

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

Stavebná fakulta

Evidenčné číslo: SvF-5343-87842

**Numerické metódy pre level set úlohy
s použitím vzdialenostnej funkcie bez
znamienka**

Diplomová práca

Študijný program: matematicko-počítačové modelovanie

Študijný odbor: matematika

Školiace pracovisko: Katedra matematiky a deskriptívnej geometrie

Vedúci záverečnej práce: doc. RNDr. Peter Frolkovič, PhD.

Bratislava 2021

Bc. Nikola Gajdošová

Pod'akovanie

Ďakujem vedúcemu diplomovej práce doc. RNDr. Petrovi Frolkovičovi, PhD. za jeho cenné poznatky, rady a pripomienky ktoré mi značne pomohli pri tvorbe práce.

Abstrakt

Práca sa bude zaoberať rekonštrukciou viacfázového interfejsu pomocou vzdialenostnej funkcie bez znamienka. Navrhujeme presnejší výpočet vzdialenostnej funkcie na zlepšenie výsledku rekonštrukcie interfejsu. Rekonštrukciu aplikujeme na pohybujúci sa interfejs, pohybom riadiacim sa rovnicou lineárnej advekcie na riešenie ktorej použijeme plne explicitnú numerickú schému.

Abstract

This work will deal with multiphase interface reconstruction using unsigned distance function. We propose more accurate numerical solution of distance function in order to improve the interface reconstruction results. We also reconstruct a moving interface where the movement is determined by linear advection equation which we solve by fully explicit numerical scheme.

Kľúčové slová

interfejs, rozhranie, fáza, indikačná funkcia, vzdialenostná funkcia, eikonalova rovnica, rekonštrukcia interfejsu, lineárna rovnica advekcie

Key words

interface, phase, indicator function, distance function, eikonal equation, interface reconstruction, linear advection equation

Obsah

1	Úvod	1
2	Level set metódy	2
2.1	Vzdialenostná funkcia	2
3	Rekonštrukcia interfejsu	4
3.1	Štandardná lineárna interpolácia	4
3.2	VIIM - Voronoi implicitný interfejs metóda	7
3.3	Rekonštrukcia cez priesečníky hrán	10
3.4	Určenie hodnoty indikačnej funkcie χ pre vyvíjajúci sa interfejs .	13
3.4.1	Rozšírenie indikačnej funkcie χ	13
4	Numerické riešenie eikonalovej rovnice	15
4.1	Schéma vyššieho rádu	15
4.1.1	Metóda extrapolácie	16
4.1.2	Fast sweeping metóda	19
5	Numerické riešenie lineárnej rovnice advekcie	22
5.1	Schéma vyššieho rádu	22
6	Numerické experimenty	25
6.1	Experiment č.1	25
6.2	Experiment č.2	26
6.3	Experiment č.3	28
6.4	Experiment č.4	30
7	Záver	33

1 Úvod

Práca je motivovaná riešením problematiky rekonštrukcie hranice viacfázových rozhraní zo vzdialenostnej funkcie bez znamienka. Viacfázové rozhrania tzv. interfejsy sa vyskytujú v prírode ako napr. zrná v poly-kryštalickej materiáli alebo bunky rybičky *Danio pruhoané* [8] a taktiež napr. v aplikáciach na úlohy toku a transportu v pórovitom prostredí [9]. V práci budeme pracovať so vzdialenostnou funkciou bez znamienka k hranici objektu, ktorý budeme rekonštruovať pomocou metódy VIIM [3] a na zlepšenie výsledkov rekonštrukcie navrhne presnejší výpočet numerickej schémy druhého rádu s využitím extrapolácie [1] a konečných diferencií druhého rádu [5]. Taktiež sa v práci zaoberáme pohybom level set funkcie riadiacim sa lineárnou rovnicou advekcie a sledovaním pohybujúceho sa viacfázového rozhrania. Lineárnu rovnicu advekcie riešime pomocou plne explicitnej numerickej schémy druhého rádu [7] a navrhne iný prístup na výpočet hodnôt indikačnej funkcie určujúcej fázy viacfázového rozhrania ako v [3].

2 Level set metódy

Level set (úrovňové množiny) metódy sa využívajú na sledovanie evolúcie tzv. interfejsu (rozhrania) pohybujúceho sa relatívne komplikovaným spôsobom. V level set metódach je interfejs $\Gamma = \Gamma(t)$, tak že $\Gamma(t) : [0, \infty) \rightarrow R^N$, reprezentovaný ako nulová čiara (nulová úrovňová množina) v čase $t \geq 0$ nejakej level set funkcie s dimenziou vyššieho rádu ako Γ . Pre naše účely zvolíme $N = 2$ a naša oblasť bude fixovaná štvorcová oblasť $\Omega \subset R^2$. Pod-oblasť ohraničenú interfejsom Γ označíme Ω^Γ , pričom $\Gamma \equiv \partial\Omega^\Gamma$. Rovnica podľa ktorej sa bude pohybovať náš interfejs daný implicitne ako nulová hodnota level set funkcie bude lineárna rovnica advekcie s externe zadaným rýchlostným vektorovým poľom

$$\partial_t \phi(x, y, t) + \vec{W}(x, y) \cdot \nabla \phi(x, y, t) = 0, \quad \phi(x, y, 0) = \phi^0(x, y), \quad (1)$$

kde $\phi(x, y, t)$ predstavuje vyvíjajúcu sa v čase level set funkciu a $\vec{W}(x, y) = (U(x, y), V(x, y))$ predstavuje rýchlostné vektorové pole.

2.1 Vzdialenostná funkcia

Na nájdenie level set funkcie, napr. v čase $t = 0$ využijeme bezznamienkovú vzdialenostnú funkciu $d = d(x) \geq 0$ k interfejsu Γ , ktorú získame vyriešením tzv. eikonalovej rovnice [3, 6, 8],

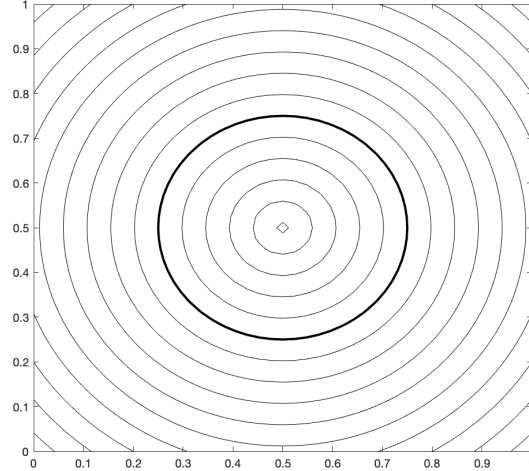
$$|\nabla d(x)| = 1, \quad x \in D, \quad d(x) = 0, \quad x \in \Gamma. \quad (2)$$

Vzdialenostná funkcia d nemá dobre definovaný gradient v bodoch interfejsu Γ , čo môže spôsobovať značné problémy v numerických metódach sledujúcich polohu interfejsu pomocou level set metódy. Z týchto dôvodov sa preferuje používať znamienkovú vzdialenostnú funkciu, definovanú ako

$$\begin{aligned} d(x) < 0, & \quad x \in \Omega^\Gamma \\ d(x) > 0, & \quad x \in \Omega/\overline{\Omega^\Gamma} \\ d(x) = 0, & \quad x \in \Gamma. \end{aligned}$$

Vzdialenostnú funkciu so znamienkom môžeme použiť v prípade, že interfejs Γ rozdeľuje výpočtovú oblasť $\Omega \subset R^2$ na dve disjunktné podmnožiny, tak že $\Omega = \Omega_1 \cup \Omega_2$ a znamienko vzdialenostnej funkcie d jednoznačne identifikuje tieto podmnožiny resp. fázy.

Ako príklad uvidíme exaktnú vzdialenostnú funkciu ku interfejsu Γ v tvare krížku s polomerom rovným 0.25 a stredom v bode (0.5, 0.5), ktorá má tvar $d(x, y) = \sqrt{(x - 0.5)^2 - (y - 0.5)^2} - 0.25$, kde $(x, y) \in \Omega$ a je znázornená na Obrázku 1.



(a) Interfejs Γ v tvare krúžku a vzdialenostná funkcia k takémuto interfejsu

Obr. 1

Pre interfejs Γ , ktorý rozdeľuje oblasť $\Omega \subset R^2$ na viac ako dve pod-oblasti, tak že,

$$\Omega = \bigcup_{i=1}^p \Omega_i, \quad p \geq 2,$$

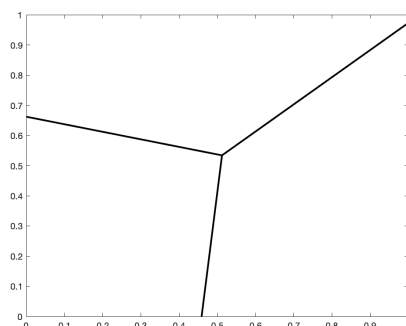
znamienko vzdialenostnej funkcie už nie je postačujúci identifikátor na určenie fázy. Z tohto dôvodu budeme v našej práci používať bezznamienkovú vzdialenostnú funkciu, aby sme mohli identifikovať interfejs, ktorý oddeľuje viaceré fázy. Na jednoznačné určenie fázy v každom bode $x \in \Omega$ si zavedieme indikačnú funkciu $\chi = \chi(x)$ [3], ktorá každému bode $x \notin \Gamma$, $x \in \Omega_i$ priradí index prislúchajúcej fázy i . Ak $x \in \Gamma$ tak pre jednoduchosť priradíme takémuto bode fázu v susediacom bode s vyššou hodnotou indexu i [3].

Definícia 1. *Trojný bod budeme nazývať bod x_t ktorého aspoň dva susediace body x_1 a x_2 nadobúdajú také hodnoty indikačnej funkcie, že platí*

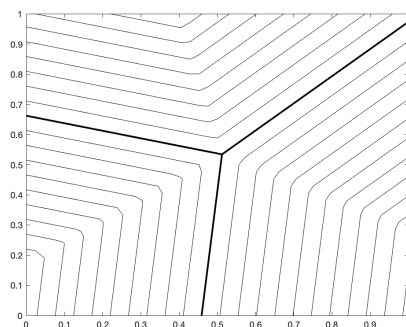
$$\chi(x) \neq \chi(x_1) \ \& \ \chi(x) \neq \chi(x_2) \ \& \ \chi(x_1) \neq \chi(x_2).$$

Pre názornosť uvedieme na Obrázku 2 príklad interfejsu deliaceho oblasť na 3 fázy, $\chi(x) = p \in \{1, 2, 3\}$, $x \in \Omega$, kde je vidno prítomnosť tzv. trojného bodu.

Tento interfejs je určený tromi polpriamkami so spoločným začiatočným bodom. Takýto interfejs budeme v práci voľne nazývať ako interfejs v tvare písmena Y.



(a) Interface Y



(b) Vzdialenostná funkcia k interfejsu Y

Obr. 2

3 Rekonštrukcia interfejsu

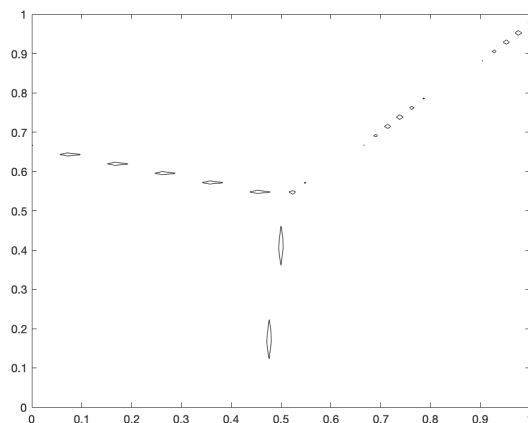
V tejto časti sa budeme zaoberať problémom zobrazenia interfejsu daného implicitne ako nulová čiara bezznamienkovej vzdialenostnej funkcie. Pre názornosť budeme používať interfejs v tvare písmena Y. Pri numerickom riešení budeme typicky pracovať s funkciami, ktoré budú dané len v bodoch štvorcovej siete. Napríklad ak budeme mať danú začiatočnú funkciu, tak numerická metóda typicky využije túto funkciu len v bodoch siete.

3.1 Štandardná lineárna interpolácia

Ak by sme mali iba bezznamienkovú vzdialenostnú funkciu bez indikačnej funkcie χ určujúcu fázy, nedokázali by sme štandardnou lineárnou interpoláciou zrekonštruovať nulovú množinu vzdialenostnej funkcie d . Pri numerickom riešení robíme štandardnú lineárnu interpoláciu využitím hodnôt v bodoch siete.

Z tohoto dôvodu štandardné nástroje vizualizácie napr. v Matlabe nevedia zobraziť nulovú čiaru takejto funkcie. Pre názornosť vid' Obrázok 3, na ktorom

vidno kontúry minimálnej hodnoty a to 0.005, ktorú dokázal softvér Matlab zobrazit'.



(a) Zobrazenie kontúry s hodnotou 0.005 štandardnou lineárnou interpoláciou

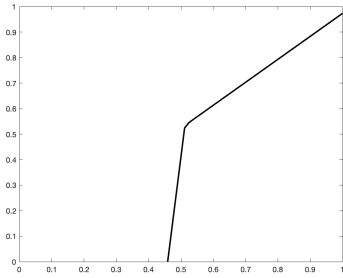
Obr. 3

Vďaka indikačnej funkcii χ dokážeme interfejs Γ zrekonštruovať po častiach aj pomocou lineárnej interpolácie. Takto zrekonštruovaný interfejs budeme značiť Γ_L .

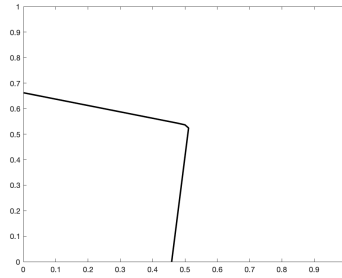
Časť interfejsu, zrekonštruovanú medzi oblasťou Ω_i a Ω_j pomocou lineárnej interpolácie, budeme značiť $\Gamma_{L,ij}$. Pre interfejs v tvare písmena Y s tromi fázami budeme mať $\Gamma_L = \Gamma_{L,12} \cup \Gamma_{L,13} \cup \Gamma_{L,23}$.

Pomocou indikačnej funkcie vieme oblasť Ω rozdeliť na dve pod-oblasti, konkrétne pod-oblasť Ω_i , kde $\chi(x) = i$ a $d(x) = |d(x)|$, pre $x \in \Omega_i$ a oblasť Ω_j , kde $\chi(x) = j$ a $d(x) = -|d(x)|$, pre $x \in \Omega_j$ a $i \neq j$ a následne použiť štandardnú lineárnu interpoláciu v bodoch siete na stykoch fáz i a j a zrekonštruovať body interfejsu $\Gamma_{L,ij}$. Pre názornosť, vid' Obrázok 4 a 5.

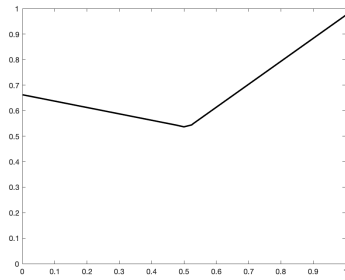
Žiaľ takýmto spôsobom strácame informáciu o interfejse okolo trojného bodu v ktorom sa stretávajú tri fázy, pretože lineárnu interpoláciu robíme vždy lokálne iba medzi dvomi fázami a teda nezískame informáciu o skutočnom gradiente okolo trojného bodu.



(a) Časť $\Gamma_{L,12}$ interfejsu Γ medzi fázami 1 a 2 zrekonštruovaná lineárnou interpoláciou

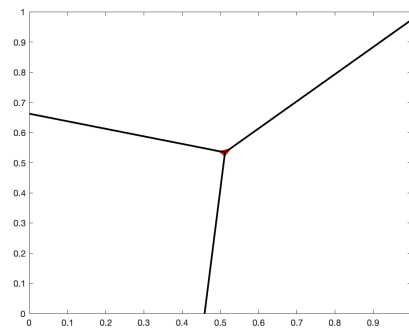


(b) Časť $\Gamma_{L,13}$ interfejsu Γ medzi fázami 1 a 3 zrekonštruovaná lineárnou interpoláciou

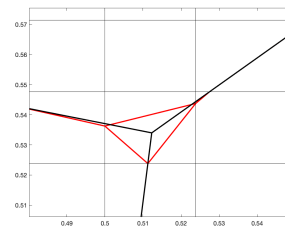


(c) Časť $\Gamma_{L,23}$ interfejsu Γ medzi fázami 2 a 3 zrekonštruovaná lineárnou interpoláciou

Obr. 4



(a) Exaktný interfejs Γ (čierna farba) a interfejs zrekonštruovaný pomocou lineárnej interpolácie Γ_L (červená farba)



(b) Priblíženie k trojnému bodu

Obr. 5

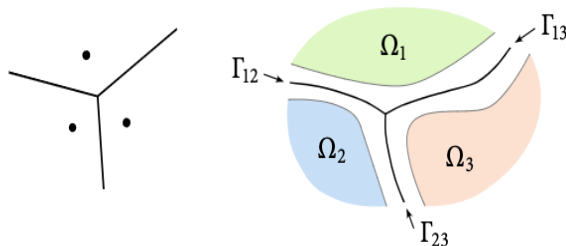
V nasledujúcich častiach si uvedieme možnosti rekonštrukcie a vizualizácie interfejsu s trojným bodom.

3.2 VIIM - Voronoi implicitný interfejs metóda

V tejto časti sa budeme zaoberať rekonštrukciou takzvaného Voronoi interfejsu.

Na úvod si pripomenieme, že Voronoi diagram množiny m uzlov v priestore R^n je dekompozícia priestoru R^n na m rôznych buniek s takou vlastnosťou, že všetky body patriace danej bunke sa nachádzajú bližšie k prislúchajúcemu uzlu bunky m ako k inému uzlu. Hranica medzi týmito bunkami je množina bodov, ktoré sú ekvidistantné aspoň k dvom uzlom a nenachádzajú sa bližšie k inému uzlu.

Vo všeobecnosti môžeme hľadať Voronoi diagram nielen pre množinu uzlov, ale taktiež pre množinu nepretínajúcich sa kriviek v R^n . Analogicky vieme rozložiť oblasť na bunky, kedy body prislúchajúce jednej konkrétnej bunke sa nachádzajú v menšej vzdialenosti ku jednej konkrétnej z kriviek spomedzi všetkých kriviek. Voronoi interfejs je teda hranica týchto buniek a budeme ho značiť Γ_V . Pre názornosť vid' Obrázok 6 [8], kde vidno tri krivky znázornené sivou farbou, ktoré vymedzili oblasti Ω_1, Ω_2 a Ω_3 . A čiernou farbou je znázornený Voronoi diagram k týmto trom oblastiam resp. krivkám.



Obr. 6: Voronoi diagram pre $m = 3$ a Voronoi interfejs $\Gamma_V = \Gamma_{12} \cup \Gamma_{13} \cup \Gamma_{23}$ hľadaný pre tri krivky [8].

Analogicky v metóde level set nám indikačná funkcia rozdelí oblasť Ω na bunky a hranicou týchto buniek je Voronoi interfejs. Namiesto interfejsu predstavujúceho body, v ktorých level set funkcia nadobúda nulovú hodnotu použijeme implicitný Voronoi interfejs, ktorý skonštruujeme následovne [3].

Budeme uvažovať nejakú ϵ hodnotu level set funkcie d , definovanú ako

$$\Gamma_\epsilon = \{x \in \Omega : d(x) = \epsilon\}$$

Pre nejaké vhodne zvolené $\epsilon > 0$ je gradient $\nabla d(x)$ dobre definovaný pre skoro všetky x susediace s Γ_ϵ . Pomocou indikačnej funkcie χ , dostaneme interfejs Γ_ϵ pozostávajúci z troch disjunktných častí

$$\Gamma_\epsilon^i = \{x \in \Omega : \chi(x) = i \quad a \quad d(x) = \epsilon\},$$

ležiacich v rôznych fázach. Následne vieme z takéhoto interfejsu zrekonštruovať (aproximovať) Voronoi interfejs Γ_V definovaný ako množina bodov $x \in \Omega$, ktoré sa nachádzajú v ekvidistantnej vzdialenosti k aspoň dvom rôznym ϵ množinám patriacim rôznym fázam a ostatné ϵ množiny sú k nemu vo väčšej vzdialenosti,

$$\Gamma_V = \{x \in D : \exists i \neq j, d(x, \Gamma_{\epsilon,i}) = d(x, \Gamma_{\epsilon,j}) \geq d(x, \Gamma_{\epsilon,k}), k \neq i, j\}. \quad (3)$$

Voronoi interfejs Γ_V vieme zrekonštruovať po častiach medzi rôznymi fázami a platí $\Gamma_V = \Gamma_{12} \cup \Gamma_{13} \cup \Gamma_{23}$. Časť ktorá predstavuje interfejs medzi fázami i a j , keď $i \neq j$ je daná implicitne ako:

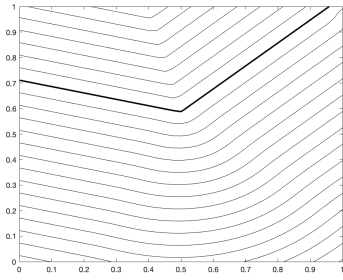
$$\Gamma_{ij} = \{x \in D : d_i(x) = d_j(x) \geq d_k(x), k \neq i, j\}, \quad (4)$$

kde d_i predstavuje vzdialenostnú funkciu k interfejsu Γ_ϵ^i , ležiaceho vo fáze s hodnotou i ,

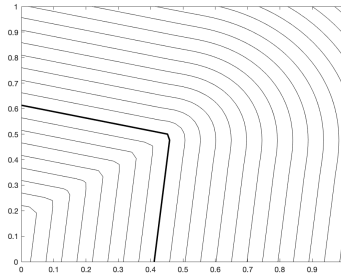
$$|\nabla d_i(x)| = 1, x \in \Omega, \quad d_i(x) = \epsilon, x \in \Gamma_\epsilon^i. \quad (5)$$

Pre názornosť vid'. Obrázok 7, na ktorom sú obrázky (a), (b), (c) 3 vzdialenostných funkcií d_1, d_2, d_3 k 3 častiam ϵ interfejsu $\Gamma_\epsilon^1, \Gamma_\epsilon^2, \Gamma_\epsilon^3$.

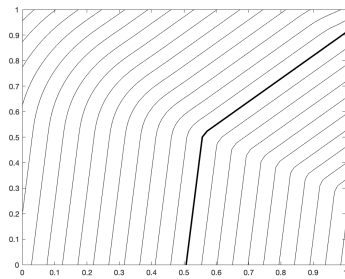
Hlavnou myšlienkou VIIM metódy, je že namiesto pôvodného interfejsu Γ reprezentujúceho nulovú hodnotu level set funkcie, budeme sledovať interfejs Γ_ϵ s dobre definovaným gradientom reprezentujúci ϵ hodnotu level set funkcie d .



(a) Časť interfejsu Γ_1^ϵ a vzdialenosťná funkcia d_1

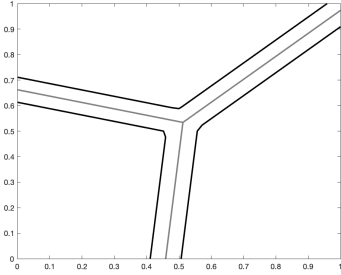


(b) Časť interfejsu Γ_2^ϵ a vzdialenosťná funkcia d_2

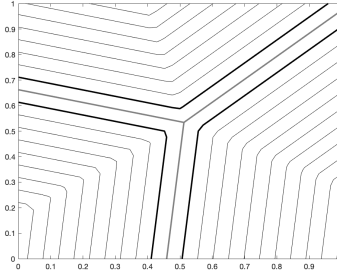


(c) Časť interfejsu Γ_3^ϵ a vzdialenosťná funkcia d_3

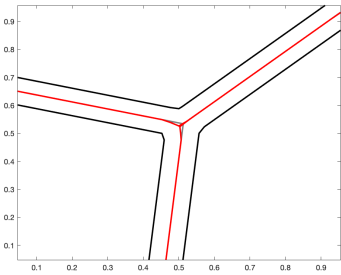
Obr. 7



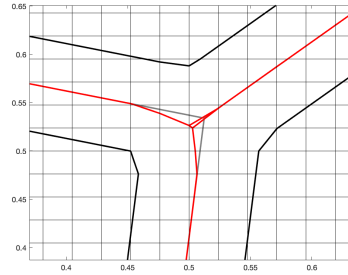
(a) Pôvodný interfejs Γ (sivá) a epsilonový interfejs Γ_ϵ (čierna)



(b) Pôvodný interfejs Γ (sivá) a epsilonový interfejs Γ_ϵ (čierna) a vzdialostná funkcia k Γ_ϵ



(c) Pôvodný interfejs Γ (sivá), epsilonový interfejs Γ_ϵ (čierna) a zrekonštruovaný VIIM Γ_V (červená)



(d) $\Gamma_\epsilon, \Gamma_V, \Gamma$ priblíženie

Obr. 8

3.3 Rekonštrukcia cez priesečníky hrán

Ak máme interfejs, ktorého analytické vyjadrenie poznáme, dokážeme ho zrekonštruovať pomocou nájdenia priesečníkov interfejsu s hranami siete, ktoré budeme značiť e_x a e_y . Majme napríklad interfejs v tvare písmena Y, ktorého jednotlivé časti vieme vyjadriť analyticky nasledovnými rovnicami:

$$y_1 = p_y + t_1(-p_x + x), \quad (6)$$

$$y_2 = p_y + t_2(-p_x + x), \quad (7)$$

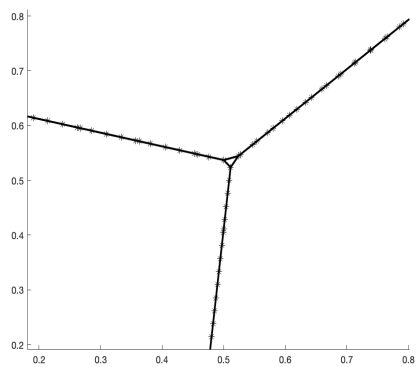
$$y_3 = p_y + t_3(-p_x + x). \quad (8)$$

Hodnota t_p predstavuje smernicu jednotlivých polpriamok a $\{p_x, p_y\}$ sú súradnice trojného bodu, v ktorom sa polpriamky stretávajú a $x \in \langle A, B \rangle$, kde $A, B \in R$ a predstavujú hranice štvorcovej oblasti Ω . Takto ak hľadáme priesečníky s x -ovou hranou e_x , teda vertikálnu hranu fixovanú v nejakej hodnote x , tak v rovnici za x dosadíme x -ovú súradnicu hrany a dostaneme y -ovú

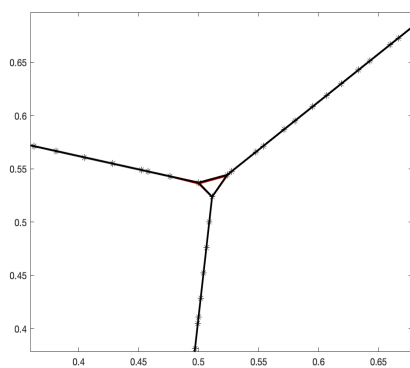
súradnicu priesečníka. Analogicky s priesečníkom s y -ovou hranou e_y , teda horizontálnu hranu fixovanú v nejakej hodnote y , dosadíme v rovniciach za y_p y -ovú hodnotu hrany a vyjadríme si x -ovú súradnicu priesečníka s hranou. Taktiež podotýkame, že priesečníky hľadáme len na hranách, ktorých koncové body sú v rôznych fázach a teda pre tieto hrany je indikovaná prítomnosť priesečníka, nulovej hodnoty vzdialenostnej funkcie.

Uvedeným postupom dostaneme neusporiadanú množinu bodov, ktorú dokážeme rozumne zobrazit' nami navrhnutou metódou bez použitia zabudovaných funkcionalít resp. iných metód. Okrem zobrazenia interfejsu, taktiež navrhujeme vypočítat' vzdialenostnú funkciu k nájdeným priesečníkom metódou opísanou v bakalárskej práci [2] ako vzdialenostná funkcia k bodom skeletu. Dostaneme presnejšiu vzdialenostnú funkciu najmä v okolí trojného bodu pretože nepracujeme s hodnotami vzdialenostnej funkcie k interfejsu v uzlových bodoch ale k vzdialenostnej funkcii k exaktným priesečníkom.

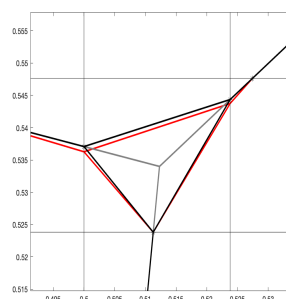
Na Obrázku č.9 vidno, že nájdené priesečníky sú exaktné a teda zhodné s priesečníkmi pôvodného interfejsu.



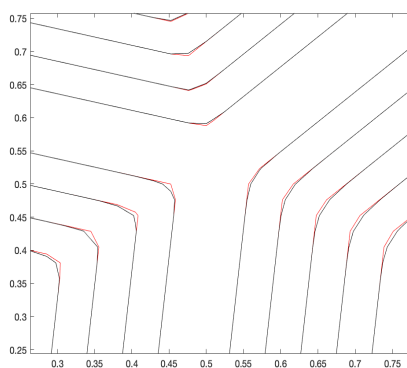
(a) Nájdené a pospájane priesečníky



(b) Porovnanie pospájaných exaktných priesečníkov (čierna) s rekonštruovaným interfejsom Γ_L pomocou lineárnej interpolácie



(c) Priblíženie k trojnému bodu, exaktný interfejs Γ (sivá)



(d) Porovnanie vzdialenostnej funkcie k priesečníkom (čierna) [2] a vzdialenostnej funkcie v bodoch siete (červená)

3.4 Určenie hodnoty indikačnej funkcie χ pre vyvíjajúci sa interfejs

V časti 1. sme zaviedli bezznamienkovú vzdialenostnú funkciu $d(x)$ a indikačnú funkciu χ na jednoznačné určenie fázy.

Keďže sa náš interfejs bude pohybovať a vyvíjať v čase podľa (1) zavedieme si časovo závislú vzdialenostnú funkciu $d(x, t)$, $x \in \Omega$, $t \in R^+$ a časovo závislú indikačnú funkciu $\chi(x, t)$, $x \in \Omega$, $t \in R^+$. Vzdialenostná funkcia a indikačná funkcia v čase $t = 0$ sú zadané a známe. Vzdialenostnú funkciu $d(x, t)$ pre $t > 0$ nájdeme riešením (1). Hodnoty indikačnej funkcie $\chi(x, t)$ pre $t > 0$ je dôležité správne určiť aby sme poznali novú polohu interfejsu, ktorú z bezznamienkovej vzdialenostnej funkcie samotnej nedokážeme zistiť. V [3] sa pracuje s funkciou $d(x, t) - \epsilon$ a teda sa sleduje interfejs Γ_ϵ a hodnoty $\chi(x, t)$ sa určia pomocou nasledovného vzťahu

$$\chi(x, t) = p \Leftrightarrow d_p(x, t) < \min\{d_q(x, t), d_r(x, t)\}, \quad (9)$$

kde $p, q, r \in \{1, 2, 3\}$ a $p \neq q \neq r$. A teda potrebujeme tri vzdialenostné funkcie $d_1(x, t)$, $d_2(x, t)$, $d_3(x, t)$ (5) a hodnota indikačnej funkcie $\chi(x, t)$ pre $t > 0$ bude rovná indexu p podľa (9) a teda indexu vzdialenostnej funkcie s najmenšou hodnotou v bode x . Navrhujeme aj pohodlnejší prístup, kedy nepracujeme s tromi vzdialenostnými funkciami ale len s jednou $d(x) - \epsilon$ bez použitia $d_1(x, t)$, $d_2(x, t)$, $d_3(x, t)$ na určenie hodnôt $\chi(x, t)$.

3.4.1 Rozšírenie indikačnej funkcie χ

Navrhujeme rozšíriť indikačnú funkciu $\chi(x, 0)$ o štvrtú fázu tak, že $\chi(x, 0) = 4$, pre $x \in \Omega$, $d(x, 0) - \epsilon < 0$. A teda lineárnu advekčnú rovnicu podľa numerickej schémy (50) budeme aplikovať pre každú oblasť Ω_p , $p \in 1, 2, 3, 4$, $x \in \Omega_p$ $\chi(x, t) = p$ samostatne.

Zavedieme si znamienkovú vzdialenostnú funkciu $s_p(x, t)$ pre každú oblasť Ω_p tak, že

$$s_p(x, t) = |d(x) - \epsilon|, \quad x \in \Omega - \Omega_p \quad (10)$$

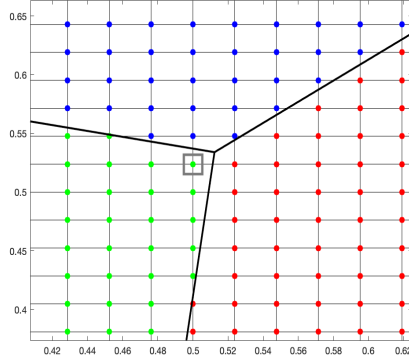
$$s_p(x, t) = -|d(x) - \epsilon|, \quad x \in \Omega_p. \quad (11)$$

Hodnoty $s_p(x, t + \tau)$ v ďalšom časovom kroku vypočítame podľa numerickej schémy (50) a v bodoch x v ktorých platí, že

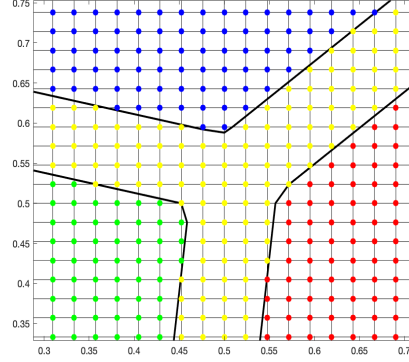
$$s_p(x, t) \cdot s_p(x, t + \tau) < 0, \quad (12)$$

treba správne zmeniť hodnotu indikačnej funkcie. Zmena znamienka môže nastať iba v bode x , ktorého aspoň jeden susedný bod nadobúda inú hodnotu fázy a teda medzi týmito bodmi sa nachádza bod interfejsu. Keďže pracujeme s interfejsom Γ_ϵ tak vieme jednoznačne určiť hodnotu novej fázy, pretože je to interfejs bez trojného bodu. Ak by sme pracovali s pôvodným interfejsom Γ tak by sme pre trojný bod sme nevedeli jednoznačne určiť hodnotu novej fázy, pretože

v susediacich bodoch sa nachádzajú body z dvoch rôznych fáz. Pre názornosť vid' Obrázok 10, kde je vyznačený trojný bod ktorého susedný bod zhora a vpravo ležia v dvoch rôznych fázach. Pre ostatné body platí, že susedné body ležia v rovnakej fáze alebo v jednej rôznej.



(a) Exaktný interfejs Γ , trojný bod a fázy znázornené rôznymi farbami



(b) Interfejs Γ_ϵ bez trojného bodu a fázy znázornené rôznymi farbami a pridaná 4. fáza znázornená žltou farbou.

Obr. 10

V prípade, že pohybujeme interfejs Γ s trojným bodom a teda pracujeme so vzdialenostnou funkciou $d(x, t)$ a nie $d(x, t) - \epsilon$ a s pôvodnou indikačnou funkciou χ bez pridanej štvrtej fázy navrhujeme podmienku podľa, ktorej by sme určili, ktorá zo susediacich fáz nahradí aktuálnu hodnotu fázy v trojnóm bode ak v ňom funkcia $s_p(x, t)$ zmenila znamienko. Funkcia $s_p(x, t)$ bude v takomto prípade definovaná ako

$$s_p(x, t) = |d(x)|, x \in \Omega - \Omega_p \quad (13)$$

$$s_p(x, t) = -|d(x)|, x \in \Omega_p . \quad (14)$$

Hodnoty $s_p(x, t + \tau)$ v ďalšom časovom kroku vypočítame podľa numerickej schémy a v bodoch x v ktorých platí, že

$$s_p(x, t) \cdot s_p(x, t + \tau) < 0 \quad (15)$$

zmeníme hodnotu fázy. Novú fázu pre trojný by sme určili podľa veľkosti Courantovho čísla vystupujúceho v numerickej schéme, tak že by sme sa rozhodli pre fázu v susednom bode s menším Courantovým číslom. Pre ostatné body vieme fázu určiť jednoznačne.

Táto podmienka však nie je postačujúca pre jednoznačne určenie fázy pre trojný bod a preto odporúčame pracovať s interfejsom Γ_ϵ a s pridanou štvrtou fázou.

4 Numerické riešenie eikonalovej rovnice

Na implementáciu rekonštrukcie interfejsu podľa VIIM metódy potrebujeme numerické riešenie vzdialenostnej funkcie,

$$|\nabla d(x)| = 1, \quad x \in D, \quad d(x) = 0, \quad x \in \Gamma. \quad (16)$$

Napríklad, ak si zoberieme interfejs Υ , tak na rekonštrukciu časti Γ_{ij} Voronoi interfejsu Γ_V medzi fázami s indikačnou funkciou s hodnotou i a j potrebujeme vzdialenostné funkcie $d_i(x)$ a $d_j(x)$, ktoré sú riešením rovnice (5). V bakalárskej práci [2] sme používali numerické riešenie eikonalovej rovnice prvého rádu s využitím extrapolácie, ale na rekonštrukciu Voronoi interfejsu sme boli motivovaný použiť metódu vyššieho rádu [5] na získanie lepších výsledkov rekonštrukcie.

4.1 Schéma vyššieho rádu

V tejto časti uvedieme numerickú schému vyššieho rádu na riešenie eikonalovej rovnice (2). Uvažujeme štvorcovú výpočtovú oblasť Ω s rovnomernou štvorcovou sieťou. Veľkosť oblasti je $L \times L$, kde $L > 0$ a veľkosť kroku je $h = L/N$, kde $N > 0$. Bod siete budeme značiť ako (x_i, y_j) , kde $x_i = i \cdot h$ a $y_j = j \cdot h$. Hodnotu vzdialenostnej funkcie v bode (x_i, y_j) budeme značiť d_{ij} , ktorá predstavuje aproximáciu exaktnej hodnoty $d(x_i, y_j)$ vzdialenostnej funkcie v tomto bode.

Prvým krokom je aproximácia gradientu v bodoch siete $\nabla d_{ij} \approx \nabla d(x_i, y_j)$ vystupujúceho v rovnici (2). Na to aby sme dosiahli metódu vyššieho rádu, použijeme konečnú diferenciu druhého rádu,

$$\nabla d_{ij} = (\partial_x d_{ij}, \partial_y d_{ij}) \approx \frac{1}{2h} (\delta_x d_{ij}, \delta_y d_{ij}), \quad (17)$$

Na rozdiel od [5] používame len konečnú diferenciu druhého rádu bez použitia premennej *switch*, ktorá za nejakých okolností použije raz konečnú diferenciu prvého rádu a raz druhého rádu.

Keďže táto metóda využíva upwind metódu na aproximáciu gradientu ∇d_{ij} , dostávame následovné možnosti výpočtu $\delta_x d_{ij}$

$$\delta_x d_{ij} = \begin{cases} 3d_{ij} - 4d_{i-1j} + d_{i-2j} & d_{i-1j} \leq \min(d_{ij}, d_{i+1j}) \\ -3d_{ij} + 4d_{i+1j} - d_{i+2j} & d_{i+1j} \leq \min(d_{ij}, d_{i-1j}) \\ 0 & d_{ij} \leq \min(d_{i+1j}, d_{i-1j}) \end{cases} \quad (18)$$

a $\delta_y d_{ij}$

$$\delta_y d_{ij} = \begin{cases} 3d_{ij} - 4d_{ij-1} + d_{ij-2} & d_{ij-1} \leq \min(d_{ij}, d_{ij+1}) \\ -3d_{ij} + 4d_{ij+1} - d_{ij+2} & d_{ij+1} \leq \min(d_{ij}, d_{ij-1}) \\ 0 & d_{ij} \leq \min(d_{ij+1}, d_{ij-1}) \end{cases} \quad (19)$$

Na aproximáciu gradientu ∇d_{ij} v bode (x_i, y_j) využívame hodnoty v susedných bodoch, v ktorých je hodnota vzdialenostnej funkcie menšia a teda sa nachádzajú bližšie k interfejsu. Eikonalovu rovnicu (2) prepíšeme do tvaru $|\nabla d|^2 = 1$ a použitím aproximácie gradientu konečnou diferenciou (18) a (19), dostaneme rovnicu v tvare

$$(\delta_x d_{ij})^2 + (\delta_y d_{ij})^2 = 4h^2 . \quad (20)$$

V ďalšej časti nadviažeme na túto metódu a vylepšíme ju tzv. metódou extrapolácie opísanou v [1, 6].

4.1.1 Metóda extrapolácie

Metódou extrapolácie chceme dosiahnuť presnejší výsledok hodnôt vzdialenostnej funkcie d_{ij} pre body ležiace blízko interfejsu Γ ako je opísané v [5] kde používajú inicializáciu exaktnými hodnotami blízko hranice interfejsu. Všimnime si, že vo výpočte gradientu (18) a (19) používame až 4 susediace hodnoty $d_{i\pm 1j}$, $d_{ij\pm 1}$, $d_{i\pm 2j}$ a taktiež $d_{ij\pm 2}$. Ak sa susediaci bod $(x_{i\pm 1}, y_j)$ alebo $(x_i, y_{j\pm 1})$ alebo $(x_{i\pm 2}, y_j)$ alebo bod $(x_i, y_{j\pm 2})$ nachádza v inej fáze ako bod (x_i, y_i) v ktorom chceme vypočítať hodnotu vzdialenostnej funkcie d_{ij} resp. leží vedľa hranice interfejsu, nepoužijeme v rovnici jeho aktuálnu hodnotu $d_{i\pm 1j}$ alebo $d_{ij\pm 1}$ alebo $d_{i\pm 2j}$ alebo $d_{ij\pm 2}$, ale navrhujeme ju v tomto bode dopočítať pomocou metódy extrapolácie [1, 6].

Zavedieme si funkciu $\phi : \Omega \rightarrow R$, ktorá bude predstavovať nejakú vhodnú level set funkciu, pre ktorú v bodoch siete (x_i, y_j) bude platiť

$$(x_i, y_j) \in \begin{cases} \Omega & \phi_{i,j} < 0 \\ \Omega \setminus \bar{\Omega} & \phi_{i,j} > 0 \\ \partial\Omega & \phi_{i,j} = 0 \end{cases} \quad (21)$$

Ďalej si zavedieme pojmy pre susediace body, ktoré považujeme za dôležité a budeme ich používať ďalej v texte.

Definícia 1. *Bod ležiaci vedľa interfejsu v smere osi x , budeme nazývať bod (x_i, y_j) pre ktorý platí, $\chi(x_i, y_j) \neq \chi(x_{i\pm 1}, y_j)$, kde χ je indikačná funkcia. Analogicky bod ležiaci vedľa interfejsu v smere osi y , budeme nazývať bod (x_i, y_j) pre ktorý platí, $\chi(x_i, y_j) \neq \chi(x_i, y_{j\pm 1})$.*

Definícia 2. *Bod ležiaci blízko interfejsu v smere osi x , budeme nazývať bod (x_i, y_j) pre ktorý platí, $\chi(x_i, y_j) \neq \chi(x_{i\pm 2}, y_j)$, kde χ je indikačná funkcia. A analogicky bod ležiaci vedľa interfejsu v smere osi y , budeme nazývať bod (x_i, y_j) pre ktorý platí, $\chi(x_i, y_j) \neq \chi(x_i, y_{j\pm 2})$.*

Pre bod ležiaci vedľa interfejsu v smere osi x platia nasledujúce vzťahy:

$$\begin{aligned} \text{ak } \chi(x_i, y_j) \neq \chi(x_{i-1}, y_j), \text{ tak } \phi_{ij}\phi_{i-1j} < 0, \\ \text{ak } \chi(x_i, y_j) \neq \chi(x_{i+1}, y_j), \text{ tak } \phi_{ij}\phi_{i+1j} < 0 \end{aligned} \quad (22)$$

Analogicky pre bod ležiaci vedľa interfejsu v smere osi y platia nasledujúce nerovnosti:

$$\begin{aligned} \text{ak } \chi(x_i, y_j) \neq \chi(x_i, y_{j-1}), \text{ tak } \phi_{ij}\phi_{ij-1} < 0, \\ \text{ak } \chi(x_i, y_j) \neq \chi(x_i, y_{j+1}), \text{ tak } \phi_{ij}\phi_{ij+1} < 0. \end{aligned} \quad (23)$$

Pre bod ležiaci blízko interfejsu v smere osi x platia nasledujúce vzťahy:

$$\begin{aligned} \text{ak } \chi(x_i, y_j) \neq \chi(x_{i+2}, y_j), \text{ tak } \phi_{ij}\phi_{i-2j} < 0, \\ \text{ak } \chi(x_i, y_j) \neq \chi(x_{i-2}, y_j), \text{ tak } \phi_{ij}\phi_{i+2j} < 0. \end{aligned} \quad (24)$$

Analogicky pre bod blízko interfejsu v smere osi y platia nasledujúce vzťahy:

$$\begin{aligned} \text{ak } \chi(x_i, y_j) \neq \chi(x_i, y_{j-2}), \text{ tak } \phi_{ij}\phi_{ij-2} < 0, \\ \text{ak } \chi(x_i, y_j) \neq \chi(x_i, y_{j+2}), \text{ tak } \phi_{ij}\phi_{ij+2} < 0. \end{aligned} \quad (25)$$

Ďalej si uvedieme vzťahy potrebné na výpočet extrapolovaných hodnôt.

Ak bod (x_i, y_j) leží vedľa interfejsu v smere osi x , potom hľadaný bod $x_{i\pm\alpha j}$ na hranici $\partial\Omega$ ležiaci medzi bodmi (x_i, y_j) a $(x_{i\pm 1}, y_j)$ môžeme vypočítať ako,

$$x_{i\pm\alpha j} = \alpha_{i\pm 1j}x_{i\pm 1j} + (1 - \alpha_{i\pm 1j})x_{ij}, \quad (26)$$

kde $\alpha_{i\pm 1j} \in (0, 1]$.

Analogicky ak bod (x_i, y_j) leží vedľa interfejsu v smere osi y , potom hľadaný bod $y_{ij\pm\alpha}$ na hranici $\partial\Omega$ ležiaci medzi bodmi (x_i, y_j) a $(x_i, y_{j\pm 1})$ môžeme vypočítať ako,

$$y_{ij\pm\alpha} = \alpha_{ij\pm 1}y_{ij\pm 1} + (1 - \alpha_{ij\pm 1})y_{ij}, \quad (27)$$

kde $\alpha_{ij\pm 1} \in (0, 1]$.

Body $x_{i\pm\alpha j}$ a $y_{ij\pm\alpha}$ budeme potrebovať ak sa bod (x_i, y_j) nachádza vedľa interfejsu a teda platia vzťahy (22) a (23).

Taktiež si uvedieme vzťahy, na výpočet hľadaného bodu na hranici $\partial\Omega$ pre bod ležiaci blízko interfejsu aj keď ich výpočet je analogický ako pre bod ležiaci vedľa interfejsu. Ak bod (x_i, y_j) leží blízko interfejsu v smere osi x potom hľadaný bod môže ležať medzi bodmi (x_{i+1}, y_j) a (x_{i+2}, y_j) a jeho polohu vypočítame ako

$$x_{i+1+\alpha j} = \alpha_{i+2j}x_{i+2j} + (1 - \alpha_{i+2j})x_{i+1j}, \quad (28)$$

alebo hľadaný bod môže ležať medzi bodmi (x_{i-1}, y_j) a (x_{i-2}, y_j) a jeho polohu vypočítame ako

$$x_{i-1-\alpha j} = \alpha_{i-2j}x_{i-2j} + (1 - \alpha_{i-2j})x_{i-1j}, \quad (29)$$

kde $\alpha_{i\pm 2j} \in (0, 1]$.

Analogicky ak bod (x_i, y_j) leží blízko interfejsu v smere osi y potom hľadaný bod môže ležať medzi bodmi (x_i, y_{j+1}) a (x_i, y_{j+2}) a jeho polohu vypočítame ako

$$x_{ij+1+\alpha} = \alpha_{ij+2}x_{ij+2} + (1 - \alpha_{ij+2})x_{ij+1}, \quad (30)$$

alebo hľadaný bod môže ležať medzi bodmi (x_i, y_{j-1}) a (x_i, y_{j-2}) a jeho polohu vypočítame ako

$$x_{ij-1-\alpha} = \alpha_{ij-2}x_{ij-2} + (1 - \alpha_{ij-2})x_{ij-1}, \quad (31)$$

kde $\alpha_{ij\pm 2} \in (0, 1]$.

Vyjadrenie $\alpha_{i\pm 1j}$ pre bod ležiaci vedľa interfejsu dostaneme z rovnice $\phi_{ij\pm\alpha} = 0 = \alpha_{i\pm 1j}\phi_{i\pm 1j} + (1 - \alpha_{i\pm 1j})\phi_{ij}$ a analogicky pre $\alpha_{ij\pm 1}$, dostaneme vzťahy

$$\alpha_{i\pm 1j} = \frac{\phi_{ij}}{\phi_{ij} - \phi_{i\pm 1j}} \quad \text{a} \quad \alpha_{ij\pm 1} = \frac{\phi_{ij}}{\phi_{ij} - \phi_{ij\pm 1}}. \quad (32)$$

Ak bod leží vedľa interfejsu v smere x aj zľava aj sprava a teda $\chi(x_i, y_j) \neq \chi(x_{i+1}, y_j)$ a $\chi(x_i, y_j) \neq \chi(x_{i-1}, y_j)$ tak kvôli upwind metóde zvolíme menšiu hodnotu medzi α_{i+1j} a α_{i-1j} . Analogicky ak bod leží vedľa interfejsu v smere y aj zhora aj zdola a teda $\chi(x_i, y_j) \neq \chi(x_i, y_{j+1})$ a $\chi(x_i, y_j) \neq \chi(x_i, y_{j-1})$ tak kvôli upwind metóde zvolíme menšiu hodnotu medzi α_{ij+1} a α_{ij-1} .

Rovnakým spôsobom pre bod ležiaci blízko interfejsu dostaneme vzťahy

$$\alpha_{i\pm 2j} = \frac{\phi_{i\pm 1j}}{\phi_{i\pm 1j} - \phi_{i\pm 2j}} \quad \text{a} \quad \alpha_{ij\pm 2} = \frac{\phi_{ij\pm 1}}{\phi_{ij\pm 1} - \phi_{ij\pm 2}}. \quad (33)$$

Pomocou týchto vzťahov si vieme vypočítať extrapolované hodnoty $d_{i\pm 1j}$ alebo $d_{ij\pm 1}$, ak sa bod (x_i, y_j) nachádza vedľa interfejsu alebo hodnoty $d_{i\pm 2j}$ a $d_{ij\pm 2}$ ak sa bod (x_i, y_j) nachádza blízko interfejsu. Pre extrapolované hodnoty v bodoch $(x_{i\pm 1}, y_j)$ a $(x_i, y_{j\pm 1})$ dostávame vzťahy

$$d_{i\pm 1j} = \frac{(\alpha_{i\pm 1j} - 1)d_{ij}}{\alpha_{i\pm 1j}} \quad \text{a} \quad d_{ij\pm 1} = \frac{(\alpha_{ij\pm 1} - 1)d_{ij}}{\alpha_{ij\pm 1}}. \quad (34)$$

Uvedieme si aj vzťahy pre výpočet extrapolovaných hodnôt v bodoch $(x_{i\pm 2}, y_j)$ a $(x_i, y_{j\pm 2})$,

$$d_{i\pm 2j} = \frac{(\alpha_{i\pm 2j} - 1)d_{i\pm 1j}}{\alpha_{i\pm 2j}} \quad \text{a} \quad d_{ij\pm 2} = \frac{(\alpha_{ij\pm 2} - 1)d_{ij\pm 1}}{\alpha_{ij\pm 2}}. \quad (35)$$

Pomocou týchto extrapolovaných hodnôt si vieme vypočítať koeficienty kvadratickej rovnice na výpočet hodnoty d_{ij} (41).

Na výpočet členov $\delta_x d_{ij}$ a $\delta_y d_{ij}$ vystupujúcich v kvadratickej rovnici (41) pre body, ktoré ležia vedľa interfejsu dostaneme nasledovné možnosti výpočtu $\delta_x d_{ij}$

$$\delta_x d_{ij} = \begin{cases} \frac{d_{ij}}{\alpha_{i-1j}} & \phi_{ij}\phi_{i-1j} < 0 \\ \frac{d_{ij}}{\alpha_{i+1j}} & \phi_{ij}\phi_{i+1j} < 0 \end{cases} \quad (36)$$

a $\delta_y d_{ij}$

$$\delta_y d_{ij} = \begin{cases} \frac{d_{ij}}{\alpha_{ij-1}} & \phi_{ij}\phi_{ij-1} < 0 \\ \frac{d_{ij}}{\alpha_{ij+1}} & \phi_{ij}\phi_{ij+1} < 0 \end{cases}. \quad (37)$$

4.1.2 Fast sweeping metóda

V tejto časti si ukážeme procedúru riešenia algebraického systému rovníc

$$(\delta_x d_{ij})^2 + (\delta_y d_{ij})^2 = 4h^2 . \quad (38)$$

Ak nahradíme v rovnici (38) členy gradientu $\delta_x d_{ij}$ a $\delta_y d_{ij}$ navrhovanými hodnotami (18) a (19) dostaneme rovnicu

$$[-3d_{ij} + 4d_{i\pm 1j} - d_{i\pm 2j}]^2 + [-3d_{ij} + 4d_{ij\pm 1} - d_{ij\pm 2}]^2 = 4h^2 , \quad (39)$$

Dôležitá myšlienka iteračnej fast sweeping metódy [4] je že sa na rovnicu (38) pozeráme ako na kvadratickú pričom v každej iterácii sa uvažuje ako neznáma len hodnota d_{ij} a ostatné susediace hodnoty $d_{i\pm 1j}$, $d_{ij\pm 1}$, $d_{i\pm 2j}$ a $d_{ij\pm 2}$ sa považujú za známe. Fast sweeping iteračná metóda [4] je založená na nelineárnej Gauss-Seidel metóde so štyrmi rôznymi usporiadaniami ("sweepings-"zametaniami") aktualizovaných neznámych hodnôt d_{ij} . Hodnoty počiatočného odhadu $d_{ij}^0 = L$ štandardne volíme dĺžku L štvorcovej oblasti Ω , ktorú považujeme za dostatočne veľkú hodnotu vzdialenosti, ktorú nemôže vzdialenostná funkcia nadobudnúť pri danom interfejsu a oblasti. Postupné iterácie d_{ij}^k dostaneme riešením kvadratickej rovnice (38) pomocou štyroch striedajúcich sa zametaní (usporiadaní)

1. $i = 1 : N, j = 1 : N$
2. $i = N : 1, j = 1 : N$
3. $i = N : 1, j = N : 1$
4. $i = 1 : N, j = N : 1$.

Upozorňujeme, že na dosiahnutie numerickej schémy druhého rádu neberieme novú aktualizovanú hodnotu vzdialenostnej funkcie len ak platí $d_{ij}^{k+1} < d_{ij}^k$ ako pri schéme prvého rádu ale pripúšťame aj možnosť $d_{ij}^{k+1} \geq d_{ij}^k$.

Ďalej si uvedieme možné podoby kvadratickej rovnice (38), ktoré môžu nastať v rôznych situáciách.

Ak sa bod (x_i, y_j) nenachádza hneď vedľa interfejsu ani v smere x ani v smere y a taktiež ani jeden zo susediacich bodov $(x_{i\pm 1}, y_j)$ alebo $(x_i, y_{j\pm 1})$ tak pre tento prípad dostávame rovnicu

$$\begin{aligned} a &= 18 \\ b &= -24d_{i\pm 1j} + 6d_{i\pm 2j} - 24d_{ij\pm 1} + 6d_{ij\pm 2} \\ c &= 16d_{i\pm 1j}^2 - 8d_{i\pm 1j}d_{i\pm 2j} + d_{i\pm 2j}^2 + 16d_{ij\pm 1}^2 - 8d_{ij\pm 1}d_{ij\pm 2} + d_{ij\pm 2}^2 - 4h^2 , \end{aligned} \quad (40)$$

kde a kvadratický koeficient, b je lineárny koeficient a c je konštantný koeficient. Riešenie tejto kvadratickej rovnice bude následovné,

$$d_{ij} = \begin{cases} \min(d_{xmin}, d_{ymin}) + h & b^2 - 4ac < 0 \\ \frac{-b + \sqrt{b^2 - 4ac}}{2a} & b^2 - 4ac \geq 0 \end{cases}, \quad (41)$$

kde $d_{xmin} = \min(d_{i-1j}, d_{i+1j})$ a $d_{ymin} = \min(d_{ij-1}, d_{ij+1})$.

Ak sa bod (x_i, y_j) nenachádza hneď vedľa interfejsu ani v smere x ani v smere y ale ak napr. interfejs leží medzi bodom $(x_{i\pm 1}, y_j)$ a $(x_{i\pm 2}, y_j)$ tak pre tento prípad si hodnotu $d_{i\pm 2j}$ vystupujúcu v (41) vypočítame pomocou metódy extrapolácie zo vzťahu (35). Analogicky pre y -ový smer.

Ak sa bod (x_i, y_j) nachádza hneď vedľa interfejsu v smere x ale nie v smere y , potom pre tento prípad dostávame rovnicu v tvare

$$\frac{4d_{ij}^2}{\alpha_{i\pm 1j}^2} + [-3d_{ij} + 4d_{ij\pm 1} - d_{ij\pm 2}]^2 = 4h^2, \quad (42)$$

čo je jednoduchá kvadratická rovnica s jedinou neznámou d_{ij} a jej koeficienty vypočítame ako

$$\begin{aligned} a &= 9 + \frac{4}{\alpha_{i\pm 1j}^2}, \\ b &= -24d_{ij\pm 1} + 6d_{ij\pm 2}, \\ c &= 16d_{ij\pm 1}^2 - 8d_{ij\pm 1}d_{ij\pm 2} + d_{ij\pm 2}^2 - 4h^2, \end{aligned} \quad (43)$$

kde a kvadratický koeficient, b je lineárny koeficient a c je konštantný koeficient. A riešenie bude vyzeráť,

$$d_{ij} = \begin{cases} \alpha_{i\pm 1j}h & b^2 - 4ac < 0 \\ \frac{-b + \sqrt{b^2 - 4ac}}{2a} & b^2 - 4ac \geq 0 \end{cases}. \quad (44)$$

Ak interfejs leží medzi bodom $(x_i, y_{j\pm 1})$ a $(x_i, y_{j\pm 2})$ tak pre tento prípad si hodnotu $d_{i\pm 2j}$ vystupujúcu v (47) vypočítame pomocou metódy extrapolácie zo vzťahu (35). Znamienka v konečnej diferencii vyberáme na základe (18).

Analogicky pre y -ový smer. Ak sa bod (x_i, y_j) nachádza hneď vedľa interfejsu v smere y , potom pre tento prípad dostávame rovnicu v tvare

$$\frac{4d_{ij}^2}{\alpha_{ij\pm 1}^2} + [-3d_{ij} + 4d_{ij\pm 1} - d_{ij\pm 2}]^2 = 4h^2, \quad (45)$$

čo je jednoduchá kvadratická rovnica s jedinou neznámou d_{ij} a jej koeficienty

vypočítame ako

$$\begin{aligned}
 a &= 9 + \frac{4}{\alpha_{ij\pm 1}^2} \\
 b &= -24d_{i\pm 1j} + 6d_{i\pm 2j} \\
 c &= 16d_{i\pm 1j}^2 - 8d_{i\pm 1j}d_{i\pm 2j} + d_{i\pm 2j}^2 - 4h^2, \tag{46}
 \end{aligned}$$

kde a kvadratický koeficient, b je lineárny koeficient a c je konštantný koeficient. Riešenie bude vyzerat',

$$d_{ij} = \begin{cases} \alpha_{ij\pm 1}h & b^2 - 4ac < 0 \\ \frac{-b + \sqrt{b^2 - 4ac}}{2a} & b^2 - 4ac \geq 0. \end{cases} \tag{47}$$

Ak interfejs leží medzi bodom $(x_{i\pm 1}, y_j)$ a $(x_{i\pm 1}, y_j)$ tak pre tento prípad si hodnotu $d_{ij\pm 2}$ vystupujúcu v (47) vypočítame pomocou metódy extrapolácie zo vzťahu (35). Znamienka v konečnej diferencii vyberáme na základe (19).

Taktiež sa môže bod (x_i, y_j) nachádzať interfejs hneď vedľa interfejsu v smere x aj v smere y , vtedy má naša rovnica tvar

$$\frac{d_{ij}^2}{\alpha_{i\pm 1j}^2} + \frac{d_{ij}^2}{\alpha_{ij\pm 1}^2} = h^2 \tag{48}$$

a jej riešenie je

$$d_{ij} = \frac{h}{\sqrt{\frac{1}{\alpha_{i\pm 1j}^2} + \frac{1}{\alpha_{ij\pm 1}^2}}}. \tag{49}$$

Použitím týchto vzťahov vieme navrhnúť fast sweeping metódu a riešiť iteračnú úlohu podľa [4].

5 Numerické riešenie lineárnej rovnice advekcie

V tejto časti sa budeme zaoberať numerickým riešením lineárnej rovnice advekcie v 2D priestore [7],

$$\partial_t \phi(x, y, t) + \vec{W}(x, y) \cdot \nabla \phi(x, y, t) = 0, \quad \phi(x, y, 0) = \phi^0(x, y), \quad (50)$$

kde $\vec{W}(x, y) = (U(x, y), V(x, y))$ a predstavuje rýchlostné vektorové pole a $U(x, y)$ je jeho vektorová zložka v smere x a $V(x, y)$ v smere y .

Rovnicu vieme prepísať do tvaru,

$$\partial_t \phi(x, y, t) + U(x, y) \partial_x \phi(x, y, t) + V(x, y) \partial_y \phi(x, y, t) = 0, \quad (51)$$

$$\phi(x, y, 0) = \phi^0(x, y). \quad (52)$$

Tento typ rovnice je časťou mnohých matematických modelov, ktoré sa používajú pri level set metódach a taktiež sledovanie pohybujúceho sa interfejsu.

5.1 Schéma vyššieho rádu

Uvedieme si numerickú schému vyššieho rádu na riešenie lineárnej rovnice advekcie. V práci [7] je opísaná semi-implicitná schéma pre 1D prípad, my sa však pokúsime opísať plne explicitnú schému pre 2D prípad. Opäť uvažujeme štvorcovú výpočtovú oblasť Ω s rovnomernou štvorčekovou sieťou. Veľkosť oblasti je $[0, L] \times [0, L]$, kde $L > 0$ a veľkosť priestorového kroku je $h = L/N$, kde $N > 0$. Bod siete budeme značiť ako (x_i, y_j) , kde $x_i = i \cdot h$ a $y_j = j \cdot h$. Uvažujeme časový interval $[0, T]$, kde $T > 0$ a veľkosť časového kroku $\tau = T/P$, kde $P > 0$. Bod času budeme značiť ako $t_n = n \cdot \tau$ a aproximovanú hodnotu exaktnej funkcie $\phi(x_i, y_j, t_n)$ budeme značiť ϕ_{ij}^n a teda $\phi(x_i, y_j, t_n) \approx \phi_{ij}^n$. Taktiež použijeme štandardnú indexáciu pre zložky vektorového poľa $V(x_i, y_j) = V_{ij}$ a $U(x_i, y_j) = U_{ij}$.

Plne explicitná schéma bude mať tvar,

$$\phi_{ij}^{n+1} = \phi_{ij}^n + \sum_{k=-2}^2 (\beta_{ijk}^x \phi_{i+1j}^n + \beta_{ijk}^y \phi_{ij+1}^n), \quad (53)$$

kde indexy x a y sa vzťahujú na priestorové premenné a β_{ijk}^x a β_{ijk}^y sú parametre metódy, ktoré určíme neskôr v texte.

Aproximovanú hodnotu gradientu $(\partial_x \phi(x_i, y_i, t_n), \partial_y \phi(x_i, y_i, t_n))$ budeme značiť $(\partial_x^{\kappa_x} \phi_{ij}^n, \partial_y^{\kappa_y} \phi_{ij}^n)$ a na jeho definíciu použijeme,

$$\begin{aligned} 2\partial_x^{\kappa_x} \phi_{ij}^n &= (1 - \kappa_x) \partial_x^- \phi_{ij}^n + (1 + \kappa_x) \partial_x^+ \phi_{ij}^n, \\ 2\partial_y^{\kappa_y} \phi_{ij}^n &= (1 - \kappa_y) \partial_y^- \phi_{ij}^n + (1 + \kappa_y) \partial_y^+ \phi_{ij}^n, \end{aligned} \quad (54)$$

kde ∂_x^- , ∂_x^+ , ∂_y^- a ∂_y^+ predstavujú štandardné konečné diferencie, kde znamienko $-$ indikuje spätnú diferenciu a znamienko $+$ indikuje doprednú diferenciu. Parametre $\kappa_x, \kappa_y \in [-1, 1]$ a sú vo všeobecnosti voliteľné. Pre názornosť uvedieme príklady voľby parametrov κ_x . Voľbou $\kappa_x = 1$ alebo $\kappa_x = -1$ dostaneme

konečnú diferenciu prvého rádu presnosti,

$$h\partial_x^{+1}\phi_{ij}^n = \phi_{i+1j}^n - \phi_{ij}^n \quad h\partial_x^{-1}\phi_{ij}^n = \phi_{ij}^n - \phi_{i-1j}^n .$$

Voľbou $\kappa_x = 0$ dostaneme konečnú diferenciu druhého rádu presnosti,

$$2h\partial_x^0\phi_{ij}^n = \phi_{i+1j}^n - \phi_{i-1j}^n$$

a analogicky pre κ_y .

Na odvodenie plne explicitnej schémy si vyjadríme presné riešenie pomocou Taylorovho rozvoju so stupňom p .

$$\phi(x, y, t + \tau) = \phi(x, y, t) + \sum_{m=1}^p \frac{1}{m!} \tau^m \partial_t^m \phi(x, y, t) + \mathcal{O}(\tau^{p+1}) . \quad (55)$$

Následne pomocou Lax-Wendroffovej procedúry zameníme všetky časové derivácie ϕ priestorovými deriváciami ϕ použitím východiskovej rovnice (50). Dostaneme vzťahy,

$$\partial_t \phi_{ij}^n = -U_{ij} \partial_x \phi_{ij}^n - V_{ij} \partial_y \phi_{ij}^n,$$

$$\begin{aligned} \partial_{tt} \phi_{ij}^n &= \partial_t (-U_{ij} \partial_x \phi_{ij}^n - V_{ij} \partial_y \phi_{ij}^n) = \\ &= -U_{ij} \partial_{xt} \phi_{ij}^n - V_{ij} \partial_{yt} \phi_{ij}^n = \\ &= -U_{ij}^n \partial_x (-U_{ij} \partial_x \phi_{ij}^n - V_{ij} \partial_y \phi_{ij}^n) - V_{ij} \partial_y (-U_{ij} \partial_x \phi_{ij}^n - V_{ij} \partial_y \phi_{ij}^n) , \end{aligned} \quad (56)$$

a použijeme ich v Taylorovom rozvoji (55) pre $p = 2$ a dostaneme,

$$\begin{aligned} \phi_{ij}^{n+1} &= \phi_{ij}^n - \tau U_{ij} \partial_x \phi_{ij}^n - \tau V_{ij} \partial_y \phi_{ij}^n + 0.5\tau^2 U_{ij} \partial_x (U_{ij} \partial_x \phi_{ij}^n + V_{ij} \partial_y \phi_{ij}^n) + \\ &\quad + 0.5\tau^2 V_{ij} \partial_y (U_{ij} \partial_x \phi_{ij}^n + V_{ij} \partial_y \phi_{ij}^n) + \mathcal{O}(\tau^3) . \end{aligned} \quad (57)$$

Aby sme dostali plne explicitnú κ schému na riešenie (50) treba správne zvoliť konečné diferencie v (57). Na dosiahnutie chyby druhého rádu treba členy násobené τ aproximovať aproximáciami druhého rádu a členy násobené $0.5\tau^2$ stačí aproximovať aproximáciami prvého rádu.

Upwind aproximačné vzťahy druhého rádu [7] vyzerajú nasledovne,

$$\begin{aligned} U_{ij} \partial_x \phi_{ij}^n &\approx [U_{ij}]^+ \partial_x^- (\phi_{ij}^n + 0.5h \partial_x^{\kappa_x} \phi_{ij}^n) + [U_{ij}]^- \partial_x^+ (\phi_{ij}^n - 0.5h \partial_x^{\kappa_x} \phi_{ij}^n) , \\ V_{ij} \partial_y \phi_{ij}^n &\approx [V_{ij}]^+ \partial_y^- (\phi_{ij}^n + 0.5h \partial_y^{\kappa_y} \phi_{ij}^n) + [V_{ij}]^- \partial_y^+ (\phi_{ij}^n - 0.5h \partial_y^{\kappa_y} \phi_{ij}^n) . \end{aligned}$$

Upwind aproximačné vzťahy prvého rádu s nezmiešanými priestorovými deriváciami [7] vyzerajú nasledovne,

$$\begin{aligned} U_{ij} \partial_x (U_{ij} \partial_x \phi_{ij}^n) &\approx ([U_{ij}]^+ \partial_x^- + [U_{ij}]^- \partial_x^+) (U_{ij} \partial_x^{\kappa_x} \phi_{ij}^n) , \\ V_{ij} \partial_y (V_{ij} \partial_y \phi_{ij}^n) &\approx ([V_{ij}]^+ \partial_y^- + [V_{ij}]^- \partial_y^+) (V_{ij} \partial_y^{\kappa_y} \phi_{ij}^n) . \end{aligned}$$

Upwind aproximačné vzťahy prvého rádu so zmiešanými priestorovými deriváciami vyzerajú nasledovne,

$$\begin{aligned} U_{ij}\partial_x(V_{ij}\partial_y\phi_{ij}^n) &\approx ([U_{ij}]^+\partial_x^- + [U_{ij}]^-\partial_x^+)(V_{ij}\partial_y^{\kappa_y}\phi_{ij}^n), \\ V_{ij}\partial_y(U_{ij}\partial_x\phi_{ij}^n) &\approx ([V_{ij}]^+\partial_y^- + [V_{ij}]^-\partial_y^+)(U_{ij}\partial_x^{\kappa_x}\phi_{ij}^n). \end{aligned}$$

Vo vyššie uvedených vzťahoch $[U_{ij}]^+ = \min\{0, U_{ij}\}$, $[U_{ij}]^- = \max\{0, U_{ij}\}$ a analogicky $[V_{ij}]^+ = \min\{0, V_{ij}\}$, $[V_{ij}]^- = \max\{0, V_{ij}\}$.

Zavedieme si značenie Courantových čísel pre x -ový smer,

$$C_{ij}^x = \frac{\tau U_{ij}}{h},$$

a analogicky pre y -ový smer,

$$C_{ij}^y = \frac{\tau V_{ij}}{h}.$$

Potom numerická schéma (57) pre napr. $\kappa^x = 0$ a $U_{ij} > 0$ a $\kappa^y = 0$ a $V_{ij} < 0$, bude vyzerať nasledovne,

$$\begin{aligned} \phi_{ij}^{n+1} &= \phi_{ij}^n - C_{ij}^x(\phi_{ij}^n - \phi_{i-1j}^n) - C_{ij}^y(\phi_{ij+1}^n - \phi_{ij}^n) - C_{ij}^x 0.25((\phi_{i+1j}^n - \phi_{i-1j}^n) - (\phi_{ij}^n - \phi_{i-2j}^n)) + \\ &+ C_{ij}^y 0.25((\phi_{ij+2}^n - \phi_{ij}^n) - (\phi_{ij+1}^n - \phi_{ij-1}^n)) - [C_{ij}^x]^2 0.25(-(\phi_{i+1j}^n - \phi_{i-1j}^n) + (\phi_{ij}^n - \phi_{i-2j}^n)) - \\ &- [C_{ij}^y]^2 0.25(-(\phi_{ij+2}^n - \phi_{ij}^n) + (\phi_{ij+1}^n - \phi_{ij-1}^n)) - C_{ij}^x C_{ij}^y 0.25(-(\phi_{ij+2}^n - \phi_{ij}^n) + (\phi_{ij+1}^n - \phi_{ij-1}^n)) - \\ &- C_{ij}^y C_{ij}^x 0.25(-(\phi_{i+2j}^n - \phi_{ij}^n) + (\phi_{i+1j}^n - \phi_{i-1j}^n)). \quad (58) \end{aligned}$$

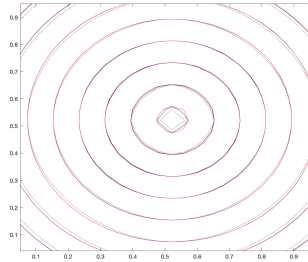
Takýmto spôsobom dostávame plne explicitnú schému druhého rádu, podmienične stabilnú pre $C_{ij}^x, C_{ij}^y \in [-1, 1]$ a $\kappa_x, \kappa_y \in [-1, 1]$.

6 Numerické experimenty

V tejto časti ukážeme zrealizované experimenty z niektorých tematických okruhov. Všetky experimenty boli realizované v prostredí Matlab a funkcie boli naprogramované samostatne.

6.1 Experiment č.1

Cieľom experimentu bolo zistiť resp. ukázať vyšší rád konverencie numerickej schémy vyššieho rádu na výpočet vzdialenostnej funkcie v porovnaní s numericou schémou prvého rádu. Počítali sme vzdialenostnú funkciu ku krúžku s polomerom 0.25 a stredom v bode $(0.5, 0.5)$. Veľkosť oblasti je $[0, 1] \times [0, 1]$.



Obr. 11: Exaktná vzdialenostná funkcia (červená), vzdialenostná funkcia vypočítaná numericou schémou prvého rádu (sivá) a vzdialenostná funkcia vypočítaná numericou schémou druhého rádu (čierna) pre $h = 0.04$.

h	$u_h - u$	EOC
0.04	0.000564	-
0.02	0.000139	2.0280
0.01	0.000038	1.8494
0.0050	0.000010	1.9359
0.0025	0.000003	1.8874

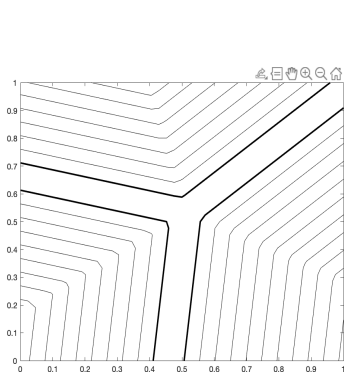
Tabuľka 1: Tabuľka hodnôt pre numericú schému druhého rádu

h	$u_h - u$	EOC
0.04	0.005730	-
0.02	0.002726	1.0508
0.01	0.001328	1.0267
0.0050	0.000654	1.0149
0.0025	0.000324	1.0079

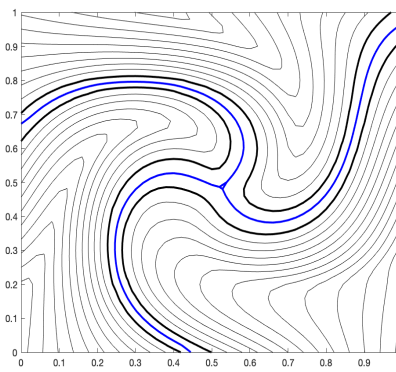
Tabuľka 2: Tabuľka hodnôt pre numerickú schému prvého rádu

6.2 Experiment č.2

Cieľom experimentu bolo zistiť resp. ukázať vyšší rád konvergencie numerickej schémy vyššieho rádu na riešenie lineárnej rovnice advekcie v porovnaní s numericou schémou prvého rádu a taktiež ukázať vplyv presnosti numerickej schémy na rekonštrukciu interfejsu. Z experimentov sme zistili, že metóda druhého rádu dáva výrazne lepšie výsledky rekonštrukcie interfejsu ako metóda prvého rádu. Počiatočná podmienka bola vzdialenostná funkcia k interfejsu v tvare písmena Y a tzv. single vortex vektorové pole definované ako $\vec{W} = 2 \cos(\frac{\pi t}{8})(-\sin^2(\pi x) \sin(\pi y) \cos(\pi y), \sin^2(\pi y) \sin(\pi x) \cos(\pi x))$. Okrajová podmienka je rovná hodnotám počiatočnej podmienky. Veľkosť oblasti je $[0, 1] \times [0, 1]$.

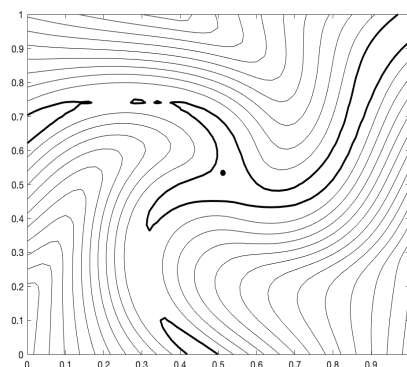


(a) Počiatočná podmienka

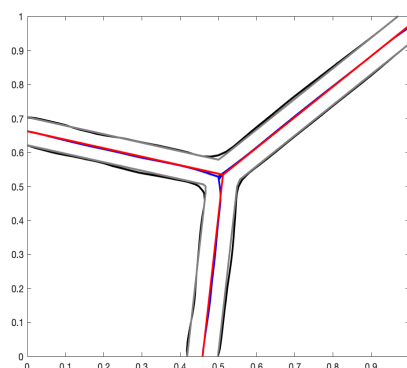


(b) Zrekonštruovaný interfejs Γ_V (modrá), Γ_ϵ (čierna) a level set funkcia v čase najväčšej deformácie vypočítaná pomocou schémy druhého rádu

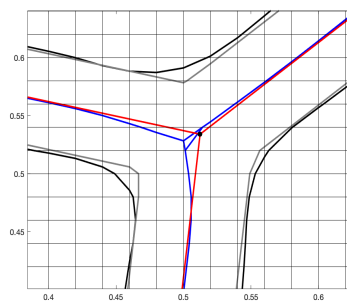
Obr. 12



(a) Interfejs Γ_ϵ a level set funkcia v čase najväčšej deformácie vypočítaná pomocou schémy prvého rádu



(b) Počiatočný interfejs Γ (červená), zrekonštruovaný interfejs Γ_V (modrá), Γ_ϵ (čierna) v konečnom čase druhého rádu



(c) Priblíženie k trojnému bod (čierny krúžok)

Obr. 13

h	$u_h - u$	EOC
0.04	0.022973	-
0.02	0.006061	1.8953
0.01	0.001614	1.8776

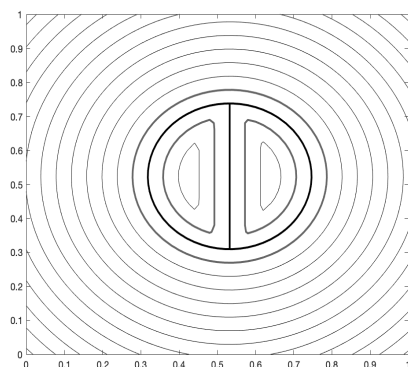
Tabuľka 3: Tabuľka hodnôt pre numerickú schému druhého rádu

h	$u_h - u$	EOC
0.04	0.025668	-
0.02	0.012425	1.0329
0.01	0.006459	0.9619

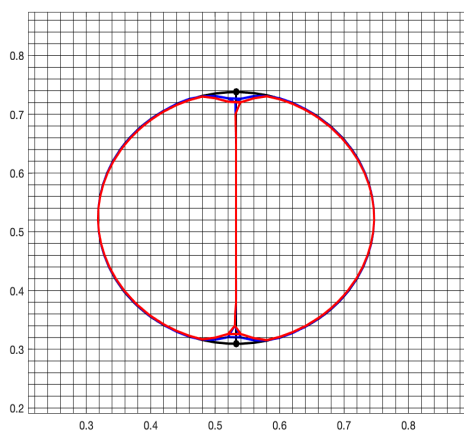
Tabuľka 4: Tabuľka hodnôt pre numerickú schému prvého rádu

6.3 Experiment č.3

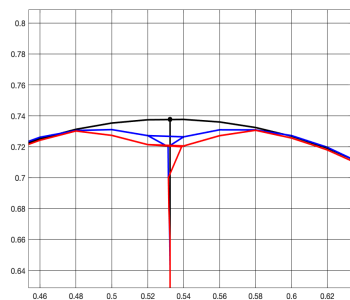
V tomto experimente sme sa pokúsili zrekonštruovať viacfázový interfejs Γ_V pomocou VIIM metódy. Porovnáme výsledky rekonštrukcie interfejsu pomocou numerickej schémy prvého rádu a druhého rádu na výpočet vzdialenostných funkcií d_1, d_2, d_3 . Z experimentov sme zistili, že numerická schéma druhého rádu dáva významne lepšie výsledky rekonštrukcie interfejsu. Náš interfejs obsahuje dva trojné body, je to krúžok so stredom v $(0.5, 0.5)$ a polomerom $(0.2, 0.2)$ rozdelený zvislou čiarou na dve časti. Priestorový krok je 0.02 a veľkosť oblasti je $[0, 1] \times [0, 1]$.



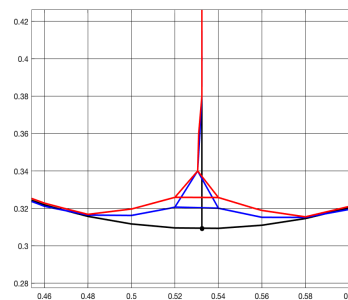
(a) Pôvodný interfejs Γ (čierna), interfejs Γ_ϵ (sivý) a vzdialenostná funkcia



(b) Pôvodný interfejs Γ (čierna), zrekonštruovaný interfejs Γ_V (modrá) pomocou schémy 2 rádu a zrekonštruovaný interfejs Γ_V (červená) pomocou schémy 1 rádu



(c) Príblíženie k hornému trojnému bodu

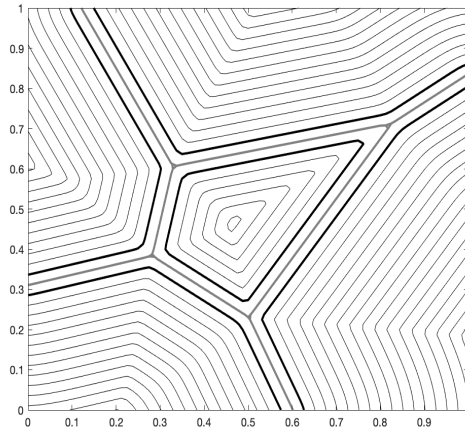


(d) Príblíženie k dolnému trojnému bodu

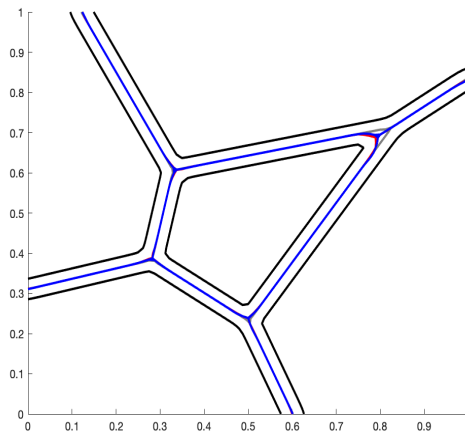
Obr. 14

6.4 Experiment č.4

V tomto experimente sme sa pokúsili zrekonštruovať netriviálny viacfázový interfejs Γ_V pomocou VIIM metódy. Porovnáme výsledky rekonštrukcie interfejsu pomocou numerickej schémy prvého rádu a druhého rádu na výpočet vzdialenostných funkcií d_1, d_2, d_3 . Z experimentov sme zistili, že numerická schéma druhého rádu dáva významne lepšie výsledky rekonštrukcie interfejsu. Náš interfejs považujeme za netriviálny Voronoi Diagram obsahujúci až 4 trojné body. Priestorový krok je 0.01 a veľkosť oblasti je $[0, 1] \times [0, 1]$.

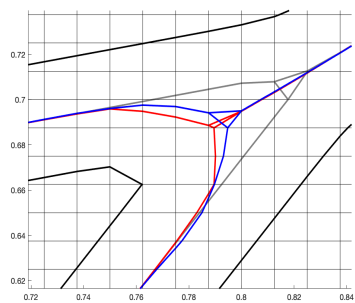


(a) Pôvodný interfejs Γ (čierna), interfejs Γ_ϵ (sivá) a vzdialenosťná funkcia

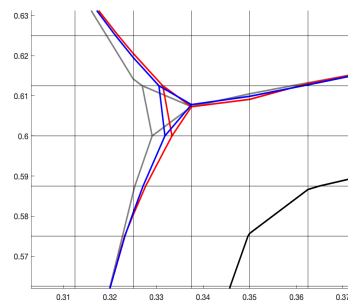


(b) Pôvodný interfejs Γ (čierna), zrekonštruovaný interfejs Γ_V (modrá) pomocou schémy 2 rádu a zrekonštruovaný interfejs Γ_V (červená) pomocou schémy 1 rádu

Obr. 15



(a) Priblíženie k hornému pravému trojnému bodu s ostrým uhlom



(b) Priblíženie k hornému ľavému trojnému bodu s tupým uhlom

Obr. 16

7 Záver

V práci sme sa zaoberali rekonštrukciou interfejsu zo vzdialenostnej funkcie bez znamienka a uviedli sme viaceré možnosti rekonštrukcie. Viacfázový interfejs sa vyskytuje často v praxi a jeho rekonštrukcia má významné aplikácie pri sledovaní dynamického vývoja interfejsu.

Implementovali sme rekonštrukciu rozhrania pomocou VIIM metódy [3] a na vylepšenie výsledkov rekonštrukcie sme navrhli numerickú schému 2. rádu na výpočet vzdialenostnej funkcie s využitím metódy extrapolácie [1,6] a konečnej diferencie 2. rádu [5]. Porovnali sme vplyv výpočtu vzdialenostnej funkcie numerickými schémami 1. a 2. rádu na rekonštrukciu interfejsu a zistili významný vplyv na výsledok rekonštrukcie.

Implementovali sme plne explicitnú schému 2. rádu presnosti na výpočet lineárnej rovnice advekcie s využitím konečných diferencií v [7]. Porovnali sme vplyv výpočtu numerickými schémami 1. a 2. rádu na sledovanie pohybujúceho sa interfejsu a zistili sme že to má dôležitý dopad na výsledok rekonštrukcie interfejsu a numerická schéma vyššieho rádu je jednoznačne lepšia. Navrhli sme používať vzdialenostnú funkciu k exaktným priesečníkom interfejsu [2] ako počiatočnú podmienku na riešenie lineárnej rovnice advekcie, ktorá dáva lepšie výsledky rekonštrukcie v blízkosti trojného bodu.

Navrhli sme iný prístup ako [3] na určenie hodnôt indikačnej funkcie v ďalšom časovom kroku, ktorý považujeme za elegantnejší a menej výpočtovo náročný.

Citácie

- [1] Róbert Blaschke. *Numerický výpočet funkcie času príchodu pre hranicu lesného požiaru*, Thesis, Slovak University of Technology in Bratislava, 2016.
- [2] N. Gajdošová *Numerické metódy na určenie skeletonu vzdialenostnej funkcie*. Thesis, Slovak University of Technology in Bratislava, 2019.
- [3] R.I. Saye and J.A. Sethian Analysis and applications of the Voronoi Implicit Interface Method. *J. Comp. Phys.*, 231 (18), p. 6051-6085, 2012.
- [4] H. Zhao. A fast sweeping method for eikonal equations. *Math. Comput.*, 74(250), p. 603-627, 2005.
- [5] J.A. Sethian Fast Marching Methods. *SIAM Review*,41(2), p. 199-235, 1999.
- [6] P. Frolkovic, K. Mikula, and J. Urbán Distance function and extension in normal direction for implicitly defined interfaces. *Discrete and continuous dynamical systems-series S*,8(5), p. 871, 2015.
- [7] P. Frolkovic, K. Mikula *High order semi-implicit schemes for linear advection equation on Cartesian grids with numerical stability analysis*, Preprint arXiv:1611.04153, 2016.
- [8] R.I. Saye, R. Ian, J.A. Sethian *The Voronoi Implicit Interface Method with Applications to Multiphase Fluid Flow and Multiscale Modelling of Foam Dynamics*, UC Berkeley, 2013.
- [9] Frolkovic, P., Gajdošová, N., Gärttner, S., Ray, N. *Voronoi implicit interface method for geometry evolution of two minerals with applications in reactive porous media Proceedings of ALGORITMY*,pp. 121-130, 2020.