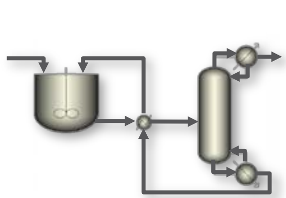
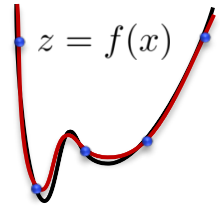




$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g(x) = 0 \end{array}$$

Learning process models from simulations

Alison Cozad[1], Nick Sahinidis[1,2], and David Miller[2]

[1] Department of Chemical Engineering, Carnegie Mellon University
[2] National Energy and Technology Laboratory

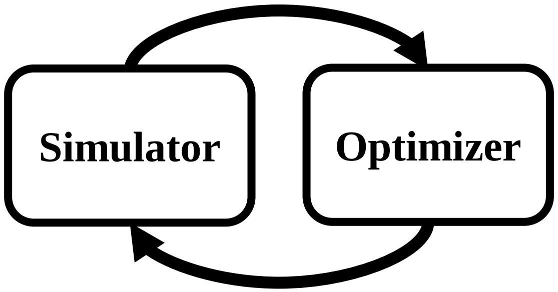
PROBLEM STATEMENT

$$
\begin{aligned}
\min \quad & f(x) \\
\text{s.t.} \quad & g(x) = 0 \\
& x^l \leq x \leq x^u
\end{aligned}
$$

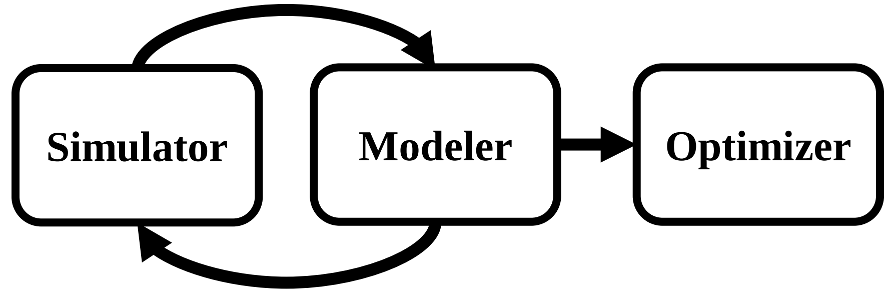
- **Challenges:**
  - **Lack of an algebraic model**
  - **Computationally costly simulations**
  - **Often noisy function evaluations**
  - **Scarcity of fully robust simulations**

SIMULATION-BASED METHODS

Direct methods



Indirect methods

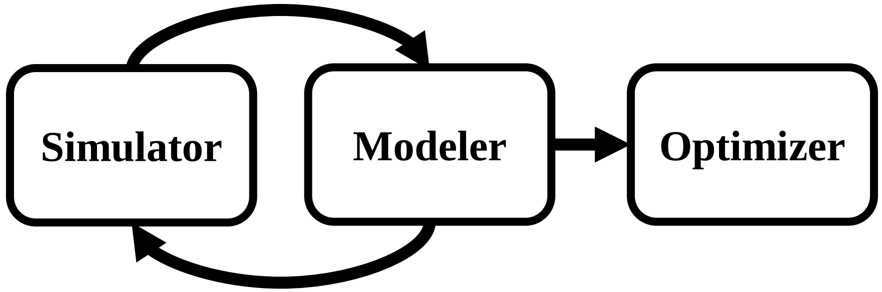


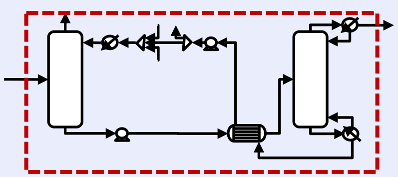
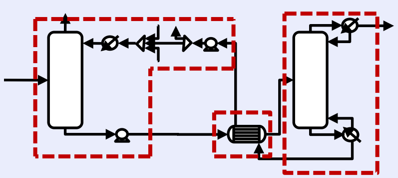
- Estimated gradient based
  - Finite element, perturbation analysis, etc.
- Derivative-free optimization (DFO)
  - Local/global
  - Stochastic/deterministic

- What is modeled?
  - Objective, objective + constraints, disaggregated system
- Type of model
  - Linear/nonlinear
  - Simple/Complex
  - Algebraic/black-box
- Optimizer
  - Derivative/derivative-free

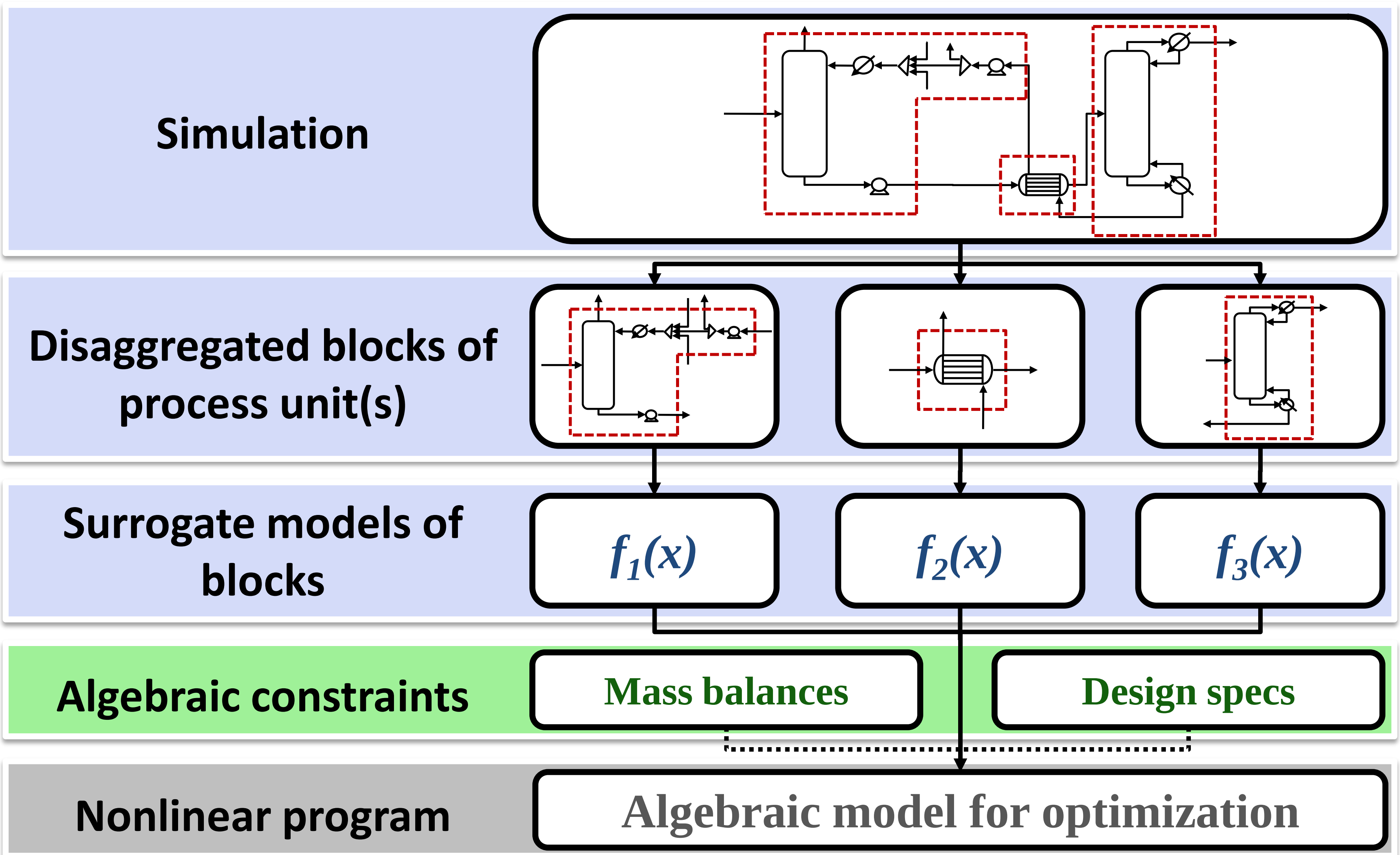
RECENT WORK IN CHEMICAL ENG

Indirect methods



	Kriging	Neural nets	Other
Full process 	▪ Palmer and Realff, 2002 ▪ Huang, et. al., 2006 ▪ Davis and Ierapetriton, 2012	▪ Michalopoulos, et. Al., 2001	▪ Palmer and Realff, 2002
Disaggregated 	▪ Caballero and Grossmann, 2008	▪ Henao and Maravelias, 2011	

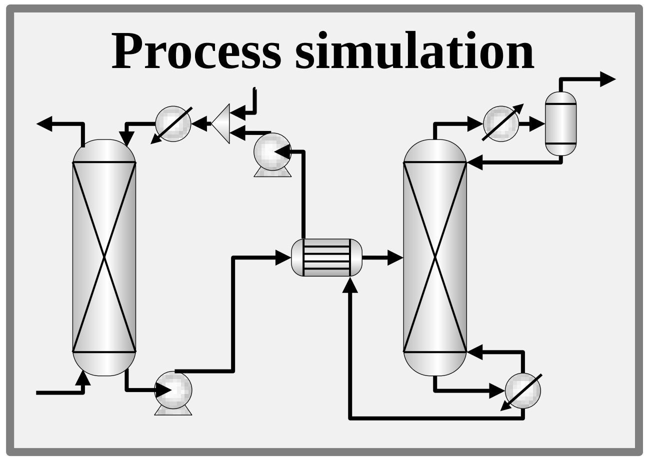
PROCESS DISAGGREGATION



MODELING PROBLEM STATEMENT

- **Build a model of output variables *z* as a function of input variables *x* over a specified interval**

$$x \in \mathbb{R}^D \qquad x^l \leq x \leq x^u \qquad \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \\ \vdots \\ x_D \end{pmatrix}$$



$$\begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_k \\ \vdots \\ z_K \end{pmatrix} \qquad z \in \mathbb{R}^K \qquad z = f(x)$$

Independent variables:
Operating conditions, inlet flow properties, unit geometry

Dependent variables:
Efficiency, outlet flow conditions, conversions, heat flow, etc.

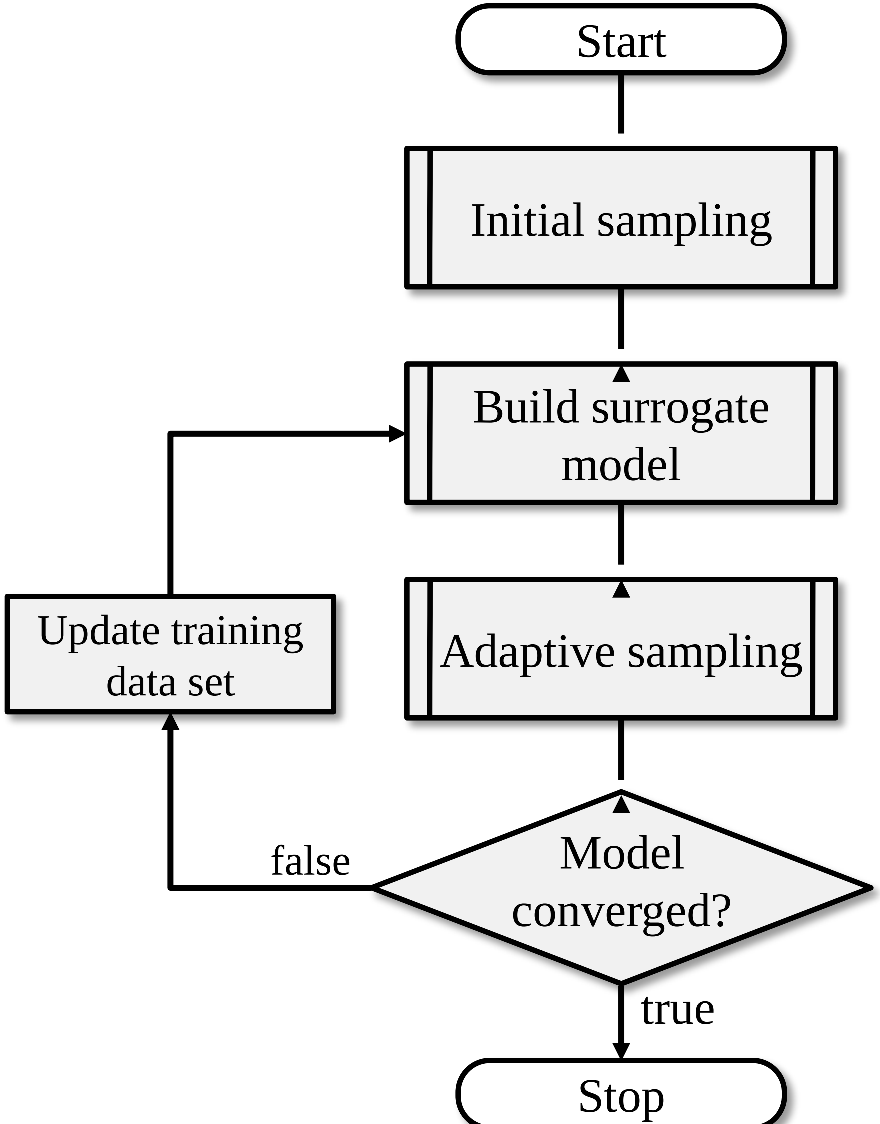
MODELING PROBLEM STATEMENT

- **Model questions:**
  - What is the functional form of the model?
  - How complex of a model is needed?
  - Will this be tractable in an algebraic optimization framework?

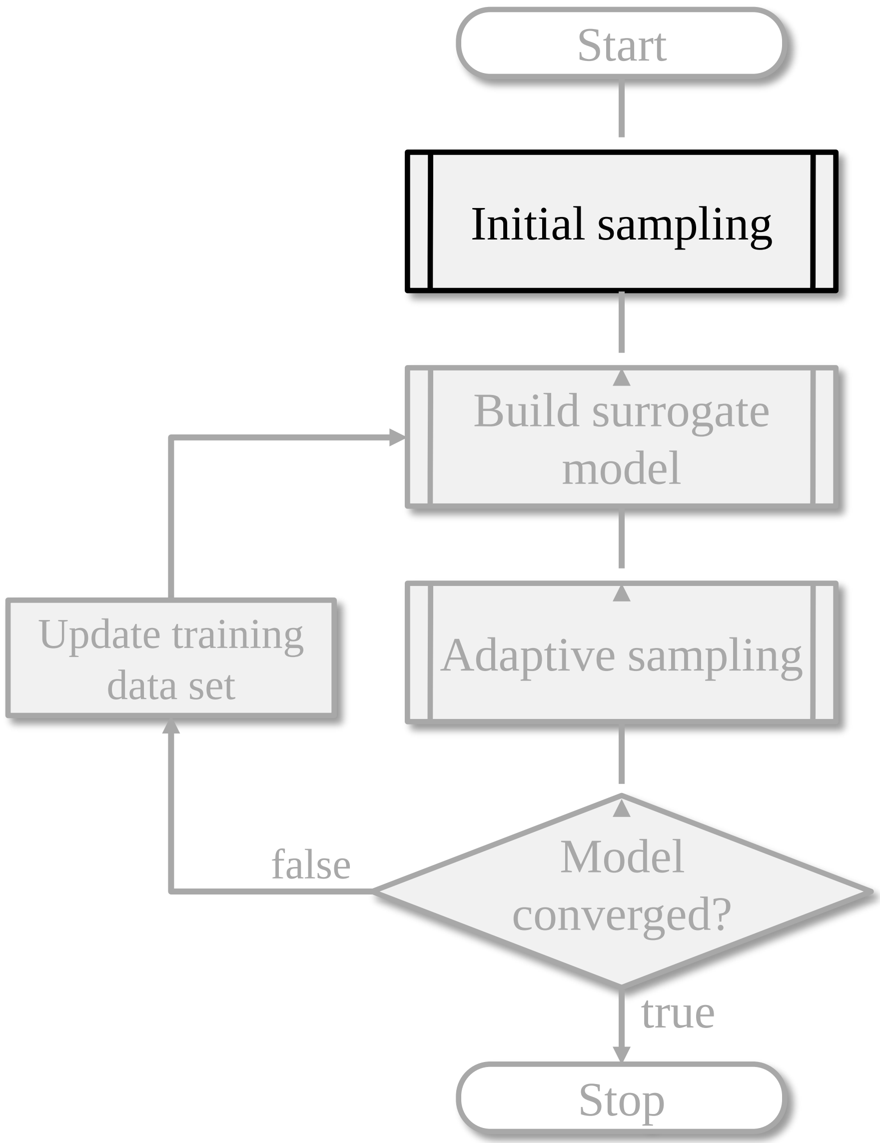
- **Sampling questions:**
  - How many sample points are needed to define an accurate model?
  - Where should these points be sampled?

- **Desired model traits:**
  - ✓ Accurate
  - ✓ Tractable in algebraic optimization: Simple functional forms
  - ✓ Generated from a minimal data set

ALGORITHMIC FLOWSHEET



ALGORITHMIC FLOWSHEET

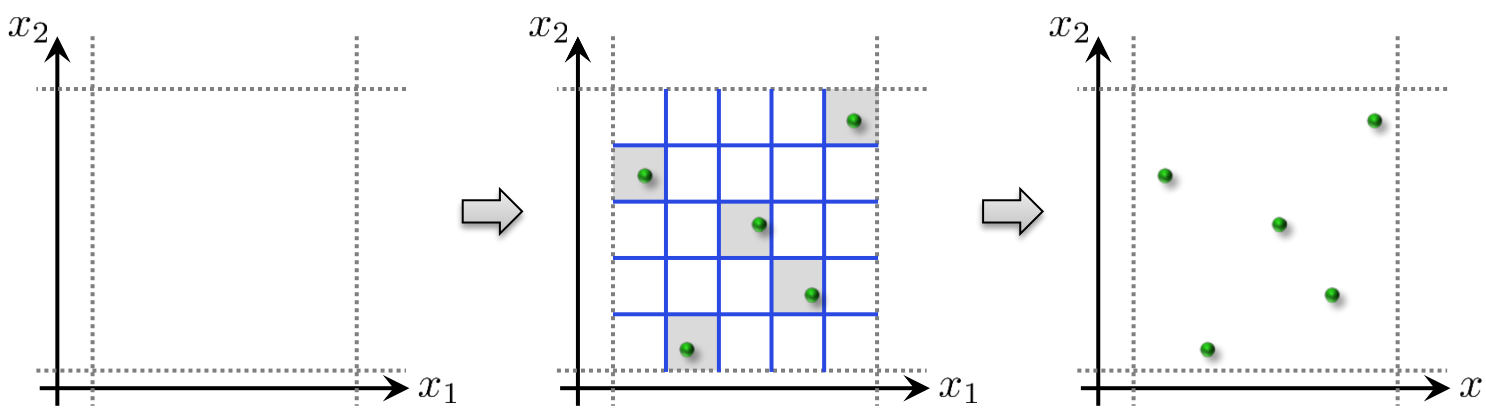


DESIGN OF EXPERIMENTS

- **Goal: To generate an initial set of input variables to evenly sample the problem space**

$$x = \begin{pmatrix} x^1 & x^2 & \cdots & x^i & \cdots & x^N \end{pmatrix} \qquad x^i = \begin{pmatrix} x_1^i \\ x_2^i \\ \vdots \\ x_d^i \\ \vdots \\ x_D^i \end{pmatrix}$$

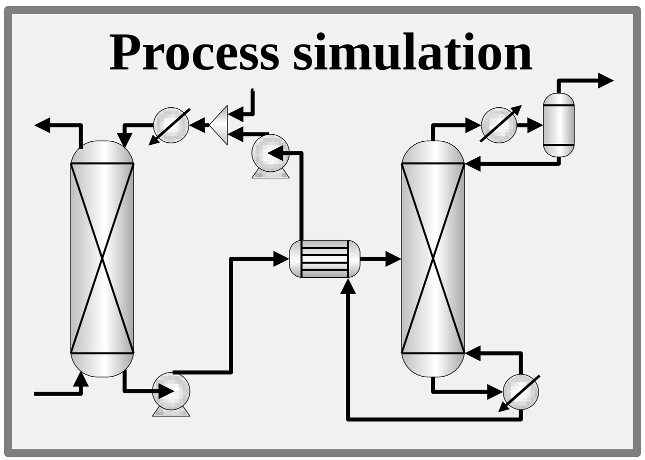
- **Latin hypercube design of experiments - Space-filling design**



INITIAL SAMPLING

- **After running the design of experiments, we will evaluate the black-box function to determine each z^i**

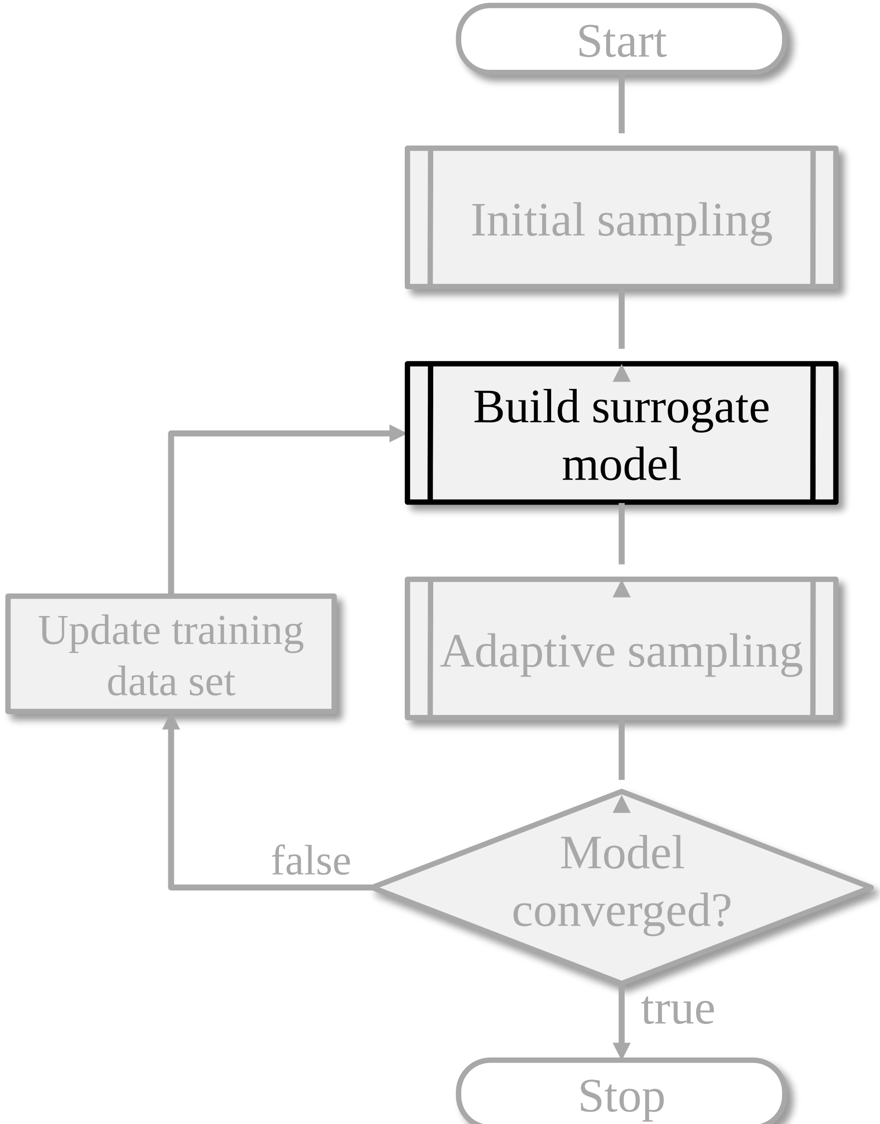
$$x = \begin{pmatrix} x^1 & x^2 & \cdots & x^i & \cdots & x^N \end{pmatrix}$$



$$z = \begin{pmatrix} z^1 & z^2 & \cdots & z^i & \cdots & z^N \end{pmatrix}$$

Initial training set

ALGORITHMIC FLOWSHEET



MODEL IDENTIFICATION

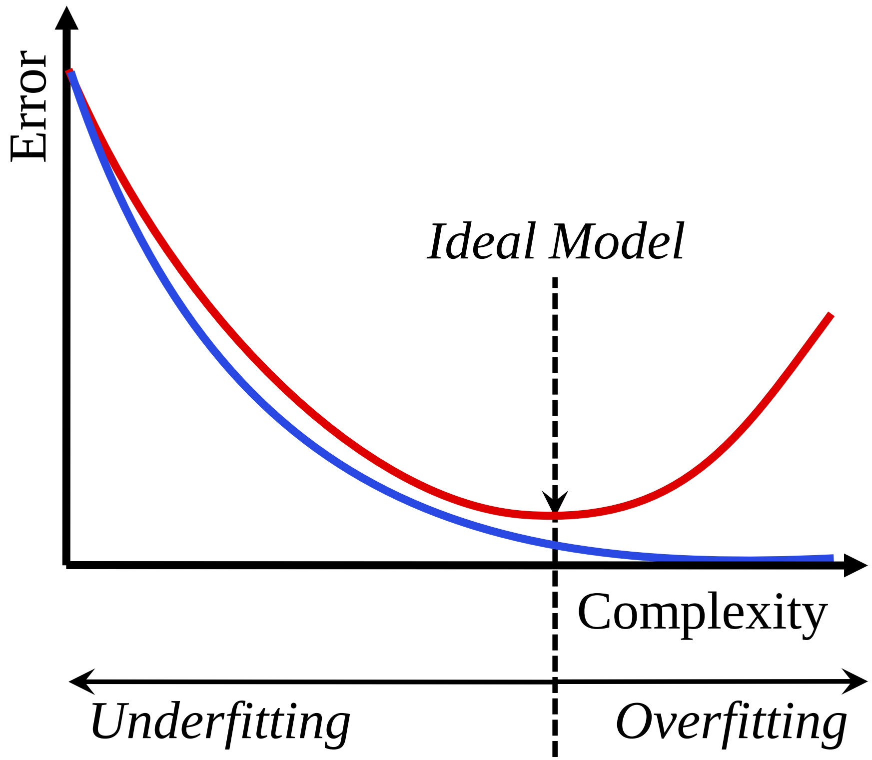
- **Goal: Identify the functional form and complexity of the surrogate models**

$$z = f(x)$$

- **Functional form:**
  - **General functional form is unknown: Our method will identify models with combinations of simple basis functions**

Category		$X_j(x)$
I.	Polynomial	$(x_d)^\alpha$
II.	Multinomial	$\prod_{d \in \mathcal{D}' \subseteq \mathcal{D}} (x_d)^{\alpha_d}$
III.	Exponential and logarithmic forms	$\exp\left(\frac{x_d}{\gamma}\right)^\alpha$, $\log\left(\frac{x_d}{\gamma}\right)^\alpha$
IV.	Expected bases	From experience, simple inspection, physical phenomena, etc.

OVERFITTING AND TRUE ERROR



- **Empirical error:**
  - **Error between the model and the sampled data points**
- **True error:**
  - **Error between the model and the true function**

SURROGATE MODEL

- **Surrogate model can have the form**

$$\hat{z} = \sum_{j \in \mathcal{B}} \beta_j X_j(x)$$

- **Low-complexity desired surrogate form**

$$\hat{z} = \sum_{j \in \mathcal{S}} \beta_j X_j(x)$$

$$\text{where } \mathcal{S} \subseteq \mathcal{B}$$

- $\mathcal{S}$ **is chosen to**
  - **Reduce overfitting**
  - **Achieve surrogate simplicity for a tractable final optimization model**

BEST SUBSET METHOD

- **Generalized best subset problem:**

$$\begin{aligned} \min_{\mathcal{S},\beta} \quad & \Phi(\mathcal{S},\beta) \\ \text{s.t.} \quad & \mathcal{S} \subseteq \mathcal{B} \end{aligned}$$

where $\Phi(\mathcal{S},\beta)$ is a goodness of fit measure for the subset of basis function, $\mathcal{S}$, and regression coefficients, β.

- **Goodness of fit:**
  - **Corrected Akaike Information Criterion (*AICc*)**
    - **Gives an estimate of the difference between a model and the true function**

$$AICc = \underbrace{N \log\left(\frac{SSE}{N}\right)}_{\textbf{Accuracy}} \; + \; \underbrace{2T + \frac{2T\,(T+1)}{N-T-1}}_{\textbf{Complexity}}$$

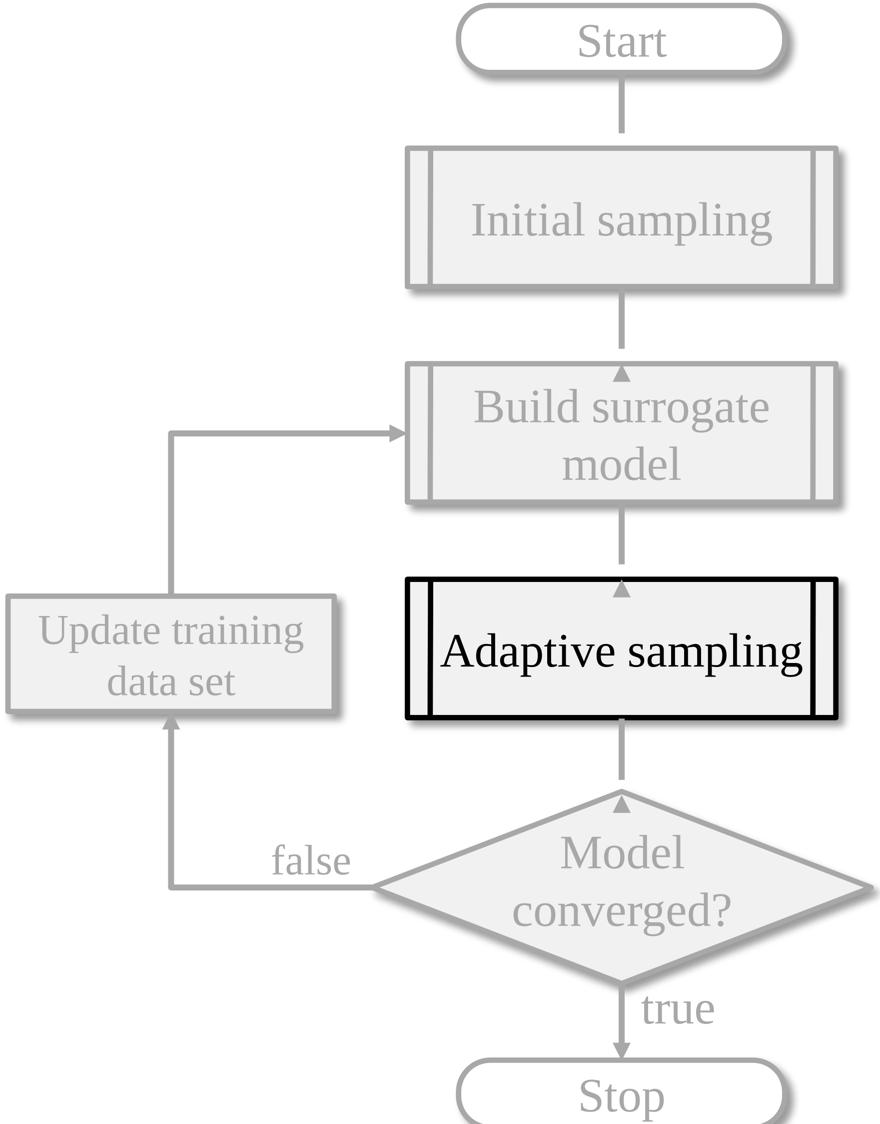
Accuracy + Complexity

FINAL BEST SUBSET MODEL

$$
\begin{aligned}
\min \quad & SE = \sum_{i=1}^{N} \left| z_i - \sum_{j \in \mathcal{B}} \beta_j X_{ij} \right| \\
\text{s.t.} \quad & \sum_{j \in \mathcal{B}} y_j = T \\
& -U(1-y_j) \le \sum_{i=1}^{N} X_{ij} \left(z_i - \sum_{j \in \mathcal{B}} \beta_j X_{ij} \right) \le U(1-y_j) && j \in \mathcal{B} \\
& \beta^l y_j \le \beta_j \le \beta^u y_j && j \in \mathcal{B} \\
& y_j \in \{0,1\} && j \in \mathcal{B} \\
& \beta_j \in \left[\beta_j^l, \beta_j^u\right] && j \in \mathcal{B}
\end{aligned}
$$

- **This model is solved for increasing values of T until the $AICc$ worsens**

ALGORITHMIC FLOWSHEET



ADAPTIVE SAMPLING

- **Goal: Search the problem space for areas of model inconsistency or model mismatch**

- **More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables**

Surrogate model

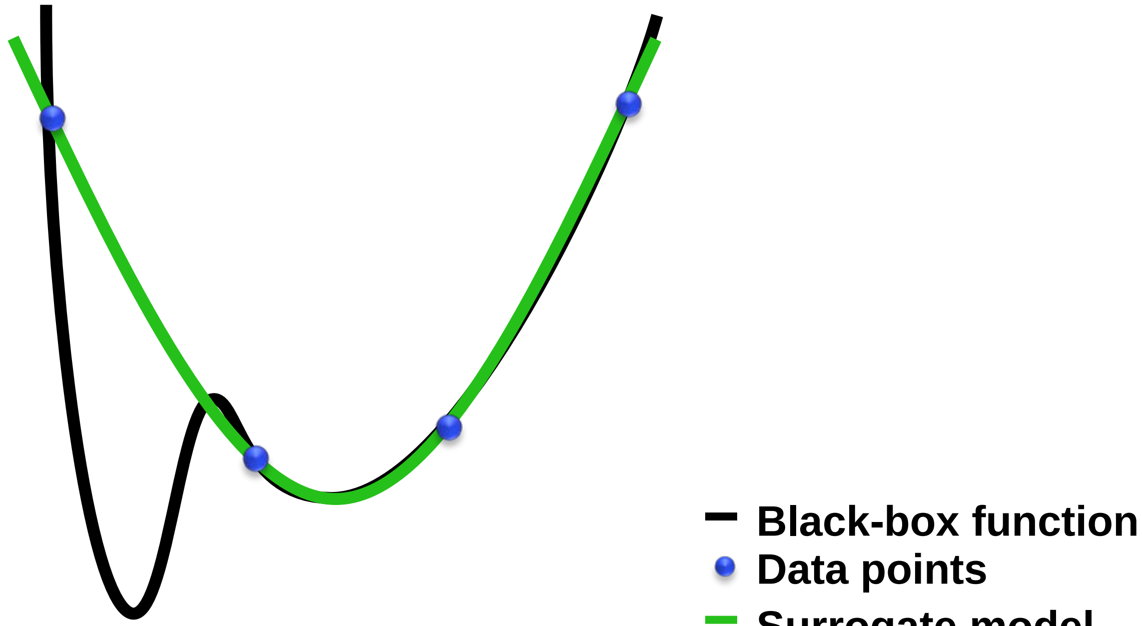
$$\max_{x} \left(\frac{z(x) - \hat{z}(x)}{z(x)} \right)^2$$

  - **Optimized using a black-box or derivative-free solver (SNOBFIT)** [Huyer and Neumaier, 08]

ADAPTIVE SAMPLING

- Goal: Search the problem space for areas of model inconsistency or model mismatch

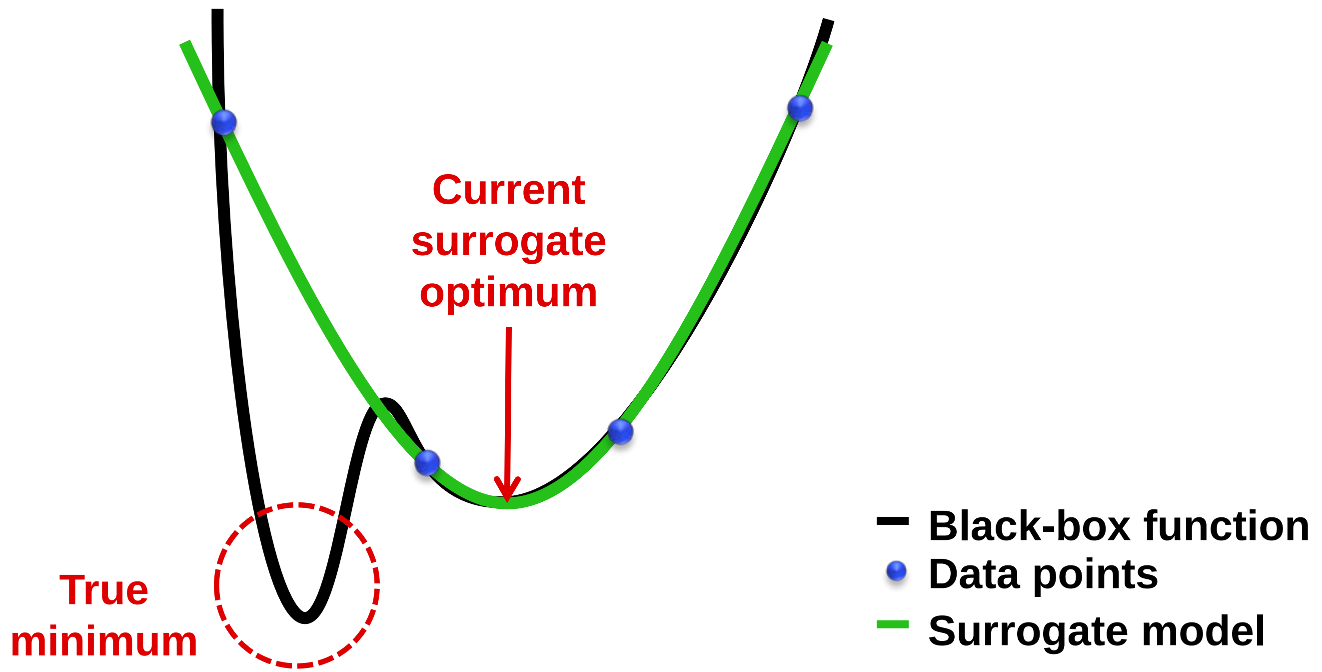
- More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables



ADAPTIVE SAMPLING

- Goal: Search the problem space for areas of model inconsistency or model mismatch

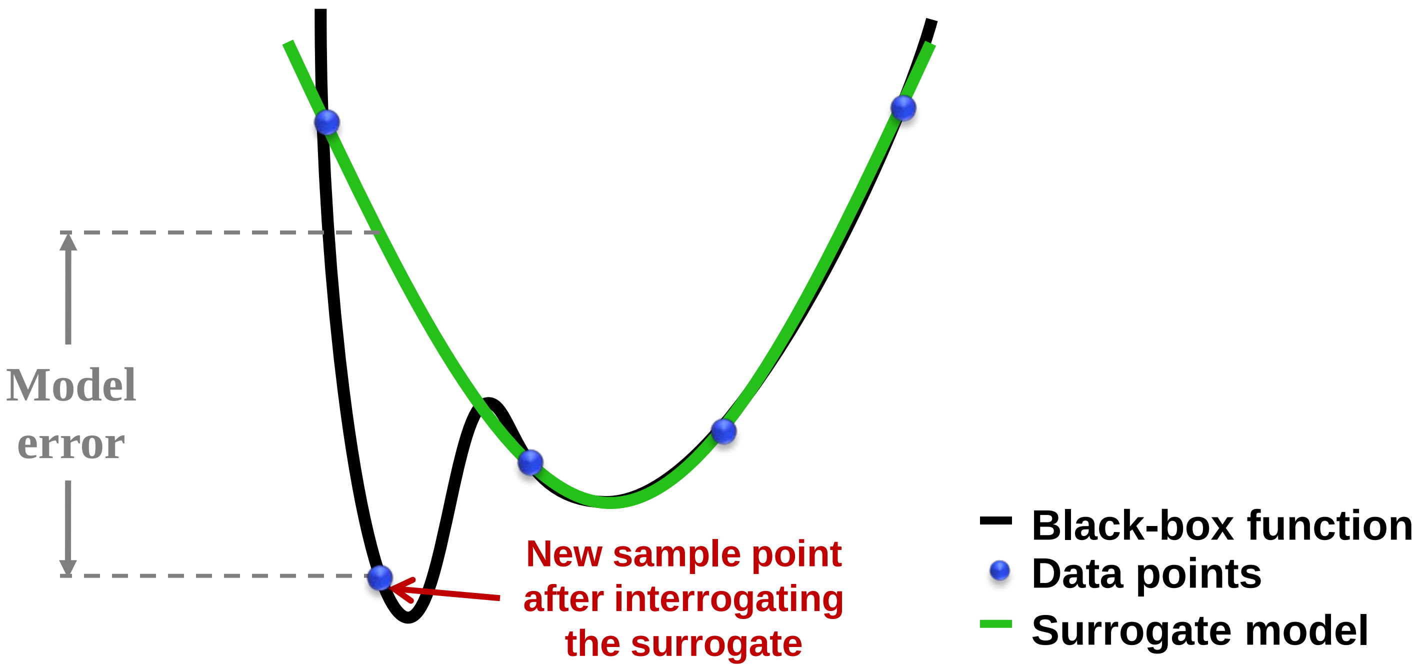
- More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables



ADAPTIVE SAMPLING

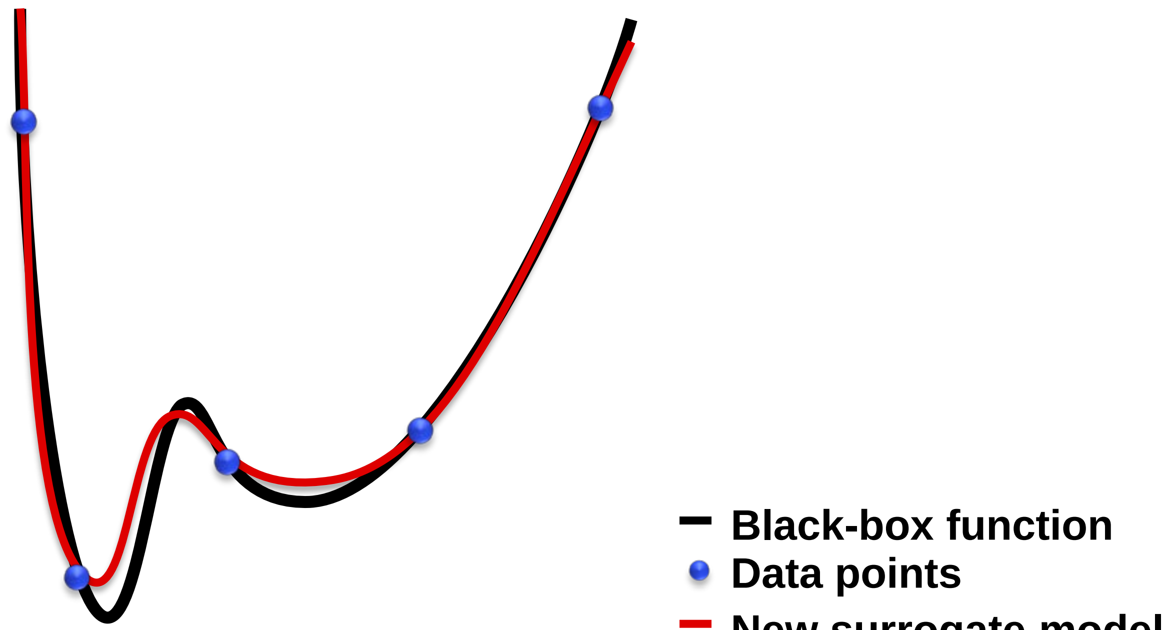
- Goal: Search the problem space for areas of model inconsistency or model mismatch

- More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables



ADAPTIVE SAMPLING

- **Goal: Search the problem space for areas of model inconsistency or model mismatch**
- **More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables**



ERROR MAXIMIZATION SAMPLING

- **Information gained using error maximization sampling:**
  1. **New data point locations that will be used to better train the next iteration's surrogate model**

  2. **Conservative estimate of the true model error**
     - **Defines a stopping criterion**
     - **Estimates the final model error**

COMPUTATIONAL TESTING

- **Surrogate generation methods have been implemented into a package:**

ALAMO

(Automated Learning of Algebraic Models for Optimization)

- **Modeling methods compared**
  - MIP – Proposed methodology
  - EBS – Exhaustive best subset method
    - Note: due to high CPU times this was only tested on smaller problems
  - LASSO – The lasso regularization
  - OLR – Ordinary least-squares regression

- **Sampling methods compared**
  - DFO – Proposed error maximization technique
  - SLH – Single latin hypercube (no feedback)

DESCRIPTION – TEST SET A

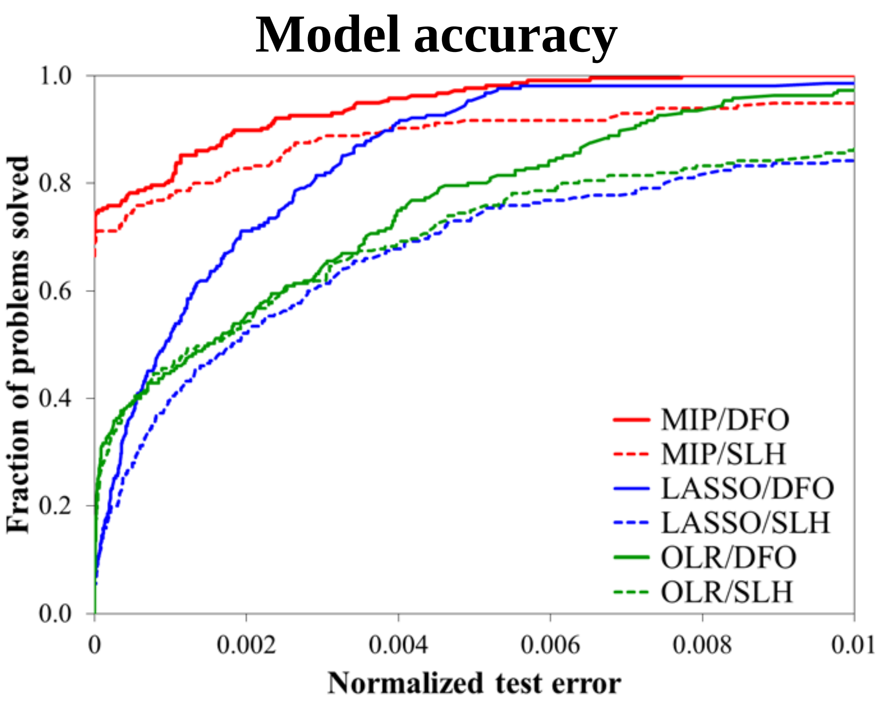
- **Two and three input black-box functions randomly chosen basis functions available to the algorithms with varying complexity from 2 to 10 terms**

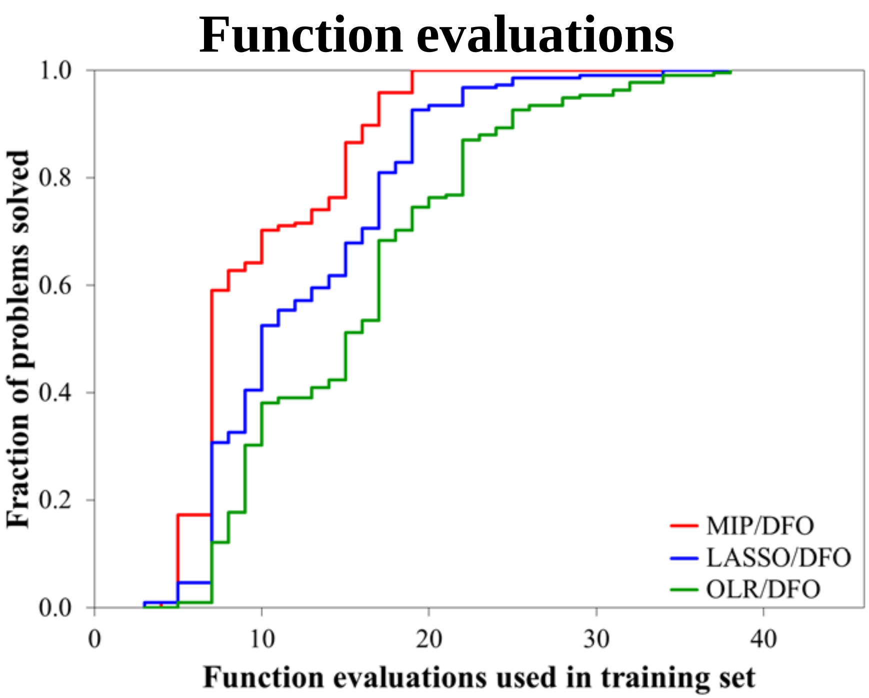
- **Basis functions allowed:**

Category		$X_j(x)$	Parameters used
I.	Polynomial	$(x_d)^\alpha$	$\alpha = \{\pm 3, \pm 2, \pm 1, \pm 0.5\}$
II.	Multinomial	$\prod_{d \in \mathcal{D}' \subseteq \mathcal{D}} (x_d)^{\alpha_d}$	for $\lvert\mathcal{D}'\rvert = 2 \quad \alpha = \{\pm 2, \pm 1, \pm 0.5\}$ for $\lvert\mathcal{D}'\rvert = 3 \quad \alpha = \{\pm 1\}$
III.	Exponential and logarithmic forms	$\exp\left(\frac{x_d}{\gamma}\right)^\alpha, \log\left(\frac{x_d}{\gamma}\right)^\alpha$	$\alpha = 1, \ \gamma = 1$

True basis function coefficients were randomly chosen from a uniform distribution where $\beta \in [-1, 1]$.

RESULTS – TEST SET A





45 test problems, repeated 5 times, tested against 1000 independent data points

MODEL COMPLEXITY – TEST SET A

No. of inputs	No. of true terms	M1/ DFO	M1/ SLH	EBS/ DFO	EBS/ SLH	LASSO/ DFO	LASSO/ SLH	OLR/ DFO	OLR/ SLH
2	2	2	[2, 2]	2	2	[6, 8]	[6, 11]	[12, 15]	[12, 15]
2	3	3	3	3	3	[5, 12]	[5, 10]	[12, 14]	[12, 14]
2	4	[3, 4]	[3, 4]	[3, 4]	[3, 4]	[8, 11]	[8, 10]	[11, 12]	[11, 12]
2	5	[2, 4]	[2, 4]	[2, 5]	[2, 5]	[3, 12]	[4, 11]	[10, 16]	[10, 16]
2	6	[5, 6]	[6, 6]	[5, 6]	[6, 6]	[7, 10]	[6, 7]	[11, 13]	[11, 13]
2	7	[4, 6]	[4, 6]	[4, 7]	[4, 7]	[7, 11]	[6, 12]	[8, 13]	[8, 13]
2	8	[4, 5]	[5, 6]	[4, 5]	[5, 6]	[6, 8]	[6, 9]	[10, 15]	[10, 15]
2	9	[4, 6]	[4, 6]	NA	NA	[6, 14]	[7, 12]	[10, 17]	[10, 17]
2	10	[4, 8]	[4, 8]	NA	NA	[5, 14]	[7, 14]	[10, 14]	[10, 14]
3	2	[2, 3]	[2, 3]	NA	NA	[6, 12]	[7, 13]	[27, 29]	[27, 29]
3	3	[3, 3]	[3, 3]	NA	NA	[8, 16]	[7, 15]	[19, 22]	[19, 22]
3	4	4	[3, 4]	NA	NA	[10, 13]	[9, 10]	[16, 21]	[16, 21]
3	5	5	5	NA	NA	[11, 17]	[9, 15]	[15, 23]	[15, 23]
3	6	[5, 6]	[6, 6]	NA	NA	[9, 18]	[10, 13]	[15, 26]	[15, 26]
3	7	7	[7, 8]	NA	NA	[10, 22]	[10, 22]	22	22

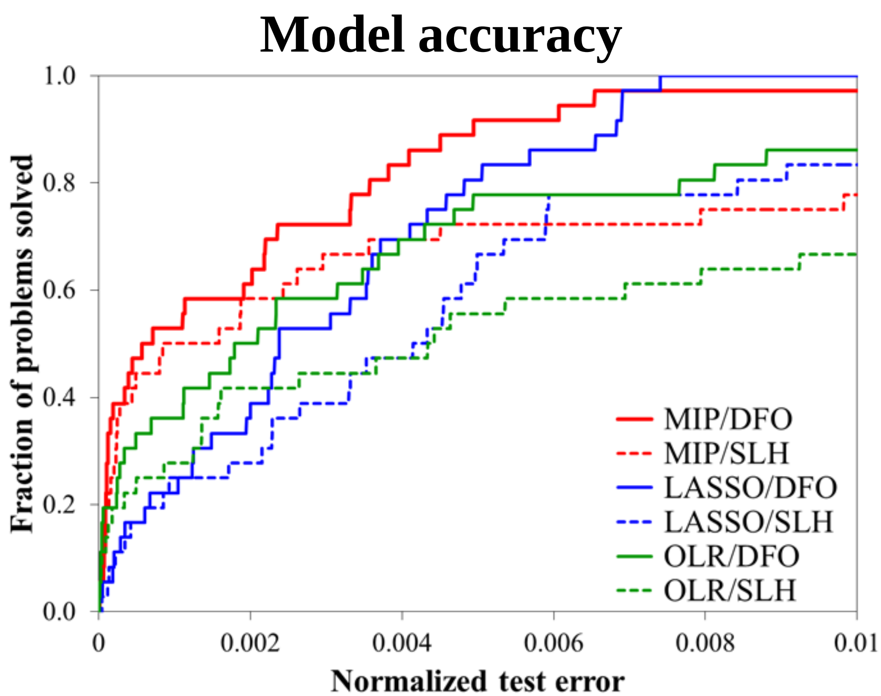
DESCRIPTION – TEST SET B

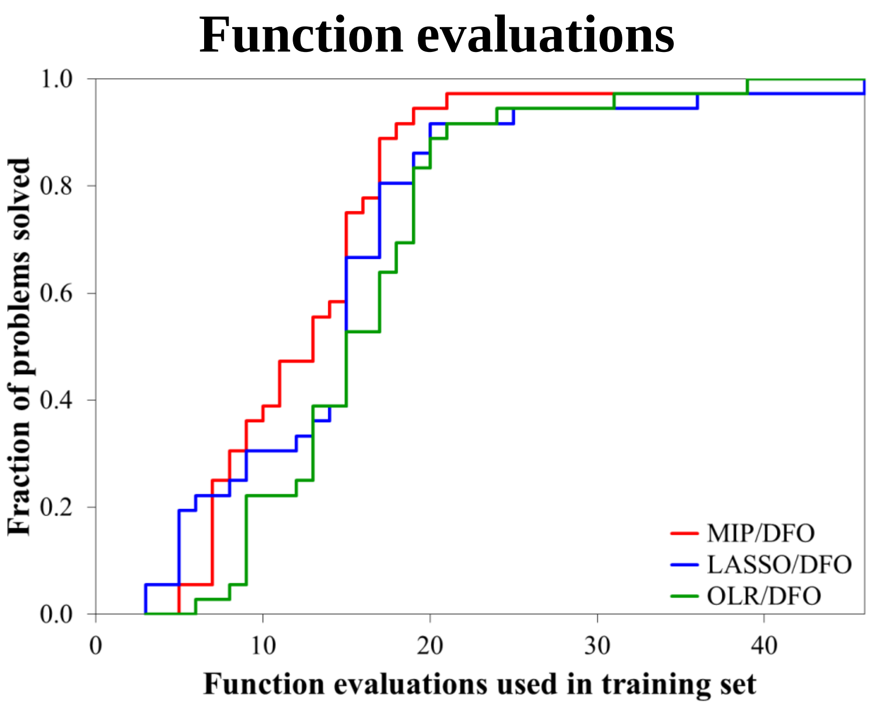
- **Two input black-box functions with basis functions unavailable to the algorithms with**

Function type	Functional form
I	$z(x) = \beta x_i^{\alpha} \exp(x_j)$
II	$z(x) = \beta x_i^{\alpha} \log(x_j)$
III	$z(x) = \beta x_1^{\alpha} x_2^{\nu}$
IV	$z(x) = \dfrac{\beta}{\gamma + x_i^{\alpha}}$

with true parameters chosen from a uniform distribution where $\beta \in [-1, 1]$, $\alpha, \nu \in [-3, 3]$, $\gamma \in [-5, 5]$, and $i, j \in \{1, 2\}$.

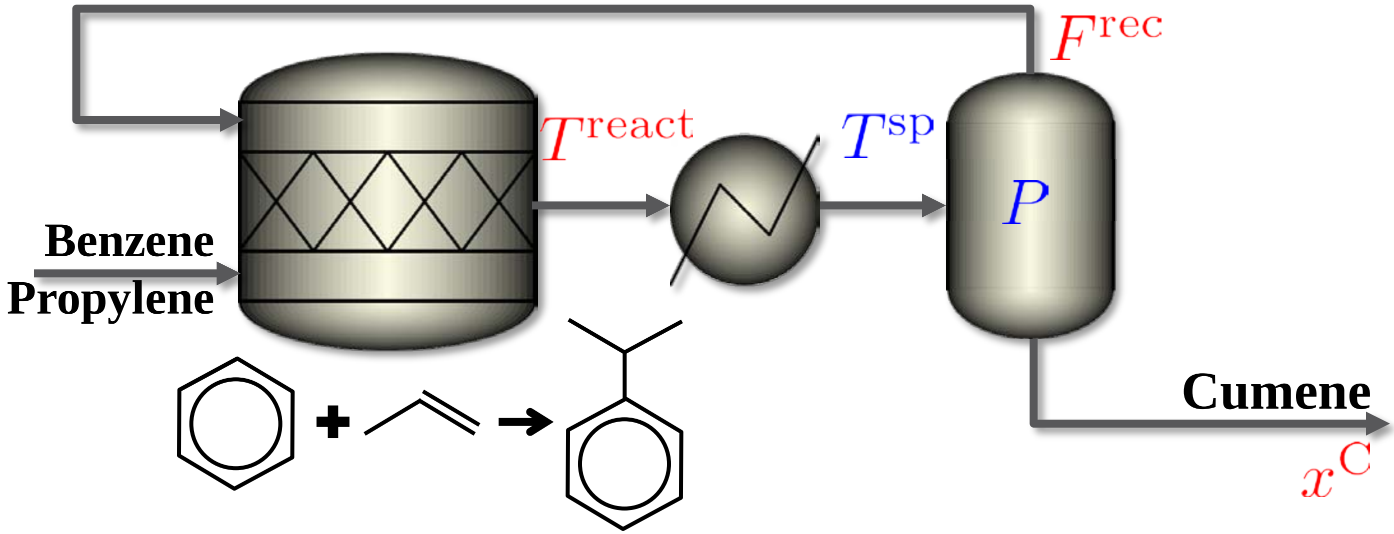
RESULTS – TEST SET B





12 test problems, repeated 5 times, tested against 1000 independent data points

TEST CASE: CUMENE PRODUCTION



Generate Models:

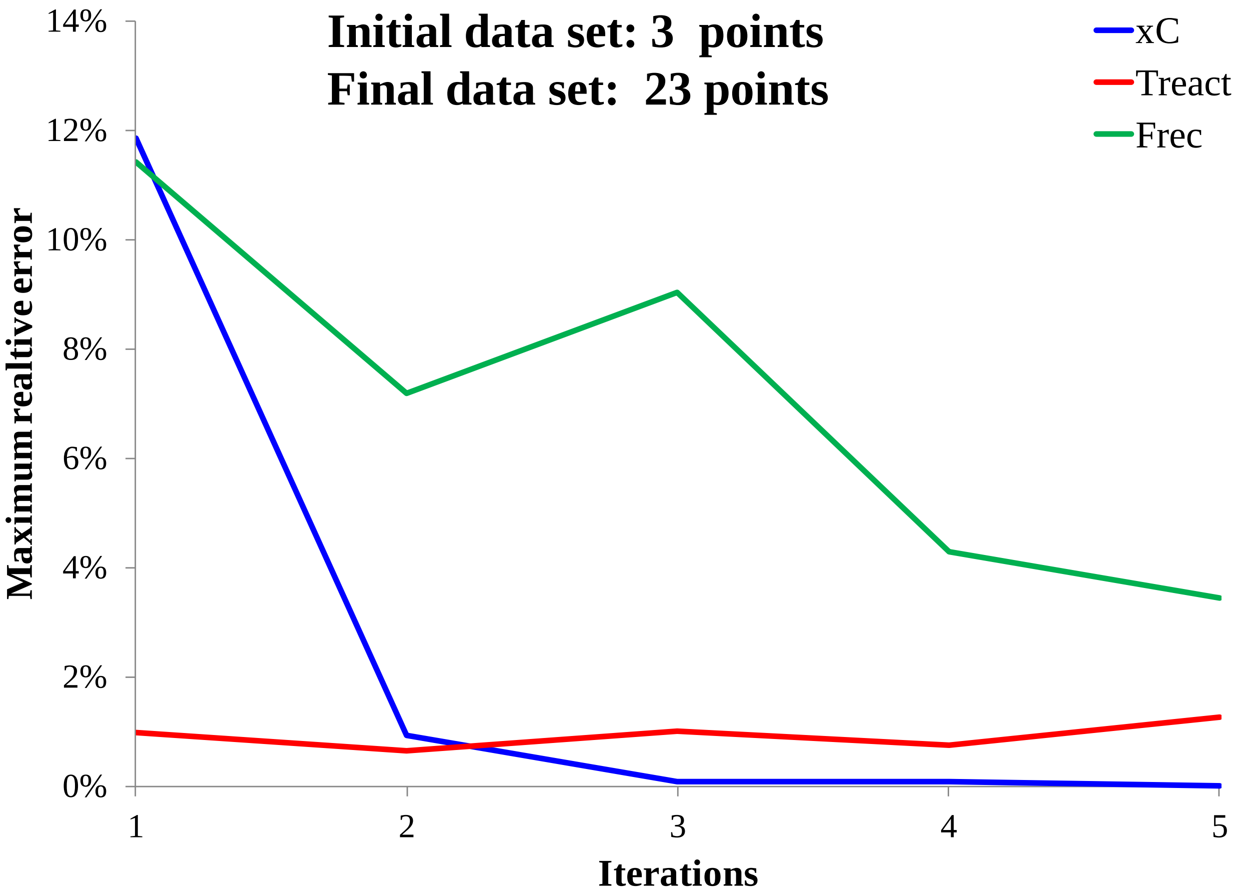
$$x^{\mathrm{C}}(T^{\mathrm{sp}}, P) = f_1(T^{\mathrm{sp}}, P)$$
$$T^{\mathrm{react}}(T^{\mathrm{sp}}, P) = f_2(T^{\mathrm{sp}}, P)$$
$$F^{\mathrm{rec}}(T^{\mathrm{sp}}, P) = f_3(T^{\mathrm{sp}}, P)$$

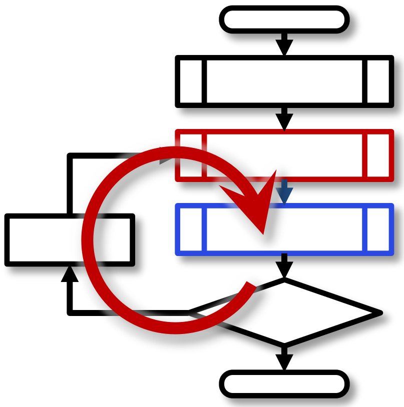
Over the Range:

$$100^\circ F \leq T^{\mathrm{sp}} \leq 250^\circ F$$
$$0.82 \text{ atm} \leq P \leq 1.36 \text{ atm}$$

Cumene production simulation is form the Aspen Plus® Library

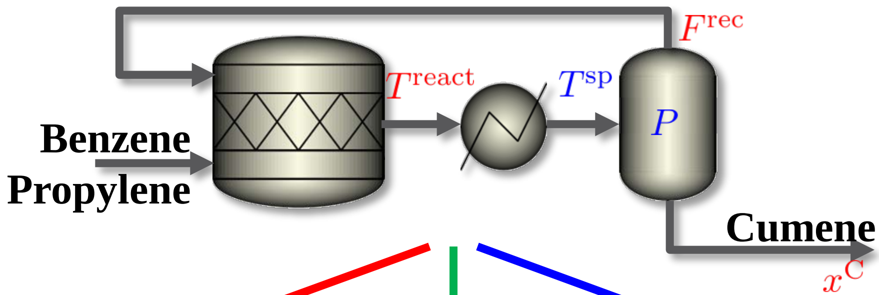
GENERATING THE SURROGATES





- **Maximum error found at each iteration may increase**
  - **Due to the derivative-free solver is given more information at each iteration**

PROCESS OPTIMIZATION



$$\max \quad x^{\mathrm{C}}$$

$$\min \quad \text{Utilities}$$

$$\begin{aligned} \min \quad & \text{CaptialCost} \\ \text{s.t.} \quad & x^{\mathrm{C}} \geq 0.95 \end{aligned}$$

$$\begin{aligned} T^{\mathrm{sp}} &= 250°\mathrm{F} \\ P &= 0.82\mathrm{atm} \\ x^{\mathrm{C}} &= 0.981 \\ T^{\mathrm{react}} &= 828.1°\mathrm{F} \\ F^{\mathrm{rec}} &= 645.7\frac{\mathrm{lb}}{\mathrm{hr}} \\ \dot{\mathrm{m}}^{\mathrm{cw}} &= 5.00 \cdot 10^{7}\frac{\mathrm{lb}}{\mathrm{hr}} \\ \mathrm{A} &= 7,706\mathrm{ft}^2 \end{aligned}$$

$\leq 5.2\%$error

$$\begin{aligned} T^{\mathrm{sp}} &= 250°\mathrm{F} \\ P &= 1.36\mathrm{atm} \\ x^{\mathrm{C}} &= 0.964 \\ T^{\mathrm{react}} &= 848.9°\mathrm{F} \\ F^{\mathrm{rec}} &= 373.16\frac{\mathrm{lb}}{\mathrm{hr}} \\ \dot{\mathrm{m}}^{\mathrm{cw}} &= 4.04 \cdot 10^{4}\frac{\mathrm{lb}}{\mathrm{hr}} \\ \mathrm{A} &= 2,364\mathrm{ft}^2 \end{aligned}$$

$\leq 3.0\%$error

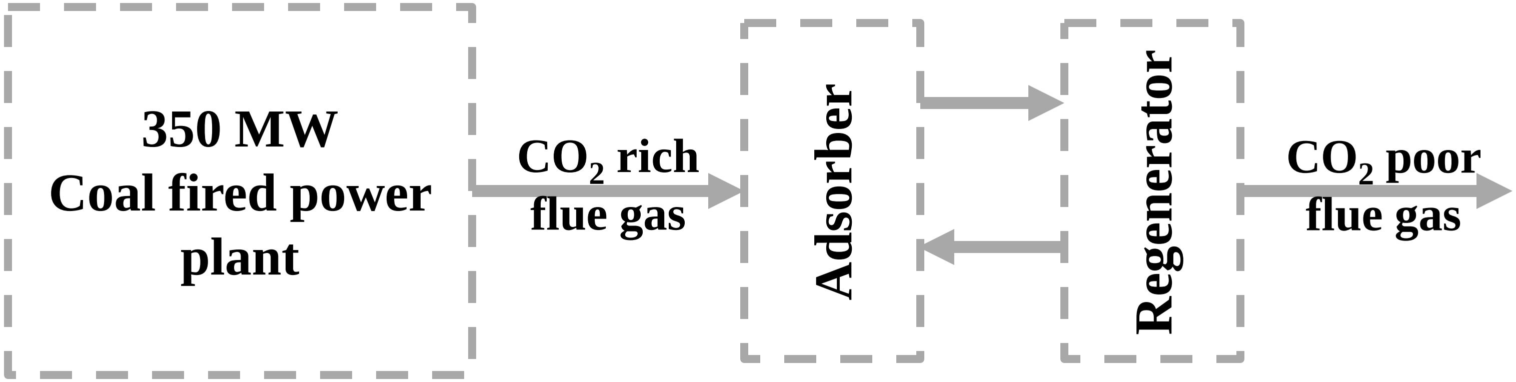
$$\begin{aligned} T^{\mathrm{sp}} &= 168.1°\mathrm{F} \\ P &= 0.82\mathrm{atm} \\ x^{\mathrm{C}} &= 0.948 \\ T^{\mathrm{react}} &= 860.4°\mathrm{F} \\ F^{\mathrm{rec}} &= 172.7\frac{\mathrm{lb}}{\mathrm{hr}} \\ \dot{\mathrm{m}}^{\mathrm{cw}} &= 4.40 \cdot 10^{4}\frac{\mathrm{lb}}{\mathrm{hr}} \\ \mathrm{A} &= 2,310\mathrm{ft}^2 \end{aligned}$$

$\leq 5.7\%$error

CARBON CAPTURE OPTIMIZATION

- **Problem statement:**

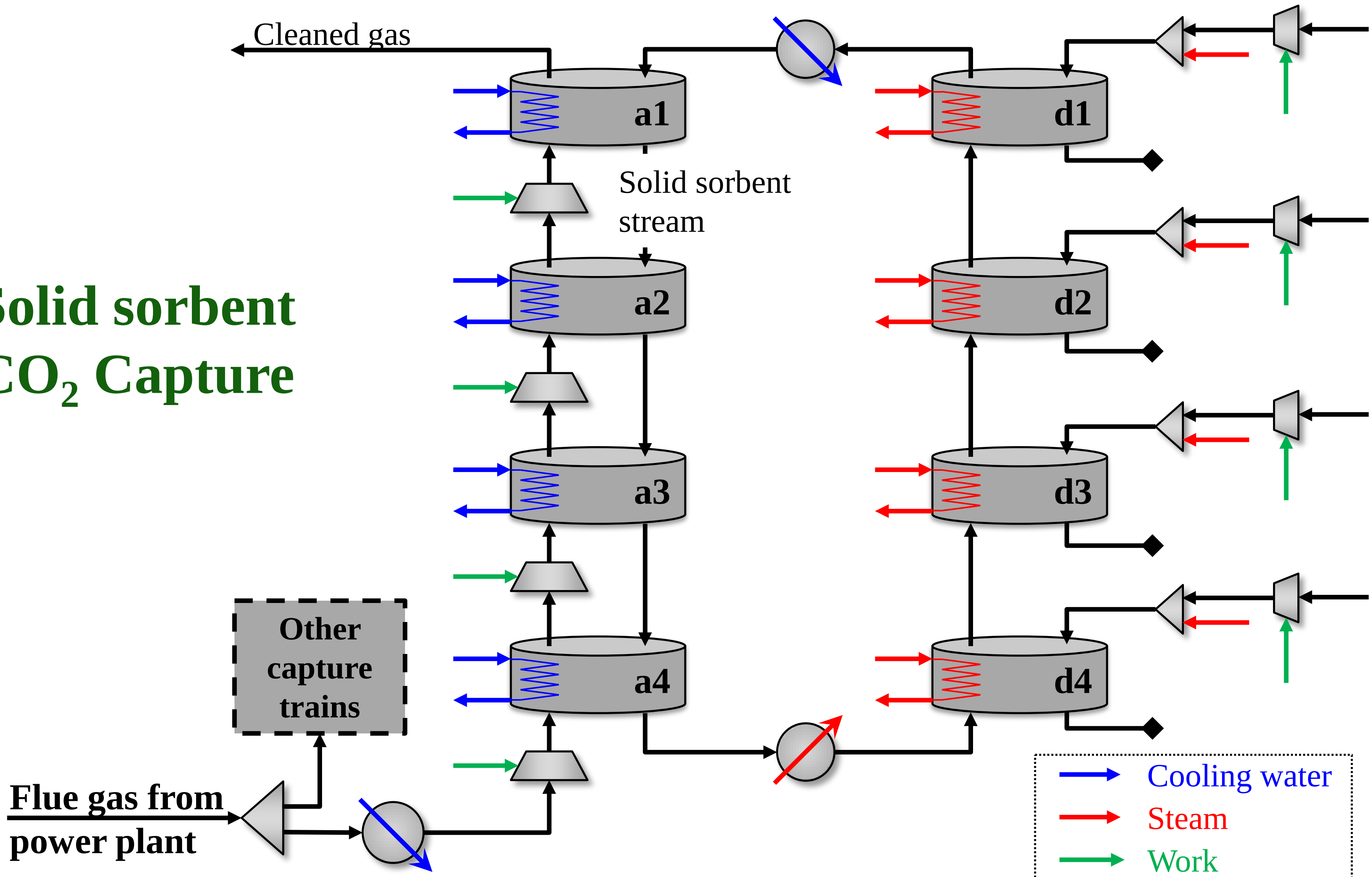
  Capture 90% of CO_2 from a 350MW power plant's post combustion flue gas with minimal increase in the cost of electricity



- **Design considerations:**
  - Capture technology
    - Bubbling fluidized bed, moving bed, fast fluidized bed, transport bed, etc.
  - Number of reactors
  - Reactor configuration and geometry
  - Operating conditions

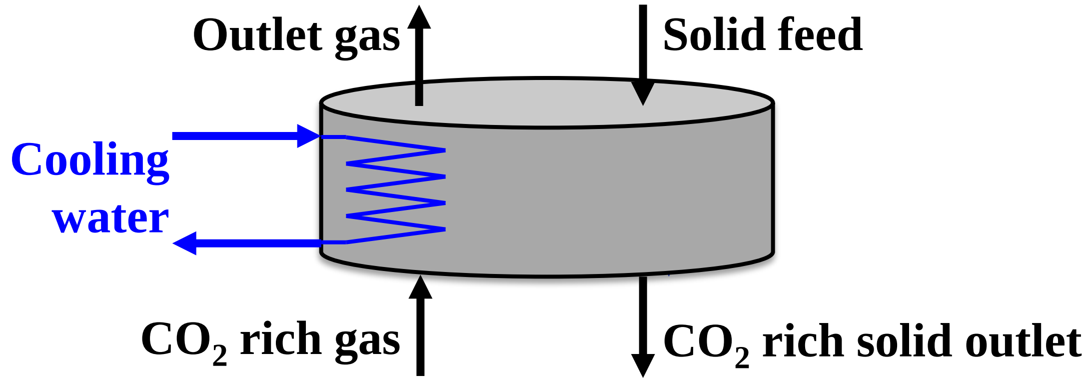
SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 Capture



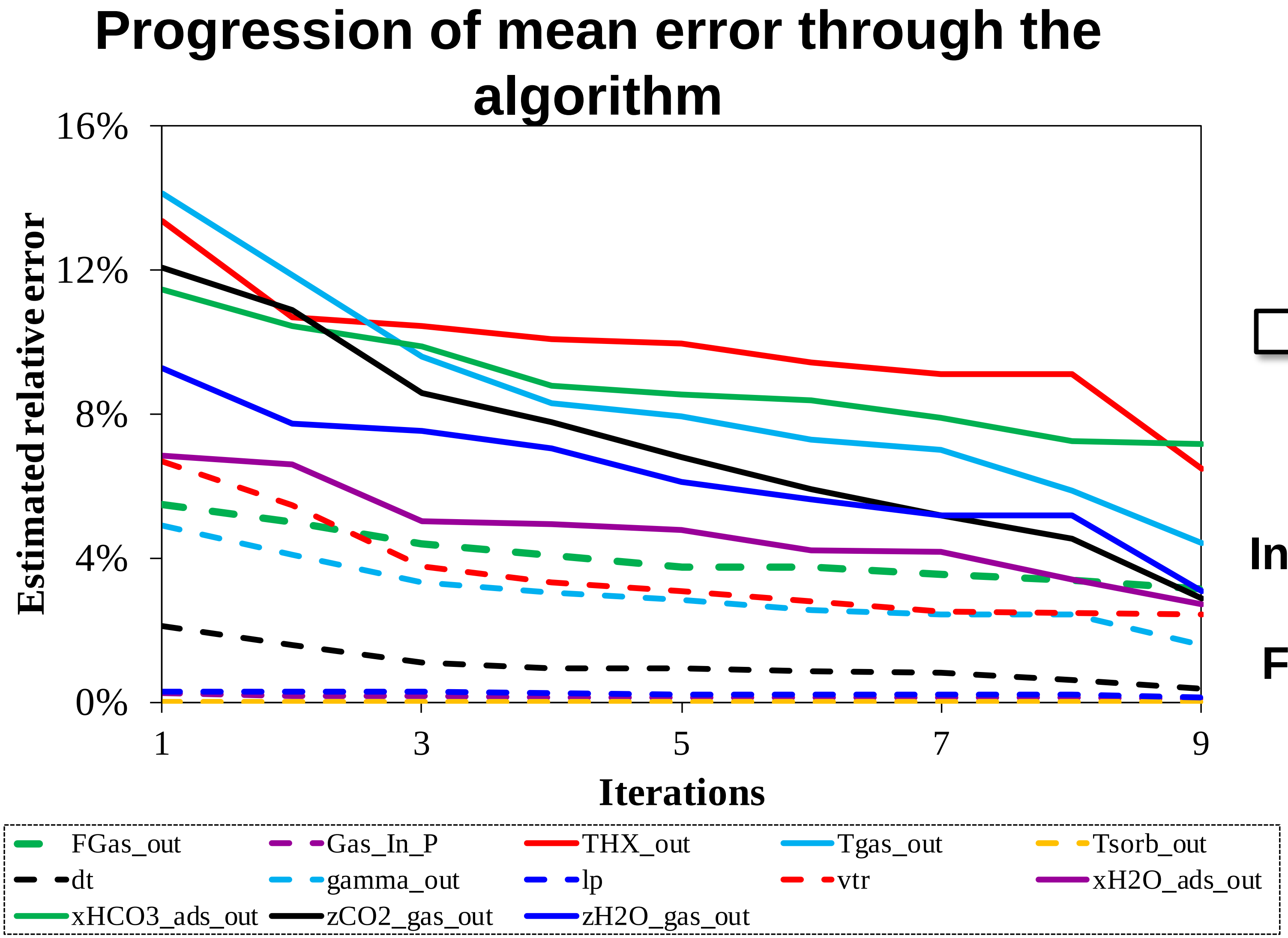
BUBBLING FLUIDIZED BED

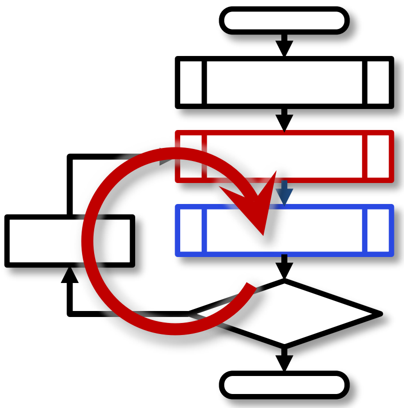
Bubbling fluidized bed adsorber diagram



- Model inputs (14 total)
  - Geometry (3)
  - Operating conditions (4)
  - Gas mole fractions (2)
  - Solid compositions (2)
  - Flow rates (4)
- Model outputs (13 total)
  - Geometry required (2)
  - Operating condition required (1)
  - Gas mole fractions (2)
  - Solid compositions (2)
  - Flow rates (2)
  - Outlet temperatures (3)
  - Design constraint (1)

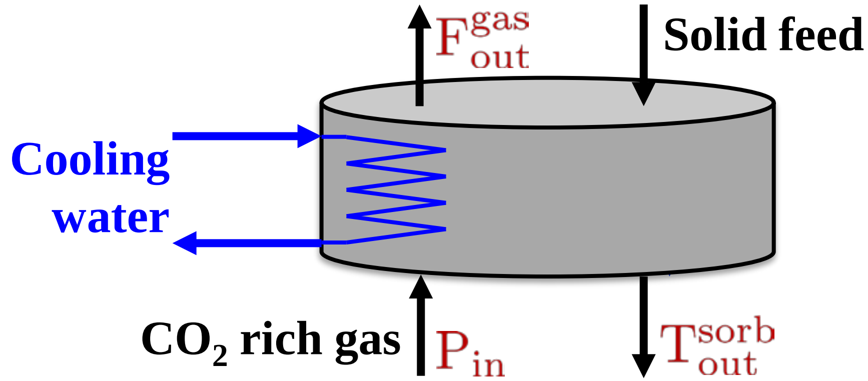
ADAPTIVE SAMPLING





Initial data set: 137 pts
Final data set: 261

EXAMPLE MODELS



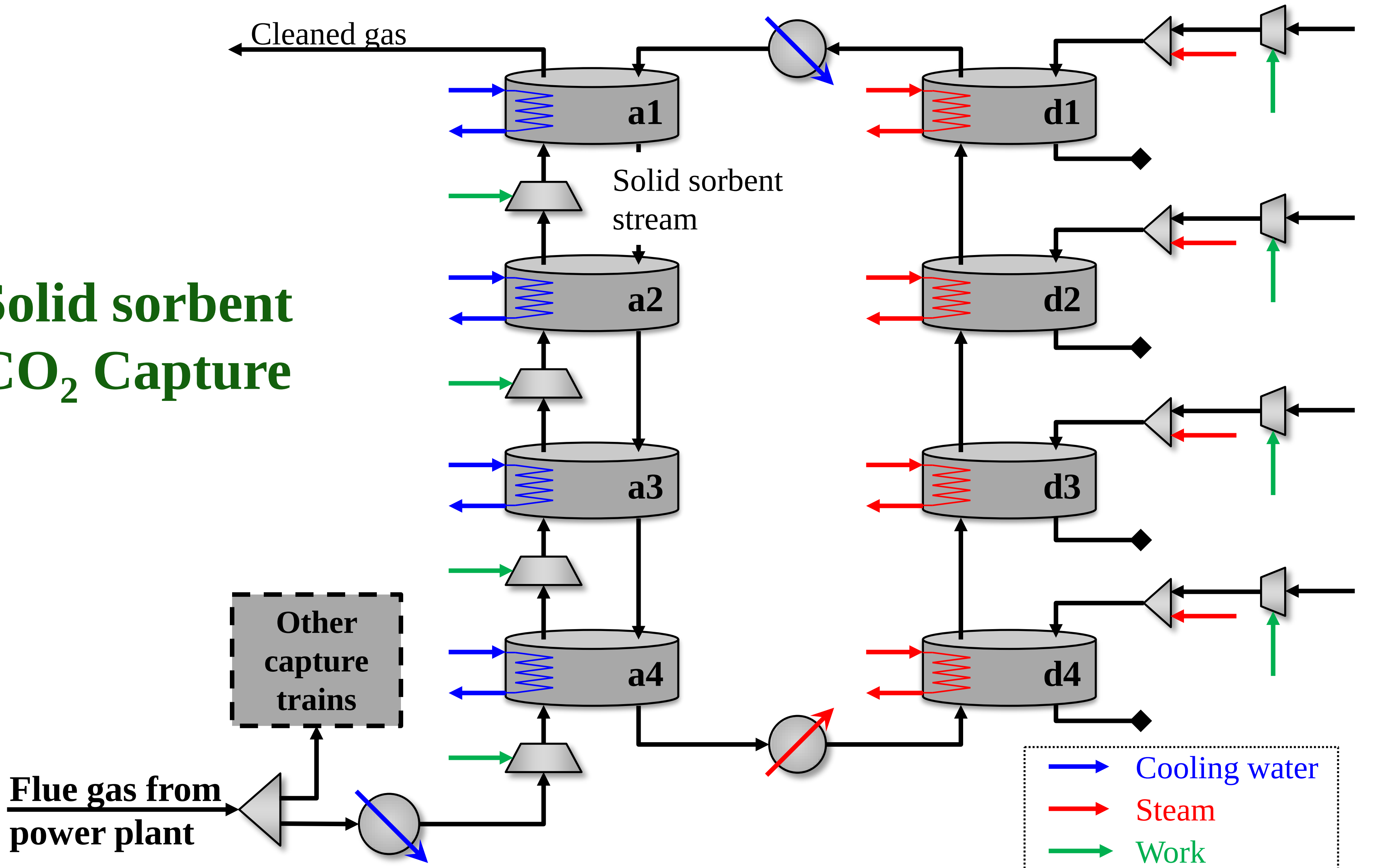
$$
\begin{aligned}
\mathrm{P_{in}} &= 1.0\,\mathrm{P_{out}} + 0.0231\,L_b - 0.0187\,\ln(0.167\,L_b) - 0.00626\,\ln(0.667\,\mathrm{v_{gi}}) - \\
&\quad \frac{51.1\,\mathrm{xHCO3^{ads}_{in}}}{\mathrm{F^{gas}_{in}}}
\end{aligned}
$$

$$
T^{sorb}_{out} = 1.0\,\mathrm{T^{gas}_{in}} - \frac{\left(1.77\cdot 10^{-10}\right)\,\mathrm{NX}^2}{\gamma^2} - \frac{3.46}{\mathrm{NX\,T^{gas}_{in}\,T^{sorb}_{in}}} + \frac{1.17\cdot 10^4}{\mathrm{F^{sorb}\,NX\,xH2O^{ads}_{in}}}
$$

$$
\begin{aligned}
F^{gas}_{out} &= 0.797\,\mathrm{F^{gas}_{in}} - \frac{9.75\,\mathrm{T^{sorb}_{in}}}{\gamma} - 0.77\,\mathrm{F^{gas}_{in}\,xCO2^{gas}_{in}} + 0.00465\,\mathrm{F^{gas}_{in}\,T^{sorb}_{in}} - \\
&\quad 0.0181\,\mathrm{F^{gas}_{in}\,T^{sorb}_{in}\,xH2O^{gas}_{in}}
\end{aligned}
$$

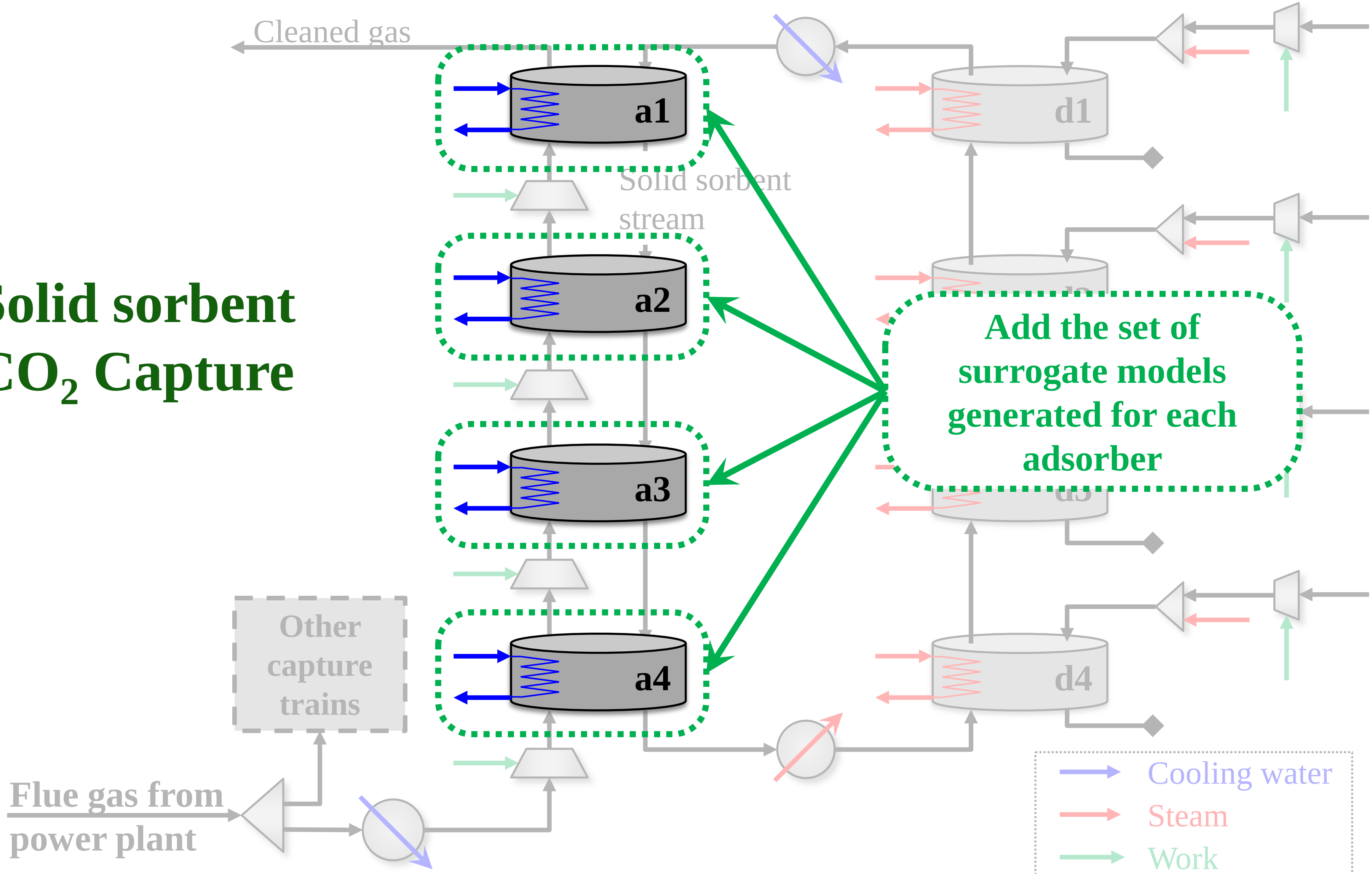
SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 Capture

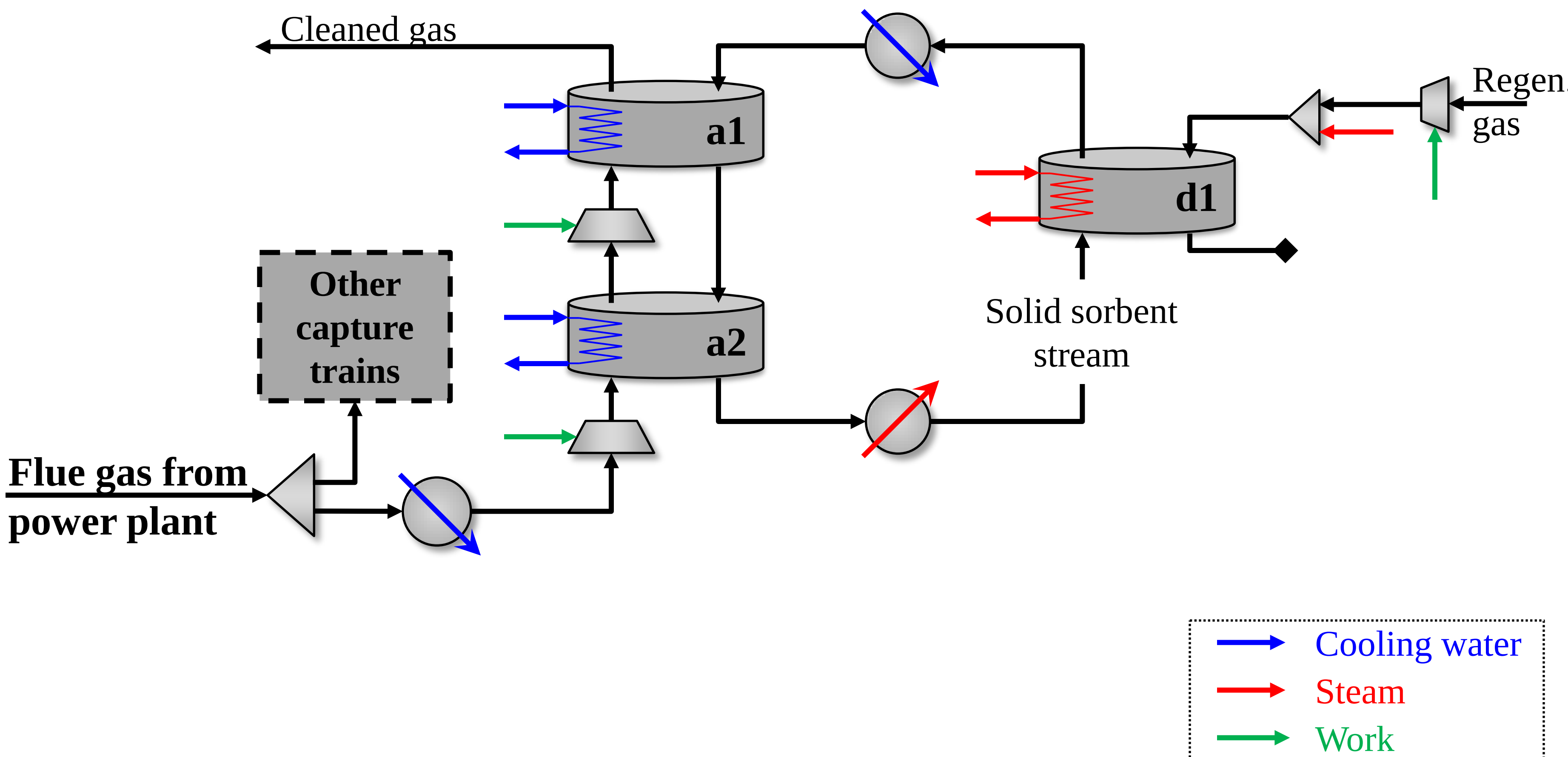


SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 Capture



PRELIMINARY RESULTS



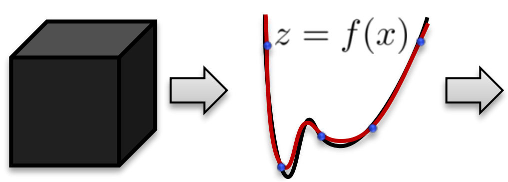
CONCLUSIONS

- **The algorithm we developed is able to model black-box functions for use in optimization such that the models are**
  - ✓ **Accurate**
  - ✓ **Tractable in an optimization framework (low-complexity models)**
  - ✓ **Generated from a minimal number of function evaluations**

- **Surrogate models can then be incorporated within a optimization framework flexible objective functions and additional constraints**

ALAMO

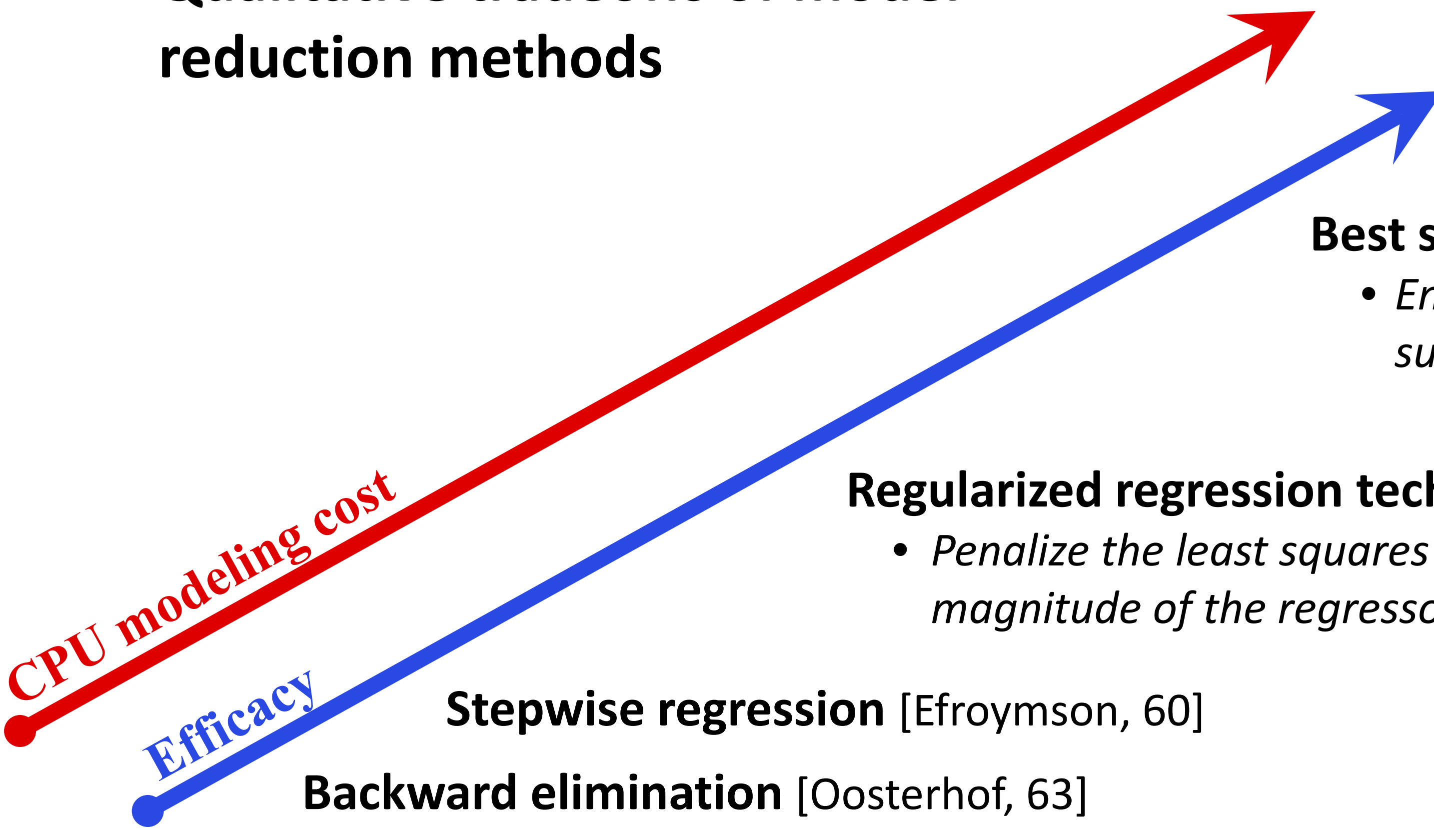
Automated **L**earning of **A**lgebraic **M**odels for **O**ptimization



$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

MODEL REDUCTION TECHNIQUES

- **Qualitative tradeoffs of model reduction methods**



Best subset methods

- *Enumerate all possible subsets*

Regularized regression techniques

- *Penalize the least squares objective using the magnitude of the regressors*

Stepwise regression [Efroymson, 60]

Backward elimination [Oosterhof, 63]

Forward selection [Hamaker, 62]

BEST SUBSET METHOD

- **Surrogate subset model:**

$$\hat{z}(x) = \sum_{j \in \mathcal{S}} \beta_j X_j(x)$$

- **Mixed-integer surrogate subset model:**

$$\hat{z}(x) = \sum_{j \in \mathcal{B}} (y_j \beta_j)\, X_j(x) \quad \text{such that} \quad \begin{array}{ll} y_j = 1 & j \in \mathcal{S} \\ y_j = 0 & j \notin \mathcal{S} \end{array}$$

- **Generalized best subset problem mixed-integer formulation:**

$$\begin{aligned} \min_{\beta, y} \quad & \Phi(\beta, y) \\ \text{s.t.} \quad & y_j = \{0, 1\} \end{aligned}$$

MIXED-INTEGER PROBLEM

- **Further reformulation**
  - **Replace bilinear terms with big-M constraints**

$$y_j \beta_j \quad \longrightarrow \quad \beta_j^l y_j \leq \beta_j \leq \beta_j^u y_j$$

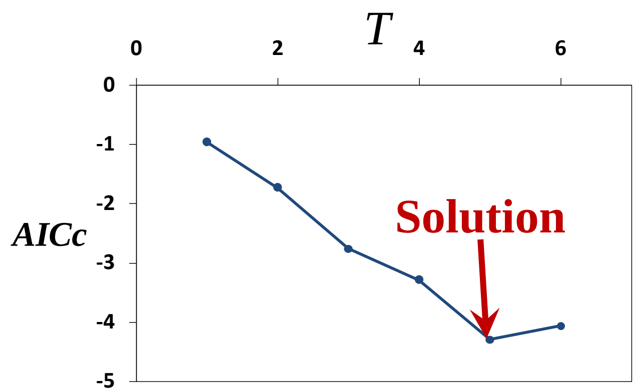
  - **Decouple objective into two problems**

$$\text{General:} \quad \min_{\beta,T,y} \Phi(\beta,T,y) \quad = \min_{T} \overbrace{\left\{ \underbrace{\min_{\beta,y} \left[\Phi_{\beta,y}(\beta,y)|_T\right]}_{\text{b) basis and coefficient selection}} + \Phi_T(T) \right\}}^{\text{a) model sizing}}$$

  - **Inner minimization objective reformulation**

PROBLEM SIMPLIFICATIONS

- **Simplifications:**
  - Outer problem
    - The outer problem is parameterized by T and a local minima is found



  - Inner problem
    - Stationarity condition used to solve for continuous variables

$$\frac{d}{d\beta_j}\sum_{i=1}^{N}\left(z_i-\sum_{j\in\mathcal{S}}\beta_j X_{ij}\right)^2 \propto \sum_{i=1}^{N} X_{ij}\left(z_i-\sum_{j\in\mathcal{S}}\beta_j X_{ij}\right)=0, \quad j\in\mathcal{S}$$

    - Linear objective used to solved for integer variables

$$\text{Objective:}\quad \sum_{i=1}^{N}\left|z_i-\sum_{j\in\mathcal{S}}\beta_j X_{ij}\right|$$