

SLOVENSKÁ POĽNOHOSPODÁRSKA UNIVERZITA V NITRE

Fakulta záhradníctva a krajinného inžinierstva

Ústav krajinného inžinierstva

Ing. Beáta Novotná, PhD., Ing. Tatiana Kaletová, PhD.

doc. Ing. Lucia Tátošová, PhD.

Využitie MATLAB v krajinnom inžinierstve

Učebná pomôcka pre študijný odbor Krajinné inžinierstvo



Nitra 2022

Vydala Slovenská poľnohospodárska univerzita v Nitre
vo Vydavateľstve SPU

Autorky: Ing. Beáta Novotná, PhD. (4,46 AH)
Ústav krajinného inžinierstva
FZKI, SPU v Nitre

Ing. Tatiana Kaletová, PhD. (2,16 AH)
Ústav krajinného inžinierstva
FZKI, SPU v Nitre

doc. Ing. Lucia Tátošová, PhD. (0,27 AH)
Ústav krajinného inžinierstva
FZKI, SPU v Nitre

Recenzenti: Dr. h. c. prof. Ing. Dušan Húska, PhD.
FZKI, SPU v Nitre

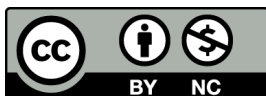
doc. Ing. Ivan Holúbek, PhD.
Ústav hospodárskej politiky a financií
FEM, SPU v Nitre

Táto elektronická publikácia vznikla vďaka finančnej podpore grantovej agentúry KEGA z projektu č. 029SPU-4/2020: Interaktívne vzdelávanie ako nástroj na analýzu a návrhy riešení vo výučbe modelovania vody v krajine.

Skriptá sú určené prioritne pre študentov dennej a externej formy štúdia I. a II. stupňa študijných programov Krajinarstvo Fakulty záhradníctva a krajinného inžinierstva Slovenskej poľnohospodárskej univerzity v Nitre.

Schválila rektorka Slovenskej poľnohospodárskej univerzity v Nitre dňa 15. 12. 2022 ako Využitie MATLAB v krajinnom inžinierstve – online učebnú pomôcku pre študijný odbor Krajinné inžinierstvo pre študentov SPU v Nitre.

Táto publikácia je publikovaná pod licenciou Creative Commons Attribution NonCommercial 4.0 International Public License (CC BY-NC 4.0).
<https://creativecommons.org/licenses/by-nc/4.0/>



ISBN 978-80-552-2564-7

Obsah

Predslov	4
ČASŤ A	4
Úvod.....	4
1. Prístup k MATLABu cez MATLAB Portal	5
2. Úvod do MATLAB	6
2.1 Začíname s MATLABom.....	6
2.2 Ako používať MATLAB.....	8
3. Skripty.....	8
3.1 Vytvorenie nového skriptu	8
4. Premenné a matice	10
4.1 Vytvorenie novej premennej.....	12
5. Základné aritmetické operácie.....	13
6. Začiatok v písaní skriptov	14
7. Galéria živých skriptov	17
8. Grafické zobrazovanie v MATLABe.....	17
8.1 Vytvorenie preddefinovaného 2D grafu	18
8.2 Vytvorenie vlastného 2D grafu.....	18
8.3 Vytvorenie 3D grafu.....	22
8.4 Vytvorenie animovaného grafu	24
8.5 Nastavenie farebného prevedenia grafu	25
8.6 Zobrazenie kontúrového grafu	27
9. Importovanie údajov z programu Excel do MATLAB.....	28
10. Vytvorenie Live Script (živý skript).....	30
11. Otázky k pochopeniu preberaného učiva.....	32
12. Technická podpora k MATLABu	33
13. Ťahák pre MATLAB.....	34
14. Možnosti využitia MATLAB v rámci krajinného inžinierstva.....	37
14.1 Odpadové hospodárstvo.....	37
14.2 Vodné hospodárstvo	40
14.2.1 Prúdenie v otvorených kanáloch.....	40
14.2.2 Využitie a spracovanie údajov zo sonaru	43
14.2.3 Ďalšie alternatívy využitia vo vodnom hospodárstve	44
14.3 Pozemkové úpravy	46
14.4 Iné typy úloh v krajinárstve.....	47
14.4.1 Lavíny.....	49
14.4.2 Zosuvy.....	49

ČASŤ B	51
Problém 1: Určenie minimálneho priemeru potrubia.....	51
Problém 2: Určenie hĺbky rovnomerného prúdenia v otvorenom kanáli	54
Problém 3: Určenie prepadovej výšky pri prepade cez ostrú hranu.....	56
Problém 4: Analýza denných teplôt.....	59
Problém 5: Nástup a ukončenie definovaných teplôt.....	63
Problém 6: Výpočet počtu definovaných períod	70
Problém 7: Hodnotenie suchých a zrážkových períod podľa časových období.....	72
Použitá literatúra.....	82

Predslov

Interaktívne vzdelávanie v prostredí MATLAB® (ďalej iba MATLAB) nás baví a preto chceme, aby ste si ho užili aj Vy - to je tiež motivácia tejto publikácie. Prvým a hlavným cieľom týchto Návodov na cvičenia je, aby ste po ich preštudovaní a pochopení prečítaného textu rýchlo a s ľahkosťou začali pracovať v MATLABe. Vyučovanie v MATLABe zmenilo význam vedeckej výpočtovej techniky. Namiesto pôvodných výpočtov, kde prevažujú pojmy, kompilácie do strojového kódu a ako výstupy sú tabuľky s číslami, teraz očakávate a užívajte si interaktívne výpočty, programovanie, grafiku a animácie. Toto všetko sa v MATLAB nachádza na jednom mieste - pod jednou strechou. MATLAB je totiž jednoduchý, výkonný a na väčšinu účelov pomerne rýchly. Myslíme si, že vzdelávanie pomocou neho sa bude páčiť aj Vám.

ČASŤ A

Úvod

Voda je nevyhnutná pre život na Zemi. To je tiež to, čo robí našu planétu takou výnimočnou. Pri pohľade z vesmíru vyzerá Zem ako jasne modrá vodnatá planéta pokrytá jemnými mrakmi. Je to jediná planéta, o ktorej v súčasnosti vieme, že má na povrchu tekutú vodu. Voda je všade okolo nás. Môže tiecť v toku v blízkosti vášho domova, spadnúť z neba ako dážď alebo sneh, alebo môže tiecť z vášho vodovodného kohútika. Je veľmi dôležité, aby sme sa o nej čo najlepšie rozhodli o zabezpečení dostupnosti vody. Bez vody by na Zemi nebol život. Zachytávanie a presúvanie vody umožnilo ľuďom prežiť a prekvitať (Arizona State University, 2022).

Nedávny pokrok v sieťových technológiách a nástrojoch spôsobuje rôzne transformácie v spôsoboch komunikácie. Zmena klímy, biodiverzita a ľudské zdravie sú neoddeliteľne prepojené a zanedbávanie týchto spojení viedlo k ekologicky deštruktívnym postupom, ktoré nedokázali zabezpečiť zdravú výživu, pričom holistický prístup by mohol pomôcť regenerovať planétu a náš blahobyť (IWMI, 2022).

Účelom tejto publikácie je bližšie spoznať a analyzovať systém a koncept vodnej bilancie. Pochopenie a uplatňovanie tohto konceptu je rozhodujúce pri zavádzaní pokročilejších tém, na dôkladnejšie pochopenie vzťahov medzi zložkami a obmedzeniami vodnej bilancie. Aby sme študentom pomohli pri rozvíjaní pochopenia vodnej bilancie, vypracovali sme túto

publikáciu, nadväzujúcu na on-line webovú aplikáciu, ktorá umožňuje jednorozmerné sledovanie a následnú aplikáciu vybraných meteorologických údajov z konkrétnej meteorologickej stanice, môže byť však aplikovateľná na ktorúkoľvek z meteorologických alebo klimatologických staníc.

V rámci aplikácie môžu študenti, po stiahnutí údajov z webu, narábať s údajmi pôdných teplôt v hĺbke 10 cm, 25 cm a 50 cm; s údajmi o teplote vzduchu, či vzdušnou vlhkosťou. Cieľom aplikácie je, aby študenti lepšie pochopili, ako jednotlivé zložky hydrologickej bilancie na seba nadväzujú, preskúmali scenáre využívania pôdy, extrémny počasie a prejavy klimatickej zmeny.

Na základe predchádzajúcich výskumov ako napr. Gannon & McGuire (2022), Mulvey et al. (2022) a iní, je zrejmé, že študenti sa môžu naučiť koncepty vodnej bilancie interaktívnymi metódami efektívnejšie ako tradičnými metódami, ako sú tabuľkové výpočty.

Podľa Fehovej (2008) najvýraznejším znakom súčasnosti je implementácia informačno-komunikačných technológií do každodenného života ľudí. Tieto zmeny ovplyvňujú nielen sféru súkromnú (využívanie voľného času, komunikáciu) a pracovnú, ale zasahujú aj do edukačného procesu. Jednou z najdôležitejších úloh v oblasti vzdelávania je vypracovanie takých programov a metódik, pri ktorých by sa počítače stali bežnými, normálnymi pracovnými nástrojmi učiteľa a súčasne by rozvíjali tvorivé myslenie študenta. Podľa uvedenej autorky výsledky rozličných diskusií vytvárajú spoľahlivé a prítiahľivé prostredie pre učenie, poskytujú pozitívnu spätnú väzbu, pomáhajú vytvoriť úhladný bezchybný text, rešpektujú individuálne tempo, rýchlosť a zručnosť, dovoľujú vrátiť sa späť k problému a začať alebo ukončiť prácu v rôznych miestach, pomáhajú v učení študentov so špecifickými poruchami učenia a handicapovaným študentom, sprístupňujú bohaté zdroje informácií, zrozumiteľne prezentujú zložité myšlienkové postupy a vzťahy pomocou grafiky, pričom ponúkajú rozvoj myslenia študentov (Kalaš, 2000).

V nasledujúcich častiach publikácie je na príklade riešenia konkrétnych úloh prezentovaná možnosť zefektívnenia výučby predmetov vyučovaných v krajinom inžinierstve pomocou počítačovej podpory programu MATLAB. MATLAB je nástroj tak pre pohodlnú interaktívnu prácu, ako aj pre vývoj širokého spektra aplikácií. Podľa (mathworks.com, 2022) je MATLAB Live Editor vedecké počítačové ihrisko pre študentov, kde si rozvíjajú výpočtovú gramotnosť, získavajú okamžitú spätnú väzbu a viaceré prostriedky reprezentácie, vyjadrovania a zapojenia sa do analýzy skúmaného problému.

Prvá – A časť publikácie predstavuje MATLAB, informuje o jeho základných nástrojoch a možnostiach využitia. Po preštudovaní základných princípov MATLAB je k dispozícii on-line test na overenie získaných vedomostí. Táto časť vyhodnocuje prácu a pochopenie preberaného textu.

Druhá – B časť nadväzuje na prvú časť, pričom používa konkrétny live script (živý skript) pre potreby interaktívneho vzdelávania vo vodnom hospodárstve.

Táto časť umožňuje v interaktívnom grafe z konkrétnej web-stránky a na základe skutočných meraní vyhodnotiť jednotlivé úlohy podľa vopred definovaných alebo voliteľných premenných.

1. Prístup k MATLABu cez MATLAB Portal

Prístup k MATLABu je cez web-stránku: <https://ch.mathworks.com/products/matlab-online.html>, pričom je ho možné využiť vo verzii:

1.) Online (Matlab je možné stiahnuť z web stránky do počítača alebo ho tiež používať v online verzii na: <https://matlab.mathworks.com/>) a

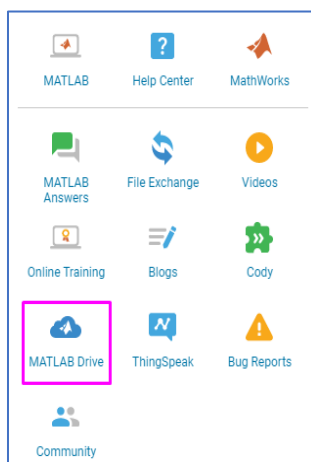
2.) Po nainštalovaní do počítača je potrebné uskutočniť nasledovné kroky (Univerzita obrany, 2022):

1. Zvoľte 'Sign in to get started' (Prihlásiť sa a začať) časti Get MATLAB and Simulink (získať MATLAB and Simulink);
2. Prihláste sa pod svojim univerzitným účtom;
3. Vytvorte si účet u MathWorks. Ak už taký účet máte, prihláste sa naň;
4. Váš účet MathWorks bude automaticky prepojený s licenciou a budete nasmerovaní na stránku pre stiahnutie inštalácie;
5. Kliknite na tlačidlo Download stiahnuť s aktuálnou verziou;
6. Zvoľte jeden z podporovaných operačných systémov a stiahnite inštaláciu.

Cloud

Data v cloude pre on-line verziu sa nachádzajú na <https://matlab.mathworks.com/>

Po prihlásení do Matlab sa zobrazia v Matlab Drive, viď. [Obr. 1.1](#).



Obr. 1.1. MATLAB Drive poskytuje spoločné úložisko založené na cloude pre súbory MATLAB (mathworks.com, 2022).

2. Úvod do MATLAB

MATLAB je unikátne a tiež univerzálne prostredie pre riešenie zložitých matematických, fyzikálnych, či biomedicínskych problémov. Ponúka tiež vývojové prostredie, v ktorom môžeme vytvoriť programy, ktoré budú riešiť daný problém s užívateľom zadanými parametrami. Úprimne povedané, s ním môžete riešiť absolútne čokoľvek. Hlavným kúzlom MATLABu je, že nemusíte riešiť, ako to naprogramujete, ale stačí vám vedieť, čo programujete. Program sám o sebe má nejaké hlavné jadro, ku ktorému sú pripojené ďalšie pluginy či knižnice (zvané Toolboxy), ktoré obsahujú vopred naprogramované funkcie. Tie potom stačí už iba používať. Základné verzie už väčšinou zahŕňajú veľké množstvo týchto Toolboxov. Sústredíte sa teda iba na váš problém a všetko ide hneď ľahšie (ltnetwork.cz, 2022).

2.1 Začíname s MATLABom

MATLAB je v podstate MATicové LABoratórium (Matrix Laboratory). Existuje veľa zdrojov na jeho výučbu a aplikáciu (Learning vibes, 2021). Bol vyvinutý za účelom uskutočnenia technických a vedeckých výpočtov a analýz. MATLAB bol vyvinutý firmou MathWork. MATLAB nie je voľne šíriteľným programom (open source program), napriek tomu sa stále častejšie používa po celom svete.

Hlavnými výhodami MATLABu sú:

- rozsiahla knižnica preddefinovaných funkcií – ak niekto vytvorí funkciu a aplikuje ju, poskytne je MathWorku a následne je doplnená priamo do MathWorku. Táto funkcia je pripravená na využitie pre ďalších užívateľov MATLABu, ktorí ju môžu jednoducho použiť. Nie je potrebné funkciu znovu vytvárať.
- môže byť prepojený s inými rôznymi typmi softvéru.
- skraca čas a náklady na projektovanie a vývoj technických produktov.

Top 10 výhod MATLABu, podľa sk.education-wiki.com (2022) sú:

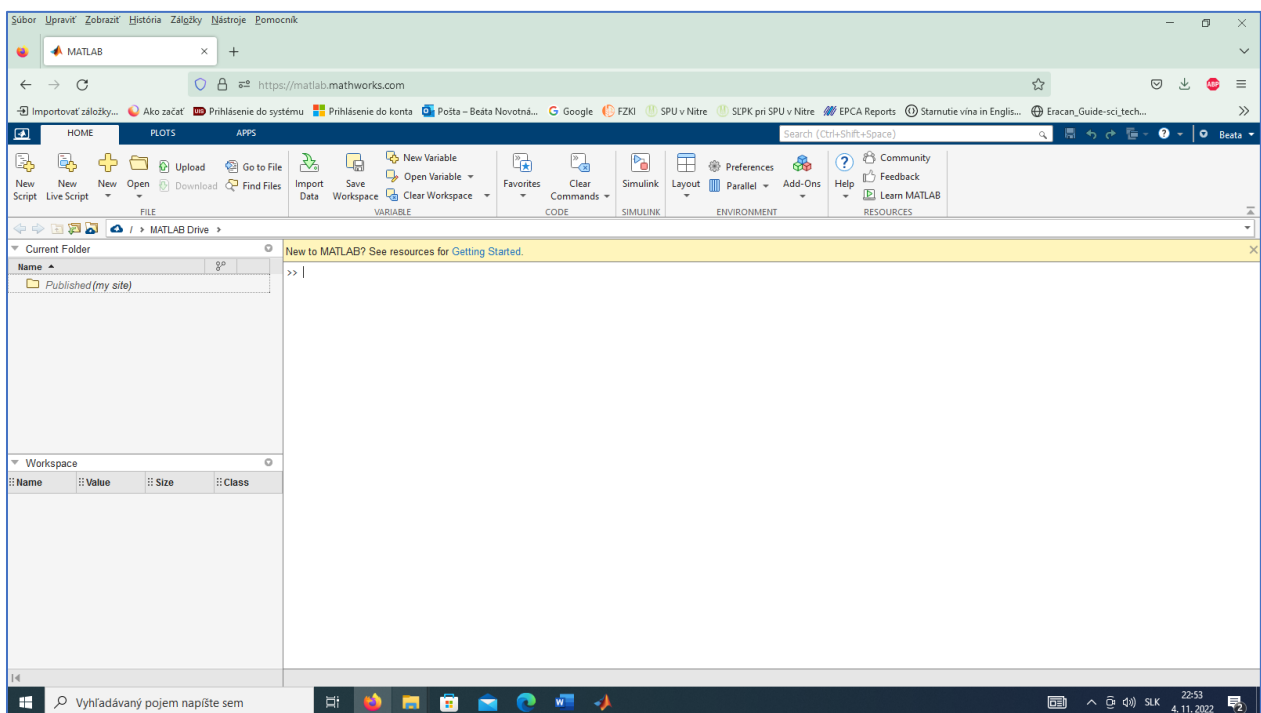
V prostredí MATLAB existuje veľa charakteristík alebo funkcií, vďaka ktorým je lepší ako mnoho iných programovacích jazykov. Niektoré z charakteristík MATLABu sú rôzne preddefinované funkcie, podpora integrácie s inými prostrediami a vizualizačné nástroje pre vlastné grafy a grafiku. Nižšie uvedené body dokazujú, že MATLAB je široko používaný v mnohých oblastiach:

1. Prostredie MATLAB má svoje preddefinované funkcie a knižnice, ktoré pomáhajú programátorom ľahko ho používať. V našom každodennom živote čelíme rôznym zložitým matematickým problémom; to sa dá vykonať v MATLABe s jedinou funkciou alebo kódom. Podobne MATLAB poskytuje celý rad nástrojov, ktoré sa používajú v mnohých oblastiach, ako je letecké inžinierstvo, komunikácia a spracovanie signálu atď.
2. MATLAB podporuje rôzne platformy ako LINUX, rôzne verzie Windows, atď. Takže programy alebo súbory aplikácií napísané na jednej platforme je možné spúšťať alebo čítať na inej platforme, čo zefektívňuje užívateľovi prácu.
3. MATLAB nepotrebuje kompilátor na vykonanie kódu, takže ak napíšeme niekoľko riadkov kódu, dostaneme okamžité výsledky, ktoré nie sú dostupné v iných programovacích jazykoch ako C, C ++. Uľahčuje to interpretáciu a implementáciu kódu v MATLABe. Tento proces je však trochu pomalý, takže sú k dispozícii kompilátory MATLAB, ktoré sa dajú použiť na rýchlejšie spustenie programov.
4. MATLAB má mnoho zabudovaných príkazov na vizualizáciu, ktoré sú potrebné pre vykreslenie a ďalšie aplikácie. Môžete ich prezerat' cez akékoľvek výstupné zariadenie, ktoré bude pripojené k zariadeniu, na ktorom je nainštalovaný MATLAB. Môže sa tiež použiť na ľahké spracovanie obrázkov a videí. Vďaka tomu je MATLAB vynikajúcim vizualizačným nástrojom, ktorý uľahčuje vykreslenie.
5. MATLAB sa používa aj na analýzu údajov. Existujú rôzne zdroje, kde je ho možné integrovať, či už ide o Excel alebo inú databázu. Po importovaní sady údajov do svojho prostredia ponúka rôzne nástroje a knižnice na vykonávanie čistiacich a iných techník, aby boli údaje vhodné na ďalšiu analýzu. Po odstránení nežiaducich informácií je potrebné vytvoriť model v závislosti od zvoleného cieľa. Má svoje nástroje, balíčky a funkcie pre vývoj modelov, ako je logistická regresia, zoskupovanie, náhodná štruktúra domén a podporné vektorové modely. Po vytvorení modelu sú k dispozícii nástroje na vylepšenie modelu, ktoré porovnávajú rôzne modely a poskytujú model, ktorý má lepšiu presnosť a je pripravený na použitie vo výrobe. Od počiatočných metód analýzy až po koniec sú všetky funkcie v MATLABe, čo z neho robí skvelý nástroj na analýzu údajov.
6. Existuje tiež MATLABov kódovač, ktorý pomáha pri konverzii kódu v MATLABe do akéhokoľvek iného jazyka, ako je C ++, JAVA atď., Čo zvyšuje čitateľnosť a pomáha programátorom ľahko prečítať kód.
7. MATLAB poskytuje nástroje a knižnice Deep Learning, ktoré sú obsiahnuté v jeho najnovšej verzii na dosiahnutie požadovaných výsledkov. Zahŕňa túto dôležitú knižnicu, ktorá sa dá použiť v mnohých ďalších oblastiach, ako je riadiaci systém, komunikácia, spracovanie obrazu a signálu.

8. MATLAB má svoje preddefinované knižnice a nástroje, ktoré umožňujú používateľom vytvárať GUI (Graphical User Interface) pre svoje príslušné programy. Je to tiež veľká pomoc pre používateľov, ktorí v MATLABe nemajú predchádzajúce skúsenosti a znalosti.
9. MATLAB Online poskytuje znalosti a používanie platformy bez sťahovania a inštalácie softvéru. Môžete ho použiť pomocou ľubovoľného webového prehľadávača, ak máte prístup na internet.
10. Má dobre napísanú dokumentáciu pre pomoc súvisiacu s akýmikoľvek funkciami alebo knižnicami používanými v jej prostredí. Na trh sú uvádzané rôzne online semináre, dvojročné aktualizácie týkajúce sa rôznych chýb, ktoré používateľom pomáhajú ľahko sa učiť.

2.2 Ako používať MATLAB

V online verzii je prostredie, ktoré je identické s verziou Matlabu po stiahnutí, vid' [Obr. 2.1](#). Skladá sa z a) Command window (príkazové okno), b) Workspace (pracovný priestor) a c) Current folder (aktuálny priečinok).



Obr. 2.1. Uživatelské prostredie MATLAB zložené s a) Command window (príkazové okno), b) Workspace (pracovný priestor) a c) Current folder (aktuálny priečinok).

3. Skripty

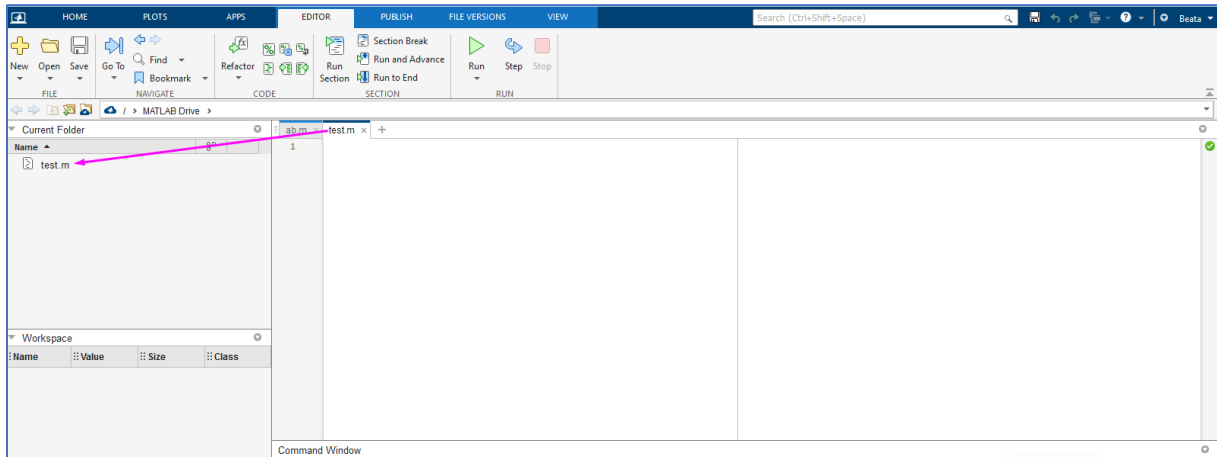
Skripty sú najjednoduchším druhom súboru kódu, pretože nemajú žiadne vstupné ani výstupné argumenty. Sú užitočné na automatizáciu sérií príkazov MATLABu, ako sú výpočty, ktoré je potrebné opakovane vykonávať z príkazového riadku alebo sériu príkazov, na ktoré sa musíme odvolávať (Mathworks.com, 2022).

3.1 Vytvorenie nového skriptu

Nový skript vytvoríme pomocou ikony *new script* (nový), vid' https://prnt.sc/ovlJzwtl_zKB, pričom sa otvorí nové okno, kde vytvoríme požadovaný nový skript, napr.:
 $a=10$

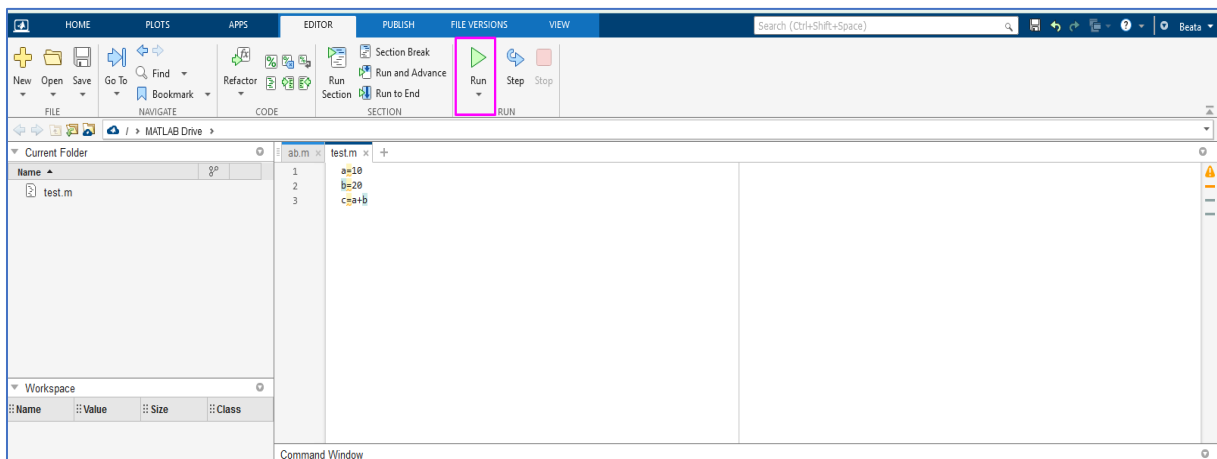
$b=20$
 $c=a+b$

Výsledný skript uložíme tlačidlom *save* (uložiť) na požadované miesto v priečinku na počítači. Zároveň sa tento skript objaví na ľavej strane v Current folder, vid' [Obr. 3.1](#). Príkazový riadok súčasne zobrazuje, kde je tento skript uložený v počítači (jeho koncové označenie je: *.m). Po vytvorení skriptu môžete do skriptu pridať kód a uložiť ho. Tento kód, ktorý generuje náhodné čísla od 0 do 100, môžete uložiť napríklad ako skript s názvom numGenerator.m (Mathworks.com, 2022).



Obr. 3.1. Príklad označenia a uloženia skriptu (*test.m*).

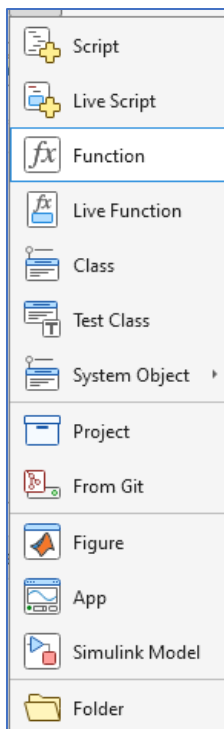
Skript spustíme pomocou ikony *Run*, vid'. [Obr. 3.2](#).



Obr. 3.2. Príklad výpočtu skriptu (spustenia skriptu) aktivovaním ikony *Run*

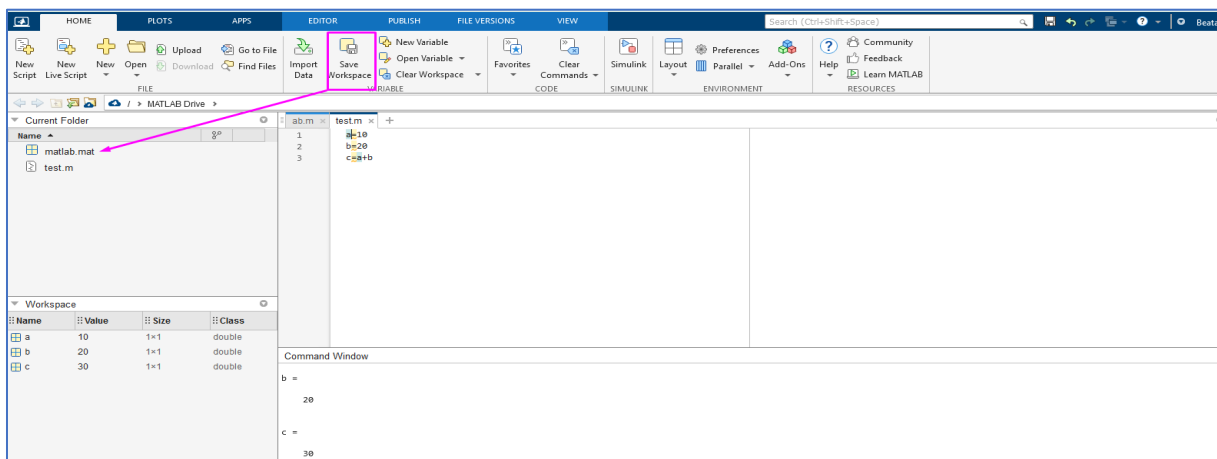
Po spustení bude skript vypočítaný a všetky premenné sa zobrazia v časti Workspace.

Aktivovaním ikony *New*, vid' obr. 4 máme možnosť vybrať si z nasledovných možností, ktoré zobrazuje [Obr. 3.3](#).



Obr. 3.3. Výber možností, ktoré sa zobrazia po aktivovaní ikony *New* (nachádza sa v hlavičke MATLAB).

Aktivovaním ikony *Save workspace* (pracovný priestor) je možnosť vytvoriť personalizovaný Workspace, ktorého názov sa nachádza rovnako v Current folder (koncovka: **.mat*) (Obr. 3.4).



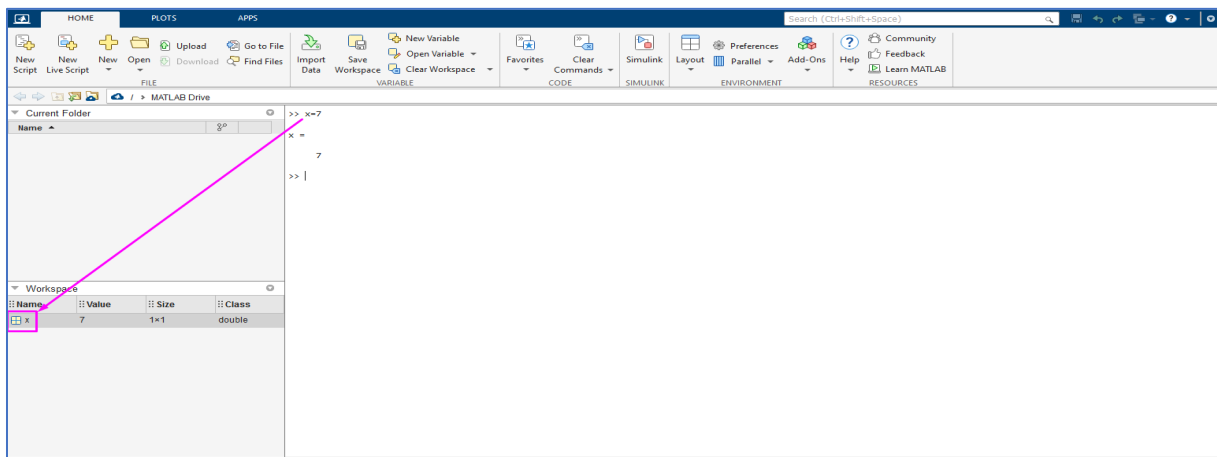
Obr. 3.4. Príklad vytvorenie personalizovaného Workspace (pracovného priestoru) s názvom *matlab.mat*.

4. Premenné a matice

Bližšie informácie, ako vytvoriť maticu s premennými sú dostupné na stránke: <https://ch.mathworks.com/matlabcentral/answers/63202-how-to-create-a-matrix-with-variables-in-it>

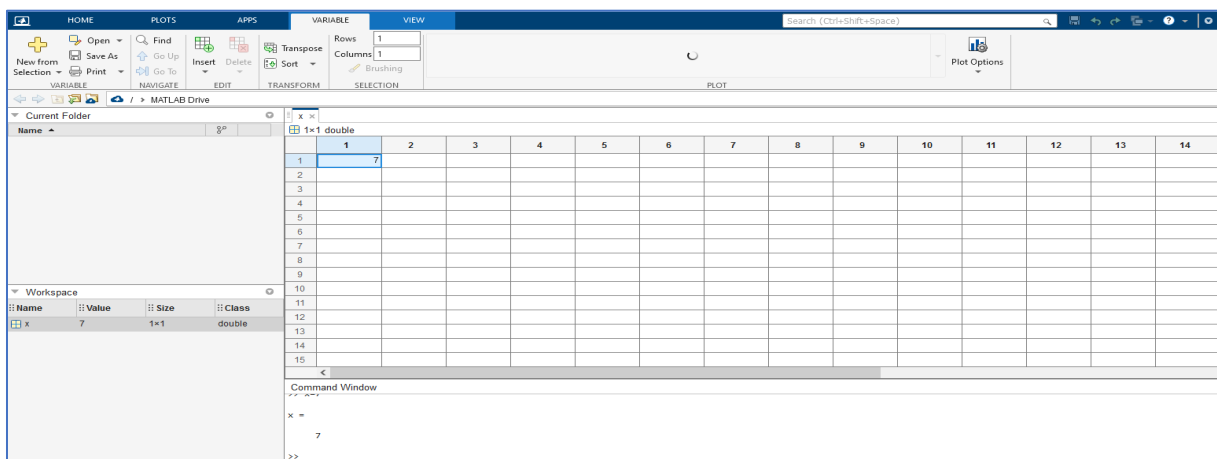
Vo všeobecnosti premenné v MATABe prináležia k určitému typu dát. Najjednoduchším typom dát je matica (matrix), z ktorej je odvodený aj názov MATAB.

Po zapísaní hodnoty do Command window sa táto hodnota rovnako zobrazí vo Workspace (Obr. 4.1).



Obr. 4.1. Príklad zápisu v MATLAB do Command window (príkazový riadok) s následným zobrazením na Workspace (pracovný priestor).

Kliknutím na označenie mriežky vo Workspace je zobrazená premenná vo formáte matice, vid' Obr. 4.2.



Obr. 4.2. Zobrazenie štruktúry zápisu premennej na príklade $x=7$.

Ak použijeme viaceré premenné napr. $y=[1\ 2\ 3]$ zobrazia sa v MATLABe v nasledovnom formáte (Obr. 4.3):

1×3 double					
	1	2	3	4	5
1	1	2	3		
2					
3					
4					
5					
6					
7					
8					

Obr. 4.3. Zobrazenie štruktúry zápisu premenných vo formáte $y=[1\ 2\ 3]$.

Premenné vo formáte $z=[2\ 4\ 6; 8\ 10\ 12]$, sú v MATLABe zapísané nasledovným spôsobom (Obr. 4.4):

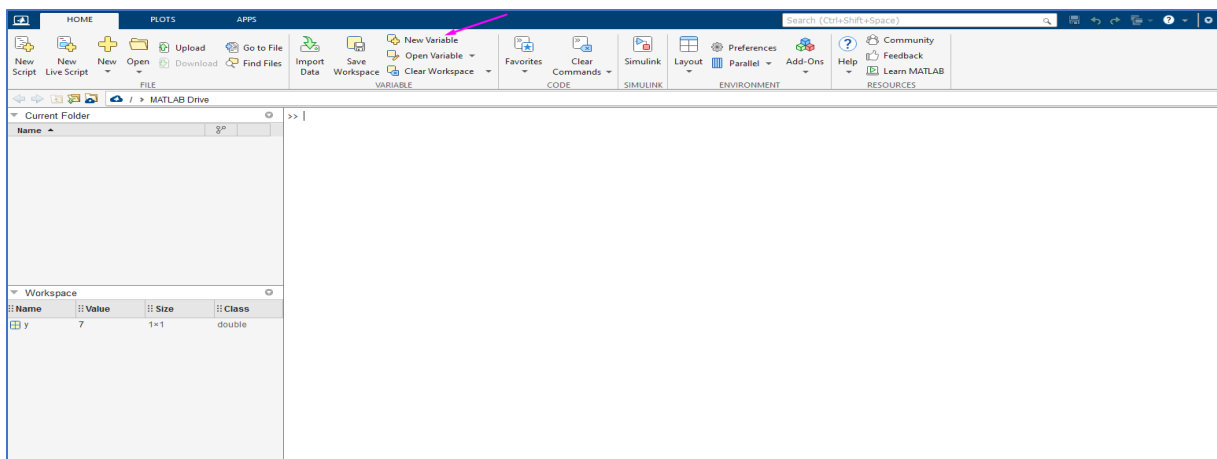
2×3 double					
	1	2	3	4	5
1	2	4	6		
2	8	10	12		
3					
4					
5					
6					
7					
8					
9					

Obr. 4.4. Príklad zobrazenia matice vo formáte: $z=[2\ 4\ 6; 8\ 10\ 12]$.

Zmeny v premenných môžeme vykonávať aj priamo v mriežke matice, napr. Obr. 4.4.

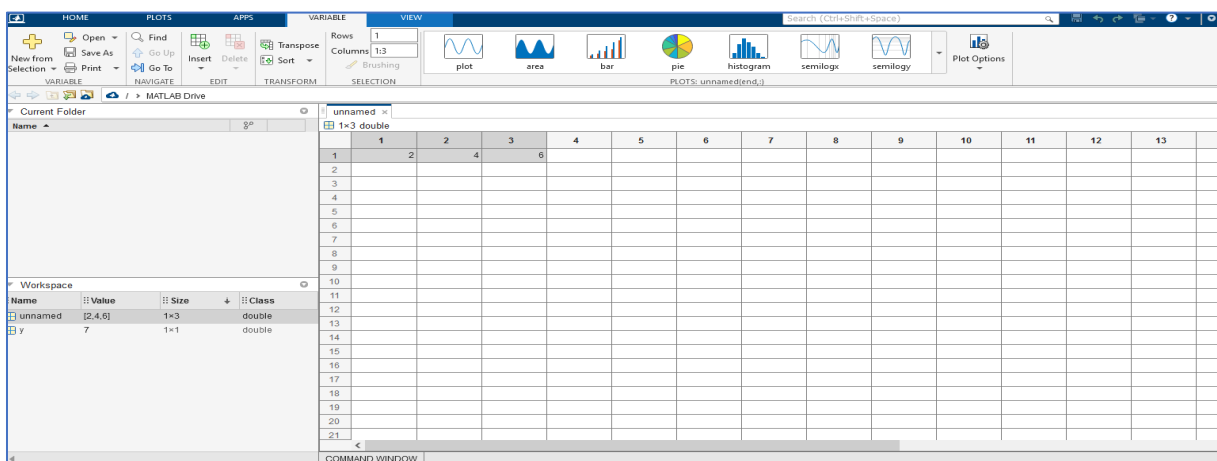
4.1 Vytvorenie novej premennej

Vyberieme si v príkazovom riadku označenie novej premennej, viď. Obr. 4.5.



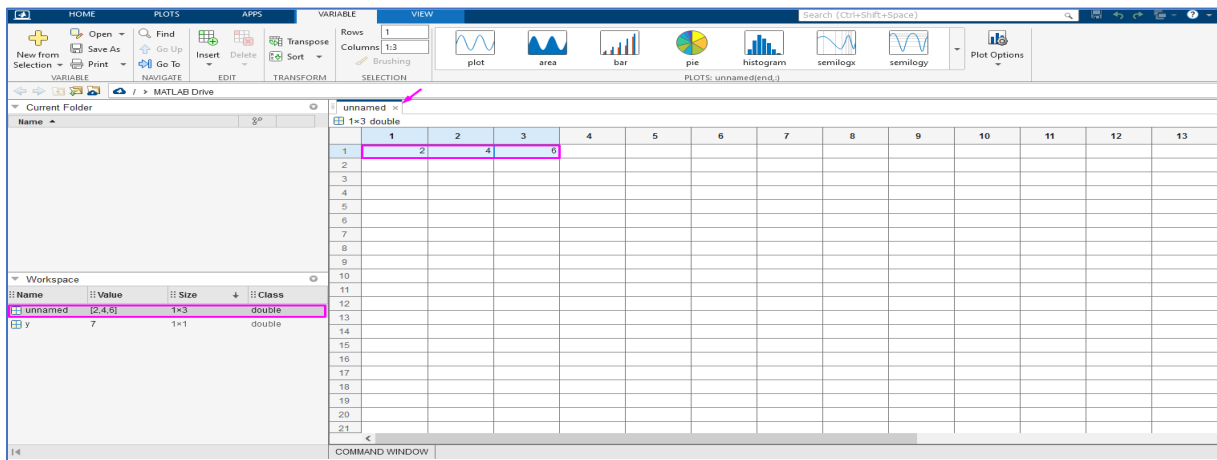
Obr. 4.5. Označenie výberu novej premennej v príkazovom riadku.

Hodnoty premenných môžeme zapísať podľa Obr. 4.6 priamo do mriežky matice.



Obr. 4.6. Zapísanie hodnôt premenných priamo do mriežky matice.

Zapísané hodnoty premenných napr. z [Obr. 4.6.](#) po zatvorení okna Command Window nájdeme vo Workspace (v pracovnom okne), kde ich vieme rovnako priamo premenovať, viď [Obr. 4.7.](#)



Obr. 4.7. Jedna z možností uloženia a pomenovania premenných priamo vo Workspace.

5. Základné aritmetické operácie

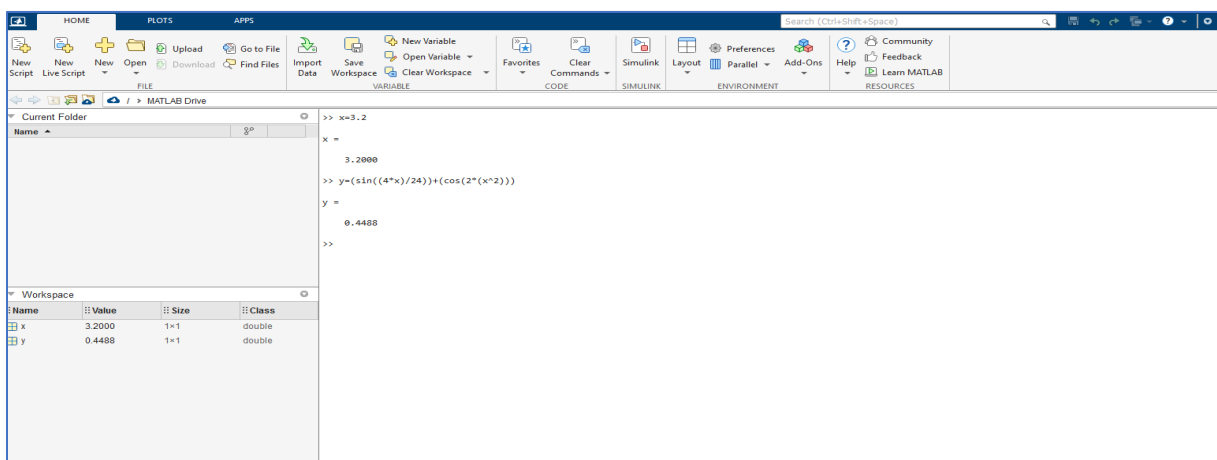
Základnými aritmetickými operáciami v MATLABe sú: sčítanie, odčítanie, násobenie, delenie, mocnina, zaokrúhľovanie. Aritmetické funkcie zahŕňajú operátory pre jednoduché operácie, ako je sčítanie a násobenie, ako aj funkcie pre bežné výpočty, ako je súčet, pohyblivé súčty, modulo-operácie a zaokrúhľovanie.

Ďalšie informácie nájdeme v časti (klikni na): [Operácie pole vs. matice.](#)

Upozornenie: MATLAB striktno rozlišuje malé a veľké písmená.

Matematické operácie, sčítanie, odčítanie, násobenie a delenie sa zadávajú do MATLABu rovnako ako je to v programe Excel. Napr. rovnicu (1), zapíšeme tak, ako uvádza [Obr. 5.1.](#)

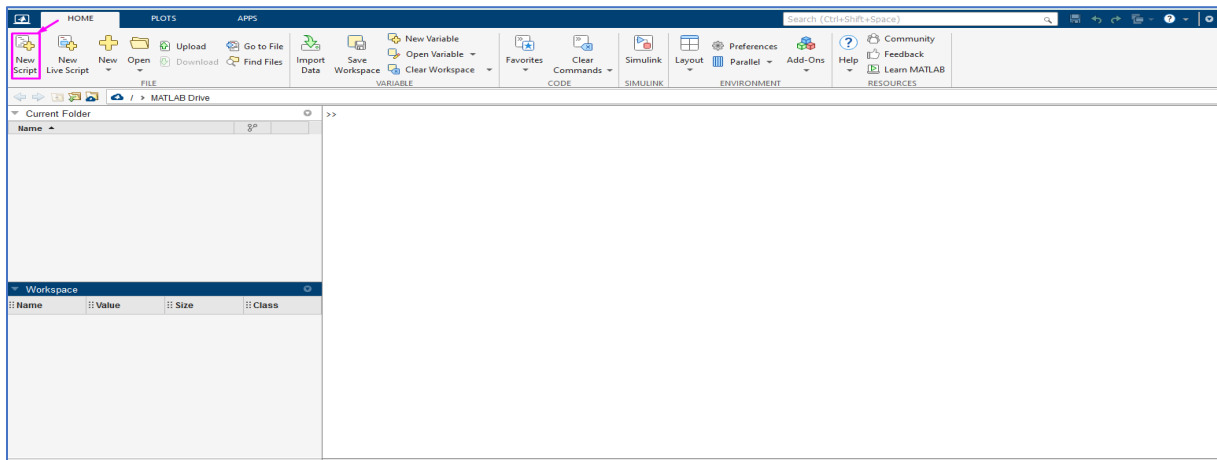
$$y = \sin \frac{4x}{24} + \cos(2x)^2 \quad (1)$$



Obr. 5.1. Príklad zápisu rovnice (1) do MATLAB.

6. Začiatok v písaní skriptov

Ak chceme vytvoriť nový skript, klikneme v príkazovom riadku na ikonu zobrazenú na [Obr. 6.1](#).



Obr. 6.1. Výber ikony *New script* zo záhlavia programu MATLAB.

Príklad 1

Určite hĺbku h_t , do ktorej sa dostane čistočka vody pri prechode nasýteným pôdnym profilom ak poznáte: čas pohybu t , počiatočnú hĺbku h_0 , nasýtenú hydraulickú vodivosť K , hydraulický sklon i .

$$\text{Rýchlosť prúdenia vody} \quad v = K \cdot i \quad (2)$$

$$\text{Dráha, ktorú prejde častica vody} \quad s_t = v \cdot t \quad (3)$$

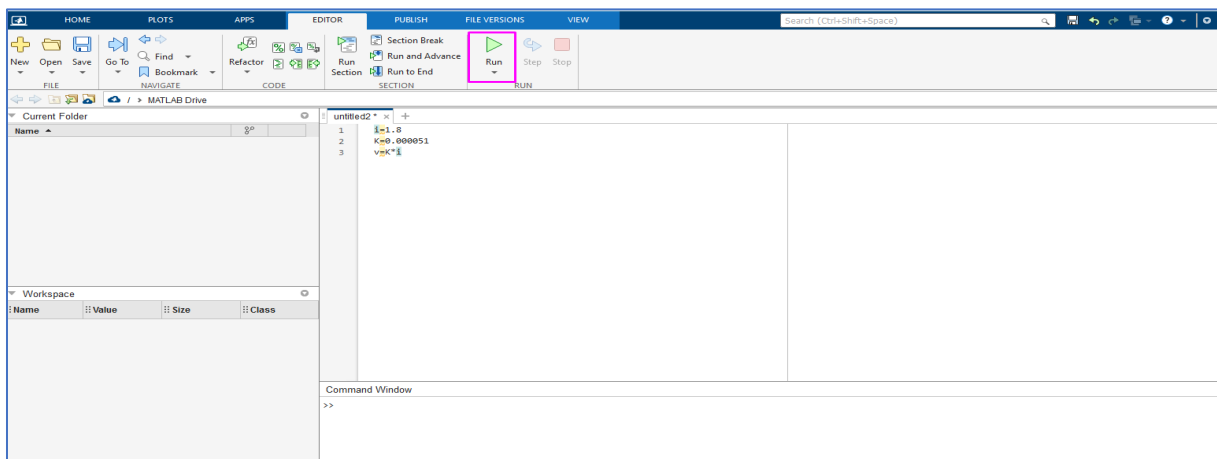
$$\text{Hĺbka, v ktorej bude čistočka vody v čase } t \quad h_t = h_0 + s_t \quad (4)$$

Zadanie

t	h_0	K	i
[s]	[m]	[m.s ⁻¹]	[bezr.]
10000	1,3	0,000051	1,8

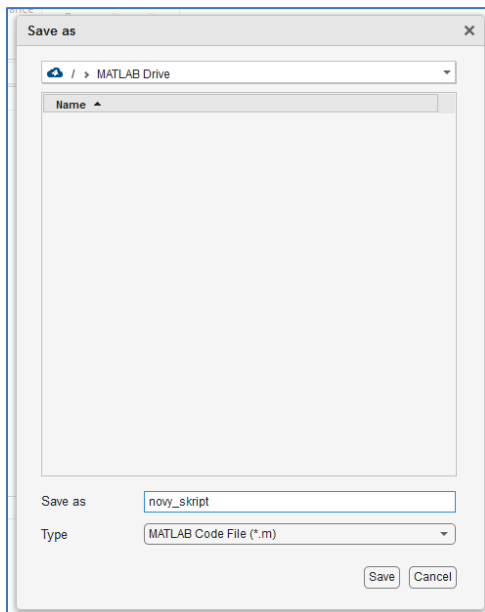
Najskôr zadefinujem v MATLAB známe premenné K a i a až potom vypočítam v , vid'. [Obr. 6.2](#). Následne vypočítam aj rovnicu (3) a (4).

Nový skript si uloží po stlačení ikony „Run“ ([Obr. 6.2](#)).



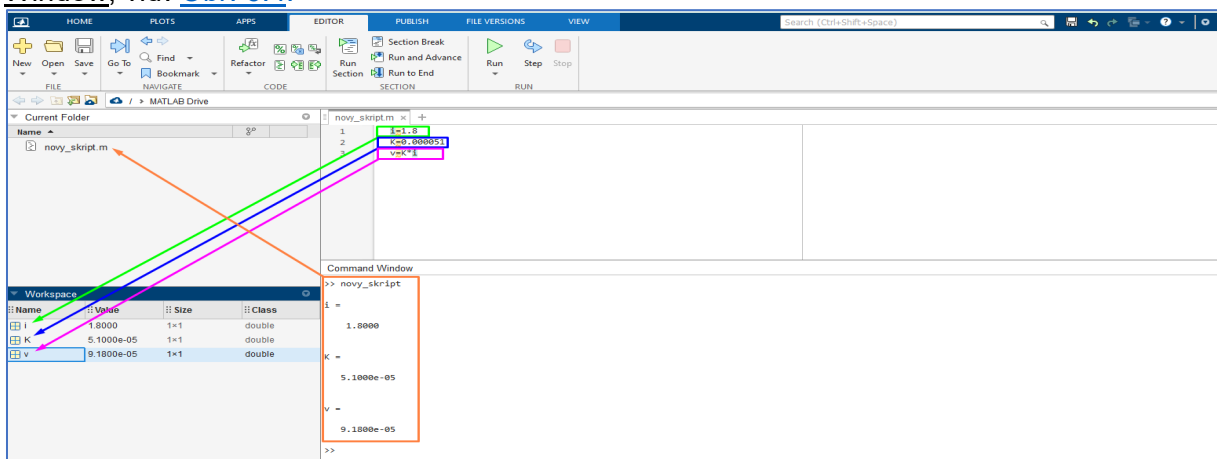
Obr. 6.2. Výpočet rovnice (2) v prostredí MATLAB.

Napokon si nový skript uložíme, vid'. [Obr. 6.3](#) pre uloženie v online verzii na MATLAB Drive.



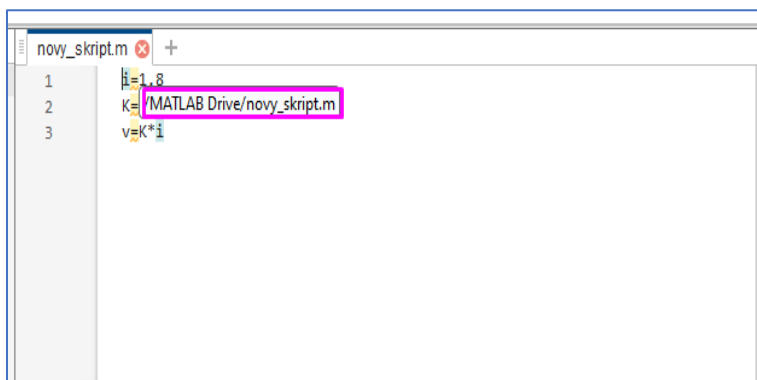
Obr. 6.3. Uloženie nového skriptu (novy_skript) v online verzii MATLAB na MATLAB Drive.

Jednotlivé členy rovnice (2) sú uložené vo Workspace, so zobrazením výsledku v Command Window, vid'. [Obr. 6.4](#).



Obr. 6.4. Zápis jednotlivých členov rovnice (2) a jej výpočet s následným zobrazením v Command Window.

[Obr. 6.5](#). znázorňuje, kam sa skript ukladá.



Obr. 6.5. Príklad uloženia nového skriptu na MATLAB Drive v prostredí MATLAB online.

Ak použijeme znak %, môžeme zaň napísať akúkoľvek poznámku a MATLAB ju nepoužije pri výpočte, viď. [Obr. 6.6](#).

```

1  i=1.8      % hydraulický sklon
2  K=0.000051 % nasýtená hydraulická vodivosť
3  v=K*i     % rýchlosť prúdenia vody
4

```

Obr. 6.6. Použitie znaku % pre vytvorenie poznámky alebo komentára pri výpočte MATLAB.

V prípade, že urobíme chybu, MATLAB automaticky upozorní, kde sa chyba nachádza, napr. chýbajúci znak „%“ viď. [Obr. 6.7](#).

```

1  i=1.8      % hydraulický sklon
2  K=0.000051 % nasýtená hydraulická vodivosť
3  v=K*i     % rýchlosť prúdenia vody
4

```

Command Window

```

1.8000
K =
5.1000e-05
v =
9.1800e-05
>> novy_skr ipt
#1: Invalid expression. Check for missing multiplication operator, missing or unbalanced delimiters, or other syntax error. To construct matrices, use brackets instead of parentheses.
>>

```

Obr. 6.7. Zobrazenie upozornenia na chybu v Command window, pri chýbajúcom znaku %.

Znak „;“ použijem v tých prípadoch, ak nemáme záujem o zobrazenie výsledku v Command window. Výsledok sa zobrazí na Workspace, ale riadky označené týmto znakom nie sú obsiahnuté v skripte popísanom v Command window, viď. [Obr. 6.8](#).

```

1  i=1.8;    % hydraulický sklon
2  K=0.000051; % nasýtená hydraulická vodivosť
3  v=K*i;   % rýchlosť prúdenia vody
4
5
6
7
8

```

Command Window

```

>> novy_skr ipt
v =
9.1800e-05
>>

```

Obr. 6.8. Možnosť použitia znaku „;“, za ktorým sa následne nezobrazí výpočet a výsledok v Command window.

7. Galéria živých skriptov

Živé skripty a živé funkcie MATLAB sú interaktívne dokumenty, ktoré kombinujú MATLAB kód s formátovaným textom, rovnicami a obrázkami v jedinom prostredí nazývanom Live Editor. Okrem toho živé skripty ukladajú a zobrazujú výstup spolu s kódom, ktorý ho vytvára.

Použitie (klikni na interaktívne prepojenie) [živých skriptov a ich funkcií](#) je pre nasledovné oblasti (ch.mathworks.com, 2022) s ich výhodami, ktoré sú:

A) Vizualne skúmanie a analýza problémov

- Píšeme, spúšťame a testujeme kód v jedinom interaktívnom prostredí.
- Spúšťame bloky kódu jednotlivo (iba živé skripty) alebo ako celý súbor a prezeráme si výsledky a grafiku s kódom, ktorý ich vytvoril.

B) Zdieľanie formátov:

- Pridávame nadpisy, podnadpisy a formátovaný text na popis procesu, ako aj zahŕňame rovnice, obrázky a hypertextové odkazy ako podporný materiál.
- Uložíme si svoje analýzy ako bohato formátované, spustiteľné dokumenty a zdieľame ich s kolegami alebo s komunitou MATLAB alebo ich skonvertujeme na dokumenty HTML, PDF, Microsoft® Word alebo LaTeX na publikovanie.

C) Vytváranie interaktívnych prednášok na vyučovanie

- Kombinujeme kód a výsledky s formátovaným textom a matematickými rovnicami.
- Vytvárame prednášky krok za krokom a postupne ich vyhodnocujeme, aby sme ilustrovali tému.
- Upravujeme kód za chodu, aby ste mohli odpovedať na otázky alebo preskúmať súvisiace témy.
- Zdieľame prednášky so študentmi ako interaktívne dokumenty alebo v tlačenej podobe a distribuujeme čiastočne dokončené súbory ako úlohy.

Live Skript Gallery (Galéria živých skriptov) sa nachádza na: <https://ch.mathworks.com/products/matlab/live-script-gallery.html>. Podľa (klikni na prepojenie) [Loren on the Art of MATLAB](#) (2022) galéria obsahuje 30 ukážkových živých skriptov. Pre ktorýkoľvek z týchto príkladov môžeme navštíviť položku File Exchange (výmena súborov) kliknutím na „View on File Exchange.“ Môžeme si tam stiahnuť príklad a spustiť ho v programe MATLAB alebo v MATLAB online.

Najdôležitejšie je, že môžeme spustiť (a dokonca upraviť) ktorýkoľvek z týchto príkladov vo svojom prehliadači kliknutím na „Run this Example“ (Spustiť tento príklad). Tým sa otvorí živý skript vo verzii MATLAB Live Editor založený na prehliadači, ktorý používa MATLAB online. Keď otvoríme konkrétny príklad a spustíme ho v prehliadači, uvidíme to, čo vyzerá ako na obrázku, ktorý je v konkrétnom skripte uvedený nižšie. Živý skript môžeme spustiť pomocou tlačidla "Run" (Spustiť) na paneli nástrojov. Jednotlivé sekcie môžeme spustiť aj z panela nástrojov.

8. Grafické zobrazovanie v MATLABe

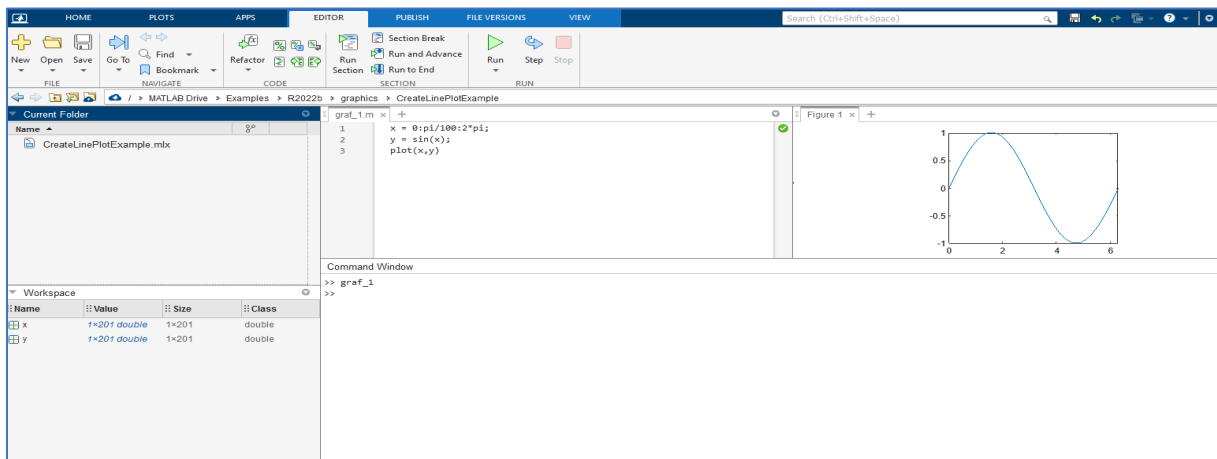
MATLAB umožňuje vykresľovanie pútavých a pre užívateľov jednoduchých grafov. Pomocou MATLABu užívateľ vytvára rôzne grafické objekty, vid'. [Galéria grafov MATLAB](#). Galéria grafov MATLAB poskytuje rôzne príklady grafického zobrazenia údajov v MATLABe. Kliknutím na „Spustiť príklad“ v Galérii otvoríte a spustíte príklady živých skriptov vo svojom prehliadači pomocou MATLAB Online™. Ďalšie možnosti nájdeme na stránke MATLAB Live Script Gallery, kde zároveň spustíme príklady živých skriptov z komunity MATLAB, vid'.: ch.mathworks.com (2022).

8.1 Vytvorenie predefinovaného 2D grafu

Voľne dostupné príklady na zostrojenie 2D grafov v MATLABe sú zadarmo k dispozícii na stránke: <https://ch.mathworks.com/help/matlab/ref/plot.html>

Na tejto stránke je možnosť vybrať si: a.) Open in MATLAB Onlive (otvoriť online) alebo b.) Copy Command (príkaz na kopírovanie).

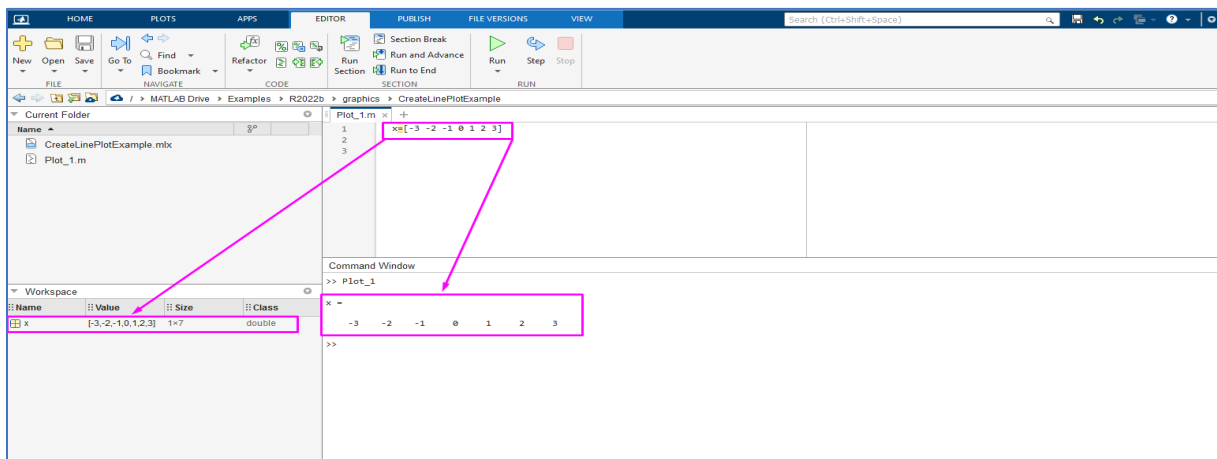
Príklad vytvorenia grafu zo stránky Mathworks.com (2022) zobrazuje [Obr. 8.1](#).



Obr. 8.1. Zobrazenie grafu na základe príkladu zo stránky Mathworks.com (2022).

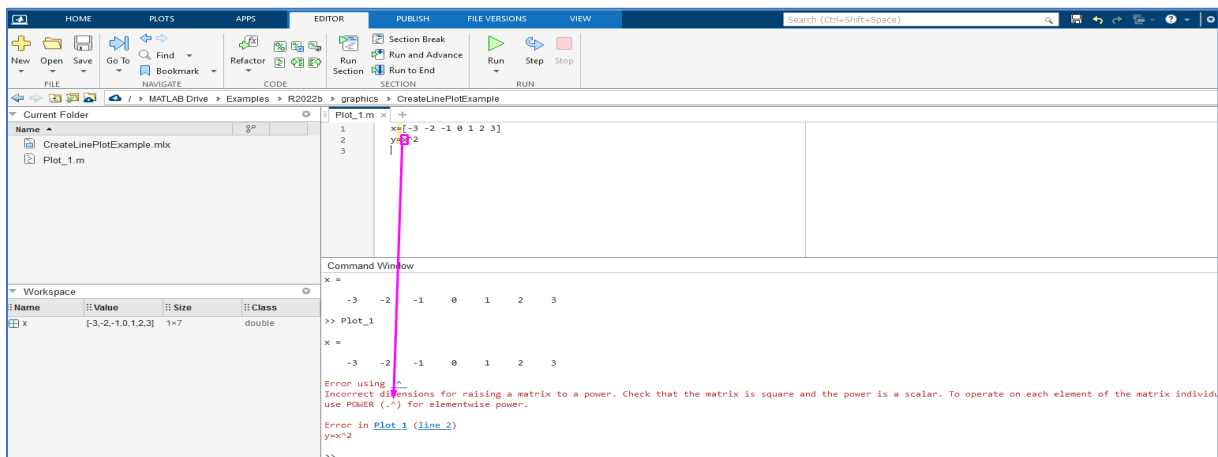
8.2 Vytvorenie vlastného 2D grafu

Vytvorenie vlastného grafu je na príklade zostrojenia grafu paraboly podľa Auraspa (2021). Najskôr zadefinujeme hodnoty x , viď [Obr. 8.2](#).



Obr. 8.2. Priebeh zostrojenia grafu paraboly (Auraspa, 2021).

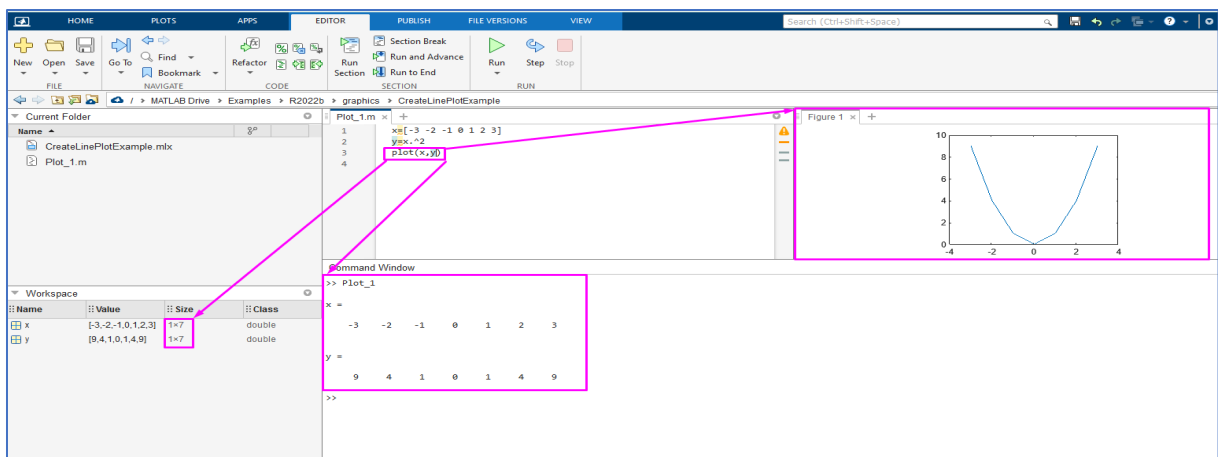
Ak zadefinujeme $y=x^2$, čo je rovnica paraboly, MATLAB upozorní na chybu, vid' [Obr. 8.3](#).



Obr. 8.3. Príklad chybového hlásenia pri zostrojení grafu paraboly podľa Aurasup (2021).

V tomto prípade je potrebné označiť x ako vektor, t. j. ako: „x.“

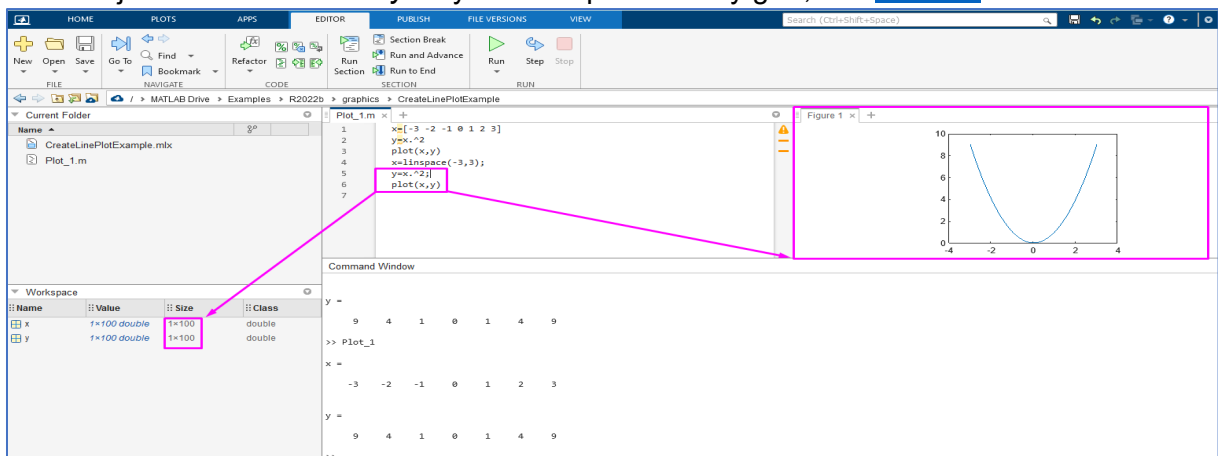
Potom je možné zostaviť graf paraboly, ktorý však nie je vizuálne vzhľadný, nakoľko je zostavený zo 7 bodov, vid' [Obr. 8.4](#).



Obr. 8.4. Zobrazenie grafu paraboly zo 7 hodnôt podľa Aurasup (2021).

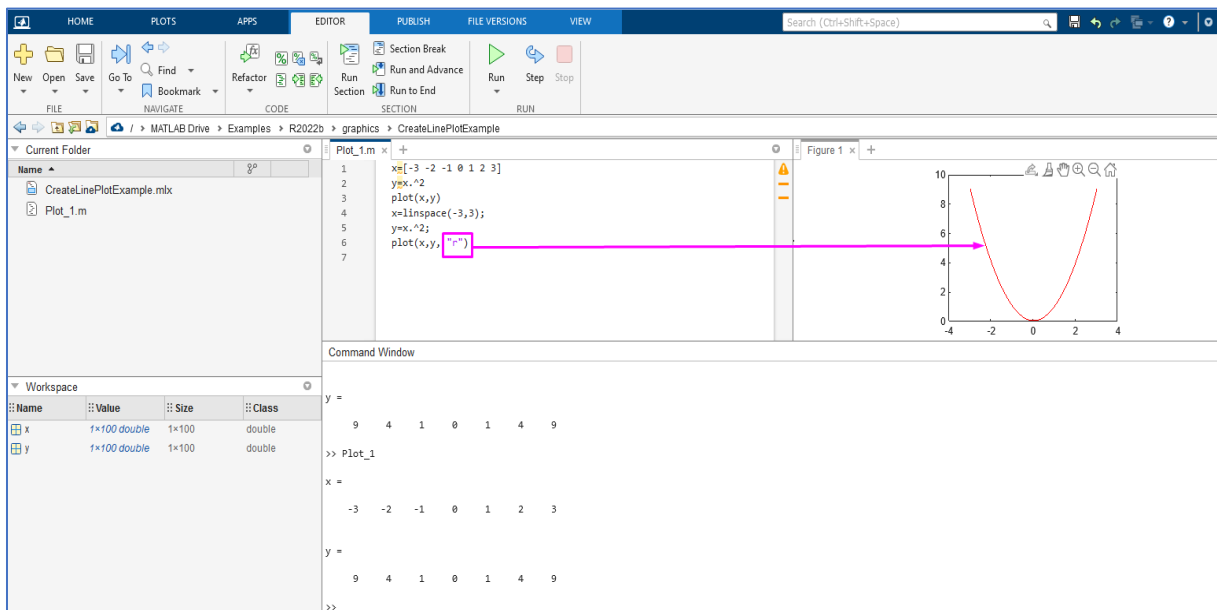
Na vytvorenie užívateľsky lepšie zobrazeného grafu je potrebné použiť príkaz „*linspace*“, ako uvádza [Obr. 8.5](#). Po takomto príkaze obsahuje hodnota vektoru x až 100 čísel.

Zadefinujeme znova hodnotu y a vykreslíme požadovaný graf, vid' [Obr. 8.5](#).



Obr. 8.5. Zobrazenie grafu paraboly na základe príkazu „*linspace*“ podľa Aurasup (2021).

V grafe môžeme zmeniť farbu zobrazenia, napr. na červenú (Obr. 8.6). Všetky farebné a grafické možnosti zobrazenia grafov v prostredí MATLAB sú podrobne uvedené v časti Input Arguments (vstupy) na stránke: <https://ch.mathworks.com/help/matlab/ref/plot.html>.



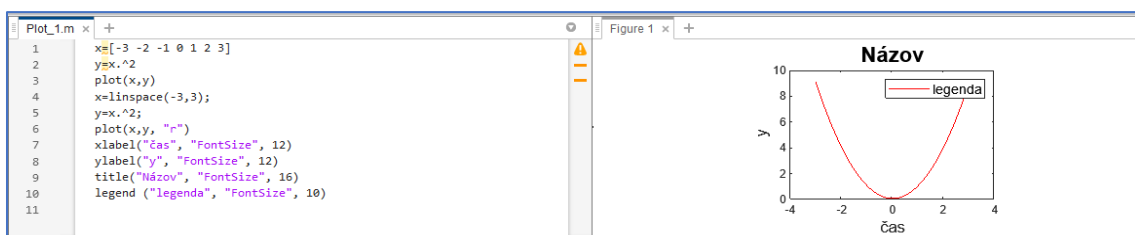
Obr. 8.6. Zmena farebného zobrazenia grafu paraboly na základe príkazu „`plot(x,y, 'r')`“.

Pre označenie osí používame príkaz `label`. Napr.: popis osi x s názvom čas a veľkosťou písma 12 vytvorím ako `xlabel('čas', 'FontSize', 12)`;

Pre zadanie názvu grafu použijeme príkaz `title`. napr.: `title('názov grafu', 'FontSize', 16)`.

Pre pomenovanie názvu legendy použijeme príkaz `legend`. Napr.: `legend('legenda', 'FontSize', 10)`.

Vyššie uvedený príklad popísania osi, názvu grafu a legendy znázorňuje [Obr. 8.7](#).



Obr. 8.7. Príklad pomenovania popisu osí, názvu legendy a pre zostrojenie grafu.

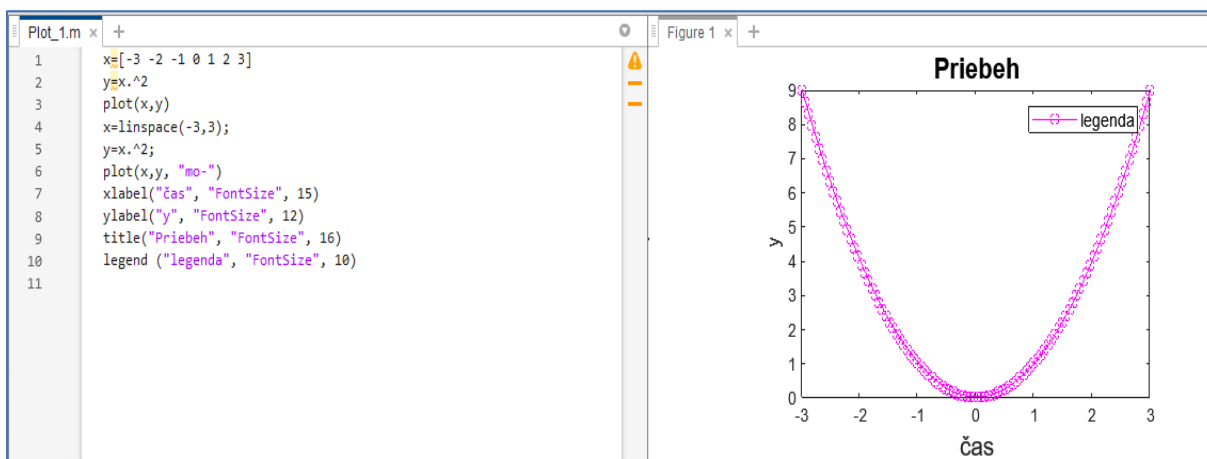
Ak chceme vyhľadať všetky dostupné možnosti na zmenu farby, hrúbky a označenia čiary v grafe, použijeme v Command window príkaz `Plot help` (pozri [Obr. 8.8](#)).

Rôzne typy čiar, symboly a farby grafu možno získať pomocou grafu (X,Y,S), kde S je reťazec znakov vytvorený z jedného prvku z ktoréhokoľvek alebo zo všetkých nasledujúcich 3 stĺpcov:

b	blue	.	point	-	solid
g	green	o	circle	:	dotted
r	red	x	x-mark	-.	dashdot
c	cyan	+	plus	--	dashed
m	magenta	*	star	(none)	no line
y	yellow	s	square		
k	black	d	diamond		
w	white	v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		

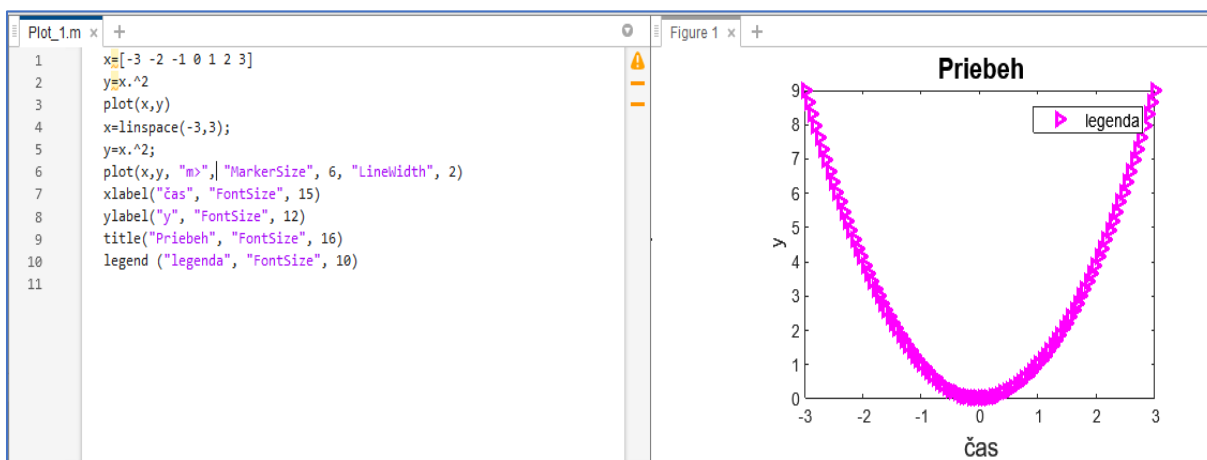
Obr. 8.8. Možnosti farebného zobrazenia (1. stĺpec), grafických značiek (2. stĺpec) a spojovníka medzi grafickými značkami (3. stĺpec) pre zostrojenie grafu v MATLAB.

[Obr. 8.9.](#) znázorňuje zmenu farby na magenta (fialová) pomocou označenia „m“, označenia grafických znakov na krúžok „o“ a ich spojenie s plnou čiarou „-“.



Obr. 8.9. Priebeh zobrazenia grafu na magenta (fialová) pomocou označenia „m“, značky grafických znakov na krúžok „o“ a ich spojenie s plnou čiarou „-“.

Pre zmenu veľkosti grafickej značky v grafe použijem príkaz „*MarkerSize*“. Napr. vid’ [Obr. 8.10.](#)

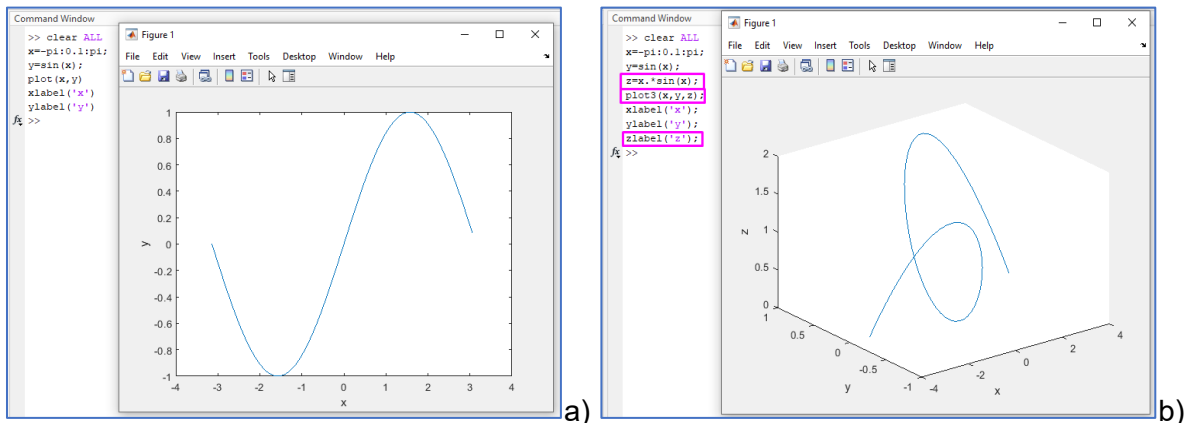


Obr. 8.10. Príklad použitia príkazu *MarkerSize* 6 na zväčšenie vyznačenia grafických znakov (v tomto prípade na veľkosť 6).

8.3 Vytvorenie 3D grafu

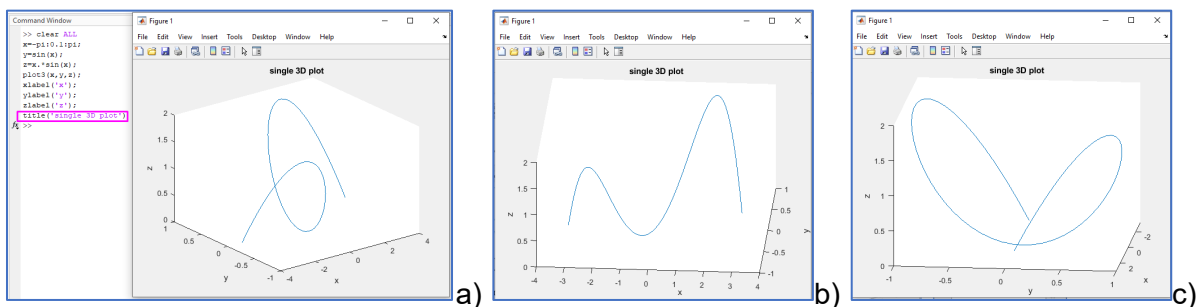
Postup zobrazenia všetkých dole uvedených grafov v tejto časti je možné sledovať aj vo videách Learning Vibes (2021). V princípe 2D graf vyžaduje 2 osi a zobrazuje 2 súradnice, ale pre 3D graf potrebujeme 3 osi a 3 súradnice.

3D graf si môžeme pripraviť aj prerobením 2D grafu ([Obr. 8.11.a](#)) pridaním osi z na 3D graf ([Obr. 8.11.b](#)).



Obr. 8.11. Príklad zmeny zobrazenia a) 2D grafu na b) 3D graf (Learning Vibes, 2021).

Graf pomenujeme a po jeho vytvorení ho otáčame ľubovoľným smerom vid'. [Obr. 8.12. a\) b\) c\).](#) Na každom z obrázkov vidíme rovnaký graf, t. j. graf si vieme pozrieť z rôznych uhlov.

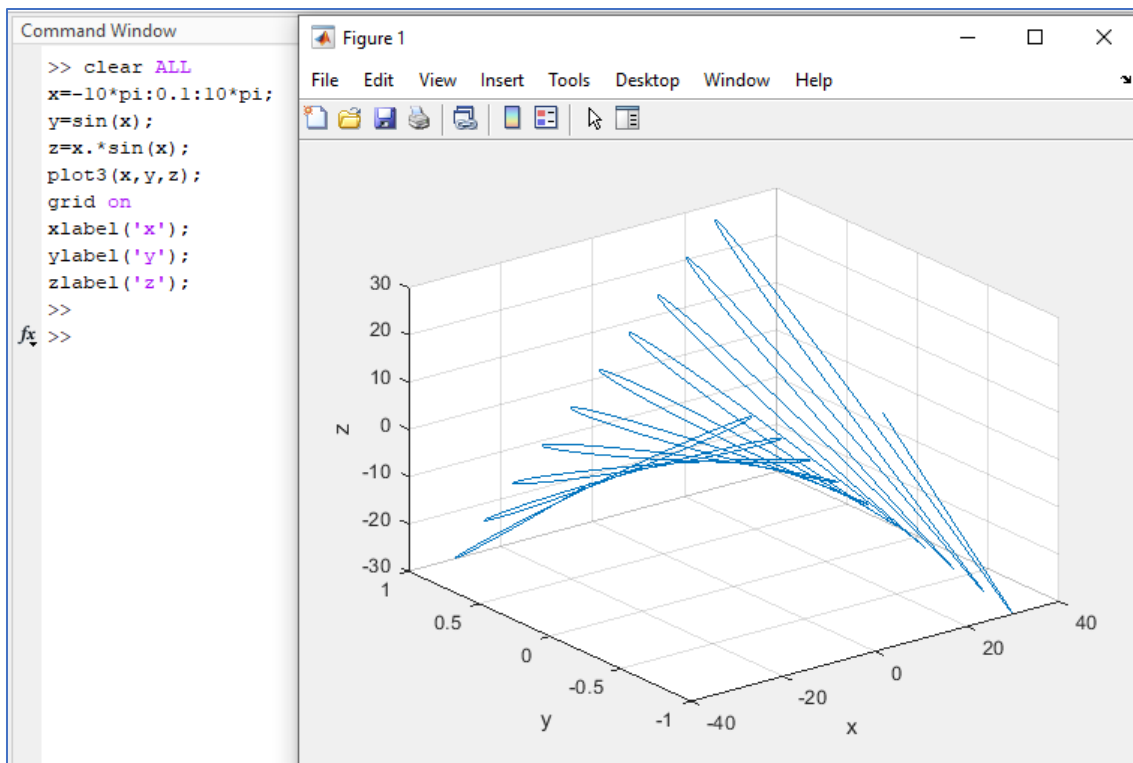


Obr. 8.12. Rôzne pohľady na rovnaký graf zobrazený v rôznom a) uhle, b) smere a c) výške zobrazenia.

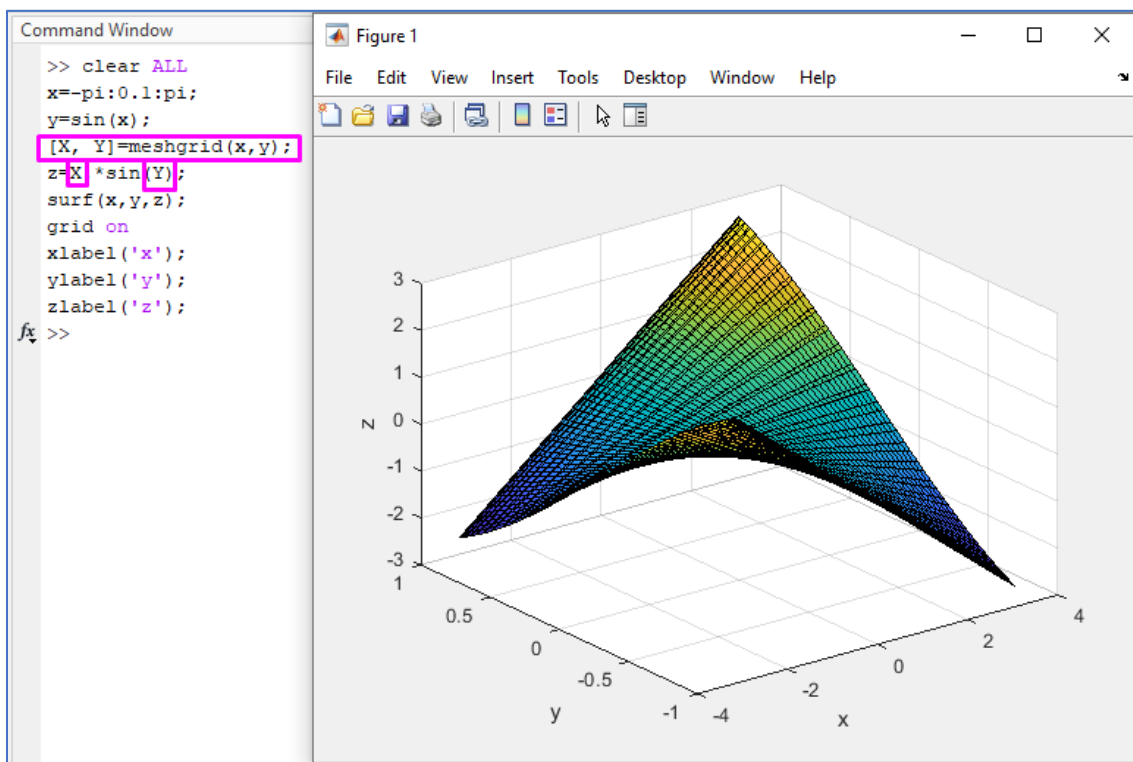
Graf si vieme nastaviť na konkrétny uhol pohľadu pridaním konkrétneho príkazu na *azimut* (*az*) (z akého uhla potrebujeme pohľad) a *elevation* (*el*) (z akej výšky sa na objekt pozeráme). Konkrétny požadovaný pohľad pridáme príkazom *view* (pohľad), napr. *view(az,el)*. Ak nepridáme tento konkrétny príkaz pre 3D graf, zobrazí sa v defaultnom (prednastavenom) zobrazení.

Vytvorený čiarový 3D graf ([Obr. 8.13.](#)) vieme jednoduchým spôsobom zmeniť na povrchový graf ([Obr. 8.14.](#)). V takomto prípade je potrebné zmeniť príkaz *plot* na *surf* a premennú z zdefinovať ako maticu príkazom *meshgrid*, napr. $[X, Y]=\text{meshgrid}(x,y)$, vid'.: [Obr. 8.14.](#)

Príkaz *mesh* v MATLABe označuje sieť čiar medzi jednotlivými bodmi.



Obr. 8.13. Vytvorenie čiarového 3D grafu podľa Learning Vibes (2021).



Obr. 8.14. Vytvorenie plošného 3D grafu pridaním matice (Learning Vibes, 2021).

8.4 Vytvorenie animovaného grafu

Príklad zobrazenia animovaného grafu uvádza [Obr. 8.15](#). Animácia sa však zobrazí až na základe zobrazenia tohto príkladu v MATLABe.

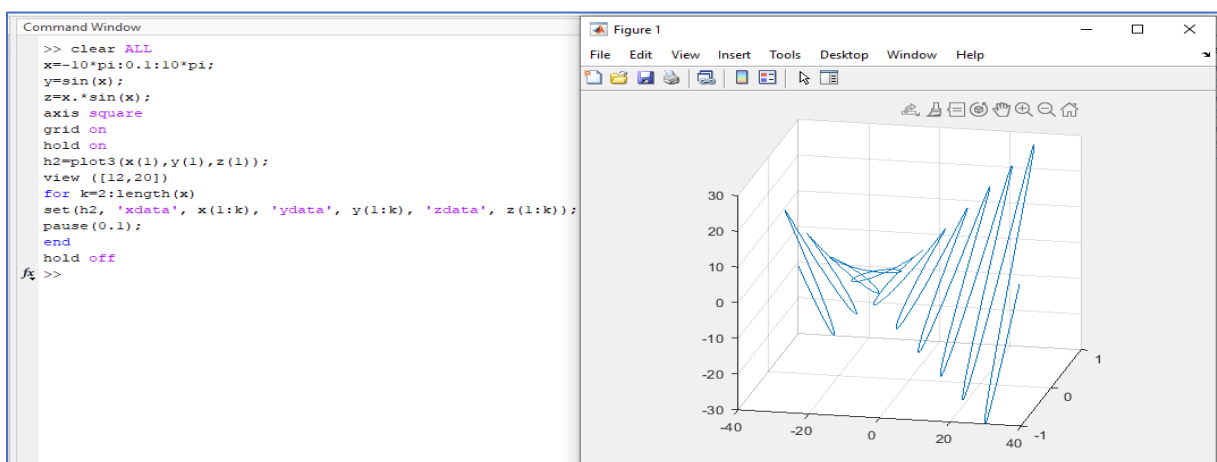
Ďalšou možnosťou na zobrazenie povrchového 3D grafu je príkaz *surf*.

Príkaz *surf* v MATLABe vloží farby medzi jednotlivé čiary.

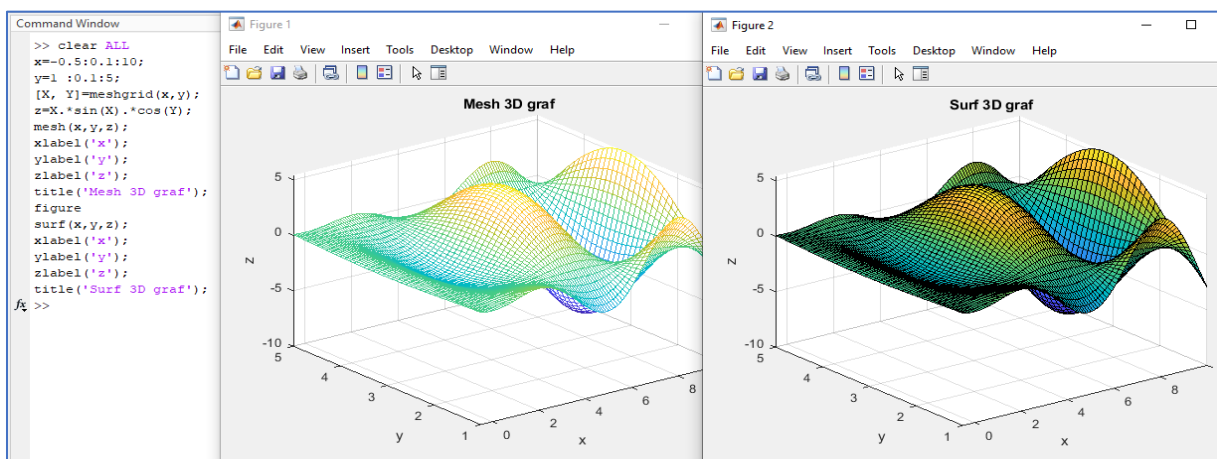
Rozdiel medzi *mesh* a *surf* 3D grafom zobrazuje [Obr. 8.16](#).

Pomocou príkaz *shading interp* odstránime čierne línie z 3D grafu, pozri [Obr. 8.17](#).

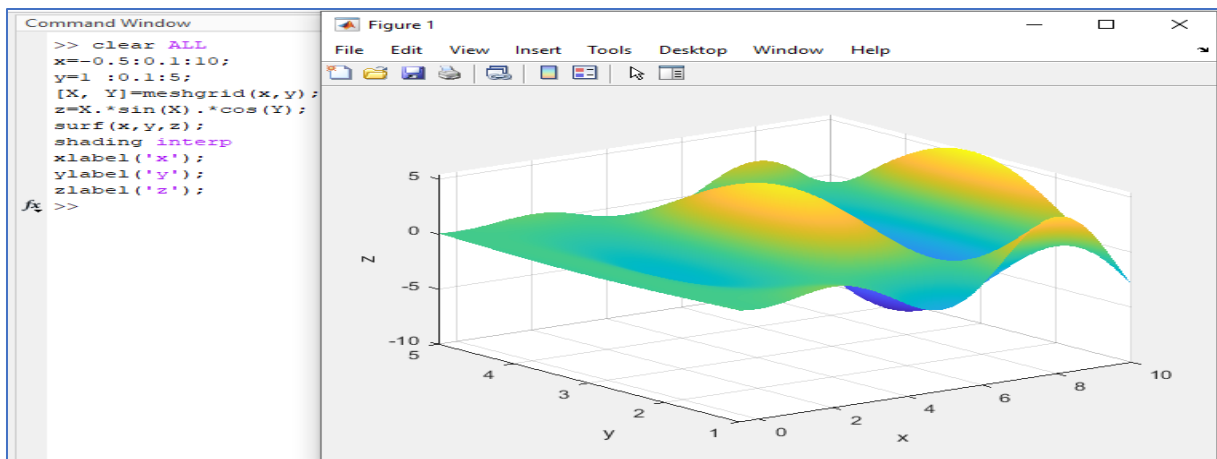
Pre doplnenie farebnej škály 3D grafu použijeme príkaz *colorbar*, viď. [Obr. 8.18](#).



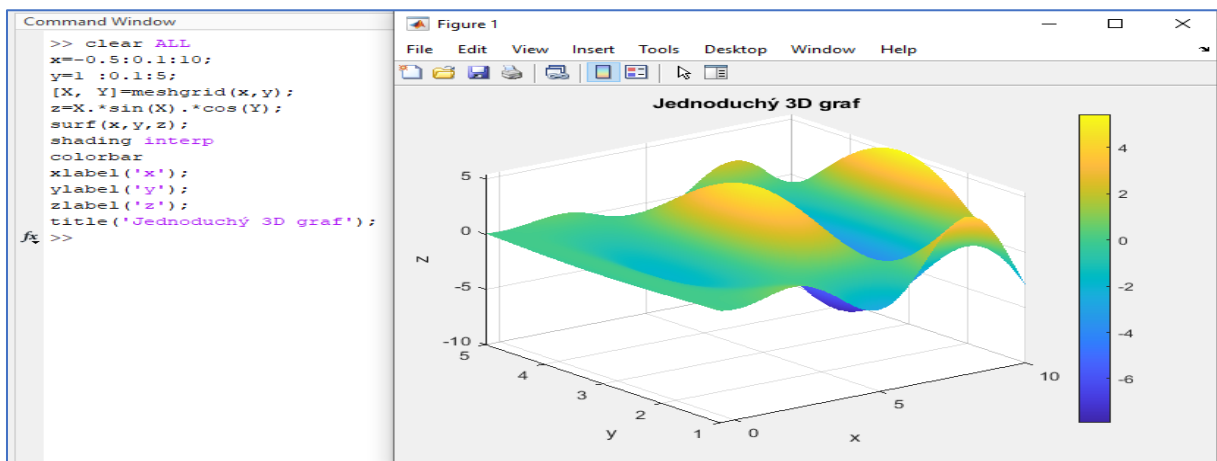
Obr. 8.15. Príklad zobrazenia animovaného grafu v prostredí MATLAB (Learning Vibes, 2021).



Obr. 8.16. Príklad zobrazenia 3D grafu v prevedení *mesh* a *surf* (Learning Vibes, 2021).



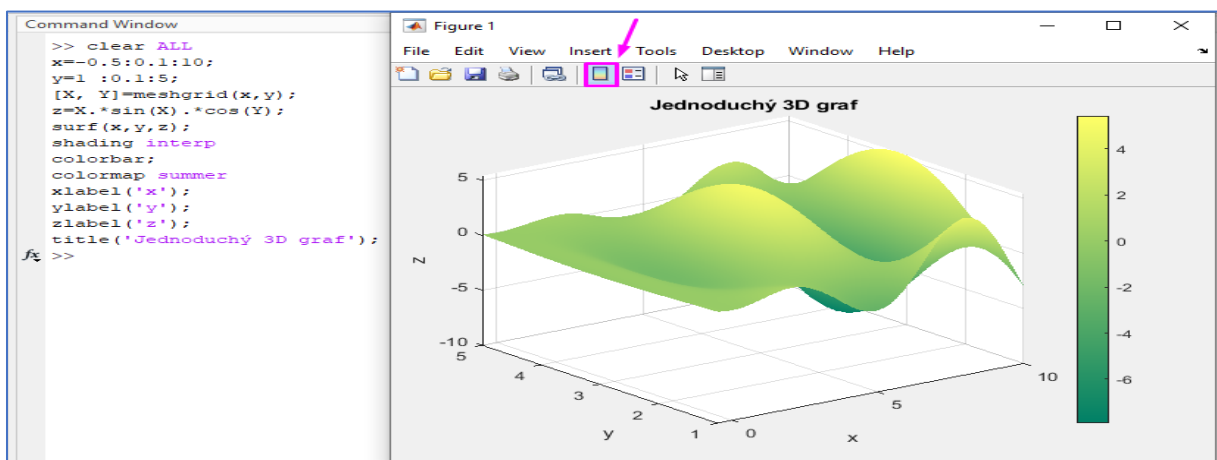
Obr. 8.17. Zobrazenie 3D povrchového (surf) grafu bez pridaných čiernych líni pomocou príkazu *shading interp* (Learning Vibes, 2021).



Obr. 8.18. Doplnenie farebnej stupnice do povrchového 3D grafu použitím príkazu *colorbar* (Learning Vibes, 2021).

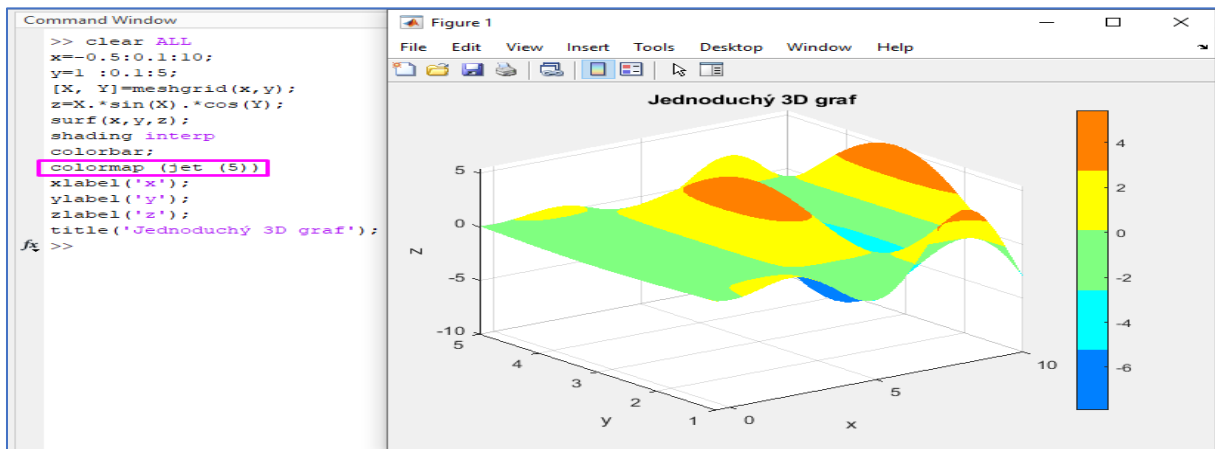
8.5 Nastavenie farebného prevedenia grafu

Farebné zobrazenie grafu je možné jednoducho zmeniť (Obr. 8.18) a to od prednastavených zobrazení až po vlastnú farebnú škálu na základe ikony zobrazenej na Obr. 8.19.



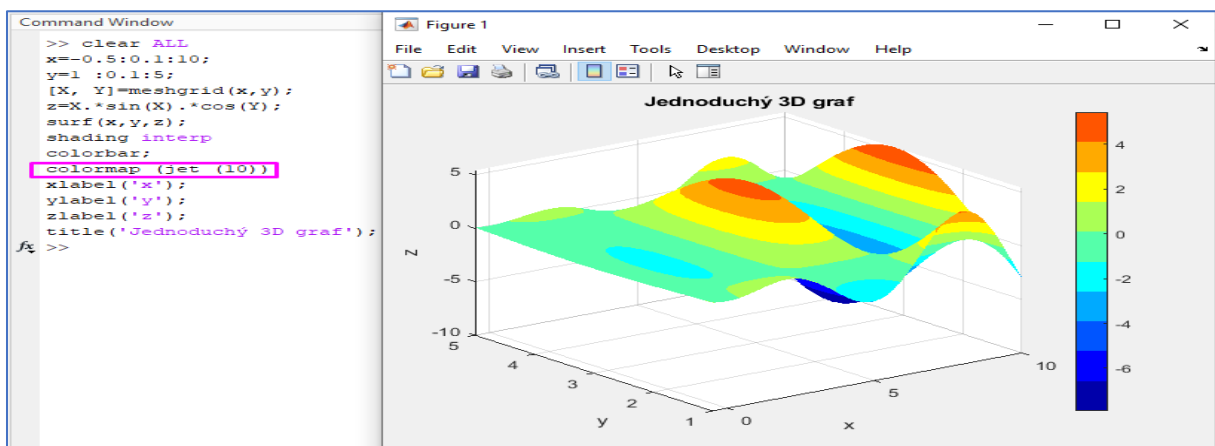
Obr. 8.19. Zmena farebného prevedenia grafu v zobrazení *colorbar summer* (Learning Vibes, 2021).

Ak potrebujeme zostrojiť graf iba z 5 farieb, zadefinujeme ich do príkazu napr. (*colorbar jet (5)*), vid'. [Obr. 8.20](#). Bližšie informácie o zmene farebného prevedenia grafov sú k dispozícii v nápovede: *Colormaps: View and modify colormap, control color scaling, add colorbar*, pozri: https://ch.mathworks.com/help/matlab/colors-1_it.html



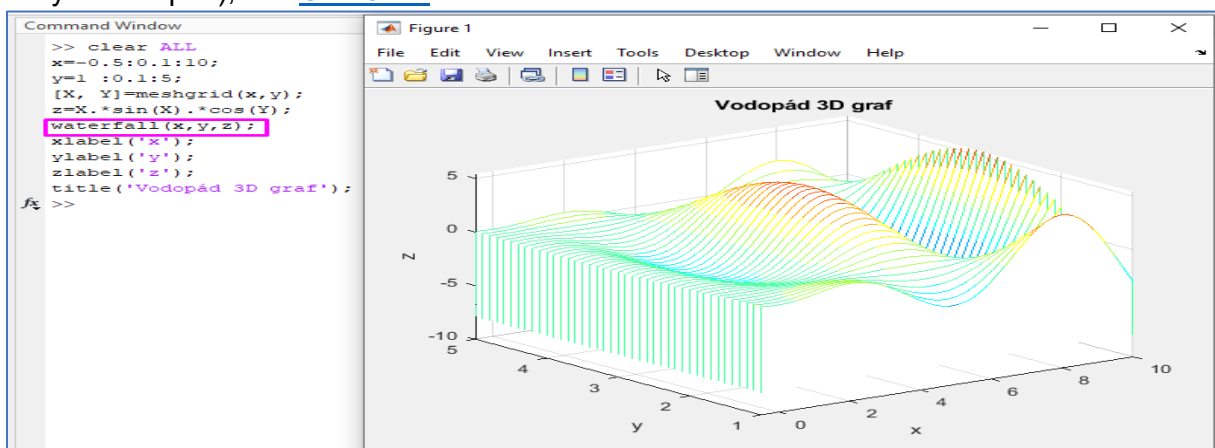
Obr. 8.20. Zobrazenie 5 farebného prevedenia 3D grafu pomocou príkazu (*colorbar jet (5)*).

[Obr. 8.21](#). zobrazuje graf zostrojený z 10 farieb, vo farebnom prevedení *jet*.



Obr. 8.21. Zobrazenie 10 škálového farebného prevedenia 3D grafu pomocou vo farebnom prevedení *jet*.

Ďalšou možnosťou zobrazenia grafu je *waterfall* (vodopád). Zobrazujeme ním (podobne ako pri vodopáde) postupnosť klesania jednotlivých línií (okraje sa podobajú vodopádu, preto sa nazýva vodopád), vid'. [Obr. 8.22](#).

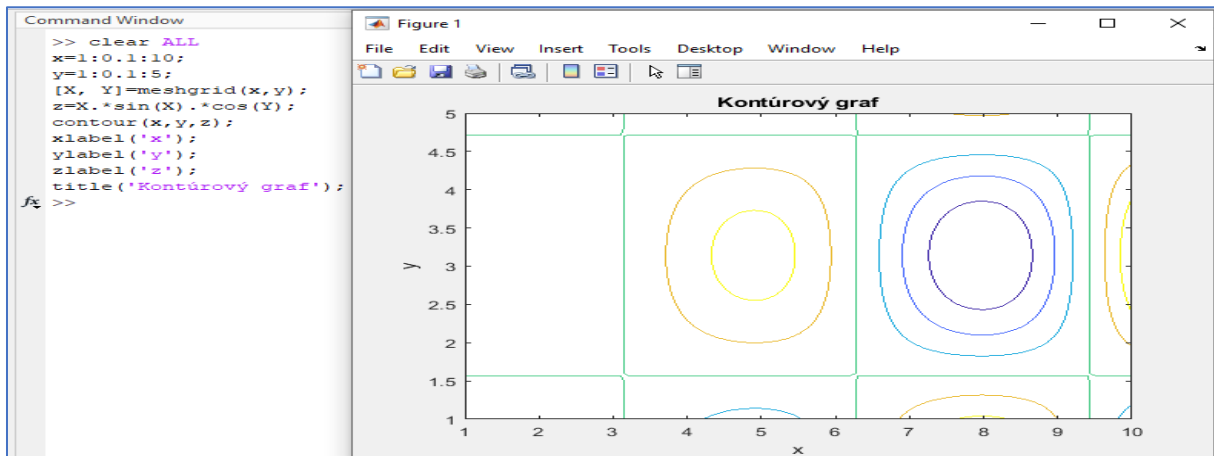


Obr. 8.22. Zobrazenie 3D grafu vo farebnom prevedení *waterfall* podľa Learning Vibes (2021).

Všetky z horeuvedených grafov môžeme zobrazovať vo zvoliteľnom sklone a výške, prípadne vybrať ich zobrazenie (azimut (az) a elevation (el)).

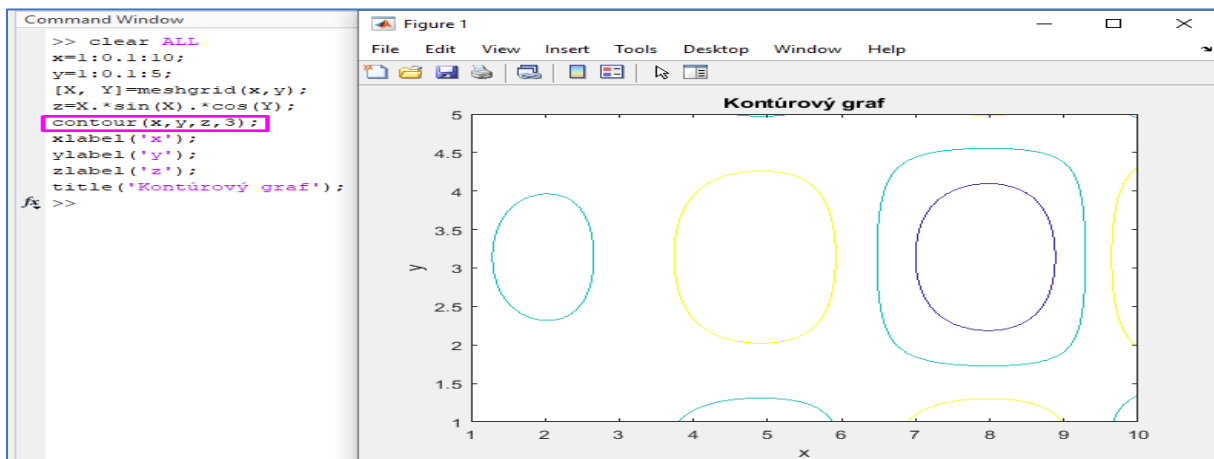
8.6 Zobrazenie kontúrového grafu

Príklad zobrazenia kontúrového grafu pomocou príkazu `contour` zobrazuje [Obr. 8.23](#), pričom počet línií v ňom je default (predefinovaný).

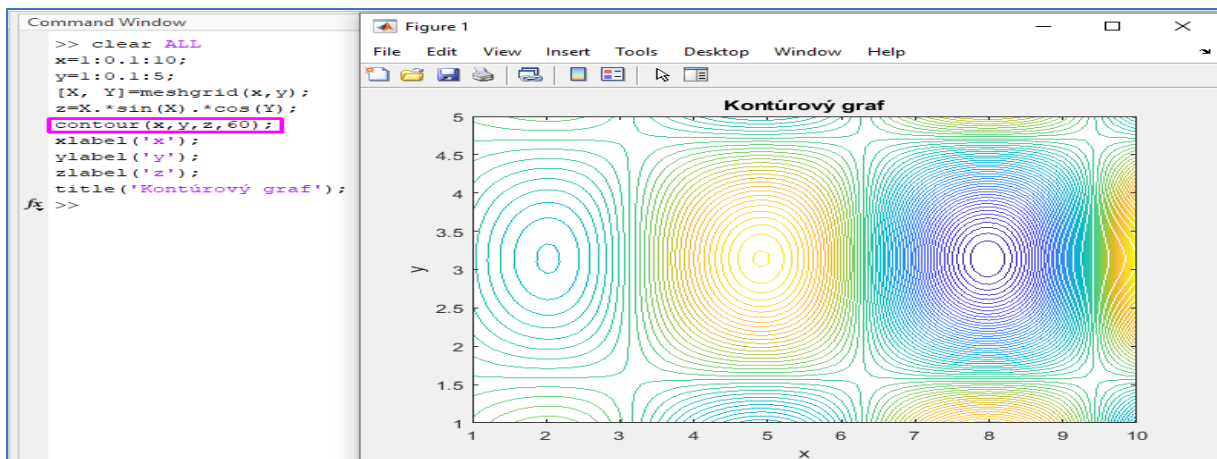


Obr. 8.23. Zobrazenie kontúrového grafu na príklade podľa Learning Vibes (2021).

Ak potrebujeme zobraziť presne definovaný počet kontúr (môžeme ich zadefinovať za príkaz, napr. ak ich potrebujeme konkrétne 3 zapíšeme: `contour(x,y,z,3)`, vid' [Obr. 8.24](#) alebo v prípade, ak ich potrebujeme 60 zapíšeme: `contour(x,y,z,60)`, vid' [Obr. 8.25](#).

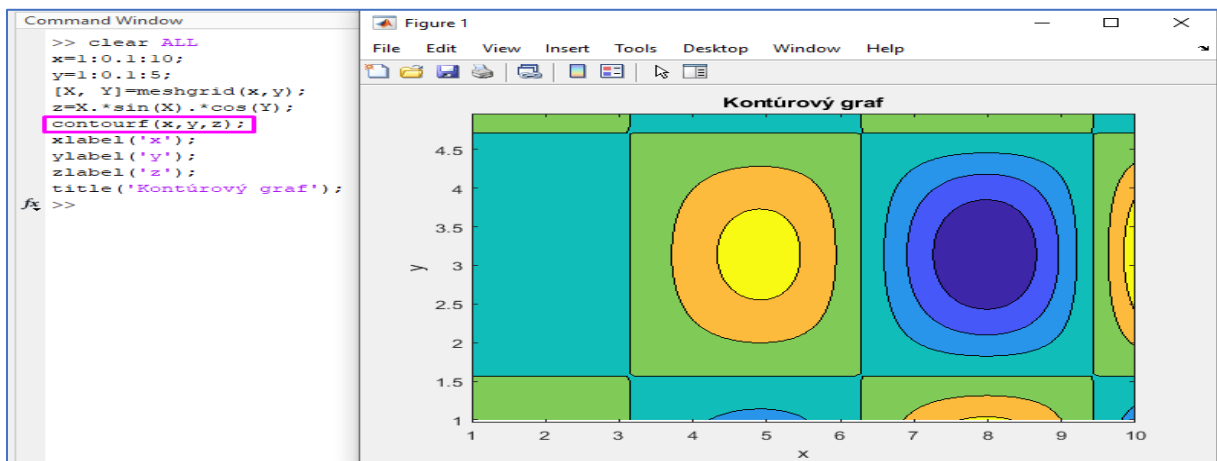


Obr. 8.24. Zobrazenie kontúrového grafu s 3 líniami (Learning Vibes, 2021).



Obr. 8.25. Zobrazenie kontúrového grafu so 60 líniami (Learning Vibes, 2021).

Použitím príkazu *contourf* (Obr. 8.26) MATLAB vyplní miesta medzi čiarami farbami. Farby reprezentujú rozličné amplitúdy osi z.

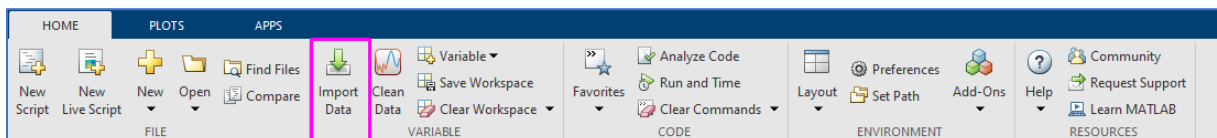


Obr. 8.26. Zobrazenie kontúrového grafu so zafarbenými priestormi medzi jednotlivými líniami pomocou príkazu *contourf*.

9. Importovanie údajov z programu Excel do MATLAB

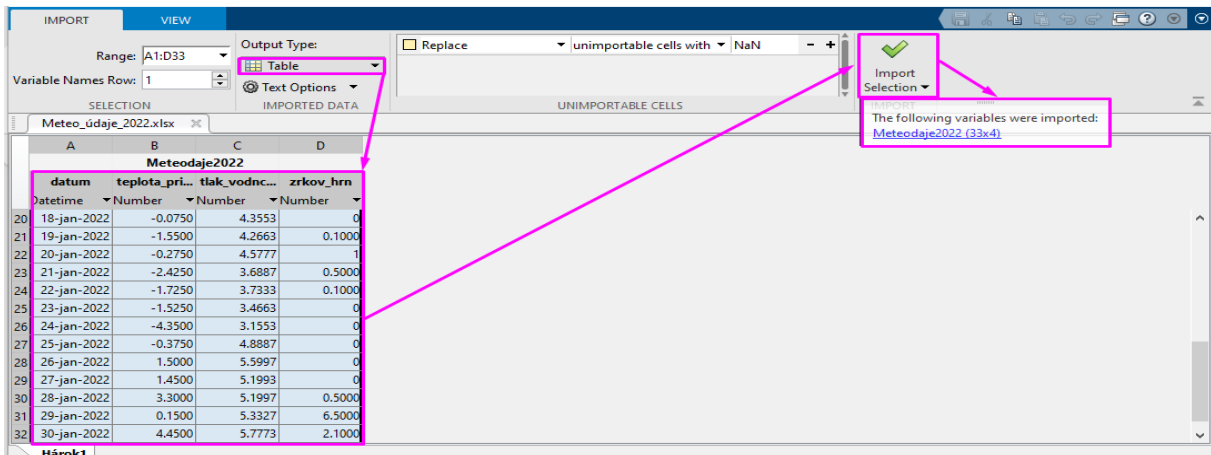
Pomocou ikony *import data* je možné vkladať do MATLABu rôzne formáty dát, vid'. Obr. 9.1.

1) Najjednoduchší spôsob, ako importovať súbory z Excelu do MATLABu pomocou ikony export, ktorá sa nachádza na lište pod Home menu, vid'. Obr. 9.1.



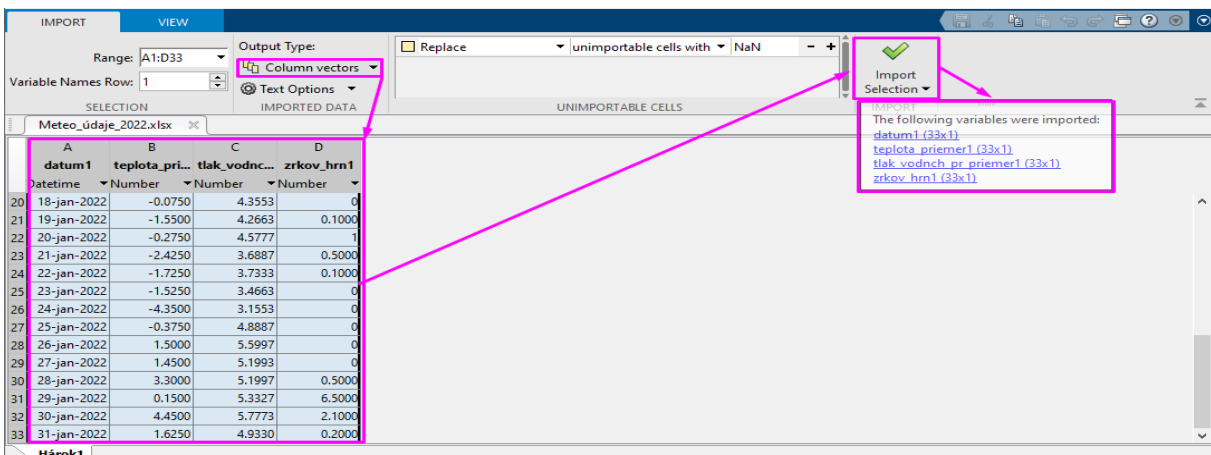
Obr. 9.1. Ikona zobrazujúca možnosti importovania dát do MATLAB.

Po vložení súboru sa otvorí okno podobné prostrediu programu Excel. Vyberieme si údaje, ktoré potrebujeme zobraziť a klikneme na ikonu *Import Selection*. Údaje sa automaticky uložia s pôvodným názvom (Obr. 9.2).



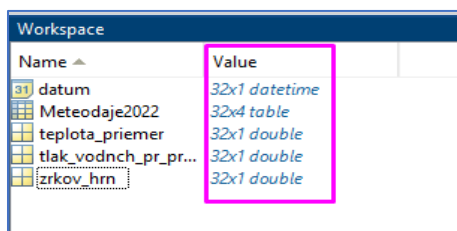
Obr. 9.2. Príklad importovania vybraných údajov z Excelu do MATLAB

Pri výbere iba niektorých častí tabuľky, vyznačíme možnosť „Column vectors“ (vektory stĺpcov), každý zo stĺpcov sa uloží ako samostatný vektor, vid'. [Obr. 9.3.](#)

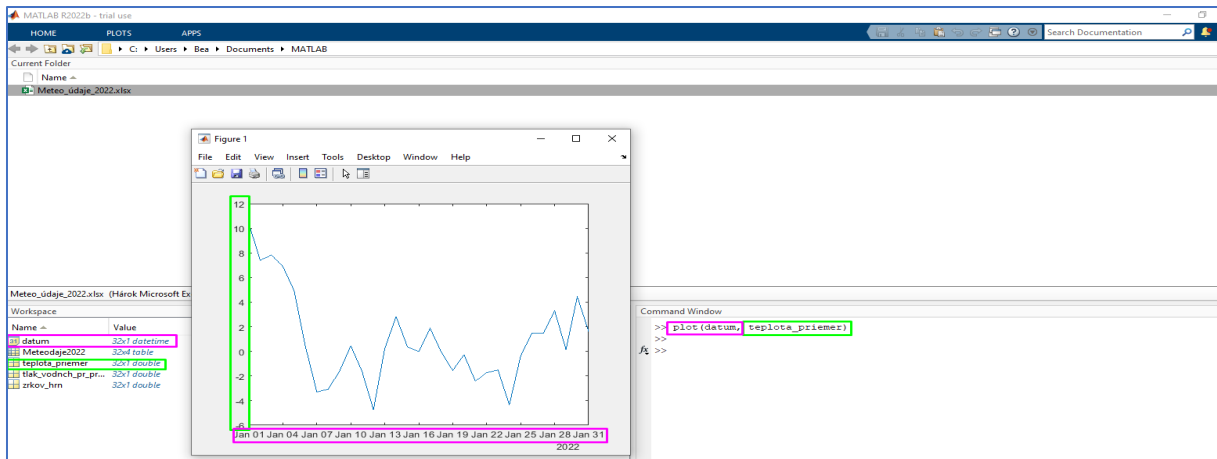


Obr. 9.3. Zobrazenie možnosti výberu z časti dát Excel súboru.

Údaje sú uložené v odlišných formátoch, vid'. [Obr. 9.4.a\)](#) a [b\)](#), t. j. v nasledovných formátoch: celá tabuľka – *table*, dátum – *datetime* a vektory ako napr. priemerná teplota – *double*.

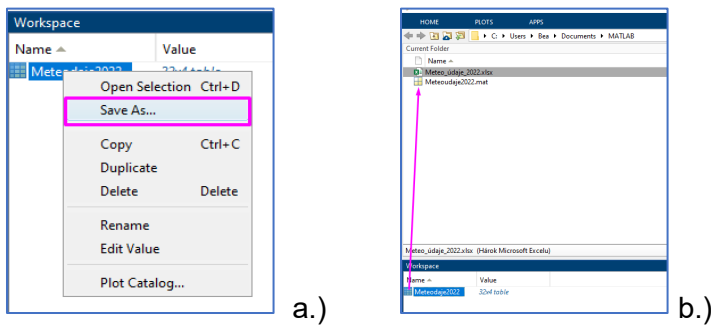


Obr. 9.4.a). Možnosti zobrazenia rôznych vložených údajov v MATLAB: celá tabuľka – *table*, dátum – *datetime*, vektor – *double*.



Obr. 9.4.b). Príklad možnosti zobrazenia grafu na základe vložených údajov zo súboru Excel. – *datetime*, vektor – *double*.

II) Ďalší spôsob importovania údajov z Excel súborov je, ak po importovaní údajov z programu Excel pomocou ikony „*Import Selection*“, uloží príslušné údaje vo formáte: *MAT-files (*.mat)*, vid'. [Obr. 9.5\) a b.\)](#).

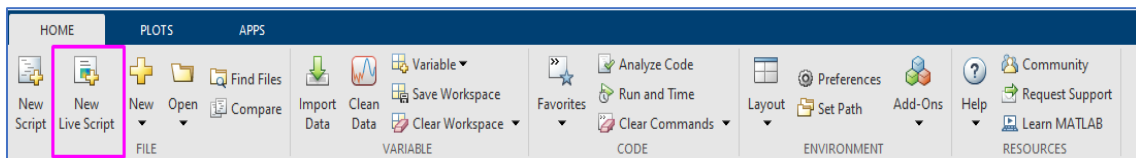


Obr. 9.5. Príklad uloženia údajov z Excel do MATLAB pomocou: a) výberu skupiny dát a b) znázornenie ich uloženia.

Takto vytvorené súbory môžeme následne načítať pomocou príkazu „*load*.“

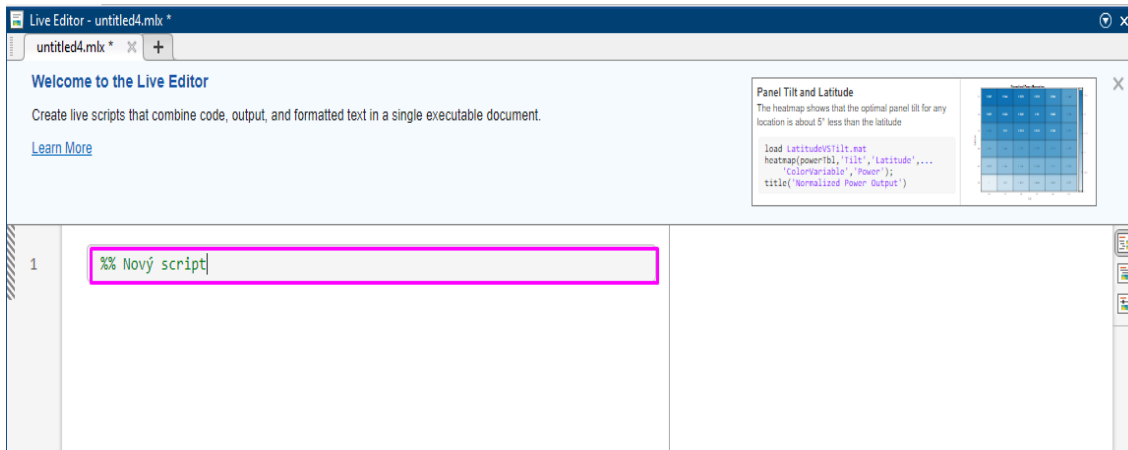
10. Vytvorenie Live Script (živý skript)

Na vytvorenie nového live scriptu použijeme ikonu „*New Live Script*“ (nový živý skript), vid'. [Obr.10.1](#).



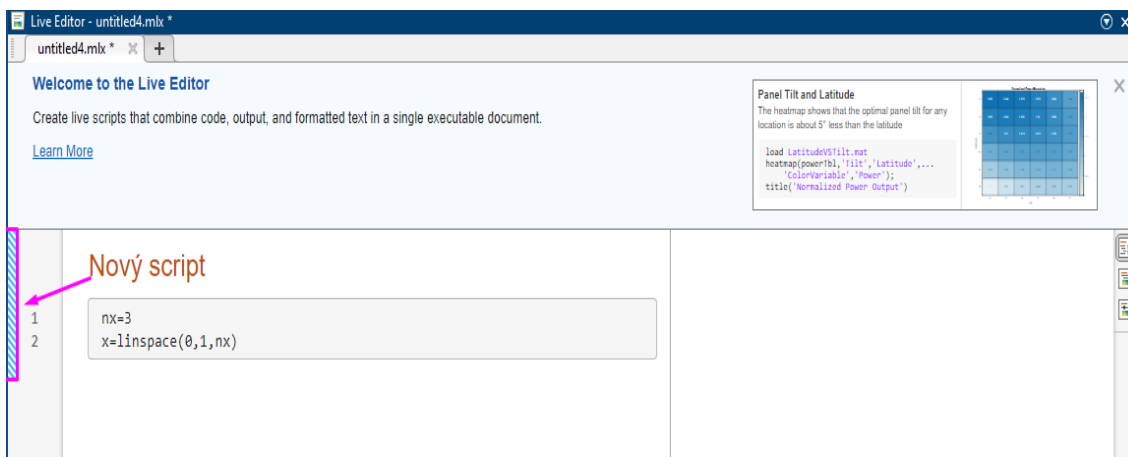
Obr. 10.1. Zobrazenie ikony na vytvorenie New Live Script (nového živého skriptu).

Po jeho spustení sa otvorí okno Live Editor, kde si názov vytvoríme ako „*%% Nový skript*“, vid'. [Obr.10.2](#).



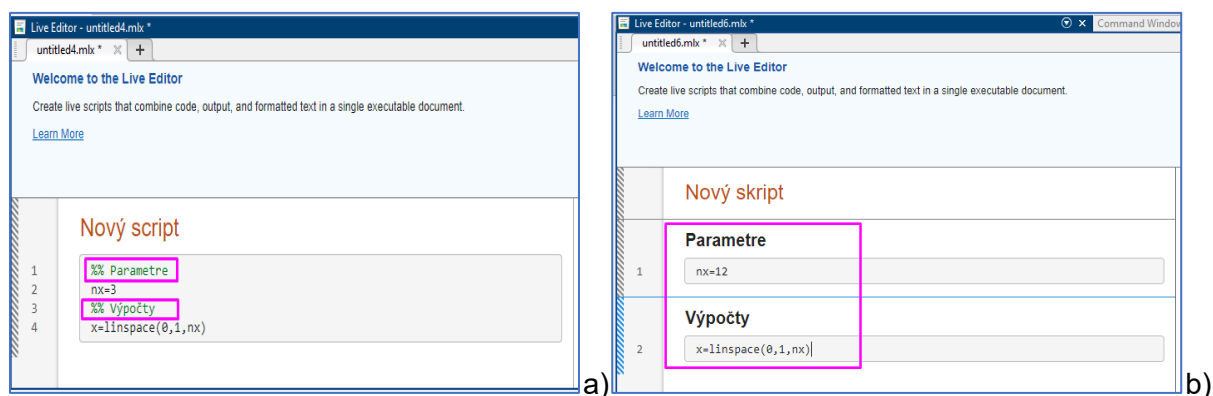
Obr. 10.2. Príklad vytvorenia nového skriptu.

Po zadefinovaní nového skriptu ho spustíme dvojklikom na modrú čiaru na jeho ľavej strane, vid'. [Obr. 10.3.](#)



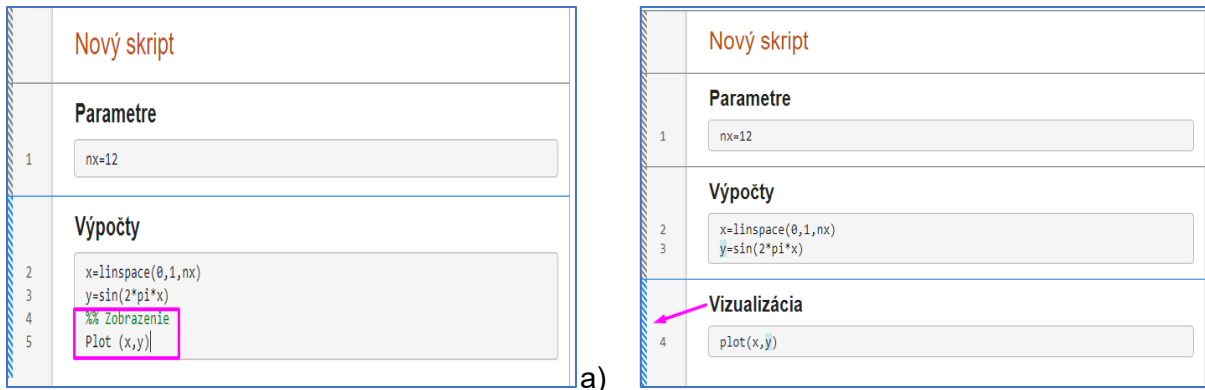
Obr. 10.3. Zobrazenie príkladu vytvorenia nového skriptu.

Do nového skriptu doplníme popis odseku opäť pomocou znakov „%%“, za ktorým pridáme ľubovoľný text-komentár, jeho zobrazenie potvrdím stlačením tlačidla „enter“, kedy sa zobrazia jednotlivé časti (napr. parametre a výpočty), vid'. [Obr. 10.4a\) a b\).](#)



Obr. 10.4. Zobrazenie príkladu: a) vytvorenia a b) zobrazenia nových odsekov v novom skripte.

Po doplnení ďalších premenných, si môžeme následne zobrazíť graf, pričom použijeme vyššie uvedené postupy ako napr.: [Obr. 10.5.a\) a b\).](#)



b) Obr. 10.5. Zobrazenie príkladu: a) vytvorenia a b) zobrazenia nových odsekov v novom skripte.

Jednu z možností zostavenia interaktívnej aplikácie zobrazuje nasledovný link: <https://ch.mathworks.com/learn/training/building-interactive-applications-in-matlab.html>

11. Otázky k pochopeniu preberaného učiva

Vyskúšame si svoje znalosti z doteraz preberaného a pochopeného učiva v teste, ktorý je dostupný na nasledovnom a) QR kóde alebo b) na doleuvedenom internetovom prepojení.

Upozornenie: spomínaný test na overenie znalostí z preberaného učiva je dostupný iba pre prihlásených užívateľov s platným univerzitným e-mailovým kontom Slovenskej poľnohospodárskej univerzity v Nitre.

a)




b) Link, na ktorom si otestujete svoje znalosti je nasledovný: <https://forms.office.com/e/WiWSVQKxnL>

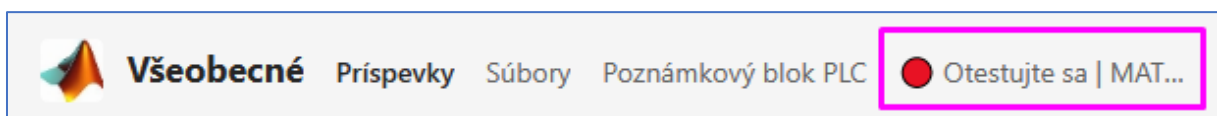
Test je rovnako prístupný aj po prihlásení do príslušného MS Tím, kde sa nachádzajú tiež aktualizácie ohľadom MATLAB a iné zaujímavé informácie a skripty, ktoré sú a budú priebežne dopĺňané. Na prihlásenie do MS Tím použite a) doleuvedený QR kód alebo b) internetové prepojenie.

a)



b) Link, na ktorom je prístupný a aktualizovaný MS Tím je: <https://lnk.sk/fheu>

Test na overenie znalostí z preberaného učiva je dostupný po prihlásení sa do príslušného MS Tímu, ak klikneme na tlačidlo  Otestujte sa | MATLAB.



Za správne zodpovedané otázky v teste môžete získať celkovo 20 bodov.

12. Technická podpora k MATLABu

Na svojich web stránkach firma Humusoft.cz ponúka v časti blog aktuality a technické články, napr.: [Tvorbá algoritmov s Live Editor Tasks](#). Výhodou je, že stránka je v slovenskom jazyku. Pokiaľ si chceme príklad vyskúšať, môžeme skopírovať dáta z Dema verzie, z uvedeného blogu, alebo z Live Skriptu, ktorý pripravila Heather Gorr a je dostupný na stránke File Exchange (Heather Gorr, 2022).

Firma Humusoft ponúka na svojich stránkach (<https://www.humusoft.cz/events/www-seminars/>) on-line kurzy v českom a slovenskom jazyku, ktoré sú zamerané na (klikni na online prepojenie):

- [Úvod do prostredia MATLAB](#) (odporúčané pre začiatočníkov);
- [Úvod do programu Simulink](#);
- [Úvod do modelovania udalostných systémov – Stateflow](#);
- [MATLAB Live Editor](#) (odporúčané pre začiatočníkov);
- [Analýza dát: Prechádzame z Excelu k prostrediu MATLAB](#) (odporúčané pre začiatočníkov).

Uvedená stránka tiež ponúka online semináre na internete, ktoré sa konajú v reálne určenom čase. Tieto online semináre prebiehajú v anglickom jazyku. V priebehu online seminára je možné na vopred danú tému klásť prednášajúcemu otázky (ak je to umožnené telefonicky alebo písanou formou), ale tiež získavať postrehy od ďalších účastníkov. Všetky uskutočnené online semináre sú archivované a preto je možné si ich aj s oneskorením prezrieť v archíve

uskutočnených online seminárov, samozrejme už bez možnosti aktívnej účasti (Humusoft.cz, 2022).

Rovnako na stránke (Mathworks.com, 2022) sú dostupné nasledovné možnosti (klikni na on-line prepojenia):

- 3 [Upcoming live events](#) (nadchádzajúce živé podujatia);
- 4 [On-Demands Webinars](#) (webináre na požiadanie);
- 5 [Conferences](#) (konferencie);
- 6 [Training](#) (školenia).

V sekcii On-Demand Webinars and Videos (Webináre a videá na požiadanie) je prístup k záznamom už uskutočnených videí a školení, ktoré sú dostupné na:

https://ch.mathworks.com/videos/search.html?s_tid=gn_ev_rw&q=&page=1

Na stránke (Mathworks.com, 2022) sú tiež dostupné Self-Paced Online Courses (Online kurzy s vlastným tempom), viď.: <https://matlabacademy.mathworks.com/>

Užitočné praktické informácie sú dostupné na nasledujúcich internetových prepojeniach:

[Tables](#)

[Timetables](#)

[Resample and Aggregate Data in Timetable](#)

[Using the Live Editor](#)

13. Ťahák pre MATLAB

Uvedené pokyny pre MATAB je vlastne ťahák zostavený podľa Papeža (2014):

Začíname (help general)

help – nápoveda, napr. pre konkrétnu funkciu: *help sin*

doc – dokumentácia, napr. pre konkrétnu funkciu *doc sin*

lookfor – vyhľadávanie funkcie, napr. pre konkrétnu funkciu *lookfor cosine*

clc – vymazanie Command Window

Premenné

MATLAB rozlišuje veľkosť znakov. Názov premennej musí začínať písmenom. Základnou premennou je dvojrozmerné pole reálnych čísel (64 bitov).

Skalár je pole 1×1 .

Riadkový vektor dĺžky n je pole $1 \times n$.

Stĺpcový vektor dĺžky m je pole $m \times 1$.

Matice s m riadkami a n stĺpmi je pole $m \times n$.

Reťazec obsahuje text. (*help strfun*)

clear a – zmazanie premennej *a*

clear – zmazanie všetkých premenných

Vytvorenie matice/vektora

Prvky na riadku sú oddelené medzerou alebo čiarkou, bodkočiarka ukončuje riadok

$v = [1\ 2\ 3\ 4\ 5]$ riadkový vektor

$w = [5; 2; 3; 4]$ stĺpcový vektor

$A = [1,2,3; 4,5,6; 7,8,9]$ matice

Alebo tiež takto:

$A = [1,2,3$

$4,5,6$

$7,8,9]$

length – dĺžka vektora alebo najväčší rozmer poľa

size – rozmery poľa $m, n] = \text{size}(A)$

Vytvorenie špeciálnych matic/vektorov

(help elmat)

linspace(a,b,N) – pravidelné delenie intervalu $[a, b]$ obsahujúce N bodov (vrátane a, b)

$i:j:k$ – aritmetická postupnosť, prvý prvok i , diferenciacia j , horná medza k (resp. dolná medza pre záporné j)

$i:k$ – aritmetická postupnosť s diferenciou 1

zeros(m,n) pole nul $m \times n$

zeros(n) pole nul $n \times n$

ones(m,n) pole jednotiek $m \times n$

ones(n) pole jednotiek $n \times n$

eye(m,n) pole nul $m \times n$ s jednotkami na diagonále

eye(n) jednotková matica $n \times n$

rand(m,n) pole $m \times n$ náhodných čísel

rand(n) pole $n \times n$ náhodných čísel

Indexovanie premenných

Číslovanie začína od jednotky!

$v(1)$ – prvý prvok vektora v

$v(\text{end})$ – posledný prvok vektoru v

$A(2,3)$ – prvok matice A v druhom riadku, treťom stĺpci

$A(:,3)$ – tretí stĺpec matice A

$A(1,:)$ – prvý riadok matice A

$A(1:2:\text{end},:)$ – matica obsahujúca nepárny riadok matice A

$A(1:2,2:4)$ – podmatica A obsahujúca 1.– 2. riadok, 2.– 4. stĺpec

$A(1,\text{end})$ – posledný prvok prvého riadku matice A

Operácie s maticami/vektormi (help arith,help ops)

+ – sčítanie

- – odčítanie

* – násobenie

. * – násobenie po zložkách

/ – delenie

./ – delenie po zložkách

^ – umocnenie

.^ – umocnenie jednotlivých zložiek

' – komplexné združenie

.' – transpozícia

Matematické funkcie (help elfun, help matfun)

Nasledujúce funkcie majú očakávaný význam: abs, exp, log, log10, log2, sqrt, sin, asin, cos, acos, tan, atan, floor, ceil, round, max, min, norm, rank, det, inv, sort, sum.

Výstupom týchto funkcií môžu byť vektory i matice.

Skripty ("m-files")

Postupnosť príkazov, tzv. skript je uložený v textovom súbore s príponou .m. Skript je možné spustiť zadaním mena súboru (bez prípony) v príkazovom riadku alebo tlačítkom „Run“ v okne editora.

;- – potlačenie výstupu, umiestňuje sa na koniec príkazu

% – začiatok komentára (celý ďalší riadok)

... – pokračovanie príkazu/ výrazu na ďalšom riadku

Funkcie

Funkcie sú definované v jednotlivých súboroch pomenovaných rovnako ako funkcia

NazovFunkcie.m nasledovne:

```
function [out1,...,outN] = NazevFunkce(in1,...inM)
```

```
% NazevFunkce: Stručný popis (voliteľné)
```

```
% ...
```

```
príkazy;
```

Funkciu potom spustíme takto:

```
[vystup1,...,vystupN] = NazovFunkce(vstup1,...vstupM)
```

Logické operátory (help relop)

< – menší než

<= – menší alebo rovný

> – väčší než

>= – väčší alebo rovný

== – rovný

~= – rôzny

& – logické AND (“a zároveň”)

| – logické OR (“alebo”)

~ – negácia

Cykly (help lang)

```
for k = Vektor
```

```
    príkazy;
```

```
end
```

Obvykle: for k = 1:n

```
while LogickyVyraz
```

```
    príkazy;
```

```
end
```

Pro náročnejšie výpočty je žiadúce vyhýbať sa cyklom.

Vetvenie kódu (help lang)

```
if LogickýVyraz
```

```
    príkazy;
```

```
elseif LogickýVyraz % Voliteľné
```

```
    príkazy;
```

```
else % Voliteľné
```

```
    príkazy;
```

```
end
```

Vykresľovanie výsledkov (help graph2d, help raph3d)

figure – vytvorenie nového okna pre graf

close – zatvorenie aktuálneho okna

close all – zatvorenie všetkých okien

plot – základný 2D graf

semilogy – 2D graf s logaritmickejšou škálovanou osou y

imagesc – zobrazenie obrázku, vhodné pre zobrazenie prvkov matice

plot3 – zobrazenie krivky/bodov v 3D

surf – zobrazenie plochy v 3D

legend – legenda ku grafu

title – názov grafu

xlabel – popis osi x

ylabel – popis osi y

14. Možnosti využitia MATLAB v rámci krajinného inžinierstva

Oblasť krajinného inžinierstva v sebe zahŕňa problematiku spojenú so životným prostredím od monitoringu cez mapovanie, analýzu, návrh až po participáciu pri realizácii návrhov. Okrem iného v sebe zahŕňa aj prácu so softvériami umožňujúcimi modelovanie procesov v krajine od jednoduchých ako je výpočet kapacity koryta vodného toku pri rôznych geometrických parametroch cez modelovanie šírenia znečistenia toku až po analýzu spektrálnych satelitných snímok. Ako sme už naznačili, použitie MATLABu je možné vo vodnom hospodárstve, odpadovom hospodárstve, krajinnom a územnom plánovaní či diaľkovom prieskume Zeme. Možné je využiť priame zadávanie kódu alebo vytvorených aplikácií priamo pre MATLAB. Nižšie uvádzame stručný prehľad vybraných oblastí využitia MATLABu v rámci krajinného inžinierstva. Vzhľadom na rozsiahlosť možností aplikácie softvéru a jeho aplikácií a nastavieb uvedieme len vybranú malú časť využitia softvéru. S rozvojom spoločnosti a nárokov na riešenie existujúcich a novovznikajúcich problémov ani nie je možné vytvoriť ucelený prehľad nástrojov, keďže takmer denne pribúdajú nové možnosti.

Udalosti a systémy sú definované pomocou určitých matematických modelov v inžinierstve a v iných oblastiach vedy. Pomocou týchto skonštruovaných modelov sa pokúšame odhadnúť budúci stav alebo vzorec správania udalosti alebo systému. Veľká väčšina problémov, s ktorými sa stretávame v každodennom živote, sa však buď nedá presne modelovať, alebo tieto spomenuté problémy nemusia z viacerých dôvodov predstavovať určitý stav alebo podmienky. Na skúmanie a riešenie takýchto problémov je možné implementovať prístup fuzzy logiky. Teória fuzzy logiky priťahuje čoraz väčšiu pozornosť v dôsledku skutočnosti, že fuzzy riadiace systémy poskytujú úspešné výsledky v mnohých priemyselných oblastiach. Dôvodom je, že fuzzy regulátory reprezentujú ľudské myslenie a príkazy lepšie ako klasické regulátory pomocou fuzzy pravidiel. Dnes, podobne ako v iných oblastiach inžinierstva, sa fuzzy logika používa aj na riešenie veľkého množstva geodetických problémov.

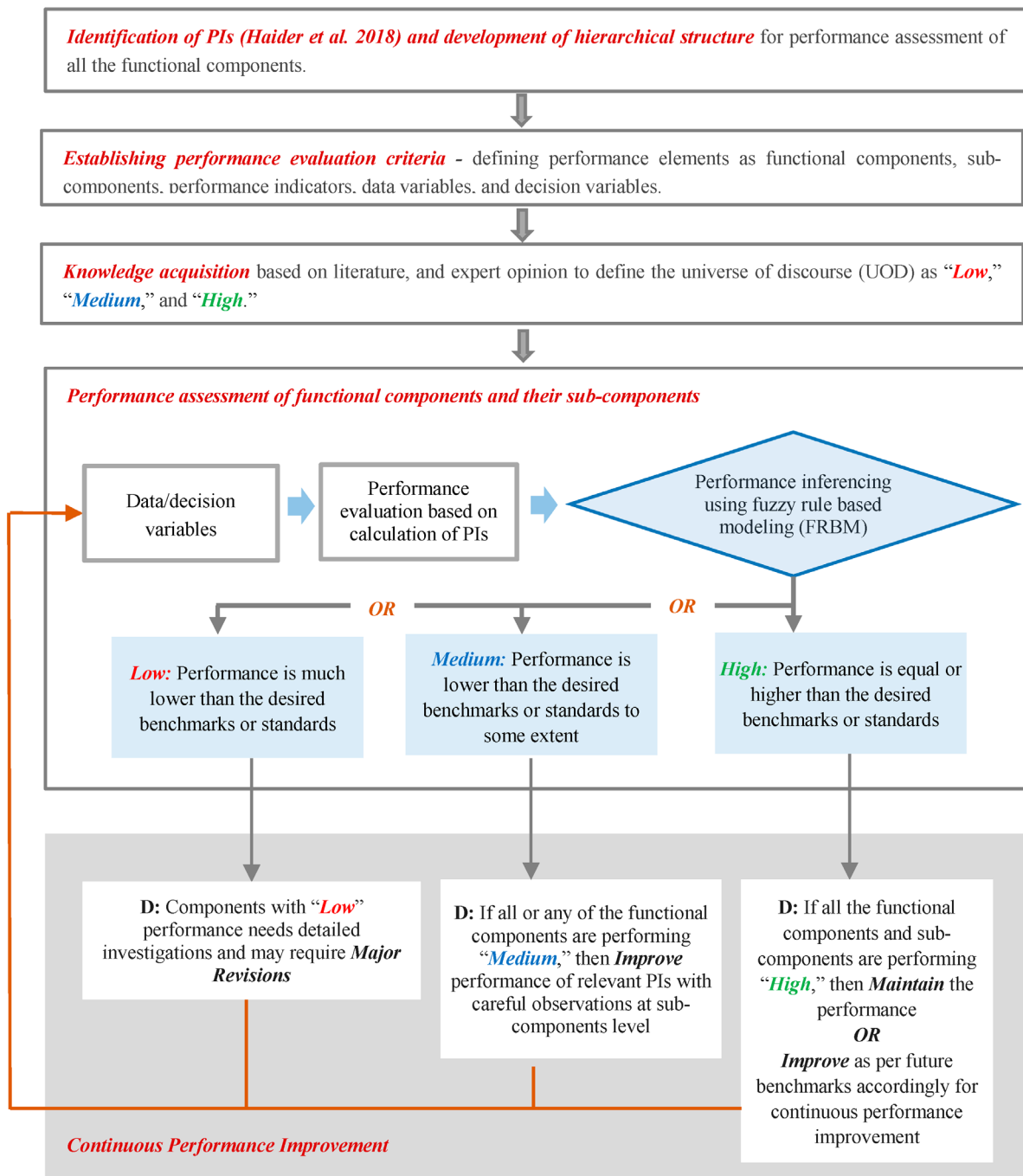
14.1 Odpadové hospodárstvo

V oblasti odpadového hospodárstva je pozornosť venovaná hospodáreniu s tuhým odpadom v obciach a mestách ako aj odpadovým vodám.

Väčšina obcí na svete čelí problémom súvisiacim s výkonom nakladania s tuhým komunálnym odpadom (Obr. 14.1). Chýbajú im však cielené medziobecné benchmarkingové procesy. Namiesto identifikácie výkonnostných medzier pre ich kľúčové zložky (napr. personálna produktivita, prevádzková spoľahlivosť atď.) a prijímania proaktívnych opatrení sa samosprávy spoliehajú predovšetkým na efektívnu reakciu na núdzové situácie. Na hodnotenie výkonnosti systémov nakladania s tuhým komunálnym odpadom bol vyvinutý nový hierarchický modelovací rámec založený na deduktívnom uvažovaní. Modelovanie založené na fuzzy pravidlách pomocou Simulink-MATLAB sa použilo na odvodzovanie výkonu na rôznych úrovniach, t. j. komponent, podkomponenty atď. Model je schopný zvládnuť inherentné neistoty kvôli obmedzeným údajom a nepresnej znalostnej báze. Výsledky modelu môžu výlučne pomôcť manažérom pracujúcim na rôznych úrovniach organizačnej hierarchie pri efektívnom rozhodovaní. Jednotlivé ukazovatele sú zároveň užitočné na porovnávanie medzi obcami (Abdulaziz AlHumid et al., 2019).

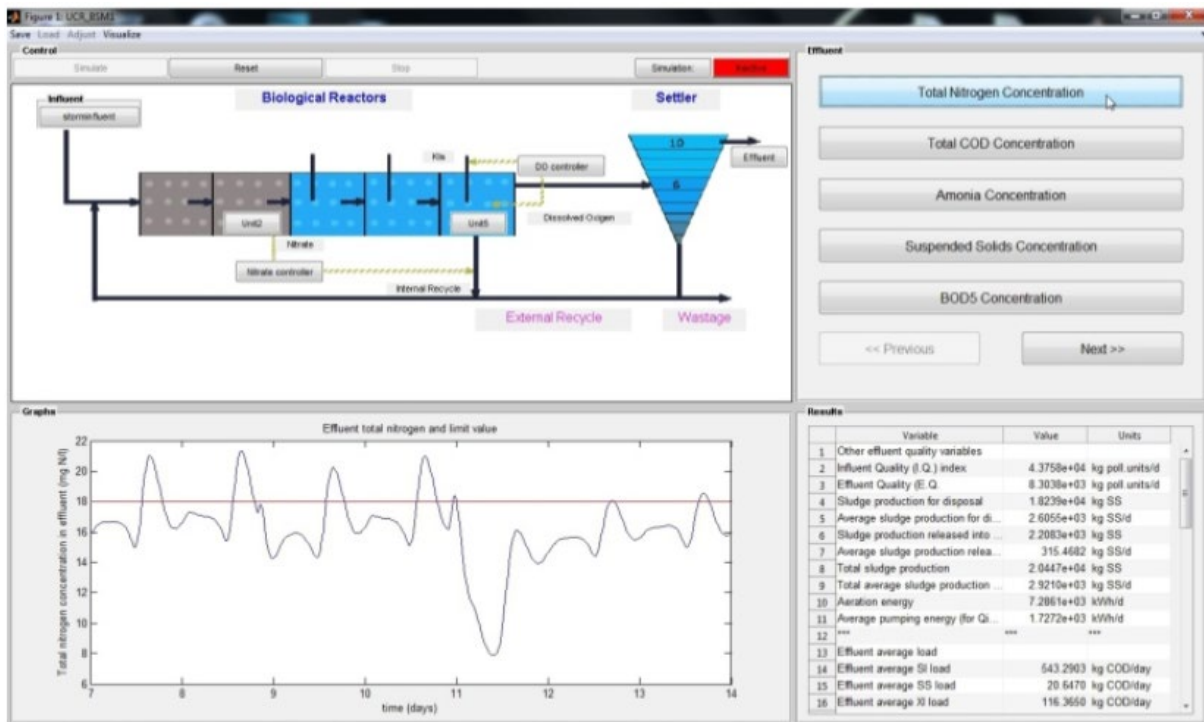
Odpad je na jednej strane faktor nežiaduci, na druhej strane môže významne prispieť k národnému dôchodku, ktorý má potenciál stať sa skvelým zdrojom príjmu, ak sa využije prostredníctvom recyklácie a opätovného použitia. Bolo vytvorené softvérové riešenie venujúce sa odpadu z domácností, ktoré bolo navrhnuté pomocou MATLAB-GUI na použitie

pri účtovaní výnosov alebo strát vzniknutých pri správe odpadového hospodárstva, najmä na úrovni obcí. Vzhľadom na deväť modulov s deviatimi parametrami ako jeden pre každý modul, formulovaný softvér pomáha pri výpočte výnosov generovaných z recyklovaného a opätovne použitého odpadu v okamihu. Softvér tiež poskytne riešenie na preplánovanie procesu opätovného využitia odpadu, ak sa zistí nejaký nesúlad medzi sumou rozpočtu a skutočne vynaloženými nákladmi. Ťažkopádne manažérske problémy, ktoré doteraz zahŕňali rôzne administratívne procesy odpadového hospodárstva, nachádzajú jednoduché elektronické riešenie v navrhnutom softvéri (Renuka & Radhakumari, 2016).



Obr. 14.1. Konceptný model pre hodnotenie výkonnosti systémov nakladania s tuhým komunálnym odpadom (Abdulaziz AlHumid et al., 2019)

Ďalšou z alternatív využitia MATLAB je oblasť odpadových vôd. Jednou z možností je využitie pri modelovaní čistenia odpadových vôd v čistiarni odpadových vôd. Simulátor BSMI (Obr. 14.2) umožňuje modifikovať nastavenia rôznych komponentov scenára, čo potom umožní analyzovať dôsledky týchto zmien a poučiť sa z nich. Takto vytvorená simulácia umožňuje získať podrobné štatistické údaje, ktoré je možné použiť na skúmanie konkrétneho správania časti procesu (Urena et al., 2011).



Obr. 14.2. Hlavné okno simulátora BSMI (Urena et al., 2011)

Nový softvér bol vytvorený na správu tuhého odpadu a odpadovej vody (SWW) na optimalizáciu životného cyklu emisií s ohľadom na náklady na uhlíkový kredit, čo je v súčasnosti výrazným problémom v celom svete. Tento softvér je prvým, ktorý kombinuje integrované systémy nakladania s pevným odpadom a odpadovými vodami do jedného rámca pri zavádzaní politiky drviča potravinového odpadu. Model/softvér ponúka platformu zahŕňajúcu niekoľko nástrojov na účtovanie emisií počas životného cyklu a optimalizáciu procesov. Poskytuje flexibilitu výberu procesov alebo úpravy vstupných parametrov, ako aj dezagregácie emisií v závislosti od rozsahu účtovania (Obr. 14.3). Grafické používateľské rozhranie je použiteľné v kontexte rozvinutých a rozvojových ekonomík s konečným cieľom pomôcť rozhodovateľom pri pridelovaní výdavkov na opatrenia na zníženie emisií (Maalouf & El-Fadel, 2020).



Obr. 14.3. Úvodná obrazovka používateľského rozhrania SWW, 1: Zadávanie údajov; 2: Nástroj na optimalizáciu; 3: Nástroj na účtovanie emisií; 4: Zobrazenie výsledkov; 5: Nástroj ekonomickej analýzy; 6: Nástroj na analýzu citlivosti; 7: Nástroj analýzy politiky (Maalouf & El-Fadel, 2020).

14.2 Vodné hospodárstvo

Oblasť vodného hospodárstva je v programe MATLAB najviac využívaná, a teda sú k dispozícii rôzne nástroje a moduly pre simulácie prúdenia vody v korytách, potrubíach alebo nádržiach.

Pokiaľ ide o problémy v oblasti vodných zdrojov a environmentálneho inžinierstva, aplikácia modelov založených na údajoch môže zahŕňať:

- simuláciu a predpoveď kvality vody,
- predikciu extrémnych hodnôt s dôrazom na povodne a suchá,
- modelovanie vodnej bilancie týkajúcej sa rôznych zložiek hydrologického systému,
- predĺženie dĺžky hydroklimatologických údajov z historických dát,
- odhadovanie cenzurovaných údajov (Araghinejad, 2014).

14.2.1 Prúdenie v otvorených kanáloch

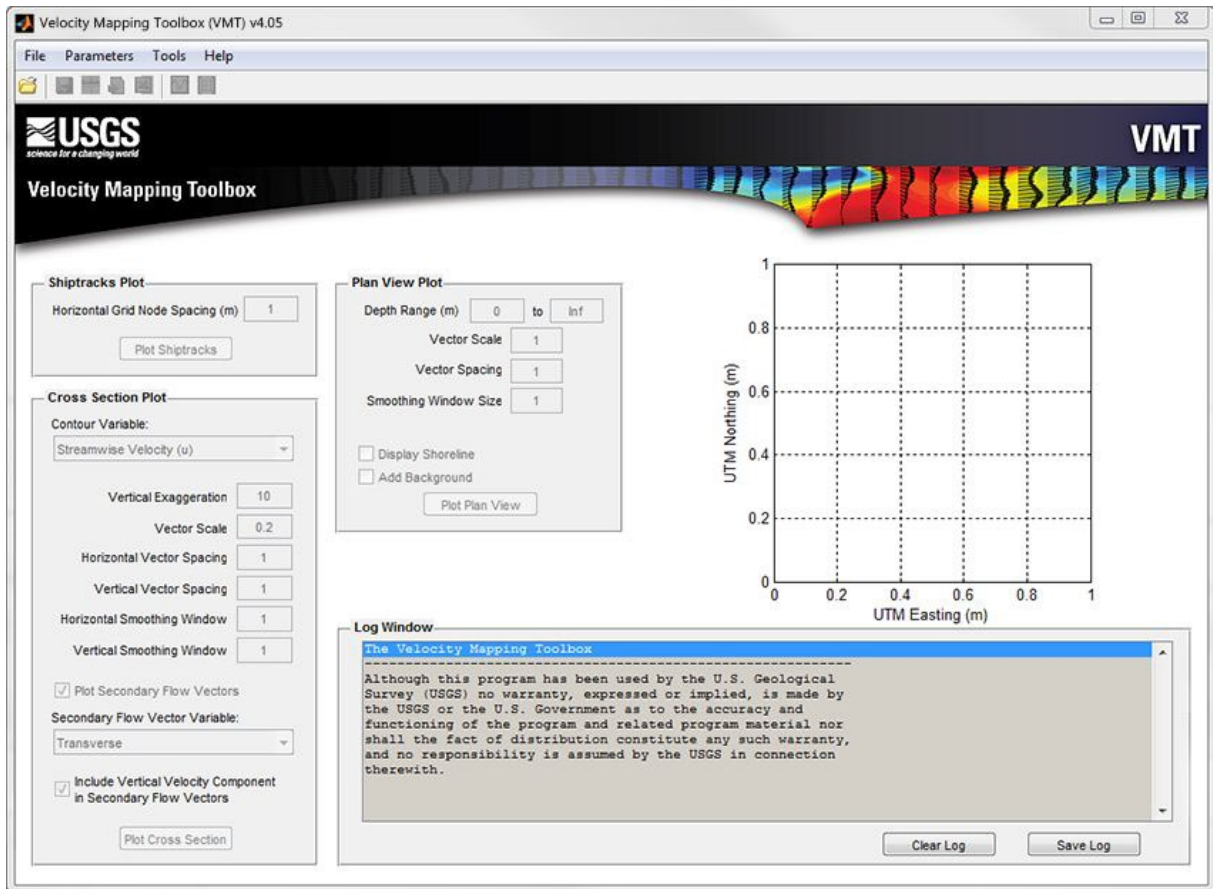
Kritická hĺbka a normálna hĺbka sú základným parametrom pri analýze rôzneho prietoku v otvorených kanáloch. Rádiace rovnice pre kritickú hĺbku sú implicitné a neexistujú žiadne analytické riešenia. V lichobežníkovom kanáli sú rádiace rovnice vysoko nelineárne v normálnych a kritických hĺbkach prúdenia, a preto riešenie implicitných rovníc zahŕňa numerické metódy. Toľko riešení už existuje vo forme empirických vzťahov, avšak ich riešenie je zdĺhavé a pravdepodobnosť výskytu chyby je väčšia. Využitie programu MATLAB získalo dôležitosť pre tieto typy výpočtových problémov a umožňujú jednoduchšie, rýchlejšie a presnejšie výpočty (Tiwari et al., 2012).

V literatúre bolo pozorované, že pre podmienky nestabilného prúdenia nemusí byť vzťah jedna ku jednej medzi hĺbkou prúdenia, priemernou rýchlosťou prierezu a odporom trenia, ako je určený z prípadov ustáleného rovnomerného prúdenia, vhodný pre tieto zložitejšie systémy prúdenia. Preto je potrebné upraviť všeobecný vzorec odporu trenia pridaním nových popisných výrazov, aby sa zohľadnila nestabilita toku a aby sa eliminovali chyby spôsobené predpokladmi rovnomerného a ustáleného toku. Rozšírený Chezyho vzorec zahŕňajúci časové aj priestorové parciálne derivácie hydraulických parametrov bol vyvinutý pomocou rozmerovej analýzy na skúmanie vzťahu medzi nestálosťou toku a odporom trenia. Výsledky ukazujú, že navrhovaný vzorec funguje lepšie ako tradičný Chezyho vzorec na simuláciu skutočných prípadov hydrografu, pričom oba koeficienty vzorca sú individuálne identifikované pre každú povodňovú udalosť a koeficienty sú vopred určené pomocou iných povodňových udalostí ako kalibračných prípadov. Hoci rozšírený vzorec Chezy, ako aj pôvodný vzorec Chezy fungujú horšie s narastajúcim stupňom nestability toku, jeho výsledky sú menej dramaticky ovplyvnené intenzitou nestability, čím sa zlepšujú odhady smerovania záplav. Výsledkom je, že pri ťažkých povodniach má tendenciu fungovať oveľa lepšie ako tradičný vzorec Chezy. V zložitejších podmienkach, pri ktorých sa vrcholné povodne môžu vyskytovať prevažne pri nestabilnom toku, môže rozšírený model Chezy poskytnúť výskumníkom, odborníkom z praxe a vodohospodárom cenný nástroj na hodnotenie a predpovedanie vplyvov záplav a na vývoj vhodnejších stratégií na zmiernenie presnejšie hodnotenia rizika (Zhou et al., 2021). Tieto výpočty je možné realizovať pomocou MATLAB kódu, ktorý je voľne dostupný na stiahnutie (<https://zenodo.org/record/4314728>; Zhou et al., 2020).

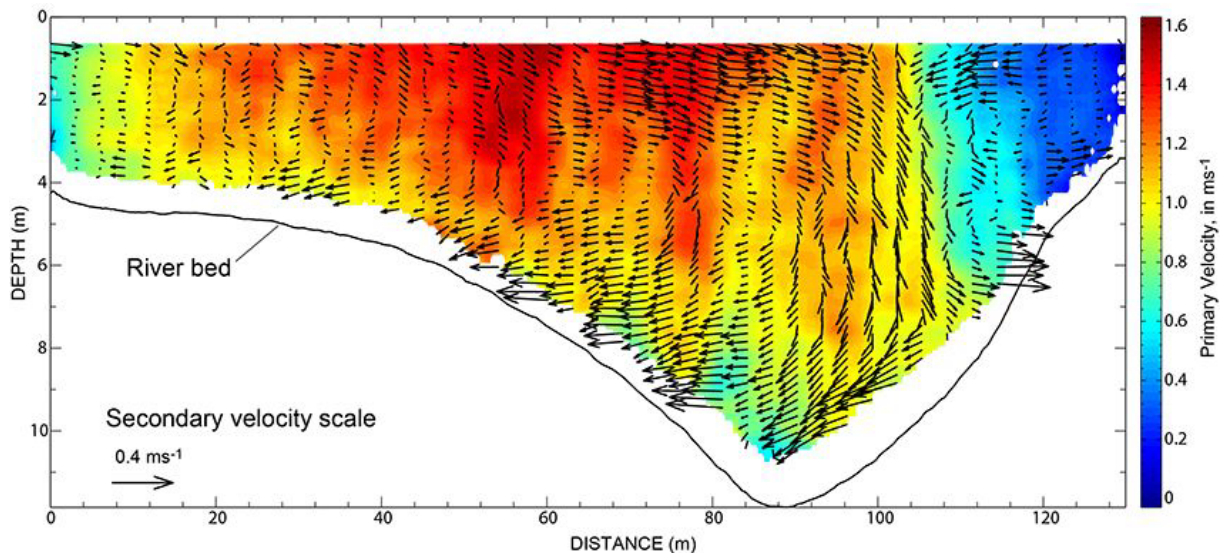
„Hydraulický odpor“ je výsledkom prúdenia v hlavnom kanáli a na brehu, ktorý vzniká tlakovým odporom a viskóznou silou. Predpovedanie hladiny vody a distribúcie prietoku je ním arbitrované v kanáli. Takýto odpor je všadeprítomne charakterizovaný určitými parametrami, ako je Manningov koeficient drsnosti (n), Chezyho koeficient odporu (c) a Darcy-Weisbachov koeficient trenia (f). Spoľahlivé výsledky simulácie záplav sa spoliehajú na presný odhad koeficientu odporu. Výsledok testu ukazuje, že so zvyšujúcou sa hĺbkou prúdenia sa koeficient drsnosti znižuje. Manningovo n sa široko používa pri výpočte prietoku v otvorenom kanáli. Hydraulický výpočet prietoku v oblasti hlavného koryta a záplavových území vyžaduje odhad koeficientov drsnosti. Napriek rozsiahlemu výskumu experimentálnych údajov sa analyzujú aj výpočtové výsledky, aby sa študovali variácie koeficientov drsnosti pre priamy aj meandrujúci kanál. Pre skúmanie konzistentného trendu kolísania drsnosti bol využitý kód vytvorený v MATLABe (Priyadarshinee & Mohanty, 2015).

Hydrológovia a ďalší vedci používajú akustické Dopplerove prúdové profily (ADCP) na meranie objemovej rýchlosti prietoku vody alebo prietoku v riekach a iných vodných cestách. Merania prietoku sa používajú na predpovedanie záplav a podmienok nízkej hladiny vody. Aj keď sú takéto merania cenné, údaje poskytnuté ADCP možno použiť aj na vytvorenie oveľa podrobnejšieho obrazu o distribúcii rýchlosti vody v rieke. Velocity Mapping Toolbox (VMT, Obr. 14.4), zabudovaný v MATLAB, umožňuje rýchlo spracovať všetky nespracované dáta zaznamenané ADCP. Výsledkom je lepšie pochopenie priestorového rozloženia toku a variácií časových tokov (Obr. 14.5), vďaka čomu je VMT cenným nástrojom pre rôzne aplikácie. Environmentálni vedci ho používajú na skúmanie distribúcie prúdenia okolo štruktúr v blízkosti biotopov voľne žijúcich živočíchov. Energetické spoločnosti ho používajú na identifikáciu najlepšej polohy pre prílivové turbíny. United States Geological Survey (USGS) použila VMT na rôzne projekty a štúdie. Použili ho napríklad na pomoc americkému armádnemu zboru inžinierov posúdiť distribúciu toku na sútoku riek Mississippi a Ohio, aby člny mohli bezpečne plávať po rozvodnených riekach počas rekordných záplav v roku 2011. Taktiež používajú VMT na štúdium meandrových ohybov v riekach a určiť, kde by mohla byť potrebná ochrana pobrežia, aby sa zabránilo erózii, vizualizovať tok v blízkych pobrežných a riečnych oblastiach Veľkých jazier, aby ste pochopili transport kontaminantov a pomohli pri štúdiách zdravia

na pláži a študovali schopnosť prítokov Veľkých jazier prenášať ikry ázijských kaprov do identifikovať vhodné biotopy neresenia pre invázne druhy (Fricker, 2014).



Obr. 14.4. Vstupné grafické rozhranie Velocity Mapping Toolbox (VMT) (Fricker, 2014).

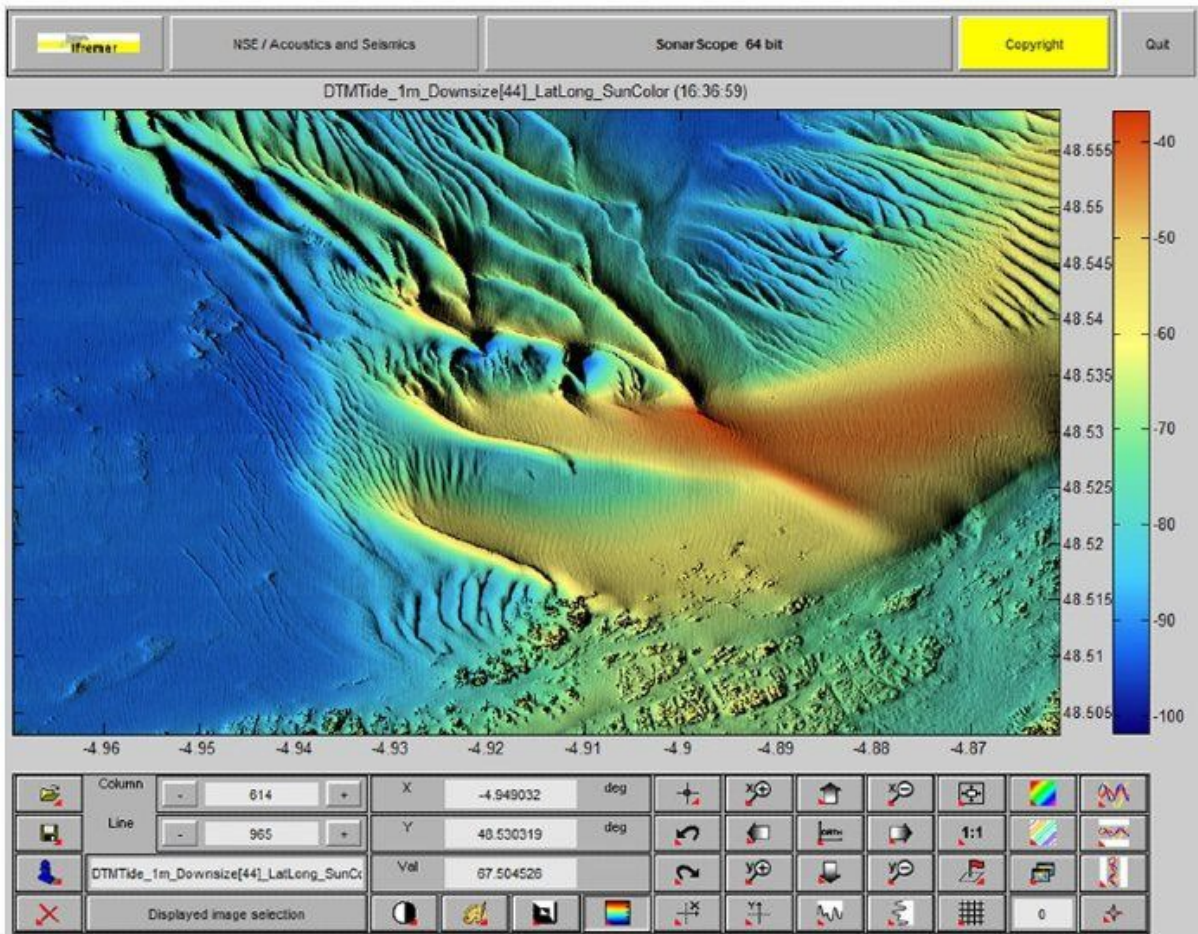


Obr. 14.5. MATLAB graf primárnych (prúdových) a sekundárnych (pričných) rýchlostí (Fricker, 2014).

14.2.2 Využitie a spracovanie údajov zo sonaru

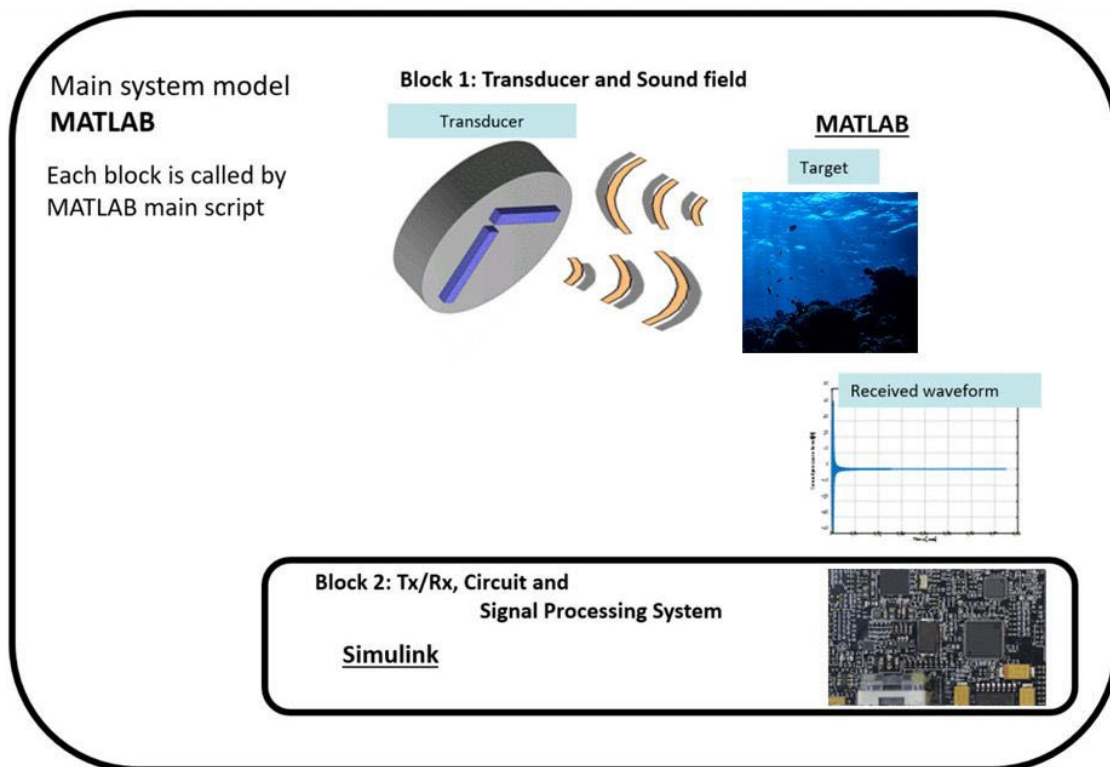
Zatiaľ čo satelitné fotografie a radar poskytujú snímky s vysokým rozlíšením prakticky každého štvorcového metra zemského povrchu, pohľady na dno oceánu sú škvŕnitejšie a menej podrobné. Presné mapy morského dna sú však nevyhnutné pre vedecký výskum a mnohé priemyselné aplikácie. Okrem záujmu o oceánografiu, geovedy a biológiu pomáha presná znalosť kontúr a zloženia morského dna energetickým spoločnostiam umiestňovať veterné farmy a vrtné plošiny, komunikačné spoločnosti plánujú, kam položiť optické káble, a odborníci na životné prostredie hodnotia účinky klimatickej zmeny v oceánoch a moriach. Pomerne nejasný pohľad na morské dno nie je spôsobený nedostatkom údajov. Lode vybavené viacúčtovými echozvučkami (MBES) a sonarovými systémami skúmajú široké oblasti mora a zhromažďujú terabajty nespracovaných oceánografických údajov. Predtým, ako vedci a inžinieri môžu použiť tieto informácie vo svojej práci, musia ich previesť na zmysluplné údaje.

MATLAB umožnil uplatniť odborné znalosti v oblasti spracovania signálov a sonaru na vývoj a nasadenie SonarScope, vysokovýkonného softvérového produktu na spracovanie, analýzu a vizualizáciu nespracovaných údajov MBES. SonarScope (Obr. 14.6.) je vysoko prispôsobiteľný a poskytuje originálny prístup k spracovaniu údajov MBES, čo z neho robí neoceniteľný nástroj pre výskumníkov a inžinierov, ktorí potrebujú testovať a kalibrovať systémy MBES. MATLAB sa ukázal ako ideálne prostredie pre vývoj SonarScope, pretože umožnil vyvíjať algoritmy, vizualizovať výsledky a potom algoritmy vylepšovať v iteratívnom cykle (Augustin, 2011).



Obr. 14.6. Rozhranie SonarScope vytvorené pomocou MATLABu. Centrálny obrázok zobrazuje aktívnu dátovú vrstvu. Tlačidlá v spodnej časti umožňujú interakciu s údajmi. Ponuky v hornej časti poskytujú rýchly prístup k nástrojom spracovania (Augustin, 2011).

Bezpilotné a autonómne podvodné vozidlá sa čoraz viac nasadzujú na mapovanie morského dna, detekciu ponorených objektov a dokončenie ďalších podmorských misií, ktoré si vyžadujú sonarové systémy s vysokým rozlíšením. Na vývoj viacúčového sonarového systému pre akustické zobrazovanie s vysokým rozlíšením musia inžinieri integrovať akustické prevodníky a ďalšie analógové komponenty s komponentmi digitálneho spracovania signálu (DSP). V minulosti boli analógový a digitálny dizajn úplne oddelené procesy: Inžinieri navrhli piezoelektrické prvky prevodníkov pomocou nástrojov založených na metódach konečných prvkov a digitálne komponenty pomocou nástrojov na spracovanie signálu. Tento prístup si často vyžadoval početné opakovania pokusov a omylov, čím sa zvyšovali náklady a dodacie lehoty. Ďalej, spustenie digitálneho dizajnu bez úplného pochopenia správania a charakteristík analógových komponentov často znamenalo rozsiahle prepracovanie digitálneho dizajnu po dokončení analógovej časti. V NEC implementovali nový dizajnový prístup pre viacúčové sonarové systémy založený na modelovom dizajne s MATLAB a Simulink. V jedinom prostredí modelujú a simulujú analógové prevodníky a zvukové pole, v ktorom pracujú, spolu s komponentmi DSP (Obr. 14.7.). Tento prístup umožňuje optimalizovať návrh celého systému skôr ako kedykoľvek predtým, overiť jeho funkčnosť a vyladiť parametre na prototypoch DSP a FPGA a dodať kompletný systém podľa plánu (Kuroda, 2018).

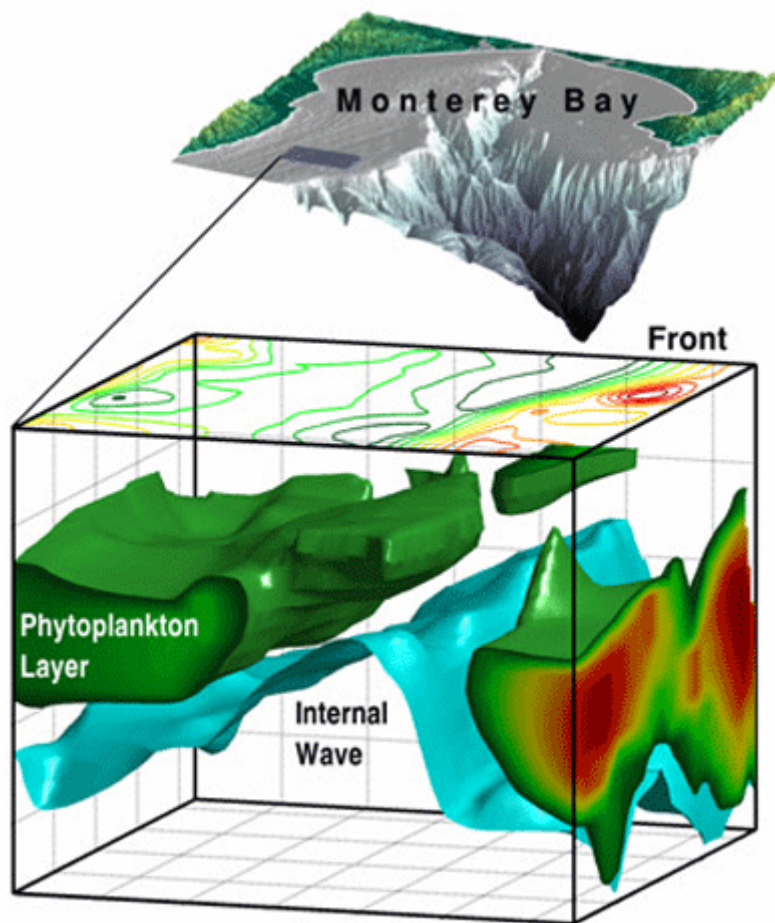


Obr. 14.7. Vysokoúrovňový diagram modelu viacúčového sonarového systému NEC (Kuroda, 2018).

14.2.3 Ďalšie alternatívy využitia vo vodnom hospodárstve

Monterey Bay Aquarium Research Institute (MBARI) v Kalifornii používa autonómne podvodné vozidlá (AUV) na rýchly prieskum fyzikálnych, chemických, biologických a geologických podmienok od povrchu oceánu až po morské dno. Výskumníci MBARI použili MATLAB na vývoj automatizovaných nástrojov na rýchlu analýzu a vizualizáciu údajov AUV. Tieto nástroje MATLAB zlepšujú prevádzku AUV a umožňujú rýchlu produkciu a vizualizáciu kvalitných vedeckých údajov (Obr. 14.8.). Výskumníci používajú údaje na skúmanie javov v hornom oceáne, vrátane prílivov červených a škodlivých rias, a na vývoj máp morského dna s vysokým

rozlíšením pre štúdium geológie, geochemického vetrania a hlbokomorských biologických spoločenstiev (MathWorks, 2007).



Obr. 14.8. Ukážka výstupov z mapovania oceánov (MathWorks, 2007).

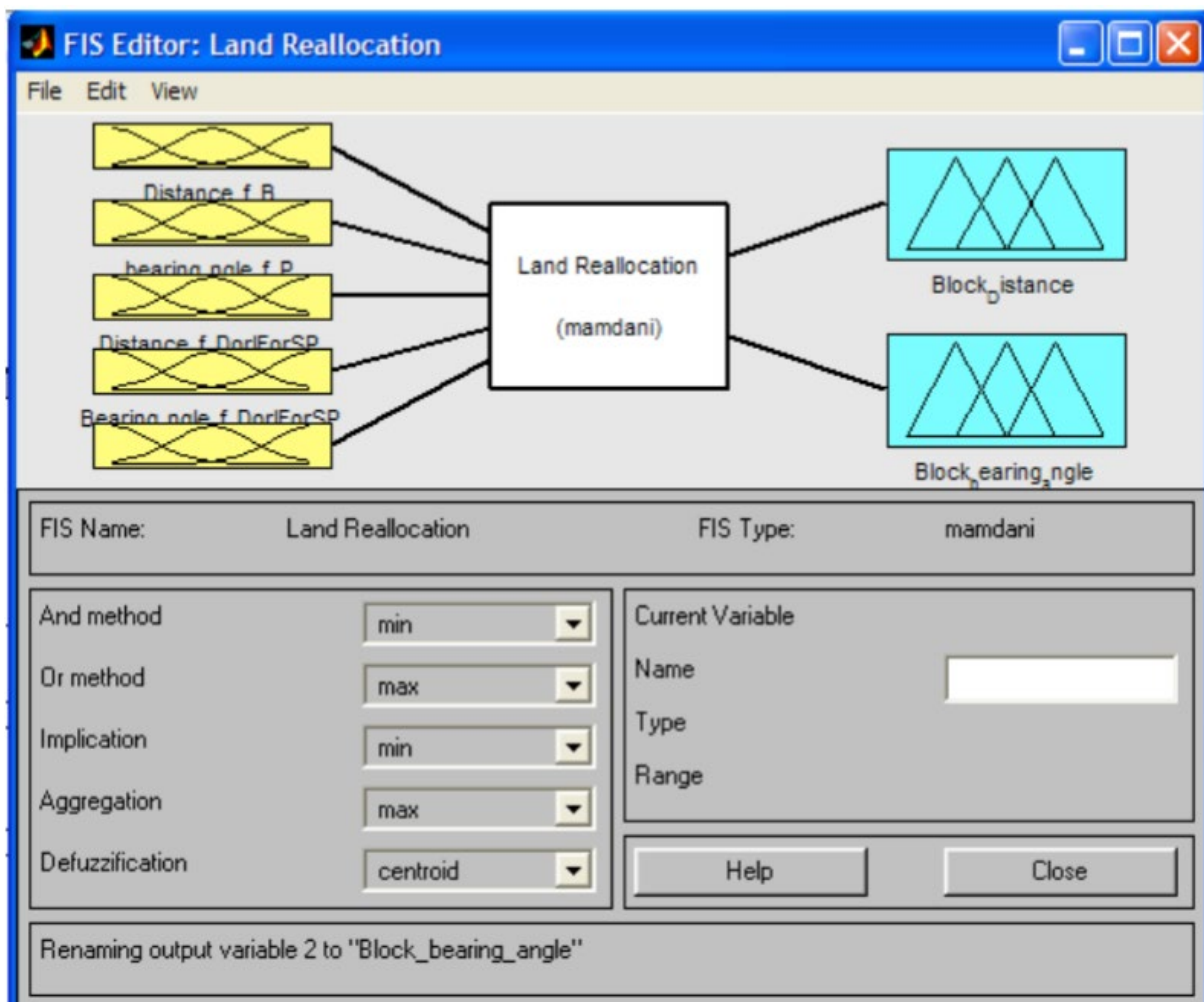
Sieť monitorovania kvality vody v reálnom čase funguje na ochranu vodných zdrojov a zisťovanie kvality vody. Pre plány manažmentu vodohospodárskych podnikov a miestnych orgánov je užitočné realizovať zmenu charakteristickú pre kvalitu vody. V oblasti pitnej vody vo vodovodnej sieti je potrebné poskytnúť veľmi dobrú reprezentáciu trendu zmeny časových radov zvyškového chlóru a tiež predložiť nový prístup k štúdiám o zmene štádia zvyškového chlóru a strednodobú predikciu pre časové rady, aby bolo možné čo najpresnejšie riadenie a dávkovanie chlóru do vodovodnej siete (Han & Huang, 2009).

CFDTool je založený na softvéri a simulačnej platforme FEATool Multiphysics a obsahuje zjednodušené používateľské rozhranie, ktoré bolo špeciálne navrhnuté a vyvinuté tak, aby boli simulácie výpočtovej dynamiky tekutín (CFD) a prenosu tepla jednoduché a zábavné! Sada nástrojov je plne integrovaná s MATLAB, všetko, čo potrebujete na nastavenie, vykonávanie a analýzu CFD simulácií, vrátane CAD geometrického modelovania, automatického generovania siete, postprocessingu a vizualizácie, ako aj vstavanej podpory pre OpenFOAM a SU2 CFD riešiče. Intuitívne GUI vám umožňuje začať s modelovaním ihneď bez toho, aby ste trávili čas učením sa zložitého riešiteľa alebo programovania syntaxu. Nelineárne výrazy a koeficienty je možné definovať a zadávať tak, ako sú napísané na papieri. Jedinečná integrácia riešiteľov vám navyše umožňuje definovať váš model raz a potom použiť ľubovoľne a všetky riešiče CFD zameniteľne, čo môže byť neoceniteľné pri testovaní a overovaní (Precise Simulation, 2023).

PIVlab je softvér na meranie rýchlosti častíc (PIV) založený na grafickom používateľskom rozhraní (GUI). Nepočíta len distribúciu rýchlosti v rámci párov obrázkov častíc, ale môže sa použiť aj na odvodenie, zobrazenie a export viacerých parametrov vzoru prúdenia. Užívateľsky prívetivé GUI so schopnosťou ovládať PIV kameru a laser umožňuje rýchle a efektívne získavanie PIV údajov a následné spracovanie údajov (Thielicke & Sonntag, 2021).

14.3 Pozemkové úpravy

V roku 2008 Tayfun Cay a Fatih Iscan z Turecka vo fuzzy logickom toolboxe softvéru MATLAB vytvorili model (Obr. 14.9), pomocou ktorého modelovali fázu prerozdelenia pozemkov pri pozemkových úpravách, ktorá nemá jednoznačný matematický model. Pomocou tejto metódy možno dosiahnuť rýchle, ekonomické a efektívne výsledky. Keďže sú dané vstupné a výstupné premenné týkajúce sa modelu, realizátori môžu realizovať svoje projekty podľa týchto premenných. Vo vstupných premenných pre určité projekty sa môže vyskytnúť množstvo rozdielov, ktoré však neovplyvnia významným spôsobom model prerozdelenia pôdy (Cay & Iscan, 2008).



Obr. 14.9. Model fuzzy systému typu Mamdani v softvéri MATLAB (Cay & Iscan, 2008).

Obdobnú úlohu riešili aj v Chorvátku. Oceňovanie katastrálnych parciel je veľmi dôležitým aspektom pri definovaní priorít pre výber novovzniknutých parciel v procese prerozdelenia. Prerozdelenie pozemkov zahŕňa nové prerozdelenie pozemkov v oblasti realizácie projektu napr. pozemkových úprav. Pritom je potrebné brať do úvahy preferencie súkromných

vlastníkov, hodnotu katastrálnych parciel a ich bonitové hodnoty, ktorých určenie je základnou úlohou pri riešení pozemkových úprav alebo realokácie pozemkov. Hlavným cieľom bolo navrhnúť model hodnotenia bonitetu súkromných katastrálnych parciel na báze Expertného systému fuzzy logiky v rámci znalostnej zložky, ktorý by znížil neistotu a zvýšil objektivitu hodnotenia. Odborné znalosti boli zahrnuté v procese hodnotenia definovaním váhových koeficientov na optimalizáciu základne pravidiel a lineárnych a nelineárnych hodnotových funkcií na štandardizáciu kritérií. Aplikáciou novovzniknutého systému v bonitetskom hodnotení, s vytvorenou bázou odborných poznatkov, došlo k skvalitneniu procesov odhadu hodnôt atribútov, najmä v podobe zníženia neistoty pri hodnotení mestských parciel, ako aj zvýšenia objektivitu o zapojenie skupiny odborníkov do tvorby modelu. Navrhnutý model tiež poskytuje rovnaký prístup všetkým zainteresovaným stranám v procese obnovy miest s rôznymi požiadavkami a želaniami. Model tiež poskytuje podporu pri vedení negociačných konaní a plánovaní realizácie realokácie pozemkov (Kilić Pamuković et al., 2021).

Vytvorený kód v prostredí MATLAB bol súčasťou aj výskumu v oblasti plánovania poľnohospodárskej činnosti a najmä hľadanie vhodnej pôdy pre dosahovanie čo najvyšších úrod, nakoľko v súčasnom poľnohospodárskom scenári je dostupnosť vhodnej pôdy na pestovanie menšia a rentabilné pridelovanie pôdy na pestovanie plodín sa javí ako ťažkopádna úloha. Optimalizácia plánovania plodín je hlavnou oblasťou výskumu v poľnohospodárstve, v ktorej je optimalizácia pôdy významnou výzvou, ktorá spadá do kategórie problémov kombinatorickej optimalizácie. Hlavným cieľom súčasného výskumu je maximalizovať čistý príjem z poľnohospodárstva prostredníctvom optimálnej alokácie pôdy. Biologicky inšpirované algoritmy sú veľmi populárne pri riešení problémov kombinatorickej optimalizácie. Social Spider Algorithm (SSA), bioinšpirovaný algoritmus, sa používa na riešenie problému optimalizácie pôdy v tomto výskume založenom na simulácii kooperatívneho správania sociálnych pavúkov. Optimálna oblasť výsadby, produktivita plodín pre rôzne poľnohospodárske podniky a požiadavky na vodu sú vypočítané SSA a doteraz získané výsledky ukázali lepšie smery pre poľnohospodárske plánovanie na zlepšenie zisku s obmedzenou rozlohou pôdy a obmedzeniami dostupnosti vody. Výsledky tohto výskumu možno použiť na odporúčania týkajúce sa pestovania plodín pre farmárov s rôznymi pôdnymi podnikmi a nedostatočnou dostupnosťou vody. V rámci budúceho výskumu plánujú tvorcovia zvážiť ďalšie optimalizačné techniky pri plánovaní plodín, ako je optimalizácia pôdy, práce, dopravy, klímy, buriny, hnojív a pesticídov (Thilagavathi & Amudha, 2019).

14.4 Iné typy úloh v krajinárstve

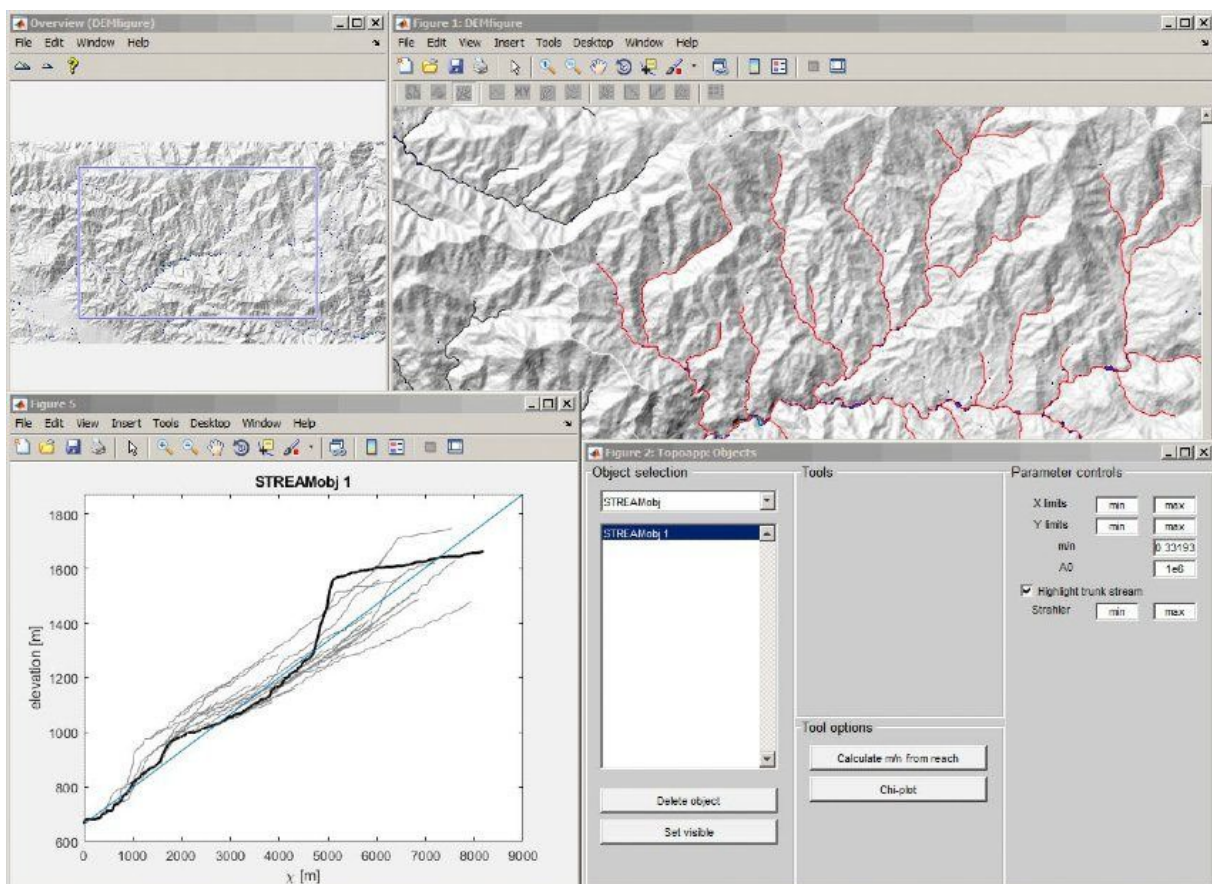
Okrem oblasti odpadového a vodného hospodárstva, či pozemkových úprav sú nástroje programu MATLAB využívané v rámci krajinárstva napr. aj pri mapovaní zosuvných území, území náchylných na pád lavín, spracovanie údajov o pôde alebo klíme a pod.

Výskumníci z Národného úradu pre oceán a atmosféru (NOAA) vyvinuli nástroje na predpovedanie toho, ako oscilácie teploty morského povrchu v Atlantickom a Tichom oceáne ovplyvnia klimatické cykly. Atlantické oceánografické a meteorologické laboratórium NOAA v Miami použilo MATLAB, Statistics a Machine Learning Toolbox™ na odhadnutie distribučnej funkcie pre tieto klimatické cykly a na výpočet pravdepodobnosti klimatických zmien v rámci určitého počtu rokov. Výskumníci zrekonštruovali minulé oscilácie analýzou prstencov zo 450-ročných stromov a potom použili metódy prevzorkovania Monte Carlo na zväčšenie veľkosti vzorky. Nástroje klimatickej projekcie umožnia vedcom a vodohospodárskym inžinierom vyvinúť dlhodobé projekcie zrážok a aktivity hurikánov (MathWorks, 2007).

Ministerstvo poľnohospodárstva Kanady (AAFC) vyvinulo systém poľnohospodárskeho mapovania na podporu ochrany pôdy a trvalo udržateľného poľnohospodárstva v Kanade.

Pomocou MATLAB AAFC analyzovala radarové snímky zo satelitov na pozorovanie Zeme a vyvinula modely, ktoré extrahujú reakcie spätného rozptylu radarov poľnohospodárskych oblastí. AAFC používa výsledky tejto analýzy na hodnotenie postupov obrábania pôdy, pomáha poľnohospodárom zlepšovať kvalitu pôdy a predchádzať erózii ornice (MathWorks, 2007).

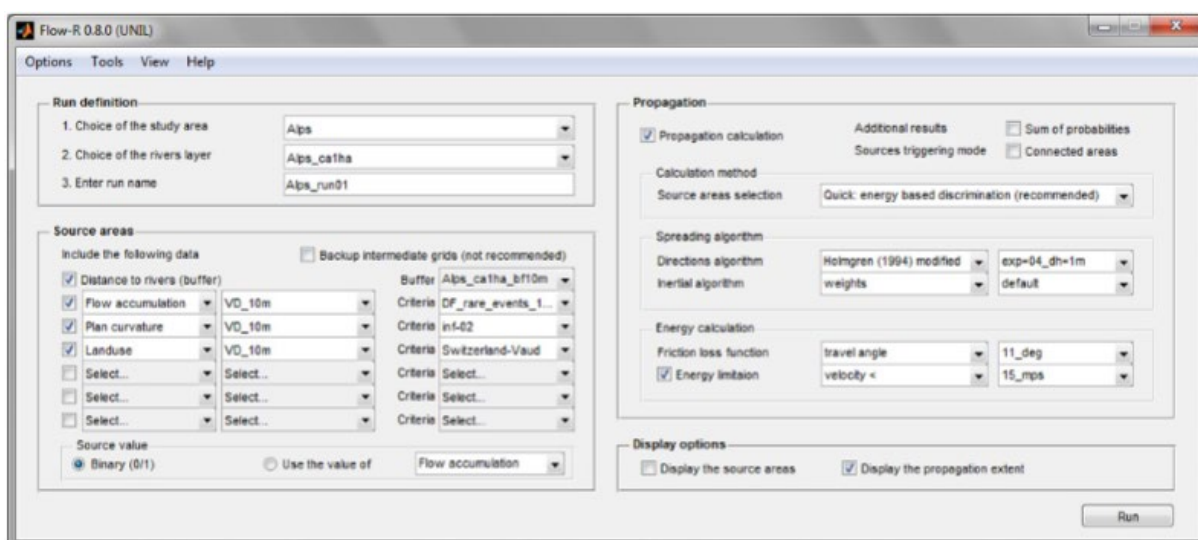
Dr. Wolfgang Schwanghart z University of Potsdam sa pred rokmi zamyslel, či vzhľadom na digitálny výškový model (DEM), existoval rýchly a elegantný spôsob, ako vypočítať, koľko vody preteká cez danú krajinu počas zrážok a po nich? Odpoveďou bolo použiť MATLAB na vyriešenie problému ako systém lineárnych rovníc. Tento prístup bol genézou TopoToolbox, súboru funkcií MATLAB, ktorý vyvinul na analýzu dráh toku v DEM. Použil MATLAB a TopoToolbox na urýchlenie vlastného výskumu, vrátane štúdií, ktoré predpovedajú, kde najpravdepodobnejšie dôjde k zosuvom pôdy, ako sa extrémne záplavy šíria horskými údoliami a skúmal geologické javy. Výskumníci z celého sveta použili súpravu týchto nástrojov na štúdium vodných tokov v ľadovcoch, vytvárania supraglaciálnych jazier a fjordov a spôsobu, akým sa deformujú siete prúdov, keď sa tektonické platne posúvajú do strany. Postupne bolo pridané používateľské rozhranie (Obr. 14.10.) na analýzu krajiny v oblastiach aktívnej deformácie. Ďalší výskumník, ktorý študuje súhrn tektonických a povrchových procesov, prispel komplexným súborom funkcií, ktoré umožňujú kolegom replikovať jeho prácu v iných regiónoch sveta (Schwanghart, 2017).



Obr. 14.10. Rozhranie TopoToolbox topoapp, ktoré poskytuje nástroj na interaktívne využívanie DEM pre tektonické procesy (Schwanghart, 2017).

14.4.1 Lavíny

Vypracovanie máp náchylnosti na trasy úlomkov skál a iných materiálov má primárny význam z dôvodu tlaku obyvateľstva v nebezpečných zónach. Posúdenie nebezpečenstva pomocou procesného modelovania v regionálnom meradle je však zložitá kvôli komplexnej povahe javu, variabilite miestnych riadiacich faktorov a neistote parametrov modelovania. Regionálne hodnotenie musí brať do úvahy zjednodušený prístup, ktorý nie je veľmi závislý od parametrov a ktorý môže poskytnúť zónovanie s minimálnymi požiadavkami na údaje. Takto bol vyvinutý distribuovaný empirický model pre regionálne hodnotenia citlivosti s použitím digitálneho modelu nadmorskej výšky (DEM). Flow-R (Flow path assessment of gravitational hazards at a Regional scale – Hodnotenie dráhy toku gravitačných rizík v regionálnom meradle) je distribuovaný empirický model na hodnotenie regionálnej citlivosti tokov materiálu, vyvinutý na Univerzite v Lausanne (Obr. 14.11). Bol úspešne aplikovaný na rôzne prípadové štúdie v rôznych krajinách. Zistilo sa, že model je relevantný aj na posúdenie iných prírodných nebezpečenstiev, ako je padanie skál alebo snehové lavíny. Umožňuje automatické vymedzenie zdrojovej oblasti a posúdenie rozsahu šírenia. Voľby množín údajov a algoritmov sú otvorené pre používateľa, vďaka čomu sú v súlade s rôznymi aplikáciami a dostupnosťou množín údajov (Horton et al., 2013).



Obr. 14.11. Snímka obrazovky hlavného okna softvéru Flow-R (Horton et al., 2013).

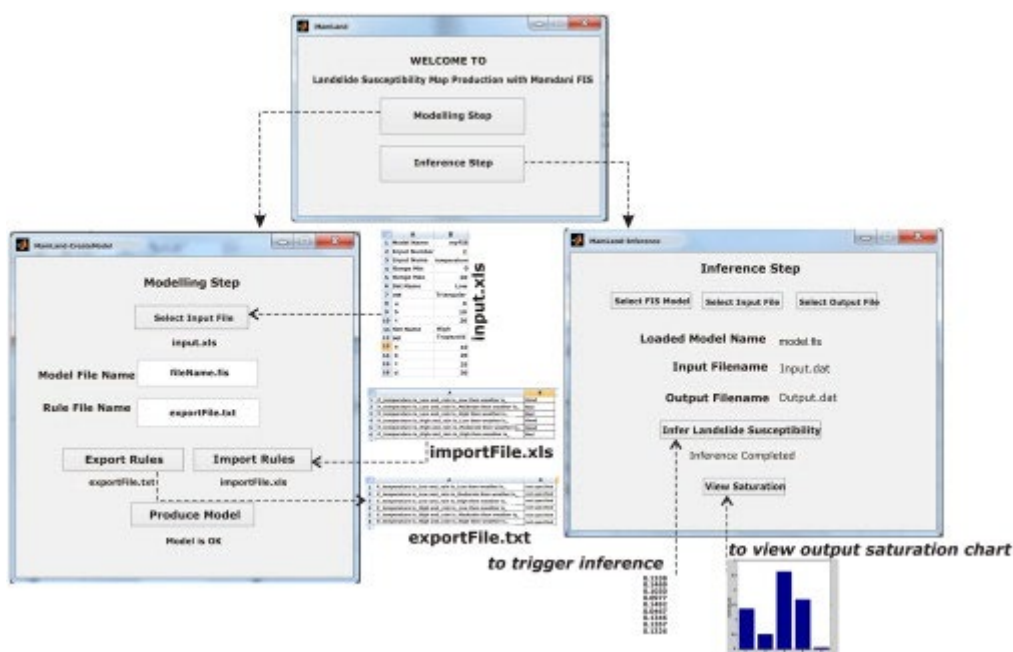
Ďalej napríklad Portenier et al. (2019) vyvinuli poloautomatický postup implementovaný v MATLABe na odvodenie snehových máp z verejne dostupných obrázkov webových kamier. Rubin et al. (2012) využili nástroje MATLABu na analýzu frekvenčných údajov a implementovali ich na záznamy zo seizmických meraní pre detekciu snehových lavín v oblastiach bez akýchkoľvek priamych pozorovaní a meraní. Podobne aj Johnson et al. (2021) využili nástroje na analýzu zvukových a video záznamov odtrhnutých lavín na detegovanie pohybu lavíny po svahu.

14.4.2 Zosuvy

Informácie týkajúce sa rozsahu nebezpečenstva sú kľúčové pre hodnotenie jeho súvisiaceho vplyvu. Kvantitatívna analýza objemu zosuvu pôdy bezprostredne po udalosti môže ponúknuť lepšie pochopenie a kontrolu prispievajúcich faktorov a ich relatívnej dôležitosti. Takéto informácie nie je možné zhromaždiť pre každú udalosť zosuvu pôdy z dôvodu obmedzení pri získavaní použiteľných prvotných údajov a nevyhnutných postupov každej použitej

technológie. Na predpovedanie zmeny objemu sa často používajú empirické pravidlá, ale výsledná presnosť je veľmi nízka. Tradičné metódy využívajú fotogrametriu alebo detekciu svetla a meranie vzdialenosti (LiDAR) na vytvorenie digitálneho modelu terénu (DTM) po udalosti. Tieto metódy sú finančne aj časovo náročné. Genetický algoritmus (GA) naprogramovaný v MATLABe sa používa na inteligentné predpovedanie zmeny nadmorskej výšky pre každý pixel obrázka. Každý DTM sa teda prevedie na obrázok tieňového reliéfu a porovná sa s jedným diaľkovo snímaným obrázkom po udalosti na účely hodnotenia podobnosti. Metóda je schopná rýchlo odhadnúť objem zmeny terénu v rámci chyby 25 %, bez oneskorenia pri získavaní párov stereo obrazu alebo bez potreby pozemných kontrolných bodov alebo profesionálneho softvéru na fotogrametriu (Wang et al., 2014).

Pre región Sinop (severné Turecko) a jeho blízke okolie bolo aplikované mapovanie náchylnosti na zosuvy s použitím úplne expertného prístupu založeného na názoroch odborníkov v dotazníku. Na tento účel bol vyvinutý ľahko použiteľný program „MamLand“ (Obr. 14.12) na konštrukciu fuzzy inferenčného systému (FIS) a použitý v MATLABe. Pomocou tohto programu je možné zostaviť mapu náchylnosti na zosuv na základe odborného posudku. V tejto štúdii bolo do FIS zahrnutých sedem kondicionálnych parametrov charakterizujúcich topografické, geologické a environmentálne podmienky. Pre študovanú oblasť sa získal súbor údajov o inventarizácii zosuvov zahŕňajúci 351 miest zosuvov. Po dokončení fázy produkcie údajov štúdie boli údaje spracované pomocou FIS typu Mamdani. V tomto systéme sa hodnotili len údaje o úprave zosuvov a údaje o inventarizácii zosuvov neboli zahrnuté do hodnotiaceho prístupu. Preto bol pomocou Mamdani FIS vytvorený súbor zobrazujúci stupne náchylnosti na zosuv pôdy pre skúmanú oblasť. Tieto stupne sa potom exportovali do prostredia GIS a vytvorila sa mapa náchylnosti na zosuv pôdy, ktorá sa vyhodnotila z hľadiska štatistickej interpretácie. Z hodnotenia sa zistilo, že táto mapa náchylnosti na zosuv pôdy, ktorá bola vytvorená spôsobom nezávislým od údajov, bola úspešná (Akgun et al., 2012).



Obr. 14.12. MamLand: rozhrania, vstupy a výstupy (Akgun et al., 2012).

Väčšina metód hodnotenia náchylnosti na zosuv pôdy je založená na matematických vzťahoch vytvorených medzi faktormi zodpovednými za spustenie javu, nazývanými podmieňujúce faktory. Tie sú zvyčajne odvodené z geografických údajov, ktoré sa bežne spracúvajú

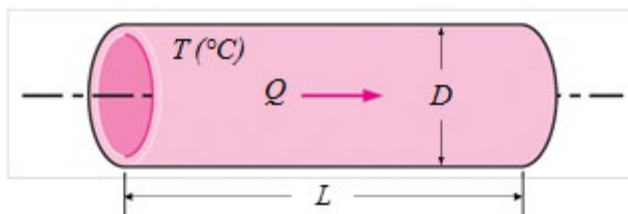
prostredníctvom technológie geografického informačného systému (GIS). Podľa prijatej metodiky je po úvodnej fáze realizovanej na platforme GIS potrebné dáta preniesť do špecifického softvéru, napr. MATLAB, na analýzu a spracovanie. Platformy na riadenie rizík založené na GIS sú preto niekedy hybridné a vyžadujú relatívne zložité adaptívne postupy pred výmenou údajov medzi rôznymi prostrediami. Gatto et al. (2022) popisuje, ako možno MATLAB použiť na odvodenie najbežnejších faktorov podmieňujúcich zosuvy pôdy, a to spravovaním geografických údajov v ich typických formátoch: rastrové, vektorové alebo bodové údaje. Konkrétne sa diskutuje o tom, ako vytvoriť matice parametrov, ktoré sú potrebné na posúdenie citlivosti, pomocou jednotiek mapovania buniek mriežky a ich mapovaním obchádzajúc GIS. Uvádza sa aplikácia týchto predbežných operácií na študovanú oblasť postihnutú plytkými zosuvmi pôdy v minulosti; výsledky ukazujú, ako možno geoúdaje spravovať tak jednoducho ako v GIS a zároveň sa môžu zobrazovať moderným spôsobom. Na podobnom princípe, avšak bez využitia prepojenia s GIS, spracovali analýzu náchylnosti územia na zosuvy aj Bhatt a Papola (2014) s využitím externého vyhodnotenia údajov nástrojom WEKA.

ČASŤ B

Táto časť obsahuje konkrétne riešené príklady na problémy pre využitie v oblasti krajinného inžinierstva.

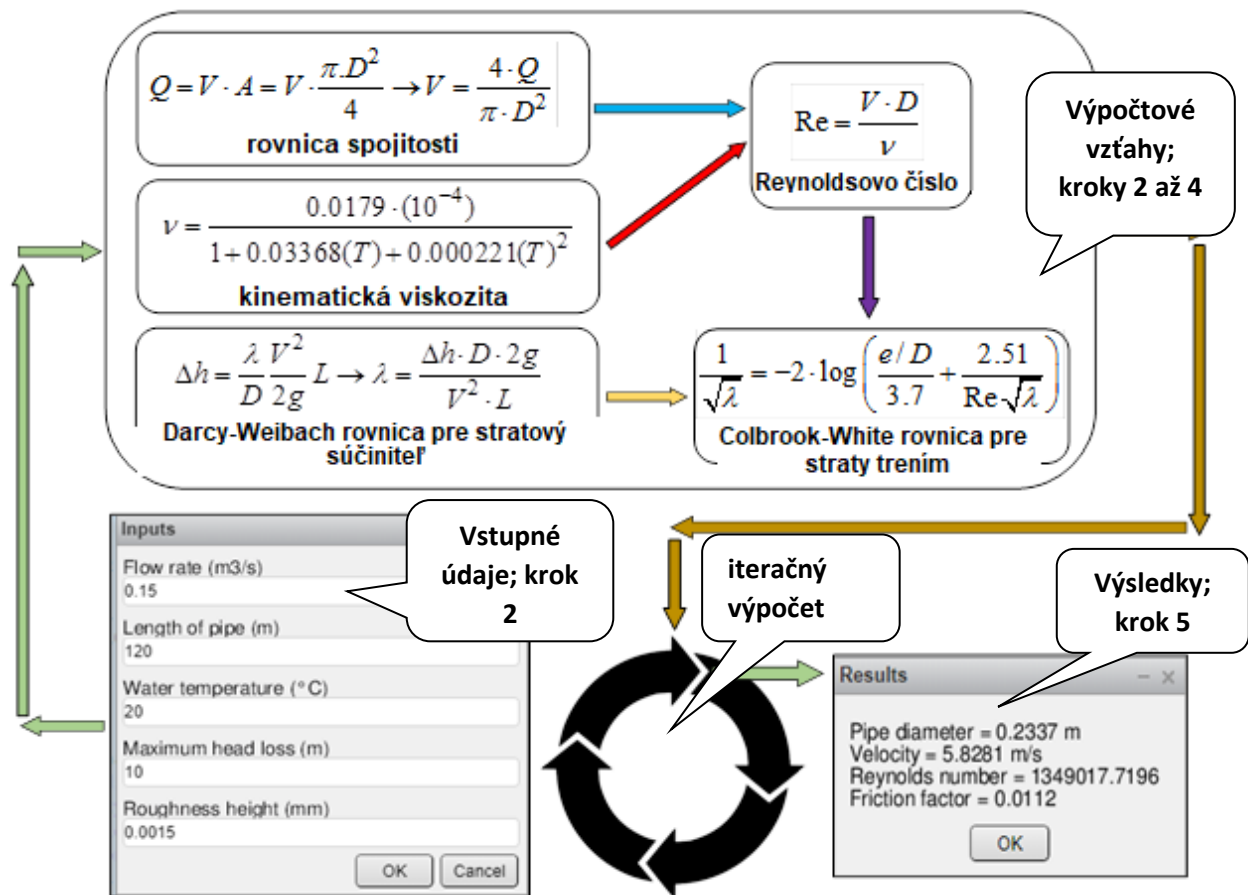
Problém 1: Určenie minimálneho priemeru potrubia

Zadanie: Voda 20°C sa má dopravovať v plastovom potrubí o dĺžke 120 m s prietokom $0.25 \text{ m}^3\text{s}^{-1}$ (Obr. P-1.1). Výška drsnosti pre materiál tohto potrubia je 0.0015 mm. Ak tlaková strata v potrubí nemá presiahnuť 10 m, určte minimálny priemer (D) potrubia tak, aby ste zabezpečili priradenú hodnotu maximálnej tlakovej straty.



Obr. P-1.1. Schéma k problému 1.

Výpočty sa vykonávajú spustením skriptov MATLABu spracovaných podľa Almoulki a Yetilmezsoy (2019) pričom budeme postupovať podľa grafického návodu (Obr. P-1.2).



Obr. P-1.2. Vizuálne znázornenie metodiky riešenia problému 1 (Almoulki & Yetilmesoy, 2019).

➤ Postup riešenia [problému 1](#) v MATLAB je nasledovný:

%Určenie minimálneho priemeru potrubia

%KROK #1

clear, clc

format short

disp(datestr(now,0))

%KROK #2, vytvorenie GUI pre zadanie vstupných parametrov

a1 = {'Prietok (m3/s)', 'Dĺžka potrubia (m)', 'Teplota vody (°C)', 'Maximalna stratova vyska (m)', 'Vyska drsnosti (mm)'};

a2 = 'Vstupy';

a3 = [1 40; 1 40; 1 40; 1 40; 1 40];

a4 = inputdlg(a1,a2,a3);

tic

```

%KROK #3
b = char(a4);
c = str2num(b);
Q = c(1); L = c(2); T = c(3); h = c(4); e = c(5);
e1 = e./1000;
%výpočet kinematickej viskozity
nu = (0.01779./(1+0.03368.*T+0.000221.*(T.^2))).*(10.^(-4));
g = 9.807;

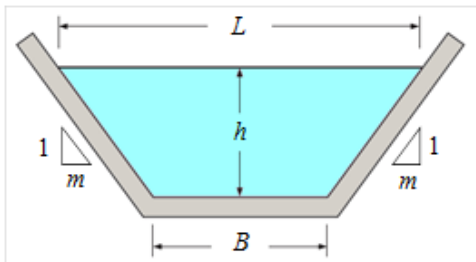
%KROK #4
Dp = 0.001;
E1 = 0;
E = 1;
while E1<E
%výpočet rýchlosti
V = (4*Q)/(pi.*(Dp.^2));
%výpočet Reynoldsovho čísla
Re = (V.*Dp)./nu;
%výpočet súčiniteľa pre straty trením podľa Darcy-Weisbach rovnice
f = (h.*Dp.*(2.*g))/(L.*(V.^2));
%výpočet stratovej výšky trením
E1 = sqrt(f)*(-2)*log10(((e1./Dp)./3.7)+(2.51./(Re.*sqrt(f))));
Dp = Dp + 0.000001;
end

%KROK #5
% Prevedenie čísel na znaky, zobrazenie výsledkov v okne, výsledky budú zobrazené na 4
desatinné miesta
Dp_s = sprintf('Priemer potrubia = %1.4f m',Dp);
V_s = sprintf('Rychlost = %1.4f m/s',V);
Re_s = sprintf('Reynoldsovo cislo = %1.4f',Re);
f_s = sprintf('Straty tremim = %1.4f',f);
d = char(Dp_s,V_s,Re_s,f_s);
msgbox(d,'Vysledky');
toc

```

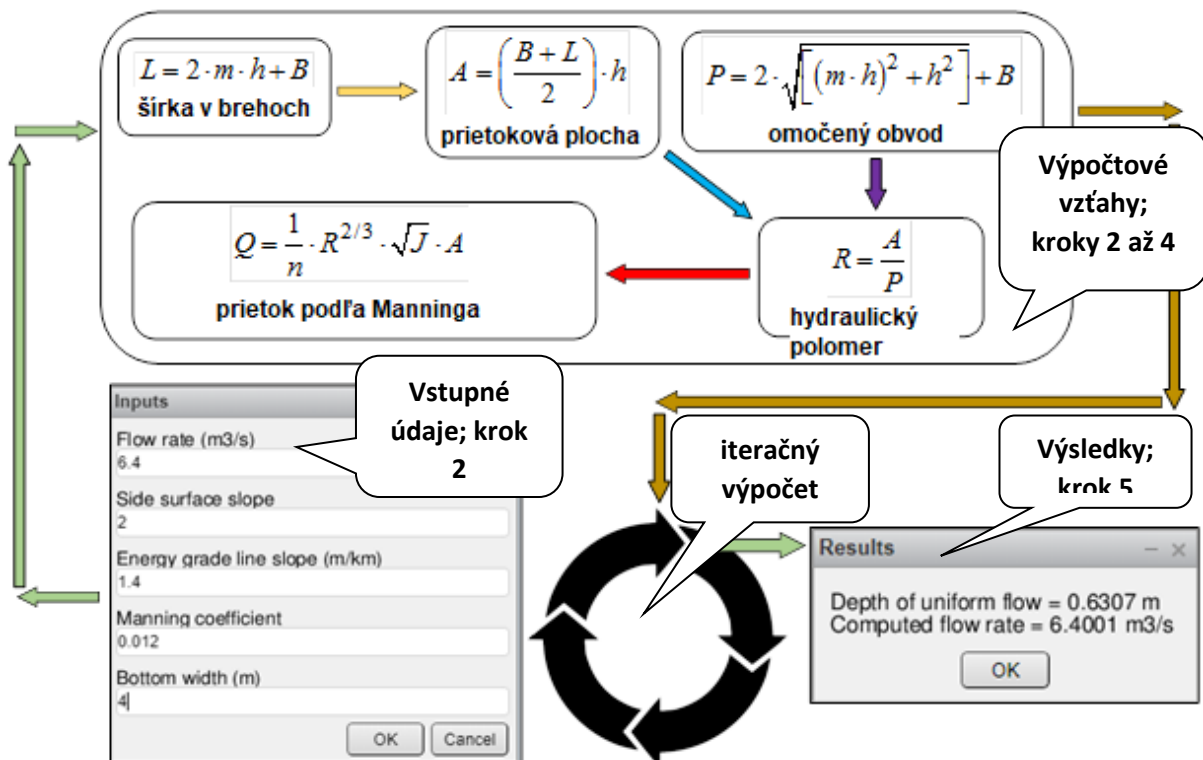
Problém 2: Určenie hĺbky rovnomerného prúdenia v otvorenom kanáli

Zadanie: Prietok cez otvorený hotový betónový žľab lichobežníkového prierezu (Obr. P-2.3) je $6.4 \text{ m}^3 \cdot \text{s}^{-1}$. Bočný sklon svahov (m) je 2:1 a sklon dna koryta je $1.4 \text{ m} \cdot \text{km}^{-1}$. Manningov koeficient pre otvorený kanál s povrchni z hotového betónu je 0.012 a šírka dna tohto kanála je 4 m. Prúdenie je stabilné a rovnomerné a sklon dna je konštantný. Predpokladá sa, že drsnosť vlhkého povrchu kanála a koeficient trenia sú konštantné. Určite hĺbku rovnomerného toku a ďalšie rozmery kanála.



Obr. P-2.3. Schéma k problému 2.

Výpočty sa vykonávajú spustením skriptov MATLABu spracovaných podľa Almoulki, Yetilmesoy (2019) pričom budeme postupovať podľa grafického návodu (Obr. P-2.4).



Obr. P-2.4. Vizualne znázornenie metodiky riešenia problému 2 (Almoulki, Yetilmesoy, 2019).

➤ V MATLABe riešime [problém 2](#) nasledovne:

%Určenie hĺbky rovnomerného toku v otvorenom kanáli

%KROK #1

clear, clc

format short

disp(datestr(now,0))

%KROK #2, vytvorenie GUI pre zadanie vstupných parametrov

a1 = {'Prietok (m³/s)', 'Sklon svahov', 'Sklon dna (m/km)', 'Manningov sučiniteľ', 'Šírka dna (m)'};

a2 = 'Vstupy';

a3 = [1 40; 1 40; 1 40; 1 40; 1 40];

a4 = inputdlg(a1,a2,a3);

tic

%KROK #3

b = char(a4);

c = str2num(b);

Q = c(1); m = c(2); J = c(3); n = c(4); B = c(5);

%prepočet sklonu na bezrozmerné číslo

J1 = J./1000;

%KROK #4

h = 0.01;

Q1 = 0;

while Q1 < Q

%výpočet šírky v brehoch

L = 2.*m.*h+B;

%výpočet prietokovej plochy

A = ((B+L)./2).*h;

%výpočet omočeného obvodu

P = abs(m.*h+h.*i).*2+B;

%výpočet hydraulického polomeru

R = A./P;

```
%výpočet prietoku použitím Chezyho a Manningovej rovnice
```

```
Q1 = (1./n).*(R.^(2./3)).*sqrt(J1).*A
```

```
h = h + 0.00001;
```

```
end
```

```
%KROK #5
```

```
% Prevedenie čísel na znaky, zobrazenie výsledkov v okne, výsledky budú zobrazené na 4  
desatinné miesta
```

```
h_s = sprintf('Hĺbka vody = %1.4f m',h);
```

```
Q1_s = sprintf('Vypocitany prietok = %1.4f m3/s',Q1);
```

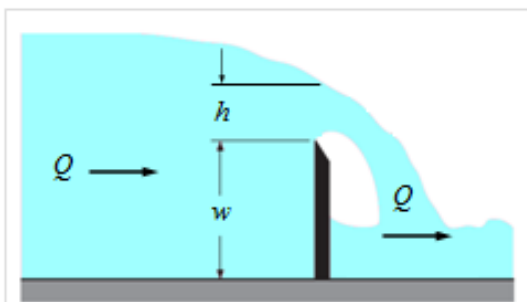
```
d = char(h_s,Q1_s);
```

```
msgbox(d,'Vysledky');
```

```
toc
```

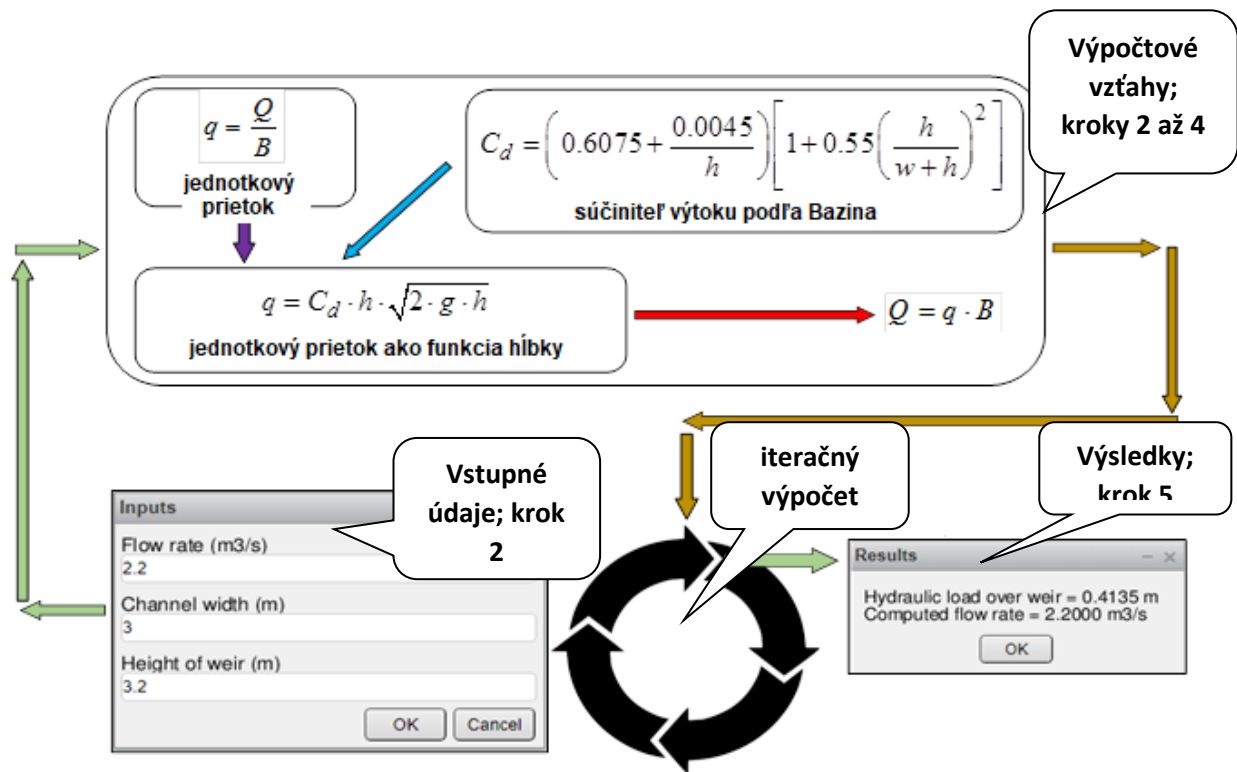
Problém 3: Určenie prepadovej výšky pri prepade cez ostrú hranu

Zadanie: Priepad s ostrou hranou je zvislá doska umiestnená v kanáli, ktorý núti kvapalinu pretekať otvorom na meranie prietoku (Obr. P-3.5). Prietok prechádzajúci cez ostrú hranu s obdĺžnikovým tvarom je $2.2 \text{ m}^3 \cdot \text{s}^{-1}$. Ak je výška tohto priepadu (w), ktorá bude umiestnená do 3 m širokého pravouhlého kanála, 3.2 m, určte prepadovú výšku (h).



Obr. P-3.5 Schéma k problému 3.

Výpočty sa vykonávajú spustením skriptov MATLABu spracovaných podľa Almoulki, Yetilmezsoy (2019) pričom budeme postupovať podľa grafického návodu (Obr. P-3.6).



Obr. P-3.6. Vizuálne znázornenie metodiky riešenia problému 3 (Almoulki & Yetilmeszo, 2019).

➤ V prostredí MATLAB je výpočet [problému 3](#) nasledovný:

%Určenie hĺbky rovnomerného toku v otvorenom kanáli

%KROK #1

clear, clc

format short

disp(datestr(now,0))

%KROK #2, vytvorenie GUI pre zadanie vstupných parametrov

a1 = {'Prietok (m3/s)', 'Sirka kanala (m)', 'Vyska priepadu (m)'};

a2 = 'Vstupy';

a3 = [1 40; 1 40; 1 40];

a4 = inputdlg(a1,a2,a3);

tic

%KROK #3

b = char(a4);

```

c = str2num(b);
Q = c(1); B = c(2); w = c(3);
%výpočet jednotkového prietoku
q = Q./B;
g = 9.807;

%KROK #4
h = 0.01;
q1 = 0;
while q1 < q
%prvý čiastkový výpočet pre súčiniteľ výtoku podľa Bazina
part1 = 0.6075 + (0.0045./h);
%druhý čiastkový výpočet pre súčiniteľ výtoku podľa Bazina
part2 = 1 + 0.55.*((h.^2)./((w+h).^2));
%výpočet súčiniteľa výtoku podľa Bazina
Cd = part1.*part2;
%výpočet jednotkového prietoku ako funkcie prepadovej výšky
q1 = Cd.*h.*sqrt(2.*g.*h);
h = h + 0.00001;
end

%KROK #5
% Prevedenie čísel na znaky, zobrazenie výsledkov v okne, výsledky budú zobrazené na 4
desatinné miesta
h_s = sprintf('Prepadova vyska = %1.4f m',h);
Q1_s = sprintf('vypocitany prietok = %1.4f m3/s',q1.*B);
d = char(h_s,Q1_s);
msgbox(d,'Vysledky');
toc

```

Problém 4: Analýza denných teplôt

Zadanie: Analyzujte získané denné teploty podľa príkladu z [Live Script Galery](#) podľa Filion (2022).

V tejto časti sa nachádza skript, ktorý bol vybraný z [Live Script Galery](#) a modifikovaný za účelom využívania študentmi Fakulty záhradníctva a krajinného inžinierstva. Tento problém je k dispozícii na precvičenie v príslušnom [MS Tím](#). Vo fialových rámčekoch sú uvedené modifikované údaje.

➤ Na riešenie [problému 4](#) vložíme do MATLABU tento konkrétny skript:

V MathWorks majú meteorologickú stanicu na vrchole jednej z ich budov. Zhromažďuje údaje o priemernej rýchlosti vetra, teplote vzduchu, vlhkosti vzduchu, atmosférickom tlaku a množstve zrážkových úhrnov, na ktorý aplikujú aj tento skript.

Tento konkrétny skript bol upravený za účelom jeho využitia študentami Fakulty záhradníctva a krajinného inžinierstva (FZKI)

Získanie údajov z meteorologickej stanice

MathWorks údaje o počasí načítava z analytickej platformy ThingSpeak, MathWorks IoT, pomocou funkcie thingSpeakRead. Dátum môžeme nastaviť pomocou posuvníkov uvedených nižšie.

Študenti FZKI použijú vlastné údaje vo formáte Excel.

```
clear
year = 2022;
month = 1 ;
day = 5;

d1 = datetime(year, month, day, 0, 0, 0);
if d1 > datetime(datestr(today))
    d1 = datetime(datestr(today));
end
d2 = d1 + days(1);

chID = 12397;
% = readtable(chID, "Fields", 1:8, "DateRange",[d1, d2], "outputFormat", "TimeTable")
%weatherData=Meteodaje2022;
%weatherData=load("Meteoudaje2022.mat")
weatherData=readtable("Meteo_údaje_2022.xlsx")
```

Warning: Column headers from the file were modified to make them valid MATLAB identifiers before creating variable names for the table. The original column headers are saved in the VariableDescriptions property.

Set 'VariableNamingRule' to 'preserve' to use the original column headers as table variable names.

weatherData = 32×4 table

	datum	teplota_priemer	tlak_vodn_ch_p_r_priemer	...
1	NaT	NaN	NaN	
2	01-Jan-2022	10.3750	10.2213	
3	02-Jan-2022	7.4000	9.0660	
4	03-Jan-2022	7.8250	7.9993	
5	04-Jan-2022	6.9000	8.8437	
6	05-Jan-2022	4.9250	7.8220	
7	06-Jan-2022	0.3750	4.6663	
8	07-Jan-2022	-3.3000	3.4663	
9	08-Jan-2022	-3.0750	3.2887	
10	09-Jan-2022	-1.6250	3.8660	
11	10-Jan-2022	0.4500	4.3553	
12	11-Jan-2022	-1.6500	3.2887	
13	12-Jan-2022	-4.7500	3.0220	
14	13-Jan-2022	0.2250	4.5773	
	⋮			

```

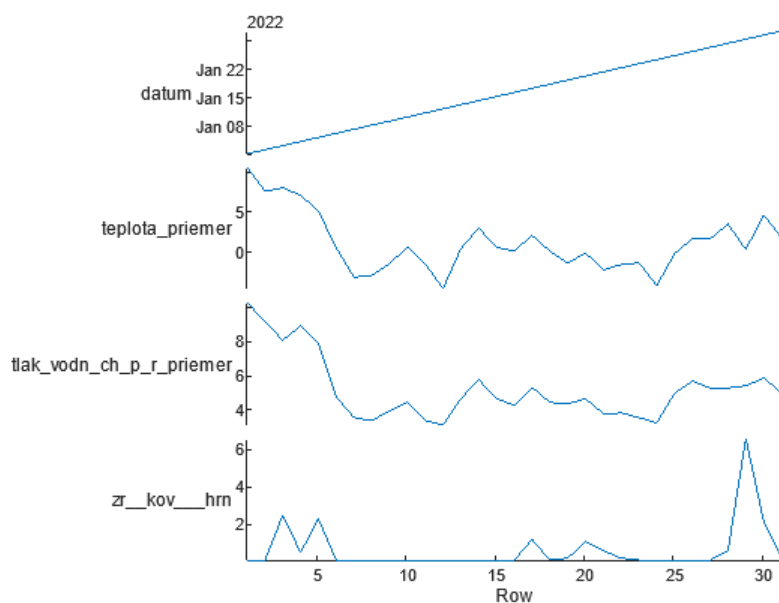
weatherData(1,:) = [];
if isempty(weatherData)
    error("No data available for " + datestr(d1,"yyyy-MMM-dd"))
end

```

Vizualizácia údajov

Teraz si vizualizujeme údaje pochádzajúce z vybranej meteorologickej stanice.

```
stackedplot(weatherData);
```



Vyhľadanie údajov s teplotami

Potrebuje vypočítať teplotné extrémny pre určitý deň, ale vieme, že v hlásených údajoch je určité množstvo nepresností. Na vyhľadanie a vyčistenie údajov o teplote môžeme použiť úlohu *Smooth Data Live Editor*.

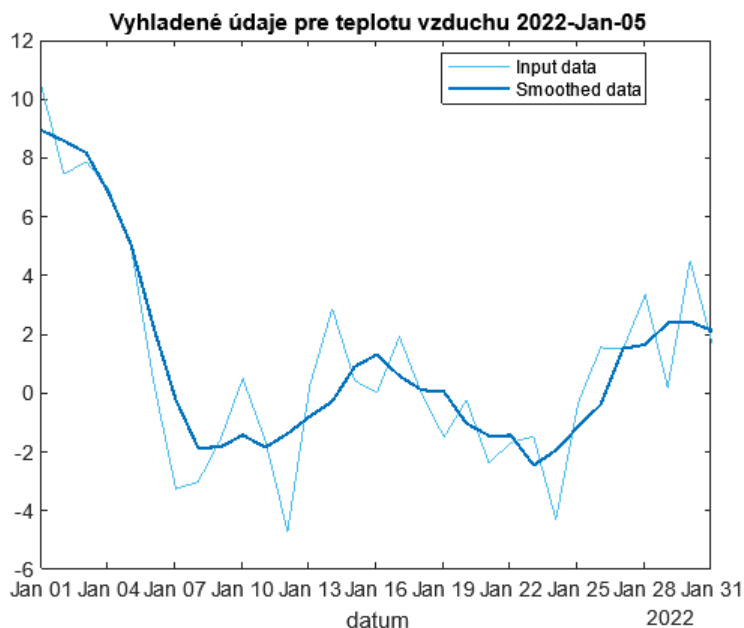
```

% Smooth input data
vyhladene_data = smoothdata(weatherData.teplota_priemer,...
    "movmean", "SmoothingFactor", 0.25, "SamplePoints", weatherData.datum);

% Display results
figure
plot(weatherData.datum, weatherData.teplota_priemer, "Color", [77 190 238]/255,...
    "DisplayName", "Input data")
hold on
plot(weatherData.datum, vyhladene_data, "Color", [0 114 189]/255, "LineWidth", 1.5,...
    "DisplayName", "Smoothed data")
hold off
legend
xlabel("datum")

```

```
xlim(weatherData.datum([1, end]))
title("Vyhladené údaje pre teplotu vzduchu " + datestr(d1,"yyyy-mmm-dd"))
legend("Location", "best")
```



Výpočet denných teplotných extrémov

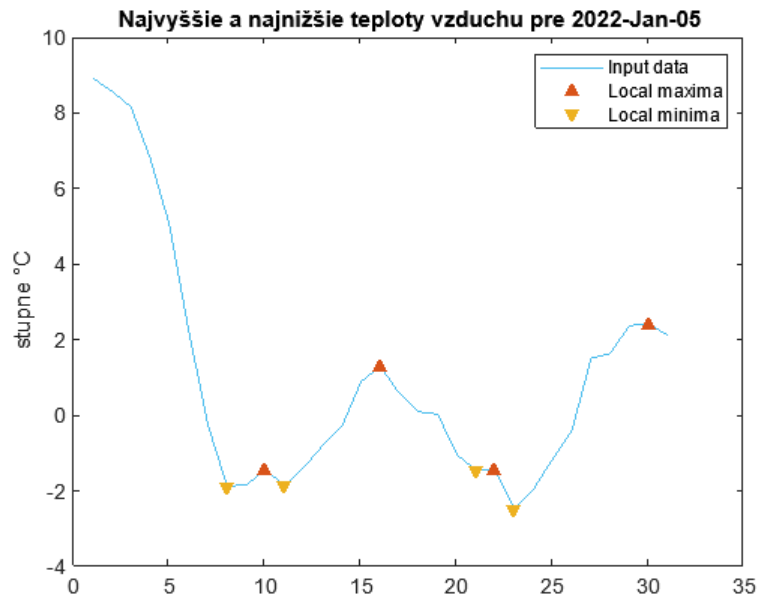
Teraz môžeme nájsť lokálne minimum a maximum v údajoch použitím **Find Local Extrema (Nájdí lokálny extrém)** na Live Editor Task

```
% Find local maxima and minima
maxIndices = islocalmax(vyhladene_data);
minIndices = islocalmin(vyhladene_data);

% Display results
figure
plot(vyhladene_data, "Color", [77 190 238]/255, "DisplayName", "Input data")
hold on

% Plot local maxima
plot(find(maxIndices), vyhladene_data(maxIndices), "^", "Color", [217 83 25]/255, ...
     "MarkerFaceColor", [217 83 25]/255, "DisplayName", "Local maxima")

% Plot local minima
plot(find(minIndices), vyhladene_data(minIndices), "v", "Color", [237 177 32]/255, ...
     "MarkerFaceColor", [237 177 32]/255, "DisplayName", "Local minima")
title("Number of extrema: " + (nnz(maxIndices)+nnz(minIndices)))
hold off
legend
legend("Location", "best")
title("Najvyššie a najnižšie teploty vzduchu pre " + datestr(d1,"yyyy-mmm-dd"))
ylabel("stupne °C")
```



Výpočet minimálnych a maximálnych teplôt a najväčšej teplotnej zmeny za deň

Môžeme dokončiť našu analýzu vypočítaním maximálnej, minimálnej a najväčšej teplotnej zmeny v priebehu sledovaného obdobia.

```

minTemp = min(vyhladene_data);
maxTemp = max(vyhladene_data);
tempChanges = diff(vyhladene_data(maxIndices | minIndices));
[~,idx] = max(abs(tempChanges));
maxChange = tempChanges(idx);

str = sprintf("Maximum Temperature = %4.1f °C\n", maxTemp);
str = [str sprintf("Minimum Temperature = %4.1f stupne °C\n", minTemp)];
str = [str sprintf("Largest Change = %4.1f stupne °C\n", maxChange)];
fprintf("%s", str)

```

Maximum Temperature = 8.9 stupne °C

Minimum Temperature = -2.5 stupne °C

Largest Change = 4.9 stupne °C

Copyright 2019 The MathWorks, Inc.

Problém 5: Nástup a ukončenie definovaných teplôt

Tento problém je k dispozícii na precvičenie v príslušnom [MS Tím](#).

Zadanie: Zdefinujte dátum nástupu a ukončenia vegetačného obdobia stanoveného číselnou hodnotou teplôt:

❖ Úlohu riešime podľa nasledovného metodického postupu:

Tabuľka 1. Priemerné mesačné teploty vzduchu v °C, Nitra, 1951-2000

Mesiac	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XI
T (°C)	-1.4	0.5	4.8	10.4	15.2	18.3	20.0	19.7	15.5	10.2	4.6	0.5

Nástup a ukončenie definovaných teplôt sa vypočíta podľa vzorcov 5 a 6.

$$\text{Nástup teplôt: } r_v = R \frac{T_n - T_2}{T_1 - T_2} \quad (5)$$

$$\text{Ukončenie teplôt: } r_p = R \frac{T_1 - T_u}{T_1 - T_2} \quad (6)$$

kde:

T_n - teplota nástupu v °C,

T_u - teplota ukončenia v °C,

T_1 - najbližšia priemerná mesačná teplota nad teplotou nástupu alebo ukončenia v °C,

T_2 - najbližšia priemerná mesačná teplota pod teplotou nástupu alebo ukončenia v °C,

R - rozdiel v dňoch medzi stredmi mesiacov s priemernou teplotou T_2 a priemernou teplotou T_1 , možno ho vyjadriť priemerným číslom, $R = 30$;

r_v - rozdiel v dňoch medzi stredom mesiaca s teplotou T_2 a dátumom nástupu teploty T_n ;

r_p - rozdiel v dňoch medzi stredom mesiaca s teplotou T_1 a dátumom ukončenia teploty T_u .

Príklad výpočtu nástupu, ukončenia a trvania teplôt:

0.0 °C: (pre tento výpočet je potrebná nasledovná výmena vzorcov: $r'_v = r_p$ a $r'_p = r_v$)

$$r'_v = 30 \frac{0.5 - 0.0}{0.5 + 1.4} = 8$$

$$r'_p = 30 \frac{0.0 + 1.4}{0.5 + 1.4} = 22$$

5.0 °C

$$r_v = 30 \frac{5.0 - 4.8}{10.4 - 4.8} = 1$$

$$r_p = 30 \frac{10.2 - 5.0}{10.2 - 4.6} = 28$$

10.0 °C

$$r_v = 30 \frac{10.0 - 4.8}{14.4 - 4.8} = 28$$

$$r_p = 30 \frac{10.2 - 10.0}{10.2 - 4.6} = 1$$

15.0 °C

$$r_v = 30 \frac{15.0 - 10.4}{15.2 - 10.4} = 29$$

$$r_p = 30 \frac{15.5 - 15.0}{15.5 - 10.2} = 3$$

Teplota 0.0 °C a menej

Dátum nástupu teploty 0.0 °C a menej možno vypočítať, ak k strednému dňu mesiaca s teplotou $T_1 = 0.5^\circ\text{C}$, t. j. k 15. decembru sa pripočíta 8 dní. Podľa tabuľky poradových čísel dní v roku (priložená [Tabuľka 1 – problém 5](#)) je 15. november 349-ty deň v roku. K tomuto dňu sa pripočíta 8 dní ($349 + 8 = 357$). To znamená, že deň nástupu $\leq 0.0^\circ\text{C}$ teploty pripadá na 357 deň, t. j. 23. december.

Dátum ukončenia teploty 0.0 °C a menej sa stanoví pripočítaním 22 dní k 15. januáru ($15 + 22 = 37$). Ukončenie teploty $\leq 0.0^\circ\text{C}$ pripadá na 38. deň, t.j. na 6. február.

Obdobie s teplotou $\leq 0.0^\circ\text{C}$ trvá 44 dní.

Teplota 5.0 °C a viac:

nástup: 15. 3. + 1 = 16. 3.

ukončenie: 15. 10. + 28 = 13. 11.

trvanie: 243 dní

Teplota 10.0 °C a viac:

nástup: 15. 3. + 28 = 13. 4.

ukončenie: 15. 10. + 1 = 16. 10.

trvanie: 187 dní

Teplota 15.0 °C a viac:

nástup: 15. 4. + 29 = 14. 5.

ukončenie: 15. 9. + 3 = 18. 9.

trvanie: 127 dní

Z uvedených údajov možno zostaviť ročnú postupnosť zmien intervalov s definovanými teplotami:

6.2. – 16. 3.	obdobie s $T \geq 0.0 \text{ } ^\circ\text{C}$	počet dní: 38
16. 3. – 13. 4.	obdobie s $T \geq 5.0 \text{ } ^\circ\text{C}$	počet dní: 28
13.4. – 14. 5.	obdobie s $T \geq 10.0 \text{ } ^\circ\text{C}$	počet dní: 31
14. 5. – 18. 9.	obdobie s $T \geq 15.0 \text{ } ^\circ\text{C}$	počet dní: 127
18. 9. – 16. 10.	obdobie s $T \geq 10.0 \text{ } ^\circ\text{C}$	počet dní: 28
16. 10. – 13. 11.	obdobie s $T \geq 5.0 \text{ } ^\circ\text{C}$	počet dní: 28
13. 11. – 23. 12.	obdobie s $T \geq 0.0 \text{ } ^\circ\text{C}$	počet dní: 41
13. 12. – 6. 2.	obdobie s $T \leq 0.0 \text{ } ^\circ\text{C}$	počet dní: 44

Obdobie kladných teplôt s $T \geq 0.0^\circ\text{C}$ trvá 320 dní.

Veľké vegetačné obdobie s $T \geq 5.0^\circ\text{C}$ trvá 243 dní.

Hlavné vegetačné obdobie s $T \geq 10.0^\circ\text{C}$ trvá 187 dní.

Obdobie vegetačného pokoja s $T \leq 10.0^\circ\text{C}$ trvá 127 dní.

Obdobie zimy s $T \leq 0.0^\circ\text{C}$ trvá 44 dní.

➤ Na vyriešenie stanoveného [problému 5](#) vložíme do MATLABu vložíme nasledovný skript:

Nástup a ukončenie definovaných teplôt

Vyčistenie obrazovky, pracovného priestoru a zatvorenie otvorených grafov.

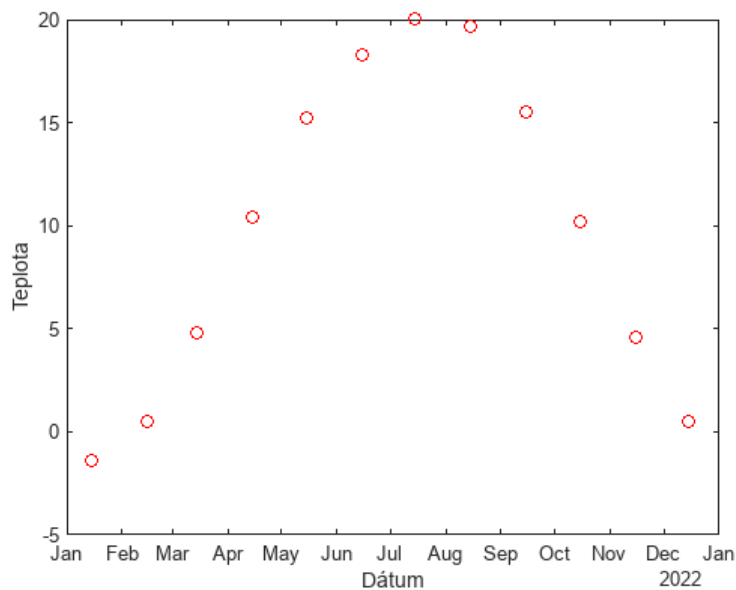
```
clear
clc
close all
```

Vytvorenie časového vektora, teplotných dát a časovej tabuľky.

```
% dáta musa byť v stĺpcoch
Date = datetime(2022,1:12,15)';
Temp = [-1.4;0.5;4.8;10.4;15.2;18.3;20.0;19.7;15.5;10.2;4.6;0.5];
originalT = timetable(Date,Temp);
```

Vykreslenie hodnôt na grafe.

```
plot(originalT.Date,originalT.Temp,'ro');
xlabel("Dátum");
ylabel("Teplota");
```



Vytvorenie novej časovej rady - kalendárny rok po dňoch.

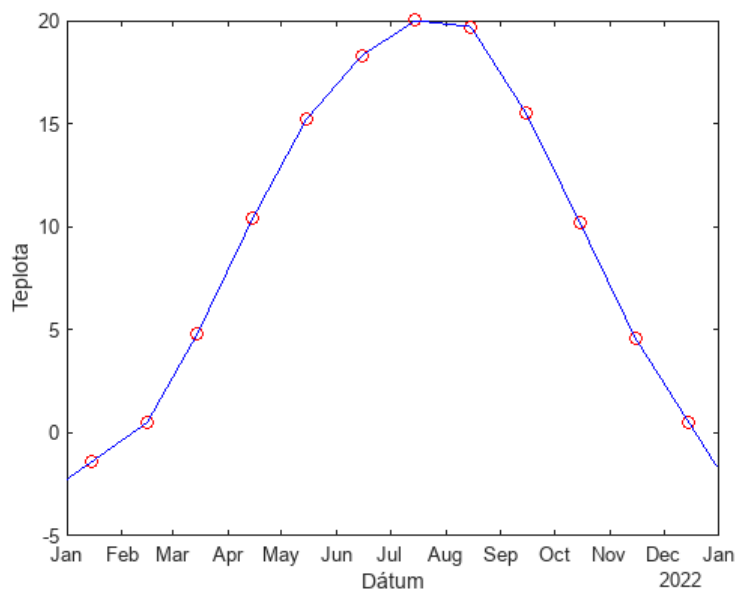
```
newTimes = datetime(2022,01,01):days(1):datetime(2022,12,31);
```

Prevzorkovanie dát lineárnou interpoláciou. Body mimo rozsahu budú tiež extrapolované lineárne.

```
interpT = retime(originalT,newTimes,'linear');
```

Doplňenie grafu o nové hodnoty.

```
hold on
plot(interpT.Date,interpT.Temp,'b');
```



Vytvorenie začiatočného a koncového času, v ktorom sa budú dáta vyhodnocovať.

```
startDen=31;
startMesiac=1;
startRok=2022;
%
endDen=31;
endMesiac=12;
endRok=2022;
%
startDT=datetime(startRok,startMesiac,startDen);
endDT=datetime(endRok,endMesiac,endDen);
```

Vytvorenie časového intervalu a výber podtabuľky

```
TR = timerange(startDT,endDT,'closed');
vyberT = interpT(TR,:);
if height(vyberT)==0
    error("Vo zvolenom intervale nie su ziadne dáta!");
end
teplota=5;
znamienko=">";
idx=eval(strcat("vyberT.Temp",znamienko,num2str(teplota)));
if ~any(idx)
    error("Podmienke nezodpovedajú žiadne dni!");
end
```

Výsledné intervaly

```
pocetDniCelkovo=sum(idx);  
disp(strcat("Počet dní zodpovedajúcich podmienke: ",num2str(pocetDniCelkovo)));
```

Počet dní zodpovedajúcich podmienke: 241

Najdlhší interval

```
[startIdx,dlзка,restIdx,restDlзка]=getMaxInterval(idx);  
disp(strcat("Nástup: ",string(vyberT.Date(startIdx))));
```

Nástup: 17-Mar-2022

```
disp(strcat("Ukončenie: ",string(vyberT.Date(startIdx+dlзка-1))));
```

Ukončenie: 12-Nov-2022

```
disp(strcat("Počet dní najdlhšie neprerušovaných: ",num2str(dlзка));
```

Počet dní najdlhšie neprerušovaných: 241

Ďalšie intervaly v dátach

```
if isempty(restIdx)  
    warning("Ďalšie intervaly sa v dátach neobjavili.");  
end
```

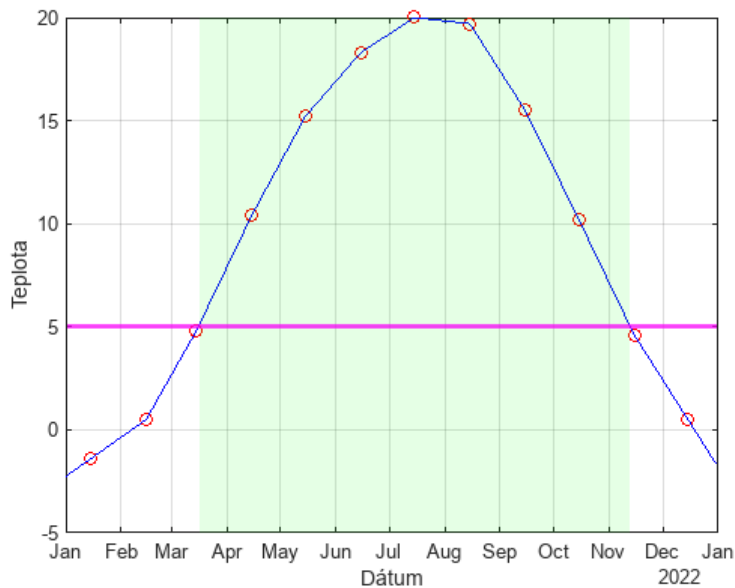
Warning: Ďalšie intervaly sa v dátach neobjavili.

```
for ii=1:length(restIdx)  
    disp(strcat("==== Interval: ",num2str(ii)," . ====="));  
    disp(strcat("Nástup: ",string(vyberT.Date(restIdx(ii)))));  
    disp(strcat("Ukončenie: ",string(vyberT.Date(restIdx(ii)+restDlзка(ii)-1))));  
    disp(strcat("Počet dní : ",num2str(restDlзка(ii))));  
end
```

Vykreslenie intervalu na grafe (zelena – najdlhší, cyan – zvyšné)

```
yl = ylim;  
yline(teplota,'m','LineWidth',2);  
area(vyberT.Date([startIdx,startIdx+dlзка-1]),...  
    [yl(2) yl(2)],"FaceColor","green","FaceAlpha",0.1,...  
    "EdgeAlpha",0,'basevalue',yl(1));  
%  
grid on  
%  
for ii=1:length(restIdx)  
    area(vyberT.Date([restIdx(ii),restIdx(ii)+restDlзка(ii)-1]),...  
        [yl(2) yl(2)],"FaceColor","cyan","FaceAlpha",0.1,...  
        "EdgeAlpha",0,'basevalue',yl(1));
```

```
end
hold off
```



```
function [startIdx,dlzka,restIdx,restDlzka]=getMaxInterval(idx)
    vd = diff([0 idx' 0]);
    starts = find(vd == 1);
    ends = find(vd == -1);
    [dlzka, idx2] = max(ends-starts);
    startIdx = starts(idx2);
    starts(idx2)=[];
    ends(idx2)=[];
    restIdx=starts;
    restDlzka=ends-starts;
end
```

Problém 6: Výpočet počtu definovaných dní v stanovenom období

Tento problém je k dispozícii na precvičenie v príslušnom [MS Tím](#).

Zadanie: Vypočítajte výskyt nižšie uvedených počtov dní, ak sú definované nasledovne:

- ❖ Úlohu riešime podľa nasledovného metodického postupu:
 - *mrazových dní* – kedy minimálna teplota bola nižšia ako 0 °C;
 - *ľadových dní* – kedy maximálna teplota bola nižšia ako 0 °C;
 - *letných dní* – maximálna teplota je vyššia ako 25 °C;
 - *tropických dní* – maximálna teplota je vyššia ako 30 °C;
 - *super tropických dní* – maximálna teplota je vyššia ako 35 °C;
 - *letných nocí* – minimálna teplota je vyššia ako 15 °C;

- *tropických nocí* – minimálna teplota je vyššia ako 20 °C;
 - *super tropických nocí* – minimálna teplota je vyššia ako 25 °C.
- Pre riešenie [problému 6](#) do MATLABu vložíme celý nasledovný skript:

Výskyt dní

Vyčistenie obrazovky, pracovného priestoru.

```
clear
clc
```

Načítanie dát zo súboru na disku. Upozornenie hovorí o tom, že názvy stĺpcov obsahujú znaky, ktoré MATLAB nemôže použiť pri pomenovaní premenných stĺpcov.

```
data=readtable("meteodata_1991_1995.xlsx");
```

Warning: Column headers from the file were modified to make them valid MATLAB identifiers before creating variable names for the table. The original column headers are saved in the VariableDescriptions property.

Set 'VariableNamingRule' to 'preserve' to use the original column headers as table variable names.

ID staníc je z uzavretej množiny dát a môžeme ich previesť na kategorické dáta. Z kategorických dát vieme rýchlo zistiť jedinečné položky. Alternatíva je využitie príkazu *unique*.

```
data.ind_kli=categorical(data.ind_kli);
categID=categories(data.ind_kli);
```

Roky sú tiež z uzavretej množiny dát a môžeme ich previesť na kategorické dáta.

```
data.rok=categorical(data.rok);
categRok=categories(data.rok);
```

Vytvoríme prázdnu tabuľku, do ktorej budeme dávať počty dní po jednotlivých zariadeniach a rokoch v cykloch.

```
dataTemp = cell2table(cell(0,10), 'VariableNames', {'ID', 'Rok', 'MrazD',...
'LadD', 'LetD', 'TropD', 'SupTropD', 'LetN', 'TropN', 'SupTropN'});
for ii=1:length(categRok)
    for jj=1:length(categID)
        % vyber podtabulky
        subTable=data(data.ind_kli==categID{jj} & data.rok==categRok{ii},:);
        if ~isempty(subTable)
            % naplnenie struktury udajmi
            subData.ID=str2double(categID{jj});
            subData.Rok=str2double(categRok{ii});
            subData.MrazD=sum(subTable.t_min__C<0);
            subData.LadD=sum(subTable.t_max__C<0);
```

```

subData.LetD=sum(subTable.t_max__C>25);
subData.TropD=sum(subTable.t_max__C>30);
subData.SupTropD=sum(subTable.t_max__C>35);
subData.LetN=sum(subTable.t_min__C>15);
subData.TropN=sum(subTable.t_min__C>20);
subData.SupTropN=sum(subTable.t_min__C>25);
% pridanie nového riadka
dataTemp=[dataTemp;struct2table(subData)];
end
end
end

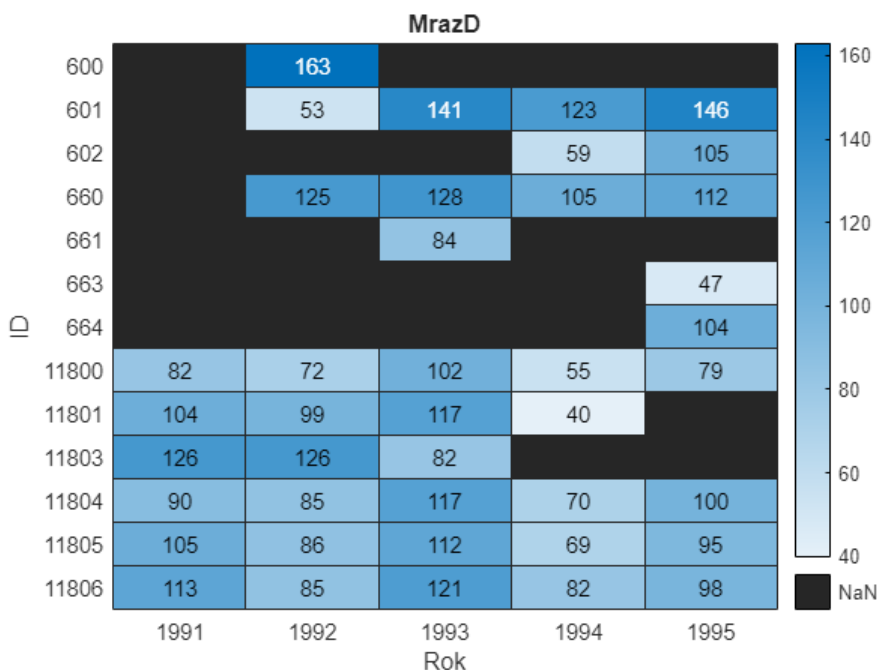
```

Porovnanie mrazových dní vybraných staníc za jednotlivé roky (pre 125 staníc bude graf neprehľadný). Čierne bunky znamenajú, že za ten rok neboli žiadne dáta vo vstupnom súbore. Môže sa použiť, aby tam neboli nulové hodnoty, to ale nezodpovedá realite.

```

dataGraph=dataTemp(dataTemp.ID<11809,:);
%heatmap(dataGraph,'Rok','ID','ColorVariable','MrazD','ColorMethod','sum');
heatmap(dataGraph,'Rok','ID','ColorVariable','MrazD','ColorMethod','none');

```



Výslednú tabuľku staníc po rokoch môžeme s teplotami zapísať do nového Excel súboru.

```
writetable(dataTemp,'PoctyDni.xlsx')
```

Problém 7: Hodnotenie suchých a zrážkových periód podľa časových období

Tento problém je k dispozícii na precvičenie v príslušnom [MS Tím](#).

Zadanie: Vyhodnoťte suché a zrážkové obdobia (periódy) podľa jednotlivých definícií:

❖ **Hodnotenie suchých períód uskutočníme za nasledovného predpokladu:**

Suchou períódou sa rozumie obdobie bez zrážok s trvaním 5 a viac dní, pričom períodu sucha neprerušia dni so zrážkami nižšími ako 1 mm.

Suché obdobia sú rozdelené do 3 skupín:

- 5 dní a viac;
- 10 dní a viac;
- 20 dní a viac.

Výpočet uskutočníme za nasledovného predpokladu:

- Ak sucho trvalo najmenej 15 dní, počas ktorých spadlo ≤ 0.9 mm zrážok, (pod 1 mm aj dni aj intervaly);
- Ak sucho trvalo najmenej 20 dní, počas ktorých spadlo ≤ 2.4 mm zrážok;
- Ak sucho trvalo najmenej 30 dní, počas ktorých spadlo ≤ 4.9 mm zrážok.

❖ **Hodnotenie zrážkových períód podľa časových období predpokladá nasledovné:**

V našich klimatických podmienkach sa zrážkové períody hodnotia aj podľa časové obdobie (týždňov, mesiacov, rokov), v ktorom spadne stanovený zrážkový limit:

- a) 5-9 za sebou nasledujúcich dní so zrážkami;
- b) 10-14 za sebou nasledujúcich dní (i vtedy, ak boli prerušené 1. dňom bez zrážok);
- c) 15-19 dní s možnosťou prerušenia dvoch za sebou nasledujúcich dní bez zrážok;
- d) 20 a viac s možnosťou prerušenia troch za sebou nasledujúcich dní bez zrážok.

➤ Pre riešenie [problému 7](#) do MATLABu vložíme celý nasledovný skript:

Hodnotenie períód

Vyčistenie obrazovky, pracovného priestoru.

```
clear  
clc
```

Načítanie dát zo súboru. Kód sa dá vytvoriť pomocou nástroja Import tool.

```
% Set up the Import Options and import the data
opts = spreadsheetImportOptions("NumVariables", 18);
% Specify sheet and range
% opts.Sheet = "denné";
% opts.DataRange = "A3:R35651";
opts.DataRange = "A3";
% Specify column names and types
opts.VariableNames = ["ind_kli", "rok", "mes", "den", "t_max", "t_min", "t_den_pr",
"vlh_rel_pr", "vie_smer7", "vie_rych7", "vie_smer14", "vie_rych14", "vie_smer21",
"vie_rych21", "vie_rych_pr", "zra_uhrn", "voda_vypar", "tlak_vod_par_pr"];
opts.VariableTypes = ["double", "double", "double", "double", "double", "double", "double",
"double", "double", "double", "double", "double", "double", "double", "double",
"double", "double"];
% Import the data
data = readtable("C:\Users\blaho\Desktop\MeteoData\denne_rok_2009.xlsx", opts,
"UseExcel", false);
```

Prázdné riadky nahradíme 0, vyberieme relevantné stĺpce a skonvertujeme na časovú tabuľku.

```
data=fillmissing(data,'constant',0);
dataSmall=data(:,{'ind_kli','den','mes','rok','zra_uhrn'});
dataSmall.Time=datetime(dataSmall.rok,dataSmall.mes,dataSmall.den);
dataTime=table2timetable(dataSmall);
clear data dataSmall opts
```

Konverzia stĺpca na kategorické dáta. Vyber kategórií.

```
dataTime.ind_kli=categorical(dataTime.ind_kli);
categID=string(categories(dataTime.ind_kli));
```

Výber zariadenia a dát na základe kategórie

```
ID=categID(1);
dataSub=dataTime(dataTime.ind_kli==ID,:);
if height(dataSub)<365
    % 665 nemalo plný rok
    error("Chýbajúce data v roku!");
end
```

Hodnotenie suchých období na základe limitu

```
pocetDni=10;
idx=dataSub.zra_uhrn<1; % 1mm
[startIdx,dlzska]=getIntervals(idx,pocetDni);
if isempty(startIdx)
    warning(sprintf("Nenasli sa suche periody s dĺžkou %d dní a viac!",pocetDni));
```

```

else
    fprintf("Celkový počet dní: %d \n",sum(dlzka));
    disp("Nájdene intervaly:");
    for ii=1:length(startIdx)
        disp("=====")
        disp(strcat("Nástup: ",string(dataSub.Time(startIdx(ii)))));
        disp(strcat("Ukončenie: ",string(dataSub.Time(startIdx(ii)+dlzka(ii)-1))));
        disp(strcat("Počet dní : ",num2str(dlzka(ii))));
    end
end

```

Celkový počet dní: 66

Nájdene intervaly:

=====

Nástup: 31-Mar-2009

Ukončenie: 10-Apr-2009

Počet dní : 11

=====

Nástup: 19-Apr-2009

Ukončenie: 03-May-2009

Počet dní : 15

=====

Nástup: 08-Jul-2009

Ukončenie: 17-Jul-2009

Počet dní : 10

=====

Nástup: 12-Aug-2009

Ukončenie: 21-Aug-2009

Počet dní : 10

=====

Nástup: 18-Sep-2009

Ukončenie: 07-Oct-2009

Počet dní : 20

```

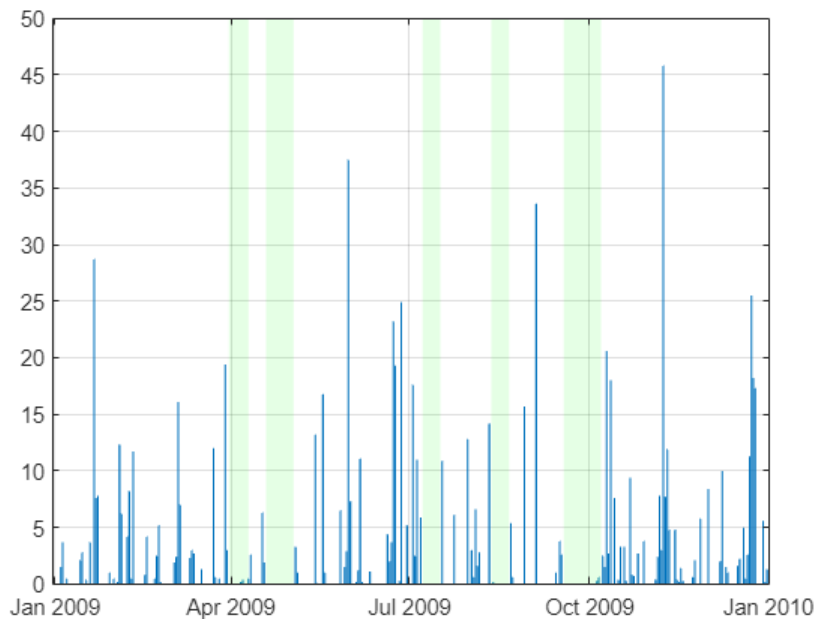
figure(1)
bar(dataSub.Time,dataSub.zra_uhrn);
grid on
hold on

```

```

yl = ylim;
for ii=1:length(startIdx)
    area(dataSub.Time([startIdx(ii),startIdx(ii)+dlzka(ii)-1]),...
        [yl(2) yl(2)],"FaceColor","green","FaceAlpha",0.1,...
        "EdgeAlpha",0,'basevalue',yl(1));
end
hold off

```



Hodnotenie suchých období na základe kumulatívnej sumy

Kritérium môže viesť v určitých situáciách na prekryvajúce sa obdobia! Skript nájde všetky, ktoré sa končia v iný deň.

```

pocetDni=10;
limit=1; % [mm]
[startIdx,dlzka]=getCumsum(dataSub.zra_uhrn',pocetDni,limit);
if isempty(startIdx)
    warning(sprintf("Nenasli sa suche periody s dĺžkou %d dní a viac!",pocetDni));
else
    fprintf("Celkový počet dní: %d \n",sum(dlzka));
    disp("Nájdene intervaly:");
    for ii=1:length(startIdx)
        disp("=====")
        disp(strcat("Nástup: ",string(dataSub.Time(startIdx(ii)))));
        disp(strcat("Ukončenie: ",string(dataSub.Time(startIdx(ii)+dlzka(ii)-1))));
        disp(strcat("Počet dní : ",num2str(dlzka(ii))));
    end
end
end

```

Celkový počet dní: 65

Nájdené intervaly:

=====

Nástup: 31-Mar-2009

Ukončenie: 09-Apr-2009

Počet dní : 10

=====

Nástup: 19-Apr-2009

Ukončenie: 03-May-2009

Počet dní : 15

=====

Nástup: 08-Jul-2009

Ukončenie: 17-Jul-2009

Počet dní : 10

=====

Nástup: 12-Aug-2009

Ukončenie: 21-Aug-2009

Počet dní : 10

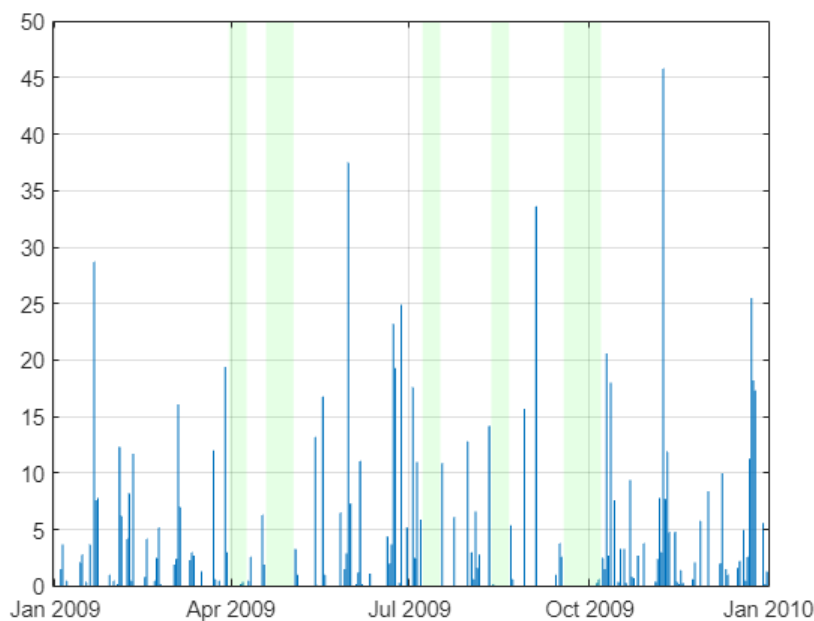
=====

Nástup: 18-Sep-2009

Ukončenie: 07-Oct-2009

Počet dní : 20

```
figure(2)
bar(dataSub.Time,dataSub.zra_uhrn);
grid on
hold on
yl = ylim;
for ii=1:length(startIdx)
    area(dataSub.Time([startIdx(ii),startIdx(ii)+dĺzka(ii)-1]),...
        [yl(2) yl(2)],"FaceColor","green","FaceAlpha",0.1,...
        "EdgeAlpha",0,'basevalue',yl(1));
end
hold off
```



Hodnotenie zrážkových periód

```
[startIdx,dlзка]=getPeriods(dataSub.zra_uhrn');
if isempty(startIdx)
    warning("Nenasli sa zrážkové periódy!");
else
    fprintf("Celkový počet dní: %d \n",sum(dlзка));
    disp("Nájdené intervaly:");
    for ii=1:length(startIdx)
        disp("=====")
        disp(strcat("Nástup: ",string(dataSub.Time(startIdx(ii)))));
        disp(strcat("Ukončenie: ",string(dataSub.Time(startIdx(ii)+dlзка(ii)-1))));
        disp(strcat("Počet dní : ",num2str(dlзка(ii))));
    end
end
```

Celkový počet dní: 35

Nájdené intervaly:

=====

Nástup: 20-Jun-2009

Ukončenie: 24-Jun-2009

Počet dní : 5

=====

Nástup: 02-Aug-2009

Ukončenie: 06-Aug-2009

Počet dní : 5

=====

Nástup: 08-Oct-2009

Ukončenie: 12-Oct-2009

Počet dní : 5

=====

Nástup: 04-Nov-2009

Ukončenie: 11-Nov-2009

Počet dní : 8

=====

Nástup: 14-Nov-2009

Ukončenie: 18-Nov-2009

Počet dní : 5

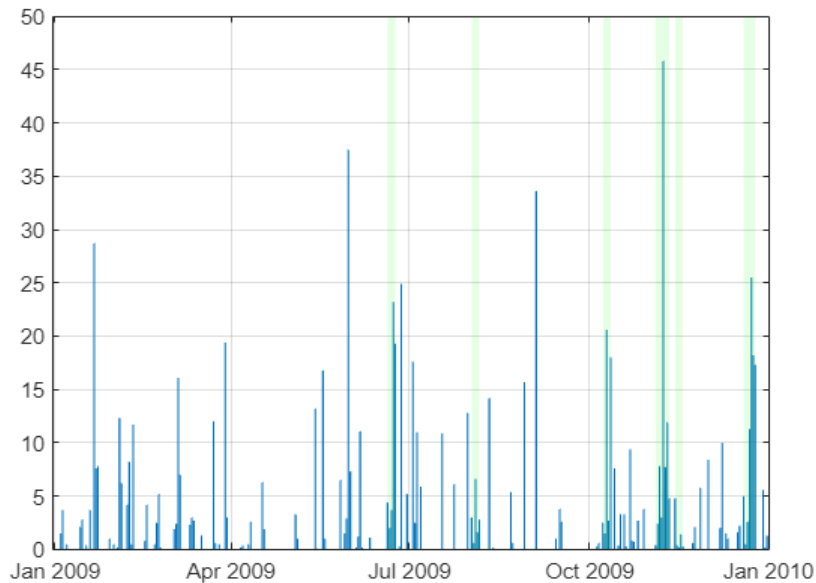
=====

Nástup: 19-Dec-2009

Ukončenie: 25-Dec-2009

Počet dní : 7

```
figure(3)
bar(dataSub.Time,dataSub.zra_uhrn);
grid on
hold on
yl = ylim;
for ii=1:length(startIdx)
    area(dataSub.Time([startIdx(ii),startIdx(ii)+dlzka(ii)-1]),...
        [yl(2) yl(2)],"FaceColor","green","FaceAlpha",0.1,...
        "EdgeAlpha",0,'basevalue',yl(1));
end
hold off
```



```

function [starts,dlzka]=getIntervals(idx,pocetDni)
vd = diff([0 idx' 0]);
starts = find(vd == 1);
ends = find(vd == -1);
dlzka=ends-starts;
dnidx=dlzka<pocetDni;
starts(dnidx)=[];
dlzka(dnidx)=[];
end

%%

function [starts,dlzka]=getCumsum(zrazky,pocetDni,limit)
starts=[];
ends=[];
lastEnd=0;
for ii=1:length(zrazky)-pocetDni-1
    suma=cumsum(zrazky(ii:end));
    dlzkaSub=find(suma<limit==0,1,'first');
    if dlzkaSub>pocetDni && dlzkaSub+ii-1>lastEnd
        starts=[starts ii];
        ends=[ends dlzkaSub+ii-1];
        lastEnd=dlzkaSub+ii-1;
    end
end
dlzka=ends-starts;
end

%%

function [starts,dlzka]=getPeriods(zrazky)
starts=[];

```



```

dlzka=[];
globStart=1;
globEnd=1;
flag=1;

if zrazky(1)==0
    rainCount=1;
    dryCount=0;
else
    dryCount=1;
    rainCount=0;
end
for ii=2:length(zrazky)
    globEnd=globEnd+1;
    if zrazky(ii-1)==0 && zrazky(ii)>0
        rainCount=1;
        dryCount=0;
    elseif zrazky(ii-1)>0 && zrazky(ii)>0
        rainCount=rainCount+1;
        dryCount=0;
    elseif zrazky(ii)==0
        dryCount=dryCount+1;

        if (rainCount>=5 || rainCount<=9) && dryCount>0
            flag=0;
        elseif (rainCount>=10 || rainCount<=14) && dryCount>1
            flag=0;
        elseif (rainCount>=15 || rainCount<=19) && dryCount>2
            flag=0;
        elseif rainCount>=20 && dryCount>3
            flag=0;
        end
    end
end

if flag==0 || ii==length(zrazky)
    if globEnd-globStart>=5
        starts=[starts globStart];
        dlzka=[dlzka globEnd-globStart];
        globStart=ii+1;
        globEnd=ii;
    else
        globStart=ii+1;
        globEnd=ii;
        rainCount=0;
    end
    flag=1;
end
end
end
end

```

Použitá literatúra

Abdulaziz AlHumid, H., Haider, H.S., AlSaleem, S., Alinizzi, M., Shafiquzaman, M. & Sadiq, R. (2019). Performance Assessment Model for Municipal Solid Waste Management Systems: Development and Implementation. *Environments* 2019, 6, 19. <https://doi.org/10.3390/environments6020019>

Akgun, A., Sezer, E.A., Nefeslioglu, H.A., Gokceoglu, C. & Pradhan B. (2012). An easy-to-use MATLAB program (MamLand) for the assessment of landslide susceptibility using a Mamdani fuzzy algorithm. *Computers & Geosciences* 38, 23–3. <https://doi.org/10.1016/j.cageo.2011.04.012>

Almoulki, T. & Yetilmmezsoy, K. (2019). MATLAB for Iteration: Hydraulic modeling for environmental engineering students. *Fluid Mech Res Int.* 2019;3(1):24–28. <https://doi.org/10.15406/fmrij.2019.03.00049>

Araghinejad, S. (2014). Introduction. In: *Data-Driven Modeling: Using MATLAB® in Water Resources and Environmental Engineering*. Water Science and Technology Library, vol 67. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-7506-0_1

Arizona State University (2022, November 11). *Water Resources Issues*. https://lsintspl3.wgbh.org/en-us/lesson/ilkaet18-ilwater?as_quest=True&next=https://www.pbslearningmedia.org/resource/ilkaet18-sci-ilwater/water-resource-issues/

Augustin, J.M. (2011). Developing and Deploying Sonar and Echosounder Data Analysis Software. https://www.mathworks.com/company/newsletters/articles/developing-and-deploying-sonar-and-echosounder-data-analysis-software.html?s_tid=srchtitle

Aurausp, M. (2021, January 14). Plotting basic Graphs in MATLAB 2021 I Part 1. <https://www.youtube.com/watch?v=B1gcp58ON2E>

Bhatt, C. & Papola, A. (2014). Analysis And Prediction Of Landslides in Uttarakhand Using Weka Tool. *International Journal of Engineering and Technical Research*, 2(8).

Cay, T. & Iscan, F. (2008). A New Land Reallocation Model for Land Consolidation. Integrating Generations, FIG Working Week 2008, Stockholm, Sweden. https://www.fig.net/resources/proceedings/fig_proceedings/fig2008/papers/ts08c/ts08c_03_cay_iscan_2925.pdf

Ch.mathworks.com (2022, November 27). MATLAB Plot Gallery. <https://ch.mathworks.com/products/matlab/plot-gallery.html>

Fechová, E. (2008). *Aplikácia programového prostredia matlab vo výučbe prírodovedných predmetov*. http://mat.fsv.cvut.cz/KomiseVSTEZ/30vstez/sbornik/fechova_30_vstez.pdf

Filion, A. (2022, December 1). Temperature changes during a day. MATLAB Central File Exchange. <https://www.mathworks.com/matlabcentral/fileexchange/73241-temperature-changes-during-a-day>

Fricker, P. (2014). Analyzing and Visualizing Flows in Rivers and Lakes with MATLAB. https://www.mathworks.com/company/newsletters/articles/analyzing-and-visualizing-flows-in-rivers-and-lakes-with-matlab.html?s_tid=srchtitle_water%20flow_10

Gannon, J.P. & McGuire, K.J. (2022). *An Interactive Web Application Helps Students Explore Water Balance Concepts*. *Front. Educ.* 7:873196. <https://doi.org/10.3389/feduc.2022.873196>

Gatto, M.P.A., Misiano, S. & Montrasio, L. (2022). On the Use of MATLAB to Import and Manipulate Geographic Data: A Tool for Landslide Susceptibility Assessment. *Geographies* 2022, 2, 341–353. <https://doi.org/10.3390/geographies2020022>

Han, X.-G. & Huang, T.-I. (2009). Analysis of Real-Time Water Quality Monitoring Data Based on Matlab Software. *2009 WRI World Congress on Software Engineering*, pp. 414-418, <https://doi.org/10.1109/WCSE.2009.336>

Heather Gorr (2022, December 2). *Calculate Heart Rate from Electrocardiogram Data*, MATLAB Central File Exchange. <https://www.mathworks.com/matlabcentral/fileexchange/73049-calculate-heart-rate-from-electrocardiogram-data>

Horton, P., Jaboyedoff, M., Rudaz, B. & Zimmermann, M. (2013). Flow-R, a model for susceptibility mapping of debris flows and other gravitational hazards at a regional scale, *Nat. Hazards Earth Syst. Sci.*, 13, 869-885, <https://doi.org/10.5194/nhess-13-869-2013>

Humusoft.cz (2022, December 12). *Nová verze Matlab® & Simulink® R2022b*. <https://www.humusoft.cz/events/www-seminars/>

Itnetwork.cz (2022, November 27). *Lekce 1 - Seznámení s uživatelským prostředím*. <https://www.itnetwork.cz/programovani/matlab/tutorial-matlab-seznameni-s-uzivatelskym-prostredim>

IWMI: International Water Management Institute (2021, December 8). *Agrol Landscape launches e-learning for systems thinking on food, land and water*. <https://wle.cgiar.org/news/agro-landscapes-launches-e-learning-systems-thinking-food-land-and-water>

Johnson, J.B., Anderson, J.F., Marshall, H.P., Havens, S. & Watson, L.M. (2021). Snow avalanche detection and source constraints made using a networked array of infrasound sensors. *Journal of Geophysical Research: Earth Surface*, 126, e2020JF005741. <https://doi.org/10.1029/2020JF005741>

Kalaš, I. (2000). Čo ponúkajú informačné a komunikačné technológie iným predmetom. In *Zborník konferencie Infovek 2000*. ISBN 80-7098-265-5.

Kilić Pamuković, J., Rogulj, K. & Jajac, N. (2021). Assessing the Bonitet of Cadastral Parcels for Land Reallocation in Urban Consolidation. *Land*, 2021, 10, 9. <https://doi.org/10.3390/land10010009>

Kuroda, J. (2018). Designing and Implementing Multibeam Sonar Systems with Model-Based Design. https://www.mathworks.com/company/newsletters/articles/designing-and-implementing-multibeam-sonar-systems-with-model-based-design.html?s_tid=srchtitle

Learning Vibes (2021, November 23). How to generate 3D plot in MATLAB. <https://www.youtube.com/watch?v=zeflwkJdQ9k>

Loren on the Art of MATLAB (2022, December 11). © 1994-2023 The MathWorks, Inc. <https://blogs.mathworks.com/loren/>

Maalouf, A. & El-Fadel, M. (2020). Optimizing emissions and carbon credit from integrated solid waste and wastewater management: A MATLAB-based model with a Graphical User Interface (v1). *MethodsX* 7, <https://doi.org/10.1016/j.mex.2020.100839>

Mathworks.com (2022, November 22). <https://ch.mathworks.com/>

MathWorks (2007). MATLAB and Simulink in the World: Earth, Ocean, and Planetary Sciences. https://www.mathworks.com/company/newsletters/articles/matlab-and-simulink-in-the-world-earth-ocean-and-planetary-sciences.html?s_tid=srchtitle

Mulvey, B., Ward, A. S., Jefferson, A. J., & Bales, J. (2022). *Innovations in Remote and Online Education by Hydrologic Scientists*. Lausanne: Frontiers Media SA. <https://doi.org/10.3389/fenvs.2022.1074801>

Papež, J. (2014). *Ťahák pre MATLAB*. <https://www2.karlin.mff.cuni.cz/~havelkova/TahakMatlab.pdf>

Portenier, C., Hüsler, F., Härer, S. & Wunderle, S. (2019). Towards a webcam-based snow cover monitoring network: methodology and evaluation. *The Cryosphere Discussions*, pp. 1–20.

Precise Simulation (2023). CFDTTool - MATLAB CFD Simulation GUI &Toolbox, GitHub. <https://github.com/precise-simulation/cfdtool/releases/tag/1.9.1>

Priyadarshinee, K. & Mohanty, L. (2015). Interpretation of Roughness in Compound Channel Using MATLAB. *International Research Journal of Engineering and Technology*, 2(3) 860-864.

Renuka & Radhakumari, C. (2016). Best out of waste: a prototype software solution using MATLAB -GUI. *International Journal of Management and Applied Science*, 2, 12. (https://www.iraj.in/journal/journal_file/journal_pdf/14-322-148489297617-25.pdf)

Rubin, M.J., Camp, T., Herwijnen, A.V. & Schweizer, J. (2012). Automatically Detecting Avalanche Events in Passive Seismic Data. *11th International Conference on Machine Learning and Applications*, pp. 13-20, <https://doi:10.1109/ICMLA.2012.12>

Schwanghart, W. (2017). Developing a MATLAB Toolbox for Digital Elevation Model Analysis. https://www.mathworks.com/company/newsletters/articles/developing-a-matlab-toolbox-for-digital-elevation-model-analysis.html?s_tid=srchtitle

sk.education-wiki.com (2022, November 23). *Výhody MATLABu*. <https://sk.education-wiki.com/1990704-advantages-of-matlab>

Tiwari, H., Das, P. & Bharti, A. (2012). MATLAB Programming Solution For Critical And Normal Depth In Trapezoidal Channels. *International Journal of Engineering*. 1(8) 2012.

Thielicke, W. & Sonntag, R. (2021). Particle Image Velocimetry for MATLAB: Accuracy and enhanced algorithms in PIVlab. *Journal of Open Research Software*, 9. Ubiquity Press, Ltd. <https://doi.org/10.5334%2Fjors.334>

Thilagavathi, N. & Amudha, T. (2019). A novel methodology for optimal land allocation for agricultural crops using Social Spider Algorithm. *PeerJ*. 2019 Sep 17;7:e7559. <https://doi:10.7717/peerj.7559>.

Univerzita obrany (2022, November 23). *Matlab* <https://matlab.unob.cz/layouts/15/start.aspx#/SitePages/Matlab%20CZ.aspx#JakZacit>

Urena, F., Alfaro, V.M., Arrieta, O. & Vilanova, R. (2011). MATLAB based graphical interface simulator for wastewater treatment plant control by using the benchmark BSM1. Proceedings of the IEEE Global Engineering Education Conference (EDUCON); 4-6; Amman, Jordan. <https://doi.org/10.1109/EDUCON.2011.5773196>

Wang, T.-S., Yu, T.-T., Lee, S.-T., Peng, W.-F., Lin, W.-L. & Li, P.-L. (2014). MATLAB code to estimate landslide volume from single remote sensed image using genetic algorithm and imagery similarity measurement. *Computers & Geosciences* 70 (2014) 238–247. <http://dx.doi.org/10.1016/j.cageo.2014.06.004>

Zhou, J., Bao, W., Tick, G.R., Moftakhari, H., Li, Y. & Cheng, L. (2020). Data and MATLAB Code for the paper entitled "A modified Chezy formula for one-dimensional unsteady frictional resistance in open channel flow" [Data set]. In *Journal of Fluid Engineering (ASME)* (1.0). Zenodo. <https://doi.org/10.5281/zenodo.4314728>

Zhou, J., Bao, W., Tick, G.R., Moftakhari, H., Li, Y. & Cheng, L. (2021). A Modified Chezy Formula for One-Dimensional Unsteady Frictional Resistance in Open Channel Flow. *ASME. J. Fluids Eng.* 2021; 143(5): 051303. <https://doi.org/10.1115/1.4049681>

Autorky:

Ing. Beáta Novotná, PhD., doc. Ing. Lucia Tátošová, PhD.

Ing. Tatiana Kaletová, PhD.

Názov:

Využitie MATLAB v krajinom inžinierstve

Vydavateľ: Slovenská poľnohospodárska univerzita v Nitre

Vydanie: prvé

Forma vydania: online

Rok vydania: 2022

AH – VH: 6,89 – 7,05

Neprešlo redakčnou úpravou vo Vydavateľstve SPU.

ISBN 978-80-552-2564-7