**RTSNET:** Deep Learning Aided Kalman Smoothing

Xiaoyong Ni, Guy Revach, Nir Shlezinger, Ruud J. G. van Sloun, and Yonina C. Eldar.

1 ETH Zürich, Switzerland, 2 Ben-Gurion University of the Negev, Israel, 3 Eindhoven University of Technology, The Netherlands, 4 Weizmann Institute of Science, Israel.

### MOTIVATION

Tracking of dynamic systems is encountered in many applications: Localization, Navigation, Task Planning, etc. Such settings can often be represented as smoothing tasks, which are typically tackled using either a Model-Based (MB) or a Data-Driven (DD) method.

In this work we aim to design a hybrid MB DD smoother.

Key idea: replace part of the MB computation by NN, in order to incorporate the advantages of both domains.

### PROBLEM FORMULATION

Consider fixed-interval smoothing: the recovery of a state block $\{x_t\}_{t=1}^T$ given a block of noisy observations $\{y_t\}_{t=1}^T$ for a fixed length $T$. The state and the observations are related via a dynamical system represented by

\[
x_t = f(x_{t-1}) + e_t, \quad e_t \sim \mathcal{N}(0, Q), \quad x_t \in \mathbb{R}^m, \quad (1a)
\]

\[
y_t = h(x_t) + v_t, \quad v_t \sim \mathcal{N}(0, R), \quad y_t \in \mathbb{R}^n. \quad (1b)
\]

In (1), $f(\cdot)$ and $h(\cdot)$ are (possibly) non-linear functions, while $e_t$ and $v_t$ are Gaussian noise signals with covariance matrices $Q$ and $R$, respectively.

### TRADITIONAL APPROACH

**Solution:**

- **Linear case:** Rauch-Tung-Striebel (RTS) Smoother achieves the optimal MMSE for linear State Space model
- **Non-linear case:** linear approximations of $f(\cdot)$ and $h(\cdot)$ through Jacobian matrices, or heuristic methods like particle smoothing

**Drawbacks:**

- Require full knowledge of the underlying model and is notably degraded in the presence of model mismatch
- Limited accuracy in highly non-linear setups

### RTSNET - OUR APPROACH

The basic design idea of RTSNet is to utilize the structure of the model-based RTS smoother and to replace modules depending on unavailable domain knowledge with trainable Recurrent Neural Networks (RNNs).

- NN-aided Kalman Gains compensate for model mismatch
- Avoid linearization and is less sensitive to non-linearities
- Not require inverting matrices while inferring rapidly with low computation complexity due to efficient RNNs
- Can be extended to carry out multiple passes via deep unfolding

### EXPERIMENTS

**Linear case:**

**Highly non-linear Lorenz Attractor case:**

![Lorenz Attractor Diagram]