial\tau$ dobivamo eksplicitnu metodu

$$\frac{y_{i,j+1}-y_{i,j}}{\delta\tau}+\mathcal{O}(\delta\tau)=\frac{y_{i+1,j}-2y_{i,j}+y_{i-1,j}}{(\delta x)^2}+\mathcal{O}\left((\delta x)^2\right),$$

a ako koristimo konaču razliku unazad za $\partial y/\partial\tau$ dobivamo potpunu implicitnu metodu

$$\frac{y_{i,j+1}-y_{i,j}}{\delta\tau}+\mathcal{O}(\delta\tau)=\frac{y_{i+1,j+1}-2y_{i,j+1}+y_{i-1,j+1}}{(\delta x)^2}+\mathcal{O}\left((\delta x)^2\right).$$

- Srednja vrijednost tih dviju jednadžbi je

$$\frac{y_{i,j+1}-y_{i,j}}{\delta\tau}+\mathcal{O}(\delta\tau)=$$

$$=\frac{1}{2}\left(\frac{y_{i+1,j}-2y_{i,j}+y_{i-1,j}}{(\delta x)^2}+\frac{y_{i+1,j+1}-2y_{i,j+1}+y_{i-1,j+1}}{(\delta x)^2}\right)+$$

$$+\mathcal{O}\left((\delta x)^2\right).$$

- Zapravo, može se pokazati da su izrazi u gornjoj jednadžbi točni do na $\mathcal{O}\left((\delta\tau)^2\right)$.
- Zanemarujući izraze $\mathcal{O}(\delta\tau)$ i $\mathcal{O}\left((\delta x)^2\right)$ dobivamo diferencijsku jednadžbu

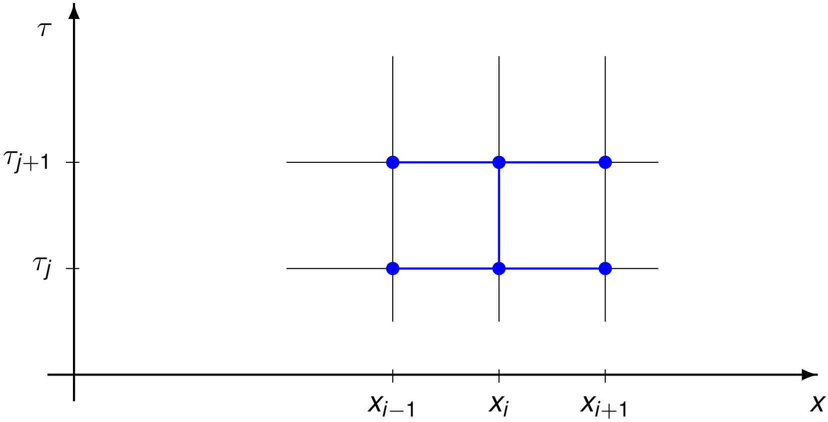
$$-\frac{\lambda}{2}y_{i-1,j+1} + (1+\lambda)y_{i,j+1} - \frac{\lambda}{2}y_{i+1,j+1} =$$

$$= \frac{\lambda}{2}y_{i-1,j} + (1-\lambda)y_{i,j} + \frac{\lambda}{2}y_{i+1,j},$$

gdje je opet

$$\lambda = \frac{\delta\tau}{(\delta x)^2}.$$

- U Crank–Nicolsonovoj metodi $y_{i-1,j+1}$, $y_{i,j+1}$ i $y_{i+1,j+1}$ implicitno ovise o $y_{i-1,j}$, $y_{i,j}$ i $y_{i+1,j}$.



Slika: $y_{i-1,j+1}$, $y_{i,j+1}$ i $y_{i+1,j+1}$ ovise o $y_{i-1,j}$, $y_{i,j}$ i $y_{i+1,j}$.

- Ponovo ćemo riješiti problem za

$$n_{min}\delta x = x_{n_{min}} \leq x_i \leq x_{n_{max}} = n_{max}\delta x,$$

gdje su $-n_{min}$ i n_{max} veliki prirodni brojevi.

- Dalje, segment $\left[0, \frac{\sigma^2 T}{2}\right]$ dijelimo na m jednaka podintervala, tako da je

$$\delta\tau = \frac{\sigma^2 T}{2m}$$

- Sada možemo riješiti diferencijsku jednadžbu za

$$n_{min} < i < n_{max}, \quad 0 < j \leq m.$$

- Rubne uvjete koristimo za određivanje $y_{n_{min},j}$ i $y_{n_{max},j}$:

$$\begin{aligned} y_{n_{min},j} &= y_{-\infty}(n_{min}\delta x, j\delta\tau), & 0 < j \leq m, \\ y_{n_{max},j} &= y_{\infty}(n_{max}\delta x, j\delta\tau), & 0 < j \leq m. \end{aligned}$$

- Za pokretanje ove iterativne metode koristimo inicijalni uvjet

$$y_{i,0} = y_0(i\delta x), \quad n_{min} \leq i \leq n_{max}.$$

- Iterativna metoda završava za $j = m$ i rješenjem

$$y_{i,m}, \quad n_{min} \leq i \leq n_{max},$$

što predstavlja aproksimaciju rješenja za $y(i\delta x, \frac{\sigma^2 T}{2})$, $n_{min} \leq i \leq n_{max}$.

- Također ćemo i ovdje morati rješavati sustave, pa nam opet treba matrični oblik diferencijske jednadžbe.
- Definiramo $(n_{max} - n_{min} - 1) \times (n_{max} - n_{min} - 1)$ matricu A

$$A = \begin{bmatrix} 1+\lambda & -\frac{\lambda}{2} & 0 \cdots & 0 & \\ -\frac{\lambda}{2} & 1+\lambda & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\frac{\lambda}{2} \\ 0 & \cdots & 0 & -\frac{\lambda}{2} & 1+\lambda \end{bmatrix},$$

i $(n_{max} - n_{min} - 1) \times (n_{max} - n_{min} - 1)$ matricu B

$$B = \begin{bmatrix} 1-\lambda & \frac{\lambda}{2} & 0 \cdots & 0 & \\ \frac{\lambda}{2} & 1-\lambda & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \frac{\lambda}{2} \\ 0 & \cdots & 0 & \frac{\lambda}{2} & 1-\lambda \end{bmatrix}.$$

- Isto tako nam još trebaju vektori

$$
\begin{aligned}
y^{(j)} &= \begin{bmatrix} y_{n_{min}+1,j} & \cdots & y_{n_{max}-1,j} \end{bmatrix}^T, \\
y_r^{(j)} &= \lambda \begin{bmatrix} y_{n_{min},j} & 0 & \cdots & 0 & y_{n_{max},j} \end{bmatrix}^T.
\end{aligned}
$$

- Tada Crank–Nicolsonovu metodu možemo napisati u matričnom obliku kao

$$
Ay^{(j+1)} = By^{(j)} + \frac{1}{2}y_r^{(j)} + \frac{1}{2}y_r^{(j+1)} = b^{(j)}.
$$

- Pokazat ćemo da je matrica A regularna pa se korak Crank–Nicolsonove metode može napisati eksplicitno kao

$$
y^{(j+1)} = A^{-1}\left(By^{(j)} + \frac{1}{2}y_r^{(j)} + \frac{1}{2}y_r^{(j+1)}\right).
$$

Algoritam (Crank-Nicolsonova metoda za difuzijsku jednadžbu)

definiraj rješavač s matricom A; $\lambda = \frac{\delta\tau}{(\delta x)^2}$;

for $i = n_{min} : n_{max}$

$\quad y(i) = y_0(i \cdot \delta x);$

end

for $j = 1 : m$

$\quad$ **for** $i = (n_{min} + 1) : (n_{max} - 1)$

$\quad\quad b(i) = \frac{\lambda}{2} y(i-1) + (1-\lambda) y(i) + \frac{\lambda}{2} y(i+1);$

$\quad$ **end**

$\quad y(n_{min}) = y_{-\infty}(n_{min} \cdot \delta x, j \cdot \delta\tau);$

$\quad y(n_{max}) = y_{\infty}(n_{max} \cdot \delta x, j \cdot \delta\tau);$

$\quad b(n_{min} + 1) = b(n_{min} + 1) + \frac{\lambda}{2} \cdot y(n_{min});$

$\quad b(n_{max} - 1) = b(n_{max} - 1) + \frac{\lambda}{2} \cdot y(n_{max});$

$\quad$ *riješi sustav* $Ay(n_{min} + 1 : n_{max} - 1) = b;$

end

Teorem

Pretpostavimo da je $y \in C^4$. Tada je red Crank–Nicolsonove metode jednak $\mathcal{O}\left((\delta\tau)^2\right) + \mathcal{O}\left((\delta x)^2\right)$.

Dokaz.

Preko Taylorovog reda. □

Stabilnost Crank–Nicolsonove metode

- Kao i do sada pretpostavimo da našu numeričku metodu izvršavamo na računalu u aritmetici konačne preciznosti.
- U tom slučaju ćemo u svakom koraku j umjesto $y^{(j+1)}$ izračunati aproksimativnu vrijednost $\tilde{y}^{(j+1)}$ koja sadrži i greške zaokruživanja.
- Neka je

$$\tilde{y}^{(j+1)} = A^{-1}\left(B\tilde{y}^{(j)} + \frac{1}{2}\tilde{y}_r^{(j)} + \frac{1}{2}\tilde{y}_r^{(j+1)}\right) + f^{(j+1)},$$

  gdje $f^{(j+1)}$ sadrži greške zaokruživanja koje su se dogodile kod računanja $A^{-1}\left(B\tilde{y}^{(j)} + \frac{1}{2}\tilde{y}_r^{(j)} + \frac{1}{2}\tilde{y}_r^{(j+1)}\right)$.
- Definirajmo ukupnu grešku

$$e^{(j)} = \tilde{y}^{(j)} - y^{(j)}, \quad j \geq 0.$$

- Možemo zaključiti da vrijedi sljedeće:

$$e^{(j+1)} = A^{-1}Be^{(j)} + \frac{1}{2}A^{-1}(\tilde{y}_r^{(j)} - y_r^{(j)} + \tilde{y}_r^{(j+1)} - y_r^{(j+1)}) + f^{(j+1)}.$$

- Usredotočimo se sada na utjecaj grešaka zaokruživanja $e^{(0)}$ kod računanja inicijalnog uvjeta (za $j = 0$).
- Zbog jednostavnosti, pretpostavimo da su svi daljnji koraci ($j > 0$) izračunati egzaktno, tj. da je

$$\tilde{y}_r^{(j)} = y_r^{(j)}, \quad f^{(j)} = 0, \quad j > 0.$$

- Tada imamo

$$e^{(j+1)} = A^{-1}Be^{(j)} = (A^{-1}B)^{j+1}e^{(0)}.$$

- Da bi metoda bila stabilna, greška $e^{(0)}$ mora biti prigušena, a za to treba biti

$$\lim_{j\to\infty}(A^{-1}B)^j e^{(0)} = 0.$$

- Znamo da će metoda biti stabilna ako i samo ako je

$$\rho(A^{-1}B) < 1,$$

- Dakle, da bi metoda bila stabilna zahtijevamo da za sve svojstvene vrijednosti $\mu_1(A^{-1}B), \ldots, \mu_{n_{max}-n_{min}-1}(A^{-1}B)$ od $A^{-1}B$ vrijedi

$$|\mu_k(A^{-1}B)| < 1, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Tražene svojstvene vrijednosti dobit ćemo iz rastava matrica A i B

$$A = I + \frac{\lambda}{2} \cdot \underbrace{\begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ 0 & & & -1 & 2 \end{bmatrix}}_{=G}.$$

$$B = I - \frac{\lambda}{2} \cdot \underbrace{\begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ 0 & & & -1 & 2 \end{bmatrix}}_{=G} .$$

- Sada jednakost $Ae^{(j+1)} = Be^{(j)}$ možemo napisati kao

$$\begin{aligned} \left(I + \frac{\lambda}{2} G\right) e^{(j+1)} &= \left(I - \frac{\lambda}{2} G\right) e^{(j)} \\ (2I + \lambda G)\, e^{(j+1)} &= (2I - \lambda G)\, e^{(j)} \\ &= (4I - (2I + \lambda G))e^{(j)}. \end{aligned}$$

- Ako definiramo $C = 2I + \lambda G$, tada je

$$Ce^{(j+1)} = (4I - C)e^{(j)},$$

Odnosno

$$e^{(j+1)} = (4C^{-1} - I)e^{(j)}.$$

- Budući da znamo svojstvene vrijednosti $\mu_k(G)$ od matrice G, tada je

$$\mu_k(C) = 2 + \lambda\mu_k(G), \quad k = 1, \ldots, n_{max} - n_{min} - 1,$$

i

$$\mu_k(4C^{-1} - I) = \frac{4}{2 + \lambda\mu_k(G)} - 1, \quad k = 1, \ldots, n_{max} - n_{min} - 1.$$

- Kako je

$$\mu_k(G) = 4\sin^2\left(\frac{k\pi}{2(n_{max} - n_{min})}\right), \; k = 1, \ldots, n_{max} - n_{min} - 1,$$

zbog toga što je $\lambda > 0$, i $\sin^2\left(\frac{k\pi}{2(n_{max}-n_{min})}\right) > 0$ za $k = 1, \ldots, n_{max} - n_{min} - 1$, vrijedi da je

$$\begin{aligned} \mu_k(G) &> 0, \\ \mu_k(C) = 2 + \lambda\mu_k(G) &> 2, \\ 0 < 4\mu_k(C^{-1}) = \frac{4}{\mu_k(C)} = \frac{4}{2 + \lambda\mu_k(G)} &< 2, \\ -1 < \mu_k(4C^{-1} - I) &< 1, \quad {\scriptstyle k=1,\ldots,n_{max}-n_{min}-1.} \end{aligned}$$

- Dakle, za bilo koji $\lambda > 0$ je $|\mu_k(4C^{-1} - I)| < 1$, što znači da je Crank–Nicolsonova metoda bezuvjetno stabilna.

- Kao i kod potpuno implicitne metode, može se vidjeti da su sve svojstvene vrijednosti matrice A pozitivne, što znači da je matrica pozitivno definitna.
- Zbog toga za rješavanje sustava $Ay^{(j+1)} = b^{(j)}$ možemo koristiti metode
  - faktorizaciju Choleskog
  - Gauss–Seidelovu i SOR metodu
  - metodu konjugiranih gradijenata

  koje su specijalno prilagođene za tridijagonalnu matricu.

Zadaci

Zadatak

Napišite M-file funkciju `cn_mkr_difuzijska()` *koja implementira Crank–Nicolsonovu metodu za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre*

- *korak δx po x koordinati i korak $\delta\tau$ po τ koordinati*
- *maksimalan broj koraka m po τ koordinati*
- *minimalan n_{min} i maksimalan n_{max} indeks čvorova po x koordinati*
- *pokazivač na funkciju inicijalnog uvjeta y_0*
- *pokazivače na funkcije rubnih uvjeta $y_{-\infty}$ i y_{∞}.*

Funkcija neka vraća

- *aproksimativno rješenje y u čvorovima sa x koordinatama $n_{min}\delta x : \delta x : n_{max}\delta x$ i τ koordinatom $\tau = m\delta\tau = \frac{1}{2}\sigma^2 T$.*

Zadatak (nastavak)

Za rješavanje sustava $Ay^{(j+1)} = b^{(j)}$ napišite posebnu funkciju koja implementira Gauss–Seidelovu metodu posebno napravljenu za tridijagonalnu matricu A.

- *Kao kriterij zaustavljanja uzmite*

$$\|y^{(j+1,k)} - y^{(j+1,k-1)}\|_2 \leq 10^{-8},$$

  gdje je $y^{(j+1,k)}$ aproksimacija rješenja sustava u k-tom koraku Gauss–Seidelove metode.

- *Za početnu iteraciju Gauss–Seidelove metode uzmite*

$$y^{(j+1,0)} = y^{(j)}.$$

Zadatak

Svoju funkciju `cn_mkr_difuzijska()` *isprobajte na sljedećem primjeru.*

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

$$y(x,0) = \sin(\pi x), \quad y(0,\tau) = y(1,\tau) = 0.$$

Egzaktno rješnje glasi

$$y(x,\tau) = e^{-\pi^2 \tau} \sin(\pi x).$$

Uzmite sljedeće parametre:

- $\delta x = 0.1$
- $n_{min} = 0$ *i* $n_{max} = 10$

Zadatak (nastavak)

Numeričko rješenje pomoću Crank–Nicolsonove metode nađite za dvije različite situacije:

1. *kada je* $\lambda = 0.05$
   - $\delta\tau = 0.0005$
   - $m = 1000$
2. *kada je* $\lambda = 1$
   - $\delta\tau = 0.01$
   - $m = 50$

U oba slučaja $\tau_m = 0.5$. *Dobivena rješenja usporedite sa egzaktnim rješenjem. Što možete zaključiti?*

Zadatak

Napišite M-file funkciju `cn_mkr_black_scholes()` *koja numerički rješava Black–Scholesovu jednadžbu pomoću Crank–Nicolsonove metode za difuzijsku jednadžbu. Funkcija neka ima ulazne parametre*

- *dogovorenu cijenu E na datumu dospijeća*
- *vrijeme do dospijeća T izraženo u godinama*
- *kamatnu stopu r*
- *volatilnost σ cijene imovine*
- *maksimalan broj koraka m po τ koordinati za rješavanje difuzijske jednadžbe*
- *minimalan n_{min} i maksimalan n_{max} indeks čvorova po x koordinati za rješavanje difuzijske jednadžbe*
- *parametar $\lambda = \frac{\delta\tau}{(\delta x)^2}$*

Zadatak (nastavak)

- *string vrsta koji označava da li se radi o "call" ili o "put" opciji.*

Funkcija treba odrediti

$$\delta\tau = \frac{\sigma^2 T}{2m}, \quad \delta x = \sqrt{\frac{\delta\tau}{\lambda}}.$$

Funkcija neka vraća

- *polje* V_0 *duljine* $n_{max} - n_{min} + 1$ *koje sadrži aproksimativna rješenja* $V(S_i, 0)$
- *polje* S *duljine* $n_{max} - n_{min} + 1$ *koje sadrži mrežu* $S_i = Ee^{i\delta x}$, $n_{min} \leq i \leq n_{max}$.

Napomena: *Ova funkcija je istovjetna funkciji* `ex_mkr_black_scholes()`, *samo što difuzijsku jednadžbu rješava pomoću Crank–Nicolsonove metode konačnih razlika umjesto eksplicitne.*

Zadatak

Svoju funkciju `cn_mkr_black_scholes()` *isprobajte na primjeru Black–Scholesovog modela za put opciju sa sljedećim parametrima.*

- $E = 10$
- $T = 0.5$
- $r = 0.05$
- $\sigma = 0.20$
- $m = 100$
- $n_{min} = -1000$ *i* $n_{max} = 1000$

Zadatak (nastavak)

Numeričko rješenje pomoću Crank–Nicolsonove metode za difuzijsku jednadžbu nađite za tri različite situacije:

1. $\lambda = 0.25$
2. $\lambda = 0.5$
3. $\lambda = 0.55$

Dobivena rješenja usporedite sa rješenjem dobivenim MATLAB-ovom funkcijom `blsprice()`*. Što možete zaključiti?*