Teaching Data-Centric Process Control (Junior Year) Using Experiential Learning

`ndcbe.github.io/controls`

Prof. Alexander (Alex) Dowling
Chemical and Biomolecular Engineering
University of Notre Dame

AIChE Annual Meeting
San Diego, CA
October 30, 2024

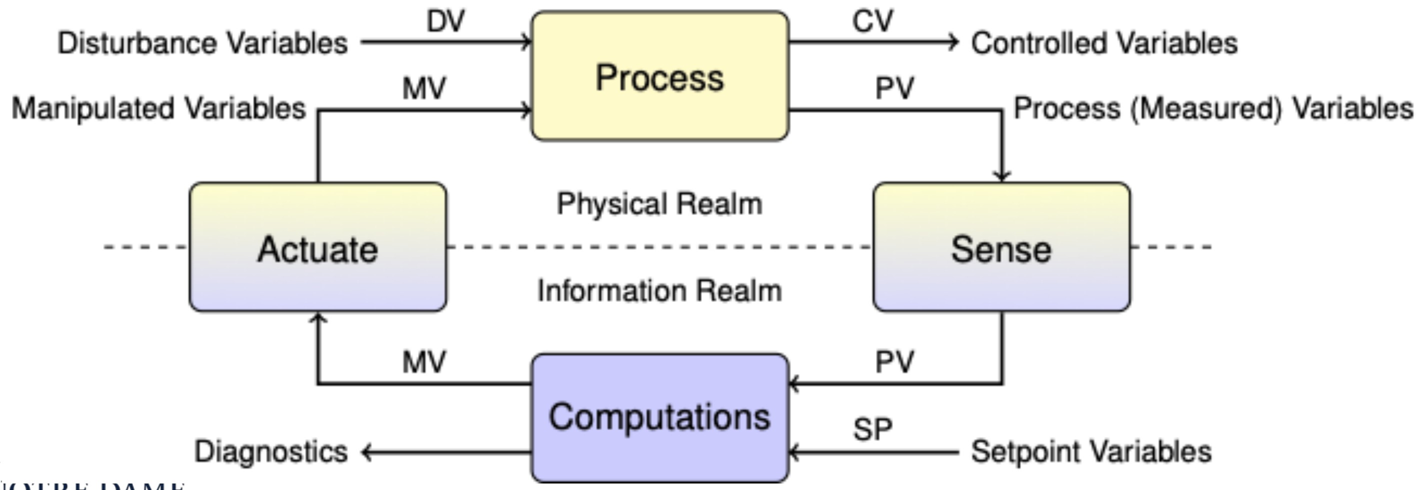
Prof. Jeff Kantor
(1954-2023)

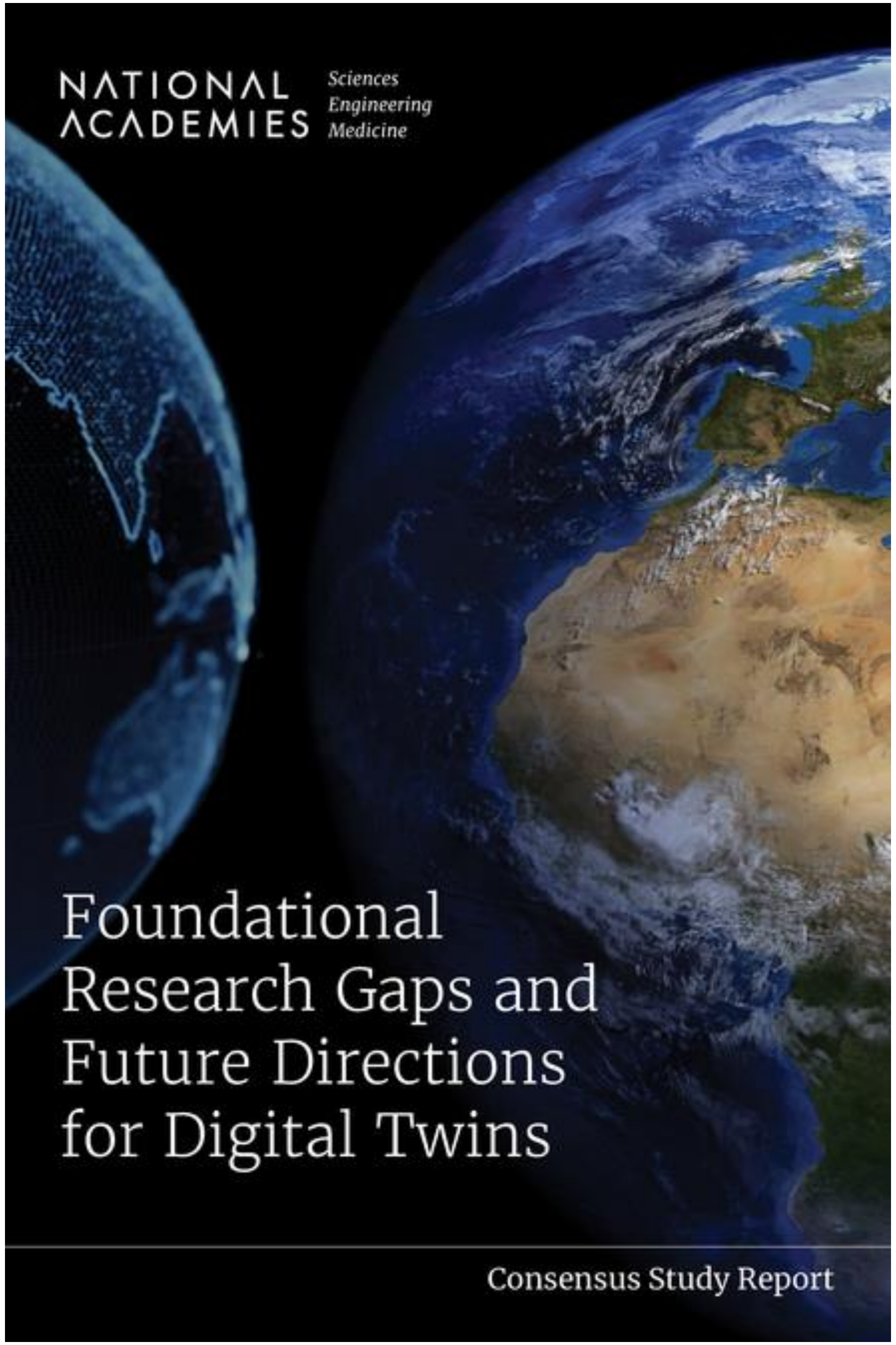
Contributors: Hailey Lynch, Molly Dougher, Maddie Watson, Zhicheng Lu, Daniel Laky

What is a Digital Twin?

*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.*

Feedback Control = Sensing + Computations + Actuation

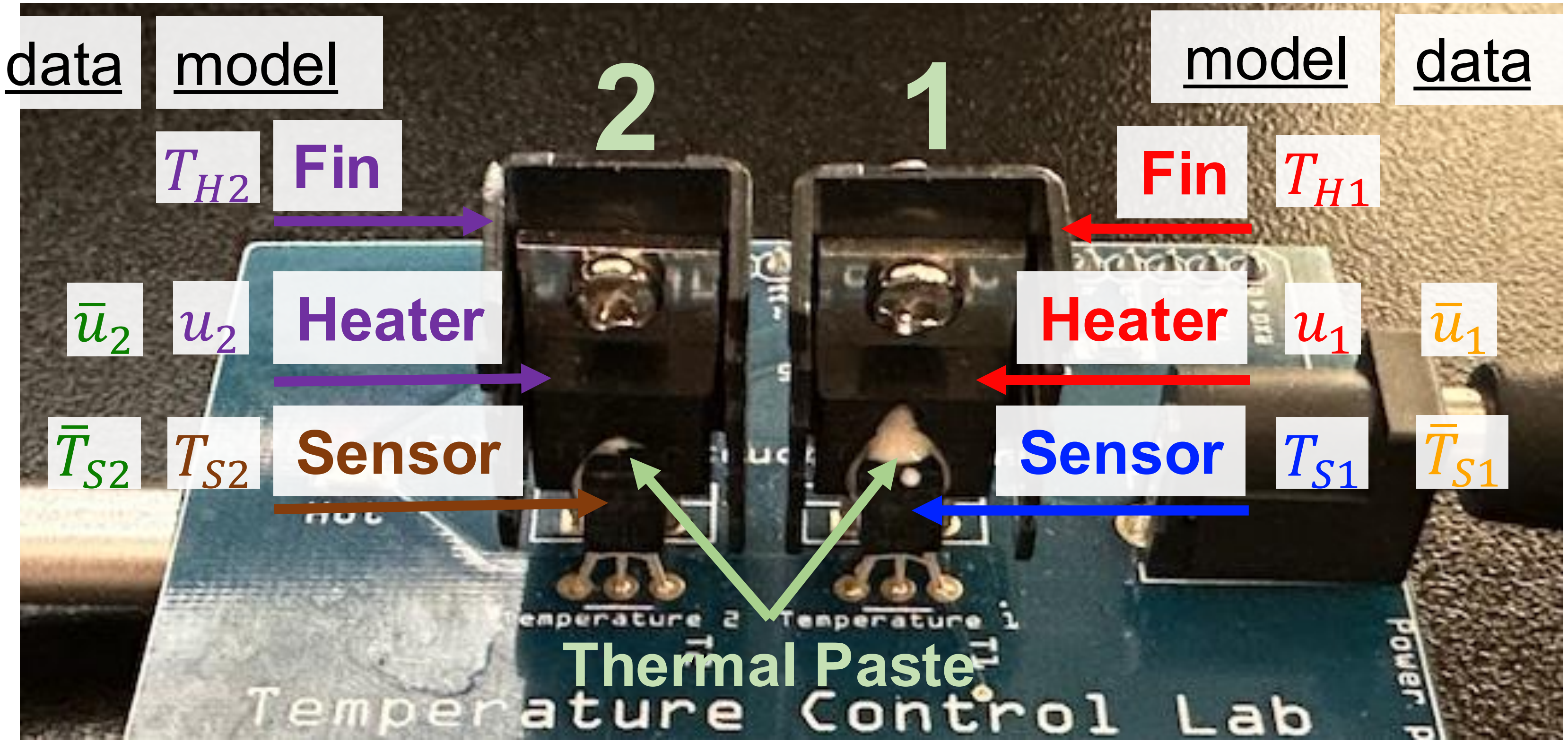



NATIONAL ACADEMIES Sciences Engineering Medicine

Foundational Research Gaps and Future Directions for Digital Twins

Consensus Study Report

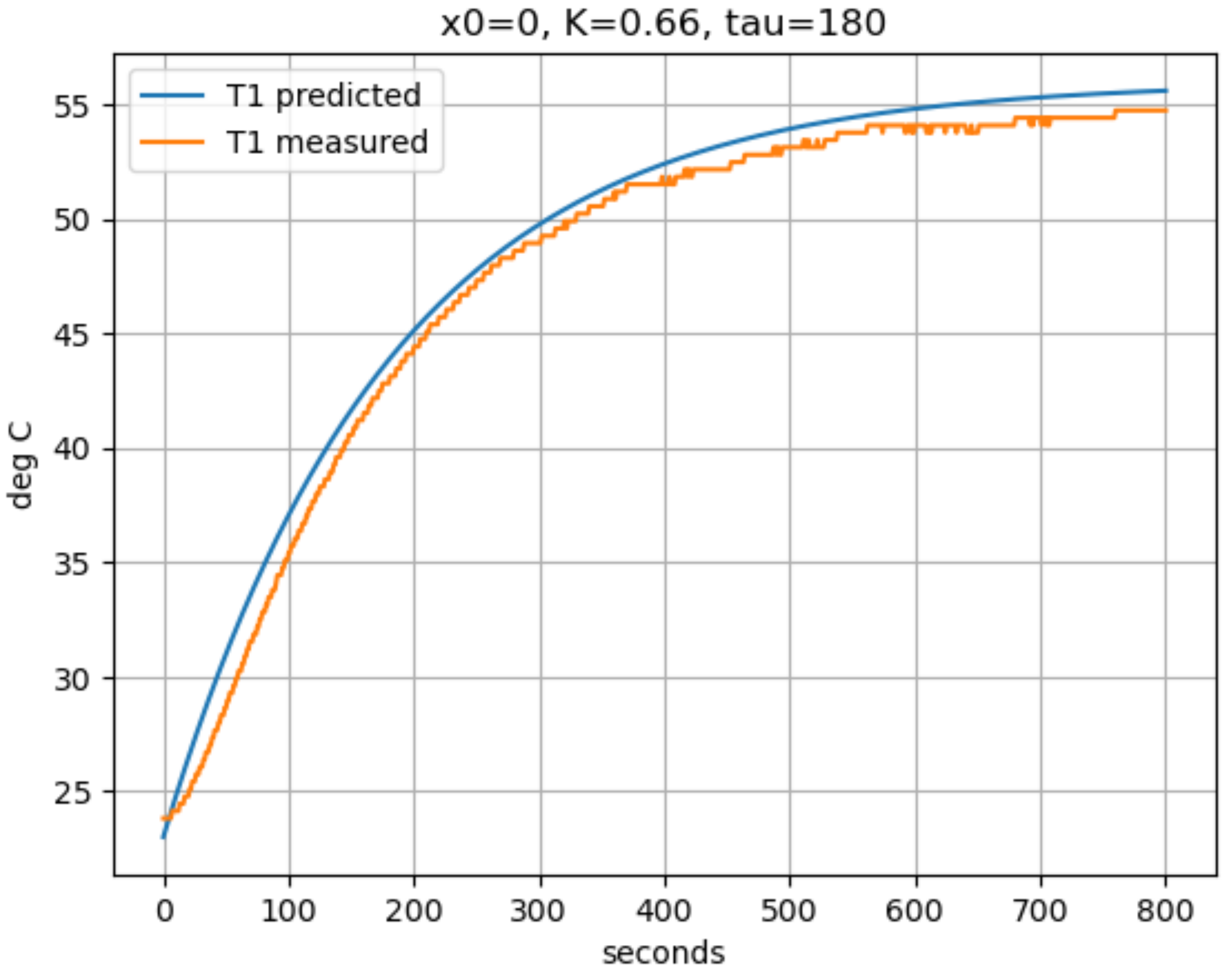
Temperature Control Lab (TC Lab)



$$C_p^H \frac{dT_{H,1}}{dt} = U_a(T_{amb} - T_{H,1}) + U_b(T_{S,1} - T_{H,1}) + \alpha P_1 u_1$$

$$C_p^S \frac{dT_{S,1}}{dt} = U_b(T_{H,1} - T_{S,1}), \qquad \theta = (U_a, U_b, C_p^H, C_p^S)^\intercal$$

Lab 1: Step Test and First-Order Model



$$\tau \frac{dx}{dt} = -x + K\bar{u}$$

$$x(t) = \underbrace{x_0 e^{-t/\tau}}_{\text{initial condition } x_0} + \underbrace{(1 - e^{-t/\tau}) K\bar{u}}_{\text{input } \bar{u}}$$

$$C_p \frac{dT_1}{dt} = U_a(T_{amb} - T_1) + \alpha P_1 u_1$$

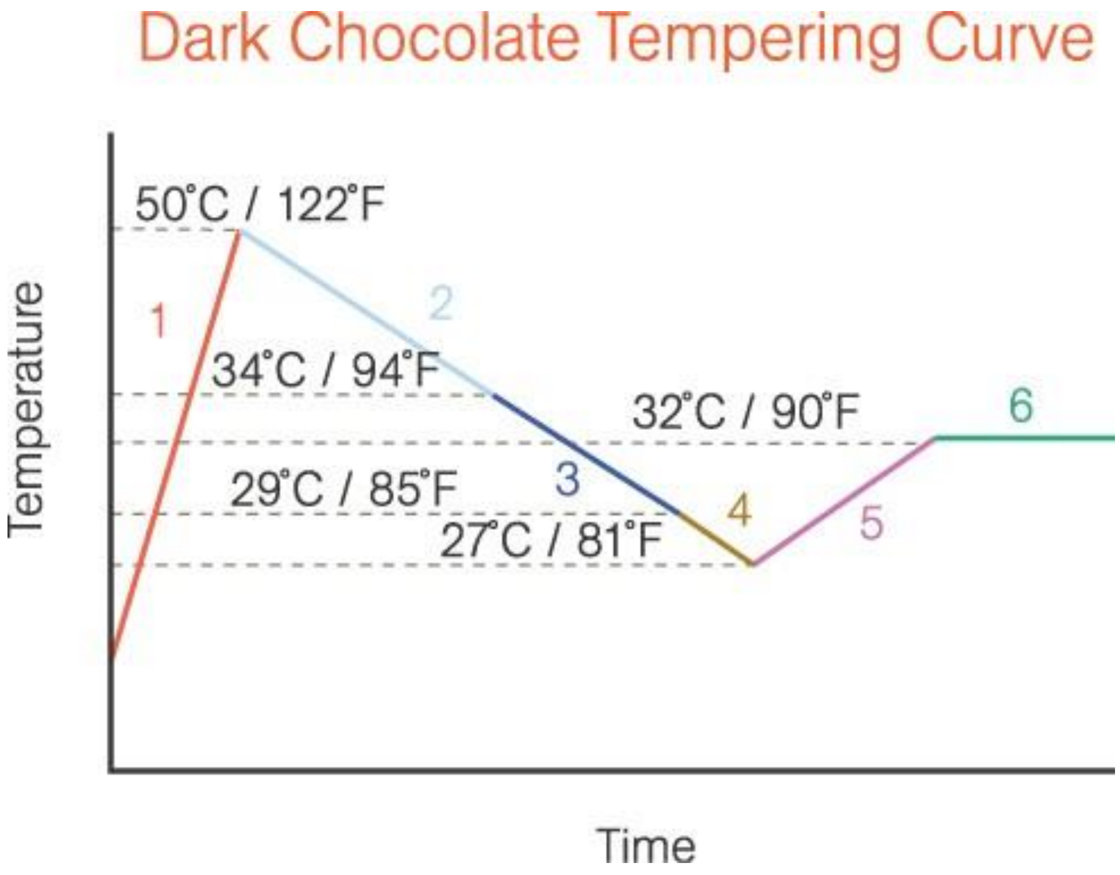
$$x = T_1 - T_{amb}$$

$$\frac{dx}{dt} = \frac{-U_a}{C_p} x + \frac{\alpha P_1}{C_p} u_1$$

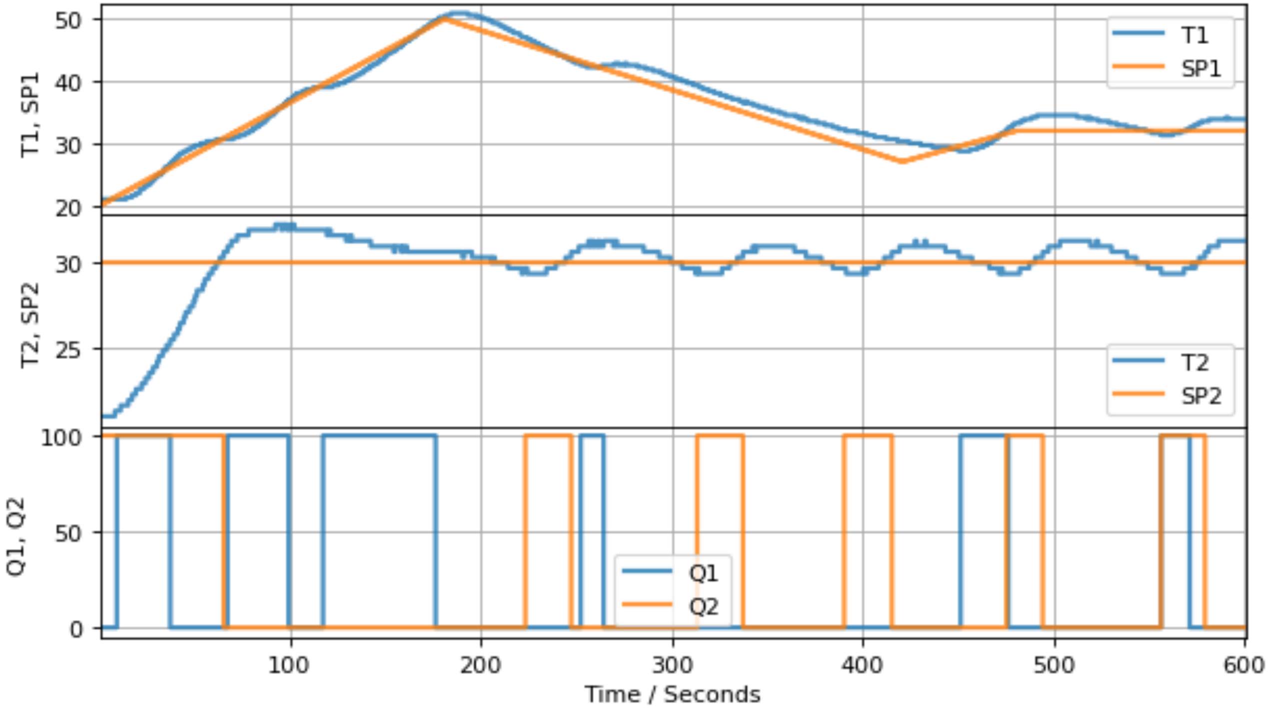
$$\frac{dx}{dt} = -\frac{1}{\tau} x + \frac{K}{\tau} \bar{u}$$

Important skill: read τ and K from plot

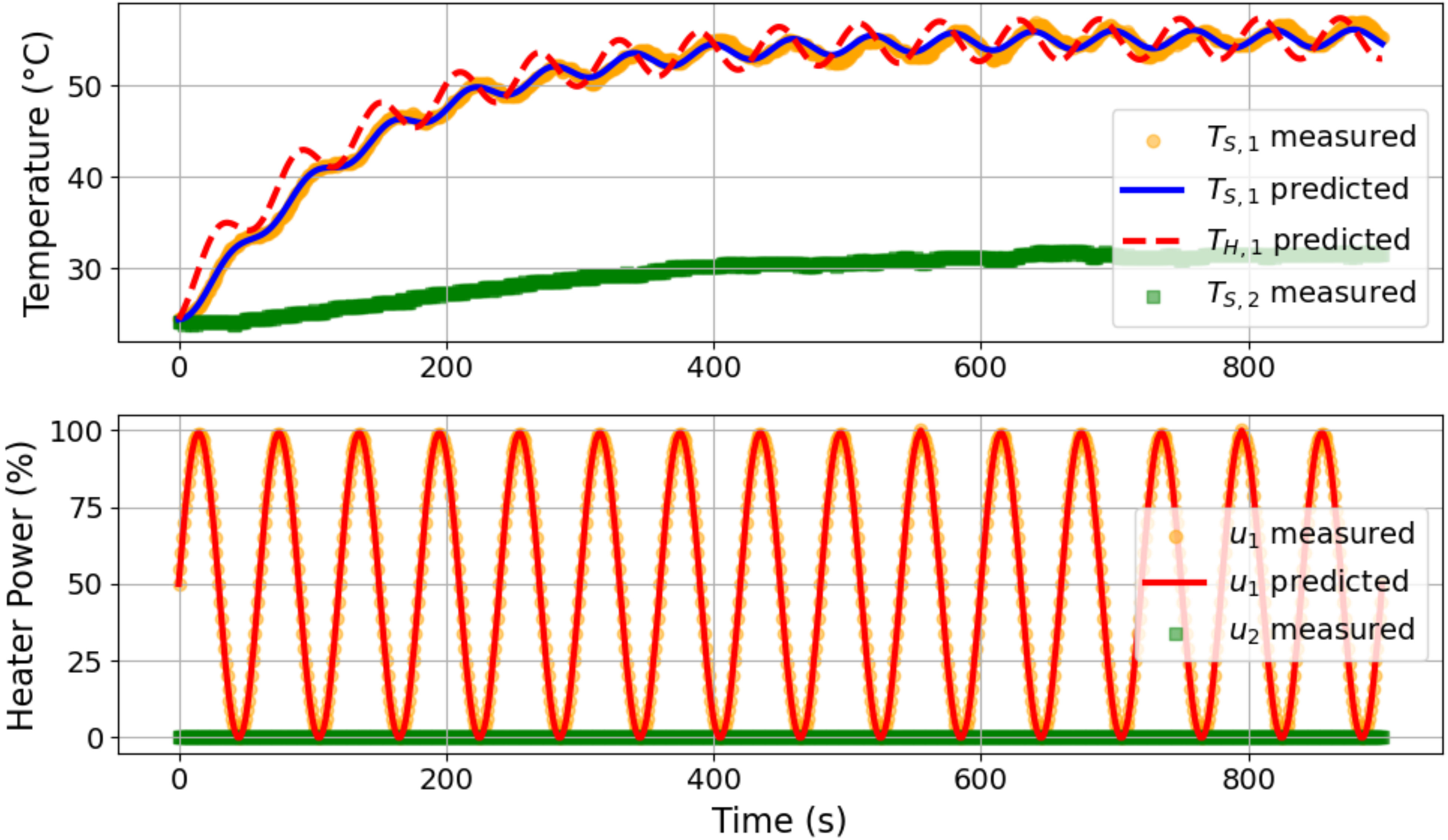
Lab 2: Relay (On/Off) Feedback Control



$$U_k = \begin{cases} U^{max} & \text{if } T_k \leq T_k^{SP} - d \\ U^{min} & \text{if } T_k \geq T_k^{SP} + d \\ U_{k-1} & \text{otherwise} \end{cases}$$



Labs 3: Data Collection and Parameter Estimation



Lab 4: Proportional Integral Control

Main idea: use mathematical model from Lab 3 to tune the controller

Control Law

$$u_1(t) = \bar{u}_1 + K_P(T_{set} - T_{S,1}(t)) + K_I \int_0^t (T_{set} - T_{S,1}(t'))dt'$$

Closed-Loop Dynamics

$$\frac{d}{dt} \underbrace{\begin{bmatrix} T^*_{H,1} \\ T^*_{S,1} \\ I \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} -\frac{U_a+U_b}{C_p^H} & \frac{U_b-\alpha P_1 K_P}{C_p^H} & \frac{\alpha P_1 K_I}{C_p^H} \\ \frac{U_b}{C_p^S} & -\frac{U_b}{C_p^S} & 0 \\ 0 & -1 & 0 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} T^*_{H,1} \\ T^*_{S,1} \\ I \end{bmatrix}}_{\mathbf{x}}$$

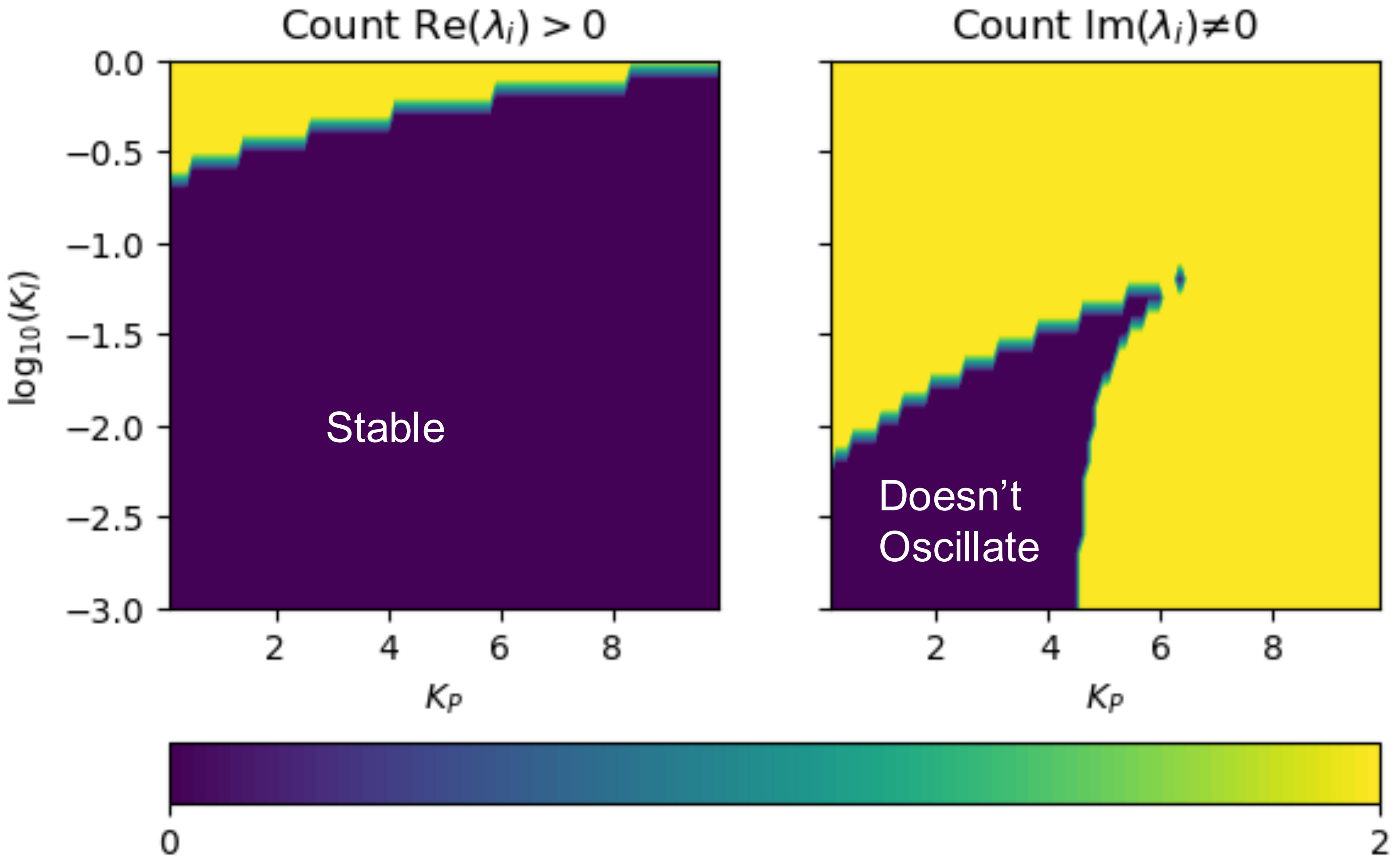
$$\underbrace{\begin{bmatrix} T^*_{S,1} \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}}_{\mathbf{C}} \underbrace{\begin{bmatrix} T^*_{H,1} \\ T^*_{S,1} \\ I \end{bmatrix}}_{\mathbf{x}}$$

Adjust K_P and K_I to tune eigenvalues of **A**

UNIVERSITY OF NOTRE DAME

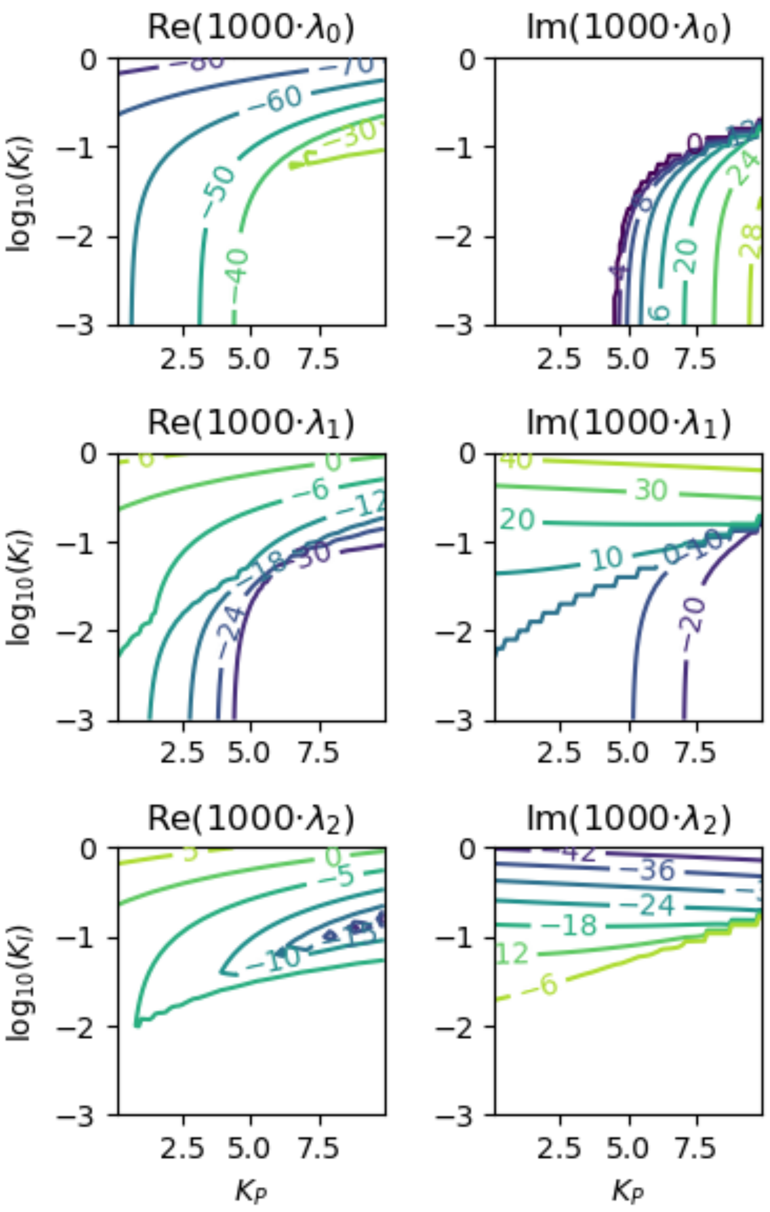
Note: we also analyze & implement velocity-form PI(D) with anti-wind-up

Lab 4: Proportional Integral Control (Stability Analysis)

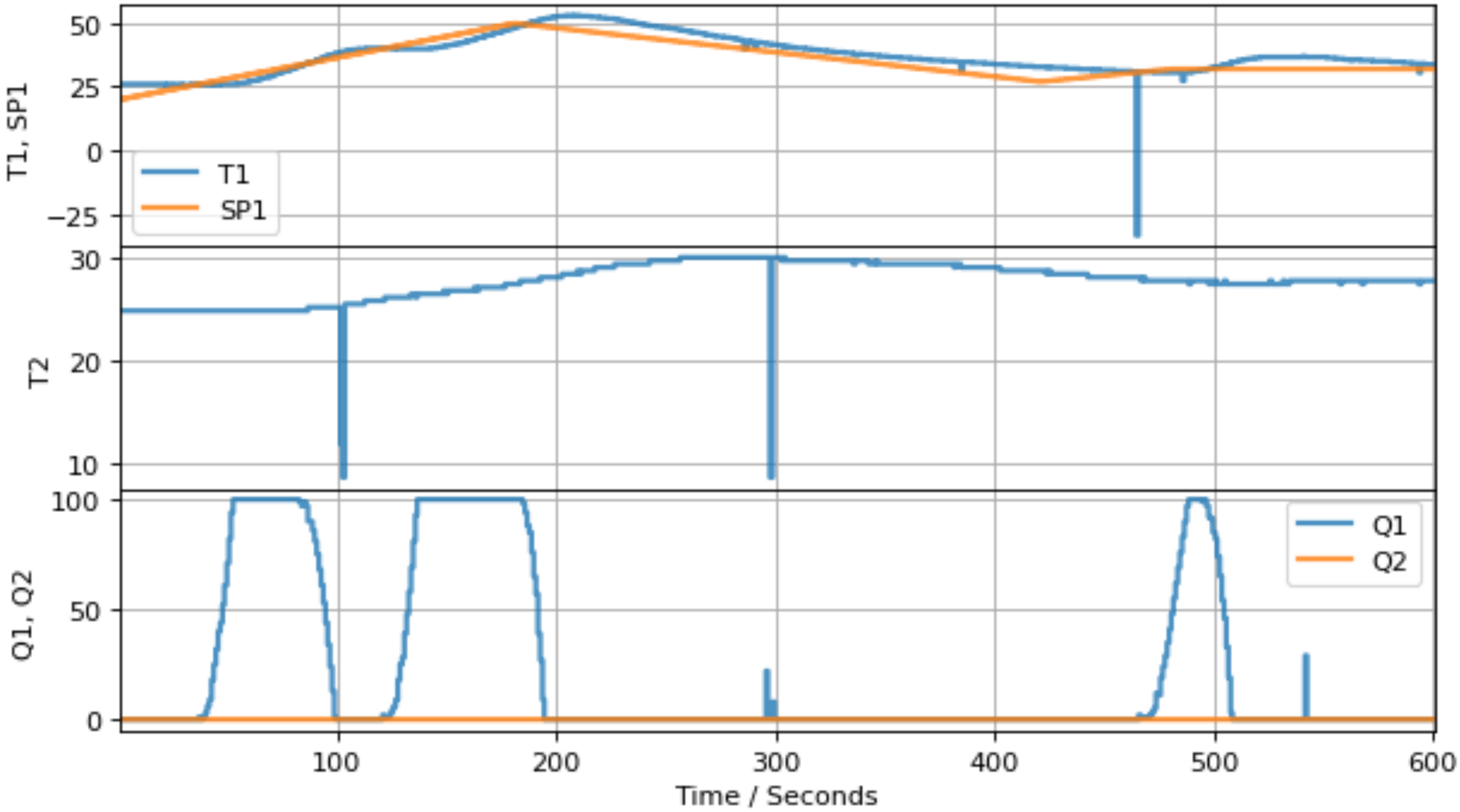


Stable

Doesn’t Oscillate

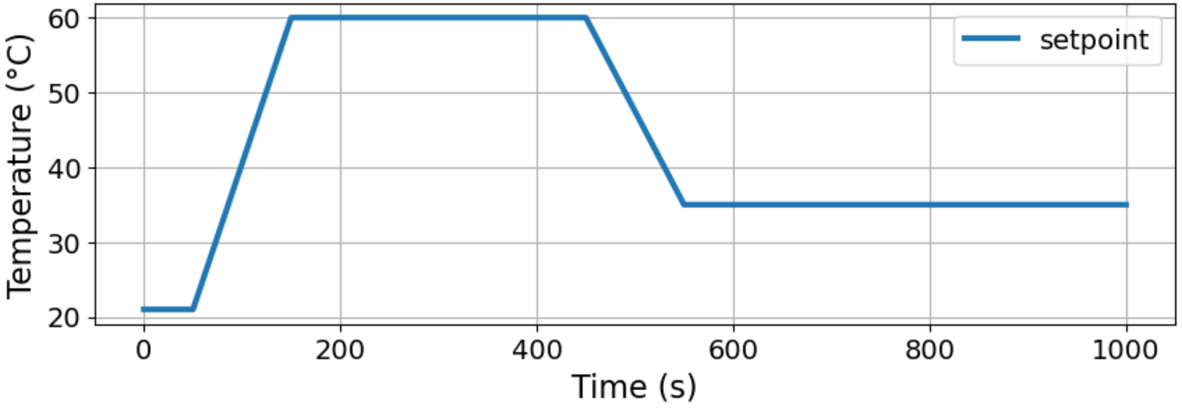


Lab 4: Proportional Integral Control (Hardware Results)

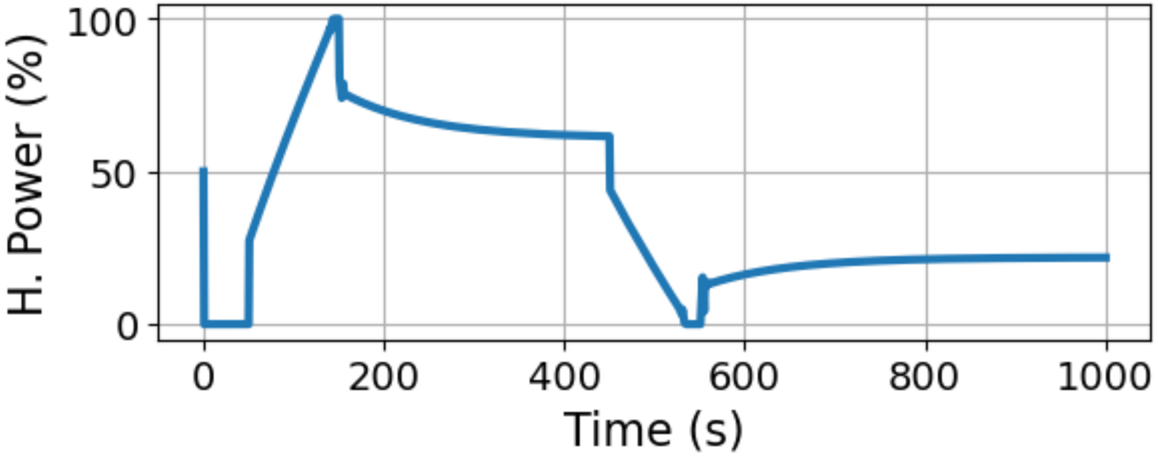


Lab 5: Open-Loop Dynamic Optimization in Pyomo

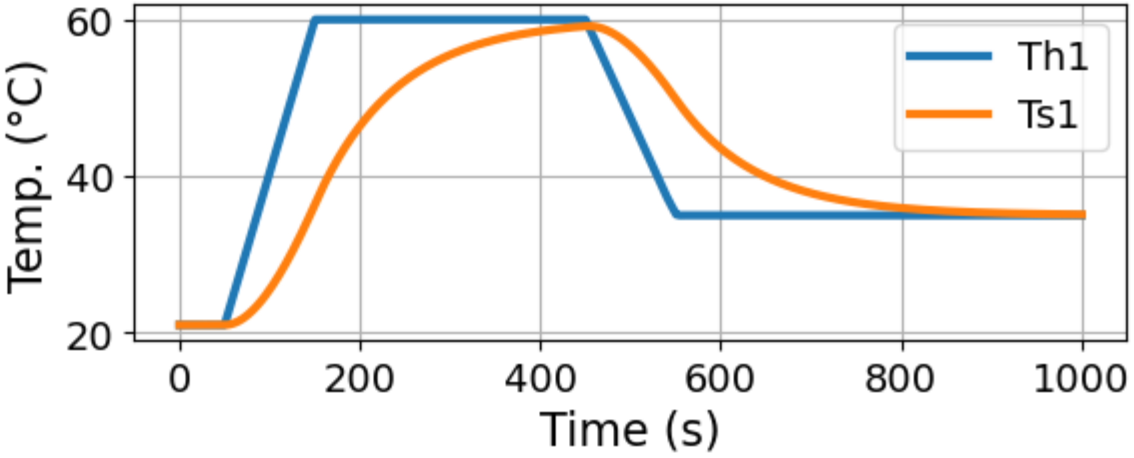
$$
\begin{aligned}
\min_{u(t)} \quad & \int_{t_0}^{t_f} \| SP(t) - T_H(t) \|^2 \, dt \\
\text{s.t.} \quad & C_p^H \frac{dT_H}{dt} = U_a(T_{amb} - T_H) + U_b(T_S - T_H) + \alpha P u(t) \\
& C_p^S \frac{dT_S}{dt} = U_b(T_H - T_S) \\
& T_H(t_0) = T_{amb} \\
& T_S(t_0) = T_{amb}
\end{aligned}
$$



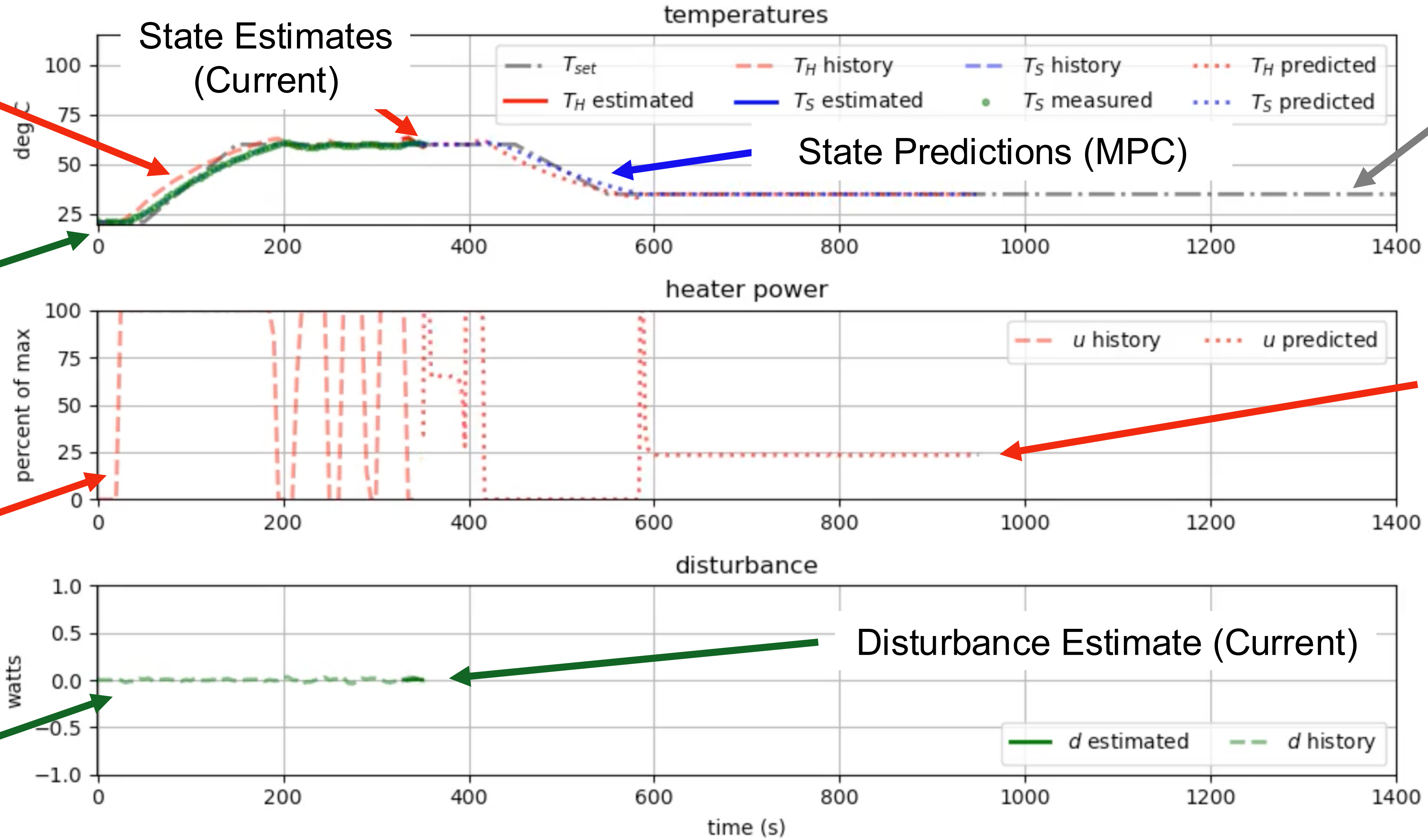
Optimal control profile



T_{h1} matches the setpoint using optimal control



Lab 6: Digital Twins (MPC, State Estimation)



State Estimates (History)

State Estimates (Current)

Set Point

State Predictions (MPC)

Sensor Data

Control Signal (MPC, Future Plan)

Control Signal (History)

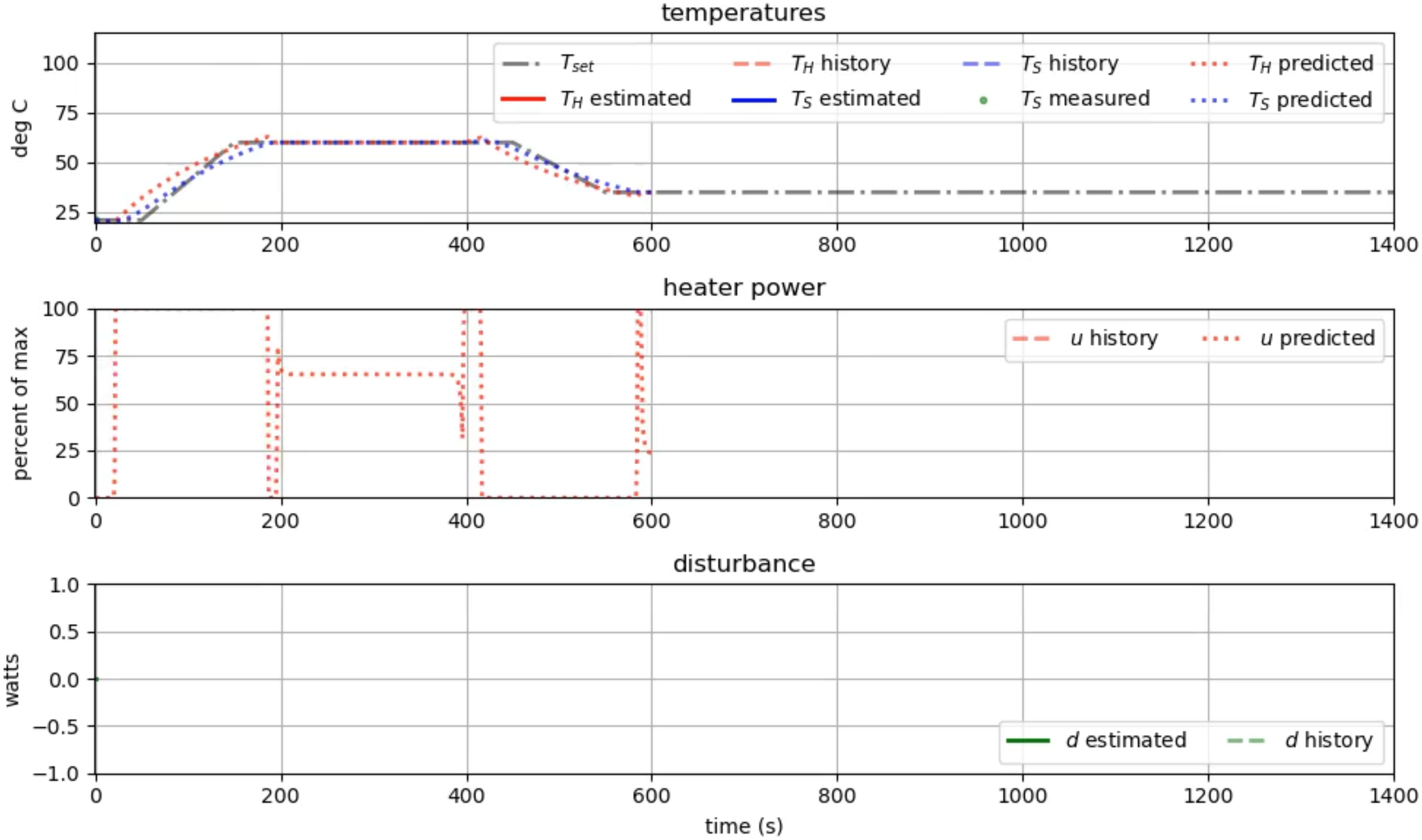
Disturbance Estimate (Current)

Disturbance Estimate (History)

MPC = model predictive control

`ndcbe.github.io/controls/notebooks/06-TCLab-Model-Predictive-Control.html`

Lab 6: Digital Twins (MPC, State Estimation)



Data-Centric Controls Should Be The Most Fun ChE Core Course!

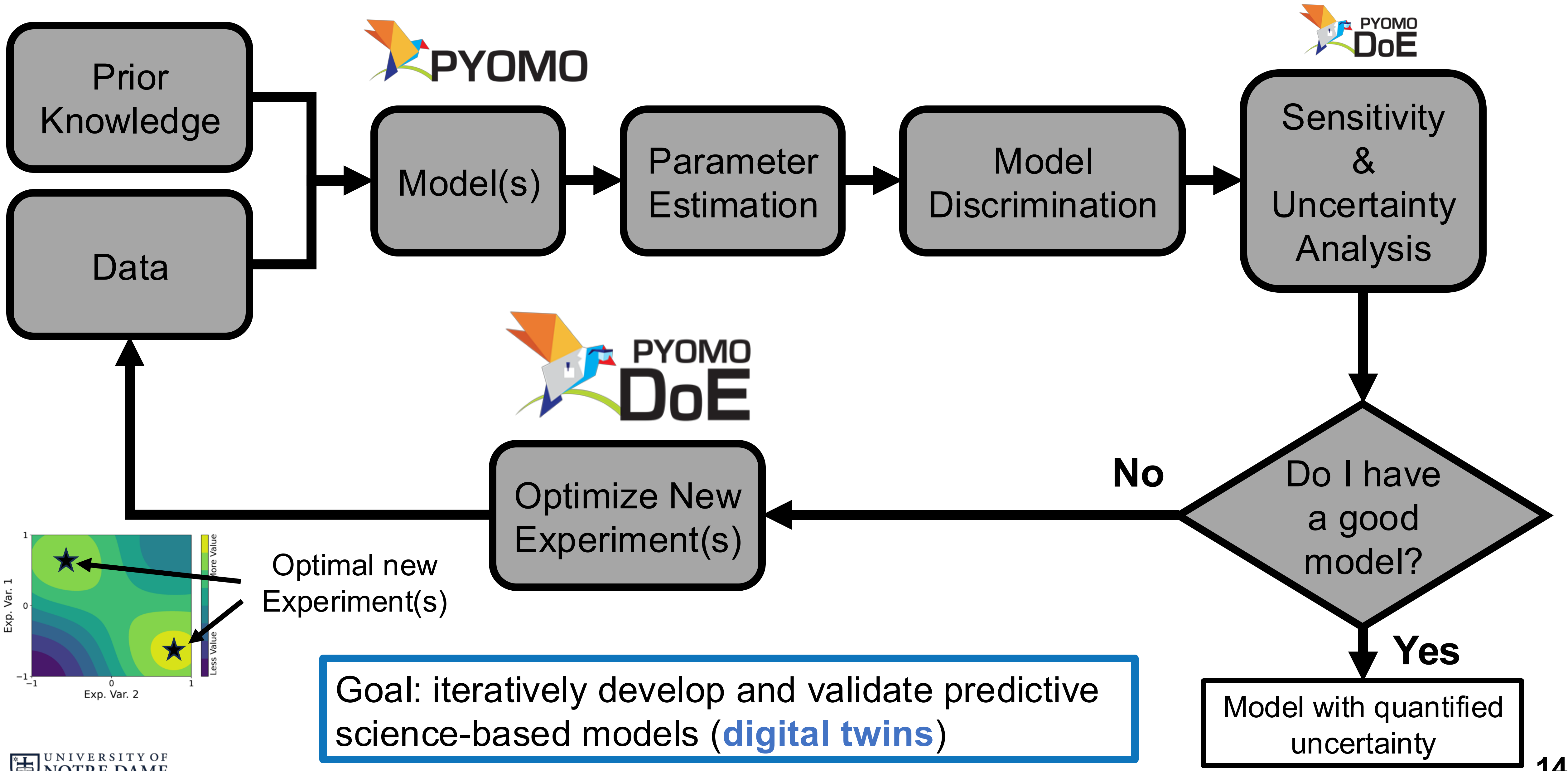
Why this works:

- Hands-on and data-centric
- No Laplace transforms, state-space modeling aligns with ChE core
- Dedicated linear algebra course (sophomore)
- Numerical methods & data analytics in Python (sophomore)
- Homework assignments re-enforce data skills and optimization (decision-making) with Pyomo
- Students get creative with open-ended final project

Ongoing improvements:

- Correlated error structure/lag models
- Four state model (both channels)
- Model-based design of experiments

Science-based (Model-based) Data Analytics Workflow



Goal: iteratively develop and validate predictive science-based models (**digital twins**)

TCLab: Eigendecomposition of the Fisher Information Matrix

ParmEst: `dowlinglab.github.io/pyomo-doe/notebooks/parmest.html`
FIM: `dowlinglab.github.io/pyomo-doe/notebooks/doe_exploratory_analysis.html`

```
FIM:
 [[517225.40941304   1360.01262476 -66404.72541298  -1002.47319402]
 [  1360.01262476   5004.3737258   12379.2662576    5238.40389773]
 [-66404.72541298  12379.2662576   65481.16908635  14190.01468139]
 [ -1002.47319402   5238.40389773  14190.01468139   5526.94375493]]

eigenvalues:
 [5.26802218e+05 6.26035823e+04 3.83207978e+03 1.61037063e-02]

eigenvectors:
 [[-9.89752804e-01 -1.35949591e-01  4.36702406e-02 -7.52086327e-05]
 [ 8.63262440e-04 -2.26164575e-01 -6.85698047e-01 -6.91857665e-01]
 [ 1.42671125e-01 -9.31600001e-01  3.33329462e-01 -2.56487437e-02]
 [ 5.79584008e-03 -2.49977462e-01 -6.45602485e-01  7.21578207e-01]]
```

U_a
U_b
$1/C_p^H$
$1/C_p^S$

U_a
U_b
$1/C_p^H$
$1/C_p^S$

Difficult to uniquely estimate U_b and C_p^S with this single experiment!

TC Lab: D-Optimal Next Experiment

`dowlinglab.github.io/pyomo-doe/notebooks/doe_optimize.html`

$$\max_{u} \quad \log\det(\mathbf{M}(u) + \mathbf{M}_0)$$

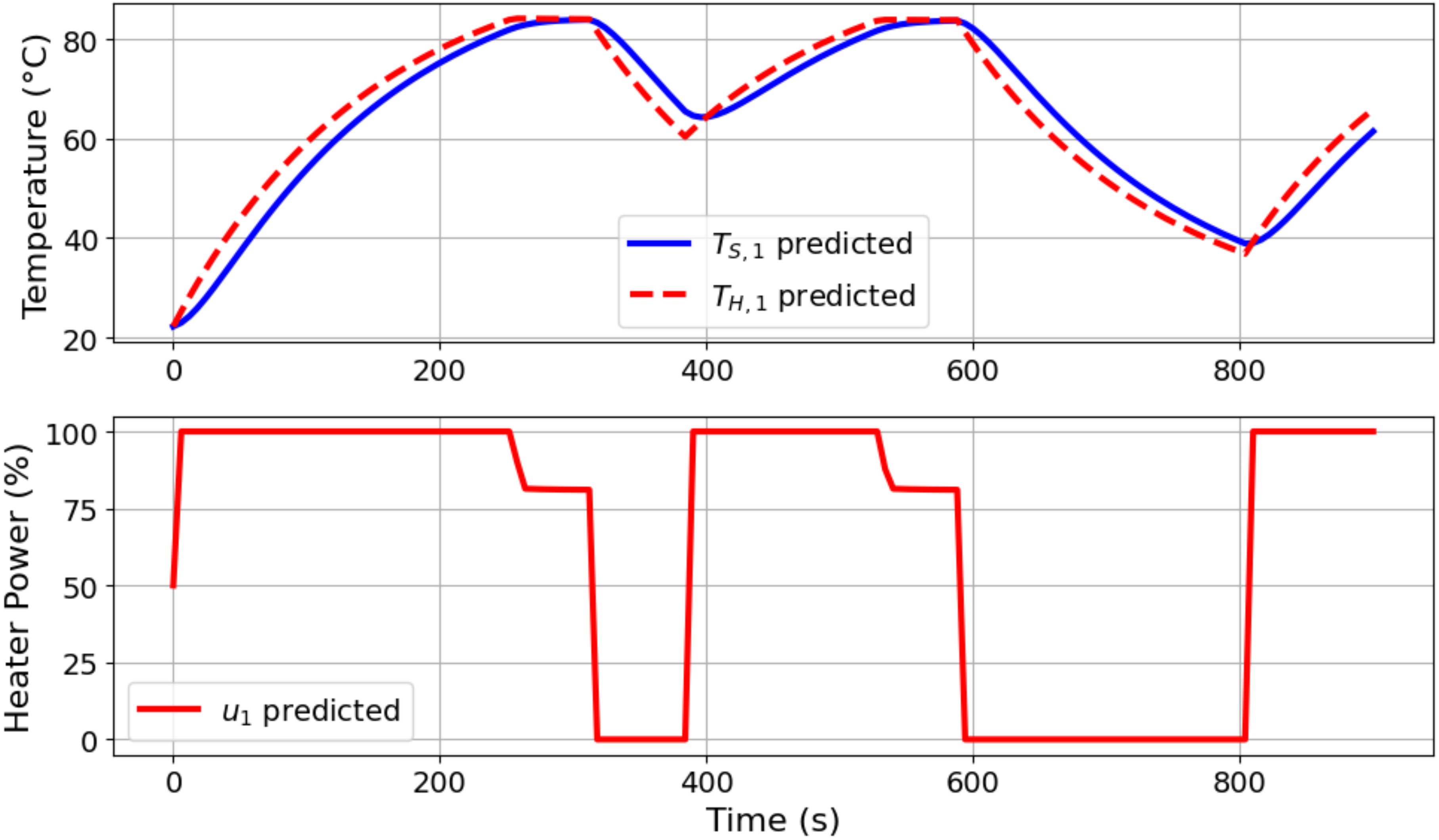
$$\text{s.t.} \quad C_p^H \frac{dT_H}{dt} = \ldots$$

$$C_p^S \frac{dT_S}{dt} = \ldots$$

$$0\% \leq u(t) \leq 100\%$$

$$T_H(t_0) = T_{amb}$$

$$T_S(t_0) = T_{amb}$$



$U_b C_p^S$ is more estimable with two experiments (sine wave test, D-optimal)!