Surrogate-based optimization of simulated energy systems

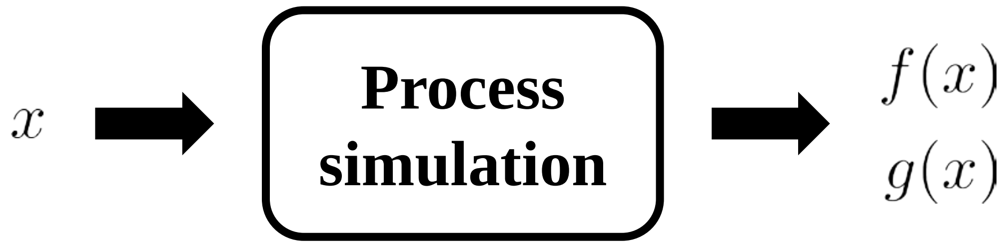
Alison Cozad[1,2], Nick Sahinidis[1,2], David Miller[2]

[1]National Energy Technology Laboratory, Pittsburgh, PA,USA

[2]Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA

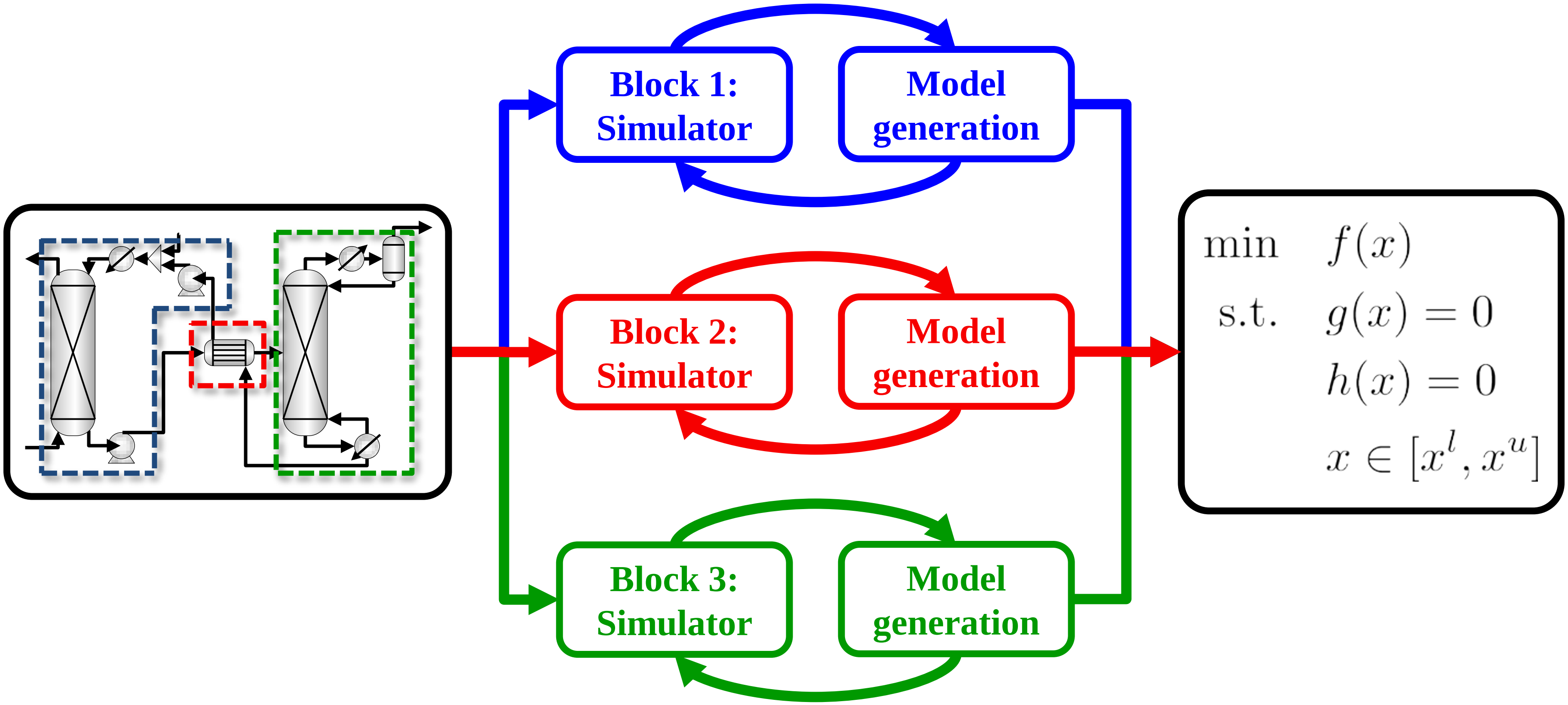
SIMULATION-BASED OPTIMIZATION

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



- **Challenges and solutions:**
  - **Lack of an algebraic model** → Build surrogate models
  - **Computationally costly simulations** → Selectively choose a minimal data set
  - **Often noisy function evaluations** → Use regression surrogate models
  - **Scarcity of fully robust simulations** → Disaggregate the process

PROCESS DISAGGREGATION



Process Simulation

Disaggregate process into process blocks

Surrogate Models

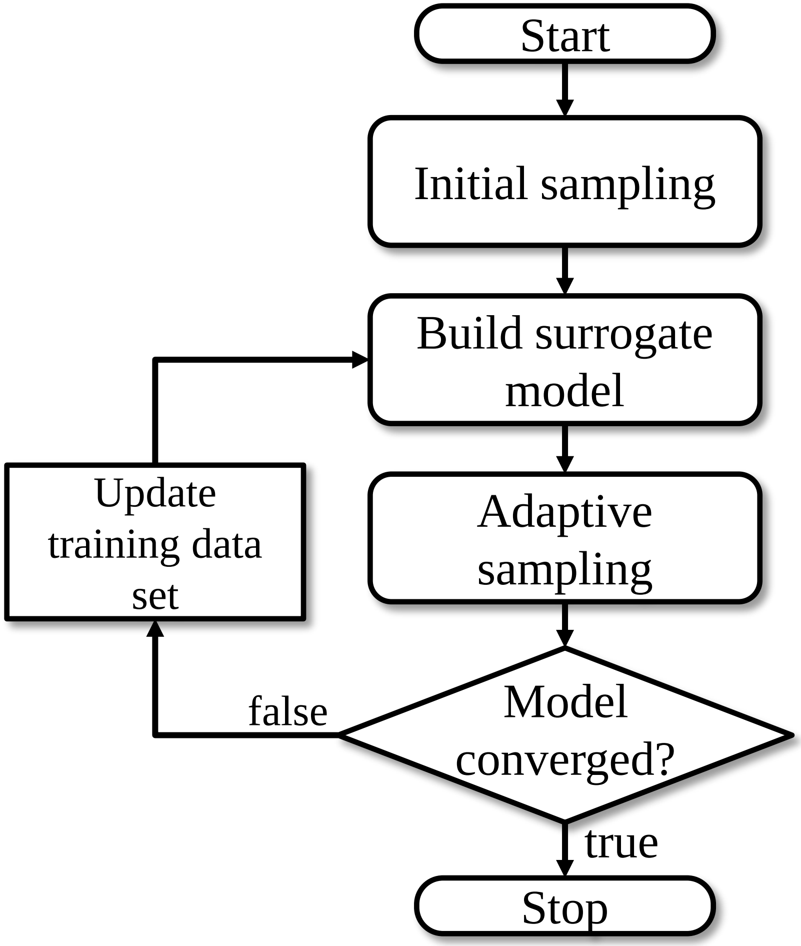
Build simple and accurate models with a functional form tailored for an optimization framework

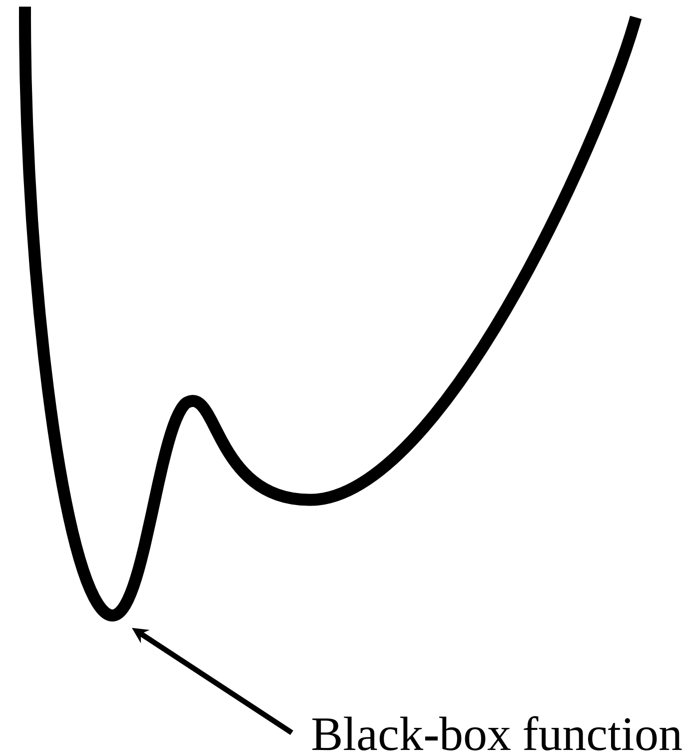
Optimization Model

Add algebraic constraints $h(x)=0$: design specs, heat/mass balances, and logic constraints

ALAMO

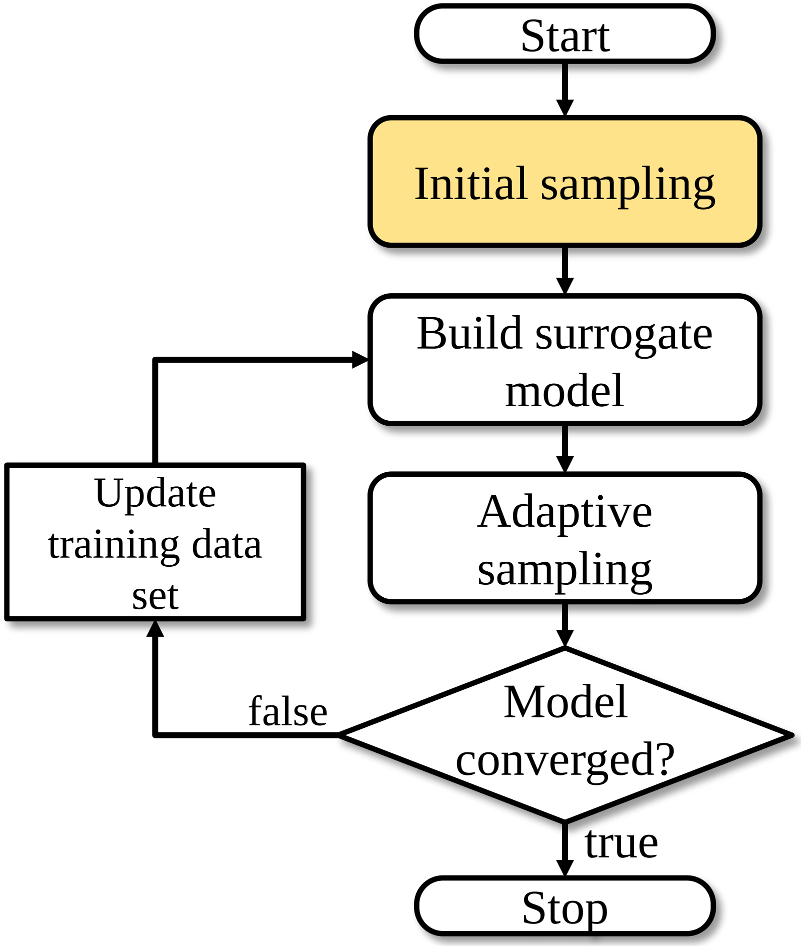
Automated Learning of Algebraic Models for Optimization

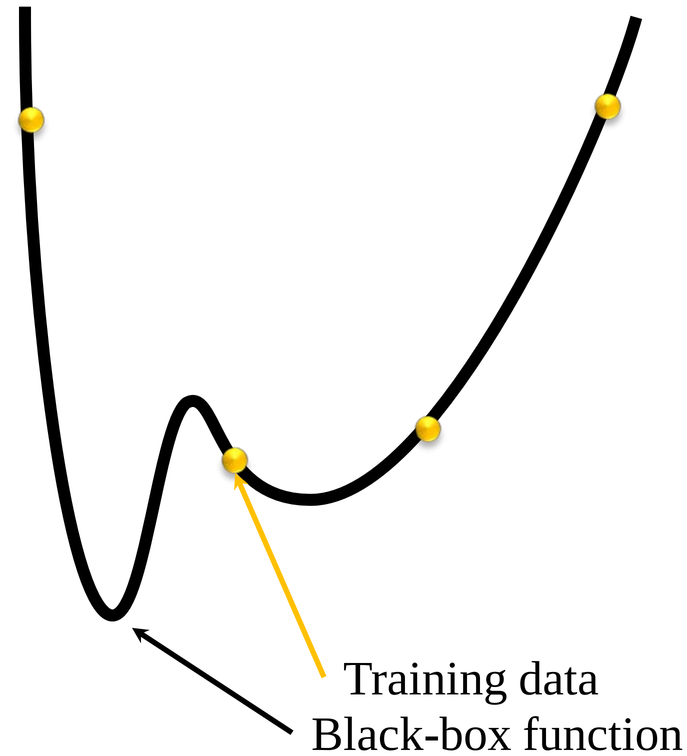




ALAMO

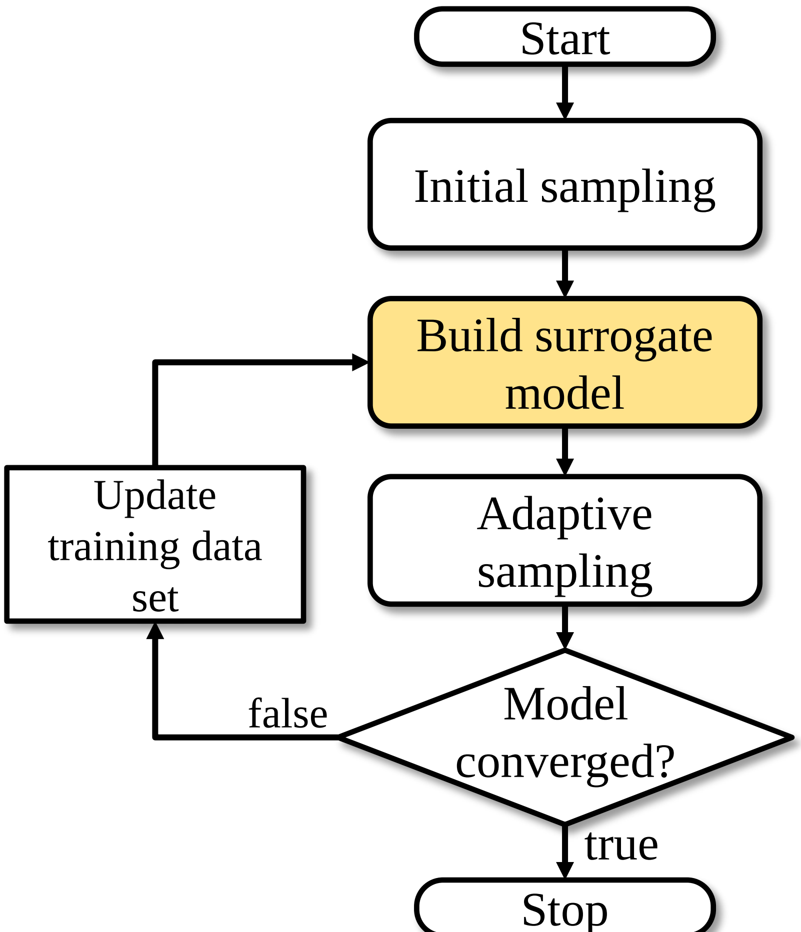
Automated Learning of Algebraic Models for Optimization

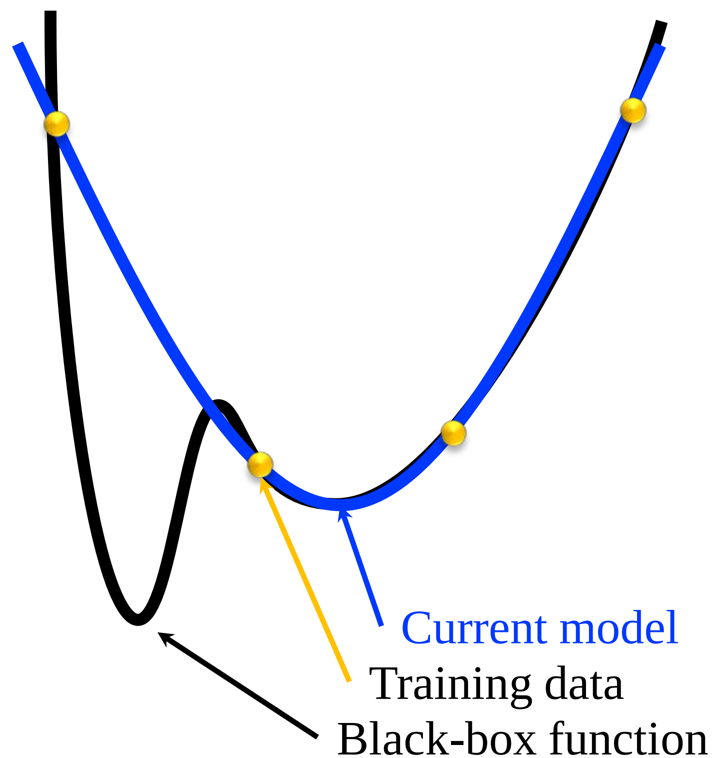




ALAMO

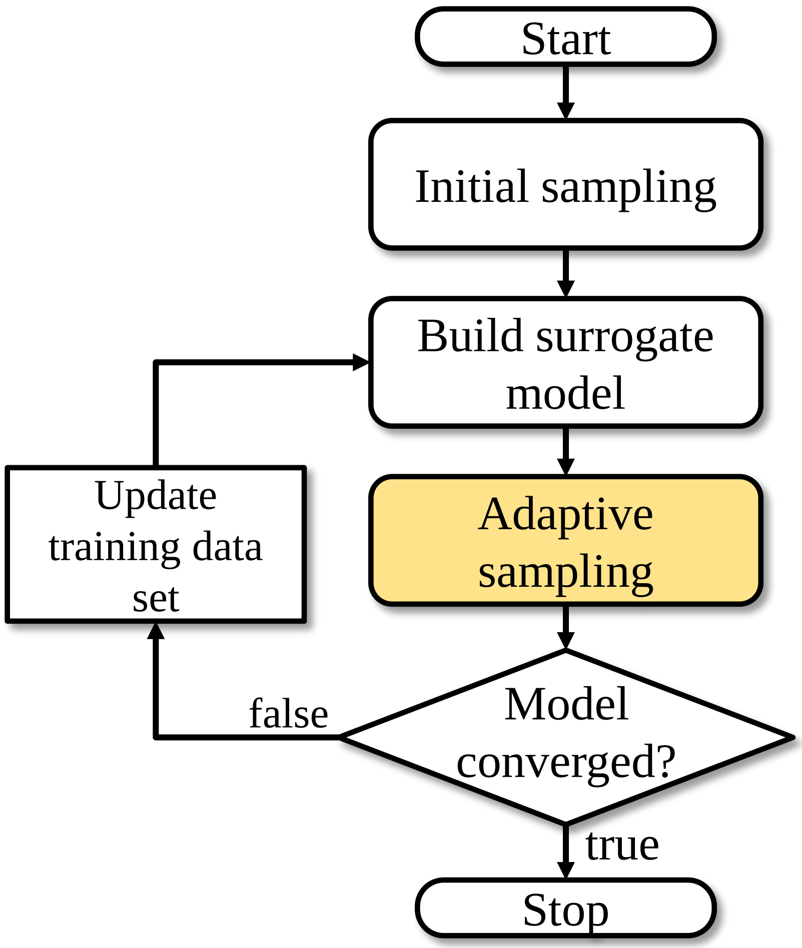
Automated Learning of Algebraic Models for Optimization

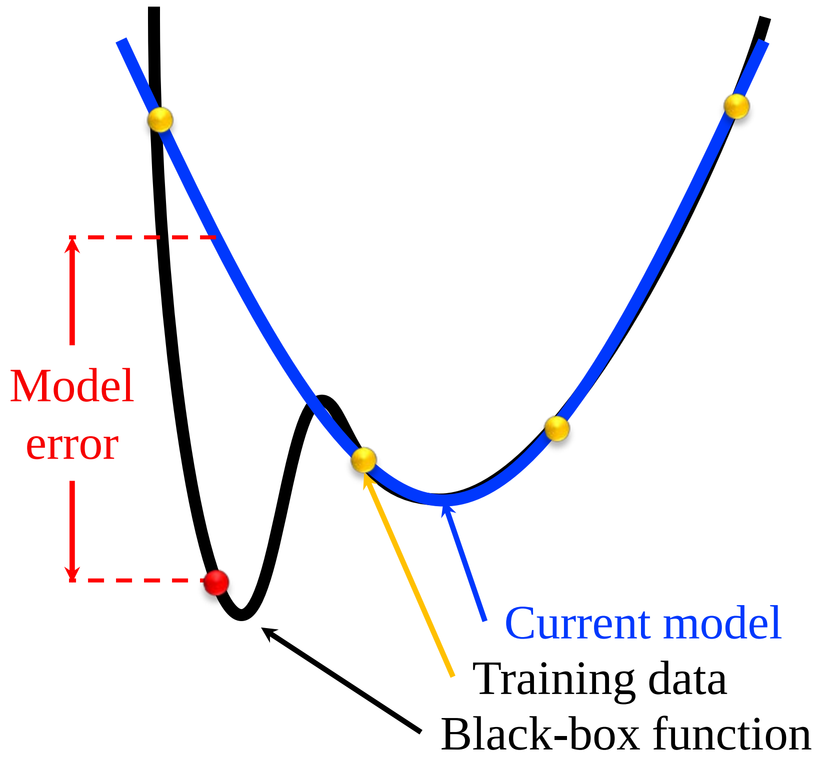




ALAMO

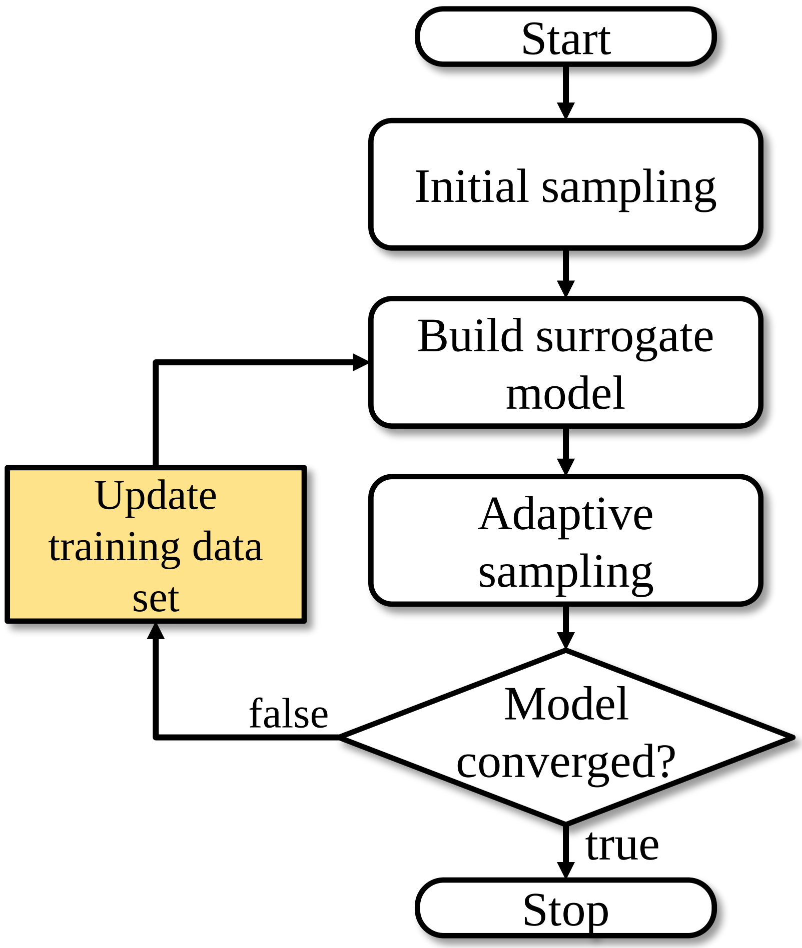
Automated Learning of Algebraic Models for Optimization

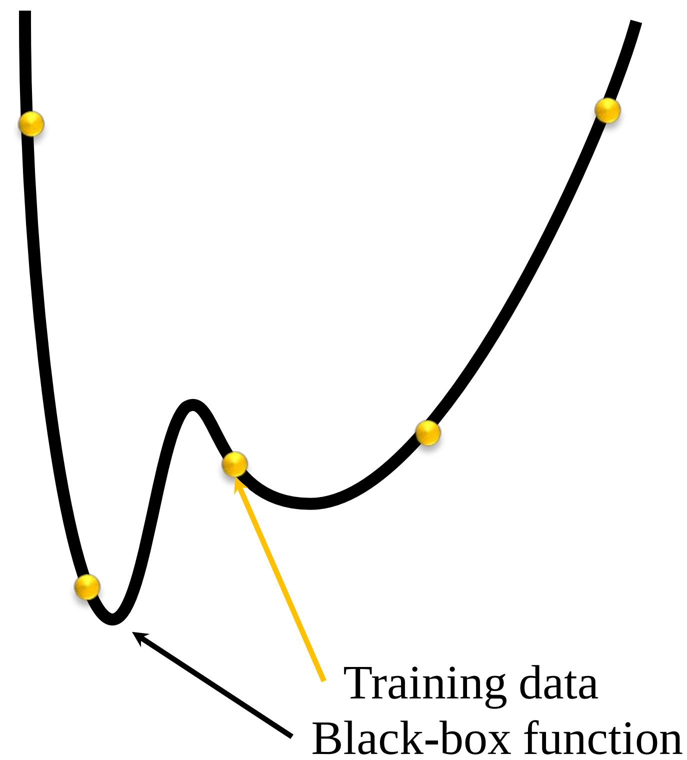




ALAMO

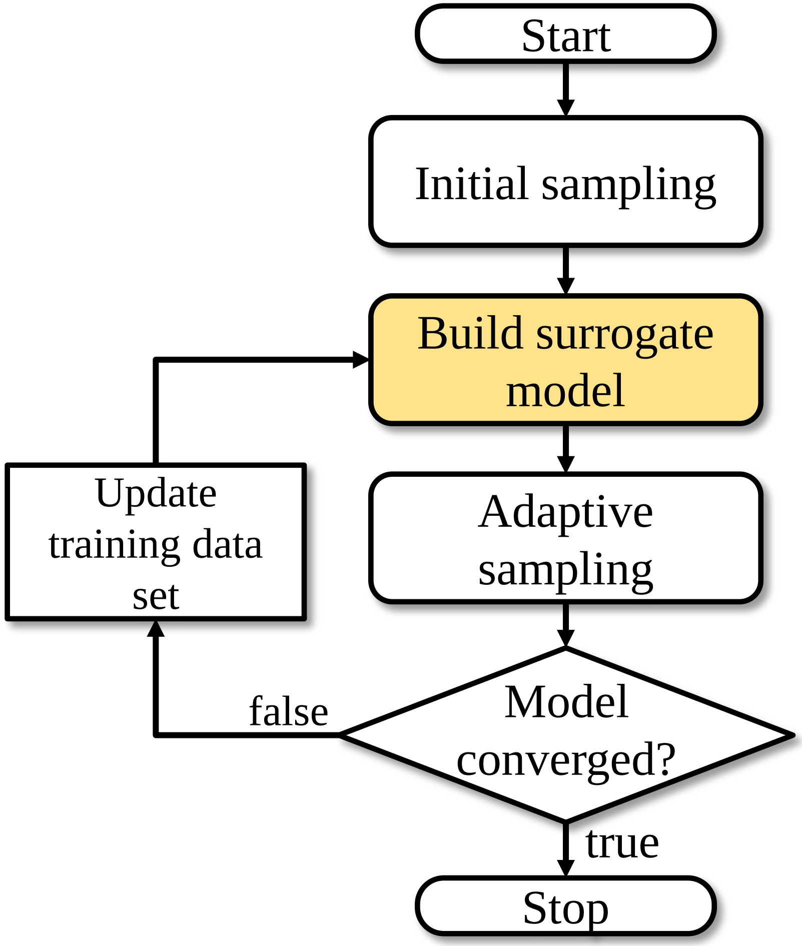
Automated **L**earning of **A**lgebraic **M**odels for **O**ptimization

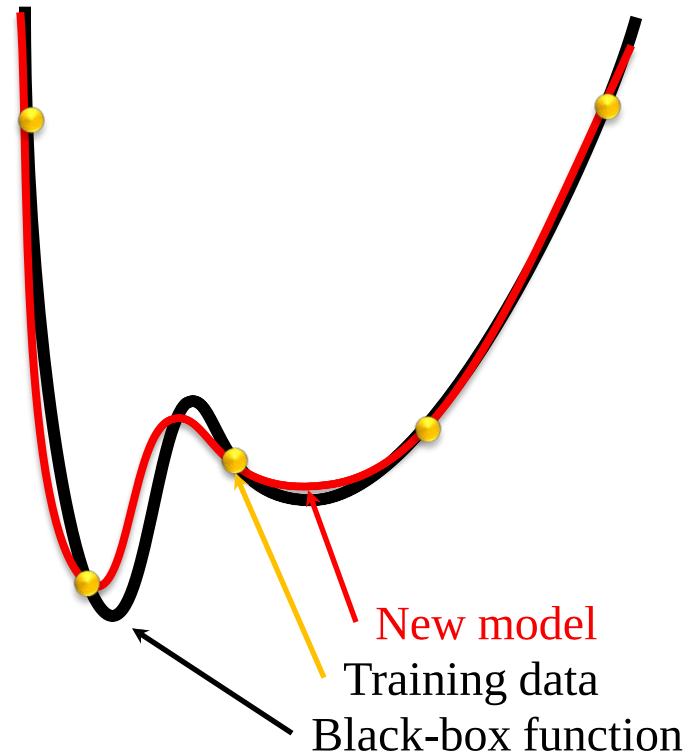




ALAMO

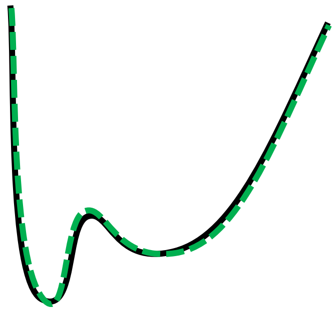
Automated Learning of Algebraic Models for Optimization





HOW TO BUILD THE SURROGATES

- **We aim to build surrogate models that are**
  - ✓ **Accurate**
    - ✓ ***We want to reflect the true nature of the simulation***
  - ✓ **Tailored for algebraic optimization**

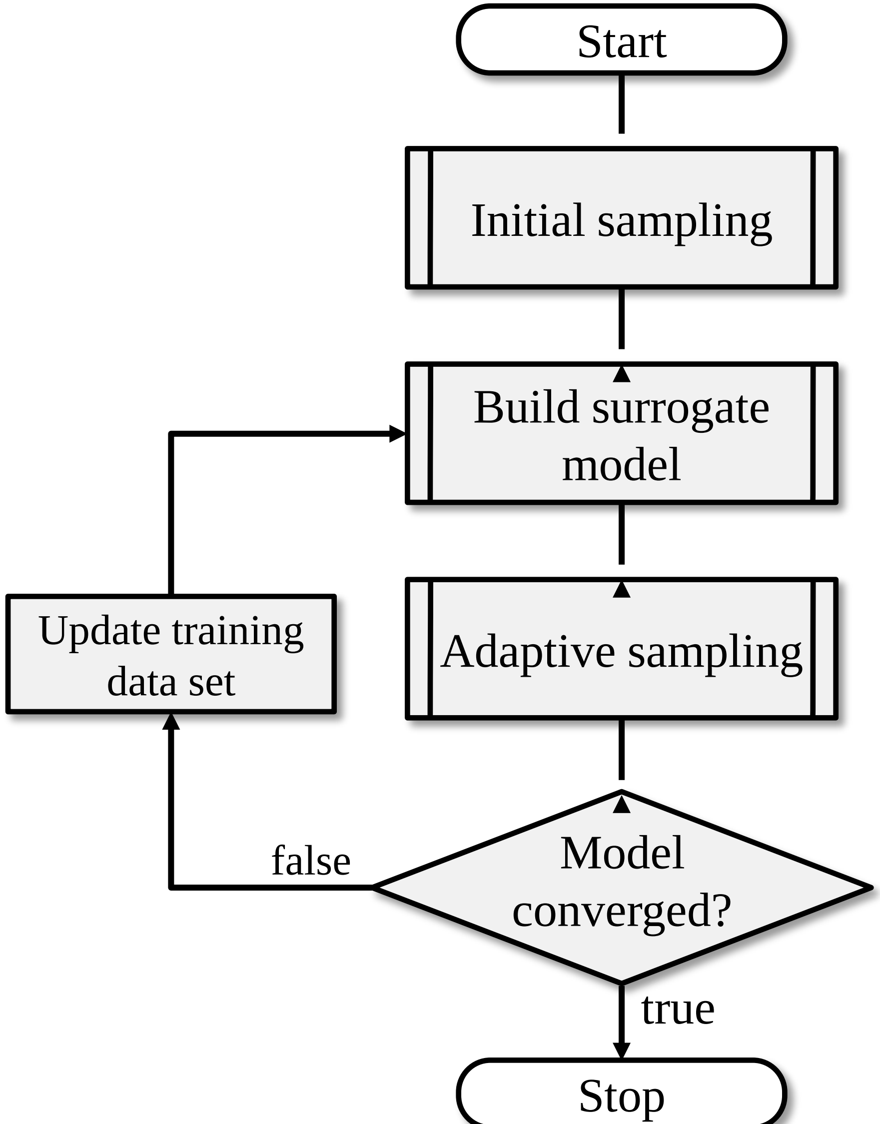


$$\hat{f}(x) = \sum_{i=1}^{n} \gamma_i exp\left(\frac{\|x\|}{\sigma^2}\right) + \beta_0 + \beta_1\, x + \ldots$$

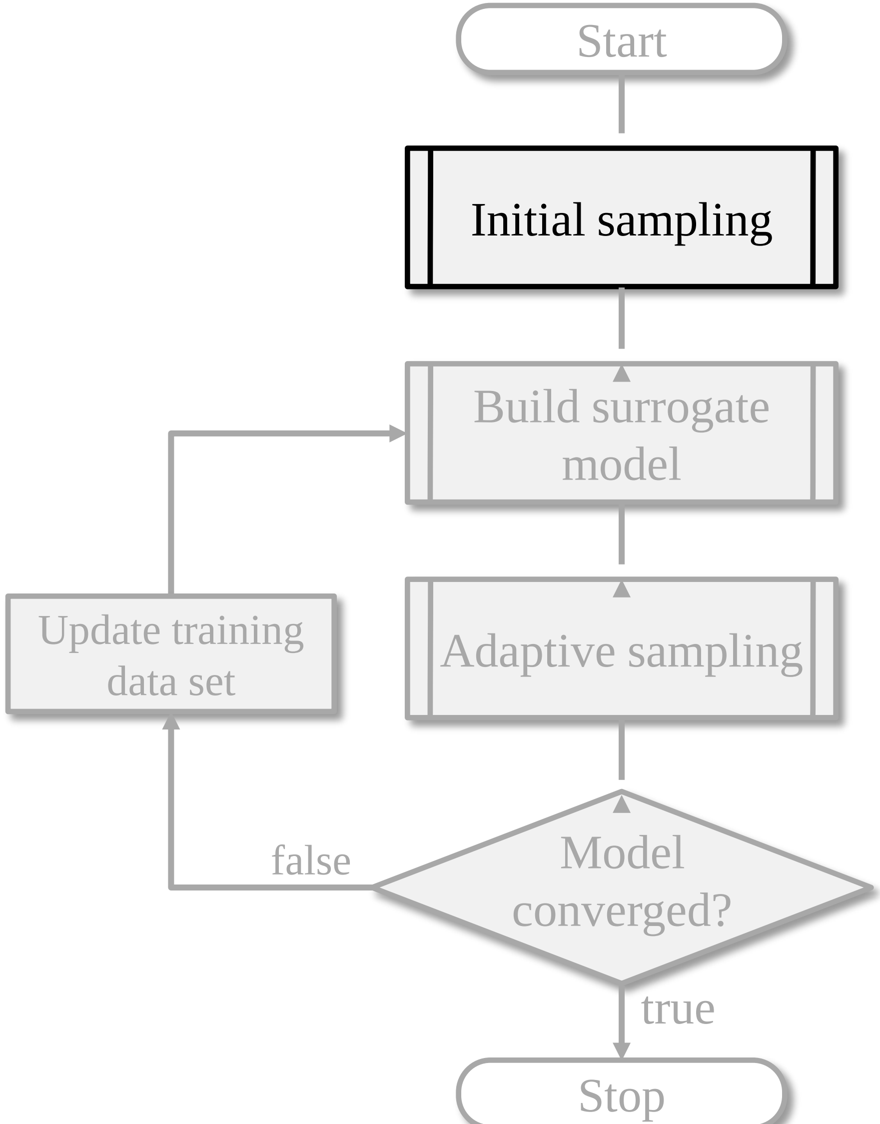
$$\hat{f}(x) = \beta_1\, x + \beta_2\, x^2 + \beta_3\, x^3 + \beta_4\, e^x$$

  - ✓ **Generated from a minimal data set**

ALGORITHMIC FLOWSHEET



ALGORITHMIC FLOWSHEET

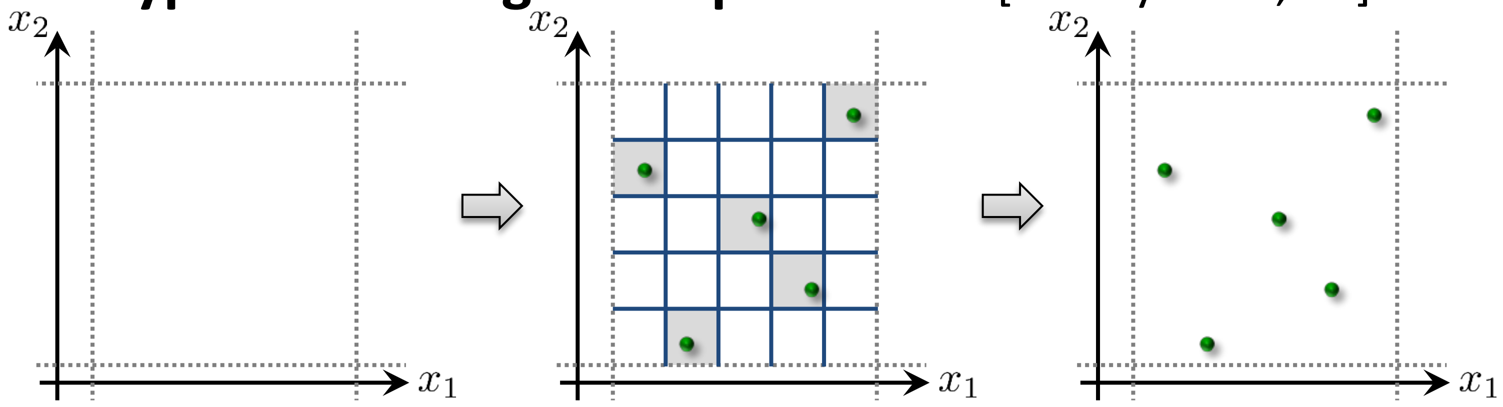


DESIGN OF EXPERIMENTS

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

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

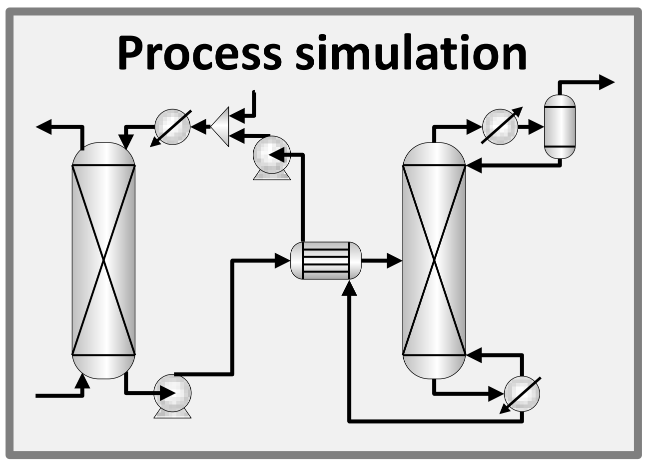
- **Latin hypercube design of experiments** [McKay et al., 79]



INITIAL SAMPLING

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

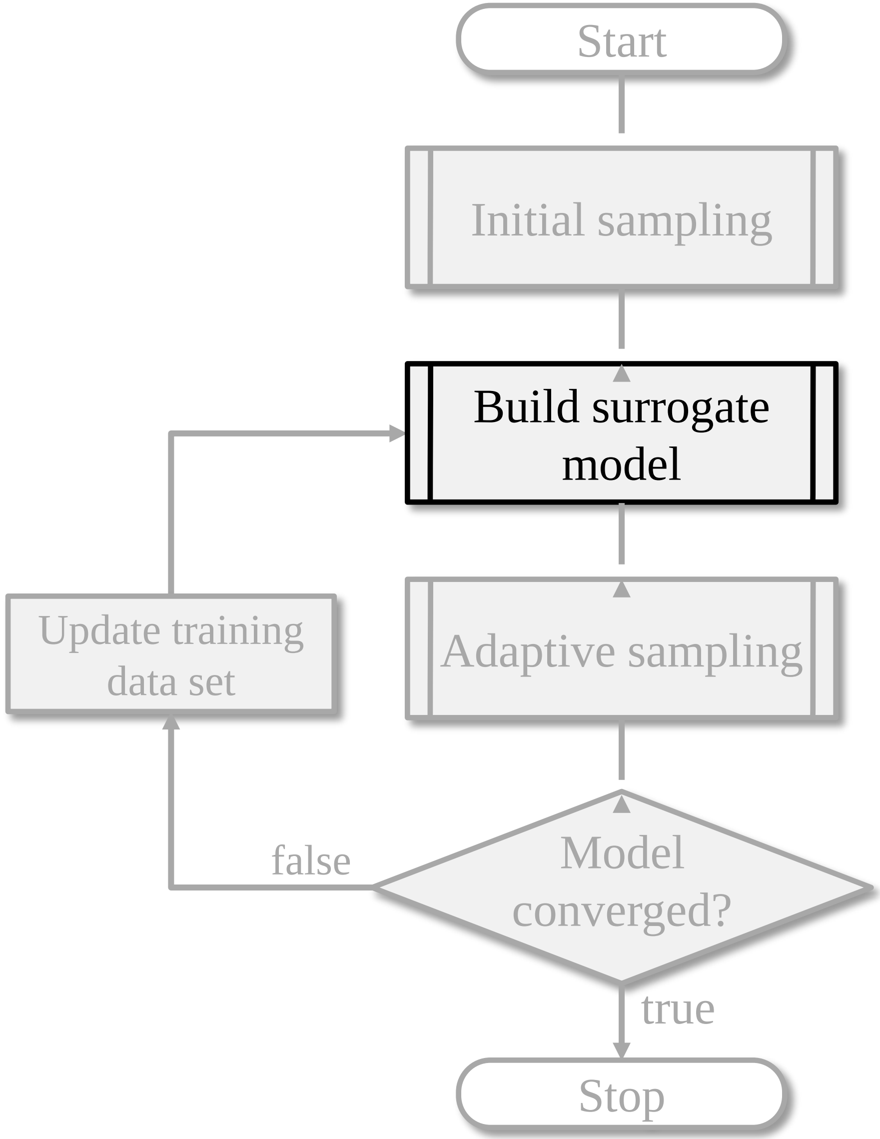
$$x = \begin{pmatrix} x^1 & x^2 & \cdots & x^i & \cdots & x^N \end{pmatrix}$$



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

Initial training set

ALGORITHMIC FLOWSHEET



MODEL IDENTIFICATION

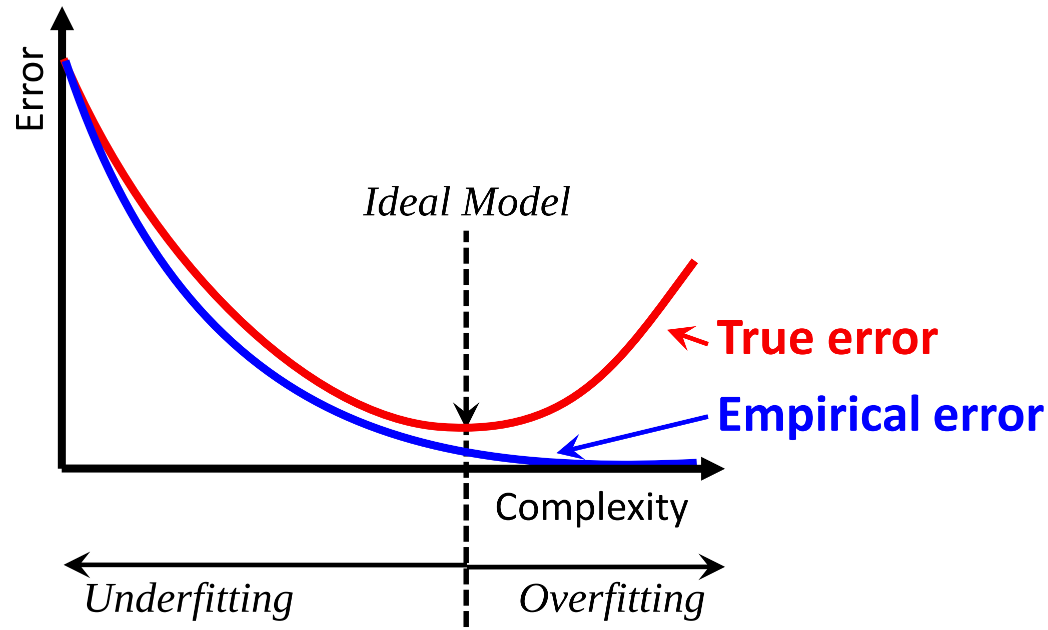
- **Goal: Identify the functional form and complexity of the surrogate models**

$$z = f(x)$$

- **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



Step 1: Define a large set of potential basis functions

$$\hat{z}(x_1) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 \frac{x_1}{x_2} + \beta_5 \frac{x_2}{x_1} + \beta_6 e^{x_1} + \beta_7 e^{x_2} + \ldots$$

Step 2: Model reduction

$$\hat{z}(x) = \beta_0 + \beta_2 x_2 + \beta_5 \frac{x_2}{x_1} + \beta_7 e^{x_2}$$

BEST SUBSET METHOD

- **Generalized best subset problem:**

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

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

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 AICC

- **Corrected Akaike information criterion (AICc)** [Hurvich and Tsai, 93]

$$AICc(\mathcal{S}, \beta) = N \log\left(\frac{1}{N}\sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{S}}\beta_j X_{ij}\right)^2\right) + 2|\mathcal{S}| + \frac{2|\mathcal{S}|\left(|\mathcal{S}|+1\right)}{N-|\mathcal{S}|-1}$$

- **Substituting the mixed integer surrogate form into AICc:**

$$AICc(\beta, y_j) = N \log\left(\frac{1}{N}\sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{B}}\left(y_j\beta_j\right)X_{ij}\right)^2\right) + 2\sum_j y_j + \frac{2\sum_j y_j\left(\sum_j y_j + 1\right)}{N-\sum_j y_j - 1}$$

$$\textbf{OR} \text{ if } \sum_j y_j = T$$

$$AICc(\beta, y_j) = N \log\left(\frac{1}{N}\sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{B}}\left(y_j\beta_j\right)X_{ij}\right)^2\right) + 2T + \frac{2T\left(T+1\right)}{N-T-1}$$

MIXED-INTEGER PROBLEM

$$\min_{\beta,T,y} \quad AICc(\beta,T,y) = N\log\left(\frac{1}{N}\sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{B}}(y_j\beta_j)\,X_{ij}\right)^2\right) + 2T + \frac{2T\,(T+1)}{N-T-1}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{B}} y_j = T$$

$$y_j = \{0,1\} \qquad j\in\mathcal{B}$$

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**

a) model sizing

$$\text{General:} \qquad \min_{\beta,T,y} \Phi(\beta,T,y) = \min_T \left\{ \min_{\beta,y} \left[\Phi_{\beta,y}(\beta,y)|_T\right] + \Phi_T(T) \right\}$$

b) basis and coefficient selection

$$AICc(\beta,T): \qquad AICc_{\beta,y}(\beta,y)|_T = N \log\left(\frac{1}{N} \sum_{i=1}^{N} \left(z_i - \sum_{j \in \mathcal{B}} (y_j \beta_j) X_{ij} \right)^2 \right)$$

$$AICc_T(T) = 2T + \frac{2T(T+1)}{N-T-1}$$

  - **Inner minimization objective reformulation**

NESTED MIXED-INTEGER PROBLEM

$$\min_{T \in \{1,\ldots,T^u\}} \quad N \log\left(\frac{1}{N}\sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{B}}(y_j\beta_j)\,X_{ij}\right)^2\right) + 2T + \frac{2T\,(T+1)}{N-T-1}$$

$$\text{s.t.} \quad \min_{\beta,y} \quad \sum_{i=1}^{N}\left(z_i - \sum_{j\in\mathcal{B}}\beta_j X_{ij}\right)^2$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{B}} y_j = T$$

$$\beta^l y_j \leq \beta_j \leq \beta^u y_j \qquad j \in \mathcal{B}$$

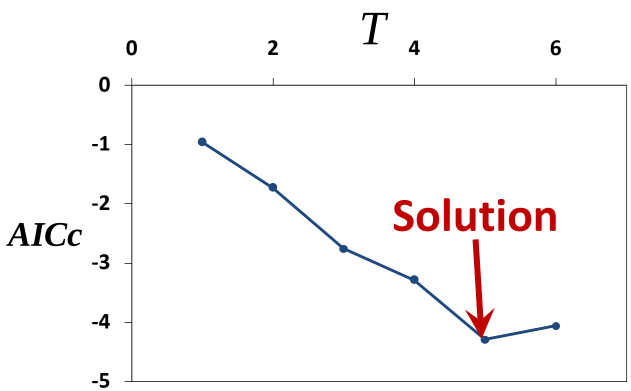
$$y_j = \{0,1\} \qquad j \in \mathcal{B}$$

a) **Model sizing**

b) **Basis and coefficient selection**

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***

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

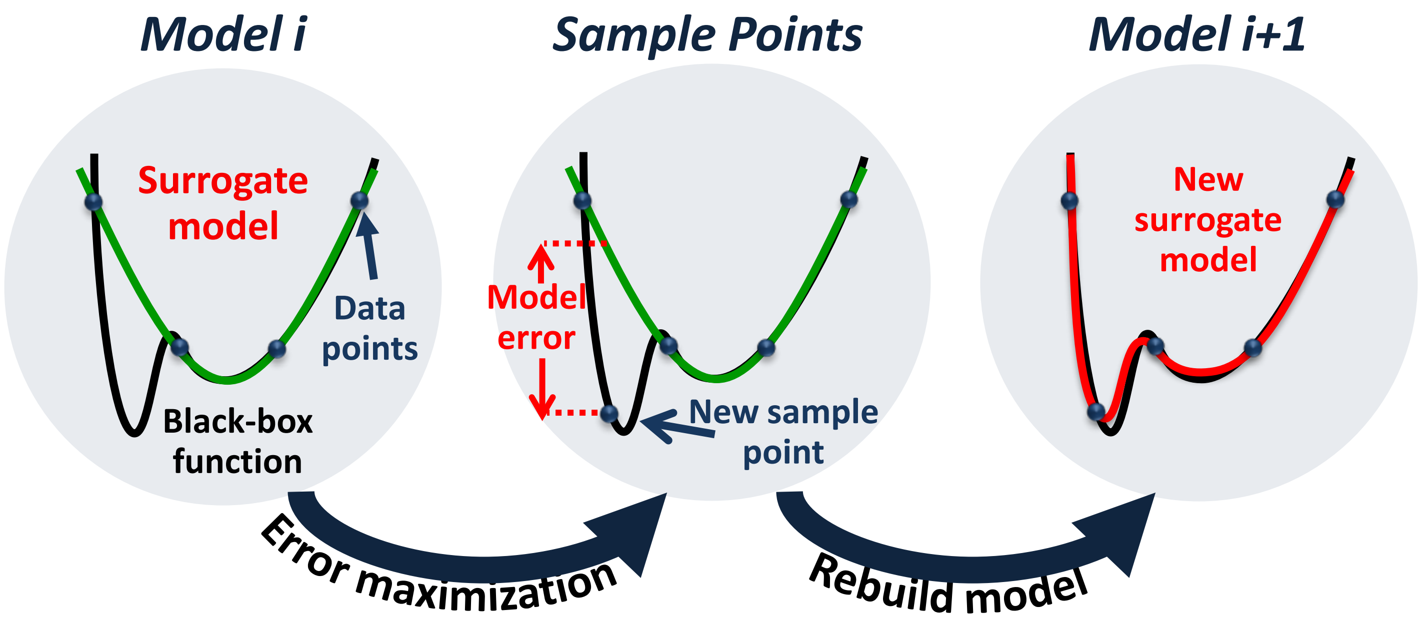
FINAL BEST SUBSET MODEL

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

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

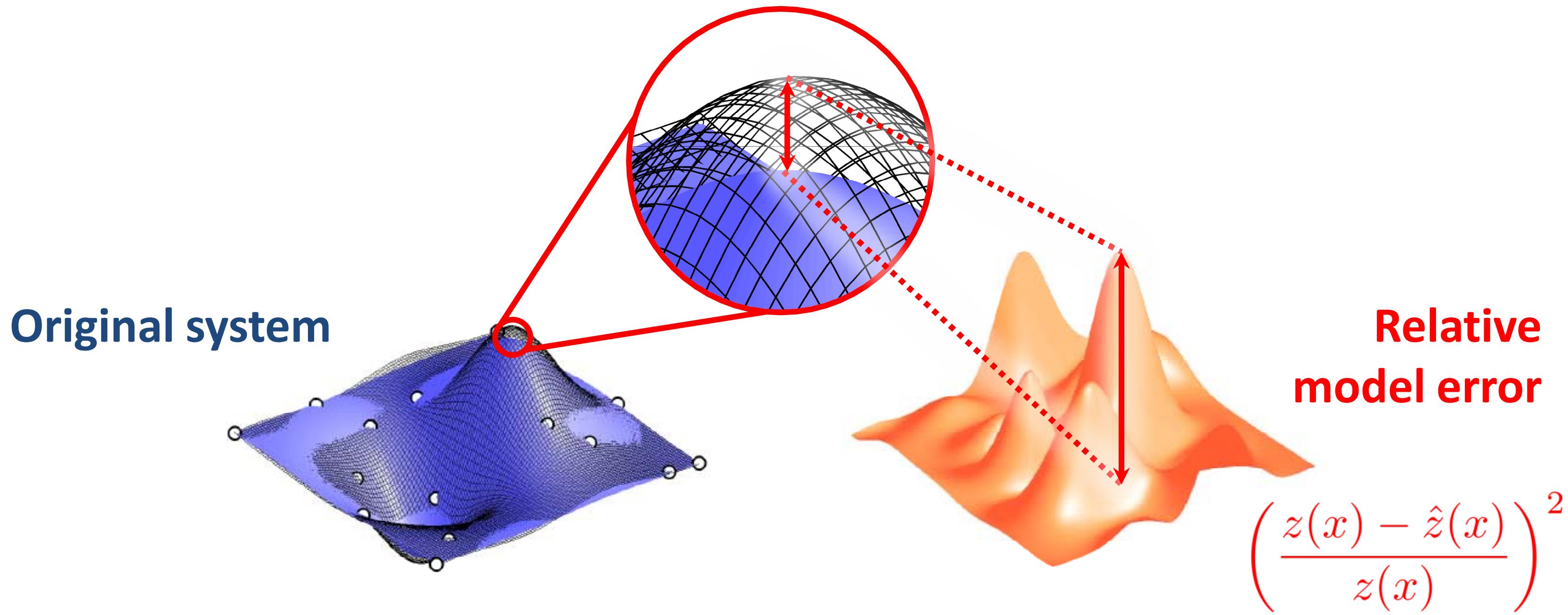
ADAPTIVE SAMPLING

- **Goal: Choose new locations to sample that can best be used to improve the model**
- **Solution: Search the problem space for areas of model inconsistency or model mismatch**



ADAPTIVE SAMPLING

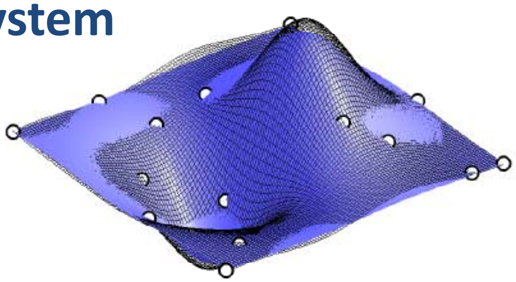
- **Goal: Choose new locations to sample that can best be used to improve the model**
- **Solution: Search the problem space for areas of model inconsistency or model mismatch**



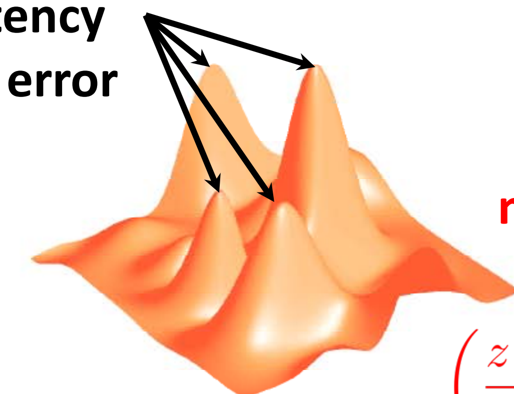
ADAPTIVE SAMPLING

- **Goal: Choose new locations to sample that can best be used to improve the model**
- **Solution: Search the problem space for areas of model inconsistency or model mismatch**

Original system



High model inconsistency or model error



Relative model error

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

ERROR MAXIMIZATION 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**

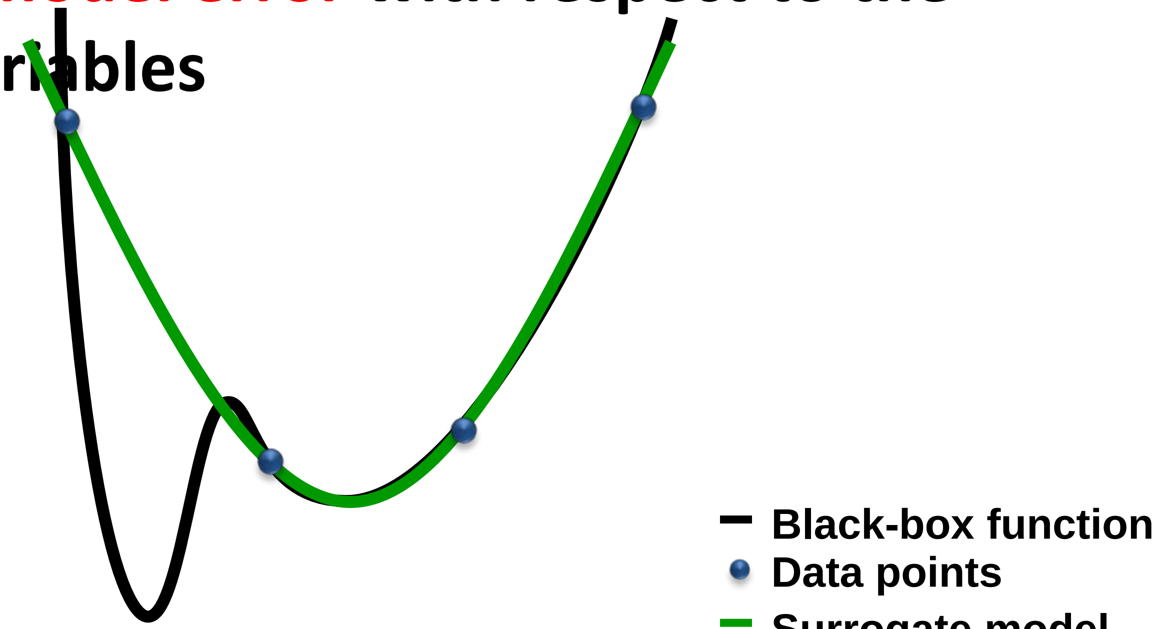
Surrogate model

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

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

ADAPTIVE SAMPLING

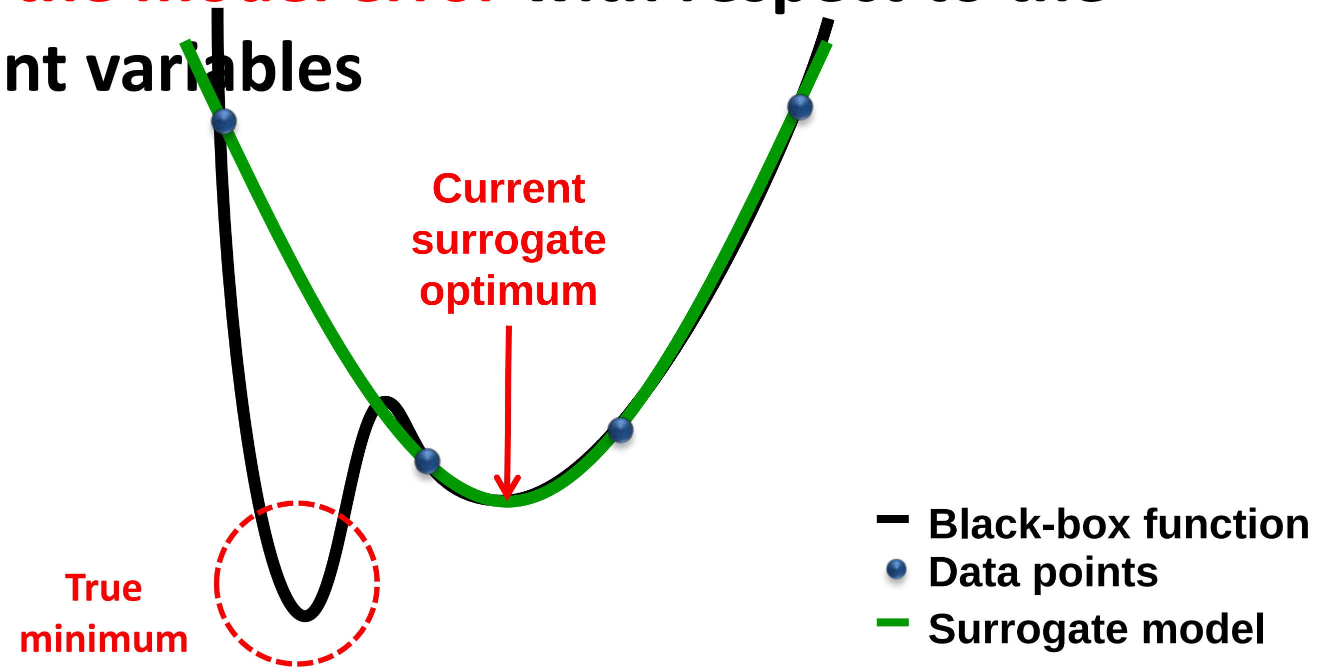
- Goal: Search the problem space for areas of model inconsistency or model mismatch
- More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables



ADAPTIVE SAMPLING

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

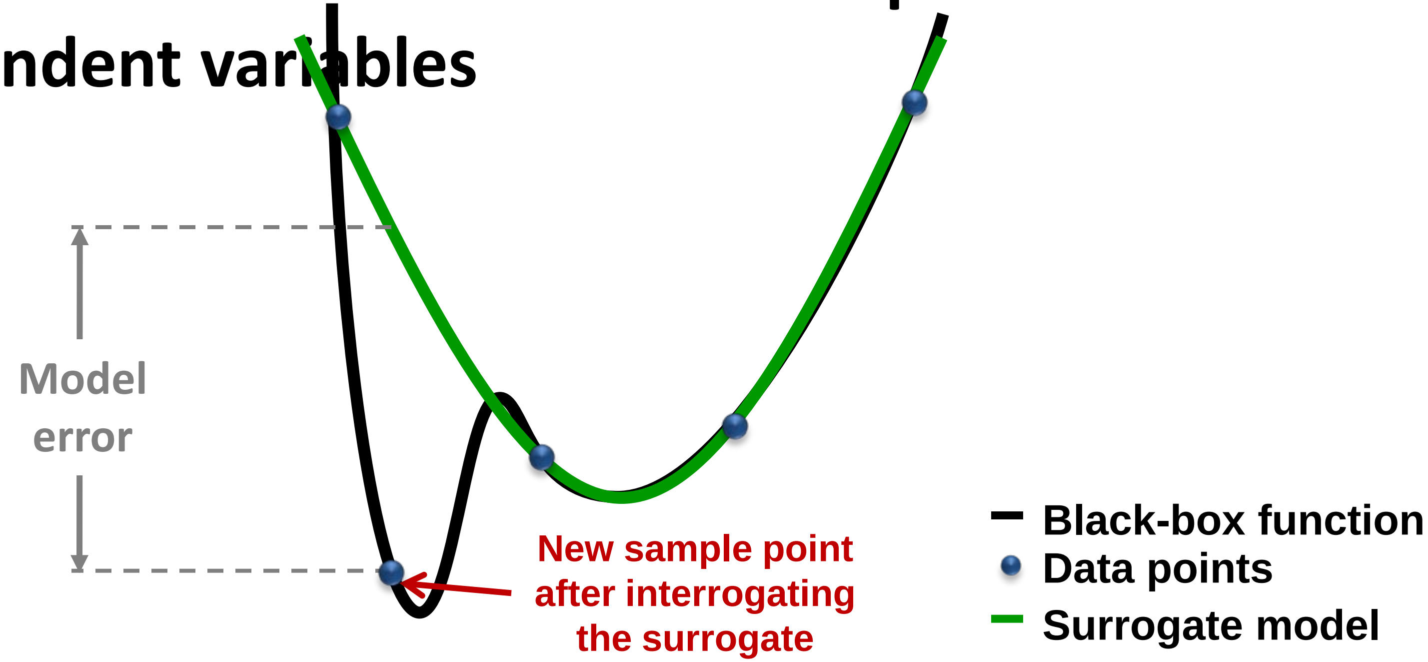
- More succinctly, we are trying to find points that **maximizes the model error** with respect to the independent variables



ADAPTIVE SAMPLING

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

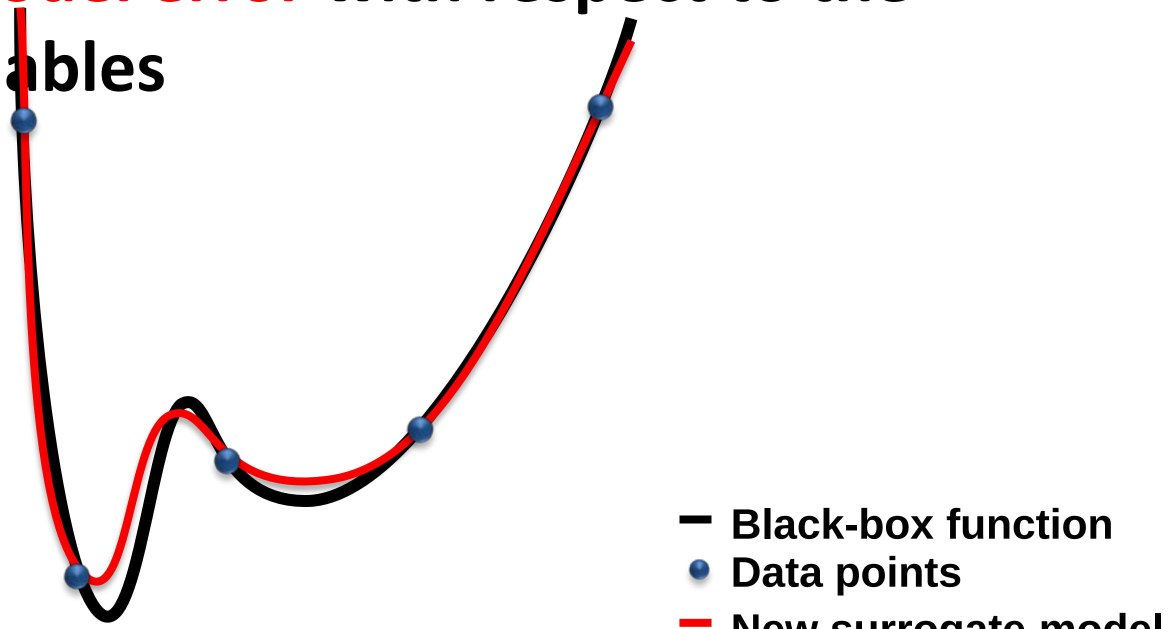
- More succinctly, we are trying to find points that maximizes the model error with respect to the independent variables



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**

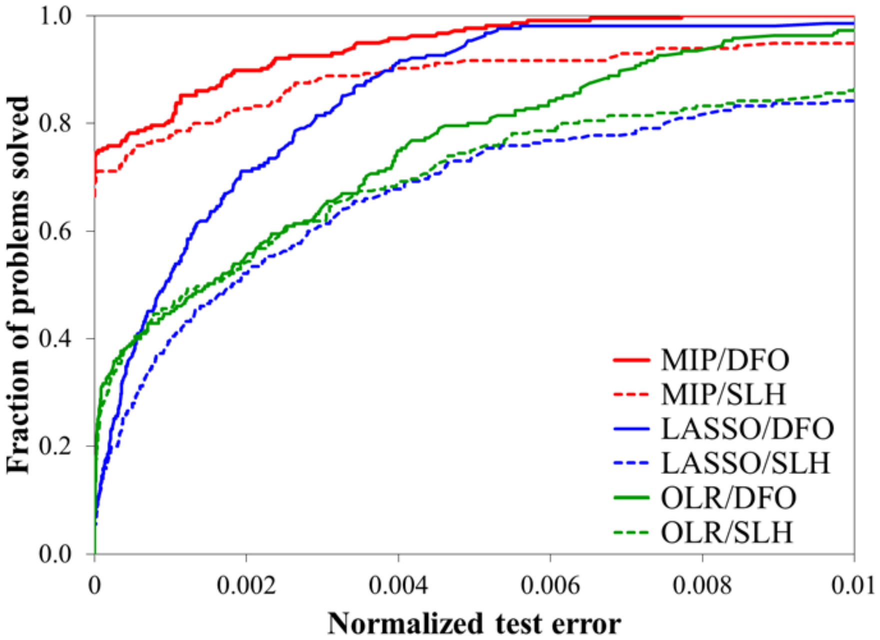
- **Basis functions allowed:**

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	$\exp\left(\frac{x_d}{\gamma}\right)^\alpha$, $\log\left(\frac{x_d}{\gamma}\right)^\alpha$	$\alpha = 1,\ \gamma = 1$

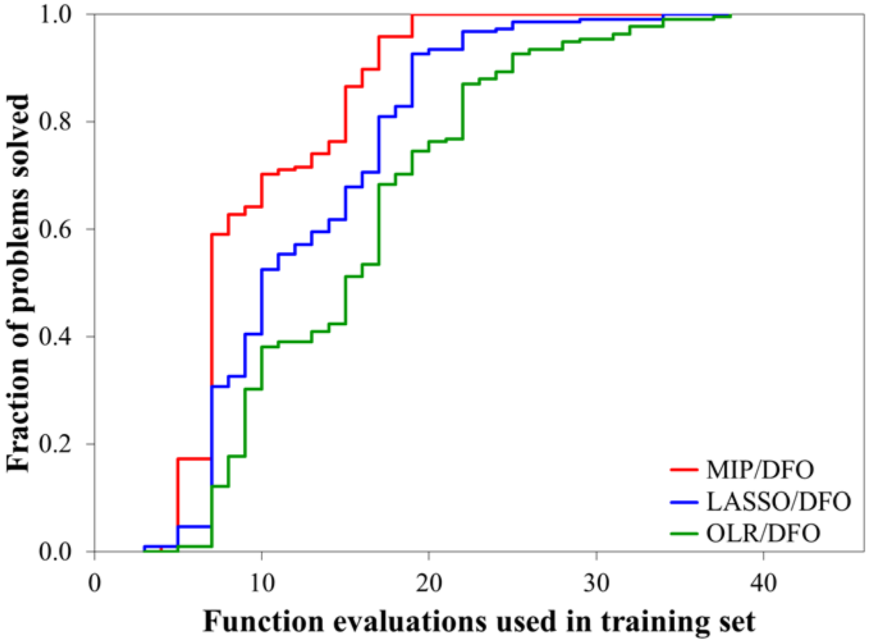
True basis function coefficients were randomly chosen from a uniform distribution where $\beta \in [-1, 1]$.

RESULTS – TEST SET A

Model accuracy



Function evaluations



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

MODEL COMPLEXITY – TEST SET A

No. inputs	No. true terms	MIP/ DFO	MIP/ 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

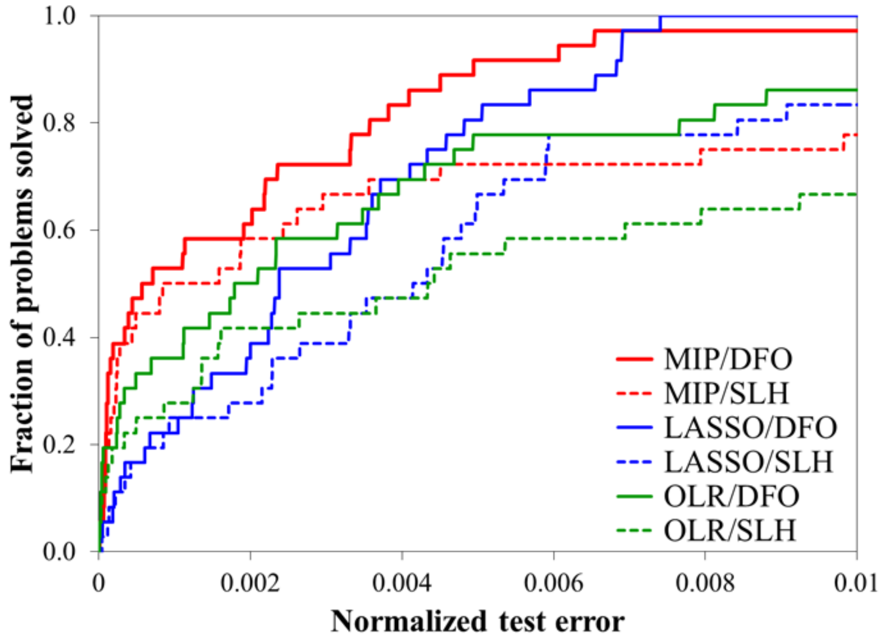
- **Two input black-box functions with basis functions unavailable to the algorithms with**

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	$z(x) = \dfrac{\beta}{\gamma + x_i^\alpha}$

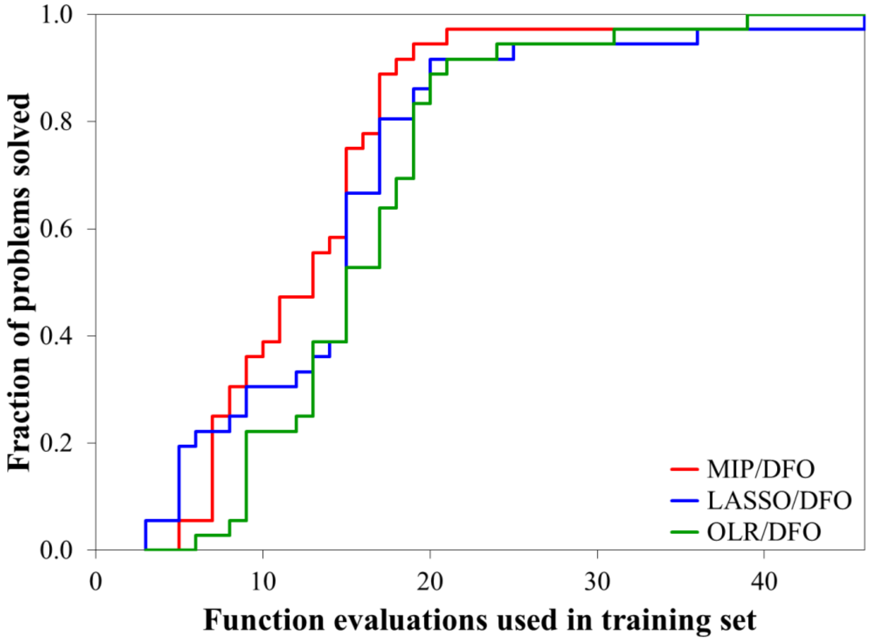
with true parameters chosen from a uniform distribution where $\beta \in [-1, 1]$, $\alpha, \nu \in [-3, 3]$, $\gamma \in [-5, 5]$, and $i, j \in \{1, 2\}$.

RESULTS – TEST SET B

Model accuracy



Function evaluations



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

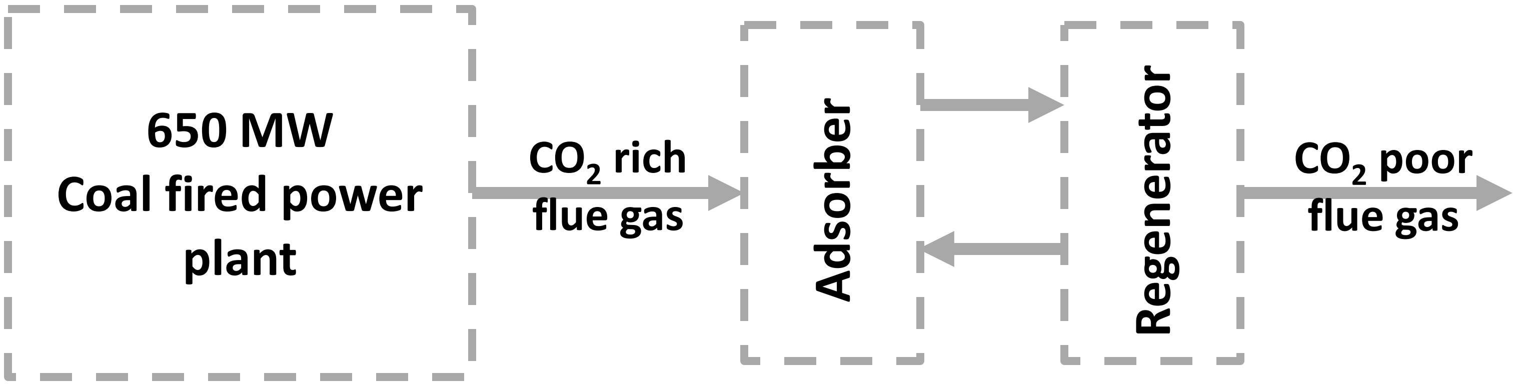
MODEL COMPLEXITY – TEST SET B

True function type	Function ID	MIP/ DFO	MIP/ SLH	LASSO/ DFO	LASSO/ SLH	OLR/ DFO	OLR/ SLH
I	a	5	5	[3, 5]	[4, 9]	[6, 17]	[6, 17]
I	b	[4, 10]	[4, 10]	[10, 14]	[5, 8]	[8, 17]	[8, 17]
I	c	[3, 10]	[6, 9]	[8, 9]	[4, 10]	[13, 17]	[13, 17]
II	a	[4, 6]	[4, 10]	[8, 15]	[7, 9]	[15, 19]	[15, 19]
II	b	[1, 7]	[1, 9]	[13, 16]	[11, 17]	[13, 30]	[13, 30]
II	c	[5, 12]	[5, 12]	[9, 13]	[9, 16]	[9, 19]	[9, 19]
III	a	[3, 4]	[1, 4]	[2, 5]	[2, 5]	[9, 20]	[9, 20]
III	b	4	[1, 4]	5	5	[9, 20]	[9, 20]
III	c	[3, 4]	[3, 4]	[5, 8]	[5, 9]	[18, 24]	[18, 24]
IV	a	[7, 8]	[4, 10]	[8, 17]	[11, 18]	[13, 19]	[13, 19]
IV	b	[8, 9]	[9, 10]	[8, 12]	[10, 14]	[9, 17]	[9, 17]
IV	c	[6, 9]	[9, 10]	[5, 13]	[4, 12]	[13, 15]	[13, 15]

CARBON CAPTURE OPTIMIZATION

- **Problem statement:**

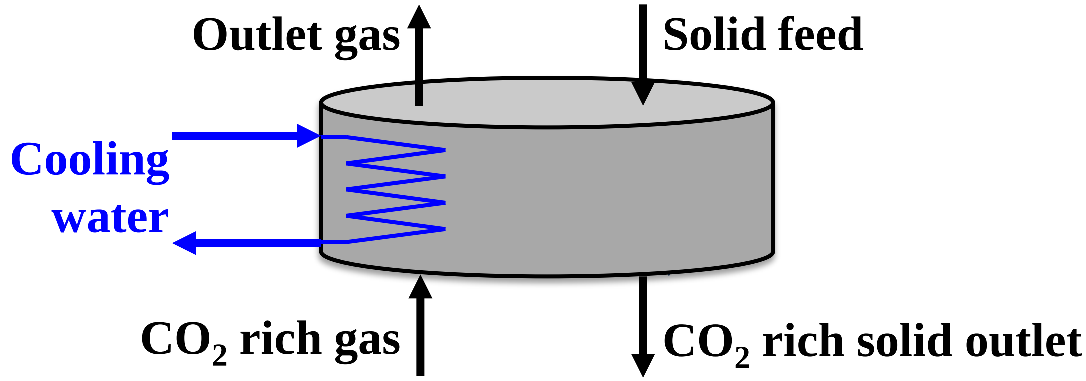
  Capture 90% of CO_2 from a 350MW power plant's post combustion flue gas with minimal increase in the cost of electricity



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

BUBBLING FLUIDIZED BED

Bubbling fluidized bed adsorber diagram



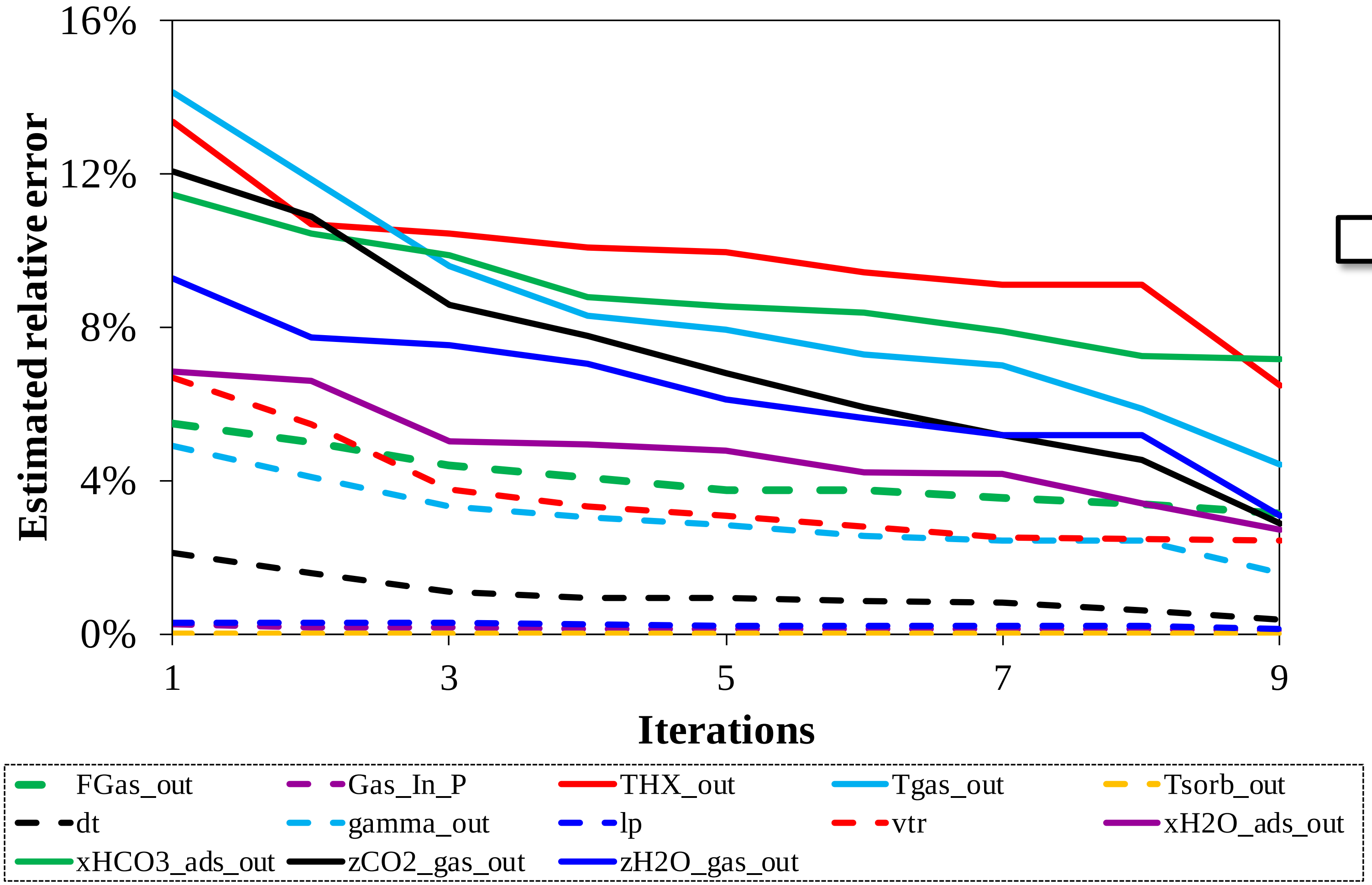
- **Model inputs (14 total)**
  - Geometry (3)
  - Operating conditions (4)
  - Gas mole fractions (2)
  - Solid compositions (2)
  - Flow rates (4)

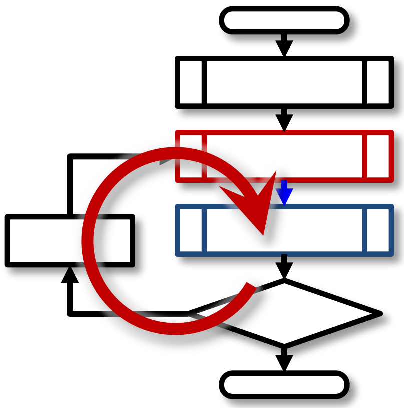
- **Model outputs (13 total)**
  - Geometry required (2)
  - Operating condition required (1)
  - Gas mole fractions (2)
  - Solid compositions (2)
  - Flow rates (2)
  - Outlet temperatures (3)
  - Design constraint (1)

Model created by Andrew Lee at the National Energy and Technology Laboratory

ADAPTIVE SAMPLING

Progression of mean error through the algorithm

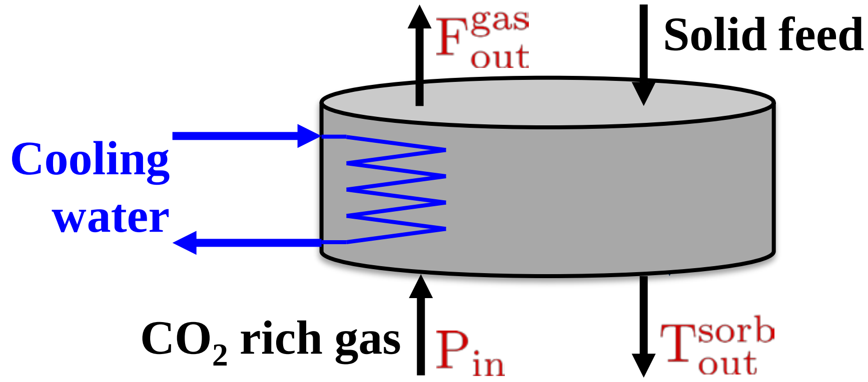




Initial data set: 137 pts

Final data set: 261

EXAMPLE MODELS



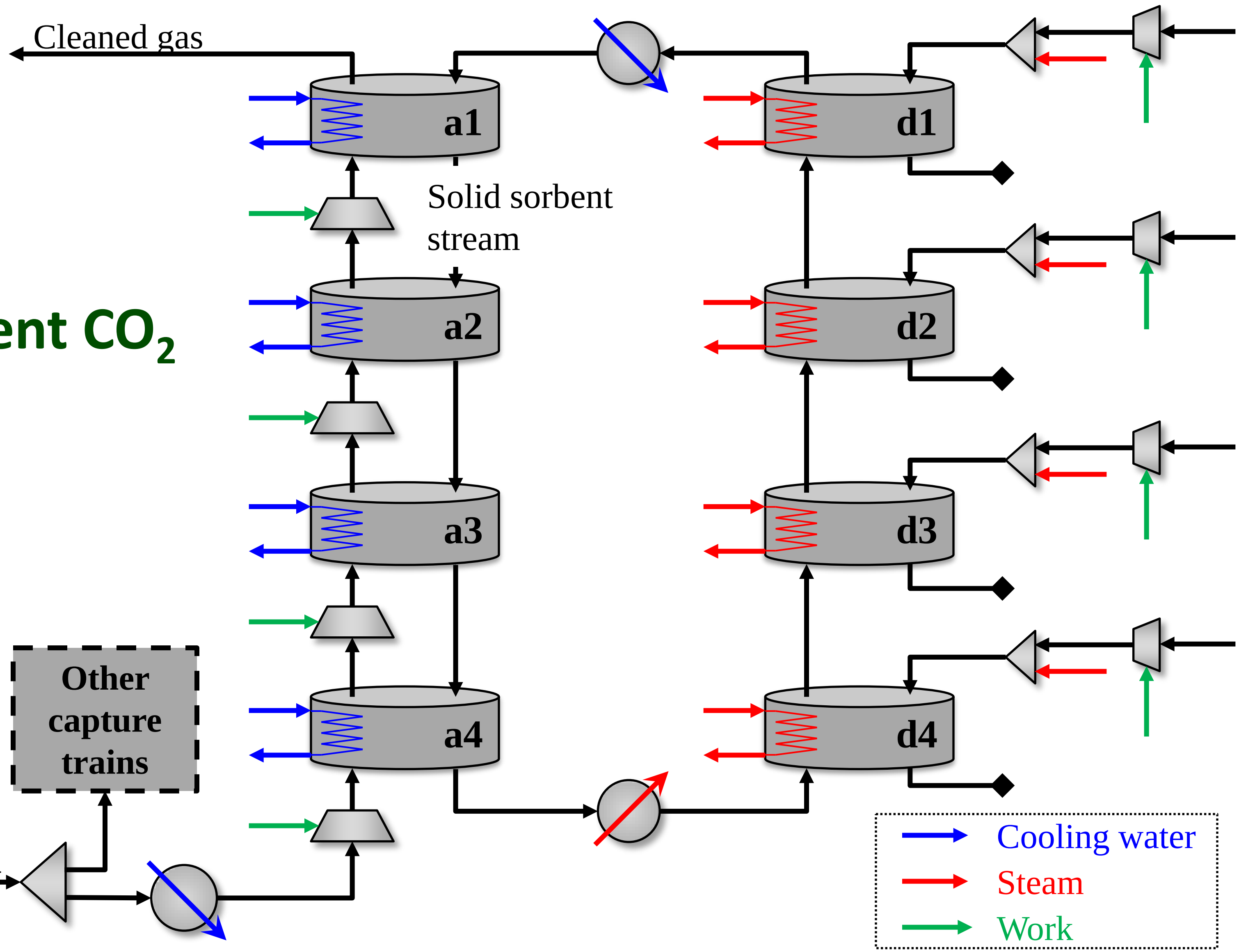
$$\mathrm{P_{in}} = 1.0\,\mathrm{P_{out}} + 0.0231\,L_b - 0.0187\,\ln(0.167\,L_b) - 0.00626\,\ln(0.667\,\mathrm{v_{gi}}) - \frac{51.1\,\mathrm{xHCO3^{ads}_{in}}}{\mathrm{F^{gas}_{in}}}$$

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

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

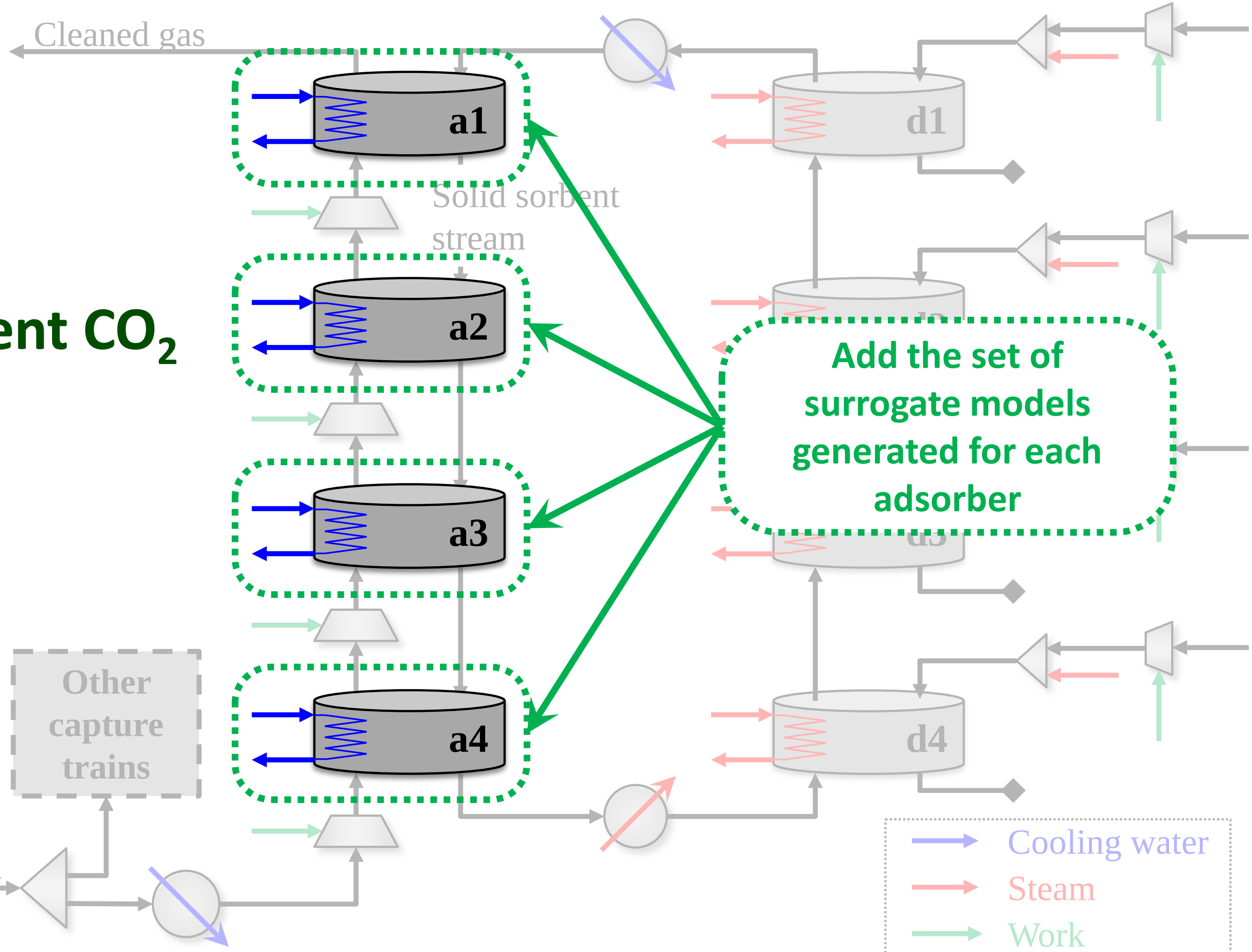
SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 Capture

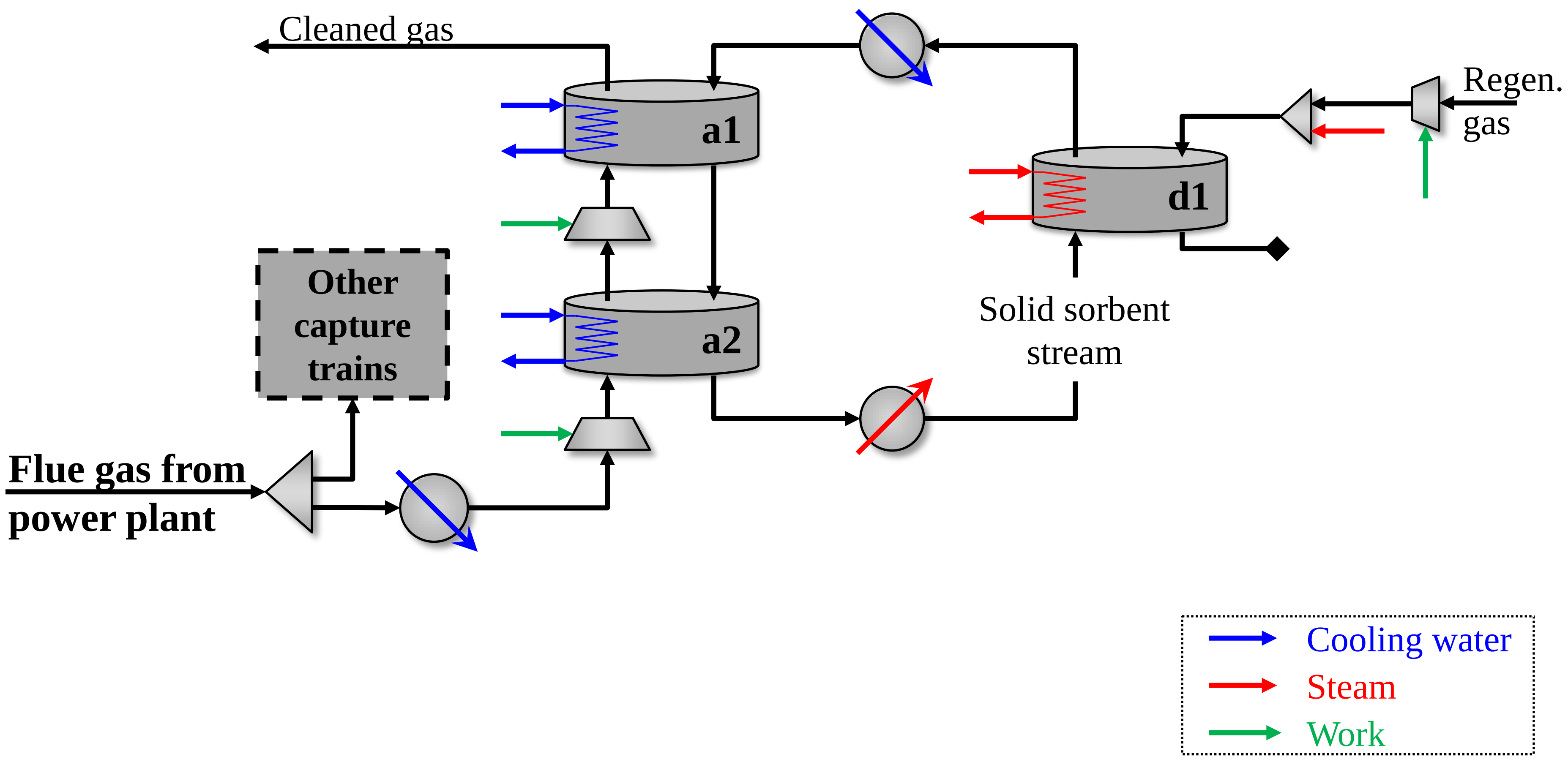


SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 Capture



PRELIMINARY RESULTS



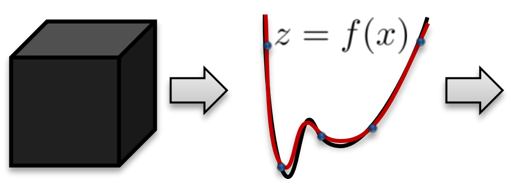
CONCLUSIONS

- **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



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