FJFI ČVUT

METODA MONTE CARLO

SEMINÁRNÍ PRÁCE

Parabolická evoluční úloha ve 2D

Autor
Vladislav BELOV

5. prosince 2017

1 Úvod

Existuje hodně způsobů řešení diferenciálních rovnic. Tato seminární práce se zabývá hledáním aproximace řešení parabolické evoluční úlohy popisující vedení tepla (resp. difuzi)[1] ve dvou dimenzích pomocí metody Monte Carlo.

Obecně parabolická evoluční úloha ve 2D na oblasti $(0,T)\times\Gamma$, kde $T\in\mathbb{R}_+$ a $\Gamma\subset\mathbb{R}^2$, vypadá následovně:

$$
\begin{aligned}
\frac{\partial}{\partial t}u(t,x) - D\cdot\triangle u(t,x) &= f(t,x),\\
u|_{\partial\Gamma} &= g(x),\\
u(0,x) &= h(x).
\end{aligned}
\tag{1}
$$

V dané notaci funkce $u=u(t,x)$ popisuje rozložení teploty na oblasti Γ (tj. $x\in\Gamma\subset\mathbb{R}^2$) v časech $t\in(0,T)$, D je termální difuzivita ($D>0$), $g=g(x)$ je okrajová podmínka evoluční úlohy $\forall t\in(0,T)$ a $h=h(x)$ je počáteční podmínka. V rámci této seminární práce položíme pravou stranu $f\equiv 0$.

2 Diskretizace úlohy

V numerické matematice se podobné úlohy řeší pomocí metody konečných diferencí (metody sítí), která spočívá v diskretizaci oblasti $(0,T)\times\Gamma$ a konstrukci diferenčního schématu (viz [Ben17]). Pro jednoduchost budeme hledat řešení úlohy (1) na čtverci $\Gamma=(-b,b)\times(-b,b)$, $b>0$, pak na jeho uzávěru $\overline{\Gamma}$ lze definovat čtvercovou síť $\overline{\omega}_h$ s krokem $h>0$, tj. $\Gamma\leftrightarrow\omega_h$ a $\partial\Gamma\leftrightarrow\overline{\omega}_h\setminus\omega_h$. Potom, rozdělíme-li interval $[0,T]$ na časové hladiny s krokem $\tau=\frac{T}{N_T}$, kde N_T je počet hladin, a označíme-li $u^k_{i,j}$ hodnoty funkce u na časových hladinách $k\in\{0,1,\dots,N_T\}$ pro všechny (i,j) body sítě $\overline{\omega}_h$, dostaneme podle [Ben17] následující diferenční schéma:

$$
\begin{aligned}
\frac{u^{k+1}_{i,j}-u^k_{i,j}}{\tau} = D\cdot\Big(\frac{u^k_{i+1,j}-2u^k_{i,j}+u^k_{i-1,j}}{h^2}+\frac{u^k_{i,j+1}-2u^k_{i,j}+u^k_{i,j-1}}{h^2}\Big),\ \forall(i,j)\in\omega_h,\ \forall k\in\{0,1,\dots,N_T-1\};\\
u^k_{i,j}=g_{i,j},\forall(i,j)\in\overline{\omega}_h\setminus\omega_h,\ \forall k\in\{0,1,\dots,N_T\};\\
u^0_{i,j}=h_{i,j},\ \forall(i,j)\in\overline{\omega}_h.
\end{aligned}
\tag{2}
$$

Takovému-to diferenčnímu schématu se říká *explicitní* a jeho chyba aproximace je $O(\tau+h^2)$. Budeme-li volit kroky τ a h v poměru $\frac{h^2}{\tau}=4$ a položíme-li $D=1$, pak (2) se přepíše následovně:

$$
\begin{aligned}
u^{k+1}_{i,j} = \frac{1}{4}\Big(u^k_{i+1,j}+u^k_{i-1,j}+u^k_{i,j+1}+u^k_{i,j-1}\Big),\ \forall(i,j)\in\omega_h,\ \forall k\in\{0,1,\dots,N_T-1\};\\
u^k_{i,j}=g_{i,j},\forall(i,j)\in\overline{\omega}_h\setminus\omega_h,\ \forall k\in\{0,1,\dots,N_T\};\\
u^0_{i,j}=h_{i,j},\ \forall(i,j)\in\overline{\omega}_h.
\end{aligned}
\tag{3}
$$

Stojí za zmínění, že schéma (3) je stabilní, neboť obecná podmínka stability pro (2) má tvar:

$$\frac{1}{2}\geq D\cdot\frac{\tau}{h^2}.$$

Získanou soustavu lineárních algebraických rovnic budeme řešit pomocí metody Monte Carlo.

3 Algoritmus výpočtu aproximace

Postup pro řešení soustavy rovnic (3) získané diskretizací úlohy (1) je velmi jednoduchý - použijeme lehce upravenou náhodnou procházku. Aproximaci $u^{k_0}_{ij}$ pro jistou pevnou časovou hladinu $k_0\in\{0,1,\dots,N_T\}$ dostaneme jako výběrovou střední hodnotu jisté náhodné veličiny $Y_{W_{ini}}$ definované následujícím způsobem ([Vir10]):

[1] Záleží na tom, jakou fyzikální interpretaci parabolické evoluční úlohy zvolíme.

1. označíme počáteční bod náhodné procházky $W_{ini} = (i, j) \in \omega_h$, který se nachází na časové hladině k_0;
2. z bodu W_{ini} přejde náhodná procházka s pravděpodobností $\frac{1}{4}$ do libovolného sousedního bodu, který je na časové hladině $k_0 - 1$;
3. pokud je náhodná procházka stále ve vnitřním bodě sítě $\overline{\omega}_h$ (tj. není na hranici $\overline{\omega}_h \setminus \omega_h$) a není na nulové časové hladině, vrátíme se na krok 2, jinak jdeme na krok 4;
4. označíme-li W_{fin} bod sítě, ve kterém se nachází náhodná procházka v daném okamžiku, pak definujeme náhodnou veličinu $Y_{W_{ini}}$ takto:
   - $Y_{W_{ini}} = h(W_{fin})$, pokud $W_{fin} \in \omega_h$, ale $W_{fin} \notin \overline{\omega}_h$;
   - $Y_{W_{ini}} = g(W_{fin})$, pokud $W_{fin} \in \overline{\omega}_h \setminus \omega_h$.

Zopakujeme-li tento postup pro všechny body $(i, j) \in \omega_h$, dostaneme aproximaci[2] řešení diferenčního schématu (3).

Označíme-li $\hat{\mu}$ výběrovou střední hodnotu náhodné veličiny Y_W pro všechna W a $\hat{s}^2$ její výběrový rozptyl, potom chybu δ vypočítané aproximace nalezneme pomocí vzorce, který přímo vyplývá z Čebyševovy nerovnosti:

$$\delta = |EY_W - \hat{\mu}| = \sqrt{\frac{\hat{s}^2}{N \cdot \varepsilon}}, \tag{4}$$

kde N je počet simulací v rámci metody Monte Carlo (tzn. $N \equiv$ počet náhodných procházek vycházejících z bodu W sítě ω_h) a $0 < \varepsilon < 1$ je hladina významnosti.

4 Implementace

Součástí této seminární práce je také implementace v programovacím jazyce Matlab, která se skládá ze čtyř souborů. Tři z nich obsahují definice funkcí, jež budou popsány podrobněji v následujících podsekcích. V čtvrtém souboru `main.m` lze nalézt pouze vyvolání řešiče parabolické parciální diferenciální rovnice, který je implementován v `parabolicPDESolver.m`.

4.1 Funkce `parabolicPDESolver`

Tato funkce odpovídá za výpočet aproximace řešení diferenčního schématu (3) podle algoritmu popsaného v sekci 3.

- Vstupní parametry:
  - `numOfSim` - počet simulací v rámci metody Monte Carlo;
  - `b` - číslo $b > 0$ určující oblast $\Gamma = (-b, b) \times (-b, b)$;
  - `meshStep` - krok h sítě $\overline{\omega}_h$;
  - `maxTime` - časová hladina k_0, pro niž hledáme aproximaci řešení;
  - `verbose` - parametr podrobného výpisu.
- Výstupní parametry:
  - `solution` - hledaná aproximace řešení schématu (3);
  - `solutionErr` - chyba aproximace řešení.

[2] Ve zdroji [Vir10] lze nalézt důkaz toho, že střední hodnoty Y_W pro všechna W řeší soustavu rovnic (3).

4.2 Funkce `meshRandomWalk`

Tato funkce odpovídá za simulaci náhodné procházky na síti $\overline{\omega}_h$.

- Vstupní parametry:
  - `xStartPos` - x-ová souřadnice bodu W_{ini};
  - `yStartPos` - y-ová souřadnice bodu W_{ini};
  - `meshStep` - analogický význam jako v podsekci 4.1;
  - `b` - analogický význam jako v podsekci 4.1;
  - `maxTime` - analogický význam jako v podsekci 4.1;
- Výstupní parametry:
  - `xFinPos` - x-ová souřadnice bodu W_{fin};
  - `yFinPos` - y-ová souřadnice bodu W_{fin}.

4.3 Funkce `calcStats`

Tato funkce odpovídá za výpočet statistických vlastností náhodné veličiny Y_W definované v sekci 3.

- Vstupní parametry:
  - `rv` - vektor hodnot náhodné veličiny Y_W;
  - `signLvl` - hladina významnosti;
  - `verbose` - parametr podrobného výpisu.
- Výstupní parametry:
  - `ex` - výběrová střední hodnota náhodné veličiny Y_W;
  - `dx` - výběrový rozptyl náhodné veličiny Y_W;
  - `err` - chyba aproximace.

5 Simulace a výsledky

Pro demonstraci funkčnosti algoritmu uvedeného v sekci 3 a jeho implementace popsané v sekci 4 byla provedena řada simulací. Okrajová a počáteční podmínky úlohy (1) byly zvoleny v následujícím tvaru[3]:

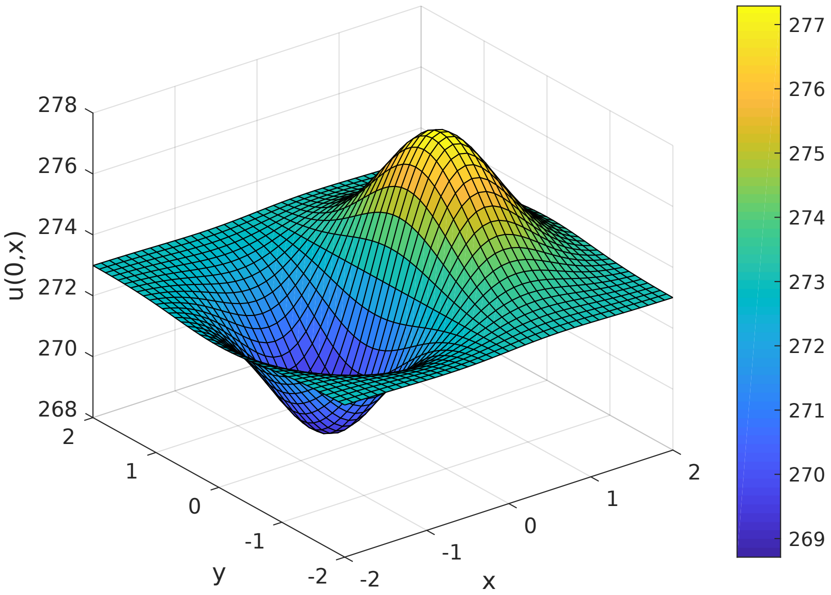
$$u|_{\partial\Gamma} = g(x) = 273; \tag{5}$$

$$u(0, x) = h(x) = 10x \cdot \exp(-x^2 - y^2) + 273. \tag{6}$$

Počáteční podmínka (6) popisuje chování funkce $u = u(t, x)$ na čtverci $\overline{\Gamma}$ v čase $t = 0$; toto chování je znázorněno na Obr. 1 (zvolíme síť $\overline{\omega}_h$ s krokem $h = 0,1)$[4]. Očekávali bychom, že časem pozorovaný systém prostřednictvím šíření tepla (resp. procesu difuze) dosáhne nějaké střední teploty na celém čtverci $\overline{\Gamma}$, což můžeme pozorovat na Obr. 2, 4, 6, 8. Na zmíněných diagramech je zobrazen průběh aproximace řešení původní úlohy ve 3D a ve 2D pro časové hladiny $k_0 \in \{50, 250\}$ a různé počty simulací $N \in \{100, 1000, 10000, 100000\}$. Na Obr. 3, 5, 7, 9 lze vidět odpovídající těmto simulacím chyby spočítané pomocí vzorce (4) na hladině významnosti 5 %. Zajímavým pozorováním je to, že chyba je maximální ve středu $\overline{\Gamma}$, tj. v bodech sítě, ze kterých se náhodné procházky nejméně pravděpodobně dostanou do hranice $\partial\Gamma$.

[3] Kdyby byla potřeba, v implementačním souboru lze tyto podmínky snadno měnit.

[4] Lze volit i menší hodnoty, což ale způsobí podstatné zvýšení času potřebného na výpočet aproximace řešení na celém čtverci $\overline{\Gamma}$.



Obrázek 1: Grafické znázornění počáteční podmínky (6).

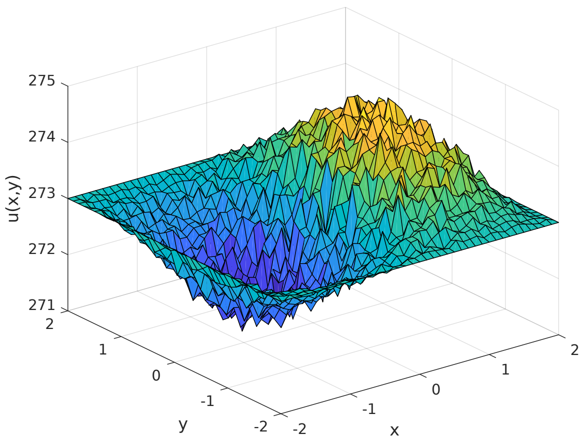
Vzhledem k tomu, že na čtverci $\overline{\Gamma}$ pro zvolenou síť bylo spočítáno řadově 10^3 chyb aproximace řešení, uvedeme v Tab. 1 pro různé počty simulací N pouze maximální chyby. Lze spekulovat o závislosti velikosti maximální chyby na zvolené časové hladině: je vidět, že pro vetší hladinu chyba je menší. Tohle může být způsobeno tím, že se systém časem stabilizuje a hodnoty skutečného řešení úlohy se přibližují k jisté střední hodnotě (v našem případě je tato hodnota rovna 273).

Počet simulací, N	Max. chyba, $k_0 = 50$	Max. chyba, $k_0 = 250$
10^2	1.1434	0.6963
10^3	0.3433	0.1830
10^4	0.1061	0.0560
10^5	0.0333	0.0176

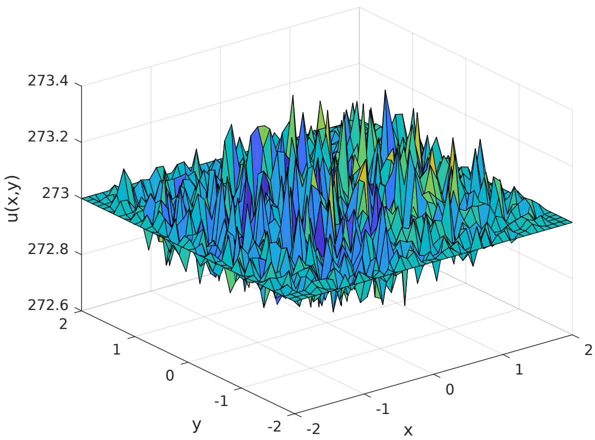
Tabulka 1: Maximální chyba aproximace řešení diskretizované parabolické evoluční úlohy pro časové hladiny $k_0 \in \{50, 250\}$.

6 Závěr

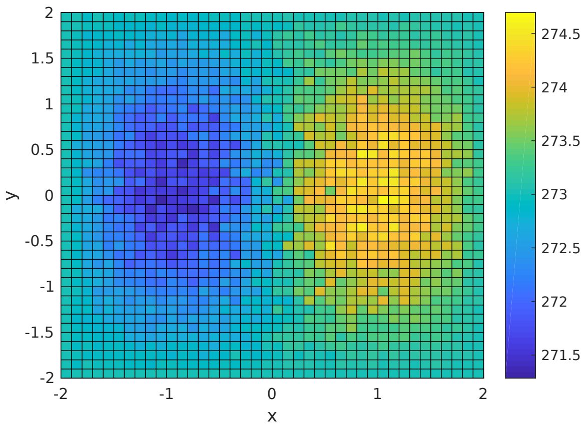
V této seminární práci bylo ukázáno, že pomocí Metody Monte Carlo lze aproximovat řešení parabolické evoluční úlohy ve 2D. Uvedený postup lze zobecnit do více dimenzí, čímž ale se o hodně zvětší časová náročnost výpočtů. Nicméně, pro aproximaci řešení v nějakém konkrétním zvoleném bodě tento postup je zcela použitelný i ve více dimenzích.



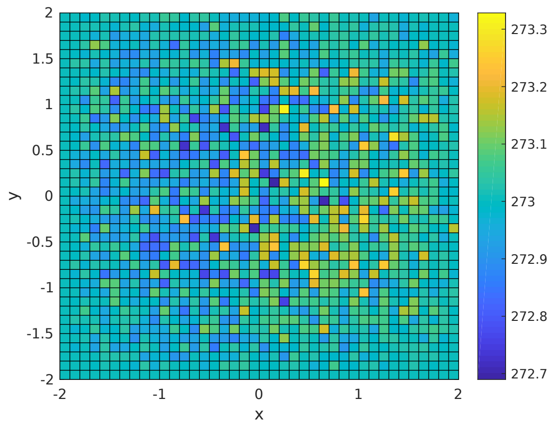
(a) Časová hladina $k_0 = 50$.



(b) Časová hladina $k_0 = 250$.

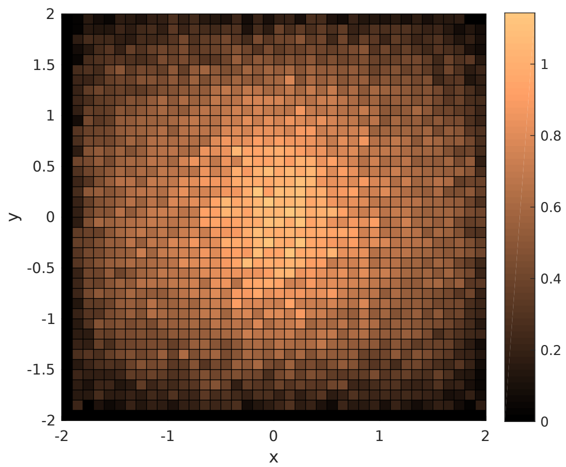


(c) Časová hladina $k_0 = 50$.

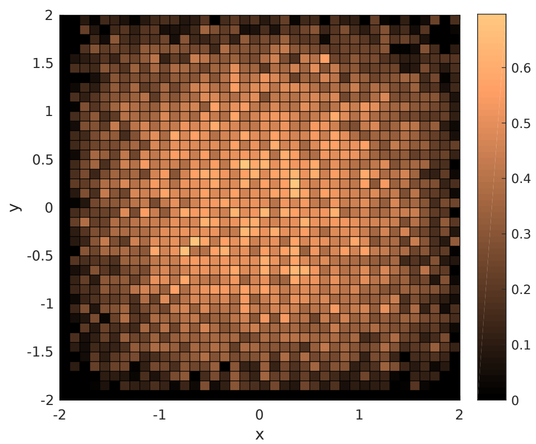


(d) Časová hladina $k_0 = 250$.

Obrázek 2: Aproximace řešení úlohy (3) na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$, která byla dosažená při počtu simulací $N = 100$ na různých časových hladinách.

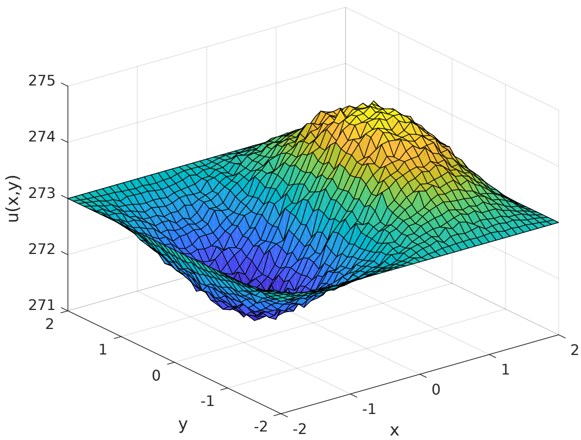


(a) Časová hladina $k_0 = 50$.

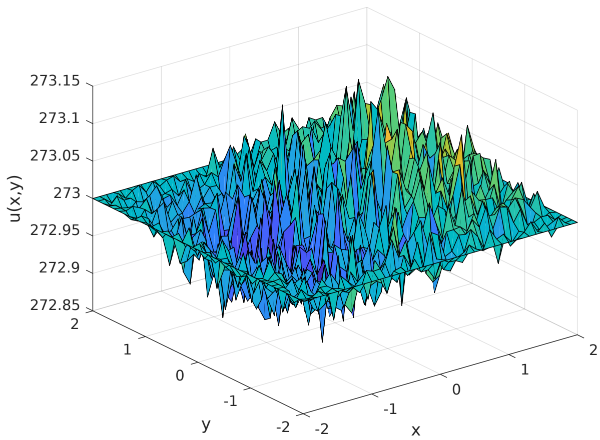


(b) Časová hladina $k_0 = 250$.

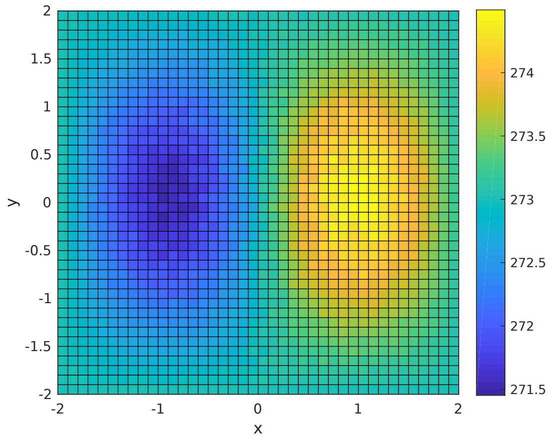
Obrázek 3: Grafické znázornění chyby aproximace řešení úlohy (3) pro počet simulací $N = 100$ na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$ (5% hladina významnosti).



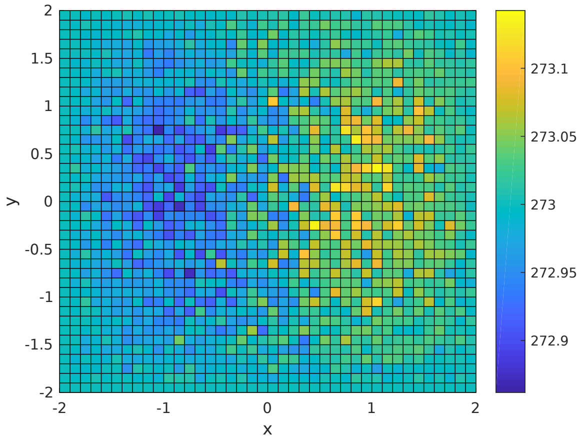
(a) Časová hladina $k_0 = 50$.



(b) Časová hladina $k_0 = 250$.

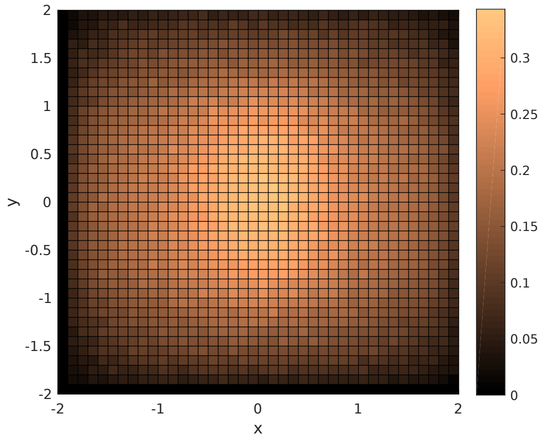


(c) Časová hladina $k_0 = 50$.

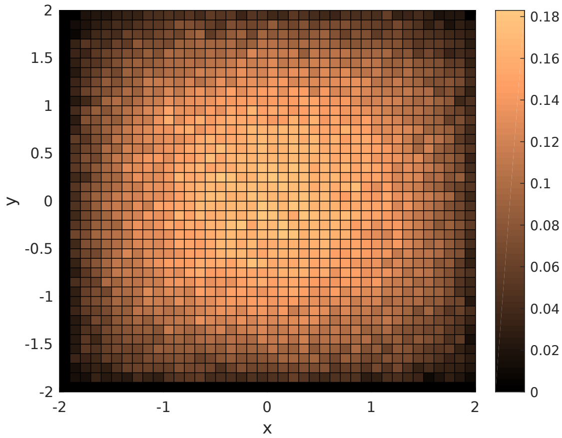


(d) Časová hladina $k_0 = 250$.

Obrázek 4: Aproximace řešení úlohy (3) na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$, která byla dosažená při počtu simulací $N = 1000$ na různých časových hladinách.

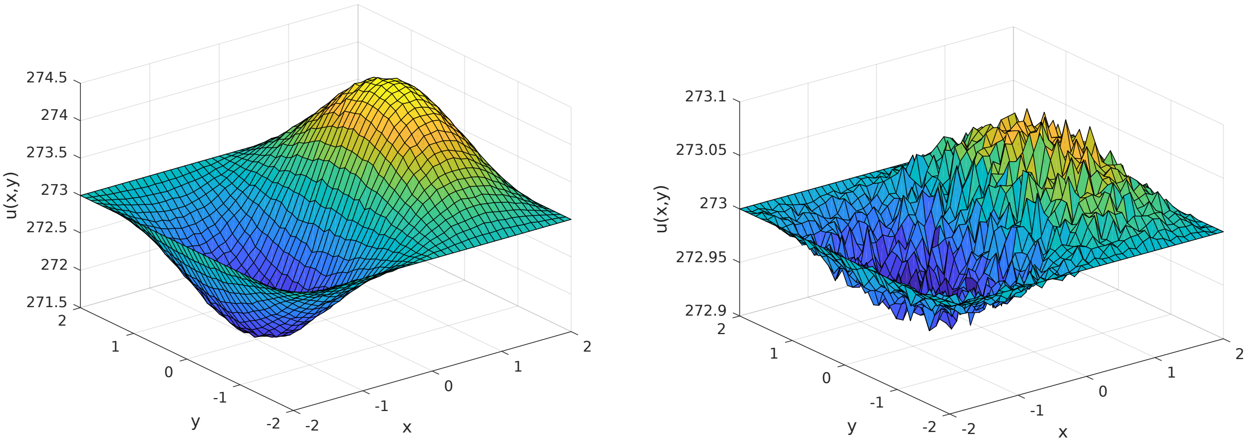


(a) Časová hladina $k_0 = 50$.



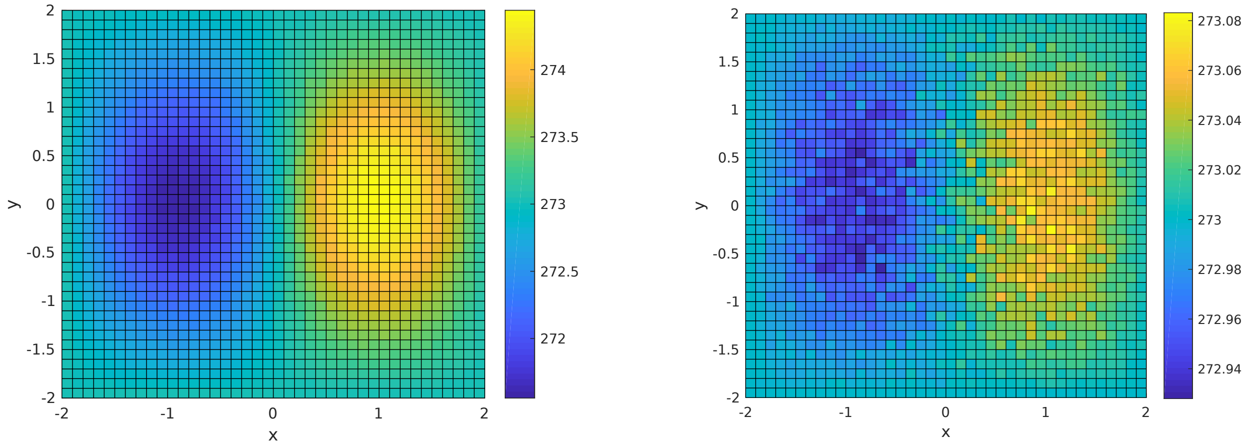
(b) Časová hladina $k_0 = 250$.

Obrázek 5: Grafické znázornění chyby aproximace řešení úlohy (3) pro počet simulací $N = 1000$ na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$ (5% hladina významnosti).



(a) Časová hladina $k_0 = 50$.

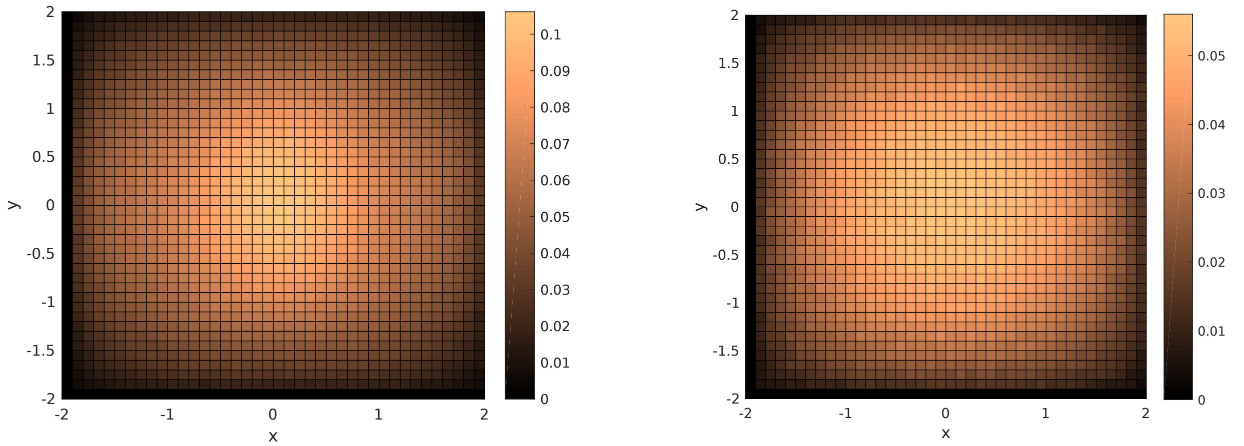
(b) Časová hladina $k_0 = 250$.



(c) Časová hladina $k_0 = 50$.

(d) Časová hladina $k_0 = 250$.

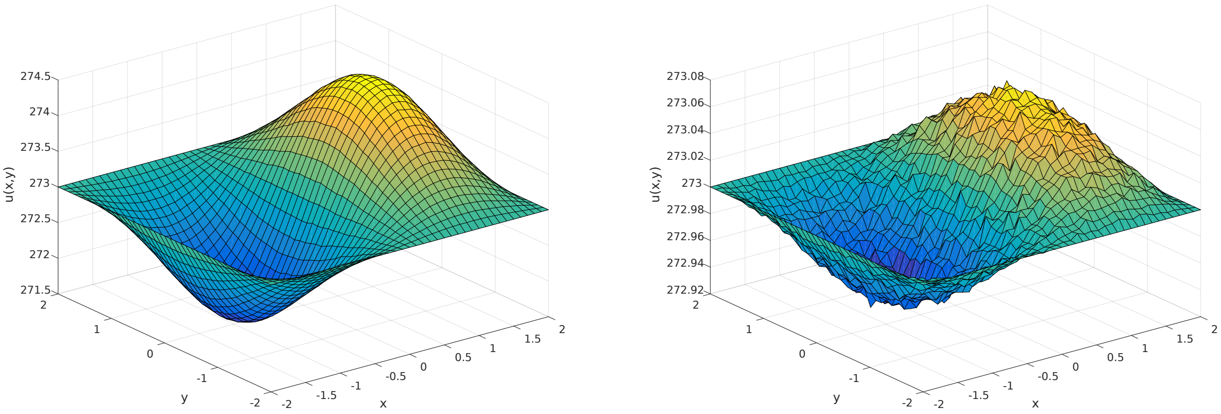
Obrázek 6: Aproximace řešení úlohy (3) na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$, která byla dosažená při počtu simulací $N = 10000$ na různých časových hladinách.



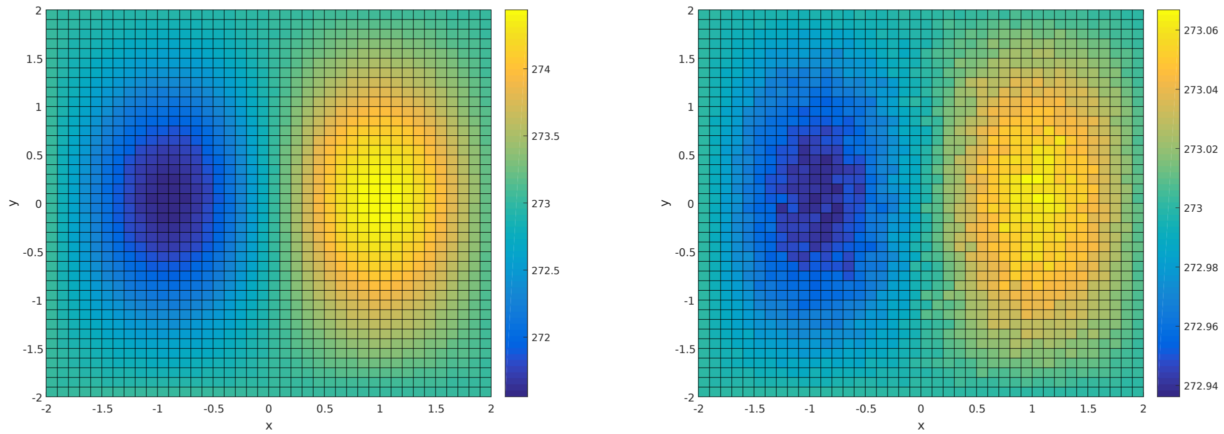
(a) Časová hladina $k_0 = 50$.

(b) Časová hladina $k_0 = 250$.

Obrázek 7: Grafické znázornění chyby aproximace řešení úlohy (3) pro počet simulací $N = 10000$ na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$ (5% hladina významnosti).

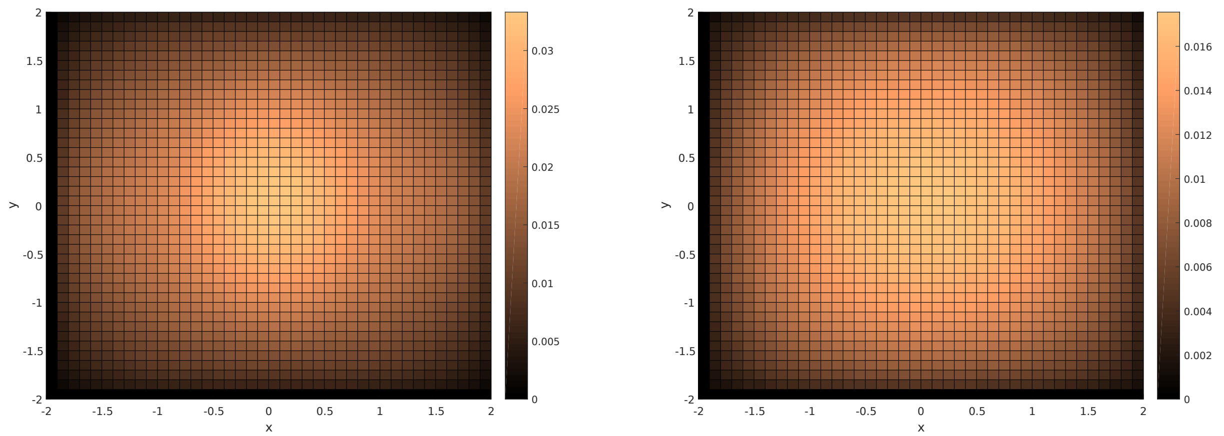


(a) Časová hladina $k_0 = 50$. (b) Časová hladina $k_0 = 250$.



(c) Časová hladina $k_0 = 50$. (d) Časová hladina $k_0 = 250$.

Obrázek 8: Aproximace řešení úlohy (3) na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$, která byla dosažená při počtu simulací $N = 100000$ na různých časových hladinách.



(a) Časová hladina $k_0 = 50$. (b) Časová hladina $k_0 = 250$.

Obrázek 9: Grafické znázornění chyby aproximace řešení úlohy (3) pro počet simulací $N = 100000$ na čtverci $\overline{\Gamma} = [-2,2] \times [-2,2]$ (5% hladina významnosti).

Reference

[Ben17] M. Beneš. Numerická matematika 2. Fakulta jaderná a fyzikálně inženýrská Českého vysokého učení technického v Praze (FJFI ČVUT), 2017.

[Vir10] M. Virius. *Metoda Monte Carlo*. Skripta. České vysoké učení technické, 2010.