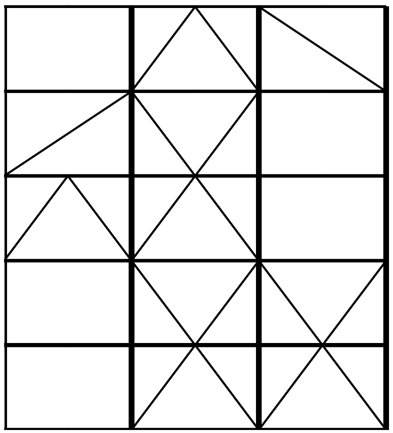
Machine Learning for Selection of Approximate Optimal Placement of Braces of Plane Frames under Static Loads

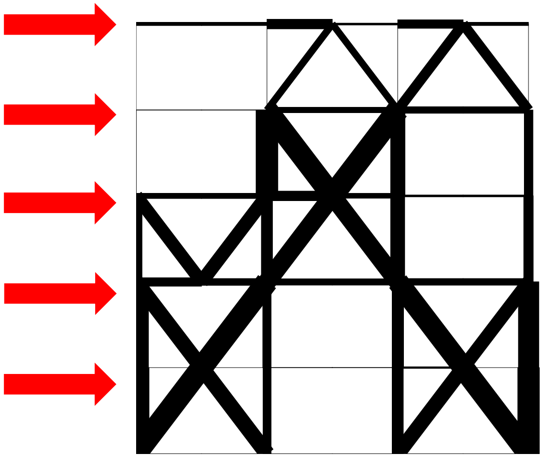
Makoto Ohsaki (Kyoto Univ.)
Toshiaki Kimura (Kyoto Univ.)
Kazuma Sakaguchi (Kyoto Univ.)

Optimal placement of braces

- Consider a process of increasing stiffness of building frame by adding braces of various patterns.
- Fix stiffnesses of beams and columns.
- Load path strongly depends on the brace locations.



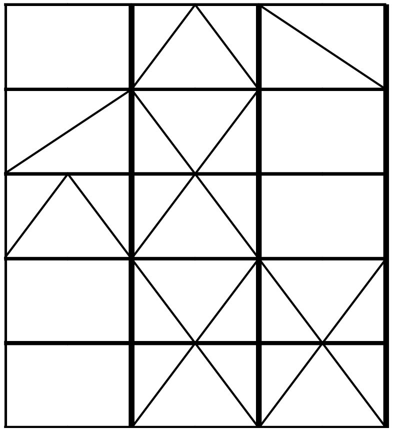
Brace locations



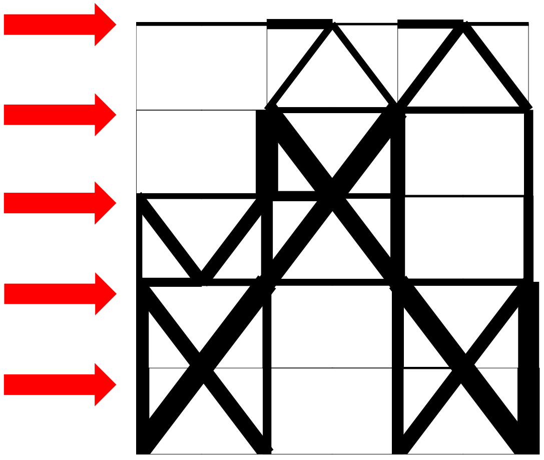
Axial forces

Purpose of this study

- Find optimal locations of braces of plane frames.
- Minimize increase of stress in existing beams and columns.
- Constraint on interstory drift angle.
- Use machine learning to reduce computational cost.



Brace locations



Axial forces

Application of machine learning to structural optimization

- Neural network for prediction (approximation) of structural responses (1990s -)
- Optimal member grouping
- Optimal parameters for heuristics
- Shape optimization of periodic structures
- Identification of features of feasible solutions
- Optimal search region/direction in heuristic approach

Difficulty in combinatorial problems

- Combinatorial problem:
  - Difficult to use mathematical programming.
  - Use heuristics.
  - Large computational cost for problem with many variables and complex structural response.

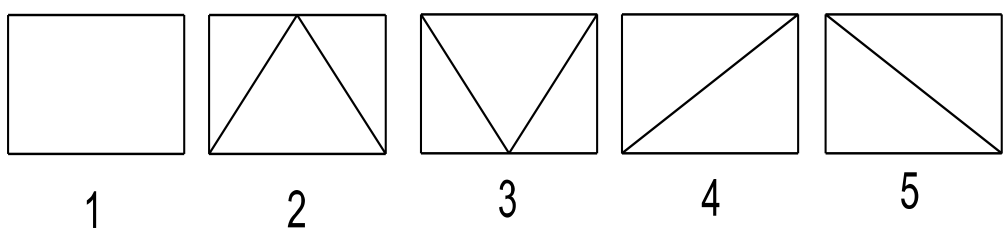
Use machine learning for reduction of computational cost.

Design variables

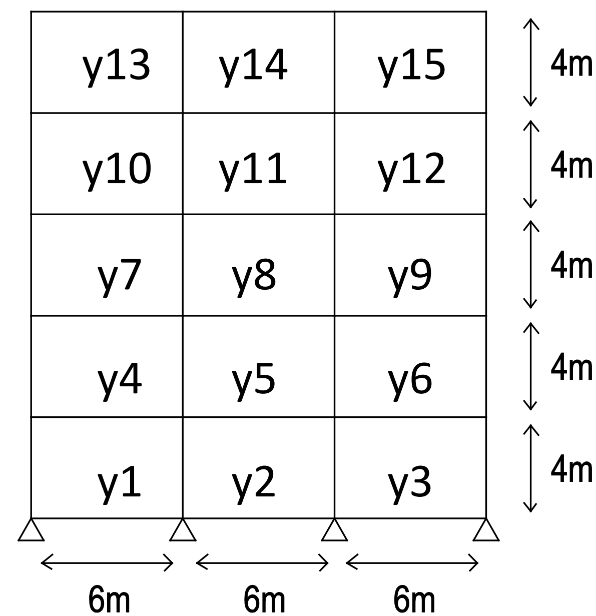
- Five patterns denoted by y = 1, 2, 3, 4, 5

$$\boldsymbol{y} = (y_1, y_2, y_3, \dots, y_m), (y_i \in \{1, 2, 3, 4, 5\})$$

- 5-story 3-span frame:
  Number of locations = 15



Five patters of braces



Optimization problem

- Objective function:

  minimize maximum stress σ_{max} of beams and columns

- Constraints:

  interstory drift angle $\theta_{max} \leq 0.005$

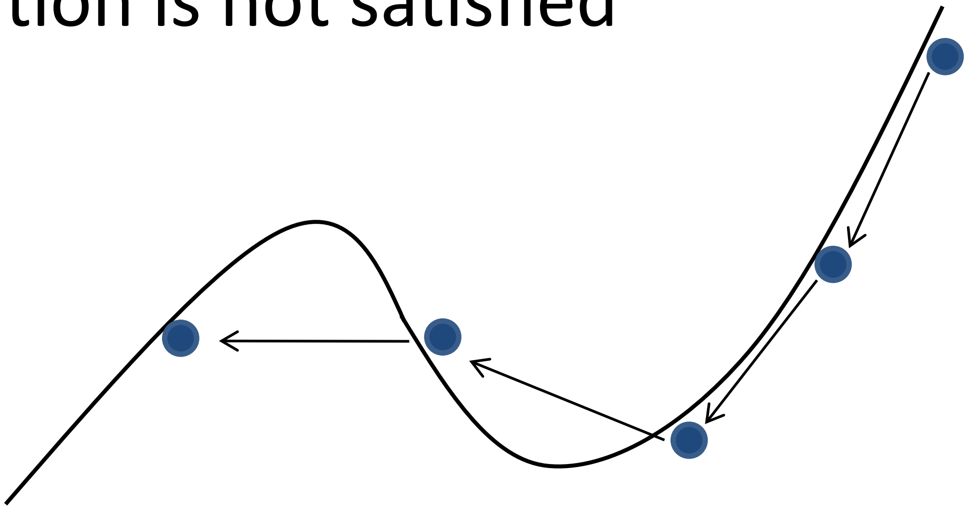
  number of braces in each story ≤ 2

- Use support vector machine (SVM) for classification of *approximate optimal solution* and *non-optimal* solution

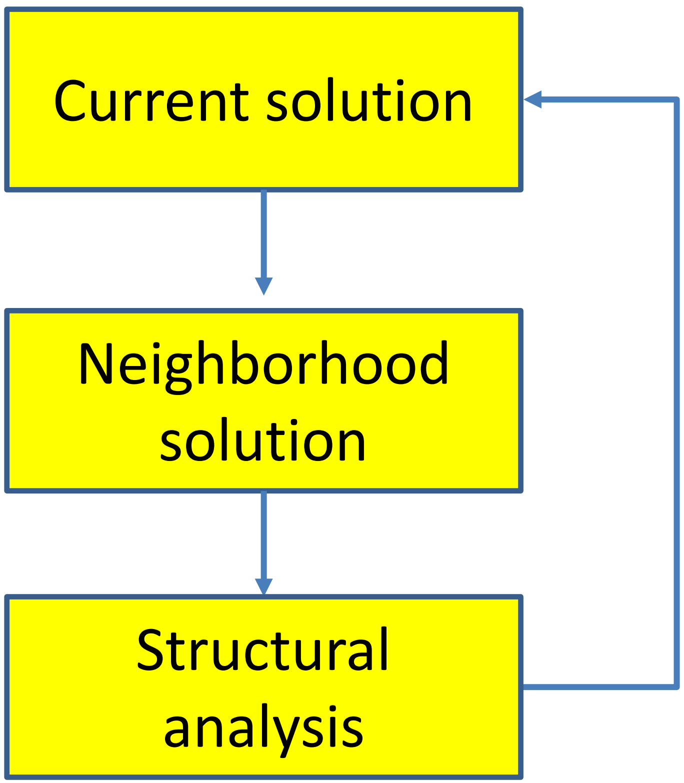
Simulated annealing (SA)

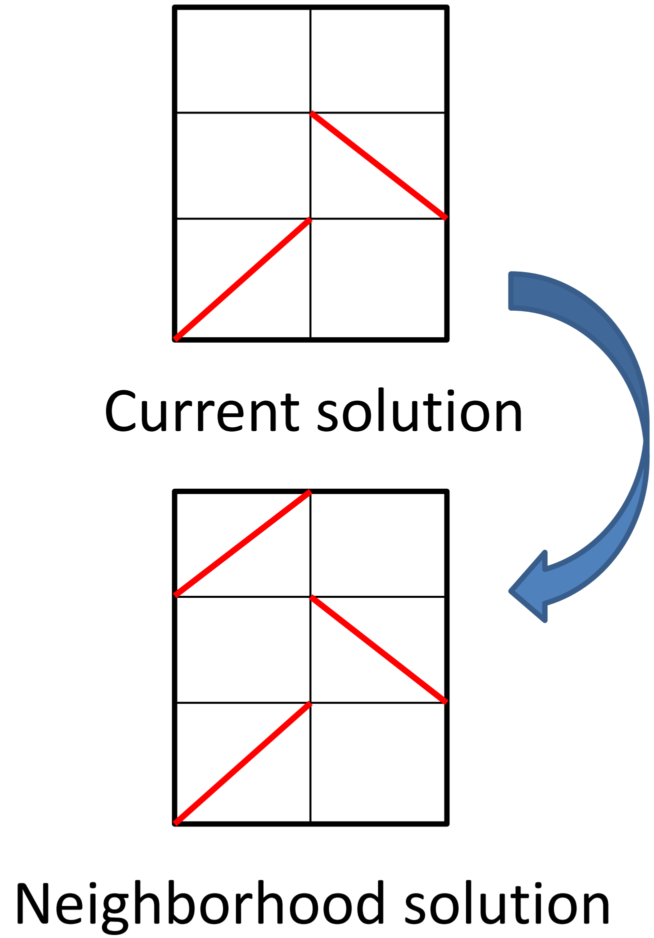
1. Randomly generate initial solution.
2. Randomly generate neighborhood solution and accept it probabilistically even it does not improve the objective value.
3. Reduce the temperature parameter if the termination condition is not satisfied and go to 2.

$$P_r = \exp\left(-\frac{\Delta F}{cT}\right)$$



Outline of SA



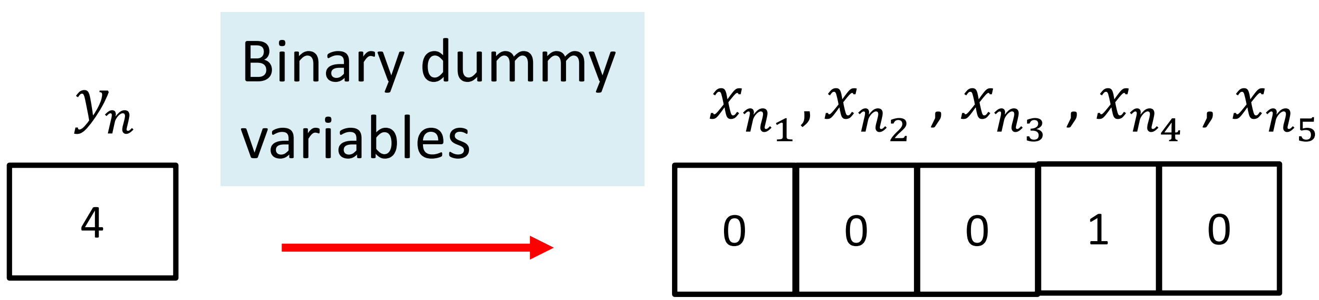


Large computational time for structural analysis.
→ Do not carry out analysis for non-optimal solution.
→ Reduce computational cost.

Machine learning using SVM

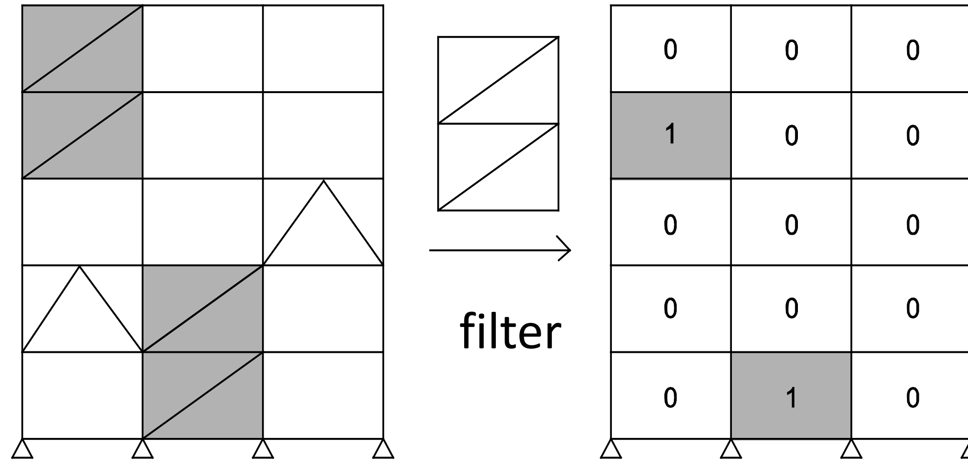
- SVM handles only ordered variables

- Integer variable $y_i \in \{1,2,3,4,5\}$ is converted to five dummy binary variables $x_{i1}, x_{i2}, x_{i3}, x_{i4}, x_{i5} \in \{0,1\}$

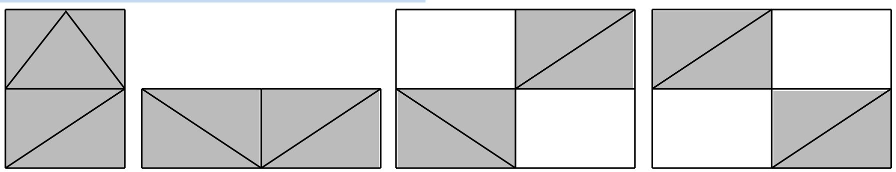


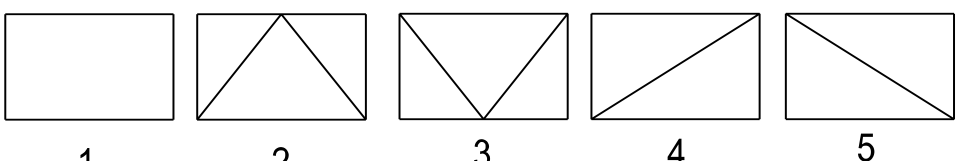
Convolution

- Relation to neighboring brace is important
  → Extract features by convolution filter



4 directions





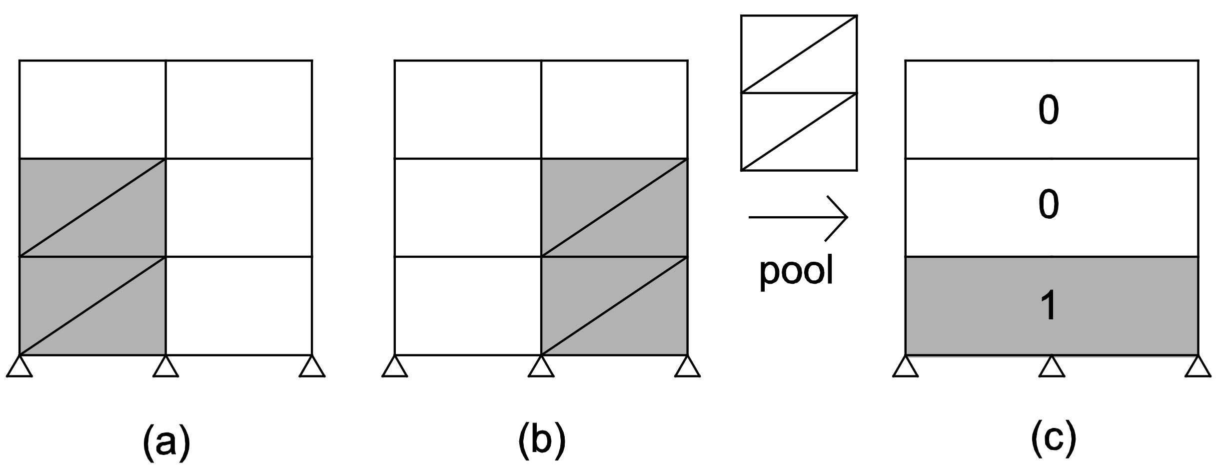
Patterns of filter: 5 x 5 x 4 = 100
Number of locations: 5 x 3 = 15
Total number of variables: 100 x 15 = 1500

Pooling

- Many variables after convolution

  → Reduce number of variables by pooling

Example: combine two same features in the same story (Max. pooling in horizontal dir.).



Learning for 2 classes

- Randomly generate 10000 solutions

- Approximate optimal solutions:
  top 10% solutions
  Label = +1
- Non-optimal solutions:
  worst 10% solutions
  Label = –1

Accuracy of learning

- Label = +1: Approximate optimal
- Label = –1: Non-optimal
- Small ratio of ‘False-Negative’ (missing approx. optimal)

  → High accuracy of learning

		Actual Label	
		+1	–1
Predicted Label	+1	True Positive (TP)	False Positive (FP)
	–1	False Negative (FN)	True Negative (TN)

Learning results by SVM

	FN	FP
Without convolution /pooling	17/1000	35/1000
With convolution Without pooling	10/1000	10/1000
With convolution and pooling	15/1000	18/1000

Contribution of filter variable

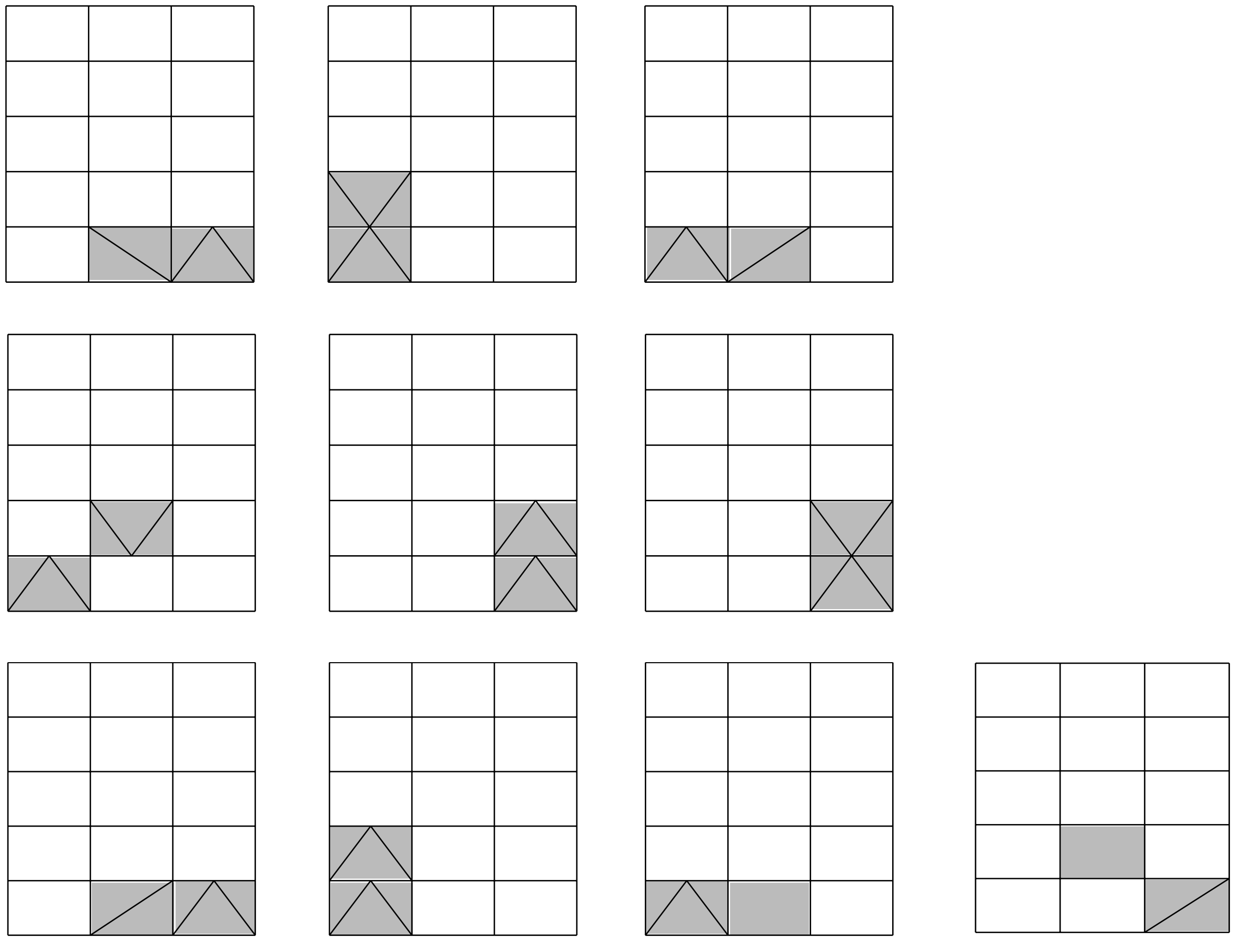
- Calculation of score using linear kernel

$$S(x) = \frac{1}{a} \cdot \boldsymbol{\beta} \cdot \boldsymbol{x} + \boldsymbol{b}$$

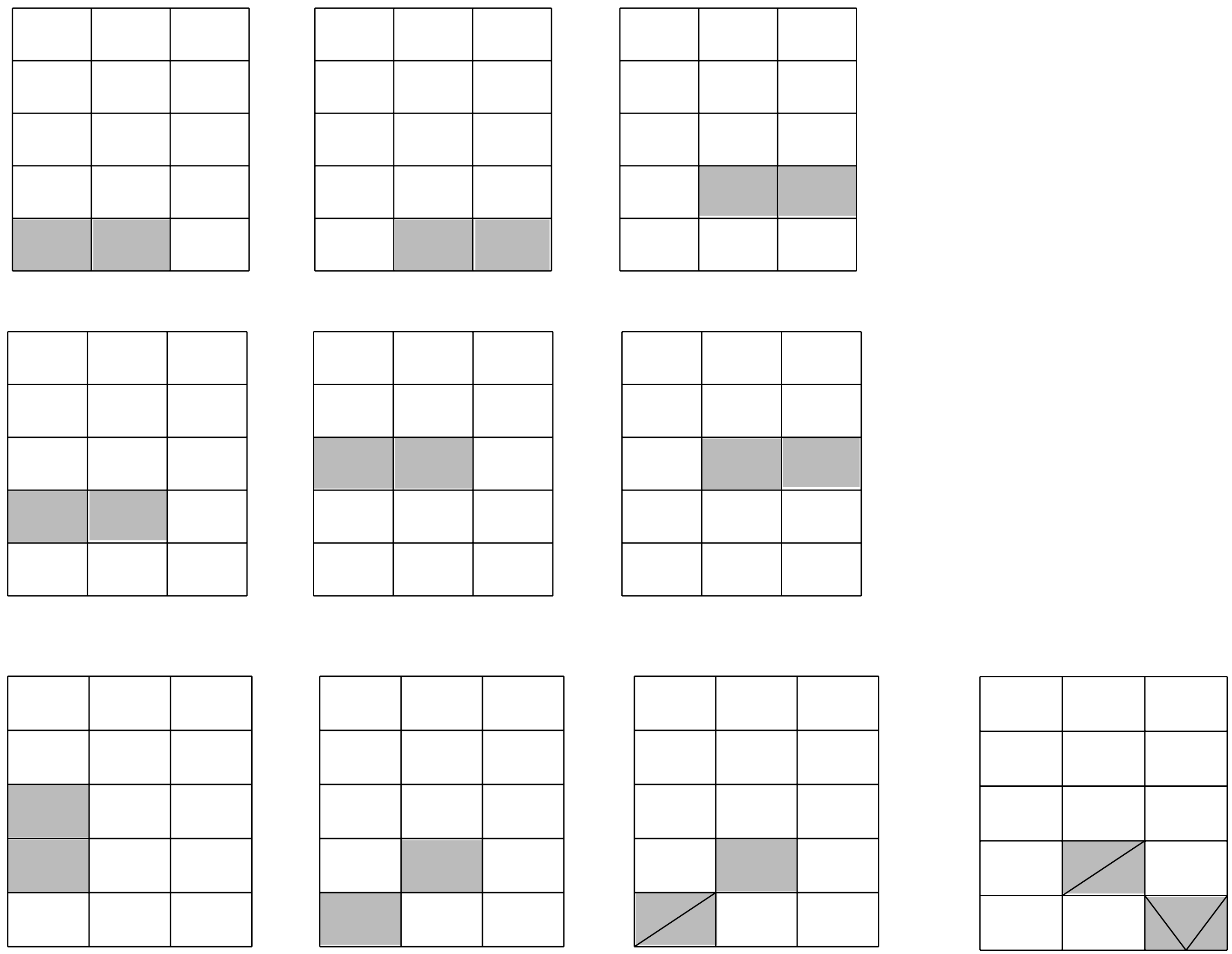
$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m)$: coefficient vector

- If β_i is large positive, $x_i = 1$ contributes to approximate optimal solution.
- If β_i is small negative, $x_i = 1$ contributes to non-optimal solution.

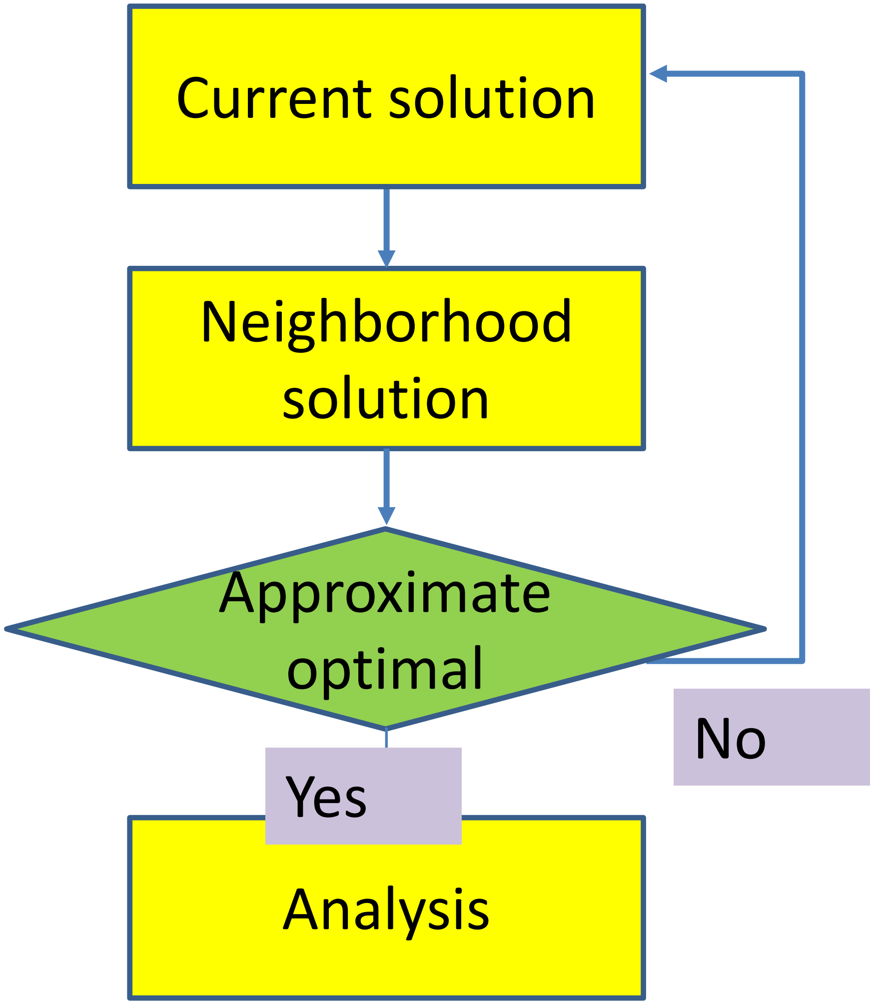
Filters characterizing approximate optimal solutions

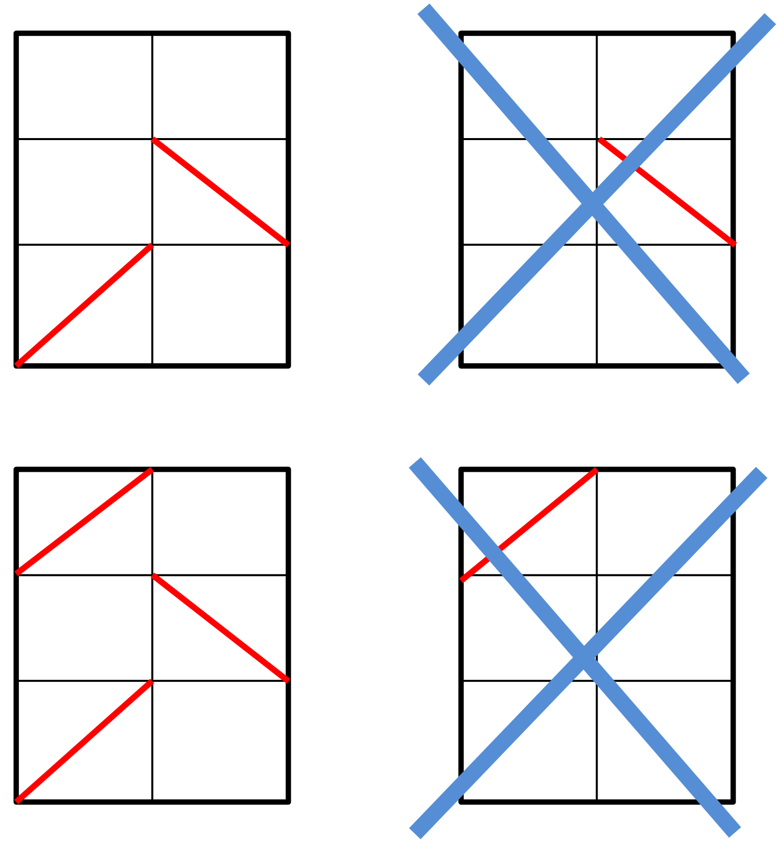


Filters characterizing non-optimal solutions



SA with machine learning



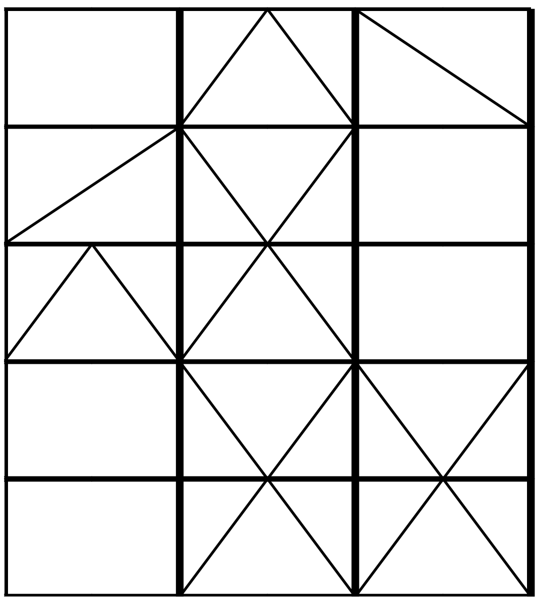


Neighborhood solutions

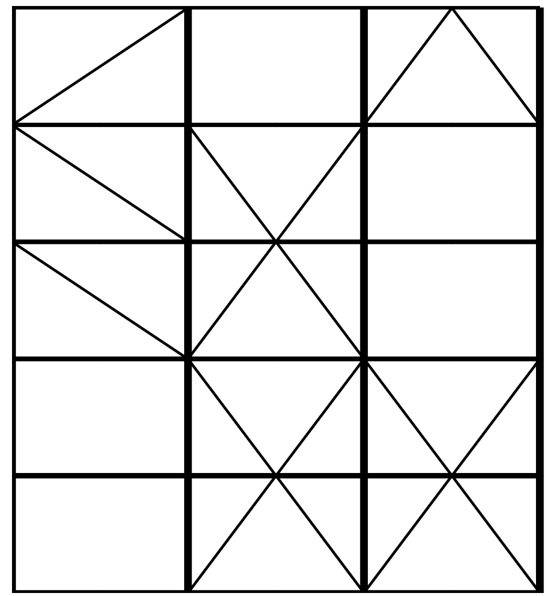
Comparison of computational time

		SA	SA+SVM
Learning	Analysis (s)	---	2093
	Learning (s)	---	12.4
Optimization	Prediction (s)	---	483.7
	Analysis (s)	14314.3	7961.6
	Optimization incl. learning (s)	14314.3	10550.7
	Number of analyses	67368	35710
	Optimal objective value	84.83 N/mm^2	87.08 N/mm^2

Optimization result



SA



SA with SVM

Utilization of learning results of small frame to large frame

Use learning result of a frame model to optimize the same frame
→ not effective for large frame

⇩

Use learning result of a small frame to optimize a large frame
→ effective for large frame

Utilization of learning results of lowrise frame to highrise frame

Estimation using RBF kernel for SVM

$$S(\mathbf{x}) = \sum_{i=1}^{n} \left\{ f_i \alpha_i^* \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x} \right\|_2^{\,2} \right) \right\} + b^*$$

$S(\mathbf{x})$: Score of data $\mathbf{x}$

γ : Scaling parameter

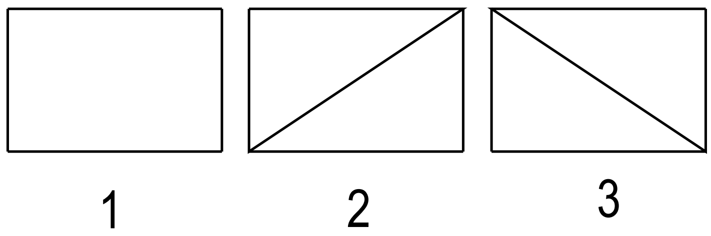
f_i : Label of ith learning data

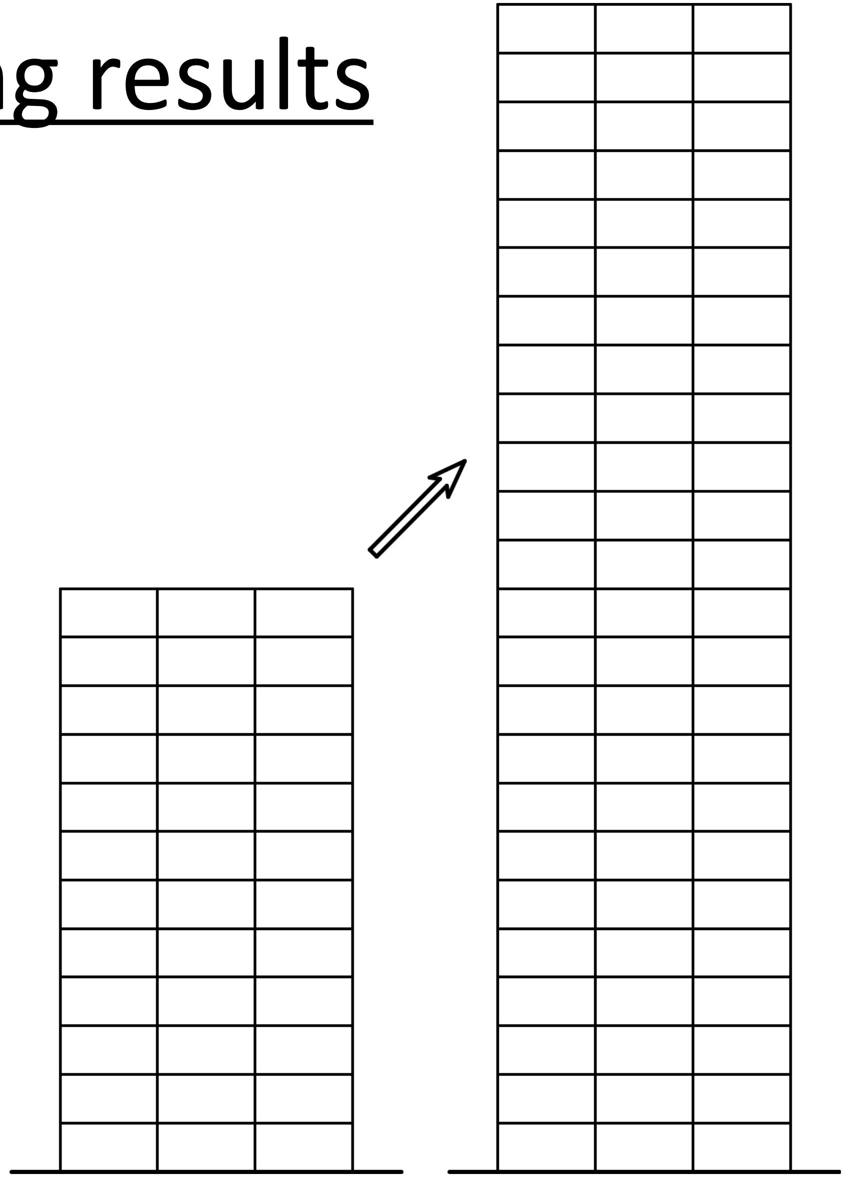
α_i^* : Weight of ith learning data

b^* : Bias parameter

Utilization of learning results of lowrise frame to highrise frame

Three types of braces including no-brace





Lowrise (12-story) frame

Highrise (24-story) frame

Conversion of feature variables

X^L: Converted feature variables of lowrise frame
X^H: Feature variables of highrise frame
⇨ Reassemble to 3-row matrix corresponding to three types of braces
C: Conversion matrix

$$X^L = CX^H$$

$$\overset{X^L : n_1 \times 3}{\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots \\ x_{n_1 1} & x_{n_1 2} & x_{n_1 3} \end{bmatrix}} = \overset{C : n_1 \times n_{f2}}{\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1 n_{f2}} \\ c_{21} & c_{22} & \cdots & c_{2 n_{f2}} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n_1 1} & c_{n_1 2} & \cdots & c_{n_1 n_2} \end{bmatrix}} \overset{X^H : n_2 \times 3}{\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots \\ x_{n_2 1} & x_{n_2 2} & x_{n_2 3} \end{bmatrix}}$$

Identification problem

C: Conversion matrix

b^*: Bias parameter

Find C and b^*

minimize $V_c = \frac{1}{\sigma^2} \sum_{\mathbf{x}_i \in (\mathrm{fn} \cup \mathrm{fp})} S(\mathbf{x}_i)^2 + w_1 \sum_{i,j} |c_{ij}| + w_2 R$

subject to $c_{ij} \geq 0 \quad (i = 1,2,\ldots,n_{F1},\ j = 1,2,\ldots,n_{F2})$

Penalty term to obtain sparse matrix

$\mathbf{x}_i$: ith data

fp, fn: Data sets of FN and FP

$S(\mathbf{x})$: Score function

σ^2 : Variance of score

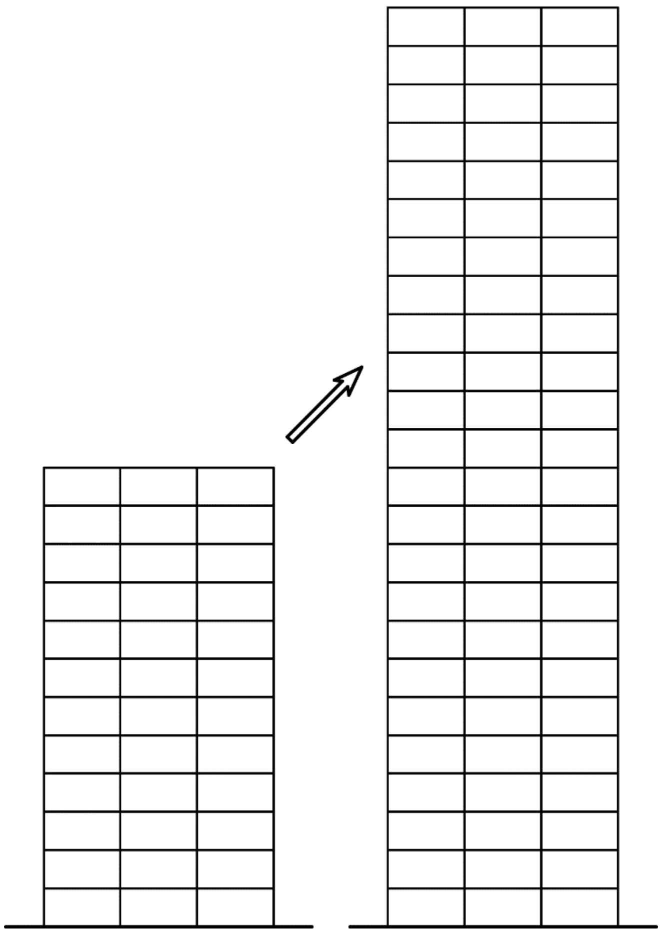
$w_1,\ w_2$: Weight parameters

$R\ (<0)$: Correlation coef. between score and index

Algorithm of extimation of properties of highrise frame

- **12-story 3-span**
  ⇨ **24-story 3-span**

1. Generate 10000 data for 12-story frame.
2. Carry out learning for 12-story frame.
3. Generate 1000 data for 24-story frame.
4. Solve optimization (identification) problem for C and *b**.
5. Estimate properties of 24-story 3-span frame.



Matrix and bias

$C =$

0.243	0.005	0.001	0.000	0.001	0.001	0.012	0.000	0.002	0.001	0.005	0.001	0.008	0.002	0.000	0.004	0.003	0.001	0.002	0.000	0.004	0.009	0.001	0.002
0.000	0.360	0.016	0.003	0.000	0.017	0.003	0.000	0.000	0.000	0.002	0.000	0.003	0.000	0.000	0.001	0.000	0.010	0.000	0.000	0.001	0.008	0.001	0.004
0.031	1.497	0.108	0.046	0.012	0.116	0.007	0.019	0.041	0.005	0.010	0.013	0.184	0.000	0.020	0.023	0.008	0.018	0.027	0.009	0.003	0.062	0.156	0.012
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.460	0.026	0.008	0.000	0.034	0.013	0.002	0.010	0.003	0.008	0.002	0.058	0.002	0.001	0.009	0.001	0.007	0.008	0.000	0.006	0.023	0.039	0.005
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.015	0.000	0.000	0.000	0.004	0.000	0.000	0.006	0.000	0.000	0.000	0.001	0.000	0.000	0.007	0.000	0.003	0.000	0.001	0.003	0.000	0.000	0.000	0.001
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

$\times 10^{-2}$

$$b^* = 0.2811$$

: Non-zero value

: Large value

: Largest value

Variables in 2nd story has large effect on property of highrise frame

Estimation result of highrise frame

	Stress minimization	
	FN	FP
Case 1	0/1000	12/1000
Case 2	7/1000	0/1000
Case 3	261/1000	254/1000

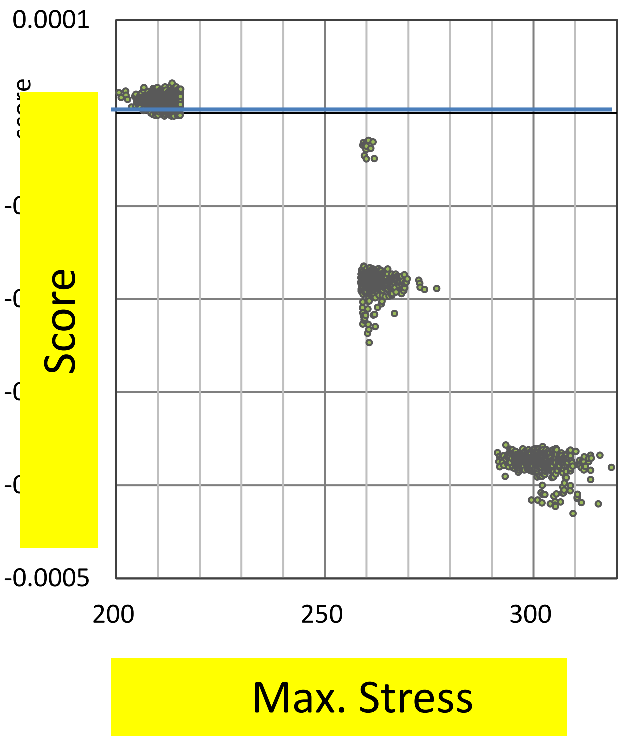
Case 1: Proposed method (optimize matrix and bias)

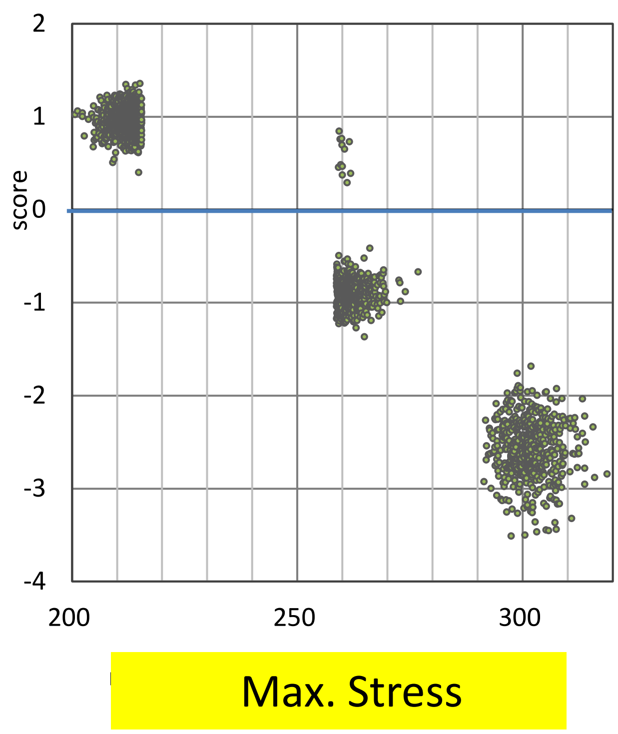
Case 2: Carry out learning for highrise frame

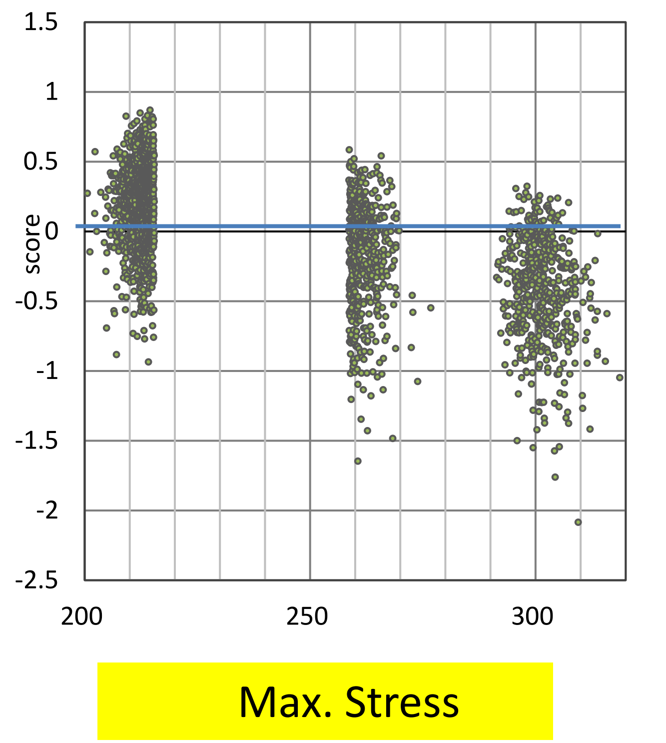
Case 3: Average of two stories and optimize bias only

Relation between stress and score

Case 1	Case 2	Case 3







Case 1: Proposed method (optimize matrix and bias)

Case 2: Carry out learning for highrise frame

Case 3: Average of two stories and optimize bias only

Conclusions

- A method based on SA for optimization of brace locations of building frames.
- Minimize maximum additional stress of beams and columns under horizontal static loads representing seismic loads.
- Distinct classification of approximate optimal and non-optimal solutions is effective to improve the accuracy of learning and prediction.
- Learning result of lowrise frame can be used for predicting the performance of highrise frame by optimizing the conversion matrix and bias.