



**MATEMATICKO-FYZIKÁLNÍ
FAKULTA**
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Štěpán Adam Chovaneček

**Parareal metoda pro numerické řešení
počátečních úloh**

Katedra numerické matematiky

Vedoucí bakalářské práce: prof. RNDr. Vít Dolejší, Ph.D.,
DSc.

Studijní program: Matematika

Studijní obor: Obecná matematika

Praha 2025

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V dne

Podpis autora

Mé poděkování za příjemnou spolupráci, trpělivost a velice cenné připomínky, které mi umožnily práci rozvíjet požadovaným směrem, patří vedoucímu práce prof. RNDr. Vítu Dolejšimu, Ph.D., DSc.. Zároveň bych za obdržanou podporu v průběhu celého studia rád poděkoval své rodině a přítelkyni.

Název práce: Parareal metoda pro numerické řešení počátečních úloh

Autor: Štěpán Adam Chovaneček

Katedra: Katedra numerické matematiky

Vedoucí bakalářské práce: prof. RNDr. Vít Dolejší, Ph.D., DSc., Katedra numerické matematiky

Abstrakt: V práci uvažujeme nelineární časově závislou rovnici vedení tepla, ze které metodou konečných prvků odvodíme soustavu obyčejných diferenciálních rovnic. Zaměřujeme se na paralelní metodu parareal pro tento systém rovnic. K diskretizaci soustavy ODR je použita implicitní Eulerova metoda a k řešení příslušné nelineární soustavy algebraických rovnic iterační Kačanova metoda. Zvláštní pozornost věnujeme adaptivní volbě časového kroku. Konkrétní úlohu pak řešíme v jazyce fortran a zaměřujeme se na konvergenci paralelního řešení k řešení jemně spočtenému. Pozorujeme také teoretické paralelní zrychlení z pohledu provedených časových kroků.

Klíčová slova: obyčejné diferenciální rovnice, metoda parareal, metoda konečných prvků

Title: Parareal method for the numerical solution of initial value problems

Author: Štěpán Adam Chovaneček

Department: Department of Numerical Mathematics

Supervisor: prof. RNDr. Vít Dolejší, Ph.D., DSc., Department of Numerical Mathematics

Abstract: In this thesis we consider a nonlinear time-dependent heat equation, from which we derive a system of ordinary differential equations using the finite element method. We focus on the parallel parareal method for this system of equations. The implicit Euler method is used to discretize the ODR system and the iterative Kachanov method is used to solve the corresponding nonlinear system of algebraic equations. Special attention is paid to the adaptive choice of the time step. We then solve the concrete problem in the Fortran language and focus on the convergence of the parallel solution to the fine computed solution. We also observe the theoretical parallel acceleration in terms of the time steps performed.

Keywords: ordinary differential equations, parareal method, finite element method

Obsah

Úvod	2
1 Formulace úlohy	3
1.1 Prostory funkcí	3
1.2 Silná formulace	4
1.3 Slabá formulace	5
1.4 Slabá formulace s hraniční penalizací	6
2 Numerické řešení	8
2.1 Prostorová semidiskretizace úlohy pomocí MKP	8
2.2 Numerická kvadratura	10
2.3 Časová diskretizace	10
2.4 Řešení nelineárních algebraických rovnic	10
2.5 Adaptivní volba časového kroku	11
3 Paralelní řešení	13
3.1 Metoda vícenásobné střelby	13
3.2 Metoda parareal	14
3.3 Praktická realizace	17
4 Numerické experimenty	18
4.1 Konvergence metody parareal	20
4.2 Teoretické zrychlení paralelizace	25
Závěr	28
Seznam použité literatury	29
Seznam obrázků	31
Seznam tabulek	32

Úvod

Numerické výpočty implementované v počítačích mohou být v komplexnějších a mohutnějších úlohách bržděny malou výpočetní silou procesoru. Tuto bariéru se snažíme dlouhodobě oddalovat a dostat se za hranice nynějších možností. Avšak postupně se tyto hranice zdají být těžce prostupitelné, ne-li dokonce neprolomitelné. Alternativní cestu pokroku skýtá použití paralelní výpočtové síly. Je tedy záhodno zkoumat a vyvíjet numerické metody uzpůsobené pro paralelní počítání. Oblastí, které se dostalo této pozornosti jen pomálu je například řešení soustavy obyčejných diferenciálních rovnic. Vskutku takové problémy jsou ve své podstatě sekvenční a jejich paralelizace se může zdát zcestná. My si však ukážeme, že toto odvětví má mnohé co nabídnout.

V této práci se seznámíme s metodou parareal a aplikujeme ji na soustavu obyčejných diferenciálních rovnic, kterou získáme standardní metodou konečných prvků z parabolické diferenciální rovnice, konkrétně nelineární rovnice vedení tepla. Základní myšlenka metody parareal se zakládá nejprve na vyřešení soustavy stabilní numerickou metodou s hrubým časovým krokem. Jedno takové řešení nám časový interval rozdělí na několik podoblastí, na kterých pak můžeme řešit příslušné podúlohy paralelně, a tedy si můžeme dovolit přesnější a dražší metody. Máme-li spočtené všechny části, provedeme korekturu a můžeme iterovat do dosažení požadované přesnosti. K tomu se přirozeně nabízí otázky odvození metody, detaily implementace a samotná konvergence a těmi se budeme zabývat.

Metoda parareal je v literatuře popsána na parciálních diferenciálních rovnicích (Maday a Turinici, 2002; Trindade a Pereira, 2004), dále také na obyčejných diferenciálních rovnicích (Amodio a Brugnano, 2009). V této práci budeme vycházet především z textu Gandra a Luneta (Gander a Lunet, 2024). Obsah práce je následující. V kapitole 1 provedeme formulaci úlohy v silné a následně slabé podobě. V kapitole 2 úlohu diskretizujeme pomocí metody konečných prvků a řekneme si, jak řešit příslušnou soustavu obyčejných rovnic i s aspektem adaptivní volby časového kroku. V kapitole 3 odvodíme schéma metody parareal a dokážeme vztah jejího řešení s příslušným sériovým řešičem. V kapitole 4 provedeme numerické experimenty. Používám implementaci metody parareal mého vedoucího práce, přičemž ostatní skripty pro analýzu chyb jsou mé vlastní.

1. Formulace úlohy

Pro ilustraci řešení parareal metody si jako testovací úlohu zvolíme nelineární rovnici vedení tepla. Chceme ji diskretizovat metodou konečných prvků (MKP), a proto přejdeme klasickým postupem k variační formulaci. Tu navíc upravíme do takového stavu, aby snáze řešila případnou nekonzistenci počátečních a okrajových podmínek. Při těchto úpravách zároveň zachováme symetrii úlohy. Budeme tak mít dvě definice slabého řešení. Nakonec se také podíváme na otázku existence řešení a jeho existence.

1.1 Prostory funkcí

Pro práci s funkcemi v následujících sekcích budeme potřebovat zavést několik značení a prostorů.

Nechť $a, b \in \mathbb{R}$, $a < b$, $\Omega = (a, b)$ a X značí *Banachův prostor* s normou $\|\cdot\|_X$. Symbolem X^* značíme *duální prostor* k X , tedy prostor všech spojitých lineárních funkcí na X s hodnotami v \mathbb{R} . Nechť dále $(\Omega, \mathcal{S}, \mu)$ je měřitelný prostor s mírou. Pak pro měřitelnou funkci $f : \Omega \rightarrow [0, \infty]$ budeme jejím *essenciálním supremem* myslet

$$\text{esssup } f = \inf\{\alpha \in \mathbb{R}; \mu(\Omega \cap f^{-1}(\alpha, \infty]) = 0\}.$$

Dále uvažujme prostory funkcí:

- $L^2(\Omega) = \{f : \Omega \rightarrow \mathbb{R} \text{ měřitelná funkce; } \|f\|_{L^2(\Omega)} < \infty\}$, kde

$$\|f\|_{L^2(\Omega)} = \left(\int_{\Omega} |f|^2 d\mu \right)^{1/2},$$

- $L^\infty(\Omega) = \{f : \Omega \rightarrow \mathbb{R} \text{ měřitelná funkce } \|f\|_{L^\infty(\Omega)} < \infty\}$ s normou

$$\|f\|_{L^\infty(\Omega)} = \text{esssup } |f|,$$

- *Sobolevův prostor* $H^1(\Omega)$ značí množinu všech funkcí z prostoru $L^2(\Omega)$, jejichž všechny parciální derivace 1. řádu jsou taktéž v $L^2(\Omega)$. Tyto derivace uvažujeme ve smyslu slabé derivace. Pro funkce $f \in H^1(\Omega)$ uvažujeme normu

$$\|f\|_{H^1(\Omega)} = \left(\|f\|_{L^2(\Omega)}^2 + \|\nabla f\|_{L^2(\Omega)}^2 \right)^{1/2},$$

kde ∇f značí gradient funkce f . Poznamenejme, že my se v práci budeme ve finále zabývat pouze reálnými funkcemi jedné proměnné, tudíž gradient bude splývat s derivací funkce,

- *Bochnerův prostor* $L^2(a, b; X) = \{f : (a, b) \rightarrow X \text{ silně měřitelná funkce; } \|f\|_{L^2(a, b; X)} < \infty\}$ s normou

$$\|f\|_{L^2(a, b; X)} = \left(\int_a^b \|f(t)\|_X^2 dt \right)^{1/2},$$

- *Bochnerův prostor* $L^\infty(a, b; X) = \{f : (a, b) \rightarrow X \text{ silně měřitelná funkce; } \|f\|_{L^\infty(a, b; X)} < \infty\}$ s normou

$$\|f\|_{L^\infty(a, b; X)} = \text{esssup } \|f\|_X.$$

Všechny výše uvedené prostory funkcí uvažujeme klasicky jako třídy ekvivalencí. Za tohoto předpokladu pak lze dokázat, že všechny zmíněné prostory jsou úplné vzhledem ke své normě (více detailů k definicím prostorů funkcí viz Bulíček a kol. (2018)).

1.2 Silná formulace

Uvažujme nelineární časově závislou rovnici vedení tepla v jedné prostorové dimenzi, kterou můžeme obecně vyjádřit jako

$$\frac{\partial u(x, t)}{\partial t} - \frac{\partial}{\partial x} \left(k(u) \frac{\partial u(x, t)}{\partial x} \right) = f(x, t) \quad x \in \Omega, t \in I, \quad (1.1)$$

kde

- $\Omega = (L, R) \subset \mathbb{R}$ reprezentuje prostorovou oblast,
- $I = (t_0, T) \subset \mathbb{R}$ značí časový interval,
- $u : \Omega \times I \rightarrow \mathbb{R}$ je neznámá funkce přiřazující dvojici (x, t) hodnotu $u(x, t)$,
- $k : \mathbb{R} \rightarrow \mathbb{R}$ je Lipschitzovsky spojitá funkce závislá na u , pro kterou existují konstanty k_0, k_1 takové, že $0 < k_0 \leq k(u) \leq k_1 \forall u \in \mathbb{R}$,
- $f : \Omega \times I \rightarrow \mathbb{R}$ odpovídá zdrojové funkci.

Tuto rovnici doplníme počáteční podmínkou

$$u(x, t_0) = u_{IC} \quad \forall x \in \Omega \quad (1.2)$$

a okrajovými podmínkami

$$u(L, t) = u_L, \quad u(R, t) = u_R \quad \forall t \in I. \quad (1.3)$$

Pod výše zmíněnou funkcí $k(u)$ splňující definující nerovnosti si můžeme pro $\alpha, \beta, \varepsilon > 0$ představit funkci (motivovanou úlohou proudění vody v porézním prostředí) tvaru:

$$k(u) = \begin{cases} \alpha + \varepsilon, & \text{je-li } u > 0, \\ \alpha \exp(\beta u) + \varepsilon, & \text{je-li } u \leq 0. \end{cases} \quad (1.4)$$

1.3 Slabá formulace

Nyní budeme přecházet k tzv. slabé formulaci úlohy, která je vhodná pro numerické metody, jako je MKP. Nejprve si však připomeňme, že v našem případě je hranice $\partial\Omega$ dvoubodová množina $\{L, R\}$, a tedy platí rovnost

$$\int_{\partial\Omega} g(x) dx = \lim_{x \rightarrow R_-} g(x) - \lim_{x \rightarrow L_+} g(x). \quad (1.5)$$

Také budeme potřebovat vnější jednotkovou normálu na hranici $\partial\Omega$, jenž má tvar

$$n(x) = \begin{cases} 1, & x = R, \\ -1, & x = L. \end{cases} \quad (1.6)$$

Nechť tedy $u \in C^2(\Omega \times I)$ splňuje rovnici (1.1) s počáteční podmínkou (1.2) a okrajovými podmínkami (1.3). Pak rovnici (1.1) vynásobíme testovací funkcí v z prostoru

$$V_0 = \{v \in H^1(\Omega); v(L) = v(R) = 0, \text{ ve smyslu stop}\} \quad (1.7)$$

a zintegrujeme přes oblast Ω :

$$\int_{\Omega} \frac{\partial u}{\partial t} v dx - \int_{\Omega} \frac{\partial}{\partial x} \left(k(u) \frac{\partial u}{\partial x} \right) v dx = \int_{\Omega} f v dx. \quad (1.8)$$

Druhý integrální člen se po aplikaci Greenovy věty rovná

$$- \int_{\partial\Omega} v k(u) \frac{\partial u}{\partial x} n dS + \int_{\Omega} k(u) \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} dx, \quad (1.9)$$

kde ovšem první člen díky definici prostoru V_0 přispívá pouze nulou. Dohromady potom dosazením do (1.8) získáváme

$$\int_{\Omega} \frac{\partial u}{\partial t} v dx + \int_{\Omega} k(u) \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} dx = \int_{\Omega} f v dx. \quad (1.10)$$

Pro definici slabého řešení (1.1) budeme potřebovat funkci $u^* \in H^1(\Omega)$, která navíc splňuje okrajové podmínky $u^*(L) = u_L$ a $u^*(R) = u_R$. V našem případě můžeme takovou funkci jednoduše zkonstruovat následujícím předpisem:

$$u^*(x) := u_L + (u_R - u_L) \frac{x - L}{R - L} \quad (1.11)$$

Poznamenejme, že ve složitějších případech, kdy předpis funkce u^* neznáme, chápeme okrajové podmínky ve smyslu stop.

Počáteční podmínku (1.2) realizujeme pomocí L^2 skalárního součinu:

$$\int_{\Omega} u(\cdot, t_0) v dx = \int_{\Omega} u_{IC} v dx. \quad (1.12)$$

Nyní máme vše potřebné pro následující definici řešení:

Definice 1. Řekneme, že funkce $u(x, t)$ je slabé řešení rovnice (1.1), pokud platí:

- (i) $u \in L^\infty(t_0, T; L^2(\Omega)) \cap L^2(t_0, T; H^1(\Omega))$ pro s.v. $t \in I$,
- (ii) $\frac{\partial u}{\partial t} \in L^2(t_0, T; (H^1(\Omega))^*)$ pro s.v. $t \in I$,
- (iii) $(u - u^*)(\cdot, t) \in V_0$ pro s.v. $t \in I$,
- (iv) u splňuje (1.10) $\forall v \in V_0, s.v. t \in I$,
- (v) u splňuje (1.12) $\forall v \in L^2(\Omega)$.

Budeme-li předpokládat, že $f \in L^2(\Omega \times I)$, pak lze dokázat (viz Bulíček a kol., 2018, Věta 7.2.3), že řešení dle Definice 1 existuje.

1.4 Slabá formulace s hraniční penalizací

V další kapitole se budeme zabývat diskretizací rovnice (1.10), nicméně okrajové podmínky budeme realizovat pomocí tzv. *hraniční penalizace*. Výhodou tohoto přístupu je snadná realizace nekonzistentních okrajových a počátečních podmínek. V Definici 1 pak nebudeme muset uvažovat podmínku (iii).

Uvažujme tedy v odvození slabé formulace testovací funkci v pouze z prostoru $H^1(\Omega)$. Potom se žádný člen po aplikaci Greenovy věty v (1.9) nevynuluje. K této rovnici přidáme na levou stranu člen, který představuje avizovanou penalizaci. Chceme však, aby byl roven nule pro řešení u splňující okrajové podmínky (1.3), a proto bude mít následující tvar:

$$\alpha \int_{\partial\Omega} (u - u^*)v \, dS, \quad \alpha > 0, \quad (1.13)$$

kde integrál $\int_{\partial\Omega}$ chápeme ve smyslu (1.5).

Dospěli jsme tím ke tvaru rovnice:

$$\int_{\Omega} \frac{\partial u}{\partial t} v \, dx + \int_{\Omega} k(u) \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \, dx - \int_{\partial\Omega} v k(u) \frac{\partial u}{\partial x} n \, dS + \alpha \int_{\partial\Omega} (u - u^*)v \, dS = \int_{\Omega} f v \, dx. \quad (1.14)$$

Třetí člen rovnice nám však nyní zapříčinil nesymetrii v prostorové části rovnice, což můžeme vyřešit přičtením dalšího členu k levé straně rovnice, který se pro u splňující okrajové podmínky (1.3) bude rovnat nule:

$$- \int_{\partial\Omega} k(u) \frac{\partial v}{\partial x} (u - u^*) n \, dS. \quad (1.15)$$

Shrnutím všech přídavek dostáváme následující rovnici:

$$\begin{aligned} & \int_{\Omega} \frac{\partial u}{\partial t} v \, dx + \int_{\Omega} k(u) \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \, dx - \int_{\partial\Omega} v k(u) \frac{\partial u}{\partial x} n \, dS \\ & - \int_{\partial\Omega} k(u) \frac{\partial v}{\partial x} (u - u^*) n \, dS + \alpha \int_{\partial\Omega} (u - u^*)v \, dS = \int_{\Omega} f v \, dx. \end{aligned} \quad (1.16)$$

Všechny uvedené modifikace nás vedou na novou definici slabého řešení:

Definice 2. Řekneme, že funkce $u(x, t)$ je slabé řešení rovnice (1.1), pokud platí:

- (i) $u \in L^\infty(t_0, T; L^2(\Omega)) \cap L^2(t_0, T; H^1(\Omega))$ pro s.v. $t \in I$,
- (ii) $\frac{\partial u}{\partial t} \in L^2(t_0, T; (H^1(\Omega))^*)$ pro s.v. $t \in I$,
- (iii) u splňuje (1.16) $\forall v \in H^1(\Omega)$, s.v. $t \in I$,
- (iv) u splňuje (1.12) $\forall v \in L^2(\Omega)$.

Existence řešení dle Definice 2 je mimo rozsah této bakalářské práce (více viz Babuška (1973)).

2. Numerické řešení

Slabé řešení chceme nalézt numericky, je tedy třeba přejít z nekonečné dimenze do konečné. Učiníme tak s pomocí MKP v prostoru čímž získáme systém ODR. Ten následně diskretizací v čase převedeme na soustavu algebraických rovnic, avšak z povahy původní úlohy bude nelineární a povede na iterativní proces. Odvodíme si také možnosti volby časového kroku.

2.1 Prostorová semidiskretizace úlohy pomocí MKP

Interval Ω rozdělíme na M uzlů $L = x_0 < x_1 < \dots < x_{M-1} = R$. Potom prostorové prvky tvoří množinu $\mathcal{T}_h = \{[x_{i-1}, x_i], \quad i = 1, \dots, M-1\}$, kde uvažujeme klasické $h = \max_{i=1, \dots, M-1} |x_{i-1}, x_i|$. Přecházíme tedy do prostoru s konečnou dimenzí a řešení hledáme v prostoru

$$V_h = \{v \in C(\Omega); v|_K \in P_1(K), \forall K \in \mathcal{T}_h\}. \quad (2.1)$$

Poznamenejme, že funkce u^* definovaná dříve v (1.11) je v prostoru V_h .

Analogicky jako pro slabé řešení dle Definice 2 chceme, aby přibližné řešení u_h splňovalo pro $\forall v_h \in V_h$ rovnici

$$\begin{aligned} & \int_{\Omega} \frac{\partial u_h}{\partial t} v_h dx + \int_{\Omega} k(u_h) \frac{\partial u_h}{\partial x} \frac{\partial v_h}{\partial x} dx - \int_{\partial\Omega} v_h k(u_h) \frac{\partial u_h}{\partial x} n dS \\ & - \int_{\partial\Omega} k(u_h) \frac{\partial v_h}{\partial x} (u_h - u^*) n dS + \alpha \int_{\partial\Omega} (u_h - u^*) v_h dS = \int_{\Omega} f v_h dx \end{aligned} \quad (2.2)$$

a aby v počátečním čase t_0 splňovalo analogicky jako v (1.12) počáteční podmínku

$$\int_{\Omega} u_h^0 v_h dx = \int_{\Omega} u_{IC} v_h dx \quad \forall v_h \in V_h. \quad (2.3)$$

Definice 3. Řekneme, že funkce $u_h(x, t)$ je přibližné semi-prostorové řešení rovnice (1.1), pokud platí:

- (i) $u_h(\cdot, t) \in V_h(\Omega)$,
- (ii) $\frac{\partial u_h}{\partial t}(\cdot, t) \in C(\bar{I})$,
- (iii) u_h splňuje (2.2) $\forall v \in V_h(\Omega), \forall t \in I$,
- (iv) $u_h(\cdot, t_0) = u_h^0(\cdot) \quad \forall t \in I$, pro u_h^0 z (2.3).

Uvažujme bázové funkce $\varphi_i(x), i = 0, \dots, M-1$, prostoru V_h s lokálním nosičem splňující $\varphi_j(x_i) = \delta_{ij} \quad \forall i, j = 0, \dots, M-1$, kde δ_{ij} je Kroneckerovo delta:

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \quad (2.4)$$

Potom přibližné řešení můžeme zapsat ve tvaru

$$u_h(x, t) = \sum_{j=0}^{M-1} u_j(t) \varphi_j(x) \quad (2.5)$$

a k získání přibližného řešení u_h nám stačí nalézt koeficienty $u_j(t)$.

Dosadíme-li do rovnice (2.2) za funkci v_h báze funkce φ_i , a řešení u_h vyjádříme tvarem (2.5), můžeme problém zformulovat maticově. Definujme si nejprve následující vektory a matice:

- vektor neznámých časově závislých koeficientů:

$$\mathbf{u}(t) = (u_0(t), u_1(t), \dots, u_{M-1}(t))^T, \quad t \in I,$$

- vektor odpovídající počáteční podmínce:

$$\mathbf{u}^0 = (u_0^0, u_1^0, \dots, u_{M-1}^0)^T,$$

- matice hmotnosti \mathbb{M} , která je dána jako:

$$M_{ij} = \int_{\Omega} \varphi_i \varphi_j dx, \quad i, j = 0, \dots, M-1,$$

- matice tuhosti $\mathbb{K}(\mathbf{u}(t))$, která je definována jako:

$$\begin{aligned} K_{ij}(\mathbf{u}(t)) &= \int_{\partial\Omega} \left(\alpha \varphi_i \varphi_j - k(u_h) n \left(\varphi_i \frac{\partial \varphi_i}{\partial x} + \frac{\partial \varphi_j}{\partial x} \varphi_j \right) \right) dS \\ &+ \int_{\Omega} k(u_h) \frac{\partial \varphi_i}{\partial x} \frac{\partial \varphi_j}{\partial x} dx, \quad i, j = 0, \dots, M-1, t \in I, \end{aligned}$$

- vektor pravé strany $\mathbf{f}(\mathbf{u}(t), t)$, který je definován jako:

$$\begin{aligned} f_i(t) &= \int_{\partial\Omega} \left(\alpha u^* \varphi_i - u^* k(u_h) n \frac{\partial \varphi_i}{\partial x} \right) dS \\ &+ \int_{\Omega} f \varphi_i dx, \quad i = 0, \dots, M-1, t \in I. \end{aligned}$$

Připomeňme, že s ohledem na (1.5) a (1.11) platí

$$\int_{\partial\Omega} u^* \varphi_i dS = u_R \varphi_i(R) - u_L \varphi_i(L) \quad \forall i = 0, \dots, M-1. \quad (2.6)$$

Hledání přibližného řešení se tedy redukuje na vyřešení soustavy ODR:

$$\mathbb{M} \frac{d\mathbf{u}(t)}{dt} + \mathbb{K}(\mathbf{u}(t)) \mathbf{u}(t) = \mathbf{f}(\mathbf{u}(t), t), \quad t \in I, \quad (2.7)$$

s počáteční podmínkou

$$\mathbf{u}(t_0) = \mathbf{u}^0. \quad (2.8)$$

Protože matici \mathbb{M} lze jednoduše invertovat, můžeme (2.7) zapsat ekvivalentním tvarem, který se nám bude později hodit:

$$\frac{d\mathbf{u}(t)}{dt} = \mathbb{M}^{-1} (\mathbf{f}(\mathbf{u}(t), t) - \mathbb{K}(\mathbf{u}(t)) \mathbf{u}(t)) =: \mathbf{g}(\mathbf{u}(t), t), \quad t \in I. \quad (2.9)$$

2.2 Numerická kvadratura

Integrály definující \mathbb{M} , \mathbb{K} a \mathbf{f} vyjádříme pomocí numerické kvadratury. Protože bázové funkce φ_i jsou po částech lineární, členy v integrandu M_{ij} obsahují nejvýše kvadratické polynomy, tedy kvadraturní metody druhého řádu by je řešily přesně. Naproti tomu K_{ij} a f_i už obsahují obecné funkce. Při implementaci jsme zvolili Gaussovy kvadraturní vzorce se třemi uzly, což je dostačující.

2.3 Časová diskretizace

Soustava ODR (2.7) patří mezi tzv. „stiff“ úlohy, a proto je rozumné použít k jejímu řešení implicitní metody. Dalším důvodem je v úvodu zmiňovaná metoda parareal, kde hrubý řešič typicky vyžaduje volbu dlouhého časového kroku. Zde použijeme implicitní Eulerovu metodu.

Rozdělme interval (t_0, T) na posloupnost N časových kroků $t_0 < t_1 < t_2 < \dots < \dots < t_{n-1} < t_n < \dots < t_N = T$. Označme $\mathbf{u}^n \in \mathbb{R}^M$, $n = 1, 2, \dots$ jako aproximaci řešení $\mathbf{u}(t_n)$ v těchto bodech, přičemž \mathbf{u}^0 odpovídá počáteční podmínce (2.8). Dále označíme časový krok $\tau_n = t_n - t_{n-1}$, $n = 1, 2, \dots$. Potom můžeme schéma zapsat jako:

$$\mathbb{M} \frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\tau_n} + \mathbb{K}(\mathbf{u}^n) \mathbf{u}^n = \mathbf{f}(\mathbf{u}^n, t_n). \quad (2.10)$$

Tato metoda je prvního řádu a hlavně je (lineárně) nepodmíněně stabilní. Nevýhodou této implicitní metody je ovšem nutnost vyřešit v každém časovém kroku nelineární soustavu rovnic.

2.4 Řešení nelineárních algebraických rovnic

Typickým přístupem pro řešení nelineární soustavy je Newtonova metoda. Ta však vyžaduje hladkost dat a naše funkce $k(u)$ v (1.4) nemá obecně spojitou první derivaci v $u = 0$. Použijeme proto jiný způsob linearizace, tzv. Kačanovu metodu (viz Kachanov, 1959).

Hodnoty \mathbf{u}^n , $n = 1, \dots, N$ ve schématu (2.10) budeme iterativně aproximovat hodnotami $\mathbf{u}^{n,l}$ pro $l = 1, 2, \dots$ následujícím způsobem:

$$\mathbb{M} \frac{\mathbf{u}^{n,l} - \mathbf{u}^{n,l-1}}{\tau_n} + \mathbb{K}(\mathbf{u}^{n,l-1}) \mathbf{u}^{n,l} = \mathbf{f}(\mathbf{u}^{n,l-1}, t_n), \quad (2.11)$$

ekvivalentně můžeme zapsat jako

$$\mathbb{A} \mathbf{u}^{n,l} := \left(\frac{1}{\tau_n} \mathbb{M} + \mathbb{K}(\mathbf{u}^{n,l-1}) \right) \mathbf{u}^{n,l} = \mathbf{f}(\mathbf{u}^{n,l-1}, t_n) + \frac{1}{\tau_n} \mathbb{M} \mathbf{u}^{n,l-1} =: \mathbf{b}. \quad (2.12)$$

Vztah (2.12) představuje soustavu lineárních algebraických rovnic, kterou řešíme přímo pomocí MUMPS (viz Amestoy a kol., 2001, 2019). Pro iniciaci použijeme $\mathbf{u}^{n,0} = \mathbf{u}^{n-1}$. Iterační proces opakujeme dokud neobdržíme při užití euklidovské normy výsledek

$$\frac{\|\mathbf{u}^{n,l} - \mathbf{u}^{n,l-1}\|}{\|\mathbf{u}^{n,l}\|} \leq 10^{-3}. \quad (2.13)$$

V takovém případě proces zastavíme a stanovíme $\mathbf{u}^n := \mathbf{u}^{n,l}$.

2.5 Adaptivní volba časového kroku

V následující části uvažujeme rovnici odvozenou dříve (2.9) ve tvaru

$$\frac{d\mathbf{u}(t)}{dt} = \mathbf{g}(\mathbf{u}(t), t), \quad t \in I. \quad (2.14)$$

Při diskretizaci časového intervalu nastává otázka, jaký zvolíme časový krok. Nej-jednodušší pro implementaci by bylo ekvidistantní, ale to není optimální, protože řešení naší úlohy se vyvíjí dynamicky. K přesnějšímu zachycení chování bychom mohli zvolit velmi krátké časové kroky τ_n , ale takové rozhodnutí by bylo nepřiměřeně drahé. S krátkými kroky bychom se na druhou stranu mohli výrazně vzdálit od přesného řešení.

Nabízí se tedy volba adaptivní, která používá navíc informace z aktuální časové vrstvy. Protože použitá Eulerova metoda je jednokroková, bude se nám výběr kroků lépe řídit. Její nepodmíněná stabilita nám navíc výrazně zmenšuje riziko nárůstu chyb při změnách časového kroku.

Jednou z možností odvození optimálního časového kroku může být srovnání chyby Eulerovy metody (E) s jiným implicitním schématem, které je druhého řádu, např. Crankovo-Nicolsonové (C). Chceme tedy zkoumat rozdíl předpisů

$$\begin{aligned} (E) \quad \mathbf{u}^n &= \mathbf{u}^{n-1} + \tau_n \mathbf{g}(\mathbf{u}^n, t_n), \\ (C) \quad \mathbf{u}_C^n &= \mathbf{u}^{n-1} + \frac{\tau_n}{2} \left(\mathbf{g}(\mathbf{u}^n, t_n) + \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1}) \right), \end{aligned} \quad (2.15)$$

kde $\mathbf{g}(\mathbf{u}^n, t_n)$ je značení odpovídající (2.9). Dostáváme tím odhad chyby

$$\begin{aligned} \text{EST}_1 &:= |\mathbf{u}^n - \mathbf{u}_C^n| = \frac{\tau_n}{2} |\mathbf{g}(\mathbf{u}^n, t_n) - \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1})|, \\ \Leftrightarrow \tau_n \text{EST}_1 &= \frac{\tau_n^2}{2} |\mathbf{g}(\mathbf{u}^n, t_n) - \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1})|. \end{aligned} \quad (2.16)$$

Podotkněme, že hodnota EST_1 je počítatelná. Nyní hledáme takový časový krok τ_{opt} , pro který bude platit $\text{EST}_1 \approx \text{TOL}_1$ nebo také $\tau_n \text{EST}_1 \approx \tau_n \text{TOL}_1$ pro předepsanou toleranci TOL_1 . Pomocí vztahů (2.16) odvodíme

$$\tau_{opt} = \sqrt{\frac{2 \tau_n \text{TOL}_1}{|\mathbf{g}(\mathbf{u}^n, t_n) - \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1})|}} = \tau_n \sqrt{\frac{\text{TOL}_1}{\text{EST}_1}}. \quad (2.17)$$

Provedeme-li výpočet na n -té hladině s krokem τ_n a spočtený odhad EST_1 je větší než TOL_1 , provedeme n -tou iteraci znovu s optimálně voleným krokem τ_{opt} . Pokud se rozhodneme pokračovat dále, jako časový krok τ_{n+1} vezmeme právě spočtený optimální krok τ_{opt} . Jedná se totiž o nejlepší aproximaci nadcházejícího optimálního kroku, kterou máme k dispozici. Abychom se v praxi vyvarovali opakování výpočtu při přesažení tolerance pouze o malé číslo, násobí se výsledný optimální časový krok bezpečnostní konstantou, tedy $\alpha \tau_{opt}$, $\alpha \in (0,1)$.

Další možností odvození je pomocí odhadu lokální chyby aproximace, která je u Eulerovy metody pomocí Taylorova rozvoje pro jisté $\delta \in (0, 1)$ tvaru

$$L_n = \frac{\tau_n^2}{2} \mathbf{u}''(t_{n-1} + \delta\tau_n) \quad n = 1, 2, \dots, N. \quad (2.18)$$

Označme symbolem $(\mathbf{u}^n)''$ aproximaci derivace $\mathbf{u}''(t_n)$. Druhou derivaci $(\mathbf{u}^n)''$ neznáme, ale můžeme ji aproximovat pomocí zpětné diference

$$(\mathbf{u}^n)'' \approx \frac{(\mathbf{u}^n)' - (\mathbf{u}^{n-1})'}{\tau_n} \approx \frac{\mathbf{g}(\mathbf{u}^n, t_n) - \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1})}{\tau_n}. \quad (2.19)$$

Definujeme-li

$$\text{EST}_2 := \frac{\tau_n}{2} |\mathbf{g}(\mathbf{u}^n, t_n) - \mathbf{g}(\mathbf{u}^{n-1}, t_{n-1})| \approx \frac{\tau_n^2}{2} |\mathbf{u}''(t_n)| \approx |L_n|, \quad (2.20)$$

pak pro obdržení $\text{EST}_2 \approx \text{TOL}_1$ chceme, aby platilo

$$\tau_{opt} = \sqrt{\frac{2 \text{TOL}_1}{|\mathbf{u}''(t_n)|}} = \tau_n \sqrt{\frac{\text{TOL}_1}{\text{EST}_2}}. \quad (2.21)$$

Vidíme, že jsme vzhledem k přibližným rovnostem dostali stejný výraz, jako při prvním postupu odvození, protože $\text{EST}_1 = \text{EST}_2$. To je díky provázanosti Eulerovy metody s metodou Crank-Nicolsonové, která je esenciálně pouze průměrem explicitní a implicitní Eulerovy metody.

Volbou implicitní Eulerovy metody máme zaručenu nepodmíněnou stabilitu řešení. To však neznamená, že je při praktických výpočtech nestabilní chování vyloučeno. Pokud by taková situace nastala, pak řešení začne oscilovat, čímž vzrostou také aproximované hodnoty druhé derivace $(\mathbf{u}^n)''$ a tím i EST_2 . Tedy při narušení podmínky stability se navrhaný optimální krok τ_{opt} sníží. Tímto způsobem je pak v adaptivní volbě časového kroku implicitně kontrolována stabilita. Více detailů k numerickému řešení ODR lze nalézt například v Hairer a kol. (1993)

V naší implementaci jsme užili jiný přístup volby adaptivního kroku založený na nespojitě Galerkinově metodě v čase (více viz Dolejší a Feistauer, 2015). Ten však přesahuje rámec této práce, a proto ho vynecháme.

3. Paralelní řešení

Odvozenou soustavu ODR (2.9) chceme řešit paralelně, k čemuž nám poslouží metoda parareal. V následujícím si přiblížíme její odvození metodou vícenásobné střelby, ukážeme si detailní popis algoritmu a prozkoumáme teoretické aspekty, především konvergenci.

3.1 Metoda vícenásobné střelby

Uvažujme soustavu ODR ve tvaru:

$$\frac{d\mathbf{u}(t)}{dt} = \mathbf{g}(\mathbf{u}(t), t), \quad t \in (t_0, T], \quad \mathbf{u}(t_0) = \mathbf{u}^0, \quad (3.1)$$

kde $\mathbf{u}(t) \in \mathbb{R}^M$ je vektorová funkce neznámých a $\mathbf{g}(\mathbf{u}(t), t) \in \mathbb{R}^M$ je vektorová funkce pravých stran. Rozdělme interval $(t_0, T]$ na N podintervalů pomocí dělení $\{t_n\}_{n=0}^N$ tak, že $t_0 < t_1 < \dots < t_N = T$. Známe-li přesné řešení \mathbf{u} , vyhodnotíme ho v právě vytvořených uzlech, čímž dostaneme sadu N parametrů $\mathbf{U}^n := \mathbf{u}(t_n) \in \mathbb{R}^M$, které použijeme jako počáteční podmínky. Můžeme tedy soustavu (3.1) ekvivalentně zapsat jako N na sebe navazujících počátečních úloh

$$\begin{aligned} \partial_t \mathbf{u}_n(t) &= \mathbf{g}(\mathbf{u}_n(t), t) \quad t \in (t_n, t_{n+1}], \quad n = 0, 1, \dots, N-1, \\ \mathbf{u}(t_n) &= \mathbf{U}_n \quad n = 0, 1, \dots, N-1, \end{aligned} \quad (3.2)$$

kde $\mathbf{u}_n(t) \in \mathbb{R}^M$ je vektorová funkce neznámých a ∂_t značí parciální derivace podle času $\frac{\partial}{\partial t}$. Všimněme si, že zde značíme parciální derivaci, protože řešení \mathbf{u}_n závisí nejen na čase, ale také parametrech \mathbf{U}_n , tedy $\mathbf{u}_n(t) = \mathbf{u}_n(t, \mathbf{U}_n)$.

Tím jsme původní soustavu sice rozdělili na N menších soustav, avšak k jejich vyřešení potřebujeme znát parametry \mathbf{U}_n , které ale vychází z přesného řešení \mathbf{u}_n . Zkusme tedy tyto parametry chytře aproximovat tak, abychom získali paralelnost úlohy. Protože původní řešení $\mathbf{u}(t)$ je spojitě pro $t \in (t_0, T]$, musí pro parametry \mathbf{U}_n platit následující soustava rovnic:

$$\begin{aligned} \mathbf{U}_0 &= \mathbf{u}_0(t_0) = \mathbf{u}^0, \\ \mathbf{U}_1 &= \mathbf{u}_0(t_1) = \mathbf{u}_0(t_1, \mathbf{U}_0), \\ \mathbf{U}_2 &= \mathbf{u}_1(t_2) = \mathbf{u}_1(t_2, \mathbf{U}_1), \\ &\vdots \\ \mathbf{U}_N &= \mathbf{u}_{N-1}(t_N) = \mathbf{u}_{N-1}(t_N, \mathbf{U}_{N-1}). \end{aligned} \quad (3.3)$$

Zavedením vektoru $\mathbf{U} := (\mathbf{U}_0, \mathbf{U}_1, \dots, \mathbf{U}_N)^T \in \mathbb{R}^{(N+1)M}$ a odečtením pravých stran soustavy (3.3) od levých dostáváme následující nelineární problém

$$\mathbf{F}(\mathbf{U}) := \begin{pmatrix} \mathbf{U}_0 - \mathbf{u}^0 \\ \mathbf{U}_1 - \mathbf{u}_0(t_1, \mathbf{U}_0) \\ \mathbf{U}_2 - \mathbf{u}_1(t_2, \mathbf{U}_1) \\ \vdots \\ \mathbf{U}_N - \mathbf{u}_{N-1}(t_N, \mathbf{U}_{N-1}) \end{pmatrix} = 0. \quad (3.4)$$

Na tuto nelineární soustavu použijeme Newtonovu iterační metodu, tedy pro počáteční odhad \mathbb{U}^0 hledáme

$$\mathbb{U}^{k+1} = \mathbb{U}^k - (\mathbf{F}'(\mathbb{U}^k))^{-1} \mathbf{F}(\mathbb{U}^k), \quad k = 0, 1, 2, \dots, \quad (3.5)$$

kde $\mathbf{F}'(\mathbb{U}^k)$ značí Jacobiho matici funkce \mathbf{F} v bodě \mathbb{U}^k , která je rovna

$$\mathbf{F}'(\mathbb{U}^k) = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ -\frac{\partial \mathbf{u}_0}{\partial \mathbf{U}_0}(t_1, \mathbf{U}_0^k) & 1 & \cdots & 0 & 0 \\ 0 & -\frac{\partial \mathbf{u}_1}{\partial \mathbf{U}_1}(t_2, \mathbf{U}_1^k) & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{\partial \mathbf{u}_{N-1}}{\partial \mathbf{U}_{N-1}}(t_N, \mathbf{U}_{N-1}^k) & 1 \end{pmatrix}. \quad (3.6)$$

Díky jednoduchému tvaru matice $\mathbf{F}'(\mathbb{U}^k)$ můžeme jednoduše zkonstruovat její inverzi, my však místo toho provedeme následující úpravy. Odečteme-li od rovnice (3.5) vektor \mathbb{U}^k a vynásobíme-li poté celou rovnici zleva maticí $\mathbf{F}'(\mathbb{U}^k)$, dostaneme ekvivalentní rovnici

$$(\mathbf{F}'(\mathbb{U}^k))(\mathbb{U}^{k+1} - \mathbb{U}^k) = -\mathbf{F}(\mathbb{U}^k), \quad k = 0, 1, 2, \dots. \quad (3.7)$$

Vyjádríme-li nyní řádky předchozí rovnosti, dostaneme po drobné úpravě pro $k = 0, 1, 2, \dots$ tvar metody vícenásobné střelby

$$\begin{aligned} \mathbb{U}_0^{k+1} &= \mathbf{u}^0, \\ \mathbb{U}_{n+1}^{k+1} &= \mathbf{u}_n(t_{n+1}, \mathbf{U}_n^k) + \frac{\partial \mathbf{u}_n}{\partial \mathbf{U}_n}(t_{n+1}, \mathbf{U}_n^k) (\mathbf{U}_n^{k+1} - \mathbf{U}_n^k), \quad n = 0, 1, \dots, N-1. \end{aligned} \quad (3.8)$$

Vyvozené schéma nám umožňuje řešit zprvu čistě sekvenční úlohu paralelně. Zbývá pouze aproximovat operátory tak, aby šly v praxi lépe použít.

3.2 Metoda parareal

Vycházíme ze schématu metody vícenásobné střelby (3.8). Chceme nahradit derivaci podle parametru jiným výrazem, k čemuž nám pomůže Taylorův rozvoj. Necht' jsou $n = 0, 1, \dots, N$ a $k = 0, 1, 2, \dots$ zvolena pevně a řešení jsou dostatečně hladká. Potom rozvíjíme přesné řešení $\mathbf{u}_n(t_{n+1}, \mathbf{U}_{n+1}^k)$ v bodě \mathbf{U}_n^k

$$\mathbf{u}_n(t_{n+1}, \mathbf{U}_{n+1}^k) = \mathbf{u}_n(t_{n+1}, \mathbf{U}_n^k) + \frac{\partial \mathbf{u}_n}{\partial \mathbf{U}_n}(t_{n+1}, \mathbf{U}_n^k) (\mathbf{U}_n^{k+1} - \mathbf{U}_n^k) + O(\|\mathbf{U}_n^{k+1} - \mathbf{U}_n^k\|^2). \quad (3.9)$$

Zanedbáním zbytku druhého řádu tedy můžeme člen s parciální derivací aproximovat jako

$$\frac{\partial \mathbf{u}_n}{\partial \mathbf{U}_n}(t_{n+1}, \mathbf{U}_n^k) (\mathbf{U}_n^{k+1} - \mathbf{U}_n^k) \approx \mathbf{u}_n(t_{n+1}, \mathbf{U}_{n+1}^k) - \mathbf{u}_n(t_{n+1}, \mathbf{U}_n^k). \quad (3.10)$$

Dosazení do (3.8) nás poté vede k definici přibližné metody vícenásobné střelby jako

$$\mathbf{U}_{n+1}^{k+1} \approx \mathbf{u}_n(t_{n+1}, \mathbf{U}_n^k) + \mathbf{u}_n(t_{n+1}, \mathbf{U}_{n+1}^k) - \mathbf{u}_n(t_{n+1}, \mathbf{U}_n^k). \quad (3.11)$$

Toto vyjádření však nedává úplný smysl, protože první a třetí člen pravé strany se vzájemně odečítají. Pakliže nebudeme uvažovat přesná, nýbrž přibližná řešení, žádný člen se prakticky neodečte a my tím získáme schéma, které už lze paralelizovat. Konkrétně pro první člen v (3.11) použijeme jemný řešič a pro zbylé dva hrubý (viz následující (3.12)).

Vraťme se nyní k řešení problému (3.1). Řekněme, že chceme proces konstruovat pro N procesorů. Rozložíme časový interval $(t_0, T]$ na několik podintervalů $(t_n, t_{n+1}]$ s krokem $\tau = \frac{T-t_0}{N}$, kde $t_0 < t_1 < \dots < t_N = T$. Přesné hodnoty řešení v uzlech t_i aproximujeme $\mathbf{u}(t_n) \approx U_n \in \mathbb{R}^M$ pro $n = 0, \dots, N$.

Hlavní signaturou metody parareal jsou dva propagační operátory:

- *hrubá aproximace* $G(t_{n+1}, t_n, U_n)$ řešení v čase t_{n+1} vycházející z času t_n a s počáteční podmínkou U_n ,
- *jemná aproximace* $F(t_{n+1}, t_n, U_n)$ řešení v čase t_{n+1} vycházející z času t_n a s počáteční podmínkou U_n .

Potom schéma *metody parareal* v daném cyklu $k = 0, 1, \dots$ a jemu odpovídající hodnotě U_n^k lze zapsat následujícím způsobem:

1. **Iniciace:** Výpočet prvního odhadu hrubou metodou $U_{n+1}^0 = G(t_{n+1}, t_n, U_n^0)$ s počáteční podmínkou $U_n^0 = \mathbf{u}^0$ pro $n = 0, \dots, N$. Je nutné počítat *sériově*.

iterace $k = 0, 1, \dots$

2. **Jemný výpočet:** Jemné řešení $F(t_{n+1}, t_n, U_n^k)$ z uzlů získaných v předchozí iteraci pro $n = 0, \dots, N - 1$. Tuto část lze provést *paralelně*.
3. **Korekce:** Oprava hrubých uzlů pro další iteraci užitím (3.11):

$$U_{n+1}^{k+1} = F(t_{n+1}, t_n, U_n^k) + G(t_{n+1}, t_n, U_n^{k+1}) - G(t_{n+1}, t_n, U_n^k). \quad (3.12)$$

Zde je nutno opět vést výpočet *sériově*.

4. **Zastavovací kritérium:** Přejít do dalšího cyklu, pokud nejsou nárůsty hodnot menší než předepsaná tolerance, ergo $\|U_n^{k+1} - U_n^k\| < \text{TOL}_2$, $n = 0, \dots, N$.

Podotkněme, že korekční krok algoritmu odpovídá dříve zmíněné aproximaci předpisu (3.11). Metoda parareal je tedy ve své podstatě metoda vícenásobné střelby se speciální aproximací Jacobiho matice z (3.5).

Máme-li nyní k dispozici paralelní metodu, kterou jsme schopni bez problému implementovat, je na místě se ptát, jaké má vlastnosti. Konkrétně jak je to s konvergencí metody. Protože uvažovaná počáteční podmínka je neměnná, dostaneme v každé k iteraci na prvním podintervalu $(t_0, t_1]$ stejnou hodnotu, přesněji $F(t_1, t_0, \mathbf{u}^0)$. To je ale to samé, jako bychom na $(t_0, t_1]$ řešili pouze jemným řešičem. Na podintervalu $(t_1, t_2]$ pak pro $k > 0$ budeme vycházet pokaždé ze stejné počáteční podmínky $F(t_1, t_0, \mathbf{u}^0)$, tedy jsme opět ve stejné situaci. Toto pozorování nás potom vede na následující větu:

Věta 1. *Pro metodu parareal platí*

$$U_n^k = F(t_n, t_0, \mathbf{u}^0) \quad k \geq n, \quad n = 0, \dots, N.$$

Důkaz. Budeme postupovat matematickou indukcí podle n .
Pro $n = 0$ triviálně dostáváme

$$U_0^k = \mathbf{u}^0 = F(t_0, t_0, \mathbf{u}^0).$$

Předpokládejme nyní, že tvrzení platí pro n a necht' $k \geq n + 1$. Pak z indukčního předpokladu pro $k \geq n$ a pro $k - 1 \geq n$ dostáváme $U_n^k = U_n^{k-1} = F(t_n, t_0, \mathbf{u}^0)$. To můžeme následně využít v rozepisu korekčního kroku

$$\begin{aligned} U_{n+1}^k &= F(t_{n+1}, t_n, U_n^{k-1}) + G(t_{n+1}, t_n, U_n^k) - G(t_{n+1}, t_n, U_n^{k-1}) \\ &= F(t_{n+1}, t_n, F(t_n, t_0, \mathbf{u}^0)) + G(t_{n+1}, t_n, F(t_n, t_0, \mathbf{u}^0)) \\ &\quad - G(t_{n+1}, t_n, F(t_n, t_0, \mathbf{u}^0)) \\ &= F(t_{n+1}, t_n, F(t_n, t_0, \mathbf{u}^0)) \\ &= F(t_{n+1}, t_0, \mathbf{u}^0), \end{aligned}$$

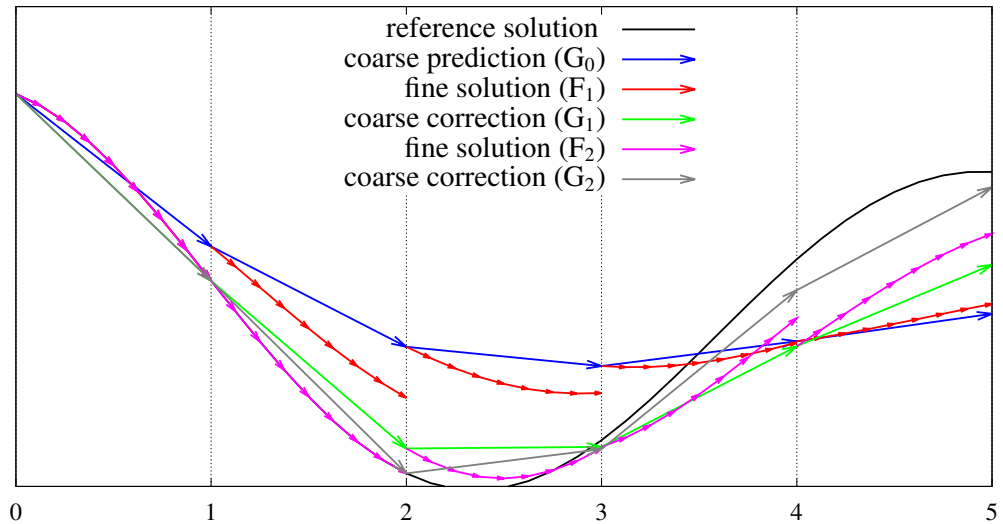
což bylo dokázati. □

Uvedená věta nám tedy říká, že po iteraci N zkonvergujeme k řešení daném jemným řešičem F . Použití pararealu v praxi má však smysl, pokud $k \ll N$, protože jinak bychom mohli úlohu řešit už od začátku sekvenčně pouze jemným řešičem. Na tento aspekt se podíváme později při numerických experimentech. Lze také dokázat jiná tvrzení o metodě (viz Gander a Lunet, 2024), my si zde však postačíme s Větou 1.

Princip metody parareal si graficky znázorníme na jednoduché úloze. Uvažujme obyčejnou stacionární diferenciální rovnici v jedné dimenzi

$$\frac{du(x)}{dx} = g(u(x)), \quad x \in (0, 5], \quad u(0) = u^0, \quad (3.13)$$

která vede na řešení $u(x)$ dané sinusoidou. Volbou $N = 5$ tedy rozdělíme interval na 5 podintervalů. Provedeme-li výpočet metodou parareal s užitím pevného kroku, máme Větou 1 zaručenu konvergenci v páté iteraci. Podrobné srovnání referenčního řešení spočteného jemným řešičem a řešení metody parareal pro první 2 iterace zachycuje Obrázek 3.1. Přirozeně vidíme, že po první iteraci jsou obě řešení na prvním podintervalu totožná. V souladu s Větou 1 se pak také po druhé parareal iteraci korekční řešení v bodě 2 shoduje s referenčním. Toto chování by pokračovalo postupně až do finální páté iterace.



Obrázek 3.1: Princip metody parareal na úloze (3.13).

3.3 Praktická realizace

Při samotném výpočtu je třeba zvolit za $G(\cdot, \cdot, \cdot)$ a $F(\cdot, \cdot, \cdot)$ vhodné numerické metody pro výpočet ODR. Hrubý řešič chceme volit levný, protože ho budeme používat sériově. Zároveň však k němu chceme mít dostatečnou důvěru, protože jeho vypočtené hodnoty používáme jako počáteční podmínku pro všechna částečná řešení. Jinými slovy chceme, aby měl velkou oblast stability, ideálně nezávislou na zvoleném časovém kroku. Jak už bylo zmíněno dříve, v této práci bude naší volbou implicitní Eulerova metoda, která je relativně levná a je nepodmíněně stabilní.

Jemný řešič naopak můžeme vybírat dražší, neboť ho budeme používat paralelně. S vyšší přesností metody budeme mít také přesnější výsledné řešení. V našem případě je však kvůli stiff vlastnosti úlohy za jemný řešič volena implicitní metoda, konkrétně opět implicitní Euler, ale s výrazně kratším časovým krokem, přesněji s menší tolerancí TOL_1 uvažovanou pro volbu adaptivního kroku.

4. Numerické experimenty

V poslední části práce si ukážeme chování metody parareal při praktickém počítání na úloze (1.1), konkrétněji

$$\frac{\partial u(x,t)}{\partial t} - \frac{\partial}{\partial x} \left(k(u) \frac{\partial u(x,t)}{\partial x} \right) = 0 \quad x \in \Omega = (0,1), t \in I = (0,80), \quad (4.1)$$

s počáteční podmínkou

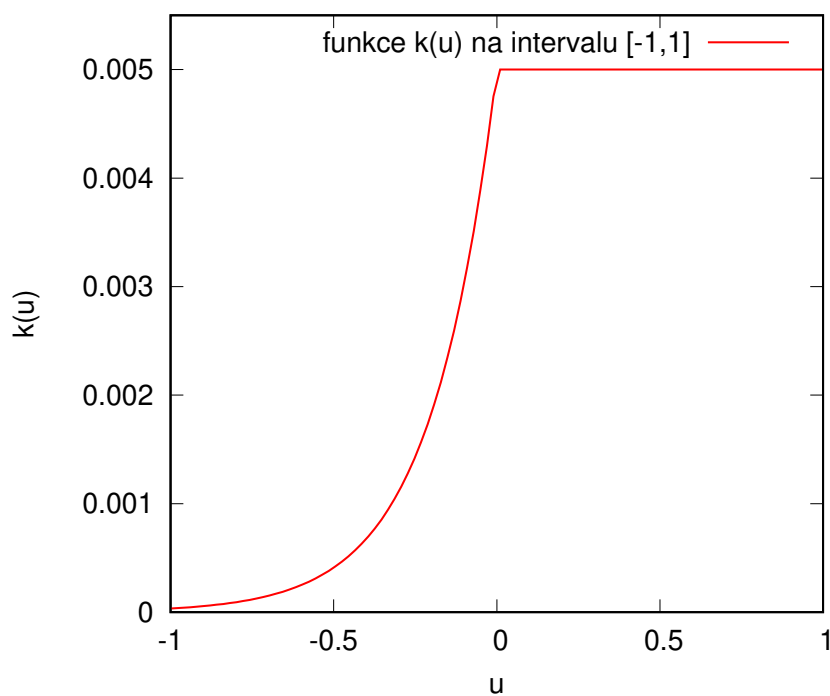
$$u(x,0) = -1 \quad \forall x \in (0,1), \quad (4.2)$$

okrajovými podmínkami

$$u(0,t) = 1, \quad u(1,t) = -1 \quad \forall t \in (0,80) \quad (4.3)$$

a funkcí

$$k(u) = \begin{cases} 5 \cdot 10^{-3}, & \text{je-li } u > 0, \\ 5 \cdot 10^{-3} \cdot \exp(5u), & \text{je-li } u \leq 0. \end{cases} \quad (4.4)$$



Obrázek 4.1: Funkce $k(u)$ na intervalu $[-1, 1]$.

Poznamenejme, že tato funkce $k(u)$ nespĺňuje původní podmínku odražení od nuly na celé reálné ose. Protože však řešení u naší úlohy bude mít omezený obor hodnot, je tato volba rozumná.

Všimněme si také, že v takto definované úloze dochází k nekonzistenci mezi počáteční a okrajovou podmínkou. Podmínky jsou vynucovány již zmíněným přidáním členem hraniční penalizace (viz (1.13)). Řešení tedy hledáme podle Definiční 2.

Implementace byla provedena na linuxovém prostředí v jazyce Fortran a k vizualizaci výsledků používáme software gnuplot. Následující experimenty se zaměřují na dvě hlavní oblasti:

- porovnání řešení spočteného metodou parareal s referenčním řešením,
- teoretické porovnání časové náročnosti sériového a paralelního výpočtu.

Referenčním řešením budeme v textu chápat řešení spočtené čistě sériovým jemným řešičem. Každé paralelně spočtené řešení má tedy příslušné referenční se stejnými parametry. Volbu takového srovnání motivuje Věta 1.

Na vstupu kódu můžeme modifikovat následující parametry:

- $M = \frac{1}{h}$ odpovídá počtu vzešlých ODR, kde h je krok prostorové sítě,
- N obsahuje počet oblastí z dělení časového intervalu,
- TOL_1 udává počáteční časový krok jemného řešiče a předepsanou toleranci pro jeho následnou adaptivní volbu.

Protože příslušný systém ODR chceme řešit paralelně, je vhodné časový interval $[0, 80]$ rozdělit rovnoměrně tak, aby na každém podintervalu proběhlo přibližně stejné množství časových kroků. Před zahájením hlavního výpočtu tedy nejprve jednou hrubě sériově předpočítáme řešení adaptivně s tolerancí $25 \cdot TOL_1$. Obdržíme tak jistou posloupnost časových uzlů $\{\tilde{t}_j\}_{j=0}^{nstep}$, kde $nstep$ je hromadný počet provedených časových kroků. Nyní chceme vytvořit menší posloupnost $\{t_n\}_{n=0}^N$, kterou následně použijeme k hlavnímu výpočtu. Okrajové body nastavíme na okrajové body řešeného intervalu, tedy $t_0 = 0$ a $t_n = 80$. Vnitřní body určíme následujícím způsobem pro $n = 1, \dots, N - 1$:

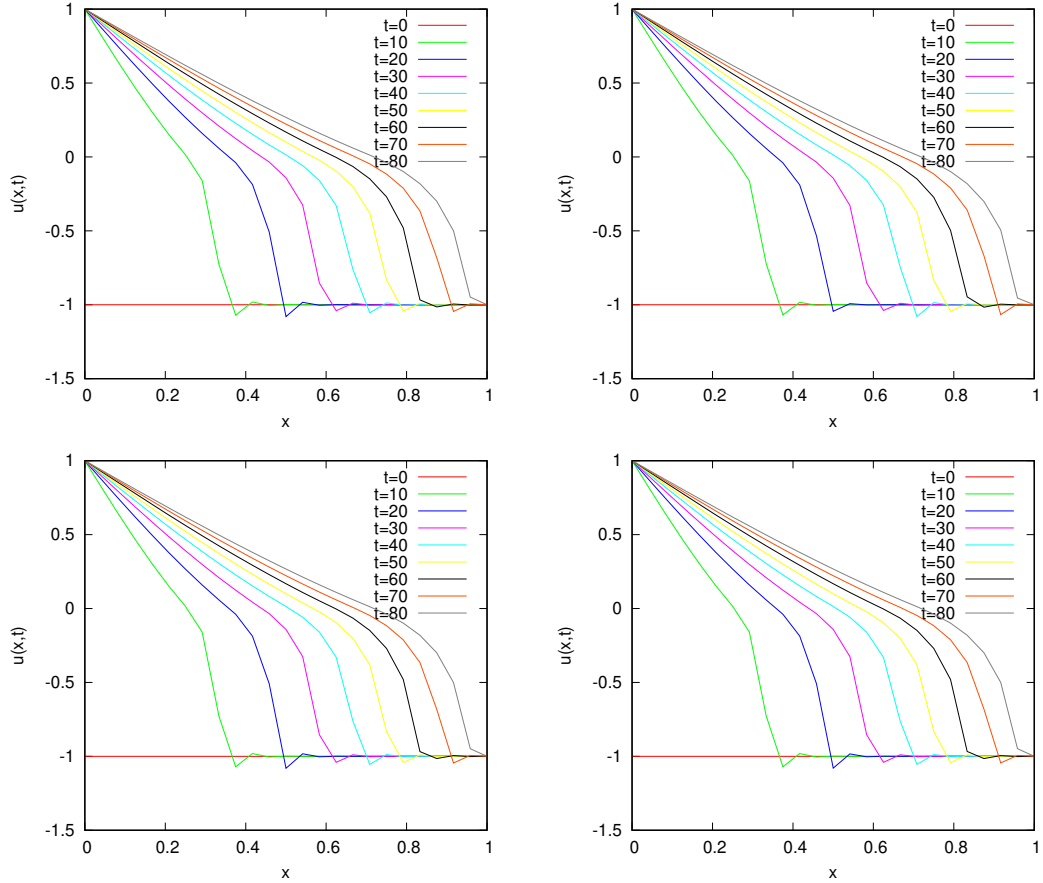
$$j_{rat} = \frac{n}{N} \cdot nsteps - \text{obecně neceločíselné,}$$

$$j = \lfloor j_{rat} \rfloor \in \mathbb{N}, \text{ značí dolní celou část čísla } j_{rat},$$

$$j_{rest} = (j_{rat} - j) \in (0, 1), \text{ neceločíselná část indexu } j,$$

$$t_n = \tilde{t}_j + j_{rest} \cdot (\tilde{t}_{j+1} - \tilde{t}_j) - \text{příslušný časový krok s korekcí.}$$

Abychom mohli jednodušeji porovnávat různá řešení, především paralelní se sériovými, budeme výsledky vypisovat v pevně specifikovaných časových hladinách $0, 10, \dots, 80$ a pro každou z nich pak M hodnot příslušných prostorové diskretizaci. Takové hodnoty budou pravděpodobně neshodné s dělením z předchozího odstavce, a proto využijeme interpolaci pomocí vedlejších hodnot.

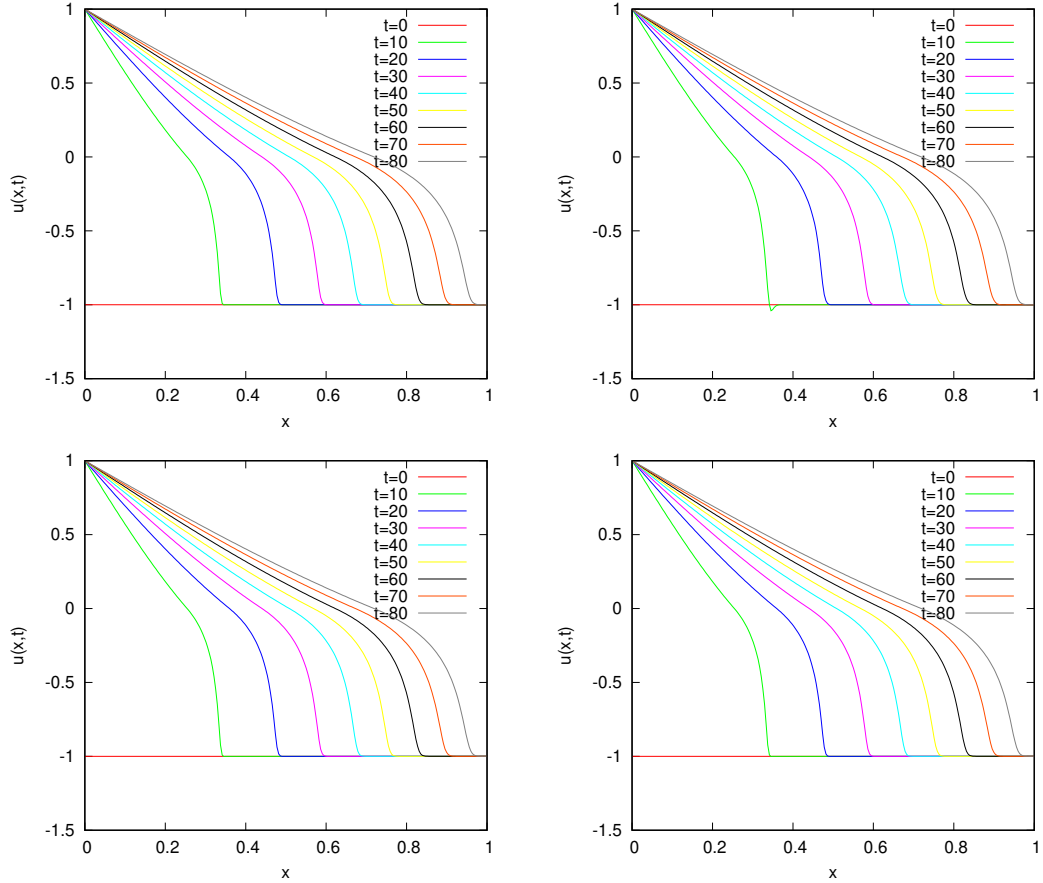


Obrázek 4.2: Porovnání řešení s parametry $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$ referenčního řešení (vlevo nahoře), parareal iterace 0 (vpravo nahoře), parareal iterace 1 (vlevo dole) a parareal iterace 2 (vpravo dole).

4.1 Konvergence metody parareal

V předchozí sekci jsme si dokázali, že parareal má zaručenou konvergenci k referenčnímu řešení po N iteracích. Prakticky bychom však chtěli, aby tolerovaná konvergence proběhla mnohem dříve. Tuto odchylku budeme tedy vždy porovnávat v poslední časové vrstvě $t = 80$, protože je nejdále od počátečního času. Pro kompaktnější zápis si zavedeme v této časové vrstvě značení chyby $\tilde{e}_i = \bar{u}_i - u_i^k, i = 0, \dots, M - 1$, kde \bar{u}_i je hodnota referenčního řešení v prostorovém uzlu x_i a u_i^k je spočtené řešení pararealu v iteraci k v prostorovém uzlu x_i . Pro naše experimenty jsme pro zastavení přechodu do další iterace pararealu volili měření euklidovskou normou a toleranci jsme stanovili na 10^{-5} , tedy výpočet se ukončí, pokud nastane $\| \mathbf{u}^{k+1} - \mathbf{u}^k \| \leq 10^{-5}$, kde $\mathbf{u}^k = (u_0^k, \dots, u_{M-1}^k)$.

Podívejme se na porovnání grafů při volbě parametrů $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$. Očekáváme tedy, že k referenčnímu řešení zkonvergujeme nejvýše po 5 iteracích. Už na takto malé úloze vidíme pozitivní jev, který se konzistentně opakuje i pro jiné volby parametrů a to konkrétně blízká konvergence už v prvních dvou iteracích (viz Obrázek 4.2). Podotkněme, že vizuální nehladkost tohoto řešení je způsobena čistě malou prostorovou diskretizací (viz srovnání s $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$ na Obrázku 4.3).



Obrázek 4.3: Porovnání řešení s parametry $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$ referenčního řešení (vlevo nahoře), parareal iterace 0 (vpravo nahoře), parareal iterace 1 (vlevo dole) a parareal iterace 2 (vpravo dole).

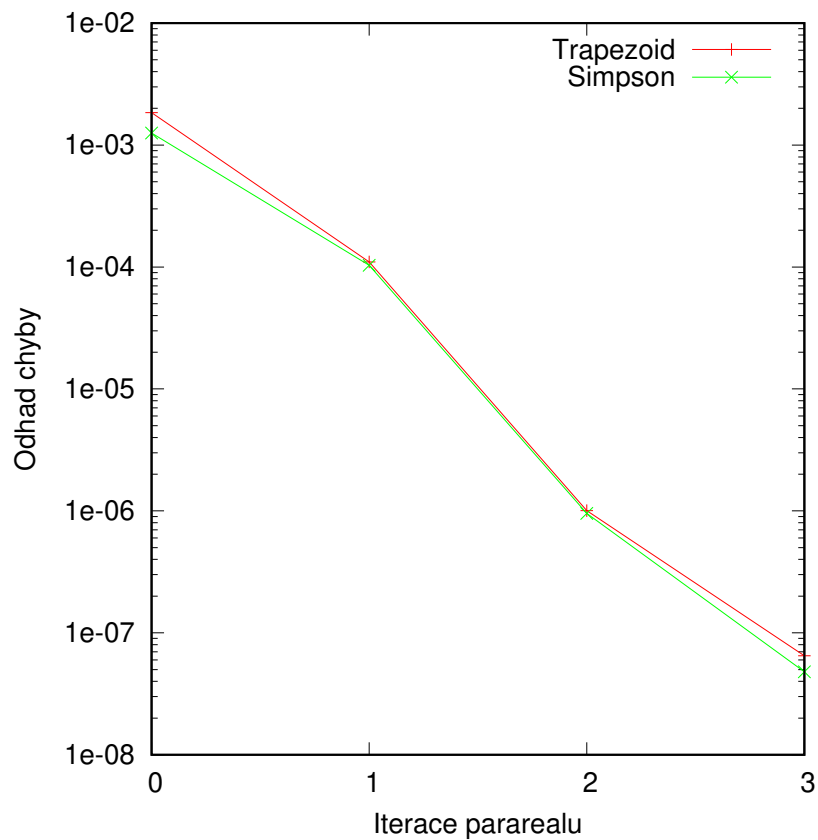
Nemá velký smysl výsledky dále porovnávat pouhým pohledem na vizualizace řešení samotných, a proto si právě zkoumanou chybu ve smyslu L^2 normy pomocí pravidel Trapezoid a Simpson. Odhady chyb poté budou mít následující tvar:

$$e_{trapez}(k) = \sqrt{\sum_{i=1}^{M-1} \frac{(x_i - x_{i-1})}{2} (\tilde{e}_i^2 + \tilde{e}_{i-1}^2)}, \quad (4.5)$$

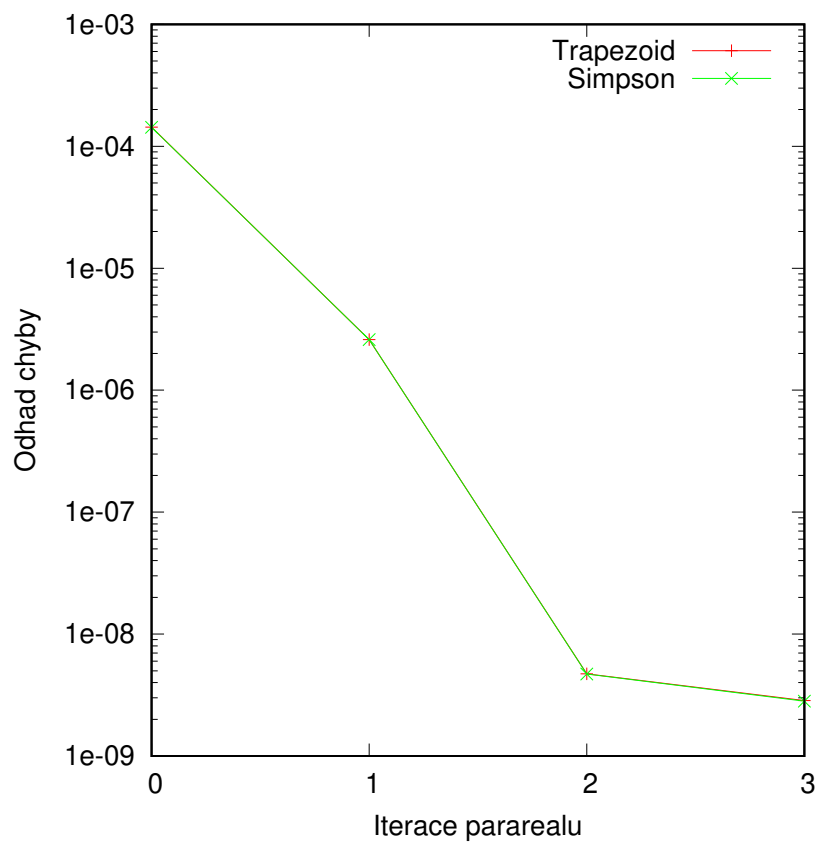
$$e_{simps}(k) = \sqrt{\sum_{i=1}^{M-1} \frac{(x_i - x_{i-1})}{6} \left(\tilde{e}_i^2 + 4 \left(\frac{\tilde{e}_i + \tilde{e}_{i-1}^k}{2} \right)^2 + \tilde{e}_{i-1}^2 \right)}. \quad (4.6)$$

Tyto aproximace řešení v časové hladině $t = 80$ zobrazíme v závislosti na iteracích k od 0 až do poslední spočtené iterace před splněním tolerance (Obrázek 4.4).

Abychom dosáhli nižšího řádu chyby, je třeba zvolit jemnější prostorovou diskretizaci. Spočteme-li například chybu řešení s parametry $M = 500, N = 10, \text{TOL}_1 = 10^{-2}$ (Obrázek 4.5), vidíme, že se dokážeme dostat pod úroveň 10^{-8} .

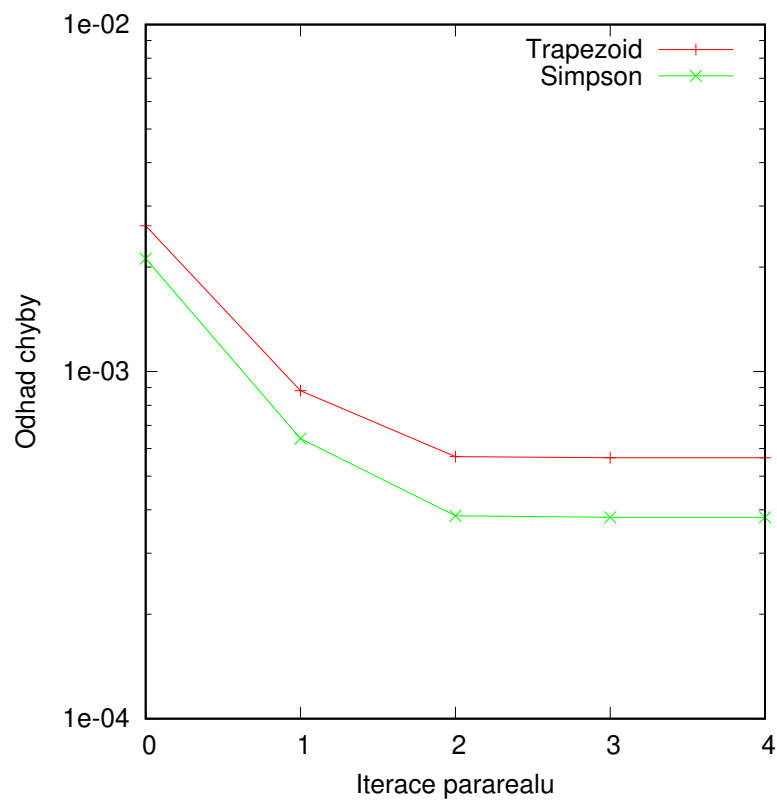


Obrázek 4.4: Konvergence odhadu chyby pro $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$.

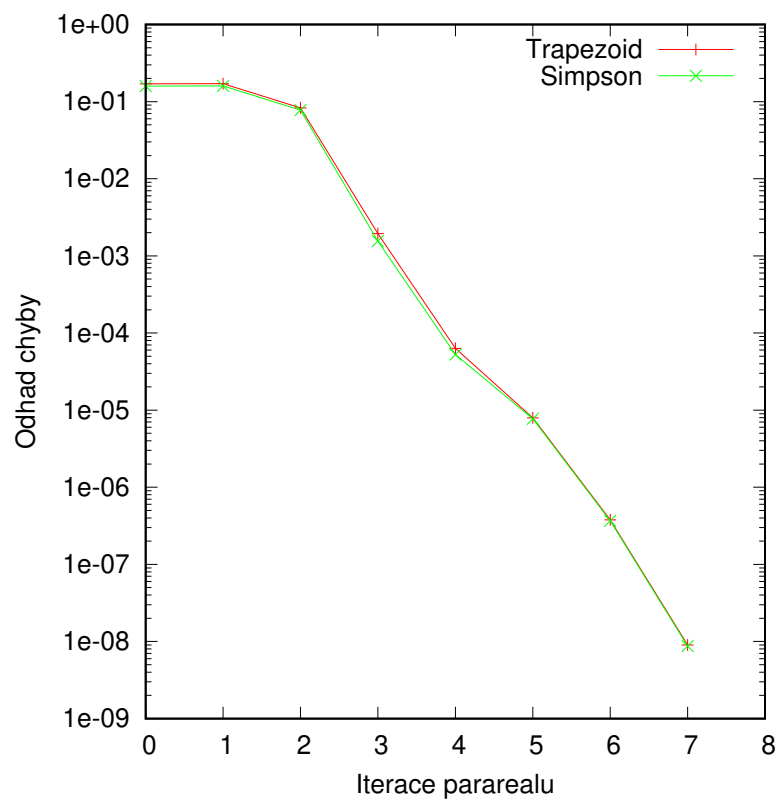


Obrázek 4.5: Konvergence odhadu chyby pro $M = 500, N = 10, \text{TOL}_1 = 10^{-2}$.

Velkou roli při výpočtu hraje adaptivní volba časového kroku, která používá informace z předchozího výpočtu. Srovnajme nyní řešení pararealu například s parametry $M = 25$, $N = 8$, $\text{TOL}_1 = 5 \cdot 10^{-1}$ s adaptivním krokem (Obrázek 4.6) a poté stejné řešení, ale s pevným časovým krokem (Obrázek 4.7). Řešení s malým pevným časovým krokem $5 \cdot 10^{-1}$ dokonverguje v předposlední iteraci k referenčnímu řešení na úrovni 10^{-9} . Na tomto konkrétním případě je vidět, jak se přesnosti řádově zvyšuje s každou iterací, za čímž stojí postupná konvergence jemných řešičů na podoblastech tak, jak popisuje Věta 1. Naopak u řešení s adaptivním krokem nám Věta 1 žádnou garanci konvergence k referenčnímu řešení nedává a v tomto případě se pohybujeme pouze kolem úrovně 10^{-3} . Do této chyby se může podepsat také zmíněná chyba interpolace v časovém uzlu. I přesto však toto adaptivní řešení dosahuje pozoruhodných výsledků. Již v nulté iteraci dosahuje chyby stejného řádu, jako řešení s pevným krokem po iteraci třetí.



Obrázek 4.6: Konvergence odhadu chyby pro $M = 25, N = 8, \text{TOL}_1 = 5 \cdot 10^{-1}$ s adaptivním krokem.



Obrázek 4.7: Konvergence odhadu chyby pro $M = 25, N = 8$ s pevným krokem $\text{TOL}_1 = 5 \cdot 10^{-1}$.

4.2 Teoretické zrychlení paralelizace

V této sekci se budeme teoreticky zabývat otázkou časového zrychlení metody parareal na úloze (4.1), konkrétně z pohledu provedených časových kroků. Hlubší analýza obsahující všechny závislosti a komunikaci mezi procesory by byla mimo rozsah této práce. Pozorování časových kroků volíme proto, že právě v této dimenzi provádíme paralelizaci.

Uvažujme situaci, kdy máme stejný počet procesorů, jako je počet časových oblastí N . Sériový referenční řešič projde celý interval $[0, 80]$ na jeden průchod s adaptivní volbou časového kroku a za celý výpočet provede celkem T_S časových kroků. Naproti tomu paralelní řešič projde nejprve první iterací $k = 0$ hrubým řešičem (coarse) a na každé oblasti $n = 1, \dots, N$ provede nějaký počet kroků $steps_n$. Následuje iterační proces $k = 1, 2, \dots$, kdy každý procesor prochází svoji příslušnou oblast jemným řešičem (fine) a když mají všechny procesory dopočteno, provede se jeden korekční sekvenční průchod opět s hrubou strukturou (coarse). Iterační proces zastavíme předepsaným kritériem, nebo napevno zvoleným maximálním počtem iterací k_{max} . Abychom získali konečný počet časových kroků metody parareal T_P , musíme všechny provedené kroky příslušné coarse částem sečíst a pro každou iteraci $k = 1, 2, \dots$ k této sumě přičíst maximální počet provedených kroků z paralelně prováděných fine sekcí. V souvislosti se značením v nadcházejících tabulkách tedy hledáme

$$\mathbf{Suma} = \sum_{k=0}^{k_{max}} \sum_{n=1}^N steps_n(k, \text{coarse}) + \sum_{k=1}^{k_{max}} \max_{n=1, \dots, N} steps_n(k, \text{fine}). \quad (4.7)$$

Takto zapsaná **Suma** odpovídá výrazu T_P , ovšem v tabulkách bude použita obecněji jako částečný kumulativní součet pro daný řešič. Naším cílem pak je, aby $T_P < T_S$.

Zkoumaný výraz T_P se minimalizuje právě tehdy, když je počet kroků v každé oblasti skoro stejný. Toho se snažíme dosáhnout adaptivním rozdělením časového intervalu. Jako indikátor neefektivity rozdělení můžeme vzít pro danou iteraci k například relativní hodnotu

$$\mathbf{Neefektivita} = \frac{\max_{n=1, \dots, N} steps_n(k) - \min_{n=1, \dots, N} steps_n(k)}{\max_{n=1, \dots, N} steps_n(k)}. \quad (4.8)$$

Výše uvedené odstavce popisují přesně Tabulky 4.1 a 4.2, kde uvažujeme řešení s parametry $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$, přičemž Tabulka 4.1 obsahuje výsledky s optimálním dělením časového intervalu a Tabulka 4.2 obsahuje výsledky s intervalem děleným uniformně. Z indikátoru neefektivity vidíme, že adaptivní dělení je efektivnější než dělení uniformní. Pro tento konkrétní příklad dostáváme v případě adaptivního dělení $T_P = 403 < 460 = T_P$. Kdybychom navíc trochu upustili od požadované přesnosti a výpočet ukončili už po druhé iteraci, dostali bychom $T_P = 278 < 460 = T_P$, což už je výrazné zrychlení. Pokud bychom uvažovali v procentech podíl $\frac{T_P}{T_S}$, tedy kolik procent z výpočetního času sériového řešiče stačí paralelnímu řešiči, dostali bychom 60 %.

Stejnou dvojici výsledků, ale pro $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$, tedy pro jemnější prostorové dělení můžeme vidět na Tabulkách 4.3 a 4.4. Docházíme zde podobných závěrů, jako v předchozím příkladu, tedy při uvažování pouze prvních dvou iterací pararealu dostáváme $\frac{T_P}{T_S} = 62\%$. Avšak protože si adaptivní volba časového kroku vynutila mnohem více kroků, je zde absolutní zrychlení výraznější.

	Oblast					Suma	Neefektivita
	1	2	3	4	5		
$k=0$: coarse	4	6	6	6	6	28	0,333
$k=1$: fine	86	94	96	97	96	125	0,113
$k=1$: coarse	4	6	6	6	6	153	0,333
$k=2$: fine	86	94	96	97	96	250	0,113
$k=2$: coarse	4	6	6	6	6	278	0,333
$k=3$: fine	86	94	96	97	96	375	0,113
$k=3$: coarse	4	6	6	6	6	403	0,333
serial	–	–	–	–	–	460	–

Tabulka 4.1: Počty operací pro řešení s adaptivním dělením časové oblasti s parametry $M = 25$, $N = 5$, $TOL_1 = 10^{-2}$.

	Oblast					Suma	Neefektivita
	1	2	3	4	5		
$k=0$: coarse	9	6	5	5	4	29	0,556
$k=1$: fine	189	92	72	62	54	218	0,714
$k=1$: coarse	9	6	5	5	4	247	0,556
$k=2$: fine	189	92	72	62	54	436	0,714
$k=2$: coarse	9	6	5	5	4	465	0,556
$k=3$: fine	189	92	72	62	54	654	0,714
$k=3$: coarse	9	6	5	5	4	683	0,556
serial	–	–	–	–	–	460	–

Tabulka 4.2: Počty operací pro řešení s uniformním dělením časové oblasti s parametry $M = 25$, $N = 5$, $TOL_1 = 10^{-2}$.

	Oblast					Suma	Neefektivita
	1	2	3	4	5		
$k=0$: coarse	25	16	11	11	11	74	0,560
$k=1$: fine	221	215	214	215	209	295	0,054
$k=1$: coarse	25	16	11	11	11	369	0,560
$k=2$: fine	221	214	214	215	209	590	0,054
$k=2$: coarse	25	16	11	11	11	664	0,560
$k=3$: fine	221	214	214	215	209	885	0,054
$k=3$: coarse	25	16	11	11	11	959	0,560
serial	–	–	–	–	–	1065	–

Tabulka 4.3: Počty operací pro řešení s adaptivním dělením časové oblasti s parametry $M = 250$, $N = 5$, $TOL_1 = 10^{-2}$.

	Oblast					Suma	Neefektivita
	1	2	3	4	5		
$k=0$: coarse	43	10	8	7	7	75	0,837
$k=1$: fine	477	200	153	130	115	552	0,759
$k=1$: coarse	43	10	8	7	7	627	0,837
$k=2$: fine	477	199	153	130	114	1104	0,761
$k=2$: coarse	43	10	8	7	7	1179	0,837
$k=3$: fine	477	199	153	130	114	1656	0,761
$k=3$: coarse	43	10	8	7	7	1731	0,837
serial	–	–	–	–	–	1065	–

Tabulka 4.4: Počty operací pro řešení s uniformním dělením časové oblasti s parametry $M = 250$, $N = 5$, $TOL_1 = 10^{-2}$.

Závěr

Na začátku práce v Kapitole 1 jsme nejprve uvedli formulaci úlohy nelineární rovnice vedení tepla v jedné prostorové a jedné časové dimenzi. Následně jsme přešli ke slabé formulaci a zavedli první definici slabého řešení této úlohy. Kapitole jsme zakončili přidáním členu hraniční penalizace spolu s členy zachovávající symetrii lineární části úlohy a zavedli jsme příslušnou definici slabého řešení.

V Kapitole 2 jsme modifikovanou rovnici převedli pomocí MKP v prostoru na nelineární systém ODR. Ten jsme dále diskretizovali v čase, což vedlo na nelineární systém algebraických rovnic, pro který jsme navrhli iterační metodu, jenž problém redukuje na lineární systém algebraických rovnic. Nakonec jsme uvedli dva možné přístupy k adaptivní volbě časových kroků.

Celou Kapitole 3 jsme věnovali odvození metody parareal pomocí metody vícenásobné střelby, dále popisu samotného parareal algoritmu a důkazu Věty 1 pojednávající o konvergenci metody.

V poslední Kapitole 4 jsme se věnovali numerickým aspektům řešení konkrétní úlohy a to zejména konvergenci parareal řešení k řešení spočteném sériově v souvislosti s Větou 1. Z praktických výpočtů jsme na zkoumané úloze vyzorovali, že s užitím adaptivního časového kroku dává metoda parareal z hlediska konvergence rozumné výsledky už po prvních dvou iteracích, což může urychlit výpočet. Také jsme experimentálně ověřili, že tato adaptivní volba nemá podle Věty 1 záruku konvergence k referenčnímu řešení. Na konec jsme uvedli data k popisu teoretického zrychlení pararealu pro paralelní výpočet. Při použití chytře zvoleného dělení časového intervalu je pak možné zrychlení dosáhnout. Je třeba brát ovšem v potaz, že při řešení složitějších úloh se může chování metody parareal zhoršit.

Metoda parareal má tedy mnohé co nabídnout, zvláště ve spojení s vhodnými nástroji, jako je například adaptivní volba kroku a dělení časového intervalu. Ukazuje nám, že i problémy, které se na první pohled mohou zdát jako nevhodné k paralelnímu zpracování, má smysl paralelizovat.

Seznam použité literatury

- AMESTOY, P., DUFF, I. S., KOSTER, J. a L'EXCELLENT, J.-Y. (2001). A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, **23**(1), 15–41.
- AMESTOY, P., BUTTARI, A., L'EXCELLENT, J.-Y. a MARY, T. (2019). Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures. *ACM Transactions on Mathematical Software*, **45**, 2:1–2:26.
- AMODIO, P. a BRUGNANO, L. (2009). Parallel solution in time of odes: some achievements and perspectives. *Applied Numerical Mathematics*, **59** (3–4), 424–435. ISSN 0168-9274. doi: 10.1016/j.apnum.2008.03.024. URL <http://dx.doi.org/10.1016/j.apnum.2008.03.024>.
- BABUŠKA, I. (1973). The finite element method with penalty. *Mathematics of Computation*, **27**(122), 221–228. ISSN 00255718, 10886842. URL <http://www.jstor.org/stable/2005611>.
- BULÍČEK, M., ČERNÝ, R., JOHN, O., MÁLEK, J., POKORNÝ, M., ROKYTA, M. a STARÁ, J. (2018). Úvod do moderní teorie parciálních diferenciálních rovnic. URL https://www2.karlin.mff.cuni.cz/~pokorny/LectureNotes/moderni_teorie_color.pdf.
- DOLEŽÍ, V. a FEISTAUER, M. (2015). *Discontinuous Galerkin Method: Analysis and Applications to Compressible Flow*. Springer International Publishing. ISBN 9783319192673. doi: 10.1007/978-3-319-19267-3. URL <http://dx.doi.org/10.1007/978-3-319-19267-3>.
- GANDER, M. J. a LUNET, T. (2024). *Time Parallel Time Integration*. Society for Industrial and Applied Mathematics. ISBN 9781611978025. doi: 10.1137/1.9781611978025. URL <http://dx.doi.org/10.1137/1.9781611978025>.
- HAIRER, E., WANNER, G. a NØRSETT, P. S. (1993). *Solving Ordinary Differential Equations: Nonstiff Problems*. Springer Berlin Heidelberg. ISBN 9783540788621. doi: 10.1007/978-3-540-78862-1. URL <http://dx.doi.org/10.1007/978-3-540-78862-1>.
- KACHANOV, L. (1959). Variational methods of solution of plasticity problems. *Journal of Applied Mathematics and Mechanics*, **23**(3), 880–883. ISSN 0021-8928. doi: [https://doi.org/10.1016/0021-8928\(59\)90184-4](https://doi.org/10.1016/0021-8928(59)90184-4). URL <https://www.sciencedirect.com/science/article/pii/0021892859901844>.
- MADAY, Y. a TURINICI, G. (2002). A parareal in time procedure for the control of partial differential equations. *Comptes Rendus. Mathématique*, **335**(4), 387–392. ISSN 1778-3569. doi: 10.1016/S1631-073X(02)02467-6. URL [http://dx.doi.org/10.1016/S1631-073X\(02\)02467-6](http://dx.doi.org/10.1016/S1631-073X(02)02467-6).

TRINDADE, J. M. F. a PEREIRA, J. C. F. (2004). Parallel-in-time simulation of the unsteady navier–stokes equations for incompressible flow. *International Journal for Numerical Methods in Fluids*, **45**(10), 1123–1136. ISSN 1097-0363. doi: 10.1002/fld.732. URL <http://dx.doi.org/10.1002/fld.732>.

Seznam obrázků

3.1	Princip metody parareal na úloze (3.13).	17
4.1	Funkce $k(u)$ na intervalu $[-1, 1]$	18
4.2	Porovnání řešení s parametry $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$ referenčního řešení (vlevo nahoře), parareal iterace 0 (vpravo nahoře), parareal iterace 1 (vlevo dole) a parareal iterace 2 (vpravo dole). .	20
4.3	Porovnání řešení s parametry $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$ referenčního řešení (vlevo nahoře), parareal iterace 0 (vpravo nahoře), parareal iterace 1 (vlevo dole) a parareal iterace 2 (vpravo dole). .	21
4.4	Konvergence odhadu chyby pro $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$. . .	22
4.5	Konvergence odhadu chyby pro $M = 500, N = 10, \text{TOL}_1 = 10^{-2}$. .	22
4.6	Konvergence odhadu chyby pro $M = 25, N = 8, \text{TOL}_1 = 5 \cdot 10^{-1}$ s adaptivním krokem.	24
4.7	Konvergence odhadu chyby pro $M = 25, N = 8$ s pevným krokem $\text{TOL}_1 = 5 \cdot 10^{-1}$	24

Seznam tabulek

4.1	Počty operací pro řešení s adaptivním dělením časové oblasti s parametry $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$	26
4.2	Počty operací pro řešení s uniformním dělením časové oblasti s parametry $M = 25, N = 5, \text{TOL}_1 = 10^{-2}$	26
4.3	Počty operací pro řešení s adaptivním dělením časové oblasti s parametry $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$	27
4.4	Počty operací pro řešení s uniformním dělením časové oblasti s parametry $M = 250, N = 5, \text{TOL}_1 = 10^{-2}$	27