Dalle Molle Institute for Artificial Intelligence
USI – SUPSI

# Meta Learning for System Identification

Marco Forgione

In collaboration with Dario Piga, Matteo Rufolo, Riccardo Busetto, Gabriele Maroni, Filippo Pura,…

SUPSI

# Motivation

- Machine learning models can effectively describe a wide variety of dynamical systems
- However, training is computationally expensive and data-hungry
- Can we automate the design of algorithms, model structures, optimizers for a class of system identification problems?
- This is meta learning, aka learning to learn!

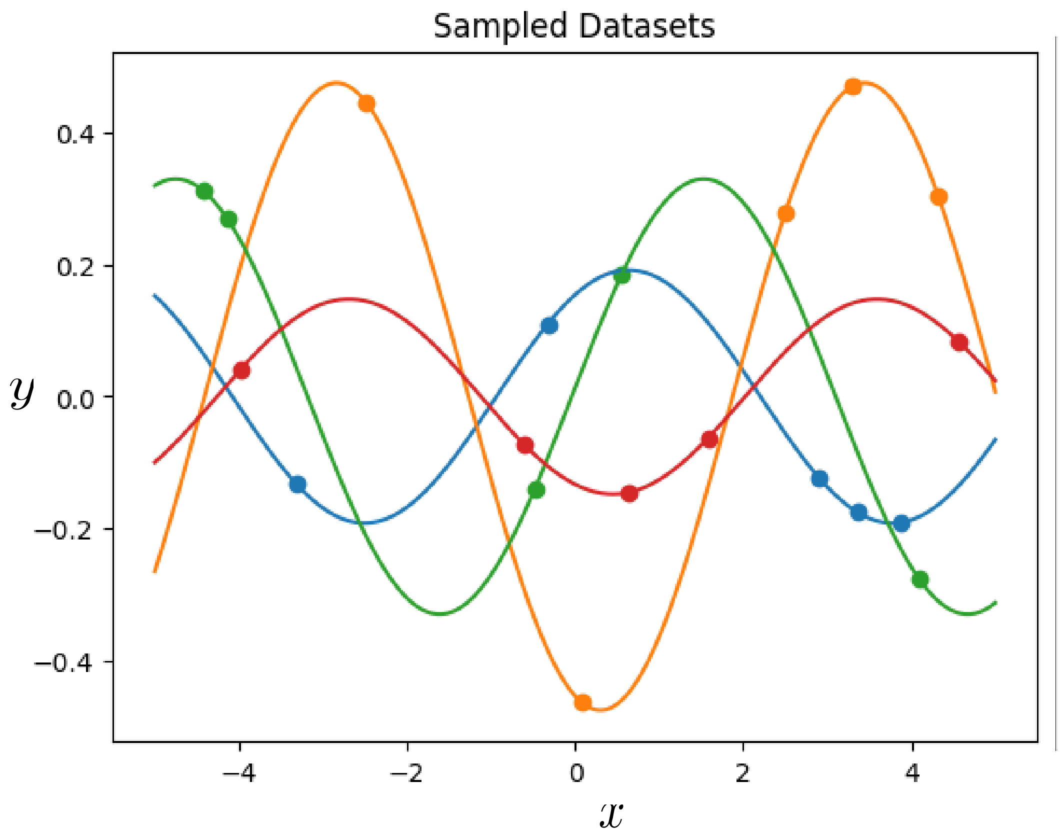
# The “sines” meta learning toy example

- Regression datasets $x \to y$: sines with randomized phase and amplitude

$$y(x) = A\sin(x + \psi)$$

- Simple structure, but just $K = 5$ points per dataset

- Would be trivial with $K = 50$ points per dataset, or if we the knew the true model structure



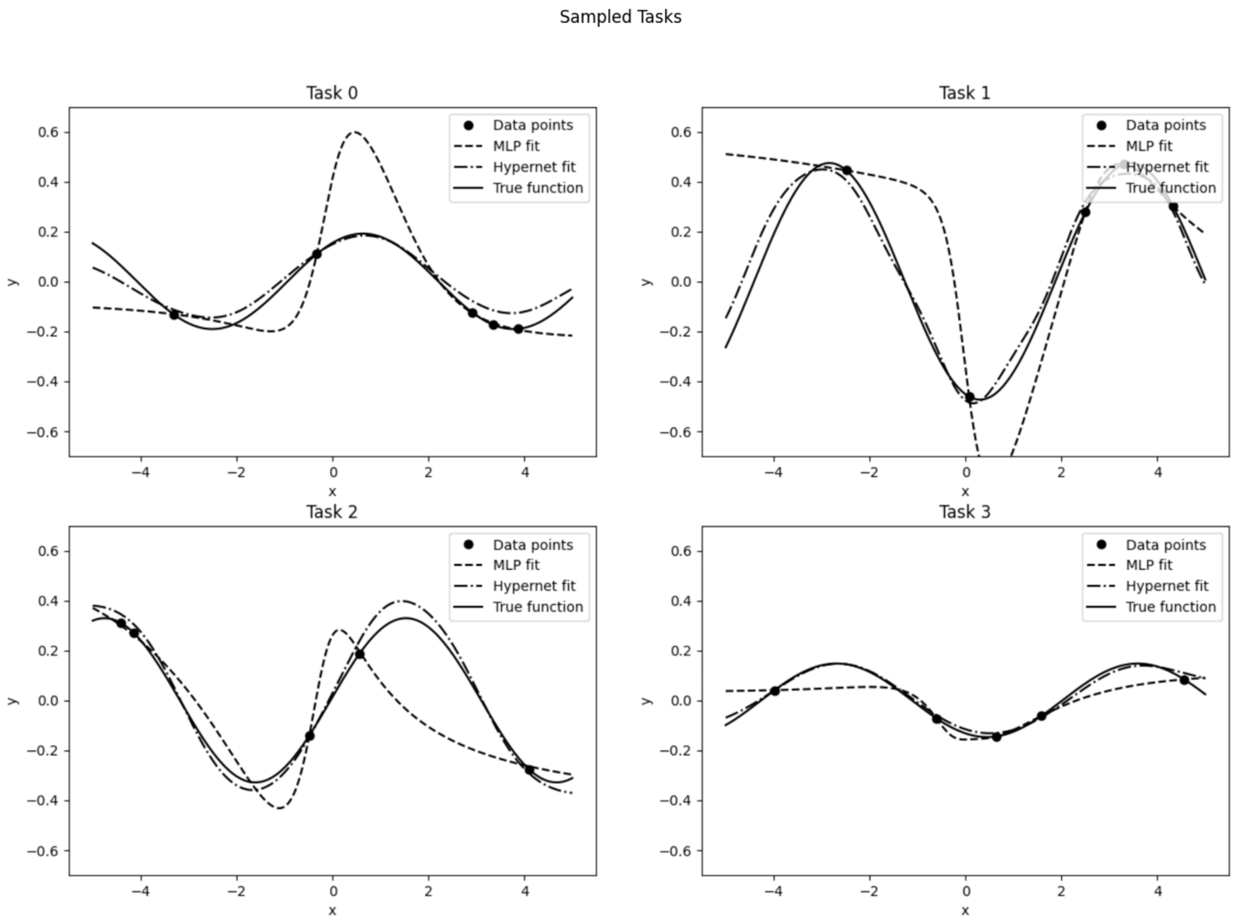


Example from:

Finn, Chelsea, Pieter Abbeel, and Sergey Levine. "Model-agnostic meta-learning for fast adaptation of deep networks.” International conference on machine learning. PMLR, 2017.

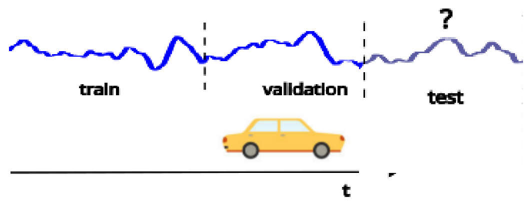
# The “sines” meta learning toy example

- With a black-box structure (eg, a small neural net) we overfit the $K = 5$ data points
- If we knew the structure, $K = 2$ data points would be enough to find $A, \psi$
- What if we observe $K = 5$ points from many datasets? Can we learn the pattern?
- Results with existing meta learning approaches



# From standard to meta system identification

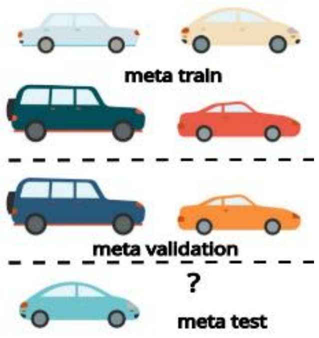
- System Identification
  - **Dataset**: I/O trajectory $D = (u_{1:N}, y_{1:N})$ of one particular system
  - **Objective**: estimate a model from $D$




$$\begin{aligned} \theta \quad &= \mathrm{Alg}(D) \\ \hat{y}^*_{1:N} &= M_\theta(u^*_{1:N}) \end{aligned}$$

  - $\mathrm{Alg}$ is fixed (or *manually* tuned)

- Meta System Identification
  - **Meta dataset**: collection of datasets $\mathfrak{D} = \{D^{(1)}, D^{(2)}, \dots\}$ from similar systems
  - **Objective**: learn to make prediction on all systems in $\mathfrak{D}$



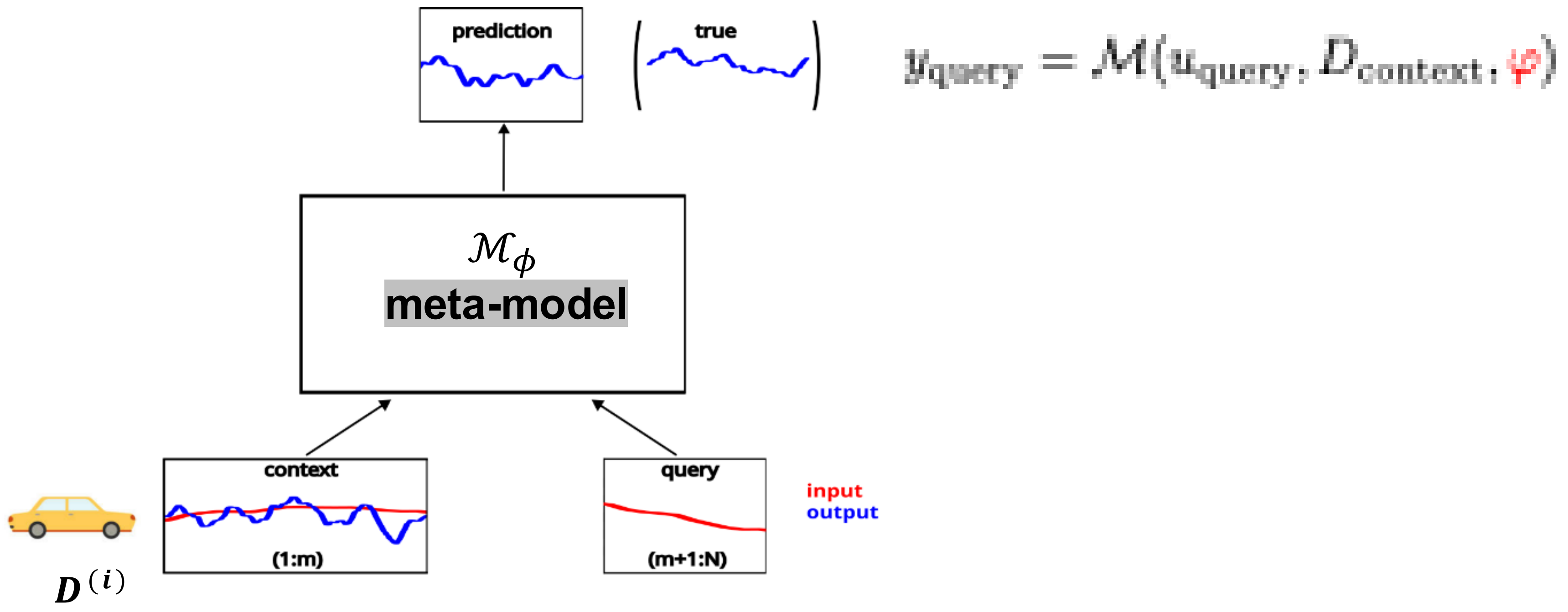

model-based meta learning
$\hat{y}^*_{1:N} = M_{\mathrm{Alg}_\varphi(D)}(u^*_{1:N})$

in-context learning
$\hat{y}^*_{1:N} = \mathcal{M}_\varphi(u^*_{1:N}, D)$

  - $\mathrm{Alg}$ is learned!

# In-context system identification

- Particular dynamical system:
  - Input/output context data, input query → predicted query output:





Forgione, M., Pura, F., & Piga, D. (2023). From system models to class models: An in-context learning paradigm. *IEEE Control Systems Letters*, *7*, 3513-3518.

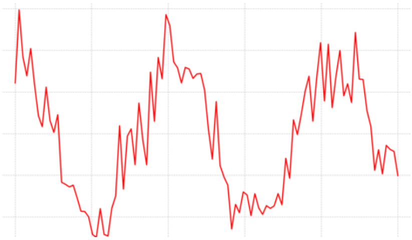
# Large Language Model analogy

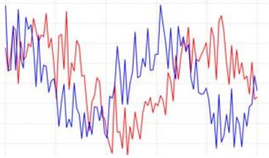
**Output**

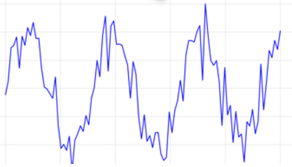
- *Milan: Santa Maria delle Grazie, where you can see Leonardo da Vinci's 'The Last Supper'. The Milan Cathedral (Duomo di Milano) is another must-visit for its breathtaking Gothic architecture.*
- *Turin: Royal Palace of Turin, which offers a glimpse into the lives of the Italian monarchy. The Egyptian Museum in Turin, is also a must for history buffs.*


**Context**
*A user is passionate about history and is planning a trip to North Italy.*

**Query**
*Can you recommend some places to visit?*

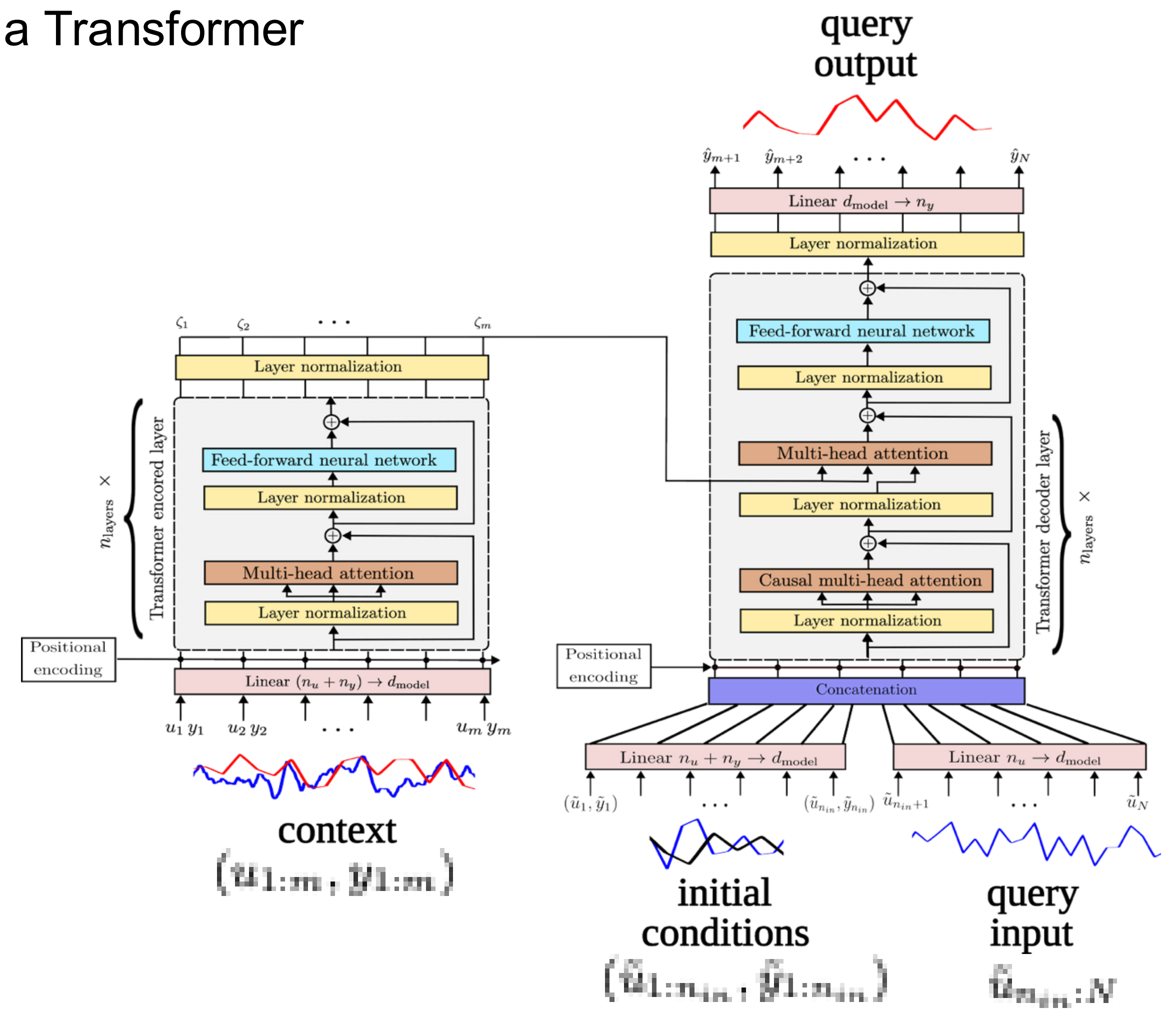






# Meta model architecture

- Meta-model extended to process the query's input/output initial condition
- Parameterized as a Transformer




Rufolo, M., Piga, D., Maroni, G., & Forgione, M. (2025). Enhanced Transformer architecture for in-context learning of dynamical systems. ECC 2025.

# Meta model training

## STEPS

1. Collect a meta dataset of similar systems from a simulator:

$$\mathfrak{D} = (D^{(1)}, D^{(2)}, \dots), \; i = 1, 2, \dots, \infty$$

2. Meta training offline to estimate a meta-model $\mathcal{M}_\phi$:

$$\hat{\phi} = \arg\min_{\phi} \sum_i \mathrm{MSE}(D^{(i)}, \mathcal{M}_\phi)$$

3. Make predictions on new (real) system, in a zero-shot fashion (no training):

$$\hat{\boldsymbol{y}}^* = \mathcal{M}_{\hat{\phi}}(\boldsymbol{u}^*, \boldsymbol{D})$$

- Technically, it is just a supervised learning problem – standard training.
- The trained $\mathcal{M}$ may be seen as a learned identification + simulation algorithm…
- … or as a *specialized* question answering system – a kind of SYSID GPT.

# Meta Training – System class

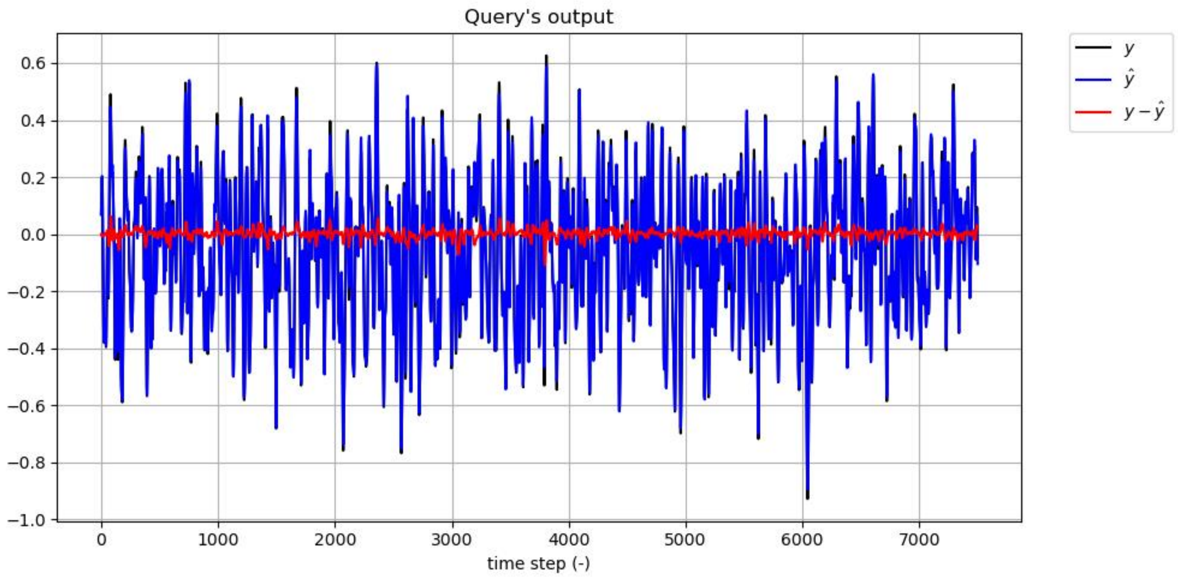
## Wiener-Hammerstein Systems

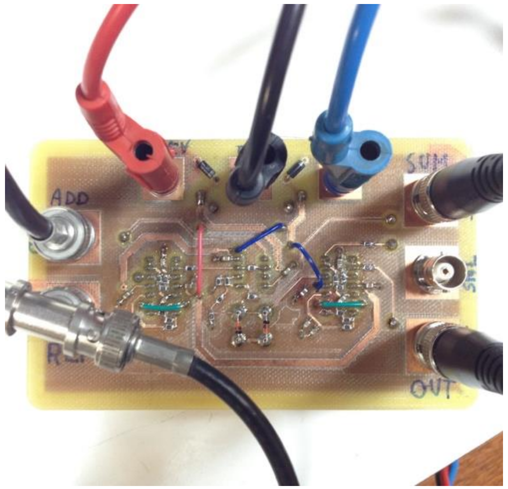

- Sequential LTI $\rightarrow F(\cdot) \rightarrow$ LTI
- Random LTI, order $\leq 5$
- $F(\cdot)$: random feed-forward NN

- Meta training on ~100 M simulated Wiener-Hammerstein systems
- The input $u_{1:N}$ is a multisine signal with random phase and spectrum
- White noise added to the simulated output
- Transformer model with ~5 M parameters, training time ~1 day on an Nvidia 4090 GPU…

# Testing – Nonlinear benchmarks (www.nonlinearbenchmark.org)

- Real Wiener-Hammerstein benchmark



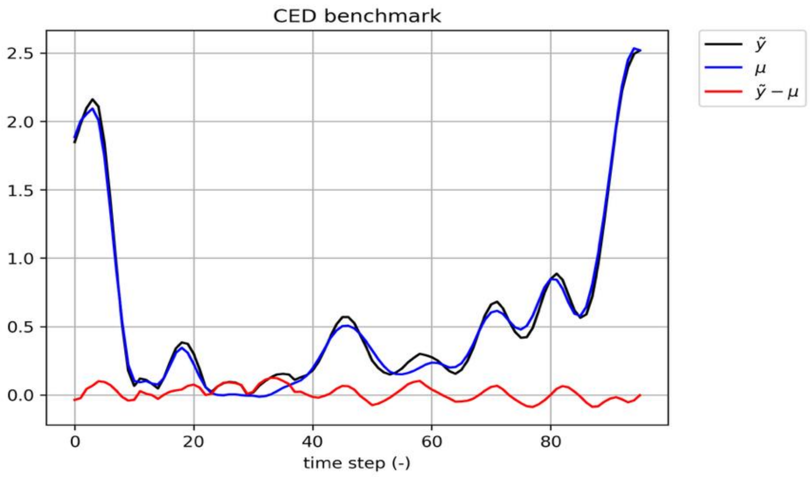

| RMSE * 1000 | Inference time (s) |
|---|---|
| 1.4 | 5.91 |

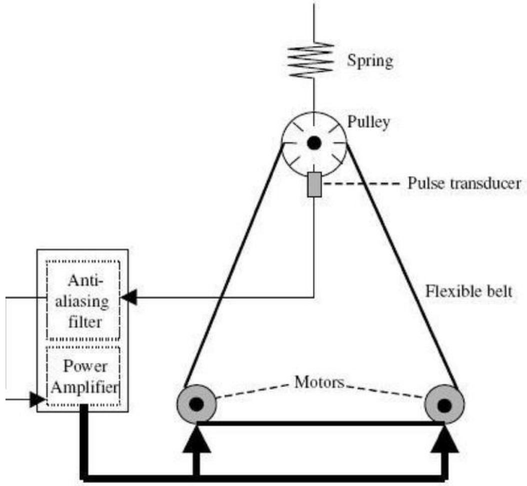
State-of-the-art

Rufolo, M., Piga, D., Pura, F., & Forgione, M. (2025). In-Context Learning for System Identification: Recent Advances. Submitted to a journal, available here.

# Testing – Nonlinear benchmarks

- Coupled Electric Drives (CED)



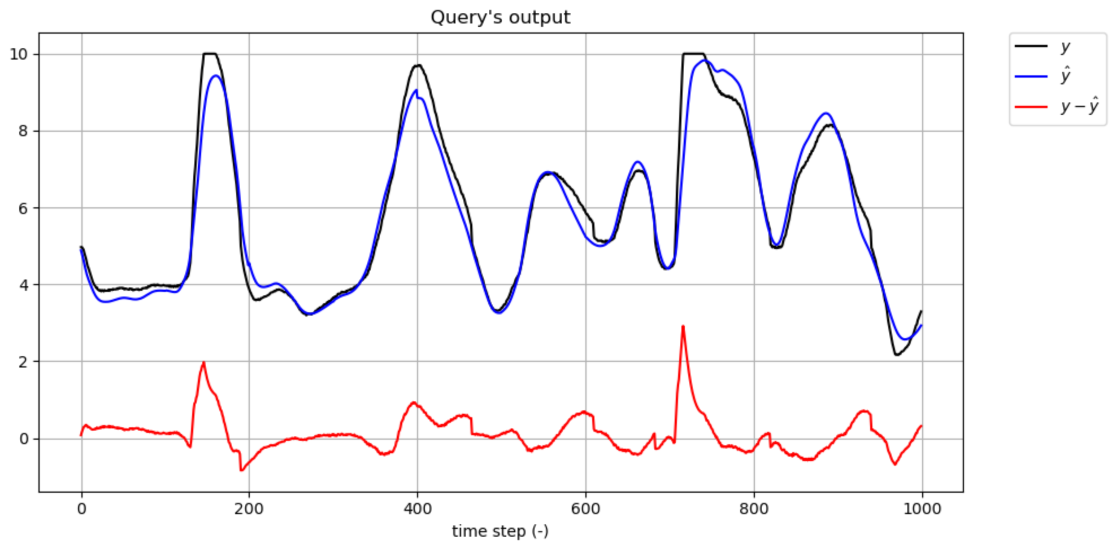

| RMSE | Inference time (s) |
|---|---|
| [0.068,0.055] | 0.17 |

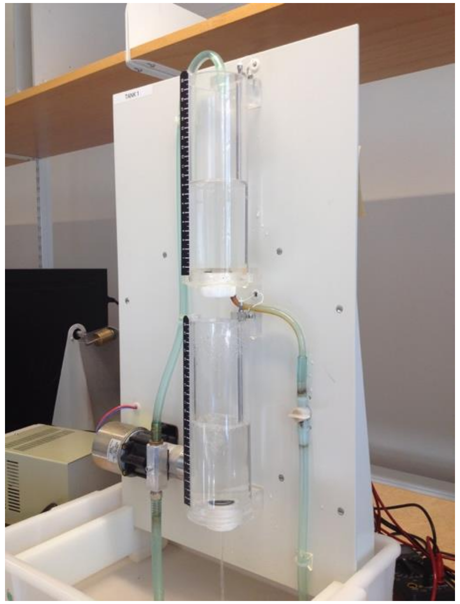
State-of-the-art

Rufolo, M., Piga, D., Pura, F., & Forgione, M. (2025). In-Context Learning for System Identification: Recent Advances. Submitted to a journal, available here.

# Testing – Nonlinear benchmarks

- Cascaded tanks with Overflow



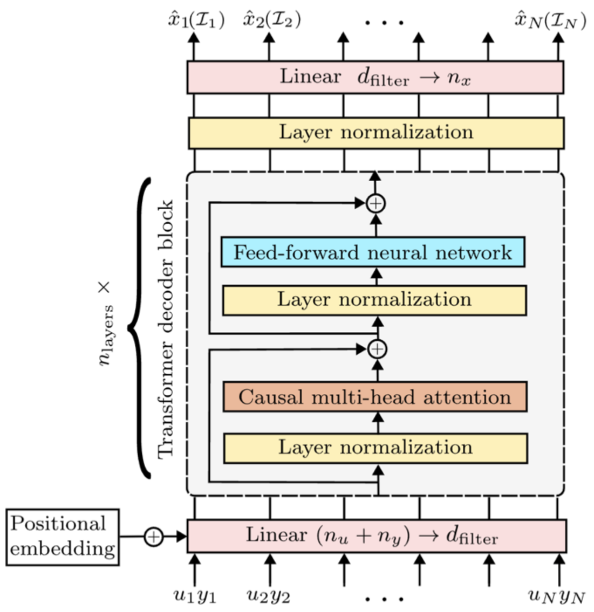
| RMSE | Inference time (s) |
|---|---|
| 0.429 | 0.22 |

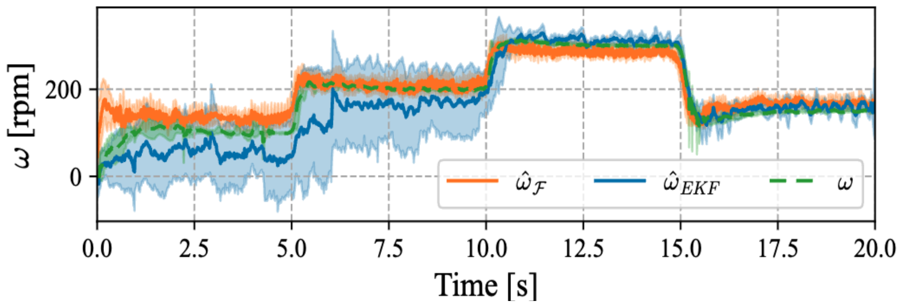
State-of-the-art

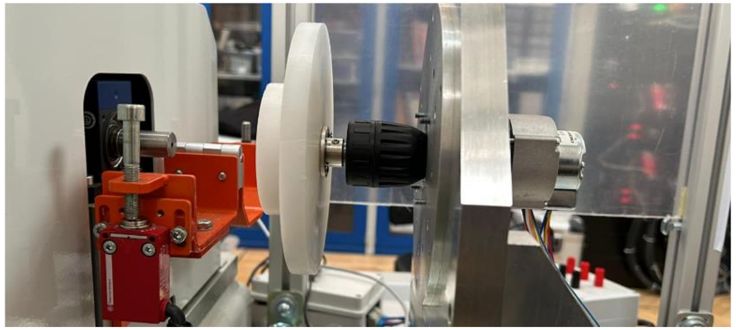
Rufolo, M., Piga, D., Pura, F., & Forgione, M. (2025). In-Context Learning for System Identification: Recent Advances. Submitted to a journal, available here.

# Other objectives: meta state estimation

- A meta state estimator that generalizes to different systems from the context



- Experiments on speed estimation of BLDC motors


Colombo, A., Busetto, R., Breschi, V., Forgione, M., Piga, D., & Formentin, S. (2025). In-Context Learning for Zero-Shot Speed Estimation of BLDC motors. arXiv preprint arXiv:2504.00673.

Busetto, R., Breschi, V., Forgione, M., Piga, D., & Formentin, S. (2024). In-context learning of state estimators. IFAC-PapersOnLine, 58(15), 145-150. Presented at SYSID 2024.

# Whitening the meta-learning box

The meta-learning grayscale

model-based meta learning

$$\hat{y}^*_{1:N} = M_{\mathrm{Alg}_{\varphi}(D)}(u^*_{1:N})$$

in-context learning

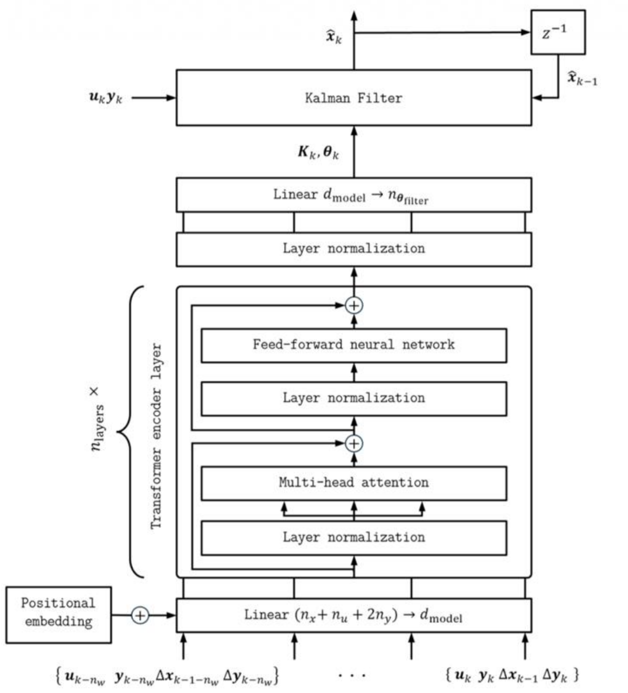
$$\hat{y}^*_{1:N} = \mathcal{M}_{\varphi}(u^*_{1:N}, D)$$

- In-context learning: prediction is a black-box function of dataset and new data.
- Other extreme: learn hyper-parameters $\varphi$ of algorithm $\mathrm{Alg}_{\varphi}$ that returns (interpretable) model parameters
  - Learn the learning rate, initial value for gradient-based optimization (MAML), etc

Hospedales, T., Antoniou, A., Micaelli, P., & Storkey, A. (2021). Meta-learning in neural networks: A survey. IEEE transactions on pattern analysis and machine intelligence, 44(9), 5149-5169.

# Meta state estimation: whitening the box

- Transformer returns estimator gain $K$ + physical parameters $\theta$ instead of predictions directly.




Model-based meta learning  In-context learning

- On-going work with Université de Poitiers (Hugo Koide, Guillaume Mercère)

# Conclusions

**Takeaways:**

- In-context system identification, estimation, & control
- Off-line: learn how to map data into prediction
- On-line: provide context data, query input → query output (zero shot)

**Current activities:**
- Real-world applications
- Whitening the meta-learning box
- Providing guarantees