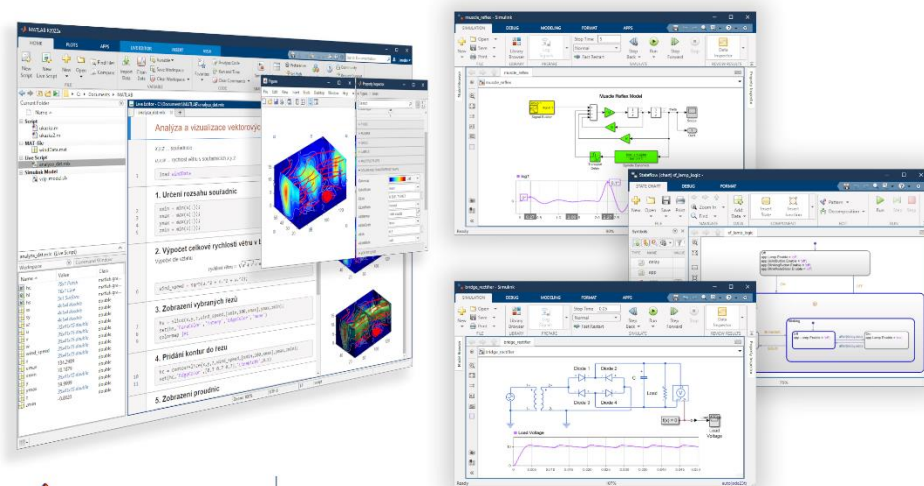


Novinky v prostředí MATLAB



Michal Blaho

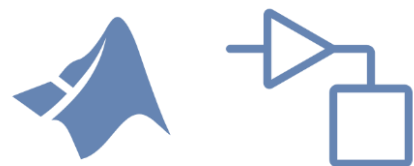
blaho@humusoft.sk

www.humusoft.cz

info@humusoft.cz

www.mathworks.com

3,744



MATLAB®
& SIMULINK®



Integrácia



AI



MATLAB®
& SIMULINK®



Integrácia



AI



Pohodnosť



Výkon



Verifikácia

Lokálne funkcie

```
x = 1:10;  
n = length(x);  
avg = mymean(x,n);
```

```
function a = mymean(v,n)  
% MYMEAN Local function  
  
    a = sum(v)/n;  
end
```

Lokálne funkcie: Definícia funkcií kdekoľvek v skripte

```
x = 1:10;  
n = length(x);
```

```
function a = mymean(v,n)  
% MYMEAN Local function
```

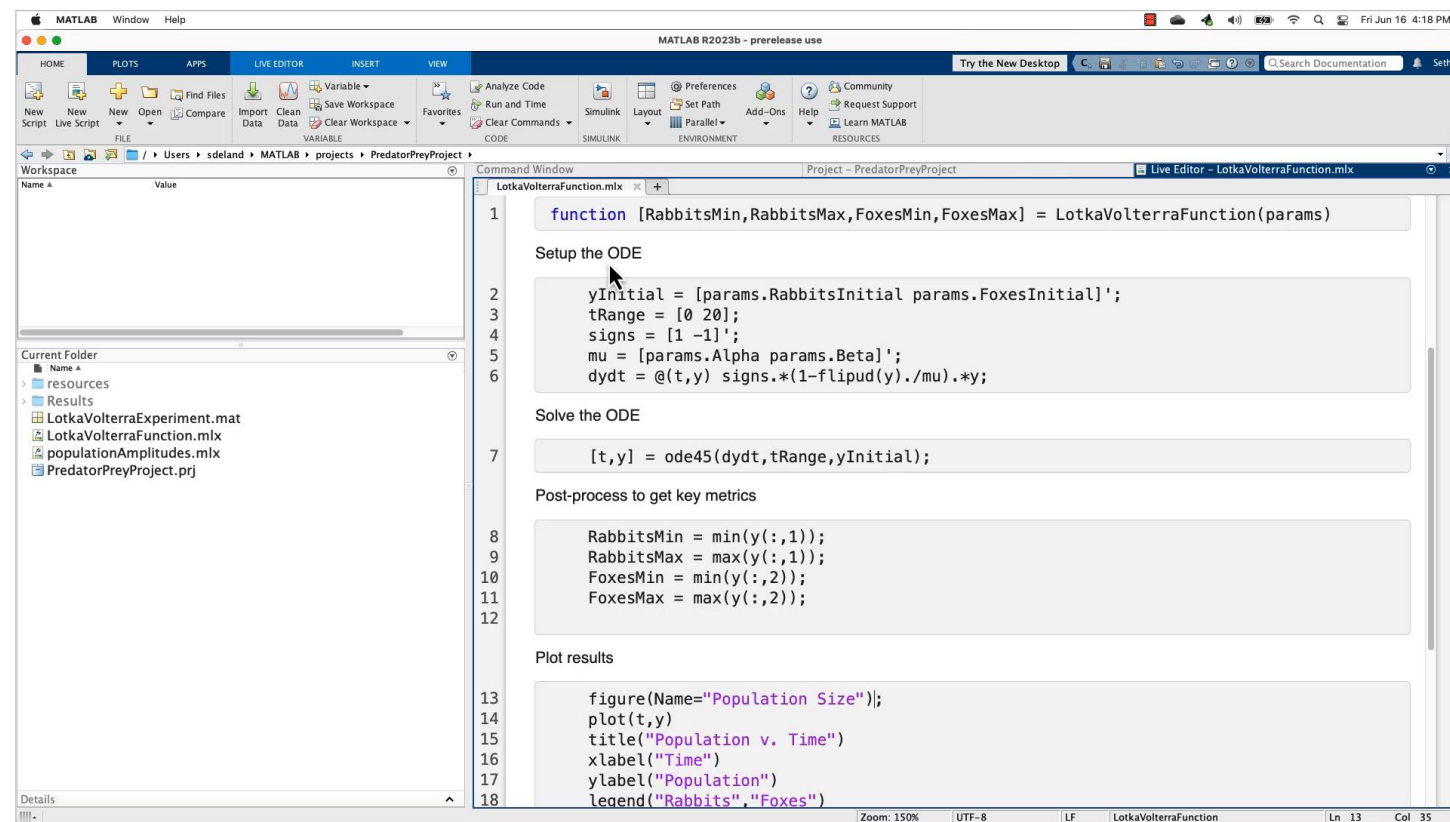
```
    a = sum(v)/n;  
end
```

```
avg = mymean(x,n);
```

Riešenie úloh – s využitím aplikácií

Tvorba **experimentov** na spúšťanie MATLAB kódu

– Vizualizácia, filtrovanie a porovnávanie výsledkov



```

1 function [RabbitsMin,RabbitsMax,FoxesMin,FoxesMax] = LotkaVolterraFunction(params)
2
3 Setup the ODE
4 yInitial = [params.RabbitsInitial params.FoxesInitial]';
5 tRange = [0 20];
6 signs = [1 -1]';
7 mu = [params.Alpha params.Beta]';
8 dydt = @(t,y) signs.*(1-flipud(y)./mu).*y;
9
10 Solve the ODE
11 [t,y] = ode45(dydt,tRange,yInitial);
12
13 Post-process to get key metrics
14 RabbitsMin = min(y(:,1));
15 RabbitsMax = max(y(:,1));
16 FoxesMin = min(y(:,2));
17 FoxesMax = max(y(:,2));
18
19 Plot results
20 figure(Name="Population Size");
21 plot(t,y)
22 title("Population v. Time")
23 xlabel("Time")
24 ylabel("Population")
25 legend("Rabbits","Foxes")
  
```

Experiment Manager App

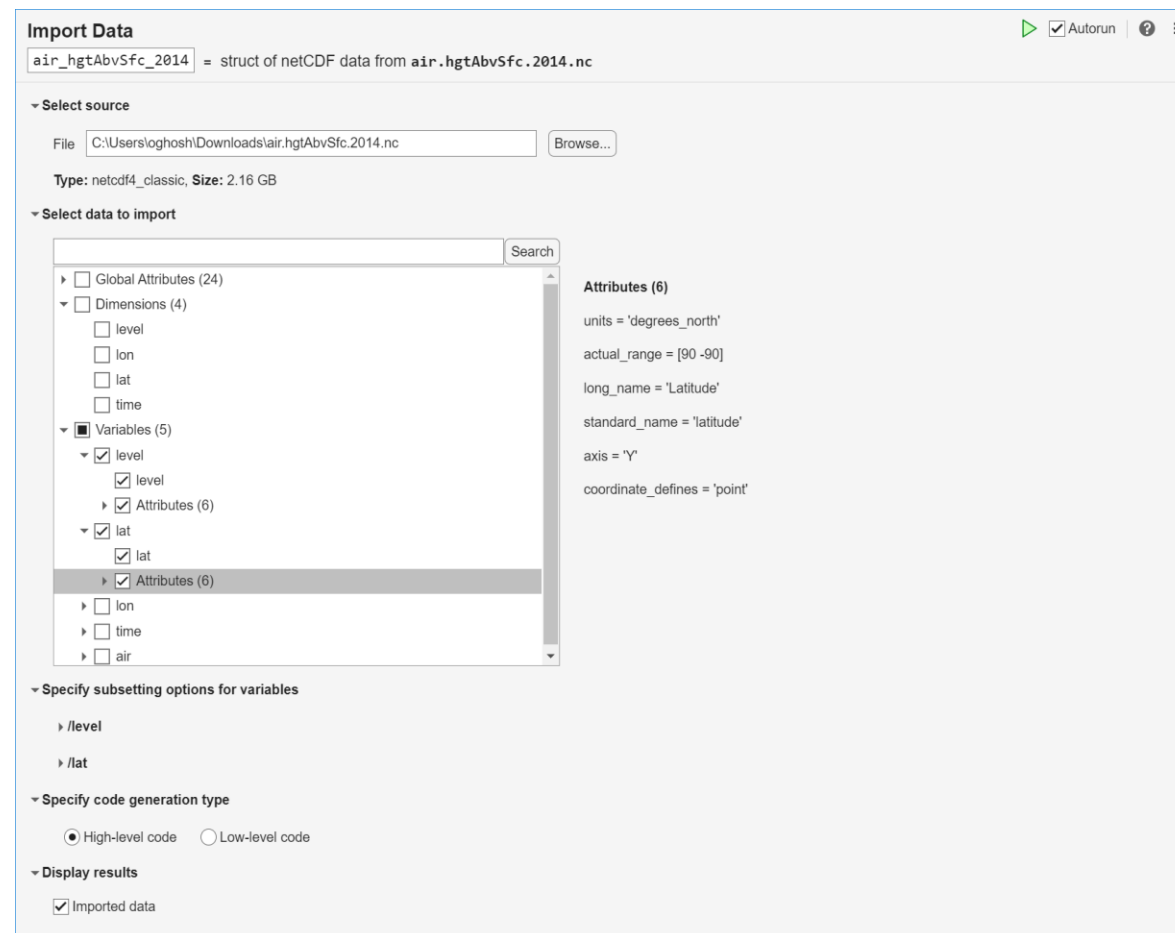
Riešenie úloh – s využitím aplikácií a Live Tasks

Tvorba experimentov na spúšťanie MATLAB kódu

- Vizualizácia, filtrovanie a porovnávanie výsledkov

Interaktívna vizualizácia a náhľad hierarchických dátových formátov

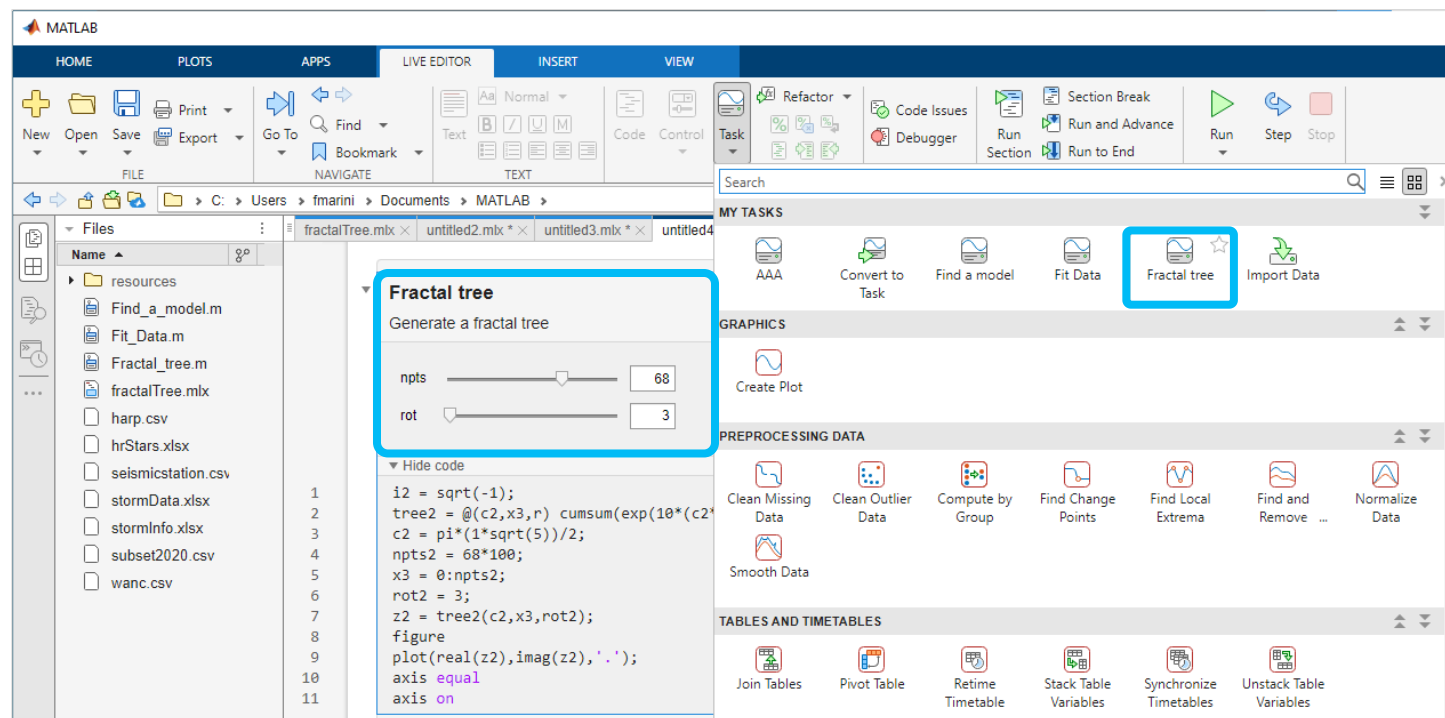
- NetCDF
- HDF5



Import Data Live Editor Task

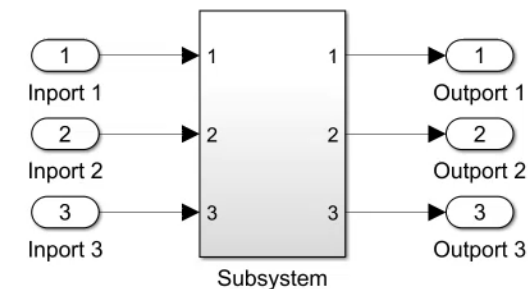
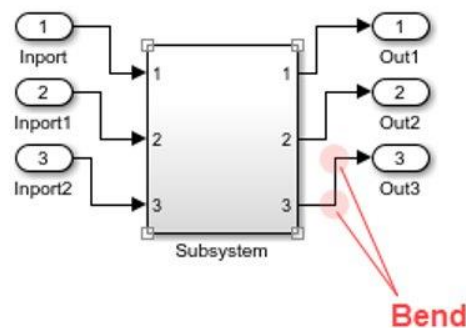
Ľahšia tvorba vlastných Live Editor Tasks

Konverzia výberu *(kódu a interaktívneho ovládania)* do vlastného Live Editor tasku



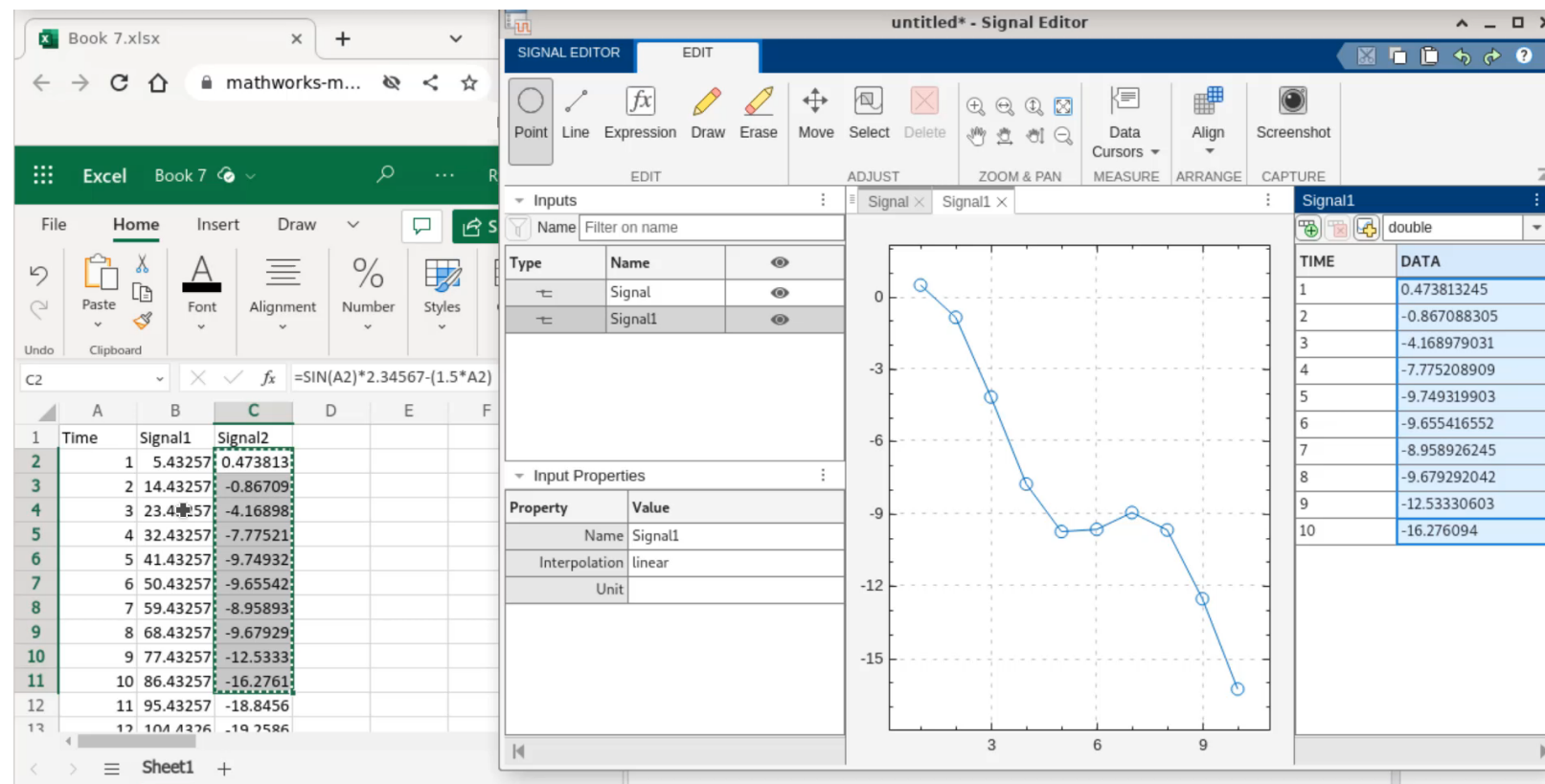
Rýchla editácia modelov

Automatické zachovanie tvaru čiar signálu pri presúvaní a zmene veľkostí blokov



Interaktívna tvorba, úprava a vizualizácia signálov

Copy, cut, paste **dát**
Excelu do alebo z
 Signal Editora



The screenshot illustrates the integration between Microsoft Excel and the Signal Editor software. On the left, an Excel spreadsheet (Book 7.xlsx) is open, showing a table with columns 'Time', 'Signal1', and 'Signal2'. The formula bar shows the expression $=\text{SIN}(A2)*2.34567-(1.5*A2)$. On the right, the Signal Editor window (untitled* - Signal Editor) is shown. It features a toolbar with various editing tools, a plot area displaying a blue line graph of 'Signal1' over time, and a data table on the right side. The data table in Signal Editor has columns 'TIME' and 'DATA', with values corresponding to the 'Signal1' column in the Excel spreadsheet.

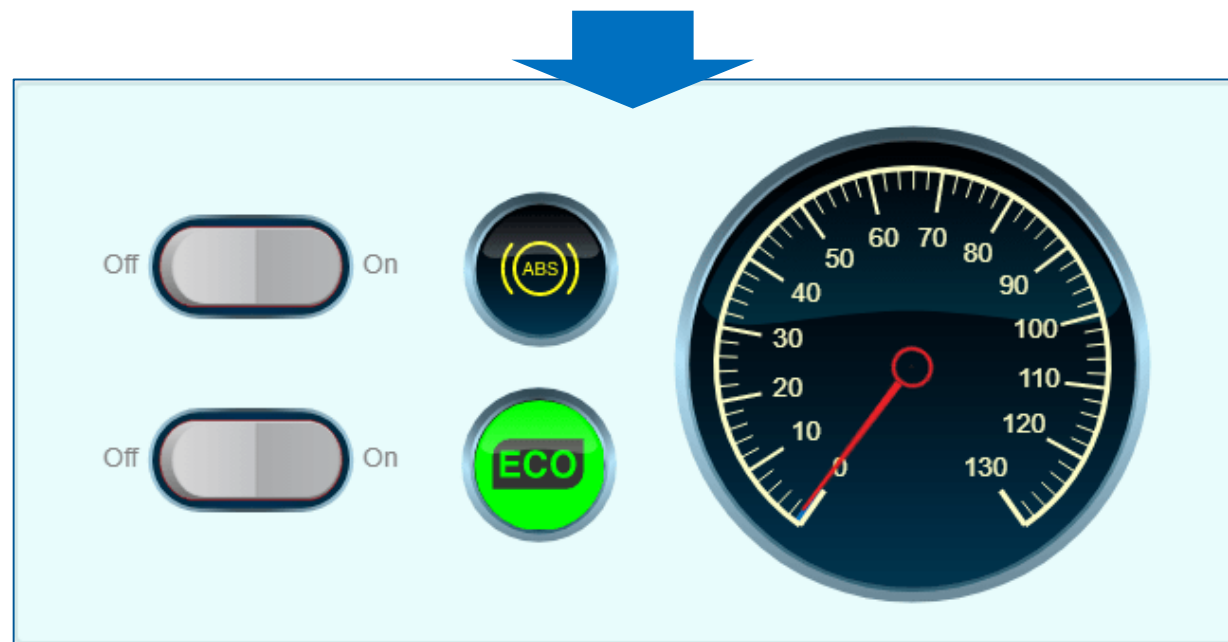
TIME	DATA
1	0.473813245
2	-0.867088305
3	-4.168979031
4	-7.775208909
5	-9.749319903
6	-9.655416552
7	-8.958926245
8	-9.679292042
9	-12.53330603
10	-16.276094

Tvorba ovládania simulácií pomocou Dashboard

Dashboard bloky



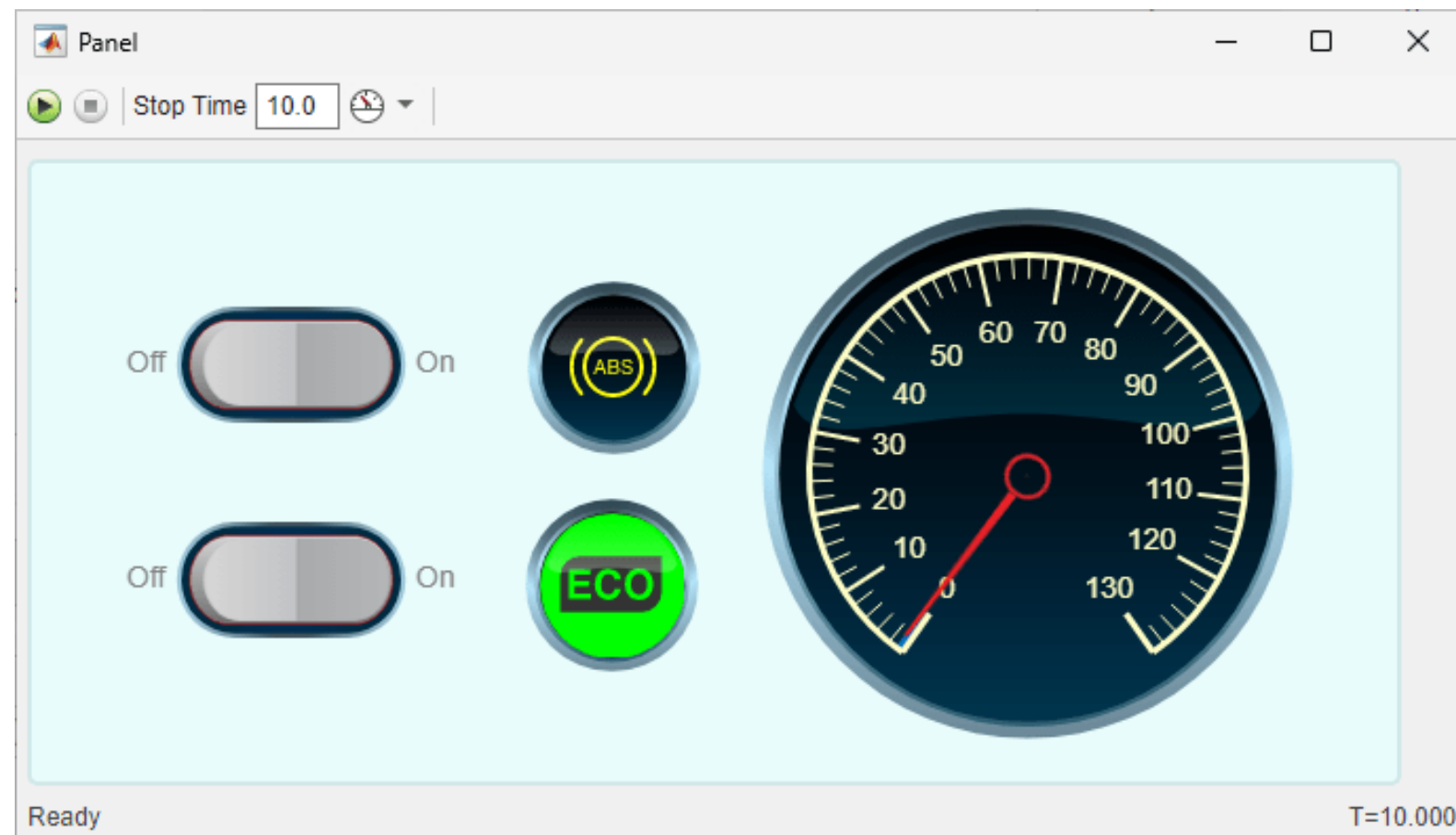
Zoskupenie a presunutie dashboard blokov do panelu



Dashboard panel

Nasadenie ovládania simulácií ako aplikácie

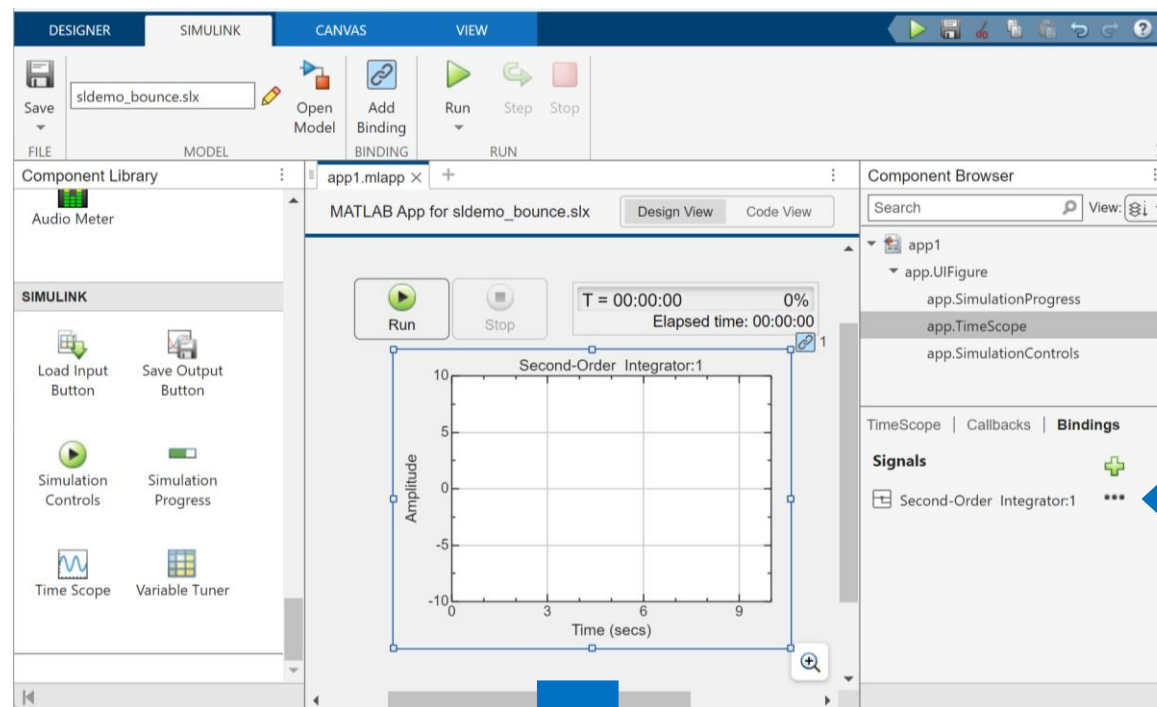
Transformácia
 dashboard panelov
 na aplikácie



Tvorba simulačných aplikácií bez písania kódu

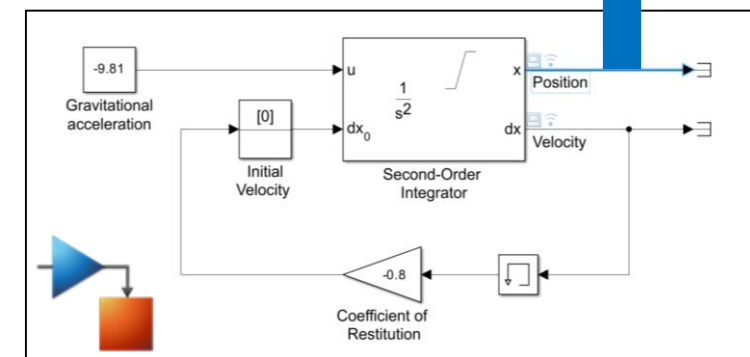
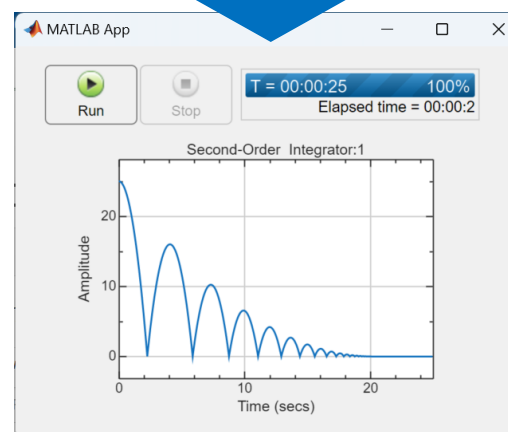
Integrácia aplikácií s modelmi Simulinku, vrátane väzby signálu

Pripravené špecifické grafické komponenty pre Simulink



Väzba signálu s modelom

Aplikácia





MATLAB®
& SIMULINK®



Integrácia



AI



Pohodnosť



Výkon



Verifikácia

MATLAB na počítačoch Mac

MATLAB a Simulink
bežia natívne na Apple silicon

Výšší výkon a
zlepšenie životnosti batérie na
počítačoch MacBook



Vylepšenia výkonu pri každom vydaní

Performance

- Language and Programming: Improved performance for reading and writing reading and writing class property values

```
classdef StorageClass
    properties
        data
        average
    end
end
```

This code is about 17.5x faster than in the previous release.

```
s = StorageClass(1:1e6);
timeit(@()s.movingAverage)
```

The approximate execution times are:

R2023a: 0.497 s

R2023b: 0.0284 s

The code was timed on a Windows[®] 10, Intel[®] Xeon[®] CPU E5-1650 v3 @ 3.50 GHz test system.

Najlepšie príklady dokumentu

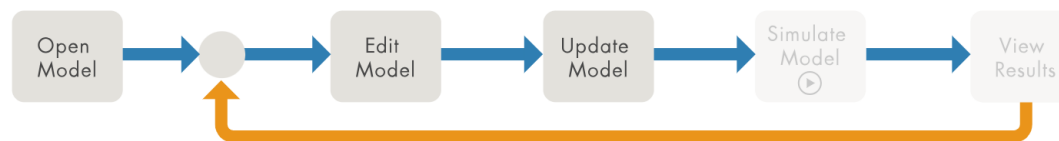
Release notes

Každá poznámka vydania zahŕňa:

- Kód príkladu
- Merané zlepšenie výkonu
- Použitý hardvér

Zlepšenie výkonu simulácie v závislosti na postupe

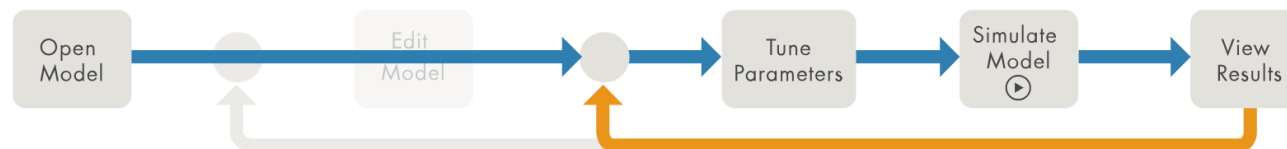
Edit-Update-Repeat



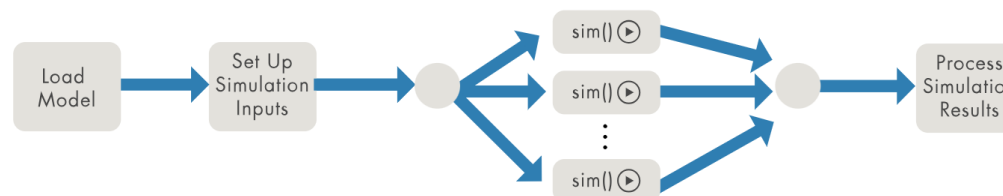
Edit-Sim-Repeat



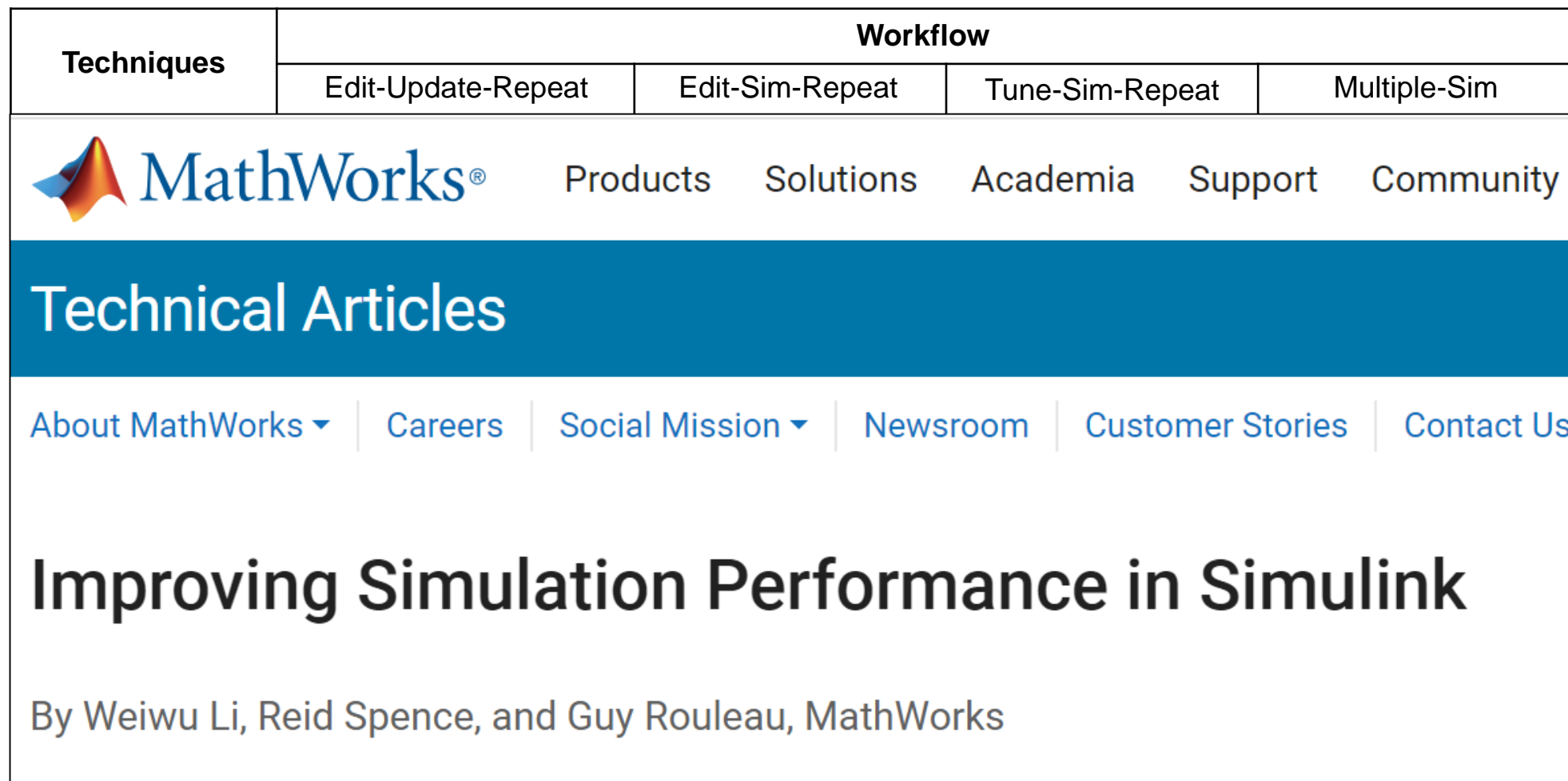
Tune-Sim-Repeat



Multiple-Sim



Využitie špecifických techník na zlepšenie výkonu

Techniques	Workflow			
	Edit-Update-Repeat	Edit-Sim-Repeat	Tune-Sim-Repeat	Multiple-Sim
				

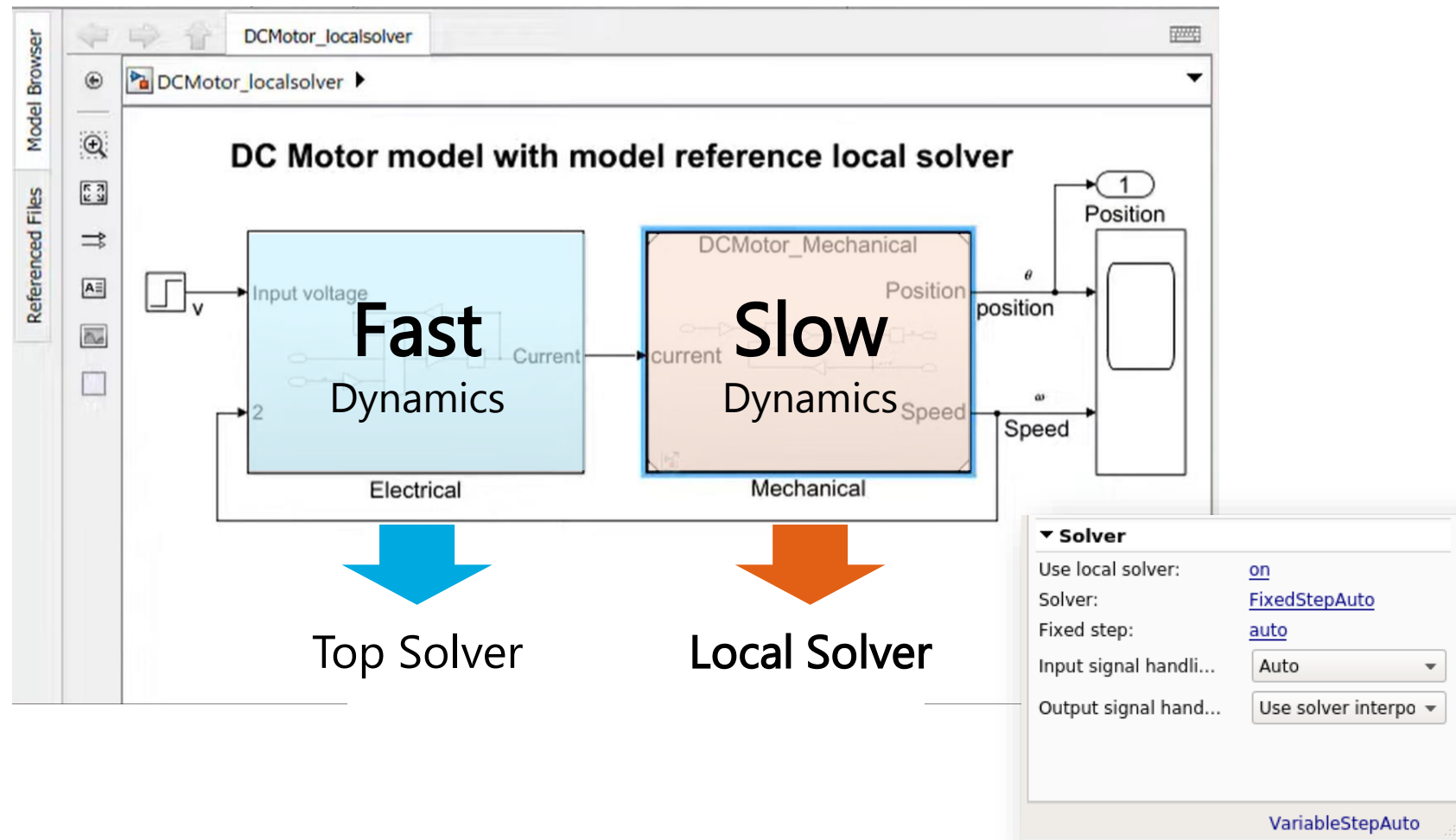
Využitie špecifických techník na zlepšenie výkonu



Techniques	Workflow			
	Edit-Update-Repeat	Edit-Sim-Repeat	Tune-Sim-Repeat	Multiple-Sim
Simulation Mode		X	X	X
Fast Restart			X	X
Simulation Cache	X	X	X	X
Model Reference - Parallel Build	X	X		
Model Reference - Incremental Loading & Rebuilding	X	X		
Simulink Profiler	X	X	X	X
Solver Profiler		X	X	X
Modify Your Models		X	X	X
Parallel Simulation				X

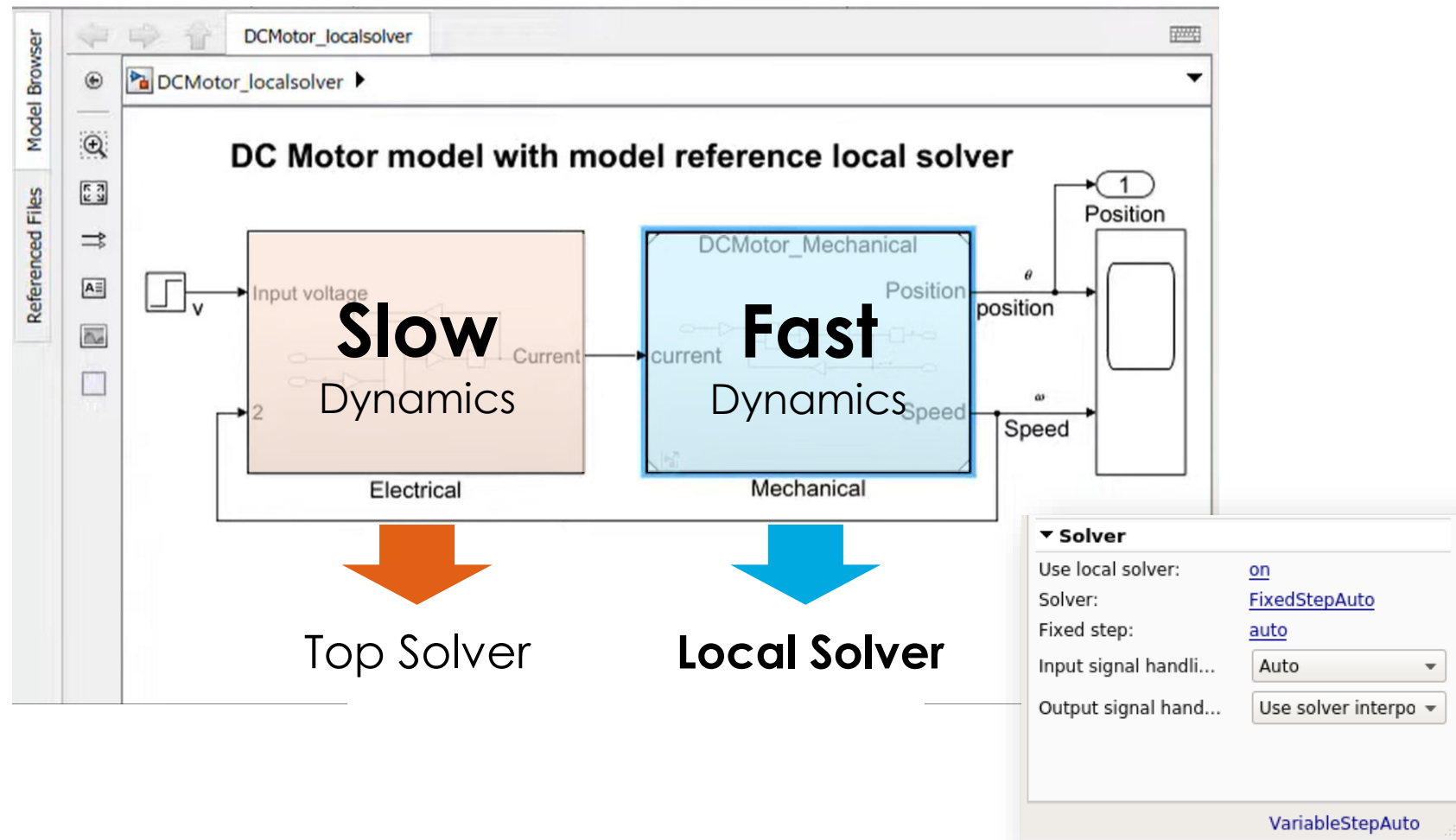
Zrýchlenie simulácií využitím lokálneho solvera

Rozdelenie systémov rôznej dynamiky s lokálnym solverom



Zrýchlenie simulácií využitím lokálneho solvera

Podporuje rýchlejšiu lokálnu dynamiku





MATLAB®
& SIMULINK®



Integrácia



AI



Pohodnosť



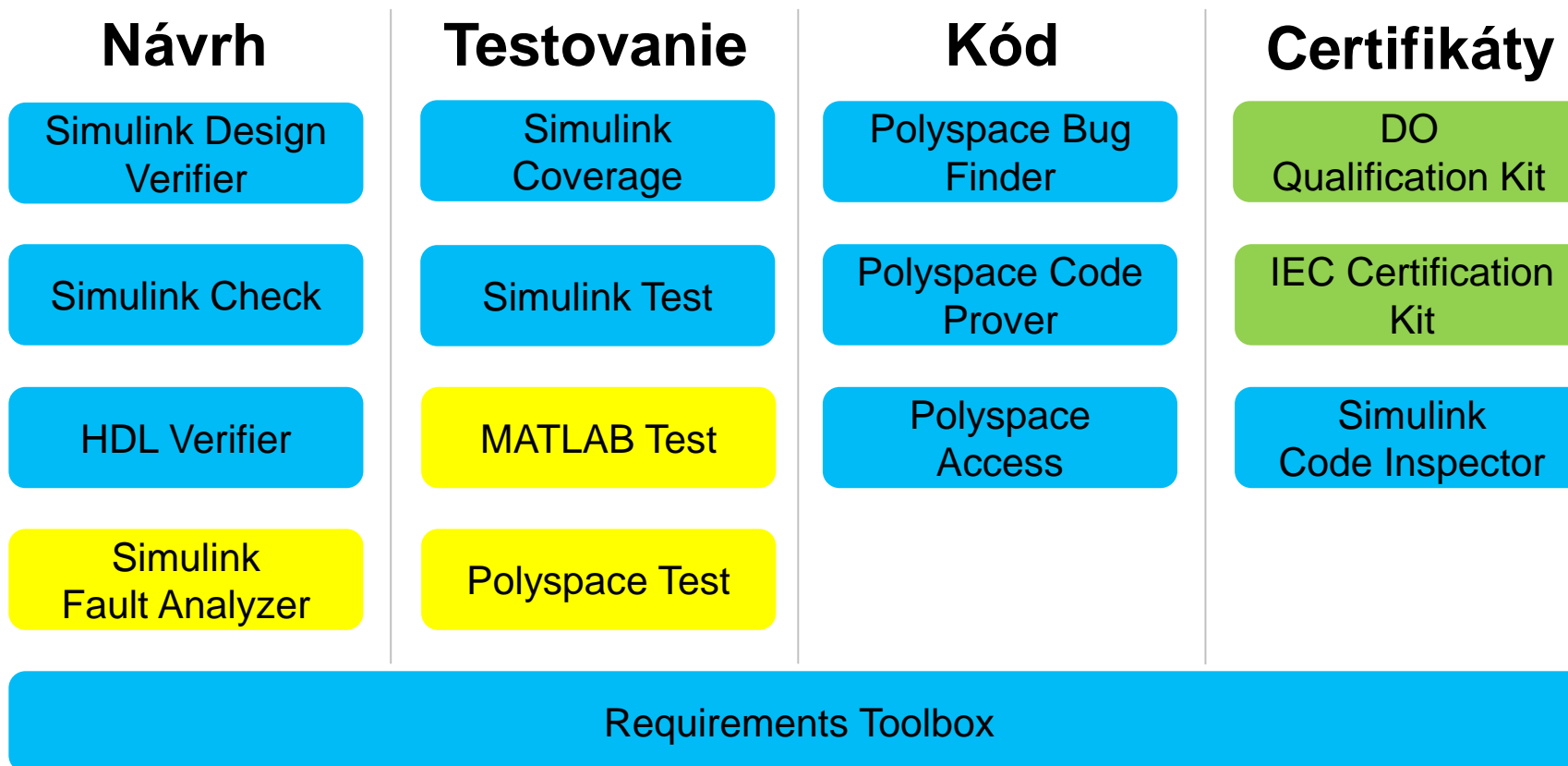
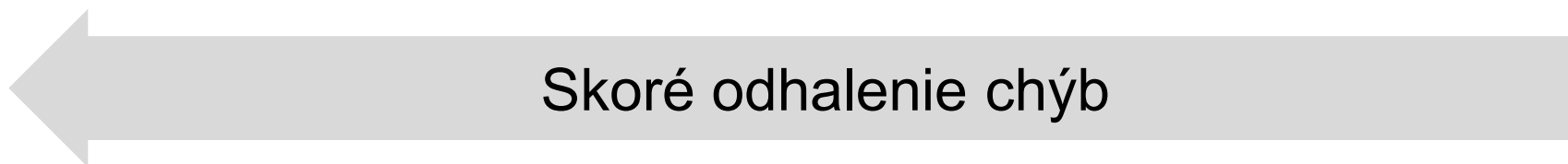
Výkon



Verifikácia

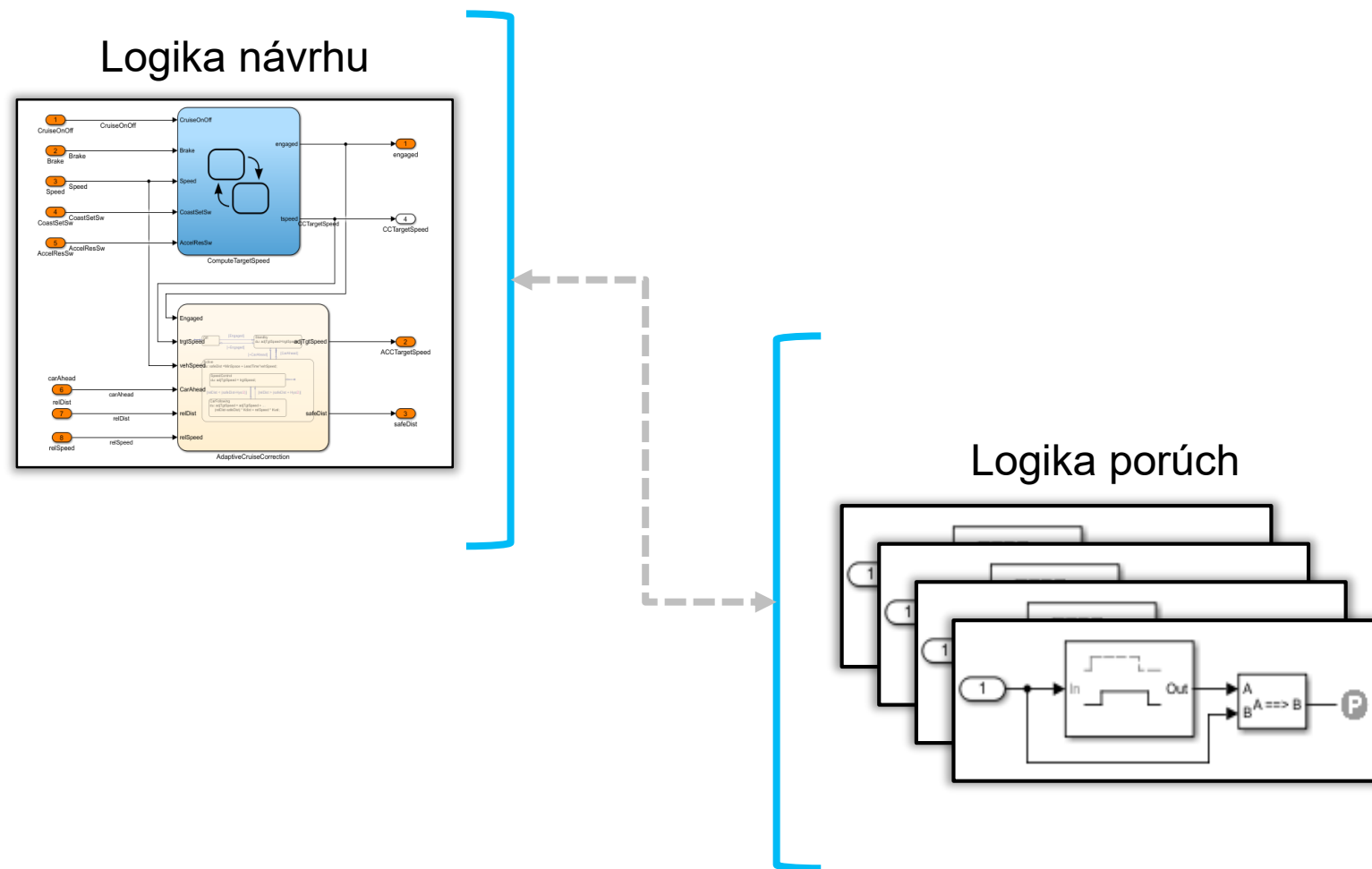
Nové produkty

Výrazné aktualizácie



Poruchy a analýza ich vplyvu – Simulink Fault Analyzer

Modelovanie porúch
bez zmien v návrhu



Poruchy a analýza ich vplyvu – Simulink Fault Analyzer

Modelovanie porúch
bez zmien v návrhu

Správa porúch cez
viaceré domény

Fault Table			
Fault		Conditional	
Enable	Model Element/Fault Name	Active Fault	Trigger
<input checked="" type="checkbox"/>	Environment/Constant6/Outport/1		
	⚡ HighTemperatureFault	<input type="checkbox"/>	Conditional: highSpeedCondition
	⚡ LowTemperaturFault	<input checked="" type="checkbox"/>	Conditional: SampleConditional
<input checked="" type="checkbox"/>	Environment/Constant7/Outport/1		
	⚡ HighPressureFault	<input checked="" type="checkbox"/>	Timed: 20
	⚡ LowPressureFault	<input type="checkbox"/>	Always On
<input checked="" type="checkbox"/>	Environment/Constant2/Outport/1		
	⚡ Grade_fault	<input checked="" type="checkbox"/>	Always On
	⚡ Grade_fault_1	<input type="checkbox"/>	Always On
<input checked="" type="checkbox"/>	Environment/Constant3/Outport/1		
	⚡ wind_x_fault	<input checked="" type="checkbox"/>	Always On
<input checked="" type="checkbox"/>	Passenger Car/Electric Plant/Simscape/Inductor1/Inductor		
	⚡ Inductor1_fault	<input checked="" type="checkbox"/>	Behavioral

Poruchy a analýza ich vplyvu – Simulink Fault Analyzer

Modelovanie porúch bez zmien v návrhu

Správa porúch cez viaceré domény

Simulácia, skúmanie a analýza efektov porúch

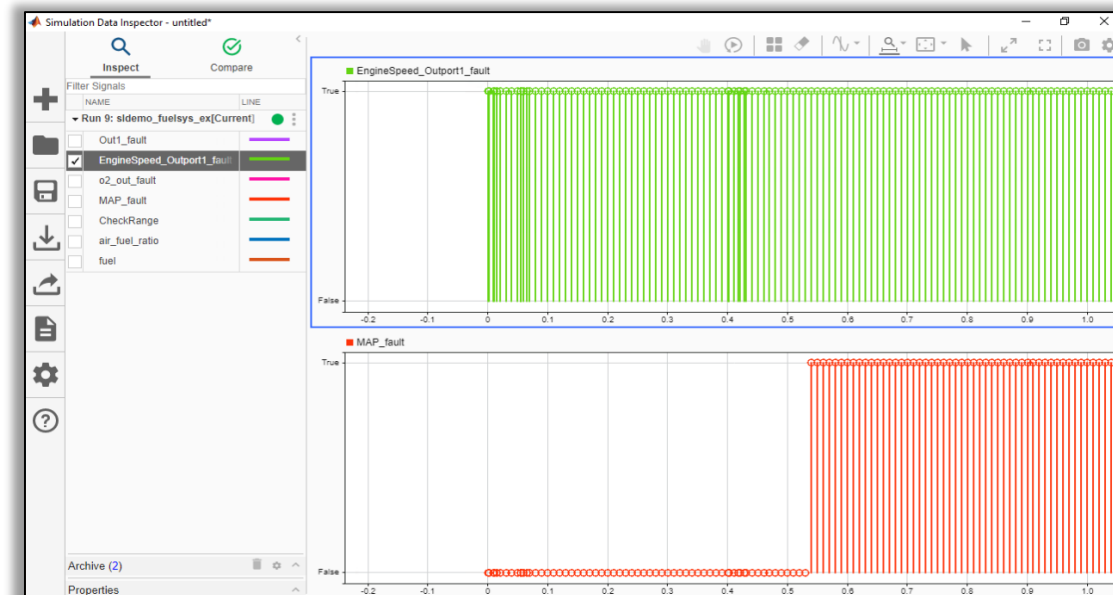
Details: Design Study

Specification Run Options

Root Parameter Set

Fault Set_1

<input checked="" type="checkbox"/>	Fault	Component
<input checked="" type="checkbox"/>	HighTemperatureFault	EvReferenceApplic...
<input checked="" type="checkbox"/>	HighPressureFault	EvReferenceApplic...
<input checked="" type="checkbox"/>	LowTemperaturFault	EvReferenceApplic...
<input checked="" type="checkbox"/>	LowPressureFault	EvReferenceApplic...
<input checked="" type="checkbox"/>	Grade_fault	EvReferenceApplic...
<input checked="" type="checkbox"/>	Grade_fault_1	EvReferenceApplic...
<input checked="" type="checkbox"/>	wind_x_fault	EvReferenceApplic...



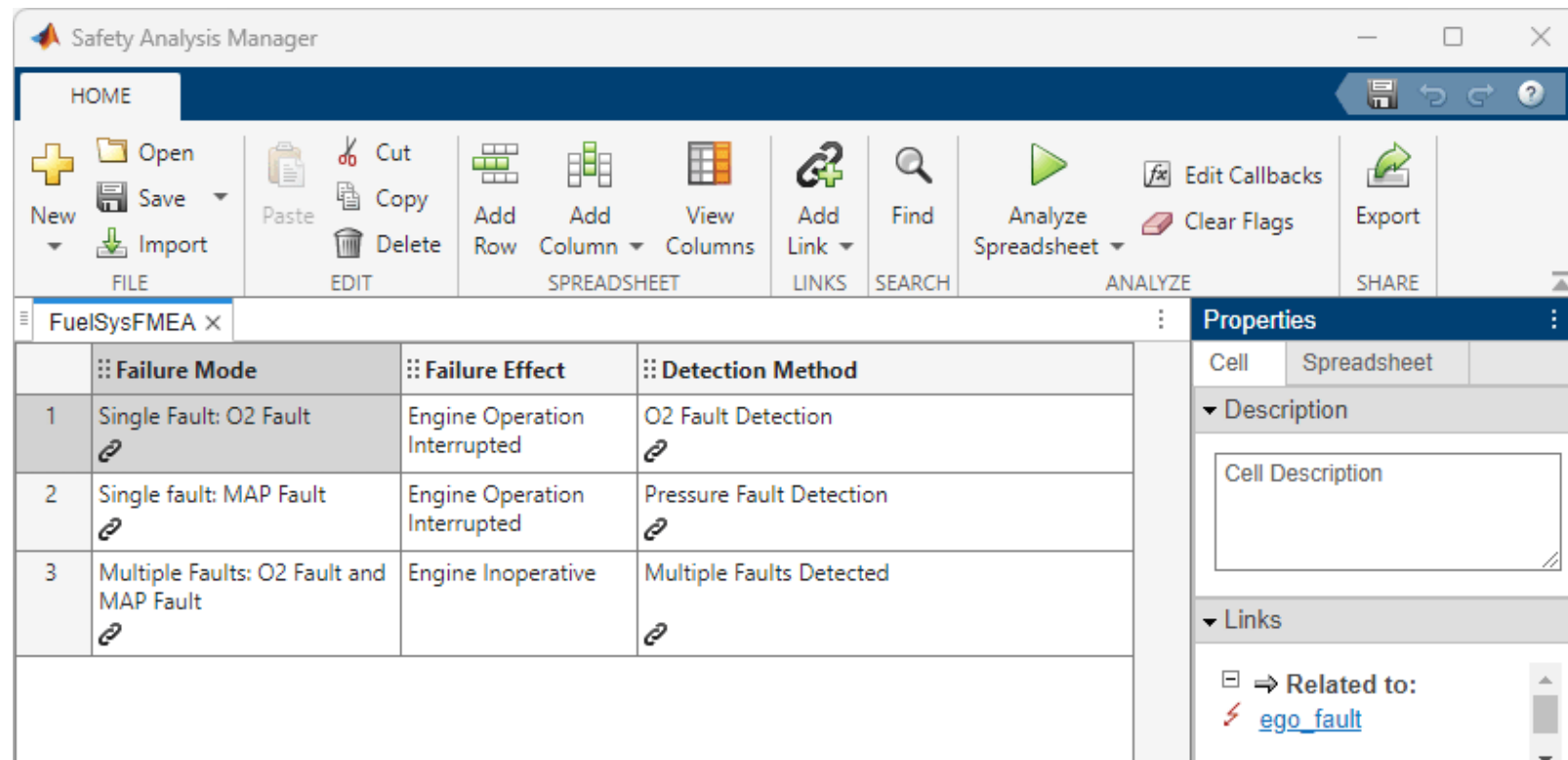
Poruchy a analýza ich vplyvu – Simulink Fault Analyzer

Modelovanie porúch bez zmien v návrhu

Správa porúch cez viaceré domény

Simulácia, skúmanie a analýza efektov porúch

Systematická analýza bezpečnosti



The screenshot shows the Safety Analysis Manager application window. The main area displays a table with three columns: Failure Mode, Failure Effect, and Detection Method. The table contains three rows of data. To the right of the table is a Properties panel with tabs for Cell and Spreadsheet, and sections for Description and Links.

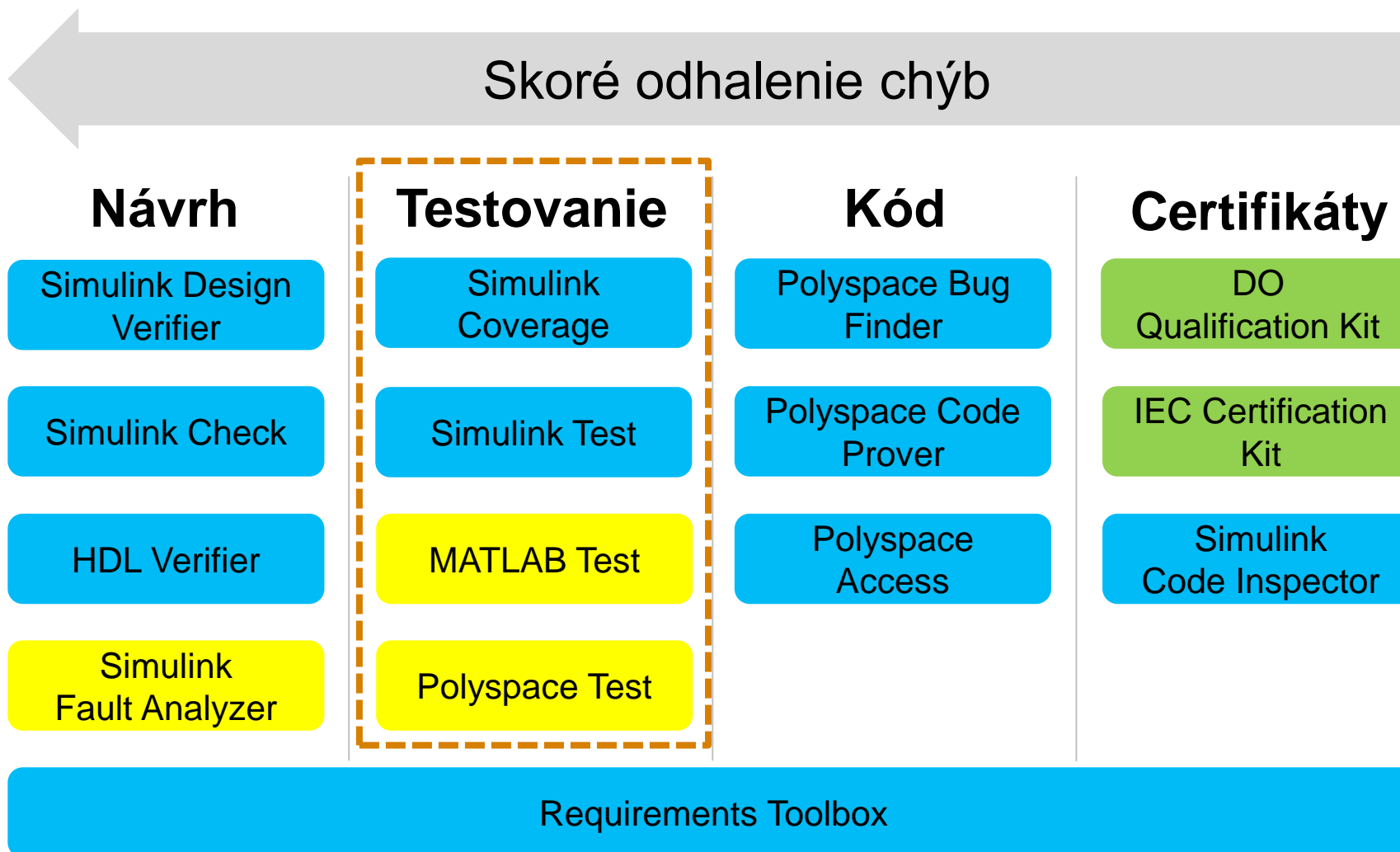
	Failure Mode	Failure Effect	Detection Method
1	Single Fault: O2 Fault	Engine Operation Interrupted	O2 Fault Detection
2	Single fault: MAP Fault	Engine Operation Interrupted	Pressure Fault Detection
3	Multiple Faults: O2 Fault and MAP Fault	Engine Inoperative	Multiple Faults Detected

Properties panel details:

- Cell | Spreadsheet
- Description: Cell Description
- Links: Related to: ego_fault

Nové produkty

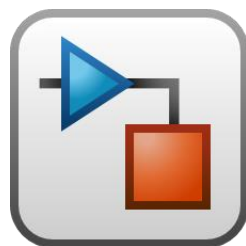
Výrazné aktualizácie



Rodina produktov testovania

R2015a

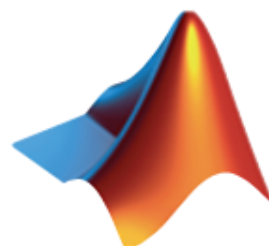
Simulink Test



Model

R2023a

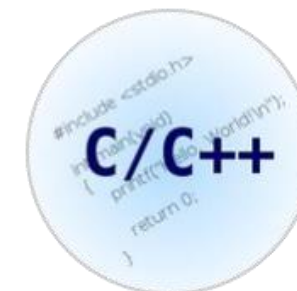
MATLAB Test



MATLAB Kód

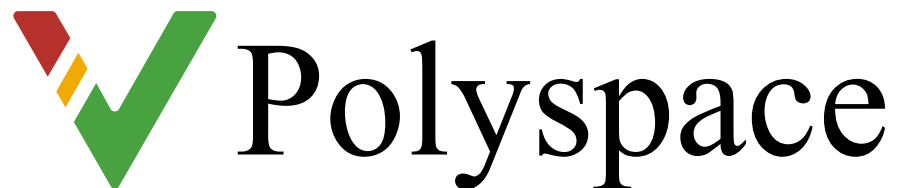
R2023b

Polyspace Test

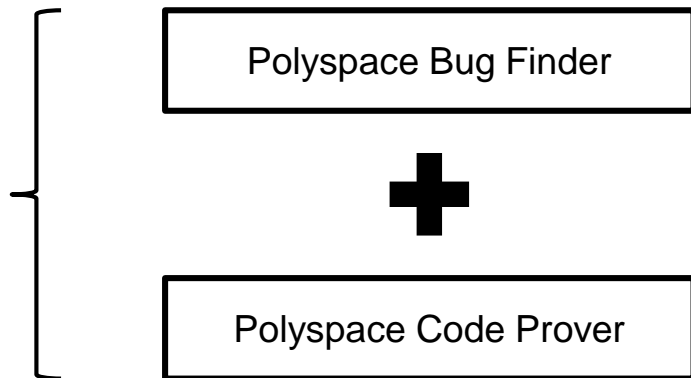


C/C++ kód

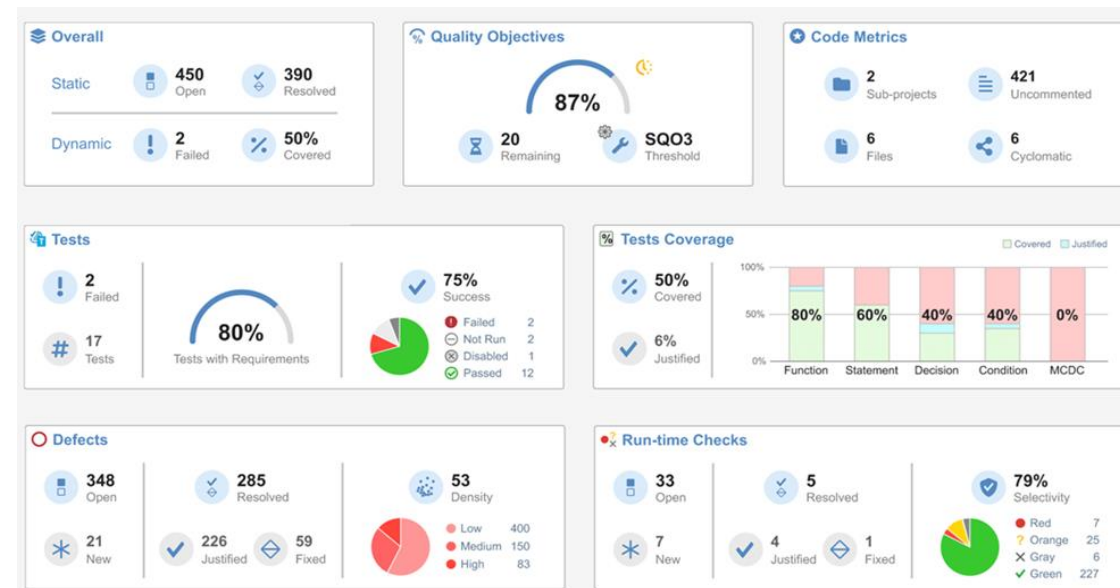
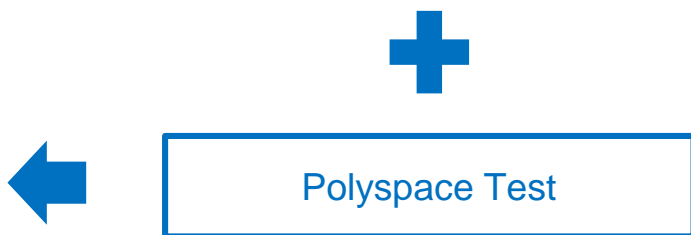
Testy pre C/C++ kód vo vnorených systémoch



Statická Analýza



Dynamické Testovanie



Centrálne správa a kombinácia statickej analýzy s dynamickým testovaním

ISO 26262 DO-178 IEC 61508 IEC 62304



MATLAB®
& SIMULINK®



Integrácia



AI



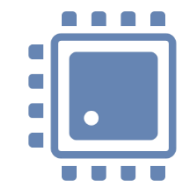
Jazyky



Simulácia



Vizualizácia



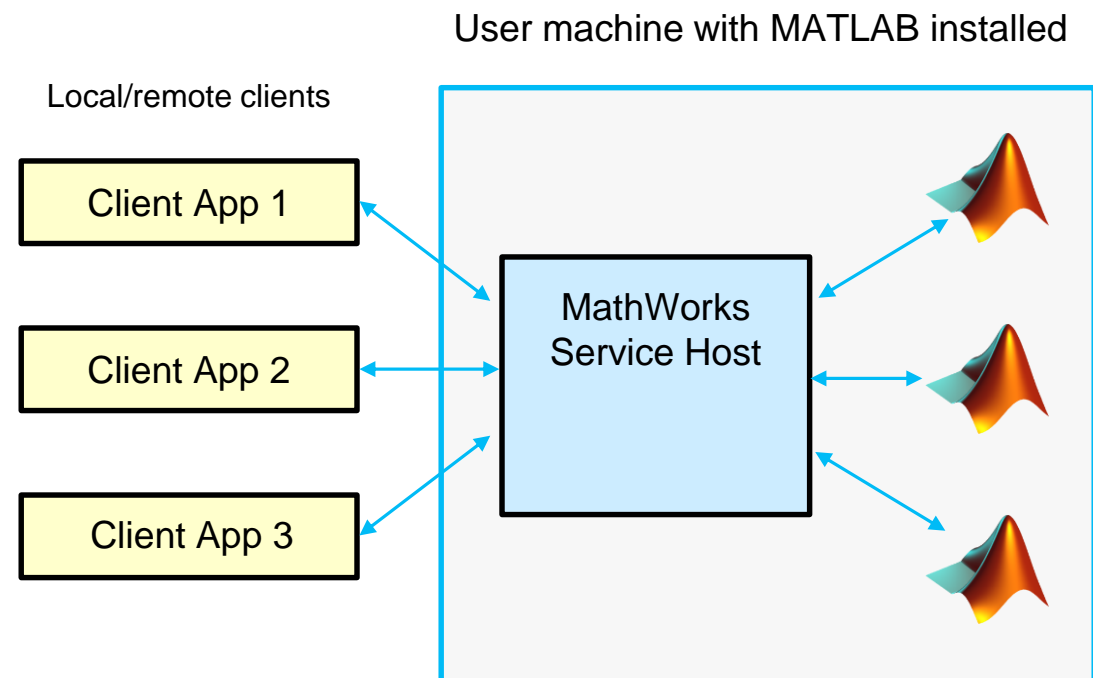
Hardvér

Volanie MATLABu lokálne alebo vzdialenie pomocou REST

Využitie vlastného klienta na volanie
 Prostredia MATLAB s využitím
MATLAB Function Service

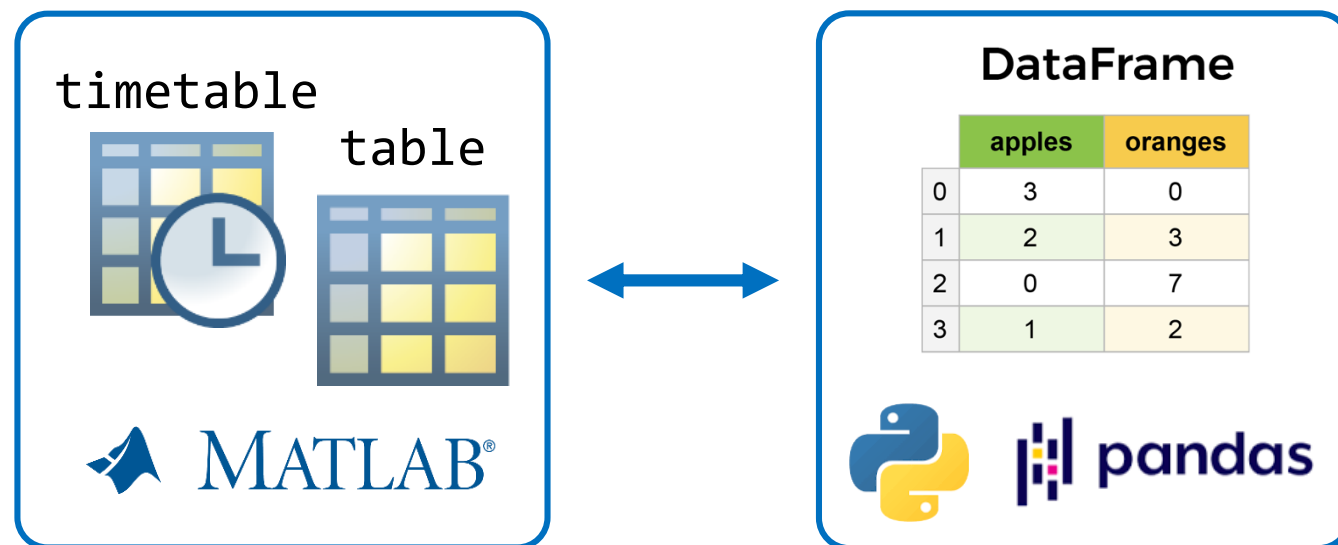
Služba využíva
 HTTPS protokol

REST Function Service



Využitie prostredia MATLAB s Pythonom

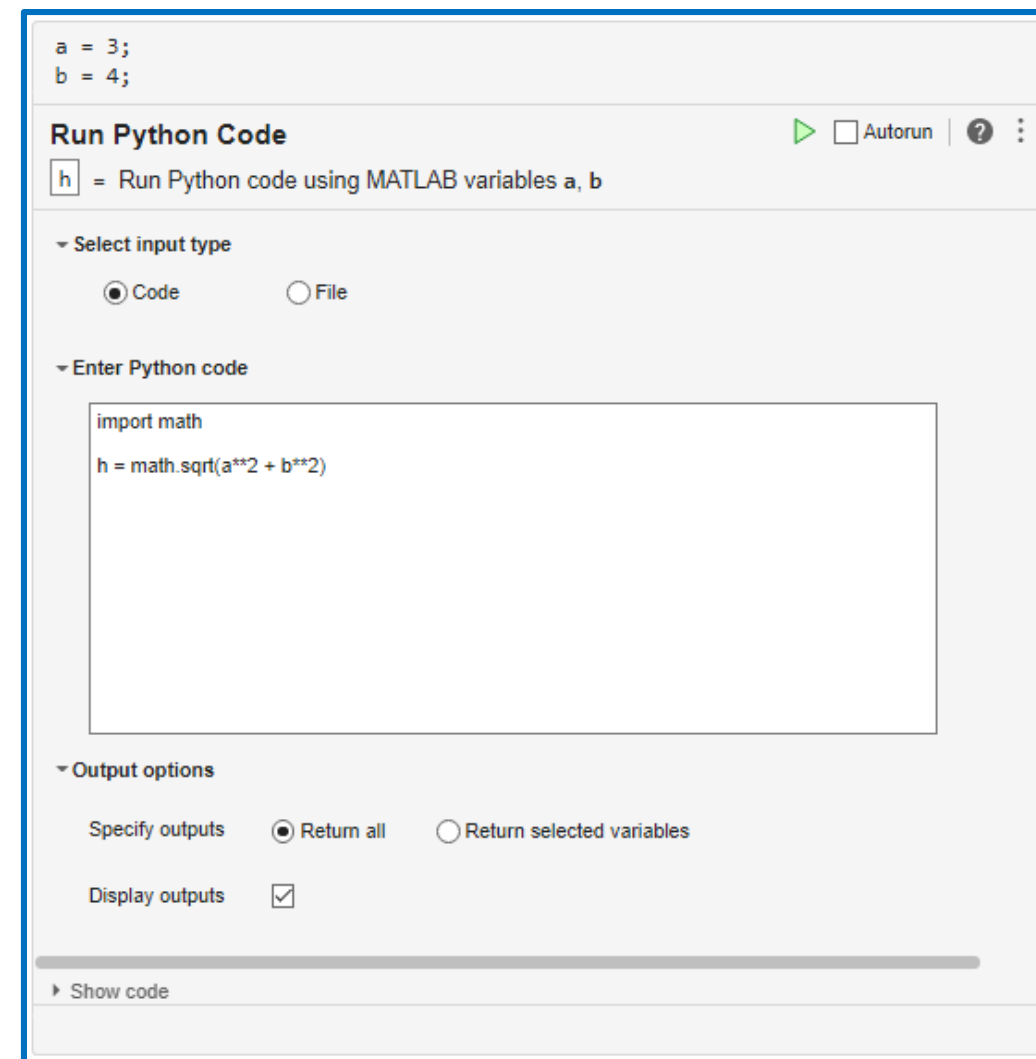
Automatická konverzia medzi
 MATLAB **table** alebo **timetable**
 a Python **Pandas DataFrame**



Využitie prostredia MATLAB s Pythonom

Automatická konverzia medzi
MATLAB `table` alebo `timetable`
a Python Pandas DataFrame

Interaktívne spúšťanie Python kódu s
[Run Python Code Live Editor task](#)



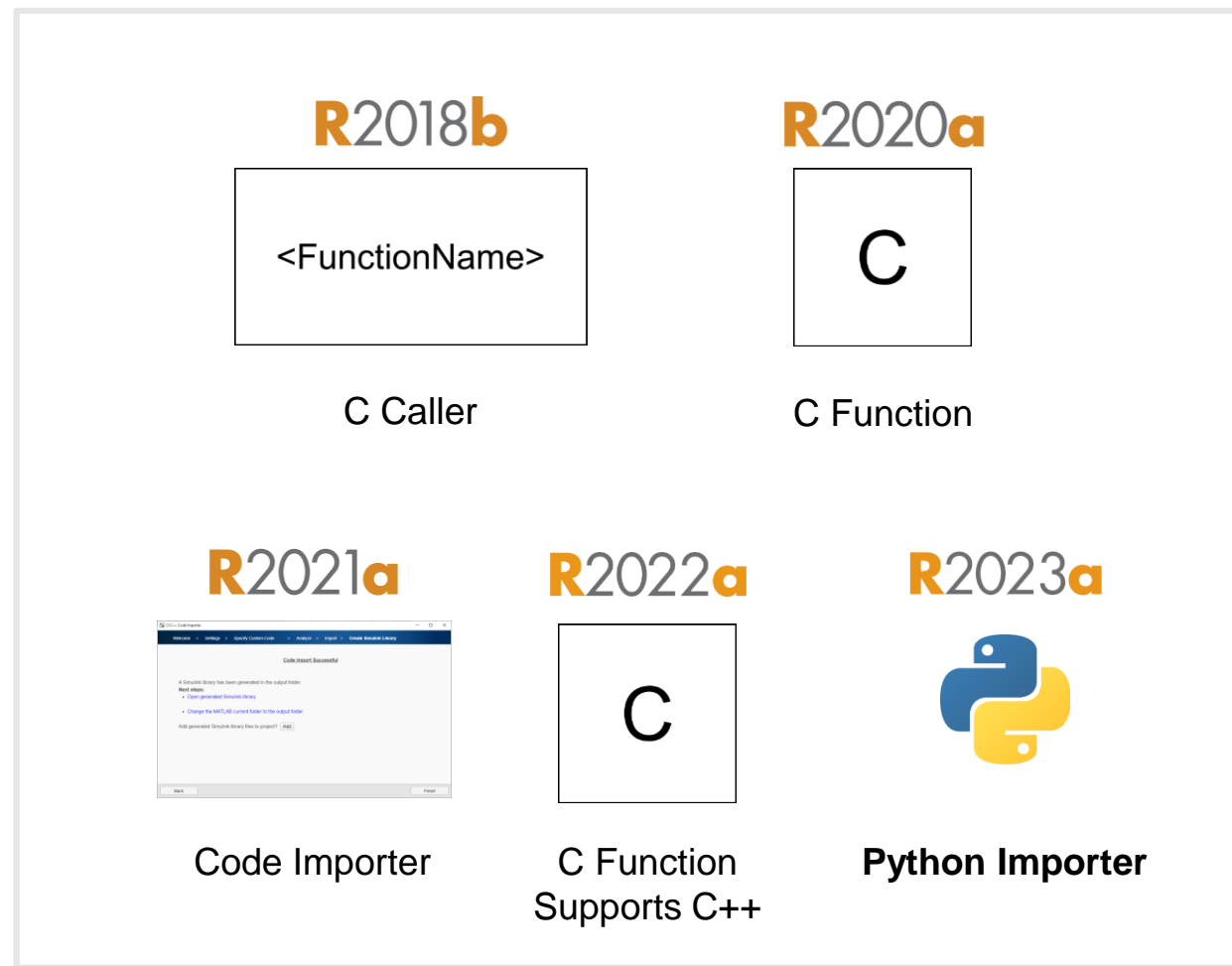
Simulink ako Simulačno Integračná Platforma

Ekosystém a
interoperabilita so
100+ jazykmi a
nástrojmi tretích strán



Import vlastného kódu do prostredia Simulink

Tvorba vlastných komponentov s využitím C, C++ a Python



Import funkcií Pythonu z tried

Python importér podporuje funkcie špecifikované v triedach Pythonu

```
class room:

    def __init__(self, length, breadth, height):
        self.length = length
        self.breadth = breadth
        self.height = height

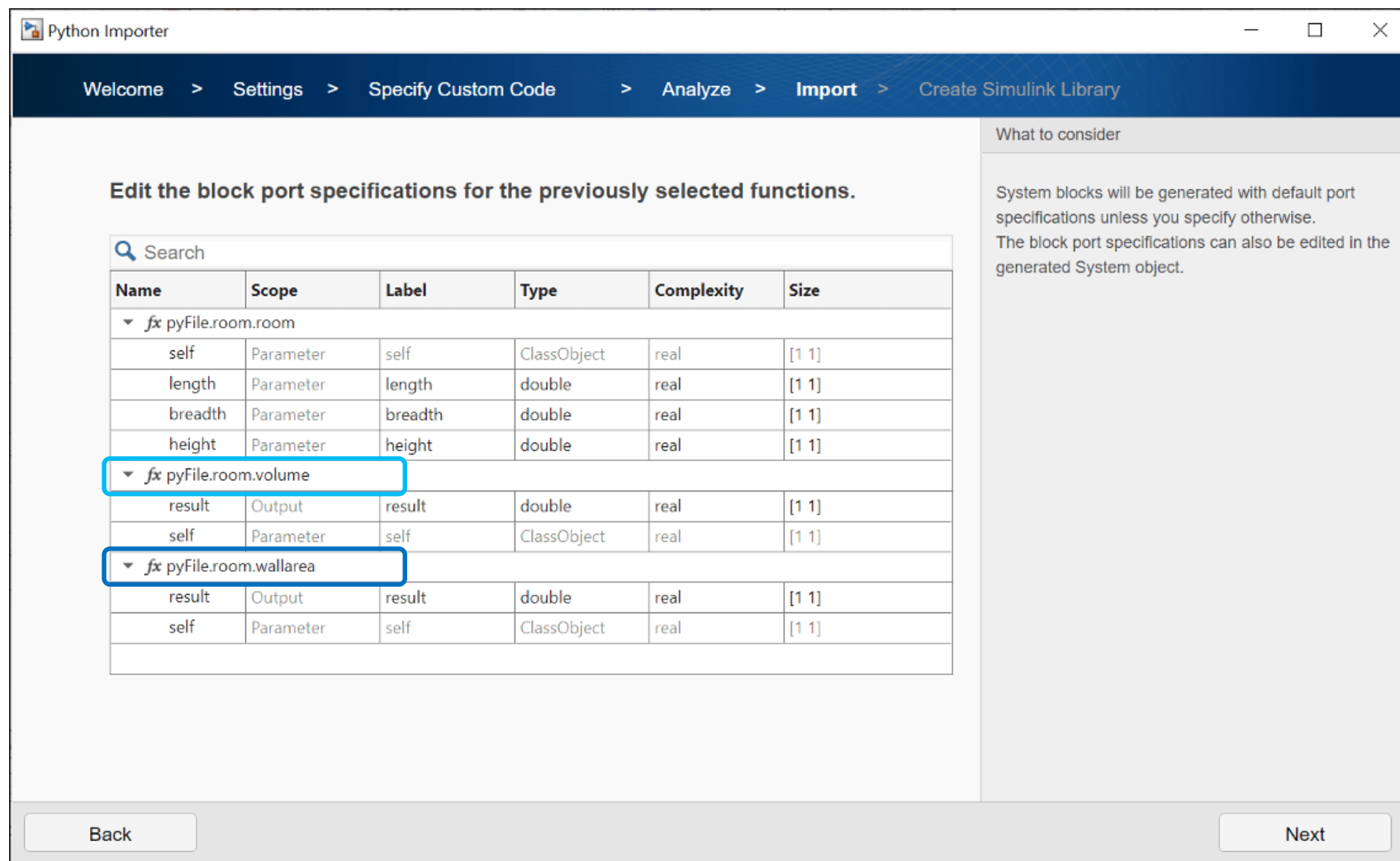
    def volume(self):
        result = self.length * self.breadth * self.height
        return result

    def wallarea(self):
        result = 2 * (self.length * height + self.breadth * height)
        return result
```

Definícia triedy Pythonu

Import funkcí Pythonu z tried

Grafické rozhranie poskytuje návod na import krok za krokom



Python Importer

Welcome > Settings > Specify Custom Code > Analyze > **Import** > Create Simulink Library

Edit the block port specifications for the previously selected functions.

Search

Name	Scope	Label	Type	Complexity	Size
▼ fx pyFile.room.room					
self	Parameter	self	ClassObject	real	[1 1]
length	Parameter	length	double	real	[1 1]
breadth	Parameter	breadth	double	real	[1 1]
height	Parameter	height	double	real	[1 1]
▼ fx pyFile.room.volume					
result	Output	result	double	real	[1 1]
self	Parameter	self	ClassObject	real	[1 1]
▼ fx pyFile.room.wallarea					
result	Output	result	double	real	[1 1]
self	Parameter	self	ClassObject	real	[1 1]

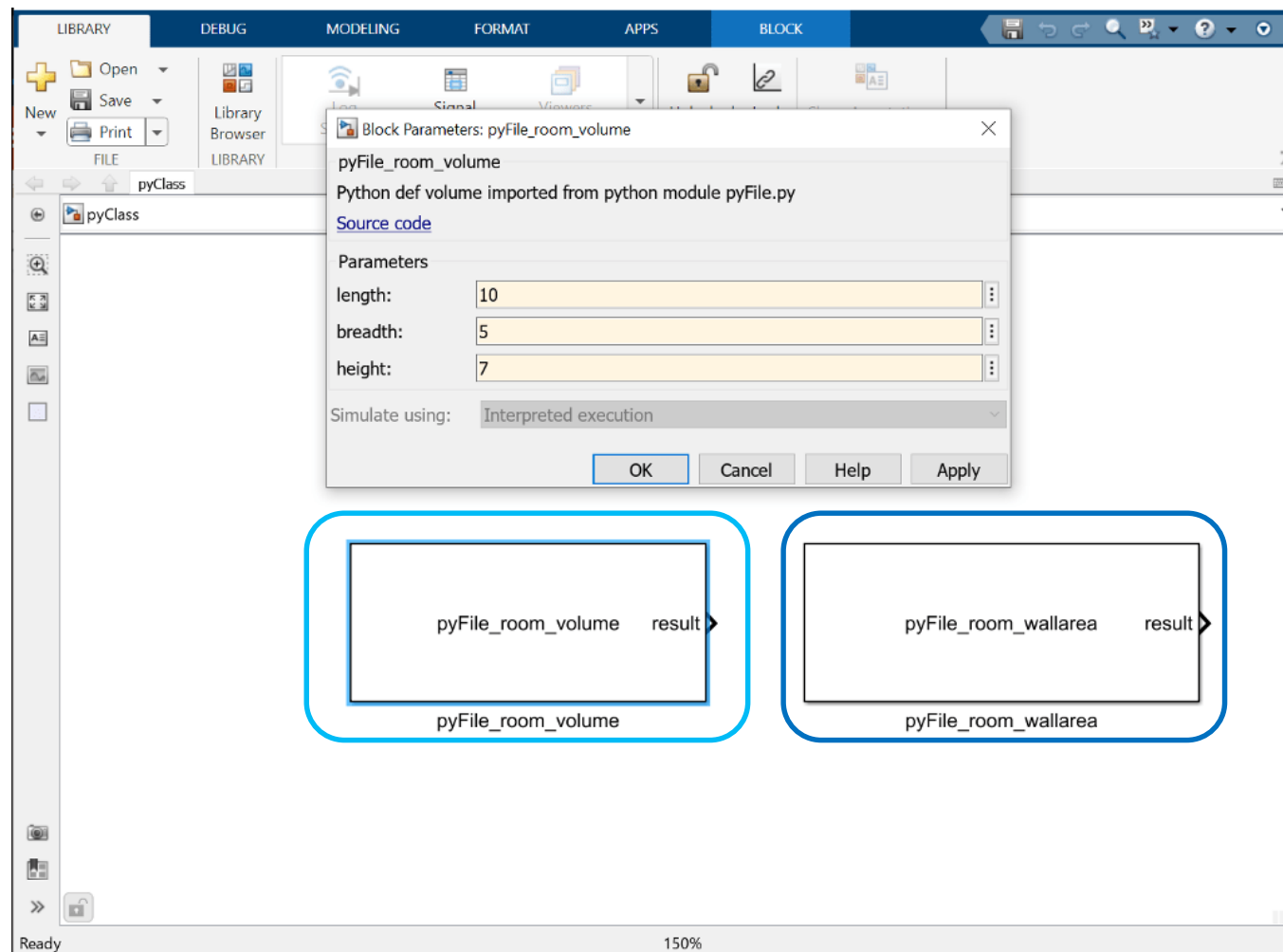
What to consider

System blocks will be generated with default port specifications unless you specify otherwise. The block port specifications can also be edited in the generated System object.

Back Next

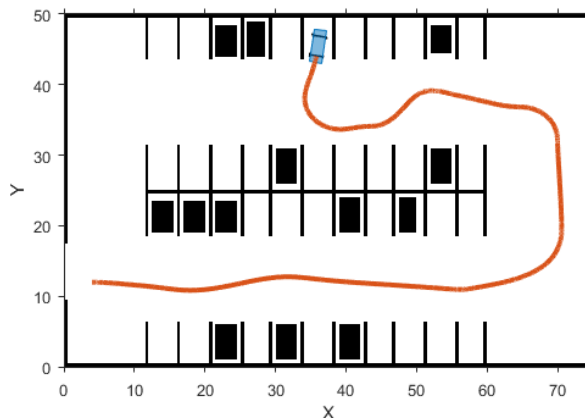
Import funkcií Pythonu z tried

Export ako **vlastný blocksets** pre simuláciu

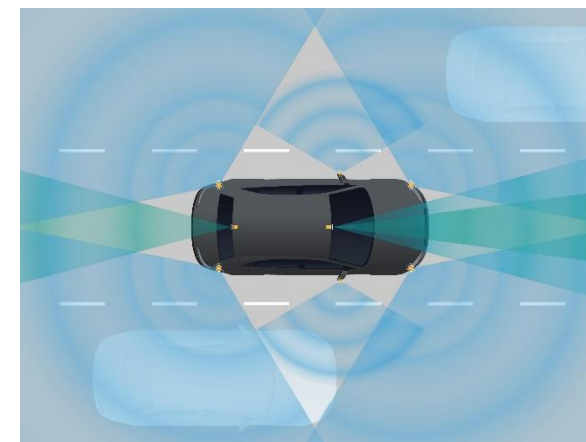


Podpora pre neohraničených signálov s premenlivou dĺžkou

Flexibilita pre modelovanie signálov bez špecifikácie konečnej dĺžky signálu



Autonómny systém parkovania



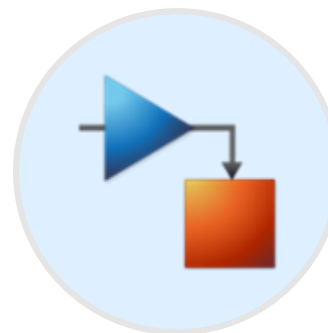
Radarový systém

Rozmer signálov je neznámi pri kompilácií a môže narastať/zmenšovať počas simulácie

Podpora pre neohraničených signálov s premenlivou dĺžkou

Poskytuje **mapovanie** medzi signálmi Simulinku a dynamickými poliami v C++

Ľahká **výmena dát** medzi ďalšími komponentami externého softvéru



Neohraničené signály s premenlivou dĺžkou

Rozmer signálu **Inf**

Pamäť alokovaná **za behu**

Dynamické polia

Dáta s **meniteľnou veľkosťou**

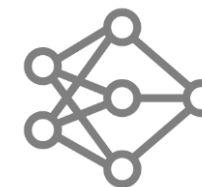
Dynamická pamäť



MATLAB®
& SIMULINK®



Integrácia



AI



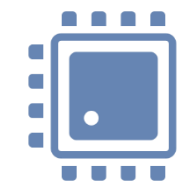
Jazyky



Simulácia



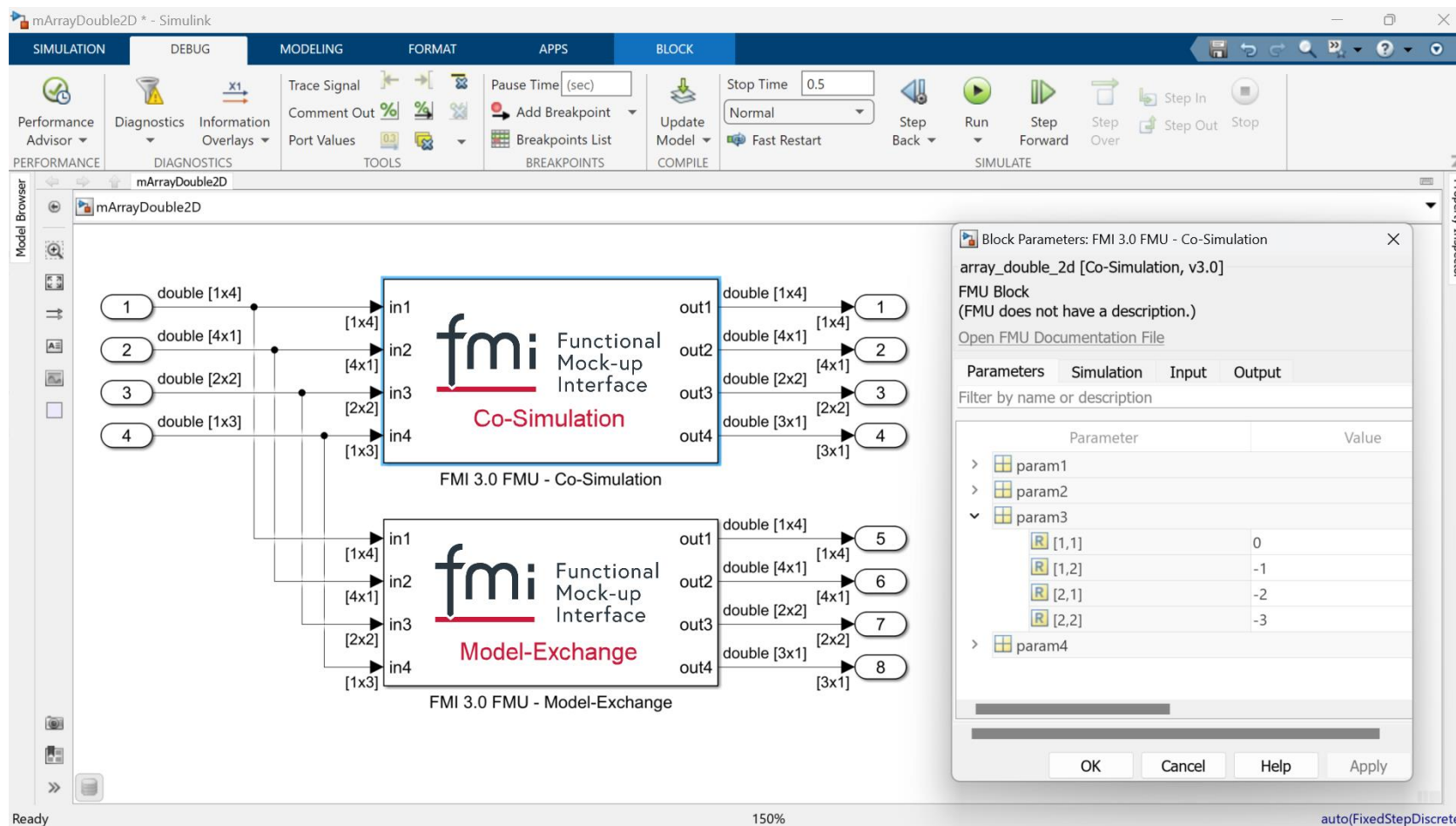
Vizualizácia



Hardvér

Import FMI 3.0 Function Mockup Units (FMUs)

Blok FMU podporuje
FMI verziu 3.0



The screenshot displays the Simulink environment with two FMU blocks. The top block is labeled 'FMI 3.0 FMU - Co-Simulation' and the bottom block is 'FMI 3.0 FMU - Model-Exchange'. Both blocks have four input ports (in1 to in4) and four output ports (out1 to out4). The Co-Simulation block's inputs are connected to signal sources with values 1, 2, 3, and 4. The Model-Exchange block's inputs are also connected to signal sources with values 1, 2, 3, and 4. The outputs of both blocks are connected to signal scopes with values 1 through 8.

The 'Block Parameters: FMI 3.0 FMU - Co-Simulation' dialog is open, showing the following parameters:

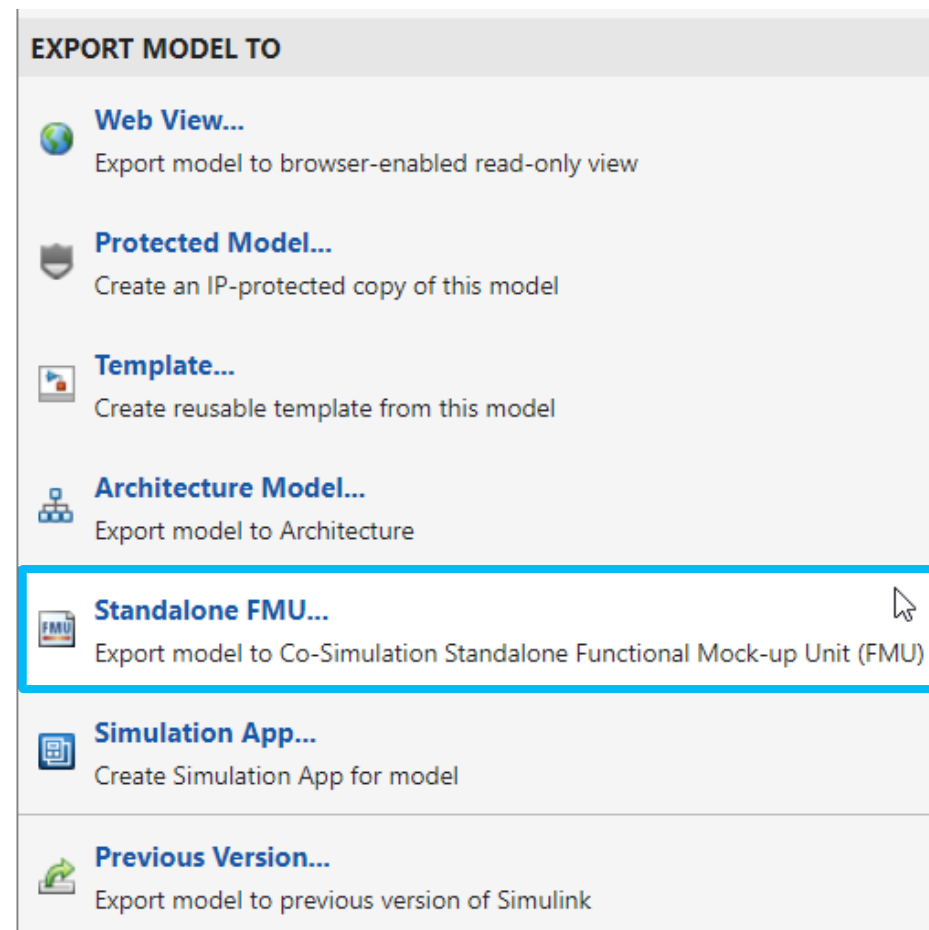
Parameter	Value
param1	
param2	
param3	
R [1,1]	0
R [1,2]	-1
R [2,1]	-2
R [2,2]	-3
param4	

Export simulácií ako FMI 3.0 FMU

Tvorba samostatných FMU z modelov prostredia Simulink alebo zdrojového kódu

Validácia FMI 3.0 FMU

Export FMU na použitie v iných simulačných prostrediami



Ovládanie simulácií použitím Simulation objektu

Konfigurácia, beh a interakcia so simuláciou vrátane krokovania

Prístup k simulačnému stavu a výstupom

Nasaditeľné do ďalších prostredí s compilerom

initialize
start
step
pause
resume
stop
terminate

```

Command Window
>> mdl = 'sldemo_suspn_3dof';
>> open_system(mdl);
>> sm = simulation(mdl);
>> initialize(sm)
>> start(sm);
>> pause(sm);
>> sm.Status

ans =

    "paused"

>> sm.Time

ans =

    16.8811

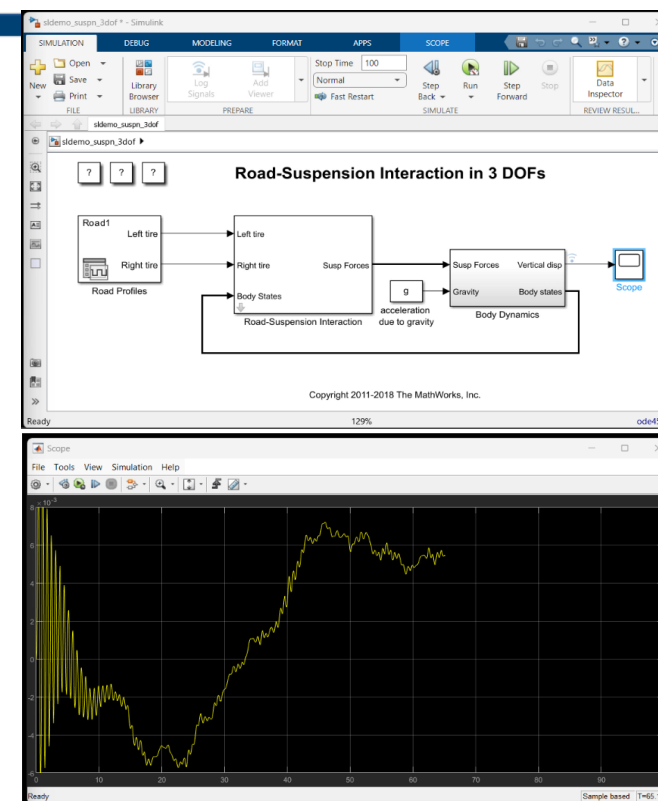
>> step(sm, PauseTime=40);
>> resume(sm)
>> stop(sm);
>> sm.Status

ans =

    "inactive"
    
```

Simulácia skriptom

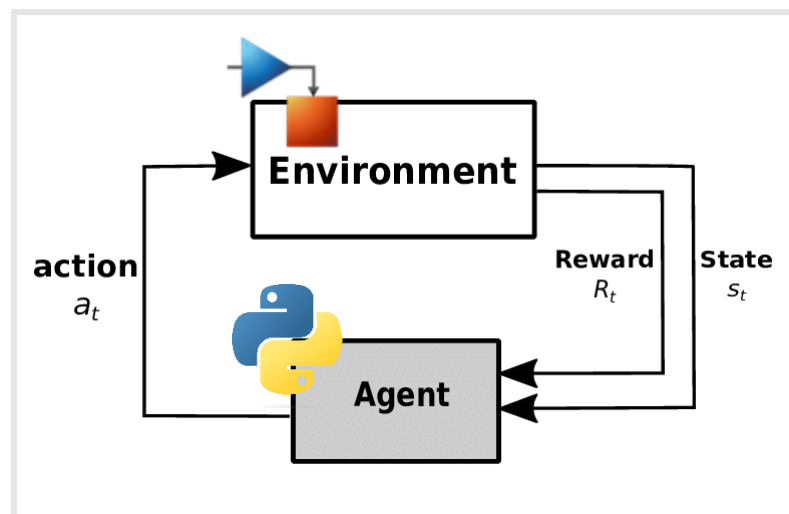
Model prostredia Simulink



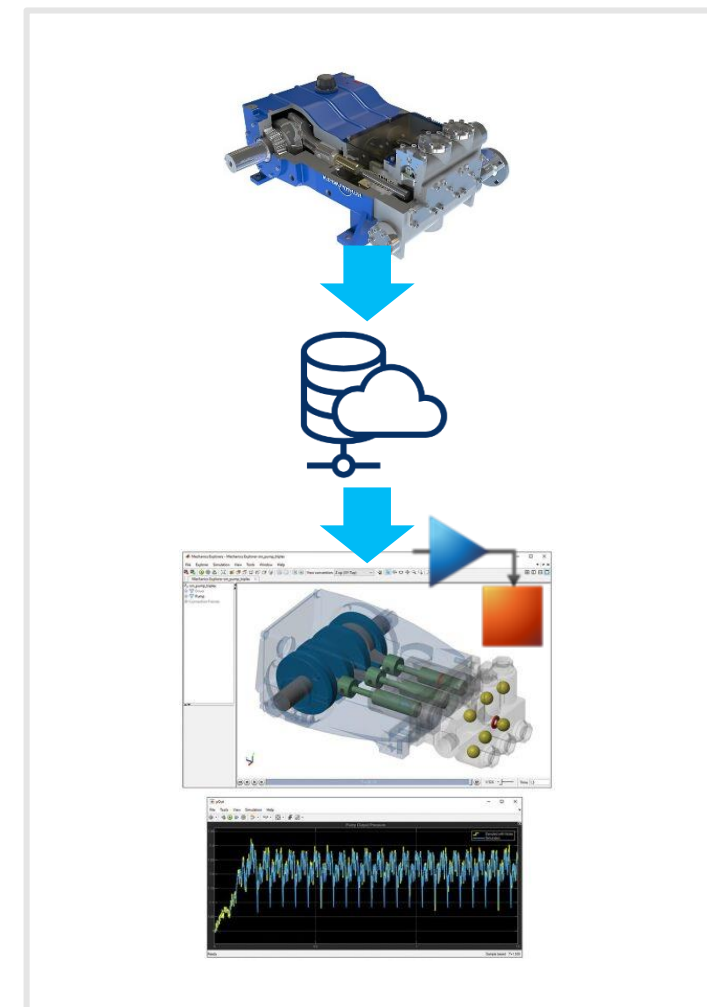
Výstupy simulácie

Ovládanie simulácií použitím Simulation objektu

Umožňuje integráciu simulácie s novými aplikáciami ako sú reinforcement learning a digitálne dvojčatá



Reinforcement Learning



Digitálne dvojčatá



MATLAB®
& SIMULINK®



Integrácia



AI



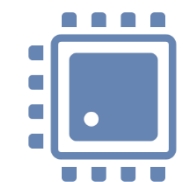
Jazyky



Simulácia



Vizualizácia

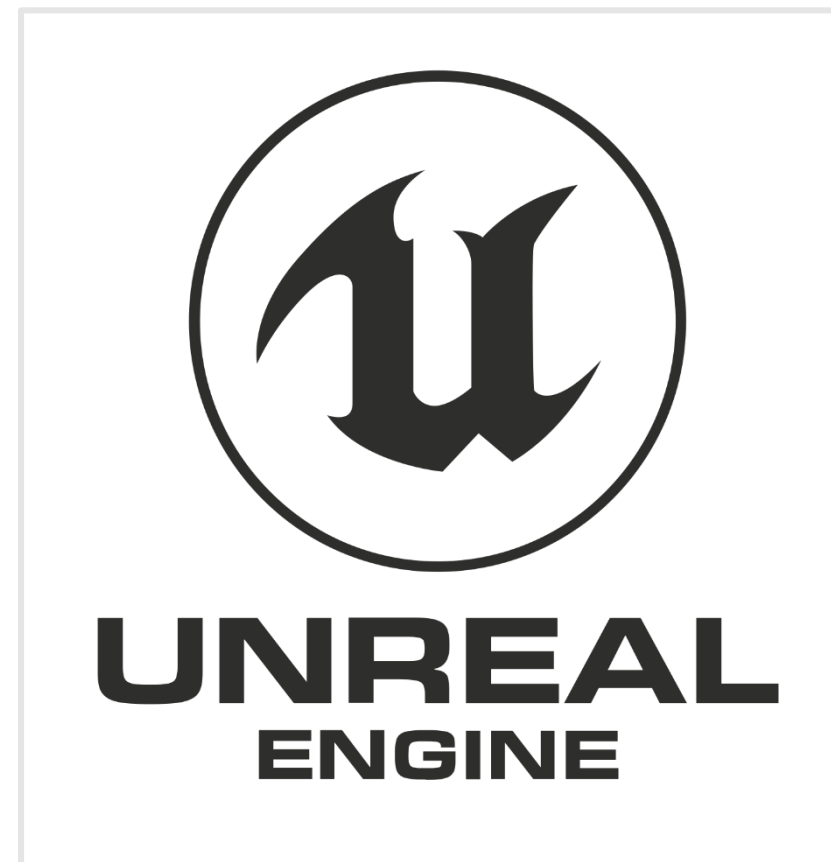


Hardvér

Vizualizácia 3D simulácií

Simulink 3D Animation poskytuje základné 3D prostriedky, platformu a **integráciu s Unreal Enginom** pre vertikálne produkty

3D Vizualizácia



Vizualizácia 3D simulácií

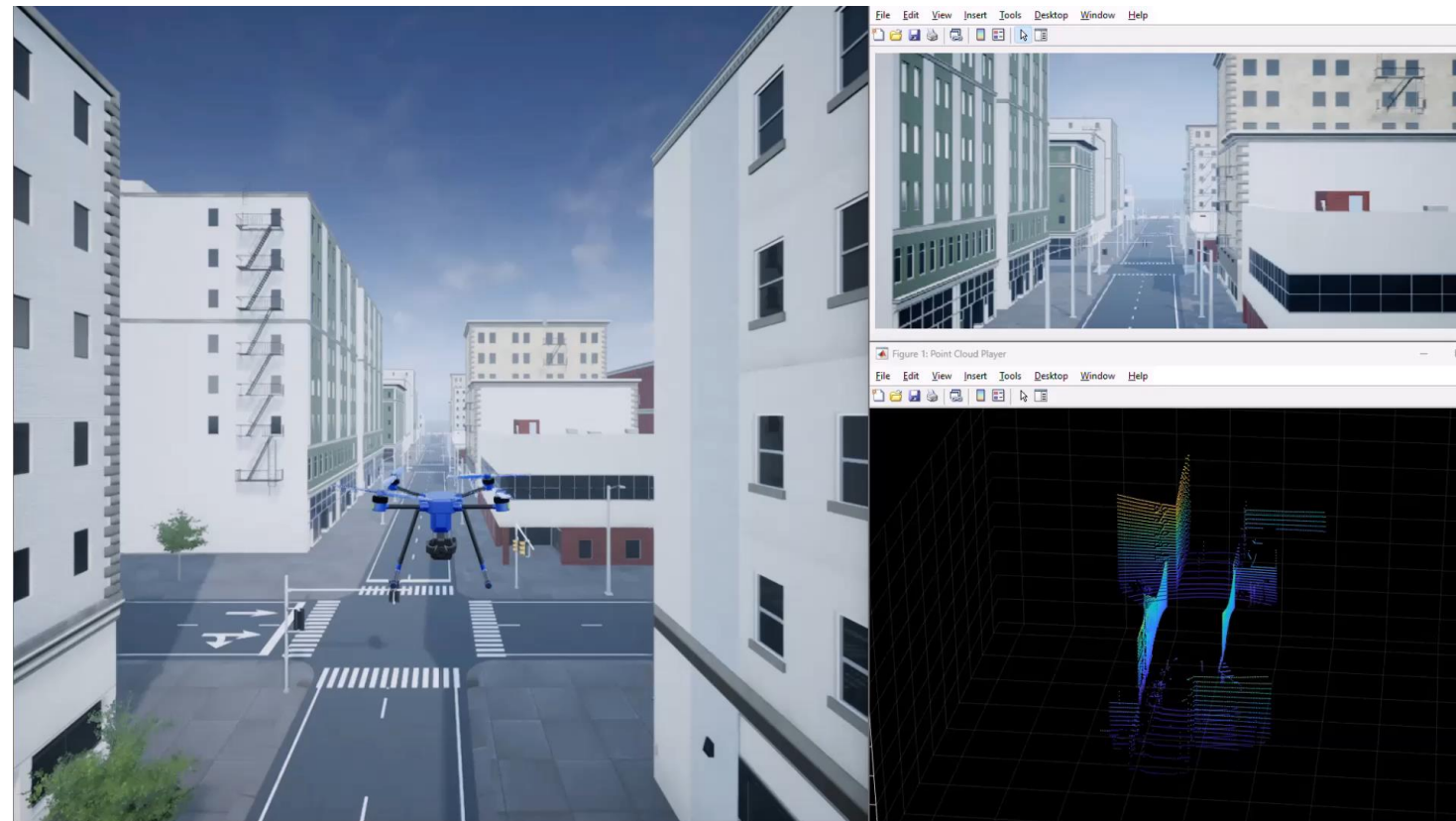
Realistické 3D
scény, akcie a
senzory pre
simuláciu
dynamických
systemov



Aerospace Blockset

Vizualizácia 3D simulácií

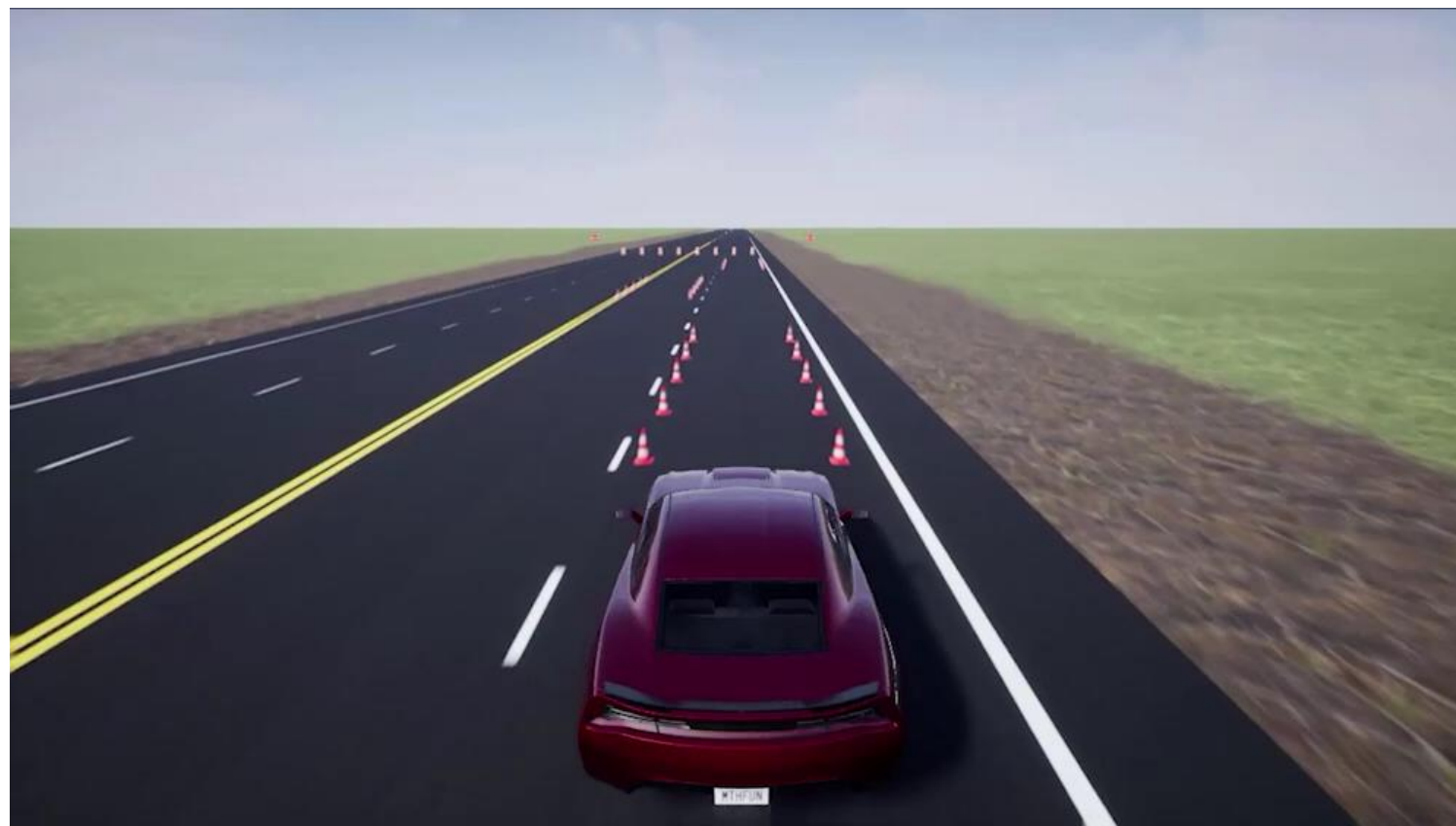
Realistické 3D scény, akcie a senzory pre simuláciu dynamických systemov



UAV Toolbox

Vizualizácia 3D simulácií

Realistické 3D
scény, akcie a
senzory pre
simuláciu
dynamických
systemov



Vehicle Dynamics Blockset

Vizualizácia 3D simulácií

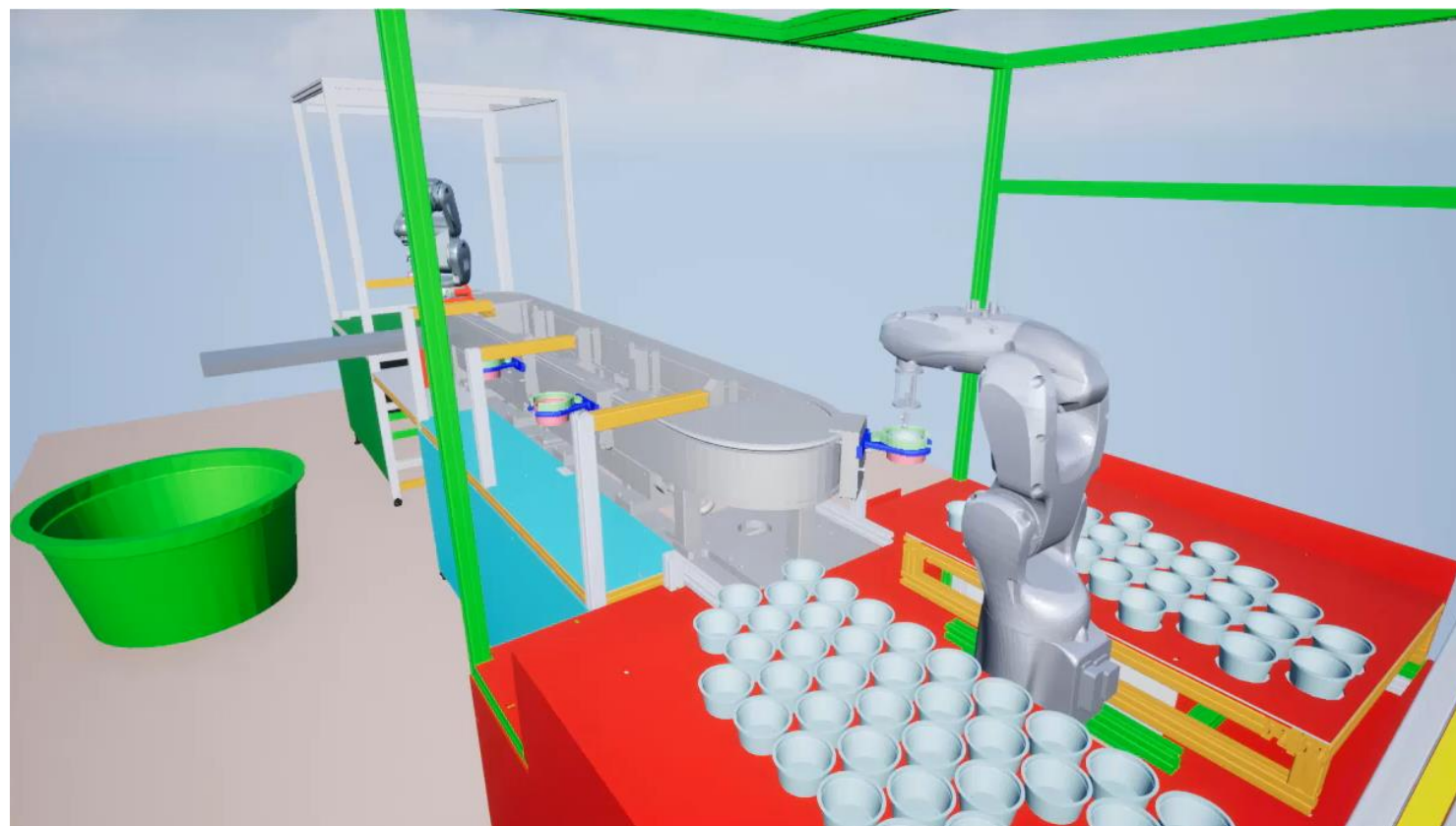
Realistické 3D
scény, akcie a
senzory pre
simuláciu
dynamických
systemov



Automated Driving Toolbox

Vizualizácia 3D simulácií

Realistické 3D
scény, akcie a
senzory pre
simuláciu
dynamických
systemov



Robotics System Toolbox



MATLAB®
& SIMULINK®



Integrácia



AI



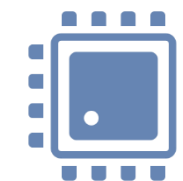
Jazyky



Simulácia

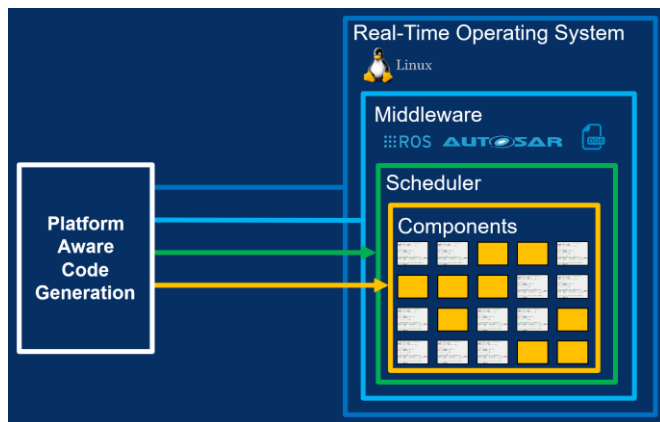


Vizualizácia

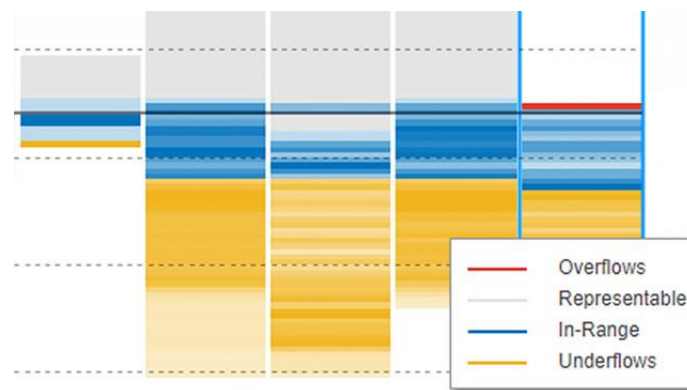


Hardvér

Simulink pre vývoj produkčného softvéru



Generovanie kódu podľa platformy



Numerická efektívnosť využívajúca paralelizmus

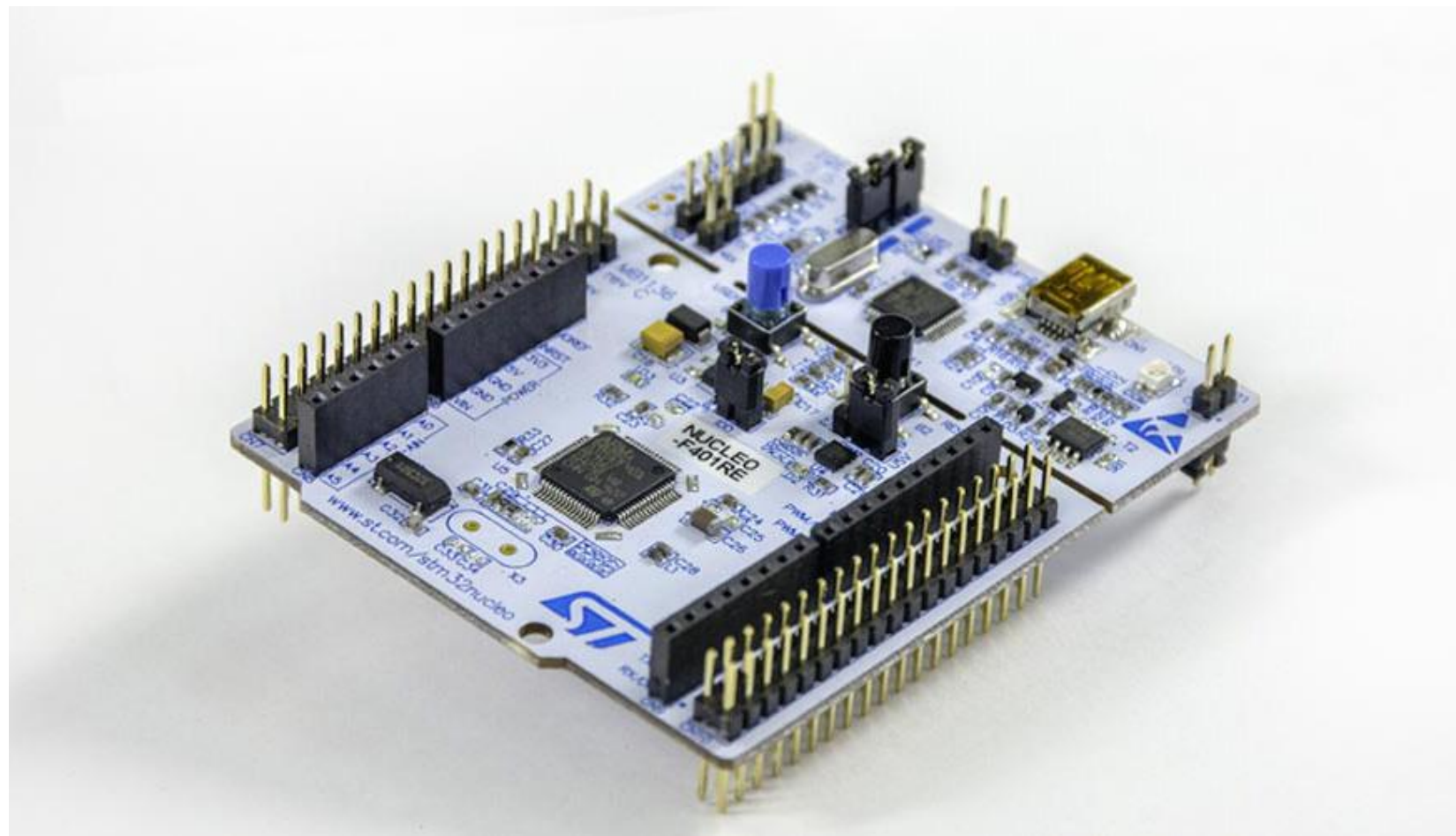


Podpora špecializovaného hardvéru

ST Microelectronics

Podpora pre dual-
core zariadenia

Podpora pre 4 nové
rodiny zariadení:
U5, L4, L5, WB



Infineon

Podpora pre **AURIX**
TC3x

Podpora pre **AURIX**
TC4x PPU
accelerator



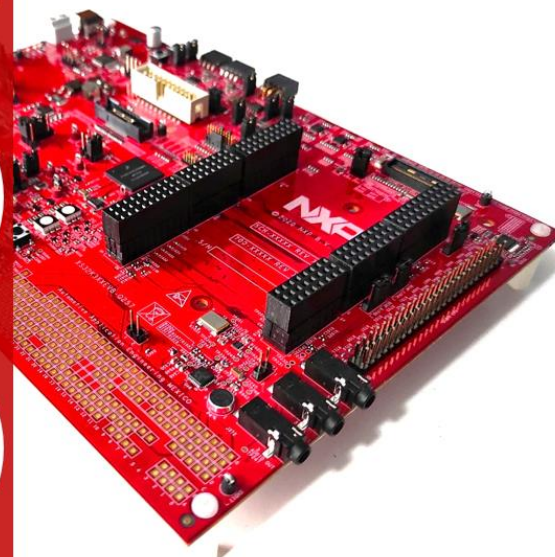
Embedded Coder

Embedded Coder Support Package for Infineon® AURIX™ TC3x Microcontrollers

NXP

Podpora pre S32M2,
S32K396, LAX
(S32R45), LPC553x
a BMS

Podpora pre S32K3,
S32ZE a HCP



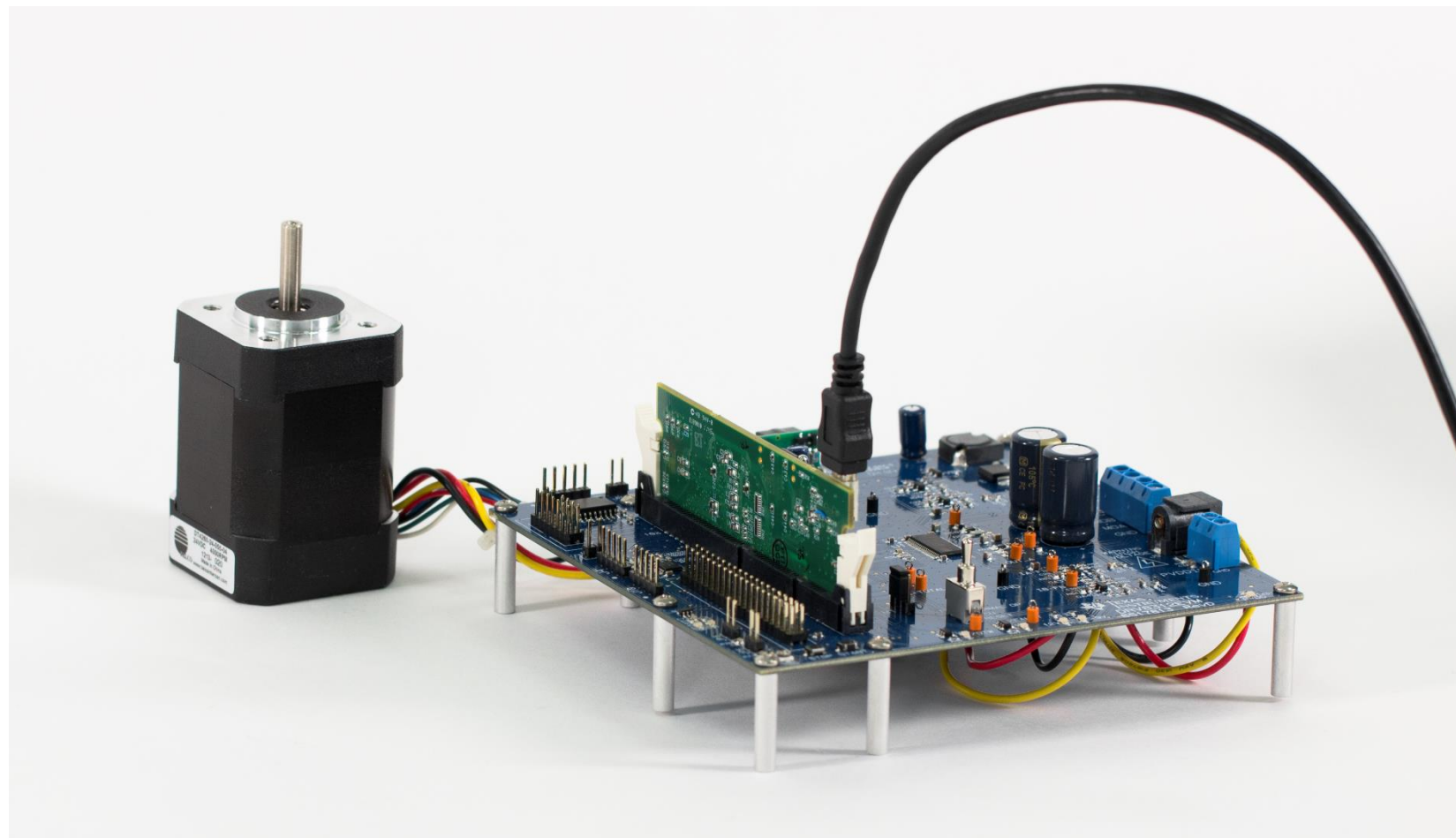
S32K3xx



HCP

Texas Instruments (TI)

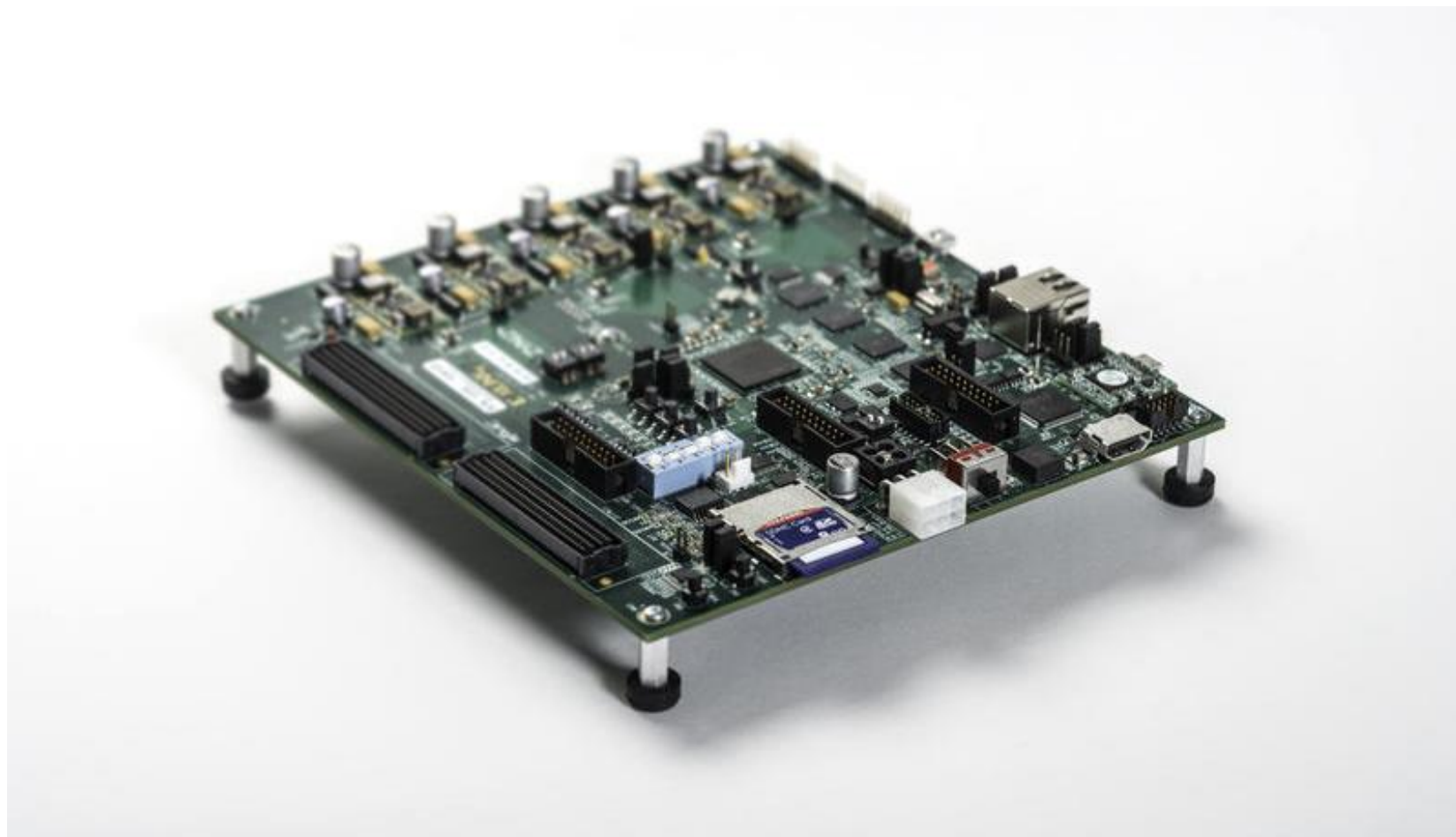
Návrh, simulácia a
implementácia
aplikácií pre TI
C2000
Microkontroléry



AMD

Návrh, analýza a
prototypovanie pre
Versal Adaptive SoC,
Zynq SoC a Xilinx FPGA
zariadenia

Generovanie a
nasadenie HDL kódu a
Embedded softvéru pre
Xilinx FPGA a SoC
zariadenia



SoC Blockset, HDL Coder



MATLAB®
& SIMULINK®



Integrácia



AI



Nové možnosti v rámci celého pracovného postupu AI

Príprava dát

AI Modelovanie

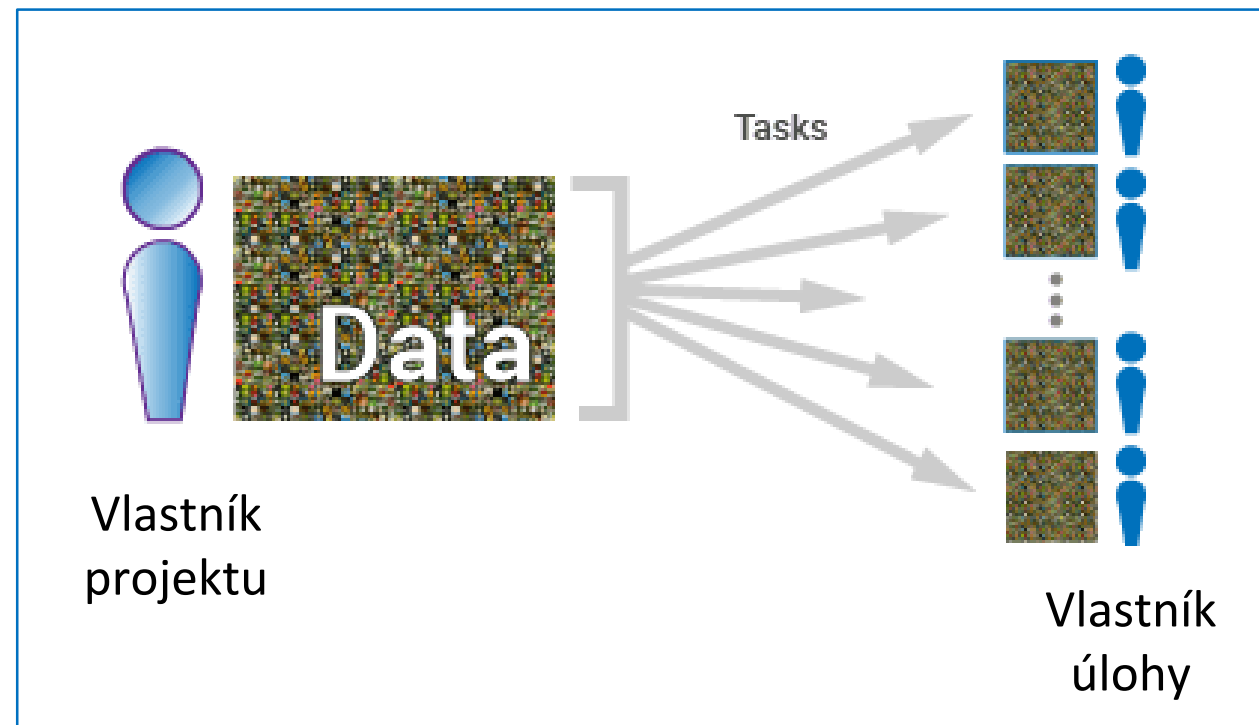
Simulácia
& Testovanie

Nasadenie

Kolaboratívne, viacužívateľské, tímové označovanie

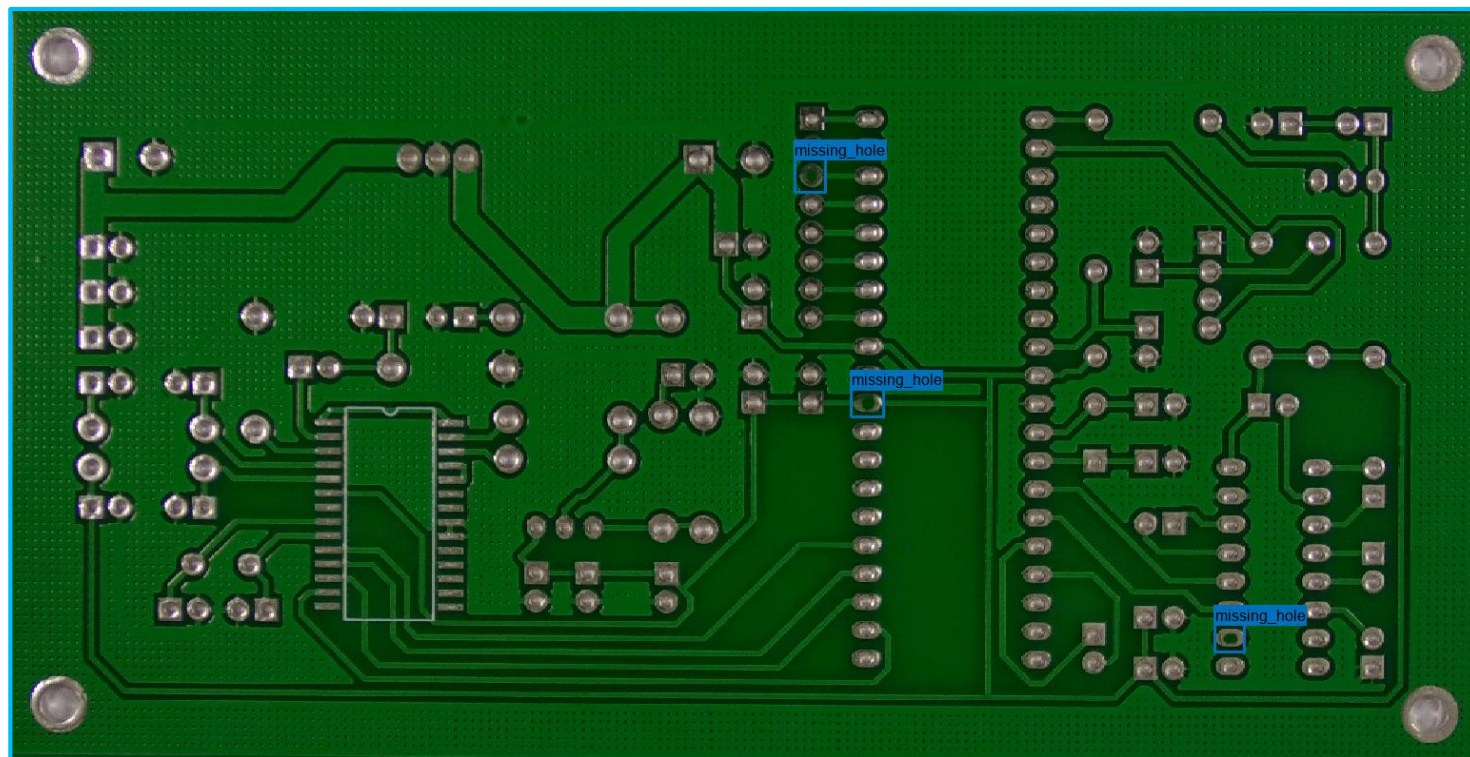
Distribúcia, monitorovanie a kontrola úloh označovania v rámci tímu

Tvorba spustiteľnej aplikácie na označovanie, kde členovia tímu môžu označovať a kontrolovať úlohy bez licencie prostredia MATLAB



Detekcia objektov v obrázkoch s YOLOX sieťami

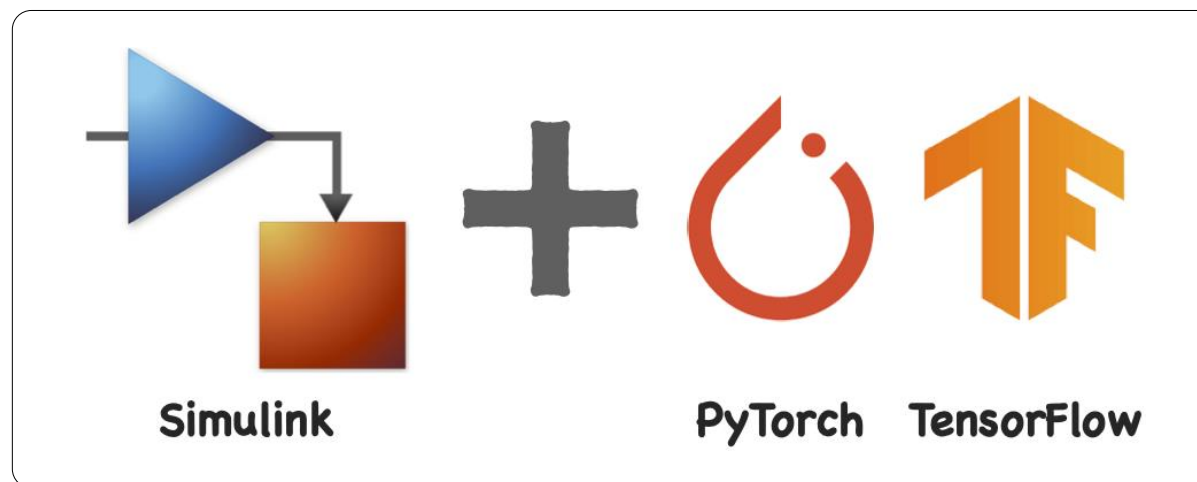
Tvorba predtrénovaných
alebo vlastných YOLOX
detektorov objektov



Príklad automatizovanej vizuálnej inšpekcie
Detekcia, lokalizácia a klasifikácia defektov
na doskách plošných spojov (PCB) s využitím
YOLOX detektora objektov

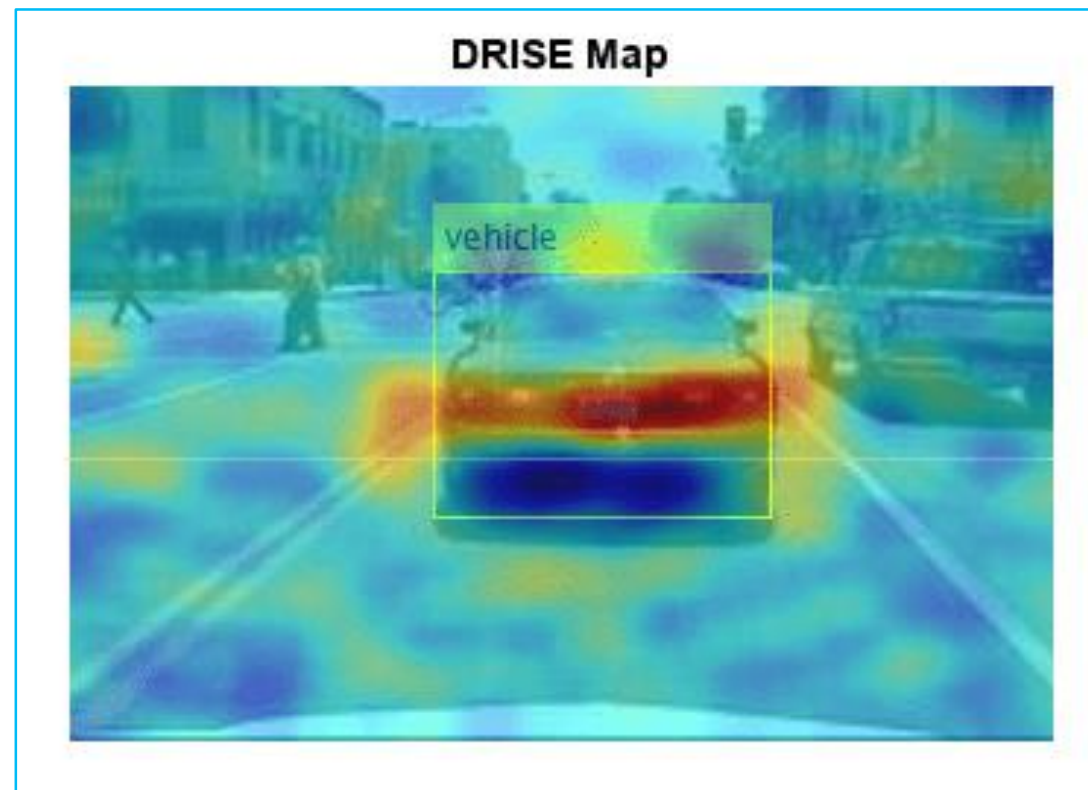
Spúšťanie modelov CNN z Pythonu v prostredí Simulink

Spolupráca TensorFlow a PyTorch modelov v Simulinku spolu s inými blokmi pre deep learning a machine learning



Vysvetlenie predikcie detekcie objektov pomocou D-RISE

Generovanie vizuálneho vysvetlenia pre predikciu výsledkov siete na detekcie objektov



Generovanie všeobecného CUDA kódu pre deep learning

Generovanie CUDA kódu pre deep learning, ktorý nepotrebuje knižnice NVIDIA pre deep learning





MATLAB®
& SIMULINK®



Integrations



AI



Ease of Use



Performance



Verification



Languages



Simulation



Visualization



Hardware

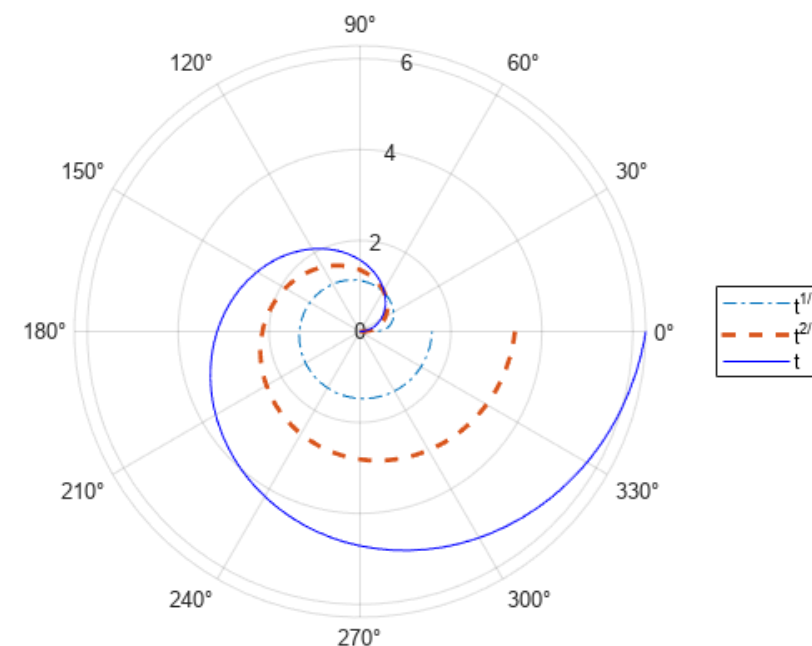


MATLAB®
PRODUCT FAMILY

SIMULINK®
PRODUCT FAMILY

Matematika a optimalizácia

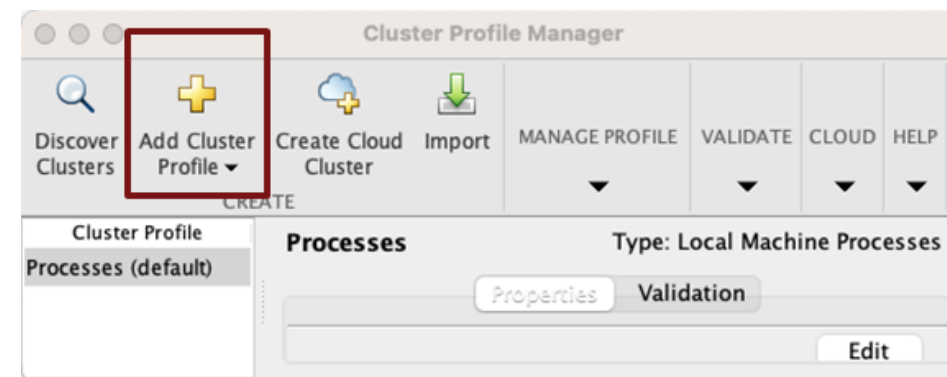
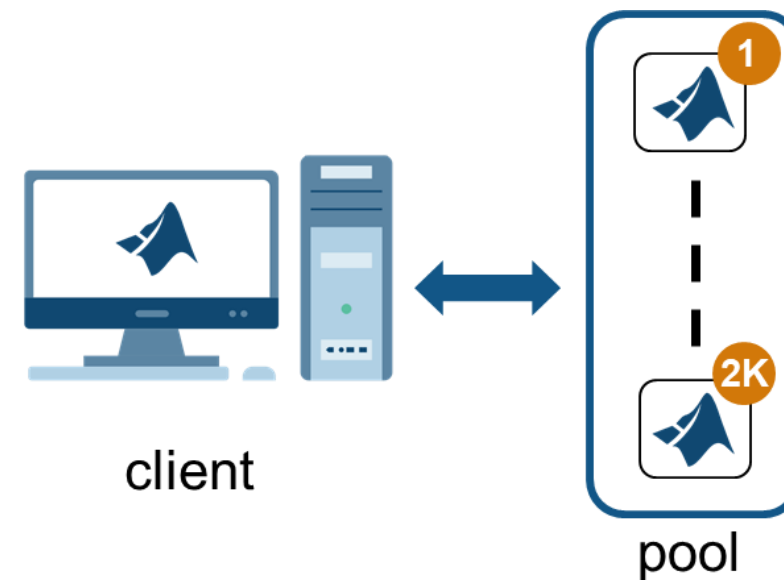
- Funkcia `fplot`
 - symbolická funkcia – polárne súradnice
- HiGHS algoritmus
 - nový východzí algoritmus `intlinprog` a `linprog`
 - rýchlejšie a spoľahlivejšie výsledky
- Urýchlenie `surrogateopt`
 - úlohy s celočíselnými obmedzeniami
- Generovanie kódu pre `fmincon`
 - podpora hardvéru - single-precision floating point
- Funkcie pre objekty máp
 - vypočet plôch, obvodov a dĺžky čiar



$$\min_x 8x_1 + x_2 \text{ subject to } \begin{cases} x_2 \text{ is an integer} \\ x_1 + 2x_2 \geq -14 \\ -4x_1 - x_2 \leq -33 \\ 2x_1 + x_2 \leq 20. \end{cases}$$

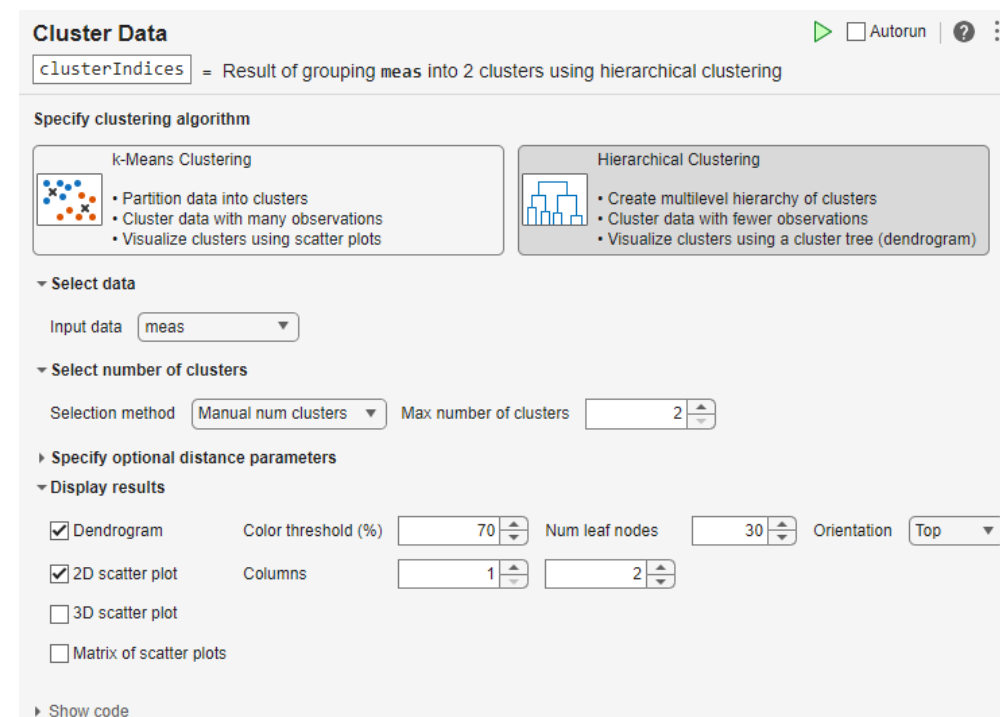
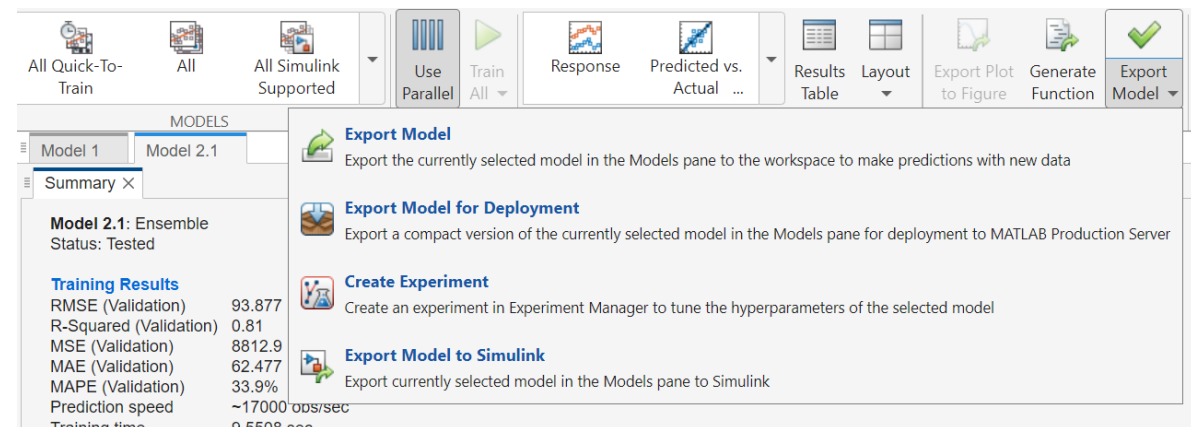
Paralelné výpočty

- Škálovateľnosť
 - Parallel pools – do 2000 workerov
 - MATLAB Job Scheduler – 10000 workerov
 - parfor – špecifikácia počtu Threadov
- Administrácia – História jobov
 - MATLAB Job Scheduler
- Profil klástra
 - AWS® Batch, Grid Engine, HTCondor
- GPU výpočty – CUDA 12.2
- Podpora gpuArray
 - 30+ vylepšení v MATLABe a 5 Toolboxoch



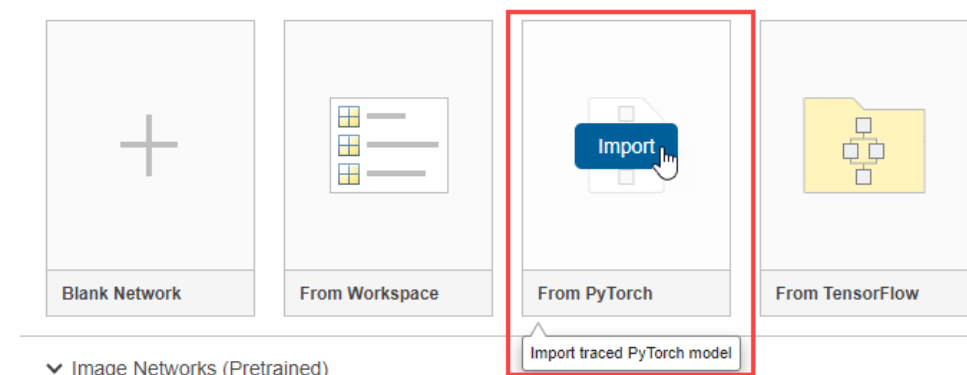
Strojové učenie

- Cluster Data Live Editor Task
 - hierarchické kláštrovanie
- Export modelov do Simulinku
 - Classification Learner, Regression Learner
- Aplikácie majú 3 záložky
 - Learn, Test, a Explain
- Direct forecasting
 - samostatné modely pre kroky horizontu
- Multivariate analysis of variance (MANOVA)
- Podpora ďalších modelov v Simulinku



Deep Learning

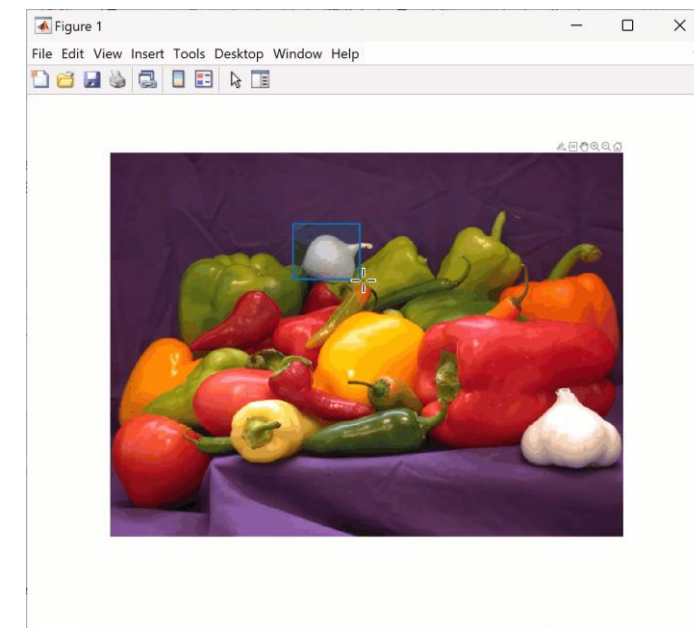
- Neural Network Analyzer
 - prístup k výsledkom z príkazového riadku
- Nový pracovný postup
 - funkcia trainnet a dlnetwork objekty
 - širšia podpora architektúr
- Transfer Learning
 - funkcia imagePretrainedNetwork
- Deep Network Designer
 - import modelov PyTorch, TensorFlow
 - odomknutie učiacich sa parametrov vrstiev
- Trénovanie siete pomocou L-BFGS solvera



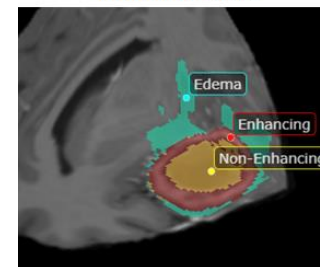
```
net = imagePretrainedNetwork("squeezeNet", NumClasses=numClasses)
options = trainingOptions("adam", ...
    ValidationData=augimdsValidation, ...
    ValidationFrequency=5, ...
    Plots="training-progress", ...
    Metrics="accuracy", ...
    Verbose=false);
net = trainnet(augimdsTrain,net,"crossentropy",options);
```

Spracovanie obrazu a počítačové videnie

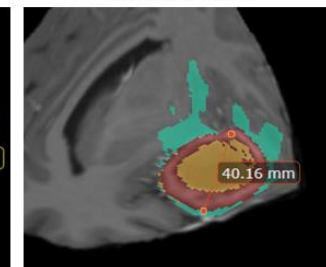
- Podpora formátu OpenLABEL
 - groundTruth na ASAM OpenLABEL (JSON)
- Segmentácia
 - Segment Anything Model (SAM)
- Volumetrická vizualizácia
 - renderovanie, anotácie
- Image Viewer App
 - interaktívne ROI, meranie vzdialeností a plôch
- Hyperspektrálne obrázky
 - práca s výrezom obrázku
 - predspracovanie, detekcia, segmentácia



Point Annotations



Line Annotation



Volumetric Rendering

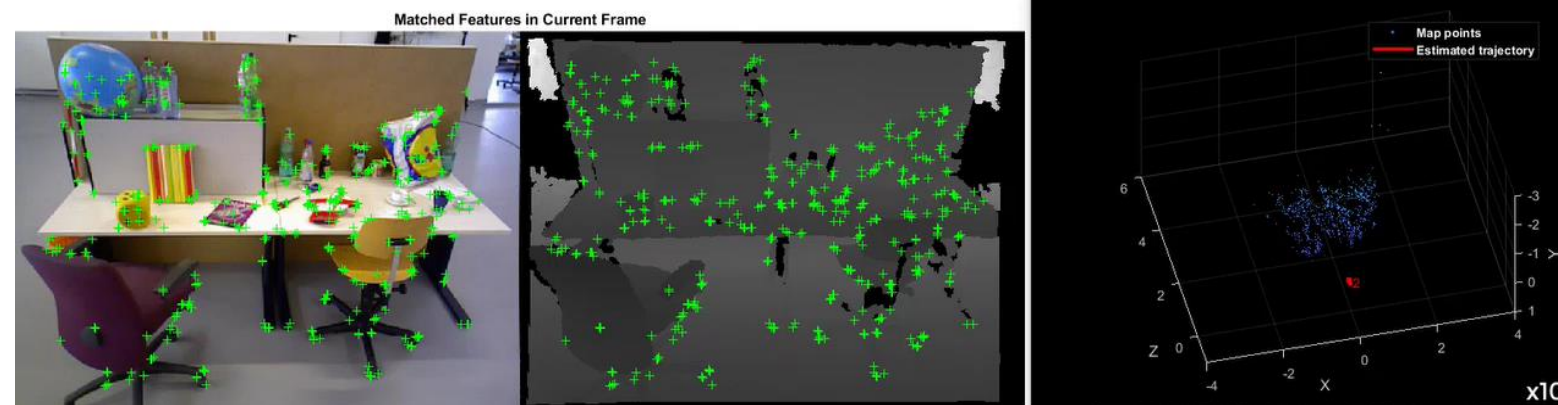
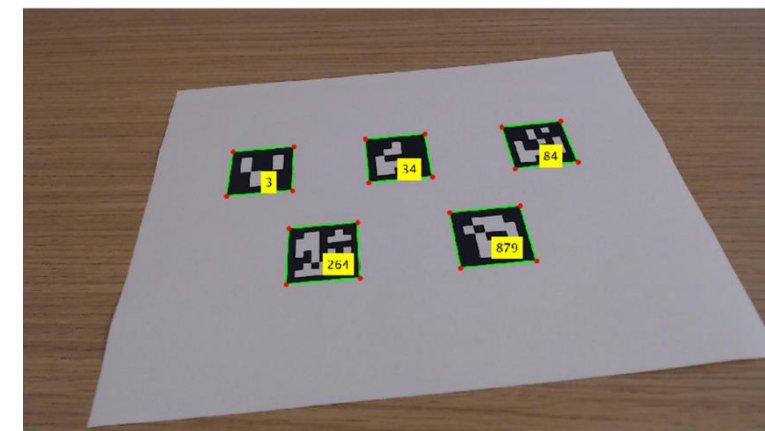


Cinematic Rendering



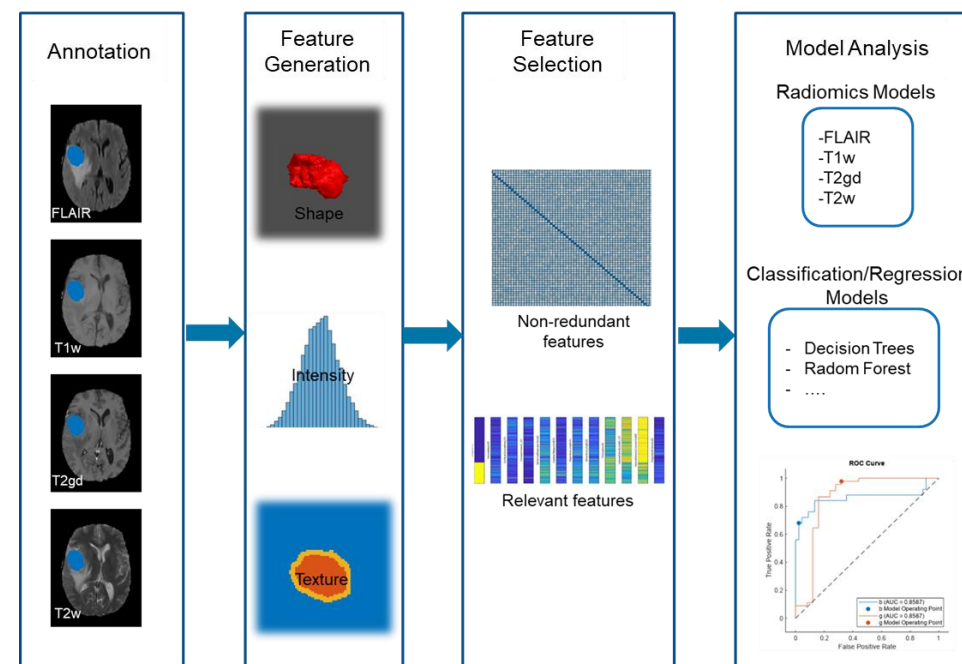
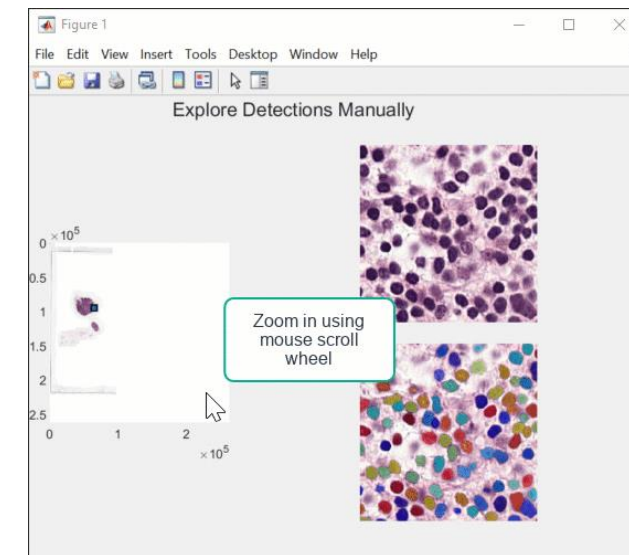
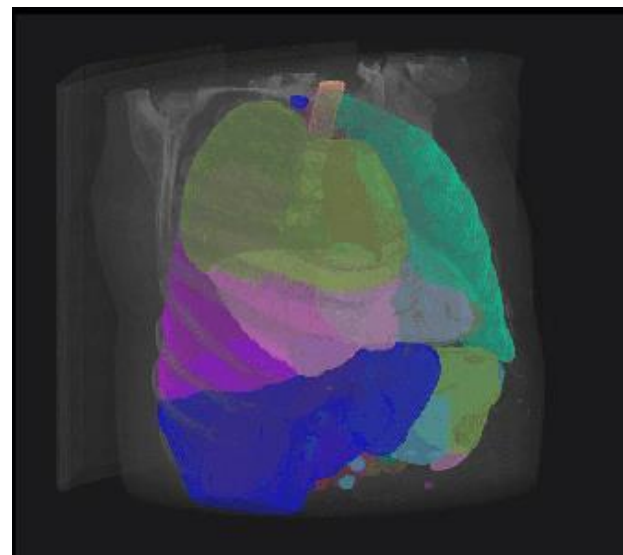
Počítačové videnie

- ArUco popisky
 - čítanie značiek a odhad pozície
- Opätovná identifikácia a sledovanie objektu
 - multi-object tracking pomocou AI
- Monocular vSLAM
 - monovslam - monokulárna kamera
- Stereo vSLAM
 - stereovslam - stereo kamera
- RGB-D SLAM
 - rgbdvslam - RGB-D kamera



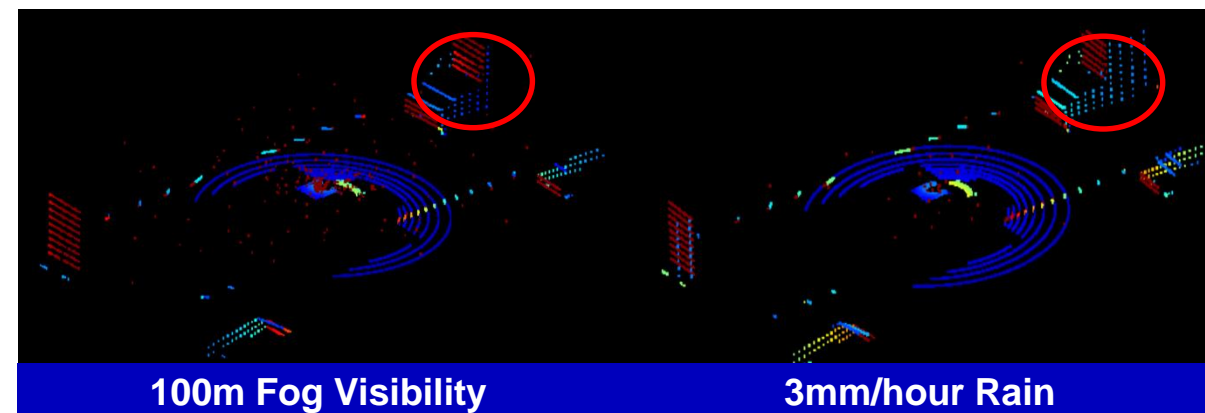
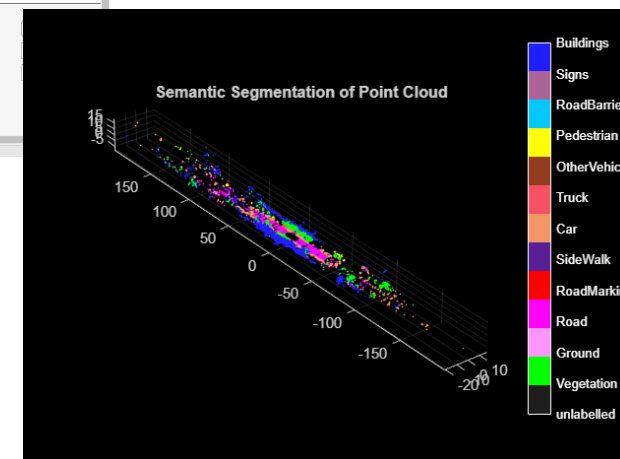
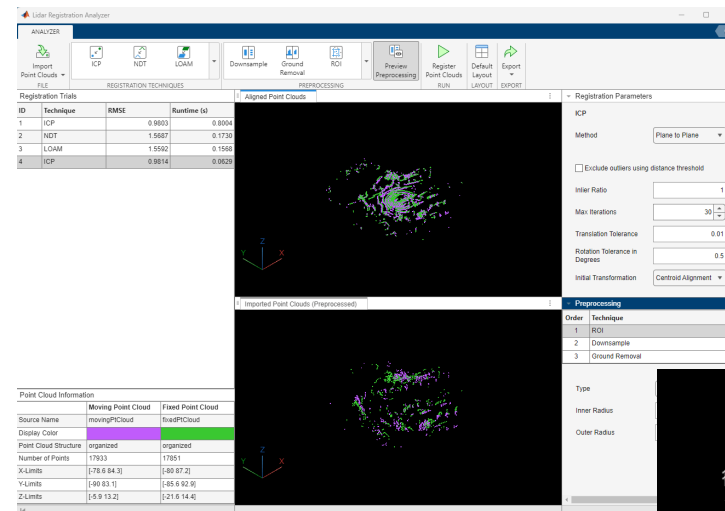
Medicínske obrázky

- Segmentácia orgánov a kostí
 - Medical Open Network for AI (MONAI)
 - automaticky alebo interaktívne
 - vyžaduje podporný balíček
- Segmentácia buniek pomocou AI
 - Interface for Cellpose Library
 - natrénované alebo vlastné siete
- Rádiomika
 - extrakcia príznakov z obrázkov (IBSI norma)
 - tvar
 - rozloženie intenzity
 - textúra



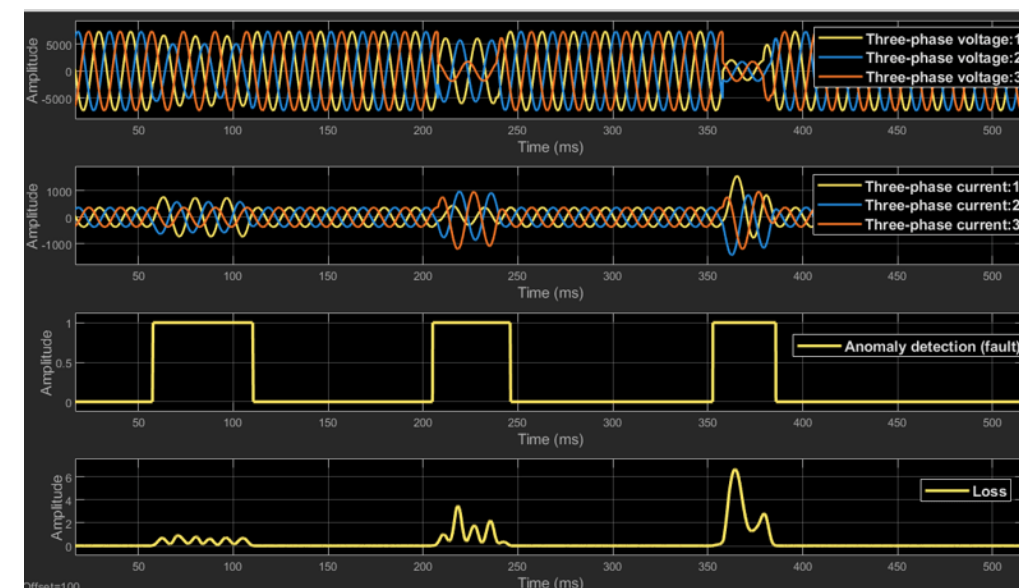
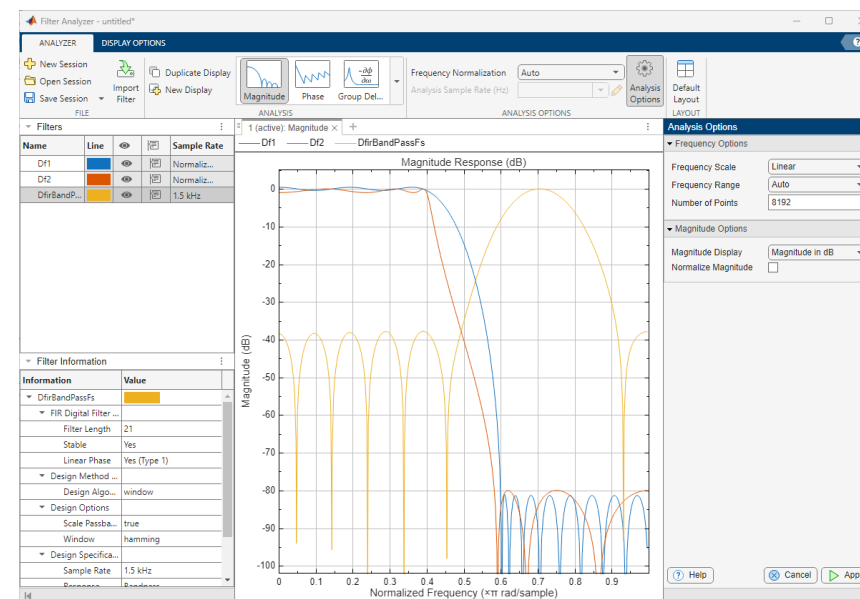
Práca s Lidarom

- Lidar Registration Analyzer App
 - ladenie algoritmov registrácie
 - import, vizualizácia, predspracovanie
- Sémantická segmentácia
 - RandLA-Net vhodná pre veľké mračná bodov
 - elektrické vedenia – letecké dáta
- Simulácia snímača
 - simulácia skreslenia pohybu
 - pridanie do RoadRunner Scenario
- Import a export (FBX, OBJ)
- Podpora Hokuyo zariadení



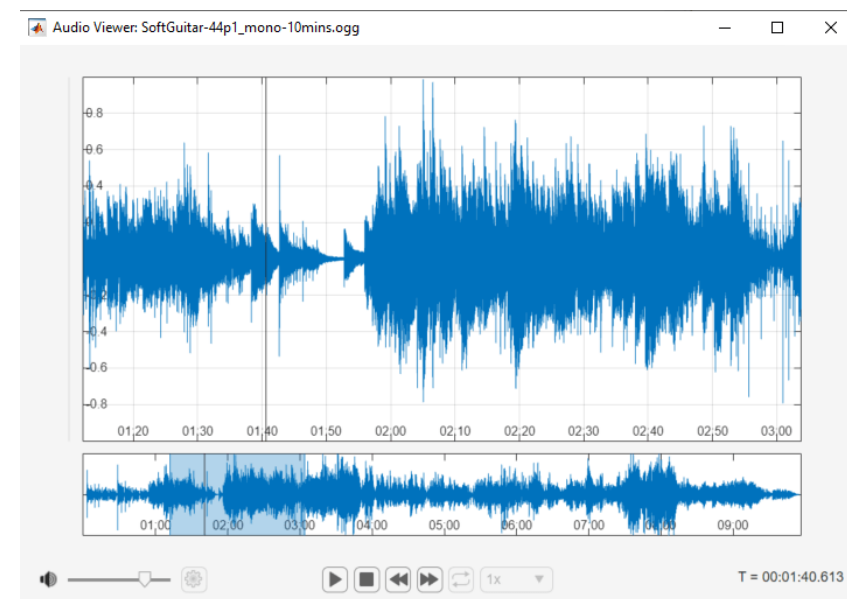
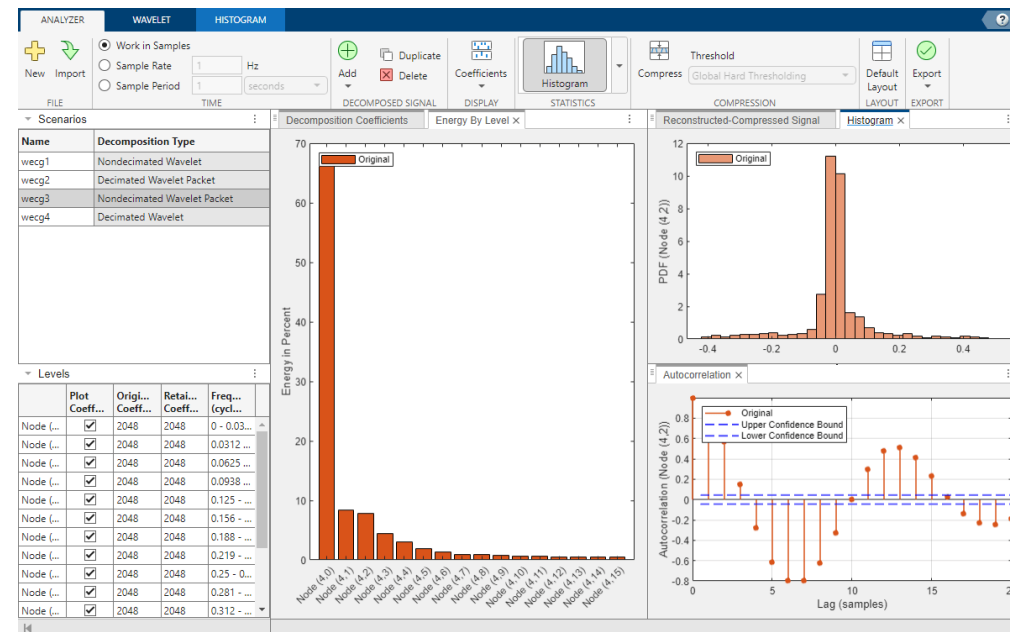
Spracovanie signálov

- Filter Analyzer app
 - zobrazenie, analýza a porovnanie filtrov
- Výber prediktorov pre AI
 - objekt `signalTimeFrequencyFeatureExtractor`
- Signal Analyzer App
 - nahradenie chýbajúcich dát
 - prehrávanie audia
- blok Deep Signal Anomaly Detector
 - detekcia pomocou LSTM v reálnom čase
- Ladenie parametrov filtra
 - plynulý prechod k požadovanej hodnote



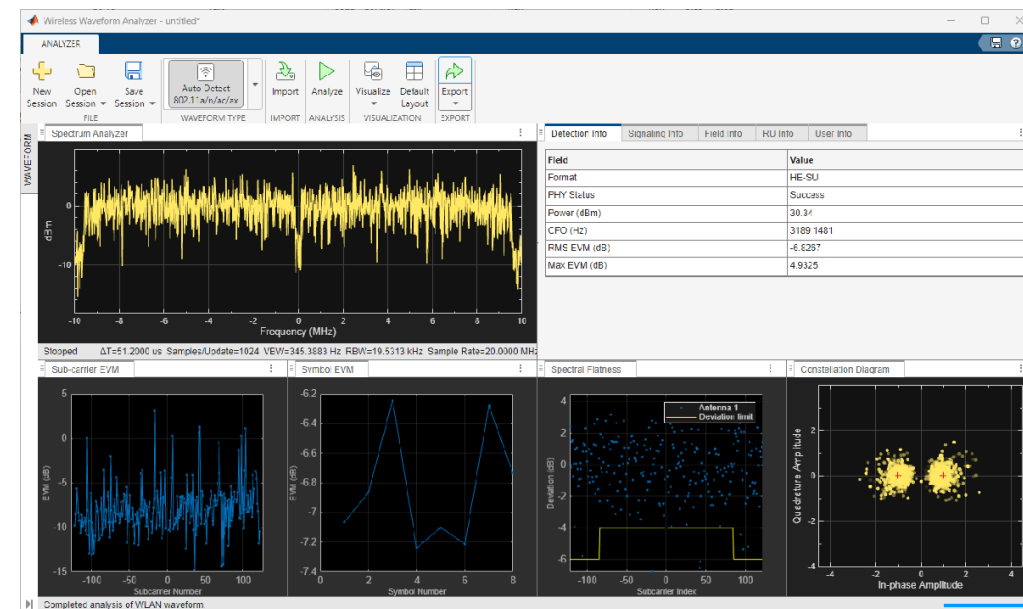
Spracovanie signálov a audia

- Wavelet Signal Analyzer App
 - nové transformácie
 - autokorelácia, variancia koeficientov
- Kvalita a zrozumiteľnosť reči
 - Virtual Speech Quality Objective Listener (ViSQOL)
 - Short-Time Objective Intelligibility (STOI)
- Prepis a syntéza reči
 - rozhranie s modelmi SpeechBrain a TorchAudio
- Oddelenie rečníkov
- Extrakcia príznakov z audia pre AI
- Audio Viewer app



Bezdrôtové komunikácie

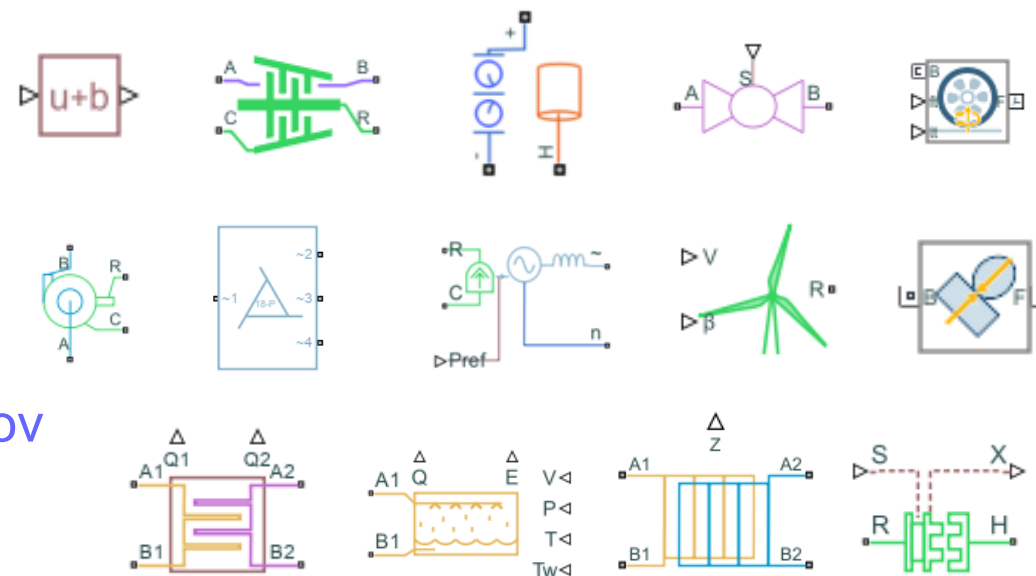
- 6G Exploration Library for 5G Toolbox
 - modelovanie a simulácia 6G waveforms
 - pokročilé funkcie pre PDSCH, PUSCH a SRS
- WLAN Waveform Analyzer app
 - analýza, vizualizácia a meranie WLAN waveforms
 - modelovanie IEEE 802.11be multi-link operation (MLO)
- Satellite Scenario
 - platformy (autá, lode, ...) – prístup a komunikácia
 - generovanie trajektórií na základe bodov
- Podpora ďalších zariadení na testovanie algoritmov



Fyzikálne modelovanie

- Základný modul

- podpora N-rozmerných signálov a premenných
- selektívne logovanie – individuálne premenné blokov
- inkrementálne kompilovanie – iba pre zmenu
- kompilácia znovupoužiteľných komponentov vo vlákne



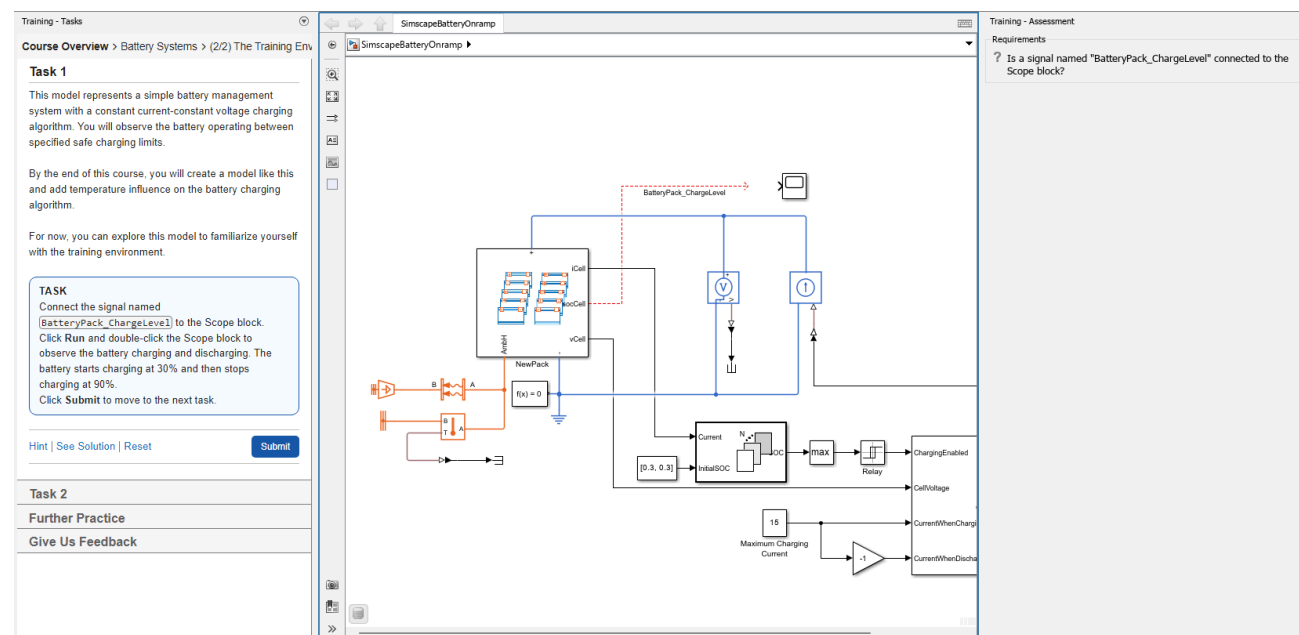
- Modelovanie porúch

- systematické modelovanie a analýza

- Nové a vylepšené bloky

- Interaktívne kurzy

- Simscape Battery Onramp
- Power Systems Simulation Onramp



Training - Tasks

Course Overview > Battery Systems > (2/2) The Training Env

Task 1

This model represents a simple battery management system with a constant current-constant voltage charging algorithm. You will observe the battery operating between specified safe charging limits.

By the end of this course, you will create a model like this and add temperature influence on the battery charging algorithm.

For now, you can explore this model to familiarize yourself with the training environment.

TASK
Connect the signal named `BatteryPack_ChargeLevel` to the Scope block. Click Run and double-click the Scope block to observe the battery charging and discharging. The battery starts charging at 30% and then stops charging at 90%. Click Submit to move to the next task.

Hint | See Solution | Reset

Task 2

Further Practice

Give Us Feedback

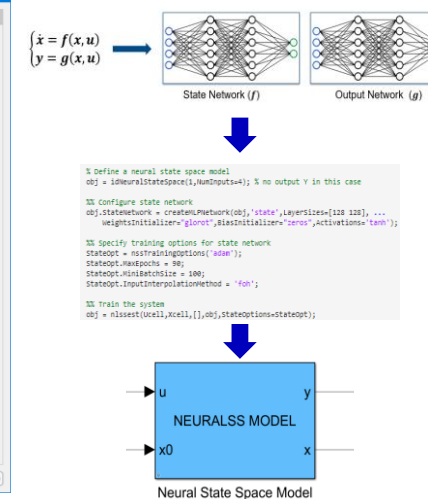
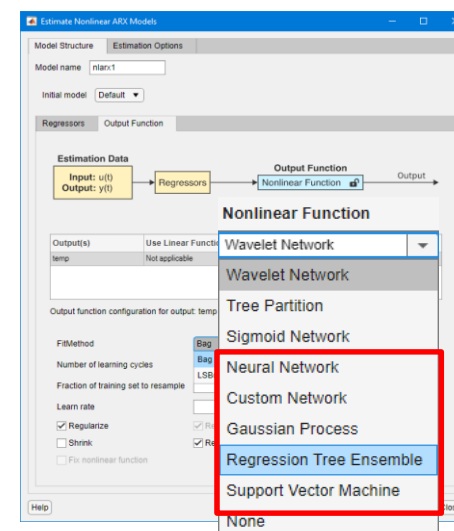
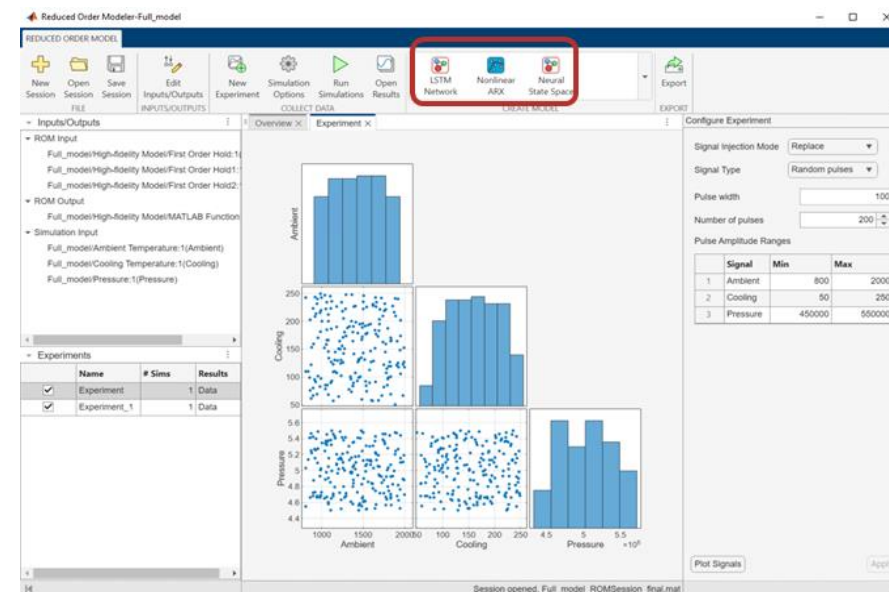
Training - Assessment

Requirements

? Is a signal named "BatteryPack_ChargeLevel" connected to the Scope block?

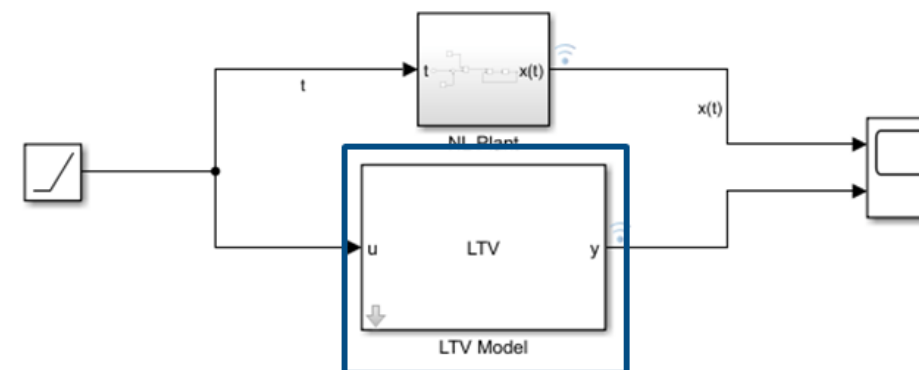
Identifikácia systémov

- Reduced Order Modeler app
 - redukované modely založené na AI
 - urýchlenie simulačných modelov
 - tvorba experientov, tréningovanie a export modelu
- Nonlinear ARX, Hammerstein-Wiener model
 - neurónové siete
 - strojové učenie
 - inicializácia modelu v System Identification app
- Neural state-space model
 - nelineárny stavový model založený na AI
 - neural state-space live task



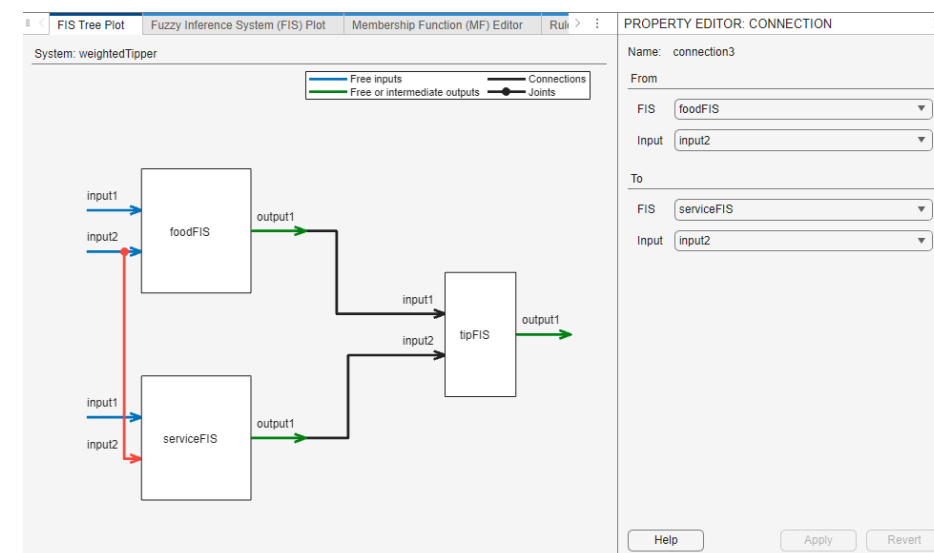
Riadiace systémy

- Control System Toolbox
 - blok LTV System – lineárny časovo-variantný
 - sparse state-space models – Model Reducer app
- Model Predictive Control Toolbox
 - riešiče pre urýchlenie MPC
 - alternating direction method of multipliers (ADMM)
 - C/GMRES
- Fuzzy Logic Toolbox
 - tvorba a ladenie FIS stromov
 - blok Simulinku pre FIS stromy
 - generovanie MATLAB kódu z FIS



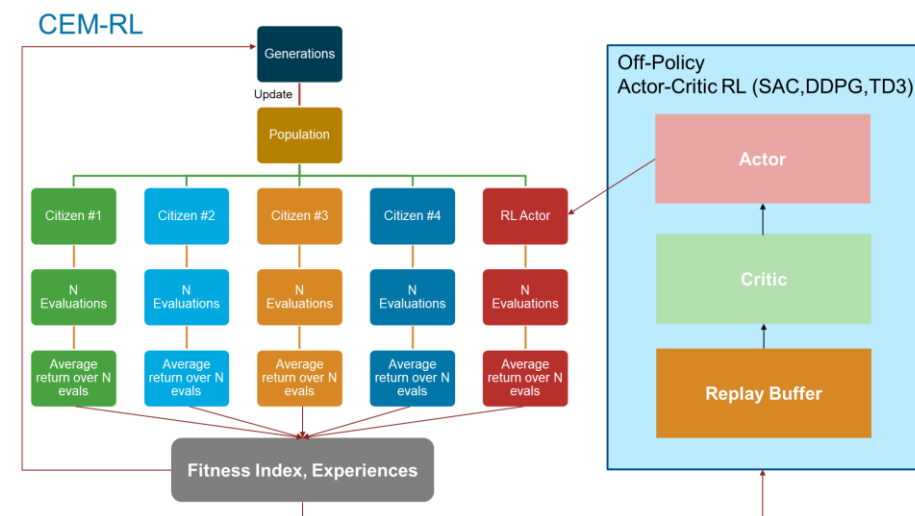
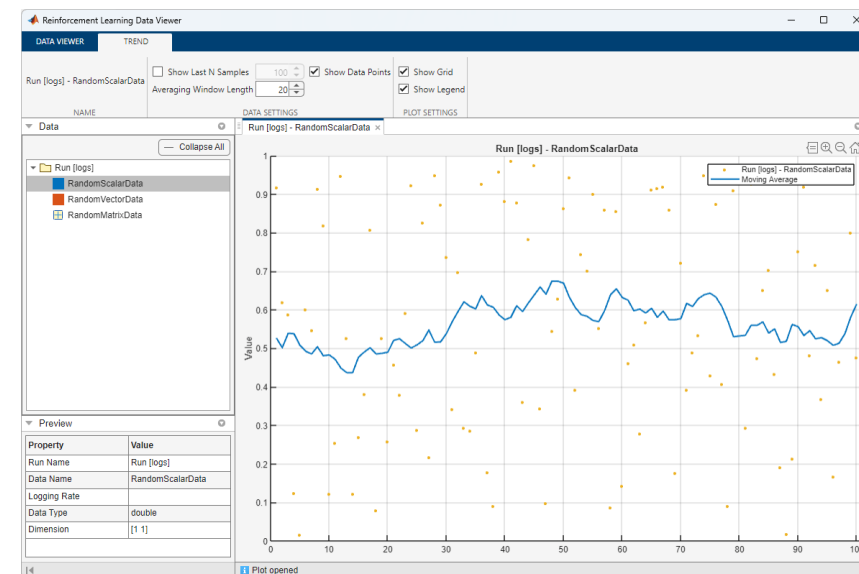
Syntax

```
ltvSys = ltvss(DataFcn)
ltvSys = ltvss(DataFcn,ts)
ltvSys = ltvss(DataFcn,ts,tcheck)
ltvSys = ltvss(__,Name=Value)
```



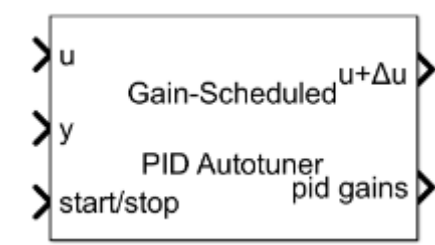
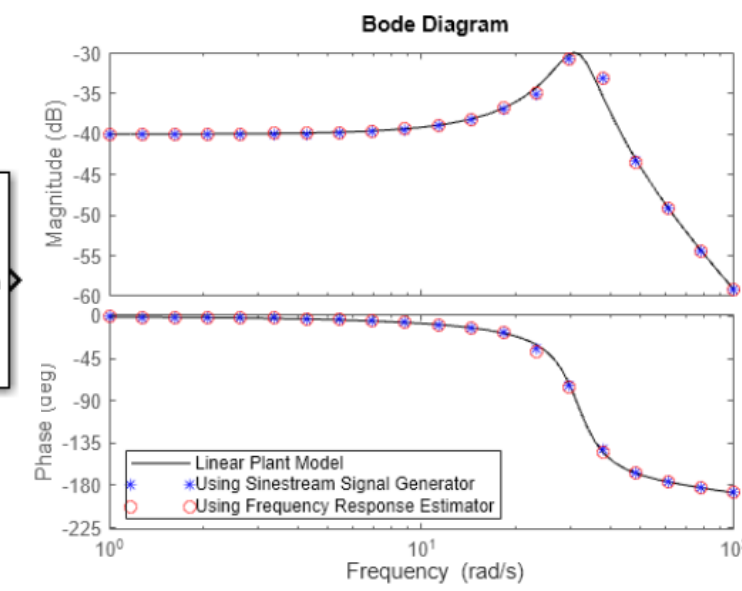
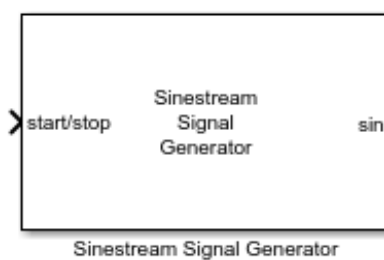
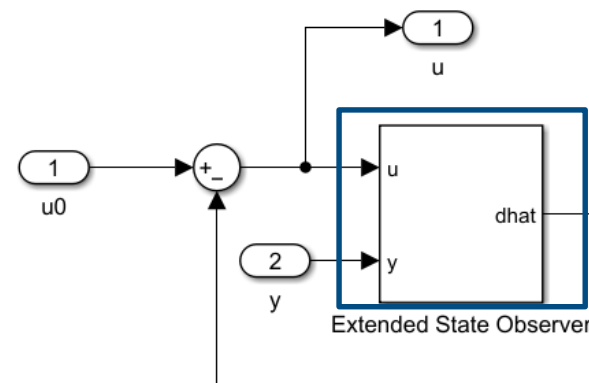
Reinforcement Learning

- Normalizácia vstupov
 - stabilnejšie trénovanie
- Reinforcement Learning Data Viewer
 - vizualizácia logovaných dát
- Vylepšenie trénovania
 - DQN, DDPG, TD3, SAC, PPO, TRPO
- Evolutionary reinforcement learning
 - stratégie pre efektívnejšie výpočty
- Multi-agent reinforcement learning
 - trénovanie viacerých agentov v MATLABe



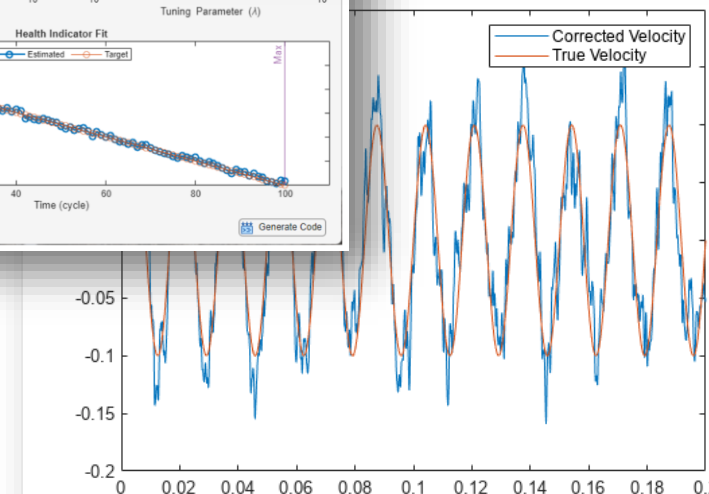
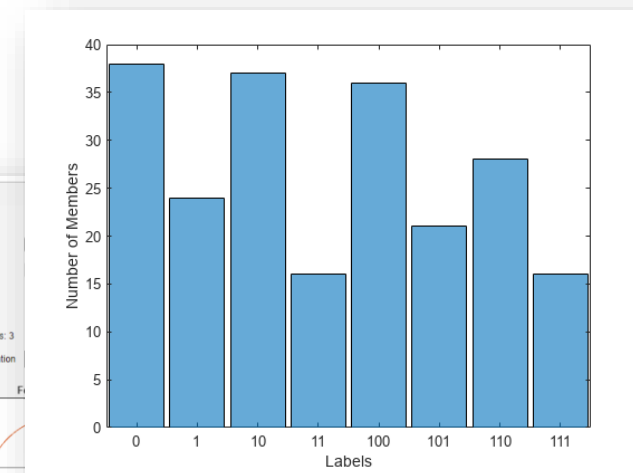
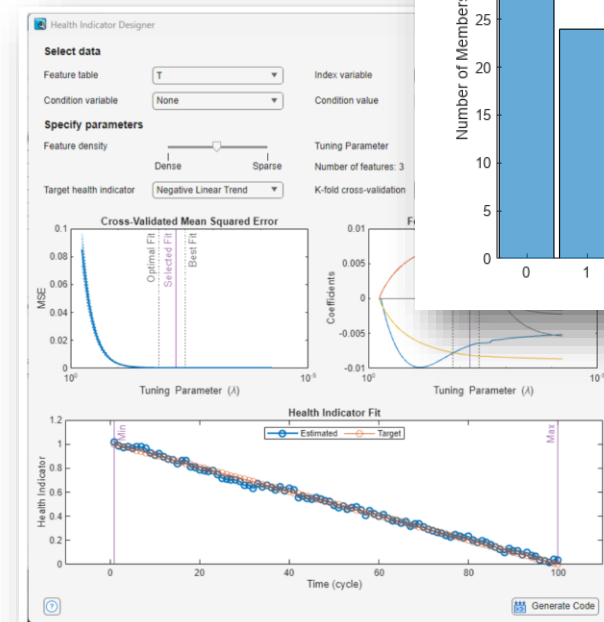
Optimalizácia riadenia a návrhu

- Bloky pre prácu s poruchami
 - Extended State Observer
 - Disturbance Compensator
- Odhad frekvenčnej odozvy
 - Sinestream Signal Generator
 - PRBS Signal Generator
- Ladenie v reálnom čase
 - Gain-Scheduled PID Autotuner
 - Closed-Loop PID Autotuner
- Simulink Design Optimization
 - optimalizácia s referenčnými konfiguráciami



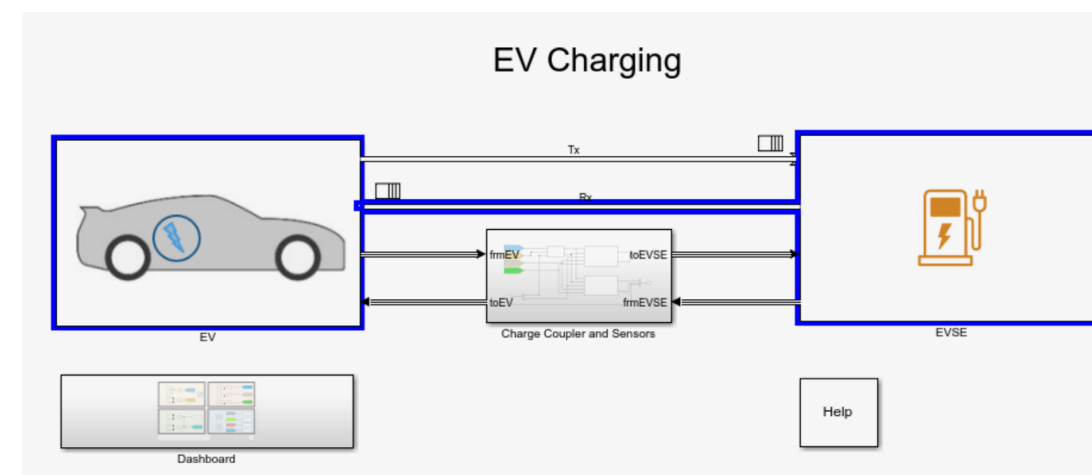
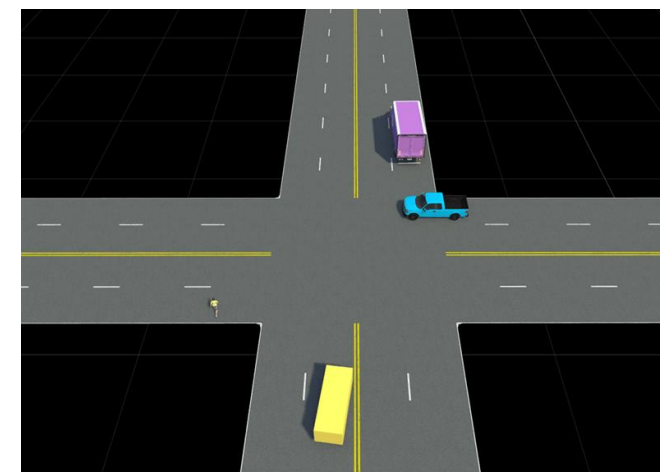
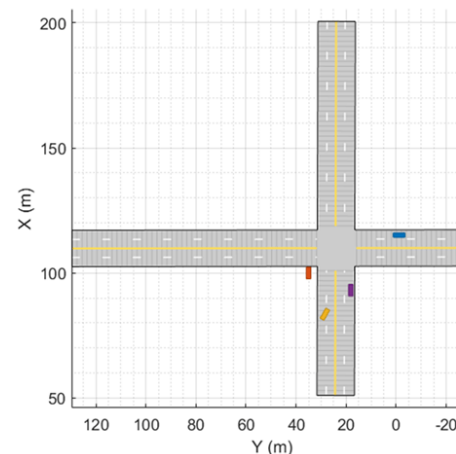
Prediktívna údržba

- Zjednodušenie monitorovania RUL
 - Health Indicator Designer app
 - kombinácia indikátorov do jedného
 - generovanie kódu pre MATLAB
- Vyvážené dátové podmnožiny
 - funkcia subset nepotrebuje indexi explicitne
- Konverzia vibračných dát na ISO štandard
 - funkcia `convertVibration`
- Extrakcia prediktorov v aplikáciách a taskoch
 - Diagnostic Feature Designer
 - Extract Spectral Features Live Editor task



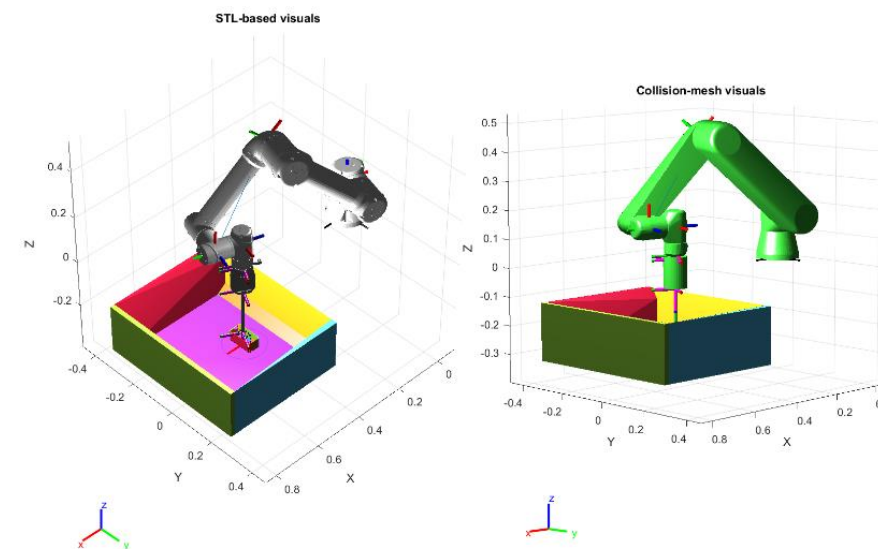
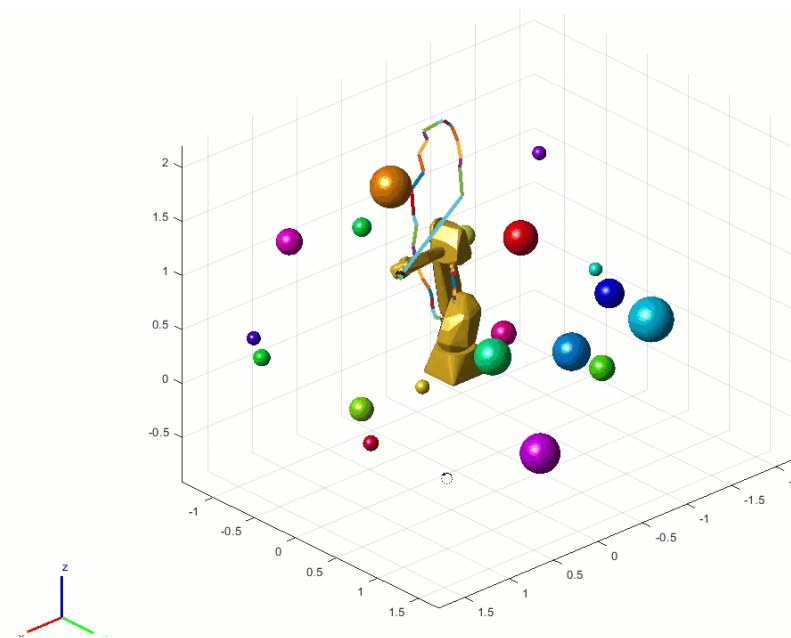
Automobily

- Export scenára do RoadRunnera
 - drivingScenario – cesty, dynamické objekty
- Import RoadRunner scény
 - exportované FBX súbory
- Referenčná aplikácia nabíjania
 - nabíjanie EV pomocou NACS (Tesla)
- Virtual Vehicle Composer
 - vylepšenie rozhranie a rýchlosť
- Práca s modelmi pneumatík
 - Extended Tire Features support package



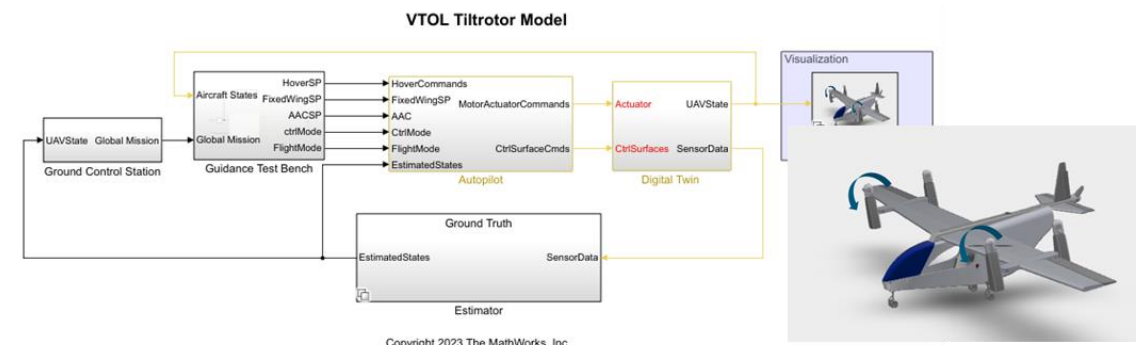
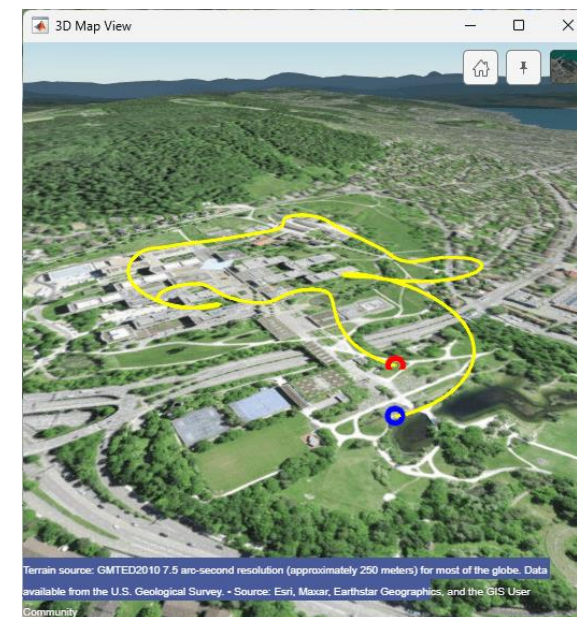
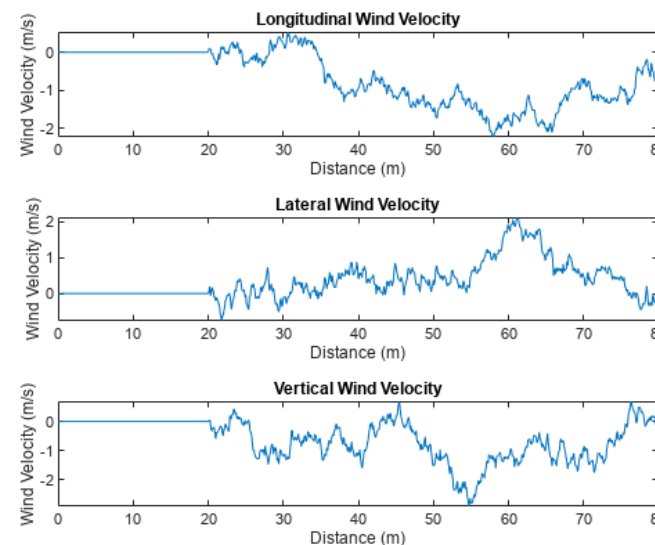
Autonómne systémy a robotika

- Podpora robotov
 - Techman manipulator robots
 - rozšírenia pre Universal Robots
- Optimalizácia trajektórie
 - dICHOMP – urýchlenie plánovania s AI
- Detekcia kolízií
 - dekompozícia nekonvexných meshov
 - geometria na štruktúru – pre plánovanie
- Floating Joints
 - 6-DOF dynamika a kinematika
- Generovanie UDRF z rigidBodyTree objektu



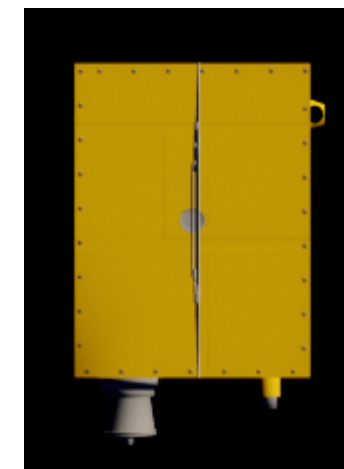
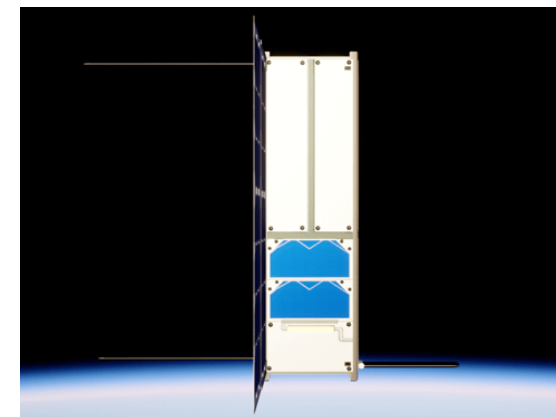
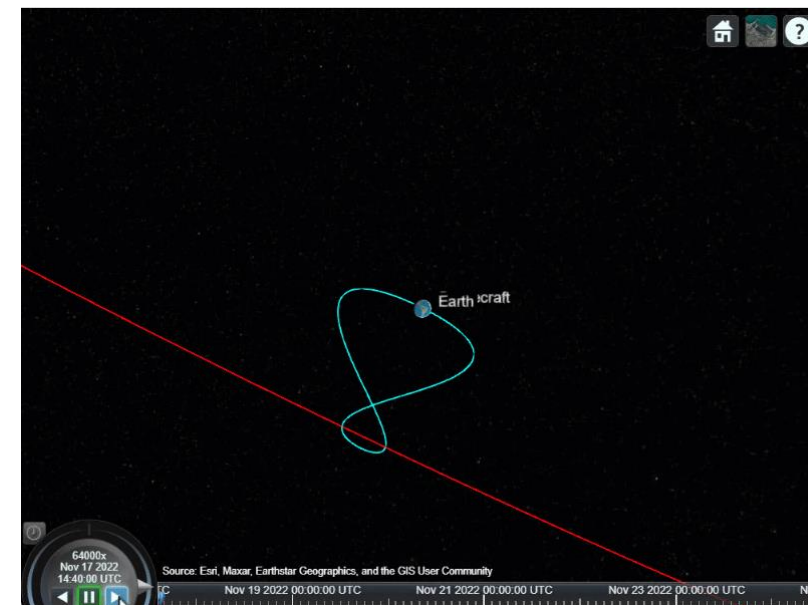
UAV zariadenia

- UAV scenáre
 - modelovanie efektov vetra
 - Urýchlenie simulácie s lidarom
- Vizualizácia logov
 - zobrazenie letu na 3D mape
- PX4 Autopiloty
 - Firmware v1.14
 - podpora Cube Orange Plus a Pixhawk 6c
- Referenčný príklad
 - VTOL UAV Controller Template
 - VTOL UAV controller template to PX4 hardware for HIL simulation



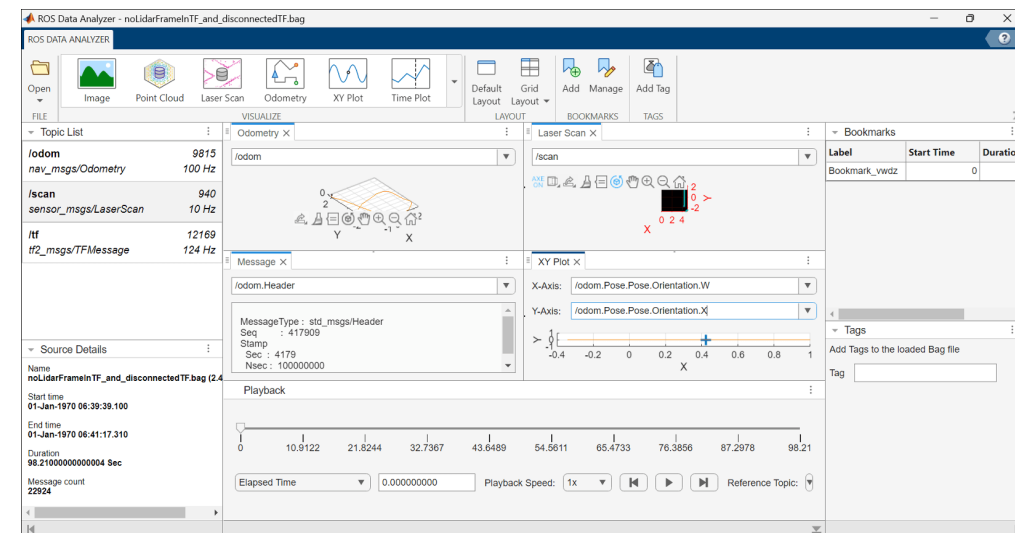
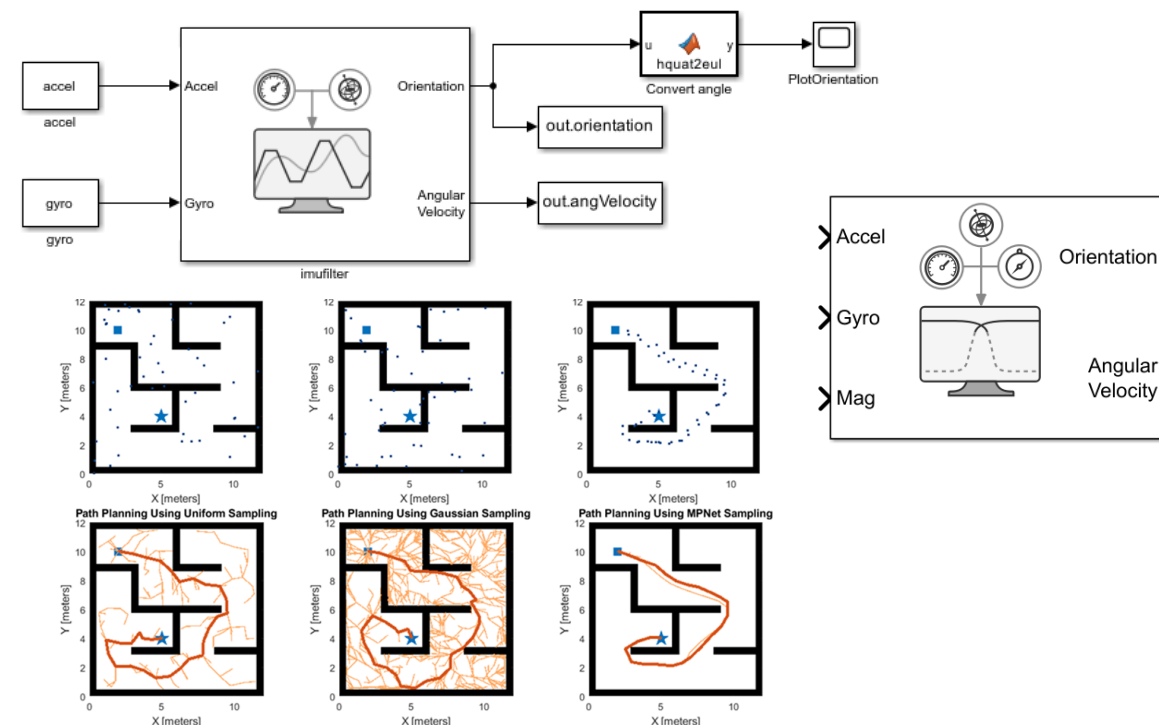
Aerospace

- Satellite Scenario
 - nový numerický propagátor – poloha a rýchlosť satelitov
 - zahrnutie ďalších platforiem
 - rendrovanie 3d modelov v scenáriu
 - urýchlenie simulácie – 4x pre 2000 satelitov
- Podpora JSBSim modelu
 - tvorba fixed-wing z JSBSim modelu
- Simulácia v 3D prostredí
 - blok Simulation 3D Spacecraft – CubeSats, SmallSats
 - Use Earth center as origin (ECEF)
 - bloky lietadiel – Simulation 3D Aircraft



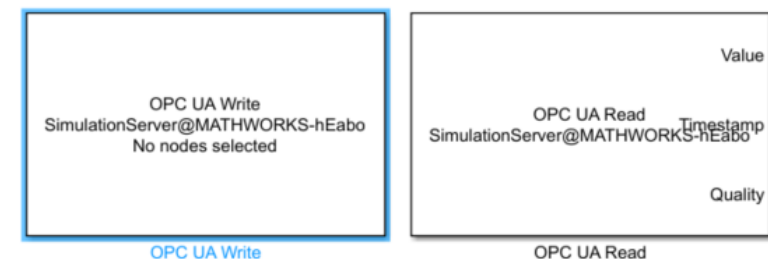
Autonómne systémy a robotika

- Odhad orientácie
 - blok IMU Filter
 - blok ecompass
- Plánovanie trajektórie s AI
 - Motion Planning Networks (MPNet)
- Generovanie kódu – objekty sledovania
 - generateCode ()
 - trackerGNN, trackerJPDA, trackerPHD, ...
- Vizualizácia ROS dát
 - ROS Data Analyzer app
- Rozšírená podpora Simulinku pre ROS2



Priemyslené komunikácie

- Bloky Simulinku pre OPC UA
 - OPC UA Read
 - OPC UA Write
- Metódy pre OPC UA
- MATLAB Online
 - podpora standardu MQTT
- Vehicle Network
 - CAN bloky podporujú 64bit integer
 - CAN Pack, CAN Unpack – podpora ARXML
 - podpora pre SocketCAN (Linux)
 - ovládače zariadení – potrebné doinštalovať



Block Parameters: OPC UA Read

OPC UA Read

Read data from an OPC UA server. The Value port outputs node values of different datatypes to a Simulink bus. The optional quality port is a vector of boolean values indicating if the quality of the value read is good. The optional Timestamp port is a double vector represented as datenum or seconds since start.

Parameters

Server: SimulationServer@MATHWORKS-hEabo

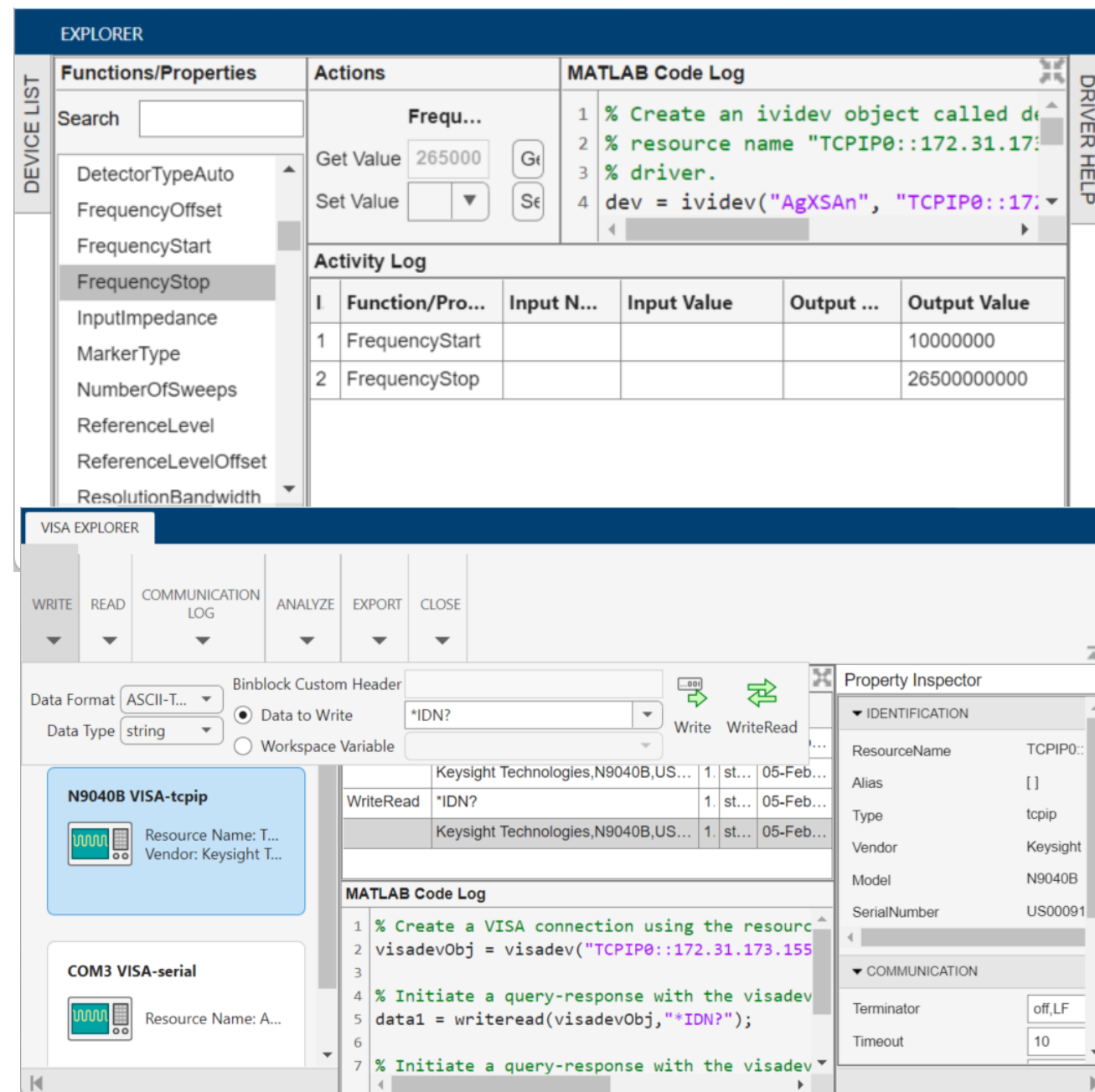
Nodes

Filter table contents

Serial No.	Name	Namespace Index
1	Sinusoid	3

Aplikácie pre zariadenia

- Instrument Explorer App
 - Pripojenie zariadení pomocou IVI a VXIplug&play ovládačov
 - prechádzanie vlastností a metód ovládačov
 - generovanie MATLAB kódu na základe interakcií
- Simulink
 - blok VISA
 - UDP bloky podporujú rapid accelerator
- Príkazy find
 - serialportfind, tcpclientfind, ...



The image shows two screenshots of software interfaces. The top screenshot is the 'EXPLORER' window, which is divided into several sections:

- DEVICES LIST:** A vertical list of device properties such as DetectorTypeAuto, FrequencyOffset, FrequencyStart, FrequencyStop (highlighted), InputImpedance, MarkerType, NumberOfSweeps, ReferenceLevel, ReferenceLevelOffset, and ResolutionBandwidth.
- Actions:** A section with input fields for 'Get Value' (265000) and 'Set Value', and buttons for 'Get' and 'Set'.
- MATLAB Code Log:** A text area showing MATLAB code:


```
1 % Create an ividev object called de
2 % resource name "TCPIP0::172.31.17:
3 % driver.
4 dev = ividev("AgXSAn", "TCPIP0::17:
```
- Activity Log:** A table with columns: I, Function/Pro..., Input N..., Input Value, Output ..., Output Value.

I	Function/Pro...	Input N...	Input Value	Output ...	Output Value
1	FrequencyStart				10000000
2	FrequencyStop				26500000000

The bottom screenshot is the 'VISA EXPLORER' window, which includes:

- Buttons:** WRITE, READ, COMMUNICATION LOG, ANALYZE, EXPORT, CLOSE.
- Data Format:** ASCII-T... (dropdown), Data Type: string (dropdown).
- Binblock Custom Header:** *IDN? (input field).
- Radio Buttons:** Data to Write (selected), Workspace Variable.
- Resource Selection:** A list of resources including 'N9040B VISA-tcpip' (selected) and 'COM3 VISA-serial'.
- MATLAB Code Log:**

```
1 % Create a VISA connection using the resourc
2 visadevObj = visadev("TCPIP0::172.31.173.155
3
4 % Initiate a query-response with the visadev
5 data1 = writeread(visadevObj,"*IDN?");
6
7 % Initiate a query-response with the visadev
```
- Property Inspector:** A panel showing device details:
 - IDENTIFICATION: ResourceName (TCPIP0::...), Alias ([]), Type (tcpip), Vendor (Keysight), Model (N9040B), SerialNumber (US00091).
 - COMMUNICATION: Terminator (off,LF), Timeout (10).

Generovanie kódu – 4000 funkcií, 39 Toolboxov



- 5G Toolbox
- Aerospace Toolbox
- Antenna Toolbox
- Audio System Toolbox
- Automated Driving Toolbox
- Bluetooth Toolbox
- Communications Toolbox
- Computer Vision Toolbox
- Control System Toolbox
- Deep Learning Toolbox
- DSP System Toolbox
- Fixed-Point Designer
- Fuzzy Logic Toolbox
- Image Acquisition Toolbox
- Image Processing Toolbox
- Industrial Communication Toolbox
- Instrumental Control Toolbox
- Lidar Toolbox
- Mapping Toolbox
- Mixed-Signal Blockset
- Model Predictive Control Toolbox
- Navigation Toolbox
- Optimization Toolbox
- Phased Array System Toolbox
- Predictive Maintenance Toolbox
- Radar Toolbox
- Reinforcement Learning Toolbox
- Robotics System Toolbox
- ROS Toolbox
- Satellite Communications Toolbox
- Sensor Fusion and Tracking Toolbox
- SerDes Toolbox
- Signal Processing Toolbox
- Stats & Machine Learning Toolbox
- System Identification Toolbox
- UAV Toolbox
- Vision HDL Toolbox
- Wavelet Toolbox
- WLAN System Toolbox

Simulácia a testovanie v reálnom čase

- Simulink Desktop Real-Time
 - platformy Windows, macOS a [Linux](#)
- Dostupné Linuxové platformy
 - Debian 11, Debian 12
 - Ubuntu 22.04, 20.04
 - Red Hat 9.2, 8.7
 - SUSE 15, 12
- Rozhranie pre SLDRT
 - funkcie pre ovládanie z príkazového riadku
- Simulink Real-Time
 - platformovo špecifická podpora - system target files (STFs)



```
openExample('sldrtex_counter');
myRunStatus = SLDRT.run('sldrtex_counter',Wait="off");
myRunStatus.Running
```

```
ans =
    logical
     1
```




Discover What's New

Get more out of MATLAB and Simulink by downloading the latest release.

[Download now](#)


Major Updates

- **Computer Vision Toolbox** – Deploy YOLOX object detection; conduct team-based labeling; perform real-time visual SLAM.
- **Deep Learning Toolbox** – Support architectures such as transformers; import and co-simulate PyTorch and TensorFlow models.
- **GPU Coder** – Generate generic CUDA for deep learning; use single memory manager and profile code for MEX code generation.
- **Instrument Control Toolbox** – Use the Instrument Explorer app to manage devices with IVI and VXIplug&play drivers without writing code.
- **Satellite Communications Toolbox** – Model multiplatform scenarios and perform visibility and communications link analyses on them.
- **UAV Toolbox** – Design and deploy flight controller for a vertical take-off and landing (VTOL) UAV with PX4 hardware-in-the-loop simulation; interface with PX4 Cube Orange Plus and Pixhawk 6c autopilots.

Help Center

Help Center ▾
🔍

☰ CONTENTS

« Documentation Home

Category

- MATLAB
- Simulink
- 5G Toolbox
- Aerospace Blockset
- Aerospace Toolbox
- Antenna Toolbox
- Audio Toolbox
- Automated Driving Toolbox
- AUTOSAR Blockset
- Bioinformatics Toolbox
- Bluetooth Toolbox
- C2000 Microcontroller Blockset
- Communications Toolbox
- Computer Vision Toolbox

Documentation Examples Functions Blocks Apps Videos Answers
Trial Software Product Updates

Release Notes R2024a

Explore Release Notes

Select a Product ▾

Resources

- [Installation and Licensing Changes](#)
- [System Requirements](#)
- [Bug Reports](#)
- [Bug Fixes](#)

New Products and Major Updates

<p>Bioinformatics Toolbox</p> <p>Read, analyze, and visualize genomic and proteomic data</p>	<p>Computer Vision Toolbox</p> <p>Design and test computer vision systems</p>	<p>Deep Learning Toolbox</p> <p>Design, train, analyze, and simulate deep learning networks</p>	<p>GPU Coder</p> <p>Generate CUDA[®] code for NVIDIA[®] GPUs</p>
<p>Instrument Control Toolbox</p> <p>Control test and measurement instruments and communicate with computer peripherals</p>	<p>Satellite Communications Toolbox</p> <p>Simulate, analyze, and test satellite communications systems and links</p>	<p>Simulink 3D Animation</p> <p>Simulate and visualize dynamic systems in a 3D environment</p>	<p>UAV Toolbox</p> <p>Design, simulate, and deploy UAV applications</p>

Online služby



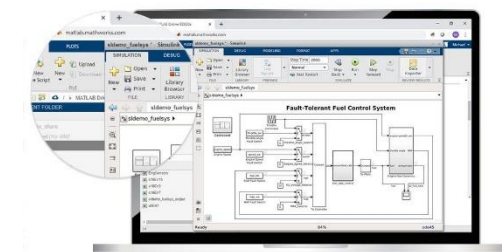
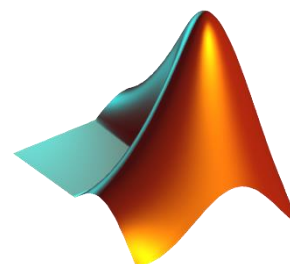
ThingSpeak



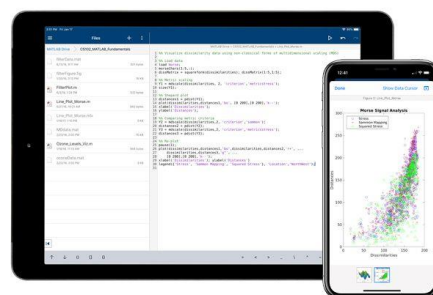
MATLAB Online



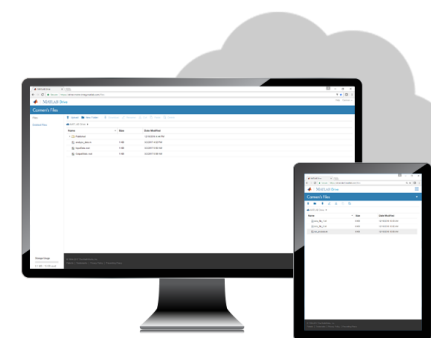
MATLAB Grader



Simulink Online



MATLAB Mobile



MATLAB Drive

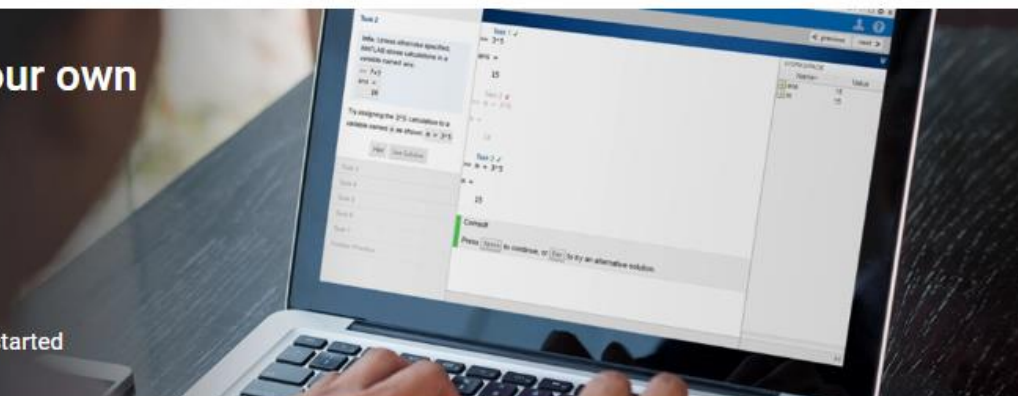
MATLAB a Simulink vlastným tempom

Self-Paced Online Courses

Home | My Courses

Learn MATLAB and Simulink at your own pace

- Complete **hands-on exercises**.
 - Take **entire courses** or focus on **specific topics**.
- ▶ Watch an introduction (1 min) » View courses you've started



Self-Paced Courses

Getting Started (18)

MATLAB (6)

Simulink (7)

AI, Machine Learning, and Deep Learning (6)

Math and Optimization (6)

Image and Signal Processing (6)

Getting Started



MATLAB Onramp

14 modules | 2 hours | Languages

Get started quickly with the basics of MATLAB.



Simulink Onramp

14 modules | 2 hours | Languages

Get started quickly with the basics of Simulink.

<https://matlabacademy.mathworks.com>