To spend **more effort** on regions with **lower diversity** improves MOEA/D

Using Diversity as a Priority Function for Resource Allocation on MOEA/D

Yuri Lavinas, Claus Aranha, Tetsuya Sakurai

1 Intro

- How to distribute effort given subproblem's difficulty?
- We estimate difficulty using priority functions.
- We give more effort on harder subproblems.

2 Methods

- 2-Norm vs. R.I. vs Random vs standard MOEA/D
- UF functions set (100 dim), 21 repetitions, 70000 function evaluations, population size = 350
- Analysis: IGD and proportion of non-dominated solutions

3 Results

IGD	None	Norm	R.I.	Random
UF1	0.140 (0.013)	0.109 (0.016)	**0.090 (0.012)**	0.093 (0.014)
UF2	0.082 (0.006)	0.060 (0.005)	0.060 (0.005)	**0.060 (0.004)**
UF3	0.260 (0.012)	**0.168 (0.025)**	0.183 (0.335)	0.214 (0.030)
UF4	0.100 (0.023)	**0.095 (0.002)**	0.095 (0.003)	**0.095 (0.002)**
UF5	1.759 (0.080)	**0.972 (0.056)**	1.056 (0.064)	1.085 (0.073)
UF6	0.121 (0.027)	0.100 (0.016)	**0.078 (0.014)**	0.079 (0.016)
UF7	0.125 (0.018)	**0.061 (0.006)**	0.068 (0.005)	0.074 (0.005)
UF8	0.286 (0.012)	**0.229 (0.014)**	0.257 (0.020)	0.232 (0.006)
UF9	0.451 (0.012)	**0.385 (0.020)**	0.420 (0.017)	0.400 (0.018)
UF10	3.693 (0.200)	2.380 (0.241)	**2.364 (0.272)**	2.639 (0.253)
Non-dominated	None	Norm	R.I.	Random
UF	0.34 (0.04)	**0.84 (0.06)**	0.58 (0.10)	0.69 (0.05)

- In 6 out of 10 functions 2-Norm had better results: lower IGD and higher rate of non-dominated
- 2-Norm as priority function effectively improves the performance of MOEA/D

Extra figures

Basic MOEA/D framework with only priority functions and no other variant. This algorithm is similar to the MOEA/D-DE with exception of lines 4 and 7.

Algorithm 1 MOEA/D with priority functions

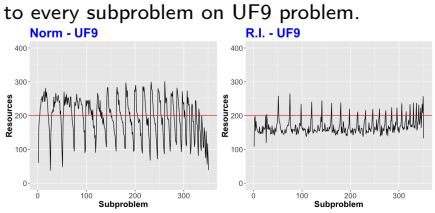
1: Initialize the weight vectors λ_i, the neighborhood B_i, the priority value u_i every subproblem $i = 1, ..., N$.
2: **while** *Termination criteria* **do**
3: **for** 1 to N **do**
4: **if** *rand()* $< u_i$ **then**
5: Generate an offspring y for subproblem i.
6: Update the population by y.
7: Evaluate and after ΔT generations, keep updating $\boldsymbol{u}$ by a priority function.

Norm of the difference of current solutions and its parents.

Algorithm 1 2-Norm

1: Input: X^t decision vectors of solutions; X^{t-1}, decision vectors from the previous solutions; N, the population size.
2: **for** i=1 to N **do**
3: u[i] = $||X_i^t - X_i^{t-1}||$
4: u = scale (u) // between 0 and 1
5: **return** u

Amount of resources allocated by Norm, R.I to every subproblem on UF9 problem.



- Norm as priority function forces solutions that are similar to update more often, leading to a higher exploration
- To give higher priority for regions with lower diversity the algorithm spends more effort in regions that are not yet well explored

Scan me

← Download the paper

筑波大学
University of Tsukuba