

4D_system_grid_search_vyssi_cisla

November 10, 2025

```
[1]: from pathlib import Path
from scipy.io import loadmat
import sys
import os

dataset_path = Path('data') / 'data.mat'
if not dataset_path.exists():
    alt = Path.cwd().parent / 'data' / 'data.mat'
    if alt.exists():
        dataset_path = alt
    else:
        raise FileNotFoundError(f"data.mat not found under {Path.cwd()} or its ↵parent")

notebook_path = os.getcwd()
print (f"Current notebook path: {notebook_path}")
project_root = os.path.dirname(notebook_path)
if project_root not in sys.path:
    sys.path.insert(0, project_root)
print (f"Added {project_root} to sys.path")

mat_data = loadmat(dataset_path)
print(mat_data.keys())
```

```
Current notebook path: /home/luky/skola/KalmanNet-for-state-estimation/TAN
Added /home/luky/skola/KalmanNet-for-state-estimation to sys.path
dict_keys(['__header__', '__version__', '__globals__', 'hB', 'souradniceGNSS',
'souradniceX', 'souradniceY', 'souradniceZ'])
```

```
[2]: import torch
import matplotlib.pyplot as plt
from utils import trainer
from utils import utils
from Systems import DynamicSystem
import Filters
import torch.nn.functional as F
from torch.utils.data import TensorDataset, DataLoader
```

```

import numpy as np
from scipy.io import loadmat
from scipy.interpolate import RegularGridInterpolator
import random

torch.manual_seed(42)
np.random.seed(42)
random.seed(42)
if torch.cuda.is_available():
    torch.cuda.manual_seed_all(42)

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Používané zařízení: {device}")

```

Používané zařízení: cuda

```

[3]: mat_data = loadmat(dataset_path)

souradniceX_mapa = mat_data['souradniceX']
souradniceY_mapa = mat_data['souradniceY']
souradniceZ_mapa = mat_data['souradniceZ']
souradniceGNSS = mat_data['souradniceGNSS']
x_axis_unique = souradniceX_mapa[0, :]
y_axis_unique = souradniceY_mapa[:, 0]

print(f"Rozměry 1D osy X: {x_axis_unique.shape}")
print(f"Rozměry 1D osy Y: {y_axis_unique.shape}")
print(f"Rozměry 2D dat výšek Z: {souradniceZ_mapa.shape}")

terMap_interpolator = RegularGridInterpolator(
    (y_axis_unique, x_axis_unique),
    souradniceZ_mapa,
    bounds_error=False,
    fill_value=np.nan
)
print("...interpolaci funkce vytvořena.")

def terMap(px, py):

    points_to_query = np.column_stack((py, px))

    return terMap_interpolator(points_to_query)

```

Rozměry 1D osy X: (2500,)
Rozměry 1D osy Y: (2500,)
Rozměry 2D dat výšek Z: (2500, 2500)
...interpolaci funkce vytvořena.

1 4D model

```
[4]: import torch
from math import pi
from Systems import DynamicSystemTAN

state_dim = 4
obs_dim = 3
dT = 1
q = 1

F = torch.tensor([[1.0, 0.0, dT, 0.0],
                  [0.0, 1.0, 0.0, dT],
                  [0.0, 0.0, 1.0, 0.0],
                  [0.0, 0.0, 0.0, 1.0]]))

Q = q* torch.tensor([[dT**3/3, 0.0, dT**2/2, 0.0],
                      [0.0, dT**3/3, 0.0, dT**2/2],
                      [dT**2/2, 0.0, dT, 0.0],
                      [0.0, dT**2/2, 0.0, dT]]))

R = torch.tensor([[3.0**2, 0.0, 0.0],
                  [0.0, 1.0**2, 0.0],
                  [0.0, 0.0, 1.0**2]]))

initial_velocity_np = souradniceGNSS[:, 1] - souradniceGNSS[:, 0]
initial_velocity = torch.from_numpy(initial_velocity_np)

initial_position = torch.from_numpy(souradniceGNSS[:, 0])
x_0 = torch.cat([
    initial_position,
    initial_velocity
]).float()
print(x_0)

P_0 = torch.tensor([[25.0, 0.0, 0.0, 0.0],
                   [0.0, 25.0, 0.0, 0.0],
                   [0.0, 0.0, 0.5, 0.0],
                   [0.0, 0.0, 0.0, 0.5]]))

def h_nl_robust(x: torch.Tensor) -> torch.Tensor:
    # ... (implementace s clampingem, jak jsme si ukázali dříve) ...
    # Získání hranic mapy
    min_x, max_x = x_axis.unique.min(), x_axis.unique.max()
    min_y, max_y = y_axis.unique.min(), y_axis.unique.max()

    # Oříznutí pozic POUZE pro dotaz do mapy
    px_safe = x[:, 0].clone().clamp(min_x, max_x)
```

```

py_safe = x[:, 1].clone().clamp(min_y, max_y)
vel_safe = 200.0 # ciste kvuli numericke explozi

vyska_terenu_np = terMap(px_safe.detach().cpu().numpy(), py_safe.detach().
cpu().numpy())
vyska_terenu = torch.from_numpy(vyska_terenu_np).float().to(x.device)

# Zbytek výpočtu s původními rychlostmi
eps = 1e-12

vx_w, vy_w = x[:, 2], x[:, 3]
vx_w = x[:, 2].clamp(-vel_safe, vel_safe)
vy_w = x[:, 3].clamp(-vel_safe, vel_safe)
norm_v_w = torch.sqrt(vx_w**2 + vy_w**2).clamp(min=eps)
cos_psi = vx_w / norm_v_w
sin_psi = vy_w / norm_v_w
vx_b = cos_psi * vx_w - sin_psi * vy_w
vy_b = sin_psi * vx_w + cos_psi * vy_w

result = torch.stack([vyska_terenu, vx_b, vy_b], dim=1)

# Pojistka pro případ, že by terMap přesto vrátila NaN
if torch.isnan(result).any():
    print("Varování: NaN hodnoty v měření detekovány, nahrazuji nulami.")
    result[torch.isnan(result)] = 0

return result
import torch.nn.functional as func

def h_nl_differentiable(x: torch.Tensor, map_tensor, x_min, x_max, y_min, y_max) -> torch.Tensor:
    batch_size = x.shape[0]

    px = x[:, 0]
    py = x[:, 1]

    px_norm = 2.0 * (px - x_min) / (x_max - x_min) - 1.0
    py_norm = 2.0 * (py - y_min) / (y_max - y_min) - 1.0

    sampling_grid = torch.stack((px_norm, py_norm), dim=1).view(batch_size, 1, 2)

    vyska_terenu_batch = func.grid_sample(
        map_tensor.expand(batch_size, -1, -1, -1),
        sampling_grid,
        mode='bilinear',

```

```

        padding_mode='border',
        align_corners=True
    )

vyska_terenu = vyska_terenu_batch.view(batch_size)

eps = 1e-12
vx_w, vy_w = x[:, 2], x[:, 3]
vel_safe = 100.0 # ciste kvuli numericke explozi
vx_w = x[:, 2].clamp(-vel_safe, vel_safe)
vy_w = x[:, 3].clamp(-vel_safe, vel_safe)
norm_v_w = torch.sqrt(vx_w**2 + vy_w**2).clamp(min=eps)
cos_psi = vx_w / norm_v_w
sin_psi = vy_w / norm_v_w

vx_b = cos_psi * vx_w - sin_psi * vy_w
vy_b = sin_psi * vx_w + cos_psi * vy_w

result = torch.stack([vyska_terenu, vx_b, vy_b], dim=1)

return result

x_axis_unique = souradniceX_mapa[0, :]
y_axis_unique = souradniceY_mapa[:, 0]
terMap_tensor = torch.from_numpy(souradniceZ_mapa).float().unsqueeze(0).
    unsqueeze(0).to(device)
x_min, x_max = x_axis_unique.min(), x_axis_unique.max()
y_min, y_max = y_axis_unique.min(), y_axis_unique.max()

h_wrapper = lambda x: h_nl_differentiable(
    x,
    map_tensor=terMap_tensor,
    x_min=x_min,
    x_max=x_max,
    y_min=y_min,
    y_max=y_max
)

system_model = DynamicSystemTAN(
    state_dim=state_dim,
    obs_dim=obs_dim,
    Q=Q.float(),
    R=R.float(),
    Ex0=x_0.float(),
    P0=P_0.float(),
    F=F.float(),
    h=h_wrapper,
)

```

```

        x_axis_unique=x_axis_unique,
        y_axis_unique=y_axis_unique,
        device=device
    )

tensor([ 1.4875e+06,  6.3955e+06,  4.3225e+00, -4.1456e+01])
INFO: DynamicSystemTAN inicializován s hranicemi mapy:
  X: [1476611.42, 1489541.47]
  Y: [6384032.63, 6400441.34]

```

```

[5]: import torch
from torch.utils.data import TensorDataset, DataLoader
from Systems import DynamicSystemTAN
from copy import deepcopy
import numpy as np
import random
from utils import utils
TRAIN_SEQ_LEN = 100
VALID_SEQ_LEN = 200
NUM_TRAIN_SETS = 50
TRAJ_PER_SET_TRAIN = 7
NUM_VALID_SETS = 20
TRAJ_PER_SET_VALID = 5
BATCH_SIZE = 256

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Používané zařízení: {device}")

num_gnss_points = souradniceGNSS.shape[1]
print(f"Načteno {num_gnss_points} GNSS bodů pro výběr počátečních podmínek.")
original_system_model = system_model
default_P0 = torch.diag(torch.tensor([25.0, 25.0, 0.5, 0.5], device=device)).  

    ↪float()

print("Generuji trénovací data s náhodnými počátečními podmínkami z GNSS...")
all_x_train = []
all_y_train = []
for i in range(NUM_TRAIN_SETS):
    print(f"  Generuji trénovací sadu {i+1}/{NUM_TRAIN_SETS}...")

    start_index = random.randint(0, num_gnss_points - 2)

    initial_pos_np = souradniceGNSS[:2, start_index]
    next_pos_np = souradniceGNSS[:2, start_index + 1]
    initial_vel_np = next_pos_np - initial_pos_np
    Ex0_sampled = torch.cat([

```

```

        torch.from_numpy(initial_pos_np),
        torch.from_numpy(initial_vel_np)
    ]).float().to(device)

P0_current = default_P0

temp_model = deepcopy(original_system_model)
temp_model.Ex0 = Ex0_sampled
temp_model.P0 = P0_current

x_batch, y_batch = utils.generate_data_for_map(
    temp_model,
    num_trajectories=TRAJ_PER_SET_TRAIN,
    seq_len=TRAIN_SEQ_LEN
)
all_x_train.append(x_batch)
all_y_train.append(y_batch)

x_train = torch.cat(all_x_train, dim=0)
y_train = torch.cat(all_y_train, dim=0)
print(f"Finální trénovací data: x={x_train.shape}, y={y_train.shape}")

print("Generuje validační data s náhodnými počátečními podmínkami z GNSS...")
all_x_val = []
all_y_val = []
for i in range(NUM_VALID_SETS):
    print(f"  Generuje validační sadu {i+1}/{NUM_VALID_SETS}...")
    start_index = random.randint(0, num_gnss_points - 2)
    initial_pos_np = souradniceGNSS[:2, start_index]
    next_pos_np = souradniceGNSS[:2, start_index + 1]
    initial_vel_np = next_pos_np - initial_pos_np
    Ex0_sampled = torch.cat([
        torch.from_numpy(initial_pos_np),
        torch.from_numpy(initial_vel_np)
    ]).float().to(device)
    P0_current = default_P0

    temp_model = deepcopy(original_system_model)
    temp_model.Ex0 = Ex0_sampled
    temp_model.P0 = P0_current

    x_batch, y_batch = utils.generate_data_for_map(
        temp_model,
        num_trajectories=TRAJ_PER_SET_VALID,
        seq_len=VALID_SEQ_LEN
    )
    all_x_val.append(x_batch)

```

```

    all_y_val.append(y_batch)

x_val = torch.cat(all_x_val, dim=0)
y_val = torch.cat(all_y_val, dim=0)
print(f"Finální validační data: x={x_val.shape}, y={y_val.shape}")

print("\nPočítám normalizační statistiky (průměr a std) z trénovacích dat...")

# Získáme state_dim (což je 4) z modelu
state_dim = original_system_model.state_dim

# Tvar x_train je [N_traj, Seq_Len, State_Dim]
# Tvar x_train_flat bude [N_traj * Seq_Len, State_Dim]
x_train_flat = x_train.view(-1, state_dim)

# Vypočítáme průměr a std pro každou ze 4 komponent stavu
x_mean = x_train_flat.mean(dim=0).to(device)
x_std = x_train_flat.std(dim=0).to(device)

x_std[x_std == 0] = 1.0

print(f" Vypočtený průměr (x_mean): {x_mean.cpu().numpy()}")
print(f" Vypočtená odchylka (x_std): {x_std.cpu().numpy()}")

train_dataset = TensorDataset(x_train, y_train)
val_dataset = TensorDataset(x_val, y_val)

train_loader = DataLoader(train_dataset, batch_size=BATCH_SIZE, shuffle=True)
val_loader = DataLoader(val_dataset, batch_size=BATCH_SIZE, shuffle=False)

print("\nDataLoadery jsou připraveny pro trénink.")

TEST_SEQ_LEN = 600
NUM_TEST_TRAJ = 10

print(f"\nGeneruje {NUM_TEST_TRAJ} testovacích trajektorií o délce"
      f" {TEST_SEQ_LEN}...")

x_test, y_test = utils.generate_data_for_map(
    system_model,
    num_trajectories=NUM_TEST_TRAJ,
    seq_len=TEST_SEQ_LEN
)

test_dataset = TensorDataset(x_test, y_test)
test_loader = DataLoader(test_dataset, batch_size=1, shuffle=False)

```

Používané zařízení: cuda

Načteno 1276 GNSS bodů pro výběr počátečních podmínek.

Generuje trénovací data s náhodnými počátečními podmínkami z GNSS...

Generuje trénovací sadu 1/50...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y:[6384032.63-6400441.34]

Generuje 7 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)

Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)

Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7

Úspěšnost (platné trajektorie / pokusy): 100.00%

Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])

Generuje trénovací sadu 2/50...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y:[6384032.63-6400441.34]

Generuje 7 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)

Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)

Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 8

Úspěšnost (platné trajektorie / pokusy): 87.50%

Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])

Generuje trénovací sadu 3/50...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y:[6384032.63-6400441.34]

Generuje 7 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)

Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)

Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 4/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 5/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 7)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 8)

Generování dat dokončeno.
Celkový počet pokusů: 8
Úspěšnost (platné trajektorie / pokusy): 87.50%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 6/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 7/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 8/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 9/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 10/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 11/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 12/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 13/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 14/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 7)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 8)

Generování dat dokončeno.
Celkový počet pokusů: 8
Úspěšnost (platné trajektorie / pokusy): 87.50%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 15/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 16/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 7)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 10)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 15)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 17)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 26)

Generování dat dokončeno.
Celkový počet pokusů: 26
Úspěšnost (platné trajektorie / pokusy): 26.92%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 17/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 18/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 19/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 7)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 8)

Generování dat dokončeno.
Celkový počet pokusů: 8
Úspěšnost (platné trajektorie / pokusy): 87.50%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 20/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 21/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 22/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 23/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 24/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 25/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 26/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 27/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 28/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 29/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 30/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 31/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 32/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 33/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 34/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 35/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 36/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 37/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 38/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 39/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 40/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 41/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 42/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 43/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 44/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 45/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 46/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 47/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 48/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 49/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Generuje trénovací sadu 50/50...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 7 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/7 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/7 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/7 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/7 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/7 (Pokusů: 5)
Úspěšně vygenerována trajektorie 6/7 (Pokusů: 6)
Úspěšně vygenerována trajektorie 7/7 (Pokusů: 7)

Generování dat dokončeno.
Celkový počet pokusů: 7
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([7, 100, 4])
Finální trénovací data: x=torch.Size([350, 100, 4]), y=torch.Size([350, 100, 3])
Generuje validační data s náhodnými počátečními podmínkami z GNSS...
Generuje validační sadu 1/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.

Celkový počet pokusů: 5
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 2/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 2)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 9)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 12)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 18)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 30)

Generování dat dokončeno.
Celkový počet pokusů: 30
Úspěšnost (platné trajektorie / pokusy): 16.67%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 3/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.
Celkový počet pokusů: 5
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 4/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahození)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 2)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 5)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 6)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 8)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 11)

Generování dat dokončeno.
Celkový počet pokusů: 11
Úspěšnost (platné trajektorie / pokusy): 45.45%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 5/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 7)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 8)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 11)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 12)

Generování dat dokončeno.

Celkový počet pokusů: 12

Úspěšnost (platné trajektorie / pokusy): 41.67%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 6/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 17)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 19)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 31)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 32)

Generování dat dokončeno.

Celkový počet pokusů: 32

Úspěšnost (platné trajektorie / pokusy): 15.62%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 7/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 10)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 11)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 12)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 16)

Generování dat dokončeno.

Celkový počet pokusů: 16

Úspěšnost (platné trajektorie / pokusy): 31.25%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 8/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 24)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 37)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 61)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 102)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 198)

Generování dat dokončeno.
Celkový počet pokusů: 198
Úspěšnost (platné trajektorie / pokusy): 2.53%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 9/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahodení)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.
Celkový počet pokusů: 5
Úspěšnost (platné trajektorie / pokusy): 100.00%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 10/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahodení)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 5)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 7)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 11)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 18)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 22)

Generování dat dokončeno.
Celkový počet pokusů: 22
Úspěšnost (platné trajektorie / pokusy): 22.73%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 11/20...
INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]
Generuje 5 platných trajektorií (metoda zahodení)...
Úspěšně vygenerována trajektorie 1/5 (Pokusů: 4)
Úspěšně vygenerována trajektorie 2/5 (Pokusů: 31)
Úspěšně vygenerována trajektorie 3/5 (Pokusů: 57)
Úspěšně vygenerována trajektorie 4/5 (Pokusů: 150)
Úspěšně vygenerována trajektorie 5/5 (Pokusů: 176)

Generování dat dokončeno.
Celkový počet pokusů: 176
Úspěšnost (platné trajektorie / pokusy): 2.84%
Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 12/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47] ,
Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 5)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 10)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 12)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 13)

Generování dat dokončeno.

Celkový počet pokusů: 13

Úspěšnost (platné trajektorie / pokusy): 38.46%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 13/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47] ,

Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.

Celkový počet pokusů: 5

Úspěšnost (platné trajektorie / pokusy): 100.00%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 14/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47] ,

Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.

Celkový počet pokusů: 5

Úspěšnost (platné trajektorie / pokusy): 100.00%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 15/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47] ,

Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahození)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 5)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 7)

Generování dat dokončeno.

Celkový počet pokusů: 7

Úspěšnost (platné trajektorie / pokusy): 71.43%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 16/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 5)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 6)

Generování dat dokončeno.

Celkový počet pokusů: 6

Úspěšnost (platné trajektorie / pokusy): 83.33%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 17/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 12)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 24)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 32)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 33)

Generování dat dokončeno.

Celkový počet pokusů: 33

Úspěšnost (platné trajektorie / pokusy): 15.15%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 18/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y: [6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.

Celkový počet pokusů: 5

Úspěšnost (platné trajektorie / pokusy): 100.00%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])
Generuje validační sadu 19/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],
Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 2)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 3)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 5)

Generování dat dokončeno.

Celkový počet pokusů: 5

Úspěšnost (platné trajektorie / pokusy): 100.00%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Generuje validační sadu 20/20...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y:[6384032.63-6400441.34]

Generuje 5 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/5 (Pokusů: 1)

Úspěšně vygenerována trajektorie 2/5 (Pokusů: 4)

Úspěšně vygenerována trajektorie 3/5 (Pokusů: 6)

Úspěšně vygenerována trajektorie 4/5 (Pokusů: 8)

Úspěšně vygenerována trajektorie 5/5 (Pokusů: 9)

Generování dat dokončeno.

Celkový počet pokusů: 9

Úspěšnost (platné trajektorie / pokusy): 55.56%

Celkový počet vygenerovaných trajektorií: torch.Size([5, 200, 4])

Finální validační data: x=torch.Size([100, 200, 4]), y=torch.Size([100, 200, 3])

Počítám normalizační statistiky (průměr a std) z trénovacích dat...

Vypočtený průměr (x_mean): [1.4839081e+06 6.3910025e+06 -1.8309392e+00

-1.8771821e+00]

Vypočtená odchylka (x_std): [3152.9805 3853.882 18.48875 23.184551]

DataLoadery jsou připraveny pro trénink.

Generuje 10 testovacích trajektorií o délce 600...

INFO: Generátor dat používá hranice X:[1476611.42-1489541.47],

Y:[6384032.63-6400441.34]

Generuje 10 platných trajektorií (metoda zahodení)...

Úspěšně vygenerována trajektorie 1/10 (Pokusů: 5)

Úspěšně vygenerována trajektorie 2/10 (Pokusů: 118)

Úspěšně vygenerována trajektorie 3/10 (Pokusů: 256)

Úspěšně vygenerována trajektorie 4/10 (Pokusů: 275)

Úspěšně vygenerována trajektorie 5/10 (Pokusů: 282)

Úspěšně vygenerována trajektorie 6/10 (Pokusů: 326)

Úspěšně vygenerována trajektorie 7/10 (Pokusů: 385)
Úspěšně vygenerována trajektorie 8/10 (Pokusů: 431)
Úspěšně vygenerována trajektorie 9/10 (Pokusů: 456)
Úspěšně vygenerována trajektorie 10/10 (Pokusů: 562)

Generování dat dokončeno.

Celkový počet pokusů: 562

Úspěšnost (platné trajektorie / pokusy): 1.78%

Celkový počet vygenerovaných trajektorií: torch.Size([10, 600, 4])

```
[6]: # import torch
# import torch.nn as nn
# from torch.utils.data import TensorDataset, DataLoader
# import numpy as np
# import os
# import random
# from copy import deepcopy
# from state_NN_models import StateKalmanNet_v2
# from utils import trainer

# torch.manual_seed(42)
# np.random.seed(42)
# random.seed(42)

# state_knet2 = StateKalmanNet_v2(
#     system_model=original_system_model,
#     device=device,
#     hidden_size_multiplier=12,
#     output_layer_multiplier=4,
#     num_gru_layers=2
# ).to(device)

# trained_model = trainer.train_state_KalmanNet_sliding_window(
#     model=state_knet2,
#     train_loader=train_loader,
#     val_loader=val_loader,
#     device=device,
#     epochs=200,
#     lr=1e-3,
#     clip_grad=1.0,
#     early_stopping_patience=20,
#     tbptt_k=2,
#     tbptt_w=8,
#     optimizer_=torch.optim.AdamW,
#     weight_decay_=1e-3,
# )
```

```
[11]: import pandas as pd
import numpy as np
import torch
import torch.nn.functional as F
from IPython.display import display
import time
from state_NN_models import StateKalmanNet_v2
from utils import trainer

# --- 1. Definice prostoru pro HPO (Grid Search) ---
print("Spouštím Grid Search pro HPO...")
start_time_hpo = time.time()

# Zde definujte hodnoty, které chcete testovat
hidden_multipliers = [6, 8, 10]
output_multipliers = [2, 4]
gru_hidden_dim_multipliers = [4]

results_list = []

# --- 2. Spuštění smyček Grid Search ---
total_runs = len(hidden_multipliers) * len(output_multipliers) * len(gru_hidden_dim_multipliers)
run_count = 0

for h_mult in hidden_multipliers:
    for o_mult in output_multipliers:
        for g_mult in gru_hidden_dim_multipliers:
            run_count += 1
            run_id = f"h{h_mult}_o{o_mult}_g{g_mult}"
            print(f"\n{'='*80}")
            print(f"Běh HPO {run_count}/{total_runs}: {run_id} (Hidden:{h_mult}, Output: {o_mult}, GRU: {g_mult})")
            print(f"{'='*80}")

            start_time_run = time.time()

            # Inicializace metrik pro případ selhání
            best_train_loss = float('inf')
            best_val_loss = float('inf')
            final_test_mse = float('inf')

            # --- 3. BLOK TRY...EXCEPT PRO ODCHYCENÍ NESTABILITY ---
            try:
                # --- 3a. Inicializace modelu ---
                current_model = StateKalmanNet_v2(
                    system_model=original_system_model,
```

```

        device=device,
        hidden_size_multiplier=h_mult,
        output_layer_multiplier=o_mult,
        gru_hidden_dim_multiplier=g_mult,
        num_gru_layers=1
    ).to(device)
    print(f"Model inicializován: {current_model}")

    # --- 3b. Trénování modelu ---
    training_results = trainer.
    ↪train_state_KalmanNet_sliding_window_grid_search(
        model=current_model,
        train_loader=train_loader,
        val_loader=val_loader,
        device=device,
        epochs=200,
        lr=1e-3,
        clip_grad=1.0,
        early_stopping_patience=30,
        tbptt_k=2,
        tbptt_w=8,
        optimizer_=torch.optim.AdamW,
        weight_decay_=5e-3,
        verbose=False
    )

    best_train_loss = training_results['best_train_loss']
    best_val_loss = training_results['best_val_loss']
    current_model = training_results['model']

    print(f"Trénování dokončeno. Nejlepší Train Loss: {best_train_loss:.6f}, Val Loss: {best_val_loss:.6f}")

    # --- 3c. Evaluace na testovací sadě ---
    print("Evaluace na testovacích datech...")
    current_model.eval()

    test_mse_list = []
    with torch.no_grad():
        for x_true_seq_batch, y_test_seq_batch in test_loader:
            y_test_seq_gpu = y_test_seq_batch.squeeze(0).to(device)
            x_true_seq_gpu = x_true_seq_batch.squeeze(0).to(device)
            initial_state = x_true_seq_gpu[0, :].unsqueeze(0)
            TEST_SEQ_LEN = x_true_seq_gpu.shape[0]

            current_model.reset(batch_size=1,
        ↪initial_state=initial_state)

```

```

        model_preds = []
        for t in range(1, TEST_SEQ_LEN):
            step_output = current_model.step(y_test_seq_gpu[t, :
→] .unsqueeze(0))

                if current_model.returns_covariance:
                    x_filtered_t = step_output[0]
                else:
                    x_filtered_t = step_output
                model_preds.append(x_filtered_t)

        full_x_hat_model = torch.cat([initial_state, torch.
→cat(model_preds, dim=0)], dim=0)
        mse = F.mse_loss(full_x_hat_model[1:], x_true_seq_gpu[1:
→]).item()

        # Zkontrolujeme, zda i MSE není náhodou NaN/Inf
        if not np.isfinite(mse):
            print("Varování: MSE na testovací sadě je NaN/Inf!")
            raise RuntimeError("Selhání při evaluaci (NaN MSE)")

        test_mse_list.append(mse)

        final_test_mse = np.mean(test_mse_list)
        print(f"Evaluace dokončena. Průměrné Test MSE: {final_test_mse:.6f}.")

# --- 3d. Zpracování výjimky (když se objeví NaN) ---
except RuntimeError as e:
    if "NaN" in str(e) or "Inf" in str(e):
        print(f"\n!!!!!!!!!!!!!! POZOR !!!!!!!!!!!!!!!")
        print(f"Běh {run_id} selhal kvůli numerické nestabilitě"
→(NaN/Inf).")
        print(f"Chyba: {e}")
        print(f"Tento běh bude zaznamenán s MSE = 'inf' a HPO bude"
→pokračovat.")
        print(f"!!!!!!!!!!!!!!\n")
        final_test_mse = float('inf') # Penalizace
    else:
        # Pokud je to jiná chyba, chceme, aby HPO spadlo
        raise e

run_duration = time.time() - start_time_run
print(f"Doba trvání běhu: {run_duration:.2f}s")

# --- 6. Uložení výsledků ---

```

```

        results_list.append({
            "run_id": run_id,
            "h_mult": h_mult,
            "o_mult": o_mult,
            "g_mult": g_mult,
            "best_train_loss": best_train_loss,
            "best_val_loss": best_val_loss,
            "test_mse": final_test_mse,
            "duration_s": run_duration
        })

# --- 7. Zobrazení finální tabulky ---
print("\n" + "="*80)
print(f"Grid Search HPO Dokončen! Celkový čas: {(time.time() - start_time_hpo) / 
    ↴ 60:.2f} minut.")
print("="*80)

# Vytvoření a seřazení DataFrame
results_df = pd.DataFrame(results_list)
results_df = results_df.sort_values(by="test_mse", ascending=True)

pd.set_option('display.float_format', '{:.6f}'.format)
display(results_df)

print("\nNejlepší konfigurace (podle Test MSE):")
print(results_df.iloc[0])

```

Spouštím Grid Search pro HPO...

```

=====
Běh HPO 1/6: h6_o2_g4 (Hidden: 6, Output: 2, GRU: 4)
=====
Model inicializován: StateKalmanNet_v2(
    (dnn): DNN_KalmanNet_v2(
        (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
        (input_layer): Sequential(
            (0): Linear(in_features=14, out_features=336, bias=True)
            (1): ReLU()
        )
        (gru): GRU(336, 100)
        (output_hidden_layer): Sequential(
            (0): Linear(in_features=100, out_features=24, bias=True)
            (1): ReLU()
        )
        (output_final_linear): Linear(in_features=24, out_features=12, bias=True)
    )
)

```

```

/home/luky/.local/lib/python3.10/site-packages/torch/optim/lr_scheduler.py:28:
UserWarning: The verbose parameter is deprecated. Please use get_last_lr() to
access the learning rate.
    warnings.warn("The verbose parameter is deprecated. Please use get_last_lr() "
Trénování dokončeno. Nejlepší Train Loss: 1275.860978, Val Loss: 524.533081
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 283222.715569.
Doba trvání běhu: 64.35s

=====
Běh HPO 2/6: h6_o4_g4 (Hidden: 6, Output: 4, GRU: 4)
=====
Model inicializován: StateKalmanNet_v2(
(dnn): DNN_KalmanNet_v2(
    (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
    (input_layer): Sequential(
        (0): Linear(in_features=14, out_features=336, bias=True)
        (1): ReLU()
    )
    (gru): GRU(336, 100)
    (output_hidden_layer): Sequential(
        (0): Linear(in_features=100, out_features=48, bias=True)
        (1): ReLU()
    )
    (output_final_linear): Linear(in_features=48, out_features=12, bias=True)
)
)
Trénování dokončeno. Nejlepší Train Loss: 2611.664257, Val Loss: 130.723541
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 10050.300001.
Doba trvání běhu: 61.44s

=====
Běh HPO 3/6: h8_o2_g4 (Hidden: 8, Output: 2, GRU: 4)
=====
Model inicializován: StateKalmanNet_v2(
(dnn): DNN_KalmanNet_v2(
    (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
    (input_layer): Sequential(
        (0): Linear(in_features=14, out_features=448, bias=True)
        (1): ReLU()
    )
    (gru): GRU(448, 100)
    (output_hidden_layer): Sequential(
        (0): Linear(in_features=100, out_features=24, bias=True)
        (1): ReLU()
    )
)

```

```
        (output_final_linear): Linear(in_features=24, out_features=12, bias=True)
    )
)
```

Trénování dokončeno. Nejlepší Train Loss: 292.067210, Val Loss: 138.771500

Evaluace na testovacích datech...

Evaluace dokončena. Průměrné Test MSE: 202432.297469.

Doba trvání běhu: 73.32s

```
=====
Běh HPO 4/6: h8_o4_g4 (Hidden: 8, Output: 4, GRU: 4)
=====
```

```
Model inicializován: StateKalmanNet_v2(
    (dnn): DNN_KalmanNet_v2(
        (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
        (input_layer): Sequential(
            (0): Linear(in_features=14, out_features=448, bias=True)
            (1): ReLU()
        )
        (gru): GRU(448, 100)
        (output_hidden_layer): Sequential(
            (0): Linear(in_features=100, out_features=48, bias=True)
            (1): ReLU()
        )
        (output_final_linear): Linear(in_features=48, out_features=12, bias=True)
    )
)
```

Trénování dokončeno. Nejlepší Train Loss: 112.979935, Val Loss: 128.908005

Evaluace na testovacích datech...

Evaluace dokončena. Průměrné Test MSE: 350777.235132.

Doba trvání běhu: 66.52s

```
=====
Běh HPO 5/6: h10_o2_g4 (Hidden: 10, Output: 2, GRU: 4)
=====
```

```
Model inicializován: StateKalmanNet_v2(
    (dnn): DNN_KalmanNet_v2(
        (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
        (input_layer): Sequential(
            (0): Linear(in_features=14, out_features=560, bias=True)
            (1): ReLU()
        )
        (gru): GRU(560, 100)
        (output_hidden_layer): Sequential(
            (0): Linear(in_features=100, out_features=24, bias=True)
            (1): ReLU()
        )
        (output_final_linear): Linear(in_features=24, out_features=12, bias=True)
    )
)
```

```

)
Trénování dokončeno. Nejlepší Train Loss: 228.058997, Val Loss: 189.797379
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 230869.253619.
Doba trvání běhu: 58.38s
```

```
=====
Běh HPO 6/6: h10_o4_g4 (Hidden: 10, Output: 4, GRU: 4)
=====
Model inicializován: StateKalmanNet_v2(
    (dnn): DNN_KalmanNet_v2(
        (input_norm): LayerNorm((14,), eps=1e-05, elementwise_affine=True)
        (input_layer): Sequential(
            (0): Linear(in_features=14, out_features=560, bias=True)
            (1): ReLU()
        )
        (gru): GRU(560, 100)
        (output_hidden_layer): Sequential(
            (0): Linear(in_features=100, out_features=48, bias=True)
            (1): ReLU()
        )
        (output_final_linear): Linear(in_features=48, out_features=12, bias=True)
    )
)
Trénování dokončeno. Nejlepší Train Loss: 1669.888760, Val Loss: 108.175728
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 8701.109563.
Doba trvání běhu: 61.83s
```

```
=====
Grid Search HPO Dokončen! Celkový čas: 6.43 minut.
=====
```

run_id	h_mult	o_mult	g_mult	best_train_loss	best_val_loss	\
5	h10_o4_g4	10	4	4	1669.888760	108.175728
1	h6_o4_g4	6	4	4	2611.664257	130.723541
2	h8_o2_g4	8	2	4	292.067210	138.771500
4	h10_o2_g4	10	2	4	228.058997	189.797379
0	h6_o2_g4	6	2	4	1275.860978	524.533081
3	h8_o4_g4	8	4	4	112.979935	128.908005

	test_mse	duration_s
5	8701.109563	61.832121
1	10050.300001	61.438509
2	202432.297469	73.317364
4	230869.253619	58.379710
0	283222.715569	64.352914
3	350777.235132	66.517033

```
Nejlepší konfigurace (podle Test MSE):
run_id          h10_o4_g4
h_mult          10
o_mult          4
g_mult          4
best_train_loss 1669.888760
best_val_loss   108.175728
test_mse        8701.109563
duration_s      61.832121
Name: 5, dtype: object
```

```
[10]: import pandas as pd
import numpy as np
import torch
import torch.nn.functional as F
from IPython.display import display
import time
from state_NN_models import StateKalmanNet_arch2
from utils import trainer

# --- 1. Definice prostoru pro HPO (Grid Search) ---
print("Spouštím Grid Search pro HPO...")
start_time_hpo = time.time()

# Zde definujte hodnoty, které chcete testovat
hidden_multipliers = [10, 12]
output_multipliers = [4, 6, 8]
results_list = []

# --- 2. Spuštění smyček Grid Search ---
total_runs = len(hidden_multipliers) * len(output_multipliers)
run_count = 0

for h_mult in hidden_multipliers:
    for o_mult in output_multipliers:
        run_count += 1
        run_id = f"h{h_mult}_o{o_mult}"
        print(f"\n{'='*80}")
        print(f"Běh HPO {run_count}/{total_runs}: {run_id} (Hidden:{h_mult}, Output: {o_mult})")
        print(f"{'='*80}")

        start_time_run = time.time()

        # Inicializace metrik pro případ selhání
        best_train_loss = float('inf')
```

```

best_val_loss = float('inf')
final_test_mse = float('inf')

# --- 3. BLOK TRY...EXCEPT PRO ODCHYCENÍ NESTABILITY ---
try:
    # --- 3a. Inicializace modelu ---
    current_model = StateKalmanNet_arch2(
        system_model=original_system_model,
        device=device,
        hidden_size_multiplier=h_mult,
        output_layer_multiplier=o_mult
    ).to(device)
    print(f"Model inicializován: {current_model}")

    # --- 3b. Trénování modelu ---
    training_results = trainer.
    ↪train_state_KalmanNet_sliding_window_grid_search(
        model=current_model,
        train_loader=train_loader,
        val_loader=val_loader,
        device=device,
        epochs=200,
        lr=1e-3,
        clip_grad=1.0,
        early_stopping_patience=30,
        tbptt_k=2,
        tbptt_w=8,
        optimizer_=torch.optim.AdamW,
        weight_decay_=5e-3,
        verbose=False
    )

    best_train_loss = training_results['best_train_loss']
    best_val_loss = training_results['best_val_loss']
    current_model = training_results['model']

    print(f"Trénování dokončeno. Nejlepší Train Loss: {best_train_loss:.6f}, Val Loss: {best_val_loss:.6f}")

    # --- 3c. Evaluace na testovací sadě ---
    print("Evaluace na testovacích datech...")
    current_model.eval()

    test_mse_list = []
    with torch.no_grad():
        for x_true_seq_batch, y_test_seq_batch in test_loader:
            y_test_seq_gpu = y_test_seq_batch.squeeze(0).to(device)

```

```

        x_true_seq_gpu = x_true_seq_batch.squeeze(0).to(device)
        initial_state = x_true_seq_gpu[0, :].unsqueeze(0)
        TEST_SEQ_LEN = x_true_seq_gpu.shape[0]

        current_model.reset(batch_size=1, □
        ↵initial_state=initial_state)
        model_preds = []
        for t in range(1, TEST_SEQ_LEN):
            step_output = current_model.step(y_test_seq_gpu[t, :]
        ↵].unsqueeze(0))

            if current_model.returns_covariance:
                x_filtered_t = step_output[0]
            else:
                x_filtered_t = step_output
            model_preds.append(x_filtered_t)

        full_x_hat_model = torch.cat([initial_state, torch.
        ↵cat(model_preds, dim=0)], dim=0)
        mse = F.mse_loss(full_x_hat_model[1:], x_true_seq_gpu[1:
        ↵]).item()

        # Zkontrolujeme, zda i MSE není náhodou NaN/Inf
        if not np.isfinite(mse):
            print("Varování: MSE na testovací sadě je NaN/Inf!")
            raise RuntimeError("Selhání při evaluaci (NaN MSE)")

        test_mse_list.append(mse)

        final_test_mse = np.mean(test_mse_list)
        print(f"Evaluace dokončena. Průměrné Test MSE: {final_test_mse:.6f}.")

# --- 3d. Zpracování výjimky (když se objeví NaN) ---
except RuntimeError as e:
    if "NaN" in str(e) or "Inf" in str(e):
        print(f"\n!!!!!!!!!!!!!! POZOR !!!!!!!!!!!!!!!")
        print(f"Běh {run_id} selhal kvůli numerické nestabilitě"
        ↵(NaN/Inf).")
        print(f"Chyba: {e}")
        print(f"Tento běh bude zaznamenán s MSE = 'inf' a HPO bude"
        ↵pokračovat.")
        print(f"!!!!!!!!!!!!!!\n")
        final_test_mse = float('inf') # Penalizace
    else:
        # Pokud je to jiná chyba, chceme, aby HPO spadlo

```

```

        raise e

    run_duration = time.time() - start_time_run
    print(f"Doba trvání běhu: {run_duration:.2f}s")

    # --- 6. Uložení výsledků ---
    results_list.append({
        "run_id": run_id,
        "h_mult": h_mult,
        "o_mult": o_mult,
        "best_train_loss": best_train_loss,
        "best_val_loss": best_val_loss,
        "test_mse": final_test_mse,
        "duration_s": run_duration
    })

# --- 7. Zobrazení finální tabulky ---
print("\n" + "="*80)
print(f"Grid Search HPO Dokončen! Celkový čas: {(time.time() - start_time_hpo) /"
    " 60:.2f} minut.")
print("=*80)

# Vytvoření a seřazení DataFrame
results_df = pd.DataFrame(results_list)
results_df = results_df.sort_values(by="test_mse", ascending=True)

pd.set_option('display.float_format', '{:.6f}'.format)
display(results_df)

print("\nNejlepší konfigurace (podle Test MSE):")
print(results_df.iloc[0])

```

Spouštím Grid Search pro HPO...

```
=====
Běh HPO 1/6: h10_o4 (Hidden: 10, Output: 4)
=====

Model inicializován: StateKalmanNet_arch2(
(dnn): DNN_KalmanNet_arch2(
    (GRU_Q): GRU(40, 16)
    (GRU_Sigma): GRU(56, 16)
    (GRU_S): GRU(69, 9)
    (FC1): Sequential(
        (0): Linear(in_features=16, out_features=9, bias=True)
        (1): ReLU()
    )
    (FC2): Sequential(

```

```

        (0): Linear(in_features=25, out_features=100, bias=True)
        (1): ReLU()
        (2): Linear(in_features=100, out_features=12, bias=True)
    )
    (FC3): Sequential(
        (0): Linear(in_features=21, out_features=16, bias=True)
        (1): ReLU()
    )
    (FC4): Sequential(
        (0): Linear(in_features=32, out_features=16, bias=True)
        (1): ReLU()
    )
    (FC5): Sequential(
        (0): Linear(in_features=4, out_features=40, bias=True)
        (1): ReLU()
    )
    (FC6): Sequential(
        (0): Linear(in_features=4, out_features=40, bias=True)
        (1): ReLU()
    )
    (FC7): Sequential(
        (0): Linear(in_features=6, out_features=60, bias=True)
        (1): ReLU()
    )
)
)
)

/home/luky/.local/lib/python3.10/site-packages/torch/optim/lr_scheduler.py:28:
UserWarning: The verbose parameter is deprecated. Please use get_last_lr() to
access the learning rate.
    warnings.warn("The verbose parameter is deprecated. Please use get_last_lr() "
Trénování dokončeno. Nejlepší Train Loss: 328.409976, Val Loss: 145.246674
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 669005.648188.
Doba trvání běhu: 71.16s

=====
Běh HPO 2/6: h10_o6 (Hidden: 10, Output: 6)
=====
Model inicializován: StateKalmanNet_arch2(
    (dnn): DNN_KalmanNet_arch2(
        (GRU_Q): GRU(40, 16)
        (GRU_Sigma): GRU(56, 16)
        (GRU_S): GRU(69, 9)
        (FC1): Sequential(
            (0): Linear(in_features=16, out_features=9, bias=True)
            (1): ReLU()
        )
)

```

```

(FC2): Sequential(
    (0): Linear(in_features=25, out_features=150, bias=True)
    (1): ReLU()
    (2): Linear(in_features=150, out_features=12, bias=True)
)
(FC3): Sequential(
    (0): Linear(in_features=21, out_features=16, bias=True)
    (1): ReLU()
)
(FC4): Sequential(
    (0): Linear(in_features=32, out_features=16, bias=True)
    (1): ReLU()
)
(FC5): Sequential(
    (0): Linear(in_features=4, out_features=40, bias=True)
    (1): ReLU()
)
(FC6): Sequential(
    (0): Linear(in_features=4, out_features=40, bias=True)
    (1): ReLU()
)
(FC7): Sequential(
    (0): Linear(in_features=6, out_features=60, bias=True)
    (1): ReLU()
)
)
)
)

Trénování dokončeno. Nejlepší Train Loss: 506.032642, Val Loss: 153.616348
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 508869.892539.
Doba trvání běhu: 60.45s

```

```

=====
Běh HPO 3/6: h10_o8 (Hidden: 10, Output: 8)
=====
Model inicializován: StateKalmanNet_arch2(
    (dnn): DNN_KalmanNet_arch2(
        (GRU_Q): GRU(40, 16)
        (GRU_Sigma): GRU(56, 16)
        (GRU_S): GRU(69, 9)
        (FC1): Sequential(
            (0): Linear(in_features=16, out_features=9, bias=True)
            (1): ReLU()
        )
        (FC2): Sequential(
            (0): Linear(in_features=25, out_features=200, bias=True)
            (1): ReLU()
            (2): Linear(in_features=200, out_features=12, bias=True)

```

```

)
(FC3): Sequential(
    (0): Linear(in_features=21, out_features=16, bias=True)
    (1): ReLU()
)
(FC4): Sequential(
    (0): Linear(in_features=32, out_features=16, bias=True)
    (1): ReLU()
)
(FC5): Sequential(
    (0): Linear(in_features=4, out_features=40, bias=True)
    (1): ReLU()
)
(FC6): Sequential(
    (0): Linear(in_features=4, out_features=40, bias=True)
    (1): ReLU()
)
(FC7): Sequential(
    (0): Linear(in_features=6, out_features=60, bias=True)
    (1): ReLU()
)
)
)
)
)

Trénování dokončeno. Nejlepší Train Loss: 243.865975, Val Loss: 105.969986
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 39146.537425.
Doba trvání běhu: 67.87s

```

```
=====
Běh HPO 4/6: h12_o4 (Hidden: 12, Output: 4)
=====

Model inicializován: StateKalmanNet_arch2(
(dnn): DNN_KalmanNet_arch2(
    (GRU_Q): GRU(48, 16)
    (GRU_Sigma): GRU(64, 16)
    (GRU_S): GRU(81, 9)
    (FC1): Sequential(
        (0): Linear(in_features=16, out_features=9, bias=True)
        (1): ReLU()
    )
    (FC2): Sequential(
        (0): Linear(in_features=25, out_features=100, bias=True)
        (1): ReLU()
        (2): Linear(in_features=100, out_features=12, bias=True)
    )
    (FC3): Sequential(
        (0): Linear(in_features=21, out_features=16, bias=True)
        (1): ReLU()
    )
)
```

```

)
(FC4): Sequential(
    (0): Linear(in_features=32, out_features=16, bias=True)
    (1): ReLU()
)
(FC5): Sequential(
    (0): Linear(in_features=4, out_features=48, bias=True)
    (1): ReLU()
)
(FC6): Sequential(
    (0): Linear(in_features=4, out_features=48, bias=True)
    (1): ReLU()
)
(FC7): Sequential(
    (0): Linear(in_features=6, out_features=72, bias=True)
    (1): ReLU()
)
)
)
)
Trénování dokončeno. Nejlepší Train Loss: 723.732797, Val Loss: 144.756027
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 386824.631049.
Doba trvání běhu: 50.48s
=====
```

```

Běh HPO 5/6: h12_o6 (Hidden: 12, Output: 6)
=====
Model inicializován: StateKalmanNet_arch2(
    (dnn): DNN_KalmanNet_arch2(
        (GRU_Q): GRU(48, 16)
        (GRU_Sigma): GRU(64, 16)
        (GRU_S): GRU(81, 9)
        (FC1): Sequential(
            (0): Linear(in_features=16, out_features=9, bias=True)
            (1): ReLU()
        )
        (FC2): Sequential(
            (0): Linear(in_features=25, out_features=150, bias=True)
            (1): ReLU()
            (2): Linear(in_features=150, out_features=12, bias=True)
        )
        (FC3): Sequential(
            (0): Linear(in_features=21, out_features=16, bias=True)
            (1): ReLU()
        )
        (FC4): Sequential(
            (0): Linear(in_features=32, out_features=16, bias=True)
            (1): ReLU()
        )
    )
)
```

```

)
(FC5): Sequential(
    (0): Linear(in_features=4, out_features=48, bias=True)
    (1): ReLU()
)
(FC6): Sequential(
    (0): Linear(in_features=4, out_features=48, bias=True)
    (1): ReLU()
)
(FC7): Sequential(
    (0): Linear(in_features=6, out_features=72, bias=True)
    (1): ReLU()
)
)
)
Trénování dokončeno. Nejlepší Train Loss: 993.272848, Val Loss: 82.494644
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 80242.809149.
Doba trvání běhu: 74.02s

=====
Běh HPO 6/6: h12_o8 (Hidden: 12, Output: 8)
=====
Model inicializován: StateKalmanNet_arch2(
    (dnn): DNN_KalmanNet_arch2(
        (GRU_Q): GRU(48, 16)
        (GRU_Sigma): GRU(64, 16)
        (GRU_S): GRU(81, 9)
        (FC1): Sequential(
            (0): Linear(in_features=16, out_features=9, bias=True)
            (1): ReLU()
        )
        (FC2): Sequential(
            (0): Linear(in_features=25, out_features=200, bias=True)
            (1): ReLU()
            (2): Linear(in_features=200, out_features=12, bias=True)
        )
        (FC3): Sequential(
            (0): Linear(in_features=21, out_features=16, bias=True)
            (1): ReLU()
        )
        (FC4): Sequential(
            (0): Linear(in_features=32, out_features=16, bias=True)
            (1): ReLU()
        )
        (FC5): Sequential(
            (0): Linear(in_features=4, out_features=48, bias=True)
            (1): ReLU()
        )
    )
)

```

```

)
(FC6): Sequential(
    (0): Linear(in_features=4, out_features=48, bias=True)
    (1): ReLU()
)
(FC7): Sequential(
    (0): Linear(in_features=6, out_features=72, bias=True)
    (1): ReLU()
)
)
Trénování dokončeno. Nejlepší Train Loss: 756.001312, Val Loss: 291.159637
Evaluace na testovacích datech...
Evaluace dokončena. Průměrné Test MSE: 3177283.978142.
Doba trvání běhu: 57.19s
=====
```

Grid Search HPO Dokončen! Celkový čas: 6.35 minut.

=====

	run_id	h_mult	o_mult	best_train_loss	best_val_loss	test_mse	\
2	h10_o8	10	8	243.865975	105.969986	39146.537425	
4	h12_o6	12	6	993.272848	82.494644	80242.809149	
3	h12_o4	12	4	723.732797	144.756027	386824.631049	
1	h10_o6	10	6	506.032642	153.616348	508869.892539	
0	h10_o4	10	4	328.409976	145.246674	669005.648188	
5	h12_o8	12	8	756.001312	291.159637	3177283.978142	
				duration_s			
2				67.865082			
4				74.015179			
3				50.481719			
1				60.449029			
0				71.160267			
5				57.188642			

Nejlepší konfigurace (podle Test MSE):

run_id	h10_o8
h_mult	10
o_mult	8
best_train_loss	243.865975
best_val_loss	105.969986
test_mse	39146.537425
duration_s	67.865082

Name: 2, dtype: object