SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
STAVEBNÁ FAKULTA

NUMERICKÉ MODELOVANIE VOĽNEJ HLADINY PRÚDENIA VODY PÓROVITÝM PROSTREDÍM

DIPLOMOVÁ PRÁCA

SVF-5343-26555

Študijný program:	Matematicko–počítačové modelovanie
Číslo a názov študijného odboru:	9.1.9 aplikovaná matematika
Školiace pracovisko:	Katedra matematiky a deskriptívnej geometrie
Vedúci práce:	RNDr. Peter Frolkovič, PhD.

Bratislava 2010 **Bc. Petra Zacharovská**

Poďakovanie

Ďakujem vedúcemu práce RNDr. Petrovi Frolkovičovi PhD. za cenné rady, odbornú pomoc, konzultácie a jeho trpezlivosť. Moje poďakovanie patrí aj všetkým, ktorí mi akýmkoľvek spôsobom pomohli pri vypracovávaní diplomovej práce.

Súhrn

Zacharovská, Petra, Bc.: Numerické modelovanie voľnej hladiny prúdenia vody pórovitým prostredím
Diplomová práca. Slovenská technická univerzita v Bratislave, Stavebná fakulta, Katedra matematiky a deskriptívnej geometrie. Vedúci práce: RNDr. Peter Frolkovič, PhD., Bratislava 2010, 43 s. a príloha CD nosič.

Práca sa zaoberá numerickým modelovaním prúdenia vody s voľnou hladinou pórovitým prostredím. Budeme používať originálnu formuláciu pomocou level set metódy, v ktorej je poloha vodnej hladiny daná implicitne ako nulová izočiara, alebo plocha takzvanej level set funkcie. Výhodou tejto metódy je, že numerické riešenie môžeme riešiť na pevnej rovnomernej sieti. V predchádzajúcich prácach s podobnou tématikou sa tento problém riešil na pohyblivej sieti. Výpočty budeme realizovať použitím metódy konečných diferencií. Dôraz práce je kladený na skúmanie a vývoj vhodných algoritmov pre tento typ úloh. Opíšeme matematický model problému a použité metódy, pričom zameriavať sa budeme na ich počítačovú realizáciu a prezentáciu testovacích príkladov. Porovnávať budeme dve varianty riešenia toku podzemnej vody, pričom ukážeme výhodnosť jednej z nich. Našou snahou je získať ustálený stav prúdenia podzemnej vody.

Kľúčové slová: numerické modelovanie, voľná hladina, level set, upwind, podzemná voda, metóda konečných diferencií

Abstract

Zacharovská, Petra, Bc.: Numerical modeling of groundwater flow in porous media

Diploma work. Slovak Technical University, Civil Engineering, Department of Mathematics and Constructive Geometry. Tutor: RNDr. Peter Frolkovič, PhD., Bratislava 2010, 43 s. and attachment CD medium

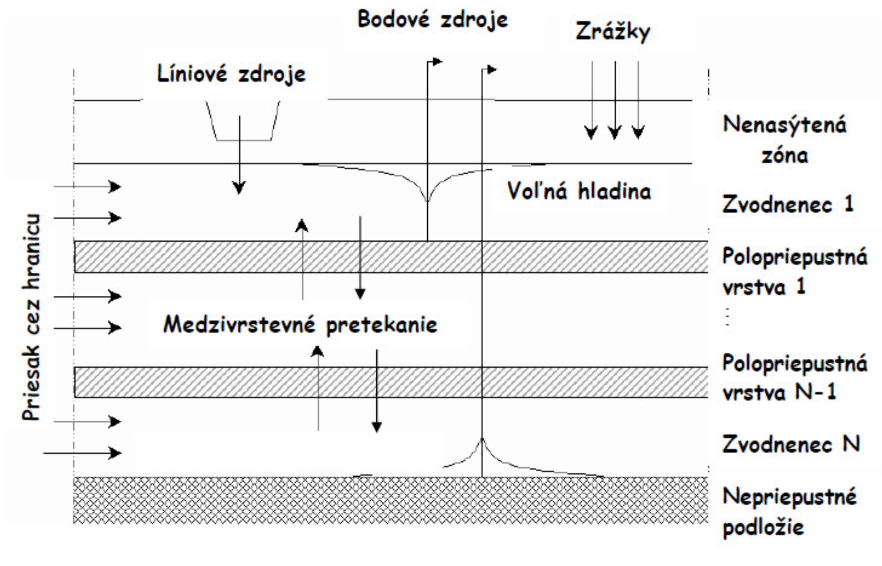
This work deals with numerical modeling of groundwater flow with dynamic table in porous media. We will use a novel level set formulation. The dynamic groundwater table is given implicitly as a zero set of the so called level set function. The advantage of this method is that we can solve the numerical solution on a fixed grid. In previous works this problem was solved on moving grids. In the computations we will use typically finite difference method. The emphasis of this work is on the study and development of the suitable algorithms for this task. We describe mathematical model and used methods with an emphasis on the computational realization and presentation of test examples. We will compare two variants of solving of the groundwater flow and we will determine the advantage of one of them. Our aim is to get steady state of dynamic groundwater table.

Kľúčové slová: numerical modeling, dynamic groundwater table, level set method, upwind, groundwater flow, finite difference method

Obsah

1 Úvod do problematiky

Voda patrí medzi jednu z najdôležitejších substancií na zemi. Skladá sa z povrchovej a podzemnej vody [11]. Voda, ktorá sa nachádza a pohybuje v pevných horninách a sypkých uloženinách, sa nazýva podzemná voda. Väčšinou vzniká vsakovaním (infiltráciou) zrážkovej vody do zemskej kôry, menej kondenzáciou vodných pár v horninách. Priestory, v ktorých sa vyskytuje podzemná voda, sa nazývajú póry a pukliny. Prostredie, ktoré je vyplnené zrnitým materiálom (prevažne pieskom) sa nazýva pórovité prostredie a tvorí ho prevažne filtračná vrstva [11] (obrázok 1).

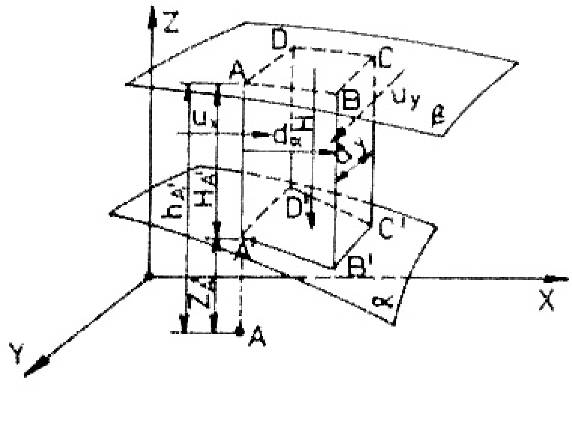


Obrázok 1: Pórovité prostredie s niekoľkými zvodnenými vrstvami [2]

Úroveň v pórovitom prostredí, po ktorú sú póry a pukliny vyplnené úplne a súvisle vodou, sa nazýva hladina podzemnej vody. Jej polohu potrebujeme často pre praktické účely poznať. Táto hladina v dôsledku pohybu podzemnej vody vytvára rôzne plynule zakrivené plochy (podobne ako terén). Keď sú priepustné vrstvy naklonené, alebo korytovito prehnuté, hladina podzemnej vody je v nich pod hydrostatickým tlakom a na hladine platí, že sa tento tlak rovná atmosferickému tlaku [11].

Cieľom našej práce je numericky sledovať pohyb voľnej hladiny pomocou takzvanej level set metódy. V mnohých doterajších prácach sa voľná hladina numericky modelovala pomocou pohyblivej siete. Hlavný rozdiel oproti tomuto prístupu je, že level set metóda nám umožní riešiť tento problém na fixnej oblasti, pričom predpokladáme, že voľná hladina je v jej vnútri. Hladina podzemnej vody bude definovaná ako nulové hodnoty nejakej level set funkcie. Na riešenie level set rovníc v práci použijeme metódu konečných diferencií.

Netriviálnou časťou numerickej diskretizácie v predkladanej práci je použitie upwind metód na riešenie hyperbolických diferenciálnych rovníc. Využili sme ich predovšetkým na modelovanie šírenia neznámej informácie do celej výpočtovej oblasti od časti jej hranice, kde je informácia známa, pričom smer šírenia musí byť daný tokom nejakého rýchlostného poľa. Tento postup je v našej level set formulácii nutný, pretože riešenie prúdenia podzemnej vody je dané len na časti výpočtovej oblasti (pod voľnou hladinou), a umožňuje nám tak riešiť túto úlohu s pohyblivou hranicou na fixnej oblasti.



Obrázok 2: Voľná hladina v 3D [10]

Na obrázku 2 môžeme vidieť jednu z najčastejších modelových situácií v tejto problematike. V našej práci sa ňou tiež zaoberáme v dvojrozmernom analogickom prípade. Na porovnanie používame riešenie získané na základe Dupuitovho princípu [10], kedy hydraulické riešenie úlohy vychádza z predpokladu, že jednotlivé prúdnice sú vodorovné, prietokové prierezy sú zvislé a rýchlosti v nich sú konštantné. Tento predpoklad je však vhodný používať len v prípadoch, kde zakrivenie prúdnic a tým odchýlka ekvipotenciálnych plôch od roviny nie je príliš veľká. Vzťah platný na riešenie voľnej hladiny s uvážením Dupuitovho princípu je

$$H = \sqrt{H_0^2 - \frac{2qx}{K}}, \tag{1}$$

kde x je vzdialenosť od rezu so známou hĺbkou H_0 a q je merný prietok [10]. Koeficient hydraulickej vodivosti je reprezentovaný K.

V matematickom modely, ktorý je prezentovaný v tejto práci, používame rýchlosť prúdenia, ktorá nemusí byť konštantná v prietokovom priereze. Obidva prístupy, t.j. s a bez Dupuitovho princípu, sú porovnané na konkrétnom príklade.

2 Matematický model problému

Naša výpočtová oblasť je pre jednoduchosť jednotkový štvorec v smere x a v smere z a označíme si ju $D \subset \mathcal{R}^2$. Krivku hladiny podzemnej vody v určitom čase $t \geq 0$ označíme $\Gamma(t) \subset D$. Dôležité je zadefinovať si oblasť pod hladinou $\Omega(t) \subset D$, pričom budeme modelovať prúdenie vody len v tejto podoblasti, kde sa voda nachádza. Oblasť nad hladinou označíme $\Omega^{out}(t) = D \setminus \Omega(t)$.

2.1 Tlak podzemnej vody

Na určenie toku podzemnej vody potrebujeme najprv vypočítať tlak $p = p(x, z, t)$, ktorý sa na povrchu voľnej hladiny rovná atmosferickému tlaku [11], čo vyjadruje podmienka, že sa na hladine rovná nule. Vzťahy pre tlak sú dané rovnicou spojitosti (2) a Darcyho zákonom (3) v tvare

$$\nabla \cdot \vec{q} = 0 \tag{2}$$

$$\vec{q} = -K\nabla(p + \rho g z). \tag{3}$$

Parameter K predstavuje koeficient hydraulickej vodivosti, g je známe gravitačné zrýchlenie a ρ je hustota vody. Spojením rovníc (2) a (3) dostaneme známu Laplaceovu rovnicu pre tlak

$$-\Delta p = 0. \tag{4}$$

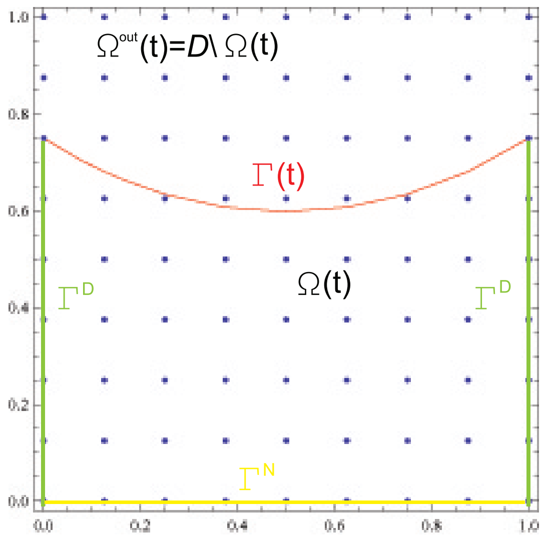
Pre rovnicu (4) zvolíme okrajové podmienky

$$p(x, z, t) = 0, \quad (x, z) \in \Gamma(t), \tag{5}$$

$$p(x, z, t) = p^D(x, z), \quad (x, z) \in \Gamma^D \tag{6}$$

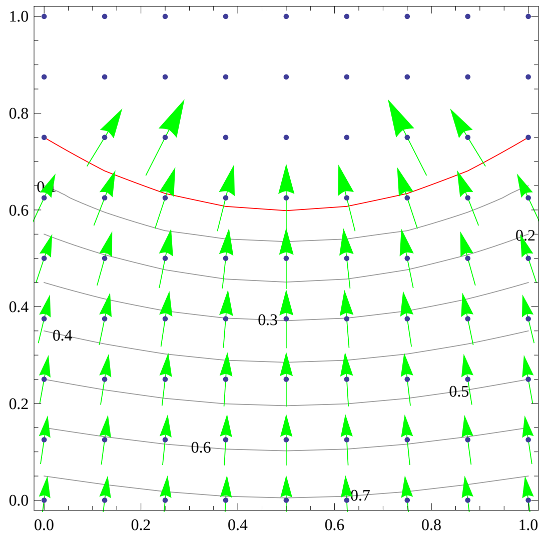
$$\vec{n}(x, z) \cdot \nabla p(x, z, t) = p^N(x, z), \quad (x, z) \in \Gamma^N, \tag{7}$$

kde $\vec{n}$ je normála na Γ^N. Príklad definície jednotlivých hraníc oblasti je zobrazený na nasledujúcom obrázku 3.



Obrázok 3: Definícia oblasti a častí jej hranice

Riešením Laplaceovej rovnice (4) s okrajovými podmienkami (5) dostávame tlak, ktorý je definovaný len na podoblasti $\Omega(t) \subset D$. Podobne môžeme pomocou rovnice (3) vypočítať rýchlosť prúdenia, ktorá takisto nie je známa na celej oblasti D. Príklad riešenia tlaku podzemnej vody môžeme vidieť zobrazený na obrázku 4 spolu s rýchlostným poľom toku podzemnej vody.



Obrázok 4: Ilustratívny numerický výsledok tlaku a toku podzemnej vody

2.2 Rovnica advekcie

Na opis dynamickej polohy voľnej hladiny $\Gamma(t)$ použijeme level set formuláciu. Nulové hodnoty nejakej level set funkcie

$$\varphi^0 = \varphi(x, z)$$

budú reprezantovať počiatočnú polohu voľnej hladiny, čo môžeme zapísať v tvare

$$\Gamma(0) = \{(x, z) \in D, \varphi^0(x, z) = 0\}.$$

Podobným spôsobom sa dá vyjadriť oblasť definovaná pod hladinou

$$\Omega(0) = \{(x, z) \in D, \varphi^0(x, z) < 0\}.$$

Naša práca sa zaoberá nájdením ustáleného stavu voľnej hladiny. Najdôležitejším krokom našej level set formulácie je nájsť rýchlosť $\vec{V} = \vec{V}(x, z, t)$, $(x, z) \in D$, ktorou sa bude voľná hladina pohybovať. Táto rýchlosť musí byť definovaná na celej oblasti a jej hodnoty $\vec{V}(x, z, t)$ pre $(x, z) \in \Gamma(t)$ musia zodpovedať fyzikálnemu opisu pohybu voľnej hladiny. Keď je rýchlosť $\vec{V}(x, z, t)$ známa, potom zmena voľnej hladiny je daná rovnicou advekcie v tvare

$$\partial_t \varphi + \vec{V} \cdot \nabla \varphi = 0, \tag{8}$$

$$\varphi(x, z, 0) = \varphi^0(x, z). \tag{9}$$

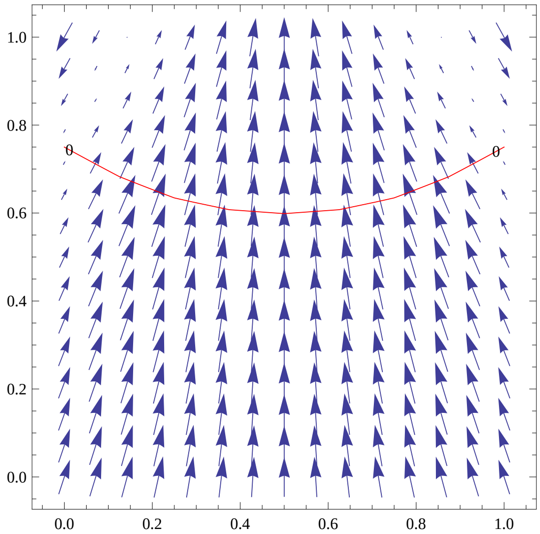
Takto je daná implicitná závislosť polohy hladiny podzemnej vody od času t, pretože platí [12, 13] pre riešenie $\varphi = \varphi(x, z, t)$ že

$$\Gamma(t) = \{(x, z) \in D, \varphi(x, z, t) = 0\}$$

a

$$\Omega(t) = \{(x, z) \in D, \varphi(x, z, t) < 0\}.$$

Tento čas t budeme nazývať reálnym časom pohybu voľnej hladiny. Na obrázku 5 je zobrazený príklad rýchlostného poľa $\vec{V}(x, z, t)$, ktorým sa bude pohybovať voľná hladina vyznačená na obrázku červenou čiarou.



Obrázok 5: Voľná hladina a rýchlostné pole $\vec{V}(x, z, t)$

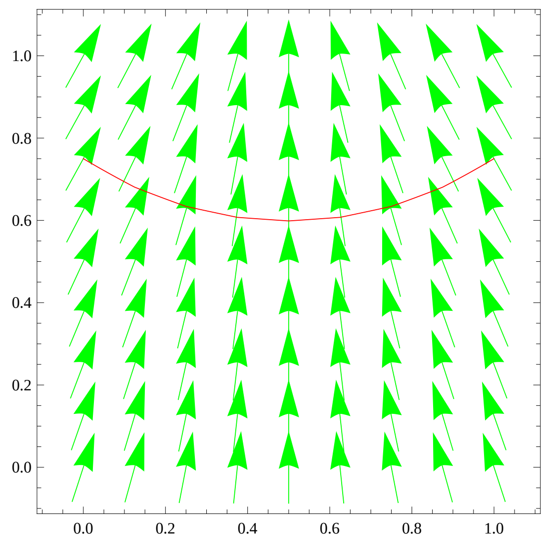
2.3 Eikonalova rovnica

Na definovanie hladiny podzemnej vody môžeme použiť ľubovoľnú level set funkciu, u ktorej len požadujeme, aby jej nulové hodnoty predstavovali voľnú hladinu. Neskôr budeme potrebovať normálu $\vec{N}$ ku hladine, keďže pohyb voľnej hladiny nás bude zaujímať práve v tomto smere, pozri aj obrázok 5. Výhodou level set formulácie je, že normálový vonkajší vektor na hladine sa dá vyjadriť (ak existuje gradient level set funkcie φ) pomocou

$$\vec{N}(x, z, t) = \frac{\nabla\varphi(x, z, t)}{|\nabla\varphi(x, z, t)|}, \quad (x, z) \in \Gamma(t).$$

Tento smer môžeme vidieť zobrazený na obrázku 6.

Funkciu φ budeme nazývať level set funkciou, ktorá definuje našu voľnú hladinu. Najbežnejšie sa používa takzvaná signed distance funkcia [12, 13] s vlastnosťou $|\nabla\varphi| = 1$ na celej výpočtovej oblasti. Ak φ má túto vlastnosť na začiatku riešenia, tá sa počas výpočtu môže stratiť. Odstránenie tohto problému nás priviedlo k myšlienke, že pri našich výpočtoch

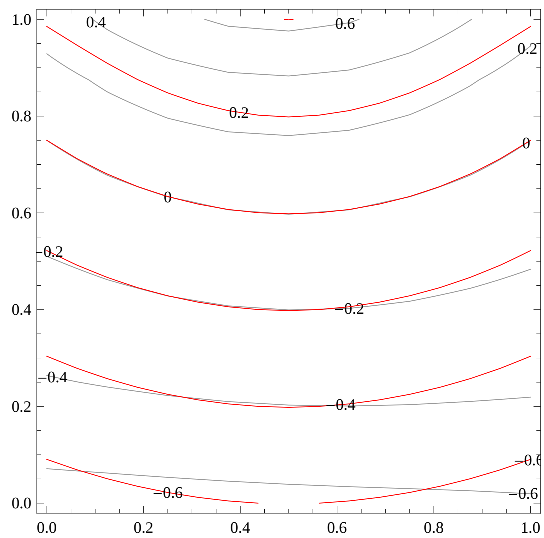


Obrázok 6: Normalový vektor ku hladine

budeme okrem level set funkcie φ, ktorá definuje voľnú hladinu, používať i druhú funkciu ϕ, pre ktorú bude platiť eikonalova rovnica

$$|\nabla \phi(x,z,t)| = 1,\ (x,z) \in D,\ \phi(x,z,t) = 0, (x,z) \in \Gamma(t),$$

pozri aj obrázok 7.



Obrázok 7: Signed distance funkcia (červené čiary) a level set funkcia (šedé čiary), ktorá definuje voľnú hladinu

Funkcia $\phi(x,z,t)$ je signed distance funkcia a získame ju hľadaním stacionárneho riešenia rovnice [4, 5, 6] v tvare

$$\partial_s \Phi(x,z,s) + |\nabla \Phi(x,z,s)| = 1 \quad (x,z) \in \Omega^{out}(t) \tag{10}$$

$$\partial_s \Phi(x,z,s) - |\nabla \Phi(x,z,s)| = -1 \quad (x,z) \in \Omega(t), \tag{11}$$

s počiatočnou podmienkou

$$\Phi(x,z,0) = \phi(x,z,t), (x,z) \in D$$

a s okrajovými podmienkami pre

$$(x,z) \in \Gamma(t), s > 0,\ \Phi(x,z,s) = 0.$$

Pre nejaký konečný čas $S > 0$ sa dosiahne stacionárne riešenie [4, 8] a môžeme zobrať $\phi(x,z,t) = \Phi(x,z,S)$. Treba poznamenať, že časová premenná s nepredstavuje reálny čas.

2.4 Extrapolovaná rýchlosť

V tejto kapitole budeme predpokladať, že pohyb voľnej hladiny $\Gamma(t)$ v smere $\vec{N}$ je daný nejakou funkciou

$$\overline{f} = \overline{f}(x,z,t),\ (x,z) \in \Gamma(t),$$

ktorá je definovaná len na hladine. Extrapolovanú rýchlosť $f = f(x,z,t)$, ktorá sa šíri v smere $\vec{N}$ a od ktorej požadujeme, že

$$f(x,z,t) = \overline{f}(x,z,t),\ (x,z) \in \Gamma(t),$$

a

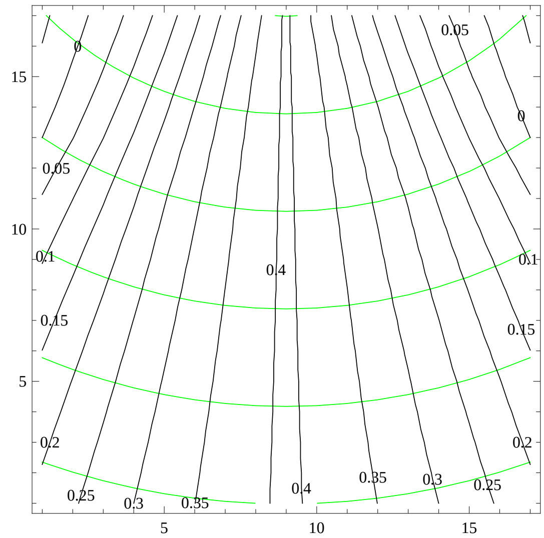
$$\nabla \phi \cdot \nabla f = 0,$$

získame riešením rovnice advekcie [1] v tvare

$$\partial_s F(x,z,s) + \nabla \phi(x,z,t) \cdot \nabla F(x,z,s) = 0 \tag{12}$$

a s okrajovou podmienkou $F(x,z,s) = \overline{f}(x,z,t)$ na $\Gamma(t)$.

Znova nám tu platí, že časová premenná s nepredstavuje reálny čas pohybu voľnej hladiny. Pre nejaký konečný čas $S > 0$ zoberieme $f(x, z, t) = F(x, z, S)$. Ilustratívny výsledok, kde izočiary extrapolovanej rýchlosti sú kolmé na izočiary signed distnace funkcie, vidíme na obrázku 8.



Obrázok 8: Extrapolovaná rýchlosť a signed distance funkcia

2.5 Opis matematického modelu

V tejto kapitole zhrnieme náš matematický model. Postup riešenia začína vyriešením tlaku podzemnej vody z rovnice (4). Keď máme definovaný tlak, môžeme určiť tok $\vec{q}$ pomocou (3), ktorý je platný aj na voľnej hladine. Následne môžme definovať pohyb voľnej hladiny pomocou rýchlosti $\overline{f} = \overline{f}(x, z, t)$, ktorá je daná len na voľnej hladine, vzťahom

$$\overline{f} = \frac{1}{\theta}\vec{N} \cdot (\vec{q} - A\vec{e}_z), \tag{13}$$

kde parameter θ vyjadruje pórovitosť prostredia, $\vec{N}$ je vonkajšia normála ku hladine a $A = A(x, z, t), (x, z) \in \Gamma(t)$ je daná rýchlosť nárastu [3].

Ďalej je potrebné vyrátať signed distance funkciu pomocou rovníc (10) - (11), ktorá nám určí smer šírenia informácie z voľnej hladiny, daný normálou ku voľnej hladine $\vec{N}$. Ďalším krokom programu je vyrátanie extrapolovanej rýchlosti f, kde pomocou rovnice (12) rozšírime túto informáciu do celej výpočtovej oblasti v smere danej normály $\vec{N}$. Následne z extrapolovanej rýchlosti f určíme rýchlosť $V(x, z, t)$ pre pohyb voľnej hladiny

$$\vec{V} = f\vec{N}, \ (x, z) \in D,$$

a pomocou riešenia rovnice (12) dosiahneme v poslednom kroku nášho algoritmu zmenu voľnej hladiny.

Celý tento proces sa opakuje v určitom počte časových krokov. Keďže hodnoty extrapolovanej rýchlosti f sa počas výpočtu zmenšujú, až dosiahnu veľmi malé hodnoty, môžeme tvrdiť, že získavame ustálené riešenie pre voľnú hladinu.

3 Numerické riešenie

Našu výpočtovú oblasť diskretizujeme pomocou bodov

$$(x_i, z_j),\ 0 \leq i, j \leq I,$$

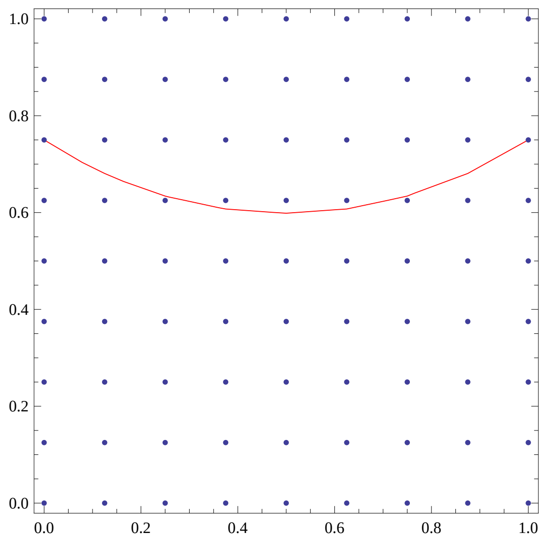
kde I máme dané, pričom platí

$$h =: x_{i+1} - x_i = z_{j+1} - z_j.$$

Level set funkciu v čase $t = 0$ určuje funkcia $\varphi^0_{ij} := \varphi^0(x_i, z_j)$ a jej hodnoty v ľubovolnom čase t^n označíme

$$\varphi^n_{ij} = \varphi(x_i, z_j, t^n).$$

Na realizáciu numerických výpočtov sme v našej práci použili metódu konečných diferencií, pričom budeme sledovať prácu [9].



Obrázok 9: Výpočtová oblasť, body siete a aproximovaná voľná hladina

Z obrázku 9 môžeme vidieť, že naša voľná hladina neprechádza gridovými bodmi, preto si označíme φ_{kl} suseda patriaceho ku bodu φ_{ij}, ktorý je mimo hladiny. Je zrejmé, že zo štyroch možných susedov $\varphi^n_{i\pm1j}$ alebo $\varphi^n_{ij\pm1}$ môžu byť najviac tri z nich nad hladinou a to $\varphi^n_{i\pm1j}$ a φ^n_{ij+1}.

Ak pre dva body platí $\varphi_{ij}^n \varphi_{kl}^n < 0$, určite sa medzi nimi nachádza nulová hodnota, ktorú získame pomocou lineárnej interpolácie

$$0 = \overline{\alpha}\varphi_{kl} + (1 - \overline{\alpha})\varphi_{ij}, \tag{14}$$

z čoho môžeme určiť parameter $\overline{\alpha}$

$$\overline{\alpha} = \frac{-\varphi_{ij}}{\varphi_{kl} - \varphi_{ij}}. \tag{15}$$

Práve tento parameter vyjadruje vzdialenosť gridového bodu (x_i, z_j) od hladiny. Tento poznatok budeme využívať pri všetkých ďalších výpočtoch.

3.1 Numerické riešenie toku podzemnej vody

Tlak a tok podzemnej vody je známy len pod hladinou. Výpočítame ich numericky z rovnice (2)-(3). Na numerickú diskretizáciu použijeme metódu štandardných konečných diferencií. Dostaneme

$$4p_{ij}^n - \widehat{p}_{i+1j}^n - \widehat{p}_{i-1j}^n - \widehat{p}_{ij+1}^n - p_{ij-1}^n = 0. \tag{16}$$

Rovnicu (16) riešime len v gridových bodoch na oblasti pod hladinou $\Omega(t)$. Okrajové podmienky (5) sme do výpočtu zahrnuli štandardne. Pre hodnoty, kde bola Dirichletova okrajová podmienka platilo

$$p_{ij}^n = p^D(x_i, z_j). \tag{17}$$

Tieto podmienky boli na bočných stranách výpočtovej oblasti. Na dolnej strane oblasti, ktorá pre naše prípady bola vždy vodorovná, sme definovali Neumannovu okrajovú podmienku. Na numerickú aproximáciu sme použili rovnicu v nasledujúcom tvare

$$p_{i,-1}^n = p_{i,1}^n + 2hp^N(x_i, z_0). \tag{18}$$

Teraz rovnica (16) platí aj pre hodnoty (x_i, z_0).

Hodnoty $\widehat{p}_{kl}^n$ predstavujú v prípade suseda nad hladinou extrapolovanú hodnotu tlaku, vypočítanú použitím (14), z čoho potom získavame pre hodnotu $\widehat{p}_{kl}^n$ vzťah

$$\widehat{p}_{kl}^n = \frac{\varphi_{kl}^n}{\varphi_{ij}^n} p_{ij}^n. \tag{19}$$

Ak (x_i, z_j) leží pod hladinou použijeme štandardnú definíciu $\widehat{p}_{kl}^n = p_{kl}^n$. Riešenie tohto problému výpočtu tlaku vedie na lineárny systém algebraických rovníc.

Zo získaného tlaku môžeme teraz prejsť na aproximáciu toku $\vec{q}_{ij}^n \approx \vec{q}(x_i, z_j, t^n)$. V našej práci sme používali dva spôsoby rátania pre tok podzemnej vody, preto v ďalšej časti práce sa budeme odvolávať na prvý spôsob ako na variantu A a druhý spôsob bude označený ako varianta B. Prvá bola publikovaná v [7], druhá metóda je popísaná prvý raz v tejto práci.

Prvý spôsob je, že na výpočet $\vec{q}_{ij}^n$ budeme používať len hodnoty pod hladinou, ktoré budú vyrátané podľa štandardných diferencií

$$\vec{q}_{ij}^n = \frac{1}{2h}(\widehat{p}_{i+1j}^n - \widehat{p}_{i-1j}^n, \widehat{p}_{ij+1}^n - p_{ij-1}^n), \tag{20}$$

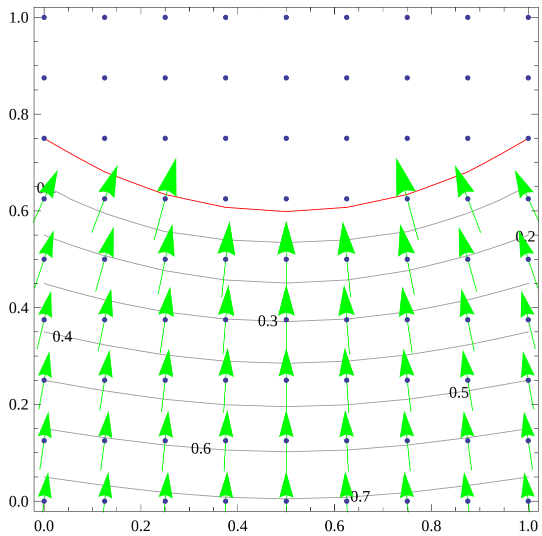
kde podľa potreby sa použije extrapolácia (19). Tlak spolu s tokom podzemnej vody môžeme vidieť na obrázku 10, kde je jasne viditeľné, že sa tok aproximuje len pre hodnoty pod hladinou.

Druhý spôsob rátania toku podzemnej vody je založený na rátaní toku z hodnôt pod hladinou a z hodnôt tesne nad hladinou. Výpočet sa rozdelí na dve časti. Výpočet (20) pre hodnoty tesne pod hladinou sa modifikuje na

$$\vec{q}_{ij}^n = \frac{1}{h}(p_{i+1j}^n - p_{ij}^n, p_{ij}^n - p_{ij-1}^n) \tag{21}$$

$$\vec{q}_{ij}^n = \frac{1}{h}(p_{ij}^n - p_{i-1j}^n, p_{ij}^n - p_{ij-1}^n) \tag{22}$$

$$\vec{q}_{ij}^n = (\frac{1}{2h}(p_{i+1j}^n - p_{i-1j}^n), \frac{1}{h}p_{ij}^n - p_{ij-1}^n). \tag{23}$$



Obrázok 10: Tok podzemnej vody (variant A)

Voľba konkrétnej aproximácie (21)-(23) závisí od toho, ktorí susedia bodu (x_i, z_j) ležia pod hladinou.

Výpočet je založený na tom, že pre hodnoty tesne pod hladinou zoberieme diferenciu platnú pre oblasť pod hladinou a teda v nej nepoužijeme žiadnu extrapoláciu.

Tok pre hodnoty tesne nad hladinou budeme rátať pomocou

$$\vec{q}^{\,n}_{ij+1} = \frac{1}{h}(\widehat{p}^{\,n}_{i+1j+1} - p^n_{ij+1}, \widehat{p}^{\,n}_{ij+1} - p^n_{ij}) \tag{24}$$

$$\vec{q}^{\,n}_{ij+1} = \frac{1}{h}(p^n_{ij+1} - \widehat{p}^{\,n}_{i-1j+1}, \widehat{p}^{\,n}_{ij+1} - p^n_{ij}) \tag{25}$$

$$\vec{q}^{\,n}_{ij+1} = \frac{1}{h}(\widehat{p}^{\,n}_{i+1j+1} - \widehat{p}^{\,n}_{ij+1}, \widehat{p}^{\,n}_{ij+1} - p^n_{ij}) \tag{26}$$

$$\vec{q}^{\,n}_{ij+1} = \frac{1}{h}(\widehat{p}^{\,n}_{ij+1} - \widehat{p}^{\,n}_{i-1j+1}, \widehat{p}^{\,n}_{ij+1} - p^n_{ij}) \tag{27}$$

$$\vec{q}^{\,n}_{ij+1} = (\frac{1}{2h}(\widehat{p}^{\,n}_{i+1j+1} - \widehat{p}^{\,n}_{i-1j+1}), \frac{1}{h}\widehat{p}^{\,n}_{ij+1} - p^n_{ij}), \tag{28}$$

kde voľba konkrétneho vzorca znova závisí od polohy bodu.

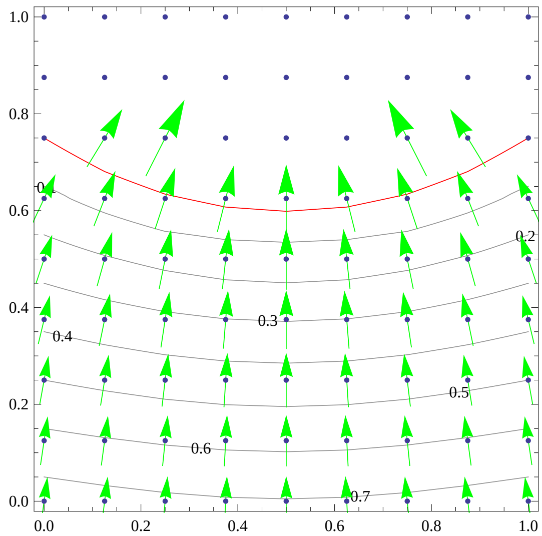
Dôležitý je postup pri rátaní toku pre body tesne nad hladinou, kde v z - smere sa vždy zoberie spätná diferencia s tým, že hodnota nad hladinou sa extrapoluje pomocou hodnoty pod hladinou. Netriviálny princíp je pre rátanie toku v x - smere, pretože nie všetky hodnoty sa dajú extrapolovať pomocou hodnoty pod hladinou. Preto sa pre takéto body použije extrapolácia jeho susedov v z - smere a zoberie sa tá diferencia, pre ktorú sa sused bodu nad hladinou dá extrapolovať. V prípade, že existuje extrapolácia pre obidvoch susedov použijeme centrálnu diferenciu.

Výhodou druhej metódy (21)-(28) je, že pomocou lineárnej interpolácie môžeme v tomto prípade aproximovať hodnoty toku na hladine. Z vyrátaných hodnôt tesne nad hladinou $\vec{q}_{ij+1}$ a tesne pod hladinou $\vec{q}_{ij}$ vypočítame pomocou interpolácie hodnotu toku

$$\vec{Q}^n_{ij} = (1-\alpha)\vec{q}^n_{ij} + \alpha\vec{q}^n_{ij+1}, \tag{29}$$

ktorá je daná presne na hladine.

Obrázok 11 pre ilustráciu zobrazuje vypočítaný tlak a tok podzemnej vody variantou B, kde vidíme, že pre tok pribudli hodnoty tesne nad hladinou.



Obrázok 11: Tok podzemnej vody (variant B)

3.2 Numerické riešenie eikonalovej rovnice

Získanie signed distance funkcie realizujeme riešením eikonalovej rovnice. Numerická schéma platná pre $1 \leq i, j \leq I - 1$ bude mať tvar [4]

$$\Phi_{ij}^{m+1} = \Phi_{ij}^{m} \pm \Delta s^m \left(1 - \frac{1}{h}\sqrt{(\Delta_x \Phi_{ij}^m)^2 + (\Delta_z \Phi_{ij}^m)^2}\right). \tag{30}$$

Pre body na hranici oblasti, hodnoty Φ fixujeme. Daná rovnica, nezávisla na reálnom čase t, bude prebiehať v krokoch $m = 0, 1, \ldots$. Správne znamienka v (30) sú dané na základe vzťahov (10) a (11). Hodnoty $\Delta_x \Phi_{ij}^m$ a $\Delta_z \Phi_{ij}^m$ definujeme pomocou

$$\Delta_x \Phi_{ij}^m = \max\{|M_{i+1j}|, |M_{i-1j}|\}$$

a

$$\Delta_z \Phi_{ij}^m = \max\{|M_{ij+1}|, |M_{ij-1}|\},$$

kde

$$M_{kl} = \begin{cases} \min\{\widehat{\Phi}_{kl}^m - \Phi_{ij}^m, 0\} & \text{ak } \phi_{ij}^m \geq 0 \\ \max\{\widehat{\Phi}_{kl}^m - \Phi_{ij}^m, 0\} & \text{ak } \phi_{ij}^m < 0 \end{cases}$$

Analogicky z predchádzajúcich častí hodnoty $\widehat{\Phi}_{kl}^m$ dostaneme extrapoláciou zo vzťahu (14).

$$\widehat{\Phi}_{kl}^n = \frac{\varphi_{kl}^n}{\varphi_{i,j}^n} \Phi_{ij}^n, \tag{31}$$

Časový krok Δs^m použitý v (30) je štandardne daný $h/2$ [4]. V našom prípade však môže byť stabilita schémy pri tomto časovom kroku porušená pre gridové body blízko hladiny.

Na základe [7] môžeme takéto body definovať špeciálny časový krok $\Delta^{crit} s_{ij}^m$

$$\Delta^{crit} s_{ij}^m = \frac{|\Phi_{ij}^m| h}{\sqrt{(\Delta_x \Phi_{ij}^m)^2 + (\Delta_z \Phi_{ij}^m)^2}}.$$

Následné použitie kritického časového kroku $\Delta^{crit}s_{ij}^m$ v rovnici (30) dostaneme zmenenú schému pre body tesne pod hladinou

$$\Phi_{ij}^{m+1} = \Phi_{ij}^m \pm \Delta s_{ij}^m \left(1 - \frac{1}{h}\sqrt{(\Delta_x \Phi_{ij}^m)^2 + (\Delta_z \Phi_{ij}^m)^2}\right), \tag{32}$$

kde

$$\Delta s_{ij}^m = \min\{\Delta s^m, \Delta^{crit} s_{ij}^m\}.$$

Z teoretického hľadiska by sme rovnicu (30) mali riešiť pre taký počet krokov m, až kým nedosiahneme ustálený stav. Z praktického hľadiska budeme počet krokov m fixovať na nejakej dostatočne veľkej hodnote M s tým, že naše požadované riešenie bude $\phi_{ij}^n = \Phi_{ij}^M$.

Dôležité je tu poznamenať, že hladiny nulových hodnôt aproximácií φ a ϕ nie sú identické, preto je nevyhnutné používať signed distance funkciu len pri výpočte normálového vektora $\overrightarrow{N}$.

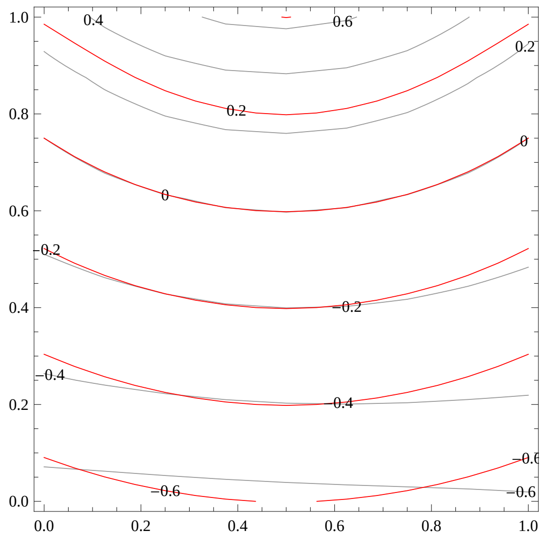
3.2.1 Príklad získania signed distance funkcie

Pre názornosť použitia signed distance funkcie uvedieme príklad získania tejto funkcie pre level set funkciu danú

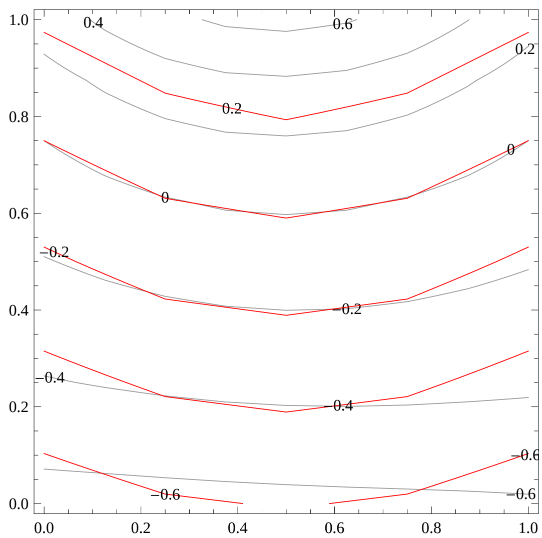
$$\varphi^0(x,y) = e^{(-0.5((x-0.4)^2+(y-0.3)^2))(r(0.0,0.75)-r(x,y))}$$

$$r(x,z) = \sqrt{(x-0.5)^2 + (z-1.5)^2}.$$

Vidíme, že na obrázku 12 majú hladinu podzemnej vody totožnú signed distance funkcia a level set funkcia (nulová čiara). Vidíme, že jednotlivé hladiny signed distance funkcie sú kvalitatívne podobné, čo je výhodné pre ďalšie výpočty. Keď sme výpočet robili na veľmi hrubej sieti 4×4 (obrázok 13), nulové hodnoty level set a signed distance funkcie boli rozdielne. Na obrázku 12 je výsledok vypočítaný na sieti 8×8 a vidíme, že tento rozdiel tu už nevidno.



Obrázok 12: Signed distance funkcia (červené čiary), Level set funkcia (šedé čiary)



Obrázok 13: Signed distance funkcia (červené čiary), Level set funkcia (šedé čiary)

3.3 Numerické riešenie extrapolovanej rýchlosti

Najdôležitejšou časťou našej práce je nájsť rýchlosť $f(x,z,t)$, ktorá je potrebná na určenie rýchlosti $\vec{V}$ pohybu voľnej hladiny. Na výpočet spomenutej rýchlosti $f(x,z,t)$ potrebujeme najprv aproximovať hodnoty $\overline{f}$ z (13), ktoré sú dané na hladine. Následne môžeme uvažovať nad jej definíciou v gridových bodoch v tvare

$$f_{ij}^n := \frac{1}{\theta_{ij}} \frac{\nabla \phi_{ij}^n}{|\nabla \phi_{ij}^n|} \cdot \left(\vec{q}_{ij}^n - A_{ij}^n \vec{e}_z \right), \tag{33}$$

kde

$$\Delta \phi_{ij}^n = \left(\frac{\phi_{i+1j}^n - \phi_{i-1j}^n}{2h}, \frac{\phi_{ij+1}^n - \phi_{ij-1}^n}{2h} \right)$$

V prvej metóde rátania toku podzemnej vody použijeme (33) pre hodnoty pod hladinou, ktoré budeme v nasledujúcej schéme fixovať

$$F_{ij}^m = \overline{f}_{ij}^n.$$

Extrapolovanú rýchlosť potom v ostatných gridových bodoch vyrátame metódy konečných diferencií pre rovnicu (12) v tvare

$$F_{ij}^{m+1} = F_{ij}^m + \frac{\Delta s}{h} \frac{1}{|\Delta \phi_{ij}|} \left(\Delta_x \phi_{ij} \Delta_x F_{ij}^m + \Delta_z \phi_{ij} \Delta_z F_{ij}^m \right), \tag{34}$$

kde

$$\Delta_x F_{ij}^m = \begin{cases} F_{ij}^m - F_{i-1j}^m & \text{ak } \Delta_x \phi_{ij} \geq 0 \\ F_{i+1j}^m - F_{ij}^m & \text{ak } \Delta_x \phi_{ij} < 0 \end{cases}$$

Analogická definícia platí pre $\Delta_z F_{ij}^m$.

Rovnicu (34) použijeme na riešenie extrapolovanej rýchlosti aj v druhom prípade s tým rozdielom, že teraz nie je potrebná fixácia hodnôt tesne pod hladinou. Zmena však v schéme (34) nastane pri rátaní diferencií

$$\Delta_x F_{ij}^m = \begin{cases} F_{ij}^m - \widehat{F}_{i-1j}^m & \text{ak } \Delta_x \phi_{ij} \geq 0 \\ \widehat{F}_{i+1j}^m - F_{ij}^m & \text{ak } \Delta_x \phi_{ij} < 0 \end{cases},$$

kde $\widehat{F}$ získame nasledujúcou extrapoláciou

$$\widehat{F}^m_{i-1j} = \frac{Fp}{\alpha} + \frac{\alpha - 1}{\alpha} F^m_{ij} \tag{35}$$

$$Fp = (1 - \alpha) f^n_{ij} + \alpha f^n_{i-1,j}. \tag{36}$$

$\vec{Q}^n_{ij}$ máme dané z rovnice (29). Analogicky to platí pre $\Delta_z F^m_{ij}$.

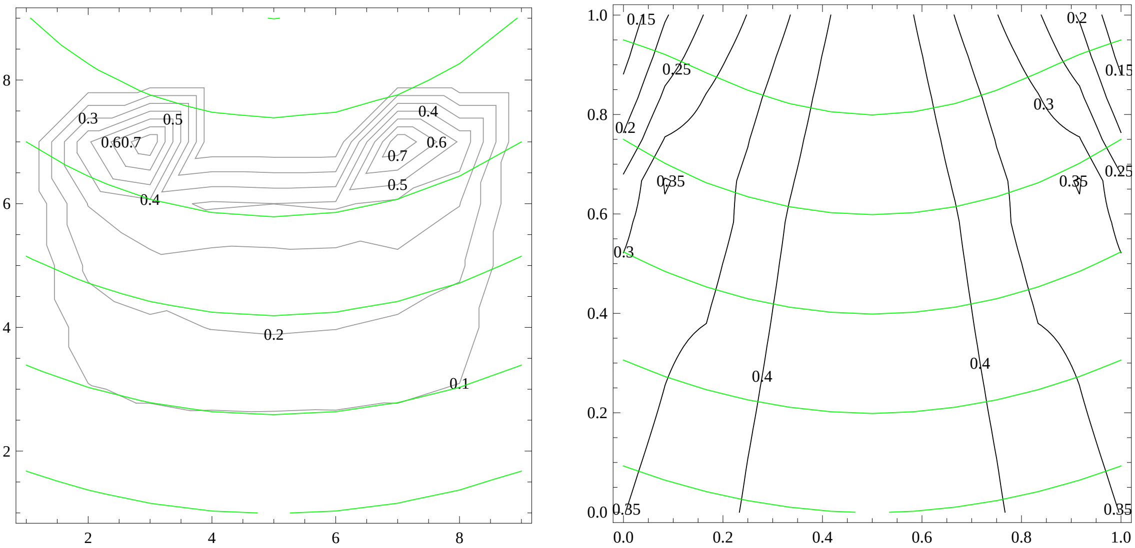
Podobne ako v už spomenutom prípade eikonalovej rovnice, aj táto schéma sa vykonáva v nemeniacom sa časovom kroku n. Výpočet prebieha v dostatočnom počte krokov m, až kým nedosiahneme ustálený stav.

3.3.1 Ukážka výpočtu extrapolovanej rýchlosti

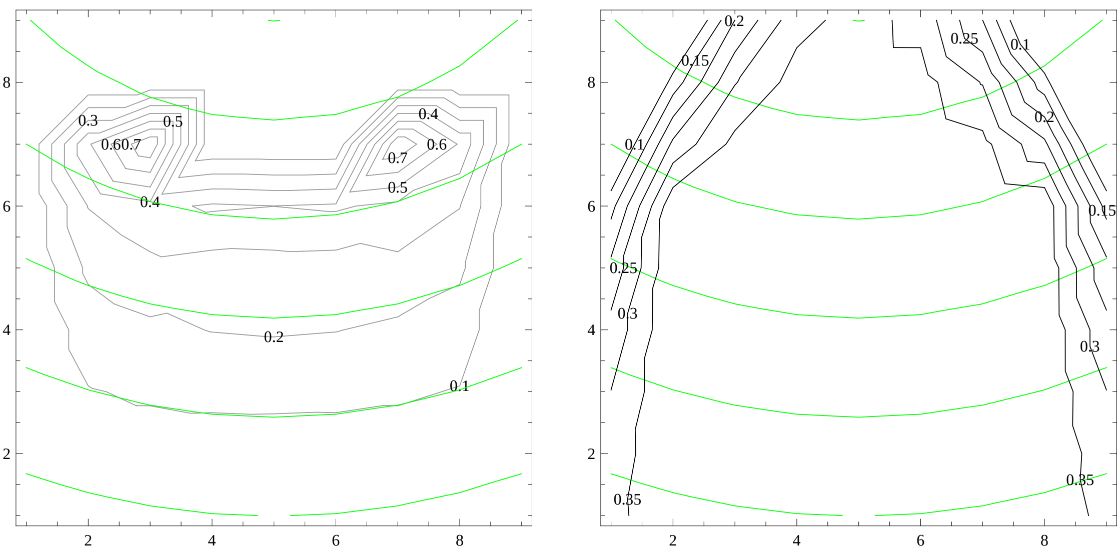
V tejto časti ukážeme ako získame extrapolovanú rýchlosť F. Vychádzame z počiatočnej podmienky, ktorú si určíme zo vzťahu (33) a je zobrazená na obrázku 15 na ľavej strane. V našej práci sme používali dva spôsoby rátania pre tok podzemnej vody, preto v ďalšej časti práce sa budeme odvolávať na prvý spôsob ako na variantu A a druhý spôsob bude označený ako varianta B.

Začiatočná podmienka sa vo variante A musí fixovať v hodnotách tesne pod hladinou, ktoré sa počas výpočtu nebudú meniť, iba sa budú šíriť z tohto miesta do celej oblasti. Varianta A je zobrazená na obrázku 14, z ktorého môžeme vidieť, že výsledná funkcia má približne kolmé izočiary na signed distance funkciu.

Vo variante B nie je potrebná fixácia hodnôt. Pre výpočet sú tu dôležité hodnoty tesne nad hladinou a tesne pod hladinou, z ktorých sa vyrátajú hodnoty platné presne na hladine z rovnice (35), ktoré budú rovnaké pre všetky časové kroky rátania extrapolovanej rýchlosti. Pri výpočte extrapolovanej rýchlosti môžeme na obrázku 15 vpravo vidieť, že získaná rýchlosť f je približne kolmá na signed distance funkciu.



Obrázok 14: Začiatočný stav a výsledný stav pre výpočet f podľa variantu A



Obrázok 15: Začiatočný stav a výsledný stav pre výpočet f podľa variantu B

3.4 Numerické riešenie rovnice advekcie

Rovnicou advekcie sme sa zaoberali v predchádzajúcej podkapitole, preto ju tu spomenieme len stručne. Hlavnou úlohou rovnice (8) je posun voľnej hladiny.

Z predchádzajúcich častí sme vyrátali všetky potrebné funkcie pre určenie rýchlosti $\vec{V}$, ktorou sa pohybuje voľná hladina. Môžeme teda písať pre jej definíciu v gridových bodoch na celej výpočtovej oblasti

$$\vec{V}_{ij}^n := f_{ij}^n \frac{\nabla \phi_{ij}^n}{|\nabla \phi_{ij}^n|}. \tag{37}$$

Samotný posun voľnej hladiny je vykonávaný v reálnom časovom kroku n a schéma má tvar

$$\varphi_{ij}^{n+1} = \varphi_{ij}^n + \frac{\Delta t^n}{h} \left(V_{ij,1}^n \Delta_x \varphi_{ij}^n + V_{ij,2}^n \Delta_z \varphi_{ij}^n \right), \tag{38}$$

kde

$$\Delta_x \varphi_{ij}^n = \begin{cases} \varphi_{ij}^n - \varphi_{i-1j}^n & \text{ak } V_{ij,1}^n \geq 0 \\ \varphi_{i+1j}^n - \varphi_{ij}^n & \text{ak } V_{ij,1}^n < 0 \end{cases}.$$

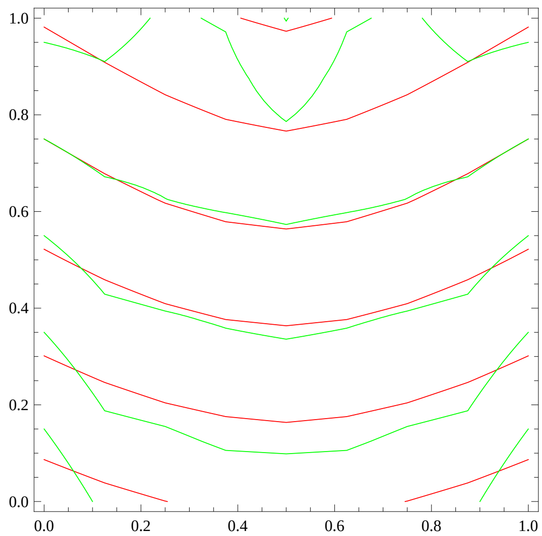
Obdobná definícia platí pre $\Delta_z \varphi_{ij}^n$.

Obidve rovnice advekcie (34) a (38) sú riešené upwind metódou, ktorá sa používa na riešenie hyperbolických diferenciálnych rovníc.

3.5 Výpočtové vylepšenia

Ako sme už v práci spomínali, hľadáme ustálené riešenie podzemnej vody. Rýchlosť získania ustáleného riešenia pohybu voľnej hladiny závisí aj od dvoch rovníc (30) a (34), ktoré sú formálne časovo závislé. Cieľom tejto kapitoly je diskusia, ako je algoritmus možne urýchliť.

V predchádzajúcich častiach sme výpočty daných rovníc robili na dostatočne veľkom počte časových krokov. V ďalšom skúmaní sme zmenili počiatočnú podmienku pre dané rovnice a počet časových krokov sme zmenšili na malý fixný počet. Vychádzali sme z predpokladu, že nasledujúci krok bude pokračovať na hodnotách výrátaných z predchádzajúceho kroku. Jedine hodnoty tesne pod hladinou a nad hladinou sa zmenili podľa pohybu level set funkcie. Hlavnou myšlienkou tohto postupu je, že sa nesnažíme nájsť riešenie rovníc (10)-(12), ale snažíme sa nájsť len ich časovo závislé riešenie, ktoré má hodnoty v okolí hladiny blízke hodnotám stacionárneho riešenia. Na základe tejto úpravy sme zaznamenali zrýchlenie výpočtu o polovicu.



Obrázok 16: Riešenie bez použitia signed distance funkcie (zelená čiara), riešenie s použitím signed distance funkcie (červená čiara)

V práci sme sa zaoberali aj potrebou použitia signed distance funkcie. Level set formulácia vyžaduje definovať si level set funkciu, ktorá bude reprezentovať našu voľnú hladinu. Možností výberu level set funkcie je niekoľko. Našou podmienkou pre výber spomenutej funkcie je, aby jej nulová izočiara bola totožná s voľnou hladinou. Ako počiatočnú podmienku si volíme signed distanc funkciu, ktorej norma gradientu je jednotková, pretože táto vlastnosť je výhodná pre správne šírenie informácie. Kvôli pohybu voľnej hladiny v každom časovom kroku sa táto vlastnosť level set funkcie môže stratiť, preto vyrátaním eikonalovej rovnice dostávame druhú level set funkciu, ktorá má jednotkovú normu gradientu. V iných aplikáciach [5] sa zvykne nahradiť level set funkcia vypočítanou signed distance funkciou, čo môže viesť k nefyzikálnej zmene polohy voľnej hladiny. Na obrázku 16 uvádzame pre ilustráciu porovnanie výpočtu voľnej hladiny bez použitia signed distance funkcie (zelené čiary) a s použitím signed distance funkcie (červené čiary). Pozorujeme, že v prípade nepoužitia signed distanc funkcie, level set funkcia, ktorá bola v čase $t = 0$ signed distance funkciou, túto vlastnosť pre $t > 0$ stratila. Uvádzané výsledky, prezentované na obrázku 16, sú riešené na sieti 8×8.

4 Programové prevedenie

Všetky programy boli vytvorené v softvéri Mathematica. Pre dosiahnutie výsledkov sme vytvorili dátový súbor obsahujúci funkcie používané na riešenie príkladov. Programy sú súčasťou práce ako príloha v podobe CD nosiča.

4.1 Pomocné programy

Na vytvorenie celkového programu bolo potrebné vytvoriť množstvo podprogramov, ktoré slúžili na získanie pomocných výsledkov. Zoznam pomocných funkcií v krátkosti opíšeme v nasledujúcom texte.

`extrbody()` – funkcia, do ktorej vstupuje level set funkcia a funkcia, ktorú chceme extrapolovať. Výsledkom sú extrapolované hodnoty a paramater α vyrátaný zo vzťahu (14).

`grady()` – funkcia, do ktorej vstupuje level set funkcia a funkcia, ktorej gradient chceme vypočítať. V tejto funkcii sa na rátanie gradientu používa extrapolácia a získame len hodnoty v smere z.

`gradx()` – obdoba predchádzajúcej funkcie s jediným rozdielom, že tu získame gradient v smere x.

`gradient()`– funkcia na výpočet gradientu pomocou štandardných diferencií. Vstupuje do nej funkcia, ktorej gradient chceme vypočítať.

`pgradient()` – funkcia na výpočet gradientu tlaku použitím variantu A, kde vstupom je tlak podzemnej vody.

`gradientExtrapolation()` – funkcia na výpočet gradientu tlaku použitím variantu B, kde vstupom je tlak podzemnej vody.

`Fhladina()` – vstupnými parametrami pre túto funkciu sú level set funkcia a hodnoty tesne nad hladinou $\vec{q}_{ij+1}$ a tesne pod hladinou $\vec{q}_{ij}$, z ktorých sa pomocou interpolácie z rovnice (29) vyráta hodnota platná na hladine, ktorá je zároveň výstupom danej funkcie.

`gradientF()` – funkcia, ktorá sa používa v upwind metóde, slúži na riešenie gradientu funkcie. Vstupom sú charakteristická rýchlosť na základe ktorej sa rozhoduje použitie diferencie a funkcia, ktorej gradient chceme vypočítať. Používame v nej aj extrapoláciu.

`gradfi()` – podobne ako v predchádzajúcej funkcii, len v tomto prípade nie je použitá extrapolácia.

4.2 Program na riešenie tlaku

Funkcia `Extrapolation3()` rieši tlak podzemnej vody. Je to samostatný program, do ktorého vstupuje funkcia, ktorej tlak chceme vypočítať. V našom prípade to je level set funkcia. Ďalšími vstupmi sú okrajové podmienky, krajné body výpočtovej oblasti $x1$, $x2$, $z1$, $z2$ a počet dielikov, na koľko bude výpočtová oblasť rozdelená. Táto funkcia nevyužíva žiadny z pomocných programov. Hlavným cieľom danej funkcie, je vytvorenie matice a pravej strany na základe rovníc (16), z ktorých sa následne vypočíta riešenie.

4.3 Program na získanie signed distance funkcie

Signed distance funkciu získame programom `cyklus2()`. Jej vstupné parametre sú level set funkcia, ku ktorej chceme získať signed distance funkciu a počet časových krokov, koľkokrát sa daný cyklus vykoná. Pre túto funkciu sú potrebné pomocné funkcie `gradx()` a `grady()`.

4.4 Program na získanie extrapolovanej rýchlosti

Program na získanie extrapolovanej rýchlosti sa nazýva `upwind()` a jedným z jeho vstupov je z predchádzajúceho programu vyrátaná signed distance funkcia, z ktorej pomocou programu `gradient()` vyrátame jej gradient. Ten bude slúžiť na smer šírenia informácie. Používame tu aj funkciu `gradientF()`, ktorá je rozhodujúcim prvkom danej schémy. Dôležitým vstupným parametrom je ešte počet časových krokov, v ktorých bude cyklus prebiehať, pretože tento parameter budeme podľa potreby zrýchlovania celkového algoritmu meniť.

4.5 Celkový program

Pospájaním všetkých predchádzajúcich podprogramov do výslednej funkcie s názvom `postup2()` dostávame celkový program pre náš matematický model. Postup začína vyrátaním tlaku, z ktorého sa podľa typu varianty použije na výpočet toku podzemnej vody pomocná funkcia `pgradient()` alebo `gradientExtrapo lation()`. Nezávisle na tom sa vypočíta signed distance funkcia samostatným programom `cyklus2()`. Z doteraz získaných hodnôt určíme počiatočnú podmienku pre program `upwind()`, z ktorého získame extrapolovanú rýchlosť $f(x, z, t)$. Potom nám stačí určiť rýchlosť $V(x, z, t)$, podľa ktorej sa bude pohybovať voľná hladina a tá bude vstupovať do posledného cyklu platného pre pohyb hladiny.

Vstupnými parametrami musia byť všetky potrebné vstupy pre podprogramy, ktoré nie sú výsledkom nejakej pomocnej funkcie. Keďže hľadáme ustálené riešenie v čase, jedným zo vstupov je aj počet časových krokov, v ktorých budeme riešenie hľadať.

Celkový výsledný postup nebol najzložitejším prvkom práce, ale vytvorenie jednotlivých podprogramov, najmä program pre extrapolovanú rýchlosť a získanie signed distance funkcie, neboli triviálnymi záležitosťami.

5 Numerické experimenty

Náš algoritmus sme testovali na dvoch príkladoch so známym presným riešením tlaku podzemnej vody. V oboch prípadoch pre rovnicu (4) sme mali definované Dirichletove okrajové podmienky na pravej a ľavej strane štvorca D a na spodnej strane výpočtovej oblasti bola daná Neumannova okrajová podmienka. Okrajové podmienky presne špecifikovali stacionárne riešenie tlaku. Keďže sme poznali presné stacionárne riešenie, mohli sme naše výsledky s ním porovnávať. Výpočty sme robili na sieťach (8×8, 12×12, 16×16) a časový krok sme vybrali $\Delta s = 0.05h$. Interval reálneho času, v ktorom sme rátali náš matematický model, bol $(0, 0.3)$ a konštanty vystupujúce vo vzťahu (3) sú dané $K = \rho = 1$. Obrázky použité na ukážku experimentov boli urobené pre výpočtovú sieť 8×8.

5.1 Príklad s ustálenou horizontálnou hladinou

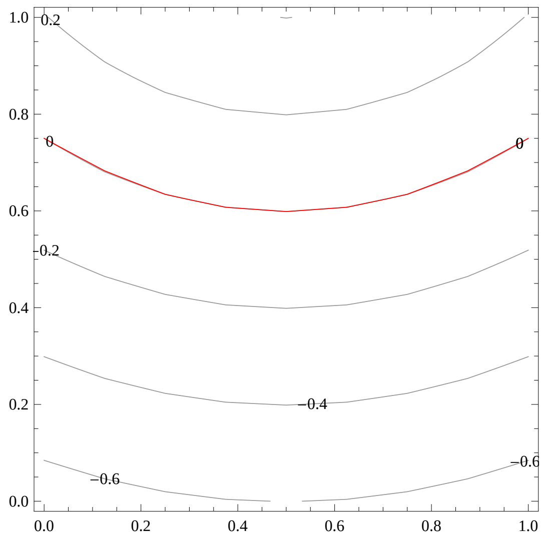
V prvom prípade sme testovali, ako sa dosiahne horizontálne ustálená voľná hladina, keď sme počiatočnú polohu vytvorili jej vychýlením do spodnej strany. Polohu hladiny v tomto príklade bude teda predstavovať level set funkcia $\varphi^0 = \Psi$ v tvare

$$\Psi(x,z) = r(0, 0.75) - r(x,z) \tag{39}$$

$$r(x,z) = \sqrt{(x-0.5)^2 + (z-1.5)^2}. \tag{40}$$

Na obrázku 17 môžeme vidieť level set funkciu, ktorej nulové hodnoty predstavujú voľnú hladinu podzemnej vody (červenou čiarou je zvýraznená voľná hladina).

Pre tento príklad uvažujeme o gravitácii, ktorá vystupuje v rovnici (3) a platí $g = 1$.



Obrázok 17: Izočiary počiatočnej level set funkcie

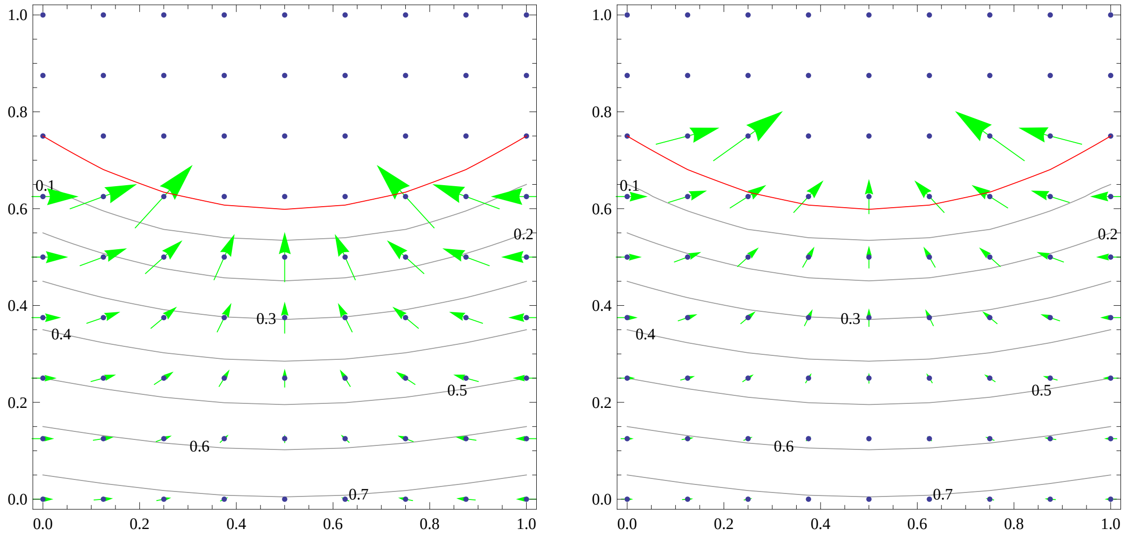
Presné stacionárne riešenie tlaku je definované vzťahom

$$P(x, z) = 0.75 - z.$$

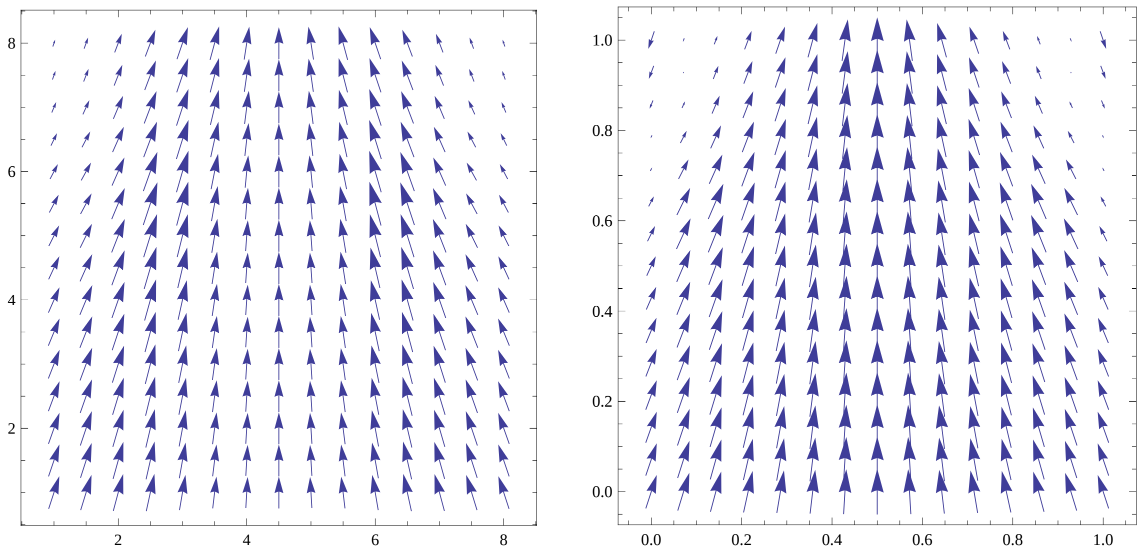
Na obrázku 18 môžeme vidieť riešenie tlaku v čase $t = 0$. Spolu s izočiarami pre tlak je na danom obrázku znázornený tok podzemnej vody (zelenými šípkami). Porovnávame variantu A a variantu B riešenia toku podzemnej vody. Rozdiel je tu len v hodnotách, v ktorých sa daný tok ráta, pričom vo variante B sa pridajú hodnoty tesne nad hladinou a hodnoty tesne pod hladinou sa mierne modifikujú podľa vzťahov (21)-(23).

Pre rýchlosť $\vec{V}$, ktorou sa pohybuje voľná hladina podzemnej vody, máme počiatočný stav zobrazený na obrázku 19. Môžeme vidieť, že pohyb hladiny bude nahor tak ako očakávame. Vidíme, že pre variant A je rýchlosť pohybu voľnej hladiny nerovnomerná.

Ustálené riešenie, ku ktorému sme dospeli našim programom, bola horizontálna čiara. Tento výsledok sme predpokladali a našim experimentom sme to aj potvrdili. Obrázok 20, ktorý sme získali programom požívajúcim rátanie toku variantou B, zobrazuje výsledky pre siete 8×8,

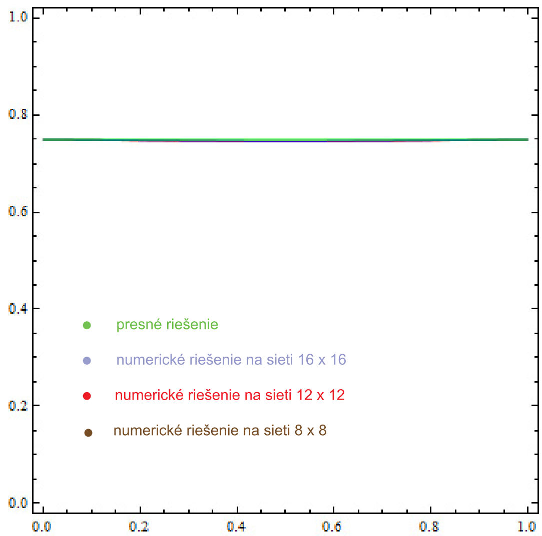


Obrázok 18: Začiatočný stav pre tlak a tok, variant A vľavo, variant B vpravo



Obrázok 19: Začiatočný stav rýchlosť $\vec{V}$, variant A vľavo, variant B vpravo

12×12 a 16×16 spolu s presným riešením. Môžeme na ňom pozorovať, že výsledok bol veľmi presný už pre najhrubšiu sieť 8×8, čo nás vedie k tvrdeniu, že pre tento príklad nie je nutné rátať tlak pre veľmi jemné výpočtové siete.



Obrázok 20: Výsledky

5.2 Príklad s počiatočnou horizontálnou hladinou

V druhom experimente sme výchadzali z počiatočného stavu v horizontálnej polohe voľnej hladiny. Level set funkcia, ktorú použijeme na definíciu počiatočnej voľnej hladiny podzemnej vody, bude mať tvar

$$\varphi^0 = z - 0.75.$$

Tento príklad bude špecifický v tom, že v ňom uvažujeme o rýchlosti nárastu A, ktorá vystupuje v rovnici (13) a je daná

$$A = -\frac{1}{\partial_z \Psi}(\nabla P \cdot \nabla \Psi).$$

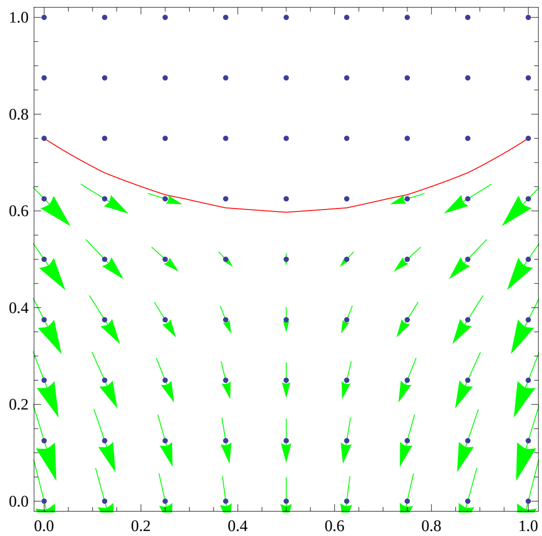
Vzťah sme dostali tak, že sme požadovali aby rýchlosť $\overline{f}(x, z) = 0$ v (13) pre body (x, z) ležiace na ustálenej voľnej hladine.

Jeden z ďalších dôležitých rozdielov medzi jednotlivými experimentami je, že tu neuvažujeme o žiadnej gravitácii, teda $g = 0$.

Presné riešenie tlaku pre ustálenú hladinu je v tvare

$$P(x, z) = \ln(r(0, 0.75)^{-1}) - \ln(r(x, z)^{-1}),$$

kde funkcia $r(x, z)$ je rovnaká ako v predchádzajúcom príklade.

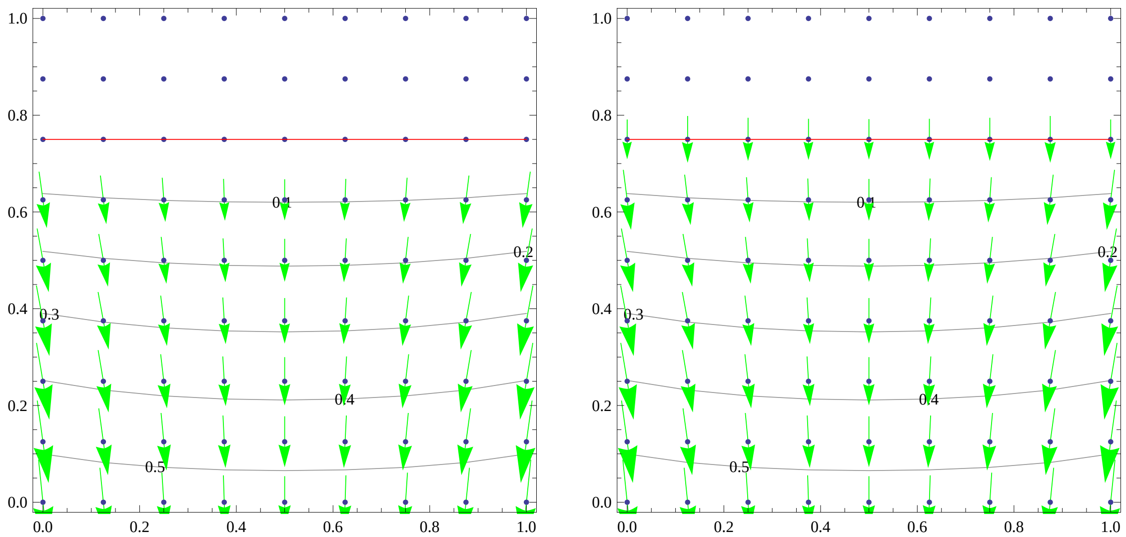


Obrázok 21: Presné riešenie tlaku spolu s tokom podzemnej vody

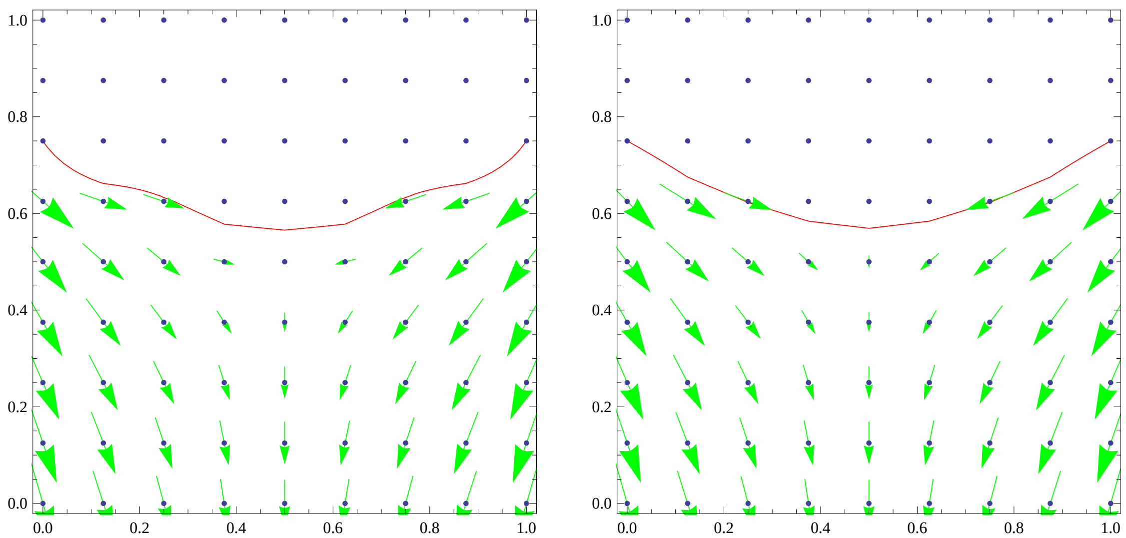
Na obrázku 21 môžeme vidieť, že pohyb podzemnej vody v ustálenej situácii je len v smere pozdĺžnom ku voľnej hladine podzemnej vody. V tomto stave už nedochádza ku zmene výšky voľnej hladiny. Voda tečie len v horizontalnom smere, čo je dôležitá podmienka aby sme dostali ustálenú polohu voľnej hladiny podzemnej vody.

Začiatočná poloha voľnej hladiny, z ktorej sme vychádzali, spolu s tokom podzemnej vody je zobrazená na obrázku 22 pre obidve varianty. Z neho vidíme, že pohyb voľnej hladiny bude smerovať nadol. Ustálený stav vyrátaný v čase $t = 0.3$ je zobrazený na obrázku 23 a predstavuje tok, ktorý je pozdĺžny ku voľnej hladine. Porovnanie riešení pre variantu A a variantu B riešenia toku podzemnej vody vedie k záveru, že výsledok z varianty B je lepší. Varianta A pre tento prípad vytvárala nežiadúce zvlnenia, ktoré boli spôsobené nespojitou zmenou extrapolovanej rýchlosti.

Na obrázku 24 sme vykreslili presné riešenie spolu s numerickými výsledkami rátanými na sieťach 8×8, 12×12 a 16×16. Vidíme, že so zvyšujúcim sa priestorovým krokom sme dostávali riešenie bližšie ku presnému riešeniu, čiže v tomto príklade zmena priestorového kroku mala vplyv na presnosť riešenia. Môžeme však vidieť, že výsledky pre siete



Obrázok 22: Začiatočný stav variant A vľavo, variant B vpravo

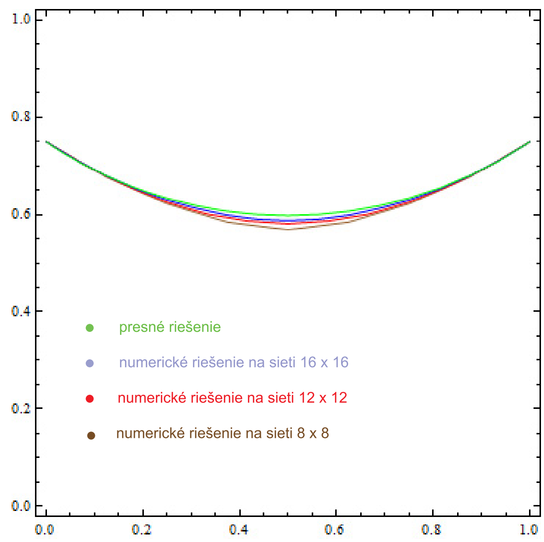


Obrázok 23: Numerické riešenie variant A vľavo, variant B vpravo

12×12 a 16×16 sa už veľmi od seba nelíšia.

5.3 Príklad s netriviálnou počiatočnou hladinou

Cieľom tohto príkladu je porovnať polohu ustálenej voľnej hladiny nášho matematického modelu so známym riešením pre prístup, keď sa využíva Dupuitov predpoklad [10].



Obrázok 24: Výsledky

Vychádzali sme zo šikmej level set funkcie danej vzťahom

$$\varphi_0(x,z) = z - \frac{3}{4} + \frac{1}{8}x. \tag{41}$$

Pre rovnicu (4) sme pre tento príklad definovali Dirichletove okrajové podmienky na hranici Γ^D v tvare

$$p^D(x1,z) = \frac{3}{4} - z \tag{42}$$

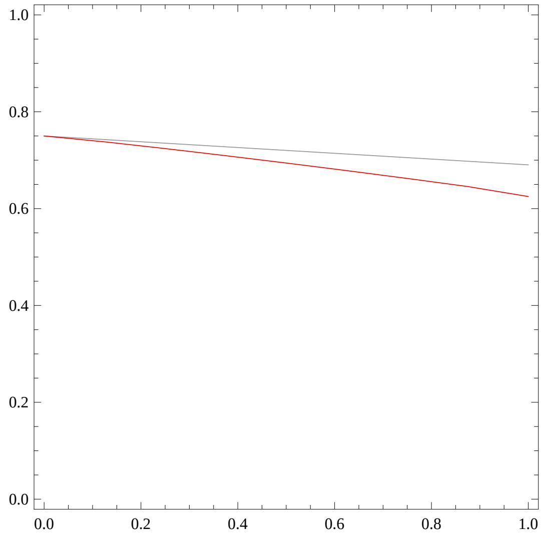
$$p^D(x2,z) = \frac{5}{8} - z, \tag{43}$$

kde $x1$ a $x2$ sú krajné body intervalu v smere x. Neumannovu okrajovú podmienku sme dali konštantnú na hranicu Γ^N ako

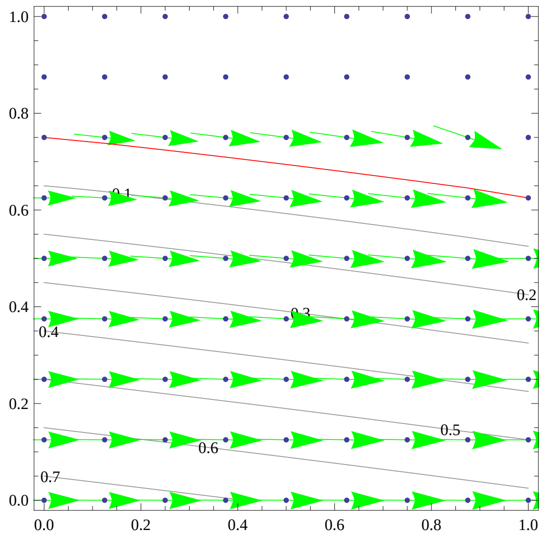
$$p^N(x,z1) = 1.0. \tag{44}$$

Presné riešenie pre tento príklad nepoznáme, ale výsledky budeme porovnávať s doterajšími zisteniami od iných autorov, ktorí sa zaoberali rovnakým problémom. Vzťah pre riešenie, s ktorým sme naše výsledky porovnávali sme uviedli v úvode, ale pre zopakovanie je

$$H = \sqrt{H_0^2 - \frac{2qx}{K}}. \tag{45}$$



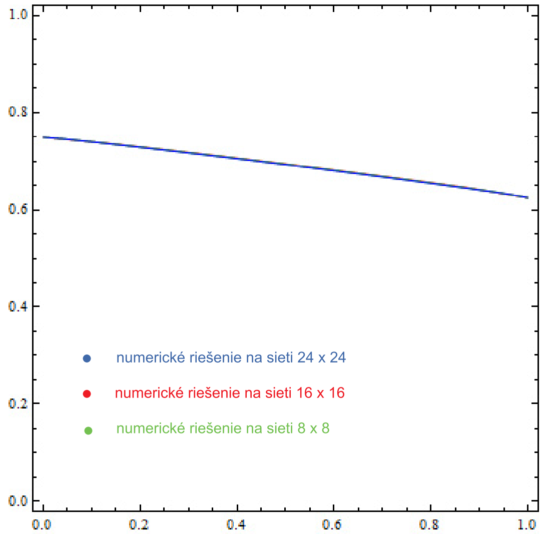
Obrázok 25: Numerické riešenie (červená čiara), Dupuitov princíp (šedá čiara)



Obrázok 26: Tok podzemnej vody

Rovnako ako v prvom experimente aj pre tento príklad uvažujeme o gravitácii, ktorá vystupuje v rovnici (3) v tvare $g = 1$.

Náš výsledok porovnávame s riešením podľa Dupuitovho princípu (obrázok 25), ktorý je zobrazený šedou čiarou a naše numerické riešenie predstavuje červená čiara. Ako sme v úvode spomínali, Dupuitov princíp je založený na tom, že uvažuje len o toku podzemnej vody, ktorý má vodorovný smer pozdĺž celej voľnej hladiny a teda zanedbávame jeho vertikálnu zložku. Ako však môžeme vidieť na obrázku 26, kde je zobrazená voľná



Obrázok 27: Výsledky

hladina spolu s tokom podzemnej vody, tok nie je vodorovný. Môžeme teda povedať, že náš prístup riešenia je odlišný pre tento typ úloh.

Porovnanie výsledkov pre rôzne výpočtové siete (8×8, 16×16 a 24×24) je zobrazené na obrázku 27. Ako z obrázku vidieť naše ustálené riešenie sa dosiahlo už pri použití siete 8×8. Následne zjemňovanie nepriniesla zlepšenie riešenia a teda môžeme povedať, že ustálený stav pre tento príklad sme dosiahli už použitím relatívne hrubej siete.

6 Záver

Práca sa zaoberá numerickým modelovaním problému ustálenia pohyblivej voľnej hladiny podzemnej vody. Na popis voľnej hladiny sme použili level set metódu, ktorá nám umožnila numericky riešiť daný problém na pevnej výpočtovej sieti. V doterajších skúmaniach sa riešenie aproximovalo na pohyblivej sieti. Myšlienka nášho matematického modelu je podľa našich vedomostí originálnym prístupom k riešeniu daného problému a taktiež zjednodušením predchádzajúcich postupov.

Náše obohatenie level set metódy spočíva v použití dvoch level set funkcií na popis voľnej hladiny. Jedna slúži na definíciu polohy voľnej hladiny a druhá len na definíciu normálového smeru k nej, čím sme sa vyhli problémom známym z iných aplikácií [5], kde použitie len jednej z nich na obidve situácie viedlo k numerickým ťažkostiam. Druhý prínos našej práce je, že sme netriviálnym spôsobom extrapolovali rýchlosť toku v smere normály od voľnej hladiny do celej výpočtovej oblasti, kde nie je inak fyzikálne definovaná. Týmto spôsob sme získali vhodné rýchlostné pole pre všetky izočiary level set funkcie.

Cieľom práce bolo zistiť vhodnosť level set metódy pre tento typ úloh, čo sa výsledkami z príkladov jednoznačne potvrdilo. Prvé pokusy z tejto práce boli už publikované v medzinárodnom zborníku [7].

Podrobne sme sa venovali dvom numerickým príkladom, u ktorých sme poznali presné riešenie a teda sme mohli naše výsledky s nim porovnávať. Rozdiel medzi numerickým a presným riešením je malý už na relatívne hrubých sieťach. V ďalšom príklade sme ukázali odchýlky od riešenia pomocou takzvaného Dupuitovovho princípu. Tento často riešený problém ustálenia hladiny medzi dvoma hrádzami s rôznou výškou zanedbáva vertikálne zložky prúdenia voľnej hladiny. Treba ešte poznamenať, že výškový rozdiel medzi jednotlivými brehmi nie je veľký. Naše výsledky, kde sa zjednodušenie pre vertikálne zložky nepoužíva, vedú k relatívne výrazným odchýlkam oproti danému Dupuitovmu prístupu. Rozdiely v riešeniach pre rôzne výpočtové siete boli minimálne a teda môžeme povedať, že ustálený stav sme dosiahli už pre relatívne hrubú sieť.

Výsledkom diplomovej práce je komplexný opis použitých metód a ich realizácia v programe Mathematica. Vďaka numerickým a grafickým schopnostiam daného softvéru sa dá veľmi podrobne sledovať správanie sa metódy pre riešenie dvojrozmerných úloh prúdenia vody s voľnou hladinou v pórovitom prostredí. V budúcnosti preto bude uľahčená realizácia a implementácia level set metódy pre riešenie realistických a praktických trojrozmerných problémov prúdenia podzemnej vody.

Literatúra

[1] Adalsteinsson. D., Sethian. A., *The Fast Construction of Extension Velocities in Level Set Methods*, J. Comput. Phys., 148 (1), 1999, s. 2-22.

[2] Baroková. D. *Numerické metódy riešenia vplyvu vodného diela na hladinový režim podzemných vôd*, Dizertačná práca, Katedra hydrotechniky SvF STU v Bratislave, 2003.

[3] Bear. J. *Hydraulics of Groundwater*, McGraw-Hill, New York, 1979.

[4] Bourgine. P., Frolkovič. P., Mikula. K., and Remešíková. M. *Extraction of the intercellular skeleton from 2d microscope images of early embryogenesis*, In Lecture Notes in Computer Science 5567 , 2009, s. 38-49.

[5] Frolkovič. P., Logashenko. D., and Wittum. G. *Flux–based level set method for two–phase flows*, In R. Eymard and J.–M. Herard, editors, Finite Volumes for Complex Applications, ISTE and Wiley, 2008, s. 415-422.

[6] Frolkovič. P., Mikula. K., *Flux–based level set method: A finite volume method for evolving interfaces*, Applied Numerical Mathematics, 57(4), 2007, s. 436-454.

[7] Frolkovič. P., Zacharovská. P., *Numerical modelling of dynamic groundwater table using level set formulation*, XVIII International Conference on Water Resources, ©CIMNE, Barcelona, 2010, accepted.

[8] Frolkovič. P. and Wehner. C., *Flux–based level set method on rectangular grids and computations of first arrival time functions*, Computing and Visualization in Science, 12 (5), 2009, s. 297-306.

[9] Gibou. F., Fedkiw. R., Cheng. L.–T., and Kang. M., *A second order accurate symetric discretization of the Poisson equation on irregular domains*, J. Comput. Phys., 176, 2002, s. 205-227.

[10] Kratochvíl. J. a kolektív, *Hydraulika*, Ediční středisko VUT Brno., 1989.

[11] Mäsiar. E., Kamenský. J., *Hydraulika podzemných vôd*, STU, Bratislava, 1983.

[12] Osher. S., Fedkiw. R., *Level Set Methods and Dynamic Implicit Surfaces*, Springer, 2003.

[13] Sethian. J., *Level Set Methods and Fast Marching Methods*, Cambridge University, Press, 1999.