Results of the IEEE WCCI 2024 Competition on Multi-objective Neural Architecture Search

Zhenyu Liang, Yifan Zhao, Zhichao Lu, and **Ran Cheng**

The IEEE World Congress on Computational Intelligence

Yokohama, Japan, June 30 - July 5, 2023 contents

Motivation

- Neural Architecture Search (NAS) automates deep learning network design, improving performance in computer vision tasks.
- NAS for multiple design criteria is a challenging multi-objective optimization problems (MOPs) suitable for evolutionary multi-objective optimization (EMO) algorithms.
- Challenges in NAS include black-box optimization, discrete variables, noisy landscapes, and many objectives, etc.
- There is still **No** tailored NAS benchmark test suite for real-time semantic segmentation available.

Competition Entries

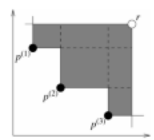
- 10 entries：5 new algorithms and 5 baseline algorithms
- The baseline algorithms were run on PlatEMO [3] and EvoX [4].

TABLE III: The information of competition entries

ID	Algorithm	Authors	Description	Label
1	DLEA	Gui Li and Guining Zhan	Dynamic Learning Evolutionary Algorithm	Participant
2	GrSMEA_NCHU	Chao He et al.	Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map	Participant
3	HypE [10]	Johannes Bader and Eckart Zitzler	An Algorithm for Fast Hypervolume-Based Many-Objective Optimization	Baseline
4	IBEA [9]	Eckart Zitzler and Simon Künzli	Indicator-Based Selection in Multi-objective Search	Baseline
5	IDEA_GNG	Bingsen Wang et al.	An improved Decomposition-based Multi-Objective Evolutionary Algorithm for Network Architecture Search	Participant
6	IMS-LOMONAS	Quan Minh Phan and Ngoc Hoang Luong	Pareto Local Search for Multi-objective Neural Architecture Search	Participant
7	MOEA/D [7]	Qingfu Zhang and Hui Li	A Multi-objective Evolutionary Algorithm Based on Decomposition	Baseline
8	MOEA-AP	Jiangtao Shen et al.	A Multiobjective Evolutionary Algorithm with Adaptive Simulated Binary Crossover and Pareto Front Modeling	Participant
9	NSGA-III [6]	Kalyanmoy Deb and Himanshu Jain	An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach	Baseline
10	RVEA [8]	Ran Cheng et al.	A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization	Baseline

Performance Indicators

- Hypervolume (HV)[11]



The HV in the two-objective case [10]

- The performance indicator that calculates HV are from pymoo [12].

Conclusion

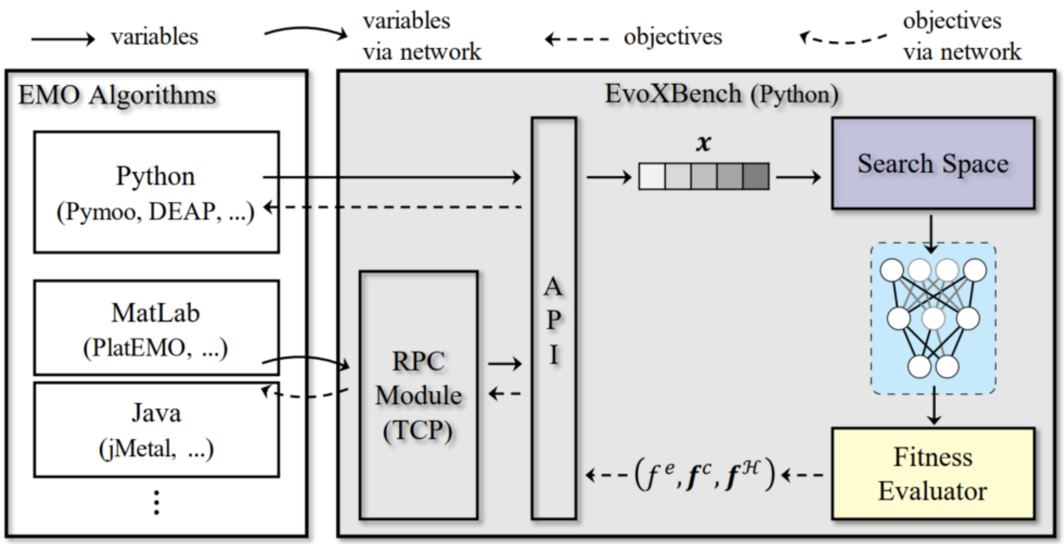
- According to the final ranking based on HV metrics, the winner is **GrSMEA_NCHU** (Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map).
- The competition showed the effectiveness of EMO algorithms in addressing multi-objective NAS tasks for real-time semantic segmentation.
- EMO algorithms offer a promising approach for automated network design in complex application scenarios such as autonomous driving.

Motivation

- Neural Architecture Search (NAS) automates deep learning network design, improving performance in computer vision tasks.
- NAS for multiple design criteria is a challenging multi-objective optimization problems (MOPs) suitable for evolutionary multi-objective optimization (EMO) algorithms.
- Challenges in NAS include black-box optimization, discrete variables, noisy landscapes, and many objectives, etc.
- There is still **No** tailored NAS benchmark test suite for real-time semantic segmentation available.

EvoXBench

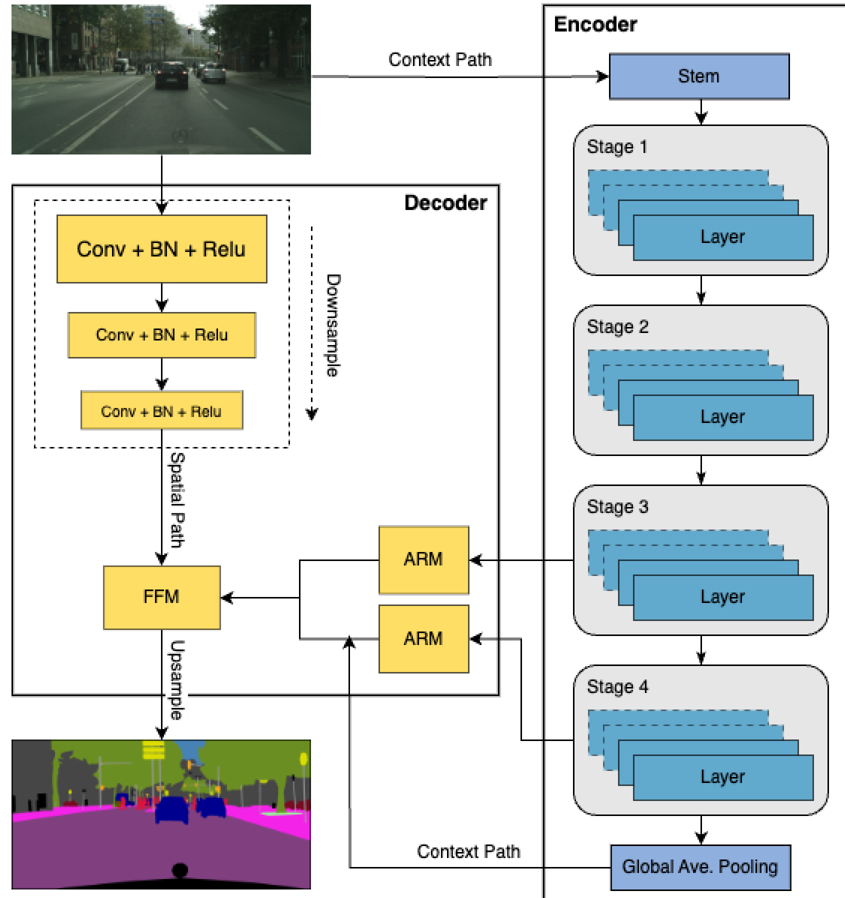
- Support multiple programming languages
- Run without the requirement of GPUs or PyTorch/Tensorflow
- Complete real-time semantic segmentation performance prediction for 100 different types of DNN architectures in 1 second.



EvoXBench [1] includes three benchmark test suites, one of which is the CitySeg/MOP, specifically designed for real-time semantic segmentation on the CityScapes dataset.

Real-time Semantic Segmentation

- Encoder-decoder or transformer structures are typically used for real-time semantic segmentation.
- Semantic segmentation needs both spatial and semantic information of input images.
- Real-time semantic segmentation has high requirements for both segmentation accuracy and inference efficiency.



Test Suite

TABLE I: Definition of the CitySeg/MOP [2] test suite

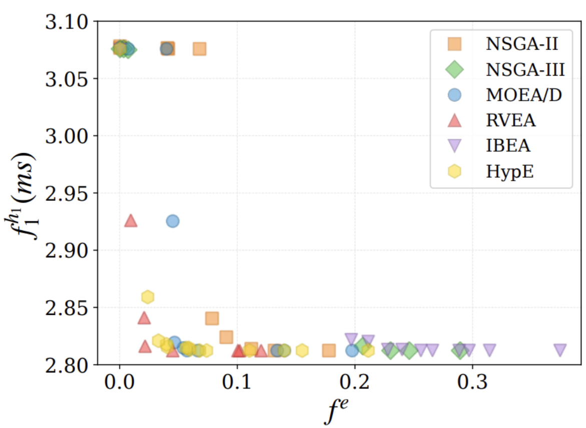
Problems	D	M	Objectives
CitySeg/MOP1	32	2	$f^e, f_1^{h_1}$
CitySeg/MOP2	32	3	$f^e, f_1^{h_1}, f_1^c$
CitySeg/MOP3	32	3	$f^e, f_1^{h_1}, f_2^c$
CitySeg/MOP4	32	4	$f^e, f_1^{h_1}, f_2^{h_1}, f_1^c$
CitySeg/MOP5	32	5	$f^e, f_1^{h_1}, f_2^{h_1}, f_1^c, f_2^c$
CitySeg/MOP6	32	2	$f^e, f_1^{h_2}$
CitySeg/MOP7	32	3	$f^e, f_1^{h_2}, f_1^c$
CitySeg/MOP8	32	3	$f^e, f_1^{h_2}, f_2^c$
CitySeg/MOP9	32	4	$f^e, f_1^{h_2}, f_2^{h_2}, f_1^c$
CitySeg/MOP10	32	5	$f^e, f_1^{h_2}, f_2^{h_2}, f_1^c, f_2^c$
CitySeg/MOP11	32	3	$f^e, f_1^{h_1}, f_1^{h_2}$
CitySeg/MOP12	32	5	$f^e, f_1^{h_1}, f_1^{h_2}, f_2^{h_1}, f_2^{h_2}$
CitySeg/MOP13	32	6	$f^e, f_1^{h_1}, f_1^{h_2}, f_2^{h_1}, f_2^{h_2}, f_1^c$
CitySeg/MOP14	32	6	$f^e, f_1^{h_1}, f_1^{h_2}, f_2^{h_1}, f_2^{h_2}, f_2^c$
CitySeg/MOP15	32	7	$f^e, f_1^{h_1}, f_1^{h_2}, f_2^{h_1}, f_2^{h_2}, f_1^c, f_2^c$

TABLE II: Definition of objectives in proposed CitySeg/MOP test suite

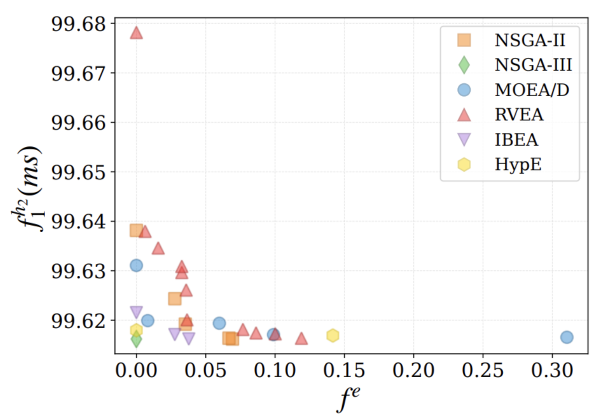
Objectives	Definition
f^e	prediction error
$f_1^{h_1}$	h_1's inference latency
$f_1^{h_2}$	h_2's inference latency
$f_2^{h_1}$	h_1's inference energy consumption
$f_2^{h_2}$	h_2's inference energy consumption
f_1^c	# of floating point operations
f_2^c	# of parameters/weights

Result of CitySeg/MOP1, 6, 15

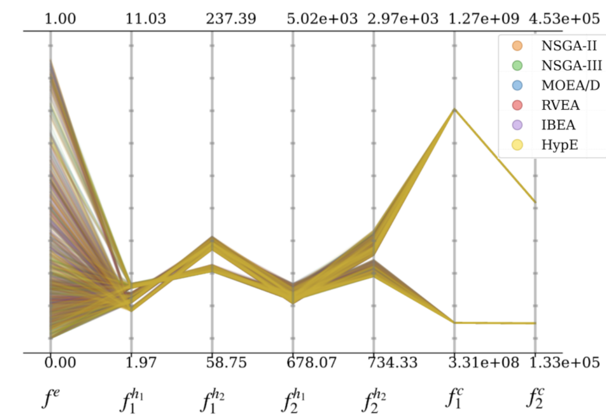
- Non-dominated solutions obtained by six representative algorithms (NSGA-II [5], NSGA-III [6], MOEA/D [7], RVEA [8], IBEA [9], and HypE [10]) on CitySeg/MOP1, CitySeg/MOP6, and CitySeg/MOP15.



(a) Result of CitySeg/MOP1.



(b) Result of CitySeg/MOP6.



(c) Result of CitySeg/MOP15.

Competition Entries

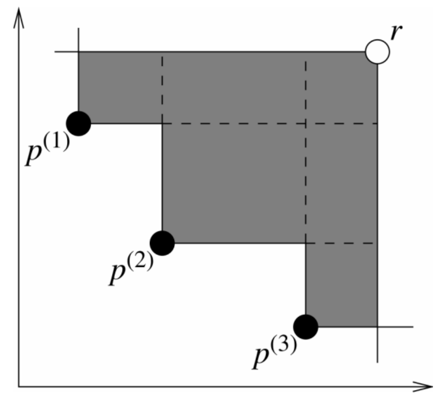
- 10 entries： 5 new algorithms and 5 baseline algorithms
- The baseline algorithms were run on PlatEMO [3] and EvoX [4].

TABLE III: The information of competition entries

ID	Algorithm	Authors	Description	Label
1	DLEA	Gui Li and Guining Zhan	Dynamic Learning Evolutionary Algorithm	Participant
2	GrSMEA_NCHU	Chao He et al.	Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map	Participant
3	HypE [10]	Johannes Bader and Eckart Zitzler	An Algorithm for Fast Hypervolume-Based Many-Objective Optimization	Baseline
4	IBEA [9]	Eckart Zitzler and Simon Künzli	Indicator-Based Selection in Multi-objective Search	Baseline
5	IDEA_GNG	Bingsen Wang et al.	An improved Decomposition-based Multi-Objective Evolutionary Algorithm for Network Architecture Search	Participant
6	IMS-LOMONAS	Quan Minh Phan and Ngoc Hoang Luong	Pareto Local Search for Multi-objective Neural Architecture Search	Participant
7	MOEA/D [7]	Qingfu Zhang and Hui Li	A Multi-objective Evolutionary Algorithm Based on Decomposition	Baseline
8	MOEA-AP	Jiangtao Shen et al.	A Multiobjective Evolutionary Algorithm with Adaptive Simulated Binary Crossover and Pareto Front Modeling	Participant
9	NSGA-III [6]	Kalyanmoy Deb and Himanshu Jain	An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach	Baseline
10	RVEA [8]	Ran Cheng et al.	A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization	Baseline

Performance Indicators

- Hypervolume (HV)[11]



The HV in the two-objective case [10]

- The performance indicator that calculates HV are from pymoo [12].

Ranking Strategy

- For each problem, each algorithm is executed 31 times and the HV is calculated.
- The mean value and standard deviation of HV is calculated for each problem and sorted.
- The statistical results are compared using Wilcoxon rank sum test.
- The score for each algorithm is determined by summing up its ranks across all problems and then normalizing the result.

Overview of the Results

TABLE IV: Statistical results (mean and standard deviation) of the HV values on CitySeg/MOP test suite. The best results of each instance are in bold red.

	DLEA	GrSMEA_NCHU	HypE	IBEA	I-DEA-GNG	IMS-LOMONAS	MOEA/D	MOEA-PA	NSGA-III	RVEA
CitySeg/MOP1	**0.9001 (0.0016)≈**	**0.9002 (0.0016)≈**	**0.8967 (0.0109)≈**	**0.8990 (0.0042)≈**	0.8989 (0.0010)-	**0.9001 (0.0018)≈**	0.8423 (0.0497)-	**0.9007 (0.0017)≈**	**0.8983 (0.0070)≈**	0.8683 (0.0225)-
CitySeg/MOP2	0.7987 (0.0012)-	**0.7994 (0.0010)≈**	0.7949 (0.0049)-	0.7739 (0.0171)-	**0.7989 (0.0014)≈**	0.7979 (0.0016)-	0.7492 (0.0408)-	**0.7993 (0.0021)≈**	**0.7991 (0.0019)≈**	0.6976 (0.0694)-
CitySeg/MOP3	0.8224 (0.0028)-	**0.8235 (0.0023)≈**	0.8134 (0.0178)-	0.7830 (0.0075)-	0.8226 (0.0029)-	0.8223 (0.0016)-	0.7582 (0.0328)-	**0.8217 (0.0045)≈**	**0.8229 (0.0035)≈**	0.7813 (0.0315)-
CitySeg/MOP4	0.6974 (0.0016)-	0.6981 (0.0006)-	0.6979 (0.0014)-	0.6006 (0.0495)-	0.6975 (0.0009)-	**0.6986 (0.0004)≈**	0.5474 (0.0840)-	**0.6983 (0.0007)≈**	**0.6983 (0.0006)≈**	0.5951 (0.0684)-
CitySeg/MOP5	0.6560 (0.0009)-	**0.6561 (0.0010)≈**	**0.6563 (0.0006)≈**	0.5612 (0.0413)-	0.6553 (0.0011)-	**0.6564 (0.0004)≈**	0.5576 (0.0443)-	**0.6561 (0.0010)≈**	**0.6565 (0.0006)≈**	0.5315 (0.1099)-
CitySeg/MOP6	**0.7719 (0.0003)≈**	**0.7719 (0.0002)≈**	**0.7706 (0.0072)≈**	**0.7692 (0.0089)≈**	**0.7719 (0.0003)≈**	0.7678 (0.0021)-	0.7107 (0.0532)-	**0.7712 (0.0038)≈**	0.7718 (0.0003)-	0.7367 (0.0223)-
CitySeg/MOP7	**0.7312 (0.0001)≈**	**0.7312 (0.0002)≈**	**0.7303 (0.0029)≈**	0.7090 (0.0334)-	**0.7311 (0.0004)≈**	0.7277 (0.0029)-	0.6745 (0.0480)-	**0.7311 (0.0002)≈**	**0.7312 (0.0002)≈**	0.6810 (0.0293)-
CitySeg/MOP8	0.7324 (0.0002)-	**0.7325 (0.0002)≈**	0.7284 (0.0126)-	0.7225 (0.0219)-	0.7322 (0.0004)-	0.7282 (0.0031)-	0.6843 (0.0336)-	0.7323 (0.0003)-	**0.7324 (0.0002)≈**	0.6889 (0.0309)-
CitySeg/MOP9	**0.5767 (0.0004)≈**	**0.5768 (0.0003)≈**	**0.5768 (0.0004)≈**	0.4923 (0.0403)-	0.5765 (0.0005)-	0.5760 (0.0005)-	0.4804 (0.0367)-	**0.5767 (0.0003)≈**	**0.5768 (0.0003)≈**	0.5405 (0.0349)-
CitySeg/MOP10	0.5472 (0.0003)-	**0.5473 (0.0003)≈**	**0.5472 (0.0004)≈**	0.4726 (0.0310)-	0.5469 (0.0006)-	0.5465 (0.0005)-	0.4606 (0.0380)-	0.5472 (0.0004)-	**0.5473 (0.0003)≈**	0.5138 (0.0320)-
CitySeg/MOP11	0.6860 (0.0051)-	0.6883 (0.0012)-	0.6840 (0.0096)-	0.6634 (0.0209)-	0.6878 (0.0014)-	**0.6904 (0.0006)+**	0.5699 (0.0345)-	0.6888 (0.0014)-	0.6884 (0.0009)-	0.6620 (0.0353)-
CitySeg/MOP12	0.4718 (0.0011)-	0.4722 (0.0014)-	0.4527 (0.0169)-	0.3889 (0.0279)-	0.4715 (0.0019)-	**0.4760 (0.0004)+**	0.3558 (0.0641)-	0.4719 (0.0011)-	0.4705 (0.0012)-	0.4358 (0.0289)-
CitySeg/MOP13	0.4219 (0.0012)-	0.4224 (0.0011)-	0.4150 (0.0091)-	0.3503 (0.0343)-	0.4216 (0.0013)-	**0.4244 (0.0002)+**	0.3459 (0.0208)-	0.4227 (0.0004)-	0.4228 (0.0006)-	0.3772 (0.0213)-
CitySeg/MOP14	0.4337 (0.0009)-	0.4338 (0.0010)-	0.4214 (0.0154)-	0.3569 (0.0236)-	0.4336 (0.0015)-	**0.4366 (0.0003)+**	0.3406 (0.0197)-	0.4324 (0.0008)-	0.4339 (0.0006)-	0.4118 (0.0171)-
CitySeg/MOP15	0.3981 (0.0007)-	0.3984 (0.0006)-	0.3956 (0.0044)-	0.3376 (0.0189)-	0.3984 (0.0006)-	**0.3993 (0.0002)+**	0.3272 (0.0261)-	0.3986 (0.0003)-	0.3980 (0.0024)-	0.3003 (0.0706)-

\+ indicates a method achieving significantly better performance.

≈ indicates a method achieving similar performance as the best-performing method.

− indicates a method achieving significantly worse performance.

Overview of the Ranks

TABLE V: Ranks according to HV values

	DLEA	GrSMEA_NCHU	HypE	IBEA	IDEA-GNG	IMS-LOMONAS	MOEA/D	MOEA-AP	NSGA-III	RVEA
CitySeg/MOP1	1	1	1	1	8	1	10	1	1	9
CitySeg/MOP2	5	1	7	8	1	6	9	1	1	10
CitySeg/MOP3	5	1	7	8	4	6	10	1	1	9
CitySeg/MOP4	7	4	5	8	6	1	10	1	1	9
CitySeg/MOP5	6	1	1	8	7	1	9	1	1	10
CitySeg/MOP6	1	1	1	1	1	8	10	1	7	9
CitySeg/MOP7	1	1	1	8	1	7	10	1	1	9
CitySeg/MOP8	3	1	6	8	5	7	10	4	1	9
CitySeg/MOP9	1	1	1	9	6	7	10	5	1	8
CitySeg/MOP10	4	1	1	9	6	7	10	4	1	8
CitySeg/MOP11	6	4	7	8	5	1	10	2	3	9
CitySeg/MOP12	4	2	7	9	5	1	10	3	6	8
CitySeg/MOP13	5	4	7	9	6	1	10	3	2	8
CitySeg/MOP14	4	3	7	9	5	1	10	6	2	8
CitySeg/MOP15	5	3	7	8	3	1	9	2	6	10
Add-up Scores	58	29	66	111	69	56	147	36	35	133
Final Rank	5	1	6	8	7	4	10	3	2	9

Winner Algorithms

GrSMEA_NCHU

Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map

Authors: Chao He, Congxuan Zhang, Ming Li, Hao Chen, and Zige Wang

Affiliations: Nanchang Hangkong University, Nanchang, China

MOEA-AP

A Multiobjective Evolutionary Algorithm with Adaptive Simulated Binary Crossover and Pareto Front Modeling

Authors: Jiangtao Shen, Junchang Liu, Huachao Dong, Xinjing Wang, and Peng Wang

Affiliations: Northwestern Polytechnical University, Xi’an, China

IMS-LOMONAS

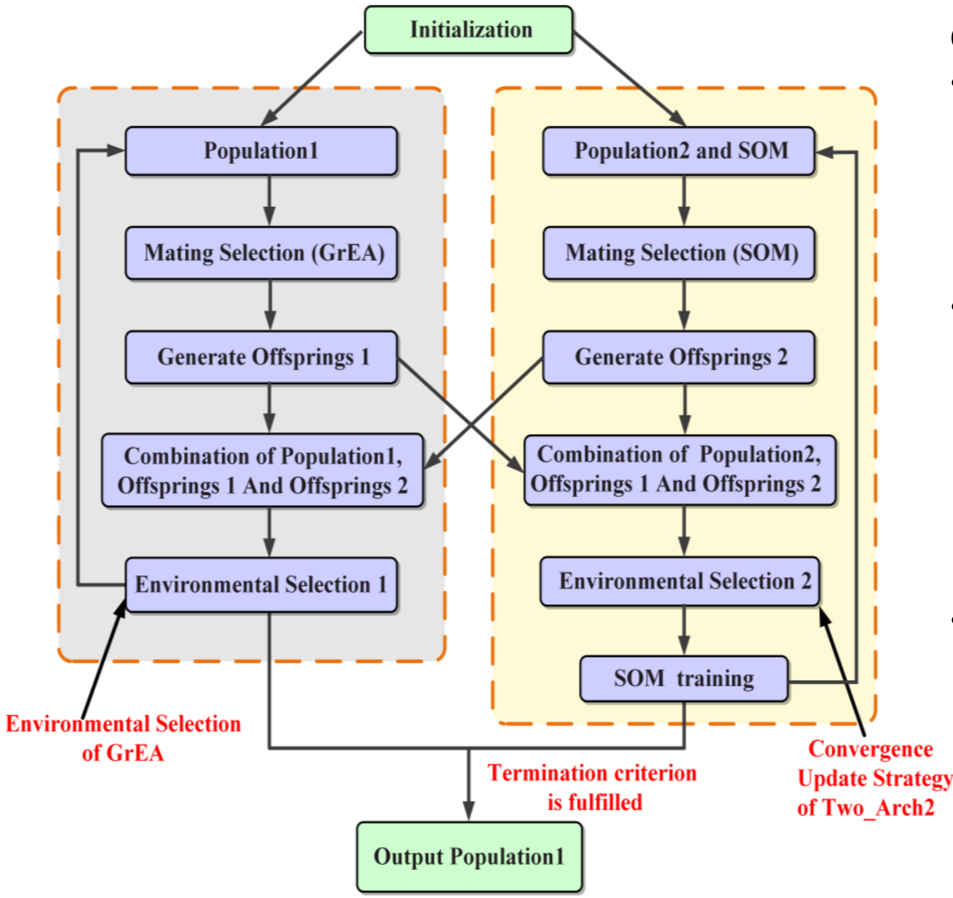
Pareto Local Search for Multi-objective Neural Architecture Search

Authors: Quan Minh Phan and Ngoc Hoang Luong

Affiliations: University of Information Technology, Ho Chi Minh City, Vietnam; Vietnam National University, Ho Chi Minh City, Vietnam

GrSMEA_NCHU

Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map



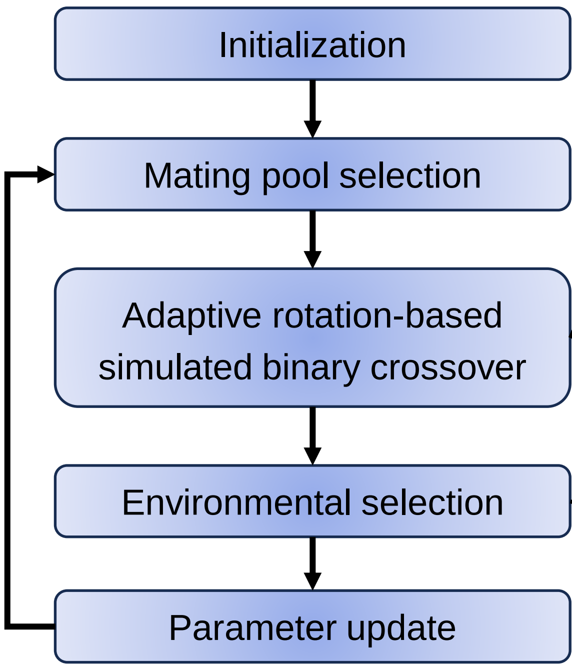
Our method consists of three major procedures:

- (1) **Two collaborative and complementary populations**: Population 1 (P1) focuses on promoting population diversity. Different from P1, Population 2 (P2) complements P1 by exploring unexploited areas.
- (2) **Constructing the mating pool and reproduction**: The mating pool of P1 is selected using the mating selection strategy of GrEA to generate offspring. For P2, utilize the neighborhood relationship information extracted by the SOM to assist in constructing the mating pool and generating the offspring.
- (3) **Environmental selection**: P1 uses the environmental selection strategy of GrEA to select solutions for the next generation. The convergence update strategy of Two_Arch2 is employed as the selection principle for P2 to improve convergence in NAS problems.

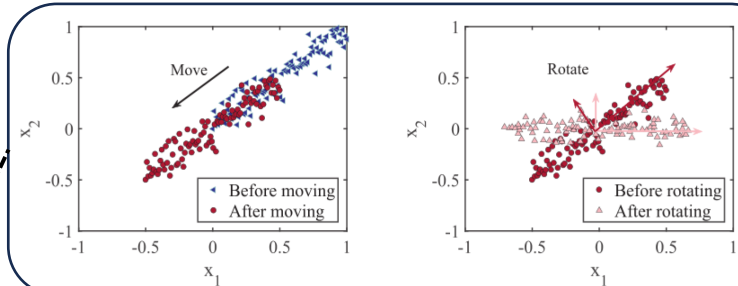
MOEA-AP

A Multiobjective Evolutionary Algorithm with Adaptive Simulated Binary Crossover and Pareto Front Modeling

Flowchart:

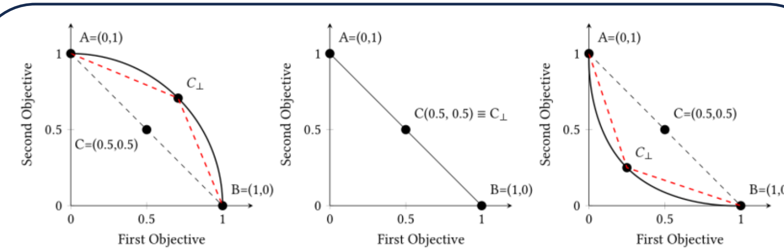


ARSBX



Generate solutions with good exploration and exploitation by moving and rotating in the **decision space**

Selection based on Pareto front modeling



$$gd(A, B) = ||A - C_\perp||_2 + ||B - C_\perp||_2$$

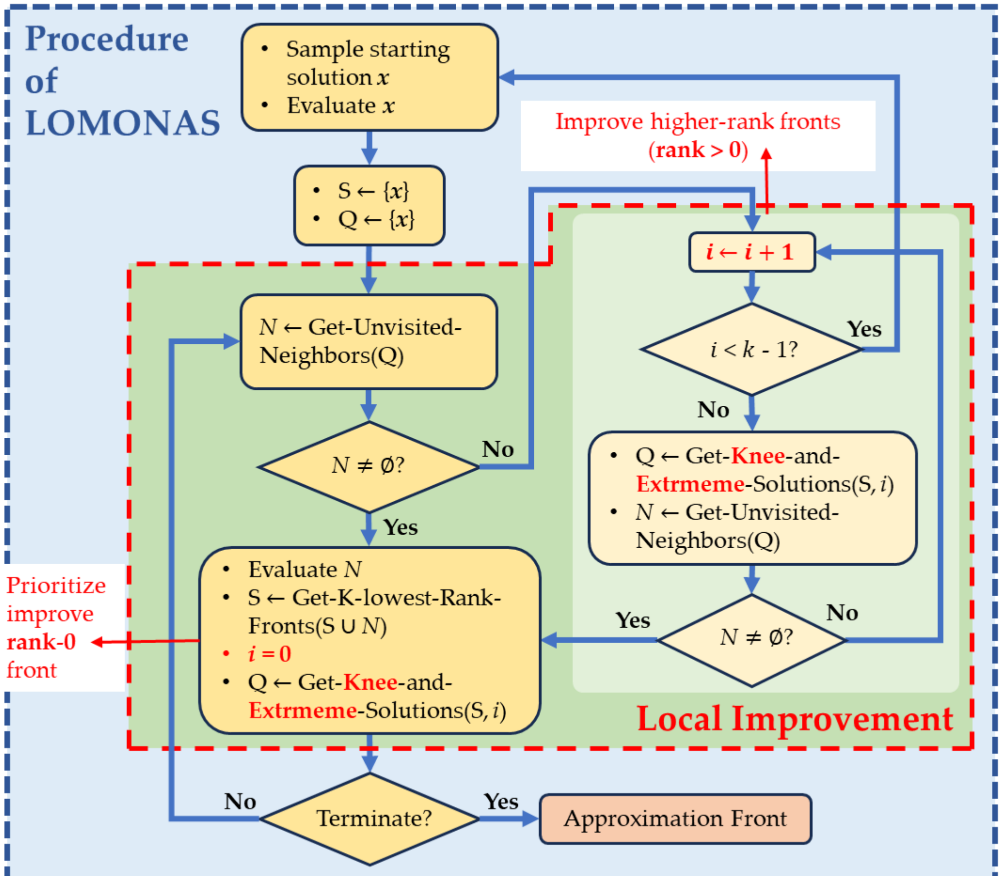
Survival score calculate:

$$geodesic\_div(A, \mathbb{F}_1) = \min_{B_\perp \neq A_\perp \in D} gd\,(A_\perp - B_\perp)$$

Select solutions with good convergence and diversity by Pareto front modeling in the **objective space**

IMS-LOMONAS: Parameter-less Pareto Local Search for MONAS

Quan Minh Phan, Ngoc Hoang Luong (University of Information Technology, VNU-HCM)



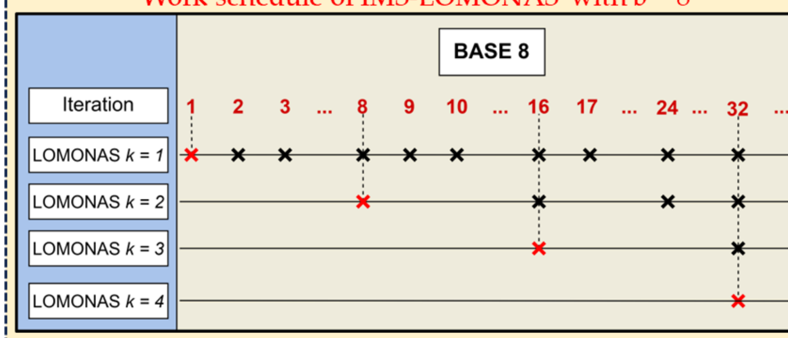
LOMONAS:

- LOMONAS is Iterated Pareto Local Search.
- Rank solutions by using Non-dominated Sorting and improve ***k*** fronts (from rank-0 to rank-(k-1) fronts) instead of only rank-0 front (non-dominated front).
- Only improve **knee** and **extreme** solutions.

IMS-LOMONAS:

- Remove the manual setting of hyparameter k in LOMONAS by using Interleaved Multi-start Scheme (IMS).
- Perform multiple LOMONAS variants with different values of k.
- Initializing new instance or executing previous LOMONAS instances is scheduled following a counter of base **b.**

Work schedule of IMS-LOMONAS with b = 8



✖ **Initialize** new instance and **perform** local improvement till #Evals equals to $1/\log_2(b)$ #Evals of the previous instance.

✕ **Perform** one local improvement

Conclusion

- According to the final ranking based on HV metrics, the winner is **GrSMEA_NCHU** (Grid-Based Evolutionary Algorithm Assisted by Self-Organizing Map).
- The competition showed the effectiveness of EMO algorithms in addressing multi-objective NAS tasks for real-time semantic segmentation.
- EMO algorithms offer a promising approach for automated network design in complex application scenarios such as autonomous driving.

Future Extension Directions of EvoXBench

- Enhanced EMO algorithm performance: Investigate novel algorithms that can handle the challenges of NAS.
- More Benchmarking: Consider expanding the range of search spaces, datasets, and hardware configurations.
- Transferability: Explore techniques for transferring knowledge learned from one NAS task to another, reducing the search space and computational costs.

Q & A

Zhenyu Liang (zhenyuliang97@gmail.com)
Yifan Zhao (AlpAcA0072@gmail.com)
Zhichao Lu (luzhichaocn@gmail.com)
Ran Cheng (ranchengcn@gmail.com)

Competition Homepage: https://www.emigroup.tech/index.php/news/ieee-cec-2024-competition-on-multiobjective-neural-architecture-search/

Github: https://github.com/EMI-Group/IEEE-CEC-NAS-Competition

Github Repository

References

[1] Z. Lu, R. Cheng, Y. Jin, K. C. Tan, and K. Deb, "Neural Architecture Search as Multiobjective Optimization Benchmarks: Problem Formulation and Performance Assessment," *IEEE Transactions on Evolutionary Computation*, 2023.
[2] Y. Zhao, Z. Liang, Z. Lu, and R. Cheng, "A Multi-objective Optimization Benchmark Test Suite for Real-time Semantic Segmentation," in *Proceedings of the Genetic and Evolutionary Computation Conference* (*GECCO*), 2024.
[3] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 73-87, 2017.
[4] B. Huang, R. Cheng, Z. Li, Y. Jin, and K. C. Tan, "EvoX: A Distributed GPU-accelerated Framework for Scalable Evolutionary Computation," *IEEE Transactions on Evolutionary Computation*, 2024.
[5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182-197, 2002.
[6] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577-601, 2014.
[7] Z. Qingfu and L. Hui, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712-731, 2007.
[8] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773-791, 2016.
[9] E. Zitzler and S. Küenzli, "Indicator-Based Selection in Multiobjective Search," in *Parallel Problem Solving from Nature - PPSN VIII*. Springer Berlin Heidelberg, 2004, pp. 832–842.
[10] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45-76, 2011.
[11] C. M. Fonseca, L. Paquete, and M. López-Ibánez, "An improved dimension-sweep algorithm for the hypervolume indicator," in *2006 IEEE international conference on evolutionary computation*, 2006: IEEE, pp. 1157-1163.
[12] J. Blank and K. Deb, "Pymoo: Multi-objective optimization in python," *IEEE Access*, vol. 8, pp. 89497-89509, 2020.