西湖大學
WESTLAKE UNIVERSITY

Multi-Objective Machine Learning

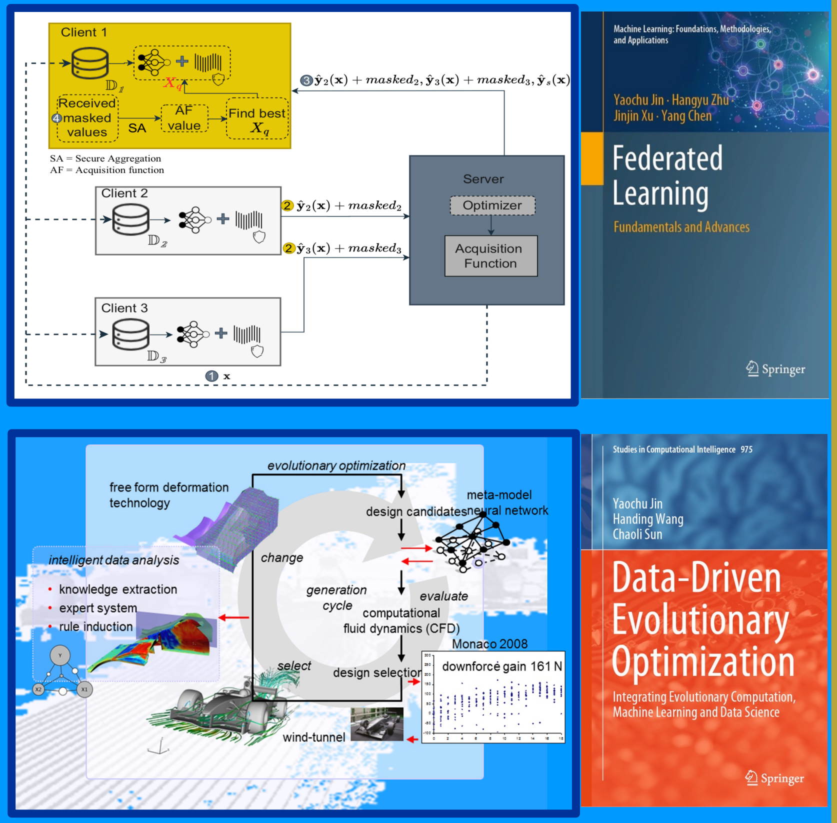
Yaochu Jin

Trustworthy and General Artificial Intelligence Laboratory

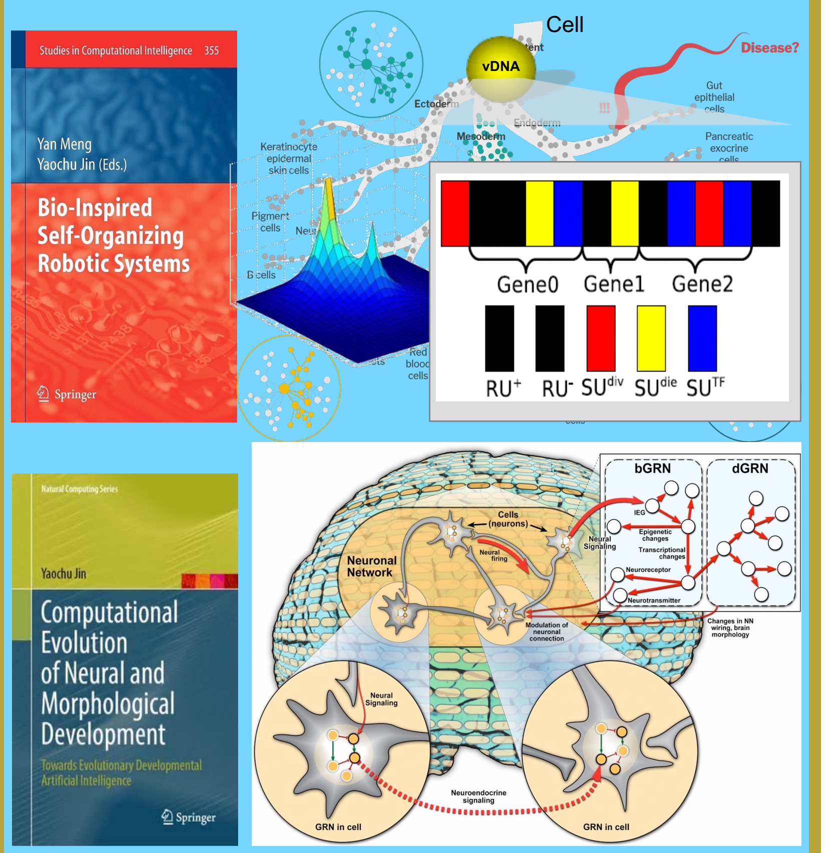
Trustworthy and General AI Lab @Westlake
可信及通用人工智能实验室

西湖大學 WESTLAKE UNIVERSITY

Trustworthy AI for Industry



Evolutionary Developmental AGI



Trustworthy AI (可信人工智能)

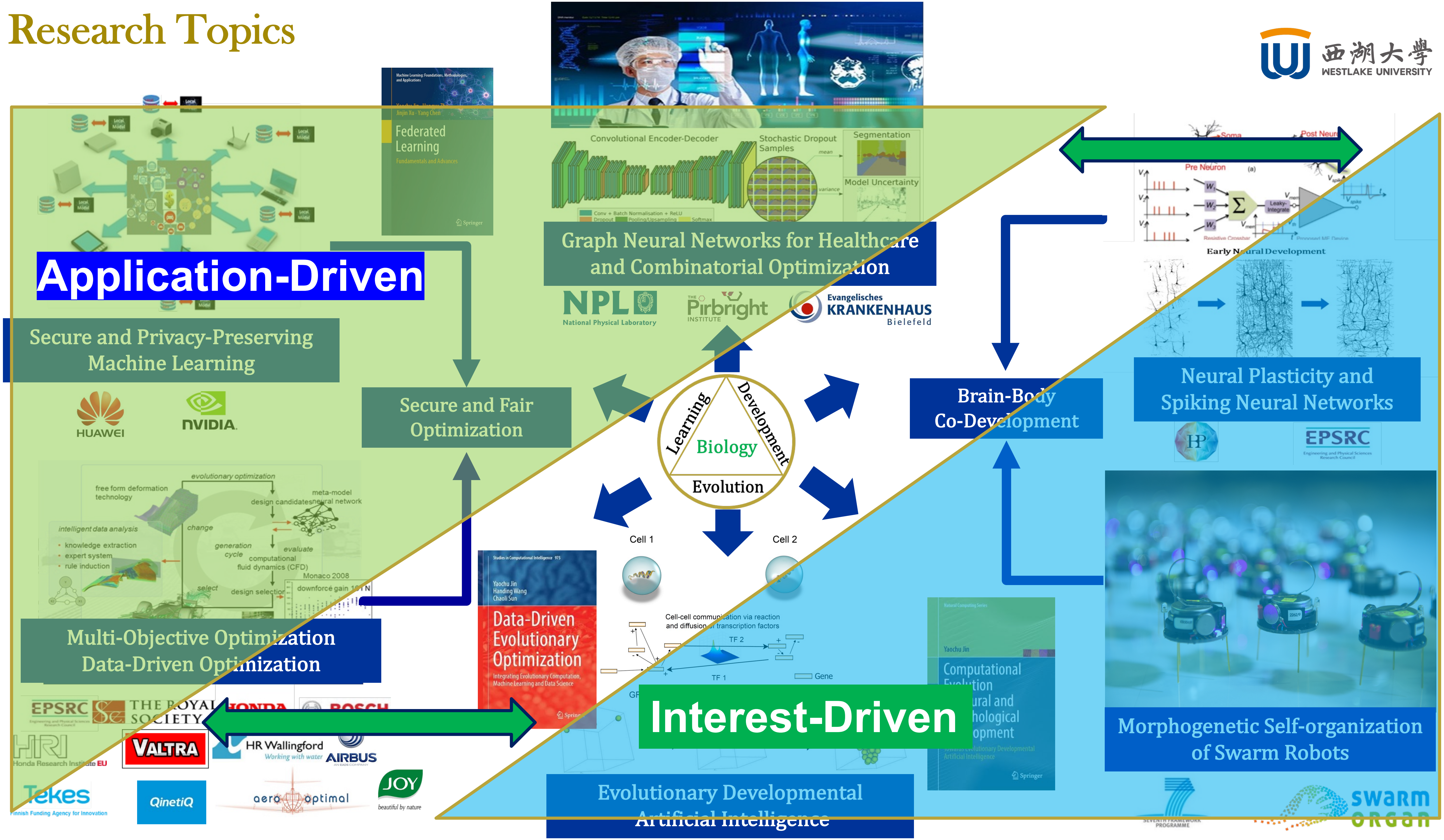
- Secure, privacy-preservation and fairness-aware learning and optimization
- Multi-modal learning and optimization
- LLMs for general optimization

General AI (通用人工智能)

- Plasticity in spiking neural networks
- Brain-body co-evolution
- Evolving autonomous learning
- Evolving human-centered LLMs

可信及通用人工智能实验室
TRUSTWORTHY AND GENERAL ARTIFICIAL INTELLIGENCE LABORATORY

Research Topics



西湖大學 WESTLAKE UNIVERSITY

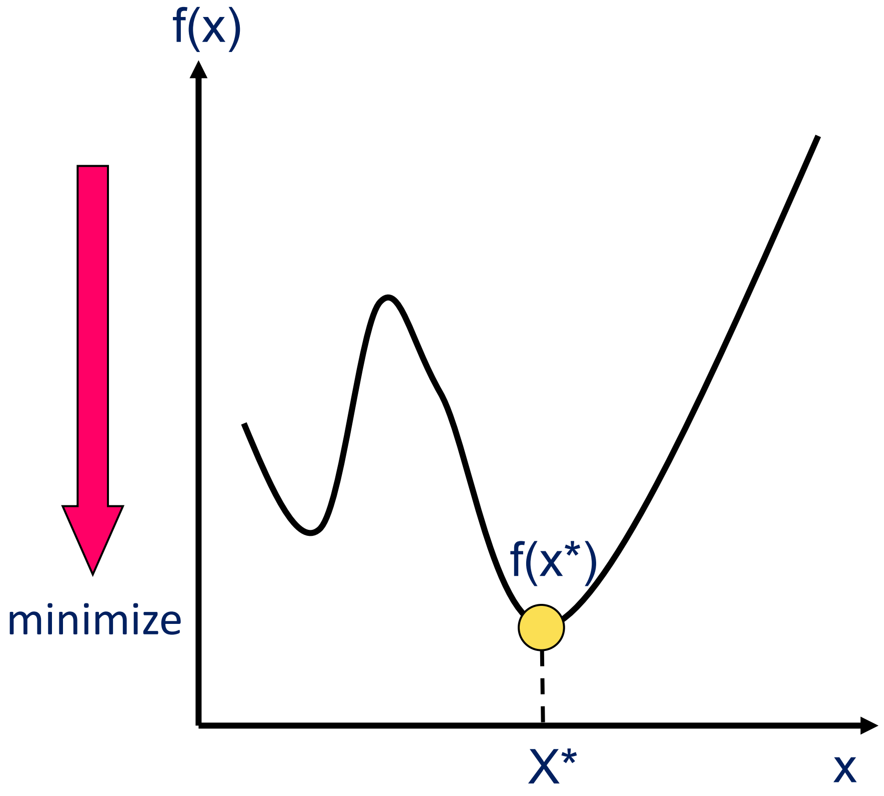
Outline

- Multi-objective evolutionary optimization
  - NSGA-II: Elitist non-dominated sorting genetic algorithm
- Multi-objective machine learning
  - Machine learning models and algorithms
  - Interpretable symbolic rule extraction from neural networks
  - Multi-objective clustering
  - Diverse feature extraction
  - Communication-efficient federated learning
  - Multi-objective adversarial learning
- Summary and future work

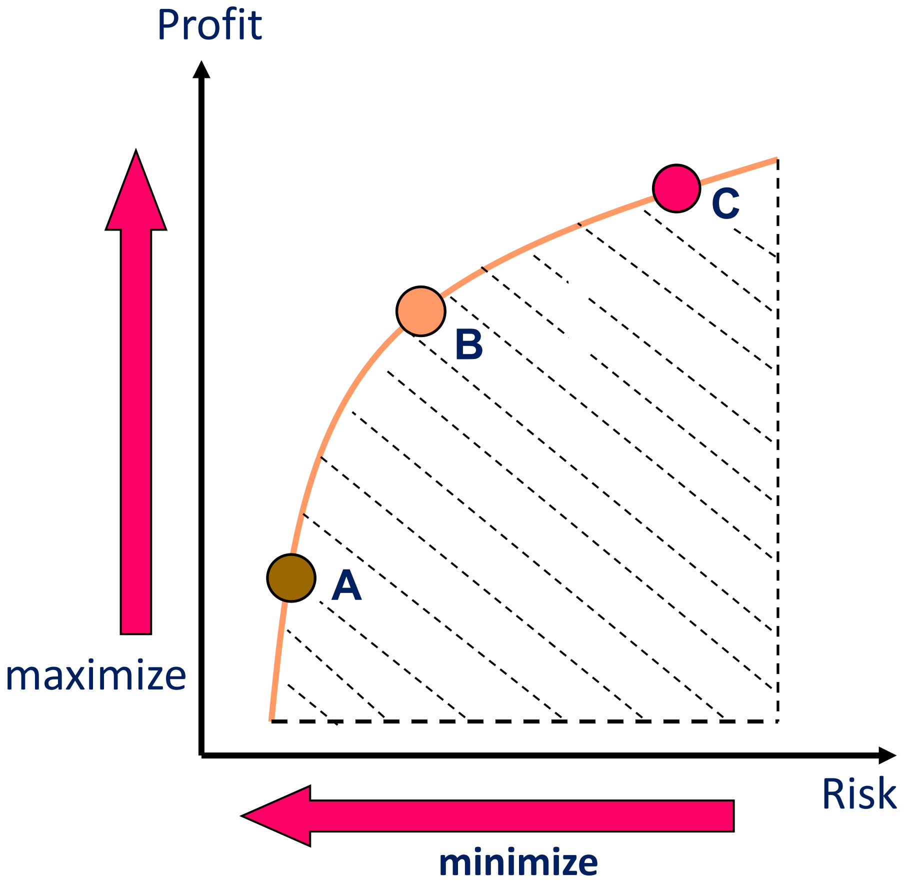
Multi-objective Evolutionary Optimization

Single and Multi-Objective Optimization

Single-objective optimization (SOO)



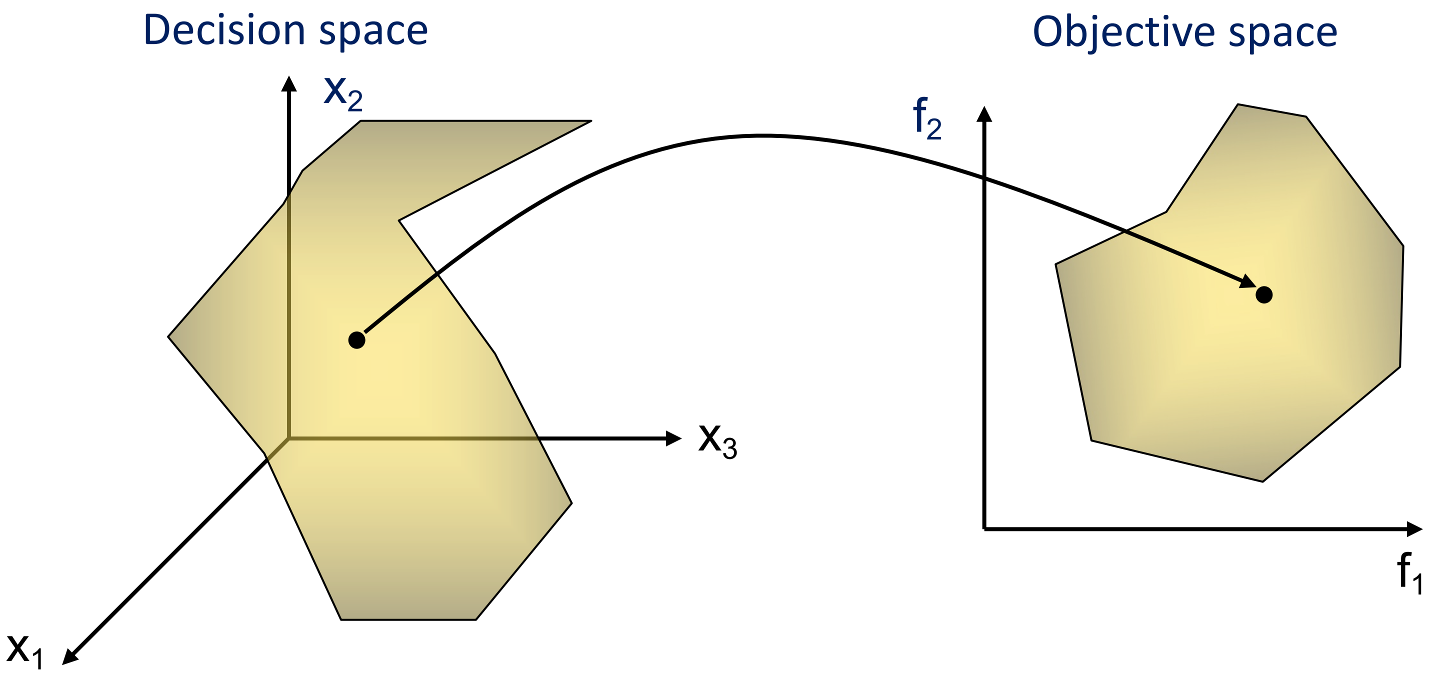
Multi-objective optimization (MOO)



- One single optimal solution can be found for SOO in most cases, whereas a finite or infinite number equally good solutions exist for MOO
- To choose a final solution, user preference is necessary

Mathematical Description of MOO

$$
\begin{aligned}
\text{minimize } & f_m(\mathbf{X}), && m = 1,2,\ldots, M; \\
\textit{s.t.} \quad & g_j(\mathbf{X}) \geq 0, && j = 1,2,\ldots, J; \\
& h_k(\mathbf{X}) = 0, \; k = 1,2,\ldots,K; && \\
& x_i^L \leq x_i \leq x_i^U, && i = 1,2,\ldots, n.
\end{aligned}
$$



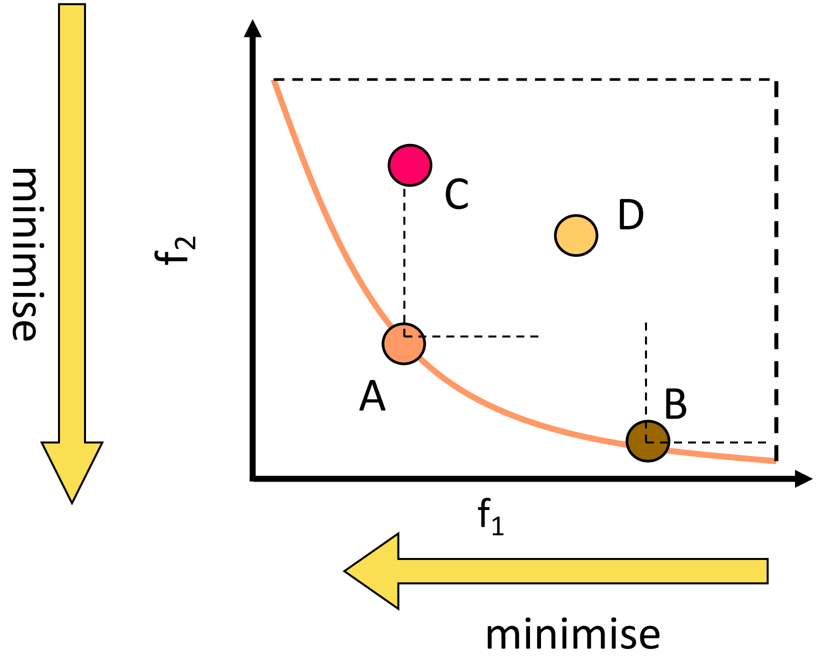
Dominance

- For minimisation problems, solution $\mathbf{X}^{(1)}$ dominates $\mathbf{X}^{(2)}$ if
  - Solution $\mathbf{X}^{(1)}$ is no worse than solution $\mathbf{X}^{(2)}$ in all objectives:

    $$\forall\ m=1,2,\ldots,M,\ \ f_m(\mathbf{X}^{(1)}) \leq f_m(\mathbf{X}^{(2)}),$$

  - Solution $\mathbf{X}^{(1)}$ is strictly better than $X^{(2)}$ at least in one objective:

    $$\exists\ m' \in 1,2,\ldots,M,\ f_{m'}(\mathbf{X}^{(1)}) < f_{m'}(\mathbf{X}^{(2)}).$$

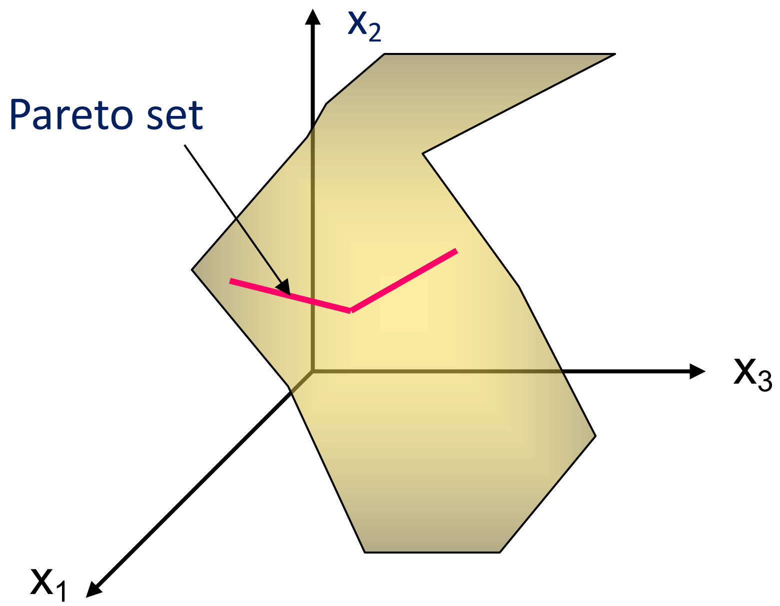


- A dominates C and D
- B is not dominated by A

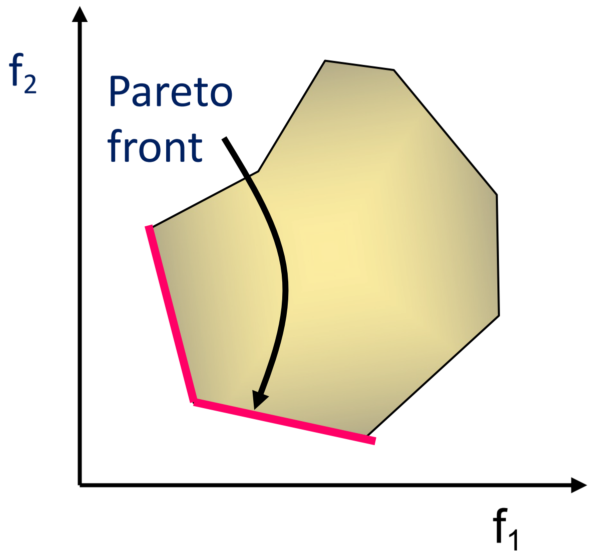
Pareto-Optimal Set and Pareto Front

- The set of all the Pareto optimal solutions is called the *Pareto set*
- The image of all Pareto optimal solutions in the objective space is termed *Pareto front*.

Decision space

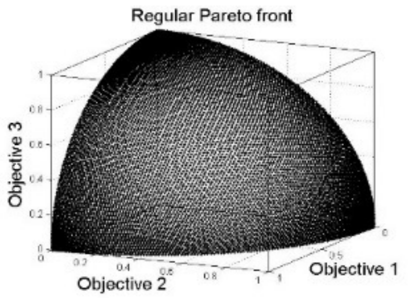


Objective space

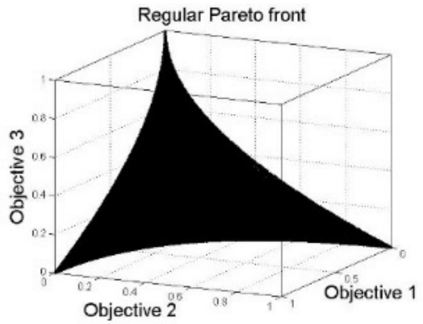


Shape of Pareto Fronts

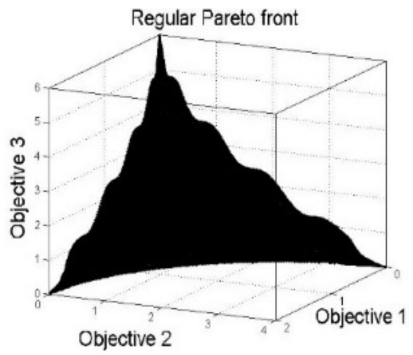
Regular Pareto Fronts



Concave

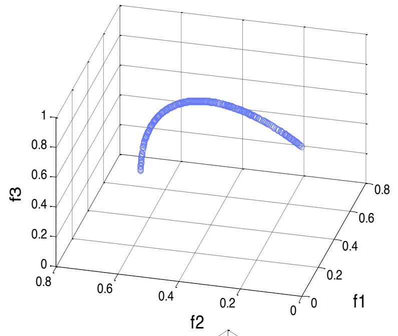


Convex

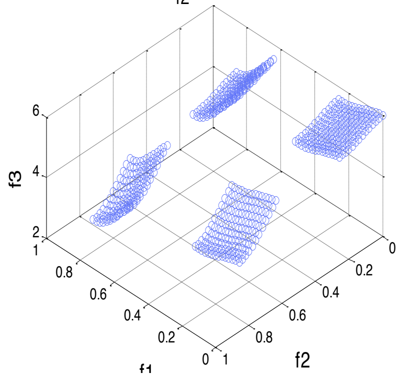


Non-convex

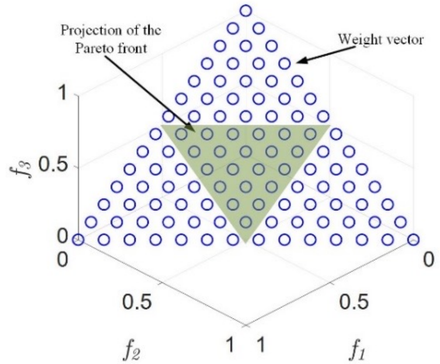
Irregular Pareto Fronts



Degenerate PF



Discontinuous PF



Inverted PF

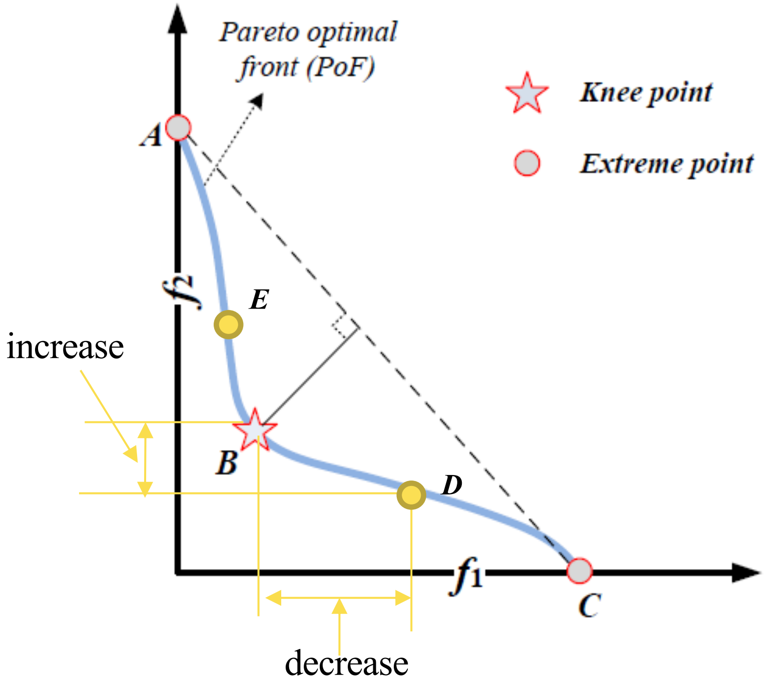
Y. Hua, Q. Liu, K. Hao, and **Y. Jin**. A survey of evolutionary algorithms for multi-objective optimization problems with irregular Pareto fronts. *IEEE/CAA Journal of Automatica Sinica*, 8(2): 303-318, 2021

Knee Points (Solutions)

- Knee points are solutions on the PoF and need a large compromise in at least one objective to gain a small improvement in other objectives [1].

- Physical significance: D→B or E→B: Much more profit on some objectives but a small unit of decrease on other objectives, in other words, it has highest cost performance.

- Geometrical features:
  - ➢ Large exterior angle [1]
  - ➢ Large distance to hyperplane [2]
  - ➢ Large hypervolume [3]



[1] K. Deb and S. Gupta, “Understanding knee points in bicriteria problems and their implications as preferred solution principles,” *Engineering Optimization*, vol. 43, pp. 1175–1204, 2011.
[2] E. Zitzler and L. Thiele, “Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach,” *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
[3] X. Zhang, Y. Tian, and Y. Jin, “A knee point driven evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015.

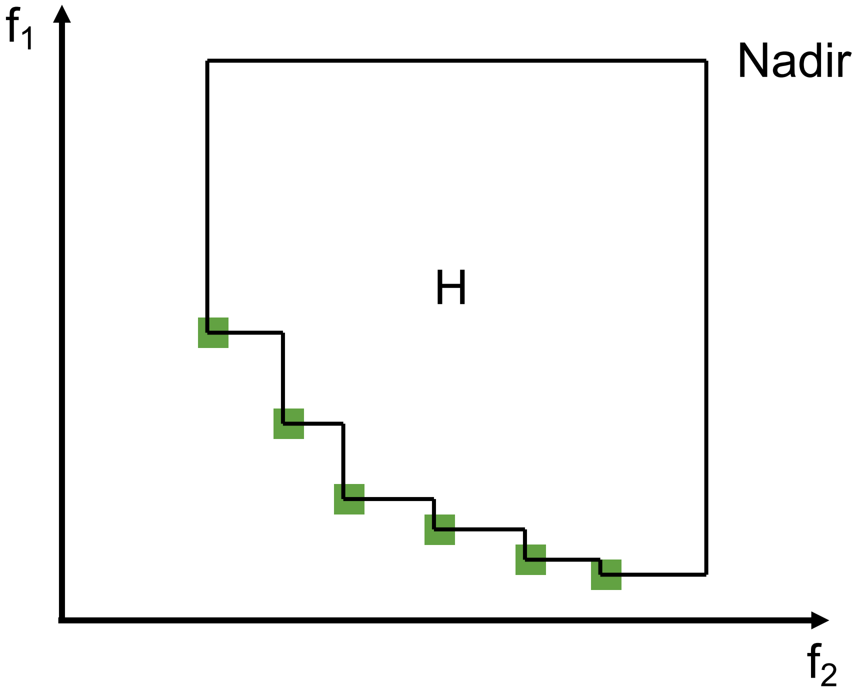
Differences Between SOO and MOO

西湖大學 WESTLAKE UNIVERSITY

	SOO	MOO
Target	• Find the global optimal solution	• Achieve the Pareto-optimal solution set or a representative subset
Performance Indices	• Accuracy • Efficiency	• Accuracy • Spread • Distribution • Efficiency
Problem Structure	• Fitness landscape (ruggedness, deceptiveness, multi-modality, correlation, etc.)	• Fitness landscape (ruggedness, deceptiveness, multi-modality, correlation, etc.) • Distribution of the Pareto-optimal solutions (finite/infinite, convexity, continuity, curve/surface, etc.)

Performance Indicator: HV

- Hypervolume (HV) is able to account for two aspects without a reference set, but the Nadir solution need to be defined
  - accuracy
  - diversity



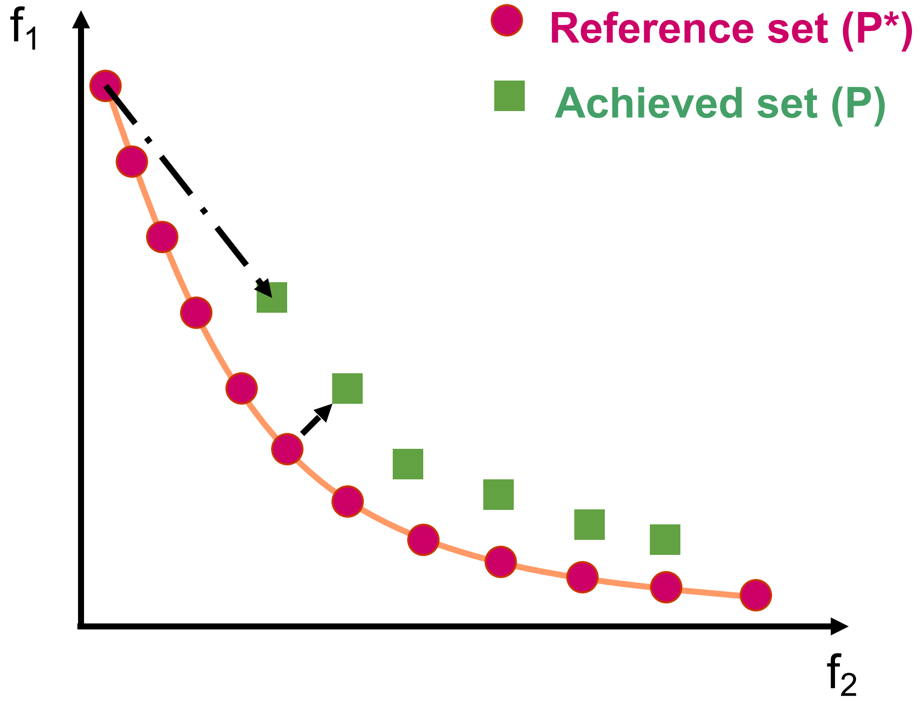
The larger H is, the better

Performance Indicator: IGD

- **Inverse generational distance** (IGD) is able to account for two aspects, if the reference set is large enough
  - accuracy
  - diversity

$$D(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|}$$

$d(v, P)$ is the minimal distance between a solution *v* in reference set P* and a solution in the achieved set P.

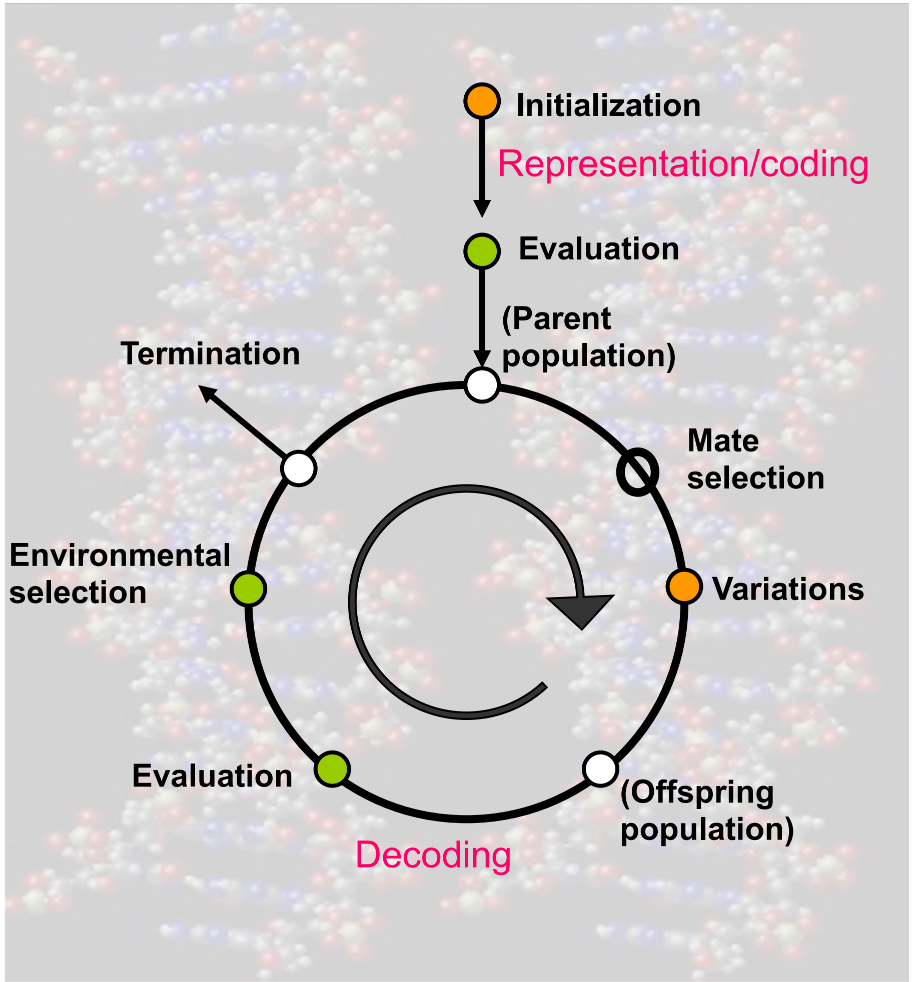


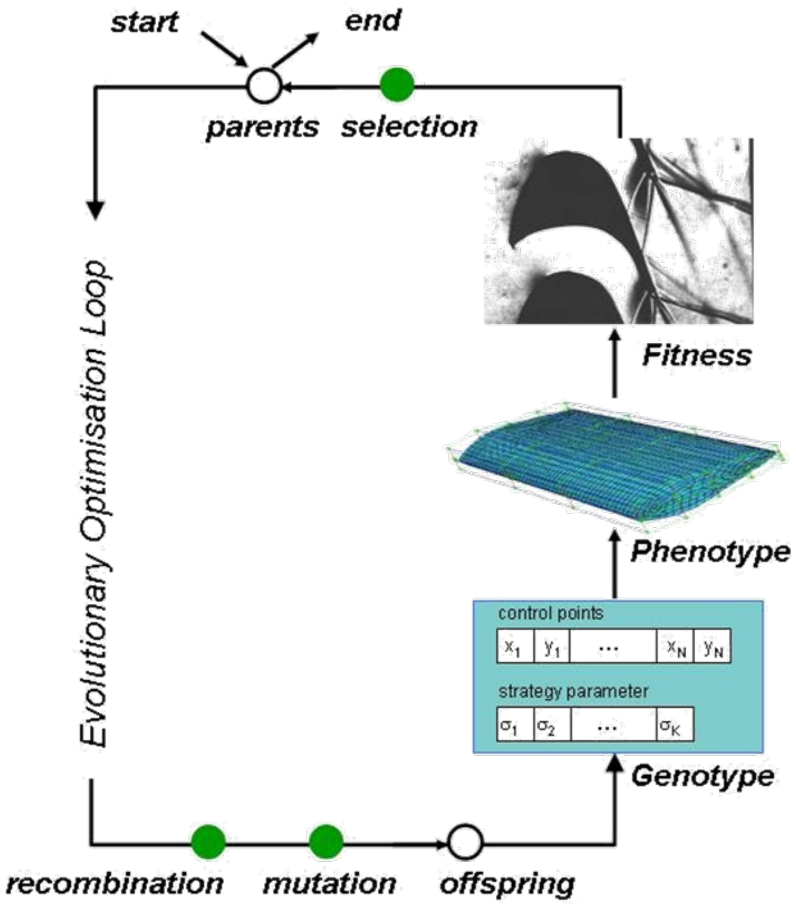
- Any issues with this performance indicator?

<u>The smaller D is, the better</u>

Evolutionary Algorithms for Optimization

Evolutionary algorithms and other meta-heuristic search methods are a class of population-based, guided stochastic search heuristics inspired from biological evolution and swarm behaviors of social animals

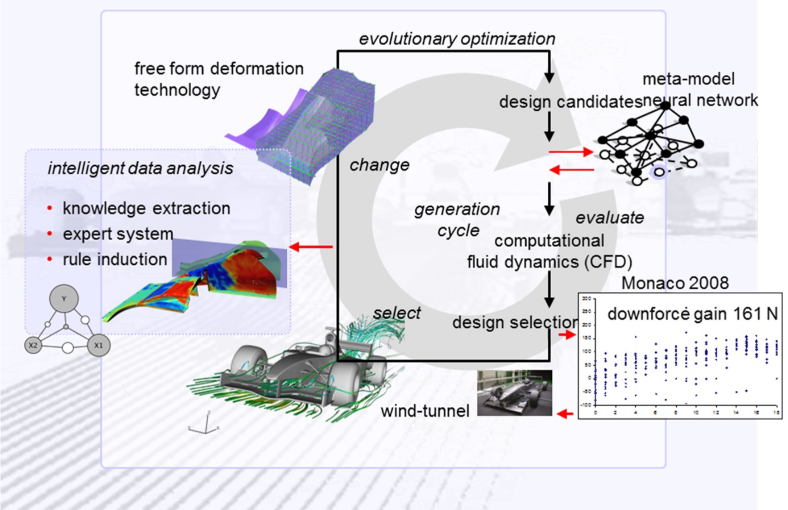




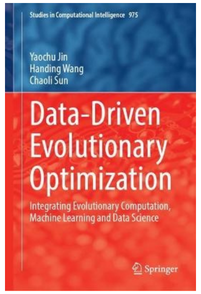
Y. Jin and B. Sendhoff. A systems approach to evolutionary multi-objective structural optimization and beyond. *IEEE Computational Intelligence Magazine*, 4(3):62-76, 2009

Challenges in Optimization of Complex Systems

- Problem formulation
- Large number of decision variables, multi- / many objectives, many constraints
- Optimization in the presence of uncertainties
  - Robust optimization
  - Dynamic optimization
  - Robust optimization over time
- Computational complexity
  - No analytic objective functions available, or data only
  - Computationally intensive
  - Experimentally costly



- **PlatEMO**, a software tool for teaching and research: https://github.com/BIMK/PlatEMO, which contains over **150** open-source algorithms and **300** benchmark and application problems



PlatEMO

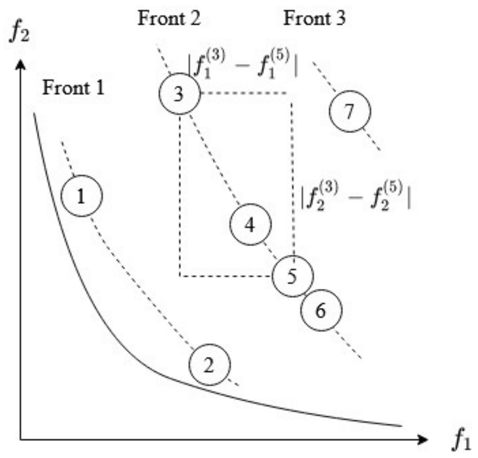
ANHUI UNIVERSITY 1928

UNIVERSITY OF SURREY

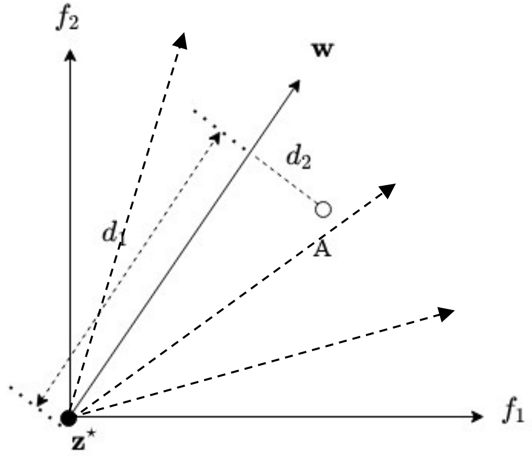
Y. Tian, R. Cheng, X. Zhang, and Y. Jin. **PlatEMO**: A MATLAB platform for evolutionary multi-objective optimization. *IEEE Computational Intelligence Magazine*, 12(4): 73-87, 2017 (**Winner of the "2019 IEEE CIM Outstanding Paper Award"**)

Evolutionary Multi-Objective Optimization

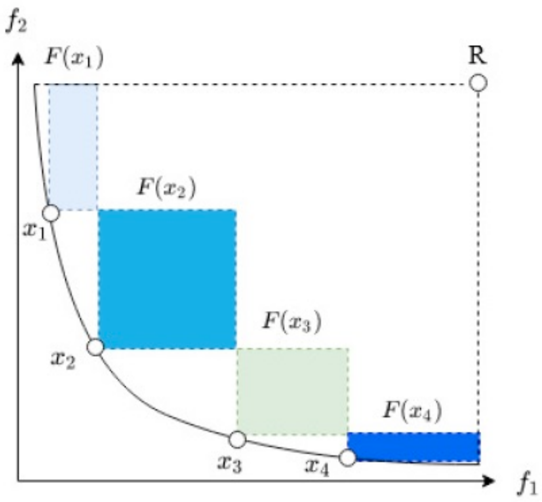
- Basic approaches to multi-objective optimization (**bi- or three-objective optimization**)
  - Pareto dominance based approaches
  - Decomposition using weight or reference vectors (cf. a scalarizing function)
  - Performance indicator based approaches



a) Pareto dominance based



b) Decomposition approaches



c) Performance indicator

K. Deb. Multi-objective optimization using evolutionary algorithms, Wiley, 2005

Many-Objective Optimization

MOPs with more than three objectives are called many-objective optimization problems (MaOPs)

- Dominance based approaches
  - Loss of selection pressure in Pareto-based approaches

- Performance indicator based approaches
  - Computational costs increases

- Decomposition based approaches
  - How many weights / reference vectors are needed to be representative?

- Solution assessment becomes tricky
  - The performance become very sensitive and also easily biased
  - Solution sets are no loner comparable
  - Diversity becomes trickier to measure

B. Li, J. Li, K. Tang, and X. Yao. Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys*, 48:13–35, 2015

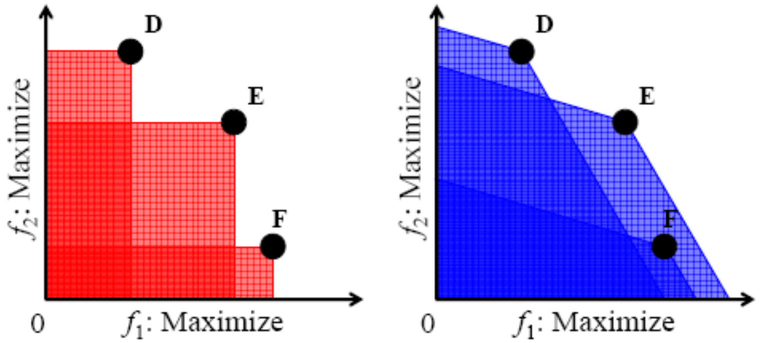
H. Ishibuchi, N. Tsukamoto, and Y. Nojima. Evolutionary manyobjective optimization: A short review. In: *Proceedings of IEEE Congress on Evolutionary Computation*, pages 2419–2426. IEEE, 2008

H. Wang, Y. Jin and X. Yao. Diversity assessment in many-objective optimization. *IEEE Transactions on Cyber*netics, 40(6):1510-1522, 2017

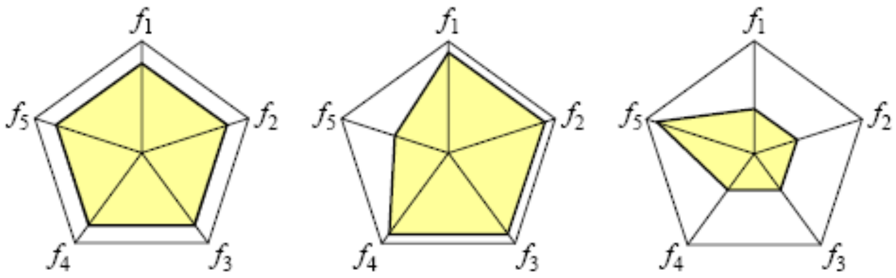
Many-Objective Optimization

Solutions:

- Reduce objective number (this is more problem formulation than optimisation)
- Modify the dominance definition, often by incorporating preferences (bias) decrease the number of non-dominated solutions
- Use performance indicator-based methods
- Use secondary selection criteria in addition to dominance
- Use decomposition (using weights, reference points, reference vectors …)



(a) Original Pareto dominance. (b) Modified dominance.



(a) Solution A. (b) Solution B. (c) Solution C.

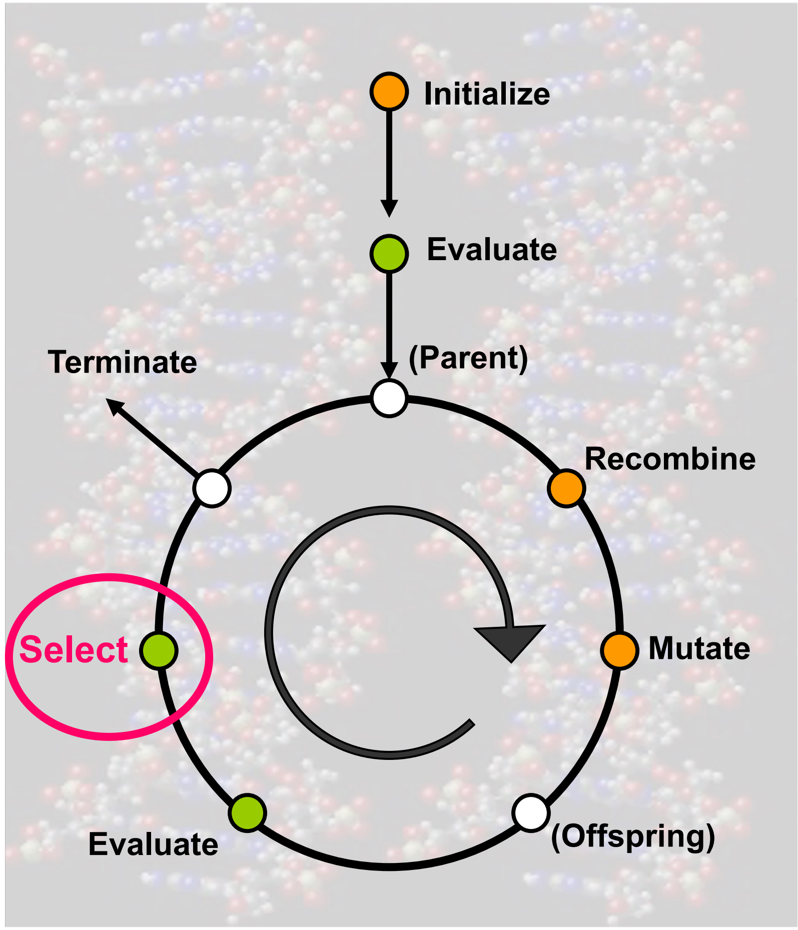
Solution B is favored if f1-f4 are more important

H. Ishibuchi, N. Tsukamoto, and Y. Nojima.Evolutionary Many-Objective Optimization: A Short Review. Proc. of *2008 IEEE Congress on Evolutionary Computation*, pp. 2424-2431, Hong Kong, June 1-6, 2008.

NSGA-II

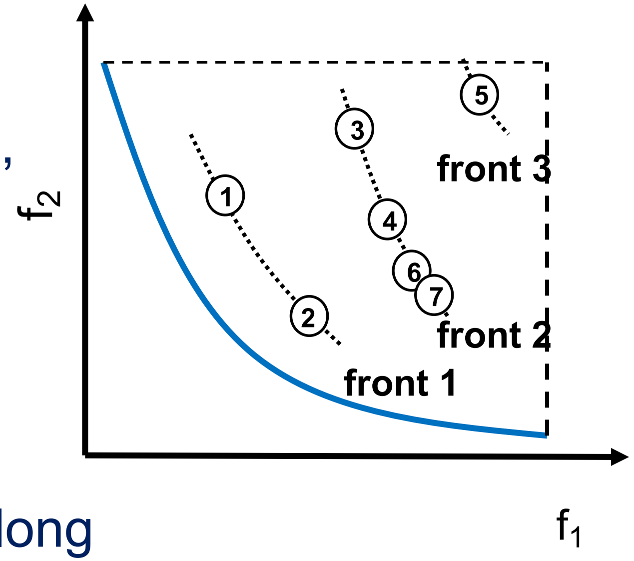
Dominance based Selection for MOO

- Different to single-objective optimization, the selection strategy must be modified – the fitness or rank based selection method is changed to **dominance (non-dominated sorting)** and **diversity** based selection



Non-Dominated Sorting

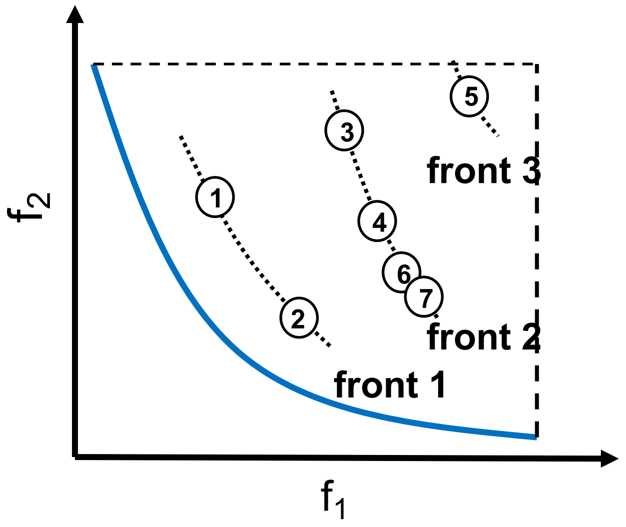
- The *basic* non-dominated sorting algorithm
  - find the non-dominated solutions in the population, which form the first non-dominated front. Assign a **rank 1** to all solutions of the first front;
  - remove the non-dominated solutions and find again the non-dominated solutions, which belong to non-dominated front 2. Assign a **rank 2**
  - Continue this process until all solutions in the population are assigned to a non-dominated front



Fast Non-Dominated Sorting

- Non-dominated sorting
  - For each solution p, record $\mathbf{n_p}$ **(number of solutions that dominate p)** and $\mathbf{S_p}$ **(list of solutions that are dominated by p)**

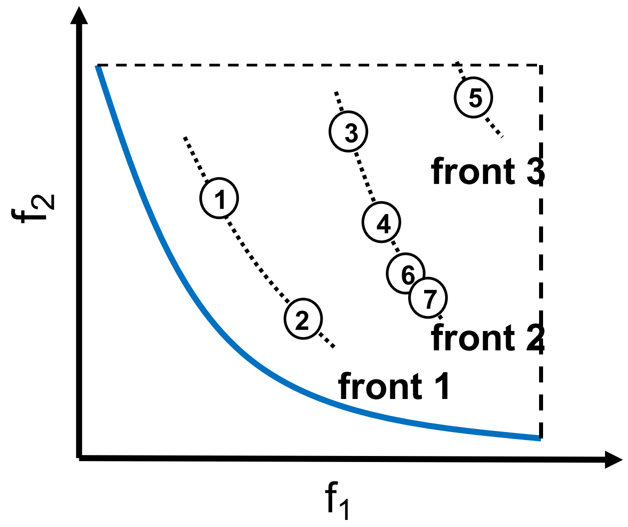
p	n_p (Number of solutions dominate p)	S_p (list of solutions dominated by p)	Rank
1	0	{3,5}	
2	0	{3,4,5,6,7}	
3	2	{5}	
4	1	{5}	
5	6	{}	
6	1	{5}	
7	1	{5}	



Fast Non-Dominated Sorting

- Non-dominated sorting
  - For all solutions p with n_p=0, assign rank 1 to them, and they form the front 1 in the set F_1.
  - Front counter i=1.

p	n_p (Number of solutions dominate p)	S_p (list of solutions dominated by p)	Rank
1	0	{3,5}	1
2	0	{3,4,5,6,7}	1
3	2	{5}	
4	1	{5}	
5	6	{}	
6	1	{5}	
7	1	{5}	



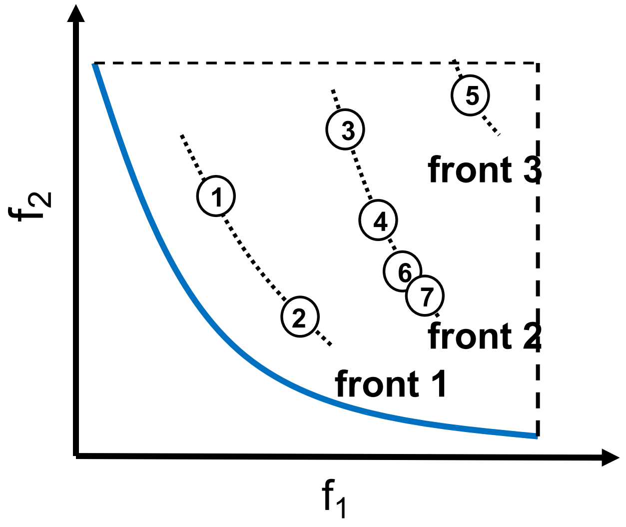
Fast Non-Dominated Sorting

- Non-dominated sorting (i=1)
  - ➢ For each solution p in F_i \\ `solutions 1 and 2 in this example`
    - ✓ For each solution in Sp \\ `{3,5} and {3,4,5,6}`

      $n_p = n_p - 1$
  - ➢ For the solutions with $n_p = 0$, assign rank $i+1$, and $i = i+1$;

p	n_p (No. of solutions dominate p)	Sp (list of solutions dominated by p)	Rank
1	0	{3,5}	1
2	0	{3,4,5,6,7}	1
3	**0**	{5}	**2**
4	**0**	{5}	**2**
5	**4**	{}	
6	**0**	{5}	**2**
7	**0**	{5}	**2**

(n_p is deduced by 1 for solutions 4, 6,7, and by 2 for solutions 3, 5)



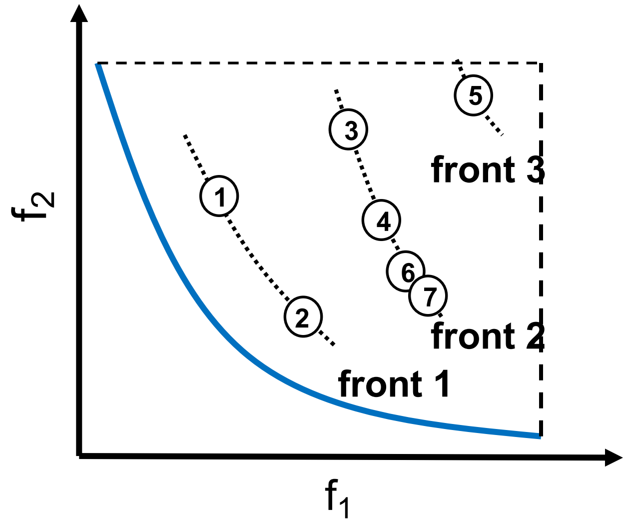
Fast Non-Dominated Sorting

- Non-dominated sorting (i=2)
  - ➢ For each solution p in F_i
    - ✓ For each solution in Sp

      $n_p = n_p - 1$

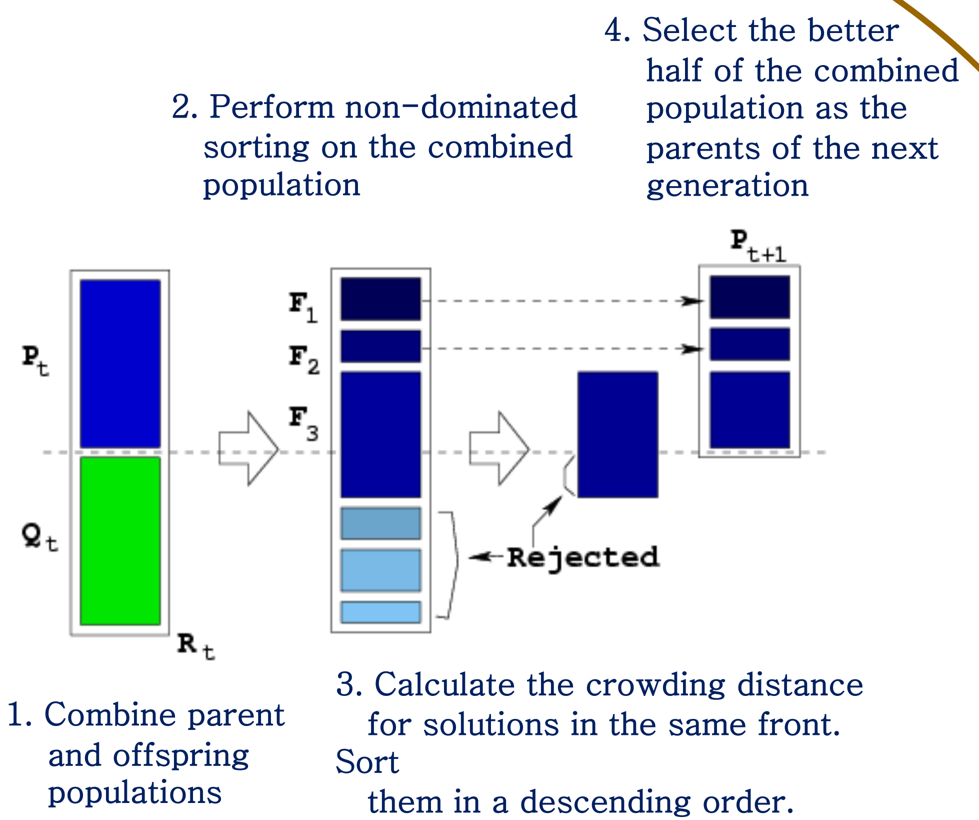
  – For the solutions with $n_p = 0$, assign rank i+1, and i=i+1;

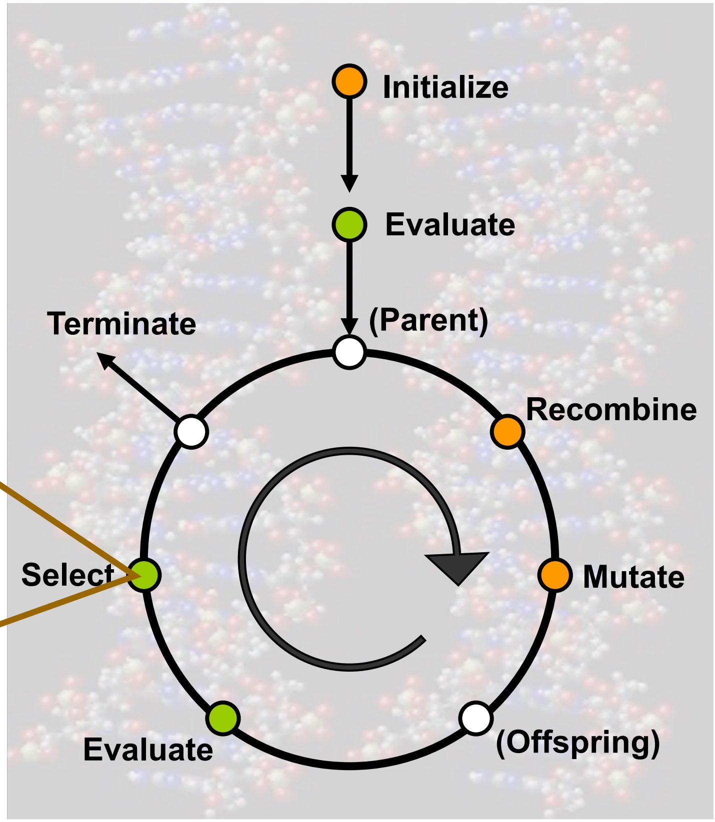
p	n_p (No. of solutions dominate p)	Sp (list of solutions dominated by p)	Rank
1	0	{3,5}	1
2	0	{3,4,5,6,7}	1
3	0	{5}	2
4	0	{5}	2
5	**0**	{}	**3**
6	0	{5}	2
7	0	{5}	2



(Sp is deduced by 4 for solution 5)

Elitist Non-Dominated Sorting Genetic Algorithm (NSGA-II)





Crowding Distance

- **Crowding distance**: For individuals in each non-dominated front, calculate the average side length of its two neighbouring solutions of solution *i* , e.g.,

$$d_4 = d'_{41}+d'_{42};\ \ d'_{41} = d_{41} / (f_1^{max} - f_1^{min});\ \ d'_{42} = d_{42} / (f_2^{max} - f_2^{min})$$

$$d_6 = d'_{61}+d'_{62};\ \ d'_{61} = d_{61} / (f_1^{max} - f_1^{min});\ \ d'_{62} = d_{62} / (f_2^{max} - f_2^{min})$$

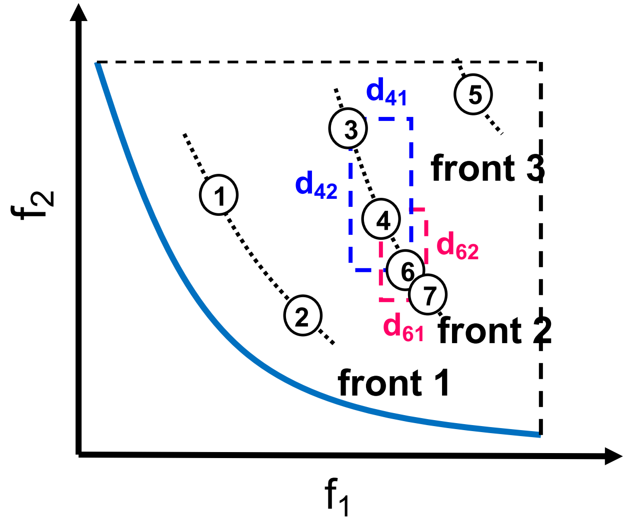
f_1^{min} and f_1^{max} is the minimum and maximum of f_1 in the current *front*;

f_2^{min} and f_2^{max} is the minimum and maximum of f_2 in the current *front*;

$$d_{41} = |f_1(I_3) - f_1(I_6)|,\ d_{42} = |f_2(I_3) - f_2(I_6)|,$$

$$d_{61} = |f_1(I_4) - f_1(I_7)|,\ d_{62} = |f_2(I_4) - f_2(I_7)|,$$

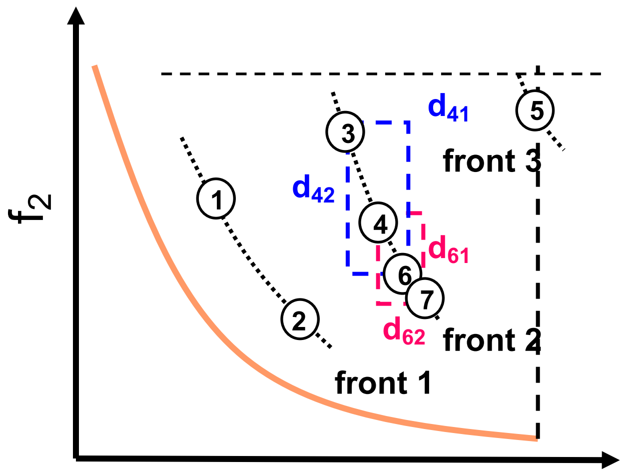
- **Assign a large distance (e.g., infinite) to the extreme solutions** -- solutions 1 and 2 for front 1, solutions 3 and 7 in front 2, and solution 5 in front 3



NSGA-II: Mate Selection: Crowed Tournament Selection

In the elitist non-dominated sorting genetic algorithm, crowded tournament selection is used for choosing two parents to generate offspring:

- Choose two solutions randomly;
- The solution with the better (lower) rank wins, e.g., 1 4 , solution 1 wins;
- If the solutions have the same rank, the one with the larger crowding distance wins, e.g., 4 6 , solution 4 wins;
- If the two solutions have the same rank and the same crowding distance, choose a winner randomly.

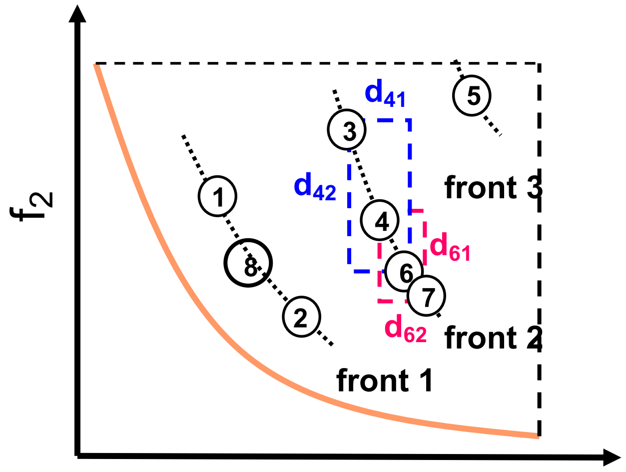


NSGA-II: Environmental Selection

Environmental selection in NSGA-II:

- Combine the parent and offspring population (elitism)
- Perform non-dominated sorting on the combined population
- Calculate the crowding distance for individuals **in the same non-dominated front**
- Rank the individuals based on the front number in an **ascending order**
- For individuals in the same front, rank them according to the crowding distance in a **descending order**
- Select N top-ranked solutions out of the $2N$ solutions in the combined population, where N is the population size

NSGA-II: Environmental Selection: An Example

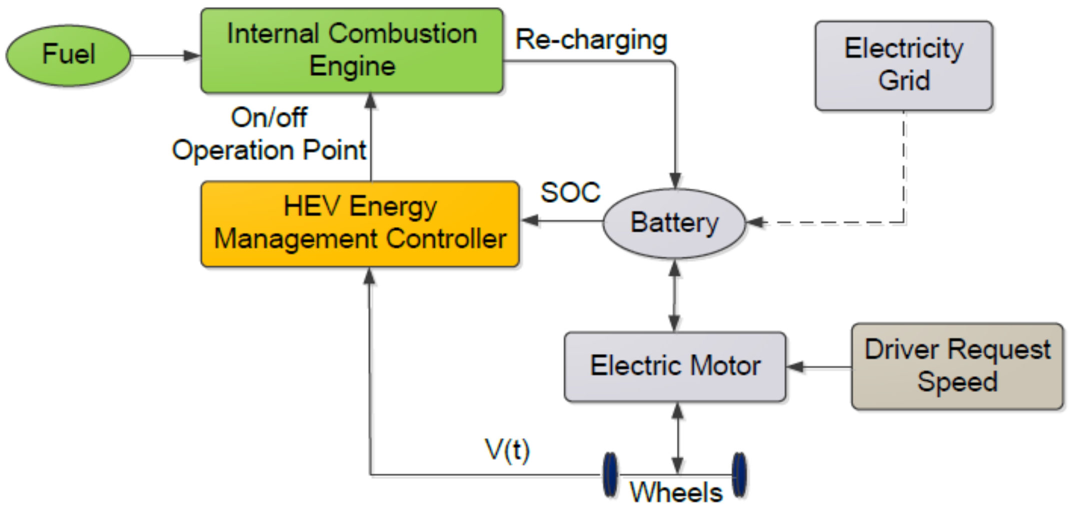


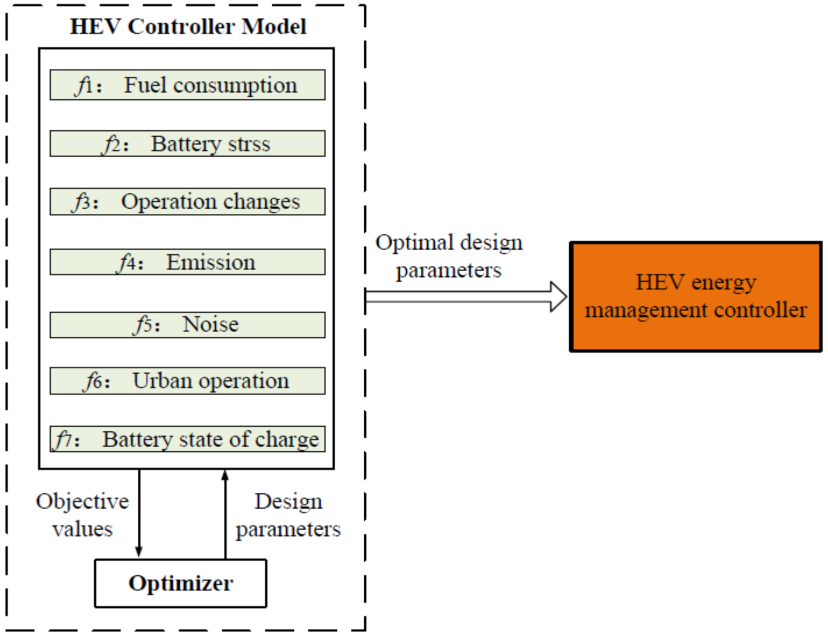
If 4 solutions are selected from the above eight, 1, 2, 8, 3 or 1,2,8,7 will be selected.

Examples of Real-World Optimization Problems

Hybrid Electric Vehicle Controller Design

西湖大學 WESTLAKE UNIVERSITY





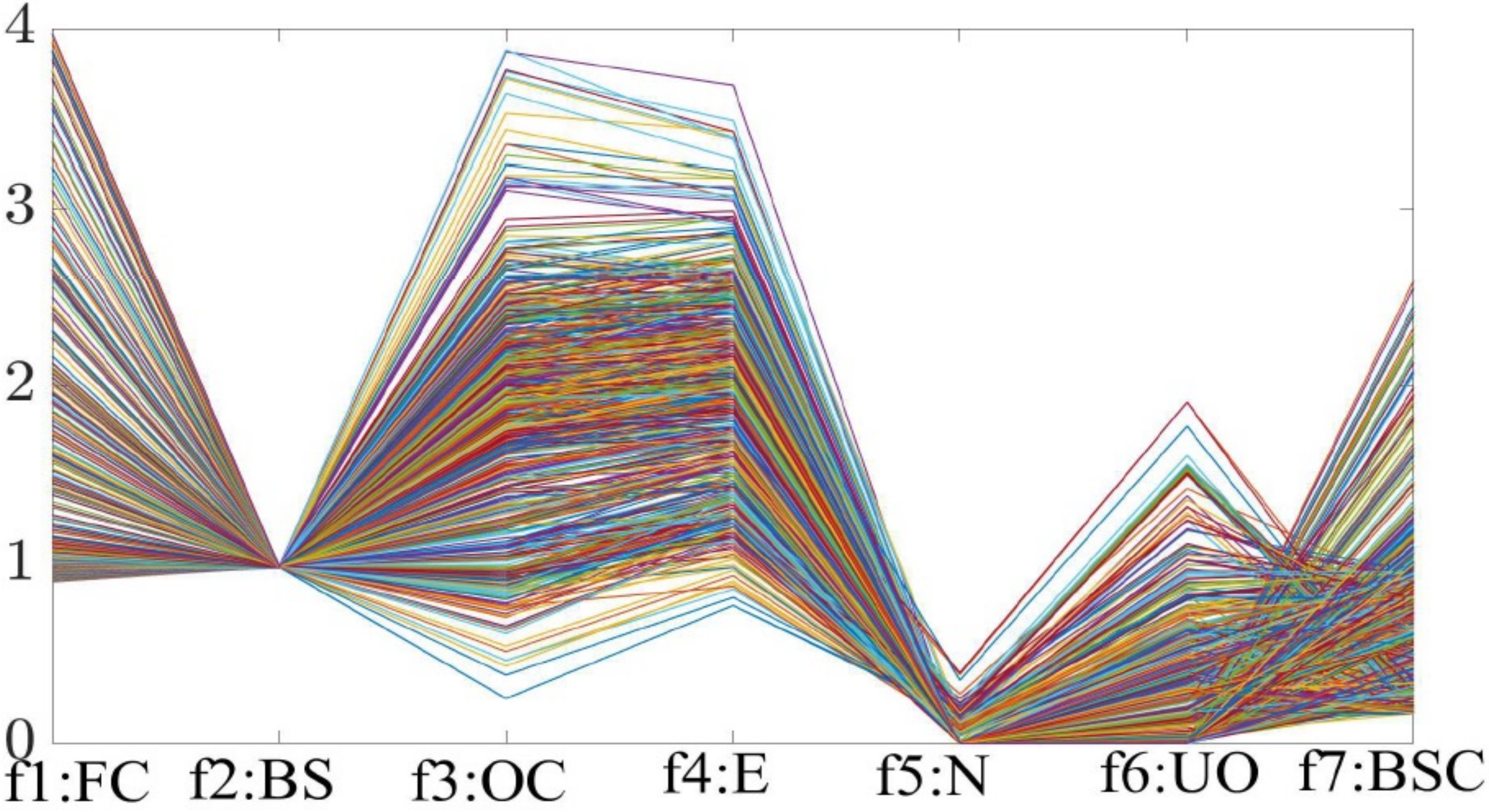
11 Decision variables:

SOC_{max} (%): SOC threshold to turn off ICE
SOC_{min} (%): SOC threshold to turn on ICE
v_1 (km/h): Lower speed for operation points
v_2 (km/h): Upper speed for operation points
rev_1 (/min): ICE speed for operation point 1
$torque_1$ (Nm): Torque for operation point 1
rev_2 (/min): ICE speed for operation point 2
$torque_2$ (Nm): Torque for operation point 2
rev_3 (/min): ICE speed for operation point 3
$torque_3$ (Nm): Torque for operation point 3
v_{off} (km/h): Speed threshold to turn off ICE

7 Objectives:

- **FC**: Fuel consumption and CO2
- **BS**: Battery stress
- **OPC**: ICE operation changes
- **Emission**: ICE emissions
- **Noise**: Perceived ICE noise
- **UO**: Urban operation
- **SOC**: Average battery state of charge level

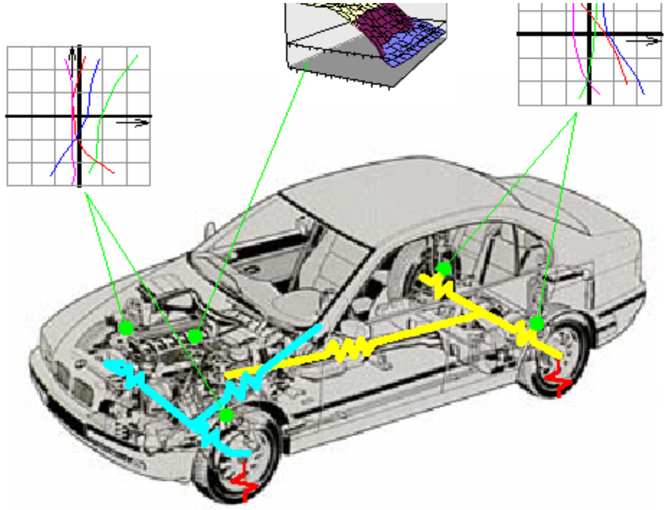
iGNG-RVEA – Hybrid Electric Vehicle Controller



Multi-Scenario Vehicle Dynamic Optimization

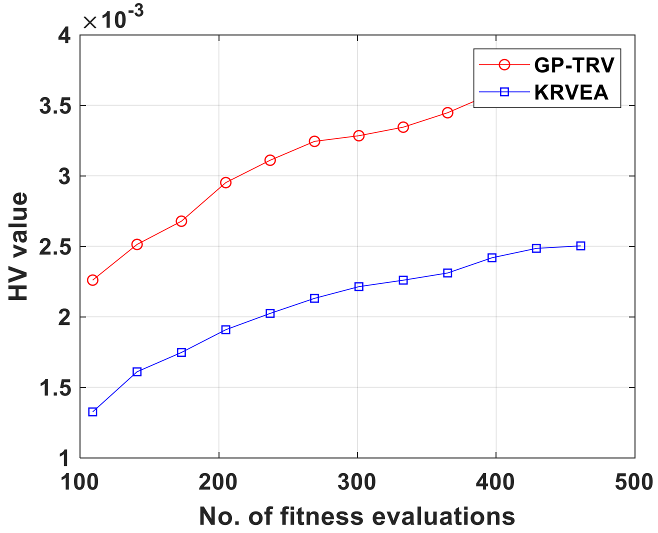
Safe, stable handling and controllability in **all driving situations** up to v_{max}

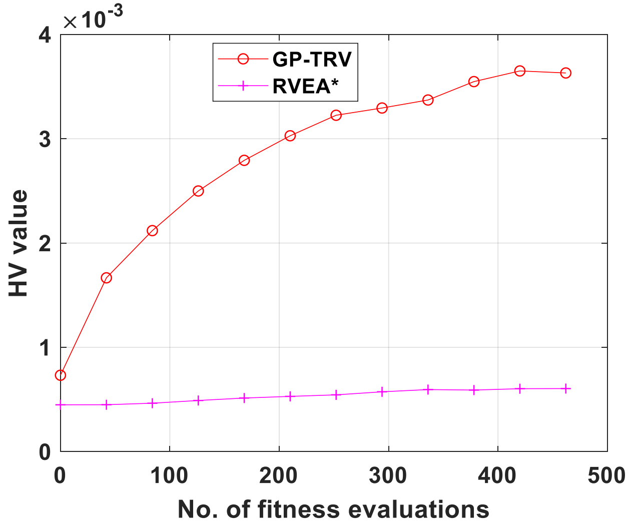
- High level of driving safety including the stability limits
- Sufficient road and vehicle reaction feedback as well as predictable vehicle behaviours
- Steady, stable and comfortable straight-line driving behaviour (e.g. under cross-wind, road surface profile irregularities)
- Comfortable and precise steering which provides good feeling for road condition

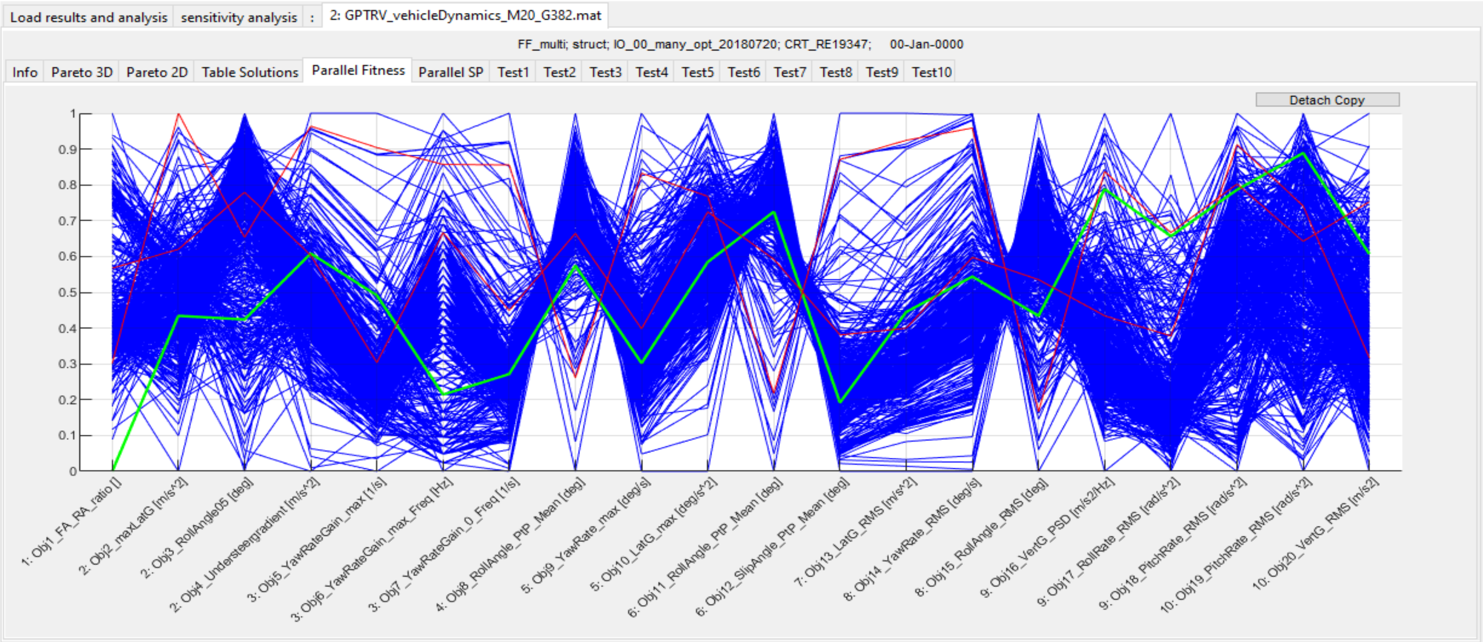


Design variable	symbol	unit
Total mass of car	m_t	[kg]
Roll inertia	I_x	[kgm^2]
Pitch inertia	I_y	[kgm^2]
Yaw inertia	I_z	[kgm^2]
Wheel base	l	[mm]
Distance between c.g. and front axle	$l_{c.g.}$	[m]
Height of c.g. above front axle	$h_{c.g.}$	[m]
Half track width front tires	$w_{ft}*$	[mm]
Half track width rear tires	$w_{rt}*$	[mm]

Comparative Results



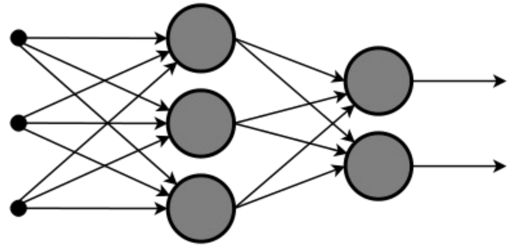




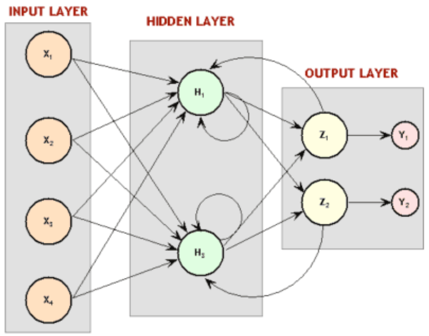
Multi-objective Machine Learning

Basic Artificial Neural Network Models

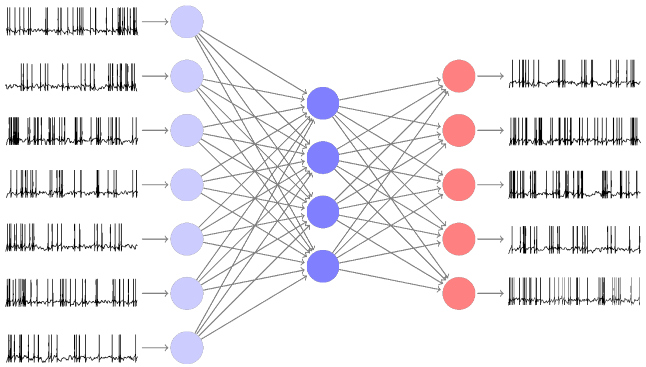
- Feed-forward neural networks
  - Multilayer perceptrons (MLPs)
  - Radial-basis-function networks (RBFNs)
- Recurrent neural networks
- Spiking neural networks
- Reservoir computing
- Other models



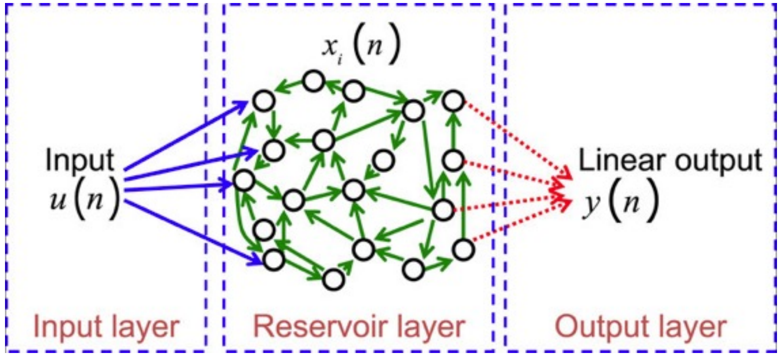
Feed-forward neural networks



Recurrent neural networks



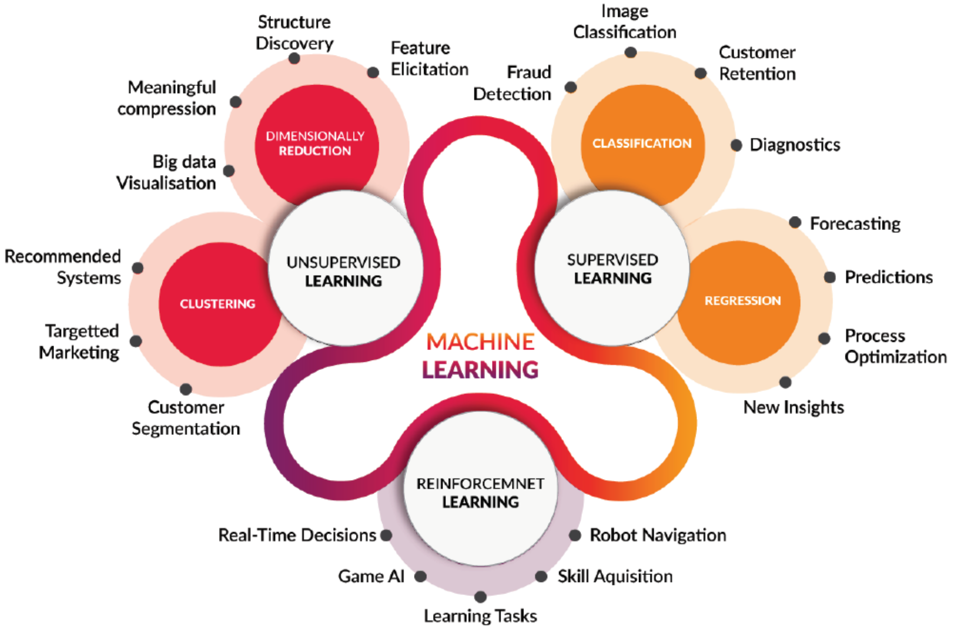
Spiking neural networks



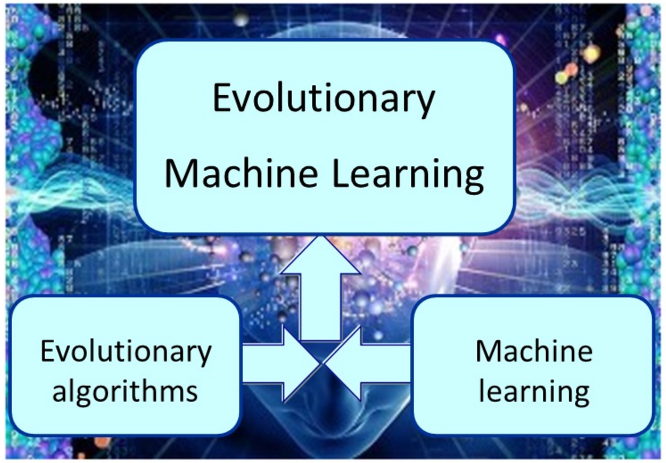
Reservoir computing

Learning Algorithms

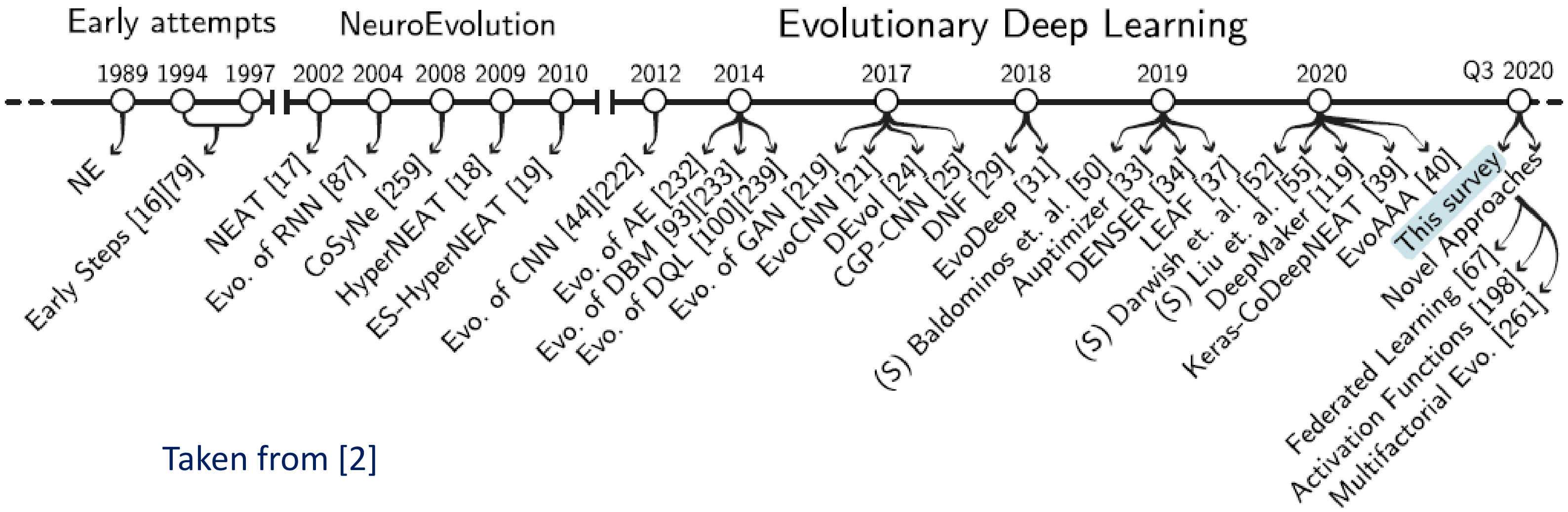
- Supervised learning
  - Need teaching signals (training samples)
  - Often known as function approximation / regression / classification
- Unsupervised learning
  - no teaching signal exists
  - figure out the structure in the observed information, often known as clustering
- Reinforcement learning
- Semi-supervised learning
- Transfer learning / multi-task learning
- Weakly supervised learning
- Self-supervised learning



Evolutionary Machine Learning



- Evolutionary learning is
  - able to solve non-convex learning problems
  - good for both (hyper)-parameter and structure optimization
  - good for multi-objective machine learning
  - good for **automated machine learning**

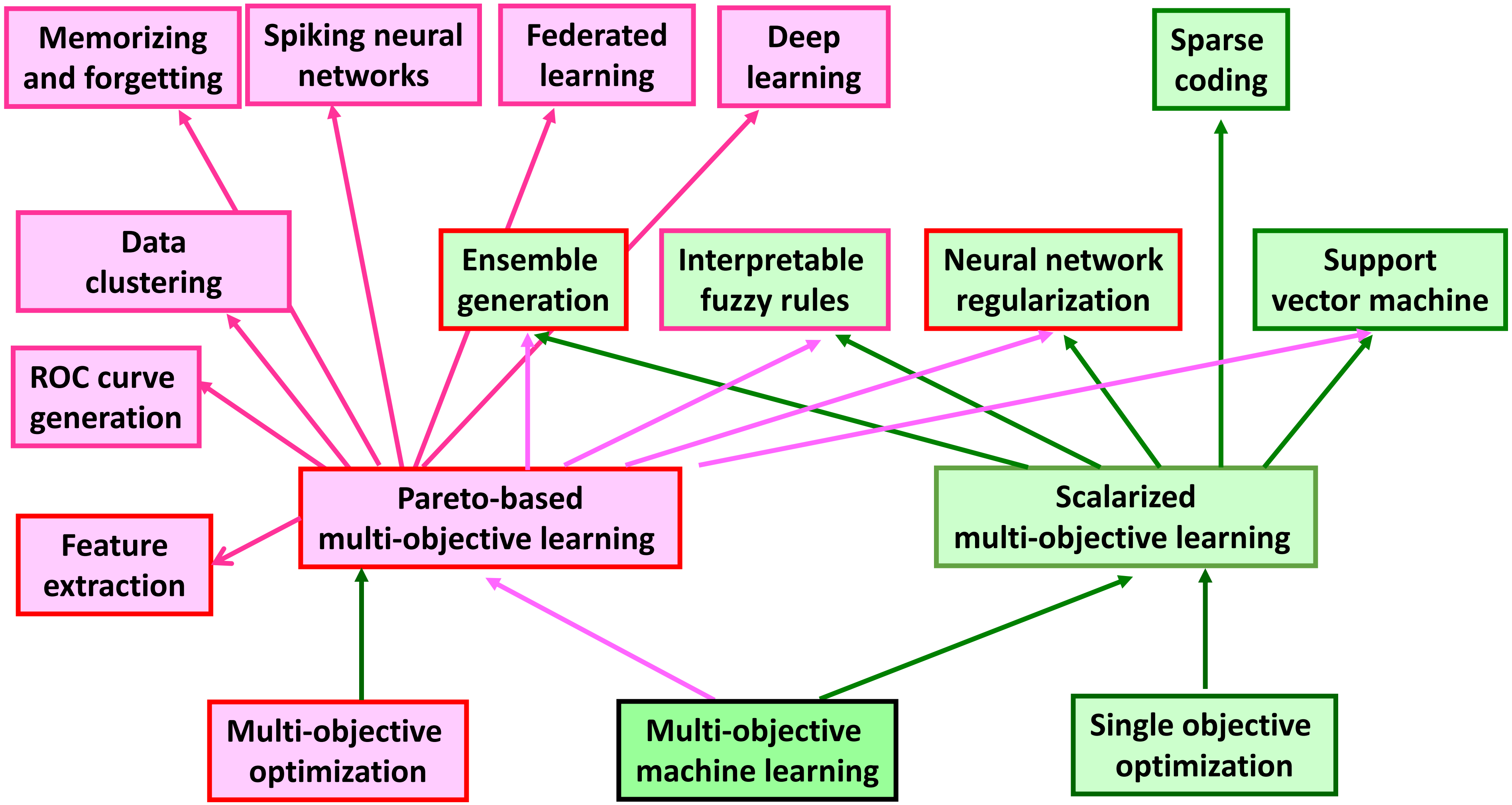


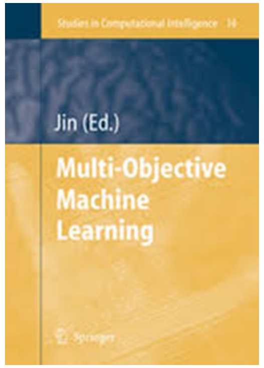
Taken from [2]

[1] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE,* 87 (9):1423-1447, 1999

[2] A.D. Martinez *et al*. Lights and shadows in Evolutionary Deep Learning: Taxonomy, critical methodological analysis, cases of study, learned lessons, recommendations and challenges. *Information Fusion,* 67:161–194, 2021

Evolutionary Multi-Objective Machine Learning





- Y. Jin and B. Sendhoff. Pareto-based multi-objective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(3):397-415, 2008
- Y. Jin (ed.) Multi-objective machine learning. Springer, 2006

Multi-objective Neural Architecture Optimization

- Y. Jin and B. Sendhoff. Pareto-based multi-objective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(3):397-415, 2008
- Y. Jin, B. Sendhoff, and E. Körner. Evolutionary multi-objective optimization for simultaneous generation of signal-type and symbol-type representations. *The Third International Conference on Evolutionary Multi-Criterion Optimization*. LNCS 3410, pp.752-766, Springer, Guanajuato, Mexico, March 9-11, 2005

Objectives in Supervised Machine Learning

西湖大學 WESTLAKE UNIVERSITY

- Tradeoff between accuracy and complexity is inherent to machine learning

$$\underbrace{AIC = -2\log(L(\theta|y, g))}_{\textbf{Accuracy}} + \underbrace{2K}_{\textbf{Complexity}}$$

- Different objectives in supervised learning
  - ➢ minimizing more than one error function
    - ❖ mean squared error
    - ❖ mean absolute error
  - ➢ minimize model complexity
    - ❖ number of hidden nodes
    - ❖ number of connections
  - ➢ maximize diversity for ensemble generation
    - ❖ structural diversity
    - ❖ functional diversity
  - ➢ maximize interpretability for interpretable rule extraction
    - ❖ number of rules / rule length
    - ❖ overlap in rules / fuzzy partition

Learning with Regularization

- A complexity term is included in the cost function

$$J = E + \lambda \Omega$$

E -- Error function, Ω -- complexity

λ -- hyper-parameter

  - Need to predefine a proper hyper-parameter

- Gaussian and Laplacian regularizers

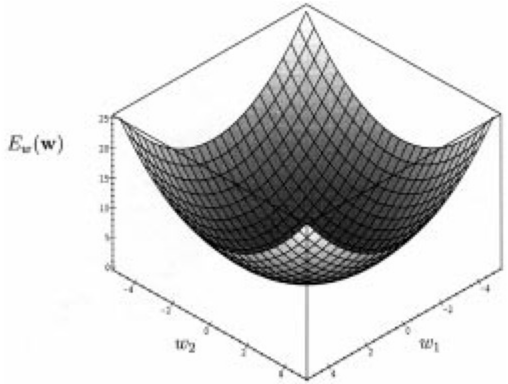
- Gaussian regularizer

$$\Omega = \sum_{i=1}^{M} w_i^2,$$

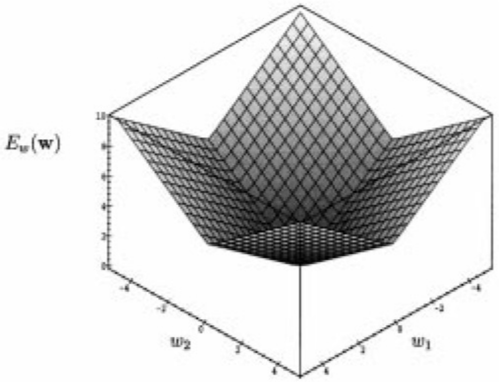
- Laplacian regularizer

$$\Omega = \sum_{i=1}^{M} |w_i|$$

  - Laplacian regularizer is believed to be more effective in reducing complexity



Gaussian



Laplacian

Pareto-based Regularization

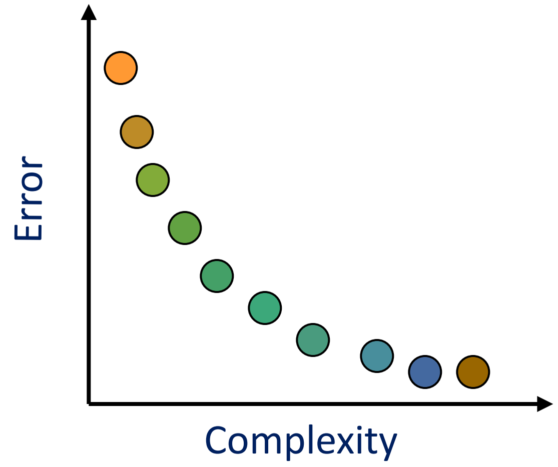
$$\min \{f_1, f_2\}$$

$$f_1 = E;$$

$$f_2 = \Omega.$$

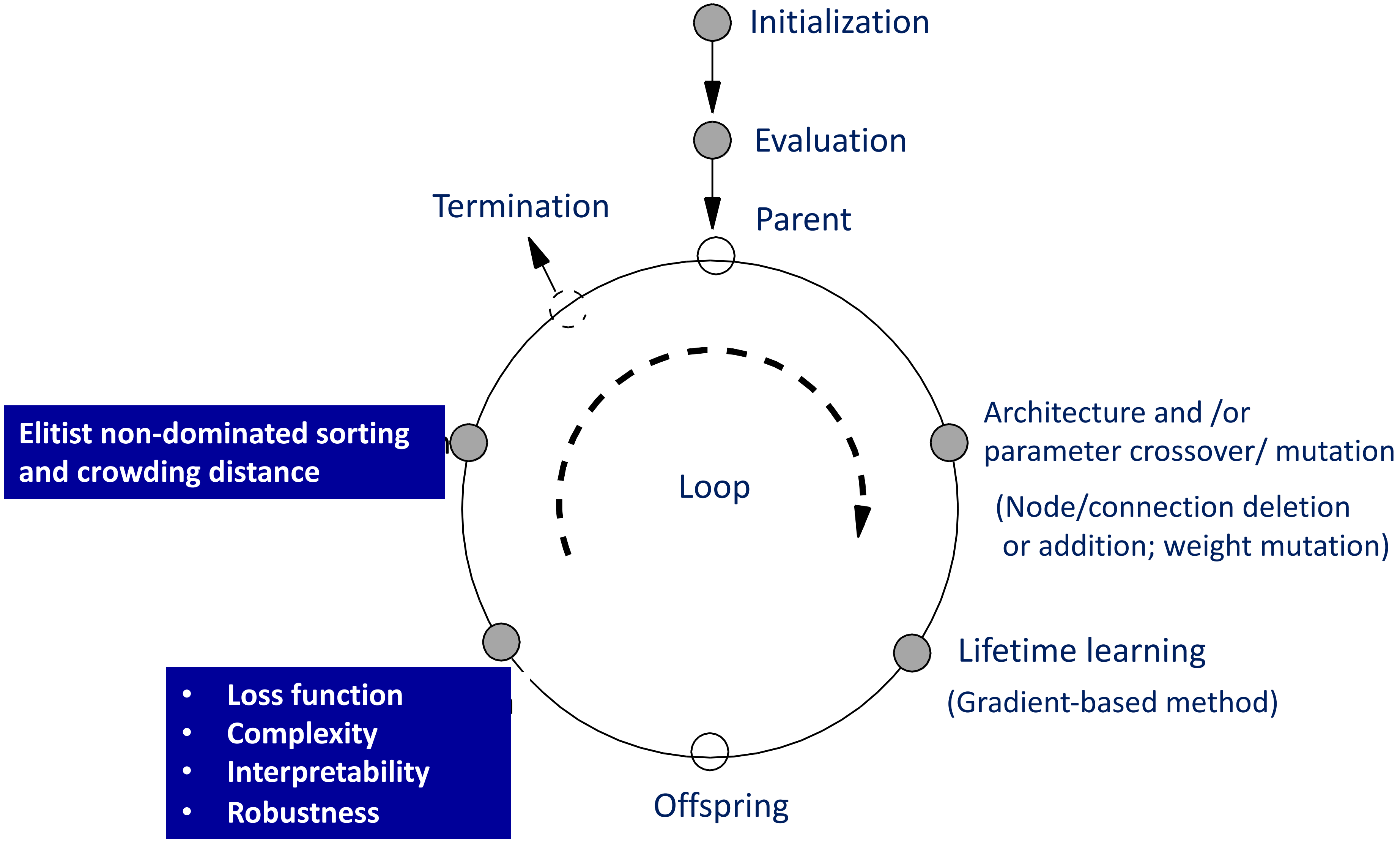
E: approximation error, Ω: complexity

- Gaussian regularizer
- Laplace regularizer
- number of connections / neurons



- Instead of a single model, multiple models with a spectrum of complexity can be obtained simultaneously

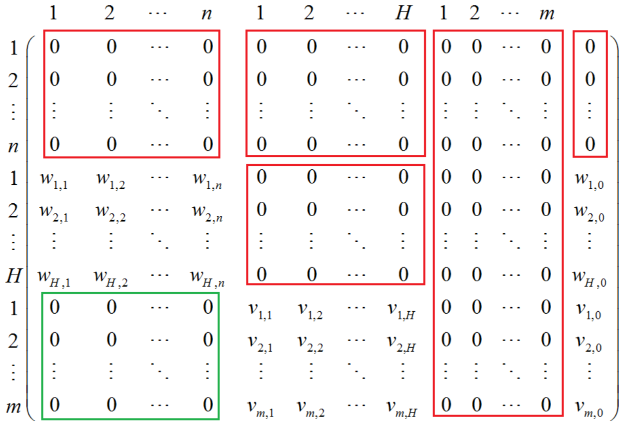
Single- and Multi-objective Evolutionary Learning



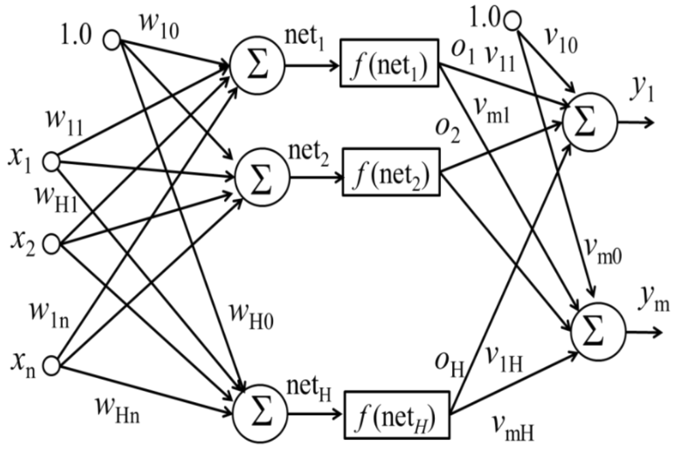
Y. Jin, B. Sendhoff. Pareto-based multi-objective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(3):397-415, 2008

Structure Optimization - Direct Representation

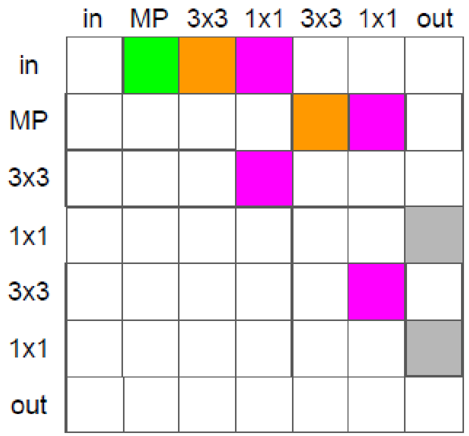
- Direct architecture and weight representation
  - a connection matrix
  - a weight matrix
- Poor scalability in particular for deep neural networks

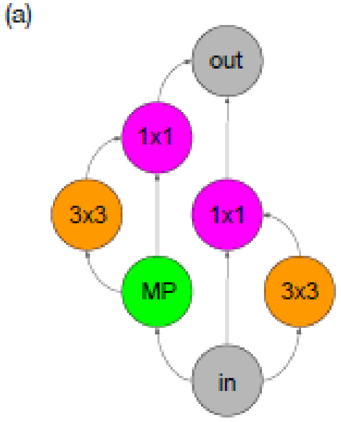


Connection matrix



Network architecture





Analyses of Pareto Front

- By analyzing the "accuracy-complexity" Pareto front, we are able to gain deeper insights into the learning problem

  - Identify Pareto-optimal solutions of low complexity from which interpretable rules can be extracted

  - Identify networks that are able to generalize on unseen data

  - Identify well-performed networks with diverse structures for building ensembles

Y. Jin, R. Grunar, and B. Sendhoff. Pareto analysis of evolutionary and learning systems. *Frontiers of Computer Science in China*, 3(1):4-17, 2009

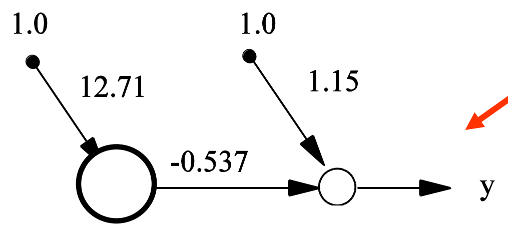
Simple Models Are Explainable

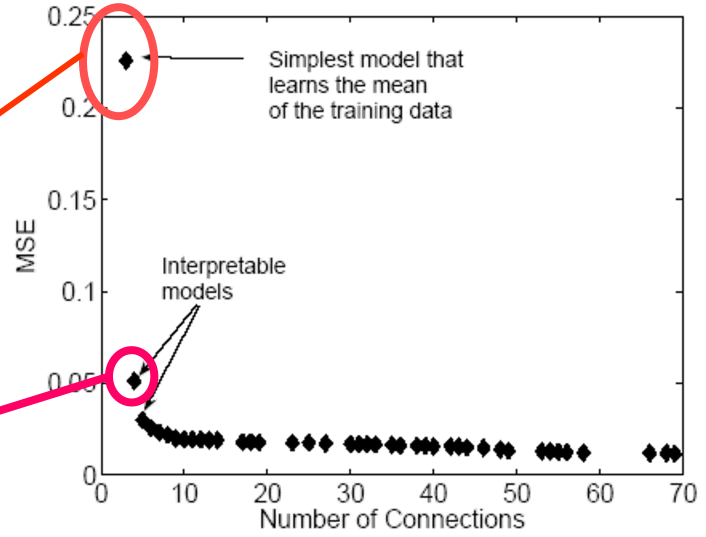
Rule Extraction Example 1: BCD Data

- Breast cancer diagnosis (BCD) data, nine attributes, two classes (benign, malignant)

- Simplest Pareto-optimal: No input feature is chosen, 3 connections

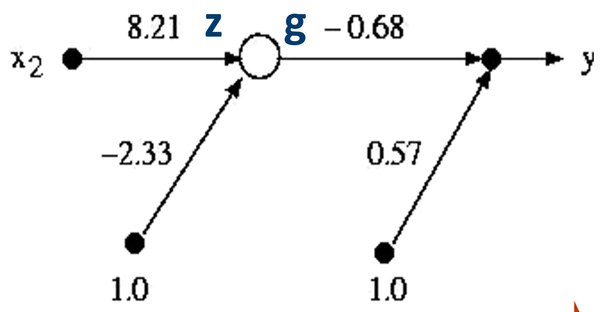
$y = 0.65$





BCD data

- Pareto-optimal NN1: only 1 input feature (x_2) is chosen, 4 connections



If $y < 0.25$, then benign
If $y > 0.75$, then malignant

$-0.68g + 0.57 < 0.25 \rightarrow$ benign
$-0.68g + 0.57 > 0.75 \rightarrow$ malignant

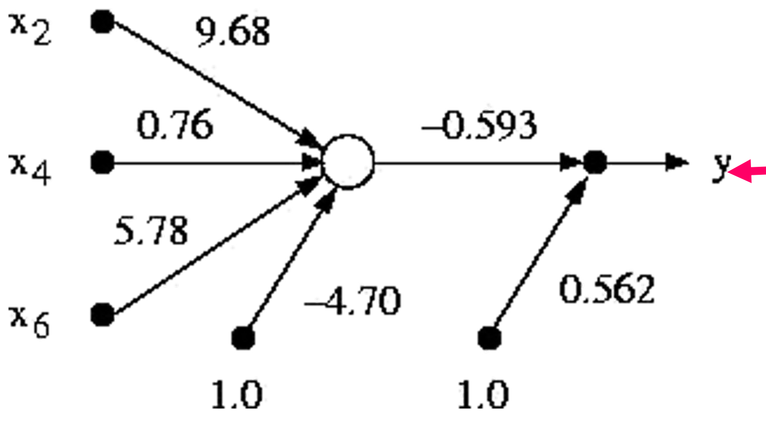
$g = z / (1 + |z|)$
$z = 8.21\, x_2 - 2.33$

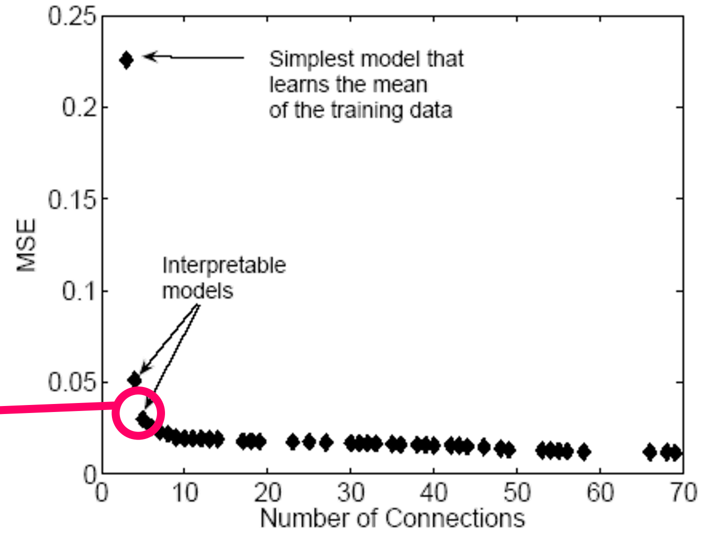
R1: If $x_2 \geq 0.5$, then malignant;
R2: If $x_2 \leq 0.2$, then benign

Rule Extraction Example 1: BCD Data

Pareto-optimal NN2:

- 6 connections
- 3 input features (x2, x4 and x6) are chosen





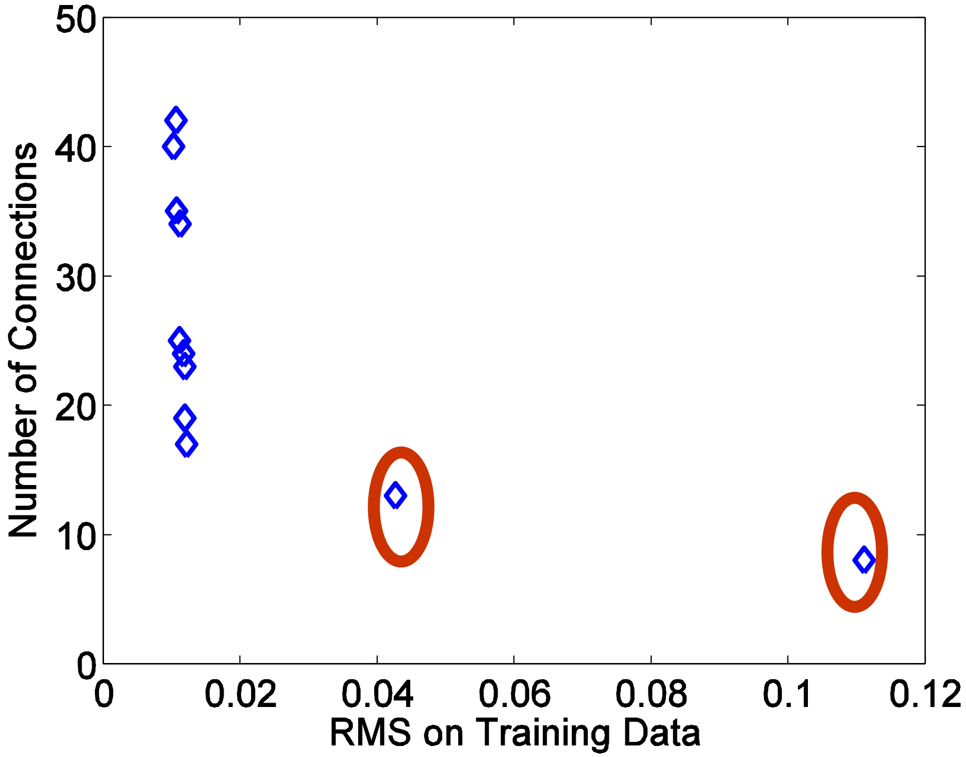
R1: If $x_2 \geq 0.6 \vee x_6 \geq 0.9 \vee x_2 \geq 0.5 \wedge x_6 \geq 0.2 \vee$
$x_2 \geq 0.4 \wedge x_6 \geq 0.4 \vee x_2 \geq 0.3 \wedge x_6 \geq 0.5 \vee$
$x_2 \geq 0.2 \wedge x_6 \geq 0.7$, then malignant;

R2: If $x_2 \leq 0.1 \wedge x_6 \leq 0.4 \vee x_2 \leq 0.2 \wedge$
$x_6 \leq 0.2$, then benign

(x_4 is too weak to play any role in the rules)

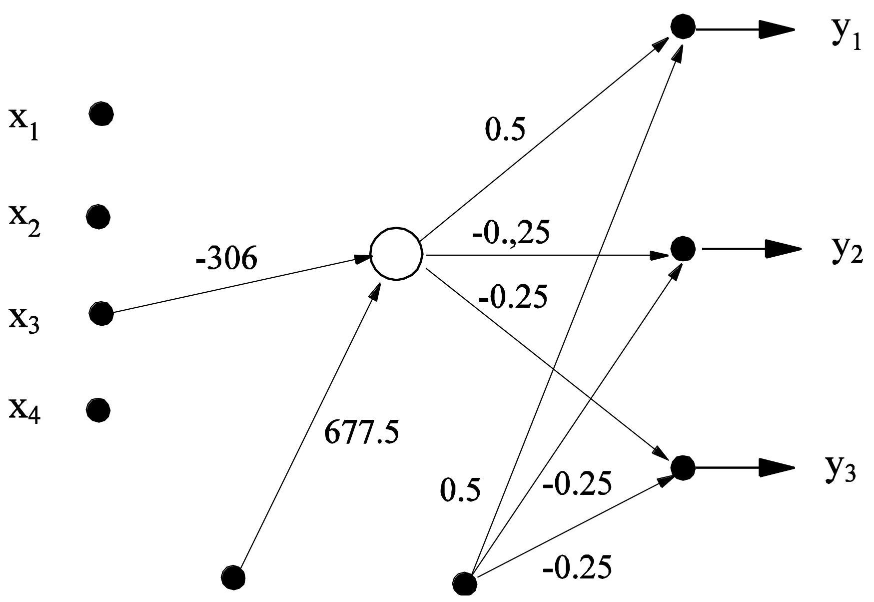
Rule Extraction Example 2: Iris Data

- 4 attributes (Sepal-length, Sepal-width, Petal-length, and Petal-width)
- 3 classes (Iris-Setosa, Iris-Versicolor, and Iris-Virginica)
- 150 data samples in total, 50 for each class (40 of which for each class are used in training)
- 11 networks are obtained

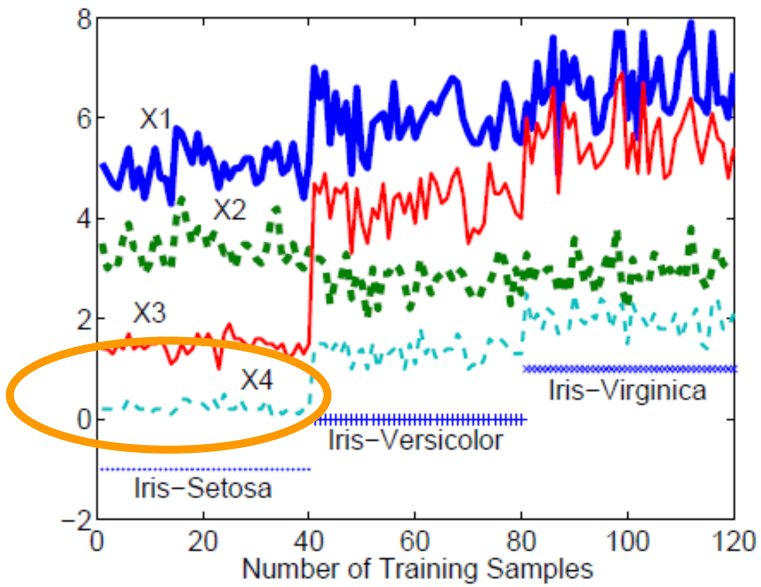


Example 2: The Simplest Network

- Only one attribute is chosen
- Class 1 can be separated form others, but not class 2 and class 3
- 8 connections

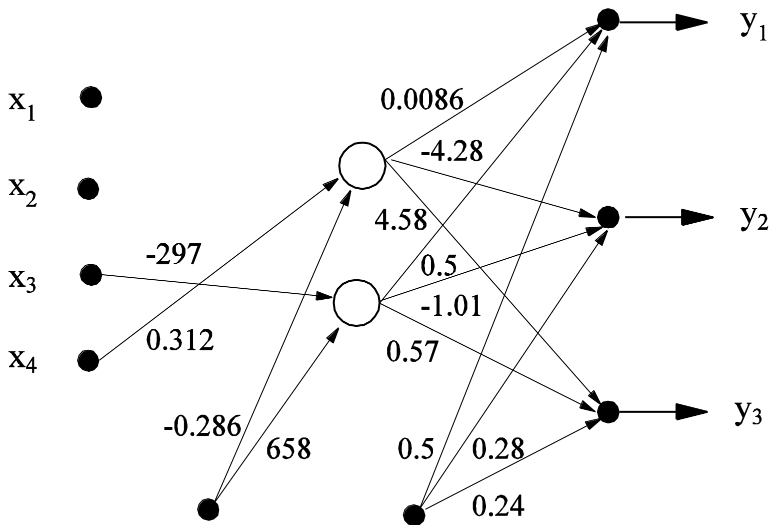


Rule: If $x_3 < 2.2$, Then Iris-Setosa



Example 2: Second Simplest Network

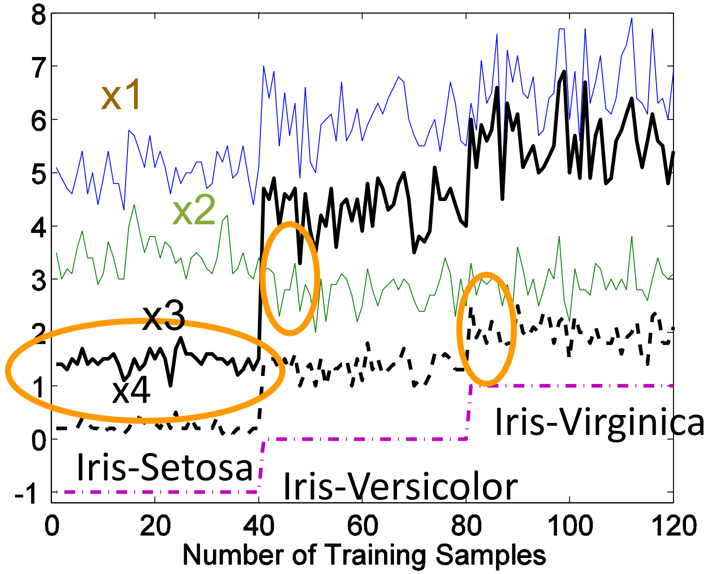
- Only two attributes are chosen
- All classes can be separated correctly



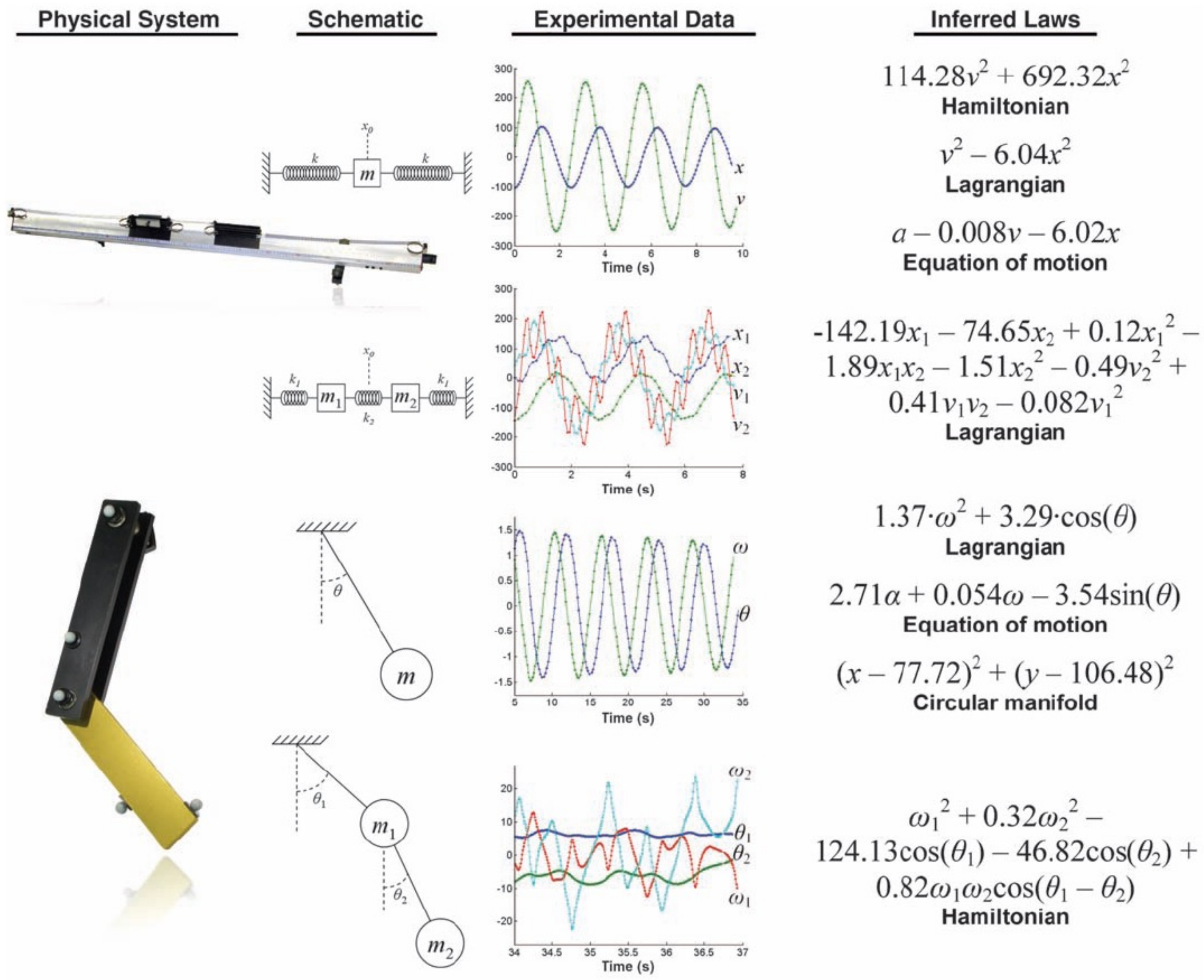
R1: If $x_3 < 2.2 \wedge x_4 < 1.0$,
Then Iris-Setosa;

R2: If $x_3 > 2.2 \wedge x_4 < 1.4$,
Then Iris-Versicolor;

R3: If $x_4 > 1.8$, Then Iris-Virginica,



A Full-Length Science Paper



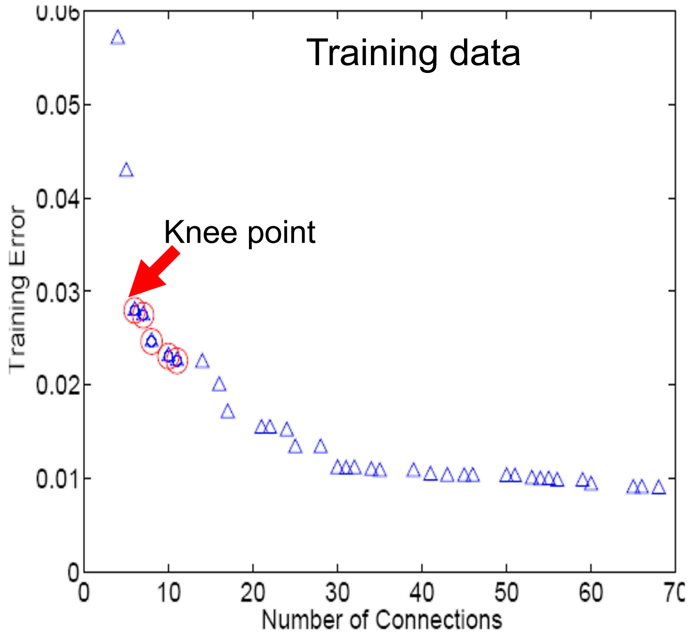
Schmidt M., Lipson H. (2009) "Distilling Free-Form Natural Laws from Experimental Data," *Science*, Vol. 324, no. 5923, pp. 81 - 85

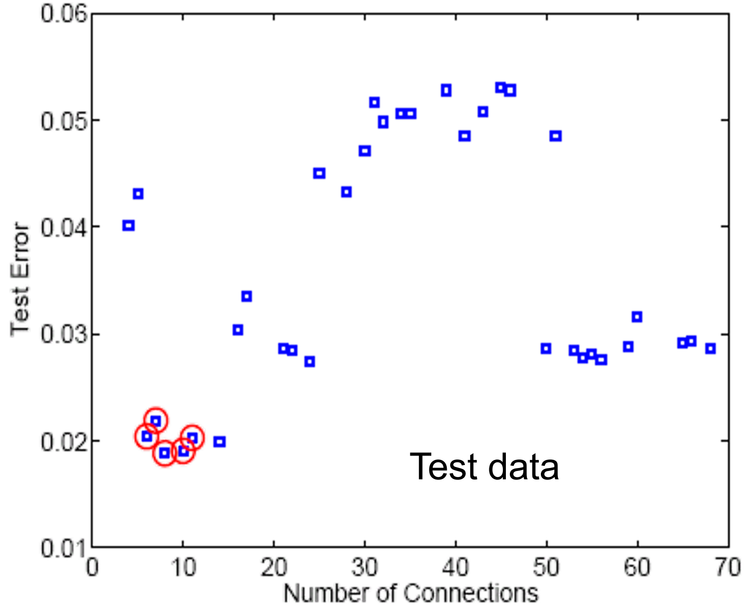
Selection of Generalizable Models

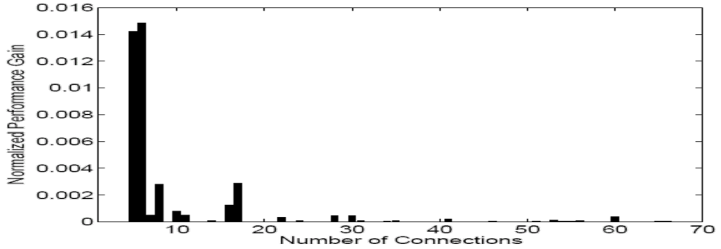
Identifying Networks That Can Generalize

- The complexity that matches the data is the one that reaches maximal normalized performance gain (NPG):

$$NPG = \frac{MSE_i - MSE_j}{N_i - N_j},$$

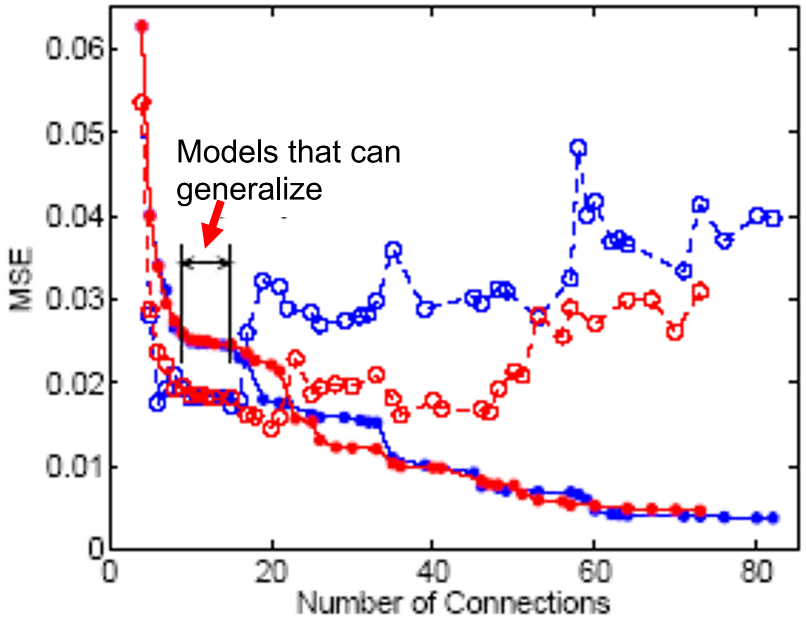




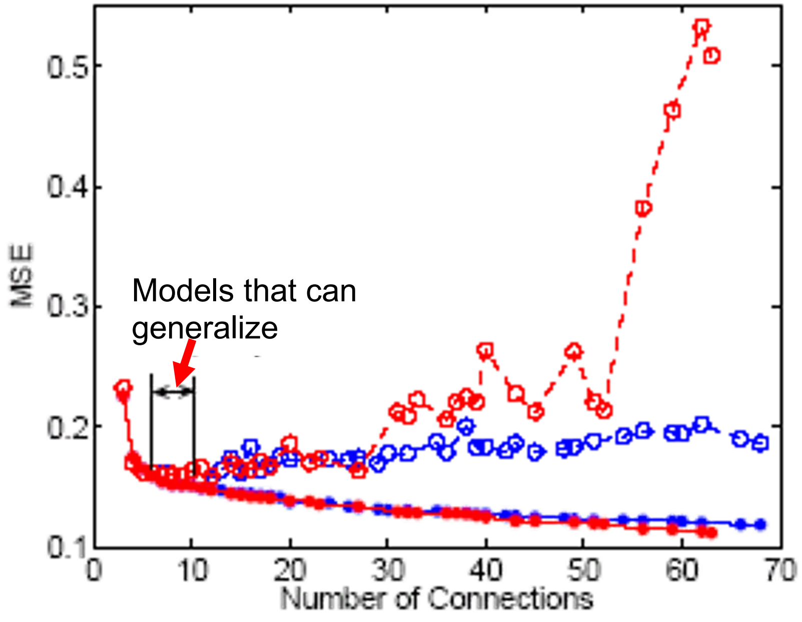


NPG

Identifying Networks That Can Generalize



Breast Cancer Data



Diabetes Data

Run 1:

Training

Run 2:

Test

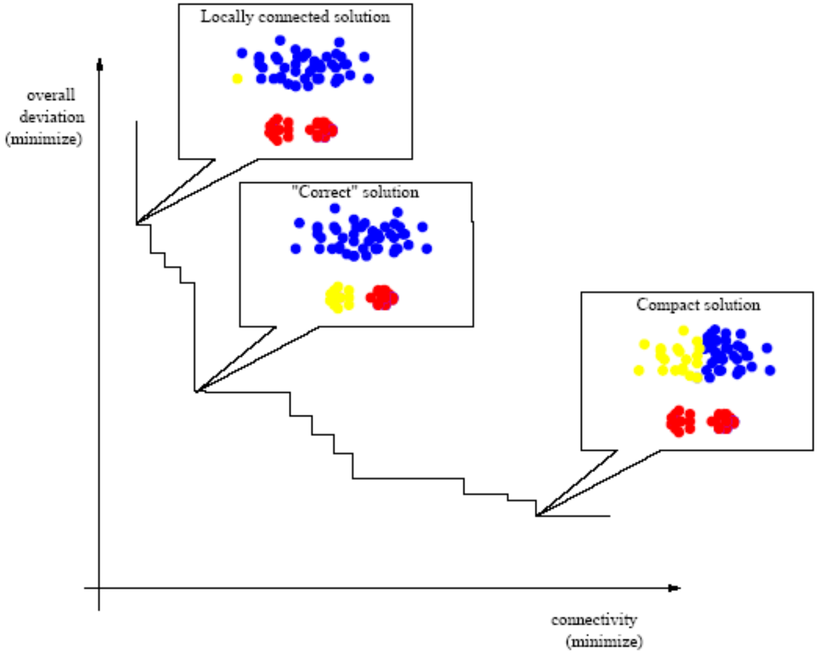
Multi-objective Clustering

J. Handl and J. Knowles. (2005) Exploiting the trade-off -- the benefits of multiple objectives in data clustering. *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005).* Pages 547-560. LNCS 3410

Multi-Objective Clustering - Objectives

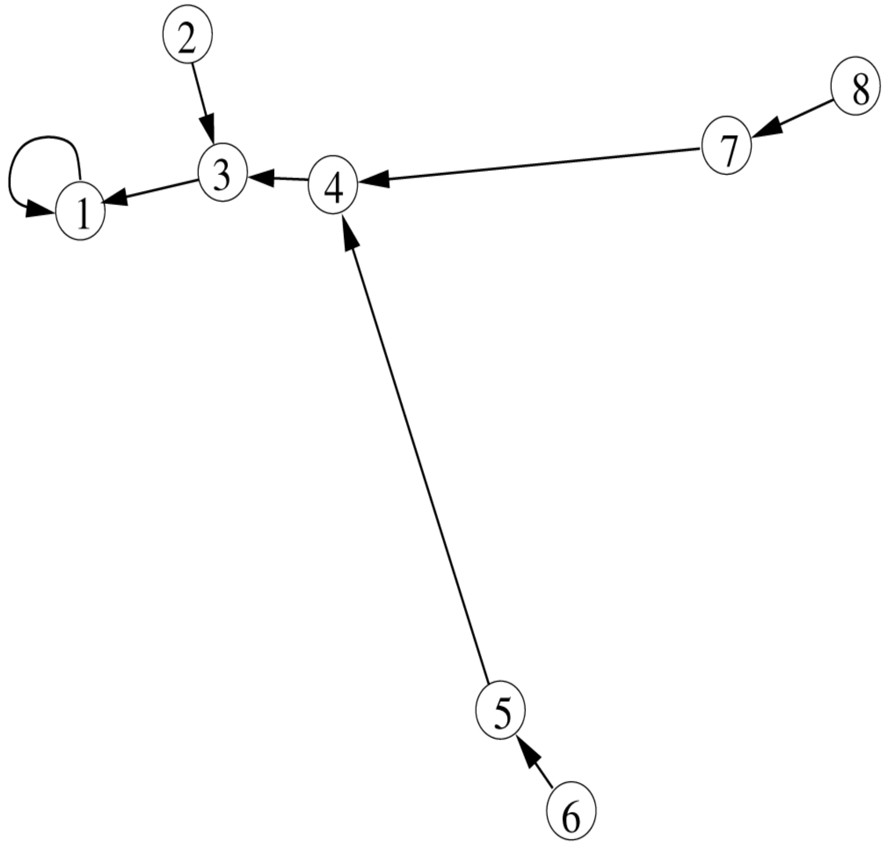
Pareto-based multi-objective clustering has shown to be helpful for determining the number of clusters (Handl and Knowles, 2005)

- Two objectives
  - ➢ Cluster compactness, described by overall deviation
  - ➢ Cluster connectivity, expressed by the degree to which neighboring data points are grouped in the same cluster



Multiobjective Clustering - Coding

Coding: locus-based adjacency scheme



Order of connection:

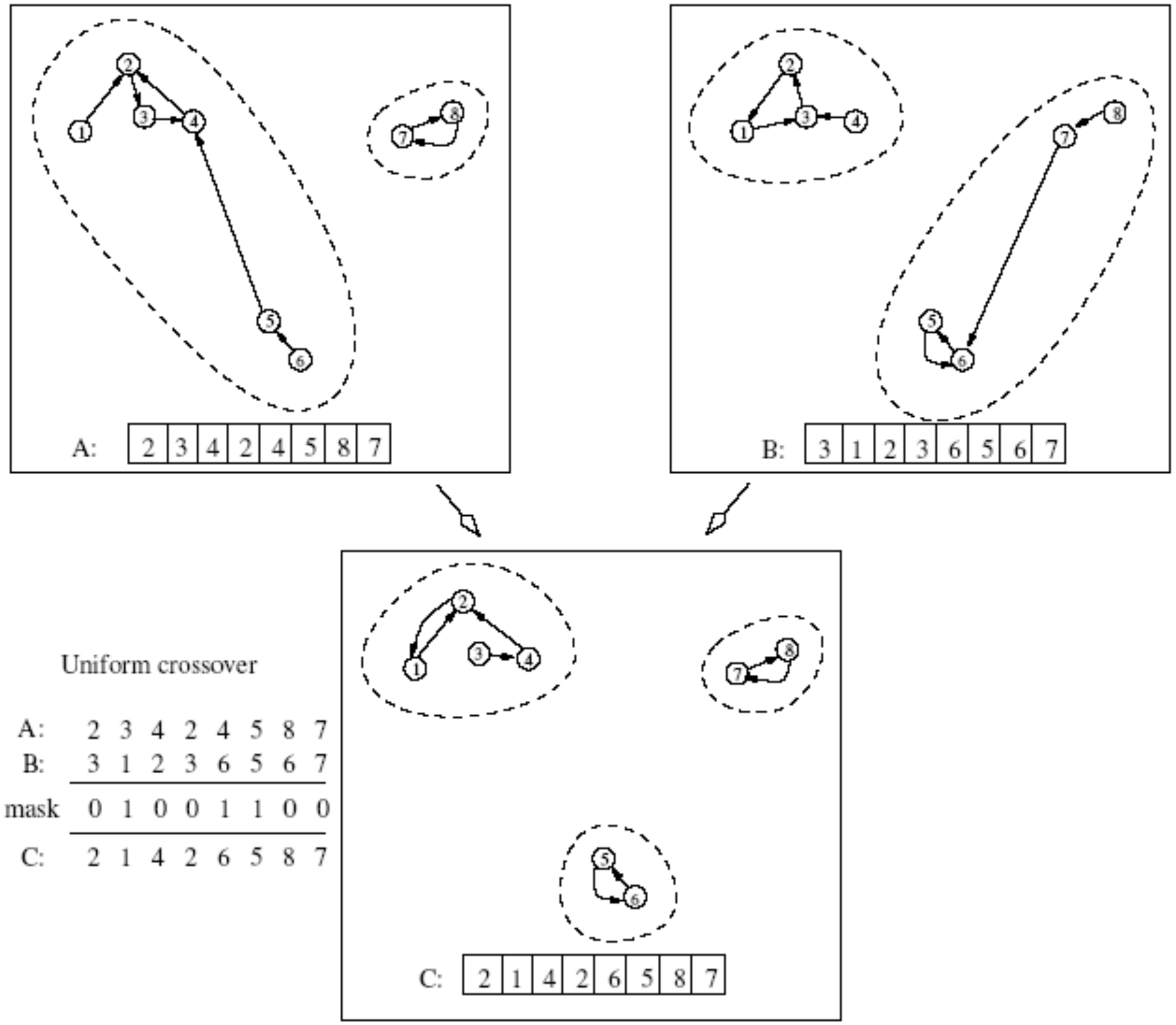
1 to 1
3 to 1
4 to 3
2 to 3
7 to 4
8 to 7
5 to 4
6 to 5

Genotype:

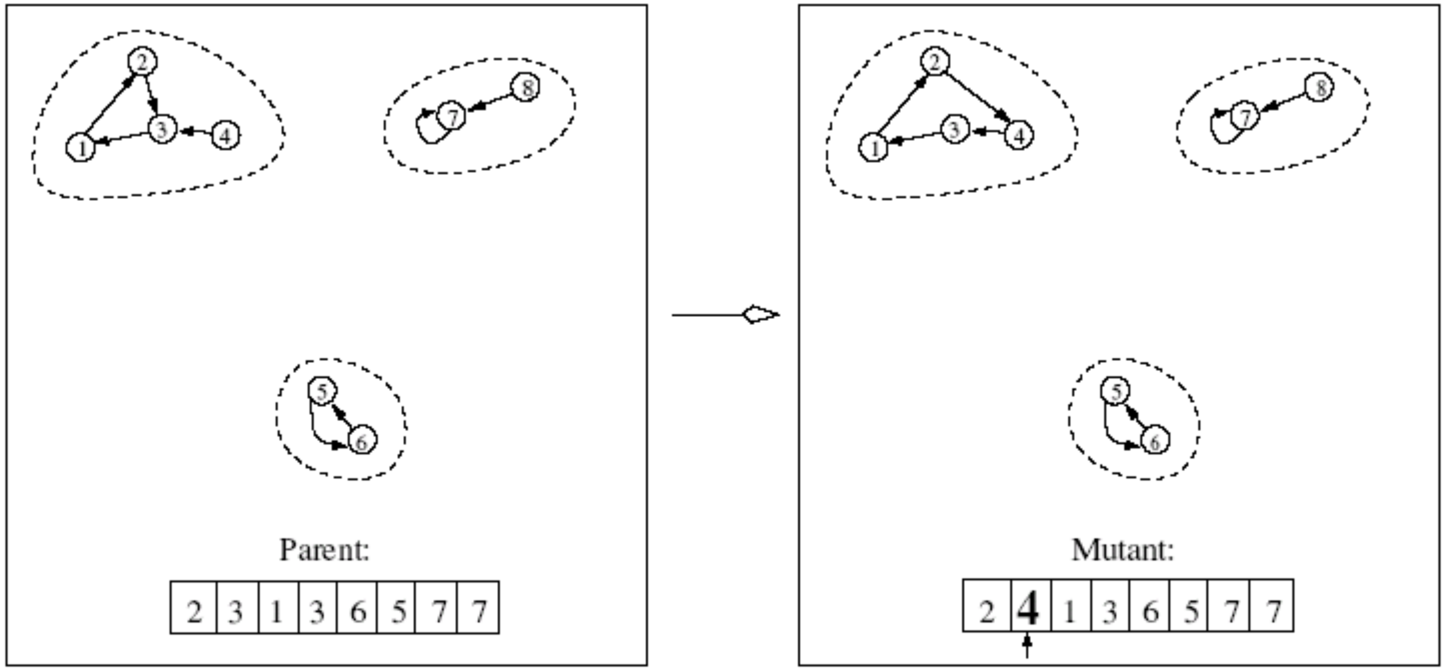
1	3	1	3	4	5	4	7

Sub-graphs need to be detected and data items in the same sub-graph are grouped in the same cluster

Multiobjective Clustering - Crossover

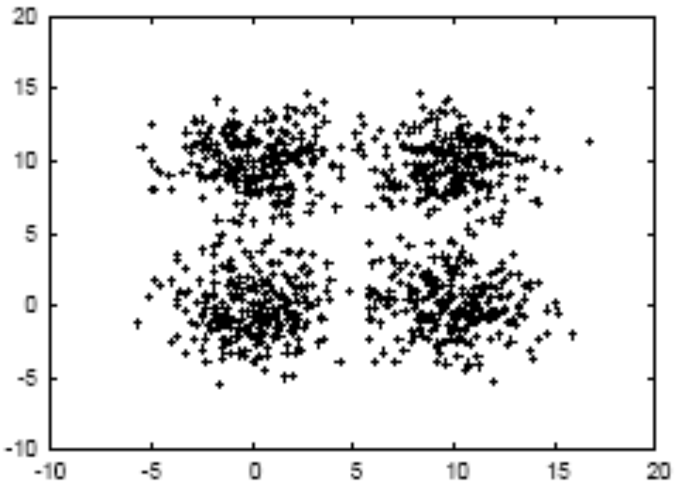


Multiobjective Clustering - Mutation

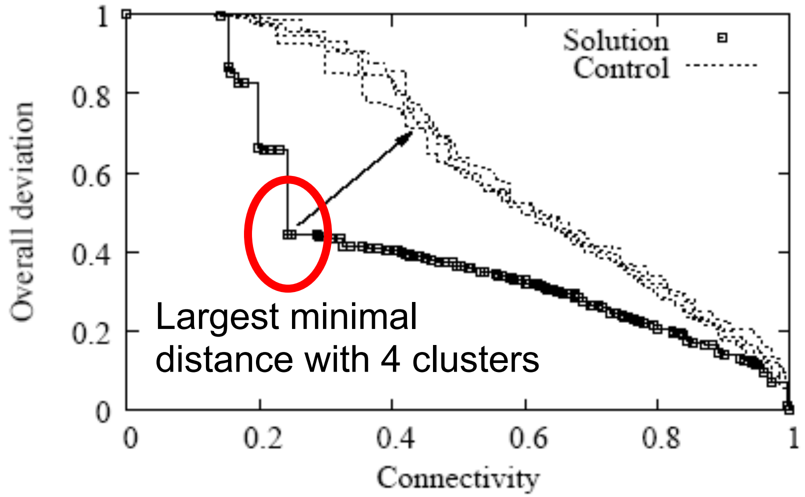


Multi-Objective Clustering - Example

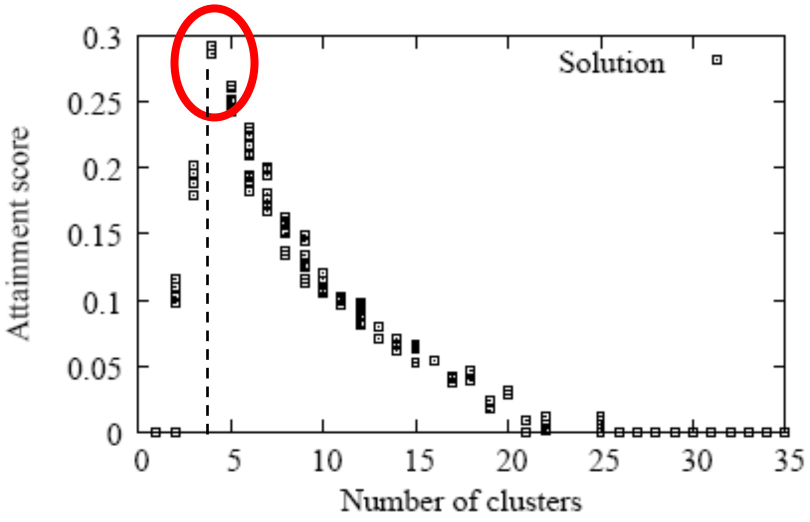
- Data set: *Square1*



- Analysis of Pareto-optimal solutions
  - Calculate attainment score (maximum distance between solution and control)



Connectivity deviation tradeoff



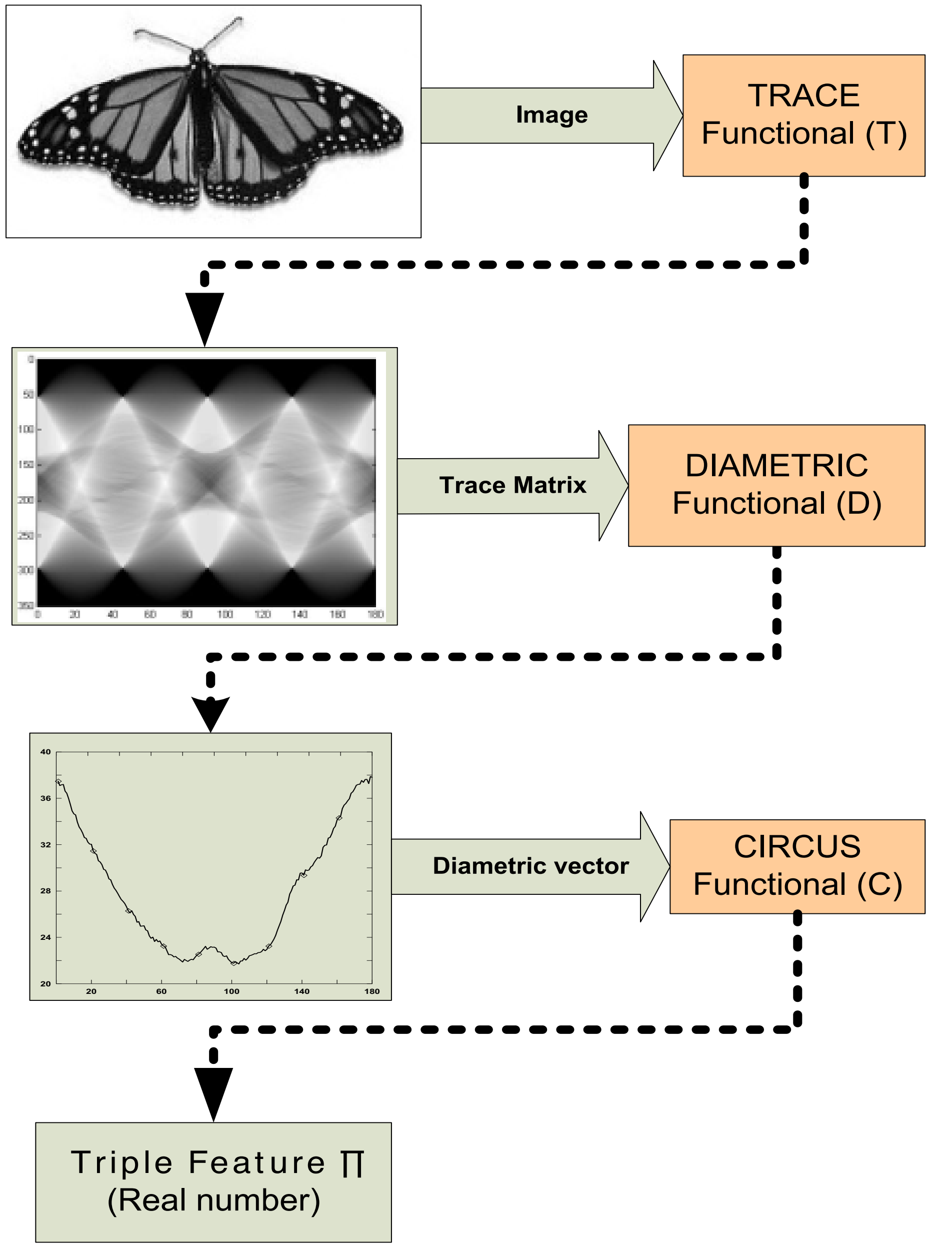
Attainment score of the Pareto-optimal solutions

Multi-Objective Feature Extraction

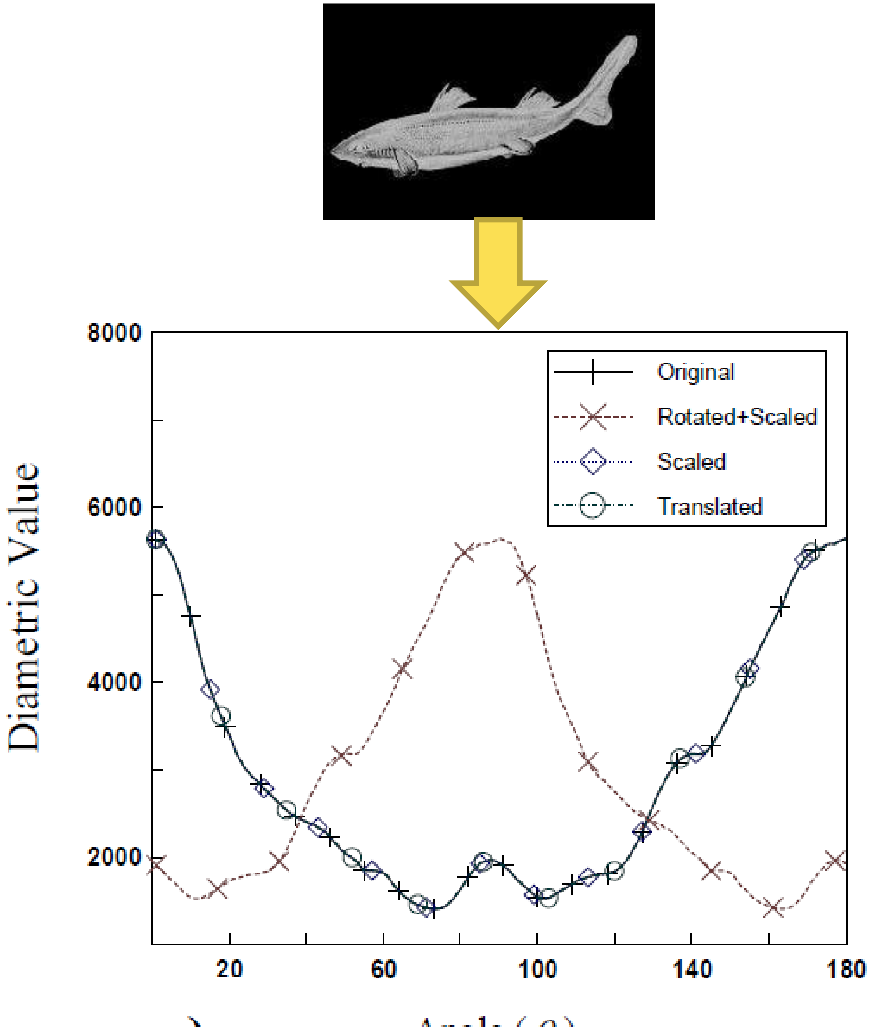
W. A. Albukhanajer, J. A. Briffa and Y. Jin. Evolutionary multi-objective image feature extraction in the presence of noise. *IEEE Transactions on Cybernetics*, 45(9):1757-1768, 2015.

W. A. Albukhanajer, Y. Jin, J. A. Briffa. Classifier ensembles for image identification using multi-objective Pareto features. *Neurocomputing,* 238:316-327, 2017.

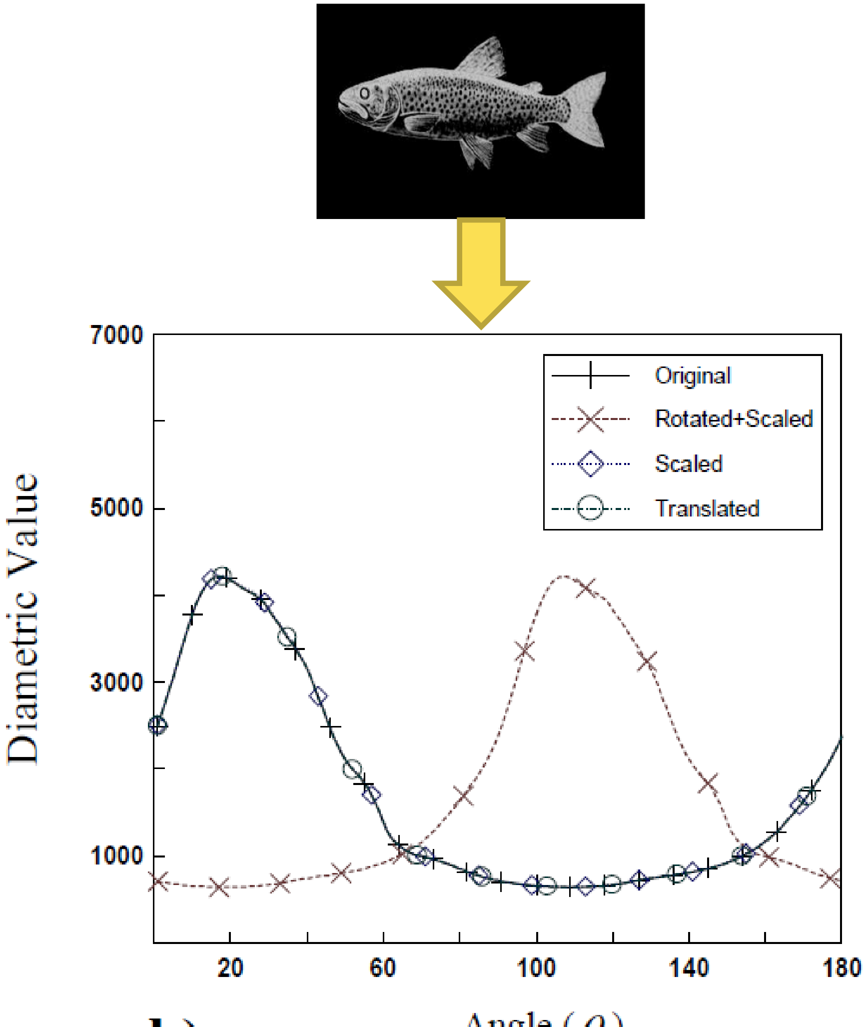
Trace Transform for Feature Extraction



Trace Transform for Feature Extraction



a)



b)

Evolutionary Trace Transform

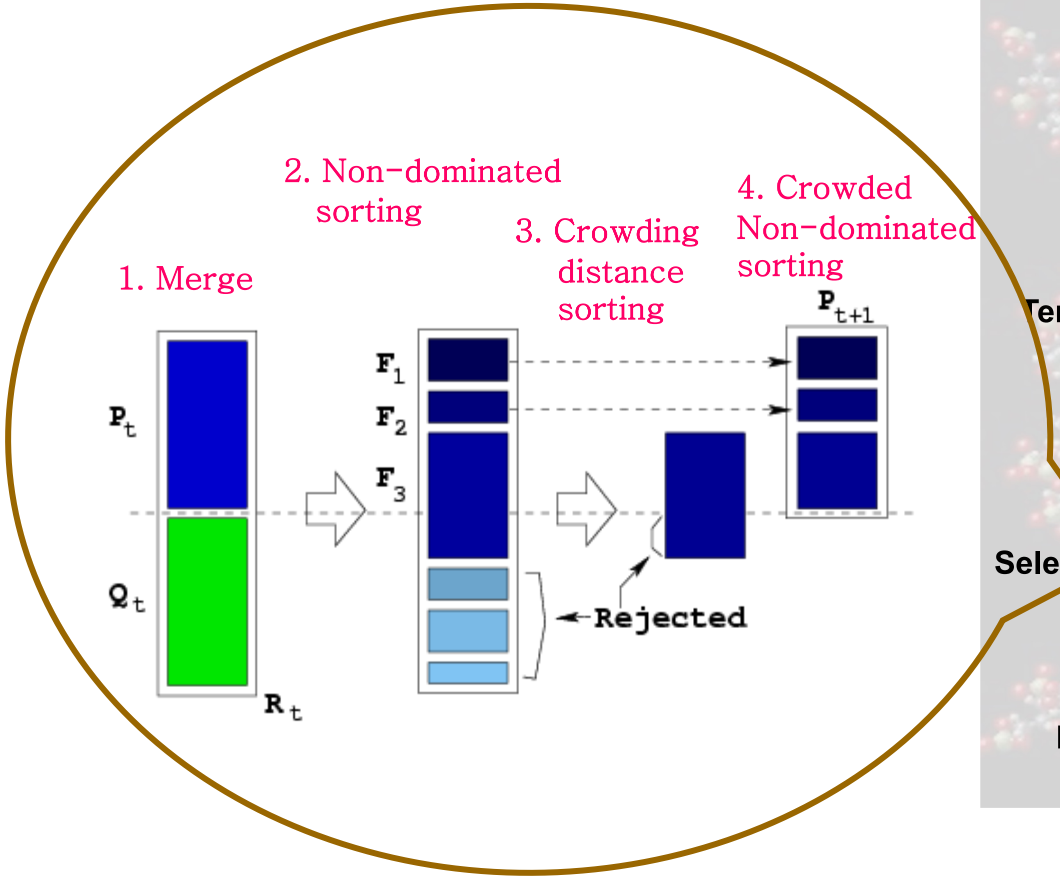
Parameters and functionals can be optimised using an evolutionary multi-objective optimisation algorithm

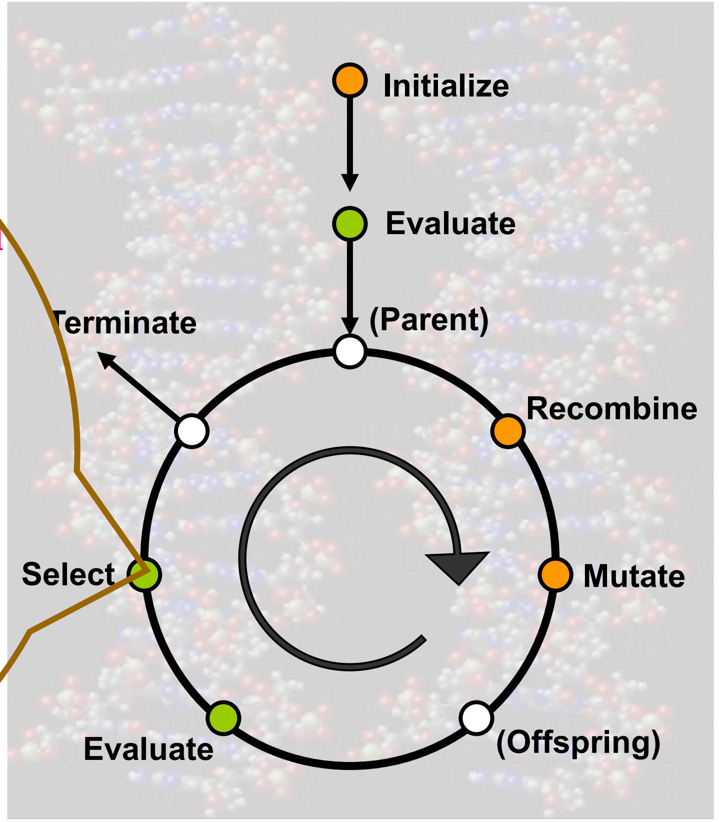
No.	Functional
T_0	$\sum_{i=1}^{n_t} \tau_i$
T_1	$\max_{i=1}^{n_t} \tau_i - \min_{i=1}^{n_t} \tau_i$
T_2	$\left(\sum_{i=1}^{n_t} \lvert\tau_i\rvert^{\frac{1}{2}}\right)^2$
T_3	$\left(\sum_{i=1}^{n_t} \lvert\tau_i\rvert^4\right)^{\frac{1}{4}}$
T_4	$\sum_{i=1}^{n_t} \lvert\tau_i'\rvert$
T_5	$\sqrt{\frac{1}{n_t}\sum_{i=1}^{n_t}(\tau_i - M)^2}, M = \frac{1}{n_t}\sum_{i=1}^{n_t}\tau_i$
T_6	$\sum_{i=1}^{n_t} \sqrt{\lvert\tau_i\rvert}$
T_7	$\max_{i=1}^{n_t} \lvert\tau_i\rvert$
T_8	$\sum_{i=c}^{n_t} (i-c)^2\tau_i,\ c = \frac{\sum_{i=1}^{n_t} i\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i\rvert}$
T_9	$\sum_{i=c}^{n_t} (i)^2\tau_i,\ c = \frac{\sum_{i=1}^{n_t} i\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i\rvert}$
T_{10}	$\sum_{i=c}^{n_t} (i)^3\tau_i,\ c = \frac{\sum_{i=1}^{n_t} i\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i\rvert}$
T_{11}	$\sum_{i=c}^{n_t} (r)^{0.5}\tau_i,\ c = \frac{\sum_{i=1}^{n_t} l\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i\rvert},\ r = \lvert l-c\rvert,\ l = 1,2,...,n_t$
T_{12}	$\sum_{i=c}^{n_t} (r)\tau_i,\ c = \frac{\sum_{i=1}^{n_t} l\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i\rvert},\ r = \lvert l-c\rvert,\ l = 1,2,...,n_t$
T_{13}	$\sum_{i=c}^{n_t} (r)^2\tau_i,\ c = \frac{\sum_{i=1}^{n_t} l\lvert\tau_i\rvert}{\sum_{i=1}^{n_t}\lvert\tau_i)\rvert},\ r = \lvert l-c\rvert,\ l = 1,2,...,n_t$

No.	Functional
D_0	$\sum_{i=1}^{n_p} \delta_i$
D_1	$\max_{i=1}^{n_p} \delta_i$
D_2	$\left(\sum_{i=1}^{n_p} \lvert\delta_i\rvert^{\frac{1}{2}}\right)^2$
D_3	$\left(\sum_{i=1}^{n_p} \lvert\delta_i\rvert^4\right)^{\frac{1}{4}}$
D_4	$\sqrt{\sum_{i=1}^{n_p} \delta_i^2}$
D_5	$\max_{i=1}^{n_p} \delta_i - \min_{i=1}^{n_p} \delta_i$
D_6	$\sum_{i=1}^{n_p} \lvert\delta_i'\rvert$
D_7	$\sum_{i=c}^{n_p} (i-c)^2\delta_i,\ c = \frac{\sum_{i=1}^{n_p} i\lvert\delta_i\rvert}{\sum_{i=1}^{n_p}\lvert\delta_i\rvert}$

No.	Functional
C_0	$\sum_{i=1}^{n_\theta} \xi_i$
C_1	$\text{median}_{i=1}^{n_\theta}\xi_i$
C_2	$\sqrt{\frac{1}{n_\theta}\sum_{x=1}^{n_\theta}(\xi_i - M)^2}, M = \frac{1}{n_\theta}\sum_{i=1}^{n_\theta}\xi_i$
C_3	$\sum_{i=1}^{n_\theta} \lvert\xi_i'\rvert$
C_4	$\max_{i=1}^{n_\theta} \xi_i$
C_5	$\max_{i=1}^{n_\theta} \xi_i - \min_{i=1}^{n_\theta} \xi_i$

Multi-objective Optimization Algorithm





Evolutionary algorithm

Elitist non-dominated sorting and selection

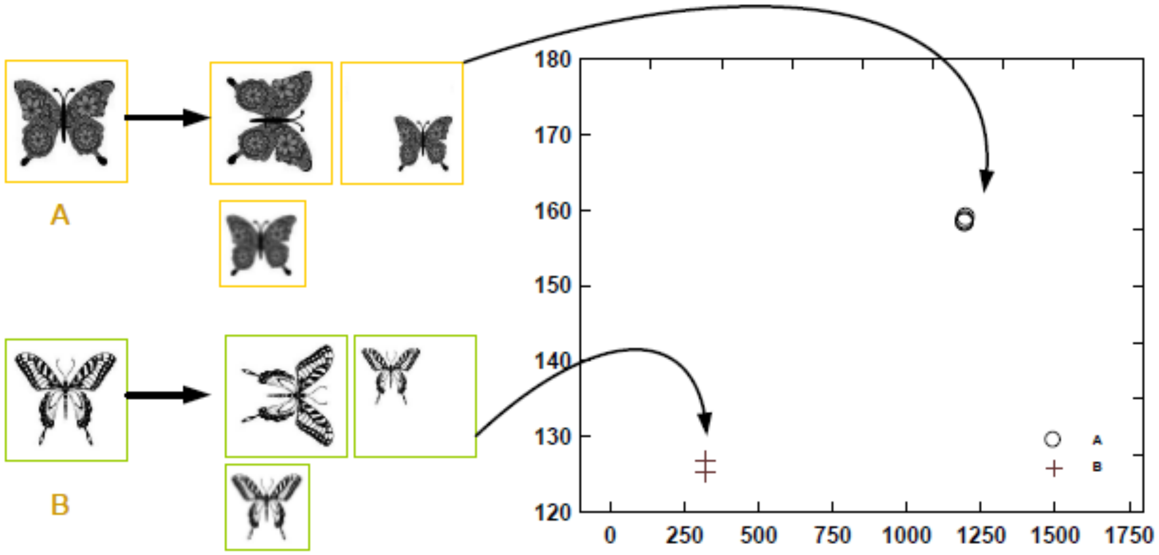
Criteria for Image Feature Extraction

1. Minimize the within-class feature variance (S_w)
2. Maximize the between-class feature scatter (S_b)

$$\min\{f_1, f_2\},$$
$$f_1 = S_w,$$
$$f_2 = \frac{1}{(S_b + \epsilon)},$$

$$S_w = \sum_{k=1}^{C} \sum_{j=1}^{N_k} (x_j^k - \mu_k)^2$$

$$S_b = \sum_{k=1}^{C} (\mu_k - \mu)^2$$



Parameter Encoding

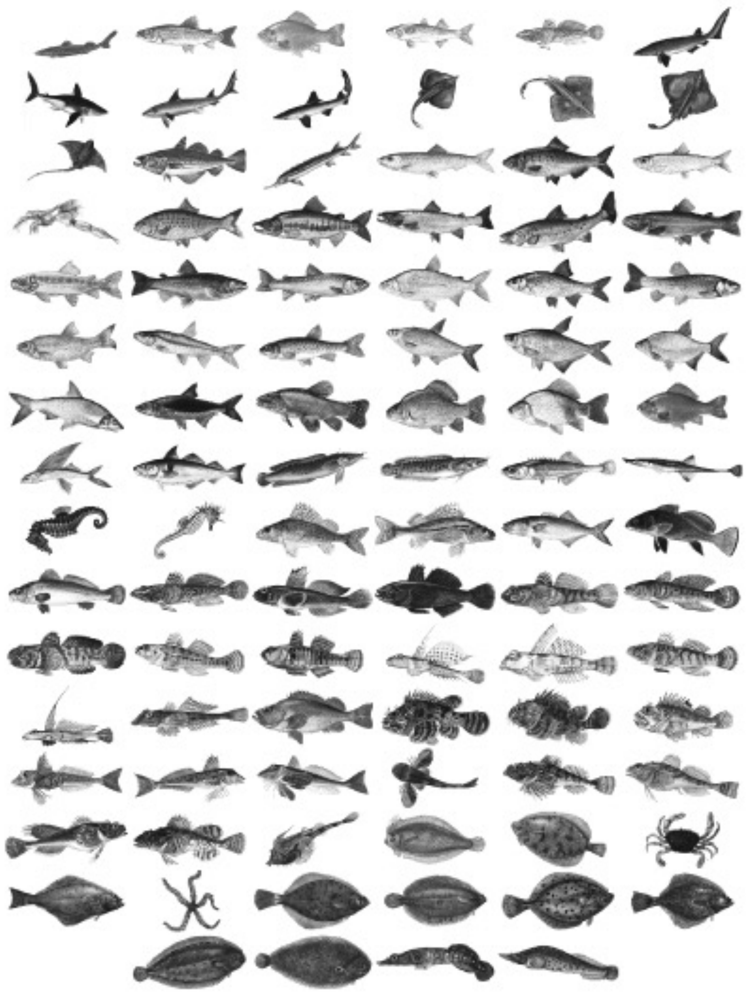
- Single features

T1	D1	C1	θ_1

- Paired features

T1	D1	C1	θ_1	T2	D2	C2	θ_2

Image Database



Fish-94 Database

COIL-20 Database

Robustness to Noise

- How can we evolve Trace transforms that are robust to noise in addition to RST transformations?

(a) 0%. (b) 1%. (c) 3%. (d) 5%. (e) 6%.

Images with salt and pepper noise.

Parameter Setting

西湖大學 WESTLAKE UNIVERSITY

- NSGA-II

Parameter	Value
N_p	150
P_m	0.125
P_c	0.9
Number of generations	200
ϵ	10^{-5}

- Training samples

ETT

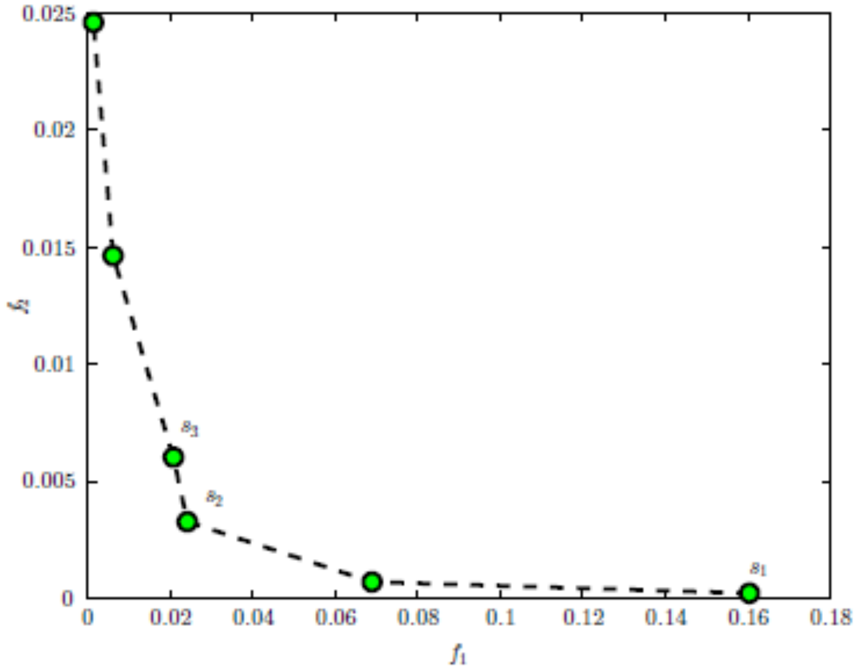
- Sample 1: A low-resolution image (64 x 64) generated from a randomly chosen original image (256x256);
- Sample 2: Random rotation [1°-359°] of Sample 1;
- Sample 3: Random translation of Sample 1 (objects remain within image boundaries);
- Sample 4: Random scale (0.1-0.9) of Sample 1.

ETTN

- Sample 1: A low-resolution image from (64x64) generated from a randomly chosen original image (256x256);
- Sample 2: Random rotation, scale and translation of Sample 1 with Gaussian noise (standard deviation=4);
- Sample 3: Random rotation, scale and translation of Sample 1 with Gaussian noise (standard deviation=6);
- Sample 4: Random rotation of Sample 1;
- Sample 5: Random scale of Sample 1.

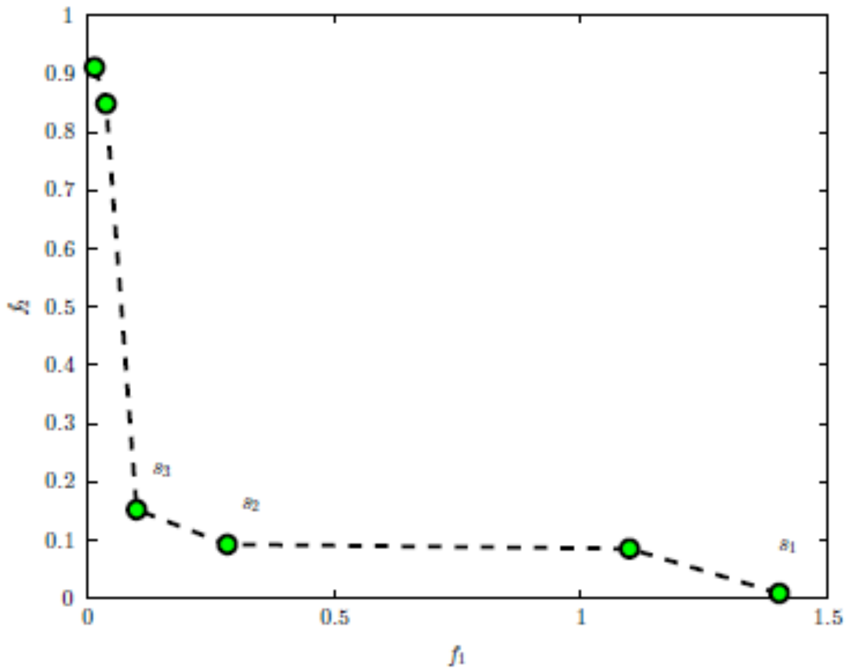
Extracted Pareto Optimal Features

Solution No.	ETT	ETTN
s_1	$T_{12}D_4C_2$	$T_0D_5C_5$
s_2	$T_6D_3C_5$	$T_0D_3C_2$
s_3	$T_0D_3C_1$	$T_0D_1C_2$



(a) ETT.

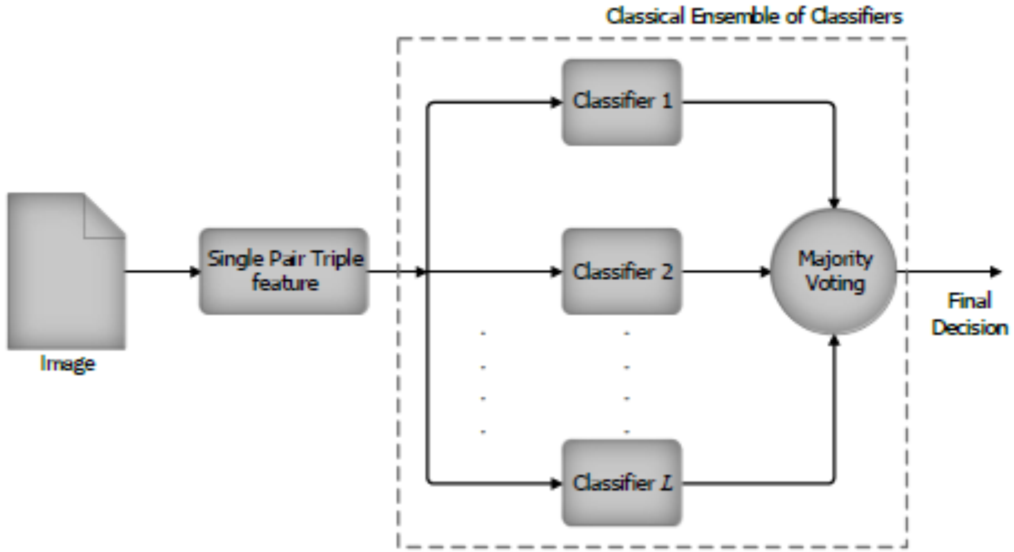
Features extracted by ETT



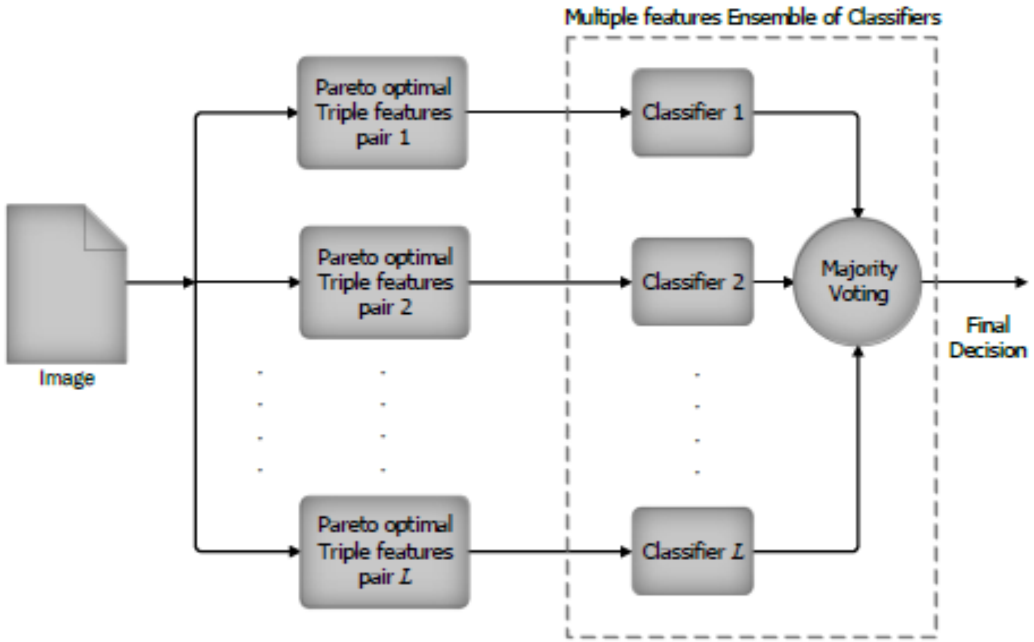
(b) ETTN.

Features extracted by ETTN

Ensemble with Pareto Optimal Features



Traditional ensembles

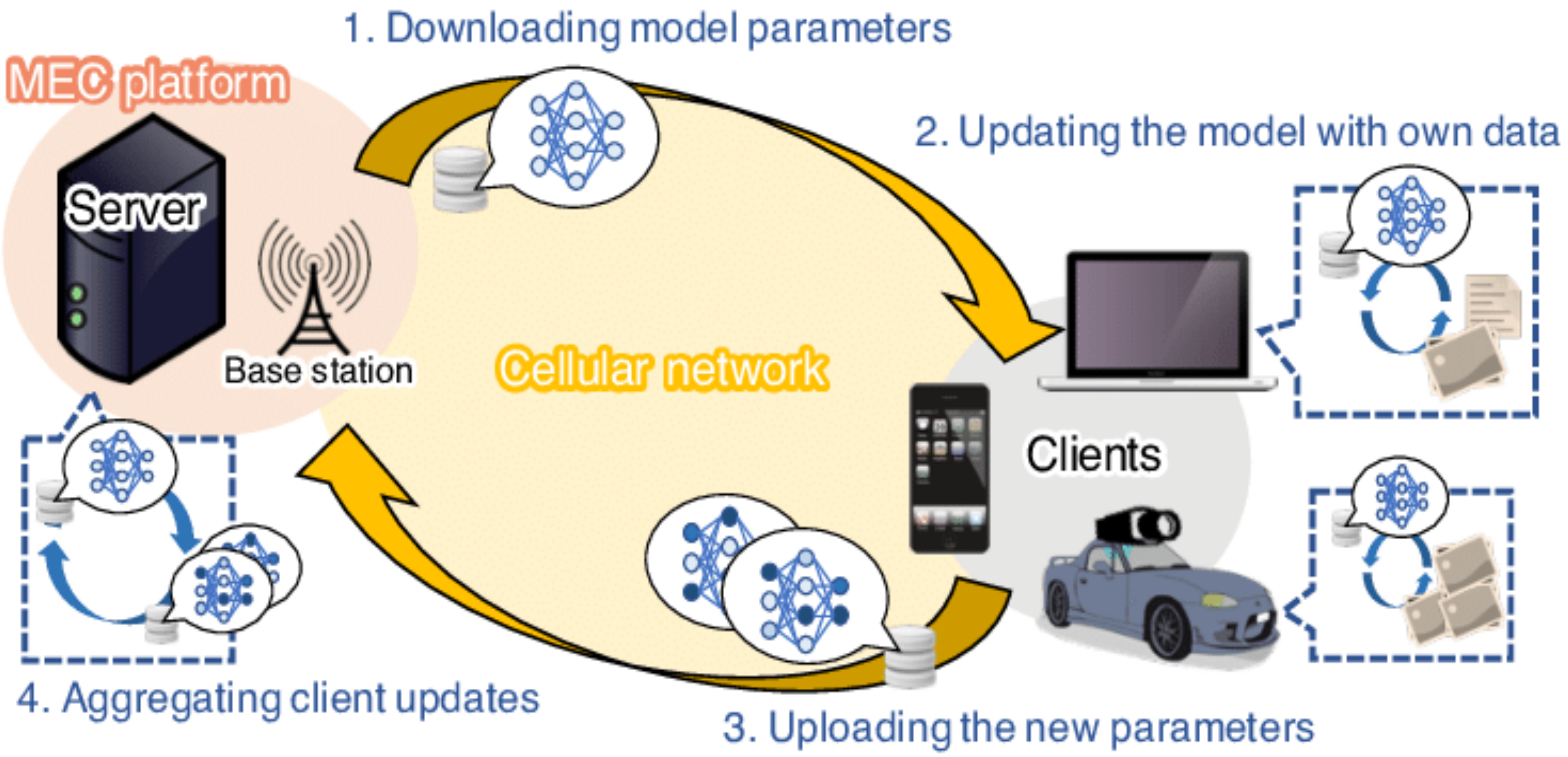


Ensembles using Pareto optimal features

Multi-Objective Evolutionary Federated Learning

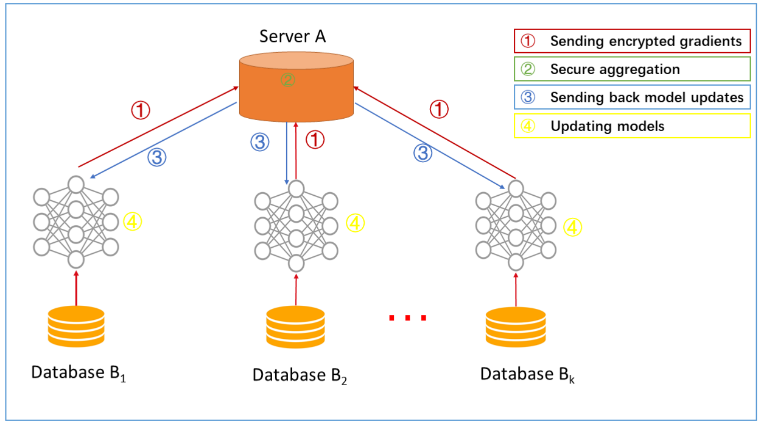
H. Zhu and **Y. Jin**. Multi-objective evolutionary federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(4): 1310-1322, 2020

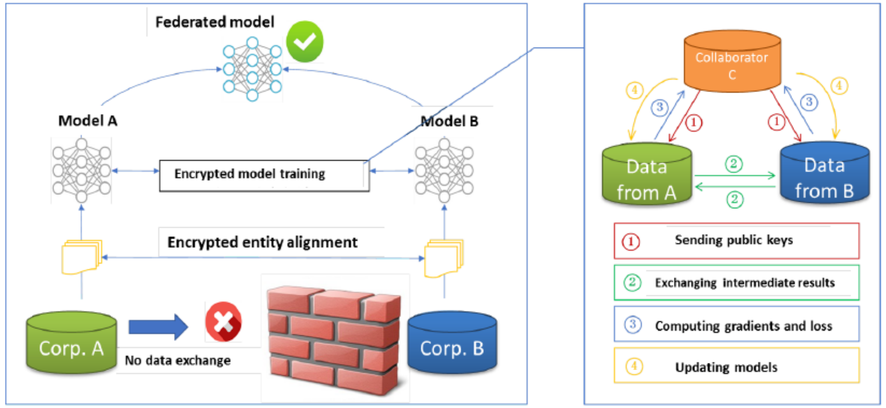
Privacy-Preserving Federated Learning



Privacy-Preserving Federated Learning

- **Federated learning** is a machine learning setting where the goal is to train a high-quality *centralized model* with training data *distributed* over a large number of clients, each with *unreliable* and relatively *slow* network connections.





a

b

	ID	X1	X2	X3	ID	X4	X5	Y
DB1	U1	9	80	600	U1	6000	600	No
	U2	4	50	550	U2	5500	500	Yes
	U3	2	35	520	U3	7200	500	Yes
DB2	U4	10	100	600	U4	6000	600	No
	U5	5	75	600	U8	6000	600	No
DB3	U6	5	75	520	U9	4520	500	Yes
	U7	8	80	600	U10	6000	600	No

Horizontal federated learning

DB1				DB2			
ID	X1	X2	X3	ID	X4	X5	Y
U1	9	80	600	U1	6000	600	No
U2	4	50	550	U2	5500	500	Yes
U3	2	35	520	U3	7200	500	Yes
U4	10	100	600	U4	6000	600	No
U5	5	75	600	U8	6000	600	No
U6	5	75	520	U9	4520	500	Yes
U7	8	80	600	U10	6000	600	No

Vertical federated learning

Main Challenges in Federated Learning

- **Extra communication cost,** computation power and storage capacity are required

- The data on each edge device
  - Class labels may be **imbalanced**
  - Attributes may not be independent and identically distributed (**Non-IID**)
  - Attributes may be **vertically partitioned**

- Vulnerable to **adversarial attacks**

- Not all clients may participate in learning in each round, and the number of clients may be huge, and the clients may be **heterogeneous** in computation and communication power

H. Zhu, J. Xu, S. Liu and Y. Jin. Federated learning on non-iid data: A survey. *Neurocomputing*, 465: 371-390, 2021

Bi-Objective Federated Learning

- Objectives
  - Maximization of the learning performance of the central model
  - Minimization of the communication cost

- Decision variables

  - The hyperparameters, such as learning rate, batch size

  - Parameters of the deep neural network

  - Structure of the deep neural network

- How to encode deep neural networks such as CNN and MLP?

Scalable Encoding of Neural Connectivity

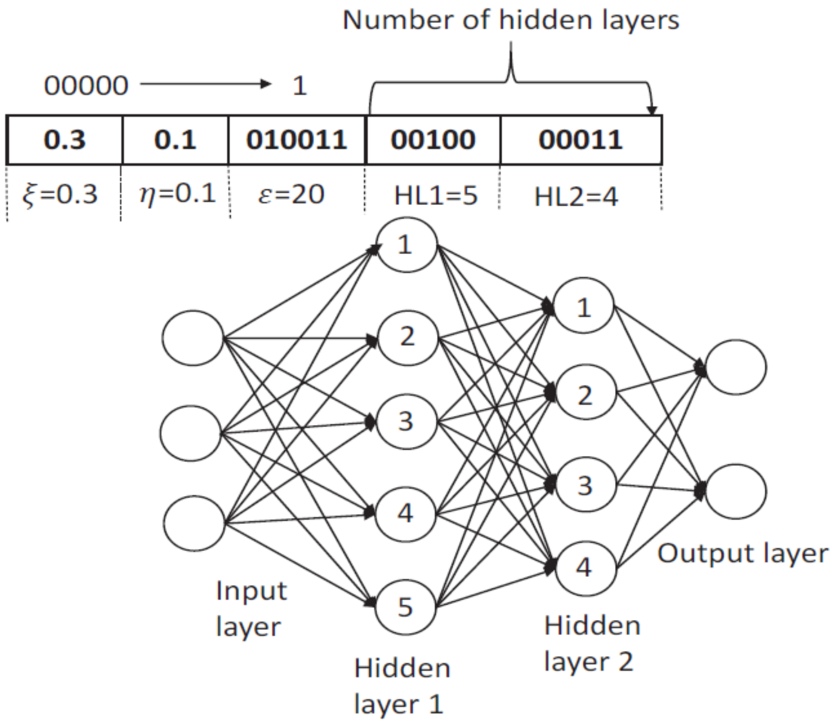
- Encoding of deep neural networks is extremely challenging since it involves a very large number if decision variables
- A modified sparse evolutionary training (SET) is adopted:
  - Use a Erdos Rnyi random graph to determine the connectivity between every two neighboring layers of the neural network

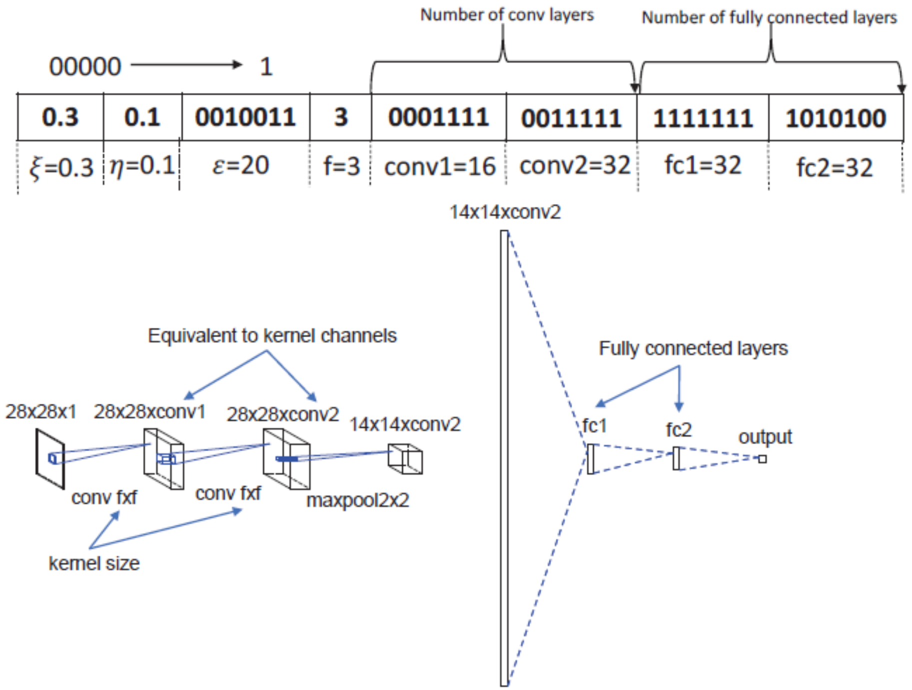
$$p(W_{ij}^k) = \frac{\varepsilon(n^k + n^{k-1})}{n^k n^{k-1}}$$
$$n^W = n^k n^{k-1} p(W_{ij}^k)$$

  - where n^k and n^{k-1} are the number of neurons in layer k and k − 1, respectively, W^k_{ij} is the sparse weight matrix between the two layers, ε is a SET parameter that controls connection sparsity, and n^W is the total number of connections between the two layers
  - It is easy to find that the connection probability would become significantly lower, if $\varepsilon \ll n^k$ and $\varepsilon \ll n^{k-1}$
  - remove a fraction ξ of the weights that have updated the smallest during each training epoch, which can be seen as the selection operation of an evolutionary algorithm
  - Removal is applied at the last SGD iteration only

H. Zhu and Y. Jin. Multi-objective evolutionary federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(4): 1310-1322, 2020

Genetic Representation





MLP

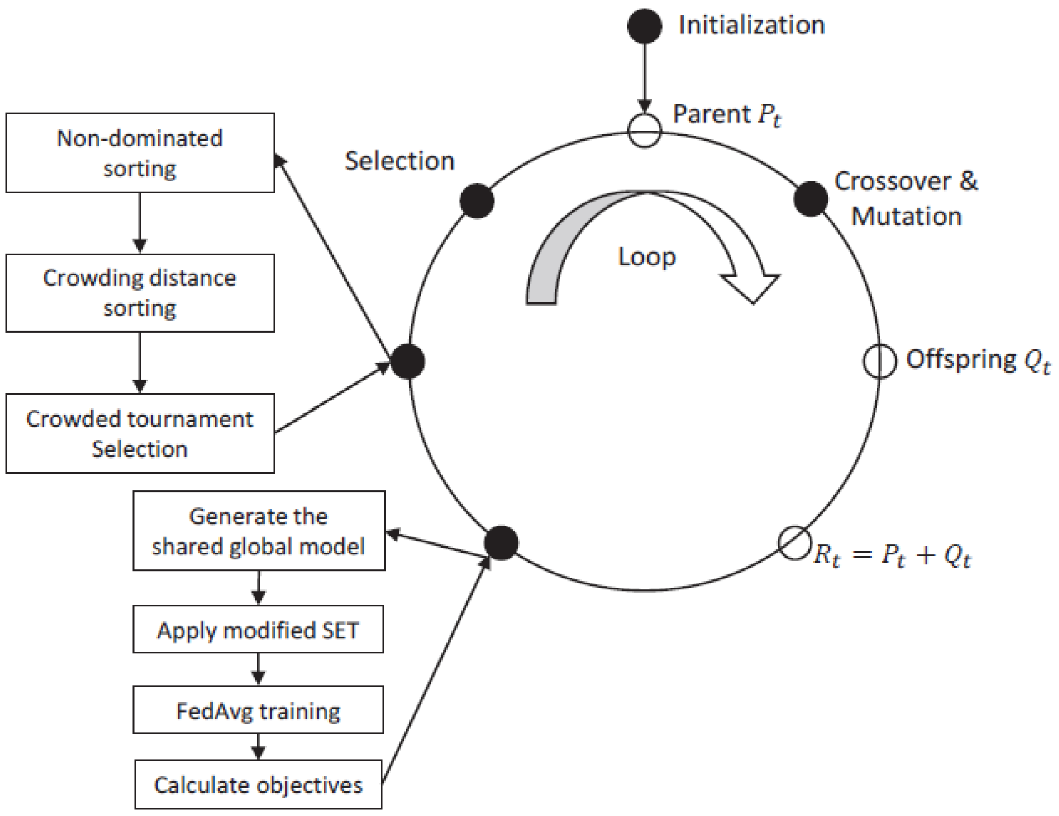
CNN

Bi-Objective Federated Learning

- Minimize the following two objectives using NSGA-II

$$E_t = 1 - A_t$$

$$\Omega_t = \sum^{K} \Omega_k / K$$



Experimental Settings

- The **standard FL**: MLP and CNN on the MNIST data
  - MLP: a learning rate of 0.1 and the batch size is 50;
  - two hidden layers, each having 200 nodes (**199,210** parameters in total) and uses the ReLu function as the activation function

  - CNN: two 3 × 3 kernel filters (the first with 32 channels and the second with 64 channels)
  - a 2 × 2 max-pooling layer, a 128 fully connected layer and finally a 10 class softmax output layer (**1,625,866** parameters in total)
  - 100 clients, mini-batch size = 50, training epoch = 5

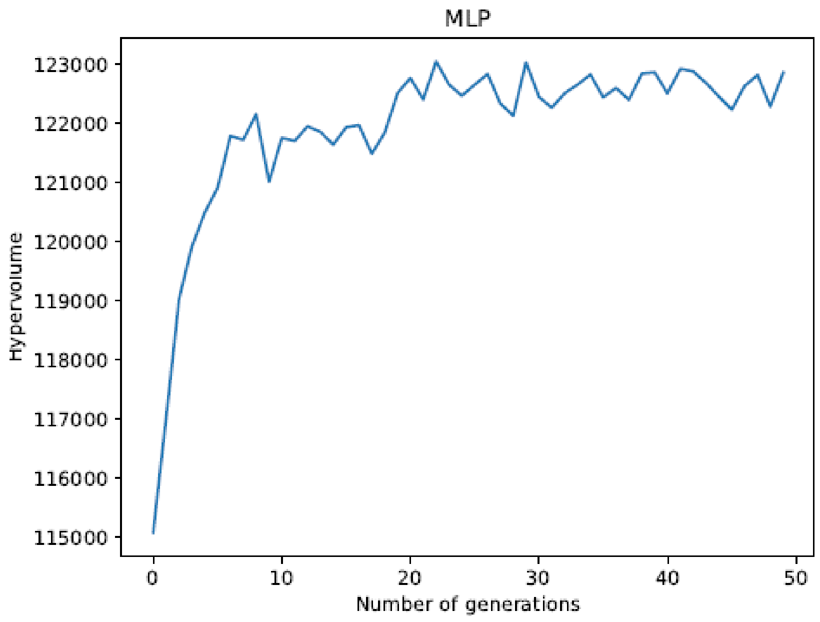
- For the evolutionary FL:
  - Population size =20, generation = 20 for IID data and 50 for non-IID data
  - Communication round = 5 for IID data and 10 for non-IID data
  - $\varepsilon = 20$ and $\xi = 0.3$ (for comparison)

Influence of the Connectivity

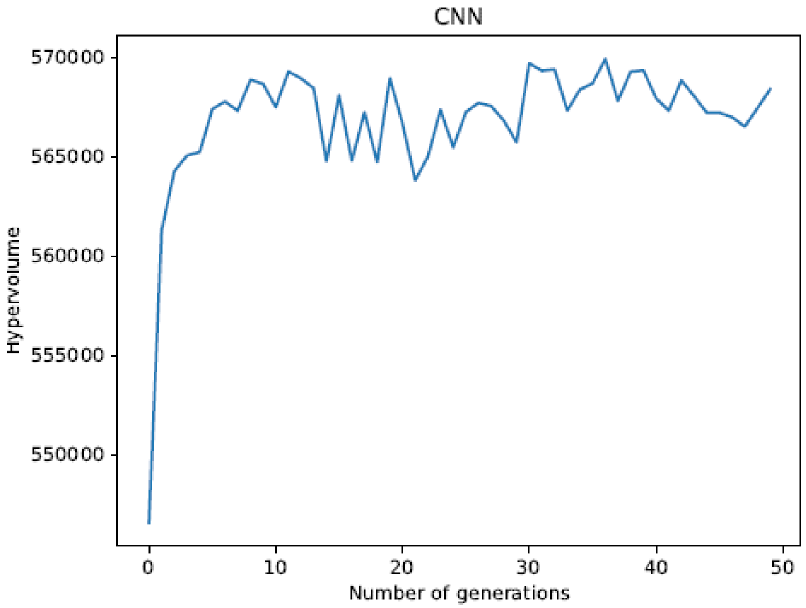
ε = 20 and ξ = 0.3

Local data distributions		IID		non-IID	
		Accuracy	Connections	Accuracy	Connections
Fully connected	MLP	98.13%	199,210	97.04%	199,210
	CNN	98.85%	1,625,866	98.75%	1,625,866
Sparsely connected	MLP	96.69%	19,360	94.45%	18,785
	CNN	98.44%	185,407	98.32%	184,543

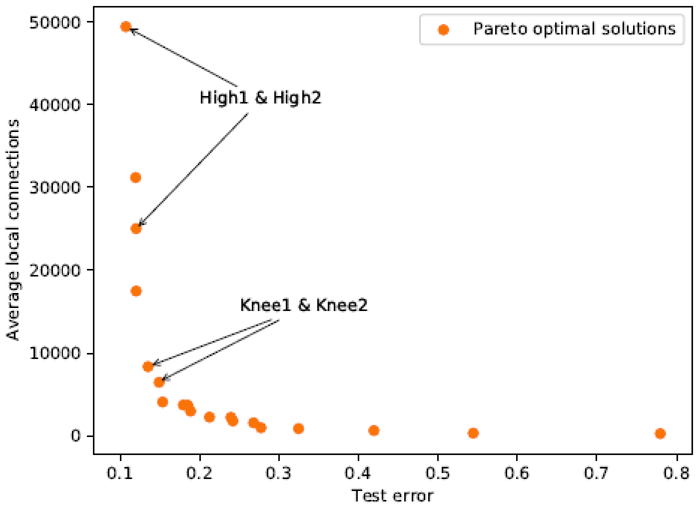
Results

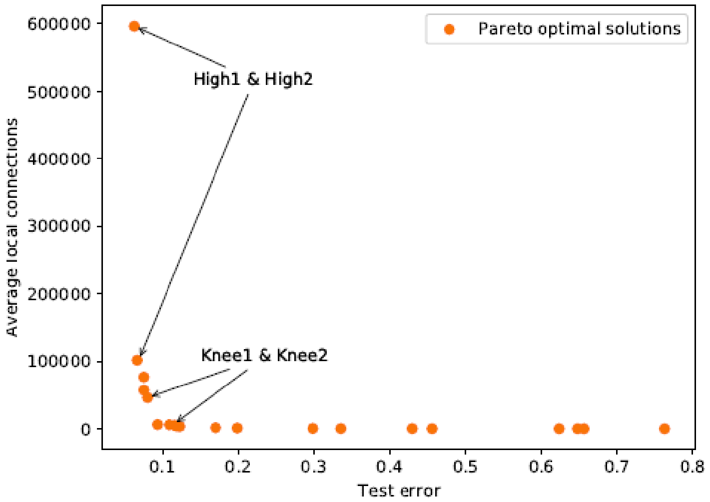


(a) Hypervolume for MLP



(b) Hypervolume for CNN





Results

HYPER-PARAMETERS OF HIGH1, HIGH2, KNEE1, AND KNEE2 FOR ***MLPs*** EVOLVED ON ***non-IID*** DATA AND THEIR VALIDATION RESULTS

Parameters	Knee1	Knee2	High1	High2	Standard
Hidden layer1	49	53	86	109	200
Hidden layer2	/	/	/	/	200
ε	10	8	66	34	/
ξ	0.1106	0.0764	0.1106	0.1566	/
Learning rate η	0.3	0.2961	0.3	0.3	0.1
Test accuracy IID	96.78%	96.41%	97.82%	97.68%	98.13%
Connections IID	7,749	5,621	45,329	22,210	199,210
Test accuracy nonIID	94.85%	94.88%	97.32%	96.21%	97.04%
Connections nonIID	8,086	6,143	45,530	24,055	199,210

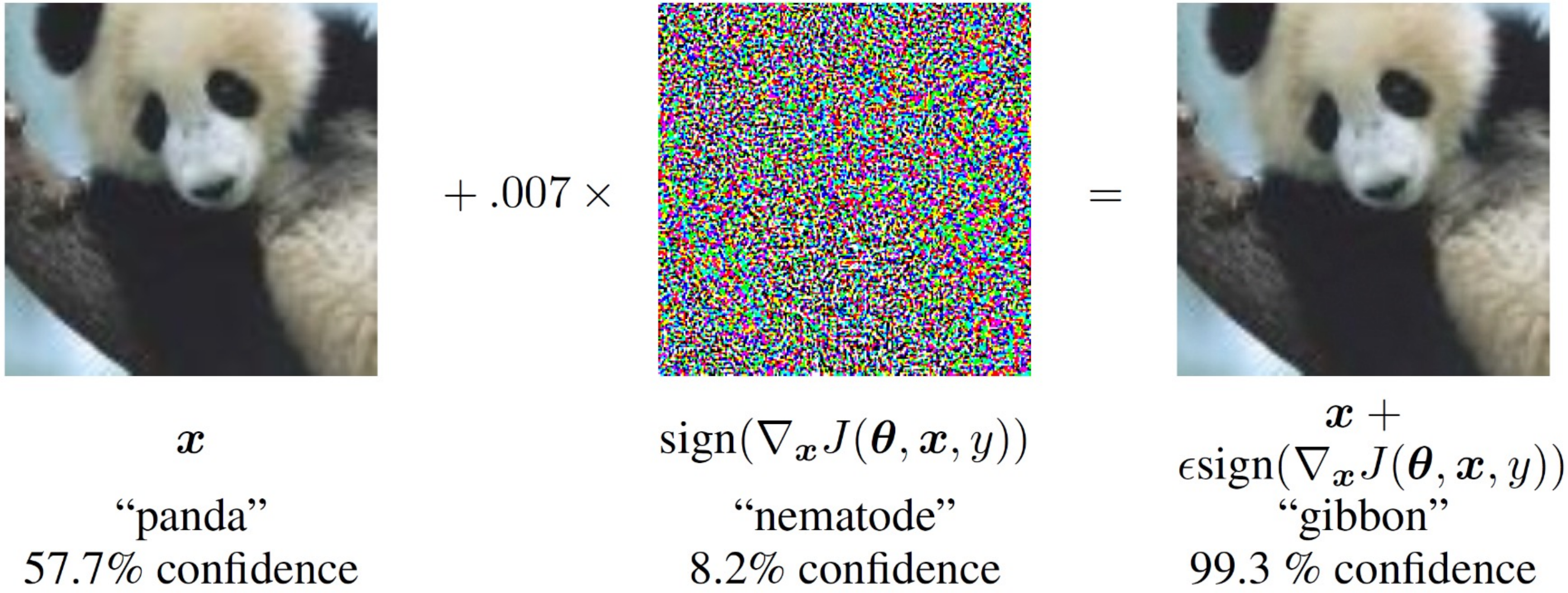
HYPER-PARAMETERS OF HIGH1, HIGH2, KNEE1, AND KNEE2 FOR ***CNNs*** EVOLVED ON ***non-IID*** DATA AND THEIR VALIDATION RESULTS

Parameters	Knee1	Knee2	High1	High2	Standard
Conv layer1	17	5	53	33	32
Conv layer2	/	/	/	/	64
Fully connected layer1	29	21	208	31	128
Fully connected layer2	/	/	/	/	/
Kernel size	5	5	5	5	3
ε	18	8	66	20	/
ξ	0.1451	0.1892	0.0786	0.1354	/
Learning rate η	0.2519	0.2388	0.2776	0.2503	0.1
Test accuracy IID	98.84%	98.15%	99.06%	98.93%	98.85%
Connections IID	48949	6262	622090	107224	1,625,866
Test accuracy nonIID	97.92%	97.7%	98.52%	98.46%	98.75%
Connections nonIID	39457	6804	553402	90081	1,625,866

Search for Robust Neural Architectures

Adversarial Robustness of Deep Neural Networks

- Deep neural networks are vulnerable to carefully designed adversarial attacks

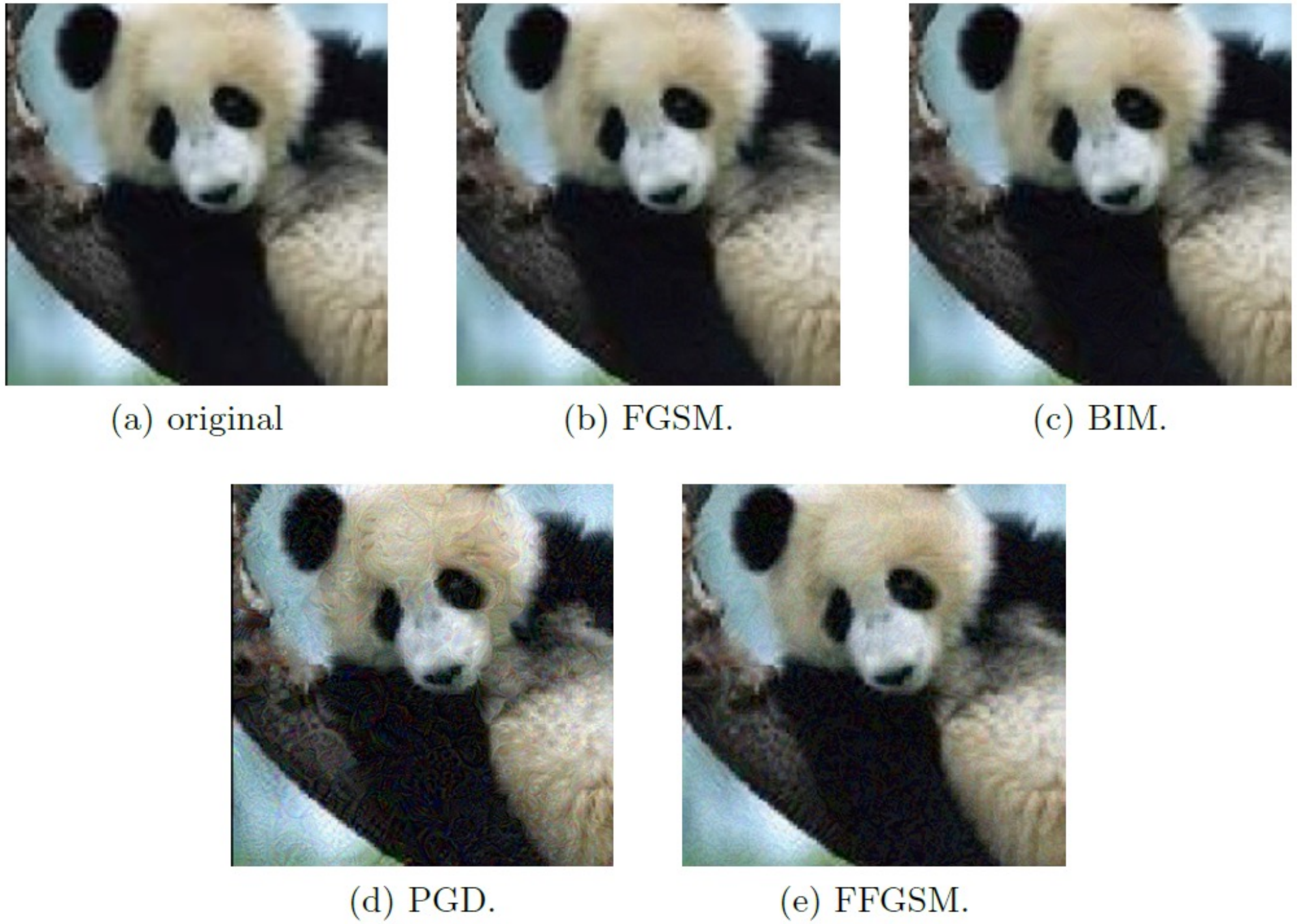


- Fast Gradient Sign Method (FGSM)

Adversarial example: $$x^* = x + \varepsilon \, sign(\nabla_x J(\theta, x, y))$$

(Goodfellow et al., 2015)

White-box Attacks



Various adversarial attacks on Inception V3

White-box attacks assume that the adversary knows detailed information of the targeted models

- model architecture
- hyper-parameters
- gradients
- training data

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: *CVPR 2016*, pp. 2818–2826.

Adversarial Robustness by Design

- Most existing work investigate the robustness of various deep learning models under **a particular type of attack**
- Is it possible to search for neural architectures that are robust to multiple adversarial attacks?
- Objectives
  - ➢ Accuracy on clean data
  - ➢ Robustness to five types of white-box attacks

Adversarial Robustness by Design

- It will be computationally extremely intensive to evaluate the performance of all candidate architectures on **the clean data and four white-box and one black-box adversarial data sets**

- One of the five attacks is **randomly selected** in each assessment to reduce the computational cost,

- To make the adversarial performances **comparable**, the adversarial error is normalized over the performance of **18 baseline DNN architectures**

$$\min : F(X) = \{f_1, f_2\}$$

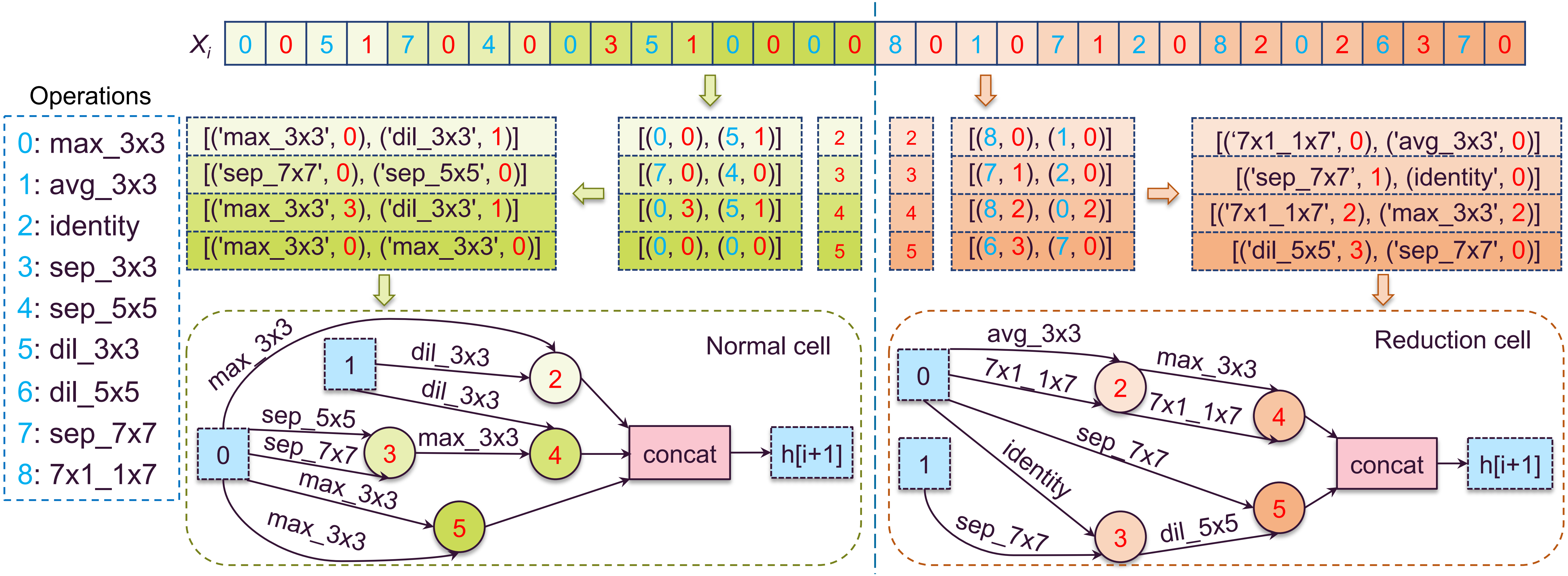
$$f_1 = Err_{clean} = (1 - \frac{1}{N}\sum I(\hat{y} == y)) \times 100\%$$

$$f_2 = \frac{Err_{ad} - \mu_i}{\sigma_i}$$

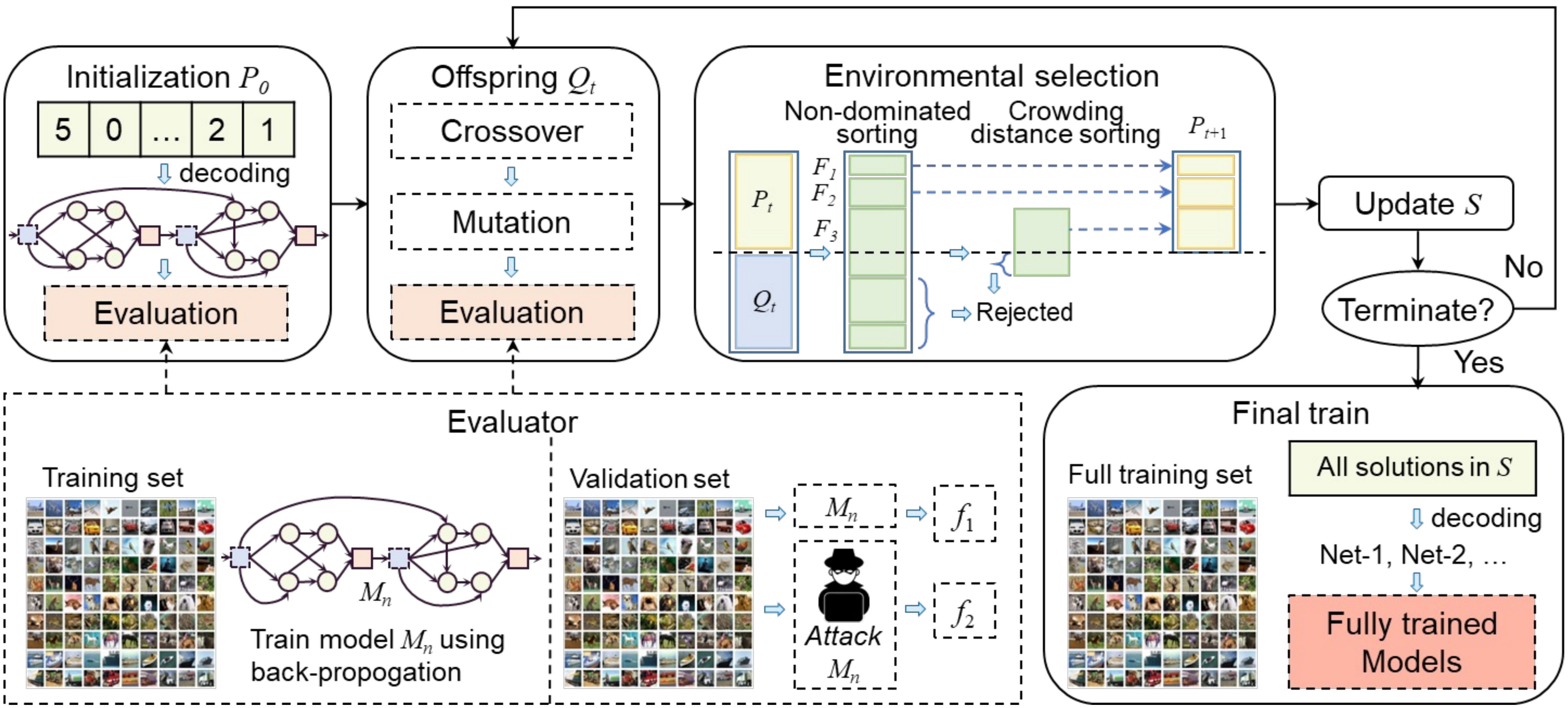
Err_{ad} the error rate on adversarial examples generated from a randomly selected type of adversarial attack, μ_{i} and σ_{i} and are the mean and the standard deviation of the error rate of different baseline architectures under the i-th adversarial attack

J. Liu and Y. Jin. Multi-objective search of robust neural architectures against multiple types of adversarial attacks. *Neurocomputing*, 453: 73-84, 2021

Neural Network Representation (Micro Search Space)



Overall Framework



Comparative Results

Datasets	Models	Clean Acc (%)	FGSM (%)	BIM (%)	PGD-7 (%)	PGD-10 (%)	PGD-20 (%)	PGD-50 (%)	FFGSM (%)	Blk-FGSM (%)
CIFAR-10	PreAct ResNet-18	83.54	55.50	**67.89**	48.47	48.45	48.43	48.43	48.47	59.05
	WideResNet-34	86.52	53.57	67.65	47.10	47.10	47.10	46.90	47.10	56.75
	RobNet-small	78.05	53.93	-	-	-	48.32	-	-	-
	RobNet-medium	78.33	54.55	-	-	-	49.13	-	-	-
	RobNet-large	78.57	54.98	-	-	-	49.44	-	-	61.92
	RobNet-large-v2	85.69	57.18	-	-	-	50.53	-	-	-
	RobNet-free	82.79	58.38	-	-	-	52.74	-	-	65.06
	E2RNAS-C46	**96.36**	-	-	-	10.21	-	-	-	-
	E2RNAS-C36	95.81	-	-	-	9.61	-	-	-	-
	E2RNAS-C25	95.14	-	-	-	7.76	-	-	-	-
	E2RNAS-C16	93.97	-	-	-	6.76	-	-	-	-
	Ours	82.82	**59.42**	66.18	**58.56**	**58.44**	**58.42**	**58.41**	**58.87**	**66.2**
CIFAR-100	PreAct ResNet-18	60.78	30.35	**47.51**	28.63	28.04	28.02	28.01	28.30	31.50
	WideResNet-34	60.57	30.84	44.93	29.53	29.11	28.61	28.61	29.34	32.94
	E2RNAS-C38	80.7	-	-	-	4.90	-	-	-	-
	E2RNAS-C36	**80.81**	-	-	-	4.00	-	-	-	-
	E2RNAS-C29	80.2	-	-	-	3.78	-	-	-	-
	E2RNAS-C20	77.03	-	-	-	3.44	-	-	-	-
	Ours	59.98	**35.72**	42.24	**35.02**	**34.55**	**34.56**	**34.56**	**35.11**	**41.59**

Multi-Fidelity Multi-objective Search of Robust Neural Architectures

J. Liu, R. Cheng, Y. Jin. Bi-fidelity evolutionary multi-objective search for adversarially robust deep neural architectures. arXiv preprint arXiv:2207.05321

Motivations – Enhance Computational Efficiency

西湖大學 WESTLAKE UNIVERSITY

- To accelerate the search process, we predict the performance of candidate architectures by combining **weight sharing** with a **predictor-based evaluator**, where the parameters directly inherited from a trained robust supernet, and the performance calculated from a partial validation set (20%) is used as a low-fidelity fitness evaluation

- We calculate the performance of architecture on the entire validation set as the **high-fidelity fitness** evaluation and a **surrogate model** is built from the high-fidelity fitness evaluation and used to approximate the high-fidelity fitness function

- A **three-objective optimization problem** is formulated to further enhance the efficiency in search for adversarially robust DNNs, where the performance predicted by a surrogate model is introduced as a third (

$$\min : F(\mathbf{x}) = \{f_1, f_2, f_3\}$$

$$f_1(\mathbf{x}) = f_1^l(\mathbf{x}) = 1 - (\frac{1}{N}\sum \mathbb{I}(\hat{y} == y))$$

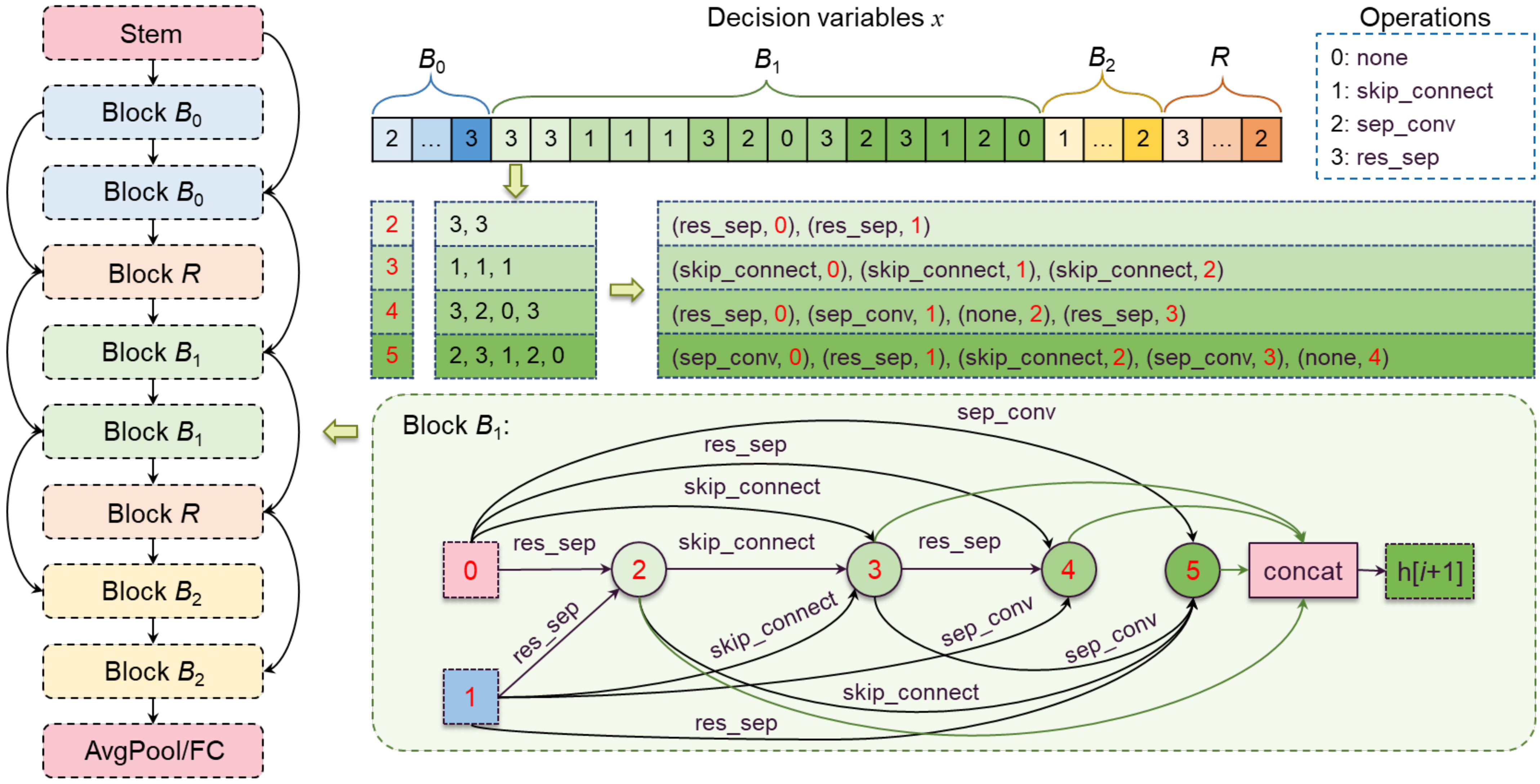
$$f_2(\mathbf{x}) = f_2^l(\mathbf{x}) = 1 - (\frac{1}{N}\sum \mathbb{I}(\hat{y}_{adv} == y))$$

$$f_3(\mathbf{x}) = f_s(\mathbf{x})$$

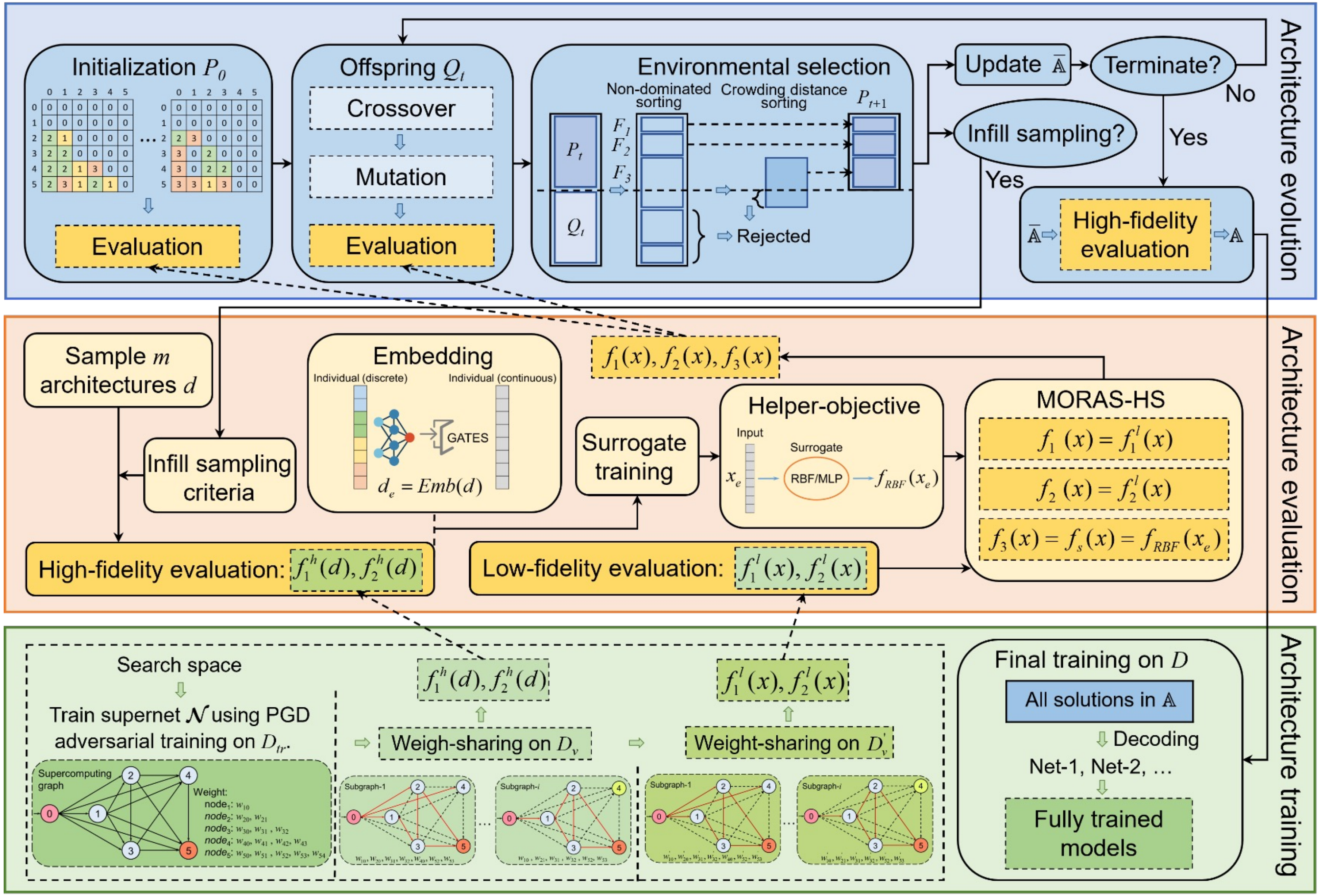
$f_1^l(\mathbf{x})$, $f_2^l(\mathbf{x})$ denote the low-fidelity fitness evaluations calculated by the error rate on the partial validation set;

$f_3(\mathbf{x})$ represents the auxiliary-objective which is predicted by the surrogate model

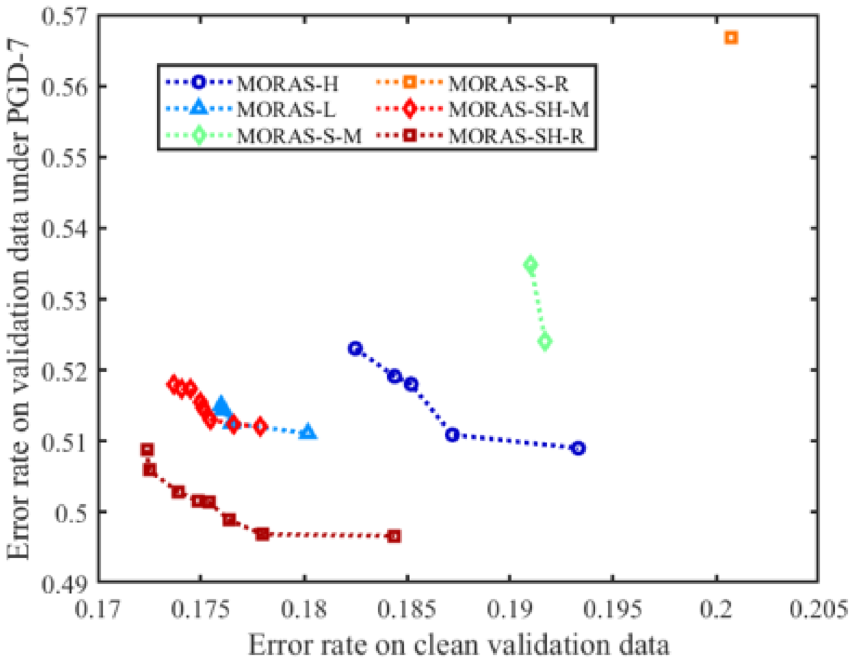
Network Representation



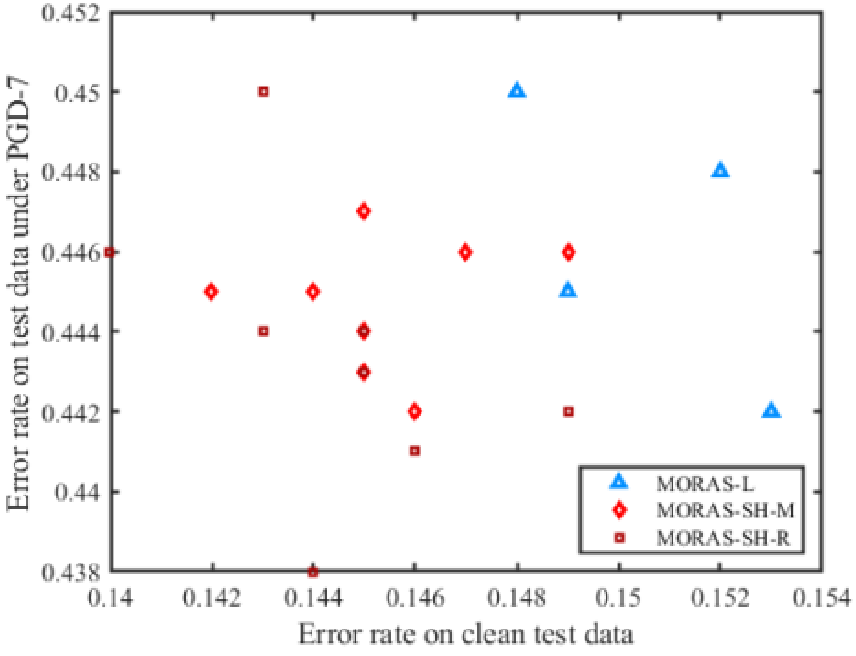
Overall Framework



Comparative Results



Pareto fronts obtained by comparative experiments, where the parameters are inherited from the supernet



The performance of architectures obtained by comparative experiments after adversarial training from scratch.

The maximum computing time is set to 3 GPU days

Comparative Results

西湖大學 WESTLAKE UNIVERSITY

TABLE I: Comparison with peer competitors under various adversarial attacks on CIFAR-10.

	Architecture	Clean (%)	FGSM (%)	PGD-7 (%)	PGD-20 (%)	PGD-100 (%)	#Para (M)	FLOPS (M)
Manually designed networks	MobileNet-V2	77.0	53.0	50.1	48.0	47.8	2.30	182
	VGG-16	79.9	53.7	50.4	48.1	47.9	14.73	626
	ResNet-18	83.9	57.9	54.5	51.9	51.5	11.17	1110
NAS-based methods	RobNet-Free	82.8	58.4	55.1	52.7	52.6	5.49	1560
	MSRobNet-1560	84.8	**60.0**	**56.2**	**53.4**	**52.9**	5.30	1588
	MSRobNet-1560-P	**85.2**	59.4	55.2	51.9	51.5	4.88	1565
Ours	MORAS-SHNet-M1	85.8	59.4	55.5	52.5	52.1	5.22	1634
	MORAS-SHNet-M2	85.4	**60.1**	55.8	52.9	52.4	5.05	1606
	MORAS-SHNet-M3	85.5	59.6	55.6	52.8	52.5	5.20	1661
	MORAS-SHNet-R1	**86.0**	59.9	55.4	52.1	51.6	5.60	1525
	MORAS-SHNet-R2	85.6	59.9	**56.2**	**53.1**	52.6	5.42	1471
	MORAS-SHNet-R3	85.1	59.9	55.8	53.0	**52.7**	5.41	1484

TABLE III: Comparison with peer competitors under various adversarial attacks on SVHN.

	Architecture	Clean (%)	FGSM (%)	PGD-7 (%)	PGD-20 (%)	PGD-100 (%)
Manually designed networks	MobileNet-V2	93.9	73.0	61.9	55.7	53.9
	VGG-16	92.3	66.6	55.0	47.4	45.1
	ResNet-18	92.3	73.5	57.4	51.2	48.8
NAS-based methods	RobNet-Free	94.2	84.0	**66.1**	**59.7**	**56.9**
	MSRobNet-1560	**95.0**	77.5	64.0	57.0	54.2
	MSRobNet-2000	94.9	**84.8**	65.3	58.8	55.1
Ours	MORAS-SHNet-M1	94.8	86.7	78.4	66.0	61.2
	MORAS-SHNet-M2	94.4	84.3	65.3	58.6	55.6
	MORAS-SHNet-M3	**95.8**	**90.6**	**85.7**	**73.7**	**66.3**
	MORAS-SHNet-R1	94.9	85.4	64.1	57.8	54.9
	MORAS-SHNet-R2	94.3	83.9	63.8	58.1	55.4
	MORAS-SHNet-R3	94.7	77.3	61.4	55.1	52.8

Summary

- Multi-objective machine learning based on Pareto-optimality provides novel perspectives on machine learning
- Models of different qualities (accuracy, complexity, interpretability, robustness, and fairness) are of great interest and deserves more attention in machine learning
- The Pareto-front achieved by evolutionary multi-objective algorithms reveals important information of the problem at hand