

Online simulační model propojených nádrží

Online simulation model of interconnected tanks

Václav Bezděk

Diplomová práce 2008



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2008/2009

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Václav BEZDĚK**

Studijní program: **N 3902 Inženýrská informatika**

Studijní obor: **Informační technologie**

Téma práce: **Online simulační model propojených nádrží**

Zásady pro vypracování:

1. Zpracujte literární rešerši na dané téma.
2. Sestavte matematický model systému propojených nádrží. Uvažujte různé tvary nádrží.
3. Navrhněte online systém (ve formě dynamických www stránek) pro simulaci systému propojených nádrží. Uživatel bude moci volit parametry daného systému.
4. Zvolte vhodné zobrazení výsledků simulačních pokusů. Při výpočtu časového průběhu sledovaných veličin použijte různých numerických metod. Proveďte srovnání výsledků.

Rozsah práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **NOSKIEVIČ, Petr.** Modelování a identifikace systémů. Ostrava : Montanex a.s., 1999. 276 s. ISBN 80-7225-030-2.
2. **VICHER, Miroslav.** Numerická matematika. Ústí nad Labem, 2003. 85 s. Skripta. ISBN 80-7044-516-5.
3. **HLAVENKA, Jiří, et al.** Vytváříme WWW stránky a spravujeme moderní web site. 7. aktualiz. vyd. Brno : CP Books, 2005. 356 s. ISBN 80-251-0801-5.
4. **JANOVSKÝ, Dušan.** Jak psát web : o tvorbě internetových stránek [online]. 2004 , 02. února 2009 [cit. 2009-02-02]. Dostupný z WWW: . ISSN 1801-045.

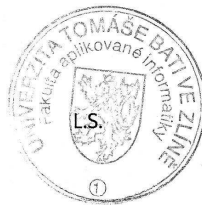
Vedoucí diplomové práce: **Ing. Bronislav Chramcov, Ph.D.**
Ústav aplikované informatiky

Datum zadání diplomové práce: **20. února 2009**

Termín odevzdání diplomové práce: **27. května 2009**

Ve Zlíně dne 13. února 2009


prof. Ing. Vladimír Vašek, CSc.
děkan




doc. Ing. Ivan Zelinka, Ph.D.
ředitel ústavu

ABSTRAKT

Cílem diplomové práce „Online simulační model propojených nádrží“ je vytvořit elektronickou podporu výuky předmětu simulace systémů. Práce je rozdělena na teoretickou a praktickou část. Teoretická část je tvořena pěti kapitolami, jejichž obsahem je problematika e-learningu a přehled podobných nástrojů na internetu. Dále popis jednotlivých simulačních modelů včetně odvození vztahů popisujících dané modely a metody řešení těchto vztahů. A také popis použitých technologií k tvorbě systému. Praktická část se zabývá samotnou tvorbou elektronické pomůcky, která je formou dynamických www stránek a je převážně vytvořena v jazyce PHP. Ve třech kapitolách je zde popsán systém jak z pohledu uživatelského ovládní, tak i z hlediska tvorby. V závěru je řešen konkrétní příklad.

Klíčová slova: simulace, model nádrže, systém propojených nádrží, výtok, obyčejné diferenciální rovnice, numerické řešení ODR

ABSTRACT

The diploma work “Online simulation model of interconnected tanks” sets its aim on creating electronic support for course in system simulation. The whole diploma thesis is divided into two parts: theoretical and experimental part. Five chapters of theoretical part are focused on principal issues of e-learning and summary of present online similar systems. Next are descriptions of simulation models including differential equations of each of models and methods to compute numerical solutions of differential equations. Used technology to create system is also included in theoretical part. Experimental part focuses on explication of creating dynamic www pages, generated in PHP. In three chapters are described user’s aspects of system and programmer’s aspects of system. In the end is solution of concrete example.

Keywords: simulation, model of tank, system of interconnected tanks, outflow, ordinary differential equations, numerical solution of ODE

Poděkování:

Na tomto místě bych chtěl poděkovat vedoucímu mé diplomové práce panu Ing. Bc. Bronislavu Chramcovovi, Ph.D. za jeho cenné rady a připomínky při vedení této práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval.

V případě publikace výsledků budu uveden jako spoluautor.

Ve Zlíně

.....
Podpis diplomanta

OBSAH

OBSAH	7
ÚVOD	10
TEORETICKÁ ČÁST	11
1 E-LEARNING	12
1.1 DEFINICE E-LEARNINGU	12
1.2 CHARAKTERISTIKA E-LEARNINGU	12
1.3 FORMY E-LEARNINGU	13
1.3.1 <i>on-line (synchronní) e-learning</i>	13
1.3.2 <i>off-line (asynchronní) e-learning</i>	13
1.4 VÝHODY E-LEARNINGU	13
1.5 NEVÝHODY E-LEARNINGU	14
1.6 STANDARDY E-LEARNINGU	15
2 DOSTUPNÉ ZDROJE K DANÉ PROBLEMATICE	16
2.1 LITERATURA	16
2.2 ELEKTRONICKÉ ZDROJE	16
3 MECHANIKA TEKUTIN	18
3.1 ZÁKLADNÍ POJMY	18
3.1.1 <i>Rovnice kontinuity (Zákon zachování hmotnosti)</i>	18
3.1.2 <i>Bernoulliho rovnice (Zákon zachování energie)</i>	19
3.1.3 <i>Výtok kapaliny z nádoby</i>	20
3.2 VÝTOK KAPALINY Z OTEVŘENÉ NÁDRŽE	21
3.3 VÝTOK KAPALINY Z NÁDRŽE S PŘÍTOKEM	22
3.4 SYSTÉM DVOU NÁDRŽÍ	23
3.5 SYSTÉM TŘÍ NÁDRŽÍ	26
3.6 VÝTOK KAPALINY Z KULOVÉ NÁDRŽE S PŘÍTOKEM	28
3.7 VÝTOK KAPALINY Z KUŽELOVÉ NÁDRŽE S PŘÍTOKEM	29
4 NUMERICKÉ METODY ŘEŠENÍ OBYČEJNÝCH DIFERENCIÁLNÍCH ROVNIC	32
4.1 RUNGOVY-KUTTOVY METODY	33
4.1.1 <i>Eulerova metoda</i>	33
4.1.2 <i>Rungovy-Kuttovy metody</i>	34
4.2 MNOHOKROKOVÉ METODY	34
4.2.1 <i>Adamsovy-Bashforthovy vzorce</i>	34
4.2.2 <i>Adamsovy-Moultonovy vzorce</i>	35

4.2.3	<i>Back differentiation formulas - BDF</i>	35
4.2.4	<i>Metody prediktor-korektor</i>	35
5	POUŽITÉ TECHNOLOGIE	37
5.1	XHTML.....	37
5.2	PHP	37
5.2.1	<i>JpGraph</i>	38
5.3	JAVASCRIPT	38
	PRAKTICKÁ ČÁST	39
6	UŽIVATELSKÝ POPIS ONLINE SIMULAČNÍHO MODELU	40
6.1	ONLINE SIMULAČNÍ MODEL	40
6.2	UMÍSTĚNÍ ONLINE SIMULAČNÍHO SYSTÉMU.....	41
6.3	ZADÁVÁNÍ PARAMETRŮ DANÉHO SYSTÉMU	41
6.4	TYPY SIMULACE	43
6.4.1	<i>Automatický odhad doby ustálení – omezení</i>	43
6.5	NUMERICKÉ METODY ŘEŠENÍ OBYČEJNÝCH DIFERENCIÁLNÍCH ROVNIC	45
6.6	ZOBRAZENÍ VÝSLEDKŮ SIMULAČNÍCH POKUSŮ	46
7	PRINCIPIÁLNÍ POPIS ONLINE SIMULAČNÍHO MODELU	47
7.1	ODHAD DOBY USTÁLENÍ.....	47
7.2	OŠETŘENÍ KRITICKÝCH STAVŮ.....	48
7.2.1	<i>Dělení nulou</i>	48
7.2.2	<i>Úplné vytečení kapaliny – nulová výška hladiny</i>	49
7.3	EXPORT PRŮBĚHU SIMULACE DO MS EXCELU	49
7.4	GRAFICKÝ VÝSTUP SIMULACE	51
7.5	OŠETŘENÍ VSTUPŮ	53
8	UKÁZKA ŘEŠENÍ KONKRÉTNÍHO PŘÍKLADU	54
8.1	ZADÁNÍ SYSTÉMU DVOU NÁDRŽÍ	54
8.2	ŘEŠENÍ PŘÍKLADU.....	54
8.2.1	<i>Eulerova metoda</i>	55
8.2.2	<i>Runge-Kutt čtvrtého řádu</i>	56
8.2.3	<i>Metoda prediktor-korektor</i>	57
8.3	ZOBRAZENÍ VÝSLEDKŮ SIMULACE	58
	ZÁVĚR	61
	CONCLUSION	62
	SEZNAM POUŽITÉ LITERATURY	63

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	65
SEZNAM OBRÁZKŮ	66
SEZNAM PŘÍLOH.....	67

ÚVOD

Významné místo mezi pomůckami využívanými při výuce mají dnes pomůcky elektronické. Jejich význam neustále roste s rozvojem moderních technologií a prostředků pro elektronickou podporu výuky. Jsou neustále posilovány výhody výpočetní techniky a zmenšovány jejich nevýhody – složitost a určitá nepřívětivost. Ve většině případu je elektronická podpora výuky účelnější, kdy řada problémů je vysvětlována animacemi, zvukovým doprovodem, videosekvencí atd. Vzájemná interaktivita bývá podpořena i testy z dané problematiky. Statický text v podobě knih nepodá a ani nemůže podat informace v dostatečně srozumitelné formě. Toto samozřejmě platí jen v některých oborech. Ale i v ostatních odvětvích se může jednat o zábavnější a přijatelnější formu výuky.

Pro svou diplomovou práci jsem zvolil téma Online simulační model propojených nádrží, jejíž výsledek bude sloužit jako elektronická podpora výuky předmětu simulace systémů.

Hlavním cílem mé práce je vytvoření online systému ve formě dynamických www stránek. Jenž bude obsahovat problematiku nádrží a jejich simulačních modelů.

Práce se skládá z části teoretické a praktické, celkově je členěna do osmi kapitol, z čehož prvních pět kapitol tvoří teoretickou část mé práce. První kapitola teoretické části se zabývá problematikou e-learningu, tedy elektronickou podporou výuky. Ve druhé kapitole najdeme přehled dostupných zdrojů o problematice nádrží a výskytu simulačních nástrojů na internetu. Sestavení matematických modelů jednotlivých typů nádrží a systémů propojených nádrží je zpracováno ve třetí kapitole teoretické části. Součástí této kapitoly je úvod do mechaniky tekutin. Obsahem čtvrté kapitoly teoretické části je řešení obyčejných diferenciálních rovnic, které jsou odvozeny v předešlé kapitole. Pátá kapitola se věnuje popisu použitých technologií k vytvoření zadaného online systému.

Praktická část práce je obsažena v posledních třech kapitolách. Šestá kapitola obsahuje uživatelský popis vytvořeného systému. Obsahem sedmé kapitoly je principiální popis důležitých částí online systému. V poslední kapitole je řešen konkrétní příklad systému dvou propojených nádrží.

I. TEORETICKÁ ČÁST

1 E-LEARNING

1.1 Definice e-learningu

Existuje řada definic, která vznikala postupem času s tím jak se měnilo samotné pojetí e-learningu, ale také s vývojem informačních a komunikačních technologií.

- *E-learning zahrnuje všechny formy výuky, které využívají jakýchkoli digitálních médií pro šíření studijních materiálů, výuku, učení a komunikaci mezi zúčastněnými stranami*

[14]

- *E-learning chápeme jako multimediální podporu vzdělávacího procesu s použitím moderních informačních a komunikačních technologií, které je zpravidla realizováno prostřednictvím počítačových sítí. Jeho základním úkolem je v čase i prostoru svobodný a neomezený přístup ke vzdělávání.*

[13]

- *E-learning označuje různé druhy učení podporované počítačem, zpravidla s využitím moderních technologických prostředků, především CD ROM.*

[15]

- *E-learning slouží lidem, společnostem a firmám jako prostředek, kterým se mohou sami vzdělat, bez pomoci nějakého lektora a to například v klidu domova a nemusí si zakupovat žádné několika stránkové knihy.*

[12]

1.2 Charakteristika e-learningu

E-learning se stává fenoménem našeho století. Využívá se pro podporu výuky jak u distančních tak i prezenčních forem výuky. Nejvíce se však e-learning využívá ve firemním vzdělávání. Ve své podstatě jde o multimediální interaktivní vzdělávání, které je praktikované prostřednictvím počítačových kurzů, které jsou distribuovány prostřednictvím CD-ROMů, Internetu, Intranetu apod.

Vzdělávání se uskutečňuje integrováním mluveného slova, audio a video sekvencí, schémat, grafiky, a v neposlední řadě také z různých testů. Čímž se zvyšuje efektivitu procesu učení - student nové informace přijímá více smysly, nejenom sluchově, ale také vizuálně. Účinnost se zvyšuje i aktivní účastí studenta na studijním programu - plněním konkrétních praktických úkolů a testů.

[16]

1.3 Formy e-learningu

Podle způsobu komunikace mezi pedagogem a studenty se setkáváme se dvěma formami e-learningu.

1.3.1 on-line (synchronní) e-learning

Výuka probíhá prostřednictvím tzv. virtuálních učeben, nabízí možnost vzájemné komunikace mezi všemi zúčastněnými navzájem. Komunikace mezi účastníky může probíhat pomocí audiokonferencí, videokonferencí nebo prostou textovou formou tzv. chatem. Tento způsob výuky je však omezen technickými možnostmi jako např. propustností datových sítí.

[16]

1.3.2 off-line (asynchronní) e-learning

V současnosti asi nejrozvinutější elektronická forma výuky. Student pracuje samostatně, není vázán časovým ani prostorovým vymezením studia. To sebou ale přináší vysoké nároky na motivaci k získávání nových poznatků. Informace a materiály jsou umístěny na internetu, odkud si je účastník může stáhnout.

[16]

1.4 Výhody e-learningu

- Časová nezávislost a individuální studium - student si sám volí dobu, kdy se bude vzdělávat, absolvuje kurzy dle vlastních potřeb, volí rychlost učení, typ a formu kurzu

- Variabilita a interaktivita – kurzy kombinují výklad s audio či videosekvencí. Studenti jsou zapojováni do výuky pomocí různých testů.
- Snížení nákladů - odpadají náklady týkající se dopravy na místo školení, ubytování, náklady na stravování, pronájem školících místností, osobu(y) lektora.
- Načasování – každý student má možnost volby doby ke studiu
- Zpětná vazba – přesné sledování průběhu kurzu, přehled znalostí, statistické výsledky testů
- Masovost a rychlost – pro uskutečnění kurzu není zapotřebí určitý počet studentů, naopak umožňuje realizaci aktuálních vzdělávacích potřeb studentů v krátkém časovém horizontu.
- Pružná reakce na požadavky aktualizace vzdělávacích modulů – obsah je průběžně aktualizován

[12], [16]

1.5 Nevýhody e-learningu

- Vysoké náklady – jedná se o jednorázové náklady na potřebné počítačové vybavení, řídicí systém a kurz
- Počítačová gramotnost – nutná je alespoň minimální počítačová gramotnost všech studentů (schopnost ovládat PC, pohybovat se v prostředí elektronického výukového programu)
- Osobní odpovědnost za rozvoj každého jednotlivce
- Vlastní motivace a sebemotivace k dalšímu vzdělávání
- Omezená výměna zkušeností a znalostí v průběhu učení
- Nemožnost navazování osobních kontaktů

[12], [16]

1.6 Standardy e-learningu

Pro vytváření e-learningových kurzů jsou stanoveny standardy, což jsou sady pravidel nebo procedur odsouhlasených a schválených standardizační organizací. Tato pravidla jsou potřebná především v oblasti tvorby kurzů a v oblasti nastavení komunikace mezi kurzy a řídicím systémem vzdělávání. Jsou důležitá pro poskytovatele řešení a vzdělávání a i pro uživatele a zákazníky. Díky garanci dodržení pravidel se tvůrci kurzů, vývojáři nástrojů, aplikací a řídicích systémů mohou věnovat řešení dalších vylepšení systémů, a to v jiných oblastech než je pouhé poskytování vzdělávacích obsahů. A také zákazníci mají svoji jistotu zaručené kompatibility zakoupených kurzů pro provozované systémy.

Mezi základní standardy patří:

AICC (Aviation Industry Computer-Based Training Committee) - mezinárodní asociace profesionálních technologicky-založených školení, vyvíjejících tréninkové směrnice pro letecký průmysl. AICC vyvíjí standardy pro Interoperabilitu školení počítačem a počítačem řízené školení, produktů průmyslových odvětví.

SCORM (The Sharable Courseware Object Reference Model) - množina specifikací, které při aplikaci na obsah kurzu vytvoří malé a znovupoužitelné výukové objekty (learning objects). Je to výsledek iniciativy Advanced Distributed Learning (ADL), SCORM-pružné moduly se mohou jednoduše spojit s jinými k vytvoření velmi modulárního úložiště výcvikových materiálů.

IMS (The Instructional Management Systems) - technická specifikace výměny dat mezi studentem, jeho kurzem a systémem pro řízení výuky. Iniciováno skupinou společností s cílem definování specifikací a přijetí otevřeného standardu pro výuku realizovanou Internetem.

IEEE (Institute of Electrical and Electronics Engineers) - největší profesní a standardizační organizace na světě, Pro počítačové sítě má největší význam standardizační orgán IEEE 802, který je specificky zaměřen na problematiku standardu lokálních sítí.

ADL (Advanced Distributed Learning) - Iniciativa amerického Ministerstva obrany k dosažení interoperability mezi počítačem a Internetově založeným výukovým softwarem, a to vývojem společné technické struktury, která by umožňovala jeho opětovné použití.

2 DOSTUPNÉ ZDROJE K DANÉ PROBLEMATICE

2.1 Literatura

Přehlednou příručku k problematice nádrží představuje publikace „Modelování a identifikace systémů“. Autorem je Petr Noskievič. V publikaci najdem kromě jiného úvod do mechaniky tekutin (Bernoulliho rovnice, rovnice kontinuity, tzv. toricelliho vztah), volný výtok z nádrže s přítokem a také systém tří propojených nádrží. Je zde vysvětlena dynamika zmíněných systémů. Dále obsahuje bloková schéma matematických modelů jednotlivých systémů vytvořených v programu Matlab-Simulink.

2.2 Elektronické zdroje

Pro hledání materiálů dané problematiky ve světě internetu jsem využíval vyhledávač Google. Do průzkumu jsem zahrnul i anglicky psané webové stránky. Podle mnoha zadávaných klíčových slov se převážná část nalezených informací týkala společnostmi nabízejících montáže a prodej nádrží. Za zmínku stojí:

- http://matlab.fei.tuke.sk/gulnServlet/JDS_index.html - realizace webové aplikace pro simulace jednoduchých dynamických soustav technické univerzity v Košicích. Obsahuje soubor simulací pro různé modely jednoduchých soustav. Skládá se z pěti příkladů dynamiky hydraulických soustav, z jednoho příkladu dynamiky tlaku vzduchu a jednoho příkladu regulace dynamiky tepelných procesů. Algoritmy jsou z programu Matlab exportovány pomocí Matlab Builder for Java do Java komponentů. Při simulacích využívá technologie Java, Java script a Java Server Pages. Bohužel v době vzniku diplomové práce je simulační část nefunkční. Lze však studovat teorii k jednotlivým modelům.

[5]

- Diplomová práce „Knihovna modelů technologických procesů“ - práce je dostupná ve studijní agendě na adrese www.stag.utb.cz. Autorem je Radim Pišan. V této diplomové práci je představena knihovna modelů technologických procesů, vytvářená v programovém prostředí MATLAB/SIMULINK. Knihovna je tvořena bloky zásobníků na kapalinu v různých konfiguracích (kulové, válcové a ve tvaru

trychtýře), průtočným výměníkem tepla a průtočným chemickým reaktorem.

[21]

- Bakalářská práce „Vytvoření skriptů pro webové rozhraní předmětu Analýza a simulace technologických procesů“ – práce je dostupná ve studijní agendě na adrese www.stag.utb.cz. Autorem je Petr Tomášek. Tato bakalářská práce se zabývá analýzou a simulací následujících technologických procesů: průtočný výměník tepla s promícháváním, zásobníky na kapalinu zapojené za sebou, trychtýřový zásobník s nekonstantním průřezem a kulový zásobník s nekonstantním průřezem. U těchto modelů se simulují statické a dynamické charakteristiky pomocí M-file souborů v programu Matlab.

[22]

V oblasti teorie se vyskytují nejčastěji základní informace k problematice mechaniky tekutin. Například Bernoulliho rovnice, rovnice kontinuity, výtok z nádoby atd.

3 MECHANIKA TEKUTIN

Pod pojmem tekutiny si většina z nás představí kapaliny, avšak pod tento pojem můžeme zahrnout i plyny. A to díky mnoha společným vlastnostem. Hlavní společný znak je tekutost, která je dána vzájemnou pohyblivostí částic. Vnitřní tření těchto částic určuje viskozitu, a proto různé látky mají různé schopnosti téci. Ideální kapalina je nestlačitelná a bez vnitřního tření. Ale skutečné kapaliny jsou stlačitelné, avšak méně stlačitelné než plyny. Kromě stlačitelnosti spočívá základní rozdíl v udržitelnosti objemu. Zatímco kapaliny díky zachování objemu vytváří vodorovnou plochu (hladinu), plyny nemají stálý objem a proto také nevytváří volný povrch.

3.1 Základní pojmy

Problematiku tekutin můžeme rozdělit na hydrostatiku a hydrodynamiku. Již podle názvu je patrné, že se jedná o popsání tekutin v rovnovážném stavu a tekutin v pohybu. Mezi nejnámější pojmy v hydrostatice vzpomeneme Pascalův zákon a Archimédovy zákony. Nás bude hlavně zajímat hydrodynamika, což je bezesporu obtížnější obor.

V našich úvahách se omezíme na jednoduchý model ideální kapaliny – bez vnitřního tření a nestlačitelnou. Proudící tekutinu můžeme popsat proudnicemi. Proudnicí myslíme křivku, jejíž tečna v každém bodě určuje směr rychlosti proudící částice. Dále také trajektorii, po níž se částice pohybuje v čase. U ideálních kapalin lze využít zákonů zachování – množství, hybnost a energie. Nás budou nejvíce zajímat zákony o zachování množství (rovnice kontinuity) a energie (Bernoulliho rovnice).

[6], [7]

3.1.1 Rovnice kontinuity (Zákon zachování hmotnosti)

Rovnici kontinuity budeme uvažovat zjednodušený tvar rovnice kontinuity, která popisuje vztah mezi rychlostí proudění v a obsahem průřezu S v jednom místě trubice, za podmínky ustáleného proudění ideální kapaliny. Jednoduše čím užší trubice, tím rychlejší proudění kapaliny.

Rovnice kontinuity pro jednorozměrné ustálené proudění, kdy platí $\frac{\partial \rho}{\partial t} = 0$, dále hustota $\rho = \rho(s)$, průřez $S = S(s)$ a rychlost $v = v(s)$ jsou pouze funkcí souřadnice s , má tvar

$$\frac{d}{ds}(\rho \cdot S \cdot v) = 0. \quad (3.1)$$

Po integraci podle souřadnice s platí

$$Q_m = \rho \cdot S \cdot v = konst, \quad (3.2)$$

Kde Q_m je hmotnostní tok. V každém průřezu jedné a téže proudové trubice musí být splněna rovnice

$$\rho_1 \cdot S_1 \cdot v_1 = \rho_2 \cdot S_2 \cdot v_2 = \rho \cdot S \cdot v = konst \quad (3.3)$$

Pro nestlačitelné kapaliny platí $\rho = konst$, takže rovnice se zjednoduší na tvar

$$Q = S \cdot v = konst., \quad (3.4)$$

kde Q je průtok a udává objem kapaliny, který protekl za jednotku času

$$Q = \frac{V}{t}. \quad (3.5)$$

[1], [6] a [8]

3.1.2 Bernoulliho rovnice (Zákon zachování energie)

Bernoulliho rovnice vyjadřuje zákon o zachování energie při proudění dokonalé kapaliny za působení tíhového zrychlení. Bernoulliho jev můžeme také popsat slovy jako, čím užší trubice, tím menší tlak a vyšší rychlost (změna rychlosti s měnícím se průřezem je důsledek rovnice kontinuity).

Odvození vychází ze zákona zachování energie, za předpokladu že součet kinetické, tlakové a polohové energie je konstantní

$$W_K + W_p + W_h = konst. \quad (3.6)$$

První člen představuje kinetickou energii hmotnostní jednotky kapaliny, druhý člen odpovídá tlakové energii hmotnostní jednotky kapaliny a třetí člen vyjadřuje polohovou energii hmotnostní jednotky kapaliny. Za těchto předpokladů dostáváme rovnici o tvaru

$$\frac{1}{2} m \cdot v^2 + p \cdot \frac{m}{\rho} + m \cdot g \cdot h = konst. \quad (3.7)$$

Kde m je hmotnost pohybující se kapaliny, v rychlost proudící kapaliny, ρ je hustota kapaliny, p je tlak kapaliny a h je výška hladiny. Při řešení proudění kapaliny je výhodnější počítat energie vztažené na jednotku hmotnosti kapaliny a energie vztažená na jeden kilogram kapalina je měrná energie $\frac{W}{m}$. Můžeme tedy přepsat rovnici do tvaru

$$\frac{v^2}{2} + \frac{p}{\rho} + g \cdot h = konst. \quad (3.8)$$

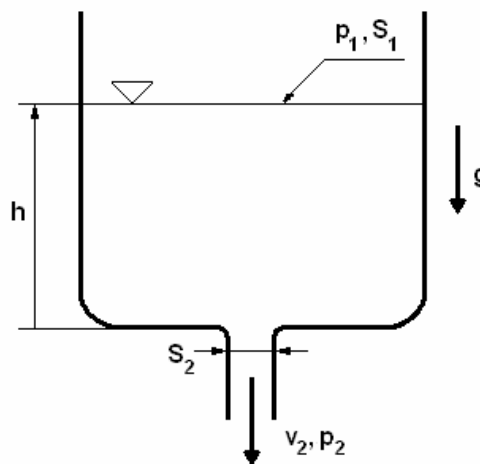
Vyjádřením zákona o zachování energie proudící ideální kapaliny ve tvaru měrných energií, dostáváme Bernoulliho rovnici ve tvaru

$$\frac{v_1^2}{2} + \frac{p_1}{\rho} + g \cdot h_1 = \frac{v_2^2}{2} + \frac{p_2}{\rho} + g \cdot h_2 = konst \quad (3.9)$$

[1], [6], [7], [8] a [19]

3.1.3 Výtok kapaliny z nádoby

Nyní odvodíme vztah pro výpočet výtokové rychlosti (tzv. Torricelliho vztah) podle obrázku 3.1. Mějme dva průřezy, hladinu, kde velikost rychlosti je zanedbatelná, tlak p_1 a výšku h . Výtokový otvor, kde rychlost má velikost v_2 a tlak p_2 .



Obrázek 3.1.: Výtok z nádoby – odvození Torricelliho vztahu

Použijeme Bernoulliho rovnici (3.9) na oba průřezy.

$$\frac{p_1}{\rho} + \frac{v_1^2}{2} + g \cdot h = \frac{p_2}{\rho} + \frac{v_2^2}{2} + g \cdot h_z \quad (3.10)$$

Proměnná v_2 nám představuje hledanou výtokovou rychlost. Tudiž můžeme využít rovnici kontinuity na oba průřezy a vyloučit rychlost v_1 , dosazením $v_1 = \frac{S_2 \cdot v_2}{S_1}$. Proměnná h_z představuje ztrátovou výšku, tj. ztráta ve výtokovém otvoru za pomoci ztrátového součinitele ξ , $h_z = \xi \cdot \frac{v_2^2}{2 \cdot g}$. Po opětovném dosazení dostáváme z Bernoulliho rovnice

$$\frac{p_1}{\rho} + \frac{1}{2} \left(\frac{S_2}{S_1} \right)^2 \cdot v_2 + g \cdot h = \frac{p_2}{\rho} + \frac{v_2^2}{2} + g \cdot \xi \cdot \frac{v_2^2}{2 \cdot g} \quad (3.11)$$

Vyjádřením v_2 dostáváme

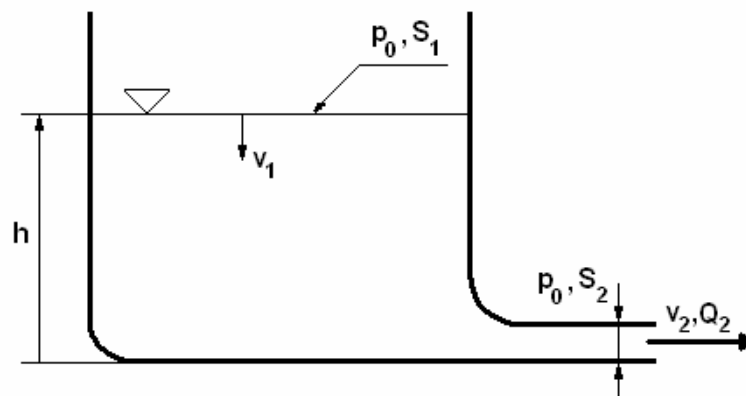
$$v_2 = \sqrt{2 \cdot \frac{g \cdot h + \frac{p_1 - p_2}{\rho}}{1 + \xi - \left(\frac{S_1}{S_2} \right)^2}} \quad (3.12)$$

Pokud je nádoba otevřená, pak tlaky nad hladinou a mimo nádobu budou stejné $p_1 = p_2 = p_0$. Uvažujeme ideální kapalinu, tudíž $\xi = 0$. Vztah nyní přechází ve Torricelliho vztah.

$$v = \sqrt{2 \cdot g \cdot h} \quad (3.13)$$

[1], [9]

3.2 Výtok kapaliny z otevřené nádrže



Obrázek 3.2.: Volný výtok z otevřené nádrže

Výška hladiny se bude v tomto modelu, který vidíme na obrázku 3.2, měnit vlivem výtoku. Po odvození výtokové rychlosti v předchozí kapitole, bude rychlost v_2 podle vzorce (3.13)

$$v_2 = \sqrt{2 \cdot g \cdot h} . \quad (3.14)$$

Rychlost v_1

$$v_1 = -\frac{dh}{dt} \quad (3.15)$$

a dosazením do rovnice kontinuity dostáváme

$$S_1 \cdot \left(-\frac{dh}{dt} \right) = S_2 \cdot \sqrt{2 \cdot g \cdot h} . \quad (3.16)$$

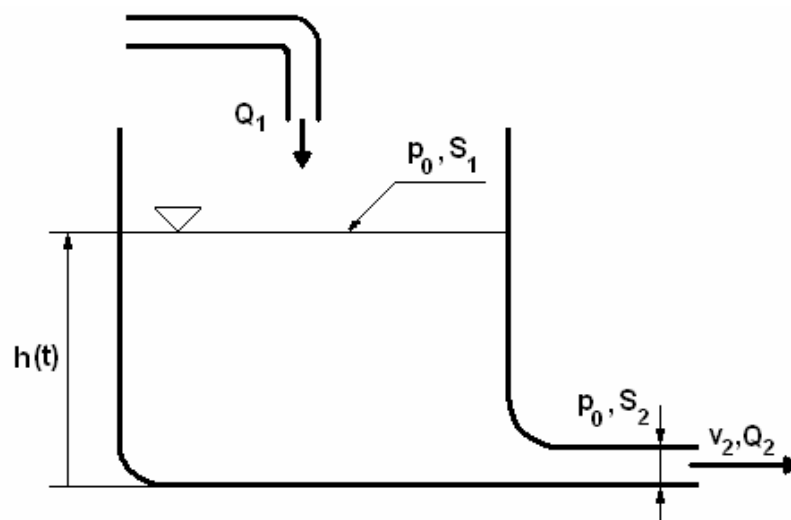
Výška hladiny h , pak bude tedy vyjádřena homogenní nelineární diferenciální rovnicí 1.řádu

$$\frac{dh}{dt} + \frac{S_2}{S_1} \cdot \sqrt{2 \cdot g \cdot h} = 0. \quad (3.17)$$

[1], [9]

3.3 Výtok kapaliny z nádrže s přítokem

Uvažujme otevřenou nádrž s volným výtokem a přítokem. Ze vztahů, které jsme si odvodily v předchozích kapitolách, můžeme podle obrázku 3.3 napsat rovnici pro změnu výšky hladiny tohoto systému.



Obrázek 3.3.: Výtok z otevřené nádrže s přítokem

Rovnice pro změnu výšky hladiny, kde Q_1 je přítok do nádrže a Q_2 je volný výtok z nádrže, má tvar

$$S_1 \cdot \frac{dh}{dt} = Q_1 - Q_2, \quad (3.18)$$

po dosazení a úpravě dostaneme rovnici, která popisuje změnu výšky hladiny, jako nelineární diferenciální rovnici

$$\frac{dh}{dt} = \frac{1}{S_1} \cdot Q_1 - \frac{S_2}{S_1} \cdot \sqrt{2 \cdot g \cdot h}. \quad (3.19)$$

Dále nás bude zajímat ustálená výška hladiny h_{ust} . Ustálený stav vyšetříme za podmínek konstantního přítoku Q_1 a vztahu pro rovnovážný stav $\frac{dh}{dt} = 0$.

$$\frac{S_2}{S_1} \cdot \sqrt{2 \cdot g \cdot h} - \frac{Q_1}{S_1} = 0 \quad (3.20)$$

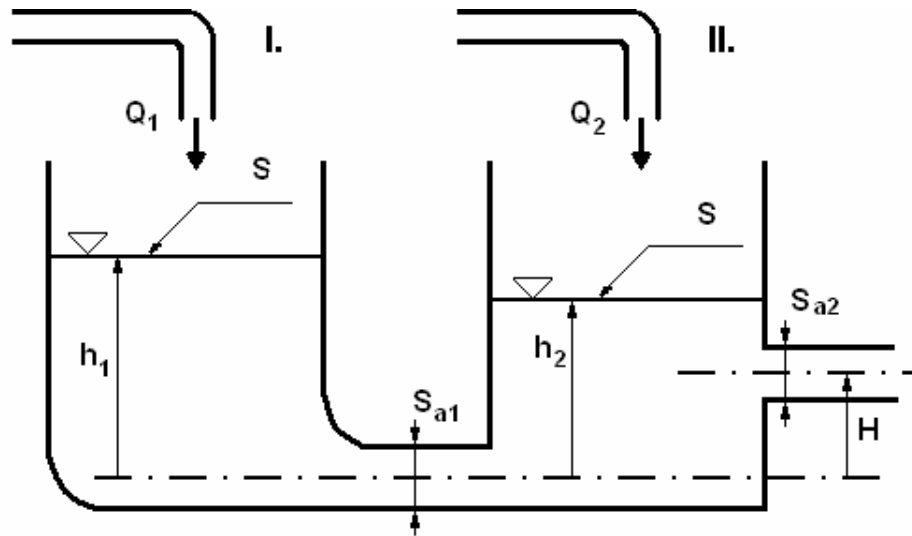
odtud

$$h_{ust} = \frac{Q_1^2}{2 \cdot g \cdot S_2^2}. \quad (3.21)$$

[1], [5]

3.4 Systém dvou nádrží

Mějme dvě nádrže o stejném průřezu S , jejichž model je na obrázku 3.4. Nádrže jsou propojeny spojovacím potrubím o průřezu S_{a1} , dále uvažujme výtokové potrubí s průřezem S_{a2} ve výšce H , s konstantními přítoky Q_1 a Q_2 . Sestavení matematického modelu vychází z popisu změn výšek hladin v jednotlivých nádržích.



Obrázek 3.4.: Systém dvou nádrží s přítokem a výtokem

Pro změnu výšky hladiny v první nádrži platí

$$S \cdot \frac{dh_1}{dt} = Q_1 - S_{a1} \cdot v_1 \quad (3.22)$$

a pro změnu výšky hladiny v druhé nádrži musíme brát v úvahu i přítok z první nádrže. Potom tedy platí

$$S \cdot \frac{dh_2}{dt} = Q_2 + S_{a1} \cdot v_1 - S_{a2} \cdot v_2, \quad (3.23)$$

kde v_1 je výtoková rychlost z první nádrže do druhé a v_2 je výtoková rychlost z druhé nádrže.

$$v_1 = \sqrt{2 \cdot g \cdot (h_1 - h_2)} \quad \text{pro } h_1 \geq h_2 \quad (3.24)$$

$$v_1 = \sqrt{2 \cdot g \cdot (h_2 - h_1)} \quad \text{pro } h_2 \geq h_1 \quad (3.25)$$

$$v_2 = \sqrt{2 \cdot g \cdot (h_2 - H)} \quad \text{pro } h_2 > H \quad (3.26)$$

$$v_2 = 0 \quad \text{pro } H > h_2 \quad (3.27)$$

Pro snadnější zápis předchozích rovnic definujeme tzv. Föllingerův vztah

$$v = F(\Delta h) = \text{sgn } \Delta h \sqrt{2 \cdot g \cdot |\Delta h|}. \quad (3.28)$$

Po tomto zjednodušení nám zůstanou výsledné rovnice nezměněny jak pro případ $h_1 \geq h_2$ tak i $h_2 \geq h_1$. A tak můžeme zapsat výslednou rovnici pro změnu výšky hladiny v první nádrži jako

$$\frac{dh_1}{dt} = \frac{1}{S} \cdot Q_1 - \frac{S_{a1}}{S} \cdot F(h_1 - h_2) \quad (3.29)$$

a změnu výšky hladiny v druhé nádrži jako

$$\frac{dh_2}{dt} = \frac{1}{S} \cdot Q_2 + \frac{S_{a1}}{S} \cdot F(h_1 - h_2) - \frac{S_{a2}}{S} \cdot F(h_2 - H). \quad (3.30)$$

Systém dvou nádrží můžeme popsat pomocí dvou stavových rovnic

$$\dot{h}_1 = \frac{1}{S} \cdot Q_1 - \frac{S_{a1}}{S} \cdot F(h_1 - h_2) \quad (3.31)$$

$$\dot{h}_2 = \frac{1}{S} \cdot Q_2 + \frac{S_{a1}}{S} \cdot F(h_1 - h_2) - \frac{S_{a2}}{S} \cdot F(h_2 - H). \quad (3.32)$$

Vypočteme ustálené hodnoty výšek hladin pro konstantní přítoky Q_1 a Q_2 , za podmínky rovnovážného stavu $\dot{h}_1 = 0$ a $\dot{h}_2 = 0$.

$$0 = Q_1 - S_{a1} \cdot F(h_1 - h_2) \quad (3.33)$$

$$0 = Q_2 + S_{a1} \cdot F(h_1 - h_2) - S_{a2} \cdot F(h_2 - H) \quad (3.34)$$

Po dosazení za nelineární funkci F vyplývá po úpravě rovnic

$$\sqrt{2 \cdot g \cdot (h_1 - h_2)} = \frac{Q_1}{S_{a1}} \quad (3.35)$$

$$\sqrt{2 \cdot g \cdot (h_2 - H)} = \frac{Q_1 + Q_2}{S_{a2}}. \quad (3.36)$$

Z posledního vztahu lze určit ustálenou výšku h_2

$$h_{2ust} = \frac{(Q_1 + Q_2)^2}{S_{a2}^2 \cdot 2 \cdot g} + H. \quad (3.37)$$

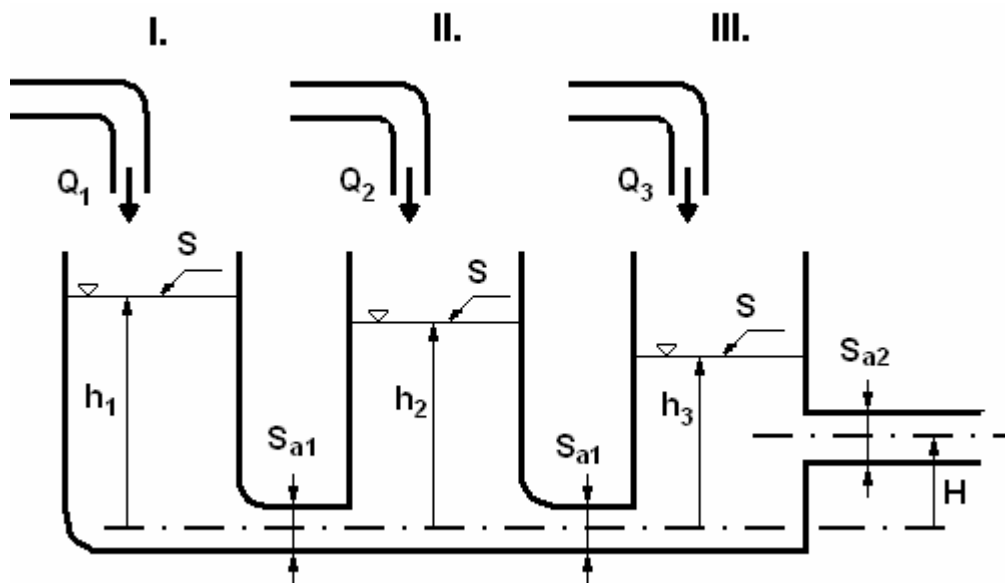
Pro h_1 platí

$$h_{lust} = \frac{(Q_1 + Q_2)^2}{S_{a2}^2 \cdot 2 \cdot g} + H + \frac{Q_1^2}{S_{a1}^2 \cdot 2 \cdot g} \quad (3.38)$$

[1]

3.5 Systém tří nádrží

Model systému tří nádrží je znázorněn na obrázku 3.5. Nádrže mají stejný průřez S a jsou propojeny spojovacím potrubím o průřezu S_{a1} . Dále uvažujeme výtokové potrubí s průřezem S_{a2} ve výšce H ve třetí nádrži a konstantní přítoky Q_1, Q_2 a Q_3 pro jednotlivé nádrže.



Obrázek 3.5.:Systém tří nádrží s přítokem a výtokem

Sestavení matematického modelu vychází z popisu změn výšek hladin v jednotlivých nádržích.

Pro změnu výšky hladiny v první nádrži platí

$$S \cdot \frac{dh_1}{dt} = Q_1 - S_{a1} \cdot v_1, \quad (3.39)$$

pro změnu výšky hladiny v druhé nádrži musíme brát v úvahu i přítok z první nádrže a výtok do třetí. Potom tedy platí vztah

$$S \cdot \frac{dh_2}{dt} = Q_2 + S_{a1} \cdot v_1 - S_{a1} \cdot v_2. \quad (3.40)$$

Změna výšky hladiny v třetí nádrži je dána přítokem z druhé nádrže a výtokem podle vztahu

$$S \cdot \frac{dh_3}{dt} = Q_3 + S_{a1} \cdot v_2 - S_{a2} \cdot v_3. \quad (3.41)$$

Po dosazení nelineární funkce F podle Föllingerova vztahu, můžeme zapsat rovnice pro změny výšek hladin jednotlivých nádrží ve tvaru

$$\frac{dh_1}{dt} = \frac{1}{S} \cdot Q_1 - \frac{S_{a1}}{S} \cdot F(h_1 - h_2), \quad (3.42)$$

$$\frac{dh_2}{dt} = \frac{1}{S} \cdot Q_2 + \frac{S_{a1}}{S} \cdot F(h_1 - h_2) - \frac{S_{a1}}{S} \cdot F(h_2 - h_3), \quad (3.43)$$

$$\frac{dh_3}{dt} = \frac{1}{S} \cdot Q_3 + \frac{S_{a1}}{S} \cdot F(h_2 - h_3) - \frac{S_{a2}}{S} \cdot F(h_3 - H). \quad (3.44)$$

System tří nádrží můžeme popsat třemi stavovými rovnicemi

$$\dot{h}_1 = \frac{1}{S} \cdot Q_1 - \frac{S_{a1}}{S} \cdot F(h_1 - h_2), \quad (3.45)$$

$$\dot{h}_2 = \frac{1}{S} \cdot Q_2 + \frac{S_{a1}}{S} \cdot F(h_1 - h_2) - \frac{S_{a1}}{S} \cdot F(h_2 - h_3), \quad (3.46)$$

$$\dot{h}_3 = \frac{1}{S} \cdot Q_3 + \frac{S_{a1}}{S} \cdot F(h_2 - h_3) - \frac{S_{a2}}{S} \cdot F(h_3 - H). \quad (3.47)$$

Ustálené hodnoty výšek hladin pro konstantní přítoky Q_1 , Q_2 a Q_3 vyplývají z podmínek rovnovážného stavu pro $\dot{h}_1 = 0$, $\dot{h}_2 = 0$ a $\dot{h}_3 = 0$.

$$0 = -S_{a1} \cdot F(h_1 - h_2) + Q_1 \quad (3.48)$$

$$0 = S_{a1} \cdot F(h_1 - h_2) - S_{a1} \cdot F(h_2 - h_3) + Q_2 \quad (3.49)$$

$$0 = S_{a1} \cdot F(h_2 - h_3) - S_{a2} \cdot F(h_3 - H) + Q_3 \quad (3.50)$$

Po dosazení za nelineární funkci F vyplývá po úpravě rovnic

$$\sqrt{2 \cdot g \cdot (h_1 - h_2)} = \frac{Q_1}{S_{a1}}, \quad (3.51)$$

$$\sqrt{2 \cdot g \cdot (h_2 - h_3)} = \frac{Q_1 + Q_2}{S_{a1}}, \quad (3.52)$$

$$\sqrt{2 \cdot g \cdot (h_3 - H)} = \frac{Q_1 + Q_2 + Q_3}{S_{a2}}. \quad (3.53)$$

Z posledního vztahu lze určit ustálenou výšku h_3

$$h_{3ust} = \frac{(Q_1 + Q_2 + Q_3)^2}{S_{a2}^2 \cdot 2 \cdot g} + H. \quad (3.54)$$

Pro h_2 platí

$$h_{2ust} = \frac{(Q_1 + Q_2 + Q_3)^2}{S_{a2}^2 \cdot 2 \cdot g} + H + \frac{(Q_1 + Q_2)^2}{S_{a1}^2 \cdot 2 \cdot g} \quad (3.55)$$

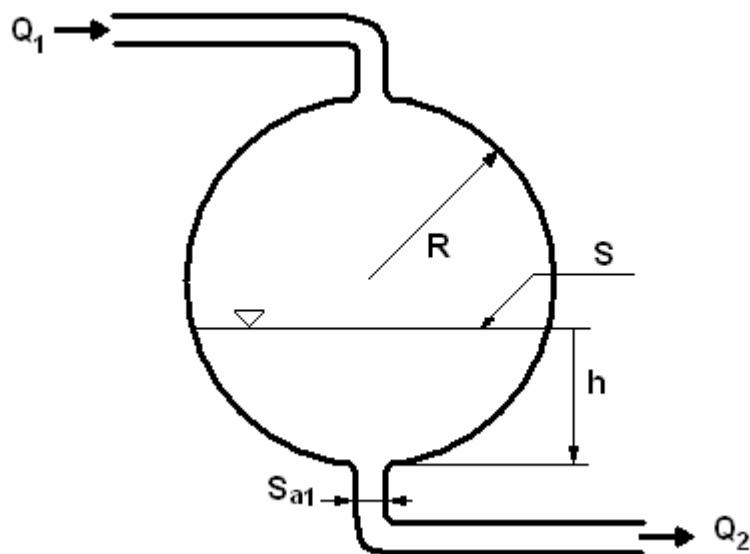
a pro h_1 dostáváme

$$h_{1ust} = \frac{(Q_1 + Q_2 + Q_3)^2}{S_{a2}^2 \cdot 2 \cdot g} + H + \frac{(Q_1 + Q_2)^2}{S_{a1}^2 \cdot 2 \cdot g} + \frac{Q_1^2}{S_{a1}^2 \cdot 2 \cdot g}. \quad (3.56)$$

[1], [5]

3.6 Výtok kapaliny z kulové nádrže s přítokem

Model kulové nádrže, který je na obrázku 3.6, má poloměr R , přítok Q_1 a výtok Q_2 . Výtok kapaliny Q_2 je dán průřezem výtokového otvoru S_{a1} a výška hladiny je určena hodnotou h .



Obrázek 3.6.: Výtok z kulové nádrže s přítokem

Vztah pro změnu výšky hladiny za určitou dobu můžeme pro kulovou nádobu napsat jako rozdíl přítokového množství kapaliny a kapaliny, která za daný čas vytekla.

$$S \cdot \frac{dh}{dt} = Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h} \quad (3.57)$$

Ovšem v případě kulové nádrže se zároveň se změnou výšky hladiny mění i poloměr průřezové plochy nádoby S . Tuto změnu průřezu můžeme přepsat do tvaru

$$S = \pi(2 \cdot R \cdot h - h^2). \quad (3.58)$$

Po dosazení do rovnice (3.57) a úpravě dostáváme diferenciální rovnici 1. řádu popisující změnu výšky hladiny v kulové nádrži.

$$\frac{dh}{dt} = \frac{Q_1 - \pi \cdot r^2 \cdot \sqrt{2 \cdot g \cdot h}}{\pi(2 \cdot R \cdot h - h^2)} \quad (3.59)$$

Nakonec nás zajímá ustálená výška hladiny h_{ust} . Ustálený stav vyšetříme za podmínek konstantního přítoku Q_1 a vztahu pro rovnovážný stav $\frac{dh}{dt} = 0$.

$$0 = Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h} \quad (3.60)$$

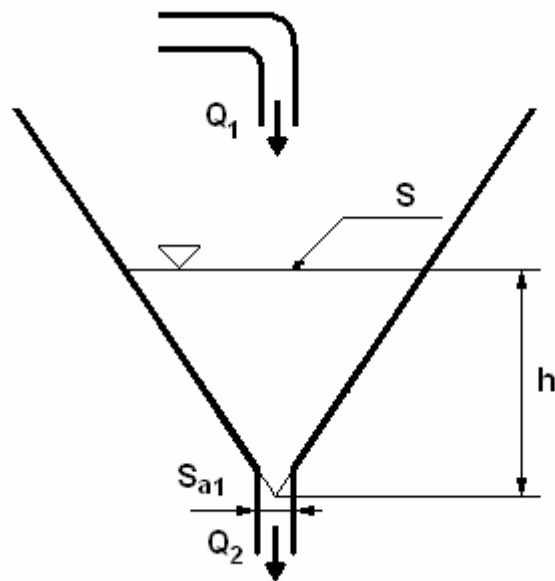
Po úpravě

$$h_{ust} = \frac{Q_1^2}{2 \cdot g \cdot S_{a1}^2}. \quad (3.61)$$

[1], [5]

3.7 Výtok kapaliny z kuželové nádrže s přítokem

Na obrázku 3.7 je zobrazen model kuželové nádrže s přítokem a volným odtokem. Kde Q_1 značí konstantní přítok kapaliny do nádrže, Q_2 množství výtokové kapaliny, S_{a1} průřez výtokového otvoru a h výška hladiny. Stejně jako v případě kulové nádrže, se jedná o model, ve kterém se s výškou hladiny h mění i celkový průřez nádrže S .



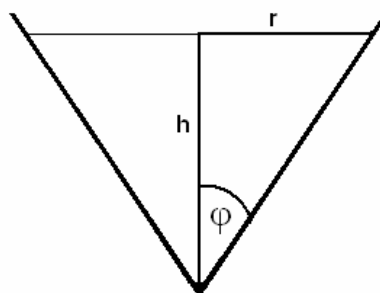
Obrázek 3.7.: Výtok z kuželové nádrže s přítokem

Uřídíme rovnici popisující daný model

$$S \cdot \frac{dh}{dt} = Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h} . \quad (3.62)$$

Velikost průřezu S je samozřejmě závislá na charakteru nádrže, respektive na úhlu kuželové nádrže. Pro velikost plochy průřezu S můžeme napsat vztah

$$S = \pi \cdot r^2 \quad (3.63)$$



Obrázek 3.8.: Kuželová nádrž – výpočet plochy průřezu

Způsob jakým zjistíme aktuální výšku průřezu je patrné z obrázku 3.8, kdy můžeme vyjádřit poloměr r ze vztahu

$$\tan \varphi = \frac{r}{h} . \quad (3.64)$$

Po dosazení těchto vztahů do rovnice (3.61) dostáváme diferenciální rovnici 1. řádu popisující změnu výšky hladiny v kuželové nádrži.

$$\pi \cdot (\tan \varphi \cdot h)^2 \cdot \frac{dh}{dt} = Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h} \quad (3.65)$$

Po úpravě (3.65) získáváme přehlednější tvar výsledné diferenciální rovnice popisující dynamiku systému kuželové nádrže.

$$\frac{dh}{dt} = \frac{Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h}}{\pi \cdot (\tan \varphi \cdot h)^2} \quad (3.66)$$

Nakonec, stejným způsobem jako u předchozích systémů, za podmínky rovnovážného stavu $\frac{dh}{dt} = 0$ a konstantního přítoku Q_1 určíme výšku hladiny v ustáleném stavu.

$$0 = Q_1 - S_{a1} \cdot \sqrt{2 \cdot g \cdot h} \quad (3.67)$$

Odtud

$$h_{ust} = \frac{Q_1^2}{2 \cdot g \cdot S_{a1}^2}. \quad (3.68)$$

Při pohledu na rovnici (3.68) je zřejmé, že systémy obsahující jednu nádrž s konstantním přítokem a volným výtokem, mají stejný vztah pro výšku hladiny ustálení bez ohledu na tvar nádrže. Vztahy (3.21), (3.61) a (3.68) jsou tedy totožné.

4 NUMERICKÉ METODY ŘEŠENÍ OBYČEJNÝCH DIFERENCIÁLNÍCH ROVNIC

V předchozí kapitole jsme získali diferenciální rovnice popisující jednotlivé modely. Nyní se budeme zabývat metodami, které vedou k jejím řešení. Diferenciální rovnice můžeme slovy popsat jako rovnice, ve kterých se proměnné vyskytují jako derivace funkcí. Diferenciální rovnice popisují většinu fyzikálních jevů a dělíme je na obyčejné diferenciální rovnice a parciální diferenciální rovnice. V této kapitole se budeme zabývat obyčejnými diferenciálními rovnicemi, ve kterých se vyskytuje derivace hledané funkce jedné proměnné. Parciální diferenciální rovnice obsahují derivace hledané funkce podle více proměnných.

Diferenciální rovnice není často možné řešit analyticky, a když už dokážeme najít vzorce pro analytické řešení, bývají často natolik složité, že práce s nimi je velice obtížná. V těchto případech používáme metody numerického řešení obyčejných diferenciálních rovnic (dále ODR). Z předchozí kapitoly je zřejmé, že se dále můžeme omezit pouze na diferenciální rovnice prvního řádu, ve kterých je derivace přímo vyjádřena. Budeme tedy hledat reálnou funkci

$$\frac{dy(x)}{dx} = f(x, y). \quad (4.1)$$

Nezbytnou součástí zadání je tzv. počáteční podmínka, která má tvar

$$y(x_0) = y_0. \quad (4.2)$$

Všechny následující metody lze zobecnit i na soustavy diferenciálních rovnic, např. systém dvou vzájemně propojených nádrží (viz. kapitola 3.1.6). Soustava dvou rovnic pro dvě neznámé funkce $y_1(x)$ a $y_2(x)$ má tvar

$$\frac{dy_1}{dx} = f(x_1, y_1, y_2) \quad y_1(x_0) = y_{10} \quad (4.3)$$

$$\frac{dy_2}{dx} = f(x_1, y_1, y_2) \quad y_2(x_0) = y_{20}. \quad (4.4)$$

Podobně postupujeme pro soustavy více rovnic.

Numerické metody můžeme rozdělit na jednokrokové a mnohokrokové. Při řešení vycházíme z počátečních podmínek v bodě x_0 a hledáme řešení v dalších krocích. U jednokrokových metod používáme k výpočtu nové hodnoty pouze jednu předcházející hodnotu.

$$y_{n+1} = f(x_n, y_n, x_{n+1}) \quad (4.5)$$

Kdežto u mnohokrokové metody používáme n-předchozích hodnot.

$$y_{n+1} = f(x_{n+1}, x_n, y_n, x_{n-1}, y_{n-1}, \dots, x_{n-i+1}, y_{n-i+1}) \quad (4.6)$$

[2], [7] a [10]

4.1 Rungovy-Kuttovy metody

4.1.1 Eulerova metoda

Jednoduchá jednokroková metoda pro řešení diferenciálních rovnic. Nahrazením derivace v diferenciální rovnici (4.1) její aproximací dostáváme při vyjádření y_{n+1} rekurentní vzorec Eulerovy metody

$$y_{n+1} = y_n + h \cdot f(x_n, y_n). \quad (4.7)$$

Eulerova metoda je sice jednoduchá, ale také velice nepřesná. Pro dosažení rozumné přesnosti musíme volit velký krok, což má za následek velký objem výpočtů. Nepřesnost je způsobena konstantní hodnotou derivace $f(x_n, y_n)$ na celém intervalu při kroku z x_n do x_{n+1} . Tuto nepřesnost se snaží odstranit modifikace Eulerovy metody. První modifikace se snaží vystihnout změnu derivace tím, že místo derivace na začátku intervalu použije derivaci uprostřed intervalu.

$$y_{n+1} = y_n + h \cdot f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}h \cdot f(x_n, y_n)\right) \quad (4.8)$$

Druhá modifikace používá průměr z derivace na začátku a na konci intervalu

$$y_{n+1} = y_n + \frac{1}{2}h \cdot [f(x_n, y_n) + f(x_n + h, y_n + hf(x_n, y_n))] \quad (4.9)$$

[7], [10]

4.1.2 Rungovy-Kuttovy metody

Dalšími modifikacemi Eulerovy metody dostáváme Runge-Kuttovy metody. Jsou to metody vyšších řádů, kdy získáváme několik odhadů derivací k_i v různých bodech. Nejčastěji používanou metodou je metoda čtvrtého řádu, tzn. odhad derivací provádíme ve čtyřech různých bodech.

$$\begin{aligned}
 k_1 &= hf(x_n, y_n) \\
 k_2 &= hf\left(x_n + \frac{1}{2}h, y_n + \frac{h}{2} \cdot k_1\right) \\
 k_3 &= hf\left(x_n + \frac{1}{2}h, y_n + \frac{h}{2} \cdot k_2\right) \\
 k_4 &= hf(x_n + h, y_n + h \cdot k_3) \\
 y_{n+1} &= y_n + h \cdot \left(\frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}\right)
 \end{aligned} \tag{4.10}$$

Obecně můžeme napsat, že zvýšením řádu dosáhneme podobné přesnosti jako u Eulerovy metody, avšak při podstatně větším kroku. Tedy i podstatně menšího objemu výpočtů. Ale také platí, že zvýšením řádu u jednokrokových metod narůstá objem výpočtů, protože v každém kroku je potřeba počítat hodnoty ve více bodech.

[7], [10]

4.2 Mnohokrokové metody

Dosud všechny uvedené metody byly jednokrokové. Mnohokrokové metody jak již bylo řečeno využívají k výpočtu nové hodnoty více předchozích hodnot.

4.2.1 Adamsovy-Bashforthovy vzorce

Adamsovy-Bashforthovy vzorce jsou explicitní vícekrokové metody, které lze odvodit integrací interpolačního polynomu sestaveného pro derivaci řešení diferenciální rovnice. Níže uvedené vzorce jsou jednobodové, lišící se v počtu kroků.

$$y_{n+1} = y_n + \frac{1}{2}h \cdot (3f_n - f_{n-1}) \tag{4.11}$$

$$y_{n+1} = y_n + \frac{1}{12}h \cdot (23f_n - 16f_{n-1} + 5f_{n-2}) \quad (4.12)$$

$$y_{n+1} = y_n + \frac{1}{24}h \cdot (55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}) \quad (4.13)$$

[7], [10]

4.2.2 Adamsovy-Moultonovy vzorce

Adamsovy-Moultonovy vzorce jsou implicitní vícekrokové metody, které lze odvodit podobným postupem jako Adamsovy-Bashforthovy vzorce. Například:

$$y_{n+1} = y_n + \frac{1}{2}h \cdot (f_{n+1} + f_n) \quad (4.14)$$

$$y_{n+1} = y_n + \frac{1}{12}h \cdot (5f_{n+1} + 8f_n - f_{n-1}) \quad (4.15)$$

[7], [10]

4.2.3 Back differentiation formulas - BDF

Předchozí vícekrokové metody jsou založené na numerické integraci. Metody BDF jsou založeny na numerické derivaci, v nichž se hledaná funkce $y(t)$ na levé straně diferenciální rovnice nahradí interpolačním polynomem a ten se poté derivuje.

$$y_{n+1} = \frac{4}{3}y_n - \frac{1}{3}y_{n-1} + \frac{2}{3}h \cdot f_{n+1} \quad (4.16)$$

$$y_{n+1} = \frac{18}{11}y_n - \frac{9}{11}y_{n-1} + \frac{2}{11}y_{n-2} + \frac{6}{11}h \cdot f_{n+1} \quad (4.17)$$

[11]

4.2.4 Metody prediktor-korektor

Výše uvedené vícekrokové metody se nejčastěji používají způsobem, který se nazývá prediktor-korektor. Prediktorem (explicitní metoda) označujeme prvotní odhad nové hodnoty. Tento odhad slouží jako startovací hodnota pro korektor (implicitní metoda) a prostými iteracemi korektoru tento odhad zpřesníme. Například pro prediktor použijeme

jednoduchou Eulerovu metodu (4.7) a pro korektor Adamsovu-Moultenovu implicitní metodu (4.14)

$$\text{Prediktor:} \quad y_{n+1}^P = y_n + h \cdot f(x_n, y_n) \quad (4.18)$$

$$f_{n+1}^P = f(x_{n+1}, y_{n+1}^P) \quad (4.19)$$

$$\text{Korektor:} \quad y_{n+1}^C = y_n + \frac{1}{2}h \cdot (f_{n+1}^P + f_n) \quad (4.20)$$

Pro výpočet nové hodnoty y_{n+1} můžeme napsat následující výpočetní postup

Pro $n = 1, 2, \dots, i-1$ hodnota y_{n+1}

1. Určíme počáteční odhad řešení y_{n+1}^P pomocí vzorce (4.18)
2. Počáteční odhad zpřesníme iterací korektoru (4.20)

$$y_{n+1}^{k+1} = y_n + \frac{1}{2}h \cdot (f(x_{n+1}, y_{n+1}^k) + f_n), \quad k = 0, 1, 2, \dots$$

Výsledná hodnota korektoru slouží opět jako vstup pro prediktor.

[2], [7] a [10]

5 POUŽITÉ TECHNOLOGIE

5.1 XHTML

XHTML (zkratka anglického *extensible hypertext markup language* – „rozšiřitelný hypertextový značkovací jazyk“) je značkovací jazyk pro tvorbu hypertextových dokumentů v prostředí WWW vyvinutý W3C. W3C je mezinárodní konsorcium, jehož členové společně s veřejností vyvíjejí webové standardy pro World Wide Web. Jeho první specifikace se označuje XHTML 1.0, jejíž cílem bylo převedení staršího jazyka HTML tak, aby vyhovoval podmínkám tvorby XML dokumentů Existuje ve třech verzích: *Strict*, *Transitional* a *Frameset*.

- Strict se používá, pokud chcete strukturovaný dokument osvobozený od formátovacích značek souvisejících s rozvržením stránky.
- Transitional je přechodnou definicí typu dokumentu pro webové stránky, který vám umožní používat překonané tagy nebo chcete-li používat ve svých dokumentech některé zavržené, ale sémantické elementy.
- Frameset vám umožňuje používat zastaralé značky jako XHTML 1.0 Transitional a přidává podporu pro rámce.

Novější specifikace XHTML 1.1 vychází ze starší specifikace XHTML 1.0 Strict. A přestože jsou nové specifikace vyvíjeny nebo již byli i vydány, jsou specifikace XHTML 1.0 a XHTML 1.1 stále hojně používány.

[3], [4] a [7]

5.2 PHP

PHP je skriptovací programovací jazyk, určený především pro programování dynamických internetových stránek. Zjednodušeně můžeme říci, že PHP je programovací jazyk vsuvek, které se dají vkládat do obyčejných HTML souborů. Jazyk je nezávislý na platformě, skripty fungují bez větších úprav na mnoha různých operačních systémech. PHP skripty jsou většinou prováděny na straně serveru, tedy při požadavku na php stránku server prochází soubor a skripty programově vyhodnocuje, takže klientovi odesílá už čisté HTML.

Jazyk PHP podporuje mnoho knihoven pro různé účely - např. zpracování textu, grafiky, práci se soubory, přístup k většině databázových systémů (mj. MySQL, ODBC, Oracle, PostgreSQL, MSSQL), podporuje celé řady internetových protokolů (HTTP, SMTP, SNMP, FTP, IMAP, POP3, LDAP...).

[4], [7]

5.2.1 JpGraph

Knihovna JpGraph, od Švédské firmy Aditus, slouží k vykreslování grafů. Je to objektově orientovaná knihovna pro PHP, která není jeho základní součástí a je kompletně psaná v jazyce PHP. Tato knihovna je volně použitelná pro nekomerční činnosti. Pro komerční účely slouží profesionální verze, kterou si můžete zakoupit na internetových stránkách firmy Aditus.

Samotná knihovna nabízí četné množství různých typů grafů, které můžeme buď jen zobrazit nebo i uložit do souboru. Použití knihovny je velice jednoduché a nevyžaduje dokonalou znalost PHP. Práci s knihovnou usnadňuje dokumentace s velkým počtem praktických příkladů. Pro představu uvedu základní typy grafů, které knihovna umožňuje generovat: bodové, sloupcové, výsečové (2D i 3D), spojnicové, prstencové, plošné a další.

[18]

5.3 JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, který se používá v internetových stránkách. Vkládá se přímo do HTML stránek a nejčastěji slouží k ovládání různých interaktivních prvků GUI (tlačítka, textová políčka) nebo k vytvoření různých efektů obrázků. JavaScript na rozdíl od PHP probíhá na straně klienta, to znamená, že se program odesílá se stránkou ke klientovi (do prohlížeče) a teprve tam je vykonáván. Jeho syntaxe patří do rodiny jazyků C/C++/Java.

[4], [7]

II. PRAKTICKÁ ČÁST

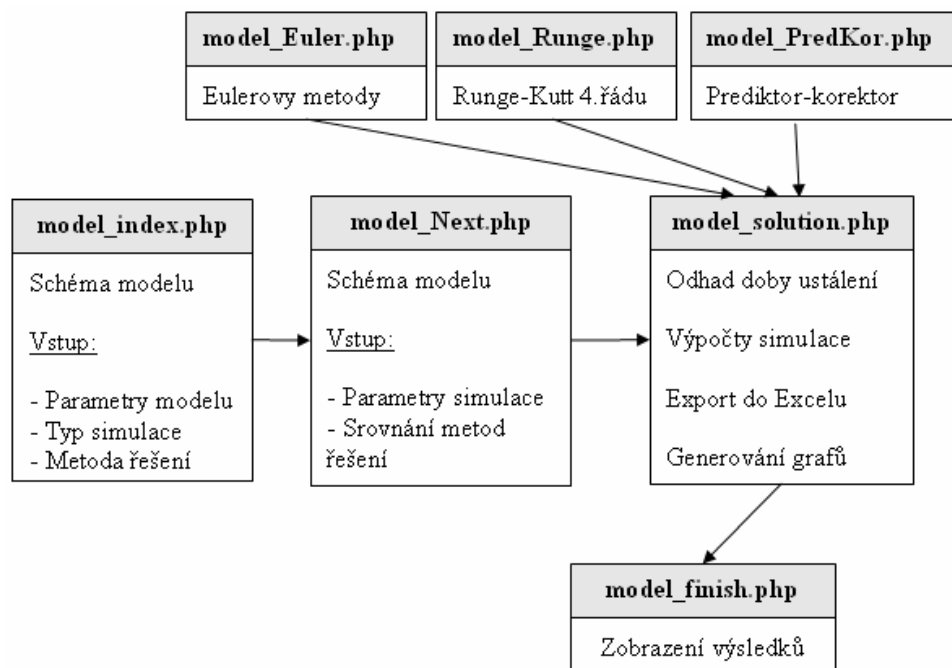
6 UŽIVATELSKÝ POPIS ONLINE SIMULAČNÍHO MODELU

6.1 Online simulační model

Online simulační systém byl vytvořen v jazyce PHP, formou dynamických webových stránek. Pro každý model je zde připravena teorie a simulace. V teorii nalezneme popis jednotlivých modelů, zejména odvození rovnic popisující jejich dynamiku. Simulační část nabízí řešení praktických příkladů na daném modelu. Systém obsahuje tyto modely:

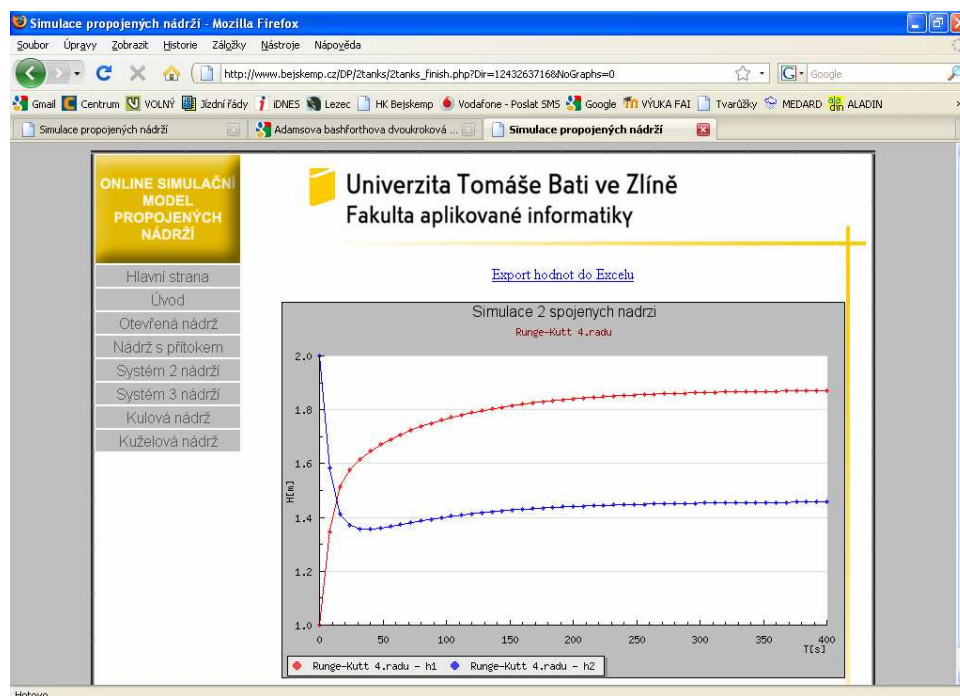
- Nádrž s volným výtokem
- Nádrž s volným výtokem a přítokem
- Systém dvou propojených nádrží
- Systém tří propojených nádrží
- Kulovou nádrž s volným výtokem a přítokem
- Kuželovou nádrž s volným výtokem a přítokem

Uživatel nejprve zadá parametry modelu, metodu numerického řešení a typ simulace. Další postup se liší podle typu simulace. Může probíhat odhad doby ustálení nebo pouze zpřesnění parametrů simulace. Po výpočtu se výsledky zobrazí v grafech a uloží do datového souboru. Struktura systému pro jednotlivé modely je znázorněna na obrázku 6.1.



Obrázek 6.1: Struktura systému pro jednotlivé modely

Na dalším obrázku můžeme vidět vzhled a rozvržení webových stránek.



Obrázek 6.2: Vzhled online systému – zobrazení výsledku průběhu simulace

6.2 Umístění online simulačního systému

V době dokončení se systém nachází na adrese: <http://www.bejskemp.cz/DP/index.html>

6.3 Zadávání parametrů daného systému

Správná funkce systému, stejně jako u většiny programů, je závislá na vstupních datech. Můžeme s trochou nadsázky říci, že „jaké hodnoty zadáme, takové dostaneme“. Systém je formou webových stránek a každý webový prohlížeč obsahuje volbu zpět. To nám zaručuje možnost opakovaného šetření průběhů pro různá nastavení v jednoduchém prostředí, které zvládne obsluhovat většina počítačově gramotných uživatelů.

Systém disponuje velkou variabilitou při zadávání údajů, avšak tato vlastnost také umožňuje zadávat nesmyslné údaje např. parametry v několika-násobně jiných řádech. V nápovědě je zdůrazněn formát zadávání desetinného čísla a důležitost fyzikální reálnosti zadaného modelu. Fyzikální reálnost není povinná, ovšem uživatel musí počítat s možností numerické nestability systému. Programová omezení při zadávání dat se vztahují pouze na základní kontrolu vstupních dat (nepovolené znaky, záporné hodnoty atd.). Na obrázku 6.3 je vstupní formulář modelu kuželové nádrže.

Zadej hodnoty: ?

Q1[m3/s]....Přítok do nádoby :

h1[m]....Počáteční výška hladiny v nádobě :
volte nemulovou počáteční hodnotu

α [°]....Úhel kuželové nádoby :
volte v intervalu <30-150>

Sa1[m2]....Průřez výtokového otvoru z nádoby :

Metoda numerického výpočtu:

Zvol typ simulace: ?

Celkový čas simulace i krok výpočtu si zvolím sám

Automatický odhad ustálení a zvolení kroku výpočtu

Automatický výpočet

Obrázek 6.3: Kuželová nádrž – úvodní formulář se vstupními poli

Ve vstupních polích jsou přednastaveny hodnoty parametrů funkčních modelů. Na obrázku 6.3 také vidíme zvolenou metodu výpočtu a zvolený typ simulace. Od zvoleného typu simulace se odvíjí další nastavení. Na obrázku 6.4 je zřejmé, že je nastaven typ simulace, kdy nastavení celkové doby simulace a hodnotu kroku volí uživatel.

Zvolená metoda : Runge Kutt 4. řádu

Zvolte čas simulace :

Zvolte velikost kroku : ?

Porovnat s Eulerovou metodou

Klasická Eulerova metoda

1.modifikace Eulerovy metody

2.modifikace Eulerovy metody

Porovnat s metodou Prediktor-Korektor

Prediktor :

$y_{i+1} = y_i + hf(x_i, y_i)$

$y_{n+1} = y_n + hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hf(x_n, y_n))$

$y_{n+1} = y_n + \frac{1}{12}h(23f_n - 16f_{n-1} + 5f_{n-2})$

Korektor :

$y_{n+1} = y_n + \frac{1}{2}h(f_{n+1} + f_n)$

$y_{n+1} = y_n + \frac{1}{12}h(5f_{n+1} + 8f_n - f_{n-1})$

$y_{n+1} = \frac{4}{3}y_n - \frac{1}{3}y_{n-1} + \frac{2}{3}hf_{n+1}$

$y_{n+1} = \frac{18}{11}y_n - \frac{9}{11}y_{n-1} + \frac{2}{11}y_{n-2} + \frac{6}{11}hf_{n+1}$

Obrázek 6.4: Kuželová nádrž – upřesnění parametrů simulace

6.4 Typy simulace

Uživatel může volit ze tří typů simulace.

- Celkový čas simulace a kroku uživatel volí sám - zde může uživatel volit jak krok tak i celkový čas simulace. Přesnost výpočtu tak záleží pouze na uživateli, podle způsobu jakým zadá parametry simulace.
- Automatický odhad ustálení a volba kroku - uživateli se zobrazí odhadovaná doba ustálení a podle této hodnoty má možnost zvolit krok simulace. Je mu automaticky nabídnut krok = Doba ustálení/80.
- Automatický režim - uživatel nemůže měnit krok ani celkový čas simulace. Velikost kroku je volena od 0,5 do 3.

Správná funkce jednotlivých typů záleží na správnosti zadání parametrů modelu. Odtud plynou některá omezení.

6.4.1 Automatický odhad doby ustálení – omezení

Zde si ukážeme na konkrétních případech zjištěné důvody, proč nemůžeme v některých situacích odhadnout dobu ustálení.

Příklad 1. Model se neustálí v reálném čase.

Uživatel zvolí nepřiměřeně velký přítok do nádrže. Voda nestačí odtékat a hladina se neustále zvětšuje. Mějme nádrž s přítokem a volným odtokem. Přítok $Q_1 = 200 \text{ m}^3/\text{s}$, průřez nádrže $S = 1 \text{ m}^2$, průřez výtokového otvoru $S_a = 0,0001 \text{ m}^2$. Výška nádrží není omezena, tudíž hladina neustále stoupá. Problém je zobrazen na obrázku 6.5.



Obrázek 6.5: Odhad doby ustálení – model se neustálí v reálném čase

Příklad 2. Hladina po ustálení kmitá

Při určitém nastavení parametrů může nastat stav, který je zobrazen na obrázku 6.6. Z obrázku je zřejmé, že odhad doby ustálení lze provést, pokud velikost rozkmitu není příliš velká.



Obrázek 6.6: Odhad doby ustálení – ustálená hladina kmitá

6.5 Numerické metody řešení obyčejných diferenciálních rovnic

Při zadávání parametrů simulačního pokusu volí uživatel hlavní metodu pro řešení. V další fázi uživatel vybírá metody vedlejší, tzn. že tyto metody jsou v grafech srovnávány s metodou hlavní. V datovém souboru zabírá každá metoda svůj list. Systém umožňuje použití metod:

- Eulerova metoda
 - Klasická Eulerova metoda (4.7)
 - 1. modifikace Eulerovy metody (4.8)
 - 2. modifikace Eulerovy metody (4.9)
- Runge-Kutt čtvrtého řádu (4.10)
- Prediktor-korektor
 - Prediktor 1: Eulerova metoda (4.7)
 - Prediktor 2: 1. modifikace Eulerovy metody (4.8)
 - Prediktor 3: Tříkrokový Adams-Bashforthův vzorec třetího řádu (4.12)
 - Korektor 1: Jednokrokový Adams-Moultonův vzorec druhého řádu (4.14)
 - Korektor 2: Dvoukrokový Adams-Moultonův vzorec třetího řádu (4.15)
 - Korektor 3: Dvoukrokový vzorec metod BDF (4.16)
 - Korektor 4: Tříkrokový vzorec metod BDF (4.17)

Na obrázku 6.7 je znázorněna možnost volby srovnání s hlavní metodou prediktor-korektor.

S volbou metody souvisí také přesnost výsledného průběhu simulačního pokusu. Tu nejvíce ovlivňuje velikost kroku. Uživatel musí brát v úvahu, že při kombinaci několika metod je nastavený krok pro všechny zvolené metody stejný. A to při skutečnosti, že Eulerova metoda vykazuje větší přesnost při nižším kroku, vede k myšlence volby menších kroků. Dalším faktorem ovlivňující přesnost výsledku je výběr vztahů pro prediktor a korektor. Je doporučeno dvojice prediktor a korektor volit tak, aby prediktor byl o jeden řád menší než korektor.

Zvolte typ metody :

Prediktor :

$y_{i+1} = y_i + hf(x_i, y_i)$

$y_{n+1} = y_n + hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hf(x_n, y_n))$

$y_{n+1} = y_n + \frac{1}{12}h(23f_n - 16f_{n-1} + 5f_{n-2})$

Korektor :

$y_{n+1} = y_n + \frac{1}{2}h(f_{n+1} + f_n)$

$y_{n+1} = y_n + \frac{1}{12}h(5f_{n+1} + 8f_n - f_{n-1})$

$y_{n+1} = \frac{4}{3}y_n - \frac{1}{3}y_{n-1} + \frac{2}{3}hf_{n+1}$

$y_{n+1} = \frac{18}{11}y_n - \frac{9}{11}y_{n-1} + \frac{2}{11}y_{n-2} + \frac{6}{11}hf_{n+1}$

Porovnat s metodou Runge Kutta

Porovnat s Eulerovou metodou

Klasická Eulerova metoda

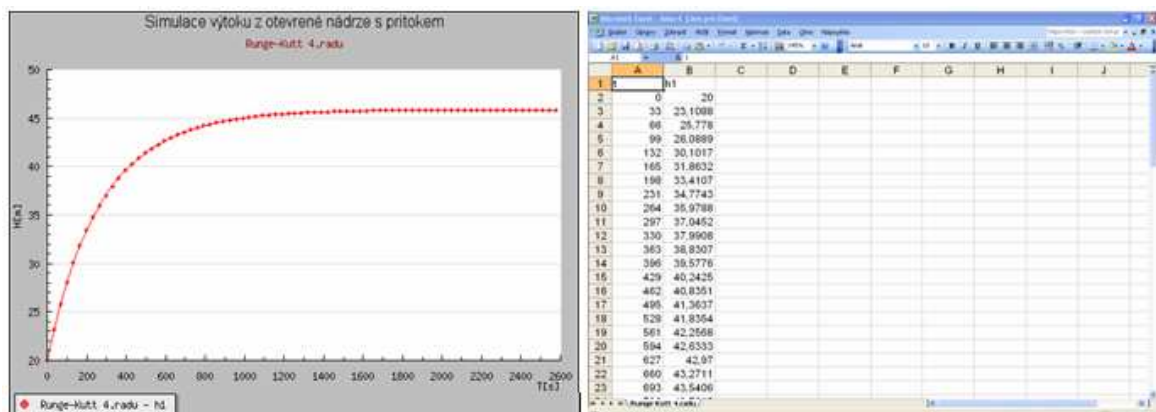
1.modifikace Eulerovy metody

2.modifikace Eulerovy metody

Obrázek 6.7: Nastavení metody numerického řešení

6.6 Zobrazení výsledků simulačních pokusů

Výsledky simulačního pokusu se automaticky zobrazí formou grafu. Takto získáme rychlý přehled o průběhu simulace. Je zde také možnost uložit výsledné hodnoty do souboru MS Excel. Tabulkový editor nám poskytuje další možnosti úpravy či dalších výpočtů, s velkou přehledností i při větším množství dat. Obrázek 6.8 nám demonstruje obě metody zobrazení výsledků.



Obrázek 6.8: Zobrazení výsledků simulace – graf a MS Excel

7 PRINCIPIÁLNÍ POPIS ONLINE SIMULAČNÍHO MODELU

V této kapitole naleznete principy důležitých částí systému, jejich vysvětlení nebo ukázky zdrojových kódů.

7.1 Odhad doby ustálení

System umožňuje zvolit takový typ simulace, který provede odhad doby ustálení výšky jedné nebo více hladin podle daného modelu. U této možnosti jsou jisté omezení, které byly vysvětleny v kapitole 6. Odhad používá metodu Runge-Kutta čtvrtého řádu a spočívá ve výpočtu odchylky několika posledních hodnot od hodnot ustálené výšky hladin. Výpočet si ukážeme na příkladu systému dvou nádrží. Výšky ustálených hladin vyšetříme vztahy (3.37) a (3.38). A protože průběžně kontrolujeme odchylku posledních deseti hodnot, vynásobíme ustálené výšky číslem deset a zaokrouhlíme. Kontrola odchylky začíná tedy až máme dostatek hodnot ke kontrole. Kvůli možnosti zacyklení je odhad omezen podmínkou maximální doby simulace, při aktuálním času $t > 7000$ je hledání ukončeno. Ukázka zdrojového kódu odhadu doby ustálení:

```
// Výpočet doby ustálení a zaokrouhlení 10 hodnot
$h2ust = (((($Q2+$Q1)*($Q2+$Q1))/(($Sa2*$Sa2)*2*$g))+$H;
$h1ust = $h2ust + (($Q1*$Q1)/(($Sa1*$Sa1)*2*$g));
$h1ust = 10 * round($h1ust,4);
$h2ust = 10 * round($h2ust,4);
$i = 1;

// Příznak pro ukončení hledání
$SolveEnd = false;

// Cyklus výpočtu odhadu doby ustálení
while ($SolveEnd == false)
{
    // Celkový čas
    $Time = $Time + $step;

    // Runge-Kutt 4.řádu
    $klh1 = . . .
    $klh2 = . . .
    . . .
    $h1 = . . .
    $h2 = . . .

    // Podmínka pro odhadnutí doby ustálení
    if($i > 12)
    {
        $pom1 = 0;
        $pom2 = 0;

        // Součet posledních 10 hodnot hladin h1 a h2
        for($j = $i - 9; $j <= $i; $j++)
        {
            $pom1 = $pom1 + round($h1[$j],4);
```

```
        $pom2 = $pom2 + round($h2[$j],4);
    }

    // Kontrola odchylky od ustálených stavů
    if (abs($hlust - $pom1) < 0.05 && abs($h2ust - $pom2) < 0.05)
    {
        $SolveEnd = true;
    }
}

$i++;
}
```

Kde proměnné \$Q1, \$Q2, \$H, \$Sa1 a \$Sa2 jsou parametry modelu, \$h1ust a \$h2ust jsou ustálené výšky hladin a \$SolveEnd je příznak pro ukončení hledání. \$pom1 a \$pom2 jsou pomocné proměnné pro kontrolu odchylek. \$h1 a \$h2 jsou výšky hladin v daném kroku.

7.2 Ošetření kritických stavů

7.2.1 Dělení nulou

V teoretické části jsme si odvodily diferenciální rovnice popisující změny výšek hladin u jednotlivých systému. Při výpočtu rovnic popisujících změnu výšky hladiny u modelu kulové nádrže a kuželové nádrže dochází k chybovým situacím. V obou případech rovnice popisující modely obsahují ve jmenovateli výšku h . Samozřejmě nulou dělit nelze a tak nezbyvá než ošetřit kritické stavy jednotlivých modelů. K těm nastává u obou modelů při úplném vytečení kapaliny a výška h je 0. U kulové nádrže, která je navíc i omezena výškou resp. poloměrem nádrže, musíme ošetřit i stav úplného naplnění.

- Při kritickém stavu platí u kuželové nádrže pro výšku hladiny h :

pokud $h \leq 0$, pak $h = 0,01$.

- U kulové nádrže pro výšku hladiny h , kde R je poloměr kulové nádrže:

pokud $h \leq 0$, pak $h = 0,01$.

pokud $h > 2 \cdot R$, pak $h = 2 \cdot R$.

Další případ, kdy nastává problém s dělení nulou, se vyskytuje v metodě prediktor-korektor. K tomuto stavu dochází opět při stavu úplného vytečení kapaliny. Konkrétně k tomu dochází při zpřesňování výsledku korektoru v podmínce, která určuje jeho přesnost.

$$\left| \frac{h_n^k - h_{n-1}^k}{h_n^k} \right| < eps \quad (7.1)$$

Kde eps je malé číslo, např. 0,001. Pokud nastane stav, kdy $h_n^k \leq 0$, pak je tato podmínka přeskočena a výška hladiny se nastaví podle modelu buď na hodnotu nula nebo na hodnotu 0,01.

7.2.2 Úplné vytečení kapaliny – nulová výška hladiny

Vzhledem ke tvaru diferenciálních rovnic popisujících jednotlivé modely, můžeme zjednodušeně říci: přítok + počáteční výška hladiny – výtok = výška hladiny. V případě nulového přítoku odečítáme od hladiny množství vyteklé kapaliny a to i když je nádrž prázdná. Tento stav je ošetřen podmínkou

pokud $h < 0$, pak $h = 0$.

Tato podmínka platí pro všechny nádrže kromě kulové nádrže a kuželové nádrže, kde je nepřipustná nulová hodnota hladin (viz. kapitola 7.2.1).

7.3 Export průběhu simulace do MS Excelu

Exportování simulovaných dat jsem záměrně volil do programu Microsoft Excel. Za prvé, je tento produkt stále nejpoužívanější tabulkový editor. A za druhé, výpis do obyčejného textového souboru nebo přímo do html strany je nepřehledný a neužitečný. Nemůžeme s nimi přímo provádět potřebné výpočty či grafické vizualizace a v případě velkého množství dat se v nich budeme ztrácet. Výsledná data ve formátu .xls jsou tedy rovnou připravena k dalším úpravám a výpočtům. Jsou přehledně a logicky rozříděny do samostatných listů podle zvolených metod numerického řešení obyčejných diferenciálních rovnic.

Při praktické realizaci využívám toho, že MS Excel umožňuje ukládání a načítání souborů uložených ve formátu XML. Strukturu takového souboru si ukážeme na jednoduchém příkladu, který vidíte na obrázku 7.1. Pod obrázkem naleznete zjednodušený zdrojový kód.

	A	B	C	D	E	F	G	H	I	J
1										
2	id	Jméno	Příjmení							
3	1	Jan	Novák							
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										

Obrázek 7.1: Export do MS Excel – konkrétní příklad

Ukázka zdrojového kódu souboru XLS ve formátu XML:

```
<?xml version="1.0"?>
<?mso-application progid="Excel.Sheet"?>
<Workbook xmlns="urn:schemas-microsoft-com:office:spreadsheet"
  xmlns:o="urn:schemas-microsoft-com:office:office"
  xmlns:x="urn:schemas-microsoft-com:office:excel"
  xmlns:ss="urn:schemas-microsoft-com:office:spreadsheet"
  xmlns:html="http://www.w3.org/TR/REC-html40">
<Worksheet ss:name="Sheet1">
<Table ss:ExpandedColumnCount="3" ss:ExpandedRowCount="3"
  x:FullColumns="1" x:FullRows="1">
  <Row>
  <Cell><Data ss:Type="String"> </Data></Cell>
  </Row>
  <Row>
  <Cell ss:StyleID="s21"><Data ss:Type="String">id</Data></Cell>
  <Cell ss:StyleID="s21"><Data ss:Type="String">Jméno</Data></Cell>
  <Cell ss:StyleID="s21"><Data ss:Type="String">Příjmení</Data>
  </Cell>
  </Row>
  <Row>
  <Cell><Data ss:Type="Number">1</Data></Cell>
  <Cell><Data ss:Type="String">Jan</Data></Cell>
  <Cell><Data ss:Type="String">Novák</Data></Cell>
  </Row>
  </Table>
</Worksheet>
</Workbook>
```

Pomocí výše uvedené struktury zapisují simulované hodnoty do souboru. Základní nastavení souboru zůstává, pouze pomocí cyklu zapíše výsledná data. Ukázka části zdrojového kódu pro exportování výsledků do souboru:

```
for($i = 0; $i < count($T);$i++)
{
  $item = "
```

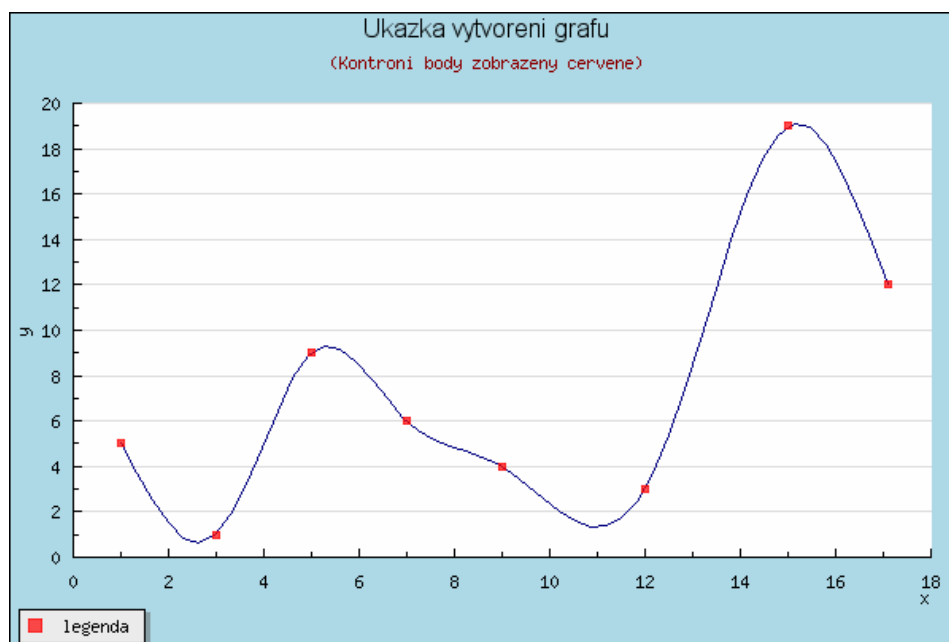
```
<Row>
  <Cell><Data ss:Type="Number">". $T[$i]. "</Data></Cell>
  <Cell><Data ss:Type="Number">".round($h1[$i],5). "</Data></Cell>
  <Cell><Data ss:Type="Number">".round($h2[$i],5). "</Data></Cell>
  <Cell><Data ss:Type="Number">".round($h3[$i],5). "</Data></Cell>
</Row>" ;

fwrite($file, $item);
}
```

Kde pole \$T obsahuje jednotlivé časy průběhu simulace při zvoleném kroku a v polích \$h1, \$h2 a \$h3 jsou uloženy hodnoty výšek hladin v jednotlivých časových okamžicích. Výsledné hodnoty jsou zapisovány do souboru po řádcích.

7.4 Grafický výstup simulace

Kromě číselného výstupu výsledků je k dispozici i grafický výstup. Graficky znázorněná simulace nás rychle informuje o celkovém průběhu, např. zda jsme zvolily správný krok, v jakém čase se hladiny ustálí apod. Pro vizualizaci výsledných dat využívám knihovny JpGraph napsanou v jazyce PHP. Konkrétně bodové grafy s vyhlazenými spojniciemi mezi datovými body se zobrazenými značkami. Grafy jsou uloženy ve formátu PNG (grafický formát určený pro bezztrátovou kompresi rastrové grafiky). Na obrázku je typově stejný graf, který slouží k vizualizaci získaných dat. Pod obrázkem naleznete způsob, jakým vykresluji grafy.



Obrázek 7.2: Grafický výstup získaných dat

Ukázka zdrojového kódu grafu s pomocí knihovny JpGraph:

```
// Potřebné knihovny
include "../src/jpgraph.php";
include "../src/jpgraph_line.php";
include "../src/jpgraph_scatter.php";
include "../src/jpgraph_regstat.php";

// Zdrojová data
$xdata = array(1,3,5,7,9,12,15,17.1);
$ydata = array(5,1,9,6,4,3,19,12);

// Nový Spline objekt (proložení dat spojnicí)
$spline = new Spline($xdata,$ydata);

// Pro hladký průběh spojnice získáme potřebný počet bodů
list($newx,$newy) = $spline->Get(50);

// Vytvoření grafu a nastavení jeho vlastností
$g = new Graph(600,400,"auto");
$g->title->Set("Ukazka vytvoreni grafu");
$g->title->SetFont(FF_ARIAL,FS_NORMAL,12);
$g->subtitle->Set('(Kontroni body zobrazeny cervene)');
$g->subtitle->SetColor('darkred');
$g->SetMarginColor('lightblue');

// Nastavení os xy (typ-lineární, formát zobrazování hodnot, popisky)
$g->SetScale('linlin');
$g->xaxis->SetLabelFormat('%1.0f');
$g->xaxis->SetLabelFormat('%1.0f');
$g->xaxis->title->Set("x");
$g->yaxis->title->Set("y");

// Nový objekt pro vykreslení kontrolních bodů
$splot = new ScatterPlot($ydata,$xdata);

// Nastavení značek (barva a velikost)
$splot->mark->SetFillColor('red@0.3');
$splot->mark->SetColor('red@0.5');

// Nový objekt LinePlot pro vykreslení prokládací křivky
$lplot = new LinePlot($newy,$newx);
$lplot->SetColor('navy');

// Nastavení legendy
$splot->SetLegend ("legenda");
$g->legend->Pos(0.01,0.94,'left','top');
$g->legend->SetLayout(LEGEND_HOR);

// „Vykreslení“ značek a spojnice do grafu a zobrazení
$g->Add($lplot);
$g->Add($splot);
$g->Stroke();
```

Jednotlivé části jsou popsány v poznámkách ve zdrojovém kódu. Řešení grafického výstupu v online simulačním systému se příliš neliší od této ukázky. Pouze ve zdrojových datech a v některých případech v počtu vykreslovaných průběhů.

7.5 Ošetření vstupů

Vstupní pole pro parametry jsou kontrolovány jazykem JavaScript před odesláním formuláře se zadanými hodnotami k pozdějším výpočtům. Funkce pro kontrolu zjišťuje klasické chyby, např. prázdná pole, pole neobsahující číselnou hodnotu. Také kontroluje nepřijatelné hodnoty u některých vstupů, např. zápornou či nulovou hodnotu plochy průřezu nádrže.

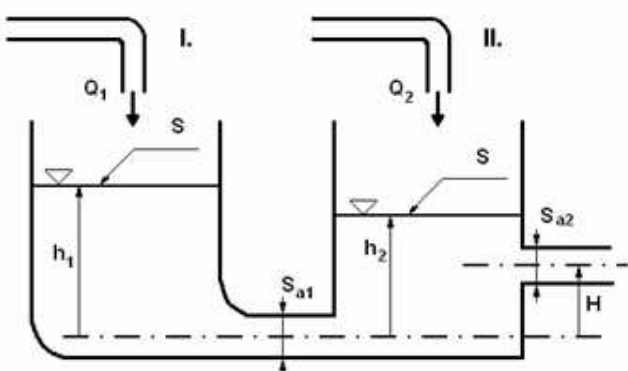
Samozřejmě se nedá ošetřit vše, takže výsledek záleží na datech, které zadá uživatel. V nápovědě jsou stanoveny některá doporučení a rady čeho se vyvarovat.

8 UKÁZKA ŘEŠENÍ KONKRÉTNÍHO PŘÍKLADU

V této části si ukážeme řešení konkrétního příkladu na systému dvou propojených nádrží.

8.1 Zadání systému dvou nádrží

Přítoky do nádrží $Q_1 = 0,02 \text{ m}^3/\text{s}$ a $Q_2 = 0,01 \text{ m}^3/\text{s}$. Plocha průřezu obou nádrží $S = 1 \text{ m}^2$, počáteční výšky hladin $h_1(0) = 1 \text{ m}$ a $h_2(0) = 2 \text{ m}$. Nádrže jsou spojeny potrubím o průřezu $S_{a1} = 0,007 \text{ m}^2$ a výtokové potrubí ve druhé nádrži je ve výšce $H = 1 \text{ m}$ s průřezem $S_{a2} = 0,01 \text{ m}^2$.



Zadej hodnoty: ?

Q_1 [m ³ /s].... Přítok do 1. nádoby :	<input type="text" value="0.02"/>
Q_2 [m ³ /s].... Přítok do 2. nádoby :	<input type="text" value="0.01"/>
h_1 [m].... Počáteční výška hladiny v 1. nádobě :	<input type="text" value="1"/> <small>volte nenulovou počáteční hodnotu</small>
h_2 [m].... Počáteční výška hladiny v 2. nádobě :	<input type="text" value="2"/> <small>volte nenulovou počáteční hodnotu</small>
S [m ²].... Průřez nádob :	<input type="text" value="1"/>
S_{a1} [m ²].... Průřez výtokového otvoru z 1. nádoby :	<input type="text" value="0.007"/>
S_{a2} [m ²].... Průřez výtokového otvoru z 2. nádoby :	<input type="text" value="0.01"/>
H [m].... Výška výtokového otvoru :	<input type="text" value="1"/>
Metoda numerického výpočtu:	<input type="text" value="Eulerova metoda"/>

Obrázek 8.1: Příklad řešení systému dvou nádrží – zadání parametrů systému

8.2 Řešení příkladu

Řešení vychází ze soustavy dvou diferenciálních rovnic (3.29) a (3.30) o dvou neznámých. Jako hlavní metoda řešení byla zvolena Eulerova metoda. A typ simulace s automatickým odhadem doby ustálení. V další části systém umožňuje volbu srovnání hlavní metody s dalšími metodami, které můžeme vidět na obrázku. Doba simulace je určena systémem

resp. jeho odhadem. Zbývá zadat hodnotu kroku. Velikost kroku je nastavena na hodnotu čtyři. Upřesníme hlavní metodu, kde jsem ponechal klasickou Eulerovu metodu a vybral srovnání s metodami Runge-Kutt čtvrtého řádu a prediktor-korektor.

Zvolená metoda : Eulerova

Očekávaný čas ustálení je : 372.5

Zvolte velikost kroku :

Zvolte typ metody :

Klasická Eulerova metoda

1.modifikace Eulerovy metody

2.modifikace Eulerovy metody

Porovnat s metodou Runge Kutta

Porovnat s metodou Prediktor-Korektor

Prediktor :

$y_{i+1} = y_i + hf(x_i, y_i)$

$y_{n+1} = y_n + hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hf(x_n, y_n))$

$y_{n+1} = y_n + \frac{1}{12}h(23f_n - 16f_{n-1} + 5f_{n-2})$

Korektor :

$y_{n+1} = y_n + \frac{1}{2}h(f_{n+1} + f_n)$

$y_{n+1} = y_n + \frac{1}{12}h(5f_{n+1} + 8f_n - f_{n-1})$

$y_{n+1} = \frac{4}{3}y_n - \frac{1}{3}y_{n-1} + \frac{2}{3}hf_{n+1}$

$y_{n+1} = \frac{18}{11}y_n - \frac{9}{11}y_{n-1} + \frac{2}{11}y_{n-2} + \frac{6}{11}hf_{n+1}$

Obrázek 8.2: Příklad řešení systému dvou nádrží - další nastavení simulace

8.2.1 Eulerova metoda

Pro obě rovnice použijeme rekurentního vztahu ve tvaru (4.7). Pro naše hodnoty dosazením získáme:

$$h1_1 = h1_0 + krok \cdot (0,02 - 0,007 \cdot \text{sgn}(h1_0 - h2_0) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|}) = 1,4080$$

$$h2_1 = h2_0 + krok \cdot (0,01 + 0,007 \cdot \text{sgn}(h1_0 - h2_0) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h2_0 - 1|}) = 1,4776$$

$$h1_2 = h1_1 + krok \cdot (0,02 - 0,007 \cdot \text{sgn}(h1_1 - h2_1) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_1 - h2_1|}) = 1,6335$$

$$h2_2 = h2_1 + krok \cdot (0,01 + 0,007 \cdot \text{sgn}(h1_1 - h2_1) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_1 - h2_1|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h2_1 - 1|}) = 1,2473$$

atd.

Výpočet pokračuje dále stejným způsobem. Ukázka zdrojového kódu Eulerovy metody:

```
for($i = 1; $i <= $Time / $step; $i++)
{
    $T[$i] = $T[$i-1] + $step;

    $h1[$i] = $h1[$i-1] + $step * (( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 )
* sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])));

    $h2[$i] = $h2[$i-1] + $step * (( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 )
* sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])) - ( $Sa2
/ $S1 ) * sqrt($outflow*2*$g*( $h2[$i-1]-$H)));

    // Podminka pro osetreni chybovych hodnot - mensich nez 0
    if ($h1[$i]<0) $h1[$i] = 0;
    if ($h2[$i]<0) $h2[$i] = 0;
}
```

Téměř všechny proměnné korespondují s názvy parametrů při zadávání a tak není třeba je popisovat. Proměnná \$outflow nabývá hodnot nula nebo jedna, podle toho zda je výška hladiny ve druhé nádrži vyšší než výtokové potrubí. Pokud ne, proměnná má hodnotu nula, kapalina nevytéká potrubím.

8.2.2 Runge-Kutt čtvrtého řádu

Pro obě rovnice získáme nejdříve odhady derivací a poté výsledné hodnoty. Podle (4.10).

$$k_1 h_1 = (0,02 - 0,007 \cdot \operatorname{sgn}(h_{1_0} - h_{2_0}) \cdot \sqrt{2 \cdot 9,81 \cdot |h_{1_0} - h_{2_0}|}) = 0,051$$

$$k_1 h_2 = (0,01 + 0,007 \cdot \operatorname{sgn}(h_{1_0} - h_{2_0}) \cdot \sqrt{2 \cdot 9,81 \cdot |h_{1_0} - h_{2_0}|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h_{2_0} - 1|}) = -0,0653$$

$$k_2 h_1 = (0,02 - 0,007 \cdot \operatorname{sgn}\left(h_{1_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_1\right) - \left(h_{2_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_2\right)) \cdot \sqrt{2 \cdot 9,81 \cdot \left|h_{1_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_1 - \left(h_{2_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_2\right)\right|} = 0,0427$$

$$k_2 h_2 = (0,01 + 0,007 \cdot \operatorname{sgn}\left(h_{1_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_1\right) - \left(h_{2_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_2\right)) \cdot \sqrt{2 \cdot 9,81 \cdot \left|h_{1_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_1 - \left(h_{2_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_2\right)\right|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot \left|h_{2_0} + \frac{\operatorname{krok}}{2} \cdot k_1 h_2 - 1\right|} = -0,0507$$

$$k_3h1 = (0,02 - 0,007 \cdot \operatorname{sgn}\left(h1_0 + \frac{krok}{2} \cdot k_2h1\right) - \left(h2_0 + \frac{krok}{2} \cdot k_2h2\right)) \cdot \sqrt{2 \cdot 9,81 \cdot \left| \left(h1_0 + \frac{krok}{2} \cdot k_2h1\right) - \left(h2_0 + \frac{krok}{2} \cdot k_2h2\right) \right|} = 0,0445$$

$$k_3h2 = (0,01 + 0,007 \cdot \operatorname{sgn}\left(h1_0 + \frac{krok}{2} \cdot k_2h1\right) - \left(h2_0 + \frac{krok}{2} \cdot k_2h2\right)) \cdot \sqrt{2 \cdot 9,81 \cdot \left| \left(h1_0 + \frac{krok}{2} \cdot k_2h1\right) - \left(h2_0 + \frac{krok}{2} \cdot k_2h2\right) \right|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot \left| \left(h2_0 + \frac{krok}{2} \cdot k_2h2\right) - 1 \right|} = -0,0541$$

$$k_4h1 = (0,02 - 0,007 \cdot \operatorname{sgn}(h1_0 + krok \cdot k_3h1) - (h2_0 + krok \cdot k_3h2)) \cdot \sqrt{2 \cdot 9,81 \cdot \left| (h1_0 + krok \cdot k_3h1) - (h2_0 + krok \cdot k_3h2) \right|} = 0,0342$$

$$k_4h2 = (0,01 + 0,007 \cdot \operatorname{sgn}(h1_0 + krok \cdot k_3h1) - (h2_0 + krok \cdot k_3h2)) \cdot \sqrt{2 \cdot 9,81 \cdot \left| (h1_0 + krok \cdot k_3h1) - (h2_0 + krok \cdot k_3h2) \right|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot \left| (h2_0 + krok \cdot k_3h2) - 1 \right|} = -0,0376$$

Výsledné hodnoty hladin h1 a h2 :

$$h1_1 = h1_0 + krok \cdot \left(\frac{k_1h1}{6} + \frac{k_2h1}{3} + \frac{k_3h1}{3} + \frac{k_4h1}{6} \right) = 1,3462$$

$$h2_1 = h2_0 + krok \cdot \left(\frac{k_1h2}{6} + \frac{k_2h2}{3} + \frac{k_3h2}{3} + \frac{k_4h2}{6} \right) = 1,5832$$

atd.

Stejným způsobem postupujeme pro získání hodnot v dalších krocích. Zdrojový kód metody Runge-Kutta čtvrtého řádu je příliš rozsáhlý a proto je umístěn v příloze.

8.2.3 Metoda prediktor-korektor

Pro explicitní metodu byla zvolena Eulerova metoda (4.7) a jako implicitní metoda vztah (4.14).

Prediktor:

$$h1_1^P = h1_0 + krok \cdot (0,02 - 0,007 \cdot \operatorname{sgn}(h1_0 - h2_0) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|}) = 1,4080$$

$$h2_1^P = h2_0 + krok \cdot (0,01 + 0,007 \cdot \operatorname{sgn}(h1_0 - h2_0) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h2_0 - 1|}) = 1,4776$$

Korektor:

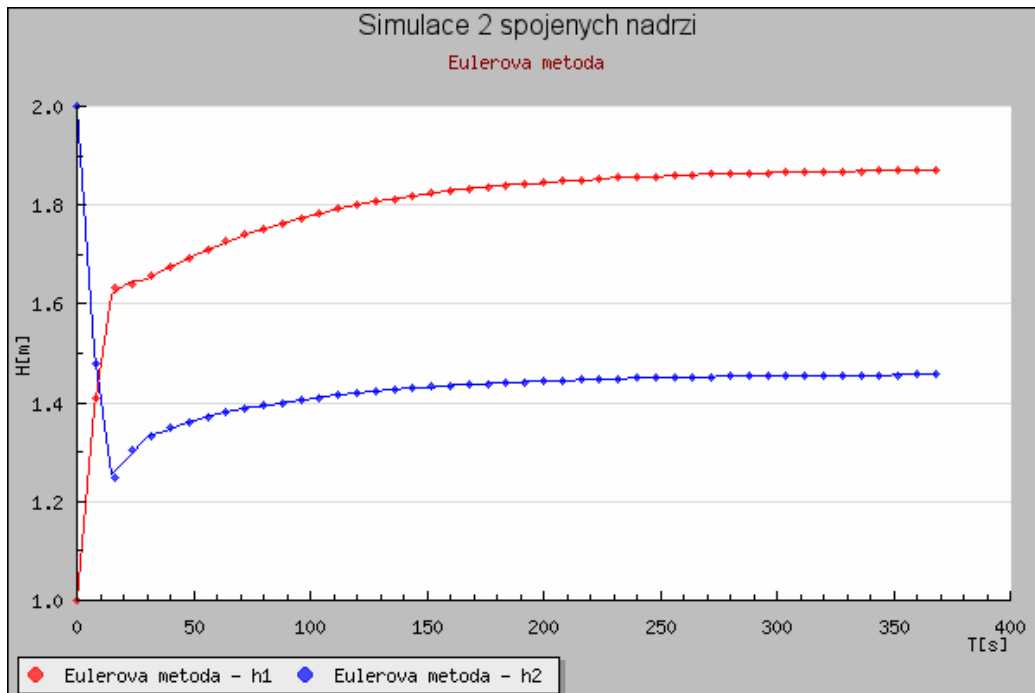
$$h1_1^K = h1_0 + \frac{1}{2}krok \cdot \left((0,02 - 0,007 \cdot \text{sgn}(h1_1^P - h2_1^P)) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_1^P - h2_1^P|} + (0,02 - 0,007 \cdot \text{sgn}(h1_0 - h2_0)) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|} \right) = 1,3167$$

$$h2_1^K = h2_0 + \frac{1}{2}krok \cdot \left((0,01 + 0,007 \cdot \text{sgn}(h1_1^P - h2_1^P)) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_1^P - h2_1^P|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h2_1^P - 1|} + (0,01 + 0,007 \cdot \text{sgn}(h1_0 - h2_0)) \cdot \sqrt{2 \cdot 9,81 \cdot |h1_0 - h2_0|} - 0,01 \cdot \sqrt{2 \cdot 9,81 \cdot |h2_0 - 1|} \right) = 1,6236$$

Následuje upřesnění hodnot hladin $h1$ a $h2$, dokud není splněna podmínka (7.1). Výsledné hodnoty jsou vstupem do prediktoru $h1_2^P$ a $h2_2^P$. Pokračujeme zpřesněním hodnot v korektoru dokud není splněna podmínka přesnosti. Tímto způsobem získáváme hodnoty v dalších krocích. Ukázku zdrojového kódu metody prediktor-korektor naleznete v příloze.

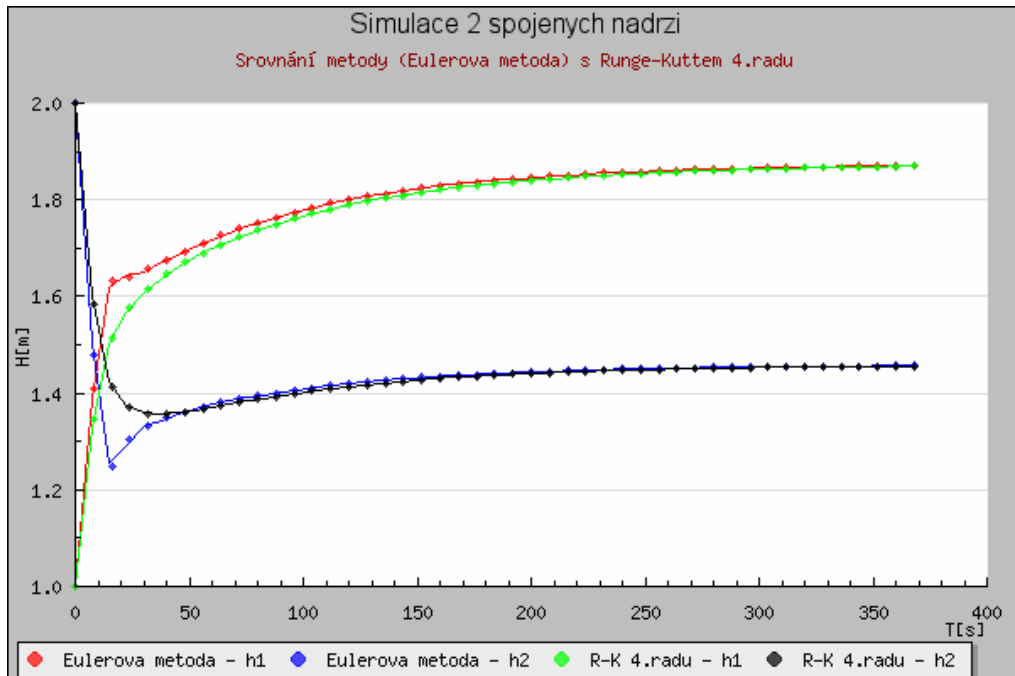
8.3 Zobrazení výsledků simulace

Průběh simulačního pokusu vyšetřeného podle předchozí kapitoly vidíme na obrázcích. Obrázek 8.3 znázorňuje průběh hlavní metody – Eulerovy metody.



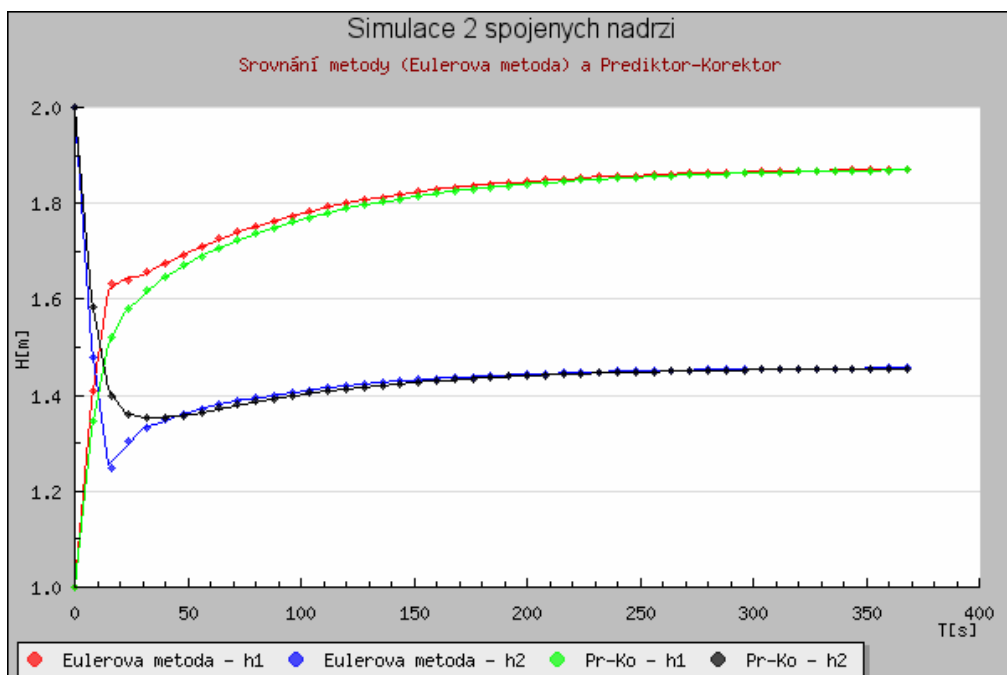
Obrázek 8.3: Příklad řešení systému dvou nádrží – graf průběhu hlavní metody

Na obrázku 8.4 naleznete srovnání Eulerovy metody s metodou Runge-Kutta čtvrtého řádu. Z obrázku je patrná přesnost jednotlivých metod při zadaném kroku. Grafy průběhu jsou k dispozici ihned po výpočtu.



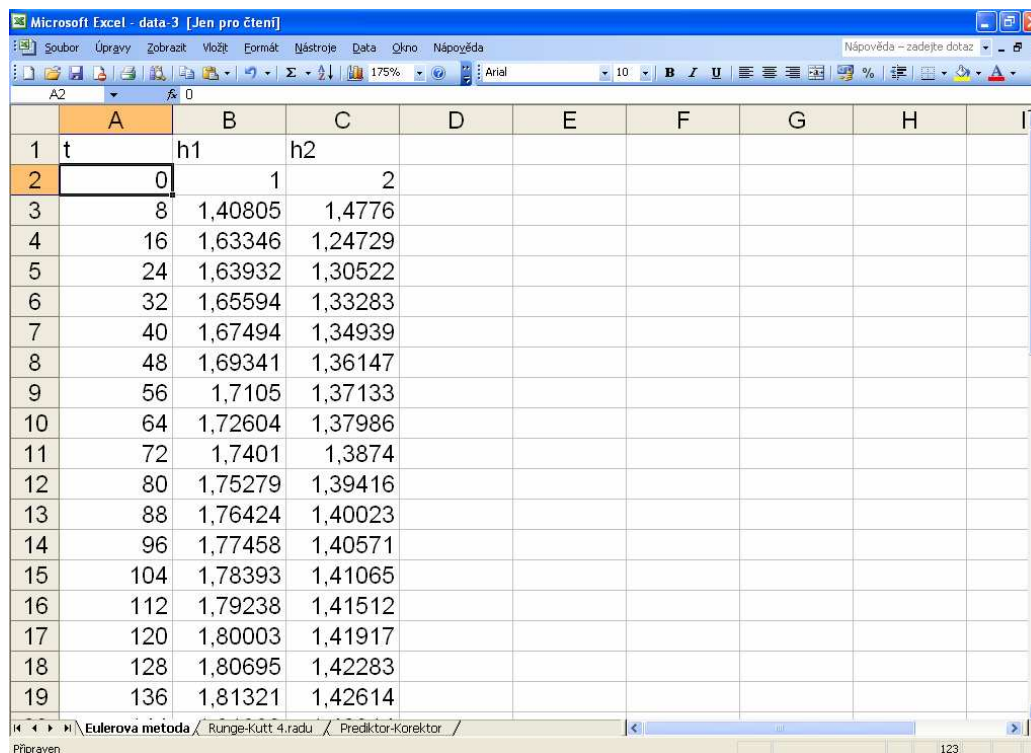
Obrázek 8.4: Příklad řešení systému dvou nádrží – srovnání metod výpočtu

Obrázek 8.5 zobrazuje graf srovnání Eulerovy metody s metodou prediktor-korektor. V tomto příkladu je prediktor zvolen o řád nižší než korektor.



Obrázek 8.5: Příklad řešení systému dvou nádrží – srovnání metod výpočtu

Na posledním obrázku 8.6 je zobrazen datový soubor MS Excel. Hodnoty jednotlivých metod jsou označeny popisky sloupců a metody jsou odděleny listy.



The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I
1	t	h1	h2						
2	0	1	2						
3	8	1,40805	1,4776						
4	16	1,63346	1,24729						
5	24	1,63932	1,30522						
6	32	1,65594	1,33283						
7	40	1,67494	1,34939						
8	48	1,69341	1,36147						
9	56	1,7105	1,37133						
10	64	1,72604	1,37986						
11	72	1,7401	1,3874						
12	80	1,75279	1,39416						
13	88	1,76424	1,40023						
14	96	1,77458	1,40571						
15	104	1,78393	1,41065						
16	112	1,79238	1,41512						
17	120	1,80003	1,41917						
18	128	1,80695	1,42283						
19	136	1,81321	1,42614						

Obrázek 8.6: Příklad řešení systému dvou nádrží – export do Excelu

ZÁVĚR

Cílem této práce bylo vytvoření elektronické podpory výuky předmětu simulace systémů ve formě dynamických www stránek.

V teoretické části je nejprve popsána problematika elektronické podpory výuky. Další část se věnuje přehledu dostupných simulačních systémů k dané problematice. Všechny nalezené aplikace či knihovny byly vytvořeny v programu Matlab. Jediná nalezená online simulace však nebyla funkční. Dále byly odvozeny jednotlivé matematické modely a popsány metody k řešení získaných vztahů. Nakonec teoretické části jsou uvedeny použité technologie k tvorbě systému.

V praktické části jsem vytvořil online systém, který obsahuje ke každému modelu teorii k odvození vztahů popisujících jejich dynamiku. Každý model obsahuje simulační část, kde lze volit parametry modelů, metody řešení a nastavení simulace. Různé tvary propojených nádrží byly ošetřeny zadáváním plochy průřezu nádrže. Nádrž tedy může nabývat libovolného tvaru, za podmínky konstantního průřezu. Pro srovnání chování různých modelů byla přidána kulová a kuželová nádrž, kdy se s hladinou mění i průřez nádrže. Jedná se tedy o ucelenou příručku problematiky nádrží. Výstup simulačních pokusů je formou grafů a datového souboru MS Excel. Systém je vytvořen v jazyce PHP, takže k jeho použití si vystačíme s webovým prohlížečem. Záměrně jsem zvolil tyto technologie pro použití a výstup výsledků, kdy nepotřebujeme žádný specializovaný program např. Matlab. Uživatel má k dispozici nápovědu v průběhu nastavování simulace. Všechna zjištěná omezení a možnosti systému jsou popsány v práci.

CONCLUSION

The aim of this diploma work was set on creation of electronic support of system simulation education.

At the beginning of theoretical part is described e-learning. Next part is focused on existing online systems. All found applications and libraries are made in Matlab. Just one is online system, but it does not work. Next is described deduction of differential equation describing each model and numerical methods for their's solutions. In the end of theoretical part is summary of technologies that I used to create.

In the experimental part, I created online system. System includes theory of each model and simulation part. In simulation part, user can set options of model or simulation and choose methods of numerical solution ODE. By input parameter of model surface we can choose various profiles of tanks. But the cross-section of tank must be constant. For comparing I added ball tank and cone tank. Both tanks have non-constant cross-section. Output of simulation experiment is stored in a file and displayed in graphs. System is created by PHP, for using is enough web browsers. All used technologies are very user-friendly. User can get help during the setting. All found limitation and possibilities are described in work.

SEZNAM POUŽITÉ LITERATURY

- [1] NOSKIEVIČ, Petr. Modelování a identifikace systémů. Ostrava : Montanex a.s., 1999. 276 s. ISBN 80-7225-030-2.
- [2] VICHER, Jaroslav. Numerická matematika. Ústí nad Labem, 2003. 85 s. Skripta. ISBN 80-7044-516-5.
- [3] HLAVENKA, Jiří, et al.. Vytváříme WWW stránky a spravujeme moderní web site. 7.aktualiz. vyd. Brno : CP Books, 2005. 356 s. ISBN 80-251-0801-5.
- [4] JANOVSKEÝ, Dušan. Jak psát web : o tvorbě internetových stránkách [online]. 2004, 02. února 2009 [cit. 2009-02-02]. Dostupný z WWW: <<http://www.jakpsatweb.cz/>>. ISSN 1801-045.
- [5] Technická univerzita v Košiciach. *Simulácia jednoduchých dynamických sústav* [online]. 2008 [cit. 2009-03-25]. Dostupný z WWW: <http://matlab.fei.tuke.sk/gulnServlet/JDS_index.html>.
- [6] REICHL, Jaroslav , VŠETIČKA, Martin . *Encyklopedie fyziky* [online]. 2006-2009 [cit. 2009-03-25]. Dostupný z WWW: <<http://fyzika.jreichl.com/index.php?sekce=browse&page=1>>.
- [7] *Wikipedie : Otevřená encyklopedie* [online]. 2002-2009 [cit. 2009-03-25]. Dostupný z WWW: <<http://cs.wikipedia.org>>.
- [8] VYBÍRAL, Bohumil. *Mechanika ideálních kapalin : Studijní text pro řešitele FO a ostatní zájemce o fyziku* [online]. 2004-2009 [cit. 2009-03-25]. Dostupný z WWW: <<http://fo.cuni.cz/texty/kapaliny.pdf>>.
- [9] JEŽEK, Jan, VÁRADIOVÁ, Blanka, ADAMEC, Josef. *Mechanika tekutin*. Praha : ČVUT vydavatelství, 2000. 150 s. Skripta.
- [10] KUČERA, Radek. *Numerické metody : Texty z numerické matematiky*. [s.l.] : [s.n.], [2007]. 152 s. Dostupný z WWW: <http://home1.vsb.cz/~kuc14/teach_NM.html>. ISBN 80-248-1198-7.
- [11] HANUŠ, Milan. *Řešení omezeného problému tří těles prostředky MATLABu*. [s.l.], 2007. 19 s. Semestrální práce.

- [12] *E-Learning* [online]. 2000 [cit. 2009-04-17]. Dostupný z WWW: <<http://lide.uhk.cz/fim/student/fssoukk1/Index.html>>.
- [13] EDO project team. *Tvorba e-learningových distančních opor (EDO)* [online]. [cit. 2009-04-17]. Dostupný z WWW: <<http://edo.upol.cz/documents.php?tid=elearning>>.
- [14] *Smectra.net* [online]. [cit. 2009-04-17]. Dostupný z WWW: <http://www.smectra.net/nejcastejsi-otazky/definice-elearningu/view?set_language=cs>.
- [15] PRŮCHA, Jan, WALTEROVÁ, Eliška, MAREŠ, Jiří. *Pedagogický slovník*. [s.l.] : [s.n.], 2003. 322 s. ISBN 80-7178-772-8.
- [16] URBAN, Jiří. *Informační server o e-learningu* [online]. 2007 [cit. 2009-04-17]. Dostupný z WWW: <<http://urbanjo2.ic.cz/index.html>>.
- [17] *Projekt eDoceo* [online]. c2002 [cit. 2009-04-17]. Dostupný z WWW: <<http://www.edoceo.cz/standards.jsp.htm>>.
- [18] Aditus Consulting. *JpGraph* [online]. 2000-2007 [cit. 2009-05-01]. Dostupný z WWW: <<http://www.aditus.nu/jpgraph/about.php>>.
- [19] SPŠS a JŠ Kolín. *Hydromechanika* [online]. [2007-2008] [cit. 2009-03-25]. Dostupný z WWW: <http://www.sps-ko.cz/documents/MEC_kratochvil/HYDROMECHANIKA_INTERNET/>.
- [20] Simulační model propojených nádrží [online]. 2005 [cit. 2009-04-21]. Dostupný z WWW: <<http://www.vyuka.fai.utb.cz>>.
- [21] PIŠAN, Radim. *Knihovna modelů technologických procesů*. [s.l.], 2008. 74 s. Diplomová práce.
- [22] TOMÁŠEK, Petr. *Vytvoření skriptů pro webové rozhraní předmětu Analýza a simulace technologických procesů*. [s.l.], 2007. 71 s. Bakalářská práce.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ODR	Obyčejné Diferenciální Rovnice
PHP	PHP Hypertext Preprocessor
XML	eXtensible Markup Language
WWW	World Wide Web
Q_i	Přítok, výtok
S	Plocha průřezu nádrže
S_{ai}	Plocha průřezu výtokového nebo spojovacího potrubí
H	Výška výtokového potrubí v nádrži
h_i	Výška hladiny
F	nelineární funkce podle Föllingerova vztahu
g	Gravitační konstanta
v	Rychlost
m	Hmotnost
p	Tlak
h_{ust}	Ustálená výška hladiny
.xls	Přípona souboru MS Excel
t	Čas

SEZNAM OBRÁZKŮ

Obrázek 3.1.: Výtok z nádoby – odvození Torricelliho vztahu	20
Obrázek 3.2.: Volný výtok z otevřené nádrže.....	21
Obrázek 3.3.: Výtok z otevřené nádrže s přítokem.....	22
Obrázek 3.4.: Systém dvou nádrží s přítokem a výtokem	24
Obrázek 3.5.:Systém tří nádrží s přítokem a výtokem.....	26
Obrázek 3.6.: Výtok z kulové nádrže s přítokem.....	28
Obrázek 3.7.:Výtok z kuželové nádrže s přítokem	30
Obrázek 3.8.: Kuželová nádrž – výpočet plochy průřezu	30
Obrázek 6.1: Struktura systému pro jednotlivé modely.....	40
Obrázek 6.2: Vzhled online systému – zobrazení výsledku průběhu simulace	41
Obrázek 6.3: Kuželová nádrž – úvodní formulář se vstupními poli.....	42
Obrázek 6.4: Kuželová nádrž – upřesnění parametrů simulace.....	42
Obrázek 6.5: Odhad doby ustálení – model se neustálí v reálném čase	44
Obrázek 6.6: Odhad doby ustálení – ustálená hladina kmitá.....	44
Obrázek 6.7: Nastavení metody numerického řešení.....	46
Obrázek 6.8: Zobrazení výsledků simulace – graf a MS Excel	46
Obrázek 7.1:Export do MS Excel – konkrétní příklad	50
Obrázek 7.2: Grafický výstup získaných dat	51
Obrázek 8.1: Příklad řešení systému dvou nádrží – zadání parametrů systému	54
Obrázek 8.2: Příklad řešení systému dvou nádrží - další nastavení simulace.....	55
Obrázek 8.3: Příklad řešení systému dvou nádrží – graf průběhu hlavní metody	58
Obrázek 8.4: Příklad řešení systému dvou nádrží – srovnání metod výpočtu	59
Obrázek 8.5: Příklad řešení systému dvou nádrží – srovnání metod výpočtu	59
Obrázek 8.6: Příklad řešení systému dvou nádrží – export do Excelu	60

SEZNAM PŘÍLOH

Příloha P I: Ukázka zdrojového kódu – systém dvou nádrží – metoda Runge-Kutt čtvrtého řádu

Příloha P II: Ukázka zdrojového kódu – systém dvou nádrží – metoda prediktor-korektor

PŘÍLOHA P I: UKÁZKA ZDROJOVÉHO KÓDU – SYSTÉM DVOU NÁDRŽÍ – METODA RUNGE-KUTT ČTVRTÉHO ŘÁDU

```
for($i = 1; $i <= $Time / $step; $i++)
{
    $T[$i] = $T[$i-1] + $step;

    //koeficienty k1 pro hladiny h1 a h2
    $k1h1[$i] = ( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) * sgn($h1[$i-1]-
$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1]));
    $k1h2[$i] = ( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) * sgn($h1[$i-1]-
$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])) - ( $Sa2 / $S1 ) *
sqrt($outflow*2*$g*($h2[$i-1]-$H));

    //koeficienty k2 pro hladiny h1 a h2
    $k2h1[$i] = ( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step/2)*$k1h1[$i])-( $h2[$i-1]+($step/2)*$k1h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step/2)*$k1h1[$i])-( $h2[$i-
1]+($step/2)*$k1h2[$i])));
    $k2h2[$i] = ( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step/2)*$k1h1[$i])-( $h2[$i-1]+($step/2)*$k1h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step/2)*$k1h1[$i])-( $h2[$i-
1]+($step/2)*$k1h2[$i])) - ( $Sa2 / $S1 ) *
sqrt($outflow*2*$g*($h2[$i-1]+($step/2)*$k1h2[$i])-$H));

    //koeficienty k3 pro hladiny h1 a h2
    $k3h1[$i] = ( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step/2)*$k2h1[$i])-( $h2[$i-1]+($step/2)*$k2h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step/2)*$k2h1[$i])-( $h2[$i-
1]+($step/2)*$k2h2[$i])));
    $k3h2[$i] = ( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step/2)*$k2h1[$i])-( $h2[$i-1]+($step/2)*$k2h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step/2)*$k2h1[$i])-( $h2[$i-
1]+($step/2)*$k2h2[$i])) - ( $Sa2 / $S1 ) *
sqrt($outflow*2*$g*($h2[$i-1]+($step/2)*$k2h2[$i])-$H));

    //koeficienty k4 pro hladiny h1 a h2
    $k4h1[$i] = ( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step)*$k3h1[$i])-( $h2[$i-1]+($step)*$k3h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step)*$k3h1[$i])-( $h2[$i-
1]+($step)*$k3h2[$i])));
    $k4h2[$i] = ( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) * sgn(($h1[$i-
1]+($step)*$k3h1[$i])-( $h2[$i-1]+($step)*$k3h2[$i])) *
sqrt(2*$g*abs(($h1[$i-1]+($step)*$k3h1[$i])-( $h2[$i-
1]+($step)*$k3h2[$i])) - ( $Sa2 / $S1 ) * sqrt($outflow*2*$g*($h2[$i-
1]+($step)*$k3h2[$i])-$H));

    //vypoctené hladiny h1 a h2
    $h1[$i] = $h1[$i-1] + $step
*($k1h1[$i]/6+$k2h1[$i]/3+$k3h1[$i]/3+$k4h1[$i]/6);
    $h2[$i] = $h2[$i-1] + $step
*($k1h2[$i]/6+$k2h2[$i]/3+$k3h2[$i]/3+$k4h2[$i]/6);

    // podminka pro osetreni chybovych hodnot - mensich nez 0
    if ($h1[$i]<0) $h1[$i] = 0;
    if ($h2[$i]<0) $h2[$i] = 0;
}
```

PŘÍLOHA P III: UKÁZKA ZDROJOVÉHO KÓDU – SYSTÉM DVOU NÁDRŽÍ – METODA PREDIKTOR-KOREKTOR

```
for($i = $i; $i <= $Time / $step; $i++)
{
    $T[$i] = $T[$i-1] + $step;

    $ph1 = $h1[$i-1] + $step * ((( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) *
    sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])));

    $ph2 = $h2[$i-1] + $step * ((( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) *
    sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])) - ( $Sa2 /
    $S1 ) * sqrt($outflow*2*$g*($h2[$i-1]-$H)));

    $kh1[$j] = $h1[$i-1] + ( $step/2 ) * ((( ( 1/$S1 ) * $Q1 ) - ( $Sa1
    / $S1 ) * sgn($ph1-$ph2) * sqrt(2*$g*abs($ph1-$ph2))) + (( ( 1/$S1 ) *
    $Q1 ) - ( $Sa1 / $S1 ) * sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-
    1]-$h2[$i-1]))));

    $kh2[$j] = $h2[$i-1] + ( $step/2 ) * ((( ( 1/$S1 ) * $Q2 ) + ( $Sa1
    / $S1 ) * sgn($ph1-$ph2) * sqrt(2*$g*abs($ph1-$ph2)) - ( $Sa2 / $S1 ) *
    sqrt($outflow*2*$g*($ph2-$H))) + (( ( 1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) *
    sgn($h1[$i-1]-$h2[$i-1]) * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])) - ( $Sa2 /
    $S1 ) * sqrt($outflow*2*$g*($h2[$i-1]-$H))));

    do
    {
        $j++;

        $kh1[$j] = $h1[$i-1] + ( $step/2 ) * ((( ( 1/$S1 ) * $Q1 ) - ( $Sa1
        / $S1 ) * sgn($kh1[$j-1]-$kh2[$j-1]) * sqrt(2*$g*abs($kh1[$j-1]-$kh2[$j-
        1]))) + (( ( 1/$S1 ) * $Q1 ) - ( $Sa1 / $S1 ) * sgn($h1[$i-1]-$h2[$i-1])
        * sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1]))));

        $kh2[$j] = $h2[$i-1] + ( $step/2 ) * ((( ( 1/$S1 ) * $Q2 ) + ( $Sa1
        / $S1 ) * sgn($kh1[$j-1]-$kh2[$j-1]) * sqrt(2*$g*abs($kh1[$j-1]-$kh2[$j-
        1])) - ( $Sa2 / $S1 ) * sqrt($outflow*2*$g*($kh2[$j-1]-$H))) + (( (
        1/$S1 ) * $Q2 ) + ( $Sa1 / $S1 ) * sgn($h1[$i-1]-$h2[$i-1]) *
        sqrt(2*$g*abs($h1[$i-1]-$h2[$i-1])) - ( $Sa2 / $S1 ) *
        sqrt($outflow*2*$g*($h2[$i-1]-$H))));

        $pom = abs(($kh1[$j]-$kh1[$j-1])/ $kh1[$j]);
        $pom2 = abs(($kh2[$j]-$kh2[$j-1])/ $kh2[$j]);

    }while( $pom > 0.001 && $pom2 > 0.001 && $j<150 );
}
```