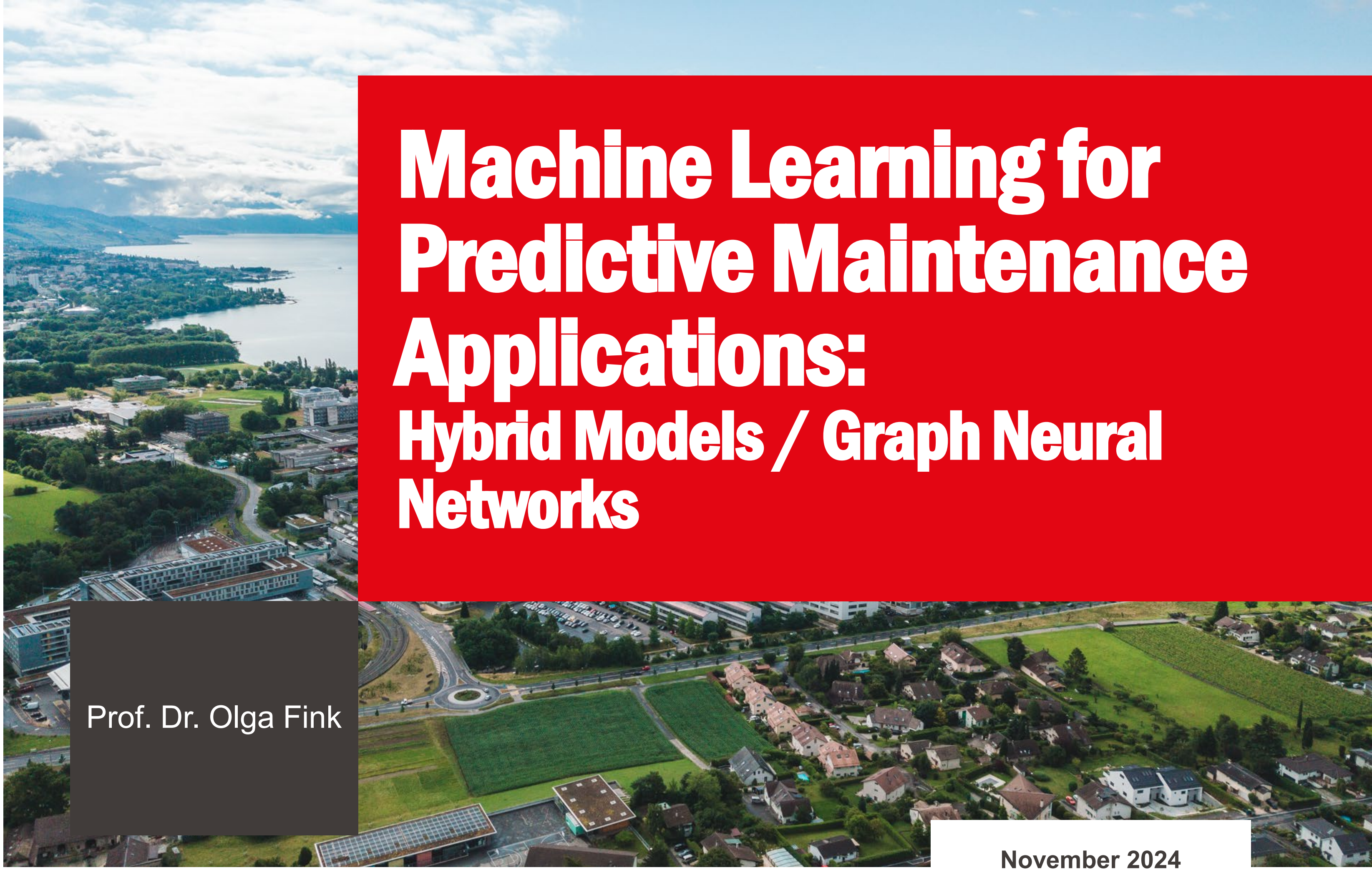
EPFL



Machine Learning for Predictive Maintenance Applications:

Hybrid Models / Graph Neural Networks

Prof. Dr. Olga Fink

November 2024

École polytechnique fédérale de Lausanne

Domain generalization

- Domain generalization focuses on training models that generalize to unseen target domains.
- Unlike domain adaptation, it does not require access to target domain data during training.
- It aims to learn robust and invariant features from source domains that perform well under different conditions.
- Useful when target environments are unknown or change dynamically.
- Helps create models that are more adaptable and resilient to distribution shifts or novel scenarios.

Continuous domain adaptation

- Continuous domain adaptation focuses on adapting a model continuously as the target domain evolves over time.
- Unlike traditional domain adaptation, it handles scenarios where data distributions change dynamically and require ongoing adjustments.
- It seeks to maintain model performance by adapting incrementally to new conditions without needing to retrain from scratch.
- Useful for real-world applications where environments and data distributions are constantly shifting.
- Often involves strategies for updating learned representations, retaining past knowledge, and mitigating catastrophic forgetting.

Test-time adaptation

- Test-time adaptation refers to adapting a model to distribution shifts in the target domain during inference, without further training on the source domain.
- It focuses on updating model parameters or representations using only test data, typically through self-supervised or unsupervised strategies.
- Allows models to respond to domain shifts or changes at the point of inference, improving performance in unseen or evolving conditions.
- Reduces the reliance on having a large amount of labeled data in the target domain.
- Useful for scenarios where models encounter dynamic environments or previously unseen data distributions at test time.

Digital Twins

1. Definition: Digital twins

- A digital twin is defined as "a living model of the physical asset or system, which continually adapts to operational changes based on the collected online data and information, and can forecast the future of the corresponding physical counterpart".

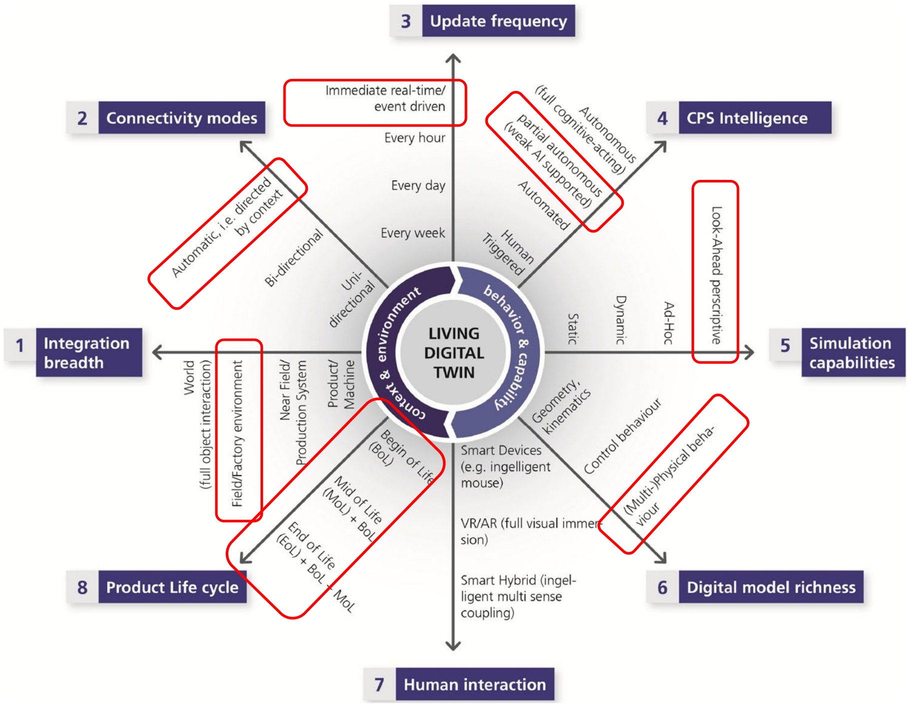
M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, Springer International Publishing, 2016, pp. 85–113.

2. Definition: Digital twins

- *A digital twin is a set of virtual information constructs that mimics the structure, context, and behavior of a natural, engineered, or social system (or system-of-systems), is dynamically updated with data from its physical twin, has a predictive capability, and informs decisions that realize value. The bidirectional interaction between the virtual and the physical is central to the digital twin.*

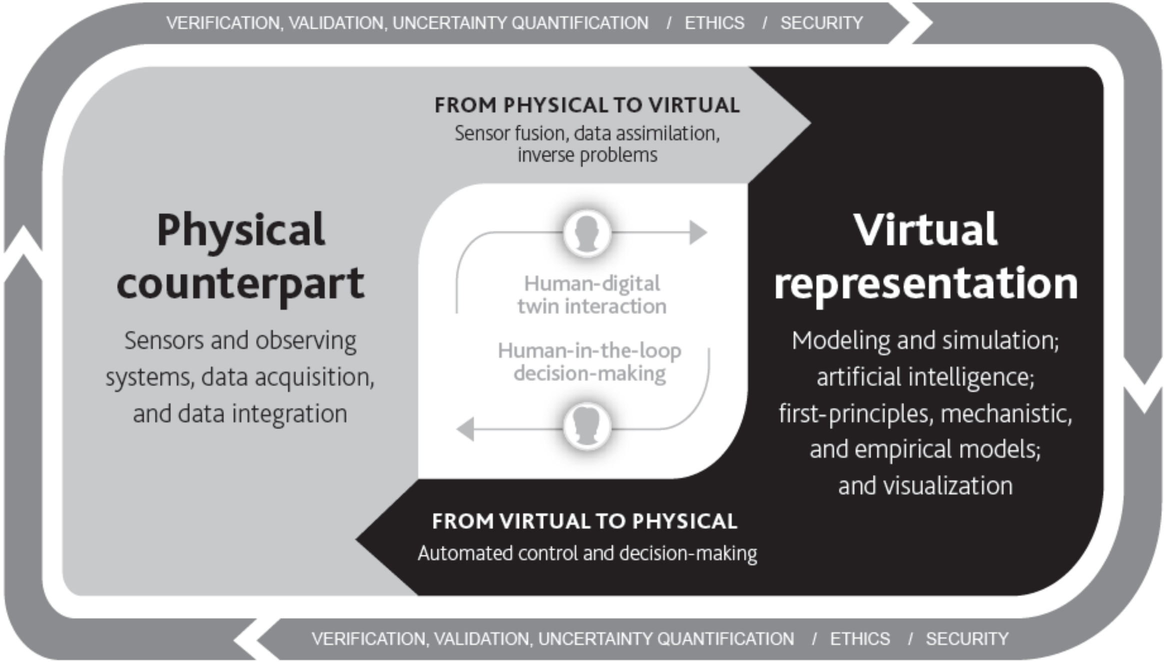
National Academies 2023: Foundational Research Gaps and Future Directions for Digital Twins

Digital Twin integration



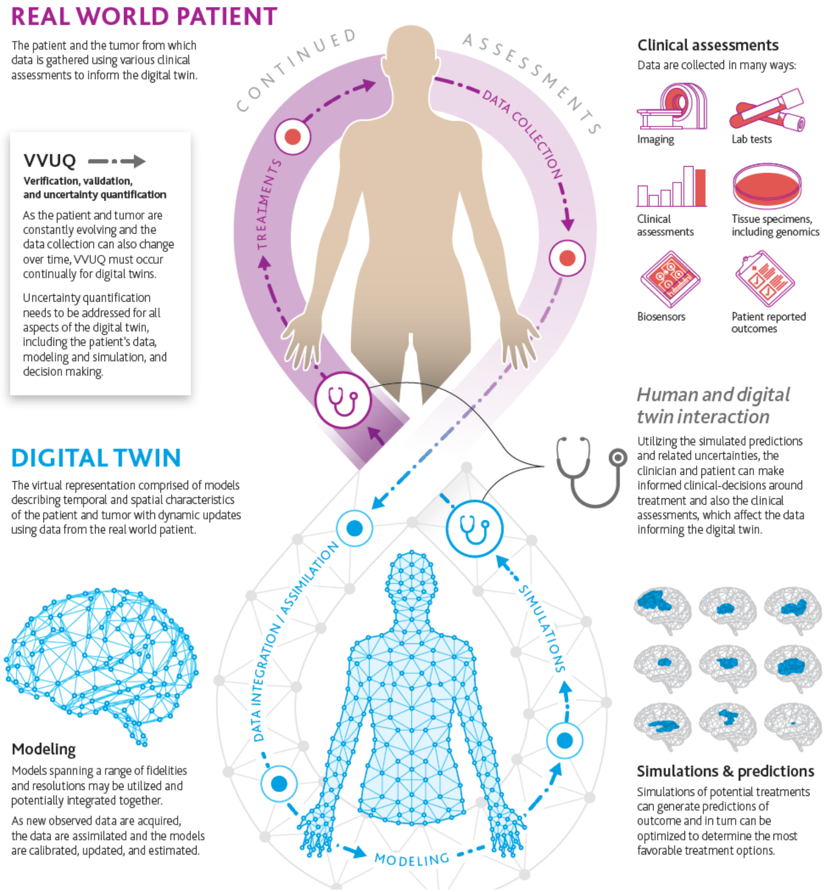
R. Stark, C. Fresemann, and K. Lindow, "Development and operation of Digital Twins for technical systems and services," *CIRP Ann.*, vol. 68, no. 1, pp. 129–132, Jan. 2019.

Elements of a digital twin



National Academies 2023: Foundational Research Gaps and Future Directions for Digital Twins

Digital twin of a patient



National Academies 2023: Foundational Research Gaps and Future Directions for Digital Twins

Verification, Validation, Uncertainty Quantification

- **Verification.** Does a computer program correctly solve the equations of the mathematical model?
- **Validation.** To what degree is a model an accurate representation of the real world, from the perspective of the intended model uses?
- **Uncertainty Quantification.** What are uncertainties in model calculations of quantities of interest?

National Academies 2023: Foundational Research Gaps and Future Directions for Digital Twins

Application fields

- **Manufacturing**
- Smart Cities
- Health care
- Energy Sector
- Aerospace and Defense
- Automotive Industry
- Building Energy Systems
- Agriculture
- Climate
- …

Application cases of DT

- Product development
- Design + Testing
- PHM / Maintenance
- Optimizing product life-cycle management
- Process Optimization
- Prescriptive Operation

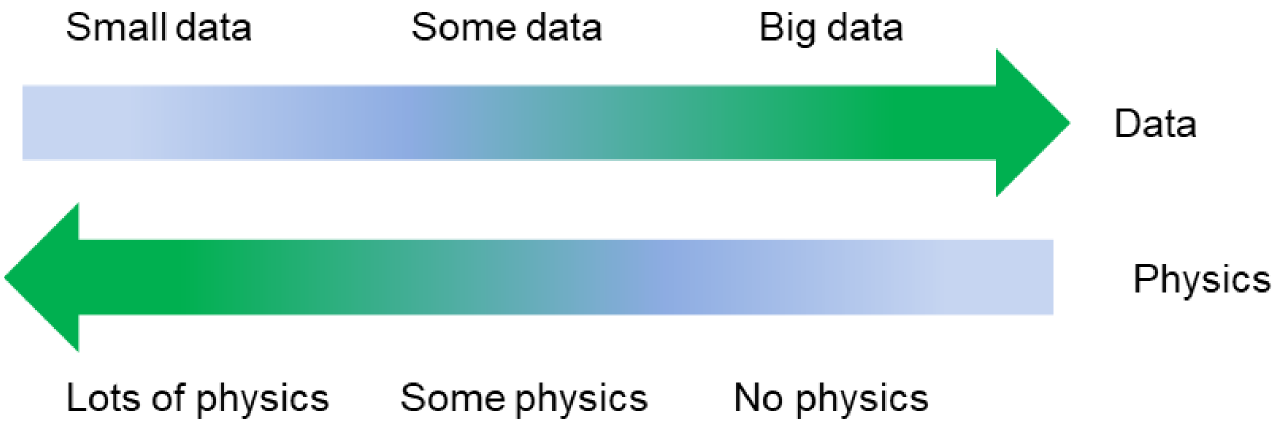
A Digital Twin should be Fit for Purpose

- **Balancing required fidelity for prediction, available resources, and acceptable costs**
- Different digital twin purposes drive different fitness requirements related to modeling fidelity, data availability, visualization, time-to-solution, etc.
- For many potential use cases, achieving fitness-for-purpose is currently intractable

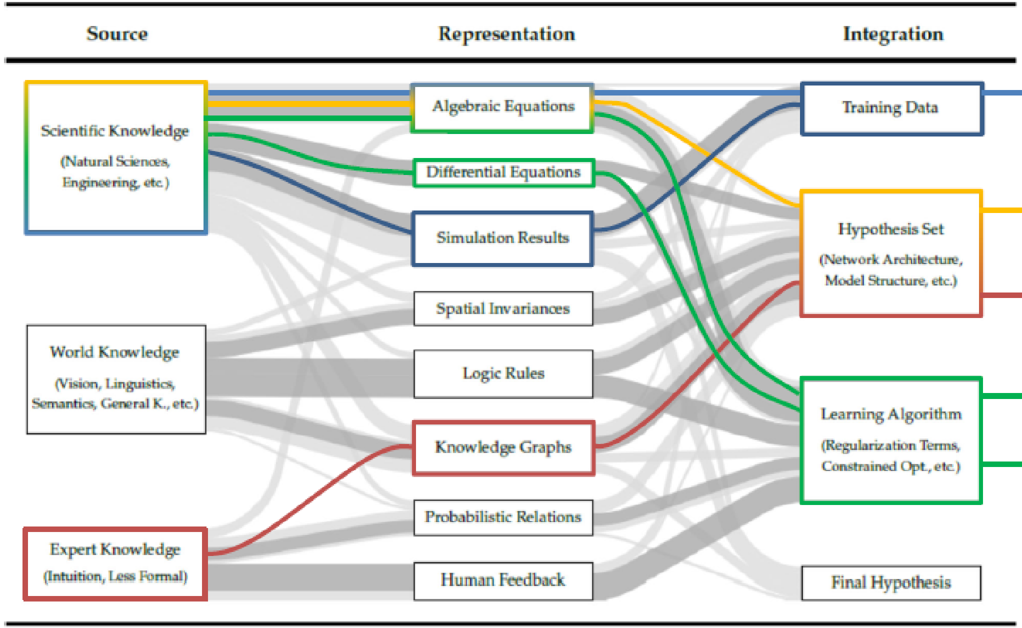
National Academies 2023: Foundational Research Gaps and Future Directions for Digital Twins

Hybrid Models

Space of potential solutions

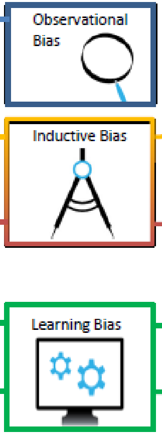


Physics-informed ML for prognostics and health management (PHM)



L. von Rueden *et al.*, (2021)

PIML methods in PHM:



M. Arias Chao, et. al (2021).

R. Nguyen, et. al (2023).

Felipe A.C.Viana (2021).

Jyoti Narwariya et. al (2020).

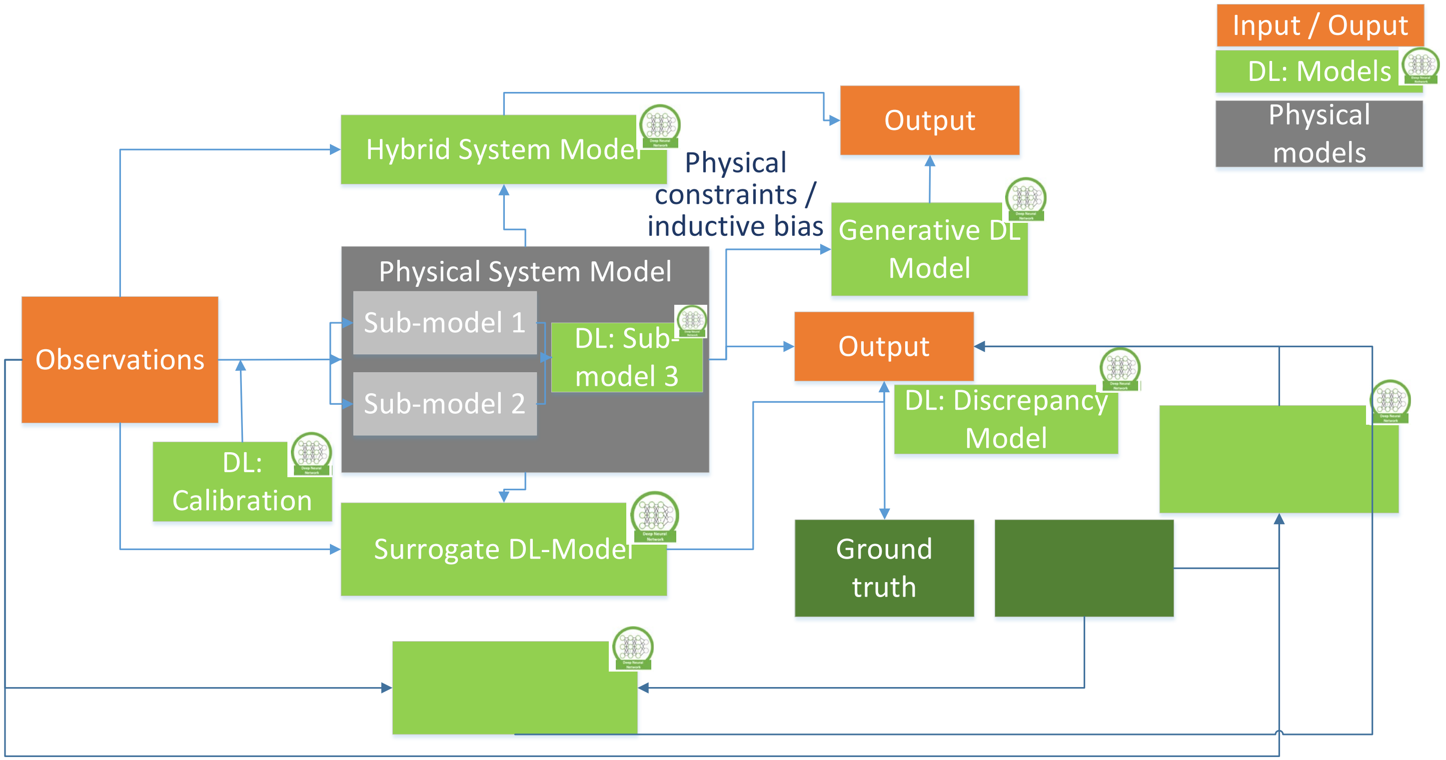
Deng, W., Nguyen, et. al (2022).

Sergio Cofre-Marte, et. Al (2021).

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. Nature Reviews Physics 2021

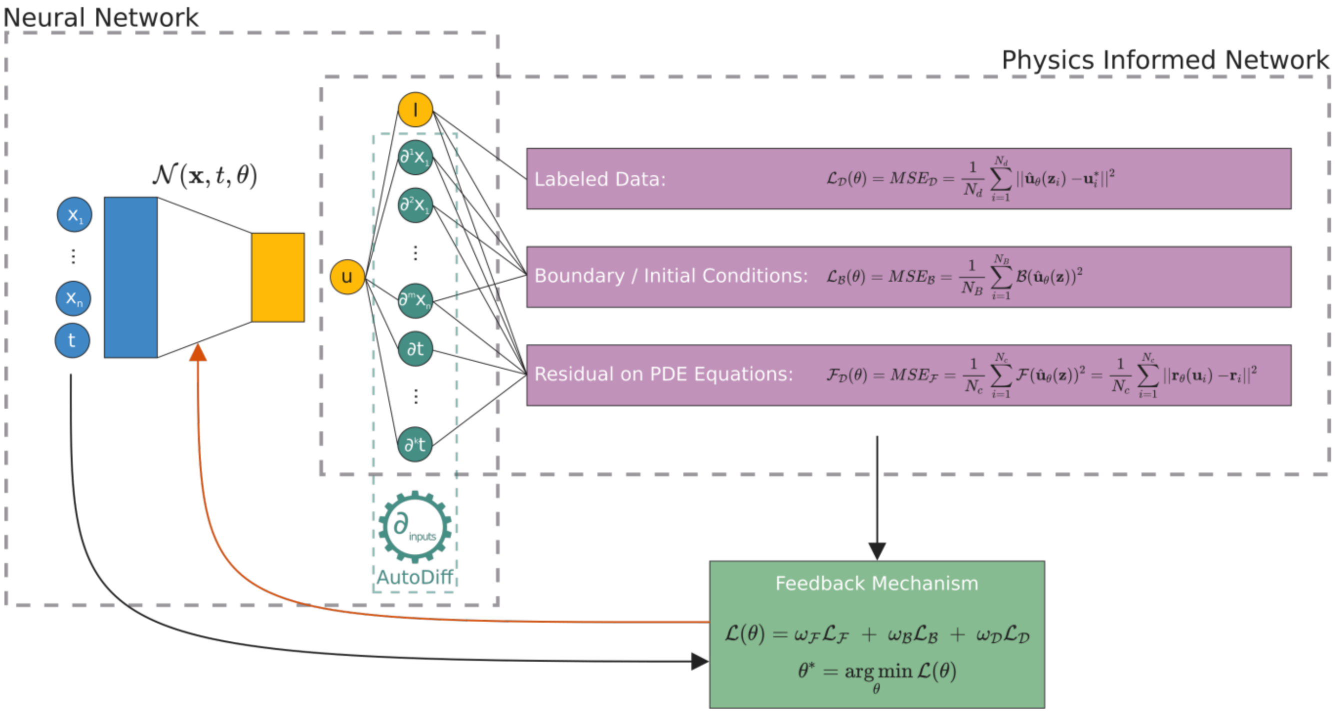
Source: M. Arias

Different ways of fusing DL and physics



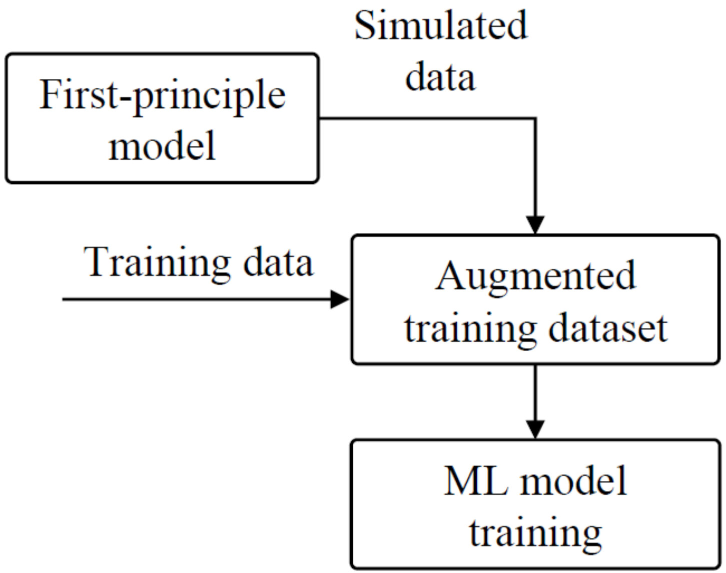
Combining Physics-Based and Deep Learning Models

Physics-informed ML



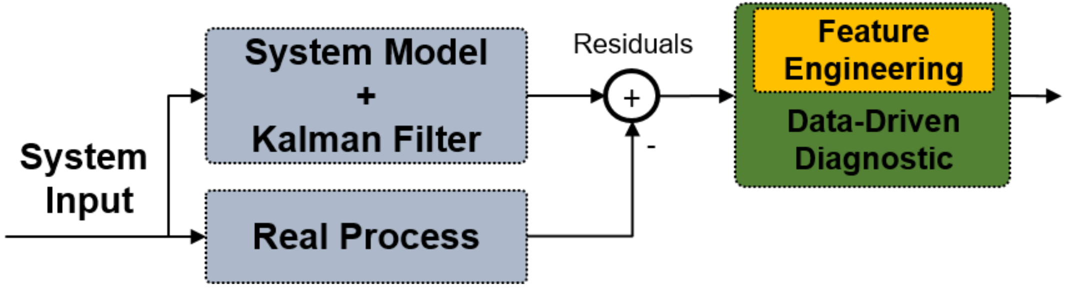
Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

Data Augmentation



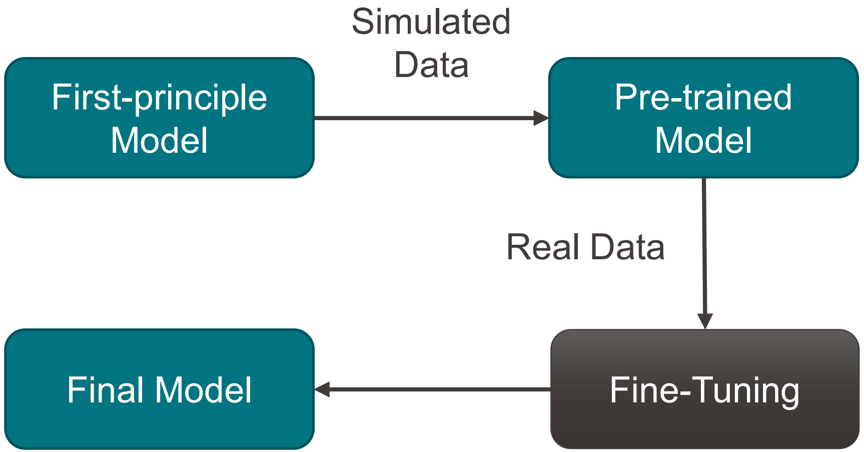
Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

Residual-model



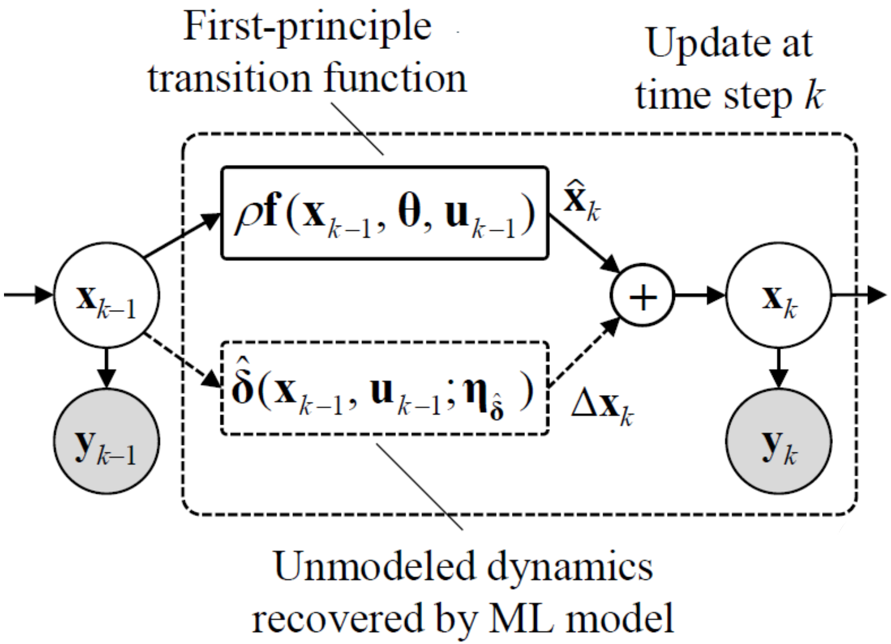
Rausch, Randal T., et al. "Integrated in-flight fault detection and accommodation: A model-based study." (2007): 962-969.

Transfer learning



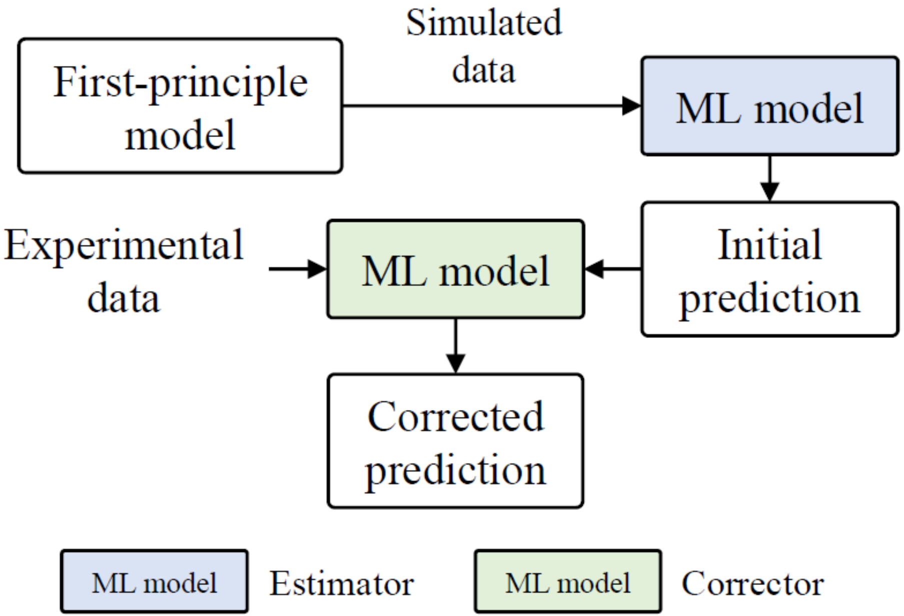
Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

Delta Learning (Missing Physics)



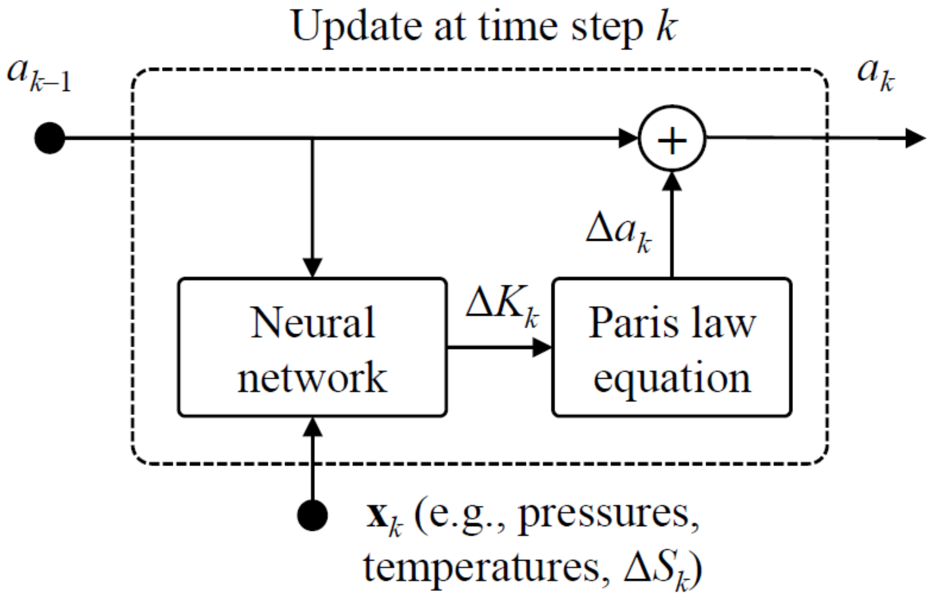
Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

Delta Learning (ML Prediction)



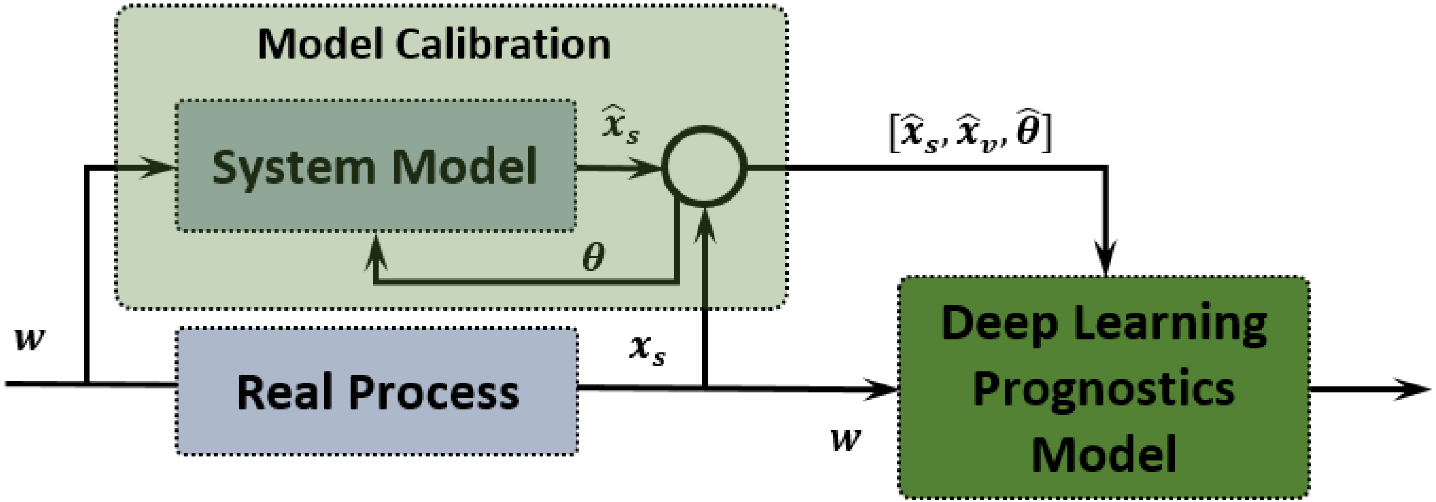
Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

ML-Assisted Prediction



Thelen, A., X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu. "A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies]{A Comprehensive Review of Digital Twin - Part 1: Modeling and Twinning Enabling Technologies.“¨, accepted for publication in Structural and Multidisciplinary Optimization journal

Fusing physical performance models and deep learning



M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, “Fusing Physics-based and Deep Learning Models for Prognostics,” Reliability Engineering & System Safety.

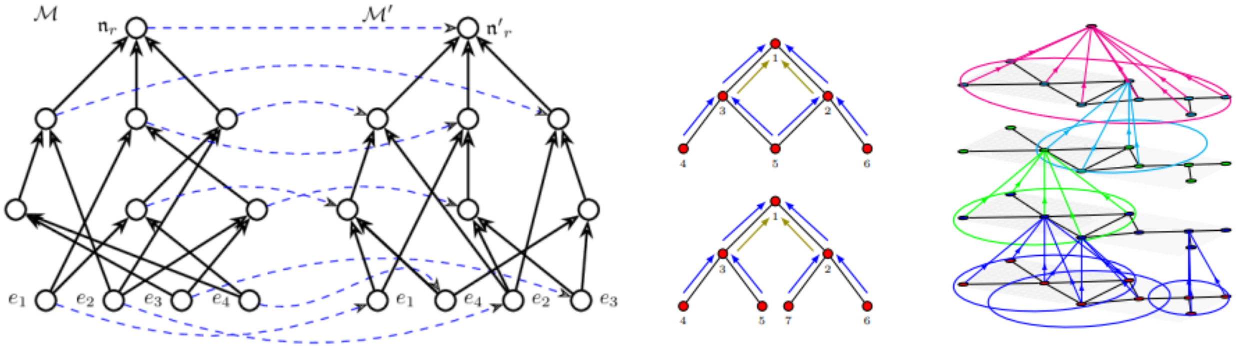
Inductive bias

Inductive bias

- Inductive bias refers to the set of assumptions a learning algorithm uses to generalize from training data to unseen data.
- It helps guide the learning process and shapes the predictions that a model makes.
- Strongly impacts how well a model generalizes to new data.
- Examples include:
  - Linear regression assumes a linear relationship between input and output.
  - Convolutional Neural Networks (CNNs) assume spatial hierarchies and local dependencies for image data.
- Enables efficient learning by reducing the hypothesis space.
- Can lead to poor generalization if the assumptions do not match the underlying data distribution.

Inductive bias / Physics Informed Neural Networks (1/2)

1. Physics is implicitly baked in specialized neural architectures with strong inductive biases (e.g. invariance to simple group symmetries).



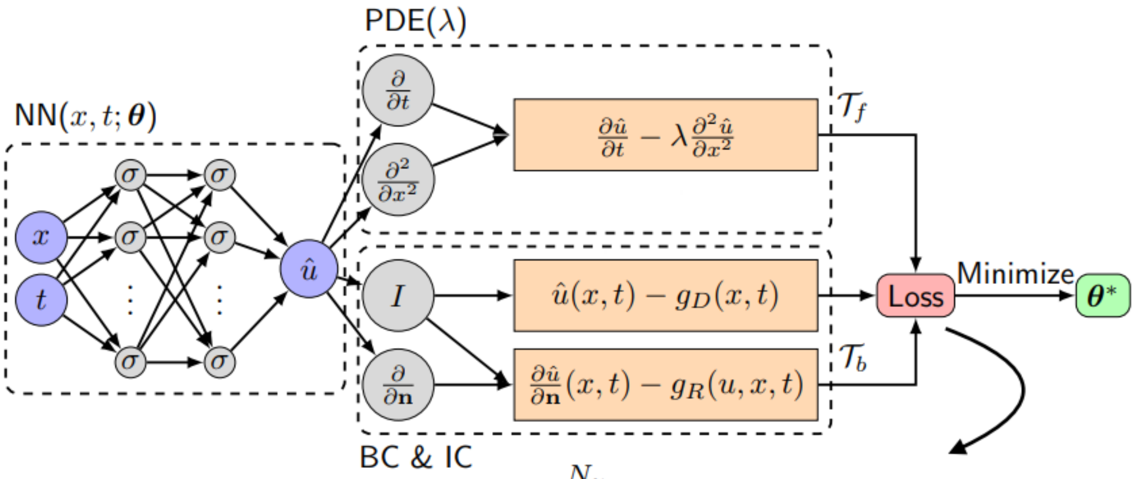
*figures from Kondor, R., Son, H.T., Pan, H., Anderson, B., & Trivedi, S. (2018). Covariant compositional networks for learning graphs. arXiv preprint arXiv:1801.02144.

Source: Perdikaris 2020

Inductive bias / Physics Informed Neural Networks

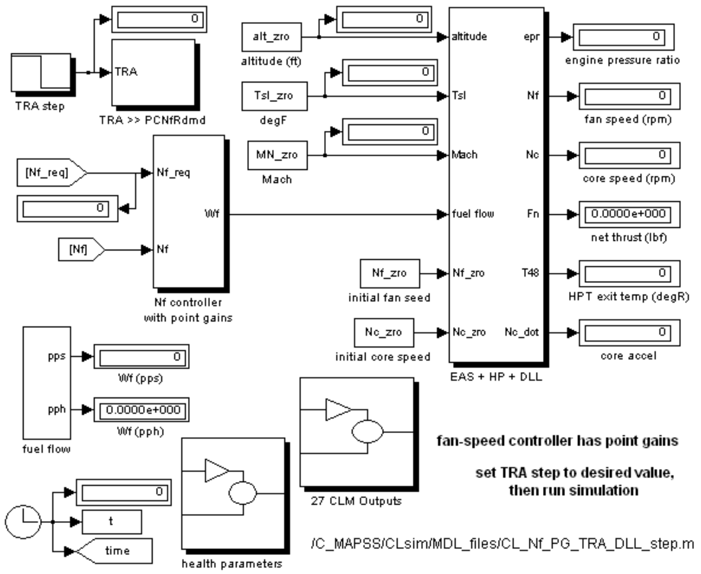
2. Physics is explicitly imposed by constraining the output of conventional neural architectures with weak inductive biases.

Psichogios & Ungar, 1992
Lagaris et. al., 1998
Raissi et. al., 2019
Lu et. al., 2019
Zhu et. al., 2019



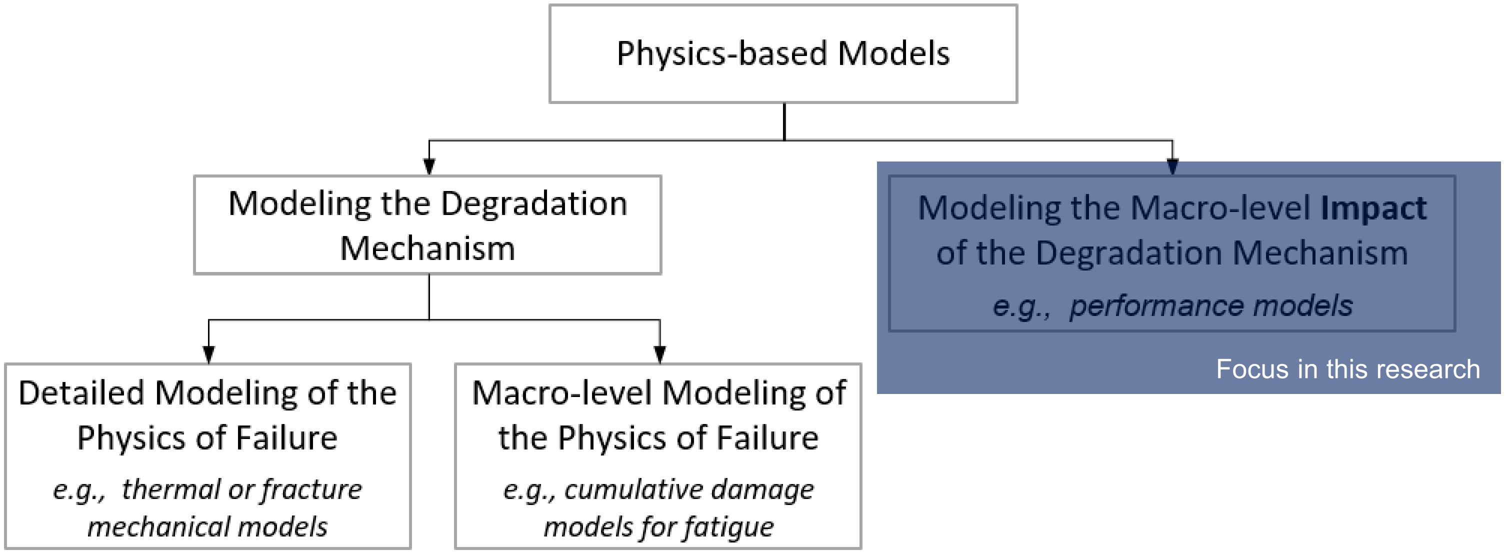
$$\mathcal{L}(\theta) := \underbrace{\frac{1}{N_u}\sum_{i=1}^{N_u}[\boldsymbol{u}_i - f_\theta(\boldsymbol{x}_i)]^2}_{\text{Data fit}} + \underbrace{\frac{1}{\lambda}\mathcal{R}[f_\theta(\boldsymbol{x})]}_{\text{Physics regularization}}$$

Source: Perdikaris 2020



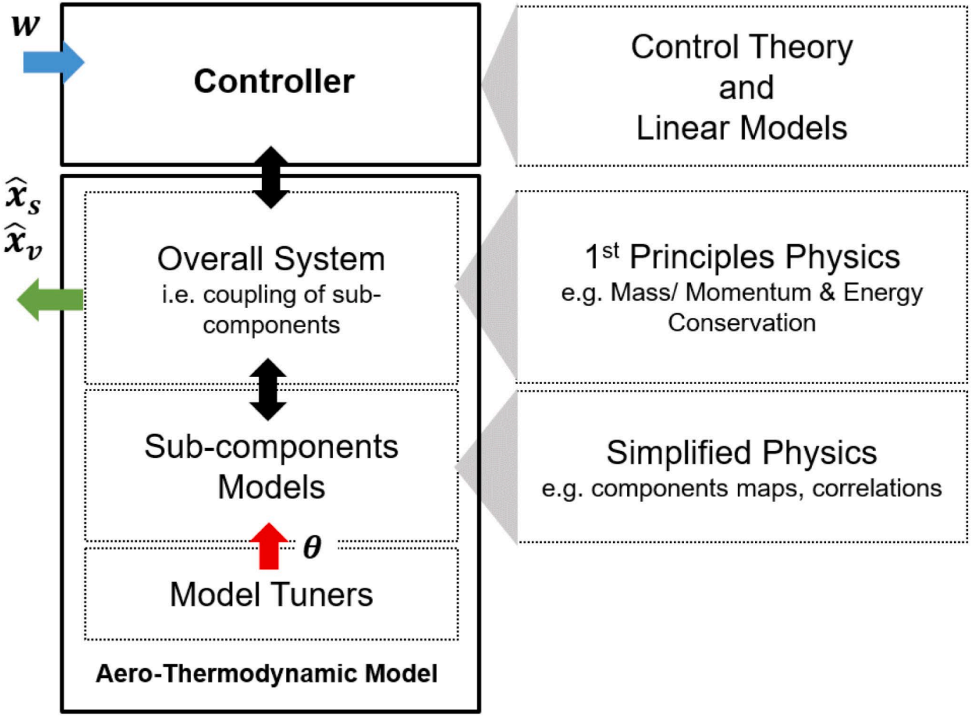
Calibration-based hybrid framework for prognostics

Physics-based models



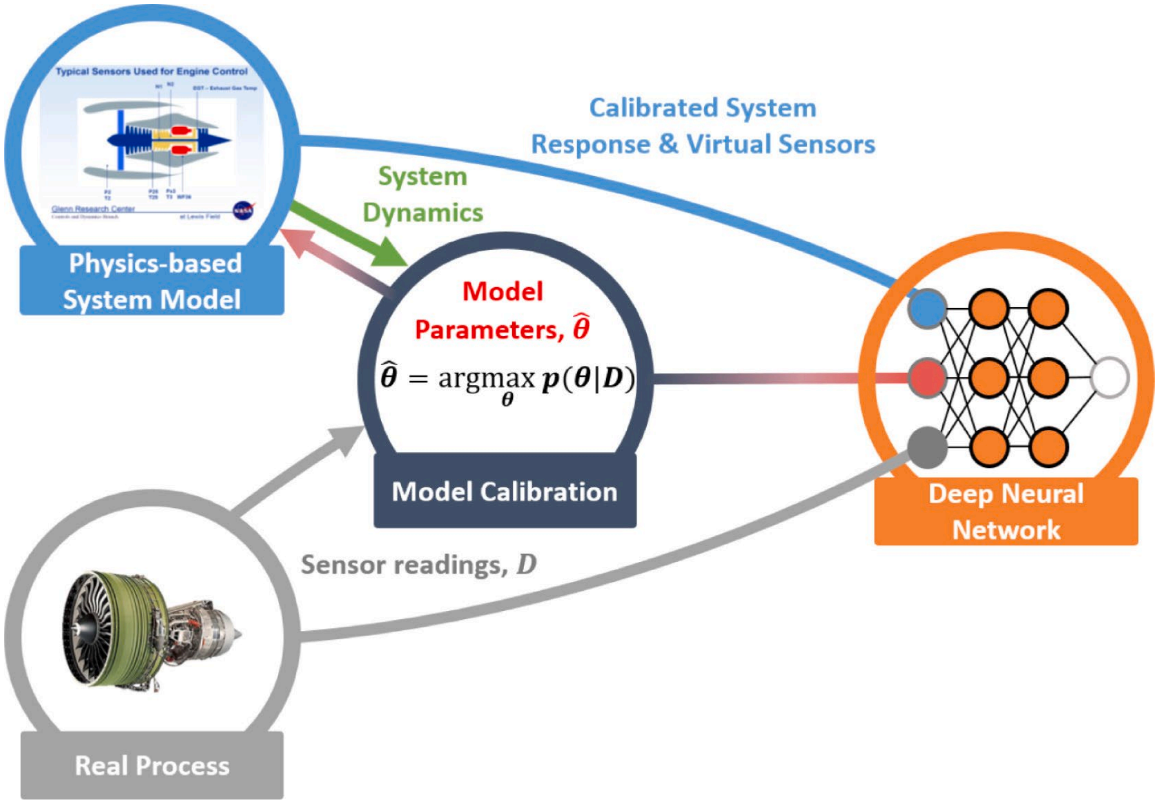
M. Arias, Algorithm for Fleet Diagnostics and Prognostics Combining Deep Learning and Physics-based Performance Models (2021)

General topology of an aero-thermodynamic performance model



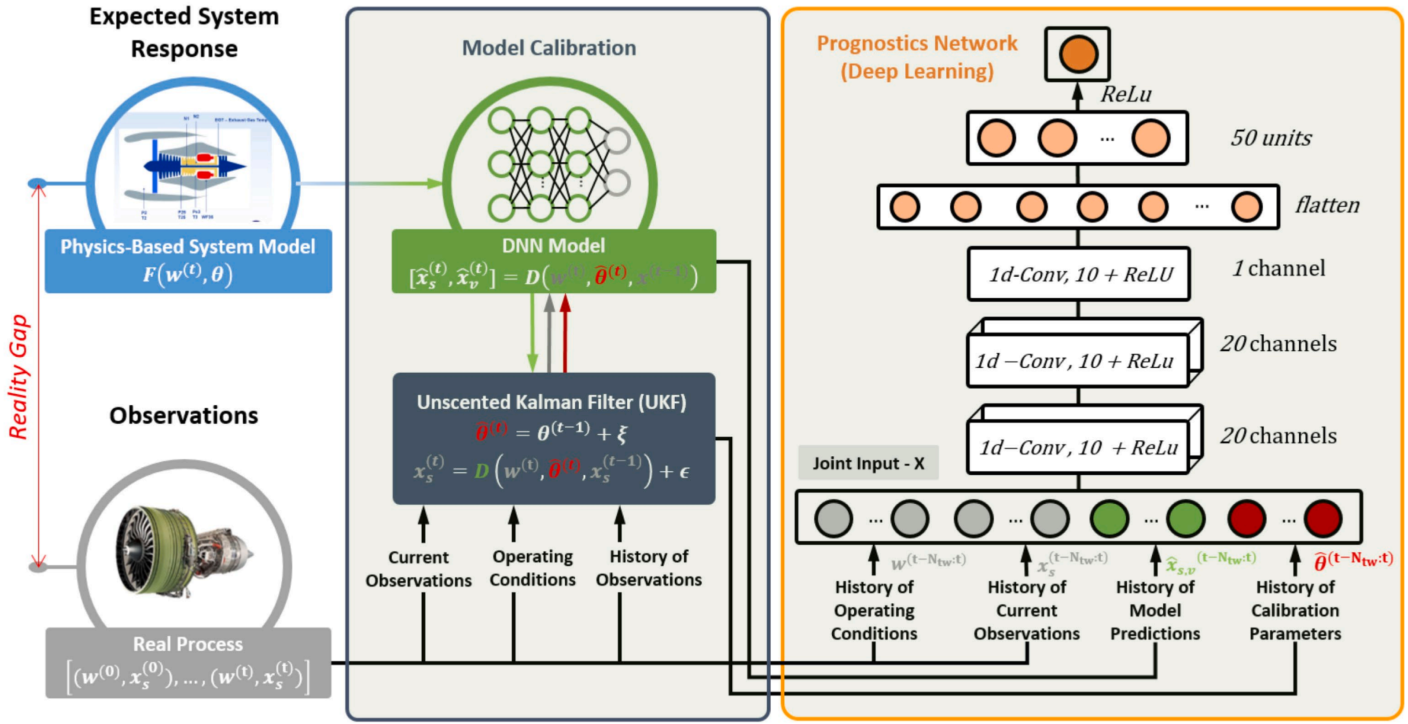
M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Basic idea of the hybrid approach



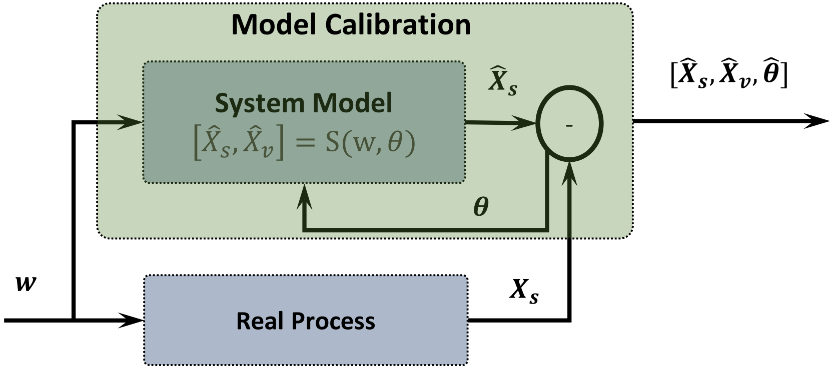
M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Proposed framework

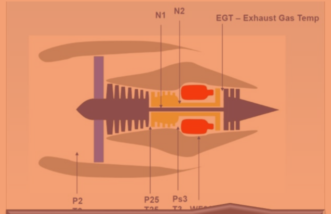


M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Enhancing the input space with inputs from physics-based models



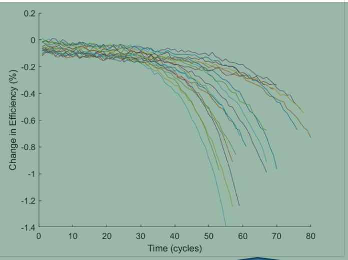
Purely data-driven



Process parameter measurements

Inferred from physics-based models

Virtual sensors



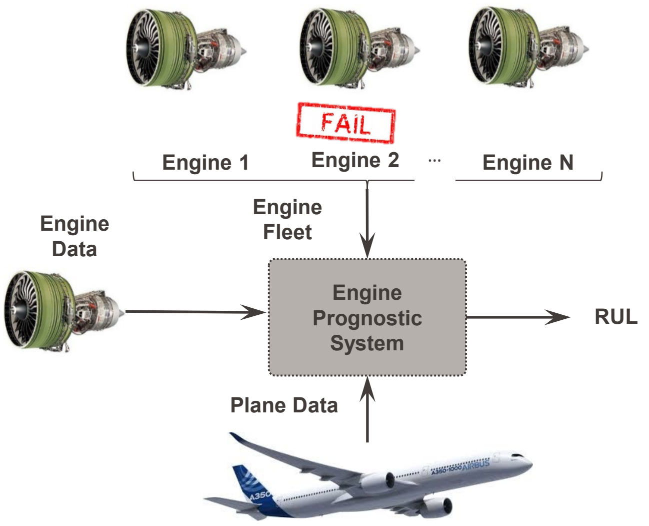
Calibration parameters

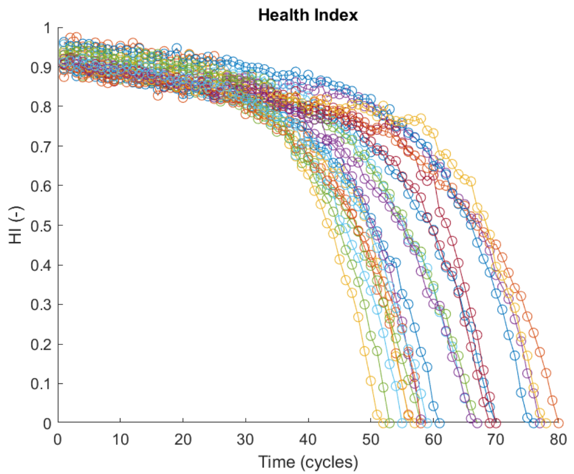
M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, “Fusing Physics-based and Deep Learning Models for Prognostics,” Reliability Engineering & System Safety.

Estimation of the Remaining Useful Life

Remaining Useful Life (RUL) of a component

– the amount of time a component can be expected to continue operating within its stated specifications

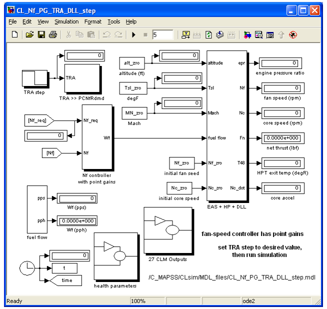




M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

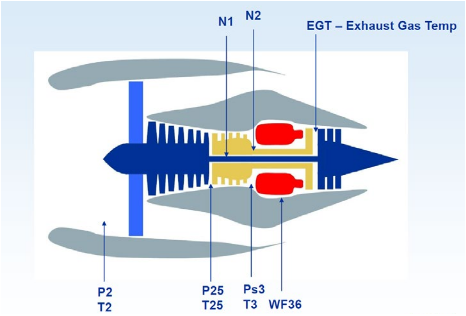
Test Case: Turbofan Engine

- **Simulation of a realistic large commercial turbofan engine**
  – Simulated **run to failure** trajectories with the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dynamical model



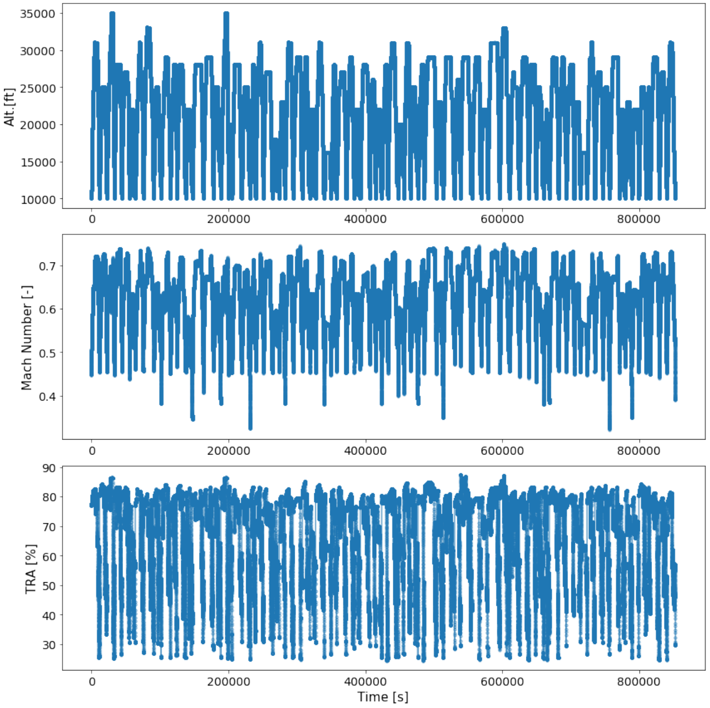
- **Real flight conditions from a commercial jet**
  – NASA **DASHlink**
  – ~500 different (1-12h) flights
  – Recordings covering climb, cruise and descend
  – Operative conditions - $w \in R^3$

- **Data from a fleet of 9 turbofan engines**
  – Internal sensors - $x_s \in R^{10}$
  – Virtual sensors - $x_v \in R^{15}$
  – Model parameters - $\theta \in R^{10}$



M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

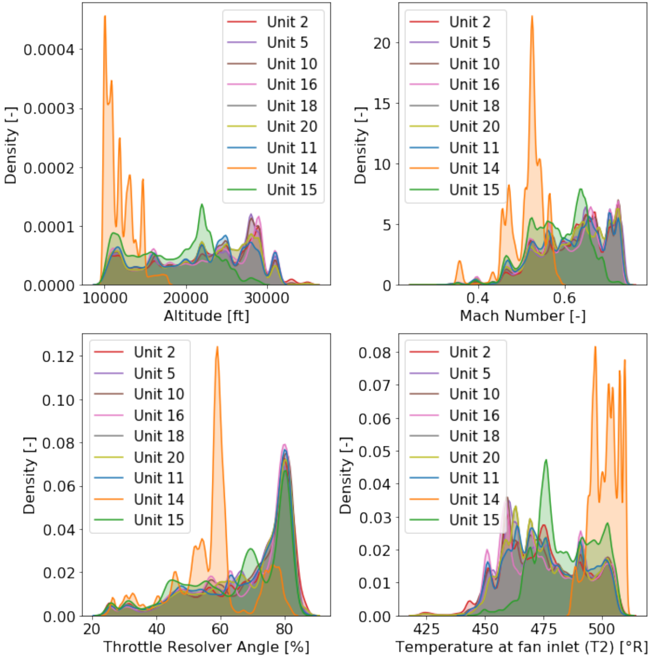
Test Case: Turbofan Dataset



- **Failure modes:** 1) HPT degradation 2) HPT and LPT flow and efficiency degradation
- **Flight conditions:** Alt > 10000 [ft]
- **Training Data:**
  - **6 engines**
  - **50-90 cycles (flights)**
  - $\boldsymbol{w}, \boldsymbol{X_s}, \hat{\boldsymbol{\theta}}$ **&** RUL
  - **5.5** x 10^6 **samples**
- **Test Data:**
  - 3 engines
  - $\boldsymbol{w}, \boldsymbol{X_s}$ & $\hat{\boldsymbol{\theta}}$
  - 1.2 x 10^6 samples

M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Kernel density estimations of the simulated flight envelopes given by recordings of altitude



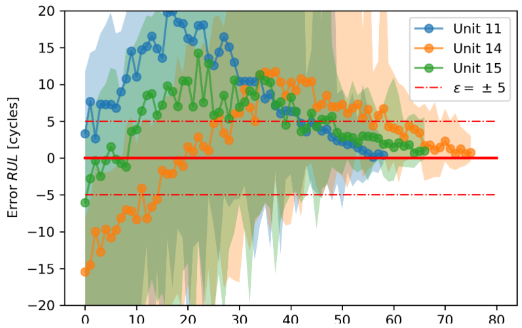
M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, “Fusing Physics-based and Deep Learning Models for Prognostics,” Reliability Engineering & System Safety.

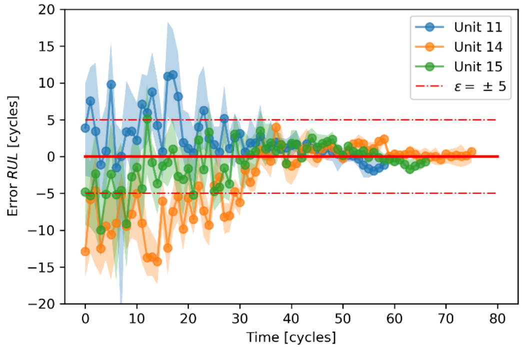
RUL prediction results

Data-driven - $\boldsymbol{X} = [\boldsymbol{w}, \boldsymbol{X_s}]$

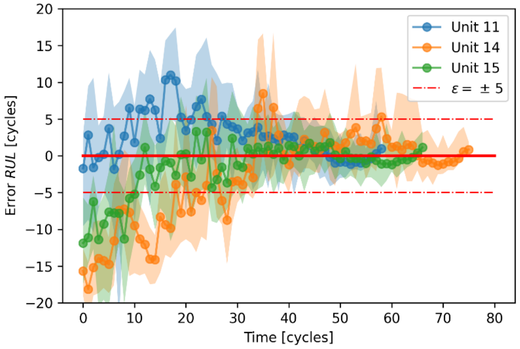
Hybrid - $\boldsymbol{X} = [\boldsymbol{w}, \widehat{\boldsymbol{X}}_s, \widehat{\boldsymbol{X}}_v, \widehat{\boldsymbol{\theta}}]$

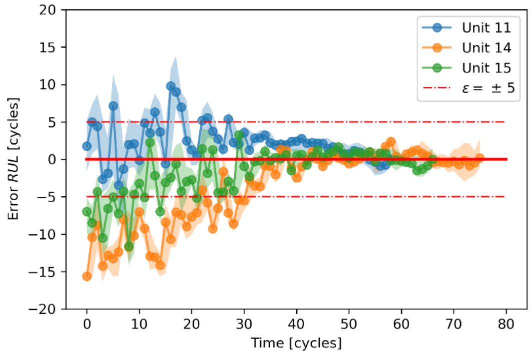
FFN





CNN





M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Detailed results RUL prediction

FFN

Metric	Data-Driven	Hybrid	Rel. Delta
RMSE [-]	7.89 ± 0.12	4.22 ± 0.10	-47%
$s \times 10^5$[-]	1.39 ± 0.04	0.44 ± 0.01	-68%

CNN

Metric	Data-Driven	Hybrid	Rel. Delta
RMSE [-]	4.95 ± 0.15	4.14 ± 0.09	-16%
$s \times 10^5$[-]	0.56 ± 0.03	0.44 ± 0.02	-21%

$$s = \sum_{j=1}^{m_*} exp(\alpha|\Delta^{(j)}|)$$

$$RMSE = \sqrt{\frac{1}{m_*}\sum_{j=1}^{m_*}\left(\Delta^{(j)}\right)^2}$$

M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, “Fusing Physics-based and Deep Learning Models for Prognostics,” Reliability Engineering & System Safety.

EOL prediction within the error bound of 5 cycles

CNN

$t_{EOL} - t_{\epsilon_y < 5}$ in [cycles]

unit	Data-Driven	Hybrid	Rel. Delta
11	11	31	195%
14	15	43	197%
15	24	37	54%
Fleet Avg.	16	37	127%

→ Prediction **horizon prolonged by 127%** on average (for some units ca. 200%)

M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, "Fusing Physics-based and Deep Learning Models for Prognostics," Reliability Engineering & System Safety.

Detailed results RUL prediction: smaller training dataset

CNN: trained on units 2,5,10,16,18,20

Metric	Data-Driven	Hybrid	Rel. Delta
RMSE [-]	4.95 ± 0.15	4.14 ± 0.09	-16%
$s \times 10^5$[-]	0.56 ± 0.03	0.44 ± 0.02	-21%

$$s = \sum_{j=1}^{m_*} exp(\alpha|\Delta^{(j)}|)$$

$$RMSE = \sqrt{\frac{1}{m_*}\sum_{j=1}^{m_*}(\Delta^{(j)})^2}$$

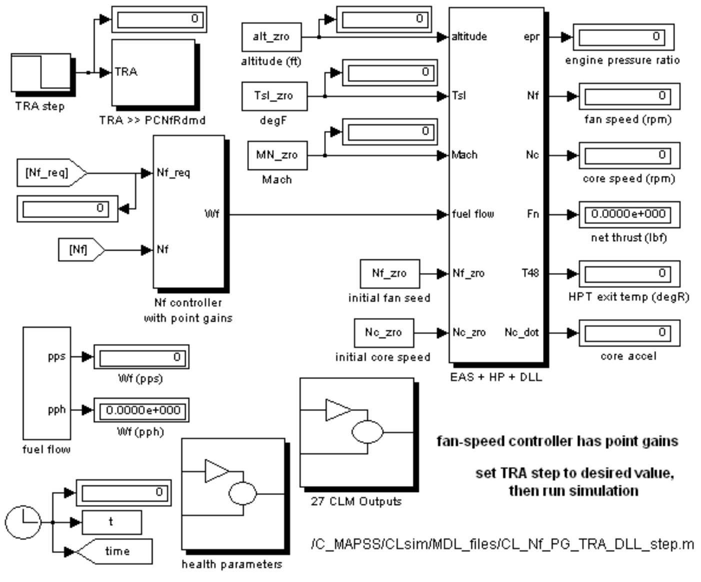
CNN: trained on units 16,18,20

Metric	Data-Driven	Hybrid	Rel. Delta
RMSE [-]	5.97 ± 0.37	4.22 ± 0.12	-29%
$s \times 10^5$[-]	0.61 ± 0.03	0.43 ± 0.02	-29%
rel. Delta RMSE [%]	17%	2%	
Rel. Delta s [%]	8%	-2%	

→ ca. 50% reduction of training data size

→ performance level maintained

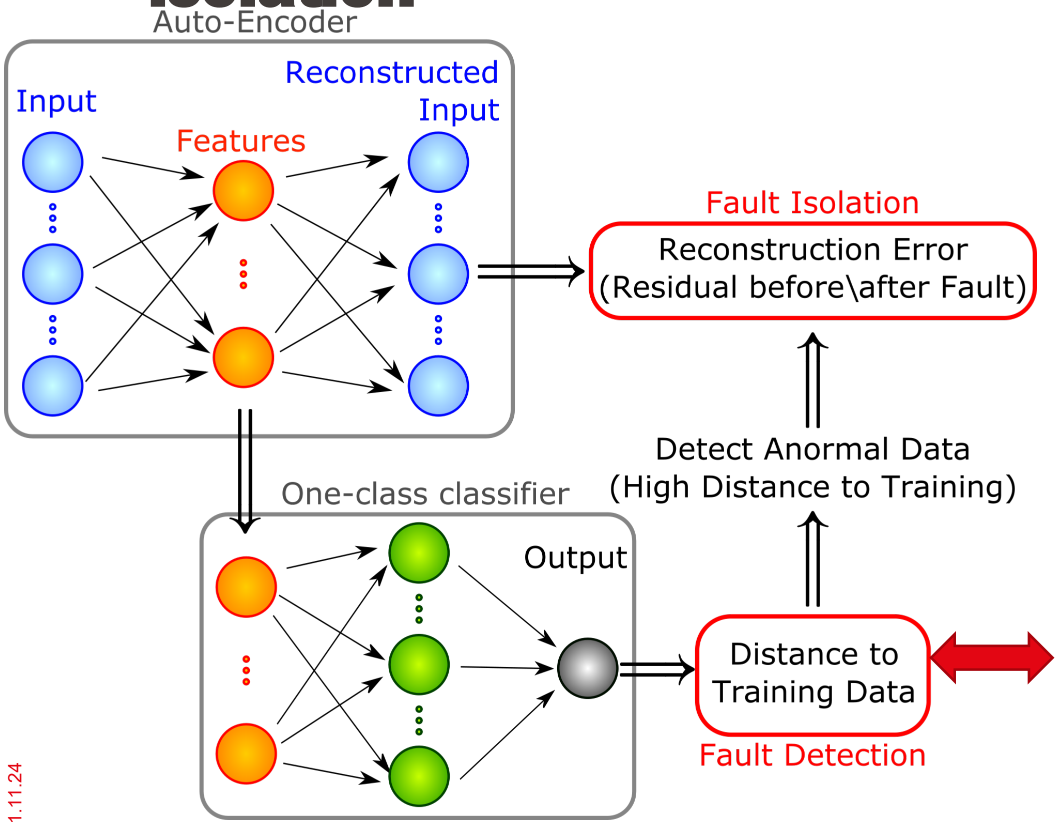
M. Arias Chao, C. S. Kulkarni, K. Goebel, and O. Fink, “Fusing Physics-based and Deep Learning Models for Prognostics,” Reliability Engineering & System Safety.

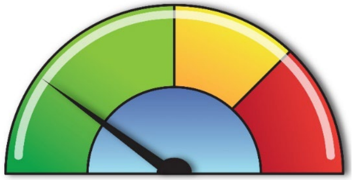


Hybrid Fault Detection and Diagnostics

How to improve not only the performance but also the interpretability and generalizability?

Analysing the reconstruction residuals for fault isolation

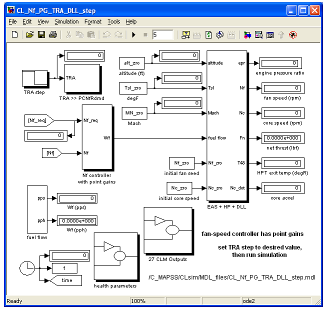


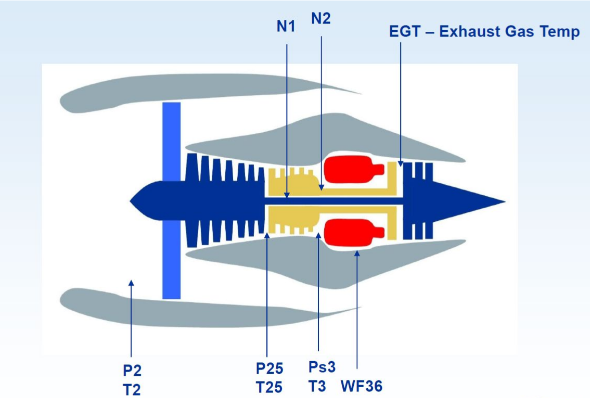


"Health Indicator"

Test Case: Turbofan Engine

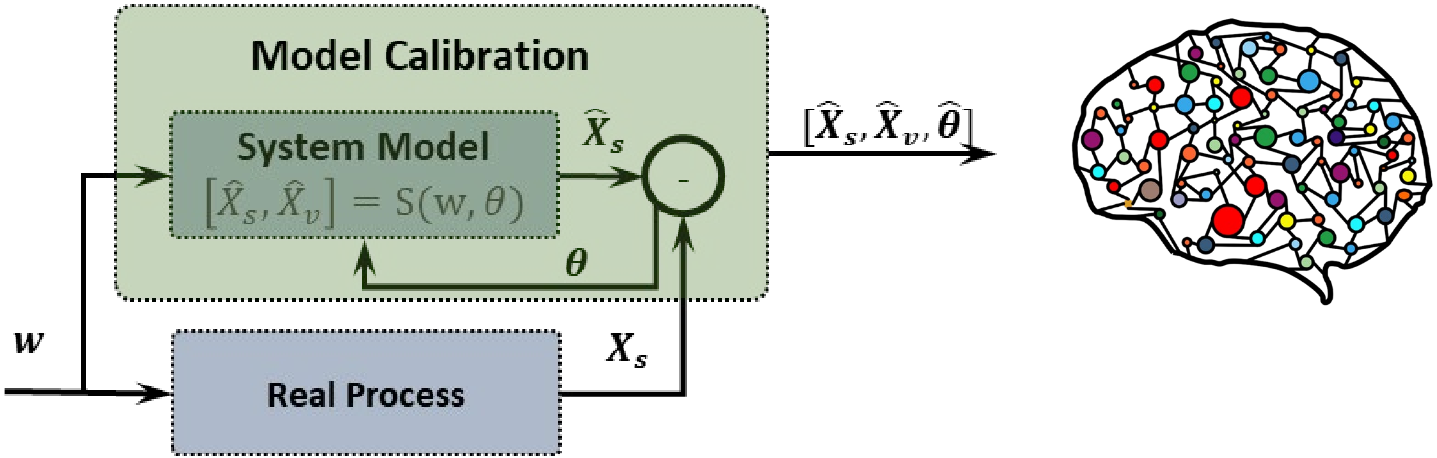
- **Simulation of a realistic large commercial turbofan engine**
- **24 flight cycles**
- **Each flight contains ca. 175 snapshots of recordings covering climb, cruise and descend conditions**
- **Real flight conditions from a commercial jet**
- **Fault: HPC Efficiency with different magnitudes**





Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

Calibration-based hybrid framework

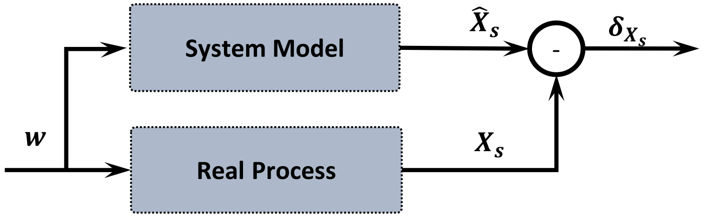


Arias Chao, M., Kulkarni, C., Goebel, K., & **Fink, O**. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

System Model - Degradation Features

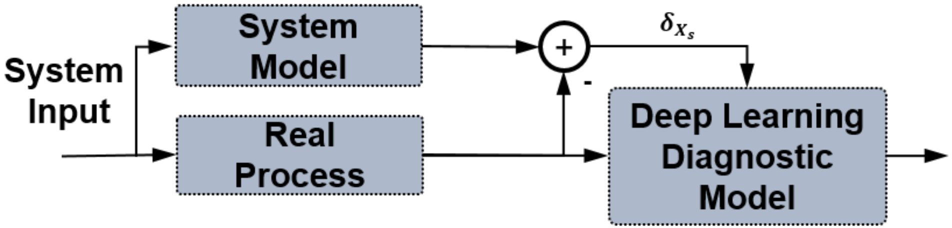
Degradation features provided by the system model:

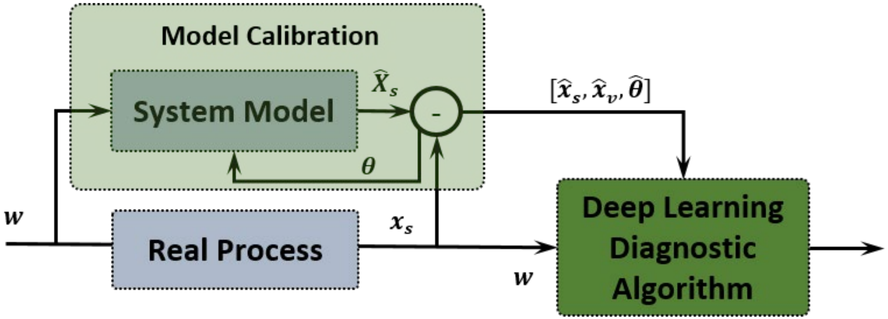
- Residual $\delta_{X_s} = \hat{X}_s - X_s$
  - Direct computation



Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

Two tested setups

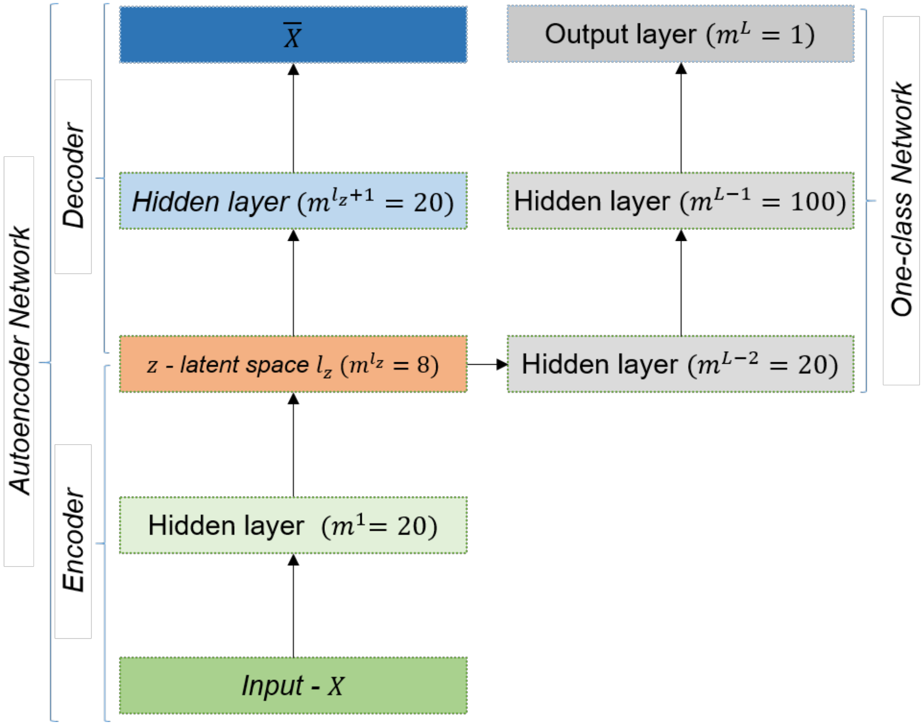




Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

EPFL

Applied network architecture



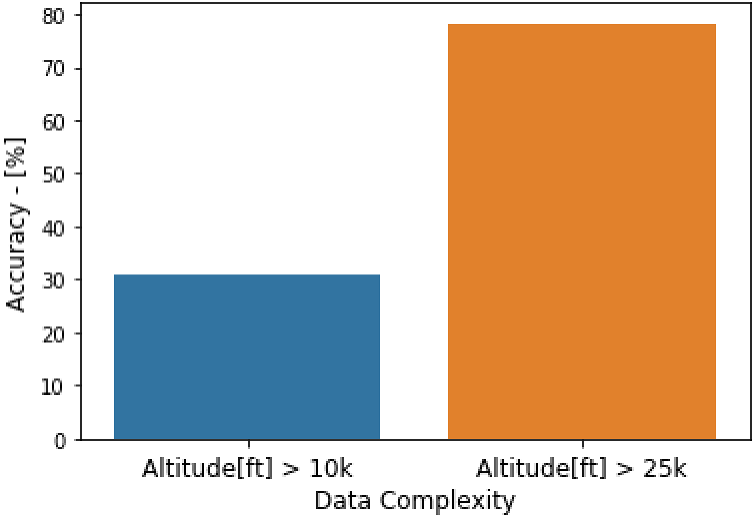
EPFL

Detection Accuracy in [%]

		AE	VAE	OC-SVM
Data-driven	$[W, X_S]$	24.5	12.9	10.0
Residual-based	$[W, X_S, \delta_{X_S}]$	98.7	99.3	79.5
Hybrid (calibration-based)	$[W, \widehat{X_S}, \widehat{X_V}, \hat{\theta}]$	100.0	97.5	96.2

Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

Influence of the data complexity (data-driven model)

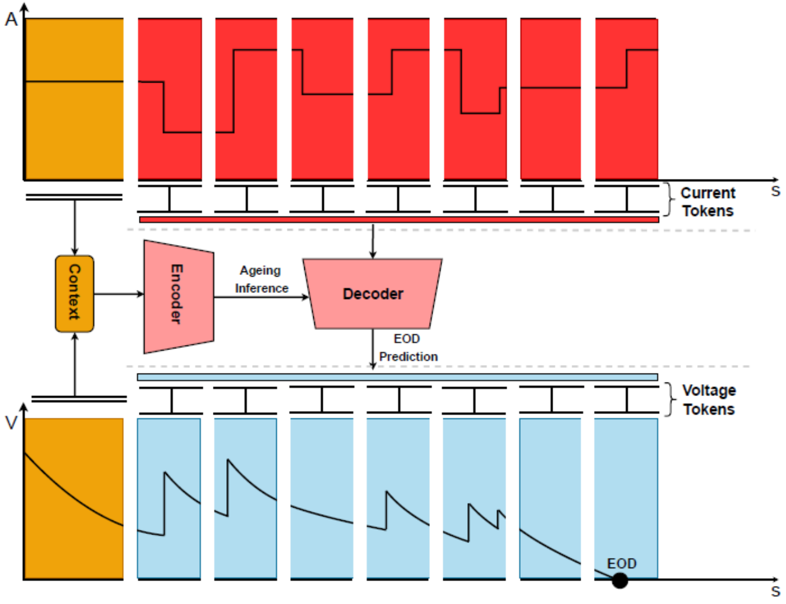


Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management

Fault isolation (with a high fault intesity)

		AE
Data-driven	$[W, X_S]$	Physical core speed Static pressure at HPC outlet
Residual-based	$[W, X_S, \delta_{X_S}]$	Delta physical core speed Delta total temp. at HPC outlet Delta total temp. at LPC outlet Delta total temp. at HPT outlet Delta total temp. at LPT outlet
Hybrid (calibration-based)	$[W, \widehat{X_S}, \widehat{X_V}, \hat{\theta}]$	HPC efficiency modifier

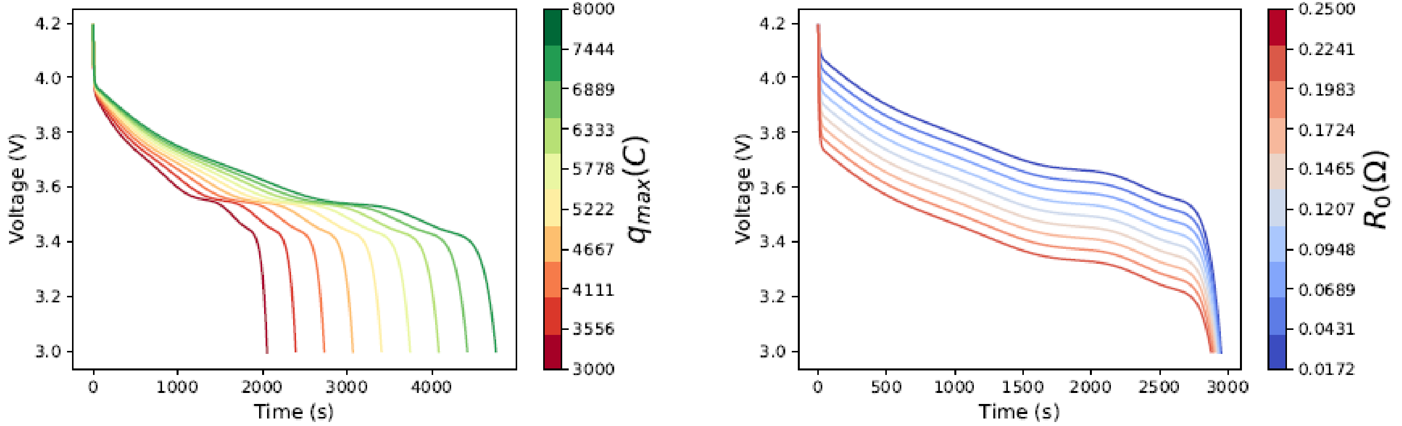
Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2019). Hybrid deep fault detection and isolation: Combining deep neural networks and system performance modelsInternational Journal of Prognostics and Health Management



Ageing-aware Battery Discharge Prediction

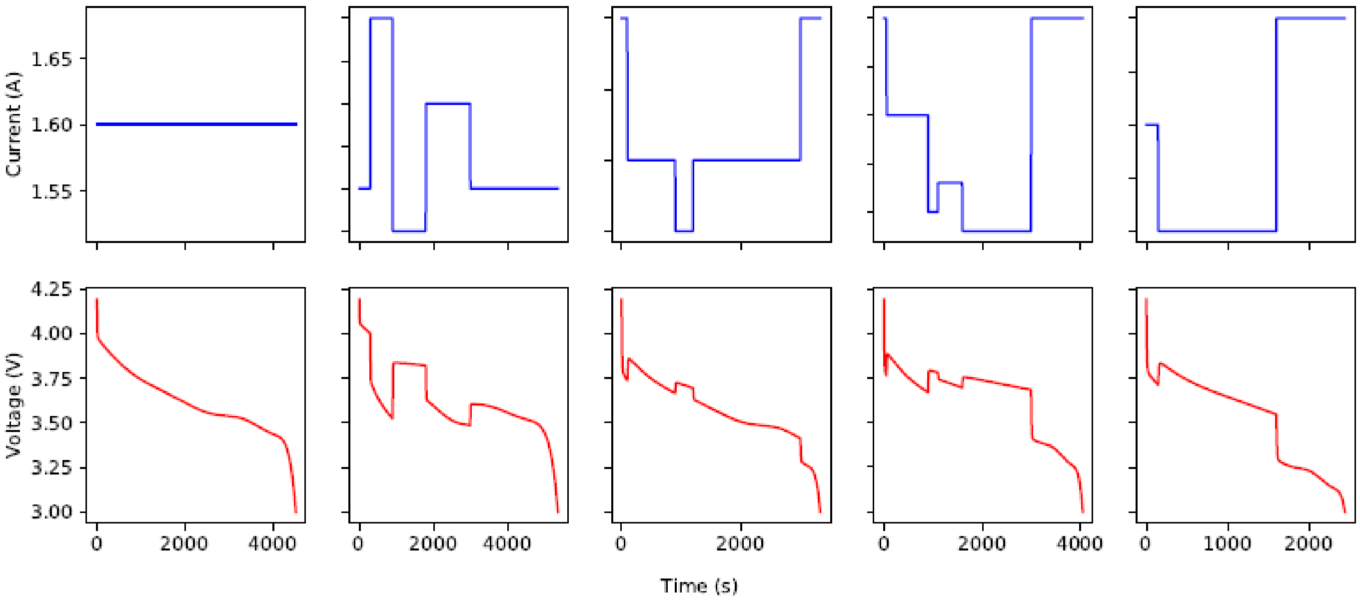
Degradation of Li-Ion batteries → importance of precise planning

Effect of varying the degradation parameters on the voltage discharge curve of a Li-ion battery



Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

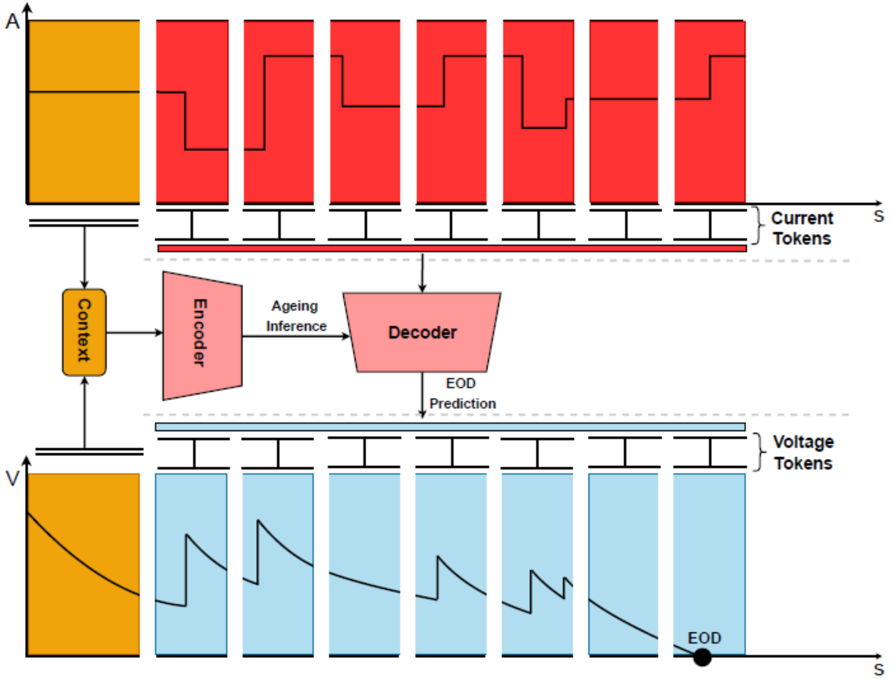
Discharge behaviors with respect to the different load profiles



Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

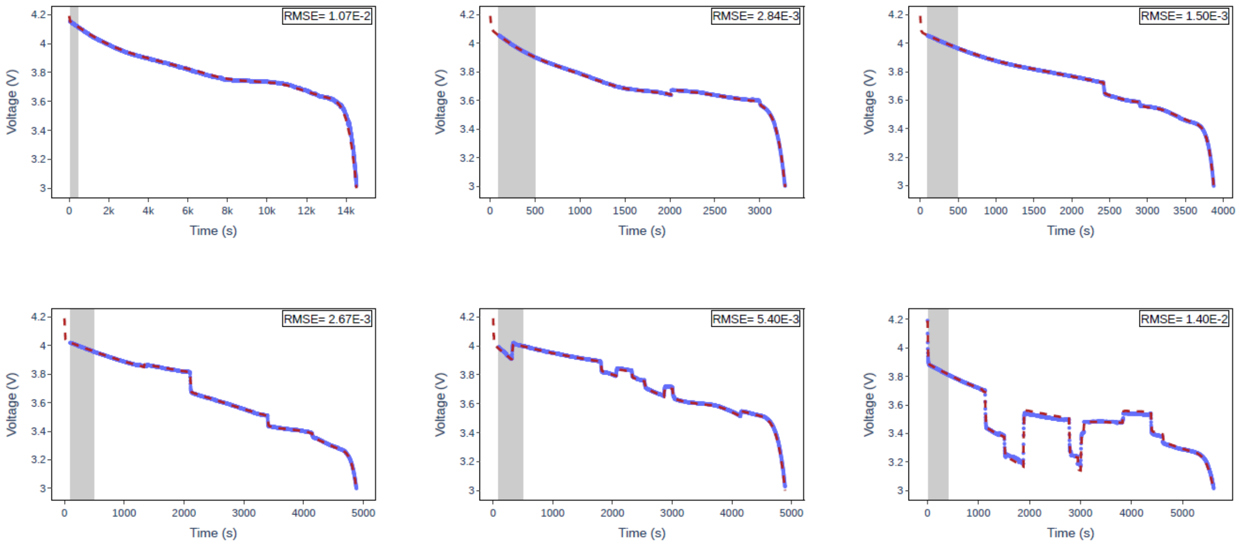
EPFL

Transformer-based long-term battery discharge prediction → Dynaformer



Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

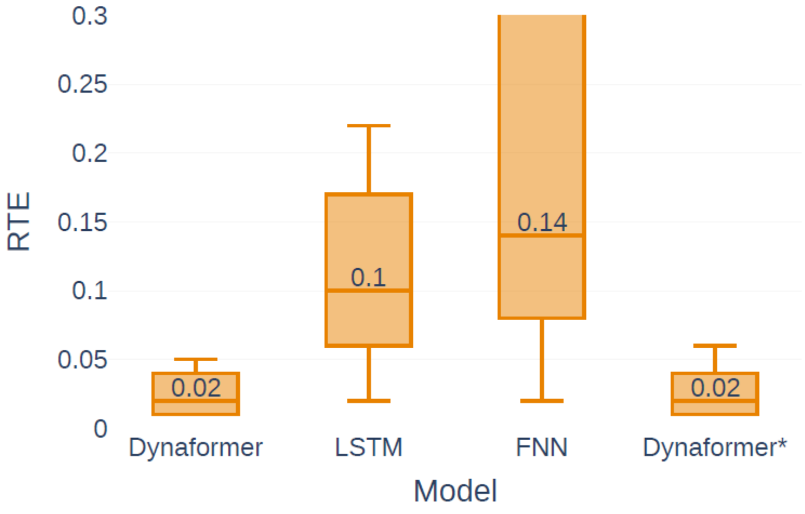
Selected results



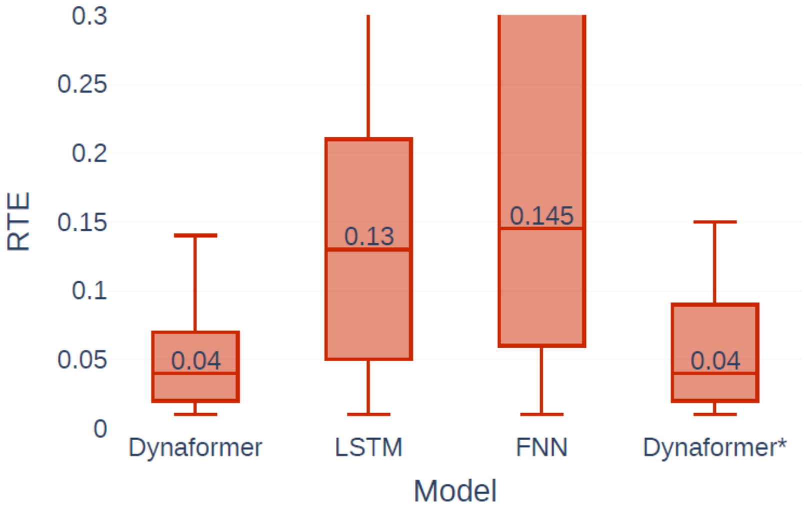
Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

Performance without fine-tuning

Interpolation



Extrapolation

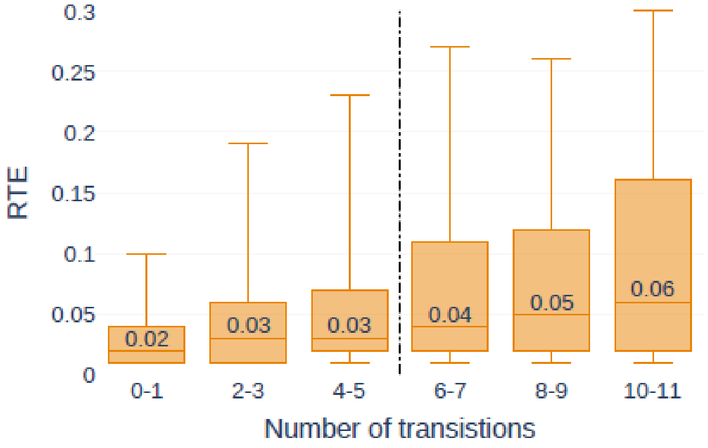


Dynaformer* → trained with variable current profiles

*RTE = relative temporal error

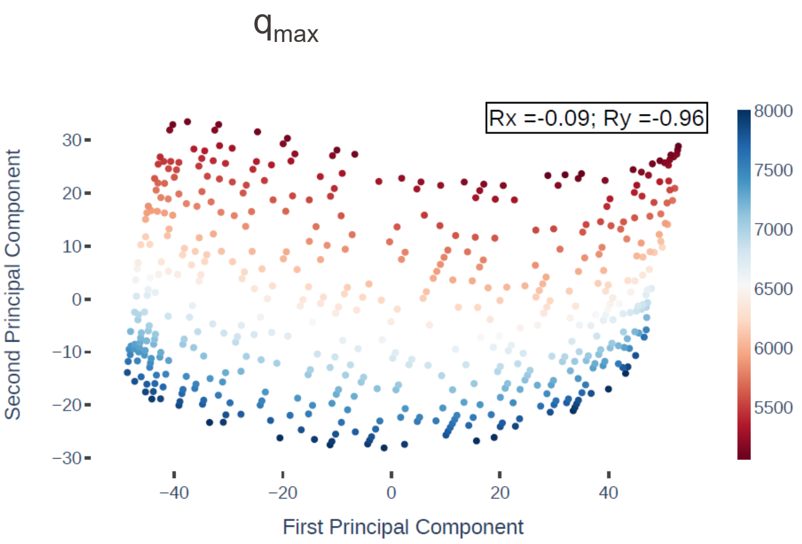
Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

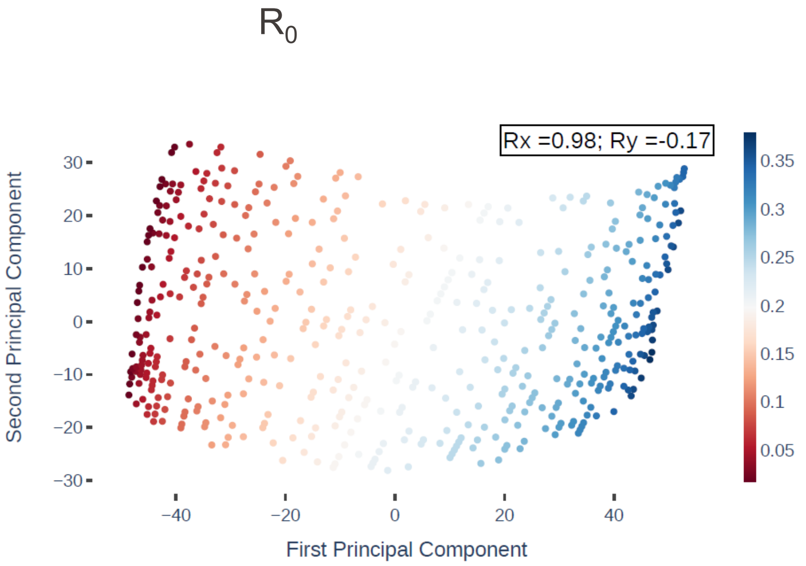
Performance dependent on the complexity of the the load profiles



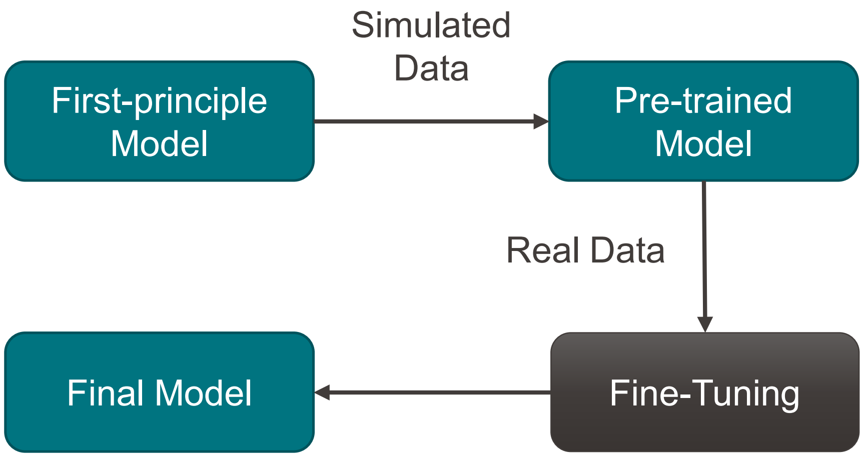
Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O.** (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review

Implicit learning of the degradation parameters in the latent space

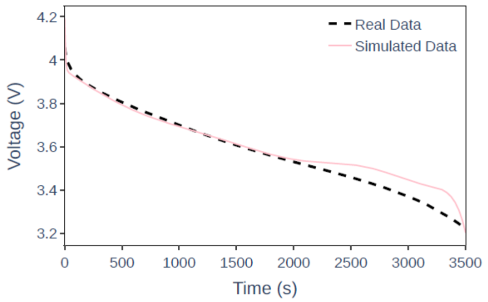


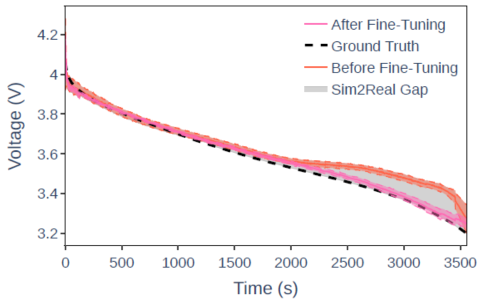


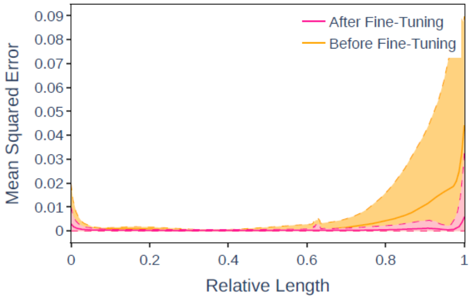
Transfer learning

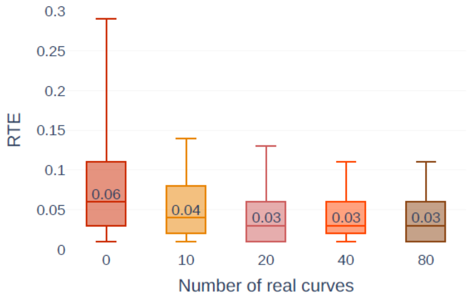


Fine-tuning on real data



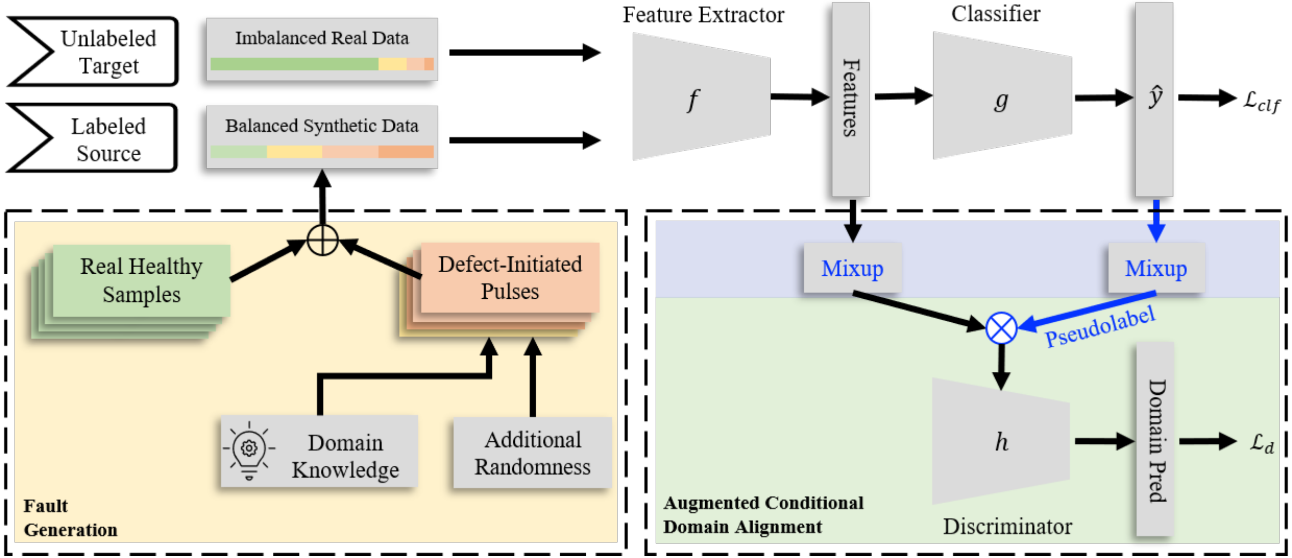






*RTE = relative temporal error

Biggio, L., Bendinelli, T., Kulkarni, C., & **Fink, O**. (2022). Dynaformer: A Deep Learning Model for Ageing-aware Battery Discharge Prediction, under review



Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis

Real Case study: Bearings of Wind Turbines

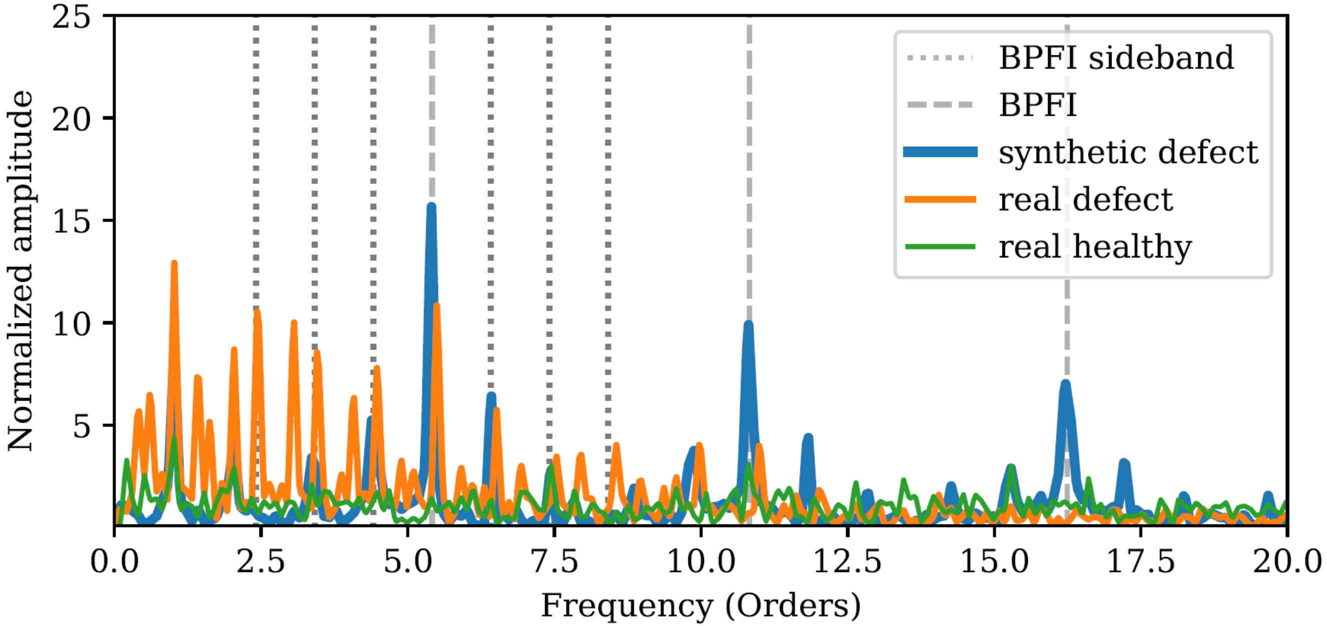
- Turbine Bearing Data
- Distinguish between 3 classes: healthy, inner race, outer race
- Without any faulty examples at training time

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Problem setup

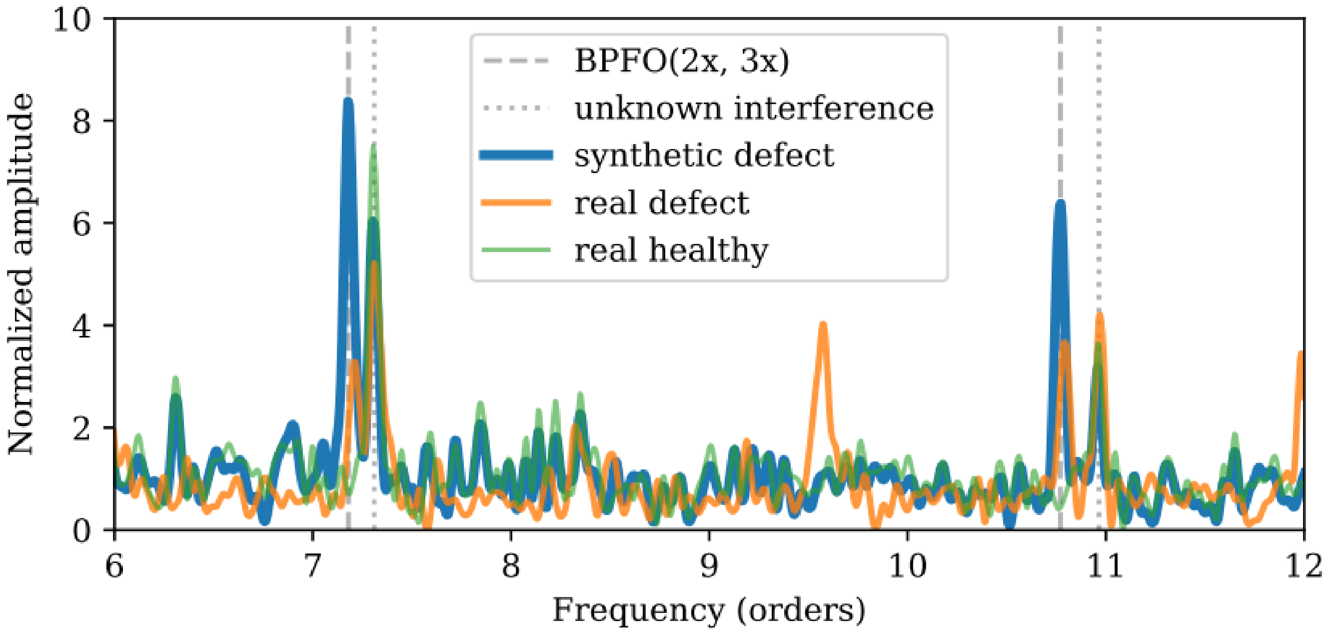
- Healthy data available
- No faulty data available at model development time
- Not just fault detection but also fault diagnostics required (distinguish between the different fault types)
- Developed models need to be tested on real faulty data
- Faulty data imbalanced
- Imbalance level unknown

Normalized full-wave rectified envelope spectrum of an inner ring defect example with corresponding defect ball pass frequency



Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

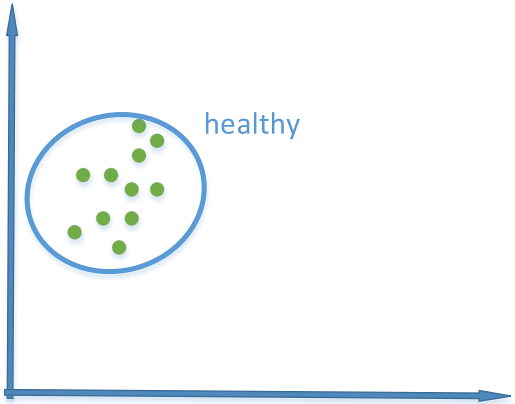
Normalized full-wave rectified envelope spectrum of an outer ring defect example with corresponding defect ball pass frequency



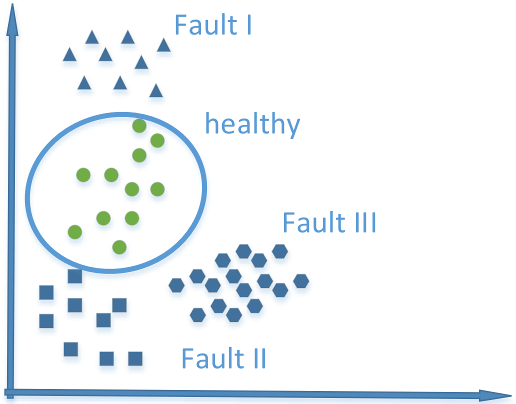
Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Synthetic2Real

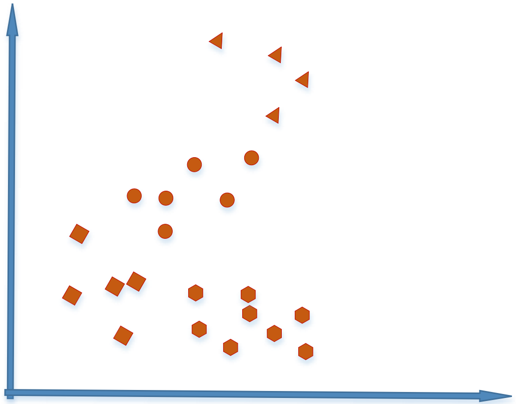
Source



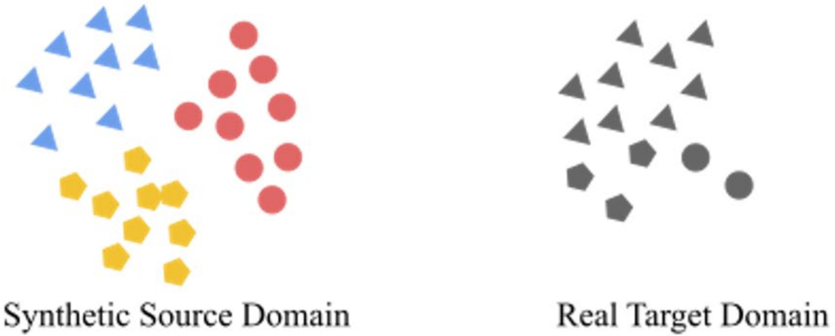
Source + synthetic faults



Target



Challenges of DA for imbalanced datasets



Generating Synthetic Data from Healthy Samples

One life-time recording of a bearing

Early part can be safely regarded as healthy

Take out this healthy part

Superposition of healthy signals + defect-initiated pulse-train signal + randomness

Inject synthetic faults to the healthy samples. Based on domain expertise.

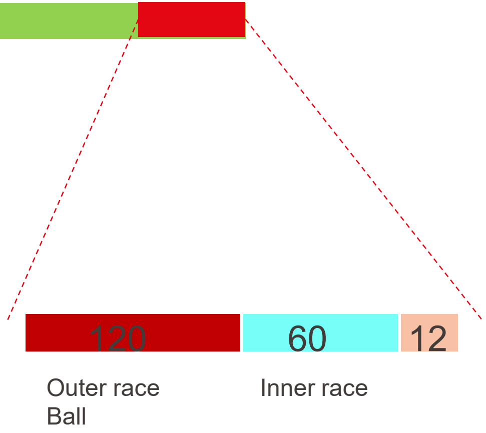
Outer race Inner race Ball

Combine them we get our synthetic data

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Naïve Synthetic2Real on imbalanced data: Toy example on CWRU

1. Four classes
2. Train only on synthetic data
3. Metric: Report average accuracy for all classes
   (Acc_healthy + Acc_fault1 + Acc_fault2 + Accu_fault3) /4

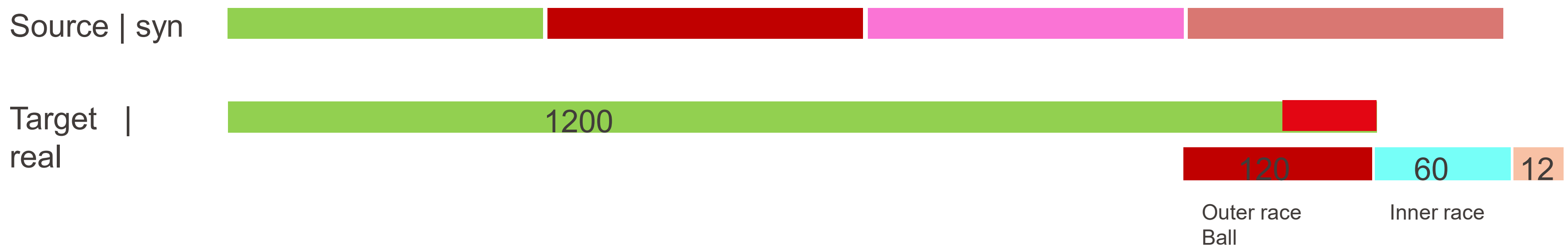


Source only Accuracy: 59.55%

The synthetic data indeed contains information that helps the classification.

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Adversarial Synthetic2Real on imbalanced data: Toy example on CWRU



How about adversarial domain adaptation ?

Method	Source-only	Adversarial
Accuracy	59.55%	72.57%

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

What is the issue ? Imbalanced data in target.

Quick verification: What if we make the number of samples equal for target ?

Target Before: exp1

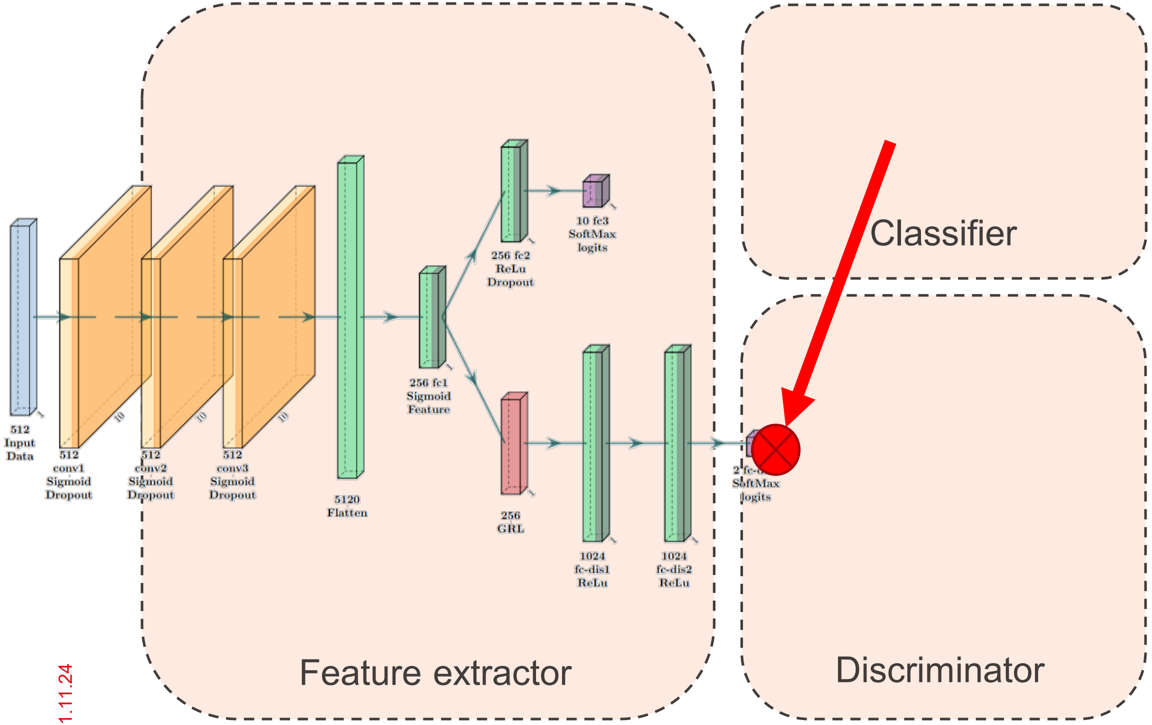
Modified Target: exp2

Method	Source-only	Exp1 Adversarial-imbalanced	Exp2 Adversarial-Balanced
Accuracy	59.55%	72.63%	82.59%

A big improvement on performance!
Simply by making it balanced!
But in reality we don't know the ground truth.

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Step 1: How to let the discriminator learn better for the rare faults.



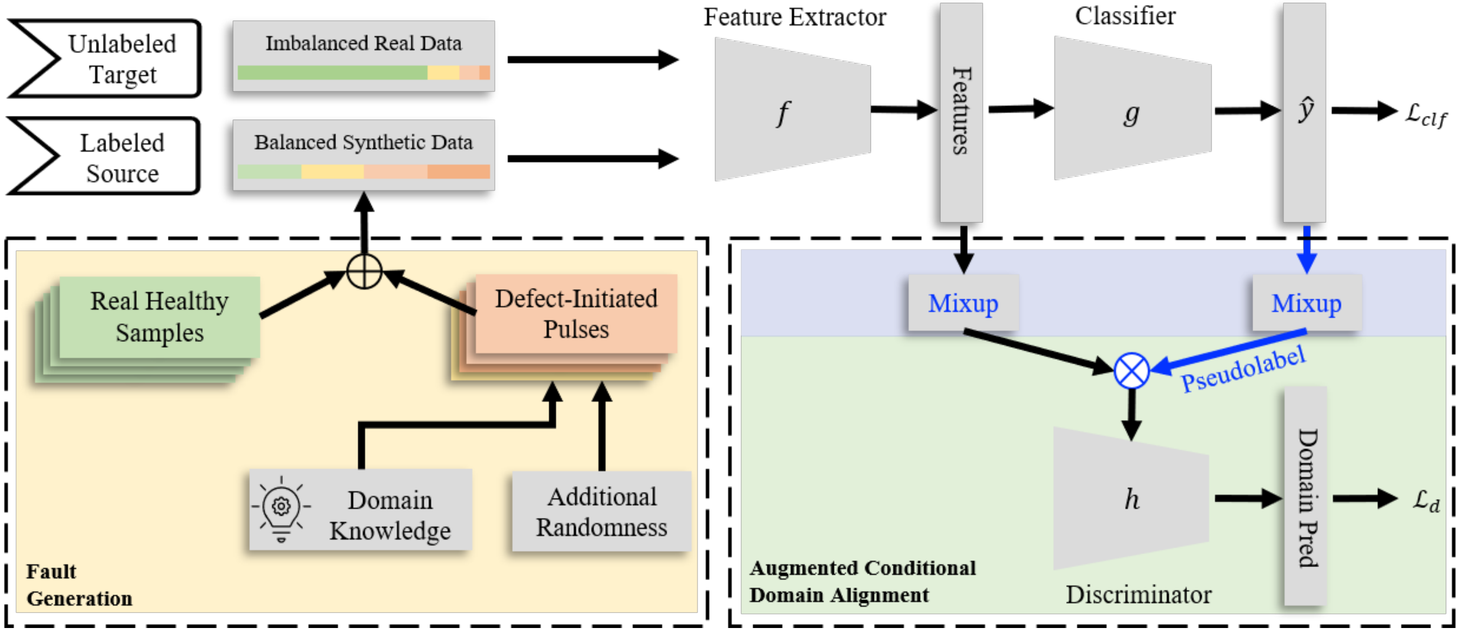
- Inspired by Conditional GAN and CDAN (conditional domain adaptation)
- Use pseudo label from the classifier and provide the class information to discriminator.

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Step 2: Provide more support for the target conditioned distribution

- Inspired by Mixup
- For features and pseudo-labels (in latent space not in input space)
- Performed separately for source and target
- This injects rare faults information to more samples in one batch

Proposed framework



Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Ablation Study

Method	Accuracy
Source only	59.55%
DANN	72.57%
Conditioned	75.38%
Conditioned + Mixup	82.29%

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Different Imbalance Levels on CWRU

	20%	15%	10%	5%	1%
Baseline	59.55	59.55	59.55	59.55	59.55
DANN	82.04	79.77	78.30	74.96	71.05
Proposed	83.49	83.04	83.32	83.30	82.34

x%: For every 100 healthy samples, x samples for the rare fault class

Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, IEEE Transactions on Instrumentation & Measurement

Real Case study

- Turbine Bearing Data from SKF
- 3 classes: Healthy, inner race, outer race
- Generated synthetic fault data from healthy
- There are much more outer race faults than inner race faults

Method	Accuracy
Source only	60.85
DANN	64.47
Conditional (proposed)	68.28
Conditional+mixup(proposed)	70.76

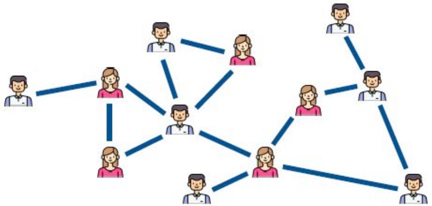
Wang, Q., C. Taal, O. Fink, Integrating Expert Knowledge with Domain Adaptation for Unsupervised Fault Diagnosis, under review

Graph Neural Networks

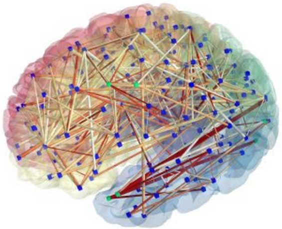
Graphs are everywhere

geographical network

traffic network



social network

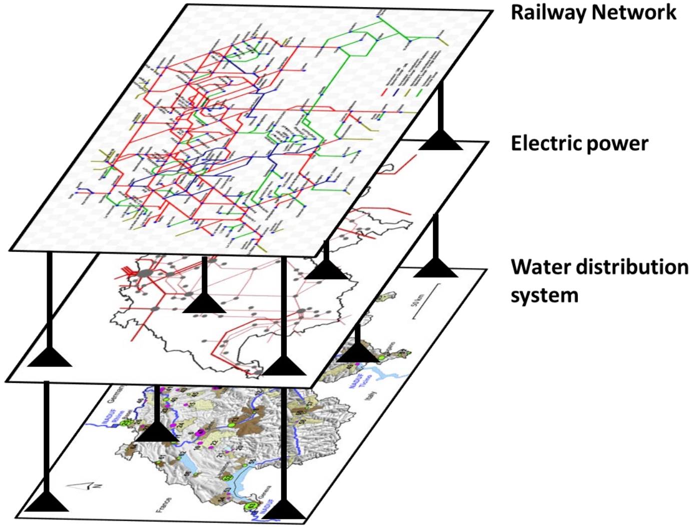


brain network

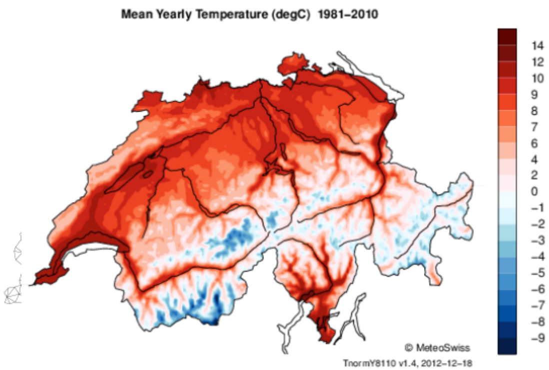
Graphs provide mathematical representation of networks

Source: P. Frossard

Infrastructure networks can be represented as graphs



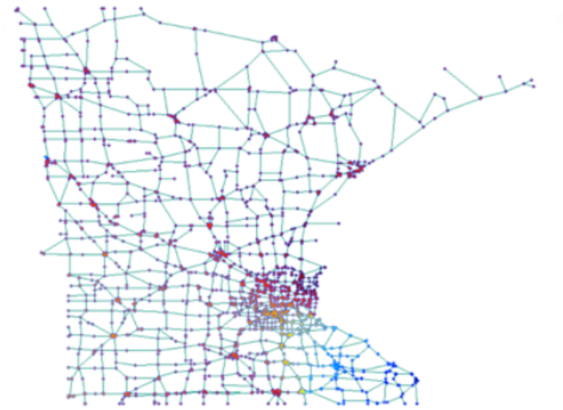
Examples of graph structured data



- nodes
  - geographical regions
- edges
  - geographical proximity between regions
- signal
  - temperature records in these regions

Source: P. Frossard

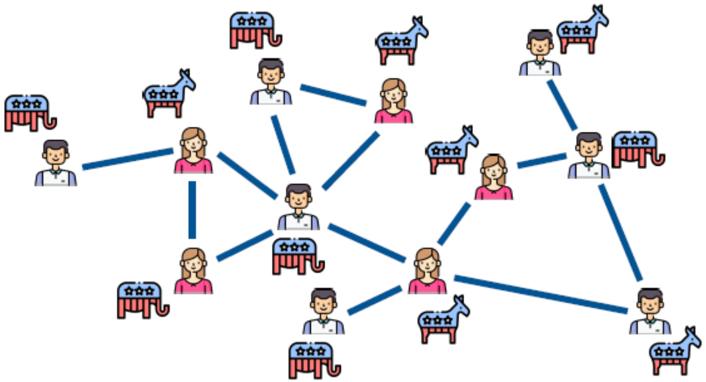
Examples of graph structured data



- nodes
  - road junctions
- edges
  - road connections
- signal
  - traffic congestion at junctions

Source: P. Frossard

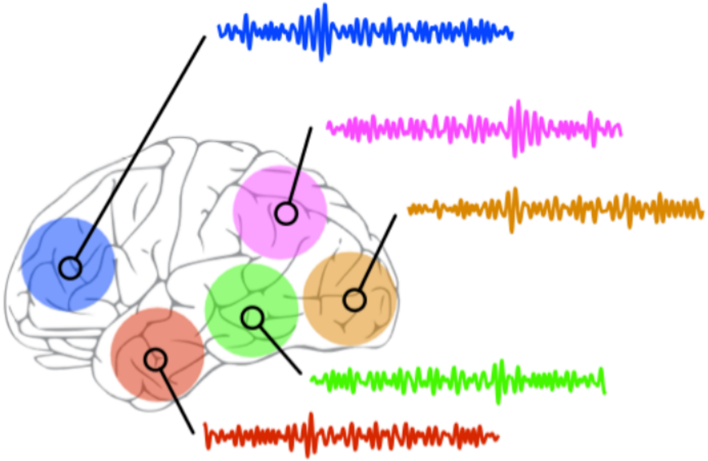
Examples of graph structured data



- nodes
  - individuals
- edges
  - friendship between individuals
- signal
  - political view

Source: P. Frossard

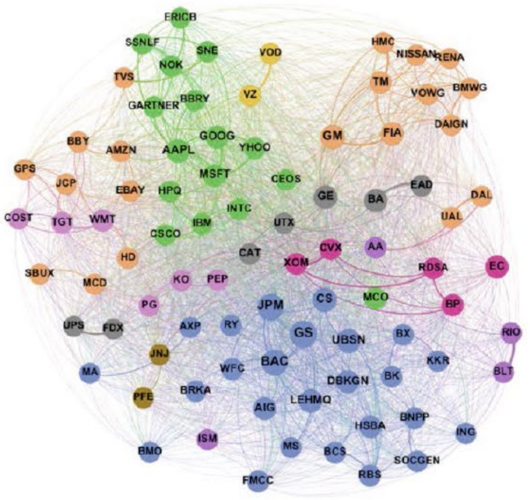
Examples of graph structured data



- nodes
  - brain regions
- edges
  - structural connectivity between brain regions
- signal
  - blood-oxygen-level-dependent (BOLD) time series

Source: P. Frossard

Examples of graph structured data



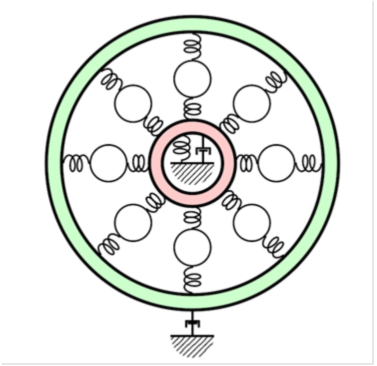
- nodes
  - companies
- edges
  - co-occurrence of companies in financial news
- signal
  - stock prices of these companies

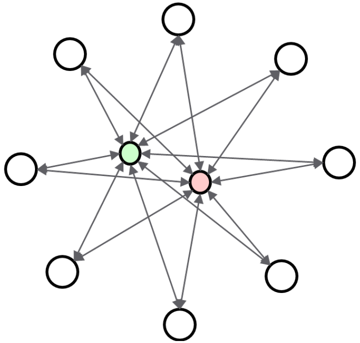
Source: P. Frossard

Examples of graph structured data

- Nodes: Sensors of a (bridge) sensor network)
- Edges: proximity of the sensors /similarity of the signals
- Signal: strain and / or accelerometer measurements

Bearing dynamics can be represented as graphs



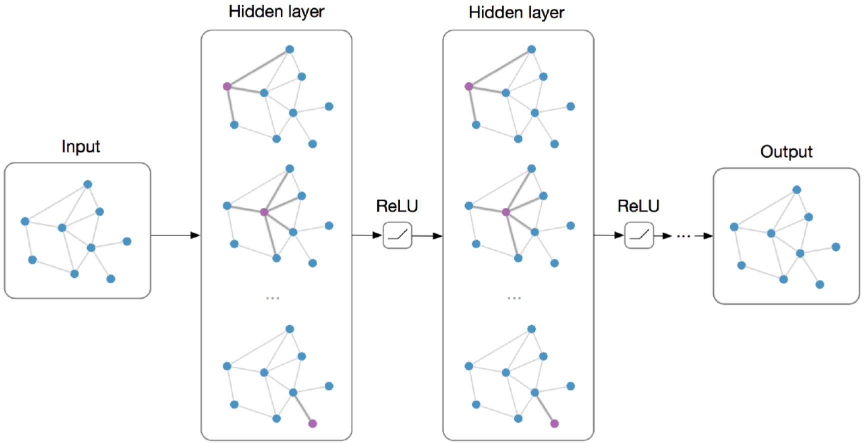


Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023

Task that can be solved with GNNs

- (supervised) graph level classification
- Node classification
- Link prediction

Graph Neural Networks



Main Idea: Pass massages between pairs of nodes and agglomerate

Alternative Interpretation: Pass massages between nodes to refine node (and possibly edge) representations

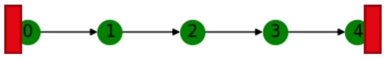
Source: T. Kipf

EPFL

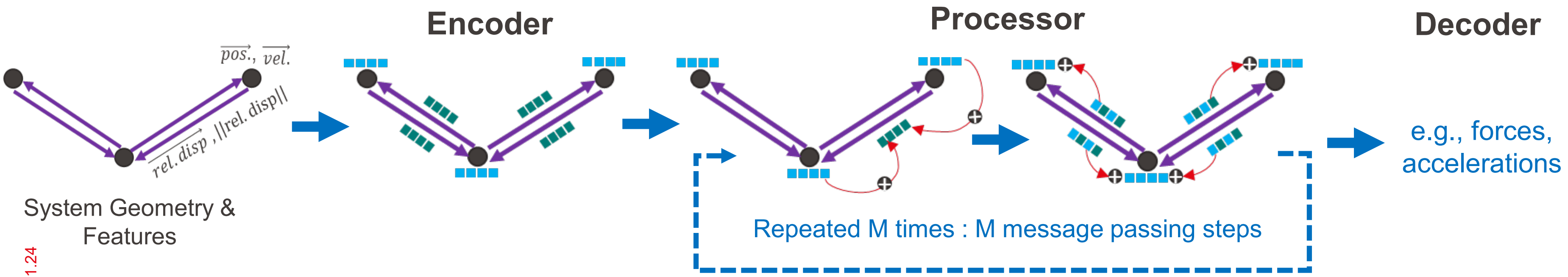
Introduction to message passing GNNs

Example : Spring Mass Systems

- Each node represents the mass
- Each edge represents the spring



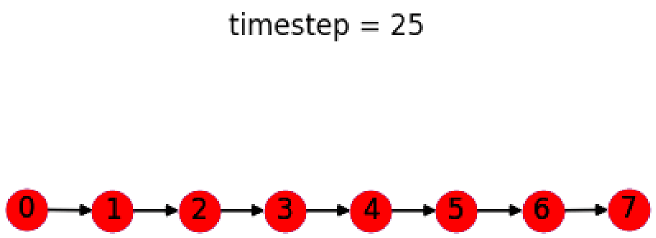
GNN : Message Passing on Graphs

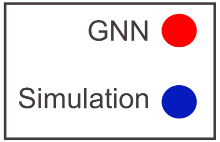


* Sanchez-Gonzalez et al. Learning to simulate complex physics with graph networks, ICML, 2020

Introduction to GNNs

Example : Spring Mass Systems learned dynamics by GNN



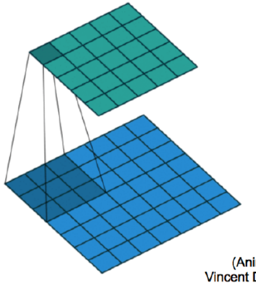


Train : 4, 5, 6, 7, 9, 10 masses
Test : 8 masses **(out of training)**

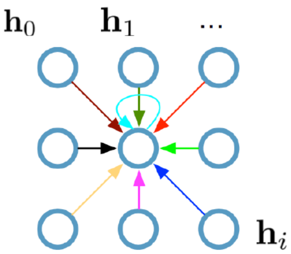
Source: V. Sharma

Recap: Convolutional Neural Networks (CNNs) on Grids

Single CNN layer with 3x3 filter:



(Animation by Vincent Dumoulin)



Update for a single pixel:

- Transform messages individually $\mathbf{W}_i \mathbf{h}_i$
- Add everything up $\sum_i \mathbf{W}_i \mathbf{h}_i$

$\mathbf{h}_i \in \mathbb{R}^F$ are (hidden layer) activations of a pixel/node

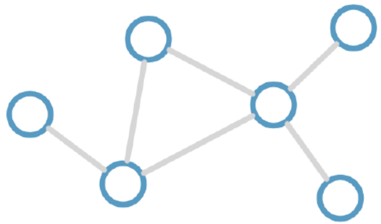
Full update:

$$\mathbf{h}_4^{(l+1)} = \sigma\left(\mathbf{W}_0^{(l)}\mathbf{h}_0^{(l)} + \mathbf{W}_1^{(l)}\mathbf{h}_1^{(l)} + \cdots + \mathbf{W}_8^{(l)}\mathbf{h}_8^{(l)}\right)$$

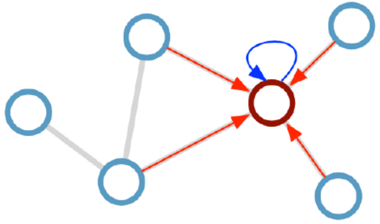
Source: T. Kipf

Graph Convolutional Networks (GCNs)

Consider this undirected graph:



Calculate update for node in red:



Desirable properties:

- Weight sharing over all locations
- Invariance to permutations
- Linear complexity O(E)
- Applicable both in transductive and inductive settings

Update rule:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{h}_i^{(l)}\mathbf{W}_0^{(l)} + \sum_{j \in \mathcal{N}_i} \frac{1}{c_{ij}} \mathbf{h}_j^{(l)}\mathbf{W}_1^{(l)}\right)$$

Scalability: subsample messages [Hamilton et al., NIPS 2017]

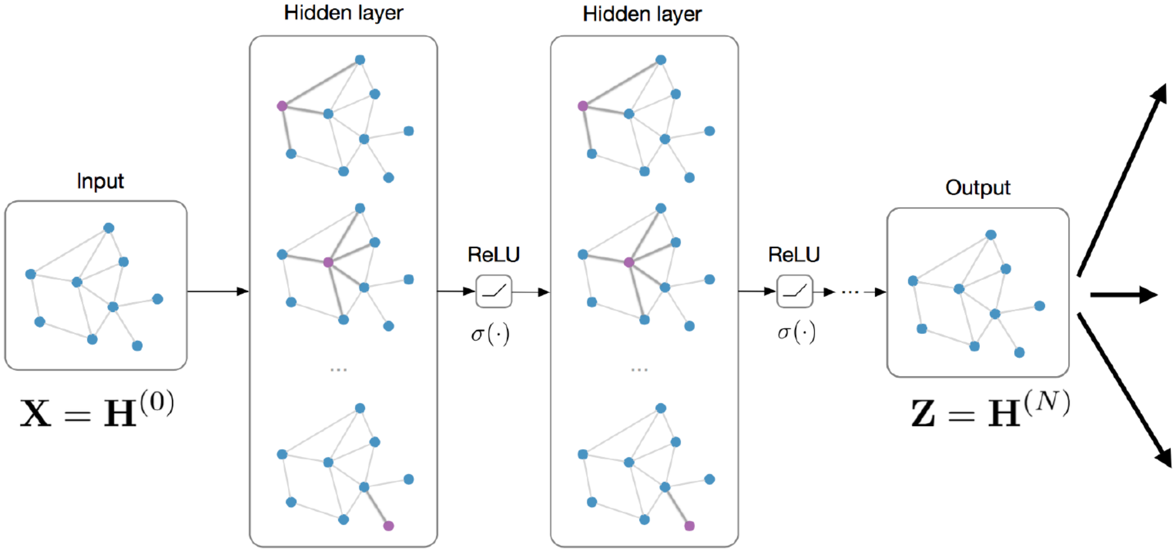
$\mathcal{N}_i$: neighbor indices

c_{ij} : norm. constant (fixed/trainable)

Source: T. Kipf

Classification and Link Prediction with GNNs / GCNs

Input: Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times E}$, preprocessed adjacency matrix $\hat{\mathbf{A}}$



$\mathbf{X} = \mathbf{H}^{(0)}$ $\sigma(\cdot)$ $\sigma(\cdot)$ $\mathbf{Z} = \mathbf{H}^{(N)}$

$$\mathbf{H}^{(l+1)} = \sigma\left(\hat{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right)$$

Node classification:

$$\mathrm{softmax}(\mathbf{z_n})$$

e.g. Kipf & Welling (ICLR 2017)

Graph classification:

$$\mathrm{softmax}(\textstyle\sum_n \mathbf{z_n})$$

e.g. Duvenaud et al. (NIPS 2015)

Link prediction:

$$p(A_{ij}) = \sigma(\mathbf{z_i}^{\mathbf{T}}\mathbf{z_j})$$

Kipf & Welling (NIPS BDL 2016)
"Graph Auto-Encoders"

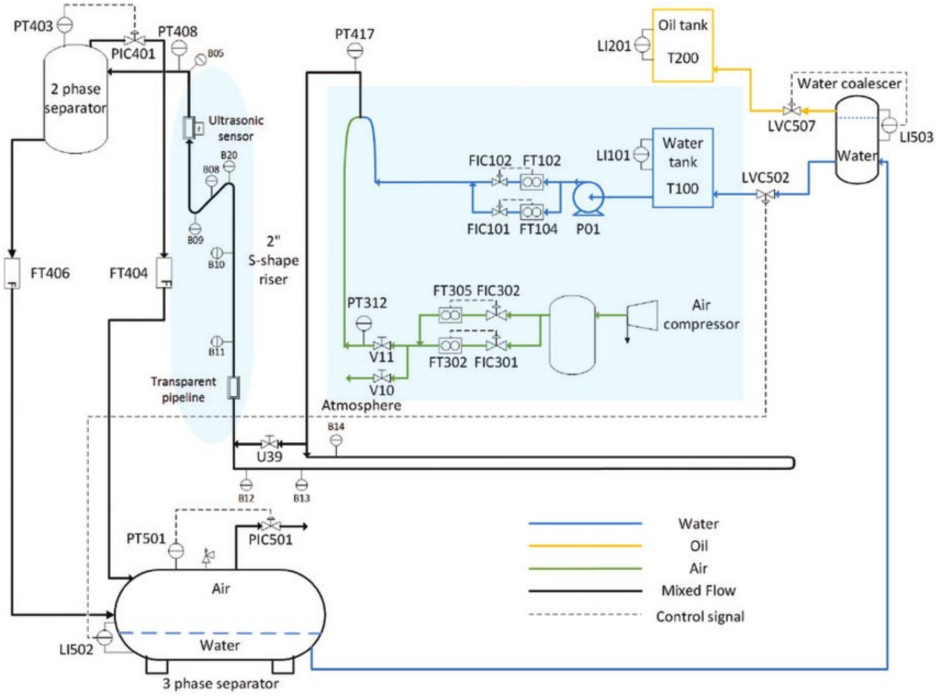
Source: T. Kipf

Example:
Industrial IoT systems

Changing / evolving relationships between time series

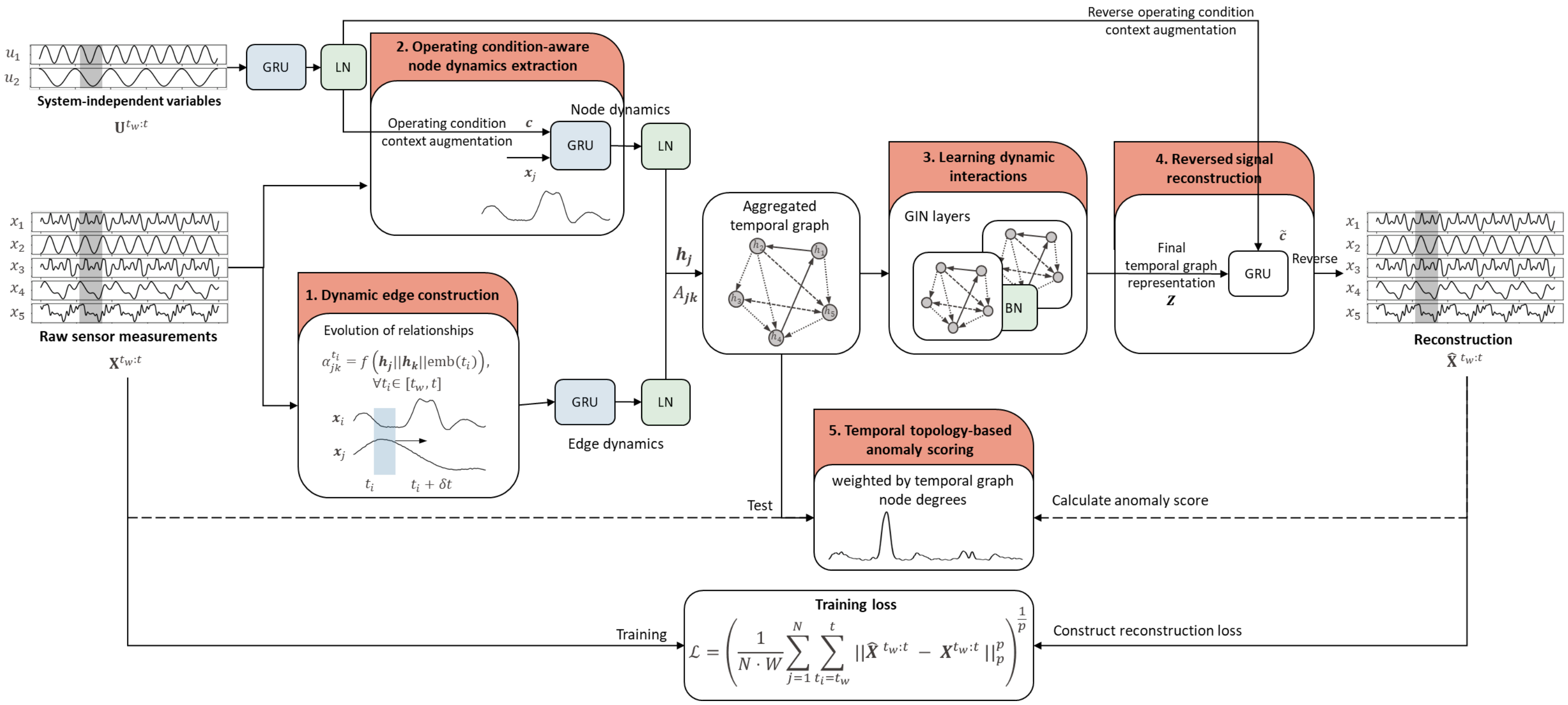
- Usually, graph topology between time series either given or derived once → static graph
- Changing environmental or operating conditions
- Impact of degradation
- Process optimization
- Maintenance
- …

Anomaly detection in multi-flow facility



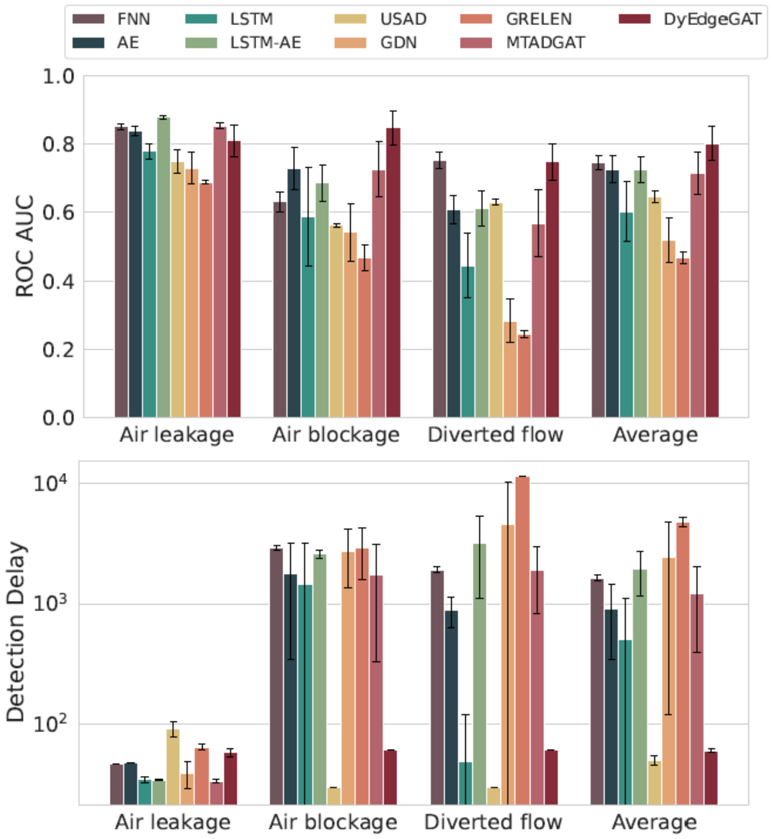
Source: Gedda, R., Beilina, L. and Tan, R., 2023. Change Point Detection for Process Data Analytics Applied to a Multiphase Flow Facility. *CMES-Computer Modeling in Engineering & Sciences*, *134*(3).

DyEdgeGAT: Dynamic Edge via Graph Attention for Early Fault Detection in IIoT Systems



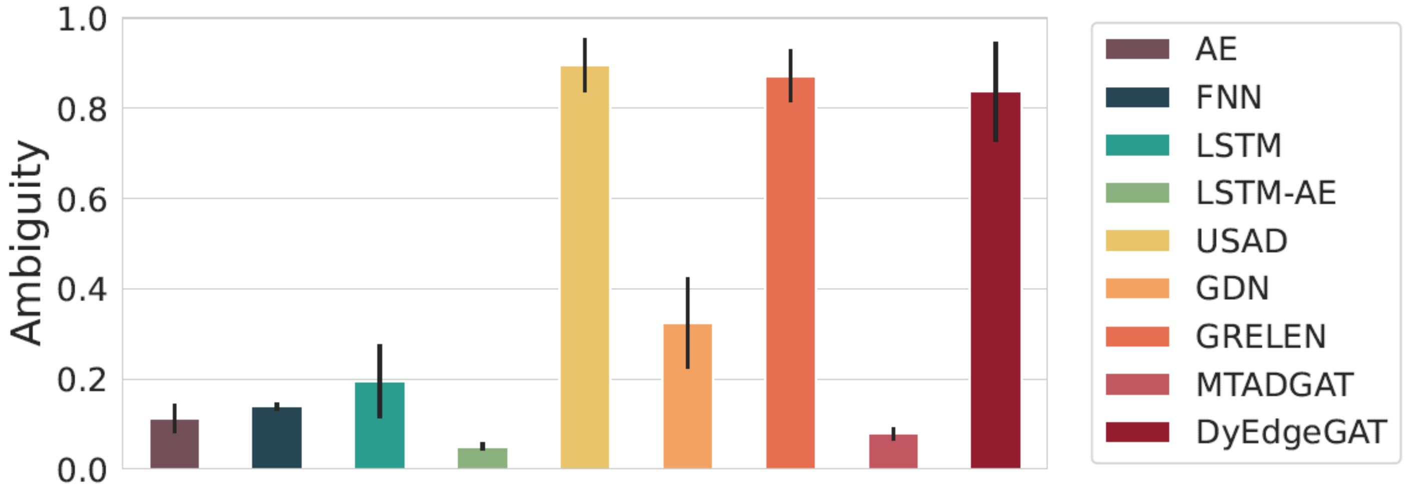
Zhao, M., & Fink, O. (2023). DyEdgeGAT: Dynamic Edge via Graph Attention for Early Fault Detection in IIoT Systems, IEEE Internet of Things Journal

Dynamic Edge via Graph Attention (DyEdgeGAT)



Zhao, M., & Fink, O. (2023). DyEdgeGAT: Dynamic Edge via Graph Attention for Early Fault Detection in IIoT Systems, IEEE Internet of Things Journal

Novel operating conditions



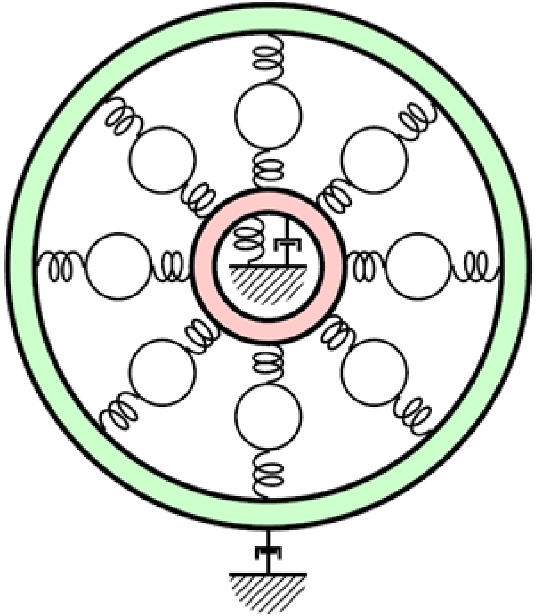
$$\text{Ambiguity} = 1 - 2 \cdot |\text{AUC} - 0.5|$$

Ambiguity quantifies the model’s inability to differentiate between normal operations and novel conditions

Zhao, M., & Fink, O. (2023). DyEdgeGAT: Dynamic Edge via Graph Attention for Early Fault Detection in IIoT Systems, IEEE Internet of Things Journal

Example: Modelling bearing as a graph

Bearing Dynamic Model

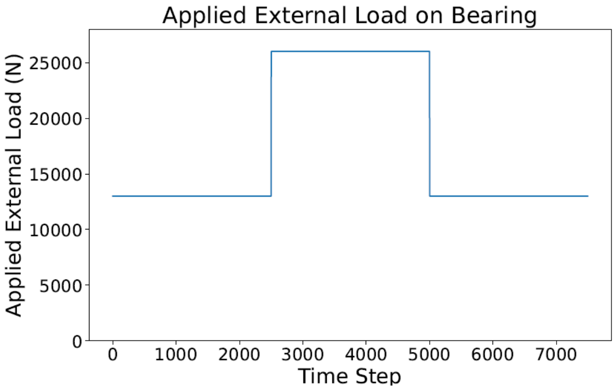


- 2D dynamic lumped bearing model, (based on P. K. Gupta, 1979)
- Differential equations of motion
- Lundberg & Palmgren model for Hertzian contact
- N209 CRB bearing (line contacts)
- Assumptions: rigid rings, zero-mass rolling elements

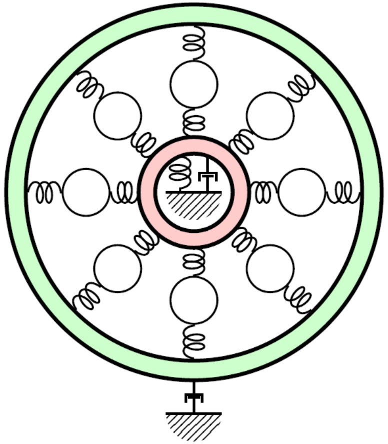
based on simple model but theory also applies to complex models!

Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023

Bearing Dynamic Model: Training set



→



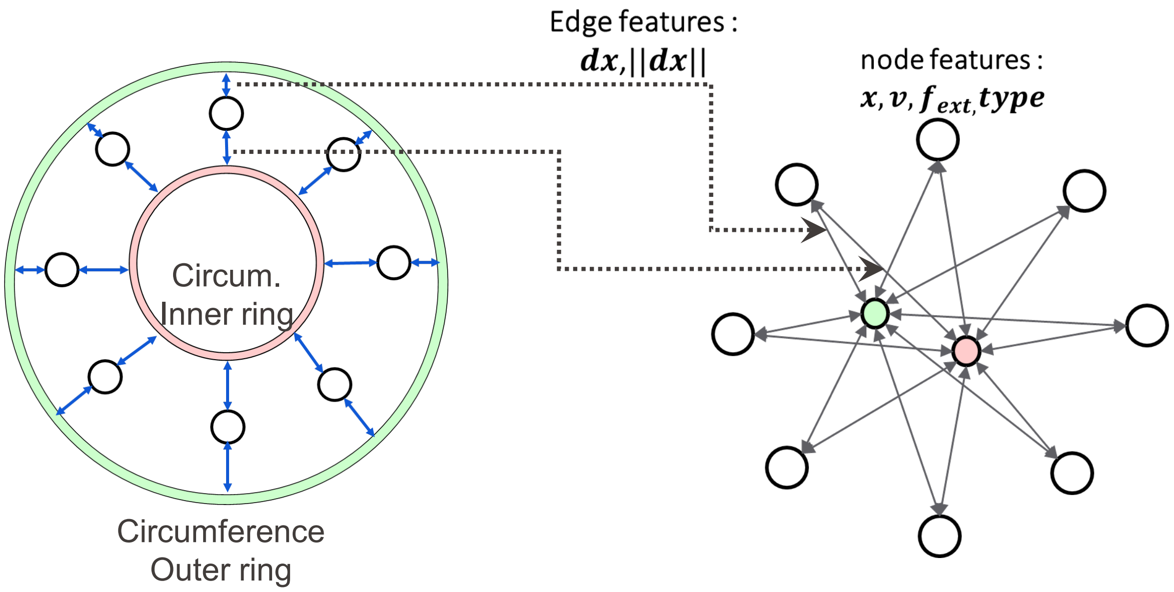
→

Positions $\vec{x}_i$
Velocities $\vec{v}_i$
Internal loads $\vec{f}_i$

- Load changes to generate dynamic step responses
- [1-20] kN OR loads, [13, 14, 15, 16] rollers
- (Rotation is not included yet)

Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023

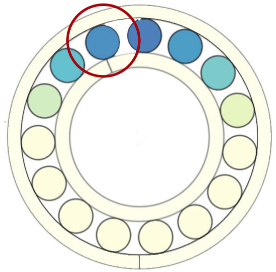
GNN Inputs and Outputs

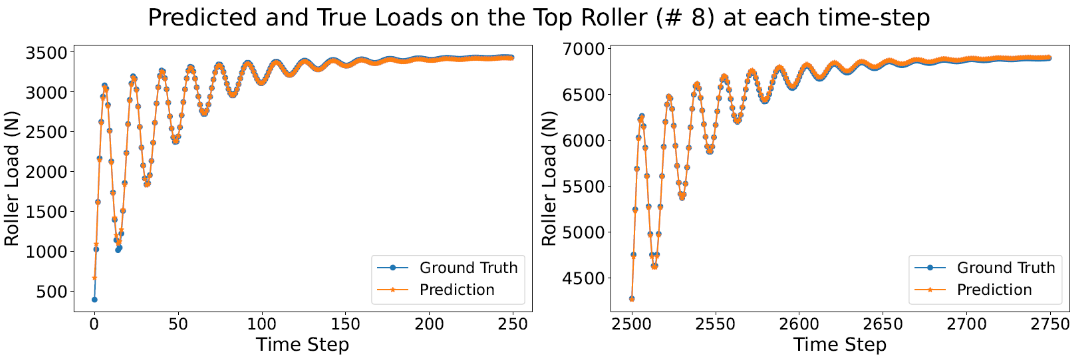


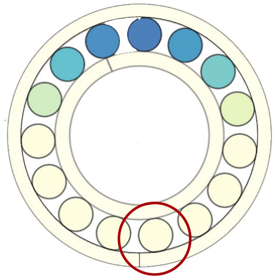
Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023

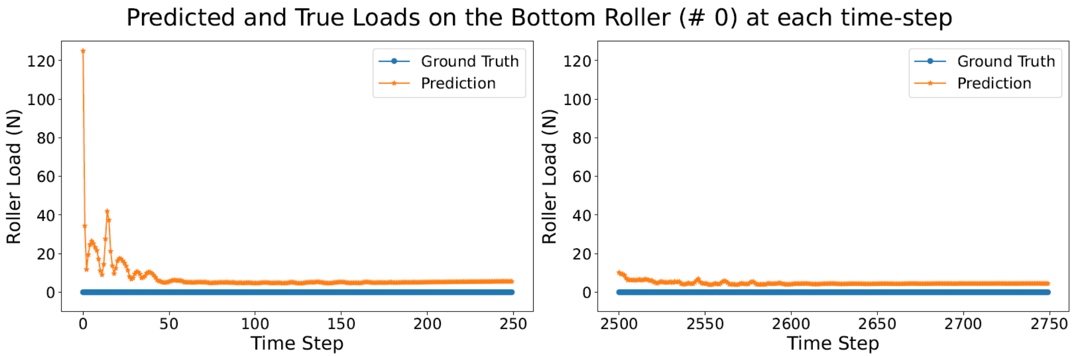
Results: Generalizability

Training : SKF N209 CRB (13, 14, 16 rollers)
Testing: 15 Rollers, 13kN





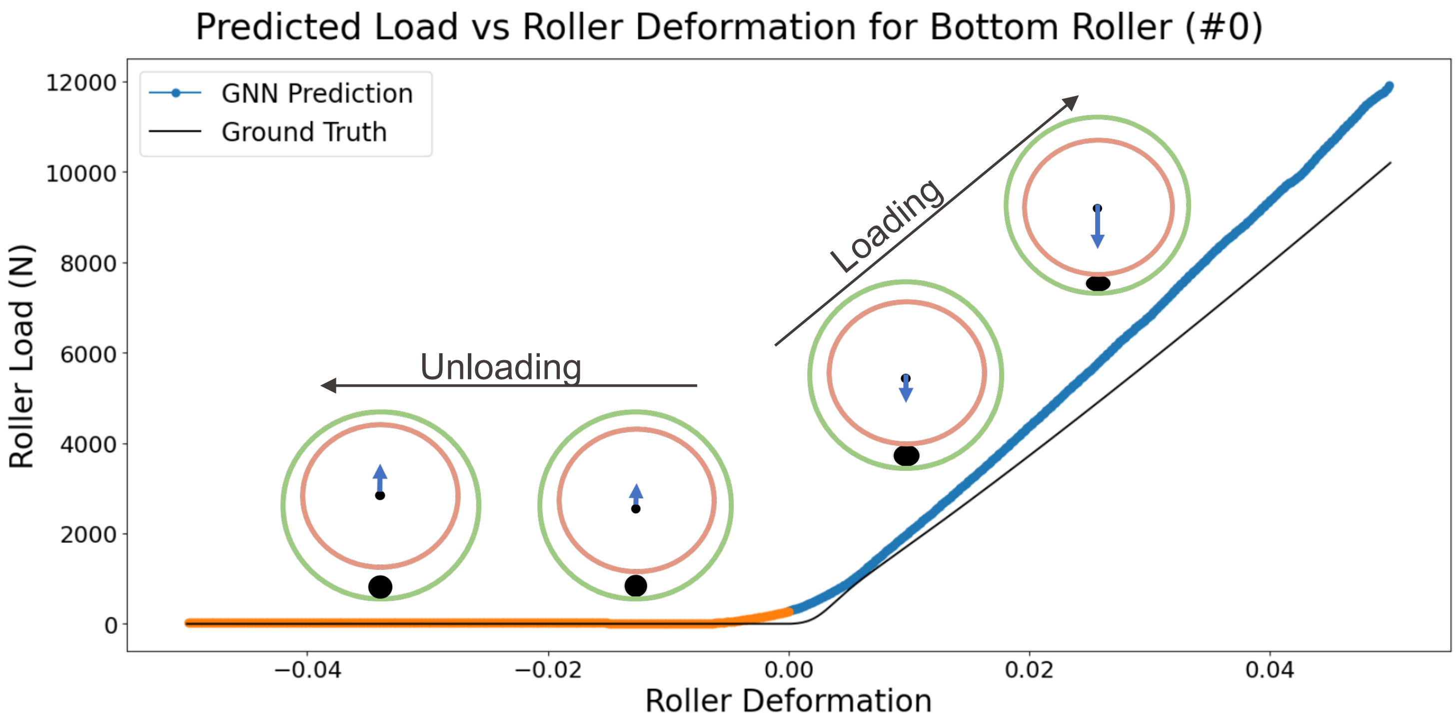




Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023

EPFL

Results: Interpretability



- **GNN can infer the unloading behavior without direct training. (Conventional Networks cannot infer this behavior)**

Sharma, V., Ravesloot, J., Taal, C. and Fink, O., 2023. Graph Neural Networks for Dynamic Modeling of Roller Bearing, PHM Society conference 2023