PASTEUR LABS

simulation.science

Perspectives & Practices for Reliable Scientific ML in Industrial Process Systems: Dynamic Surrogacy & Causal Discovery

Jordan Jalving PhD

SciML @ ICERM, March 2024

SciML for Industrial Control & Process Systems

Chemical Processes

2

Power Systems

3

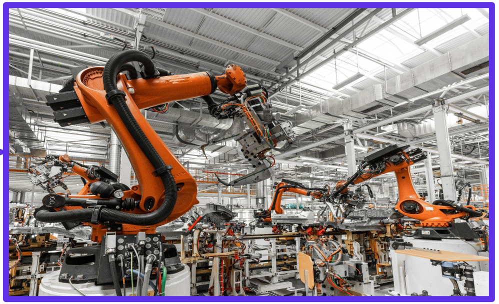
Control Room

1

Building Energy Systems

4

Manufacturing Processes



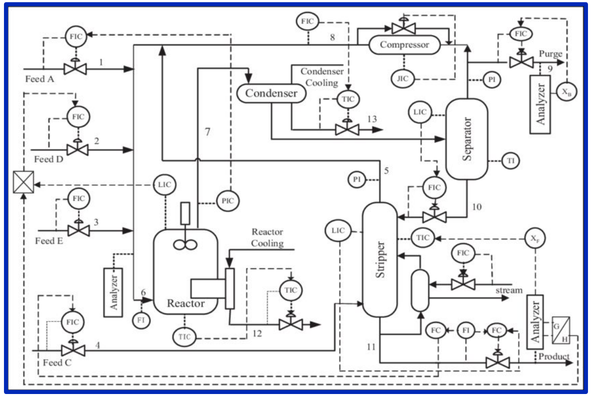
5

1: https://www.cnet.com/pictures/inside-a-power-grid-control-room-photos
2: https://www.freepik.com/free-photo/chemical-plants-vibrant-red-tanks-stand-out-against-tangle-pipes-steel-structures_134882051.htm#query=refinery&position=7&from_view=search&track=sph&uuid=121fbd90-8914-47b5-8a87-2a13348f63e6
3: https://www.freepik.com/free-photo/electricity-high-voltage-pole-sky_1242947.htm#query=energy%20utility&position=13&from_view=search&track=ais&uuid=e1b60ea8-989c-4372-a21e-11d3009b086b
4: https://www.freepik.com/free-photo/aerial-shot-photovoltaics-eco-friendly-warehouse-3d-illustration_147664728.htm#fromView=search&page=1&position=13&uuid=b74b66df-921a-4895-9393-9fc08c9248c5
5: https://www.freepik.com/free-photo/photo-automobile-production-line-welding-car-body-modern-car-assembly-plant-auto-industry_26150191.htm#fromView=search&page=1&position=2&uuid=c6d6555a-fbab-4cc5-bfcb-d01daff100ea

SciML for Industrial Control & Process Systems

Industrial Control & Process Systems (ICPS) are characterized by…

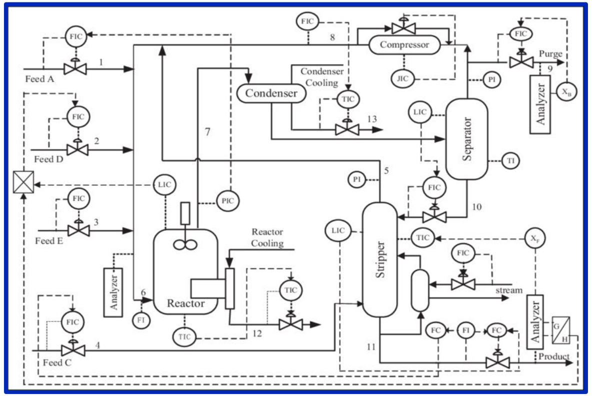
Physical Equipment Topology



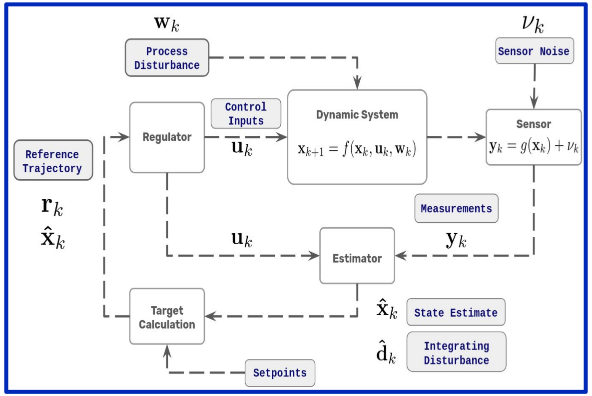
SciML for Industrial Control & Process Systems

Industrial Control & Process Systems (ICPS) are characterized by…

Physical Equipment Topology



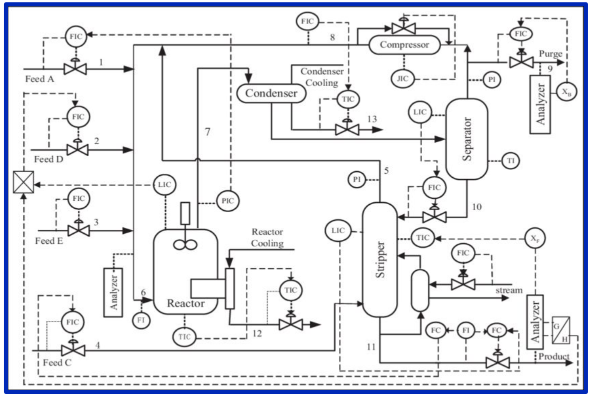
Control System / SCADA



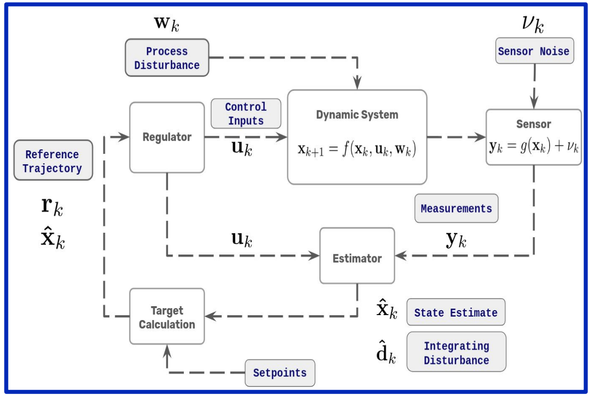
SciML for Industrial Control & Process Systems

Industrial Control & Process Systems (ICPS) are characterized by…

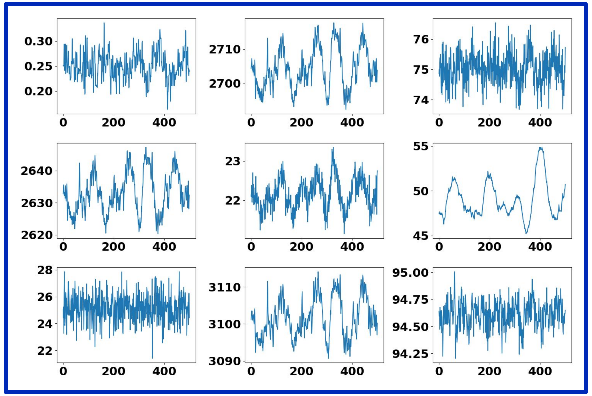
Physical Equipment Topology



Control System / SCADA



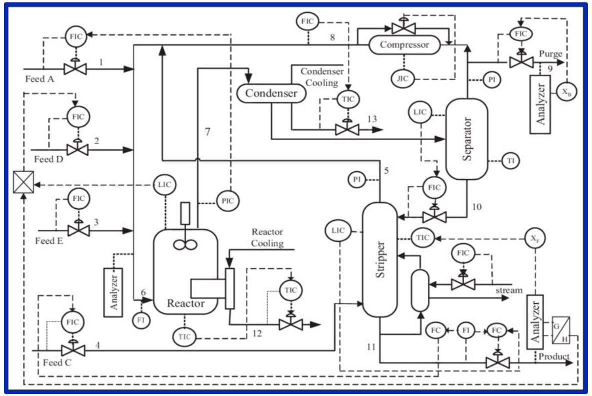
Noisy & Nonlinear Dynamics



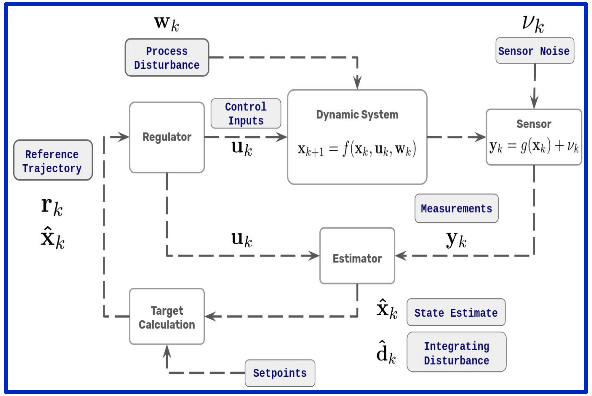
SciML for Industrial Control & Process Systems

Industrial Control & Process Systems (ICPS) are characterized by…

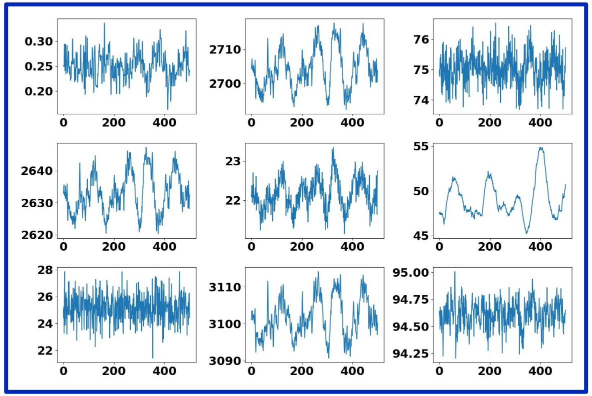
Physical Equipment Topology



Control System / SCADA



Noisy & Nonlinear Dynamics



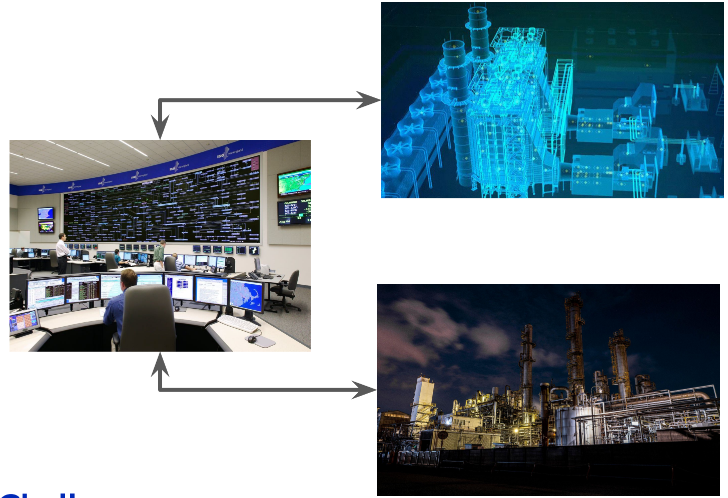
Operational Challenges: Monitoring, Maintenance, Security, Control, Training Personnel, System Diagnosis…

Where can SciML advances complement and accelerate the ICPS paradigm?

Primary ICPS Themes

Virtual Experimentation (Part 1)

Online systems benefit from online models



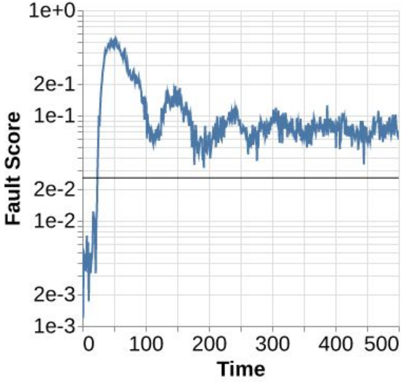
Challenges:

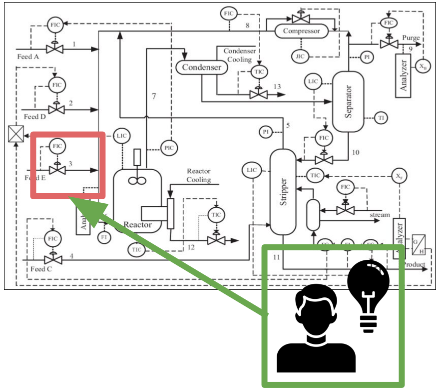
➔ Building (and updating) simulators

➔ Handling plant-model mismatch; uncertainty

Fault Identification and Analysis (Part 2)

Complicated systems require expert diagnosis





Challenges:

➔ Root cause diagnosis is ultimately manual

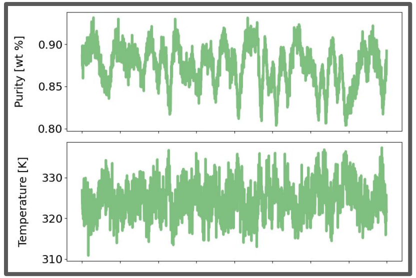
➔ Diagnosis complicated by operator turnover

1: "Digital Twin of a Petroleum Refinery (Source: GE Ventures)", https://www.arcweb.com/blog/digital-twins-define-oil-gas-40
2: https://www.freepik.com/free-photo/environmental-pollution-factory-exterior-night_16920299.htm#fromView=search&page=1&position=14&uuid=b1546320-95fa-42bd-8a7d-5b2ace5e5086

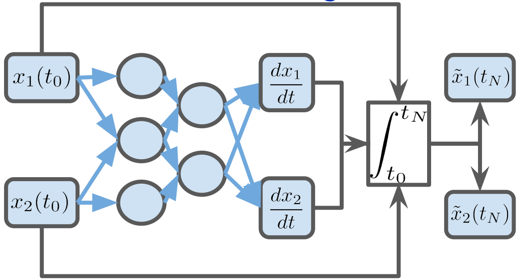
Proposed SciML Solutions for ICPS

Virtual Experimentation → **Part 1: Data-Driven & Differentiable Emulators**

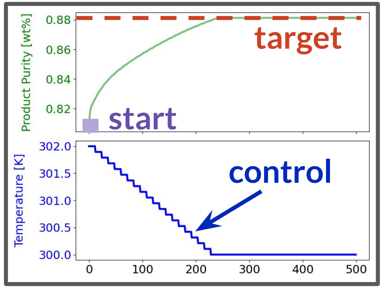
Real Time Data



Differentiable Surrogates

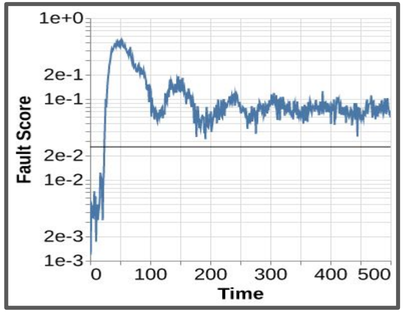


Optimal Control

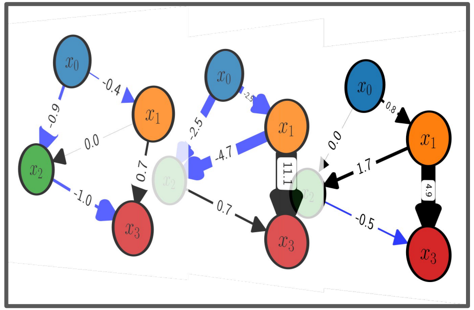


Fault Identification and Analysis → **Part 2: Time Resolved Causality**

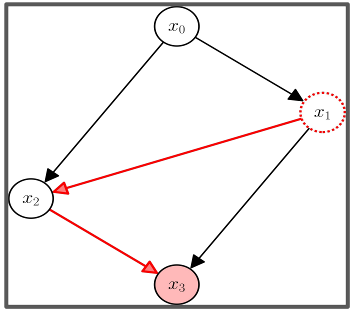
Real Time Monitoring



Causal Effects



Root Cause Diagnosis

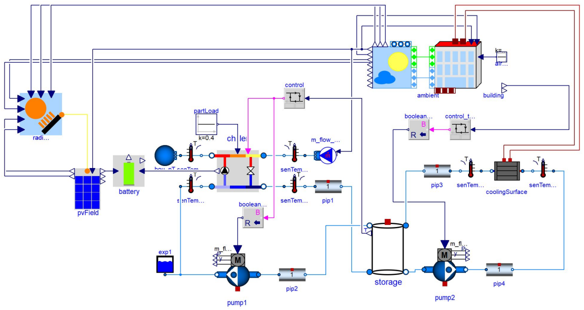


PASTEUR LABS

Part 1: Building Data-Driven (Differentiable) Emulators for ICPS

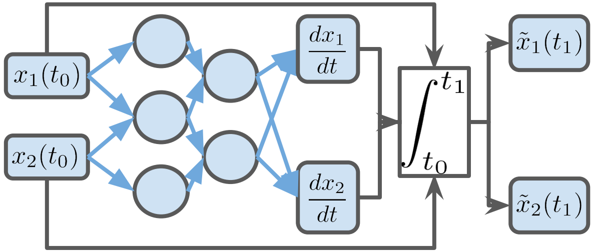
Use traditional simulation tools as emulators

Modelica Language



Complex assimilation & validation lifecycle

Build differentiable data-driven emulators

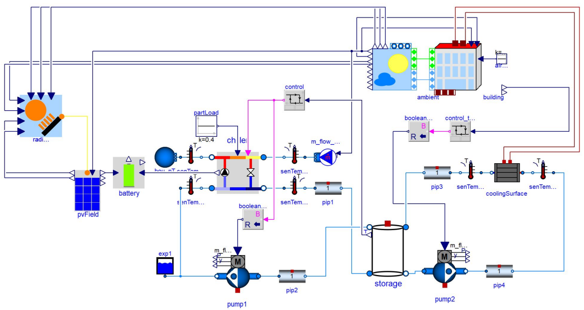


Natural data assimilation mechanism / lifelong learning

https://modelica-buildingsystems.de/features.html

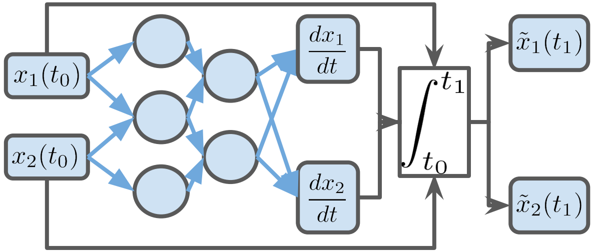
Part 1: Building Data-Driven (Differentiable) Emulators for ICPS

Use traditional simulation tools as emulators



Complex assimilation & validation lifecycle

Build differentiable data-driven emulators



Natural data assimilation mechanism / lifelong learning

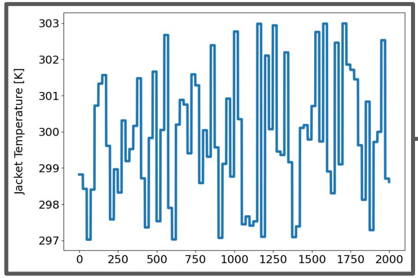
Our Focus: Differentiable emulators naturally facilitate **optimization**, **control**, and **design** activities

https://modelica-buildingsystems.de/features.html

Dynamic Surrogates for ICPS: Setting the Stage

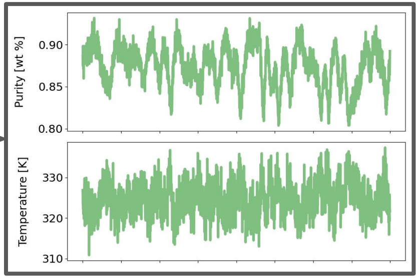
Given a history of process data...

Observed Control Input History



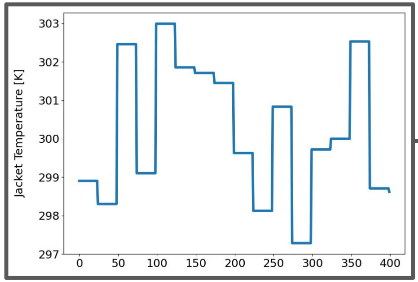
Real Plant Dynamics

Observed (Noisy) Measurements

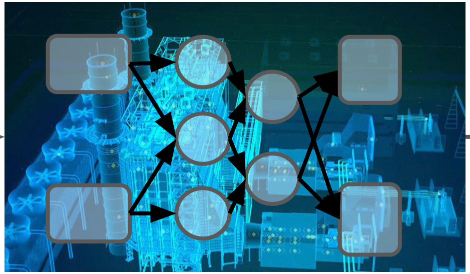


We seek dynamic surrogates that can be treated as continuous functions

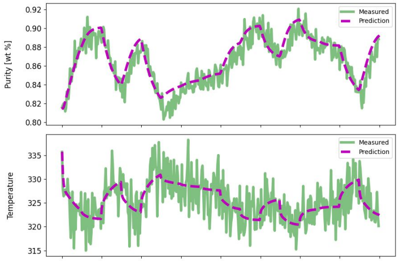
Proposed Control Input



Plant Dynamic Surrogate



Predicted Plant State



Dynamic Surrogates for ICPS: Setting the Stage

In a *generic sense*, we want to learn the following mapping

Initial Conditions

$$\mathbf{x}_0 = \begin{bmatrix} x_{0,1} \\ x_{0,2} \\ \vdots \\ x_{0,K} \end{bmatrix}$$

Input Trajectories

$$\mathbf{U} = \begin{bmatrix} u_{0,0} & u_{1,0} & \cdots & u_{M,0} \\ u_{0,1} & u_{1,1} & \cdots & u_{M,1} \\ \vdots & \vdots & \vdots & \vdots \\ u_{0,N-1} & u_{1,N-1} & \cdots & u_{M,N-1} \end{bmatrix}$$

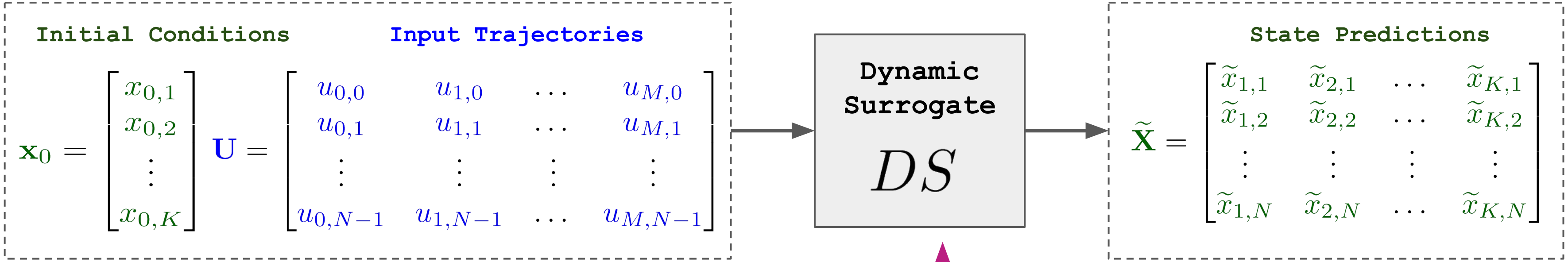
→ **Dynamic Surrogate** DS →

State Predictions

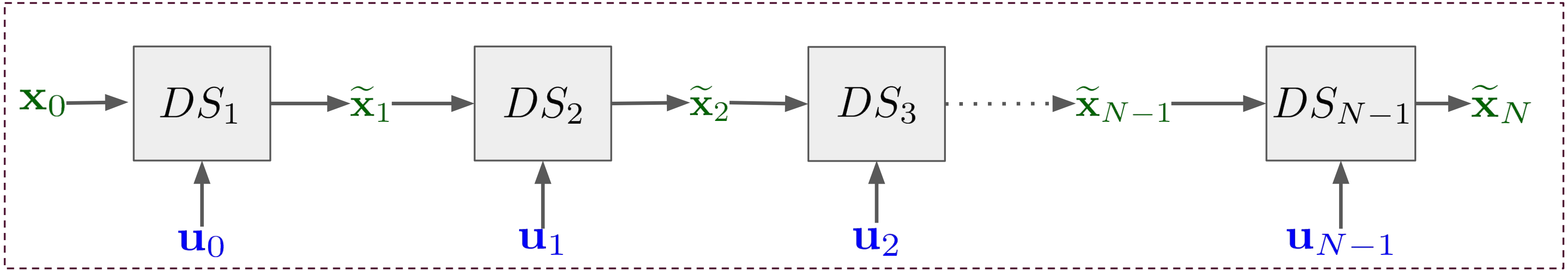
$$\widetilde{\mathbf{X}} = \begin{bmatrix} \widetilde{x}_{1,1} & \widetilde{x}_{2,1} & \cdots & \widetilde{x}_{K,1} \\ \widetilde{x}_{1,2} & \widetilde{x}_{2,2} & \cdots & \widetilde{x}_{K,2} \\ \vdots & \vdots & \vdots & \vdots \\ \widetilde{x}_{1,N} & \widetilde{x}_{2,N} & \cdots & \widetilde{x}_{K,N} \end{bmatrix}$$

Dynamic Surrogates for ICPS: Setting the Stage

In a *generic sense*, we want to learn the following mapping



A dynamic surrogate can be *unrolled* over a time interval

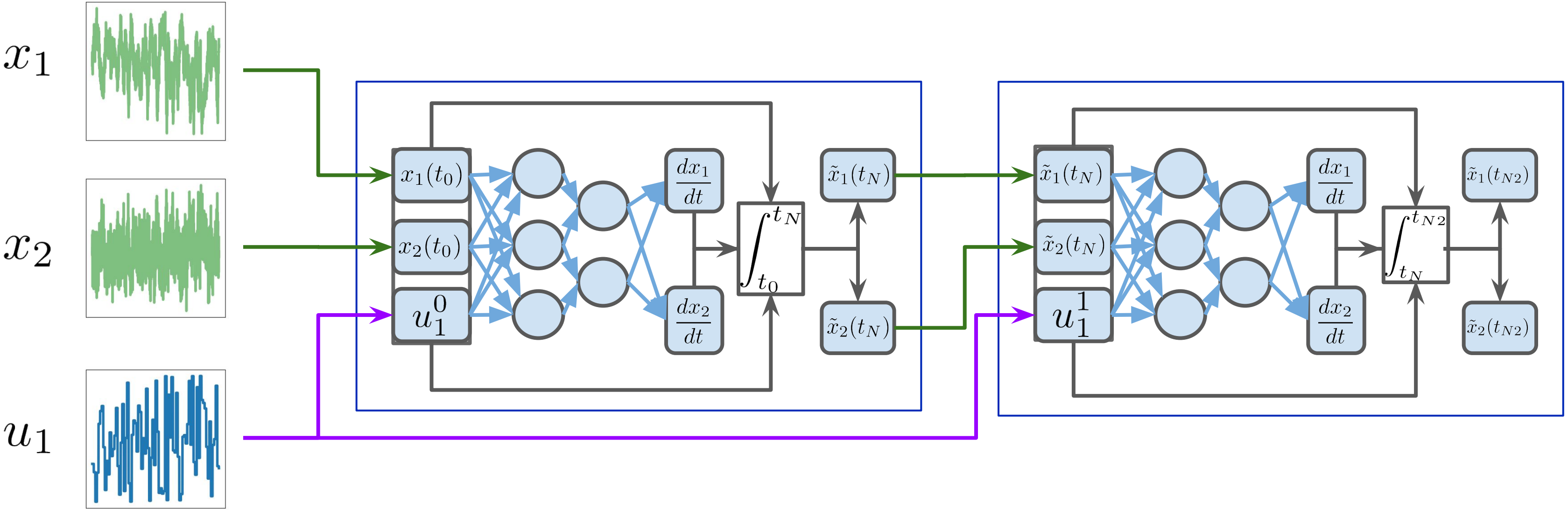


For later: a dynamic surrogate can be formulated in both **generic** and **unrolled** representations

Training Dynamic Surrogates: Neural **Controlled** ODEs

$$\underset{\theta}{\text{minimize}} \quad \sum_{i=1}^{m}\left(Q_1\|x_1^i - \hat{x}_1^i\|_2^2 + \sum_{k=1}^{N}\left(Q_x\|x_k^i - \hat{x}_k^i\|_2^2 + Q_{dx}\|\Delta x_k^i - \Delta\hat{x}_k^i\|_2^2\right)\right)$$

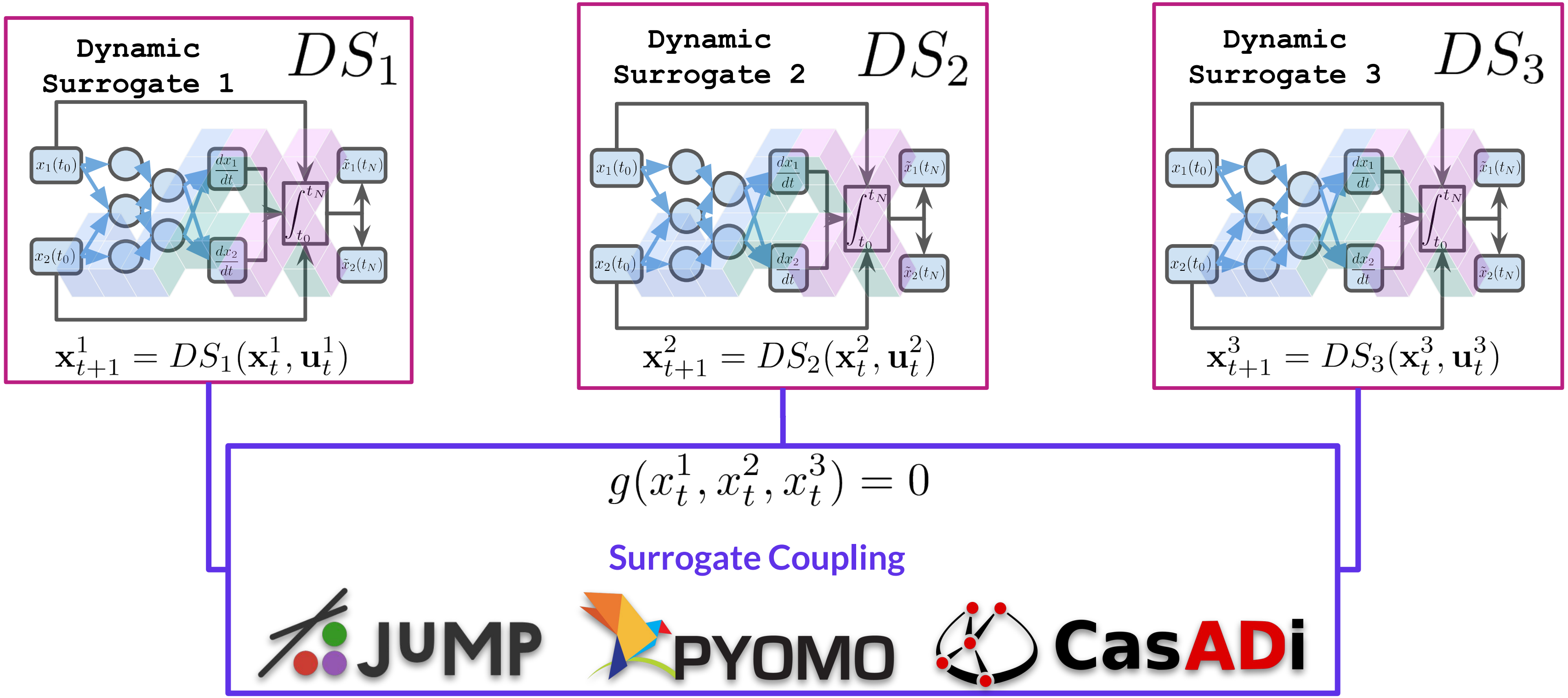
$$\text{subject to} \quad x_{k+1}^i = \text{ODESolve}\boxed{(f_\theta(x_k^i, u_k^i))}$$ → for each [state] trajectory



Neural Controlled ODEs* incorporate control actions at irregular intervals → **ICPS requirement**

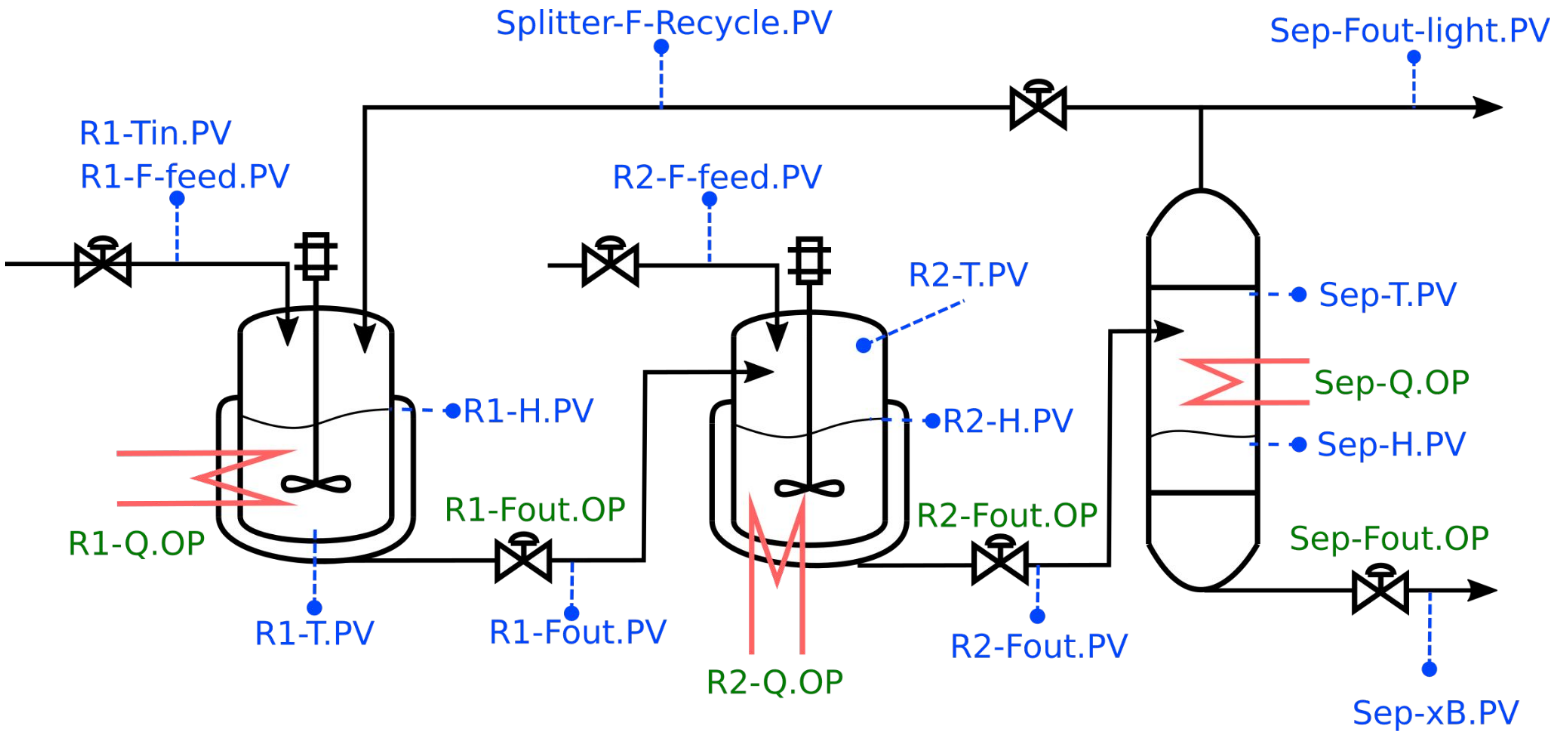
*Neuromancer-based ODEs: https://github.com/pnnl/neuromancer

Modular Dynamic Surrogates for ICPS: Optimization and Control



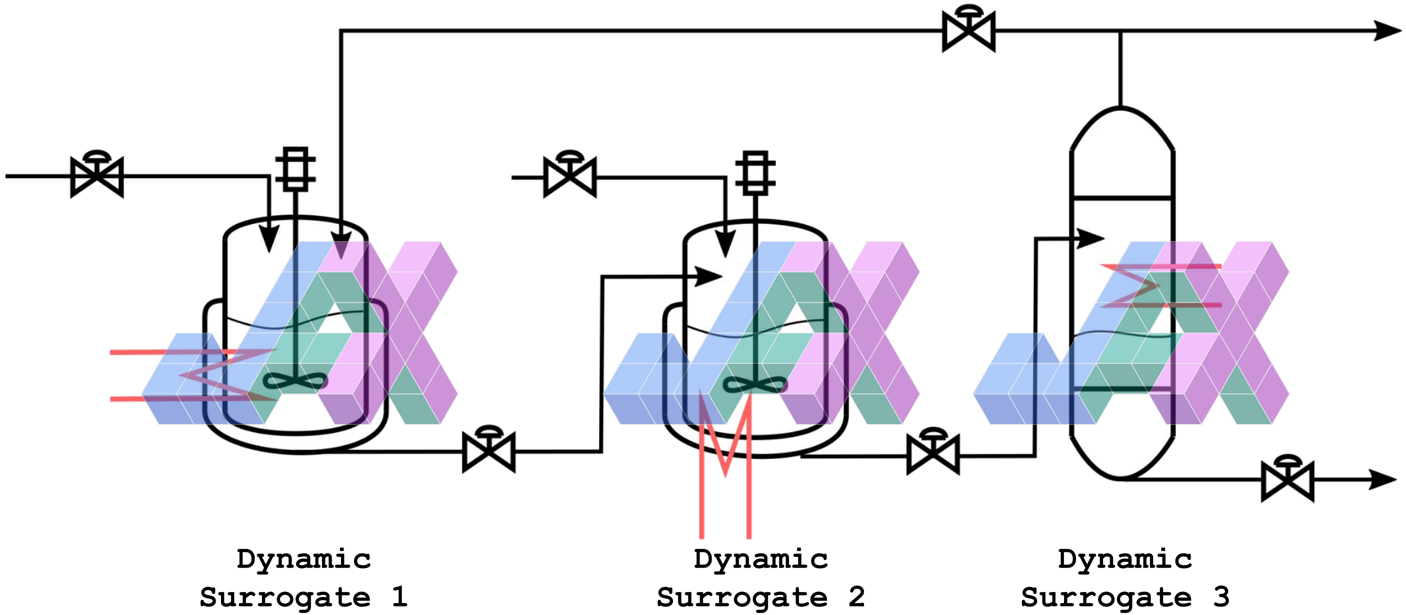
Optimization frameworks like JuMP, Pyomo, and CasADi have **sparsity exploiting AD engines**

Example: Nonlinear MPC with Neural Controlled Surrogates



Stewart, B.T., Wright, S.J., & Rawlings, J.B., Cooperative distributed model predictive control for nonlinear systems. *Journal of Process Control*, 2011

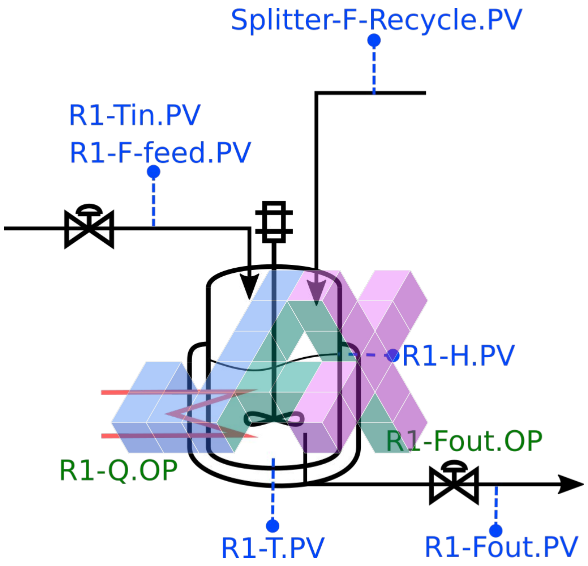
Example: Nonlinear MPC with Neural Controlled Surrogates



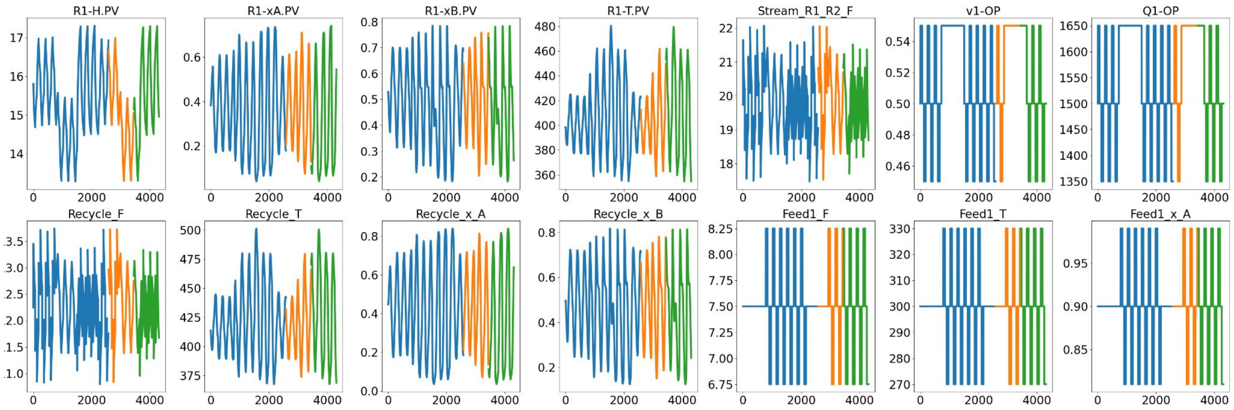
Main Idea: Train surrogates in isolation → Link within broader optimization-based framework to perform MPC

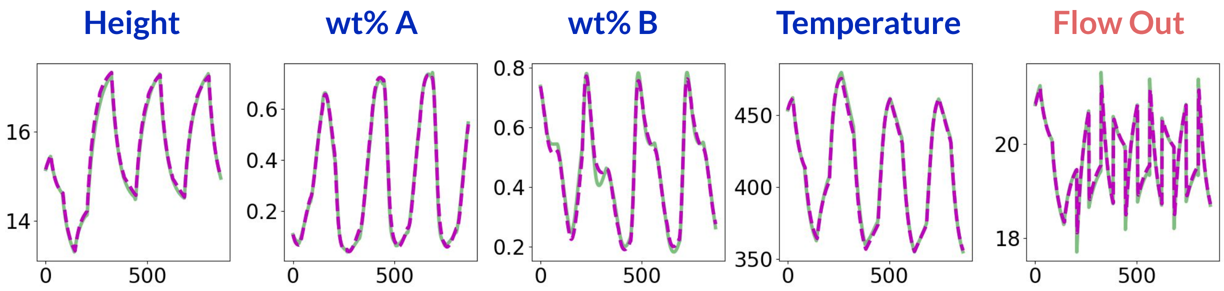
Example: Nonlinear MPC with Neural Controlled Surrogates

Train Reactor 1



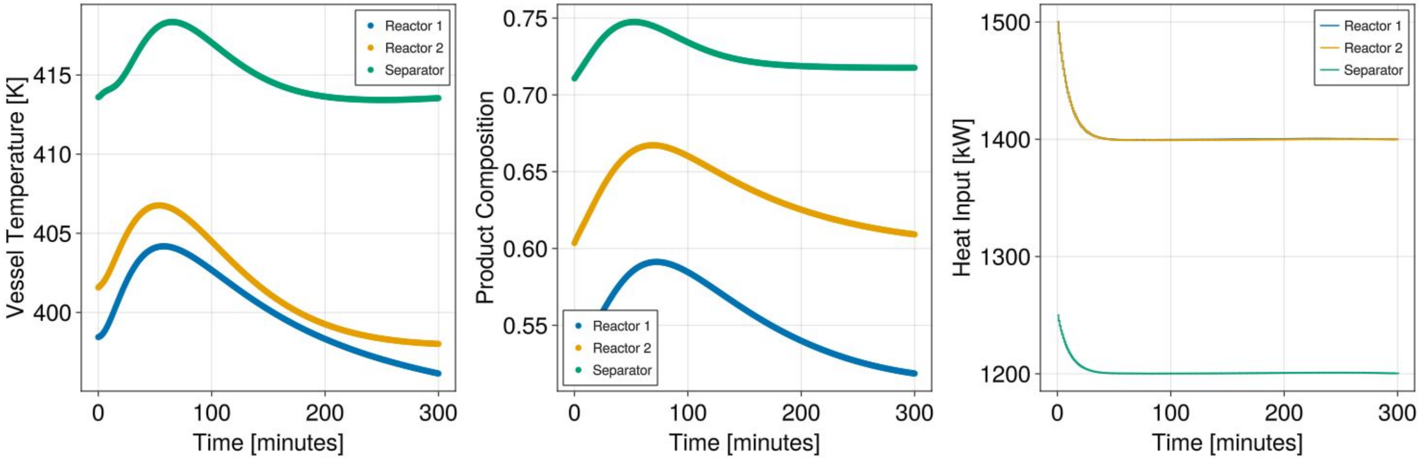
Train Dev Test





Example: Nonlinear MPC with Neural Controlled Surrogates

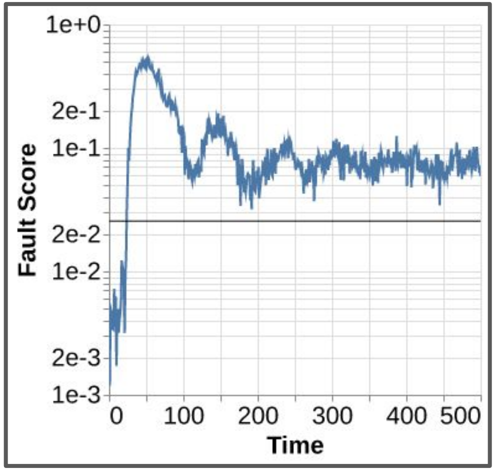
$$\begin{aligned}
\underset{\mathbf{x}_1,\mathbf{u}_1,\mathbf{x}_2,\mathbf{u}_2,\mathbf{x}_3,\mathbf{u}_3}{\text{minimize}} \quad & \sum_{t=1}^{N}\left(\mathbf{x}_{1,t}^T Q \mathbf{x}_{1,t} + \mathbf{u}_{1,t}^T R \mathbf{u}_{1,t}\right) + \sum_{t=1}^{N}\left(\mathbf{x}_{2,t}^T Q \mathbf{x}_{2,t} + \mathbf{u}_{2,t}^T R \mathbf{u}_{2,t}\right) + \sum_{t=1}^{N}\left(\mathbf{x}_{3,t}^T Q \mathbf{x}_{3,t} + \mathbf{u}_{3,t}^T R \mathbf{u}_{3,t}\right) \\
\text{subject to} \quad & \mathbf{x}_{1,t+1} = \text{ODEsolve}(f_{1,\theta}(\mathbf{x}_{1,t}, \mathbf{u}_{1,t})) \quad \text{(Reactor 1)} \quad \forall t \in \mathcal{T} \\
& \mathbf{x}_{2,t+1} = \text{ODEsolve}(f_{2,\theta}(\mathbf{x}_{2,t}, \mathbf{u}_{2,t})) \quad \text{(Reactor 2)} \quad \forall t \in \mathcal{T} \\
& \mathbf{x}_{3,t+1} = \text{ODEsolve}(f_{3,\theta}(\mathbf{x}_{3,t}, \mathbf{u}_{3,t})) \quad \text{(Separator)} \quad \forall t \in \mathcal{T} \\
& g(\mathbf{x}_{1,t}, \mathbf{x}_{2,t}, \mathbf{x}_{3,t}) \leq 0 \quad \text{(Coupling Constraints)} \quad \forall t \in \mathcal{T}
\end{aligned}$$



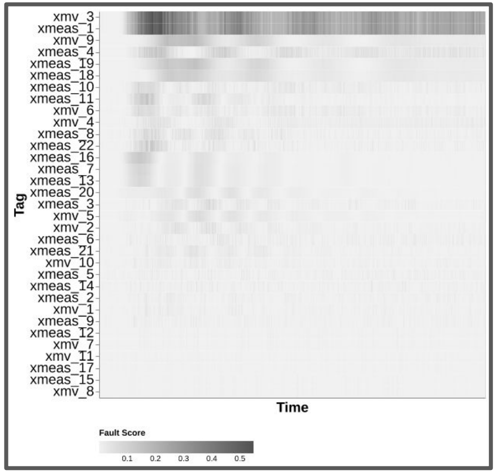
Nonlinear optimizers (like Ipopt) can robustly incorporate linked dynamic surrogates

Part 2: Fault Analysis - Time Resolved Causality

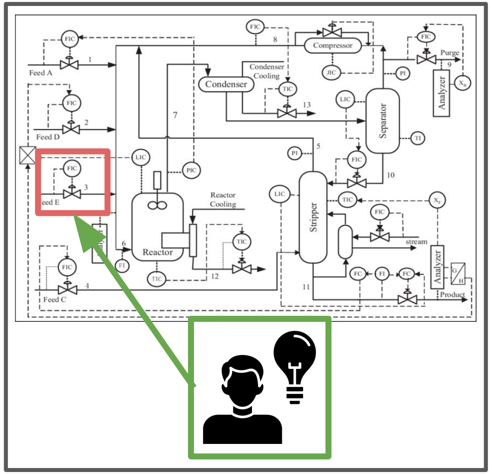
Detection
"Something went wrong"



Identification
"These sensors are a problem"



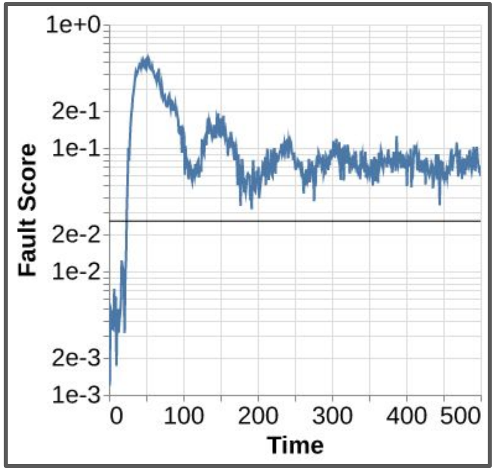
Diagnosis
"This valve was flipped"



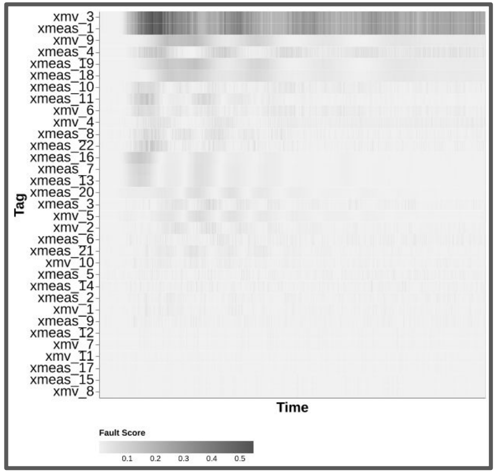
Detection and Identification techniques are mature and commercially implemented…
Bayesian-neural-network classifiers [Sun2020] | Geometric-based detection [Smith2022]

Part 2: Fault Analysis - Time Resolved Causality

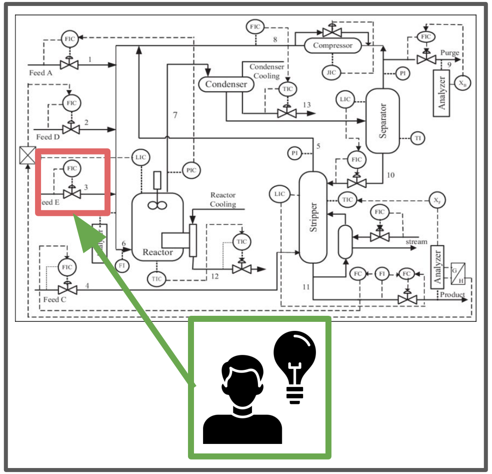
Detection
"Something went wrong"



Identification
"These sensors are a problem"



Diagnosis
"This valve was flipped"



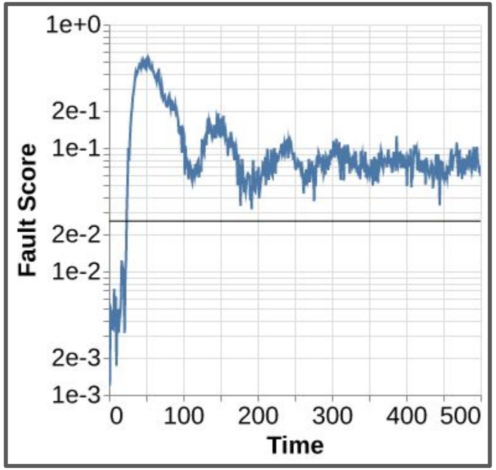
Fault diagnosis is a perpetually challenging problem

Methods often require **a-priori knowledge** of system faults. **What if the system changes? New faults?**

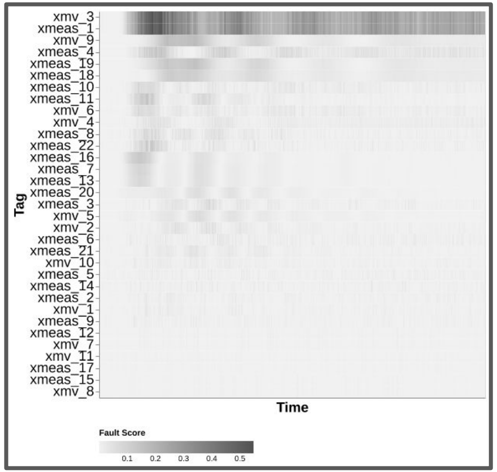
Chiang, L.H., Russell, E.L., Braatz, R.D., Fault Detection and Diagnosis in Industrial Systems, Advanced Textbooks in Control and Signal Processing, 2001

Part 2: Fault Analysis - Time Resolved Causality

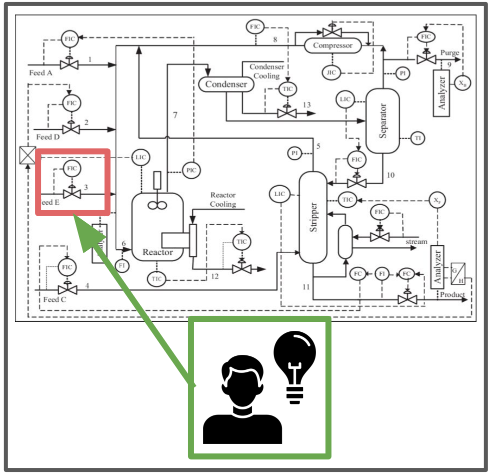
Detection
"Something went wrong"



Identification
"These sensors are a problem"



Diagnosis
"This valve was flipped"



Enter Causal Discovery

Standard framework for **determining causation** among observations. Provide graphical intuition.

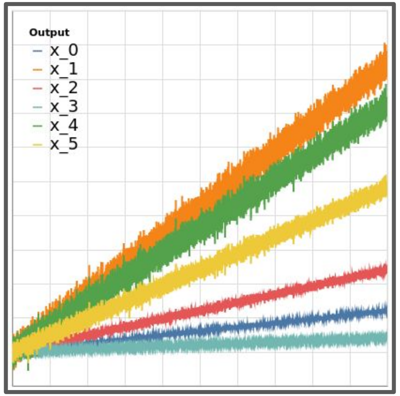
Chiang, L.H., Russell, E.L., Braatz, R.D., Fault Detection and Diagnosis in Industrial Systems, Advanced Textbooks in Control and Signal Processing, 2001

Quick Summary: Causal Discovery with LINGAM

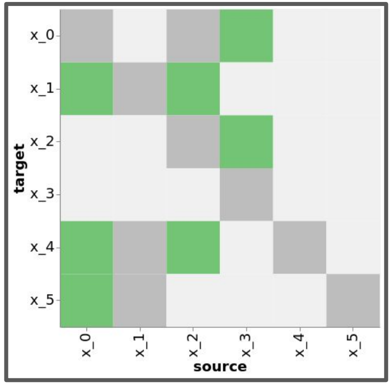
Causal discovery seeks to learn a Structural Equation Model (SEM) that describes observed variables

We specifically seek to identify a **Li**near **N**on-**G**aussian **A**cyclic **M**odel (LiNGAM)

Observed Data



Prior Knowledge

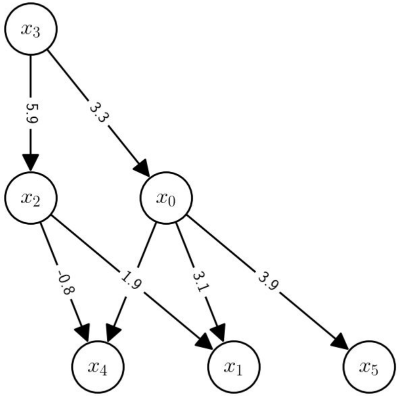


Fit Causal Model

$$x = Bx + e$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ b_{10} & 0 & 0 & 0 & 0 & 0 \\ b_{20} & b_{21} & 0 & 0 & 0 & 0 \\ b_{30} & b_{31} & b_{32} & 0 & 0 & 0 \\ b_{40} & b_{41} & b_{42} & b_{43} & 0 & 0 \\ b_{50} & b_{51} & b_{52} & b_{53} & b_{54} & 0 \end{bmatrix}$$

View Causal Graph



Causal discovery methods have been sparsely explored for ICPS applications... ***Why?***

Causal Discovery for Process Systems

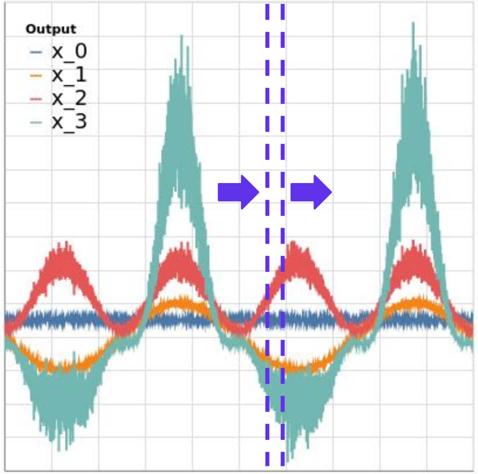
The ICPS setting violates almost all causal discovery assumptions

1. **Cycles** in the physical plant due to recycle and feedback control [assumption violation → **DAG**]
2. **Nonstationarity** in causal effects due to dynamics [assumption violation → **stationarity**]
3. **Control systems** mask physical causal relations [assumption violation -> **faithfulness**]
4. **Control** as correlated noise [assumption violation -> **iid**]
5. **Hysteresis** as a result of operator decisions [assumption violation - **Markov**]

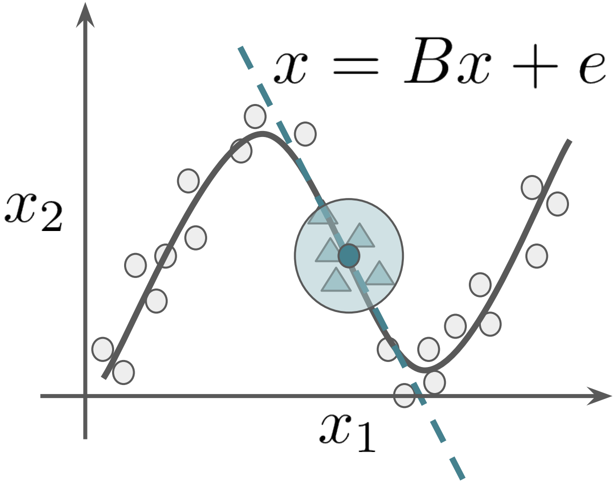
Causal Discovery for Process Systems: Just In Time Solutions

IDEA: Identify **locally linear** causal models over moving window – **Just In Time** (JIT)

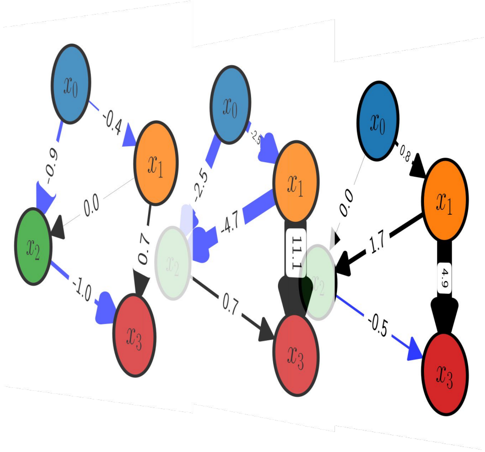
Ingest Moving Online Window



Fit Locally Linear Models
JIT-LiNGAM



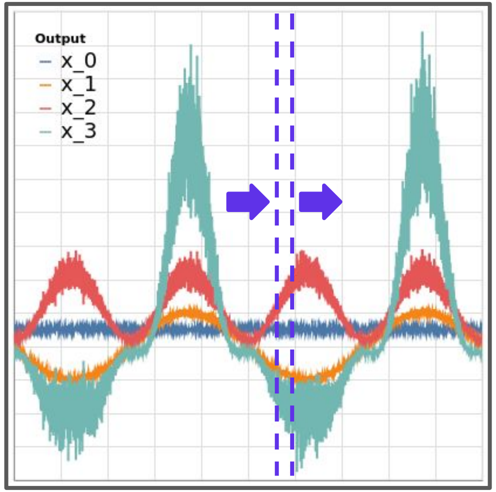
Obtain Causal Time Series



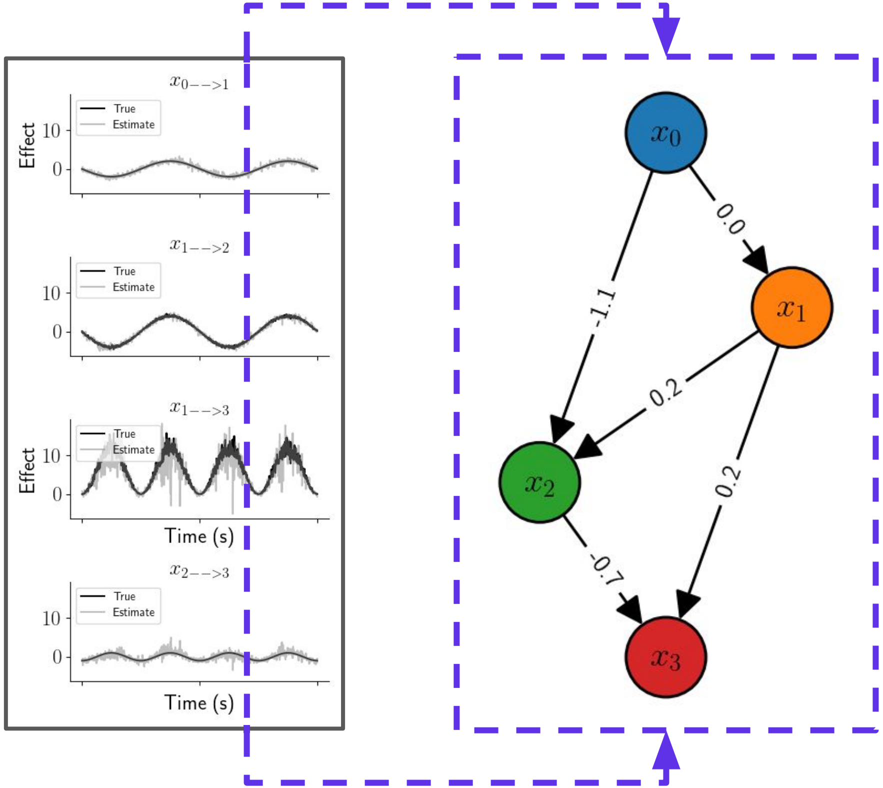
Fujiwara, D., Koyama, K., Kiritoshi, K., Okawachi, T., Izumitani, T., Shimizu, S., n.d. Causal Discovery for Non-stationary Non-linear Time-series Data Using Just-In-Time Modeling, 2023

Does JIT-LINGAM Work?

Noisy nonlinear dynamics

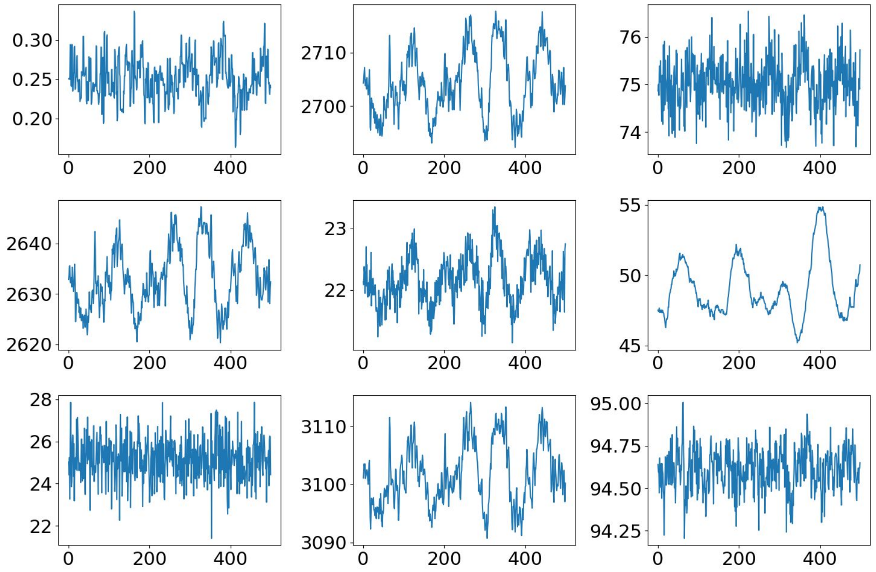


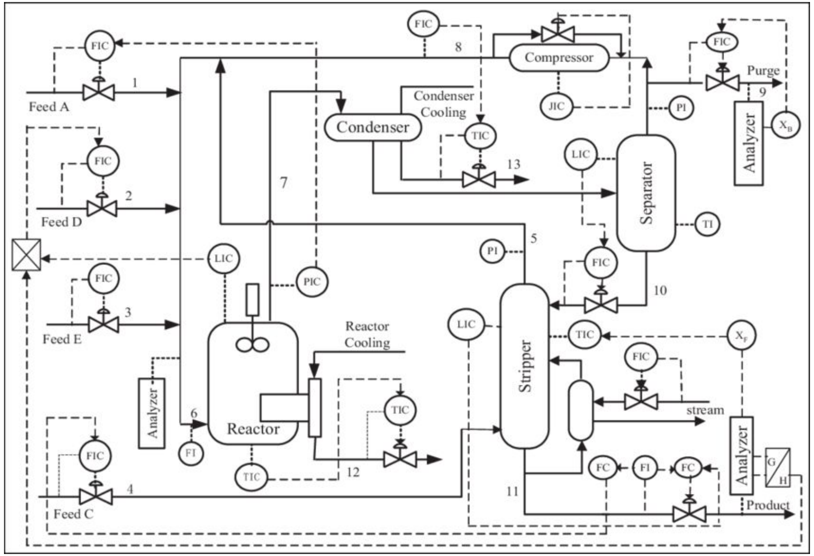
Run JIT-LiNGAM



Under limited assumptions and the right implementation → **Yes! JIT-LiNGAM works**

The Tennessee Eastman Process (TEP)

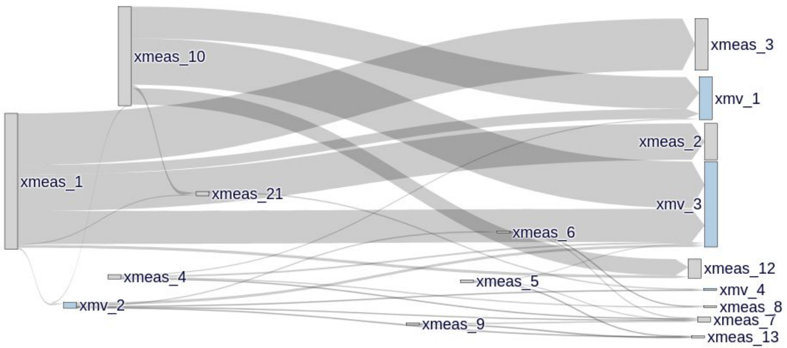




J. Downs and E. Vogel, "A plant-wide industrial process control problem," Computers and Chemical Engineering, vol. 17, no. 3, pp. 245–255, 1993, industrial challenge problems in process control.

TEP: Calculating & Comparing Causal Effects

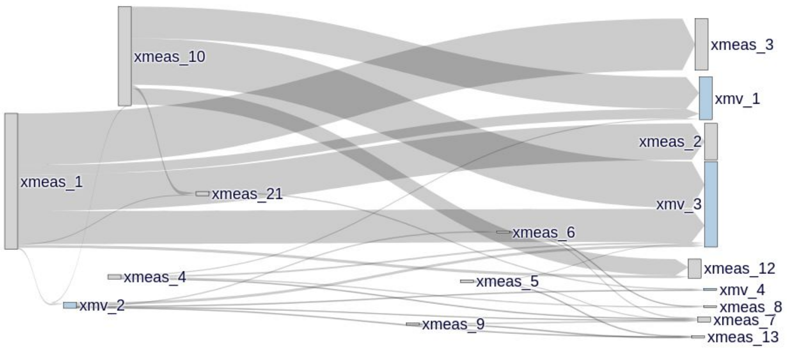
Raw TEP Causal Effects



Issue: No obvious method to compare effects

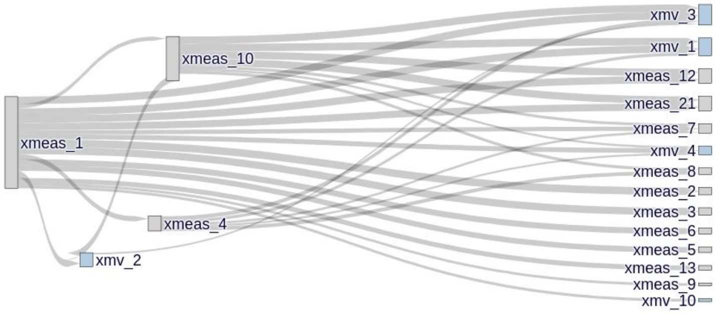
TEP: Calculating & Comparing Causal Effects

Raw TEP Causal Effects



Issue: No obvious method to compare effects

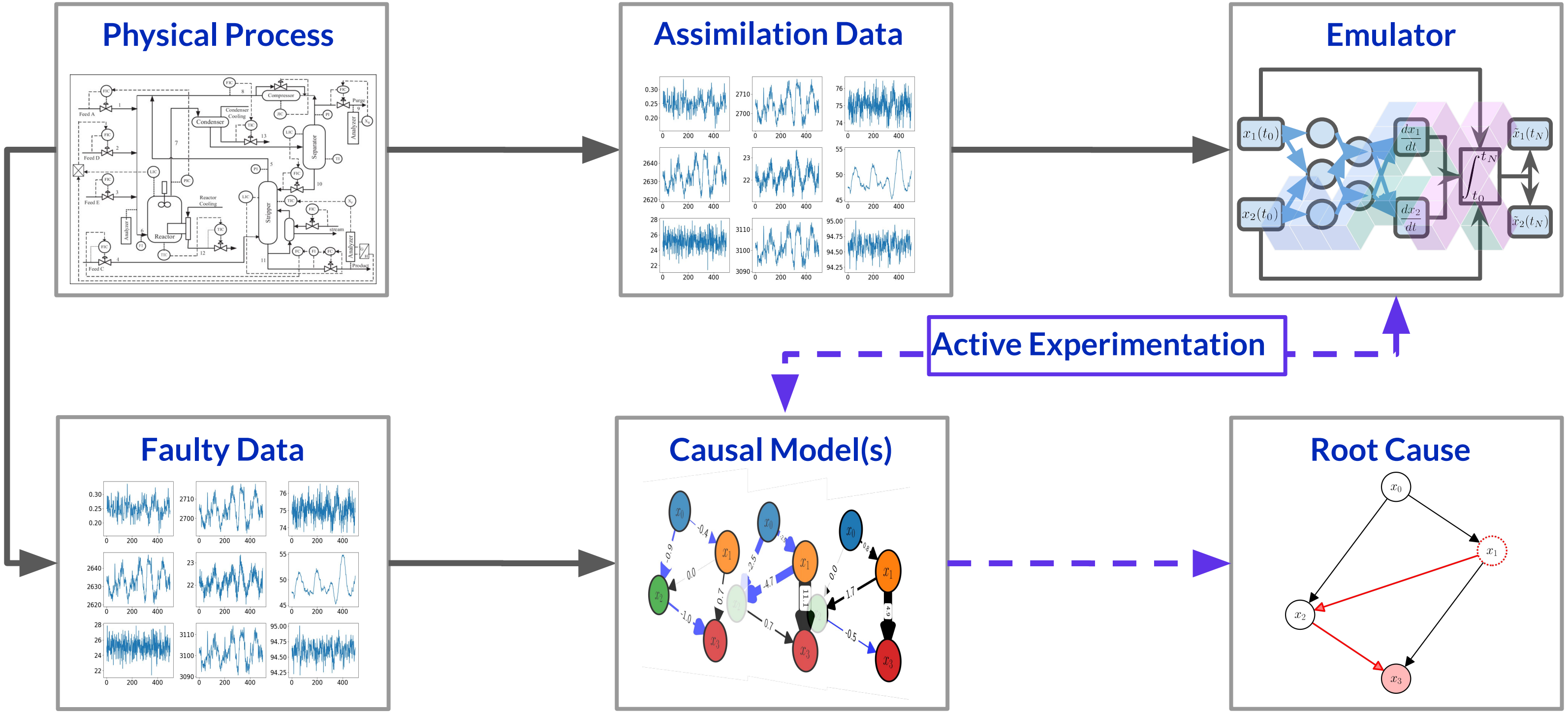
Normalized TEP Causal Effects



Valid **normalization** facilitates fault diagnosis

Going Forward: Causal Discovery with Active Experimentation

Root cause diagnosis is further aided by **active experimentation**



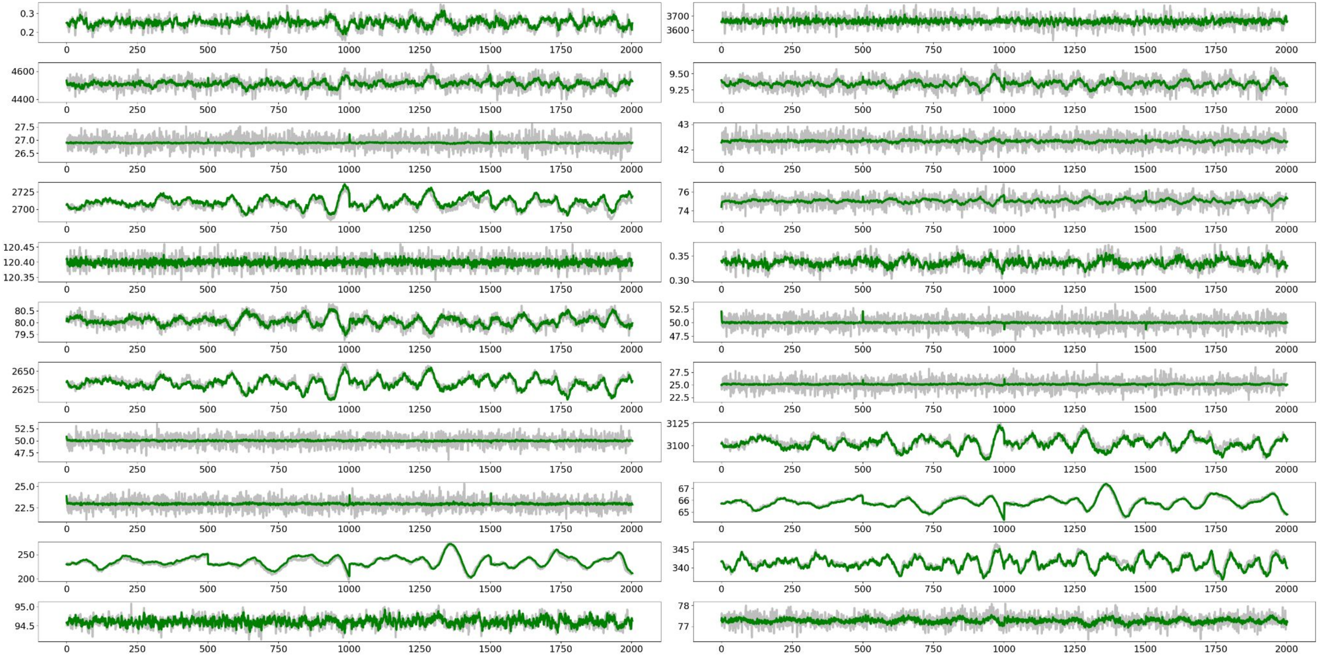
Active experimentation requires a representative **emulator** for the dynamics

Going Forward: Causal Discovery with Active Experimentation

Active experimentation requires virtual emulation. **Can we learn the TEP dynamics?**

Going Forward: Causal Discovery with Active Experimentation

Active experimentation requires virtual emulation. **Can we learn the TEP dynamics?**



Yes! Vanilla neural ODE architectures are performant under realistic ICPS settings

Closing Remarks

Dynamic Surrogates & Optimization

- ➔ In our experience, neural ODE dynamic surrogates are performant against realistic process data
- ➔ Unifying differentiable programming with optimization frameworks is a powerful paradigm
- ➔ Promising outlook towards integrating **mesh-based** surrogates for virtual experimentation

Causal Discovery and Intervention

- ➔ JiT-LINGAM is a promising method for bringing causal discovery to online systems
- ➔ Causal discovery can help develop ‘intuition’ about underlying operational mechanisms
- ➔ Optimal interventions with causal models and dynamic surrogates will lead to tailored MPC formulations

Acknowledgements

Erik Peterson, PhD
Staff Scientist - Advanced Projects Lead
Pasteur Labs

Marta D'Elia, PhD
Principal Scientist
Pasteur Labs

+ the Pasteur Labs team!

... come build with us → simulation.science/careers

PASTEUR LABS

Thank You!

Jordan Jalving PhD

jordan.jalving@simulation.science

hello@simulation.science

PASTEUR LABS

simulation.science