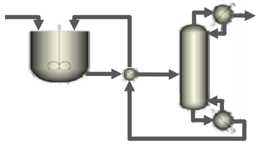
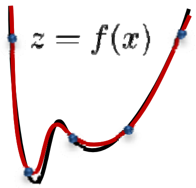




$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

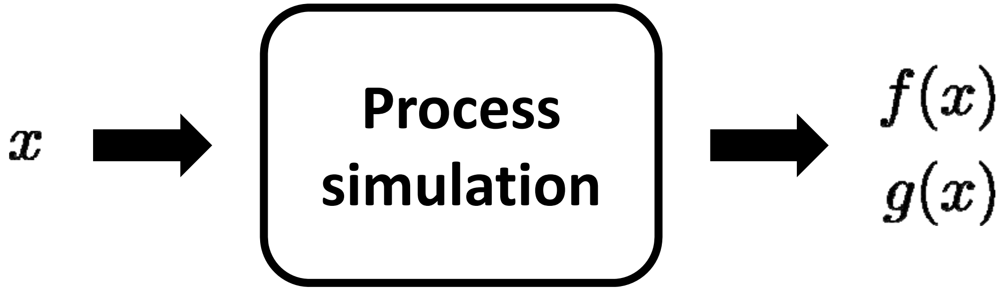
Derivative-Free Optimization Enhanced-Surrogate Models for Energy Systems Optimization

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

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

PROBLEM STATEMENT

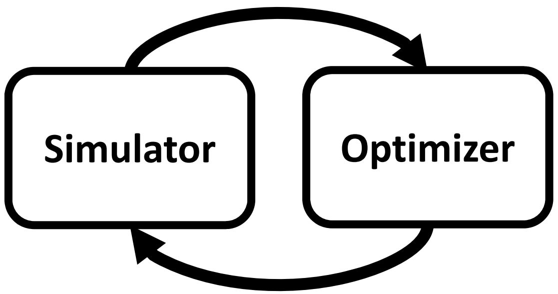
$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$



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

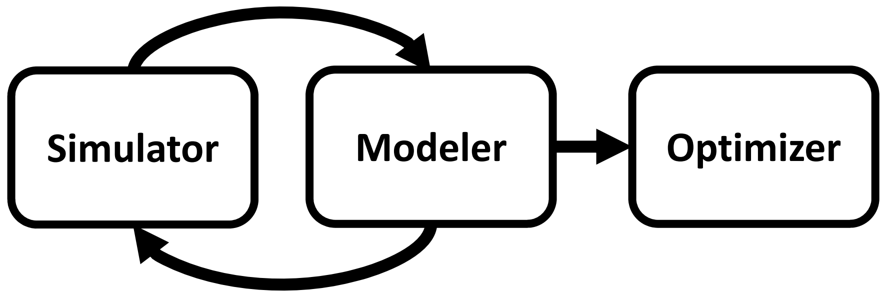
SIMULATION-BASED METHODS

Direct methods



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

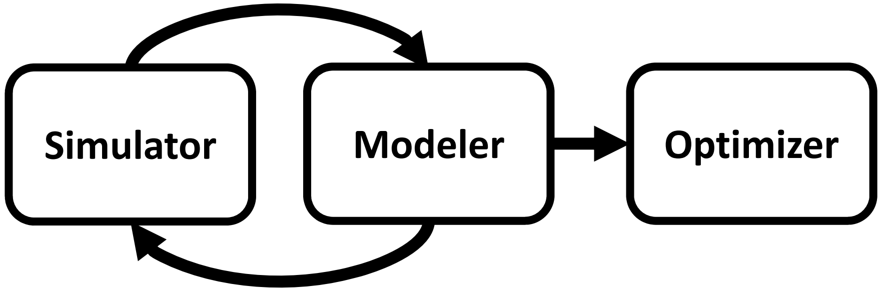
Indirect methods

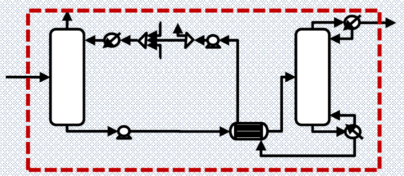
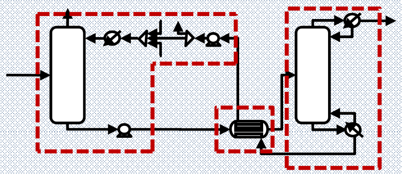


- **What is modeled?**
  - **Objective, objective + constraints, disaggregated system**
- **Type of model**
  - **Linear/nonlinear**
  - **Simple/Complex**
  - **Algebraic/black-box**
- **Optimizer**
  - **Derivative/derivative-free**

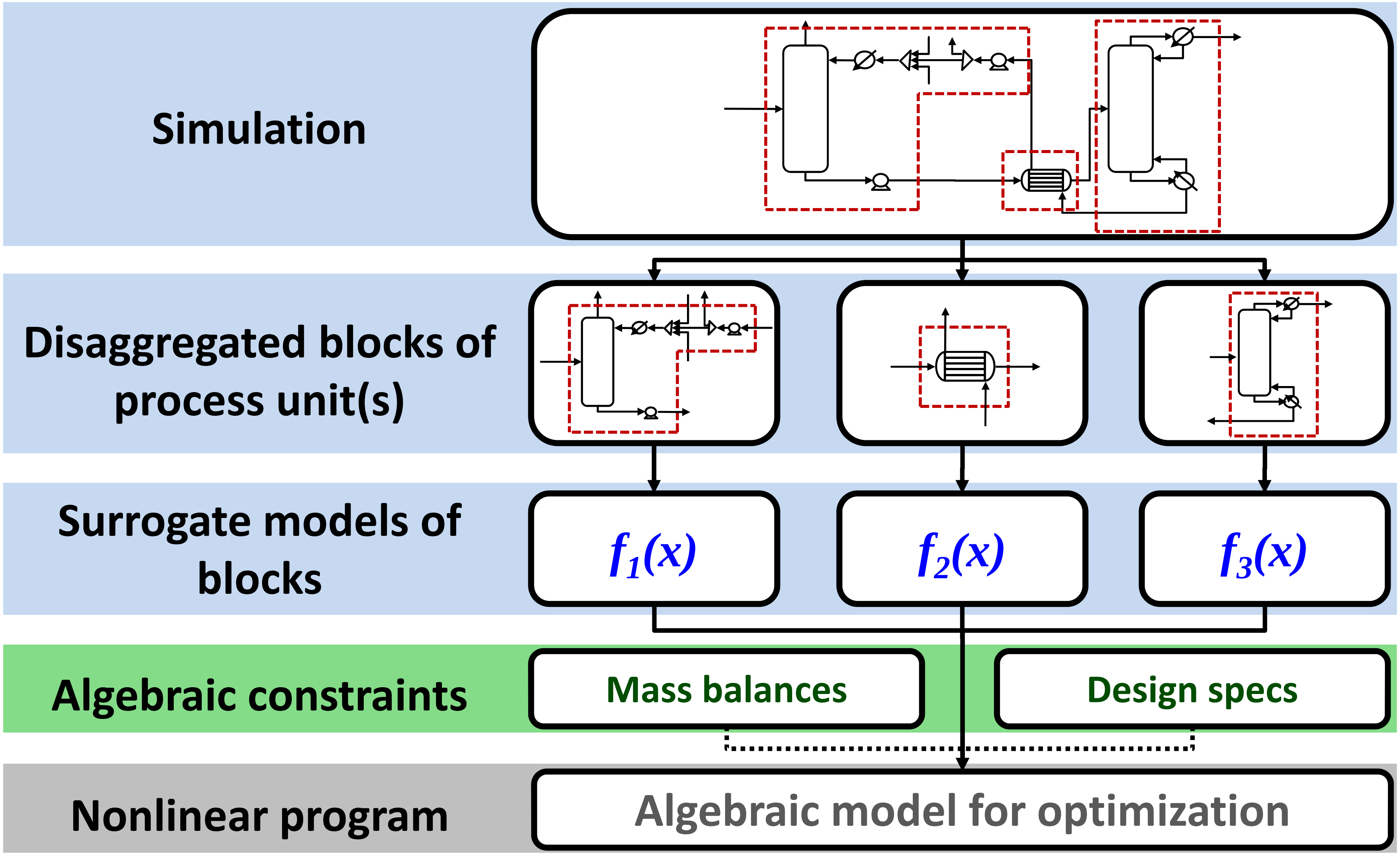
RECENT WORK IN CHEMICAL ENG

Indirect methods



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

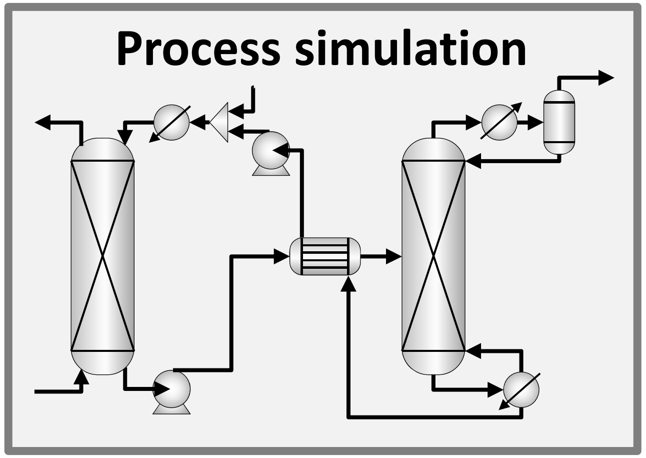
PROCESS DISAGGREGATION



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.

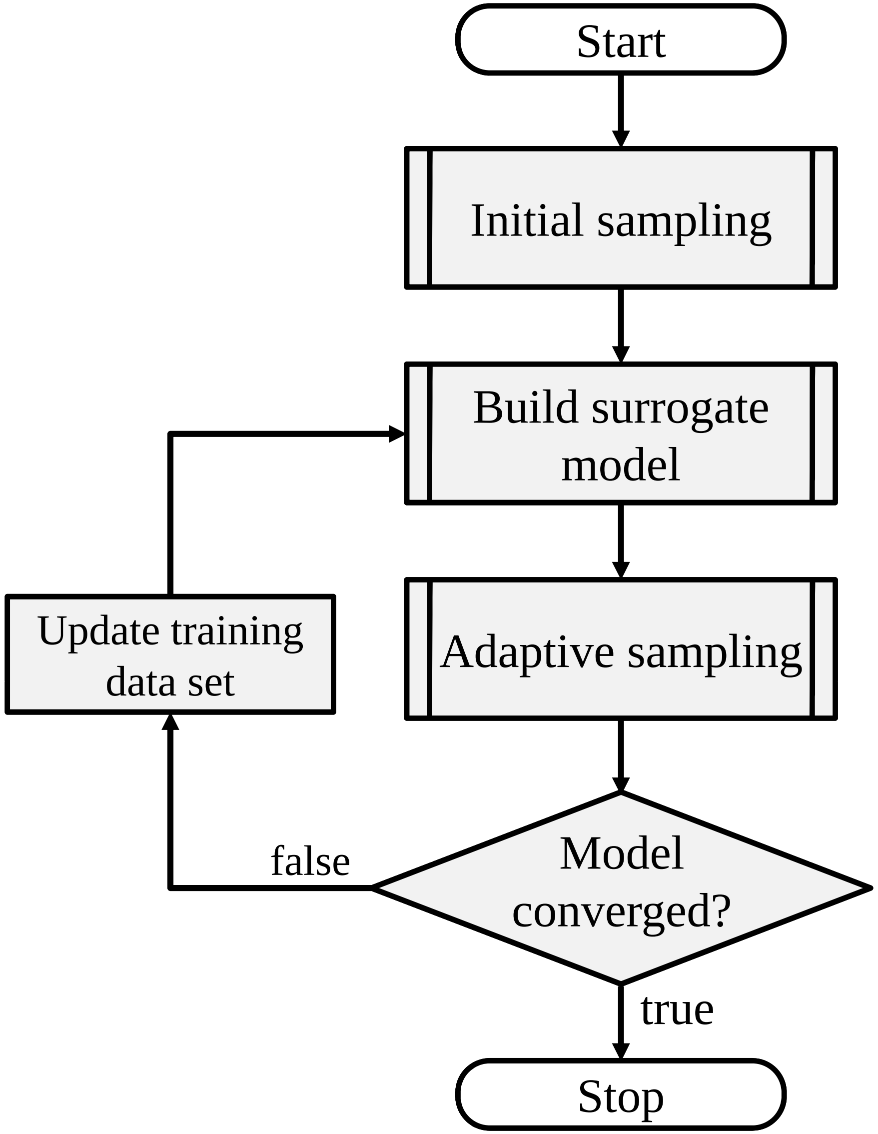
MODELING PROBLEM STATEMENT

- **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?

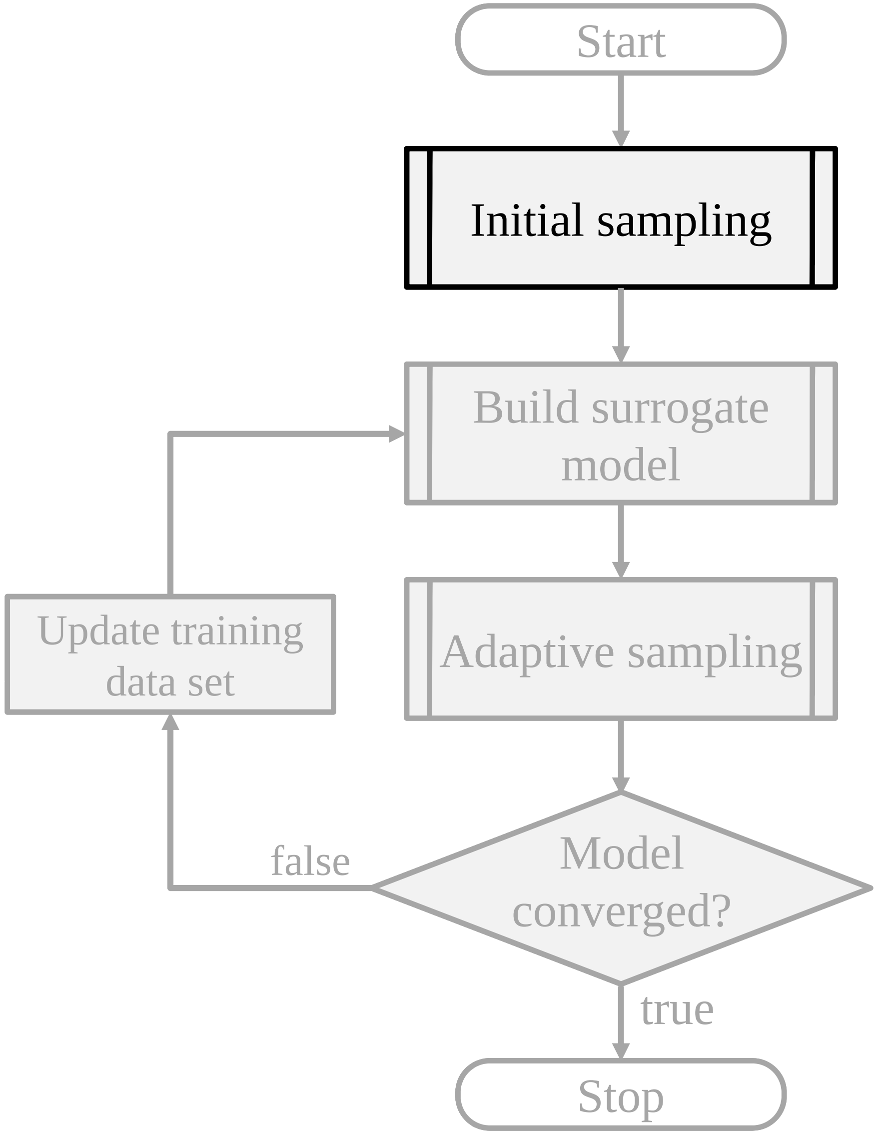
- **Sampling questions:**
  - How many sample points are needed to define an accurate model?
  - Where should these points be sampled?

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

ALGORITHMIC FLOWSHEET



ALGORITHMIC FLOWSHEET

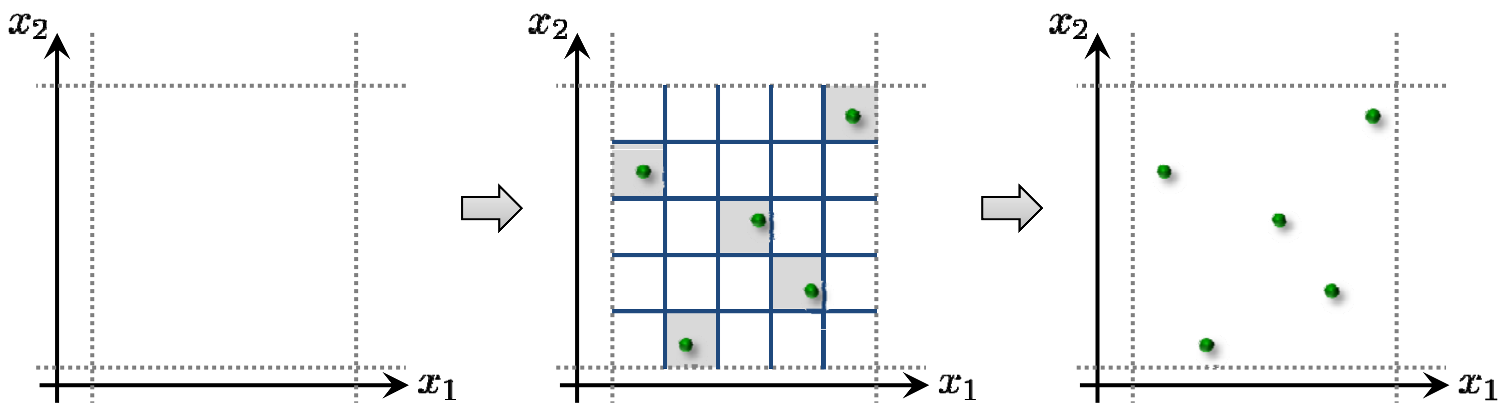


DESIGN OF EXPERIMENTS

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

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

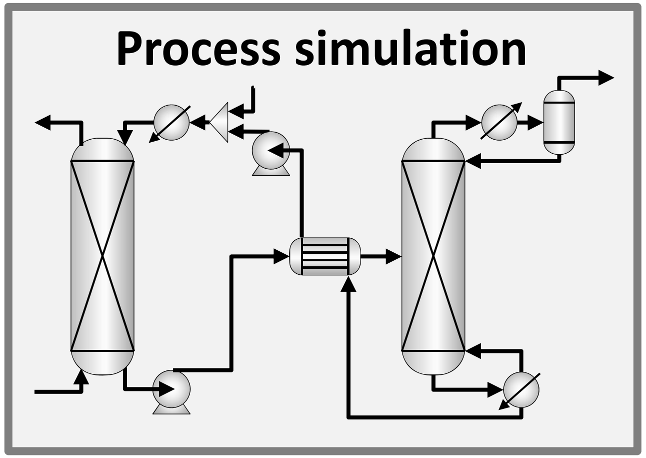
- **Latin hypercube design of experiments** [McKay et al., 79]



INITIAL SAMPLING

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

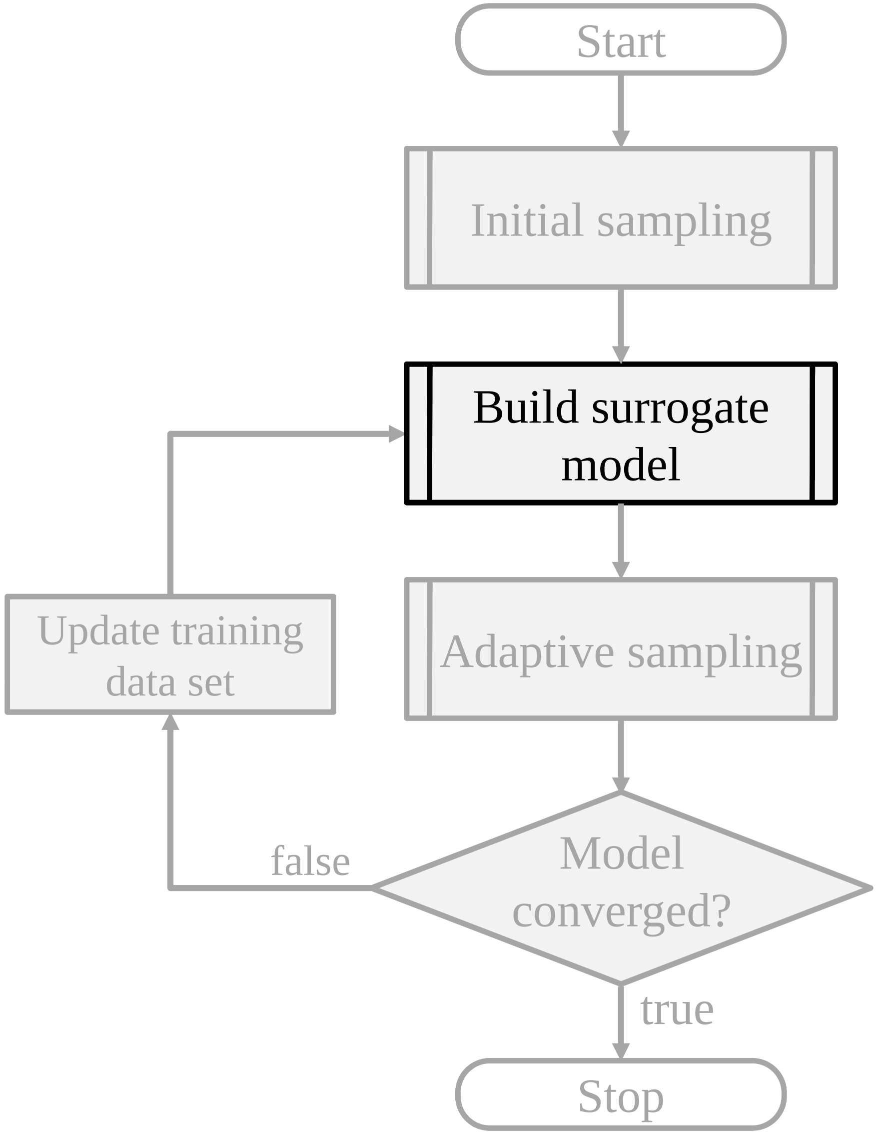
$$x = \begin{pmatrix} x^1 & x^2 & \cdots & x^i & \cdots & x^N \end{pmatrix}$$



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

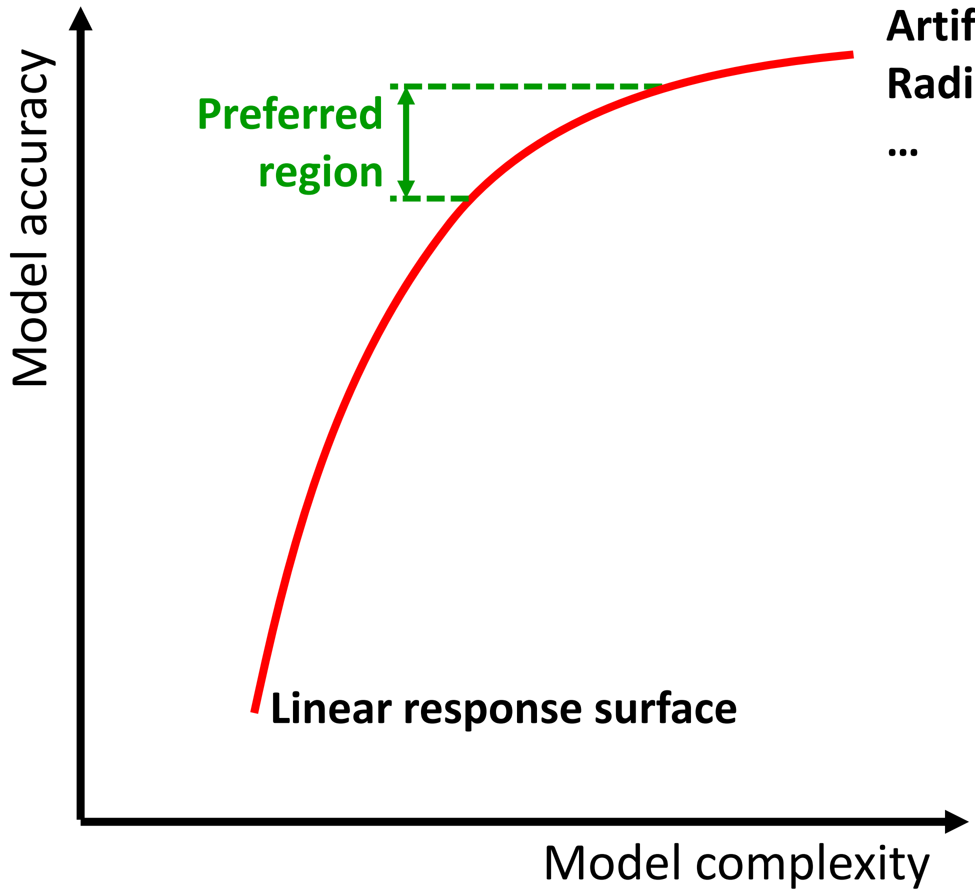
Initial training set

ALGORITHMIC FLOWSHEET



MODEL COMPLEXITY TRADEOFF

Kriging [Krige, 63]
Artificial neural networks [McCulloch-Pitts, 43]
Radial basis functions [Buhman, 00]
...



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.

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**

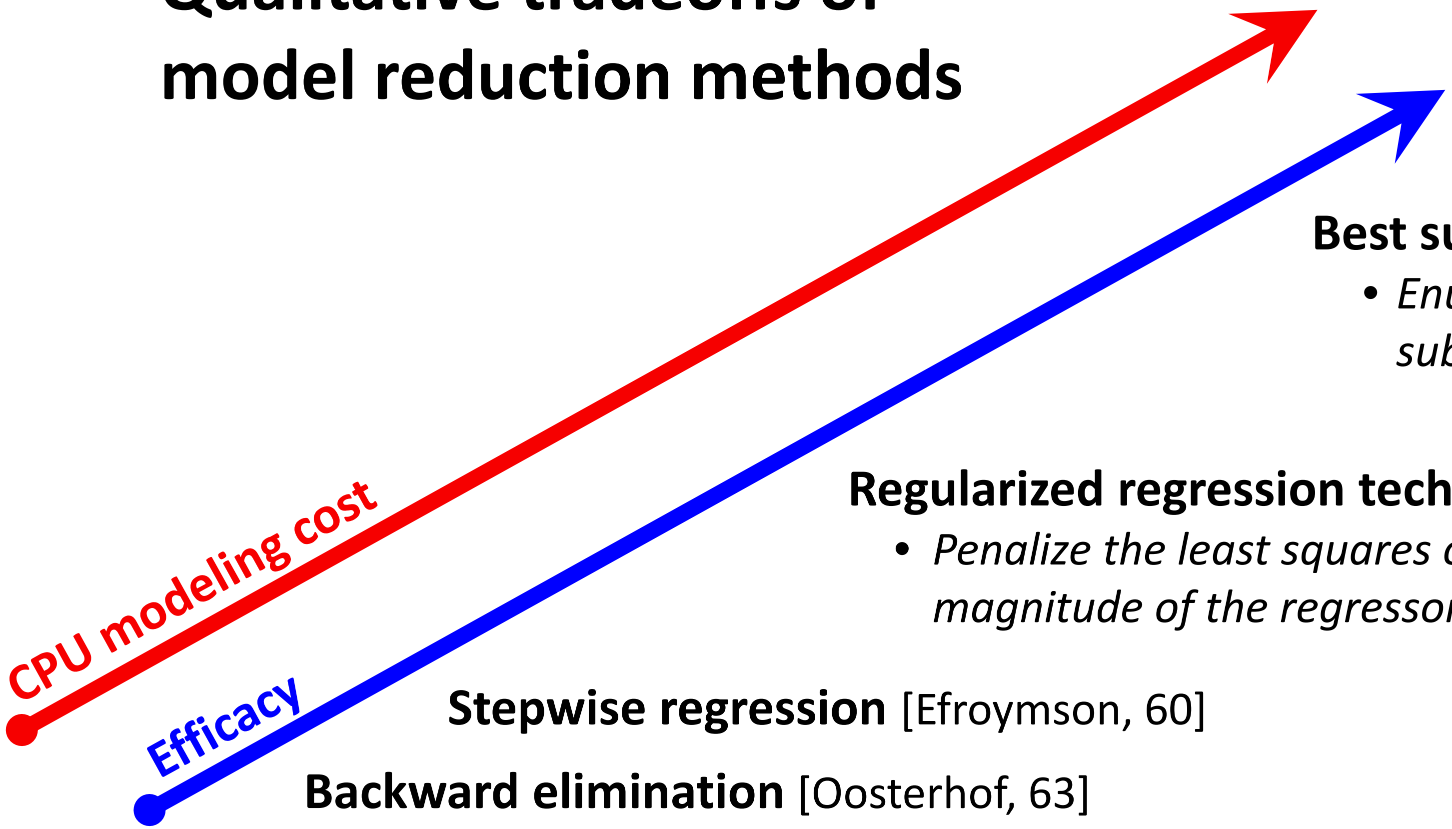
$$\hat{z} = \sum_{j \in \mathcal{S}} \beta_j X_j(x)$$

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

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

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

- **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.

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:**

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

NESTED MIXED-INTEGER PROBLEM

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

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

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

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

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

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

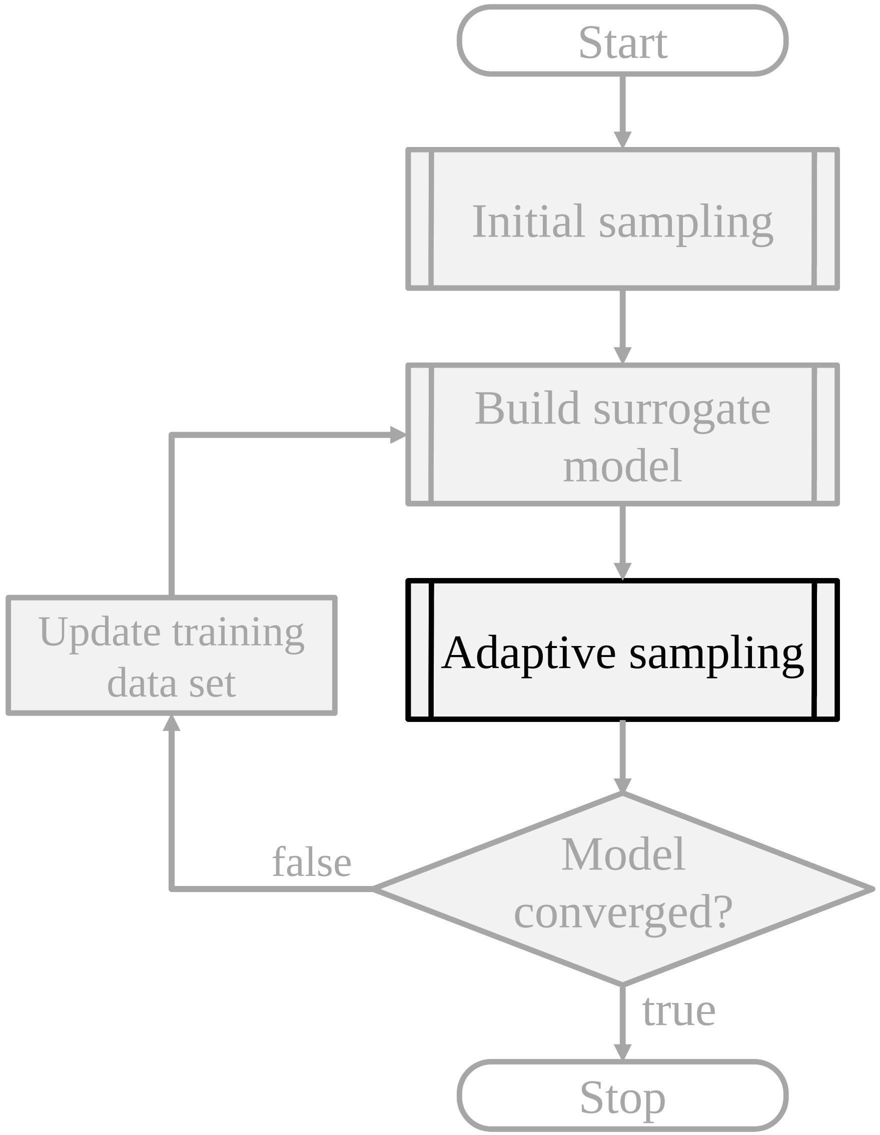
a) **Model sizing**
b) **Basis and coefficient selection**

FINAL BEST SUBSET MODEL

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

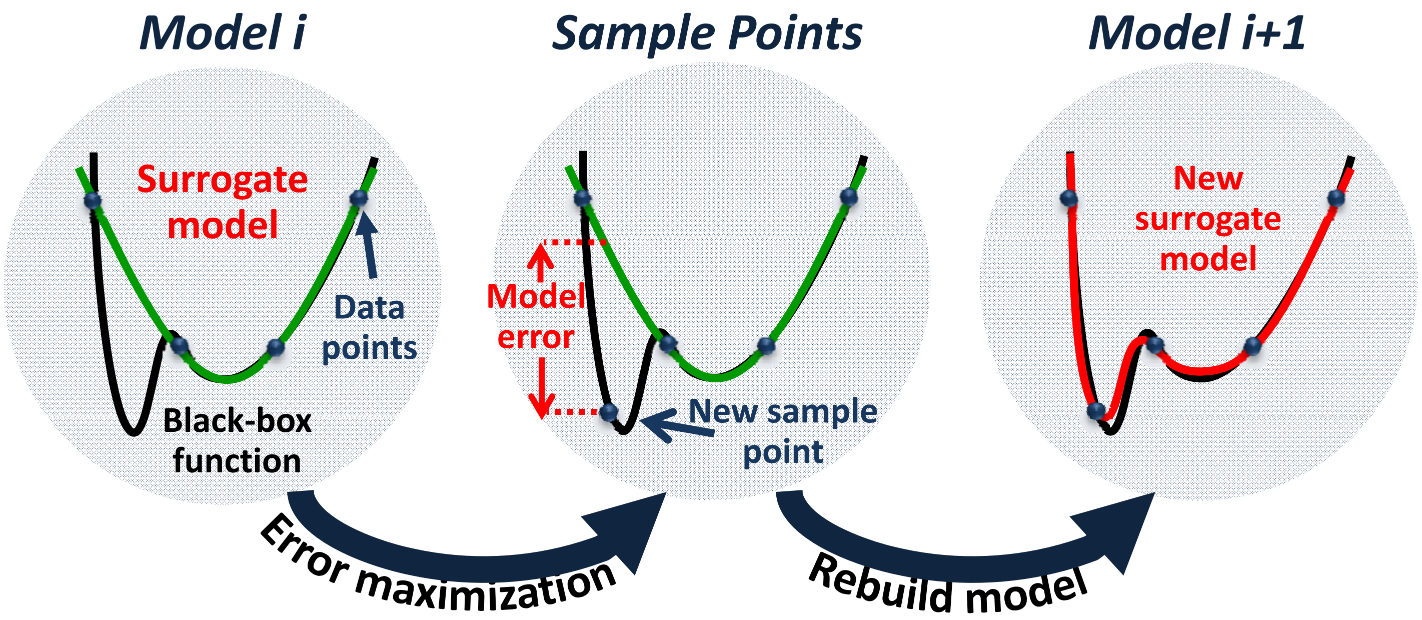
- **This model is solved for increasing values of *T* until the *AICc* worsens**

ALGORITHMIC FLOWSHEET



ADAPTIVE SAMPLING

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



ERROR MAXIMIZATION SAMPLING

- **Goal: Search the problem space for areas of model inconsistency or model mismatch**
- **More succinctly, we are trying to find a point 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]

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**

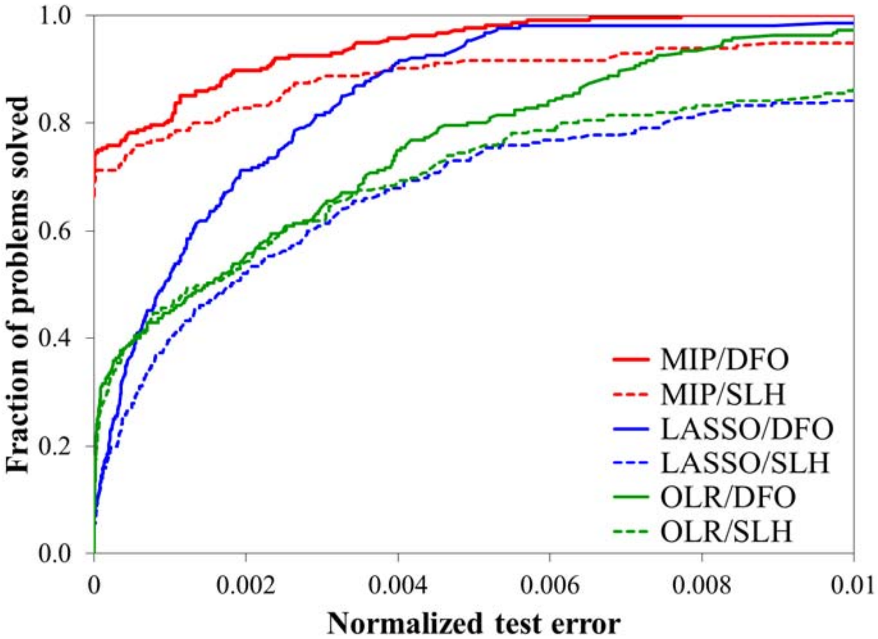
- **Basis functions allowed:**

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

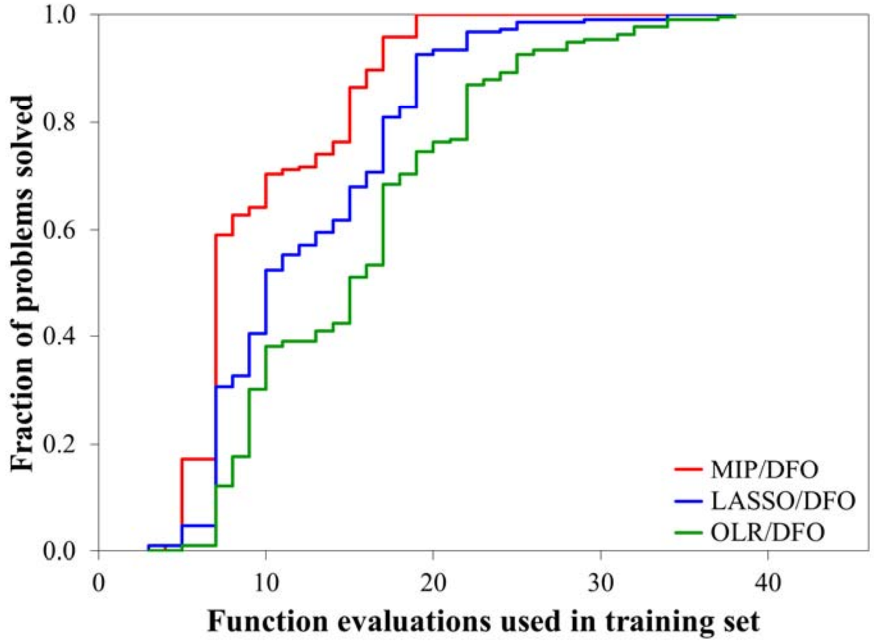
True basis function coefficients were randomly chosen from a uniform distribution where $\beta \in [-1, 1]$.

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

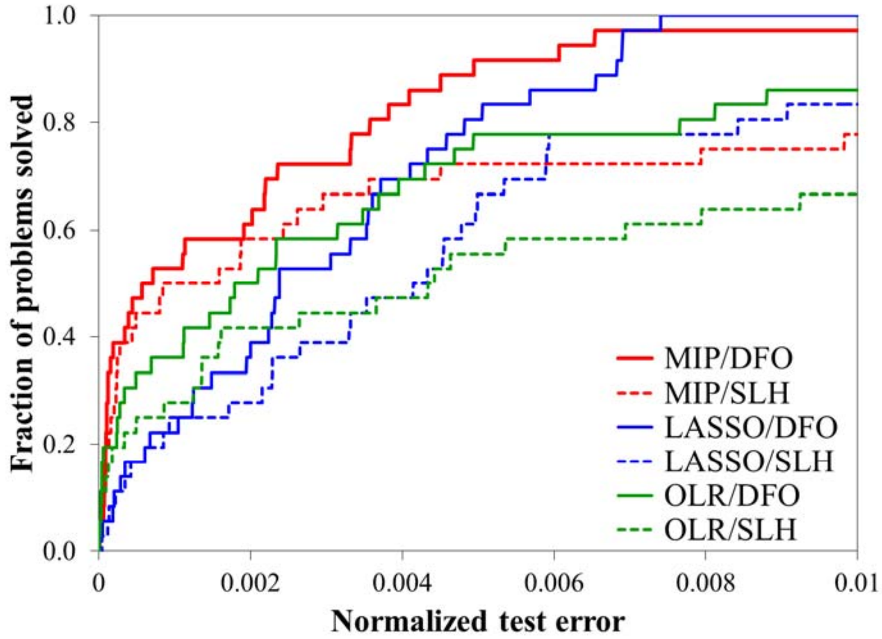
- **Two-input black-box functions with basis functions unavailable to the algorithms with**

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

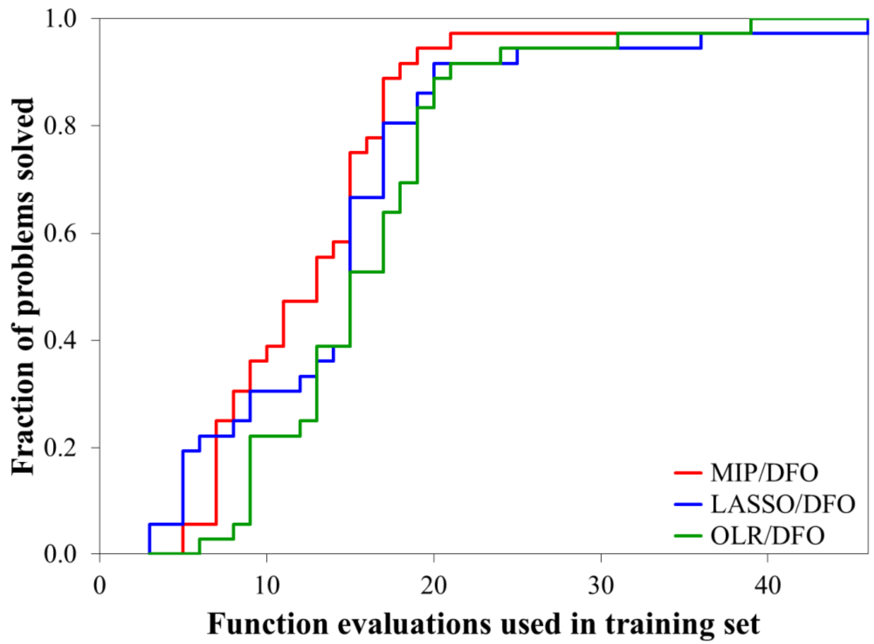
with true parameters chosen from a uniform distribution where $\beta \in [-1, 1]$, $\alpha, \nu \in [-3, 3]$, $\gamma \in [-5, 5]$, and $i, j \in \{1, 2\}$.

RESULTS – TEST SET B

Model accuracy



Function evaluations

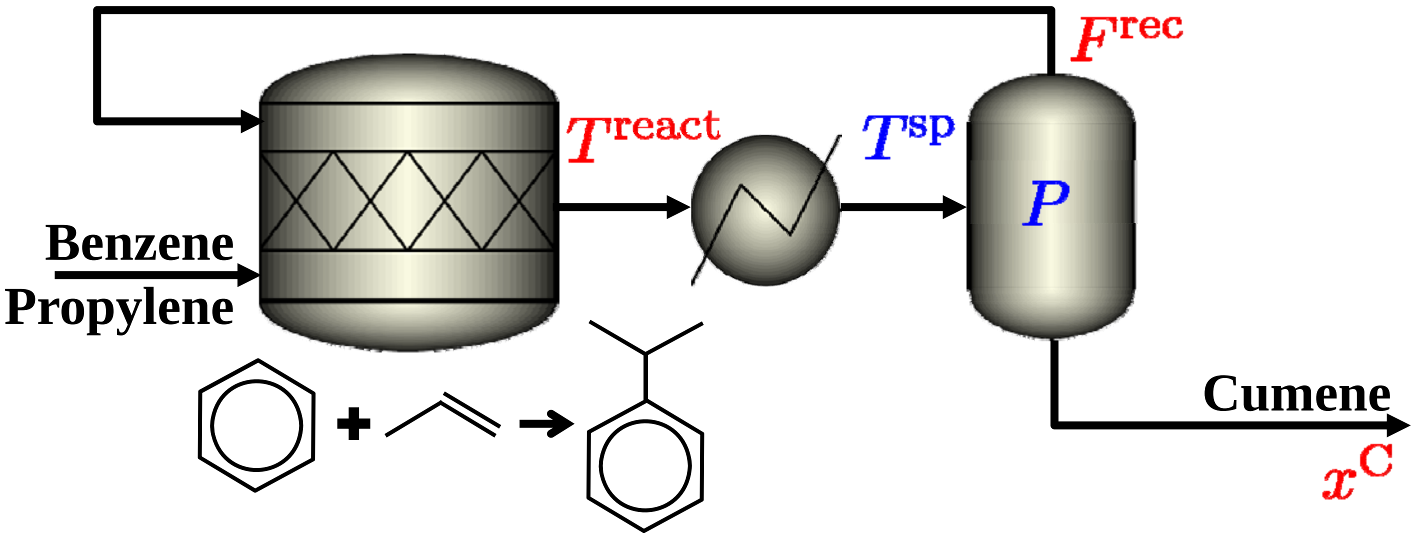


12 test problems, repeated 5 times, tested against 1000 independent data points

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]

TEST CASE: CUMENE PRODUCTION



Generate Models:

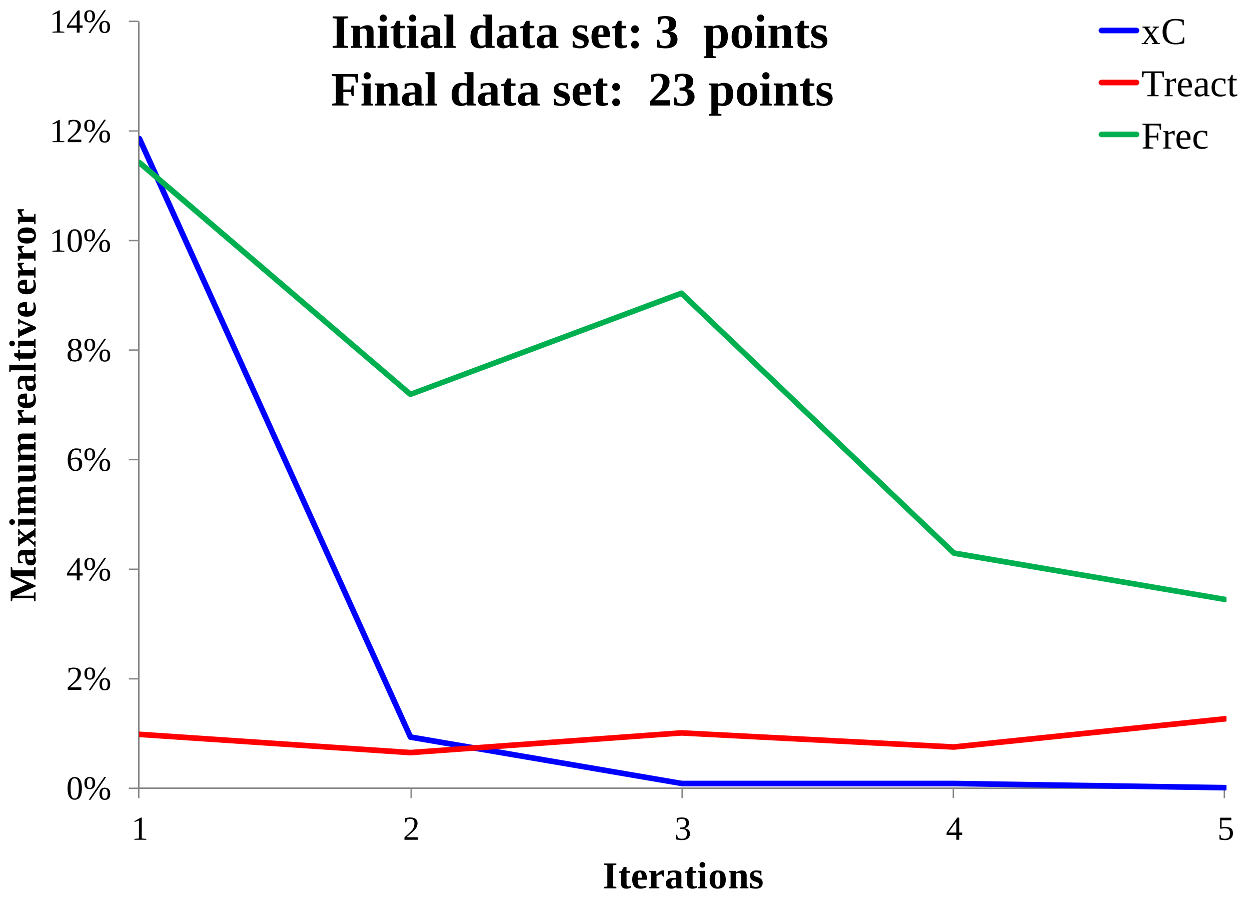
$$x^{\text{C}}(T^{\text{sp}}, P) = f_1(T^{\text{sp}}, P)$$
$$T^{\text{react}}(T^{\text{sp}}, P) = f_2(T^{\text{sp}}, P)$$
$$F^{\text{rec}}(T^{\text{sp}}, P) = f_3(T^{\text{sp}}, P)$$

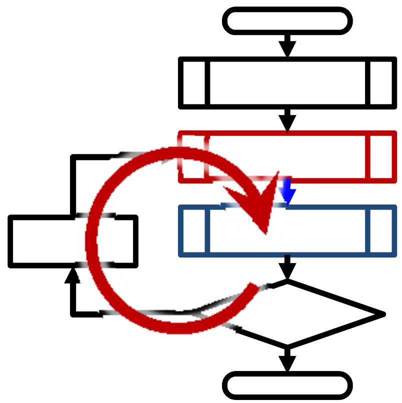
Over the Range:

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

Cumene production simulation is form the Aspen Plus® Library

GENERATING THE SURROGATES



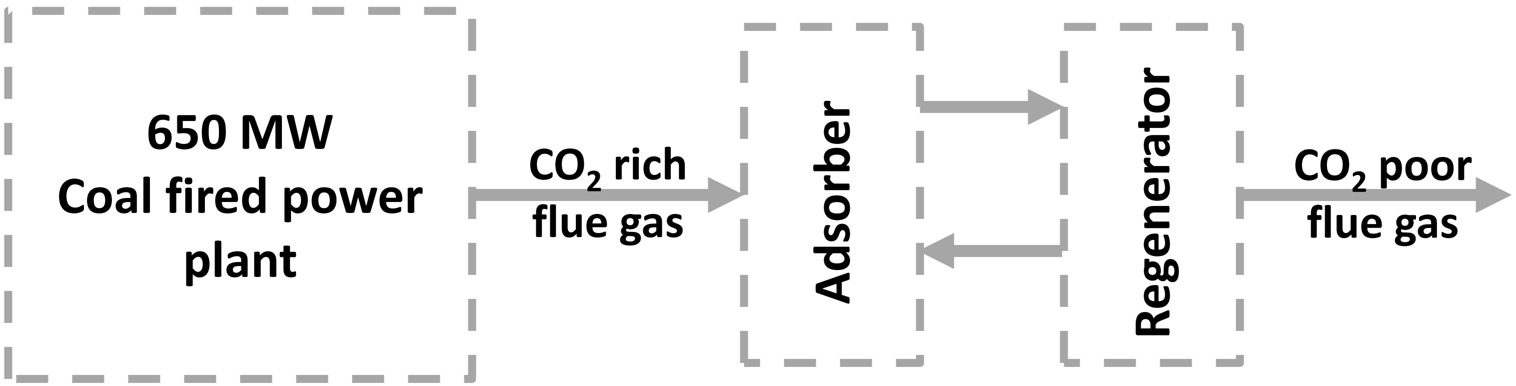


- **Maximum error found at each iteration may increase**
  - **Due to the derivative-free solver is given more information at each iteration**

CARBON CAPTURE OPTIMIZATION

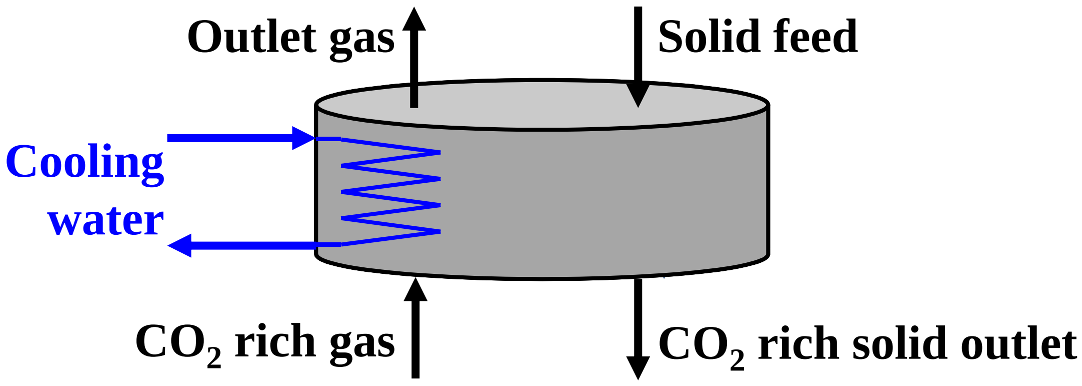
- **Problem statement:**

  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



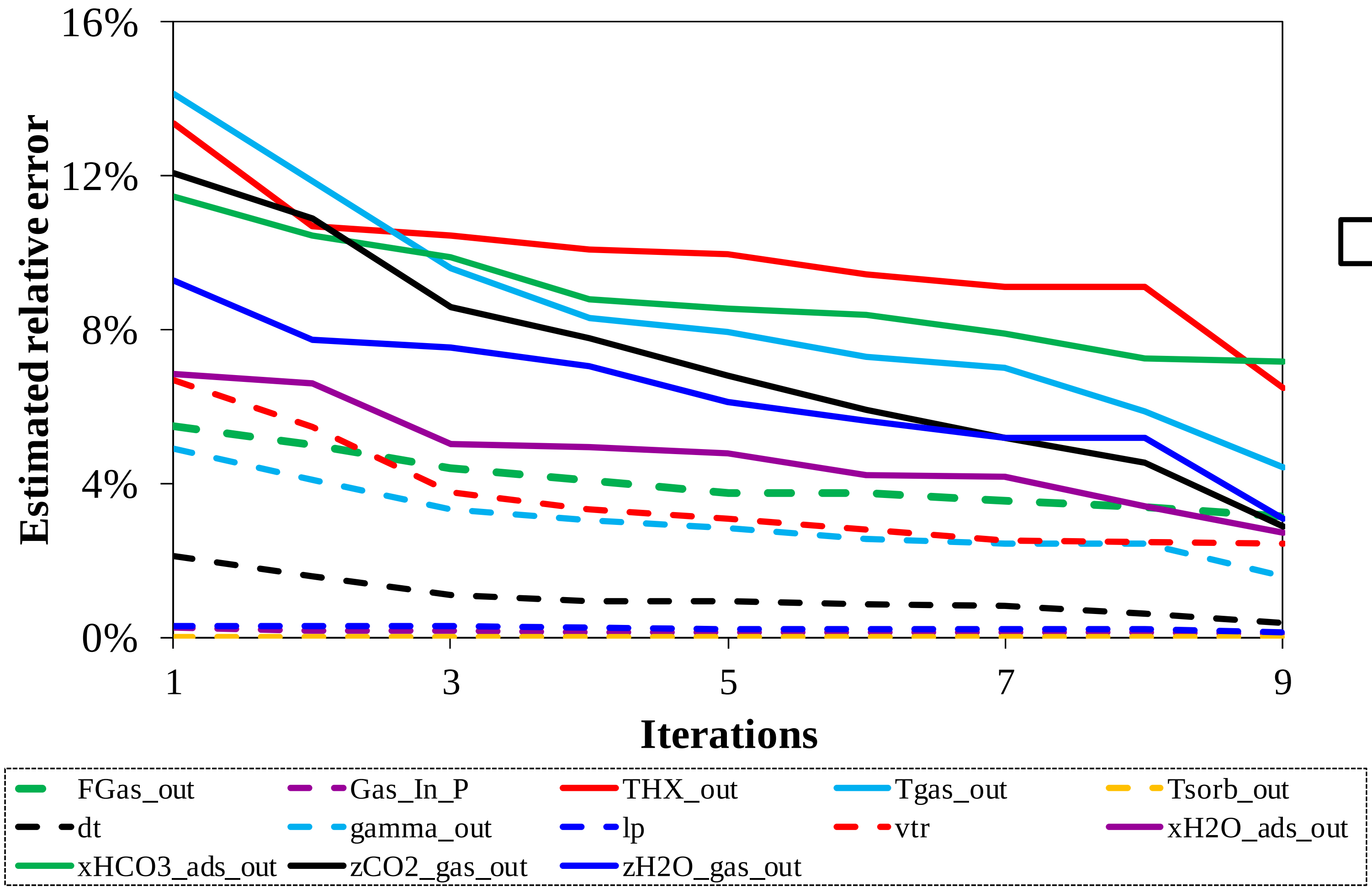
- **Model inputs (14 total)**
  - **Geometry (3)**
  - **Operating conditions (4)**
  - **Gas mole fractions (2)**
  - **Solid compositions (2)**
  - **Flow rates (4)**

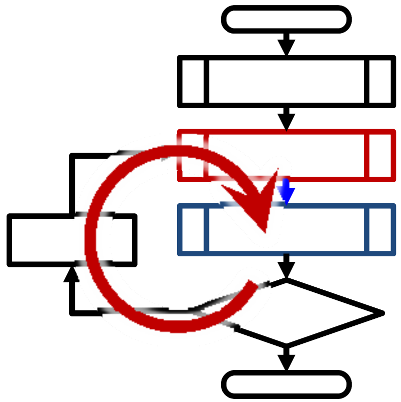
- **Model outputs (13 total)**
  - **Geometry required (2)**
  - **Operating condition required (1)**
  - **Gas mole fractions (2)**
  - **Solid compositions (2)**
  - **Flow rates (2)**
  - **Outlet temperatures (3)**
  - **Design constraint (1)**

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

ADAPTIVE SAMPLING

Progression of mean error through the algorithm

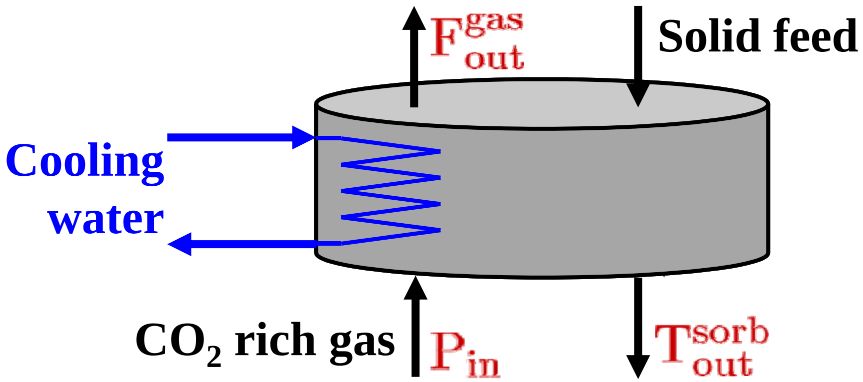




Initial data set: 137 pts

Final data set: 261

EXAMPLE MODELS



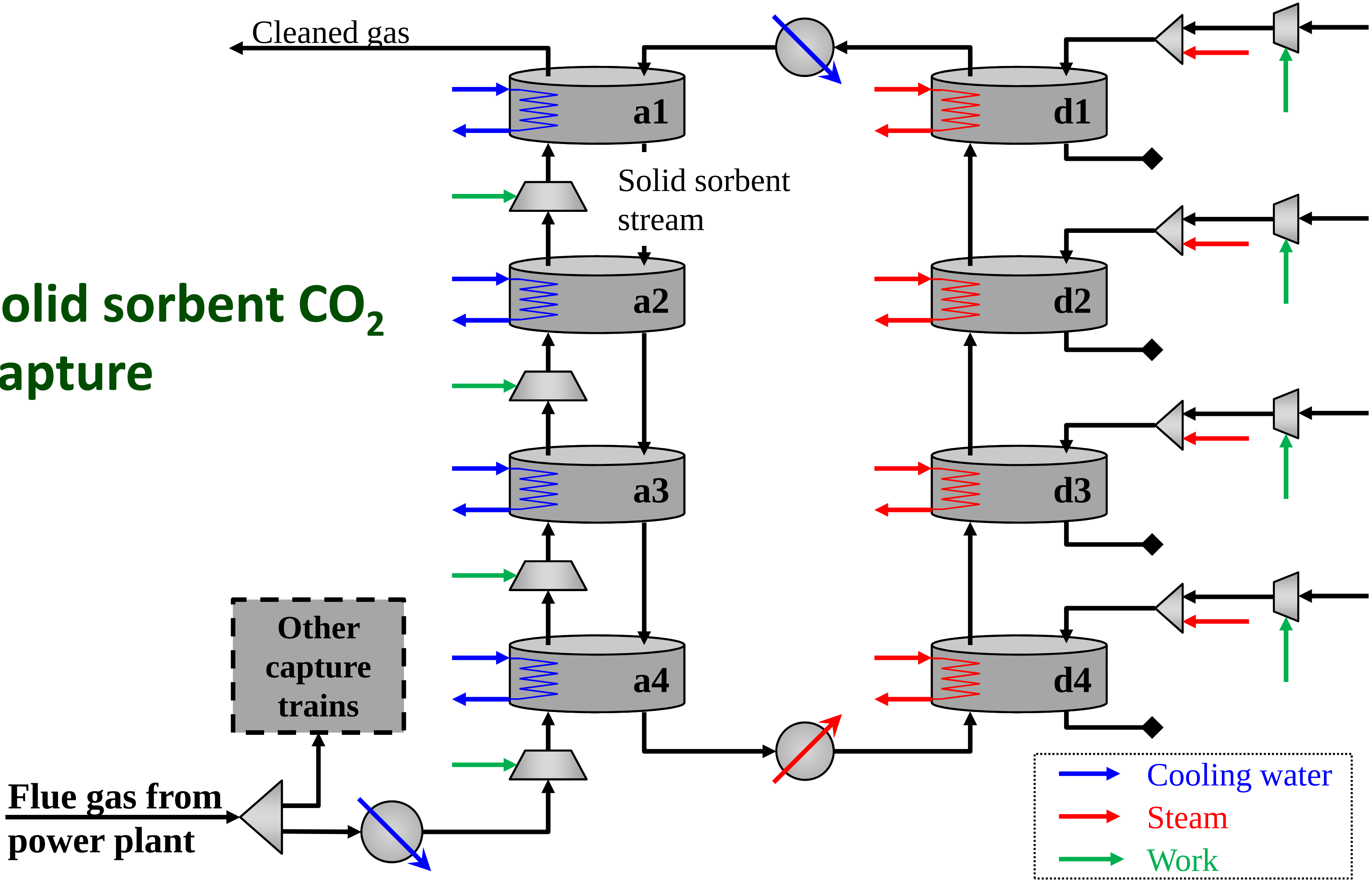
$$P_{in} = 1.0\,P_{out} + 0.0231\,L_b - 0.0187\,\ln(0.167\,L_b) - 0.00626\,\ln(0.667\,v_{gi}) - \frac{51.1\,xHCO3^{ads}_{in}}{F^{gas}_{in}}$$

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

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

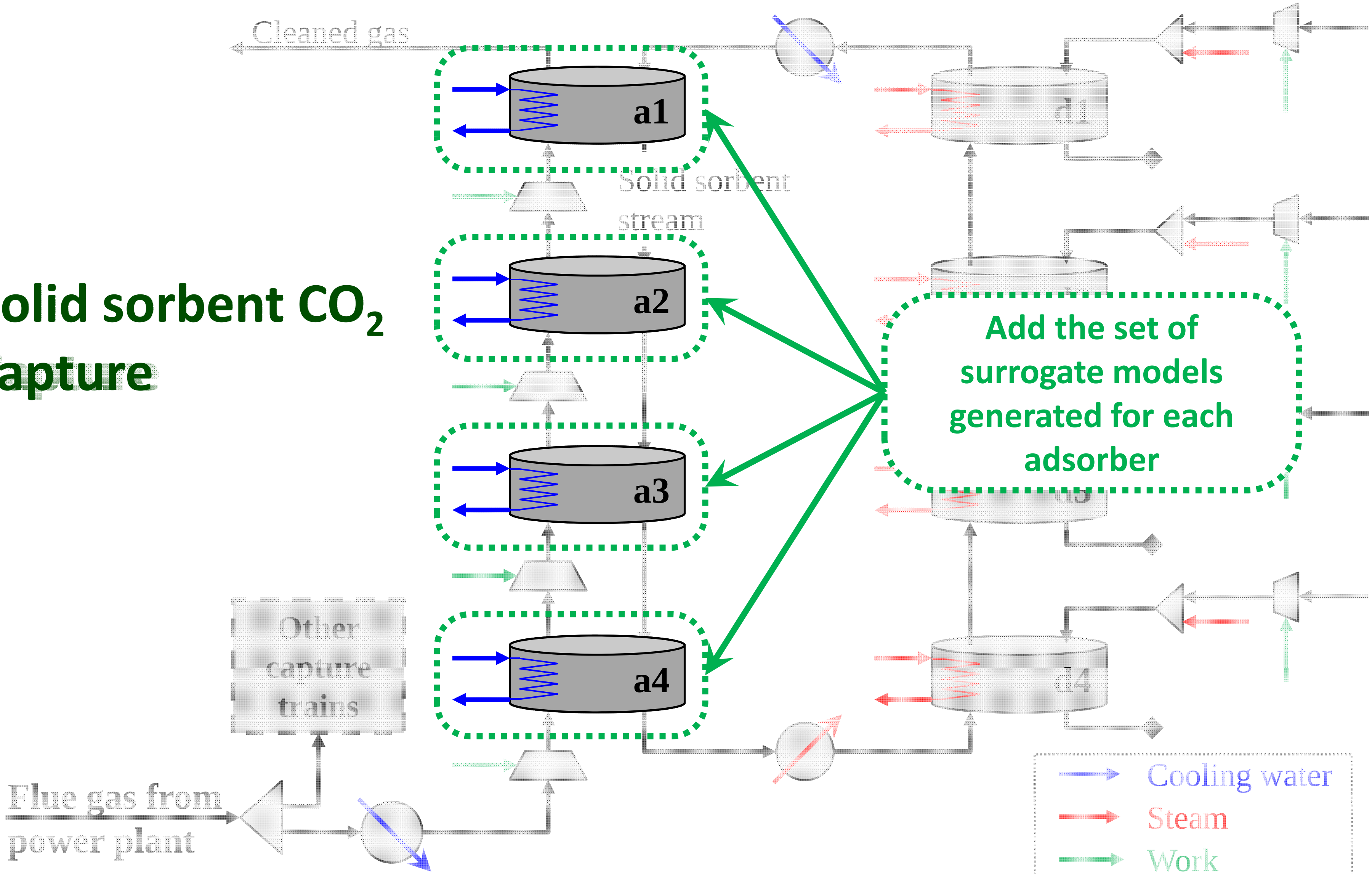
SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 capture



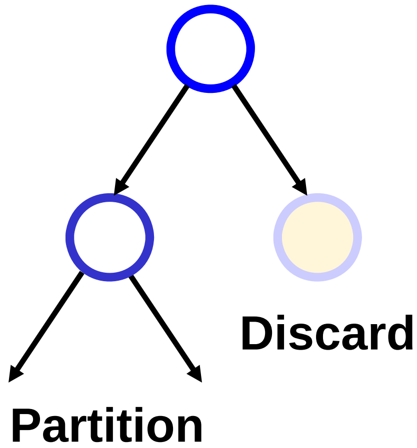
SUPERSTRUCTURE OPTIMIZATION

Solid sorbent CO_2 capture

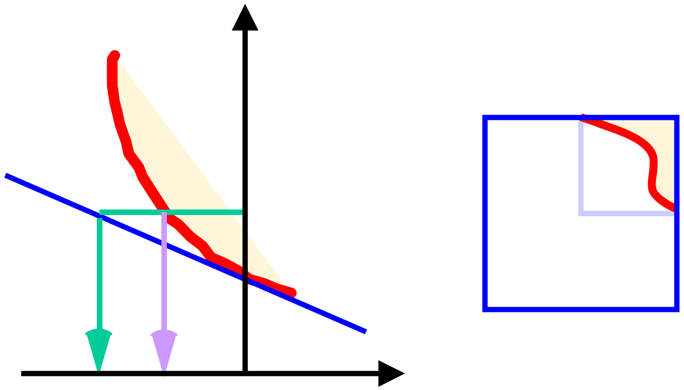


BRANCH-AND-REDUCE

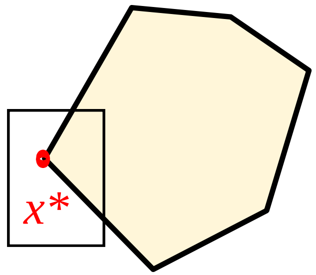
Search Tree



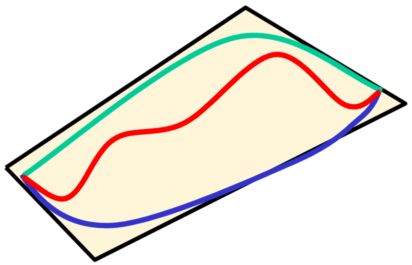
Range Reduction



Finiteness



Convexification



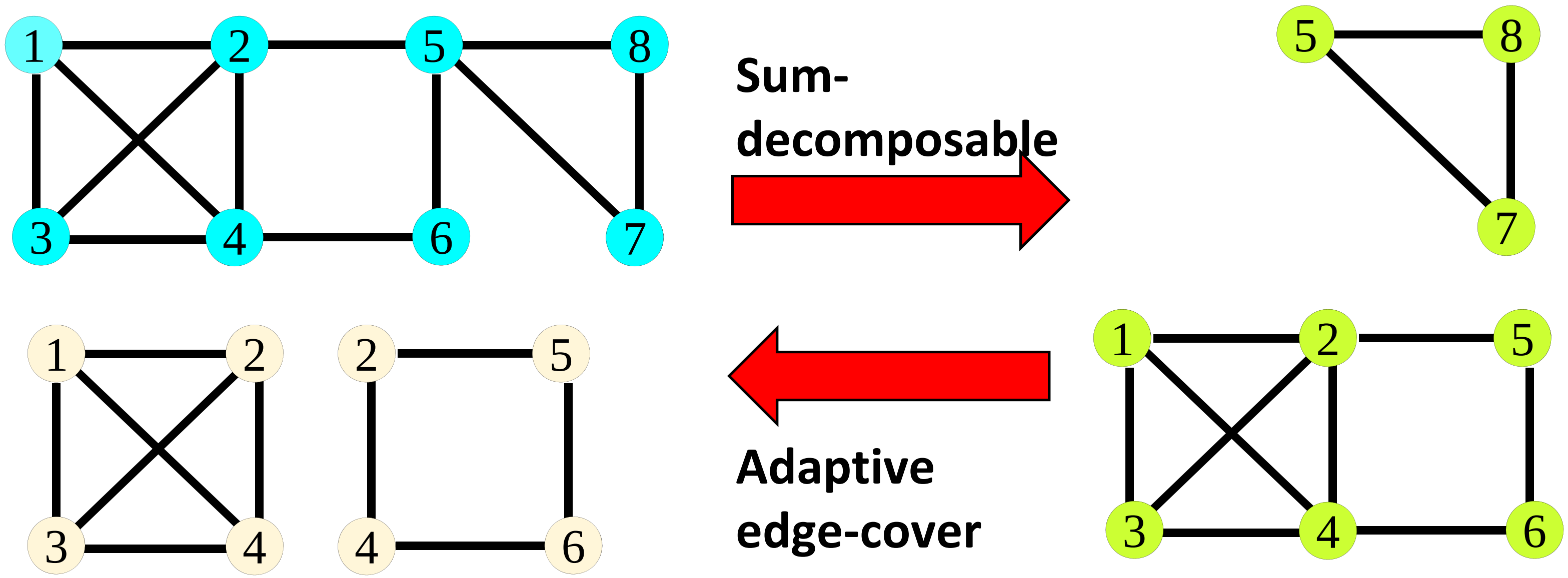
BARON HISTORY

1991-93	Duality-based range reduction
	Nonlinear constraint propagation
1994-95	Branch-and-bound system
	Finite algorithm for separable concave minimization
1996-97	Parser for factorable programs; nonlinear relaxations
	Links to MINOS and OSL
1997-98	Polyhedral relaxations; Link to CPLEX
	Compressed data storage, tree traversal, …
2002	Under GAMS
2004	Branch-and-cut
2005-07	Local search; memory management, …
2009	Multi-term envelopes
2010-11	Multi-variate and multi-constraint envelopes/relaxations
	Links to CLP and IPOPT

CONVEXIFICATION OF MULTILINEARS

Decompose **multilinear functions into** **low-dimensional dense components** **that are convexified individually**

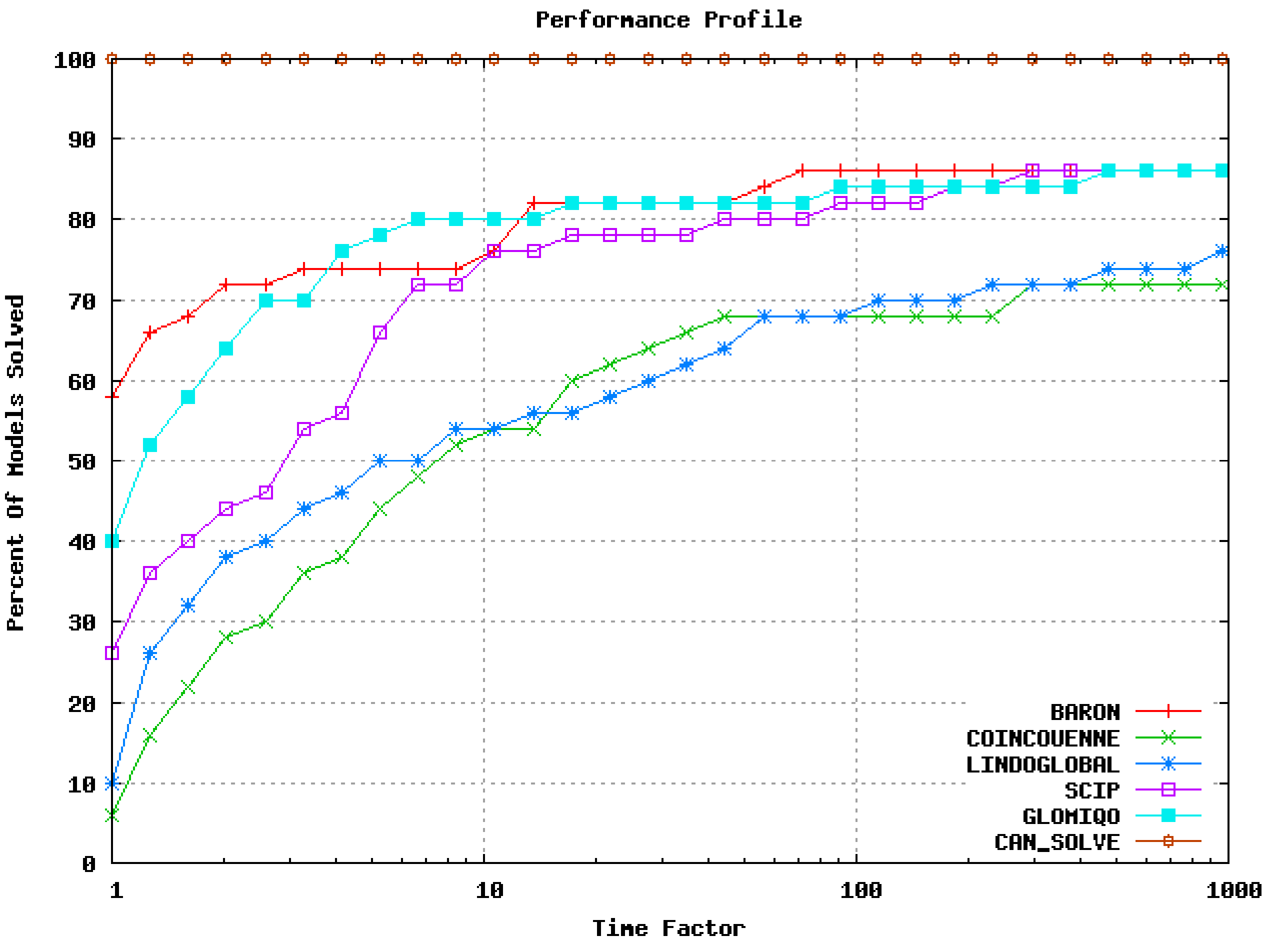
$$L(x) = x_1x_2 + x_1x_3x_4 + x_2x_3 + x_2x_4 + x_2x_5 + x_4x_6 + x_5x_6 + x_5x_7x_8$$



$$L_1(x) = x_1x_2 + x_1x_3x_4 + x_2x_3 + x_2x_4$$

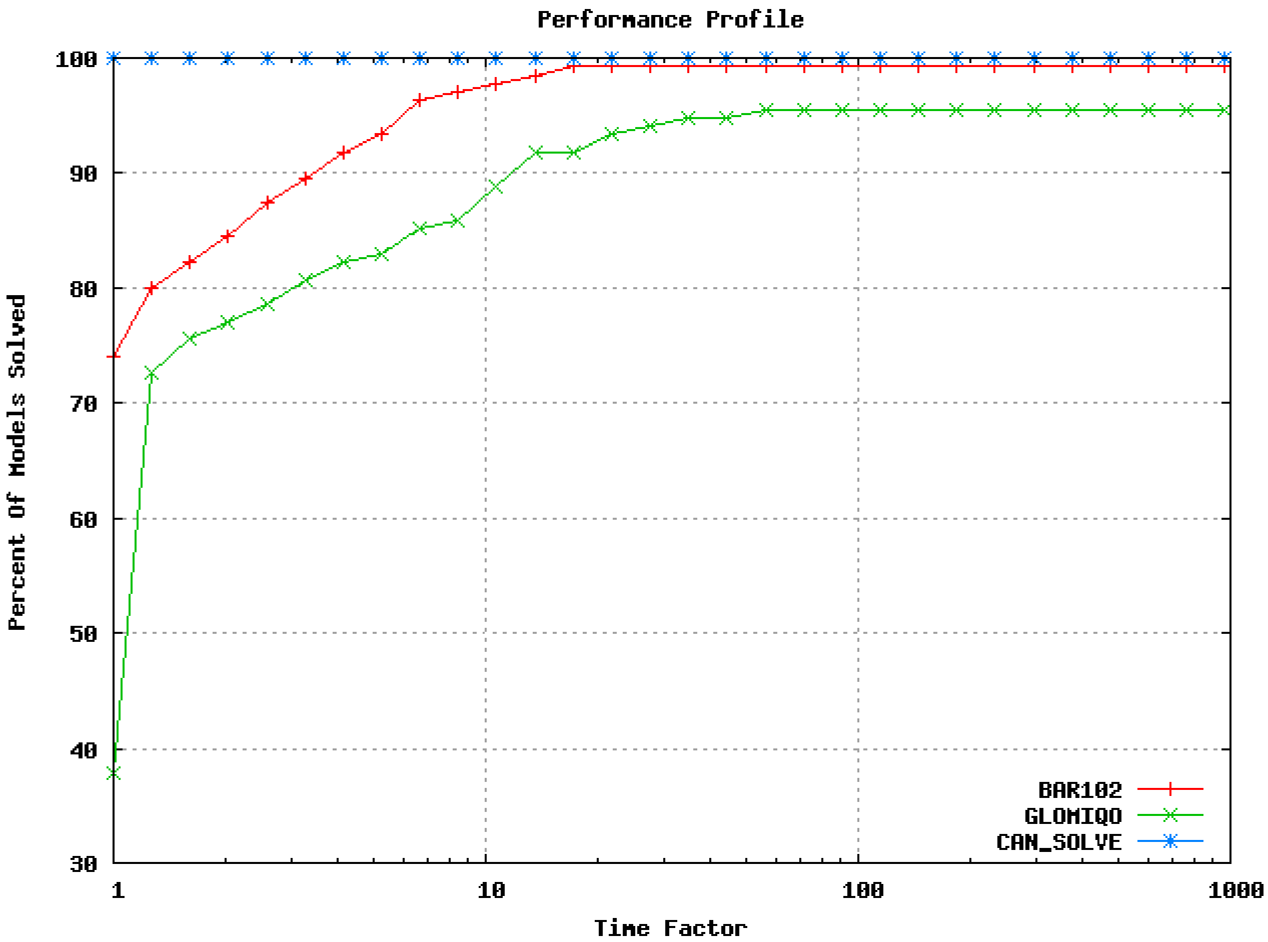
$$L_2(x) = x_2x_4 + x_2x_5 + x_4x_6 + x_5x_6, \; L_3(x) = x_5x_7x_8$$

COMPARISONS WITH OTHER CODES



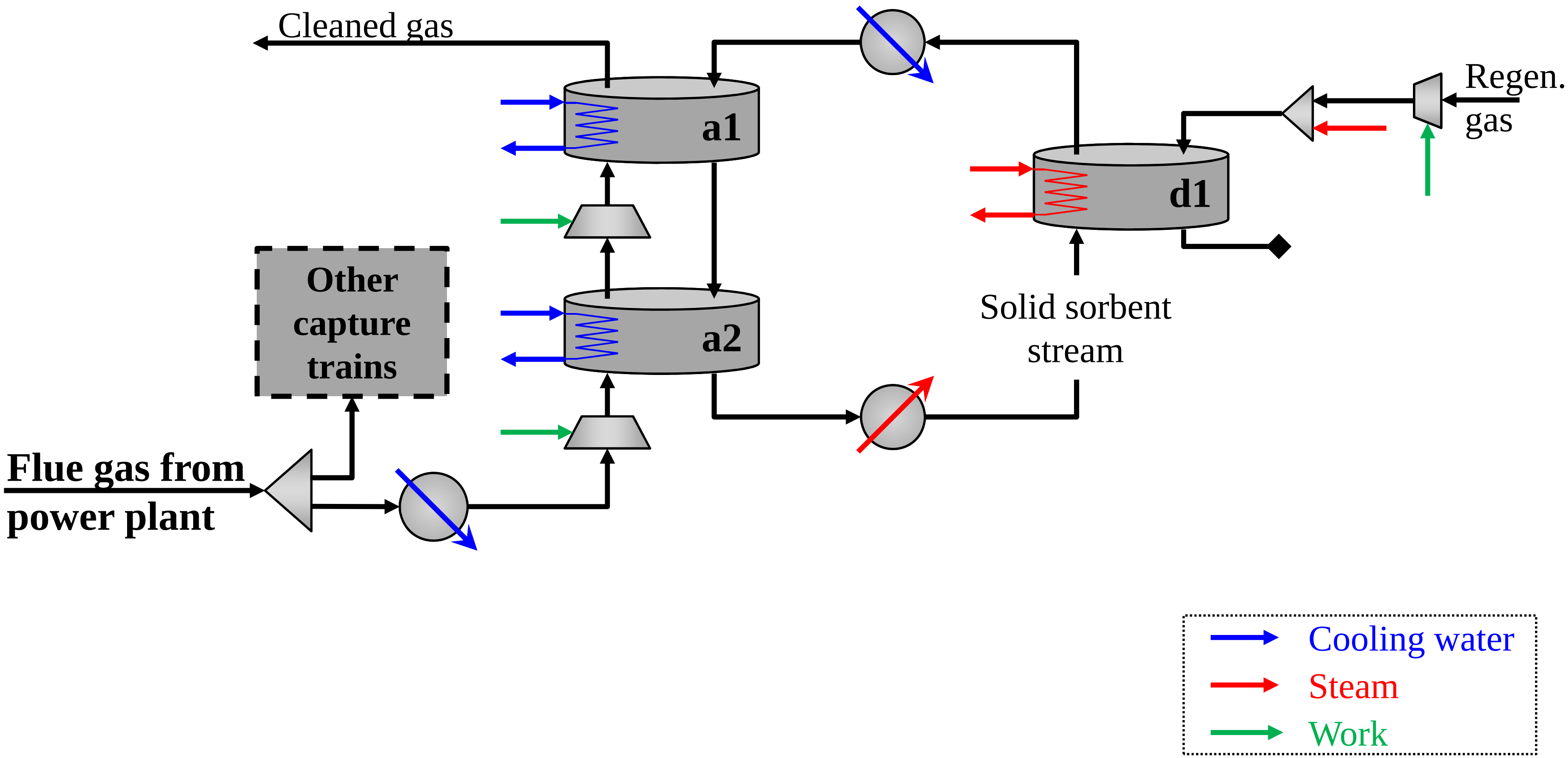
Test problems from
http://helios.princeton.edu/GloMIQO/test_suite.html#Computational%20Geometry

COMPARISONS WITH GloMIQO ON 136 QCQPs



Test problems from Bao et al. (2009)

PRELIMINARY RESULTS ON MINLP



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 Learning of Algebraic Models for Optimization