PURDUE UNIVERSITY

Mathematical Optimization and Machine Learning-aided Process Safety Applications

Can Li (Tsan)
Assistant Professor (Joined 2022)
Davidson School of Chemical Engineering

Mathematical Programming under Uncertainty

Deterministic Optimization Model

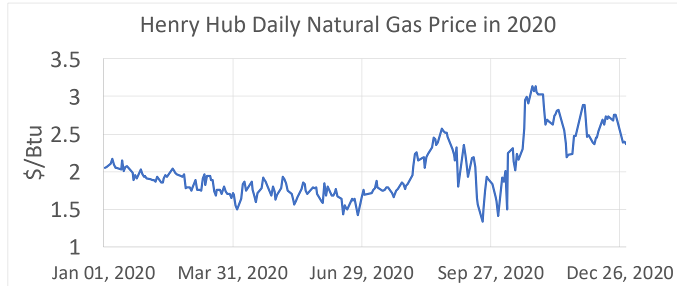
- Model **decision-making** process as an **optimization** problem

$$
\begin{aligned}
\min \quad & f(x, y, \theta) \\
s.t. \quad & g(x, y, \theta) \leq 0 \\
& h(x, y, \theta) = 0 \\
& x \in X, y \in Y
\end{aligned}
$$

- **Variables** continuous x, discrete y the capacity of process (x), whether to install a process or not (y)
- **Constraints** g, h the mass balance, to satisfy the customer demand
- **Objective** f minimize total cost
- **Parameters** θ Product demand, unit cost, thermal and kinetic properties
- The input parameters θ can be uncertain.

Sources of Parameter Uncertainty

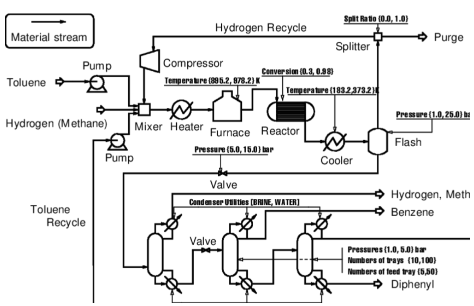
- Long-term forecasts, e.g., natural gas price



- Short-term changing conditions, e.g., extreme weather



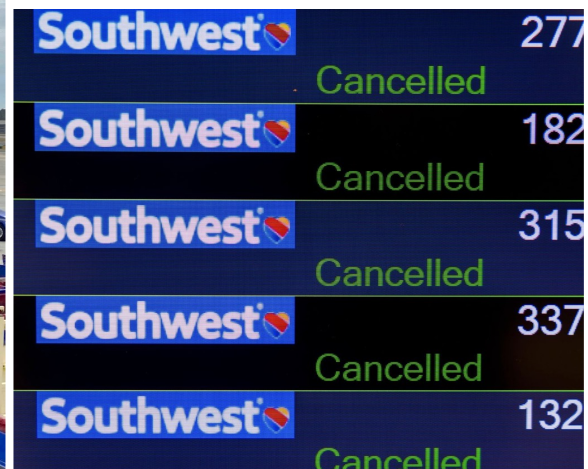
- Real-time inaccurate measurement, e.g., temperature, pressure

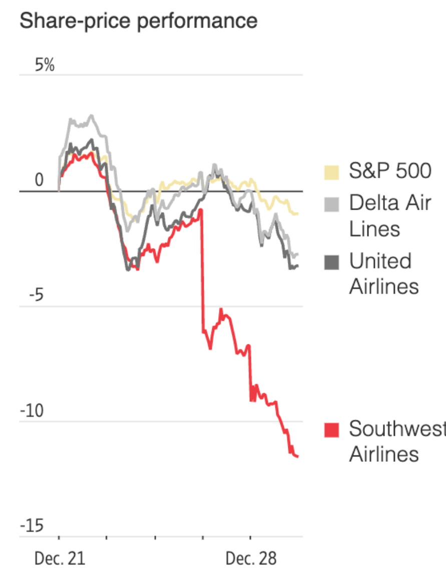


BUSINESS

How Southwest Airlines Melted Down

Dec. 28, 2022
The Wall Street Journal





“Airline executives and labor leaders point to inadequate technology systems, in particular, SkySolver, as one reason why a brutal winter storm turned into a debacle. SkySolver was overwhelmed by the scale of the task of sorting out which pilots and flight attendants could work which flights, Southwest executives said. ”

How Do We Model Uncertainty in Optimization Problems?

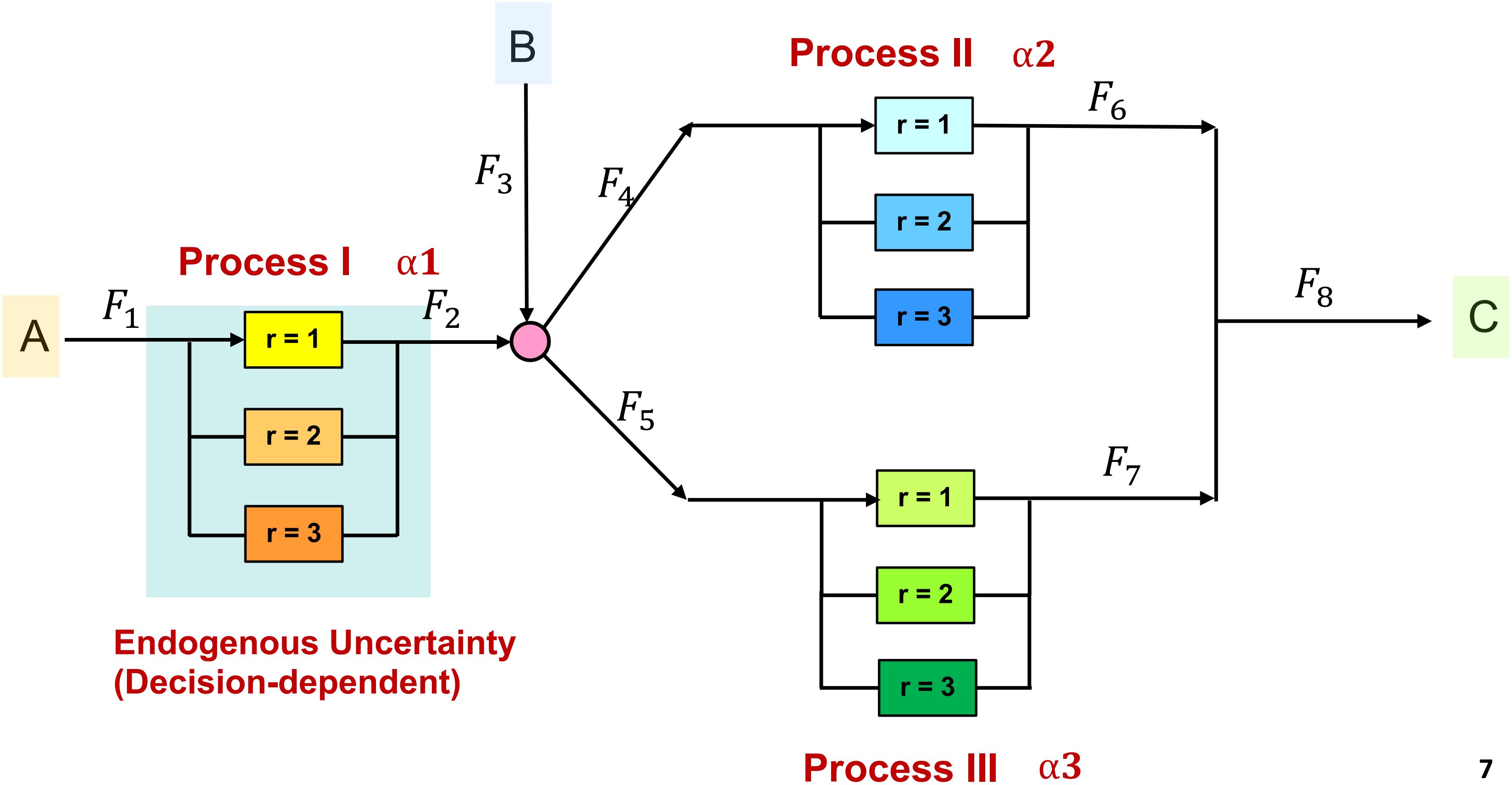
- Not a uniquely-defined problem
  - **Multiple** ways to hedge against uncertainty/risk

The jungle of stochastic optimization

(credit: Warren Powell)

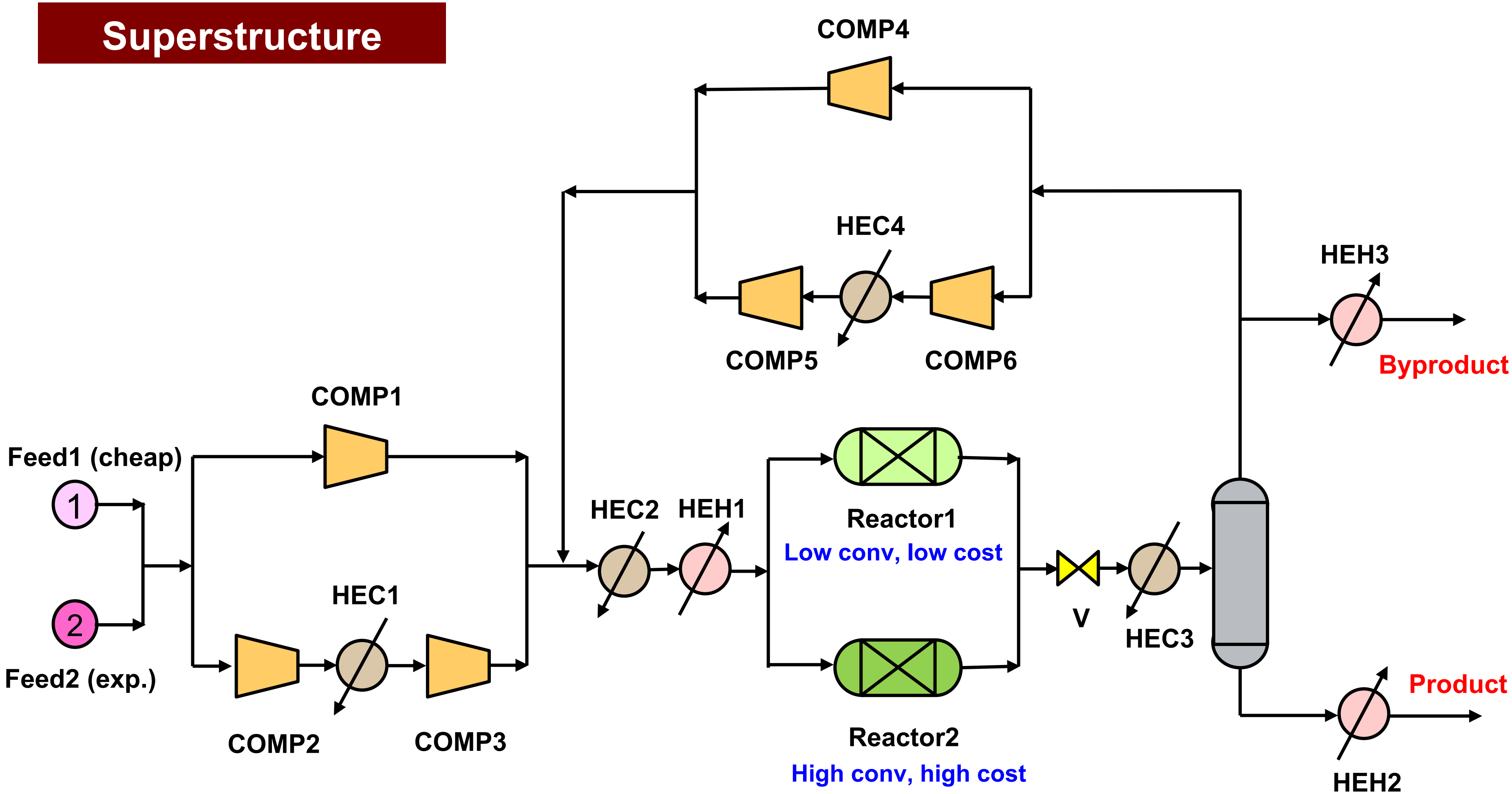
Risk-based Process Design

- Motivation: Process units may fail.
- Solution: Have backup units to improve reliability
- Trade-off: Investment cost v.s. system reliability. How many units should we install?



Industrial Methanol Synthesis

Superstructure

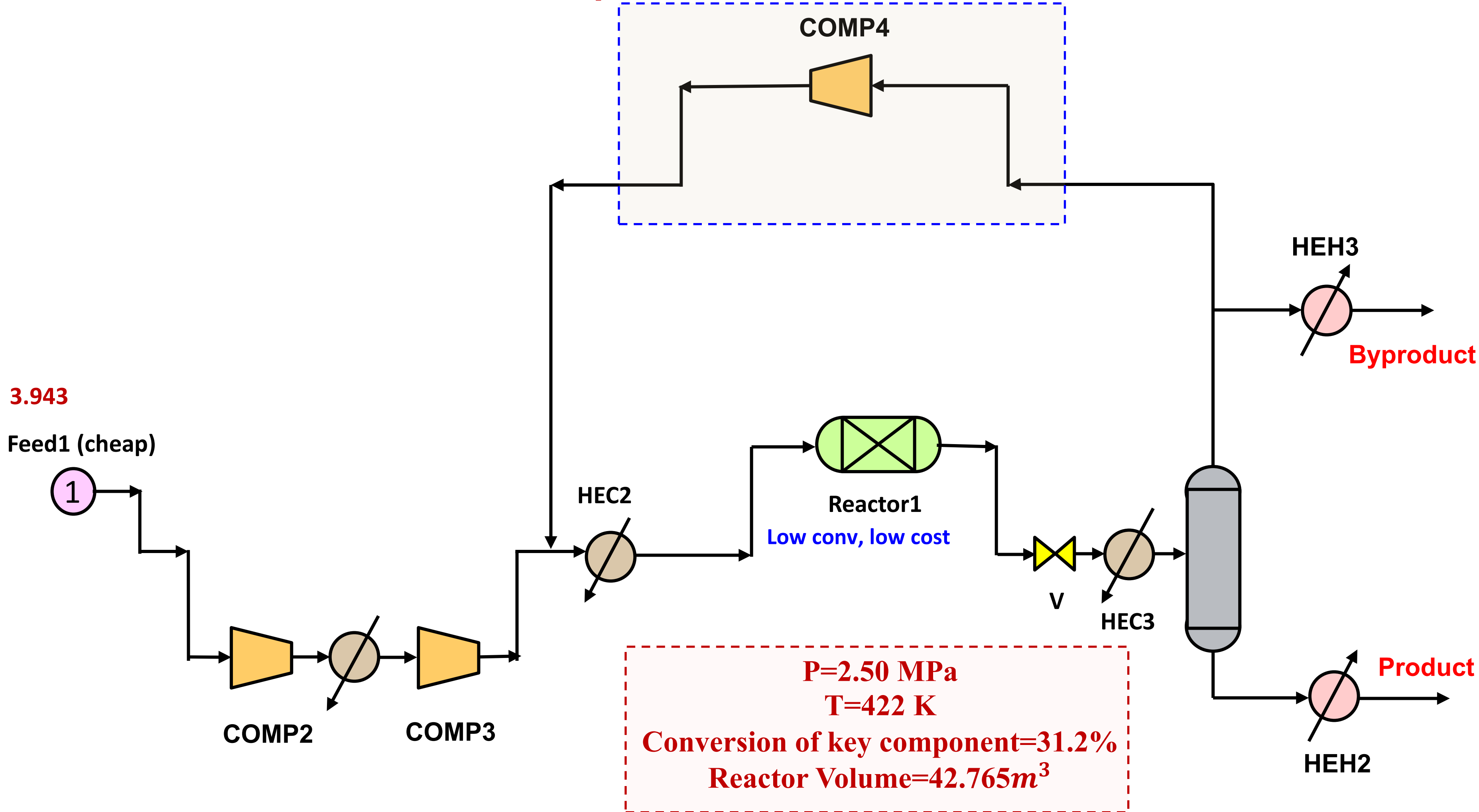


Key optimization variables in the reactors:
operating pressure and the conversion per pass

Industrial Methanol Synthesis

Deterministic Model

Profit = 4115.3749 ($1000 PER YEAR)
Too optimistic

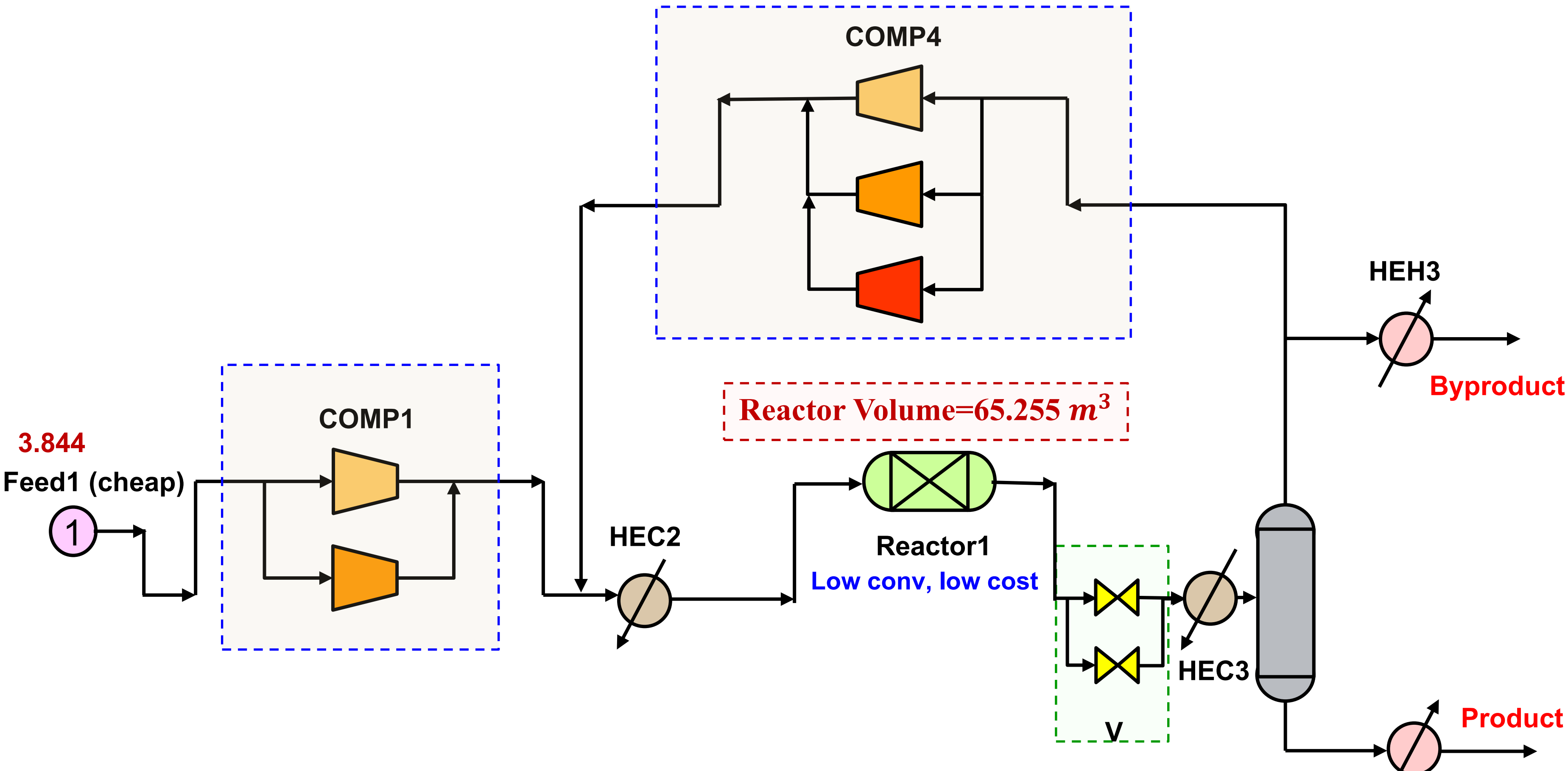


Industrial Methanol Synthesis

Stochastic Programming Model

Expected Profit = 3203.6879 ($1000 PER YEAR)

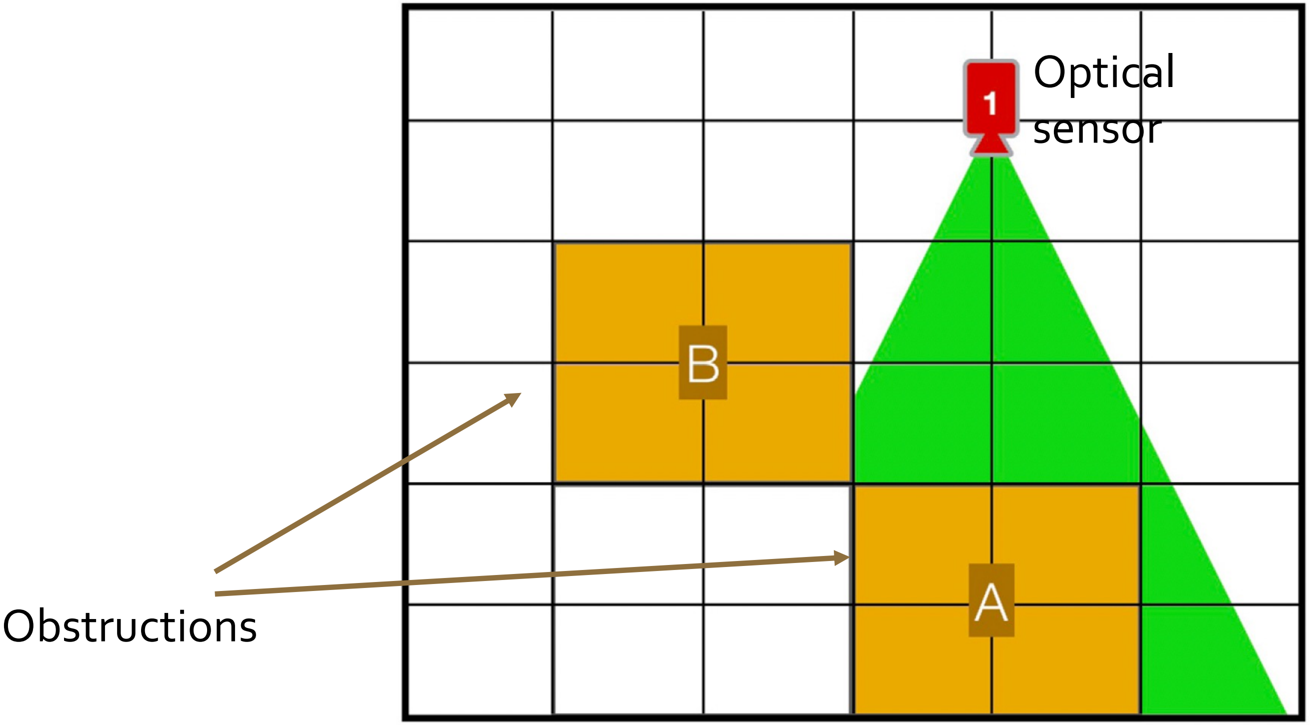
Consider demand uncertainty and reliability Simultaneously



Value of Reliable solution: 5.6% of the expected profit

Sensor Placement under Uncertainty

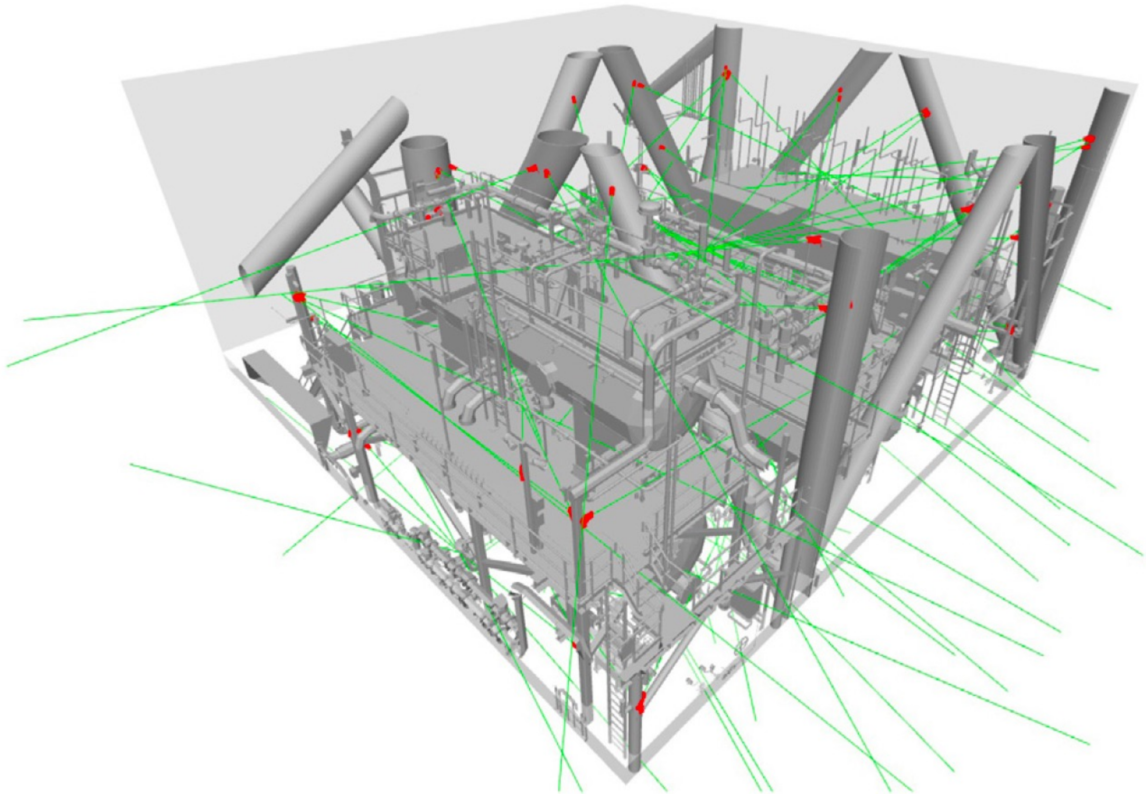
- Motivation: Determine the optimal configurations of sensors to maximize the probability of detecting safety hazards
- Flame, smoke, and heat detectors using chemical or optical sensors



Work of Prof. Carl Laird with P2SAC

Sensor Placement under Uncertainty

- Facility with 81 candidate flame detector locations (Kenexis Consulting Corporation)



T. Zhen, K.A. Klise, S. Cunningham et al. / Process Safety and Environmental Protection 132 (2019) 47–58

Mathematical Optimization for Sensor Placement

- Mixed-integer nonlinear programming (MINLP) formulation

Maximize expected coverage

$$\underset{x,\sigma}{\text{maximize}} \quad \sum_{e \in E} \sigma_e w_e$$

$$\text{subject to} \quad \sum_{l \in L} x_l \leq k$$

Place at most k sensors

$$\sigma_e = 1 - 1[\prod_{l \in L_e} (1 - p_{l,e} x_l)] \qquad \forall e \in E$$

expected coverage of entity

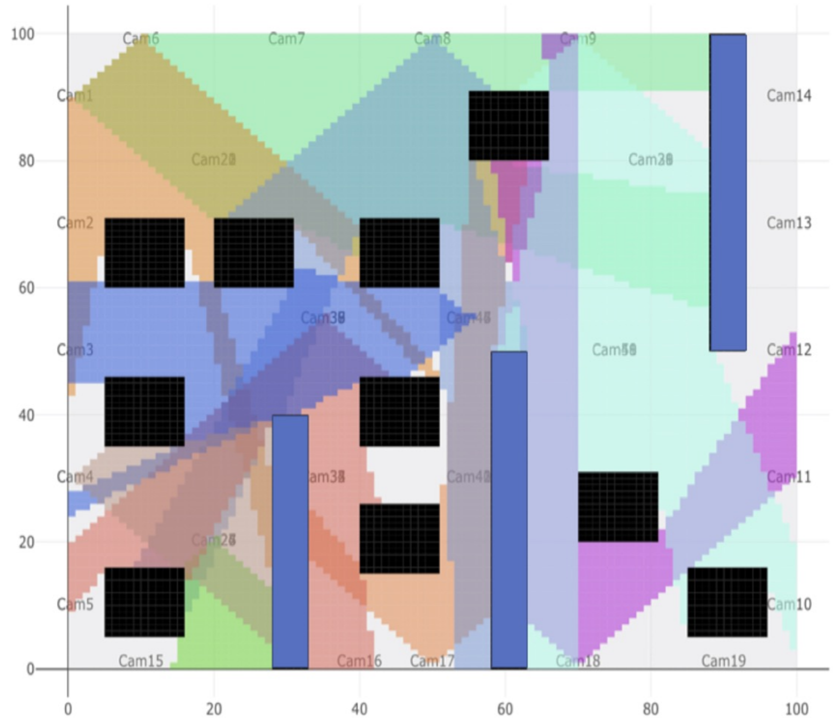
$$x_l \in \{0, 1\} \qquad \forall l \in L$$

$$0 \leq \sigma_e \leq 1 \qquad \forall e \in E$$

Binary variable, whether to place a sensor at location l

Sensor Placement under Uncertainty

- Facility with 81 candidate flame detector locations (Kenexis Consulting Corporation).
- Find the optimal configuration within 2 hours with a tailored algorithm

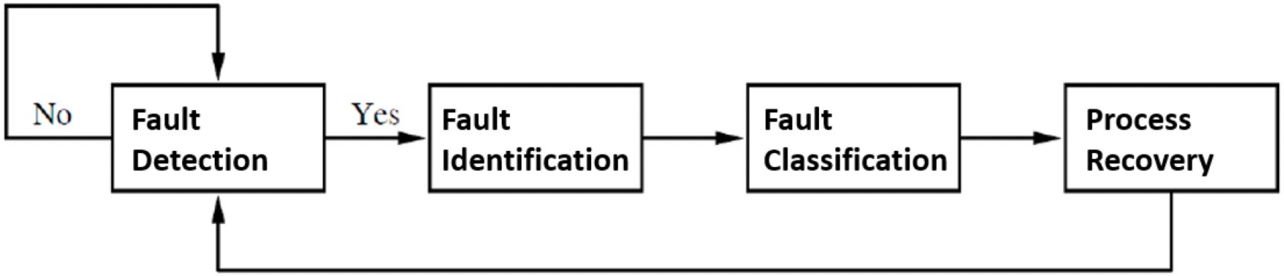


Optimal placement with $k = 10$

T. Zhen, K.A. Klise, S. Cunningham et al. / Process Safety and Environmental Protection 132 (2019) 47–58

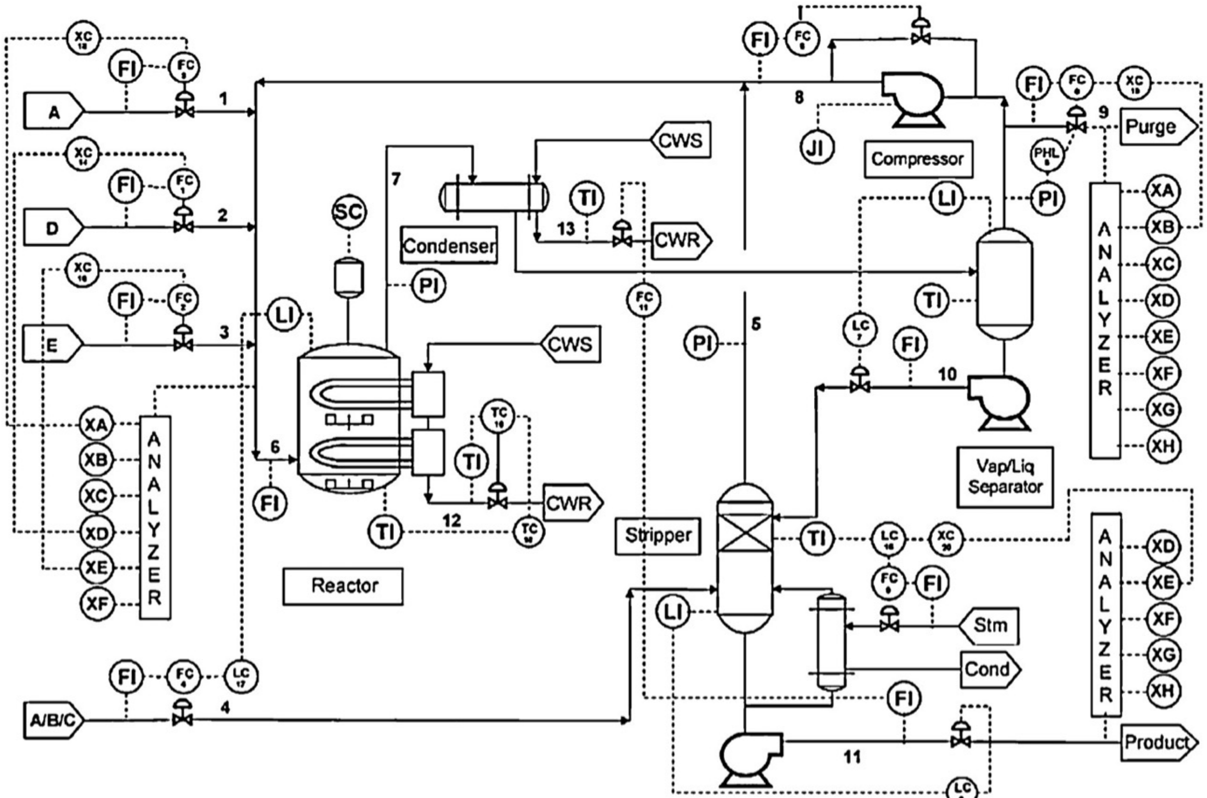
Machine Learning for Process Monitoring

- Fault **Detection**: Detect if a fault has occurred
- Fault **Identification**: Identify the variables most relevant to the fault
- Fault **Diagnosis** (or Classification): Diagnose the root cause of the fault



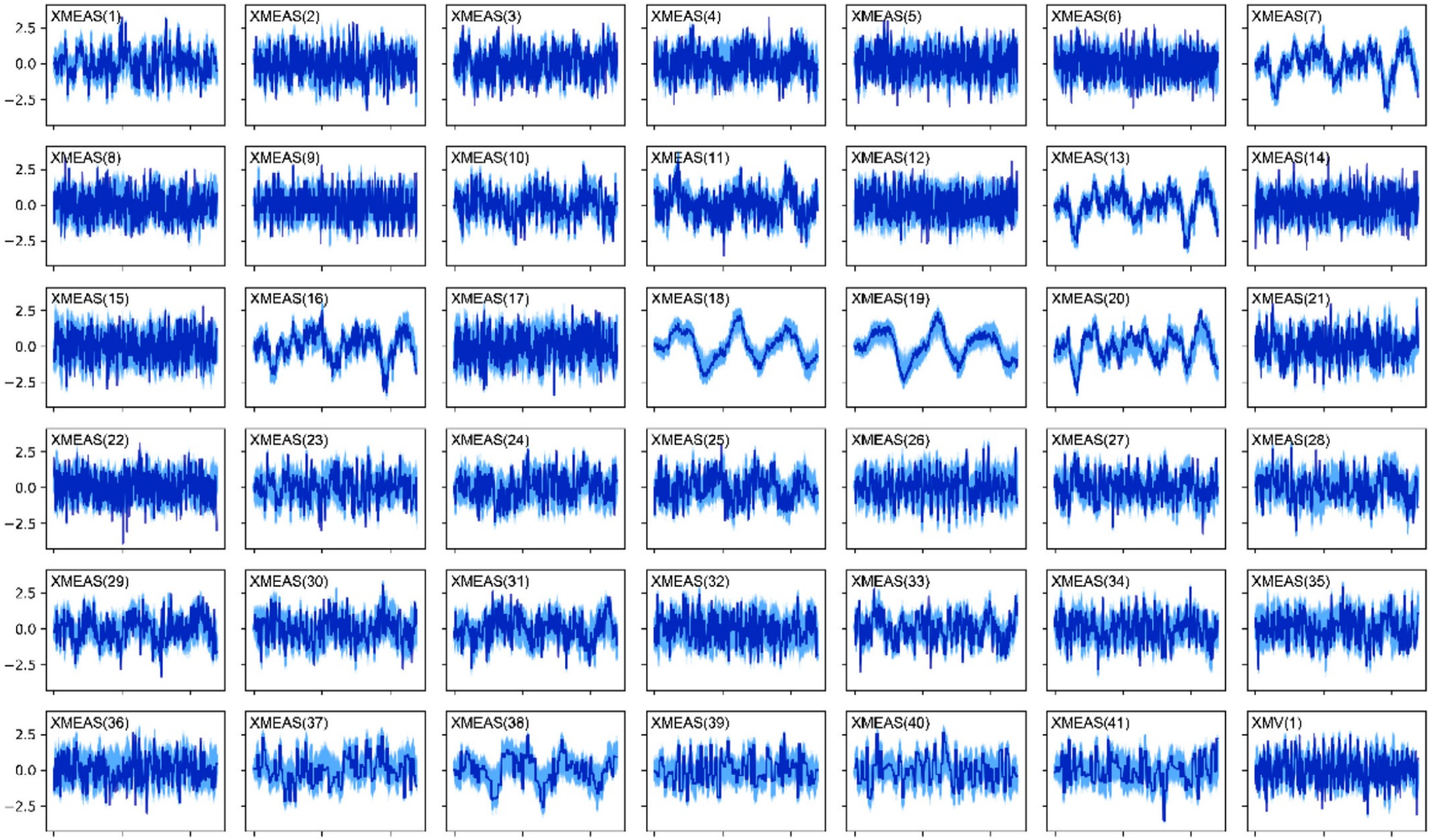
Tennessee Eastman process

- TEP is an open-source simulator written in Fortran that resembles a real chemical process by Eastman
- Time series data can be collected from over 40 sensors that measure the state variables.
- Task: From measured state variables, perform fault detection using ML/AI



Examples of State Variables with Sensor Data

- Examples include feed flow rates, temperatures, pressures



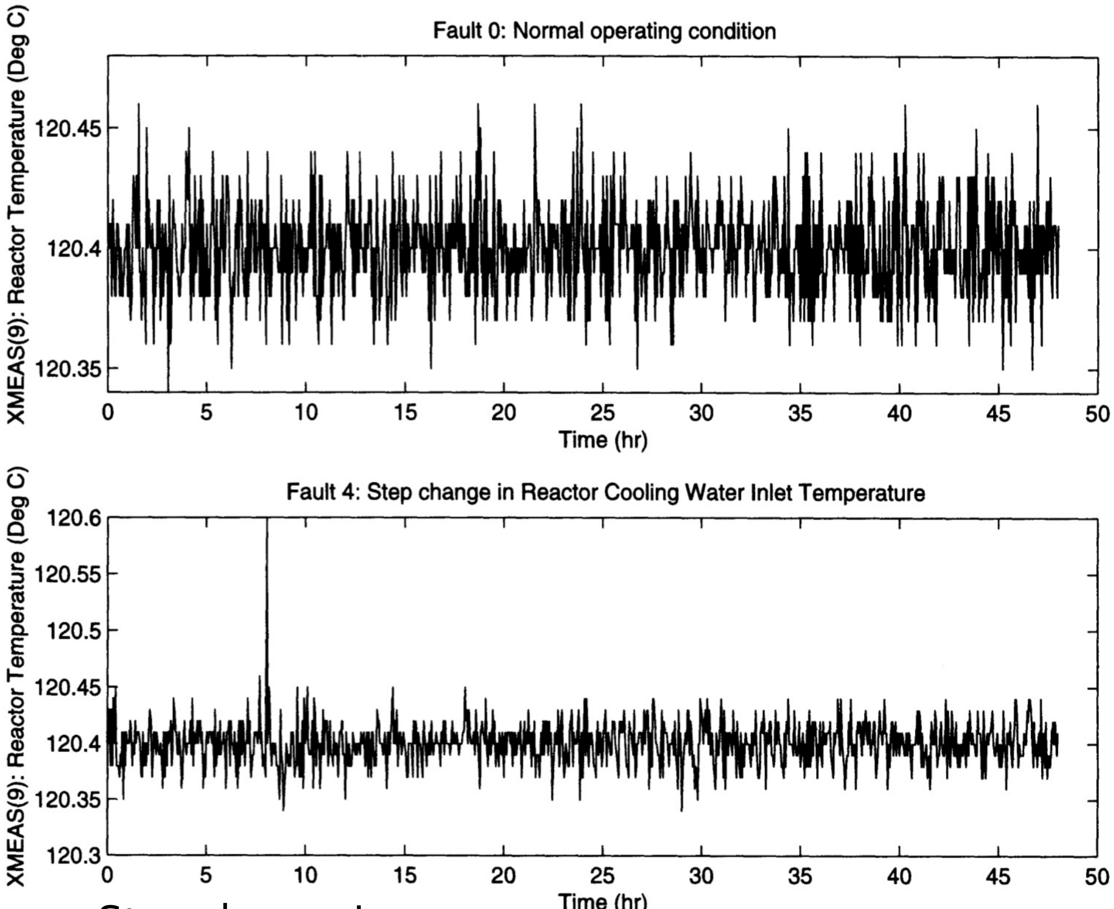
List of Potential Faults

- The following "faults" are created synthetically by the simulator
- These faults will cause the measured state variables to change from their normal operating conditions which further cause safety hazards.

Variable number	Process variable	Type
IDV (1)	A/C feed ratio, B composition constant (stream 4)	Step
IDV (2)	B composition, A/C ratio constant (stream 4)	Step
IDV (3)	D feed temperature (stream 2)	Step
IDV (4)	Reactor cooling water inlet temperature	Step
IDV (5)	Condenser cooling water inlet temperature	Step
IDV (6)	A feed loss (stream 1)	Step
IDV (7)	C header pressure loss—reduced availability (stream 4)	Step
IDV (8)	A, B, C feed composition (stream 4)	Random variation
IDV (9)	D feed temperature (stream 2)	Random variation
IDV (10)	C feed temperature (stream 4)	Random variation
IDV (11)	Reactor cooling water inlet temperature	Random variation
IDV (12)	Condenser cooling water inlet temperature	Random variation
IDV (13)	Reaction kinetics	Slow drift
IDV (14)	Reactor cooling water valve	Sticking
IDV (15)	Condensor cooling water valve	Sticking
IDV (16)	Unknown	Unknown
IDV (17)	Unknown	Unknown
IDV (18)	Unknown	Unknown
IDV (19)	Unknown	Unknown
IDV (20)	Unknown	Unknown

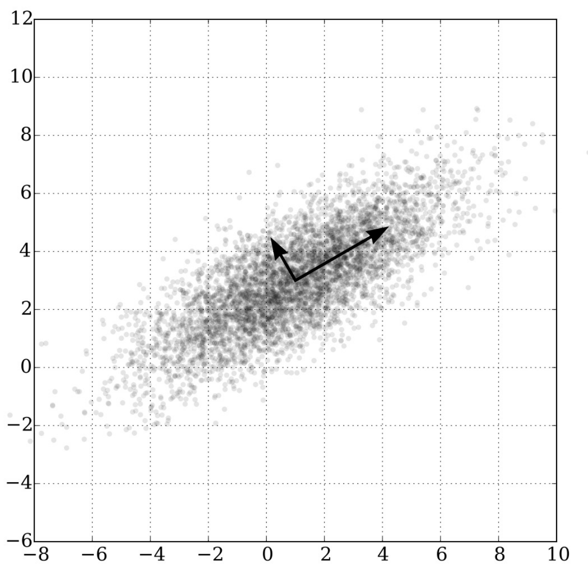
Step change in reactor cooling water temperature

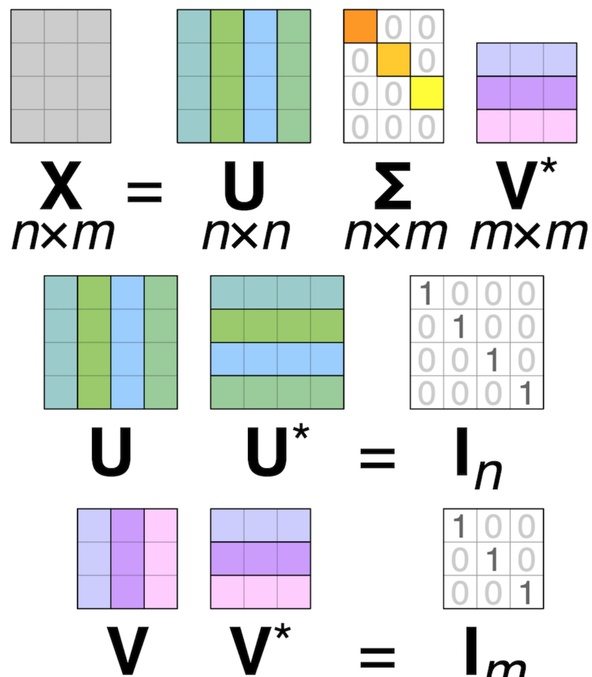
- This fault could cause runaway reaction. The controller will increase the cooling water flowrate to bring the temperature down



Step change in
reactor temperature

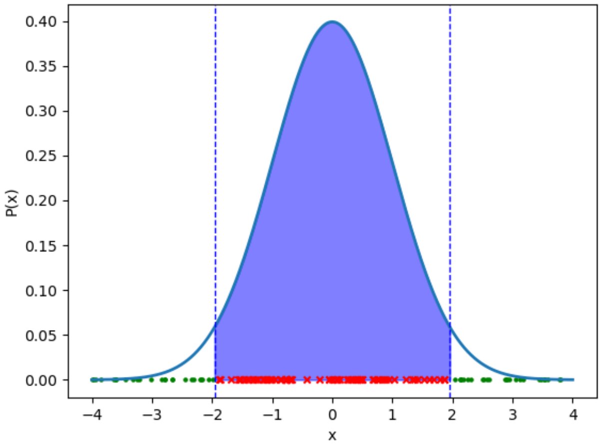
- **Principal component analysis:** identify the principal components where the data have the largest variance. The non-principal components are "noise".
- Approach: singular value decomposition



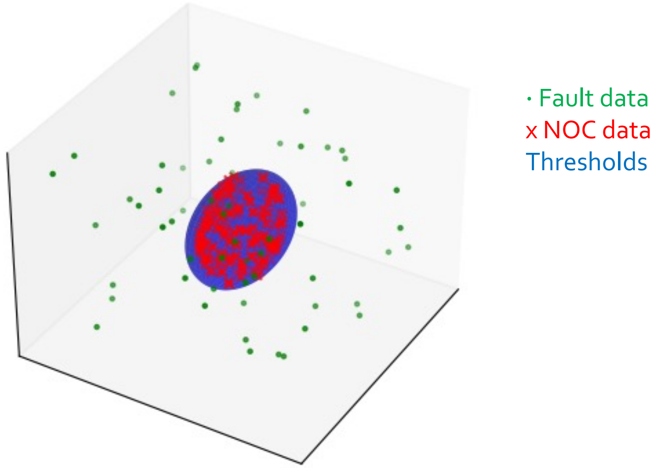


Principal component analysis

- The region within the thresholds represents the Normal Operating Condition (NOC) under random noise.
- The region outside of the thresholds represents the systematic variation from NOC.



1-dim



3-dim

PCA applied on TEP dataset

- We implemented PCA algorithm applied to TEP data set in Python.
- PCA works well on linearly correlated variables.
- Achieve a fault detection rate of almost 90%,

i.e., 90% of the faults are detected by the algorithm.

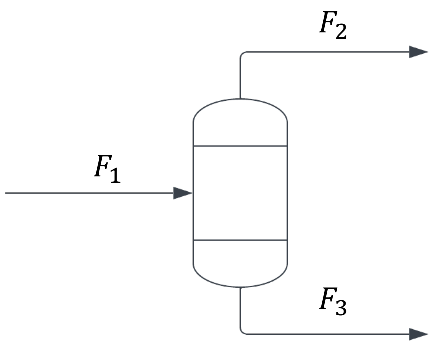
Fault	FDR %	FPR %
1	99.88	0.0
2	98.88	0.0
3	26.88	4.37
4	100.0	0.62
5	100.0	0.62
6	100.0	0.0
7	100.0	0.0
8	98.5	0.0
9	20.88	5.0
10	94.5	0.0
11	83.0	0.0
12	99.88	0.62
13	95.62	0.0
14	100.0	0.0
15	40.75	0.0
16	96.25	1.88
17	96.88	0.62
18	92.12	0.0
19	93.75	0.0
20	91.88	0.0
21	73.88	0.0

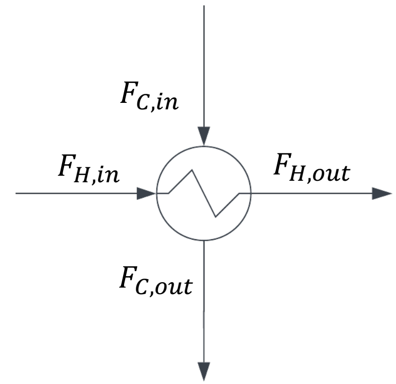
Work by PhD student Hao Chen

FDR%: Fault Detection Rate; FPR %: False Positive Rate.

Limitations of PCA

- PCA works well on linearly correlated variables.

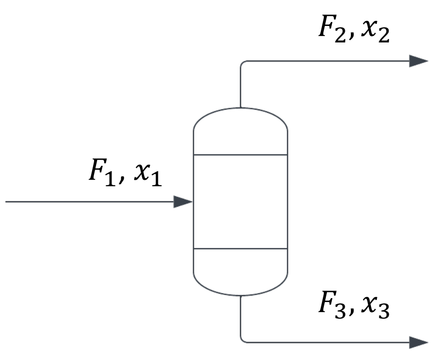


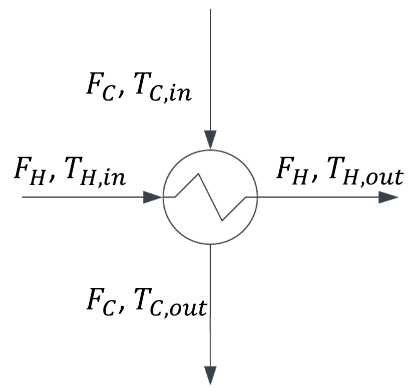


$$F_1 = F_2 + F_3$$

$$F_{H,in} = F_{H,out} \qquad F_{H,in} = F_{H,out}$$

- But chemical processes, such as flash units and heat exchangers, involve variables that are nonlinearly correlated



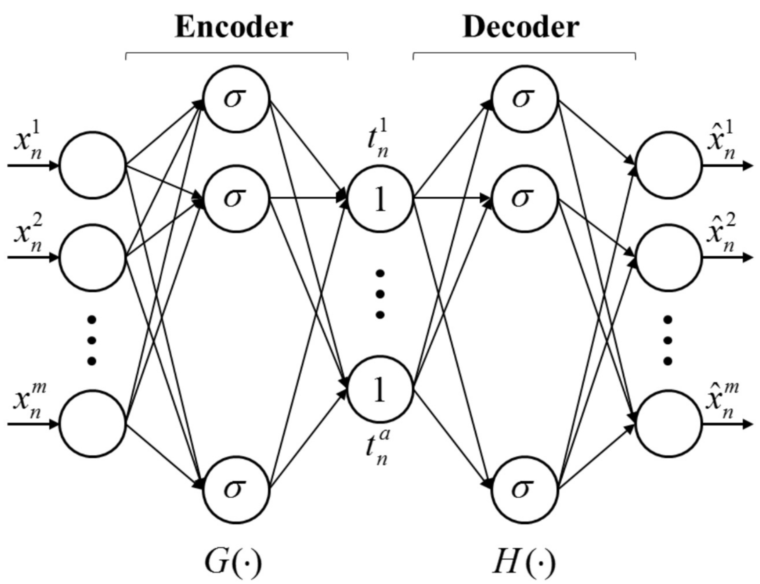


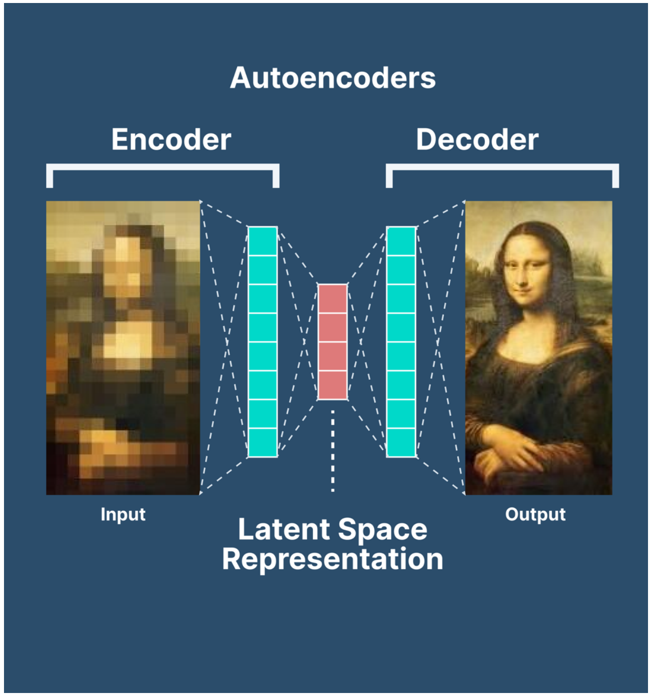
$$F_1 x_1 = F_2 x_2 + F_3 x_3$$

$$F_H c_{p,H}\left(T_{H,in} - T_{H,out}\right) = F_C c_{p,C}\left(T_{C,out} - T_{C,in}\right)$$

Deep learning methods

- Autoencoder: utilize the artificial neural network to capture the nonlinearity among variables and map to lower dimensional representations.
- Wide successful applications of autoencoder in tasks such as image reconstruction.
- Capture more complex patterns and better suited for various input data





Comparison of PCA & autoencoder results

- Implementation of autoencoder in Python using the Pytorch library.
- No significant difference between PCA and autoencoder due to the linearity of TEP data. We expect better performance of autoencoder than PCA on real industrial data such as data from refineries.

PyTorch

PCA

Fault	FDR %	FPR %
1	99.88	0.0
2	98.88	0.0
3	26.88	4.37
4	100.0	0.62
5	100.0	0.62
6	100.0	0.0
7	100.0	0.0
8	98.5	0.0
9	20.88	5.0
10	94.5	0.0
11	83.0	0.0
12	99.88	0.62
13	95.62	0.0
14	100.0	0.0
15	40.75	0.0
16	96.25	1.88
17	96.88	0.62
18	92.12	0.0
19	93.75	0.0
20	91.88	0.0
21	73.88	0.0

FDR%: 85.88% FPR %: 0.65%

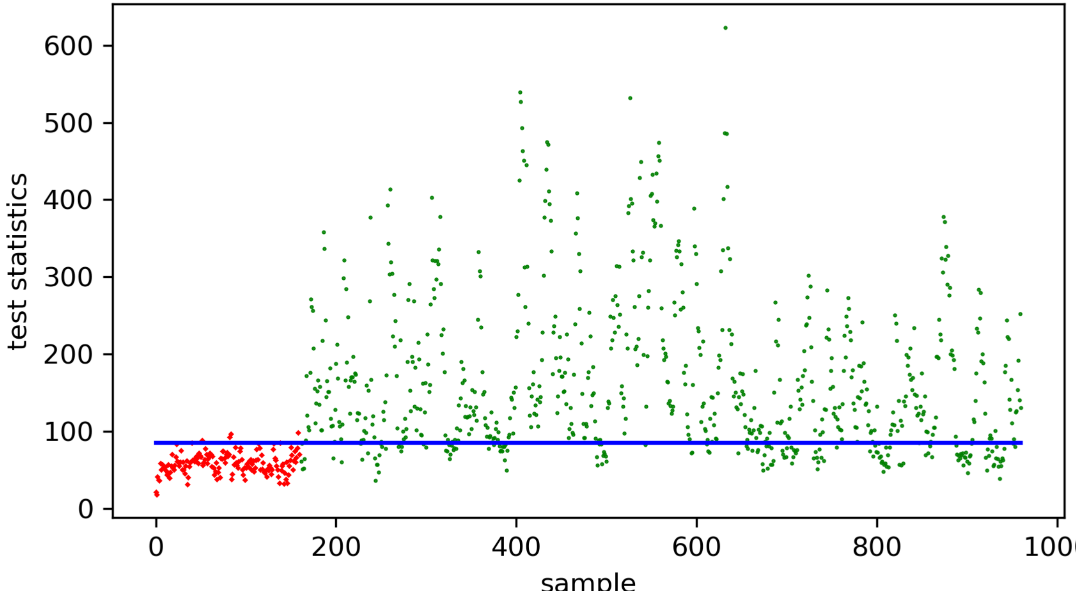
Autoencoder

Fault	FDR %	FPR %
1	100.0	0.0
2	99.5	1.25
3	41.88	0.62
4	100.0	0.0
5	100.0	0.0
6	100.0	0.0
7	100.0	0.0
8	99.25	0.0
9	40.5	2.5
10	87.12	0.62
11	91.38	0.0
12	99.75	0.0
13	96.88	0.0
14	100.0	0.0
15	41.0	0.0
16	88.0	3.75
17	98.88	0.0
18	94.5	0.0
19	78.75	0.0
20	86.38	0.0
21	76.62	1.25

FDR%: 86.68%; FPR %: 0.48%

Case Study

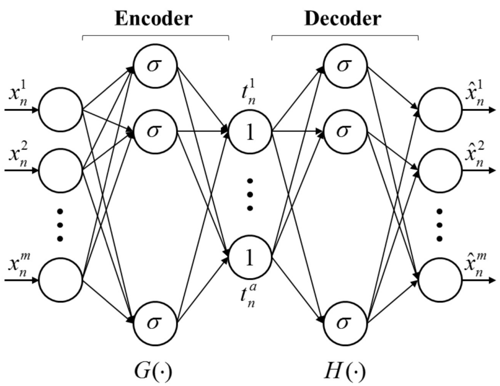
- Fault 11 (random variation in reactor cooling water inlet temperature)
- Oscillations in reactor temperature and cooling water flow rate
- Prevent runaway reaction



Future research plan

- Improve the explainability of machine learning methods

  deep learning-based models such as
  autoencoder and recurrent neural network



Computationally efficient to use online
Hard to interpret

- Develop machine learning models based on open-source Python libraries, such as Pytorch, scikit-learn. Made them open-source for P2SAC sponsors.

- Look for collaborations with industry to study real-world datasets, e.g., digital twins, data lakes.