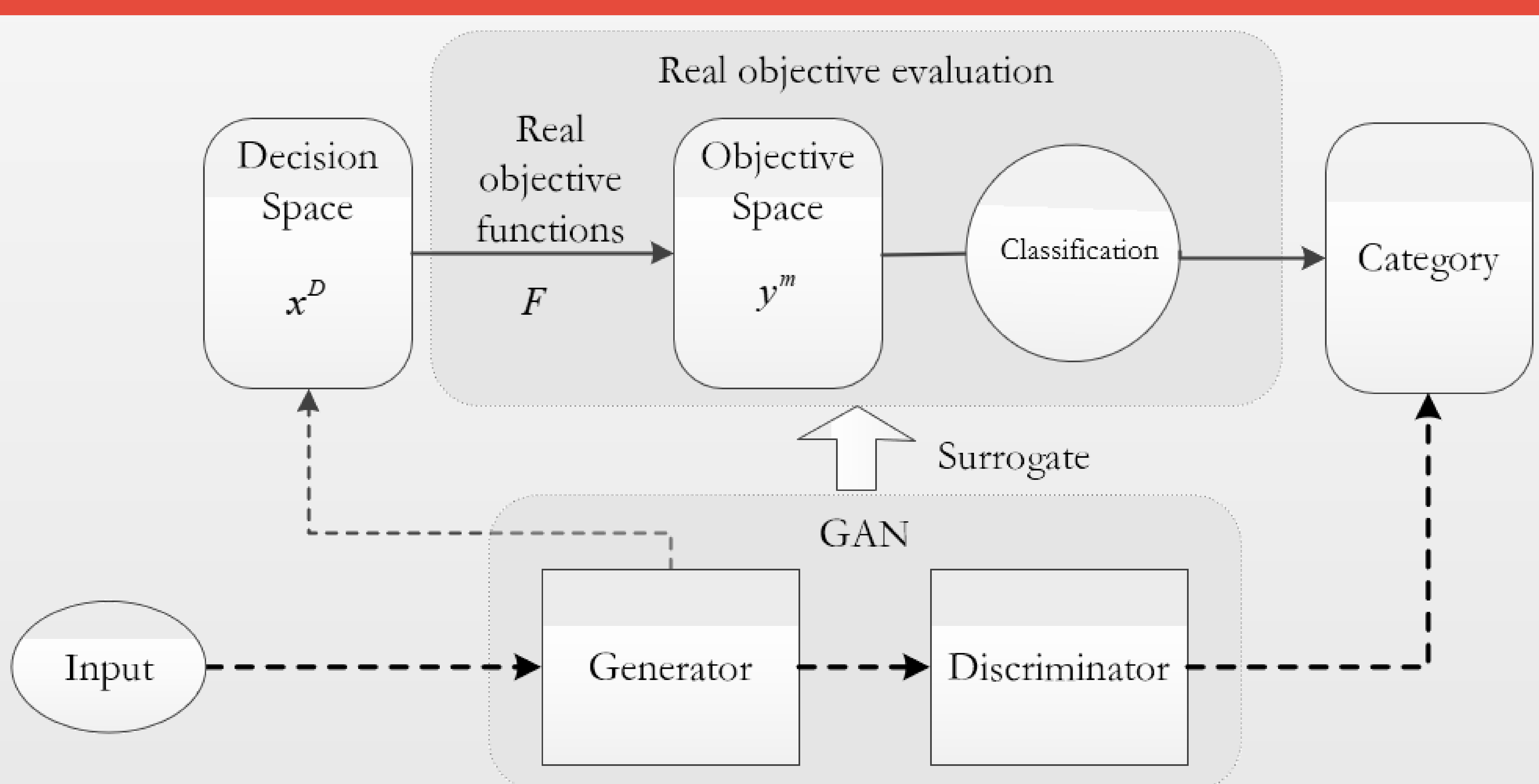
Evolutionary Multiobjective Optimization Driven by Generative Adversarial Networks (GANs)

Cheng He[1], Shihua Huang[1], Ran Cheng[1], Kay Chen Tan[2], Yaochu Jin[3]
Department of Computer Science and Engineering, Southern University of Science and Technology
Department of Computer Science, City University of Hong Kong
Department of Computer Science, University of Surrey

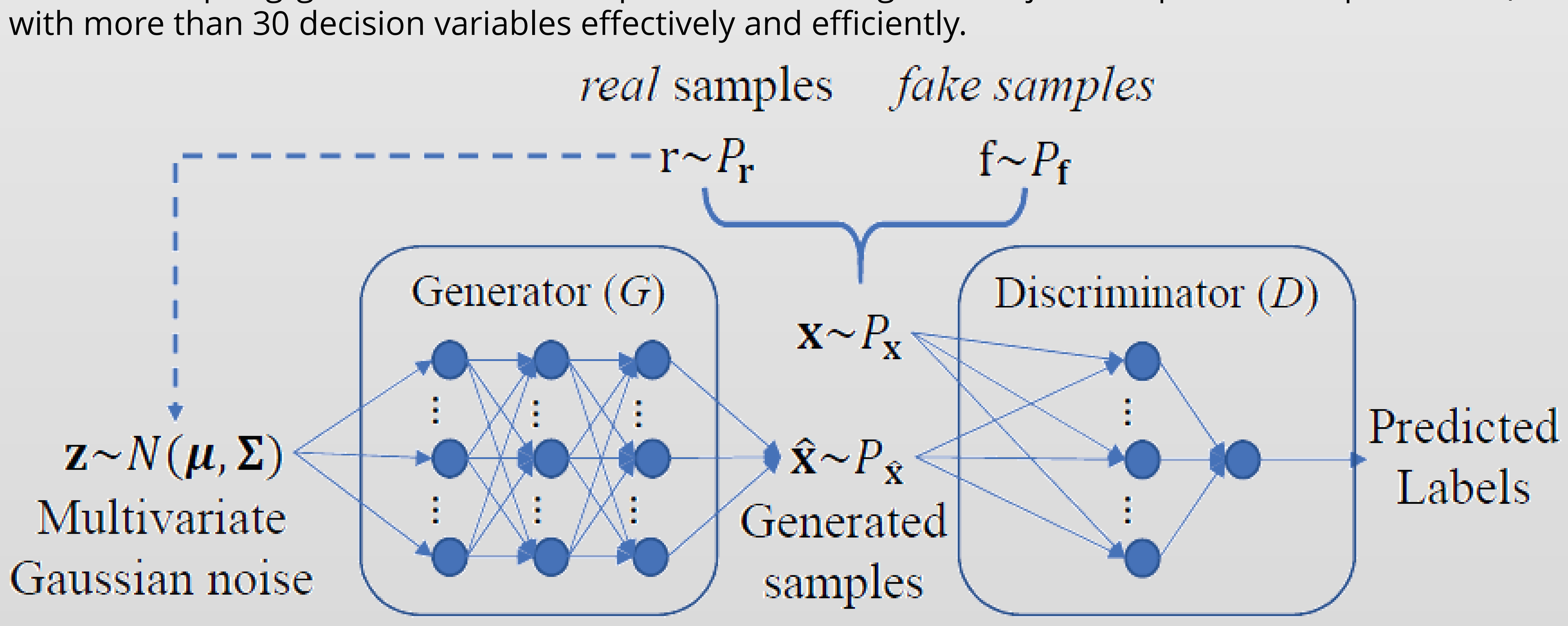
Introduction

The combination of machine learning models and evolutionary algorithms (EAs) has been promising research in evolutionary computation for decades. Generally, the models enable the algorithms to learn from the evolving population, and thus EAs can save lots of function evaluations. However, the performance of such model-based EAs is highly dependent on the training qualities of the adopted models, and it deteriorates rapidly with the increase of the problem scales due to the curse of dimensionality. This work proposes a multiobjective evolutionary algorithm driven by the generative adversarial networks (GANs), where the GANs are used to generate promising offspring solutions in high-dimensional decision space. Compared with surrogate-assisted EAs, this work provides a new way for adopting deep learning models in handling complex optimization problems. At each generation of the proposed algorithm, the parent solutions are first classified into real and fake samples to train the GANs; then the offspring solutions are sampled by the trained GANs. Thanks to the powerful generative ability of the GANs, the proposed algorithm is capable of generating promising offspring solutions in high-dimensional decision space with limited training data.

Methodology



The core idea is to apply the GAN network to learn the distribution of solutions on the Pareto optimal set (PS) for generating candidate solutions close to the PS. As a model-based algorithm, it focuses on efficient offspring generation and is capable of handling multiobjective optimization problems (MOPs) with more than 30 decision variables effectively and efficiently.



Apart from the model-based offspring generation, the training of the Gan is also interesting and skillful. Due to the unsupervised learning-based principle in GANs, the generation of preferred data is challenging. Thus, we have proposed to classify existing solutions into real and fake samples, aiming to provide additional learning direction during the training of the GAN.

Algorithm 2 General Framework of GMOEA

Input:
N (population size), m (batch size)
$P \leftarrow$ Initialize a population of size N
$GAN \leftarrow$ Initialize the GANs
while termination criterion not fulfilled **do**
  $\mathbf{X} \leftarrow$ Solution Classification /*Half of the solutions in P are classified as *fake* samples*/
  net $\leftarrow$ Model Training /*Use $\mathbf{X}$ to train the model*/
  $Q \leftarrow$ Offspring Reproduction /*Generate N offspring solutions by the proposed reproduction method*/
  $P \leftarrow$ Environmental Selection /*Select N solutions from the combination of P and Q*/
end
Return: P

$$\nabla\theta_g \frac{1}{m}\sum_{i=1}^{m}\log(1-D(G(\mathbf{z}_i))) \tag{5}$$

$$\max_D V(D) = \mathrm{E}_{\mathbf{r}\in P_\mathbf{r}}[\log(D(\mathbf{r}))] + \mathrm{E}_{\mathbf{f}\in P_\mathbf{f}}[\log(1-D(\mathbf{f}))] + \mathrm{E}_{\mathbf{z}\in P_\mathbf{z}}[\log(1-D(G(\mathbf{z})))] \tag{11}$$

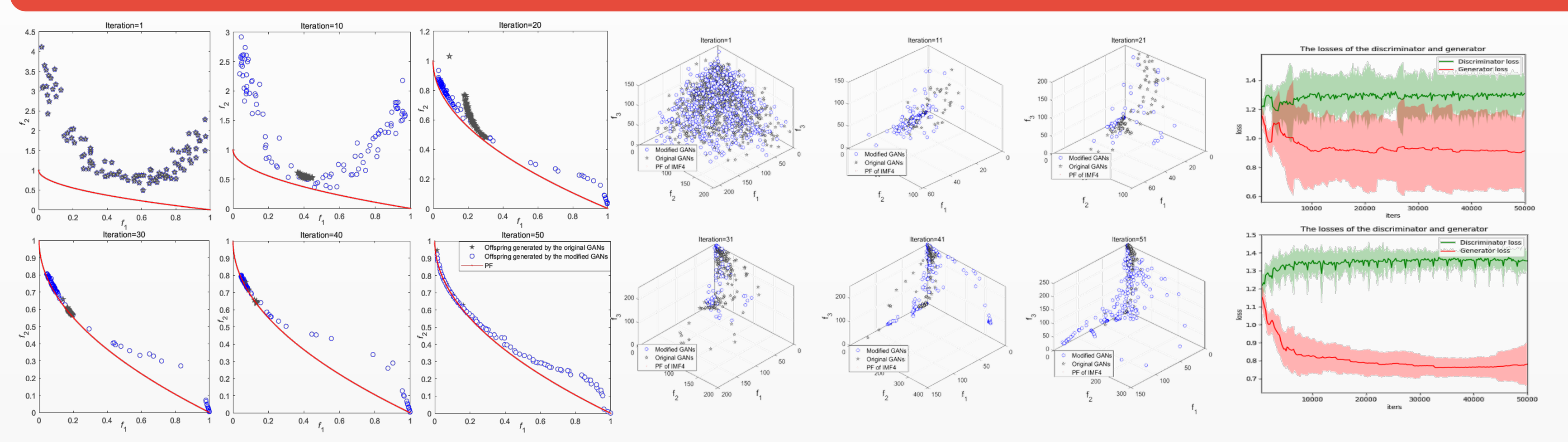
Algorithm 3 Solution Classification

Input: N' (number of fake samples), P (population).
$Fit \leftarrow$ Calculate the fitness values of candidate solutions in P according to (9)
$A \leftarrow \arg_{\mathbf{x}_i\in P} Fit(\mathbf{x}_i) < 1$
if $|A| \leq N'$ **then**
  $A \leftarrow$ Select N' candidate solutions with the minimal Fit
else
  while $A > N'$ **do**
    Delete $\arg\min_{\mathbf{x}_j\in A}\min dis(\mathbf{x}_j, A\backslash\mathbf{x}_j)$ in A
  end
end
$A \leftarrow real$
$P\backslash A \leftarrow fake$
Return: $\mathbf{X} \leftarrow \{A\bigcup(P\backslash A), \{real\}^{\mathbb{N}'\times 1}\bigcup\{fake\}^{\mathbb{N}'\times 1}\}$/*$\mathbf{X}$ is a tuple, where the first element denotes the decision vector and the second one denotes the label*/

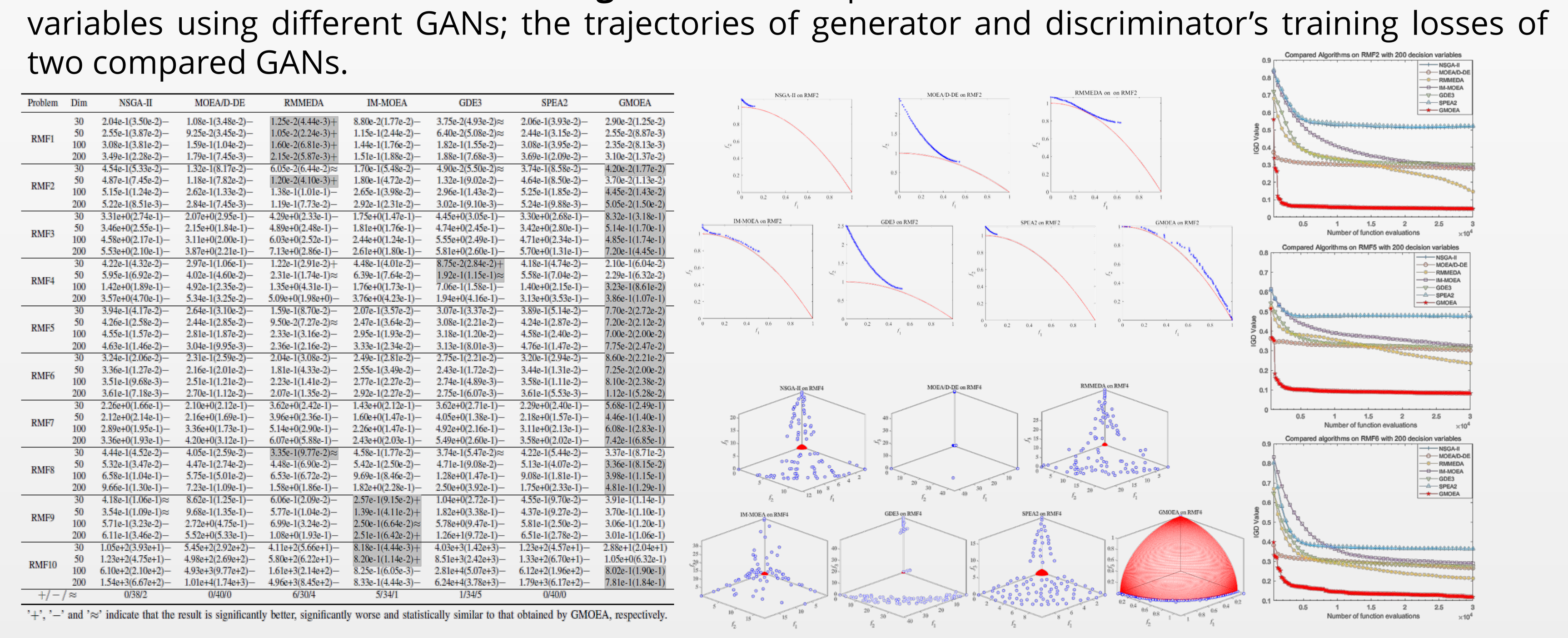
Algorithm 4 Model Training

Input: $\mathbf{X}$ (given data), m (batch size).
$\boldsymbol{\mu} \leftarrow mean(P_\mathbf{r})$ /*Mean vector of $P_\mathbf{r}$ with $P_\mathbf{r}\in\mathbf{X}$*/
$\boldsymbol{\Sigma} \leftarrow cov(P_\mathbf{r})$ /*Covariance matrix of $P_\mathbf{r}$*/
for total number of training iterations **do**
  $\mathbf{X}' \leftarrow \mathbf{X}$
  for $i \leftarrow 1 : |\mathbf{X}|/m$ **do**
    $\mathbf{T} \leftarrow$ Randomly sample m data points from $\mathbf{X}'$
    $\mathbf{X}' \leftarrow \mathbf{X}'\backslash\mathbf{T}$
    $\mathbf{Z} \leftarrow$ Sample m data points from multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$
    Update the discriminator according to (11) by using $\mathbf{T}$ and $\mathbf{Z}$
    $\mathbf{Z} \leftarrow$ Sample m data points from multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$
    Update the generator according to (5) by using $\mathbf{Z}$
  end
end

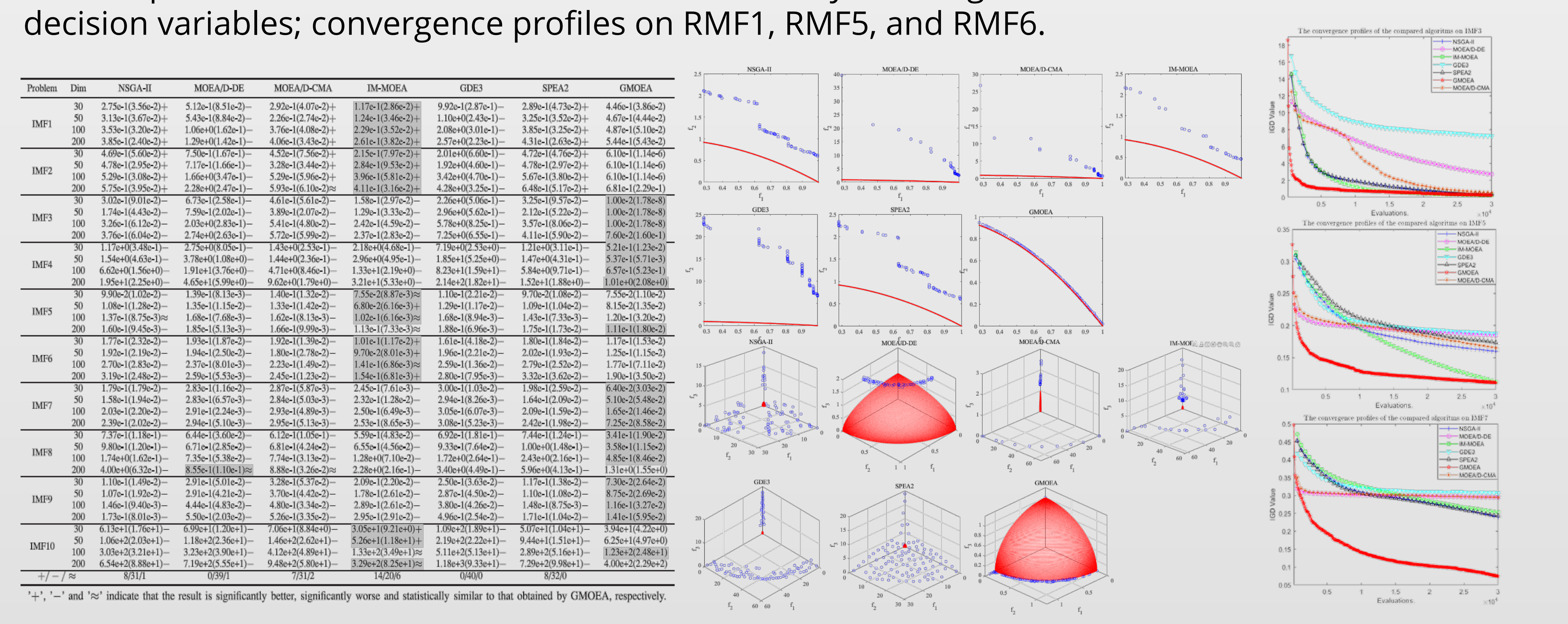
Results



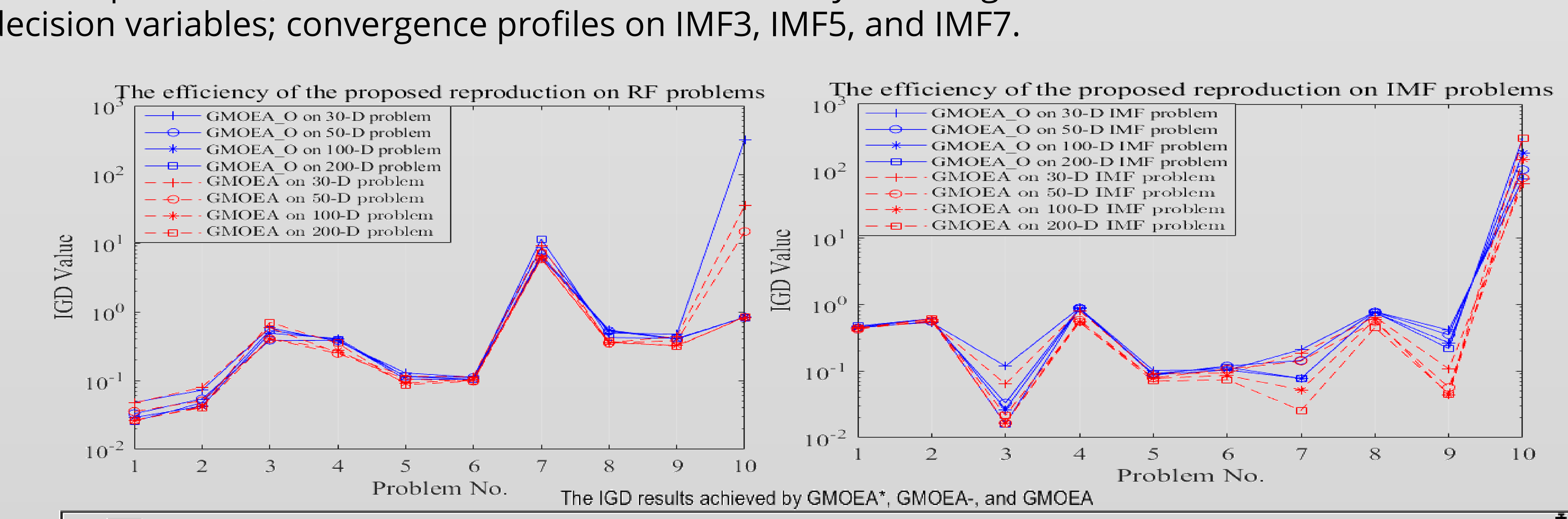
Effectiveness of the Model Training Method: Comparison on IMF1 and IMF4 with 200 decision variables using different GANs; the trajectories of generator and discriminator's training losses of two compared GANs.

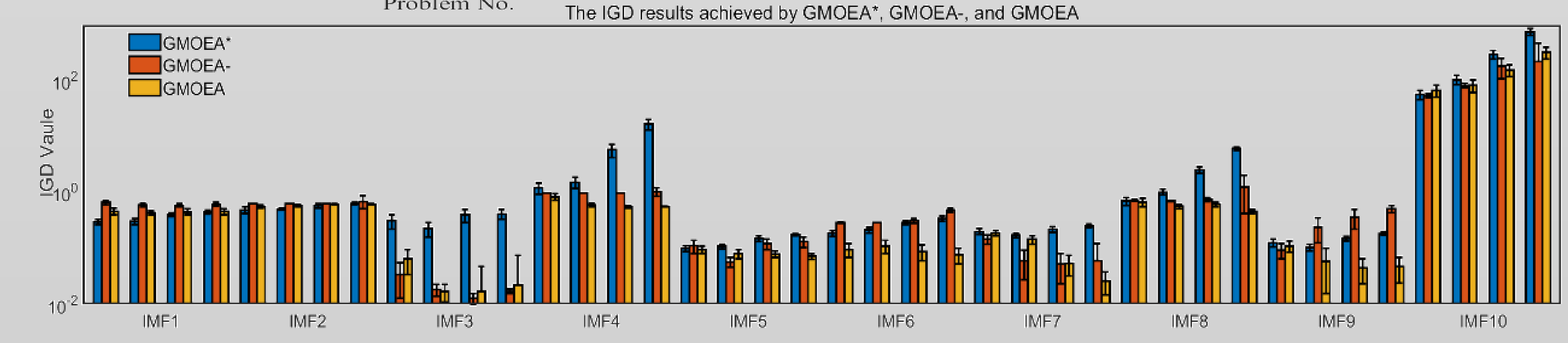


General Performance on RMF Problems: IGD results obtained by seven compared algorithms on 40 RMF problems; non-dominated fronts obtained by each algorithm on RMF2 and RMF4 with 200 decision variables; convergence profiles on RMF1, RMF5, and RMF6.



General Performance on IMF Problems: IGD results obtained by seven compared algorithms on 40 IMF problems; non-dominated fronts obtained by each algorithm on IMF3 and IMF8 with 200 decision variables; convergence profiles on IMF3, IMF5, and IMF7.





Effectiveness of Hybrid Offspring Generation: Comparisons between GMOEA_O (GMOEA without hybrid reproduction) and GMOEA on ten IMF problems with different number of decision variables. The statistics of IGD results achieved by GMOEA* (the reproduction with pure genetic operators), GMOEA- (the reproduction with pure GAN operator), and GMOEA (the reproduction with the hybrid strategy) on seven IMF problems with a number of 30, 50, 100, and 200 decision variables, respectively.

Conclusion

This article demonstrates that the MOEA driven by the GAN is promising in solving MOPs. Therefore, it deserves further efforts to introduce more efficient generative models. Besides, the extension of our proposed GMOEA to MOPs with more than three objectives (many-objective optimization problems) is highly desirable. Moreover, its applications to real-world optimization problems are also meaningful.

References

The source code is available athttps://www.chenghehust.com/assets/code/GMOEA_py.zip

@article{he2020evolutionary,
title={Evolutionary multiobjective optimization driven by generative adversarial networks (GANs)},
author={He, Cheng and Huang, Shihua and Cheng, Ran and Tan, Kay Chen and Jin, Yaochu},
journal={IEEE transactions on cybernetics},
volume={51},
number={6},
pages={3129--3142},
year={2020},
publisher={IEEE}
}