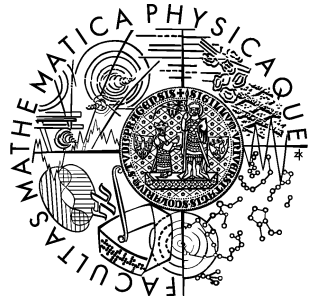
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Jan Voříšek

Numerické metody ve financích

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: Doc. RNDr. Jan Hurt, CSc.,
Studijní program: Matematika
Studijní obor: Finanční matematika

2008

Na tomto místě bych rád poděkoval Doc. RNDr. Janu Hurtovi, CSc. za trpělivost a cenné rady při vedení diplomové práce.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce a jejím zveřejňováním.

V Praze dne 7. srpna 2008 Jan Voříšek

Obsah

1 Úvod **5**

2 Opce **6**
2.1 Základní charakteristika 6
2.2 Black-Scholesova rovnice 7

3 Iterační metody **9**
3.1 Základy 9
3.2 Metody 10

4 Metoda konečných diferencí **12**
4.1 Aproximace derivací 12
4.2 Síť souřadnic 12
4.3 Metody výpočtu 13
4.4 Okrajové podmínky 16
4.5 Lineární komplementarita pro americké opce 20
4.6 Metoda maximální okrajové podmínky 23

5 Metoda konečných prvků **26**
5.1 Vážená rezidua 26
5.2 Galerkinův přístup 28
5.3 Aplikace na standardní opce 30

6 Výpočty v systému Mathematica **35**
6.1 Funkce 35
6.2 Evropské opce 38
6.3 Americké opce 41

7 Závěr **47**

Literatura **48**

Název práce: Numerické metody ve financích
Autor: Jan Voříšek
Katedra: Katedra pravděpodobnosti a matematické statistiky
Vedoucí diplomové práce: Doc. RNDr. Jan Hurt, CSc.
e-mail vedoucího: hurt@karlin.mff.cuni.cz

Abstrakt: V předložené práci studujeme metody konečných diferencí, konečných prvků a numerické algoritmy, které využijeme při aproximování plochy, kterou tvoří cena opce v závislosti na době do splatnosti a ceně podkladového aktiva. Evropské opce splňují Black-Scholesovu rovnici, americké opce splňují nerovnost Black-Scholesova typu. Plochu tvořenou cenou opce aproximujeme pomocí diskrétních hodnot vypočtených různými metodami a numerickými algoritmy. Výpočty pomocí zabudovaných funkcí systému Mathematica využijeme při hodnocení přesnosti navržených metod a srovnání teoretických výsledků s tržními hodnotami opcí. Nejvhodnější algoritmus pak použijeme pro ukázkovou demonstraci výsledků a grafických možností systému Mathematica.

Klíčová slova: Black-Scholes, opce, Mathematica, konečné diference, konečné prvky

Title: Numerical methods in finance
Author: Jan Voříšek
Department: Department of Probability and Mathematical Statistics
Supervisor: Doc. RNDr. Jan Hurt, CSc.
Supervisor's e-mail address: hurt@karlin.mff.cuni.cz

Abstract: In the present thesis we study the finite differences method, the finite element method and numerical algorithms used for the surface approximation of the option price with respect to the price of underlying asset and the time to maturity. The Black-Scholes equation gives the price of an European option, American options obey inequality of the Black-Scholes type. The surface is approximated by discrete values which may be calculated by different methods and numerical algorithms. The calculations using implemented functions in the framework Mathematica are used for the accuracy check and comparing theoretical results with the options market values. Most suitable algorithm will be used for implementing onto Mathematica demonstration.

Keywords: Black-Scholes, options, Mathematika, finite differences, finite element

Kapitola 1

Úvod

Vedle nejběžnějších obchodů, kde se platba i dodání provádějí okamžitě po uzavření dohody, je možné uzavřít dohodu, která umožňuje prodat nebo koupit určité aktivum v budoucnosti za danou cenu. Prvními předměty takto obchodovanými byly například kovy, palivo nebo zemědělské produkty. Později se tento způsob obchodování rozšířil na akcie, měnové kurzy a další finanční instrumenty. V dnešní době jsou možnosti „obchodování s budoucností“ prakticky neomezené, týkající se například počasí.

Na finančním trhu se těmto dohodám o budoucím obchodě říká deriváty, protože jejich hodnota a další charakteristiky záleží na podkladovém aktivu. Mezi hlavní typy derivátů patří *futures*, *forwards*, *opce* a *swapy*. Deriváty se obchodují na speciálních trzích. K nejznámějším patří London International Financial Futures and Options Exchange (LIFFE), Chicago Board of Options Exchange (CBOE) a Chicago Mercantile Exchange (CME) otevřené v roce 1973.

Ve stejném roce byla poprvé prezentována Black-Scholesova parciální diferenciální rovnice pro evropské opce, která za určitých idealizujících předpokladů umožňuje počítat cenu opce analyticky. Hodnotu amerických opcí, které splňují nerovnost Black-Scholesova typu, je nutné počítat numericky.

Numerické výpočty týkající se oceňování amerických opcí tvoří hlavní část této práce. Při výpočtech a hodnocení algoritmů budeme využívat možností systému *Mathematica*, který zároveň obsahuje zabudované funkce pro numerické výpočty. Při výpočtu plochy, kterou tvoří cena opce splňující Black-Scholesovu diferenciální rovnici v závislosti na době do splatnosti a ceně podkladového aktiva, můžeme využít funkce *NDSolve*. Další zabudované funkce využijeme ve vlastních algoritmech pro numerické výpočty Black-Scholesova modelu pro evropské a zejména americké opce. Výsledky srovnáme s tržními cenami opcí na index NASDAQ100.

Na přiloženém CD jsou uloženy soubory s daty a výpočty v programu *Mathematica*.

Kapitola 2

Opce

2.1 Základní charakteristika

Opce dává majiteli právo prodat nebo koupit za dohodnutou realizační cenu K určité aktivum, kterým může být například akcie, měna, index nebo zboží. Kupující platí prodávajícímu náhradu za riziko, které vzniká závazkem k prodeji při nejisté budoucí ceně podkladového aktiva. Vlastník opce se v daném časovém horizontu daném dobou splatnosti T opce podle tržní situace rozhoduje, jestli opci realizuje. Put opce opravňuje k prodeji podkladového aktiva za realizační cenu, zatímco call dává držiteli možnost podkladové aktivum koupit. V případě evropských opcí platí toto právo pouze v době splatnosti T. Pro americké opce platí právo realizace kdykoli v čase $t \leq T$ do doby splatnosti. Opce na akcie bývají většinou americké. Cena opce $V(S,t)$ závisí na době do splatnosti $T-t$ a ceně podkladového aktiva $S(t)$ v čase t. Na prostoru $S(t) \geq 0$, $0 \leq t \leq T$ tvoří cena opce $V(S,t)$ plochu, pro kterou můžeme snadno určit některé okrajové podmínky, které později využijeme při výpočtech. Dále se budeme zabývat PUT opcemi z pozice majitele. Hodnotu opce v čase splatnosti $t = T$, která je stejná pro evropské i americké opce nazýváme výplatní funkcí. Pokud je hodnota $S < K$, má opce hodnotu $K - S$, v opačném případě je opce bezcenná. V matematickém zápisu

$$V(S,T) = \max(K - S, 0).$$

Americká opce je rozšířením evropské o možnost dřívější realizace v čase $t < T$ proto je její hodnota

$$V^{am} \geq V^{ev}$$

pro libovolné t a S. Při odvození dalších okrajových podmínek uvažujeme trh bez možnosti *arbitráže*. Arbitráž znamená existenci portfolia, které při nulové počáteční investici garantuje zisk při splatnosti. V praxi si investor na takové portfolio půjčí peníze za bezrizikovou úrokovou míru r a zároveň hodnota portfolia při splatnosti bude větší než půjčka úročená bezrizikovou úrokovou mírou. Čím větší bude tento zisk tím více investorů jej bude realizovat, což vede k rychlému snížení bezrizikového zisku až

na nulu. Z principu neexistence arbitráže plyne další podmínka pro americké opce

$$V^{am}(S,t) \geq \max(K-S,0).$$

Pokud by hodnota V byla menší než $K-S$, při nákupu opce a akcie za cenu $V+S$, by okamžitá realizace znamenala výnos $K-S-V$, což je spor s neexistencí arbitráže. Obdobně odvozená podmínka

$$V^{ev}(S,t) \geq \max(Ke^{-r(T-t)}-S,0)$$

platí pro evropské opce na aktivum bez výplaty dividend. Zde musíme brát v úvahu možnost realizace pouze v době splatnosti T a proto diskontujeme hodnotu realizační ceny.

2.2 Black-Scholesova rovnice

Black-Scholesova rovnice zní:

$$\frac{\partial V(S,t)}{\partial t}+\frac{\sigma^2}{2}S^2\frac{\partial^2 V(S,t)}{\partial S^2}+(r-\delta)S\frac{\partial V(S,t)}{\partial S}-rV(S,t)=0. \tag{2.1}$$

Je to parciální diferenciální rovnice pro výpočet hodnoty opce $V(S,t)$ je odvozená za následujících předpokladů o trhu:

(a) Neexistence arbitráže

(b) Neexistence transakčních nákladů

(c) Žádný investor nemá tak velký kapitál aby jeho chování ovlivnilo ceny

(d) Všichni investoři mají okamžitý přístup ke všem informacím

(e) Všechna aktiva jsou obchodovatelná a absolutně dělitelná

(f) Existuje pouze jedna bezriziková úroková míra pro všechny investory pro půjčování a ukládání kapitálu

a předpokladů o podkladovém aktivu:

(A) bezriziková úroková míra r a volatilita podkladového aktiva σ jsou konstantní v čase $0 \leq t \leq T$. Dividendy jsou vyplácené spojitě v konstantní výši $\delta \geq 0$. Tato spojitá výplata snižuje hodnotu S v každém okamžiku dt o $\delta S dt$

(B) Cena podkladového aktiva se řídí standardním modelem geometrického Brownova pohybu vyjádřeným stochastickou diferenciální rovnicí, která po úpravě o spojité vyplácení dividend je

$$dS(t) = (\mu-\delta)S(t)\,dt+\sigma S(t)\,dW(t), \tag{2.2}$$

kde $W(t)$ je Wienerův proces.

Výhodou Black-Scholesovi rovnice je, že se v ní neobjevuje očekávaný růst akcie μ. Black-Scholesovu rovnici můžeme transformovat použitím substitucí

$$\begin{gathered} S = Ke^x, \quad t = T - \frac{\tau}{\frac{1}{2}\sigma^2}, \quad q := \frac{2r}{\sigma^2}, \quad q_\delta := \frac{2(r-\delta)}{\sigma^2}, \\ V(S,t) = V(Ke^x, T - \frac{2\tau}{\sigma^2}) =: v(x,\tau), \\ v(x,\tau) := K \ \exp\{-\tfrac{1}{2}(q_\delta - 1)x - (\tfrac{1}{4}(q_\delta - 1)^2 + q)\tau\} \ y(x,\tau) \end{gathered} \tag{2.3}$$

na diferenciální rovnici vedení tepla

$$\frac{\partial y(x,\tau)}{\partial \tau} = \frac{\partial^2 y(x,\tau)}{\partial x^2}, \tag{2.4}$$

kde $0 \leq \tau, \ x \in \mathbf{R}$. Podle postupu uvedeném v [6] nejdříve dosadíme za S

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}\frac{\partial^2 V}{\partial x^2} + (r - \delta - \frac{\sigma^2}{2})\frac{\partial V}{\partial x} - rV = 0,$$

následně použijeme substituci za q a q_δ

$$\frac{\partial^2 V}{\partial x^2} + (q_\delta - 1)\frac{\partial V}{\partial x} - qV = -\frac{2}{\sigma^2}\frac{\partial V}{\partial t},$$

použijeme novou časovou proměnnou τ

$$\frac{\partial^2 V}{\partial x^2} + (q_\delta - 1)\frac{\partial V}{\partial x} - qV = \frac{\partial V}{\partial \tau},$$

v posledním kroku se po dosazení za $V(S,t)$ a krácení dojdeme k výrazu (2.4).

Transformovaná diferenciální rovnice (2.4) je narozdíl od Black-Scholesovi rovnice (2.1) lineární, což usnadňuje implementaci numerických algoritmů. Nevýhodou při výpočtu je nutnost provádět transformaci. Při použití metody konečných diferencí navíc exponenciální transformace vede k různé vzdálenosti aproximovaných bodů plochy tvořené cenou opce v závislosti na čase a ceně podkladového aktiva. Tuto nevýhodu můžeme eliminovat použitím metody konečných prvků, která umožňuje vzdálenosti mezi aproximovanými body měnit.

Kapitola 3

Iterační metody

3.1 Základy

Pro řešení systému lineárních rovnic $Ax = b$ v $\mathbb{R}^n$ použijeme úpravu do tvaru

$$Mx = (M - A)x + b,$$

kde M je vhodná matice. Pro regulární matici M platí vzorec podobný rovnici pevného bodu lineárního zobrazení

$$x = (I - M^{-1}A)x + M^{-1}b.$$

Zavedením označení $B := (I - M^{-1}A)$ dostaneme iteraci

$$x^{(k+1)} = Bx^{(k)} + M^{-1}b, \tag{3.1}$$

kterou využijeme při důkazu konvergence metody. Během každého kroku výpočtu dochází ke vzniku chyby v důsledku zaokrouhlování. Teoretické hodnoty bez zaokrouhlovací chyby označuje vektor $x^{(k)}$, zatímco vypočtené hodnoty označíme $\bar{x}^{(k)}$ a chybu vzniklou zaokrouhlováním $e^{(k)} = \bar{x}^{(k)} - x^{(k)}$. Skutečně vypočtenou hodnotu zapíšeme ve tvaru

$$\bar{x}^{(k+1)} = B\bar{x}^{(k)} + M^{-1}b + r^{(k+1)},$$

kde $r^{(k+1)}$ představuje zaokrouhlovací chybu vzniklou během výpočtu $B\bar{x}^{(k)}$. Pro jednoduchost předpokládáme, že $r^{(k)} = 0$ pro všechny $k > 1$. Uvažujeme pouze chybu $e^{(0)}$ vzniklou při výpočtu $\bar{x}^{(0)}$. Potom platí

$$\bar{x}^{(k+1)} = B\bar{x}^{(k)} + M^{-1}b,$$

$$Be^{(k)} = B\bar{x}^{(k)} - Bx^{(k)} + M^{-1}b - M^{-1}b = \bar{x}^{(k+1)} - x^{(k+1)} = e^{(k+1)}.$$

Chyba v k-tém kroku výpočtu tak bude $e^{(k)} = B^k e^{(0)}$. Pro konvergenci potřebujeme zaokrouhlovací chyby utlumit, což vede k požadavku $B^k \to 0$ pro $k \to \infty$, který je

splněn například v případě, když jsou všechny prvky matice B v absolutní hodnotě menší než jedna. Tento požadavek je ekvivalentní $\rho(B) < 1$, což je spektrální rádius matice B (důkaz viz. [5]). Pro přirozené maticové normy platí, že $\|B\| \geq \rho(B)$, tedy $\|B\| < 1$ implikuje konvergenci. Maticové normy

$$\|B\|_\infty = \max_i \sum_{j=1}^{n} |b_{ij}|, \quad \|B\|_1 = \max_j \sum_{i=1}^{n} |b_{ij}|,$$

vedou k postačujícím podmínkám pro konvergenci

$$\sum_{j=1}^{n} |b_{ij}| < 1 \quad \text{pro} \quad 1 \leq i \leq n \quad \text{nebo} \quad \sum_{i=1}^{n} |b_{ij}| < 1 \quad \text{pro} \quad 1 \leq j \leq n.$$

Výpočet $x^{(k+1)}$ se provádí řešením soustavy $Mx^{(k+1)} = (M - A)x^{(k)} + b$. To klade požadavky na matici M vzhledem k rychlosti konvergence (3.1) a zároveň nízké náročnosti řešení soustavy pro $x^{(k+1)}$. Jednoduchý příklad je je rozdělení matice A aditivním způsobem $A = D - L - U$, kde D je diagonální, L dolní trojúhelníková a U horní trojúhelníková matice.

3.2 Metody

Jacobiho metoda
Zavedení $M := D$ implikuje $M - A = L + U$ a znamená iteraci

$$Dx^{(k+1)} = (L + U)x^{(k)} + b.$$

Označíme $B_J = D^{-1}(L + U)$.

Gauß-Seidelova metoda
Zde je zvoleno $M := D - L$, což implikuje $M - A = U$ a znamená iteraci

$$(D - L)x^{(k+1)} = Ux^{(k)} + b.$$

Použijeme označení $B_{GS} = (D - L)^{-1}U$.

Super-relaxační metoda
Podle [4] je super-relaxační metoda (successive overrelaxation, dále SOR) modifikací Gauß-Seidelovy metody použitím relaxačního parametru ω_R, který urychluje konvergenci:

$$M := \frac{1}{\omega_R} D - L \Longrightarrow M - A = \left(\frac{1}{\omega_R} - 1\right) D + U$$

$$\left(\frac{1}{\omega_R}D - L\right)x^{(k+1)} = \left(\left(\frac{1}{\omega_R} - 1\right)D + U\right)x^{(k)} + b.$$

Označíme $B_R := \left(\frac{1}{\omega_R}D - L\right)^{-1}\left(\left(\frac{1}{\omega_R} - 1\right)D + U\right)$, SOR-metoda pak vypadá následovně:

$$x^{(k+1)} = B_R x^{(k)} + \left(\frac{1}{\omega_R}D - L\right)^{-1} b$$

Gauß-Seidelova metoda je speciální případ SOR pro $\omega_R = 1$.

Volba ω_R
Pro vektory $d^{(k+1)} := x^{(k+1)} - x^{(k)}$ platí

$$d^{(k+1)} = B_R d^{(k)}. \tag{3.2}$$

Podle [5] $d^{(k)}$ konverguje k vlastnímu vektoru dominantní vlastí hodnoty $\rho(B_R)$. Tedy pokud (3.2) konverguje, pak

$$d^{(k+1)} = B_R d^{(k)} \approx \rho(B_R) d^{(k)}.$$

Potom $|\rho(B_R)| \approx \frac{\|d^{(k+1)}\|}{\|d^{(k)}\|}$ pro libovolnou vektorovou normu. Podle [7] pro určitou třídu matic A platí

$$\rho(B_{GS}) = \rho(B_J)^2,$$
$$\omega_{opt} = \frac{2}{1 + \sqrt{1 - \rho(B_J)^2}}.$$

Podle praktických zkušeností [5] podává hodnota $\omega_R = 1$ dobré výsledky.

Kapitola 4

Metoda konečných diferencí

4.1 Aproximace derivací

Zavedeme posloupnost diskrétních bodů $x_i \in \mathbf{R}$ takovou, že $x_{i-1} < x_i$ se vzdáleností h mezi jednotlivými body. Každá funkce $f(x)$ se spojitou druhou derivací splňuje

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \frac{h}{2} f''(\xi),$$

kde $x < \xi < x + h$. Tendo vztah vychází z Taylorova rozvoje a pro $h \to 0$ odpovídá definici derivace. S využitím kratšího zápisu $f_i := f(x_i)$ označíme

$$f'(x_i) = \frac{f_{i+1} - f_i}{h} + \mathcal{O}(h). \tag{4.1}$$

Analogické výrazy platí pro parciální derivace $y(x, \tau)$ a $V(t, S)$, pro x, τ, t, S převedené na diskrétní posloupnosti. V tomto případě nahradíme h v závislosti na prováděné aproximaci $\Delta x, \Delta\tau, \Delta t, \Delta S$. Z Taylorova rozvoje je možno odvodit aproximace vyššího řádu. Nesymetrickou aproximaci (4.1) lze nahradit symetrickou.

$$f'(x_i) = \frac{f_{i+1} - f_{i-1}}{2h} + \mathcal{O}(h^2), \qquad (\text{pro} f \in \mathcal{C}^3)$$

$$f''(x_i) = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + \mathcal{O}(h^2), \qquad (\text{pro} f \in \mathcal{C}^4)$$

4.2 Síť souřadnic

Nechť Δx a $\Delta\tau$ jsou vzdálenosti mezi body x_i a τ_ν a Δt a ΔS vzdálenosti mezi body t_n a S_j. Velikost kroku pro τ je $\Delta\tau := \tau_{\max}/Nt$, kde $\tau_{\max} := \frac{1}{2}\sigma^2 T$ a Nt zvolený maximální index. Velikost kroku t je $\Delta t := T/Nt$. Při volbě diskretizace pro x musíme interval $-\infty < x < \infty$ převést na konečný interval $a \leq x \leq b$. Krajní body intervalu

$a = x_{\min} < 0$ a $b = x_{\max} > 0$ musí být zvoleny tak, aby $S_{\min} = Ke^a$ a $S_{\max} = Ke^b$ tvořící interval $S_{\min} \leq S \leq S_{\max}$ zajišťovaly dostatečnou kvalitu aproximace. Pro zvolený index m je pak délka kroku $\Delta x := (b-a)/m$. Obdobně volíme omezíme intervalu pro $0 < S_0 \leq S \leq S_m < \infty$ s délkou kroku $\Delta S := (S_1 - S_m)/m$. Matematický zápis sítě souřadnic je:

$$\begin{array}{lll} t_n & := n \cdot \Delta t & \text{pro } n = 0, 1, \ldots, Nt \\ S_j & := S_0 + j \cdot \Delta S & \text{pro } j = 0, 1, \ldots, m \\ V_{jn} & := V(S_j, t_n) & \\ u_{jn} & \text{aproximace } V_{jn} & \end{array} \tag{4.2}$$

$$\begin{array}{llll} \tau_\nu & := \nu \cdot \Delta\tau & \text{pro} & \nu = 0, 1, \ldots, Nt \\ x_i & := a + i \cdot \Delta x & \text{pro} & i = 0, 1, \ldots, m \\ y_{i\nu} & := y(x_i, \tau_\nu) & & \\ w_{i\nu} & \text{aproximace } y_{i\nu} & & \end{array}$$

Takto definovaná síť souřadnic je ekvidistantní pro x a τ. Po transformaci (x, τ)-mřížky zpět na (S, t)-mřížku nebudou souřadnice $S_i = Ke^{x_i}$ stejně vzdálené. Směrem k $S_{\min}$ jejich hustota poroste, což je nevýhodné pro přesnost aproximace $V(S, t)$. Tomuto problému se při použití konečných diferencí přímo na rovnici (2.1) vyhneme. Průsečíky souřadnic τ_ν s x_i a t_n s S_j se nazývají *uzly*. Souřadnice budeme pro zjednodušení označovat pouze indexy ν a i nebo n a j.

Oproti teoretickým řešením $y(x, \tau)$, $V(S, t)$ definovaným na spojitých intervalech jsou aproximace $w_{i\nu}$, u_{jn} definované pouze v uzlech. Velikost chyby způsobené diskretizací $w_{i\nu} - y_{i\nu}$ závisí na volbě parametrů $Nt, m, x_{\min}, x_{\max}$ a velikost chyby $u_{jn} - V_{jn}$ na Nt, m, S_0, S_1. Volba $x_{\min} = -5,\ x_{\max} = 5$ je podle [5] dostatečná. Velikost chyby se pak kontroluje změnou hustoty dělení intervalů. Bohužel dopředu nevíme, jaká volba parametrů Nt, m povede k požadované přesnosti.

4.3 Metody výpočtu

Metoda dopředných a zpětných diferencí
Dosazením

$$\frac{\partial y_{i\nu}}{\partial \tau} = \frac{y_{i,\nu+1} - y_{i\nu}}{\Delta\tau} + \mathcal{O}(\Delta\tau)$$

$$\frac{\partial^2 y_{i\nu}}{\partial x^2} = \frac{y_{i+1,\nu} - 2y_{i\nu} + y_{i-1,\nu}}{\Delta x^2} + \mathcal{O}(\Delta x^2)$$

do vztahu (2.4) a zanedbáním chyb dostaneme rovnici

$$\frac{w_{i,\nu+1} - w_{i\nu}}{\Delta\tau} = \frac{w_{i+1,\nu} - 2w_{i\nu} + w_{i-1,\nu}}{\Delta x^2}$$

pro aproximaci w. Vyjádřením $w_{i,\nu+1}$ dostaneme

$$w_{i,\nu+1} = w_{i,\nu} + \frac{\Delta\tau}{\Delta x^2}(w_{i+1,\nu} - 2w_{i\nu} + w_{i-1,\nu}).$$

Po substituci $\lambda := \frac{\Delta\tau}{\Delta x^2}$ dostaneme upravenou rovnici

$$w_{i,\nu+1} = \lambda(w_{i+1,\nu} + w_{i-1,\nu}) + (1 - 2\lambda)w_{i,\nu}. \tag{4.3}$$

Protože vzorec (4.3) představuje explicitní metodu jak vypočítat $w_{i,\nu+1}$, je tato metoda pojmenována metoda dopředných diferencí. Pro pevné hodnoty aproximace w v časové vrstvě ν vypočítá všechny body v časové vrstvě $\nu + 1$. Pro $\nu = 0$ jsou hodnoty w_{i0} dány počátečními podmínkami

$$w_{i0} = y(x_i, 0) \qquad \text{pro } y \text{ z (4.4)}, \ 0 \leq i \leq m.$$

Krajní body $w_{0\nu}$ a $w_{m\nu}$ jsou dány hraničními podmínkami. Prozatím budeme uvažovat $w_{0\nu} = w_{m\nu} = 0$. Pro zjednodušení dalších výpočtů je vhodné zavést vektor všech hodnot w v čase ν

$$w^{(\nu)} := (w_{1\nu}, \ldots, w_{m-1,\nu})^\top$$

a konstantní třídiagonální matici o rozměrech $(m-1) \times (m-1)$

$$A = \begin{pmatrix} 1-2\lambda & \lambda & & & 0 \\ \lambda & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \\ 0 & & & \ddots & \ddots \end{pmatrix}$$

Potom můžeme vyjádřit řešení w v čase ν jako

$$w^{(\nu+1)} = Aw^{(\nu)} \qquad \text{pro } \nu = 0, 1, \ldots \tag{4.4}$$

Pokud bychom dosadili

$$\frac{\partial y_{i\nu}}{\partial\tau} = \frac{y_{i,\nu} - y_{i\nu-1}}{\Delta\tau} + \mathcal{O}(\Delta\tau)$$

dostaneme implicitní alternativu ke vztahu (4.3)

$$-\lambda(w_{i+1,\nu} + w_{i-1,\nu}) + (1+2\lambda)w_{i,\nu} = w_{i,\nu-1} \tag{4.5}$$

pojmenovanou metoda zpětných diferencí.

Rovnice (4.5) udává vztah času ν vzhledem k času $\nu - 1$. Známe pouze hodnotu $w_{i,\nu-1}$ na pravé straně rovnice, zatímco na levé straně jsou tři neznámé hodnoty w. Nemáme žádnou jednoduchou jednoznačnou metodu, jak vypočítat hodnoty jednu po druhé, proto musíme uvažovat současně všechny rovnice. Pomocí matice

$$B = \begin{pmatrix} 1+2\lambda & -\lambda & & & 0 \\ -\lambda & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \\ 0 & & & \ddots & \ddots \end{pmatrix}$$

je vektor $w^{(\nu)}$ definován jako řešení systému lineárních rovnic

$$Bw^{(\nu)} = w^{(\nu-1)} \qquad \text{pro } \nu = 1, \ldots, Nt. \tag{4.6}$$

Tento výpočet musíme provést pro každý čas ν. Opět uvažujeme podmínky $w_{0\nu} = w_{m\nu} = 0$. Časová náročnost implicitní metody je nízká, protože matice B je třídiagonální a konstantní. Nejprve jednou provedeme LR rozklad pro $\nu = 0$. Dále je náročnost výpočtu pro každé ν pouze řádu $\mathcal{O}(m)$.

Crank-Nicolsonova metoda

Pro předchozí metody výpočtu platí, že diskretizace $\frac{\partial y}{\partial \tau}$ je řádu $\mathcal{O}(\tau)$. Pro Crank-Nicolsonovu metodu je diskretizace řádu $\mathcal{O}(\tau^2)$. Opět uvažujeme vztah (2.4) a použijeme oba předcházející přístupy:
dopředný pro ν:

$$\frac{w_{i,\nu+1} - w_{i\nu}}{\Delta\tau} = \frac{y_{i+1,\nu} - 2y_{i\nu} + y_{i-1,\nu}}{\Delta x^2}$$

zpětný pro $\nu + 1$:

$$\frac{w_{i,\nu+1} - w_{i\nu}}{\Delta\tau} = \frac{y_{i+1,\nu+1} - 2y_{i,\nu+1} + y_{i-1,\nu+1}}{\Delta x^2},$$

což v součtu dává

$$\frac{w_{i,\nu+1} - w_{i\nu}}{\Delta\tau} = \frac{1}{2\Delta x^2}(y_{i+1,\nu} - 2y_{i\nu} + y_{i-1,\nu} + y_{i+1,\nu+1} - 2y_{i\nu+1} + y_{i-1,\nu+1}). \tag{4.7}$$

Rovnice (4.7) obsahuje tři neznámé hodnoty w v čase ν i $\nu+1$ a tvoří základ používané metody. Pro $\lambda := \frac{\Delta\tau}{\Delta x^2}$ můžeme rovnici (4.7) přepsat na

$$
\begin{aligned}
&-\tfrac{\lambda}{2}w_{i-1,\nu+1} + (1+\lambda)w_{i,\nu+1} - \tfrac{\lambda}{2}w_{i+1,\nu+1} \\
&\quad = \tfrac{\lambda}{2}w_{i-1,\nu} + (1-\lambda)w_{i\nu} + \tfrac{\lambda}{2}w_{i+1,\nu}.
\end{aligned} \tag{4.8}
$$

Pro hraniční podmínky $w_{0\nu} = w_{m\nu} = 0$ je rovnice (4.8) soustavou $m-1$ rovnic. Pomocí matic

$$
A = \begin{pmatrix}
1+\lambda & -\frac{\lambda}{2} & & & 0 \\
-\frac{\lambda}{2} & \ddots & \ddots & & \\
 & \ddots & \ddots & \ddots & \\
 & & \ddots & \ddots & \ddots \\
0 & & & \ddots & \ddots
\end{pmatrix}, B = \begin{pmatrix}
1-\lambda & \frac{\lambda}{2} & & & 0 \\
\frac{\lambda}{2} & \ddots & \ddots & & \\
 & \ddots & \ddots & \ddots & \\
 & & \ddots & \ddots & \ddots \\
0 & & & \ddots & \ddots
\end{pmatrix}
$$

můžeme soustavu zapsat ve tvaru

$$
Aw^{(\nu+1)} = Bw^{(\nu)}. \tag{4.9}
$$

Vlastní hodnoty matice A jsou reálná čísla z intervalu $[1, 1+2\lambda]$. Protože matice A nemá vlastní hodnotu nula, bude podle [5] regulární a řešení (4.9) jednoznačné. Pro explicitní metodou je potřeba volit parametry $0 < \lambda \leq \frac{1}{2}$ aby byl výpočet stabilní. Pro $\lambda = \frac{\Delta\tau}{\Delta x^2}$ je podmínka stability ve tvaru $0 < \Delta\tau \leq \frac{\Delta x^2}{2}$.

4.4 Okrajové podmínky

Po transformaci se výplatní funkce době splatnosti změní na okrajovou podmínku pro $y(x, 0)$. Pro put opci platí

$$
V(S,T) = \max(K - S, 0) = K \cdot \max(1 - e^x, 0).
$$

Podle transformace (2.3) dostaneme

$$
\begin{aligned}
y_P(x,0) &= \max(e^{\frac{x}{2}(q_\delta - 1)} - e^{\frac{x}{2}(q_\delta + 1)}, 0) \\
y_C(x,0) &= \max(e^{\frac{x}{2}(q_\delta + 1)} - e^{\frac{x}{2}(q_\delta - 1)}, 0).
\end{aligned} \tag{4.10}
$$

Black-Scholesova rovnice, její transformace (2.4) i diskretizovaná verze potřebují okrajové podmínky:

$V(S,t)$ pro $S = 0$ a $S \to \infty$ nebo
$y(x,\tau)$ pro $x_{\min}$ a $x_{\max}$ nebo
$w_{0\nu}$ a $w_{m\nu}$ pro $\nu = 1, \ldots, Nt$.

Z okrajové podmínky pro čas $t = T$ můžeme odvodit hraniční podmínky pro $t < T$. S ohledem na výplatní funkci platí

$$\begin{aligned} V_C(S,t) &= 0 \quad \text{pro } S = 0, \\ V_P(S,t) &= 0 \quad \text{pro } S \to \infty \end{aligned} \tag{4.11}$$

i pro $t < T$, protože diskontování nezmění nulovou hodnotu. $S(0) = 0$ implikuje $S(t) = 0$ a proto $V_C(0,t) = 0$. Stejně tak pro $S(0) \to \infty$ se $V_P(\infty, t) = 0$. Toto platí pro evropské i pro americké opce bez ohledu na vyplácení dividend.

Okrajové podmínky pro evropské opce
Zatím neznámé okrajové podmínky pro evropské opce bez vyplácení dividend ($\delta = 0$), kde se $V \neq 0$, mohou být odvozeny z put-call parity.

$$\begin{aligned} V_C(S,t) &= S - Ke^{-r(T-t)} \quad \text{pro } S \to \infty \\ V_P(S,t) &= Ke^{-r(T-t)} - S \quad \text{pro } S \approx 0 \end{aligned} \tag{4.12}$$

V (4.12) nezanedbáme $S \approx 0$, protože potřebujeme $S_{\min} > 0$ kvůli možnosti zadat okrajovou podmínku $x_{\min} > -\infty$. Analogické podmínky jako (4.12) platí v případě spojitého vyplácení dividend ($\delta \neq 0$). V takovém případě při transformaci nahradíme $S = \overline{S}e^{\delta(T-t)}$. Pro výpočet pomocí funkce *NDSolve* ponecháme transformované podmínky $y(x,\tau) = r_1(x,\tau),\ \text{pro } x \to -\infty,\ y(x,\tau) = r_2(x,\tau),\ \text{pro } x \to \infty,$

$$\begin{aligned} \text{call: } r_2(x,\tau) &:= \ (\exp(x + \tau(2q - q_\delta)) - 1)\ \exp(\tfrac{1}{2}(q_\delta + 1)x + \tfrac{1}{4}(q_\delta + 1)^2\tau) \\ r_1(x,\tau) &:= 0 \\ \text{put: } r_1(x,\tau) &:= \ (1 - \exp(x + \tau(2q - q_\delta)))\ \exp(\tfrac{1}{2}(q_\delta - 1)x + \tfrac{1}{4}(q_\delta - 1)^2\tau) \\ r_2(x,\tau) &:= 0 \end{aligned} \tag{4.13}$$

v kompletní podobě. Pro praktické řešení uvažujeme, že první části výrazu budou rovné nule jako důsledek $x \to -\infty$, i když se omezíme se na konečný interval $a := x_{\min} \leq x \leq x_{\max} =: b$. Po zanedbání dostáváme okrajové podmínky

$$\begin{aligned} &\text{call: } r_1(x,\tau) := 0, \quad r_2(x,\tau) := \exp(\tfrac{1}{2}(q_\delta + 1)x + \tfrac{1}{4}(q_\delta + 1)^2\tau) \\ &\text{put: } r_1(x,\tau) := \exp(\tfrac{1}{2}(q_\delta - 1)x + \tfrac{1}{4}(q_\delta - 1)^2\tau), \quad r_2(x,\tau) := 0. \end{aligned} \tag{4.14}$$

Pro diskretizaci platí $w_{0\nu} = r_1(a, \tau_\nu), \quad w_{m\nu} = r_2(b, \tau_\nu)$ pro všechny ν. V Crank-Nicolsonově rovnici (4.8), kde některé členy můžeme vyjádřit pomocí okrajových podmínek vychází.

Pro $i = 1$:

$$-\frac{\lambda}{2} w_{0,\nu+1} = -\frac{\lambda}{2} r_1(a, \tau_{\nu+1}), \quad \frac{\lambda}{2} w_{0\nu} = \frac{\lambda}{2} r_1(a, \tau_\nu),$$

pro $i = m - 1$:

$$-\frac{\lambda}{2} w_{m,\nu+1} = -\frac{\lambda}{2} r_2(b, \tau_{\nu+1}), \quad \frac{\lambda}{2} w_{m\nu} = \frac{\lambda}{2} r_2(b, \tau_\nu).$$

Po dosazení a úpravách dostaneme

$$Aw^{(\nu+1)} = Bw^{(\nu)} + d^{(\nu)}, \tag{4.15}$$

kde

$$d^{(\nu)} := \frac{\lambda}{2} \cdot \begin{pmatrix} r_1(a, \tau_{\nu+1}) + r_1(a, \tau_\nu) \\ 0 \\ \vdots \\ 0 \\ r_2(b, \tau_{\nu+1}) + r_2(b, \tau_\nu) \end{pmatrix} \tag{4.16}$$

Okrajové podmínky pro americké opce
Hodnota americké opce nemůže být nikdy menší než hodnota evropské opce, takže platí $V^{am} \geq V^{ev}$. Druhou podmínkou je, že americká opce má minimálně hodnotu hodnotu výplatní funkce, což pro evropské opce neplatí. Kdyby hodnota americké put byla $V_P^{am} < (K - S)^+ =: \max(K - S, 0)$, stačilo by tuto opci neustále realizovat a tvořit zisk, čímž by byla porušena podmínka neexistence arbitráže. Podle analogického předpokladu nemůže ani pro americkou call nastat případ $V_C^{am} < (S - K)^+$. Platí tedy nerovnosti

$$\begin{aligned} V_P^{am}(S, t) &\geq (K - S)^+ \quad \text{pro všechna } (S, t) \\ V_C^{am}(S, t) &\geq (S - K)^+ \quad \text{pro všechna } (S, t). \end{aligned} \tag{4.17}$$

Okrajové podmínky pro $V = 0$

$$V_C(S, t) = 0 \quad \text{pro } S = 0,$$

$$V_P(S, t) = 0 \quad \text{pro } S \to \infty$$

jsou stejné jako pro evropské opce, ostatní se liší. Pro americkou put platí, že pokud ji realizujeme hned po nákupu, dostaneme hodnotu K. Pro malá $S > 0$ podle (4.17) platí $V^{am} \equiv K - S$. Na druhé straně $V_P^{am} > (K - S)^+$, tedy ze spojitosti a monotónnosti

V_P vyplývá existence hodnoty $S_f(t)$, pro kterou platí $0 < S_f(t) < K$ a $V_P^{am}(S_f, t) = (K - S)^+$. V matematickém zápisu

$$\begin{aligned} V_P^{am}(S,t) &> (K-S)^+ \quad \text{pro } S > S_f(t), \\ V_P^{am}(S,t) &= K - S \quad \text{pro } S \leq S_f(t). \end{aligned} \tag{4.18}$$

Pro $S \leq S_f(t)$ a libovolné $t \in [0, T]$ se hodnota $V_P^{am}(S,t)$ rovná výplatní funkci. Hodnoty $S_f(t)$ však neznáme, proto se počítání $V^{am}(S,t)$ pro $S > S_f(t)$ nazývá problém volné okrajové podmínky. Z ekonomického hlediska znamená $S_f(t)$ pro americkou put, že v případě $V^{am} > (K-S)^+$ okamžitý prodej znamená ztrátu, protože realizační cena K nevyrovná ztrátu S a V. Naopak v případě $S \leq S_f$ je doporučeno opci prodat, protože investor může investovat do bezrizikového aktiva hodnotu K na zbývající období $(T-t)$ za bezrizikovou úrokovou míru r. Tedy investor by měl opci prodat, jakmile dosáhne hodnoty $S_f(t)$ v čase t_s, který nazveme okamžik realizace (stopping-time). Držení opce po dosažení t_s znamená snížení zisku z bezrizikového aktiva. K vypočítání $S_f(t)$ potřebujeme dodatečnou podmínku. Sklon $\frac{\partial V}{\partial S}$ pod kterým se $V^{am}(S,t)$ dotýká v bodě $S_f(t)$ přímky $K - S$ je roven -1. Případ $\frac{\partial V(S_f(t),t)}{\partial S} < -1$ nemůže nastat vzhledem podmínkám (4.17) a (4.18), které by v tomto případě byly porušeny. Z důvodu arbitráže nemůže nastat ani $\frac{\partial V(S_f(t),t)}{\partial S} > -1$. Pro bod dotyku $S_f(t)$ tak máme dvě podmínky

$$\begin{aligned} V_P^{am}(S_f(t),t) &= K - S_f(t) \\ \tfrac{\partial V_P^{am}(S_f(t),t)}{\partial S} &= -1. \end{aligned} \tag{4.19}$$

Pro americkou call formulujeme obdobné hraniční podmínky. Je potřeba aby $\delta \neq 0$, protože dřívější realizace se nevyplatí call opce na aktivum, které nevyplácí dividendy. To je vidět z $V_C^{am} \geq S - Ke^{-r(T-t)}$, což pro $t < T$ a $r > 0$ znamená $V_C^{am} > S - K$, takže pro $\delta = 0$ je $V_C^{am} = V_C^{eur}$. Pro call s $\delta \neq 0$ a $r > 0$ podmínky

$$\begin{aligned} V_C^{am}(S_f(t),t) &= S_f(t) - K \\ \tfrac{\partial V_C^{am}(S_f(t),t)}{\partial S} &= 1. \end{aligned} \tag{4.20}$$

musí platit pro $S_f(t) > K$.

Black-Scholesova rovnice (2.1) je odvozena za podmínek zakazující dřívější prodej. Americké opce splňují nerovnost Black-Scholesova typu

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + (r - \delta) S \frac{\partial V}{\partial S} - rV \leq 0 \tag{4.21}$$

(důkaz viz. [3]). Nerovnosti (4.17) a (4.21) platí pro každé (S,t). V případě ostré nerovnosti v (4.17) platí v (4.21) rovnost. Kontaktní bod S_f oddělí hodnoty V^{am} pro které je výhodnější opci prodat od těch, kdy je lepší opci držet

$$V_P^{am} = K - S \quad \text{pro } S \leq S_f \quad \text{prodat}$$
$$V_P^{am} \text{ řeší (2.1)} \quad \text{pro } S > S_f \quad \text{držet}$$

$$V_C^{am} = S - K \quad \text{pro } S \geq S_f \quad \text{prodat}$$
$$V_C^{am} \text{ řeší (2.1)} \quad \text{pro } S < S_f \quad \text{držet}$$

Pro americké put opce budeme řešit nerovnost Black-Scholesova typu i její transformovanou verzi. Potřebujeme metodu, která umožní spočítat V i přes neznámou hodnotu S_f.

4.5 Lineární komplementarita pro americké opce

Pokud $V_P^{am} > (K - S)^+$ platí pro americké put opce bez výplaty dividend ($\delta = 0$)

$$\frac{\partial V(S,t)}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V(S,t)}{\partial S^2} + (r - \delta) S \frac{\partial V(S,t)}{\partial S} - rV(S,t) = 0,$$

(viz. [8]). Pro transformovanou rovnici (2.4) platí

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

(viz. [5]). Také podmínka (4.17) je transformována na

$$V_P^{am}(S,t) \geq (K - S)^+ = K \max(1 - e^x, 0),$$

která vede k nerovnosti

$$\begin{aligned} y(x,\tau) &\geq \exp\{(q-1)x/2 + (q+1)^2\tau/4\} \ \max(1 - e^x, 0) \\ &= \exp\{(q+1)^2\tau/4\} \ \max(e^{\frac{x}{2}(q-1)}(1 - e^x), 0) \\ &= \exp\{(q+1)^2\tau/4\} \ \max(e^{\frac{x}{2}(q-1)} - e^{\frac{x}{2}(q+1)}, 0) \\ &=: g(x,\tau) \end{aligned} \tag{4.22}$$

Funkce g je transformovaná výplatní funkce a umožňuje zapsat počáteční podmínku (4.10) jako $y(x,0) = g(x,0)$. Víme také, že $y \in \mathcal{C}^1$ a $y(x,\tau) = g(x,\tau)$ pro $x \to \pm\infty$. V praxi tyto hraniční podmínky platí pro $x_{\min}$ a $x_{\max}$. Pro netransformovanou verzi Black-Scholesovi rovnice pak máme okrajové podmínky $V(S_0,t) = (K-S)^+$, $V(S_1,t) = (K-S)^+$ a $V(S,T) = (K-S)^+$.

Souhrn lineárních podmínek komplementarity pro americké opce

$$\begin{aligned}
&\left(\frac{\partial V}{\partial t}+\frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}+(r-\delta)S\frac{\partial V}{\partial S}-rV\right)(V-(K-S)^+)=0\\
&\frac{\partial V}{\partial t}+\frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}+(r-\delta)S\frac{\partial V}{\partial S}-rV\le 0\\
&V(S,t)\ge (K-S)^+\\
&V(S_0,t)=(K-S_0)^+,\quad V(S_1,t)=(K-S_1)^+\\
&V(S,T)=(K-S)^+
\end{aligned}\tag{4.23}$$

$$\begin{aligned}
&g(x,\tau):=\exp\{((q_\delta-1)^2+4q)\tau/4\}\ \max\{e^{\frac{x}{2}(q_\delta-1)}-e^{\frac{x}{2}(q_\delta+1)},0\}\\
&\left(\frac{\partial y}{\partial \tau}-\frac{\partial^2 y}{\partial x^2}\right)(y-g)=0,\\
&\frac{\partial y}{\partial \tau}-\frac{\partial^2 y}{\partial x^2}\ge 0,\quad y-g\ge 0,\\
&\lim_{x\to+\infty} y(x,\tau)=\lim_{x\to+\infty} g(x,\tau),\\
&y(x,0)=g(x,0).
\end{aligned}\tag{4.24}$$

Diskretizace pro americké opce

Diskretizaci provedeme stejně jako pro evropské opce. Jediným rozdílem je změna diferenciální rovnice (2.4) na diferenciální nerovnost $\frac{\partial y}{\partial \tau}\ge\frac{\partial^2 y}{\partial x^2}$ a její diskrétní verzi

$$\begin{aligned}
\frac{w_{i,\nu+1}-w_{i\nu}}{\Delta\tau}\ge\ &\theta\frac{w_{i+1,\nu+1}-2w_{i,\nu+1}+w_{i-1,\nu+1}}{\Delta x^2}+\\
&(1-\theta)\frac{w_{i+1,\nu}-2w_{i\nu}+w_{i-1,\nu}}{\Delta x^2},
\end{aligned}$$

kde volba $\theta=\frac{1}{2}$ znamená Crank-Nicolsonovu, $\theta=0$ explicitní a $\theta=1$ implicitní metodu. Znovu použijeme označení $\lambda:=\frac{\Delta\tau}{\Delta x^2}$ a po drobné úpravě dostaneme verzi

$$\begin{aligned}
w_{i,\nu+1}\quad &-\lambda\theta(w_{i+1,\nu+1}-w_{i,\nu+1}+w_{i-1,\nu+1})\\
&-w_{i\nu}-\lambda(1-\theta)(w_{i+1,\nu}-2w_{i\nu}+w_{i-1,\nu})\ge 0.
\end{aligned}\tag{4.25}$$

Použitím označení

$$
\begin{aligned}
b_{i\nu} &:= w_{i\nu} + \lambda(1-\theta)(w_{i+1,\nu} - 2w_{i\nu} + w_{i-1,\nu}),\ i = 2, \ldots, m-2 \\
b^{(\nu)} &:= (b_{1\nu}, \ldots, b_{m-1,\nu})^{|} \\
w^{(\nu)} &:= (w_{1\nu}, \ldots, w_{m-1,\nu})^{\top} \\
g^{(\nu)} &:= (g_{1\nu}, \ldots, g_{m-1,\nu})^{|}
\end{aligned}
$$

$$
A = \begin{pmatrix}
1+2\lambda\theta & -\lambda\theta & & & 0 \\
-\lambda\theta & \ddots & \ddots & & \\
 & \ddots & \ddots & \ddots & \\
 & & \ddots & \ddots & \ddots \\
0 & & & \ddots & \ddots
\end{pmatrix} \tag{4.26}
$$

přepíšeme (4.25) do vektorového zápisu

$$
Aw^{(\nu+1)} \geq b^{(\nu)} \quad \text{pro všechna } \nu.
$$

Nerovnost pro vektory chápeme jako nerovnost po složkách. Nerovnost $y - g \geq 0$ dává $w^{(\nu)} \geq g^{(\nu)}$ a $(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2})(y - g) = 0$ zapíšeme jako

$$
(Aw^{(\nu+1)} \geq b^{(\nu)})^{\top}\ (w^{(\nu+1)} - g^{(\nu+1)}) = 0.
$$

Počáteční hraniční podmínky jsou

$$
\begin{aligned}
&w_{i0} = g_{i0}, \quad i = 1, \ldots, m-1, \quad \text{tedy } w^{(0)} = g^{(0)}, \\
&w_{0\nu} = g_{0\nu}, \quad w_{0\nu} = g_{0\nu}, \quad \nu \geq 1.
\end{aligned}
$$

Hraniční podmínky ve vektorech $b_{1\nu}$ a $b_{m-1,\nu}$ jsou obsaženy následovně

$$
\begin{aligned}
b_{1\nu} &= w_{1\nu} + \lambda(1-\theta)(w_{2\nu} - 2w_{1\nu} + g_{0\nu}) + \lambda\theta g_{0,\nu+1} \\
b_{m-1,\nu} &= w_{m-1,\nu} + \lambda(1-\theta)(g_{m\nu} - 2w_{m-1,\nu} + w_{m-2,\nu}) + \lambda\theta g_{m,\nu+1}.
\end{aligned} \tag{4.27}
$$

Diskrétní verze (4.24) jako algoritmus pro americké opce:

Pro $\nu = 1, \ldots, \nu_{max}$:
spočítáme vektor $g := g^{(\nu)}$,
$\quad b := b^{(\nu-1)}$ z (4.26) a (4.27).
Spočítáme vektor w jako řešení úlohy

$$
Aw - b \geq 0,\ w \geq g,\ (Aw - b)^{|}(w - g) = 0. \tag{4.28}
$$

$w^{(\nu)} := w$

Iterační řešení
Pro každý čas ν v (4.28) platí lineární komplementarita. Řešení se provádí iteračně. Použijeme upravený SOR algoritmus podle Cryera (viz. [5]). Standardní SOR algoritmus řeší systém lineárních rovnic $Ax = b$. Označíme $x := w - g$ a systém rovnic $Ax = b - Ag =: c$ vyjádříme po složkách:

$$r_i^{(k)} = c_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i}^{m-1} a_{ij} x_j^{(k-1)} \tag{4.29}$$

$$x_i^{(k)} = \max \left\{ 0, x_i^{(k-1)} + \omega_R \frac{r_i^{(k)}}{a_{ii}} \right\} \tag{4.30}$$

SOR algoritmus pro americké opce řeší $Aw = b$ za podmínky $w \geq g$ po složkách.

4.6 Metoda maximální okrajové podmínky

Podle [8] využijeme při řešení problému volné okrajové podmínky pro put opce vlastností funkce $V(S,t)$, která má v bodech $S_f(t)$ parciální derivaci -1. Při numerickém řešení však tato podmínka nikdy nebude splněna přesně. Proto použijeme postup hledání minimálního S, pro které bude platit $V(S,t) = (K-S)^+$. V čase T se takové $S = K$.

Tentokrát budeme vycházet z původní Black-Scholesovi rovnice (2.1). Analogicky lineárním podmínkám komplementarity (4.23) zapíšeme problému volné okrajové podmínky

$$\begin{aligned} \max \quad & \left(\frac{\partial V(S,t)}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V(S,t)}{\partial S^2} + (r-\delta) S \frac{\partial V(S,t)}{\partial S} - rV(S,t), \right. \\ & \left. (K-S)^+ - V(S,t) \right) = 0. \end{aligned} \tag{4.31}$$

Užitím diskrétní sítě souřadnic (4.2), metody zpětných diferencí

$$\frac{\partial V_{jn}}{\partial t} = \frac{u_{j,n+1} - u_{jn}}{\Delta t} + \mathcal{O}(\Delta t)$$

$$\frac{\partial^2 V_{jn}}{\partial S^2} = \frac{u_{j+1,n} - 2\ u_{jn} + u_{j-1,n}}{\Delta S^2} + \mathcal{O}(\Delta S^2)$$

$$\frac{\partial V_{jn}}{\partial S} = \frac{u_{jn} - u_{j,n-1}}{\Delta S} + \mathcal{O}(\Delta S)$$

a označení $a2 := \frac{\sigma^2}{2}$, $a1 := r - \delta$, $a0 := -r$ dostaneme

$$\max \left(a2\ S_j^2\ \frac{u_{j+1,n} - 2\ u_{jn} + u_{j-1,n}}{\Delta S^2} + a1\ S_j\ \frac{u_{jn} - u_{j-1,n}}{\Delta S} + \left(a0 - \frac{1}{\Delta t}\right)\ u_{jn} + \frac{u_{j,n+1}}{\Delta t}, (K - S_j)^+ - u_{jn} \right) = 0 \tag{4.32}$$

za předpokladu $r \geq \delta$. Obdobně jako výpočtů transformovaná rovnice (2.4) můžeme tento systém lineárních rovnic převést do maticového zápisu

$$Bu^{(n)} = u^{(n+1)}.$$

Změnu, kterou přináší výběr maximálního prvku z řešení Black-Scholesovi rovnice a rozdílu výplatní funkce a hodnoty opce, vyjádříme vzhledem k libovolnému bodu $S_k \leq K$, který je v síti souřadnic (4.2) na k-té pozici. Vzhledem ke známému bodu S_k se nejedná o problém volné okrajové podmínky. Při řešení využijeme okrajové podmínky (4.23). Matice B je blokově diagonální a skládá se ze dvou částí. První blok je matice identity délky k druhá část pak vypadá následovně:

$$B2 = \begin{pmatrix} S_k^2 \frac{a2}{\Delta S^2} & S_k \frac{a1}{\Delta S} + a0 & S_k^2 \frac{a2}{\Delta S^2} & & 0 \\ -S_k \frac{a1}{\Delta S} & -\frac{1}{\Delta t} - S_k^2 \frac{a2}{\Delta S^2} & & & \\ & \ddots & \ddots & \ddots & \\ & & S_{m-1}^2 \frac{a2}{\Delta S^2} & S_{m-1} \frac{a1}{\Delta S} + a0 & S_{m-1}^2 \frac{a2}{\Delta S^2} \quad 0 \\ & & -S_{m-1} \frac{a1}{\Delta S} & -\frac{1}{\Delta t} - S_{m-1}^2 \frac{a2}{\Delta S^2} & \\ 0 & & & 0 & 1 \end{pmatrix}$$

Matice B je třídiagonální s rozměry $(m+1) \times (m+1)$.

$$B = \begin{pmatrix} I_k & 0 \\ 0 & B2 \end{pmatrix}. \tag{4.33}$$

Vektor $u^{(n+1)}$ na pravé straně rovnice je upraven ve stejném smyslu jako matice B.

$$u^{(n+1)} := \left((K - S_0)^+, \ldots, (K - S_{k-1})^+, -\frac{1}{\Delta} u_{k,n+1}, \ldots, -\frac{1}{\Delta} u_{m-1,n+1}, (K - S_m)^+ \right)^{\mathsf{T}}$$

Tato úprava, kde až do $k-1$-ního místa vektoru u jsou hodnoty výplatní funkce a v matici B jedničky na diagonále znamená, že se hodnota opce až do hodnoty S_k rovná výplatní funkci a pro hodnoty S_{k+1} a větší je řešením (2.1).

Pro další výpočet najdeme takové z, že $S_z = K$. Budeme hledat minimální $k \leq z$, pro které platí

$$a2\ S_k^2\ \frac{u_{k+1,n} - 2\ u_{kn} + u_{k-1,n}}{\Delta S^2} + a1\ S_j\ \frac{u_{kn} - u_{k-1,n}}{\Delta S} + \left(a0 - \frac{1}{\Delta t}\right)\ u_{kn} = -\frac{u_{k,n+1}}{\Delta t}$$

a zároveň

$$(K - S_k)^+ = u_{kn}.$$

Podle [8] je takové řešení jednoznačné. Přesné analytické řešení máme pro věčnou americkou opci, na které je možné ověřit výše zmíněný postup. Druhou vlastností funkce $S_f(t)$, kterou využijeme je monotónnost. To znamená, že hledání minimálního k nemusíme vždy začínat od z, ale stačí hledat od bodu dotyku v předcházející již vypočtené časové vrstvě. Výpočet časových vrstev probíhá zpětně od doby splatnosti T.

Kapitola 5

Metoda konečných prvků

5.1 Vážená rezidua

Myšlenku vážených reziduí (viz. [5]) si ukážeme na příkladu diferenciální rovnice

$$Lu(x) = f(x),$$

kde L je lineární diferenciální operátor. Důležité příklady jsou

$$Lu(x) \ := -u''(x) \quad \text{nebo}$$

$$Lu(x,y) \ := -\frac{\partial^2 u(x,y)}{\partial x^2} - \frac{\partial^2 u(x,y)}{\partial y^2}.$$

Řešení diferenciální rovnice u hledáme na oboru hodnot $\mathcal{D} \subseteq \mathbb{R}^n$. Rozdělíme obor hodnot na konečný počet podmnožin

$$\mathcal{D} = \bigcup_k \mathcal{D}_k. \tag{5.1}$$

Aproximace $u(x)$ počítáme na uzavřené množině $\bar{\mathcal{D}}$. Všechny hranice by měly být zahrnuty v $\mathcal{D}_k$ tak, aby $\mathcal{D}_k \cap \mathcal{D}_j = \emptyset$ pro $j \neq k$. Při aproximaci $u(x)$ pomocí $w(x)$ vyjádříme

$$w(x) := \sum_{i=1}^{N} c_i \varphi_i(x). \tag{5.2}$$

Pokud je x nezávislá proměnná, pak $c_i \in \mathbf{R}$ jsou konstantní koeficienty a φ_i funkce závislé na x, které nazveme **základní funkce**. Obvykle jsou $\varphi_1(x), \ldots, \varphi_N(x)$ dány předem a parametry $c_1, \ldots, c_N$ hledáme tak, aby $w(x) \approx u(x)$.

Pro výpočet c_i využijeme rezidua

$$R(x) := Lw(x) - f(x).$$

Hledáme takové $w(x)$, aby $R(x)$ bylo co nejmenší. Protože $\varphi_i(x)$ považujeme za dané, pak vzhledem k (5.2) potřebujeme N podmínek nebo rovnic pro výpočet $c_1, \ldots, c_N$. Proto zavedeme N váhových funkcí $\psi_1(x), \ldots, \psi_N(x)$ a požadujeme

$$\int_{\mathcal{D}} R(x)\psi_j(x)dx = 0 \quad \text{pro } j = 1, \ldots, N, \tag{5.3}$$

což znamená, že rezidua musí být kolmá na váhové funkce. Člen dx vyjadřuje integrování přes $\mathcal{D} \subseteq \mathbb{R}^n$. Soustava N rovnic

$$\int_{\mathcal{D}} Lw(x)\psi_j(x)dx = \int_{\mathcal{D}} f(x)\psi_j(x)dx \quad \text{pro } j = 1, \ldots, N, \tag{5.4}$$

obsahuje neznámé c_i jako součást $w(x)$. Pro lineární L znamená předpoklad (5.2), že

$$\underbrace{\int f(x)\psi_j(x)dx}_{=:r_j} = \int Lw(x)\psi_j(x)dx =$$

$$= \int \Big(\sum_i c_i L\varphi_i(x)\Big)\psi_j(x)dx = \sum_i c_i \underbrace{\int L\varphi_i(x)\psi_j(x)dx}_{=:a_{ij}}.$$

Integrály a_{ij} jsou prvky matice A a integrály r_j prvky vektoru r. Potom (5.4) můžeme zapsat vektorově jako

$$Ac = r, \tag{5.5}$$

kde c je vektor koeficientů $c = (c_1, \ldots, c_N)^{\mathsf{T}}$.

Příklady váhových funkcí

1) Galerkinova metoda: $\psi_j(x) := \varphi_j(x)$,
pak $a_{ij} = \int L\varphi_i(x)\varphi_j(x)dx$

2) Uspořádání: $\psi_j(x) := \delta(x - x_j)$,
kde δ znamená Diracovo delta, které splňuje $\int f(x)\delta(x - x_j)dx = f(x_j)$. Tedy platí $\int Lw(x)\psi_j(x)dx = Lw(x_j)$ a $\int f(x)\psi_j(x)dx = f(x_j)$. Systém rovnic $Lw(x_j) = f(x_j)$ umožní spočítat diferenciální rovnici ve zvolených bodech x_j.

3) Charakteristické funkce:

$$\psi_k(x) = \begin{cases} 1 & \text{v } \mathcal{D}_k \\ 0 & \text{pro } x \notin \mathcal{D}_k \end{cases}$$

Příklady základních funkcí
Pro dosažení co nejvyšší efektivnosti můžeme také zkoumat řídkost matic. Požadujeme, aby $\varphi_i(x) \equiv 0$ pro skoro všechny $\mathcal{D}_k$. Takzvaná trojúhelníková funkce je nejjednodušší příklad. Je po částech lineární a každá funkce $\varphi_i(x) \neq 0$ pouze na dvou intervalech. Jako důsledek pak platí

$$\int_{\mathcal{D}} \varphi_i(x)\varphi_j(x)dx = 0 \quad \text{pro } |i-j| > 1, \tag{5.6}$$

stejně jako podobný vztah $\int \varphi_i'(x)\varphi_j'(x)dx$. Pokročilejší základní funkce konstruujeme s využitím polynomů vyšších stupňů.

Pro $Lu(x) = u''(x)$ a $u(x), \varphi(x), \psi(x) \in (u(x) : u(0) = u(1) = 0)$ z integrace per partes vyplývá

$$\int_0^1 \varphi''(x)\psi(x)dx = -\int_0^1 \varphi'(x)\psi'(x)dx = \int_0^1 \varphi(x)\psi''(x)dx,$$

protože neintegrální části se v důsledku $u(0) = u(1) = 0$ rovnají 0. Tyto tři verze integrálu můžeme využít podle požadavků na hladkost $\varphi(x)$ a $\psi(x)$.

5.2 Galerkinův přístup

U metody konečných diferencí vede jednoduchý systém uzlů se stejně vzdálenými uzly x_i vede k exponenciální rychlosti růstu hodnot S_i. Vhodnější je pracovat se stejně vzdálenými S_i pomocí metody konečných prvků.

Trojúhelníková (hat) funkce
Definice: pro $1 \leq i \leq m-1$

$$\varphi_i(x) := \begin{cases} \frac{x-x_{i-1}}{x_i-x_{i-1}} & \text{pro } x_{i-1} \leq x < x_i \\ \frac{x_{i+1}-x}{x_{i+1}-x_i} & \text{pro } x_i \leq x < x_{i+1} \\ 0 & \text{jinde} \end{cases}$$

hraniční podmínky:

$$\varphi_0(x) := \begin{cases} \frac{x_1-x}{x_1-x_0} & \text{pro } x_0 \leq x < x_1 \\ 0 & \text{jinde} \end{cases}$$

$$\varphi_m(x) := \begin{cases} \frac{x-x_{m-1}}{x_m-x_{m-1}} & \text{pro } x_{m-1} \leq x < x_m \\ 0 & \text{jinde} \end{cases}$$

Jednoduchá aplikace
Uvažujme model s jednoduchými hraničními podmínkami

$$Lu(x) := u''(x) = f(x) \quad \text{kde } u(x_0) = u(x_m) = 0. \tag{5.7}$$

Použijeme Galerkinův přístup a dosadíme za $w(x) = \sum_{i=0}^{m} c_i\varphi_i(x)$ do rovnice (5.4) a dostaneme

$$\sum_{i=0}^{m} c_i \int_{x_0}^{x_m} L\varphi_i(x)\varphi_j(x)dx = \int_{x_0}^{x_m} f(x)\varphi_j(x)dx.$$

Aplikací integrace po částech na levé straně a dosazením polynomu $f = \sum_{i=0}^{m} f_i\varphi_i(x)$, kde $f_i := f(x_i)$ na pravé straně dostáváme systém rovnic

$$\sum_{i=0}^{m} c_i \underbrace{\int_{x_0}^{x_m} \varphi_i'(x)\varphi_j'(x)dx}_{=:a_{ij}} = \sum_{i=0}^{m} f_i \underbrace{\int_{x_0}^{x_m} \varphi_i(x)\varphi_j(x)dx}_{=:b_{ij}} \qquad j = 0, \ldots, m \tag{5.8}$$

Prozatím jsme neuvažovali hraniční podmínky. Nyní využijeme možnosti rozdělit integrál

$$\int_{x_0}^{x_m} dx = \sum_{k=0}^{m-1} \int_{\mathcal{D}k} dx$$

ke konstrukci čtvercových matic A a B o rozměrech $(m+1) \times (m+1)$. Na intervalu

$$\mathcal{D}_k = x : x_k \leq x \leq x_{k+1}$$

jsou integrály $\varphi_i'\varphi_j'$ a $\varphi_i\varphi_j$ nenulové pro

$$i, j \in \mathcal{I}_k := k, k+1. \tag{5.9}$$

$\mathcal{I}_k$ je množina indexů základních funkcí, které jdou nenulové na $\mathcal{D}_k$. Proto po integrování přes jeden interval $\mathcal{D}_k$ dostaneme matici (2×2)

$$\frac{1}{h_k}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix},$$

kde $h_i := x_i - x_{i-1}$. Jejím postupným sčítáním po diagonále dostaneme třídiagonální matici

$$A = \begin{pmatrix} \frac{1}{h_1} & \frac{-1}{h_1} & & & & & 0 \\ \frac{-1}{h_1} & \frac{1}{h_1}+\frac{1}{h_2} & \frac{-1}{h_2} & & & & \\ & \frac{-1}{h_2} & \frac{1}{h_2}+\frac{1}{h_3} & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \frac{1}{h_{m-1}}+\frac{1}{h_m} & \frac{-1}{h_m} & \\ 0 & & & & \frac{-1}{h_m} & \frac{1}{h_m} & \end{pmatrix}$$

Obdobným způsobem dostaneme matici

$$B = \begin{pmatrix} \frac{h_1}{3} & \frac{h_1}{6} & & & & 0 \\ \frac{h_1}{6} & \frac{h_1+h_2}{3} & \frac{h_2}{6} & & & \\ & \frac{h_2}{6} & \frac{h_2+h_3}{3} & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & \frac{h_{m-1}+h_m}{3} & \frac{h_m}{6} \\ 0 & & & & \frac{h_m}{6} & \frac{h_m}{3} \end{pmatrix}$$

z matice

$$\frac{h_k}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Nyní můžeme systém rovnic (5.8) přepsat do maticového zápisu

$$Ac = B\bar{f}, \tag{5.10}$$

kde f je vektor funkcí f_i. Matice A je singulární, protože $A \cdot (1, \ldots, 1)^\top = 0$, proto nemá soustava (5.10) jednoznačné řešení. Jednoznačné řešení dostaneme splněním okrajových podmínek (viz. [5]). V případě $u(0) = u(m) = 0$ dostaneme koeficienty $c_0 = c_m = 0$.

5.3 Aplikace na standardní opce

Analogicky jako u metody konečných diferencí bude výpočet pro americké opce problém volných okrajových podmínek. Funkce $g(x, \tau)$ je stejná jako v případě konečných diferencí.

$$\text{put} : g(x,\tau) := \exp\left\{((q_\delta - 1)^2 + 4q)\tau/4\right\} \ \max\left\{e^{\frac{x}{2}(q_\delta - 1)} - e^{\frac{x}{2}(q_\delta + 1)}, 0\right\}$$

$$\text{call} : 0 < \delta < r, \ g(x,\tau) := \exp\left\{((q_\delta - 1)^2 + 4q)\tau/4\right\} \ \max\left\{e^{\frac{x}{2}(q_\delta + 1)} - e^{\frac{x}{2}(q_\delta - 1)}, 0\right\}$$

Třída srovnávacích funkcí je definována jako

$$\begin{aligned}\mathcal{K} = \quad & \{v \in \mathcal{C}^0, \\ & v(x,\tau) \geq g(x,\tau) \quad \text{pro každé } x,\tau,\ v(x,0) = g(x,0), \\ & v(x_{\max},\tau) = g(x_{\max},\tau), v(x_{\min},\tau) = g(x_{\min},\tau)\}. \end{aligned} \tag{5.11}$$

Nechť $y(x,\tau)$ je přesné řešení lineární komplementarity pro americké opce. Jako řešení parciální diferenciální rovnice je $y(x,\tau)$ hladká funkce a tedy $y(x,\tau) \in \mathcal{K}$. Z nerovností

$$v \geq g, \quad \frac{\partial y(x,\tau)}{\partial \tau} - \frac{\partial^2 y(x,\tau)}{\partial x^2} \geq 0$$

vychází

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y(x,\tau)}{\partial \tau} - \frac{\partial^2 y(x,\tau)}{\partial x^2} \right) (v(x,\tau) - g(x,\tau)) dx \geq 0.$$

S ohledem na lineární komplementaritu pro americké opce (4.24) platí

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y(x,\tau)}{\partial \tau} - \frac{\partial^2 y(x,\tau)}{\partial x^2} \right) (y(x,\tau) - g(x,\tau)) dx = 0$$

a odečtením dostaneme

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y(x,\tau)}{\partial \tau} - \frac{\partial^2 y(x,\tau)}{\partial x^2} \right) (v(x,\tau) - y(x,\tau)) dx \geq 0.$$

Integrováním per partes vyjde nerovnost

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y(x,\tau)}{\partial \tau} (v(x,\tau) - y(x,\tau)) - \frac{\partial y(x,\tau)}{\partial x} \Big(\frac{\partial v(x,\tau)}{\partial x} - \frac{\partial y(x,\tau)}{\partial x} \Big) \right) dx$$

$$- \Big[\frac{\partial y(x,\tau)}{\partial x} (v(x,\tau) - y(x,\tau)) \Big]_{x_{\min}}^{x_{\max}} \geq 0.$$

Poslední člen se rovná nule, protože pro $x_{\min}$ i $x_{\max}$ platí $v = g$, $g = y$ a tedy $v = y$. Výsledek pak je

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y(x,\tau)}{\partial \tau} (v(x,\tau) - y(x,\tau)) - \frac{\partial y(x,\tau)}{\partial x} \Big(\frac{\partial v(x,\tau)}{\partial x} - \frac{\partial y(x,\tau)}{\partial x} \Big) \right) dx \geq 0, \tag{5.12}$$

pro každé $v(x,\tau) \in \mathcal{K}$. Pro přesné řešení $y(x,\tau) \in \mathcal{C}^2$ platí, že pro každé $v(x,\tau) \in \mathcal{K}$ platí nerovnost (5.12) Takové řešení se nazývá *silné* nebo *klasické řešení*. Pro $v(x,\tau) = y(x,\tau)$ bude integrál roven nule. Nyní budeme hledat takové $\hat{y}(x,\tau) \in \mathcal{K}$ aby integrál z (5.12)

byl minimální pro všechna $v(x,\tau) \in \mathcal{K}$. Řešení $\hat{y}(x,\tau)$, které je spojité, ale pouze po částech spojitě diferencovatelné, se nazývá *slabé řešení*. Použijeme aproximace

$$\sum_i w_i(\tau)\varphi_i(x) \quad \text{pro } \hat{y}(x,\tau),$$

$$\sum_i v_i(\tau)\varphi_i(x) \quad \text{pro } v(x,\tau)$$

a dosadíme do (5.12). Sumy můžeme rozdělit vzhledem k nezávislosti proměnných x a τ.

$$\int \Big\{ \Big(\sum_i \frac{dw_i}{d\tau}\varphi_i \Big) \Big(\sum_j (v_j - w_j)\varphi_j \Big) + \Big(\sum_i w_i \varphi_i' \Big) \Big(\sum_j (v_j - w_j)\varphi_j' \Big) \Big\} dx$$

$$= \sum_i \sum_j \frac{dw_i}{d\tau}(v_j - w_j) \int \varphi_i \varphi_j dx + \sum_i \sum_j w_i (v_j - w_j) \int \varphi_i' \varphi_j' dx \geq 0.$$

Ekvivalentní vektorové vyjádření vypadá

$$(v - w)^\top \Big(B\frac{dw}{d\tau} + Aw \Big) \geq 0,$$

kde matice A, B jsme již definovali. V dalším kroku dosadíme vektory

$$w^{(\nu)} := w(\tau_\nu), \quad v^{(\nu)} := v(\tau_\nu).$$

Pro každé ν platí nerovnost

$$\Big(v^{(\nu+1)} - w^{(\nu+1)} \Big)^\top \Big(B\frac{1}{\Delta\tau}\Big(w^{(\nu+1)} - w^{(\nu)} \Big) + \theta A w^{(\nu+1)} + (1-\theta) A w^{(\nu)} \Big) \geq 0, \tag{5.13}$$

která pro $\theta = 1/2$ odpovídá Crank-Nicolsonově metodě. Vzhledem k $y(x,\tau) \geq g(x,\tau)$ platí

$$\sum w_i(\tau)\varphi_i(x) \geq g(x,\tau),$$

což společně s vlastnostmi trojúhelníkové funkce φ_i ($\varphi_i(x_i) = 1$ a $\varphi_i(x_j) \neq 1$ pro $j \neq i$) a $x = x_j$ znamená, že $w_j \geq g(x_j,\tau)$. Ve výše uvedeném značení

$$w^{(\nu)} \geq g^{(\nu)}, \quad \text{analogicky } v^{(\nu)} \geq g^{(\nu)}.$$

Zavedeme označení

$$r := (B - \Delta\tau(1-\theta)A)w^{(\nu)}$$

$$C := B + \Delta\tau\theta A$$

a dosazení do (5.13) vede k nerovnosti

$$\left(v^{(\nu+1)} - w^{(\nu+1)}\right)^{\top}\left(Cw^{(\nu+1)} - r\right) \geq 0.$$

Pro každé ν musíme najít řešení, které zároveň splňuje podmínky $w^{(\nu+1)} \geq g^{(\nu+1)}$ pro každé $v^{(\nu+1)} \geq g^{(\nu+1)}$. Algoritmus pro výpočet amerických opcí

$$\begin{aligned}
&\text{Pro } \nu = 1, \ldots, Nt: \\
&\text{spočítáme } g := g^{(\nu)}, \\
&\quad r = (B - \Delta\tau(1-\theta)A)w^{(\nu-1)}. \\
&\text{Vypočítáme } w \text{ tak, aby pro každé } v \geq g \\
&\quad (v-w)^{\top}(Cw - r) \geq 0,\ w \geq g \\
&\text{a položíme } w^{(\nu)} := w.
\end{aligned} \tag{5.14}$$

Hlavní krok algoritmu můžeme přepsat do tvaru shodného s konečnými diferencemi

$$\begin{aligned}
&Cw - r \geq 0,\ w \geq g \\
&(Cw - r)\,|\,(w - g) = 0,
\end{aligned} \tag{5.15}$$

protože platí:

$$(v-w)^{\top}(Cw-r) = (v-g)^{\top}\underbrace{(Cw-r)}_{\geq 0} - \underbrace{(w-g)^{\top}(Cw-r)}_{=0},$$

proto $(v-w)\,|\,(Cw-r) \geq 0$ pro všechny $v \geq g$, tedy (5.15) $\Rightarrow$ (5.14). Na druhou stranu pokud by k-tý člen $Cw - r$ v nerovnosti

$$v^{\top}(Cw-r) \geq w^{\top}(Cw-r) \quad \text{pro každé } v \in \mathcal{K}$$

byl menší než nula, pro libovolně velké v_k by se levá strana stala libovolně malou, což je spor, tedy $Cw - r \geq 0$. Pro $w \geq g \Longrightarrow (w-g)\,|\,(Cw-r) \geq 0$. Dosazením $v = g$ do (5.14) vyjde $(w-g)^{\top}(Cw-r) \leq 0$, z čehož vychází $(w-g)^{\top}(Cw-r) = 0$ a tedy (5.14) $\Rightarrow$ (5.15).

Algoritmus (5.15) můžeme po dosazení $A = C$ a $b = r$ počítat pomocí SOR algoritmu použitého pro výpočet konečných diferencí, čemuž odpovídá jádro algoritmu

Najít řešení $Cw = r$ tak,
že $w_j^{(\nu)} \geq g(x_j, \tau_\nu)$ pro $j = 0, \ldots, m$.

Diskrétní verze diferenciální nerovnosti $\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0$ vypadá:

$$\begin{aligned}&\Big(\tfrac{h_i}{6} - \tfrac{\beta}{h_i}\Big)w_{i-1,\nu+1} + \Big(\tfrac{h_i+h_{i+1}}{3} + \beta\Big(\tfrac{1}{h_i} + \tfrac{1}{h_{i+1}}\Big)\Big)w_{i,\nu+1} + \Big(\tfrac{h_{i+1}}{6} - \tfrac{\beta}{h_{i+1}}\Big)w_{i+1,\nu+1}\\ &-\Big(\tfrac{h_i}{6} + \tfrac{\gamma}{h_i}\Big)w_{i-1,\nu} - \Big(\tfrac{h_i+h_{i+1}}{3} - \gamma\Big(\tfrac{1}{h_i} + \tfrac{1}{h_{i+1}}\Big)\Big)w_{i,\nu} - \Big(\tfrac{h_{i+1}}{6} + \tfrac{\gamma}{h_{i+1}}\Big)w_{i+1,\nu} \geq 0,\end{aligned}$$

kde $\beta = \Delta\lambda\theta$, $\gamma = \Delta\lambda(1-\theta)$. Hraniční podmínky zahrneme do r obdobně jako do b u konečných diferencí:

$$w_{0,\nu} = g(x_0, \tau_\nu), \quad w_{m,\nu} = g(x_m, \tau_\nu).$$

Kapitola 6

Výpočty v systému Mathematica

6.1 Funkce

Výpočetní systém Mathematica obsahuje řadu zabudovaných funkcí, které usnadňují a urychlují výpočty ve finanční matematice. Ve vlastních numerických algoritmech budeme využívat funkce LinearSolve, kterou využijeme při výpočtu Crank-Nicolsonova algoritmu i pro algoritmus maximální volné okrajové podmínky. Přímý výpočet diferenciální rovnice (2.4) provedeme funkcí *NDSolve* pro porovnání s Crank-Nicolsonovým algoritmem.

LinearSolve

Přestože je možné systém lineárních rovnic $Ax = b$, kde A je zadaná matice a b daný vektor nebo matice, řešit pomocí vyjádření $x = A^{-1}b$, funkce LinearSolve je navržena tak, aby x vypočítala co nejefektivněji vzhledem k zadané matici A. Kromě nejzákladnějších metod výpočtu obsahuje Gaußovu eliminaci pro výpočet numerických i symbolických matic, Choleského rozklad pro numerické matice a pro řídké matice využívá výpočty pomocí Multifrontal a Krylovovu metodu. Ve standardním nastavení je vybrána metoda podle zadané matice. V našem případě ji využijeme při výpočtu Crank-Nicolsonova algoritmu pro evropské opce, kde vždy počítáme s řídkými, přesněji třídiagonálními maticemi.

NDSolve

Je funkce, kterou můžeme využít přímo pro numerickou aproximaci diferenciálních rovnic (2.1) a (2.4). NDSolve řeší numericky obyčejné diferenciální rovnice, parciální diferenciální rovnice i jejich soustavy, které není možné řešit analyticky funkcí DSolve. Okrajové podmínky mohou být ve tvaru diferenciálních rovnic stejně tak jako čistě algebraických. Řešení NDSolve je ve formě interpolované funkce na zadaném prostoru. Počáteční podmínky je potřeba zadat tak, aby přesně určovaly řešení. V případě diferenciálních rovnic až do n-tého stupně je potřeba zadat podmínky až do $(n-1)$-ní derivace nebo okrajové podmínky v n bodech. NDSolve využívá pro interpolaci mnoho metod. Výše zmíněné metody dopředných a zpětných diferencí, jejich rozšíření na více

mezivýpočtů v jednom kroku podle Runge-Kutta, metodu implicitních diferenciálních algebraických rovnic (IDA) vyvinutou Center for Applied Scientific Computing of Lawrence Livermore National Laboratory pro řešení široké škály diferenciálních rovnic (viz. [2]) a mnoho dalších. Pokročilejší metody dále umožňují zvolit vlastní nastavení vnitřních algoritmů pro výpočet. NDSolve vedle samotných výpočetních algoritmů obsahuje funkce, které v automatickém nastavení podle velikosti chyby volí velikost kroku pro vyhodnocování a výpočetní metodu zaručující dosažení požadované přesnosti. Pro Runge-Kutta metodu se provádí výpočet stejného kroku metodami různého řádu, které mají stejné koeficienty. Z rozdílu výsledků se odhaduje chyba, podle které se volí velikost dalšího kroku. Maximální počet extrapolací, které NDSolve provádí na daném intervalu je omezen standardně na 10000, čímž je omezena minimální vzdálenost vyhodnocovaných hodnot.

Základní metoda pro extrapolaci diferenciální rovnice je Eulerova metoda. Uvažujme rovnici $y'(t) = f(t, y(t))$ s počáteční podmínkou $y(t_0) = y_0$ a daným $h > 0$, což je vzdálenost bodů t_{n+1}, t_n. Výpočet Eulerovou metodou

$$y_{n+1} = y_n + hf(t_n, y_n)$$

není příliš přesný. Přesnost je měřena srovnáním s Taylorovým rozvojem řešení. Eulerova metoda je řádu jedna, což znamená, že se mohou objevit chyby velikosti $\mathcal{O}(h^2)$. Myšlenka Runge-Kutta je vzít postupně vážené výsledky Eulerovy metody pro aproximaci Taylorovy řady. Pro metodu středního bodu

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + 0.5h, y_i 0.5hk_1) \\ y_{n+1} &= y_n + hk_2 \end{aligned} \tag{6.1}$$

lze ukázat, že chyba je řádu $\mathcal{O}(h^3)$, tedy se jedná o metodu řádu dva.
Rozšířením metody (6.1) na více mezivýpočtů, můžeme dosáhnout libovolného řádu. Tato metoda počítá

$$\begin{aligned} g_i &= y_n + h \sum_{j=1}^{s} a_{ij} k_j \\ k_i &= f(t_n + c_i h, g_i), \qquad i = 1, \ldots, s \\ y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i k_i, \end{aligned} \tag{6.2}$$

kde s je počet stupňů výpočtu a c_i, b_i, a_{ij} koeficienty vybrané tak, aby výsledky odpovídaly rozvoji Taylorovy řady do určitého řádu pro časový krok h. Obecně se uvažuje,

že platí podmínka řádkových součtů $c_i = \sum_{j=1}^{s} a_{ij}$. Pro explicitní Runge-Kutta metodu platí

$$a_{ij} = 0, \quad j \geq i, \quad j = 1, \ldots, s.$$

Koeficienty se obvykle zaznamenávají do Butcherovy tabulky, která pro explicitní Runge-Kutta metodu je

$$\begin{array}{c|ccccc} 0 & 0 & 0 & \cdots & 0 & 0 \\ c_2 & a_{2,1} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_s & a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \\ \hline & b_1 & b_2 & \ldots & b_{s-1} & b_s \end{array} \tag{6.3}$$

dolní trojúhelníková, důsledkem toho je $c_1 = 0$ a tedy $k_1 = f(t_n, y_n)$. Butcherova tabulka pro metodu středního bodu (6.1) vypadá:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 0.5 & 0.5 & 0 \\ \hline & 0 & 1 \end{array}. \tag{6.4}$$

Speciální třída Runge-Kutta metod využívaná v moderních výpočetních programech se nazývá FSAL (First Same As Last) podle struktury koeficientů, která se využívá. Butcherova tabulka má formu:

$$\begin{array}{c|ccccc} 0 & 0 & 0 & \cdots & 0 & 0 \\ c_2 & a_{2,1} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{s-1} & a_{s-1,1} & a_{s-1,2} & \cdots & 0 & 0 \\ 1 & b_1 & b_2 & \ldots & b_{s-1} & 0 \\ \hline & b_1 & b_2 & \ldots & b_{s-1} & 0 \end{array} \tag{6.5}$$

Podle [9] spočívá výhoda FSAL schématu v tom, že funkční hodnota k_s na konci jednoho integračního kroku je stejná jako první funkční hodnota k_1 v dalším integračním kroku. Obě tyto hodnoty jsou potřeba při konstrukci interpolované funkce, která se využívá při konstrukci výstupu funkce NDSolve.

Zpětná transformace

Použijeme v [5] doporučený rozsah $x_{\min}, x_{\max}$ a hustotu dělení $m = Nt = 100$. Transformace vypočtených hodnot na uzlech $x[i], \tau[\nu]$ na uzly $S[i], t[\nu]$ provedeme pomocí funkce

$$\begin{aligned} &V[i_, \nu_, W_] := \\ &\quad W[[\nu + 1, i]] \quad K\mathrm{Exp}\left[-\tfrac{x[i]}{2}\left(q_\delta - 1\right) - \tfrac{\tau[\nu]}{4}\left(\left(q_\delta - 1\right)^2 + 4q\right)\right]; \end{aligned} \tag{6.6}$$

kde W je matice vypočtených aproximovaných hodnot $y(x,\tau)$. Pro správné zobrazení hodnot $V(S,t)$ navíc budeme potřebovat hodnoty

$$\text{scale} = \text{Table}[K * N[\text{Exp}[x[i]]], \{i, m-1\}];$$

jednotlivých uzlů S_i, jejichž vzdálenost se v závislosti na použité metodě mohou měnit.

6.2 Evropské opce

Výpočet pro evropské opce provedeme iteračním postupem podle Crank-Nicolsonova algoritmu, tak využitím funkce NDSolve pro parciální diferenciální rovnice. Z [8] použijeme hodnoty parametrů pro výpočet

$$r = 0.04; \quad \sigma = 0.494515; \quad \delta = 0; \quad T = 0.052; \quad K = 76;$$

a tržní hodnoty opcí na index NASDAQ100 pro porovnání s teoretickými výsledky.

Crank-Nikolsonův algoritmus

1. Ze zadaných hodnot vypočítáme vzdálenosti jednotlivých uzlů
   $\Delta x = (x_{\max} - x_{\min})/\,m; \quad x[i_] := x_{\min} + i\Delta x;$
   $\Delta\tau = \frac{1}{2}\sigma^2 T\,/Nt\,; \quad \tau[\nu_] := \nu\Delta\tau;$
   $\lambda = \Delta\tau\,/\Delta x^2\,;$

2. Spočítáme parametry $q = \frac{2r}{\sigma^2},\ q_\delta = \frac{2(r-\delta)}{\sigma^2}$ a zavedeme vektor počátečních hodnot

   $$\text{ww} = \text{Table}\left[\text{Max}\left[\text{Exp}\left[\tfrac{x[i]}{2}\,(q_\delta - 1)\right] - \text{Exp}\left[\tfrac{x[i]}{2}\,(q_\delta + 1)\right], 0\right], \{i, m-1\}\right],$$

   který vložíme na první místo matice W.

3. Vypočítáme vektor okrajových podmínek
   $d = \text{Transpose}[$
   $\text{Join}\left[\left\{\text{Table}\left[\frac{\lambda}{2}\left(\text{Exp}\left[\frac{x_{\min}}{2}\,(q_\delta - 1) + \frac{\tau[\nu]}{4}\,(q_\delta - 1)^{\,2}\right]\right.\right.\right.$
   $\left.\left.\left. + \text{Exp}\left[\frac{x_{\min}}{2}\,(q_\delta - 1) + \frac{\tau[\nu-1]}{4}\,(q_\delta - 1)^{\,2}\right],\right.\right.$
   $\left.\left.\{\nu, Nt\}\right]\right\}, \text{Table}\,[0, \{i, m-2\}, \{j, Nt\}]]]];$

4. Vytvoříme řídké matice
   $A =$
   $\text{SparseArray}[\{\{i_, i_\} \to 1 + \lambda, \{i_, j_\}/; \text{Abs}[i-j] == 1 \to -\lambda/2\}, \{m-1, m-1\}];$
   $B =$
   $\text{SparseArray}[\{\{i_, i_\} \to 1 - \lambda, \{i_, j_\}/; \text{Abs}[i-j] == 1 \to \lambda/2\}, \{m-1, m-1\}];$

Zde se odchýlíme od popsané Crank-Nicolsonovi metody, když místo v algoritmu místo výpočtu LR-rozkladu použijeme funkci *LinearSolve*, která obsahuje speciální způsob výpočtu pro řídké matice.

$Do[ww = \text{LinearSolve}[A,\ B.W[[\nu]] + d[[\nu]]];$
$W = \text{Append}[W, ww];,$
$\{\nu, Nt\}]$

Výsledek algoritmu je matice W. Užitím výše uvedené zpětné transformace vypočítáme hodnoty $V(S,t)$

$OP = \text{Outer}\,[V[\#1, \#2, W]\&, \text{Range}[0, Nt], \text{Range}[1, m]]\,;$

a s použitím vzdálenosti jednotlivých uzlů vykreslíme plochu hodnot $V(S,t)$.

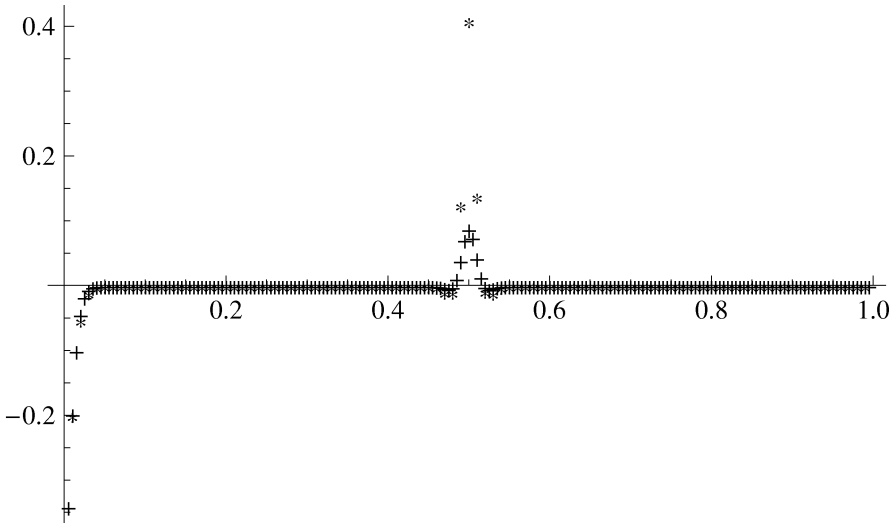
Srovnání

Funkci NDSolve využijeme pro výpočet parciální diferenciální rovnice (2.4) a výsledky srovnáme s výsledky Crank-Nicolsonova algoritmu. Nejprve porovnáme Crank-Nicolsnův algoritmus s NDSolve v čase $t = 0$. Jak ukazuje obrázek 6.1 k největšímu rozdílu dochází v x_1 a v polovině intervalu konkrétně v bodě $x_{m/2}$, což po zpětné transformaci znamená bod, kde se $S(t) = K$. Tento rozdíl vzniká v důsledku zlomu ve výplatní funkci v čase splatnosti a je možné jej snížit zvýšením počtem bodů sítě jak ukazuje obrázek 6.1, kde '*' označuje rozdíly pro $m = Nt = 100$ a '+' pro $m = Nt = 200$. Hodnoty pro různé hustoty dělení uvádí tabulka 6.1. Rozdíl v první desetině intervalu je způsobený zkrácením intervalu záporných hodnot x z $-\infty$ na a.

$m = Nt$	rozdíl v $m/2$	rozdíl v 1	odchylka
100	0.40922	−0.19974	12.7575
200	0.08726	−0.34102	7.6987
400	0.02112	−0.42546	6.6517
800	0.00525	−0.46873	6.4031

Tabulka 6.1: Rozdíl Crank-Nicolson a NDSolve

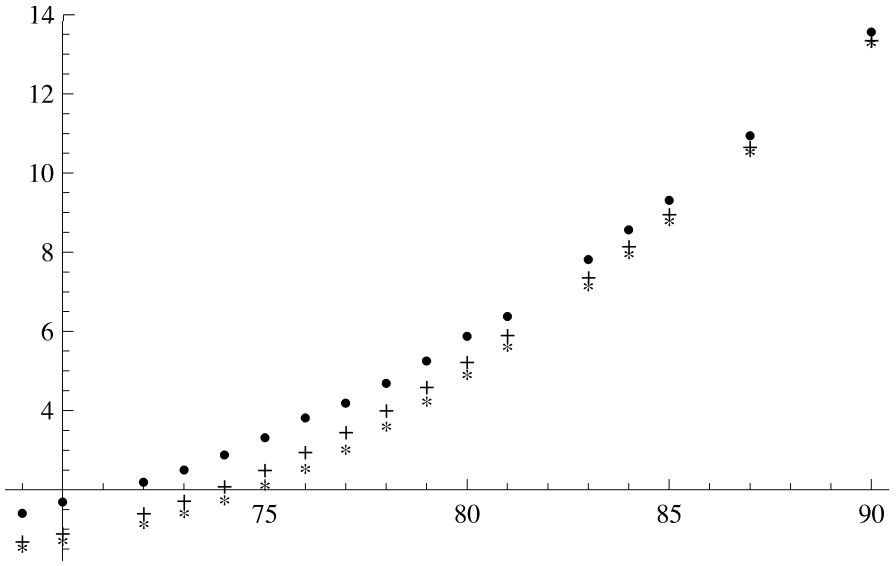
Pro srovnání teoretických výsledků tržními hodnotami je nejdříve potřeba interpolovat výsledné hodnoty Crank-Nicolsonova algoritmu pomocí funkce *ListInterpolation*, čímž se omezí chyba vzniklá tržní hodnotou podkladového aktiva mezi diskrétními hodnotami S_j a zároveň umožní srovnání s výsledkem NDSolve, který je také interpolovaný. Hodnoty opcí budeme porovnávat pro 17 různých hodnot realizační ceny K v rozmezí 69 až 90 $. Všechny tyto opce měly objem obchodování větší než 300 obchodů za den, což ukazuje jejich likviditu a jako důsledek relevantní tržní cenu. Pro tyto výpočty pro různé realizační ceny je výhodné používat transformovanou Black-Scholesovu rovnici (2.4), protože výsledná funkce potažmo matice diskrétních hodnot W nezávisí na realizační ceně K, která je využita až při zpětné transformaci hodnot $w_{i,\tau}$ na $V(S,t)$. Čtvercová odchylka NDSolve výpočtu od tržních hodnot je 6.31885.



Obrázek 6.1: Rozdíl Crank-Nicolson a NDSolve

Hodnoty čtvercových odchylek výsledků Crank-Nicolsonova algoritmu pro různou hustotu aproximovaných bodů $w_{i,\tau}$ uvádí tabulka 6.1. Se zvyšujícím se počtem bodů se odchylka Crank-Nicolsonova algoritmu od výsledku NDSolve snižuje. Na obrázku 6.2 označuje '+' označuje hodnoty z Crank-Nicolsonova algoritmu, '*' hodnoty NDSolve a '·' tržní hodnoty opcí na index NASDAQ100 v závislosti na realizační ceně.

Při rozptylu $\sigma = 0.494515$ uvedeném v [8] vycházejí hodnoty z Crank-Nicolsonova algoritmu v průměru o -0.816 nižší a hodnoty NDSolve v průměru o -0.576 nižší



Obrázek 6.2: Srovnání s tržními hodnotami

než tržní hodnoty. Toto jednostranné vychýlení, které je vidět na obrázku 6.2, může znamenat, že odhad σ je nižší než skutečná volatilita v uvažovaném období. Důvodem může být podceněný odhad volatility z historických dat nebo porušení předpokladu nezávislosti volatility na čase a její zvýšení. Pokud využijeme znalosti tržních cen opcí pro výpočet volatility minimalizací čtvercové odchylky Crank-Nicolsonovou metodou při hustotě dělení $m = Nt = 800$ vyjde volatilita podkladového aktiva $\sigma = 0,59$ a čtvercová odchylka 0.45, výpočtem NDSolve vyjde čtvercová odchylka 0.46.

6.3 Americké opce

Na výpočtech pro americké put opce budeme porovnávat čtyři algoritmy pro aproximaci plochy. Metodou konečných diferencí, metodou konečných prvků (s neekvidistantní sítí (S,t)) a metodou konečných prvků (s ekvidistantní sítí (S,t)), které počítají transformovanou rovnici (2.4). Čtvrtý algoritmus řeší problém maximální okrajové podmínky. První dvě metody můžeme vzhledem ke shodnosti sítě (S,t) porovnat v každém vypočteném bodě. Stejným způsobem můžeme porovnávat druhou dvojici metod. Výpočty provedeme pro parametry

$$r = 0.25; \quad \sigma = 0.6; \quad \delta = 0.2; \quad T = 1; \quad K = 10;$$

americké put opce použitých v [5]. Z demonstrativních důvodů neodpovídá hodnota bezrizikové úrokové míry realitě.

Metoda konečných diferencí
První a druhý krok je shodný s Crank-Nicolsonovým algoritmem pro evropské opce. Pro urychlení výpočtu spočítáme vektory

$$\text{xx=Table[x[i],\{i,0,m\}]; } \tau\tau = \text{Table}\left[\tau[i], \{i, 0, \text{Nt}\}\right];.$$

3. V případě amerických opcí nemáme konkrétní hraniční podmínky jako v případě evropských opcí, ale omezující podmínkou je hodnota opce větší než výplatní funkce

$$g = \text{Outer}\left[\text{Exp}\left[\frac{\#2}{4}\left(\left(q_\delta - 1\right)^2 + 4q\right)\right]\right.$$
$$\left.\text{Max}\left[\text{Exp}\left[\frac{\#1}{2}\left(q_\delta - 1\right)\right] - \text{Exp}\left[\frac{\#1}{2}\left(q_\delta + 1\right)\right], 0\right] \&, \text{xx}, \text{Append}\left[\tau\tau, \tau\left[\text{Nt} + 1\right]\right];$$

v transformované verzi (4.22) pro každé τ z intervalu $[0, \frac{1}{2}\sigma^2 T]$.

4. Pro výpočet SOR algoritmem definujeme parametry
$\epsilon = 10^{-5}$; tolerance chyby,
$\theta = \frac{1}{2}$; parametr pro Crank-Nicolsonovu metodu,
$\lambda = \Delta\tau\,/\Delta x^2$; $\alpha = \lambda\theta$;

Algoritmus výpočtu rozdělíme na vnější cyklus, který bude postupně probíhat přes všechny τ od 0 do $\frac{1}{2}\sigma^2 T$ a vnitřní cyklus, který SOR algoritmem hledá volný okrajový bod, dokud se mu nepřiblíží na zadanou toleranci ϵ.

Vnitřní cyklus

While [Sqrt [Total [(Drop[vv, −1] − Drop[vvn, 1])^2]] > ϵ,
If[Total[vvn] > 0, vv = Append[Drop[vvn, 1], 0]];
Do [ρ = (bb[[i]] + α(vvn[[i]] + vv[[i + 1]]))/(1 + 2α);
vvn[[i + 1]] = Max [g[[i + 1, ν + 1]], vv[[i]] + ω_R(ρ − vv[[i]])] ; , {i, m − 1}; ;

Vnější cyklus

Do[bb = Flatten[{ww[[1]] + λ(1 − θ)(ww[[2]] − 2ww[[1]] + g[[1, ν]]) + αg[[1, ν + 1]],
Table[ww[[i]] + λ(1 − θ)(ww[[i + 1]] − 2ww[[i]] + ww[[i − 1]]), {i, 2, m − 2}],
ww[[m−1]]+λ(1−θ)(g[[m+1, ν]]−2ww[[m−1]]+ww[[m−2]])+αg[[m+1, ν+1]]}];
vv = Append[Table[Max[ww[[i]], g[[i + 1, ν + 1]]], {i, m − 1}], 0];
vvn = Table[0, {m}];
Vnitřní cyklus
ww = Drop[vvn, 1]; W = Append[W, ww]; ,
$\left\{\nu, \mathrm{Nt} + \mathrm{Ceiling}\left[\frac{\mathrm{Nt}}{4}\right]\right\}$

Výsledkem algoritmu je matice W. Použijeme transformaci (6.6) pro výpočet hodnot $V(S,t)$.

Metoda konečných prvků

A) V případě, že budeme počítat na stejné síti prvků jako v případě metody konečných diferencí je první krok shodný s touto metodou.

B) Při požadavku na ekvidistantní síť S, t musí hodnoty vzdáleností mezi body x_i exponenciálně klesat. Zavedeme novou funkci

x[i_]:=$x[i] = N$[Log[Exp[$x[i-1]$] + st]],

kde st je zvolená konstanta v závislosti na hustotě dělení a intervalu x_{min}, x_{max}. Funkce pro výpočet vzdáleností bodů τ_ν zůstane stejná.
Krok dvě je shodný s metodou konečných diferencí. Navíc však vypočítáme vektor

hh = Table[N[x[i] − x[i − 1]], {i, m}];

který využijeme při výpočtu diagonálních hodnot matic A a B

bbn = Table $\left[\frac{\mathrm{hh}_{[[i]]}}{6} + \frac{\gamma}{\mathrm{hh}_{[[i]]}}, \{i, m\}\right]$;

$$\text{bbd} = \text{Table}\left[\frac{\text{hh}[[i]]+\text{hh}[[i+1]]}{3} - \gamma\left(\frac{1}{\text{hh}[[i]]} + \frac{1}{\text{hh}[[i+1]]}\right), \{i, m-1\}\right];$$
$$\text{aan} = \text{Table}\left[\frac{\text{hh}[[i]]}{6} - \frac{\Delta\tau\theta}{\text{hh}[[i]]}, \{i, m\}\right];$$
$$\text{aad} = \text{Table}\left[\frac{\text{hh}[[i]]+\text{hh}[[i+1]]}{3} + \Delta\tau\theta\left(\frac{1}{\text{hh}[[i]]} + \frac{1}{\text{hh}[[i+1]]}\right), \{i, m-1\}\right];$$

Tyto hodnoty se ve výpočtu používají opakovaně, proto jejich uložení vede ke snížení časové náročnosti výpočtu.

3. Transformovaná výplatní funkce je stejná jako u metody konečných diferencí.

4. Před spuštěním výpočtu musíme zadat

$\epsilon = 10^{-5}$; tolerance chyby,
$\theta = \frac{1}{2}$; parametr pro Crank-Nicolsonovu metodu,
$\lambda = \Delta\tau\,/\Delta x^2$; $\gamma = \Delta\tau(1-\theta)$;

Algoritmus výpočtu rozdělíme na vnější cyklus, který bude postupně probíhat přes všechny τ od 0 do $\frac{1}{2}\sigma^2 T$ a vnitřní cyklus, který SOR algoritmem hledá volný okrajový bod, dokud se mu nepřiblíží na zadanou toleranci ϵ.

Vnitřní cyklus

While [Sqrt [Total [(Drop[vv, −1] − Drop[vvn, 1])^2]] > ϵ,
If[Total[vvn] > 0, vv = Append[Drop[vvn, 1], 0]];
Do[ρ = (bb[[i]] − aan[[i]]vvn[[i]] − aan[[$i+1$]]vv[[$i+1$]])/aad[[i]];
vvn[[$i+1$]] = Max [g[[$i+1, \nu+1$]], vv[[i]] + ω_R(ρ − vv[[i]])] , {$i, m-1$}]] ;

Vnější cyklus

Do[bb =
Flatten[{bbn[[1]]g[[1, ν]] + bbd[[1]]ww[[1]] + bbn[[2]]ww[[2]] − aan[[1]] g[[1, $\nu+1$]],
Table[bbn[[i]] ww[[$i-1$]] + bbd[[i]]ww[[i]] + bbn[[$i+1$]]ww[[$i+1$]], {$i, 2, m-2$}],
bbn[[$m-1$]]ww[[$m-2$]] + bbd[[$m-1$]]ww[[$m-1$]] + bbn[[m]]g[[$m+1, \nu$]]
−aan[[m]] g[[$m, \nu+1$]]}];
vv = Append[Table[Max[ww[[i]], g[[$i+1, \nu+1$]]], {$i, m-1$}], 0];
vvn = Table[0, {m}];
Vnitřní cyklus
ww = Drop[vvn, 1]; W = Append[W, ww]; ,
$\left\{\nu, 0, \text{Nt} + \text{Ceiling}\left[\frac{\text{Nt}}{4}\right] - 1\right\}$]

Výsledkem algoritmu je matice W. Použijeme transformaci (6.6) pro výpočet hodnot $V(S,t)$.

Metoda maximální okrajové podmínky
1. Pomocí diskrétních hodnot

$$\text{ss} = N\left[\text{Range}\left[s_{\min}, s_{\max}, \Delta\text{s}\right]\right];$$
$$\text{tt} = N\left[\text{Range}\left[t_{\min}, t_{\max}, \Delta\text{t}\right]\right];$$

zavedeme síť uzlů

$$\text{grid} = \text{Outer}[\{\#1, \#2\}\&, \text{tt}, \text{ss}];$$

kterou použijeme při generování vektorů parametrů

$$\text{aa0} = \text{Table}[-r, \{\text{m}+1\}];$$
$$\text{aa1} = \text{Table}[(r-\delta)\text{grid}[[1, i, 2]], \{i, \text{m}+1\}];$$
$$\text{aa2} = \text{Table}\left[\tfrac{\sigma^2}{2}\text{grid}[[1, i, 2]]\text{\^{}}2, \{i, \text{m}+1\}\right];$$

a vektoru počátečních hodnot

$$\text{pp} = \text{Table}[p[\text{grid}[[1, i, 2]]], \{i, \text{m}+1\}];$$

kde $p[\text{S_}]$ je výplatní funkce. Najdeme největší hodnotu $S_k \leq K$ a odpovídající hodnotu výplatní funkce

$$\text{Stv} = \{\text{Position}[\text{ss}, \text{Select}[\text{ss}, \#1 \leq K\&][[-1]]][[1, 1]], K - \text{Select}[\text{ss}, \#1 \leq K\&][[-1]]\};$$

což je výchozí bod pro hledání okamžiku realizace a připravíme si vektory

$$\text{LDiag} = \text{ReplacePart}\left[\text{Drop}\left[\left(\frac{\text{aa2}}{\Delta\text{s}^2} - \frac{\text{aa1}}{\Delta\text{s}}\right), 1\right], 0, -1\right];$$
$$\text{Diag} = \text{ReplacePart}\left[\text{ReplacePart}\left[\left(-\frac{2\ \text{aa2}}{\Delta\text{s}^2} + \frac{\text{aa1}}{\Delta\text{s}} + \text{aa0} - \tfrac{1}{\Delta\text{t}}\right), 1, -1\right], 1, 1\right];$$
$$\text{UDiag} = \text{ReplacePart}\left[\text{Drop}\left[\left(\frac{\text{aa2}}{\Delta\text{s}^2}\right), -1\right], 0, 1\right];$$

pro matici B (4.33).

2. Funkce

$$\text{ReplaceAtPlaces}[\text{expr_}, \text{new_}, \text{places_}]:=$$
$$\text{Fold}[\text{ReplacePart}[\#1, \#2[[1]], \#2[[2]]]\&, \text{expr}, \text{Transpose}[\{\text{new}, \text{places}\}]]$$

upravuje matici B a vektor na pravé straně rovnice $u^{(n+1)}$ do požadovaného tvaru, aby řešení funkce

```
BoundaryValueSol[Stv_, LDiag_, Diag_, UDiag_, pp_]:=
 Module [{ModifiedLDiag, ModifiedDiag, ModifiedUDiag, RHS, ModifiedRHS, j},
 j = Stv[[1]]; RHS = -pp/Δt;
 ModifiedLDiag = ReplaceAtPlaces[LDiag, Table[0, {j - 1}], Range[1, j - 1]];
 ModifiedDiag = ReplaceAtPlaces[Diag, Table[1, {j - 1}], Range[2, j]];
 ModifiedUDiag = ReplaceAtPlaces[UDiag, Table[0, {j - 1}], Range[2, j]];
 ModifiedRHS = ReplaceAtPlaces[RHS, (pp[[#1]]&)/@Range[1, j], Range[1, j]];
 LinearSolve[SparseArray[{Band[{2, 1}] → ModifiedLDiag, Band[{1, 1}]
 → ModifiedDiag, Band[{1, 2}] → ModifiedUDiag}, m + 1], ModifiedRHS]
```

odpovídalo (4.31). Hledání maximálního bodu, který tuto podmínku splňuje, v konkrétní časová vrstvě provádí funkce

```
MaxBoundaryValueSol[Stv_, LDiag_, Diag_, UDiag_, pp_]:=
 NestWhile[{#[[1]] + {-1, ΔS},
 BoundaryValueSol[Modify[#[[1]] + {-1, ΔS}, LDiag, Diag, UDiag, pp]]}&,
   {Stv + {1, -ΔS}, pp}, #1[[2]][[Stv[[1]]]] ≤ #2[[2]][[Stv[[1]]]]&, 2, Stv[[1]], -1];
```

která postupně počítá vektor $u^{(n)}$ pro klesající hodnoty S_j dokud nedosáhne maximální hodnoty $u(S_j, t_n)$ v bodě S_{n+1}, který je okrajovou podmínkou v předchozí časové vrstvě. Voláním této funkce pro všechny časové vrstvy

```
V = NestList
 [MaxBoundaryValueSol[#1[[1]], LDiag, Diag, UDiag, #1[[2]]]&, {Stv, pp}, Nt];
```

Vzhledem k použití nerovnosti Black-Scholesova typu v netransformované podobě obsahuje výsledná matice V hodnoty $V(S, t)$, takže není potřeba zpětná transformace.

Srovnání

Srovnání provedeme pro hustotu dělení $m = Nt = 800$, které jak jsem ověřili pro evropské opce dává dostatečně přesné řešení a časová náročnost výpočtů není neúnosně vysoká. U výpočtů SOR algoritmem dalším významným parametrem ovlivňujícím časovou náročnost nastavení požadované přesnosti ϵ. Podle [5] je doporučená hodnota $\epsilon = 10^{-5}$. Konkrétní hodnoty časové náročnosti pro všechny hodnoty jsou uvedeny v tabulce 6.2 .

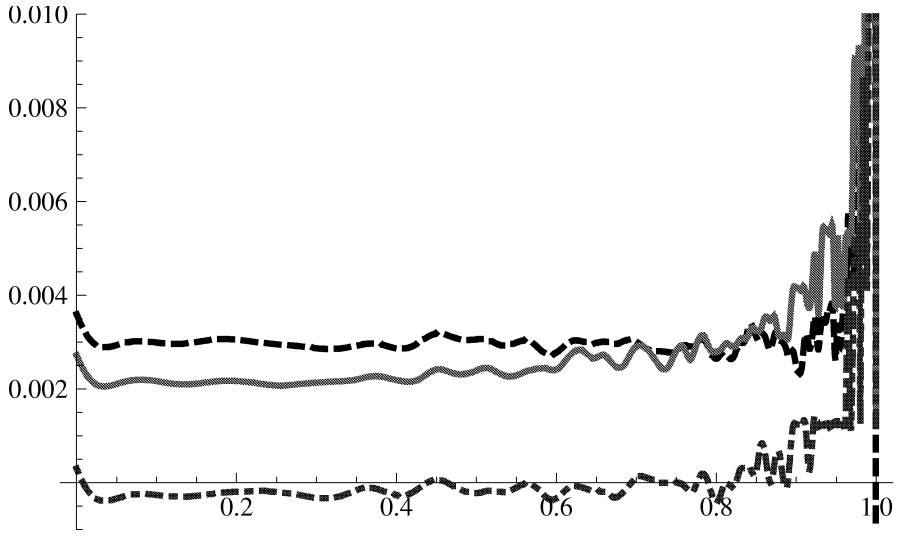
Jedním z hlavních cílů při výpočtech týkajících se amerických opcí je funkce $S_f(t)$, která určuje cenu podkladového aktiva, při které se vyplatí americkou opci realizovat. Proto se při srovnání různých metod výpočtu zaměříme na rozdíly funkčních hodnot $S_f(t)$. Funkce $S_f(t)$ je interpolovaná z aproximovaných bodů dotyku pomocí funkce *Interpolate*. V závislosti na hustotě dělení sítě je možné volit parametr *InterpolationOrder*, který ovlivňuje hladkost interpolované funkce. Jako srovnávací funkce byla vybrána $S_f(t)$ vypočtená metodu maximální okrajové podmínky (dále MMOP). Aproximací metodou konečných diferencí a metodou konečných prvků, která počítá na stejné

metoda	doba výpočtu	přesnost ϵ
konečné diference	251.302	10^{-5}
konečné prvky (A)	278.991	10^{-5}
konečné prvky (B)	512.857	10^{-5}
konečné prvky (B)	2524.209	10^{-10}
maximální hodnota	40.078	

Tabulka 6.2: Rozdíl Crank-Nicolson a NDSolve

síti prvků jako metoda konečných diferencí vychází $S_f(t)$ podle očekávání stejná. To už neplatí pro MMOP a metodu konečných prvků, které využívají také stejnou síť pro aproximování bodů, ale metody výpočtu se diametrálně odlišují. Metoda konečných prvků využívá klasické numerické řešení pomocí SOR algoritmu, zatímco metoda maximální okrajové podmínky vychází z vlastností aproximované plochy a diferenciálních rovnic.

Na obrázku 6.3 jsou znázorněny relativní odchylky výsledků dalších metod: čerchovanou a plnou křivkou metoda konečných prvků B s nastavení přesnosti $\epsilon = 10^{-10}$ a $\epsilon = 10^{-5}$, čárkovanou křivkou pak metoda konečných diferencí s přesností $\epsilon = 10^{-5}$. Pro zvyšující se přesnost se výsledky SOR algorimu přibližují výsledkům MMOP. Průměrná relativní odchylka nejpřesnějšího výpočtu je 0.00022. Obdobně jako u evropských opcí způsobuje největší nestabilitu numerického výpočtu výplatní funkce, resp. její zlom v bodě K v čase T, proto jsou v čase $t \to T$ největší odchylky mezi aproximovanými funkcemi. Aproximace SOR algoritmem v tomto případě vychází větší než MMOP. Maximum těchto odchylek je menší než dva procentní body. Maximum v intervalu $(0, 0.9\ T)$ je menší než půl procentního bodu a průměrná relativní odchylka $0,003$.



Obrázek 6.3: Relativní odchylka $S_f(t)$ vypočtených SOR od MMOP

Kapitola 7

Závěr

Přes svoje stáří je Black-Scholesova formule stále často využívaným nástrojem při oceňování opcí. Přestože při jejím odvození byly učiněny některé předpoklady, které v tržním prostředí neplatí podává podle mnoha provedených studií dobré výsledky. Porušení některých předpokladů o trhu nemůžeme při výpočtech ovlivnit. Dalším důležitým předpokladem je použití správných parametrů. Bezriziková úroková míra je parametr, který je poměrně stabilní a dobře známý. Oproti tomu volatilita podkladového aktiva je parametr, který můžeme aproximovat několika různými metodami. Podle [1] k základním metodám patří odhad směrodatné odchylky logaritmických výnosů z historických dat. Další možností je vyjít ze znalosti tržních cen opcí a použít Black-Scholesovu rovnici pro odhad budoucí volatility. Složitější odhady volatility vycházejí ze stochastického modelu volatility nebo GARCH modelu. Numerické výpočty nezbytné pro výpočet amerických opcí při požadavcích na vysokou přesnost vyplývající zejména z hustoty aproximovaných bodů jsou časově nezanedbatelně náročné, zejména při použití SOR algoritmu potažmo Gauss-Seydelovy metody, což je daň za možnost jejich obecného využití. Narozdíl od doporučení [5] je lepší vzhledem k rychlosti konvergence volit relaxační parametr ω_R větší než jedna. Výhodou využití transformace Black-Scholesovi rovnice na rovnici vedení tepla je nezávislost na realizační ceně opce, čehož lze využít při výpočtech ceny opce pro různé realizační ceny při stejné době do splatnosti. Vzhledem k nejnižší časové náročnosti a přesnosti výpočtu byla pro demonstrační ukázku zvolena metoda maximální okrajové podmínky.

Literatura

[1] Dupačová, J., Hurt, J., Štěpán, J.: *Stochastic Modeling in Economics and Finance*, Kluwer Academic Publishers, Praha, 2002.

[2] Hindmarsh, A., Taylor, A.: *User Documentation for IDA, A Differential-Algebraic Equation Solver for Sequential and Parallel Computers*, Lawrence Livermore National Laboratory, UCRL-MA-136910, 1999.

[3] Lamberton, D., Lapeyre, B.: *Introduction to Stochastic Calculus Applied to Finance*, Chapman & Hall, London, 1996.

[4] Ralston, A.: *Základy numerické matematiky*, Academia, Praha, 1973.

[5] Seydel, R.: *Tools for Computational Finance*, Springer, Berlin, 2002.

[6] Shaw, W.: *Modelling Financial Derivatives with Mathematica*, Cambridge University Press, Cambridge, 1998.

[7] Stoer, J., Bulirsch, R.: *Introduction to Numerical Analysis*, Springer, New York, 2002.

[8] Stojanovic,S.: *Computational Financial Mathematics using Mathematica*, Birkhäuser Boston, Boston, 2003.

[9] Wolfram, S.: *The Mathematica Book*, Wolfram Media, Champaign, 2003.