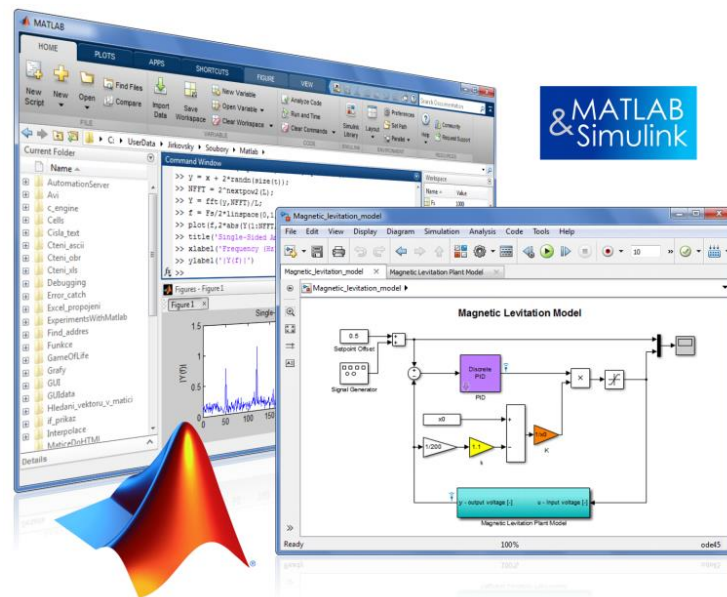


TCC 2019

Tipy a triky pro MATLAB



Proč se vyhýbat funkcím *eval*, *evalc*, *evalin*, *assignin*, či *load* (volaný bez výstupu)?

- Neočekávané přepisování proměnných
- Bezpečnostní rizika, čitelnost kódu aj.
- Rychlost výpočtu
 - automatická optimalizace kódu
- Využití výhod nástrojů jako Code Analyzer
- *Dají se nahradit jinými funkcemi, např.:*
 - *struct* (s dynamickým indexováním) / *cell array*,
 - *sprintf*, *str2func*, *feval*, ...

Proč a jak se vyhnout globálním proměnným?

- Ztrácíme přehled
 - jakákoliv funkce je může změnit
- ***Co používat místo globálních proměnných:***
 - předávání argumentů
 - persistentní proměnná

Proč se vyhýbat příkazu *clear all*?

- **Maže více než proměnné**
 - zpomalení výpočtů
 - persistentní proměnné

Value of ItemType	Items Cleared							
	Variables in scope	Scripts and functions	Class definitions	Persistent variables	MEX functions	Global variables	Import list	Java classes on the dynamic path
all	✓	✓		✓	✓	✓	From command prompt only	
classes	✓	✓	✓	✓	✓	✓	✓	
functions		✓		✓	✓			
global						✓		
import							✓	
java	✓	✓		✓	✓	✓		✓
mex					✓			
variables	✓							

➤ ***Jak mazat proměnné?:***

– *clear*

- samotný příkaz „*clear*“ maže všechny proměnné z workspace

– **používat funkce**

- mají lokální workspace

Můžu mít více funkcí v jednom souboru?

- **Lokální / Vnořené funkce**
 - lze předávat jako argumenty ve formě function handle
- **Třída obsahující pouze statické metody**
 - ukázka: *MyFcns*
 - nemusím zakládat objekt
 - `>> res = MyFcns.scitani(5,2)`

```
MyFcns.m x +
1  classdef MyFcns
2      methods(Static)
3          function z = scitani(x,y)
4              z = x + y;
5          end
6
7          function z = nasobeni(x,y)
8              z = x * y;
9          end
10
11         function z = deleni(x,y)
12             z = x / y;
13         end
14     end
15 end
16
17
```

Jak přidávat adresáře do cest MATLABu?

- **Interaktivně: „Set Path“**

- přidat cesty pro dané sezení

- „Add Folder...“

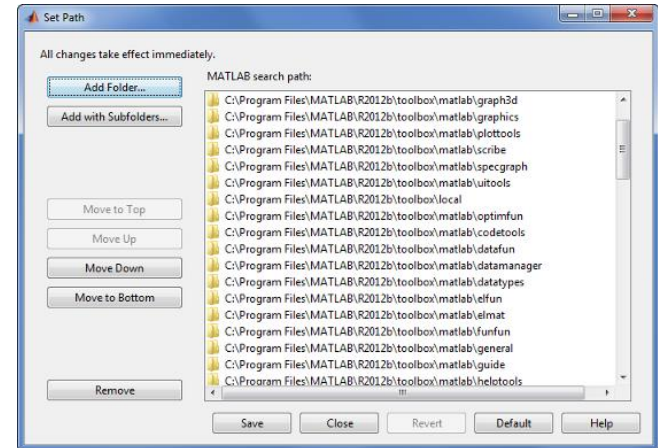
- uložit pro budoucí sezení

- „Save“

- uloží do souboru *pathdef.m*

- obsahuje cesty do instalačních adresářů dané verze MATLABu

- vytvoření nového *pathdef.m* může způsobit problémy v jiných verzích MATLABu



- **Pomocí příkazů**

- *path*, *addpath*, *savepath*

- **Pro budoucích sezení vytvořit soubor *startup.m***

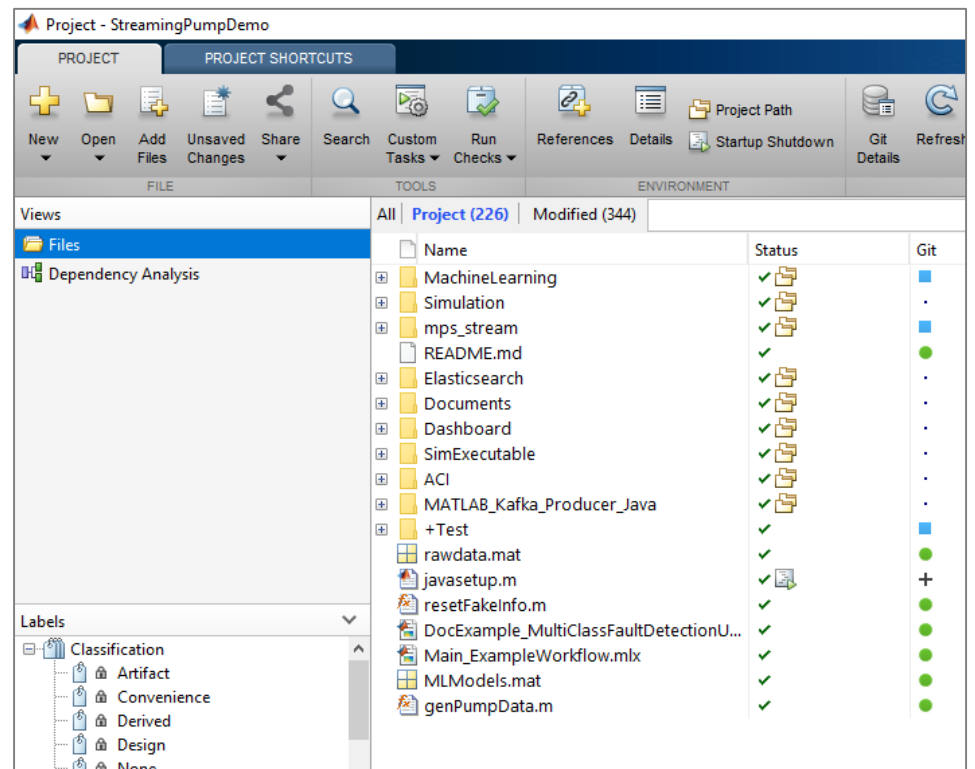
- spustí se automaticky při startu

```
startup.m x +
1 - addpath(genpath('C:/MATLAB/MujProjekt'))
```

- **MATLAB Projects**

MATLAB Projects

- Správa cest
- Startup / Shutdown tasks
- Označení souborů
- Integrace s verzovacími systémy
- Analýza závislostí



Jak pracovat s velkými daty v paměti?

- „Copy-on-Write“

```
>> A = rand(12000);
```

```
>> B = A;
```

```
>> A(1,1) = 1;
```

- Předit výraz jako vstupní argument

```
>> B = f2(rand(12000));
```

- Přiřazovat „In-Place“

```
>> A = A.*1.1;
```

```
function Y = f2(X)
    X = X.*1.1; % X is an independent copy of A
    Y = X;      % Y is a shared copy of X
end
```

- Použití vhodných datových typů

- *categorical místo cell array of strings,*

- *sparse*

- *datastore, tall array, když se data nevejdou do paměti*

Jak urychlit výpočet?

- **Proč je výpočet pomalý?**

- **checkcode**

- warnings

- **MATLAB Profiler**

- **Často výpočet urychlí:**

- **předalokace**

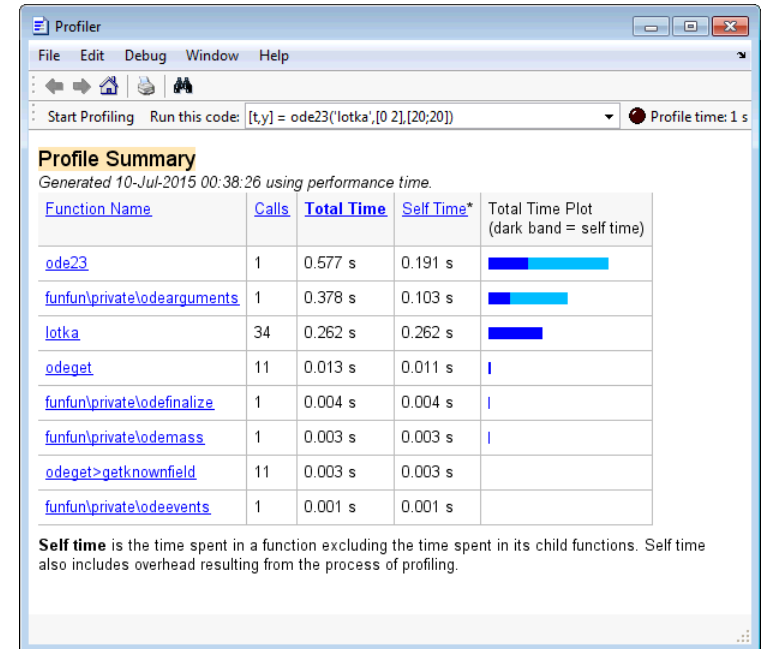
- **vektorizace**

- **minimalizace časově kritických operací**

- přístupování k disku, zobrazení výstupů

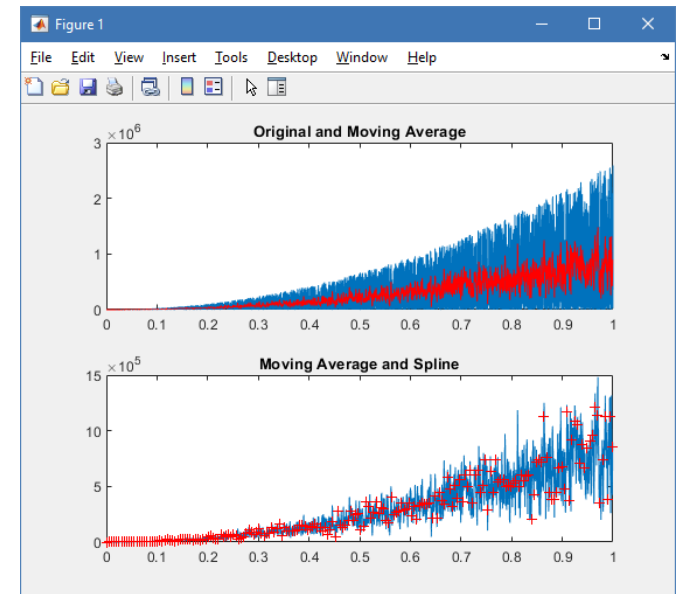
- **hardware**

- paralelní, GPU výpočty



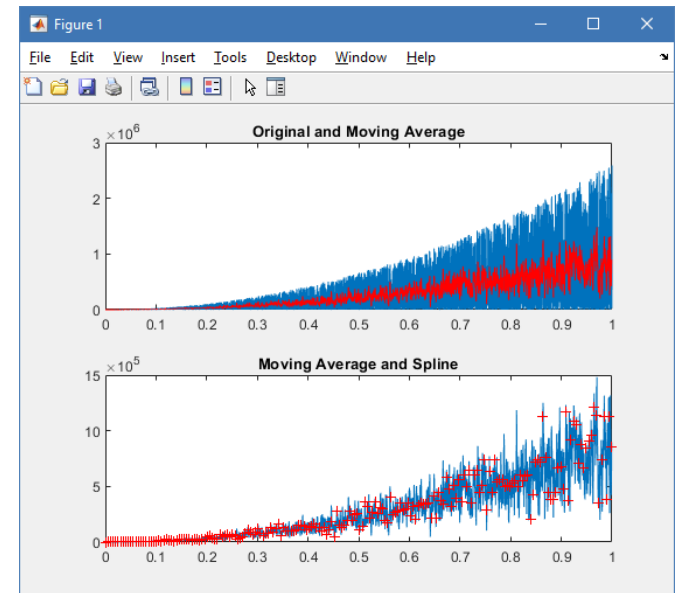
Příklad: Prokládání dat

- Načtení dat z několika souborů
- Výběr konkrétního vzorku
- Proložení dat spline funkcí
- Zápis výsledků do sešitu Microsoft Excel



Shrnutí příkladu

- K analýze kódu využít nástroj Profiler
- Odhaleny časově kritické operace
- Redukován zápis a čtení souborů
- Opětovné využívání okna figure



Co se stane, když v MATLABu zavoláme...

>> *spy*

>> *xpbombs*

>> *fifteen*

>> *why*

Primární zdroje informací

1. Dokumentace

www.mathworks.com/help

2. MATLAB Answers

www.mathworks.com/answers

Dotazy a připomínky

- ...