M2 AIC

TC2: Introduction to Optimization

Black-Box Optimization Benchmarking with the COCO platform - Multiobjective Optimizer adaptive IBEA (ϵ-indicator)

Group 1: Martin BAUW, Robin DURAZ, Jiaxin GAO,
Hao LIU, Luca VEYRON-FORRER

1. Introduction
2. The algorithm
   - Overview of IBEA
   - Selection and variation
3. Our implementation
   - Code structure
   - Improvements regarding the execution speed
4. CPU timing and results
   - CPU timing and results
   - Comparision with NSGA 2 and Random Search
5. Conclusion
6. Bibliography

- steps description

Binary quality indicators

Recombination

For recombination, we use a Simulated Binary Crossover (SBX) operator. A uniform probability pick in $[0,1]$ written u determines the parameter used in computing the features (decision space coordinates) of the children.

- if the uniform probability pick ≤ 0.5:

$$\beta_q = (2u)^{\frac{1}{\mu+1}} \tag{1}$$

- else:

$$\beta_q = (\frac{1}{2(1-u)})^{\frac{1}{\mu+1}} \tag{2}$$

Recombination

Thus, we can compute the children's coordinates:

- first child:

$$child0[j] = 0.5((1+\beta_q)parent0[j] + (1-\beta_q)parent1[j]) \quad (3)$$

- second child:

$$child1[j] = 0.5((1-\beta_q)parent0[j] + (1+\beta_q)parent1[j]) \quad (4)$$

Mating selection and mutation

Polynomial mutation operator:
The mutation operator modifies individuals by changing small parts in the associated vectors according to a given mutation rate.

- if the uniform probability pick ≤ 0.5:

$$\sigma_L = (2u)^{\frac{1}{\mu+1}} - 1 \tag{5}$$

$$p_{mut}[j] = ind[j] + \sigma_L(ind[j] - Lo) \tag{6}$$

- else:

$$\sigma_R = (2(1-u))^{\frac{1}{\mu+1}} \tag{7}$$

$$p_{mut}[j] = ind[j] + \sigma_R(Up - ind[j]) \tag{8}$$

Computer specifications and batch options

- Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
- Quad core CPU with 16GB RAM

Everything ran with a budget of 100

- Three batchs for dimensions 2, 3, 5, 10, 20
- First batch running alone, and two others together
- One batch for dimensions 40

Options chosen to run the algorithm

- Population size : 100
- Maximum number of generation : 100
- Scaling factor : 0.05
- Mutation rate : 0.01
- Recombination and mutation mu : 1
- Population initialization in range (-5, 5)

Dimension Batch	2	3	5
Batch 1 on 3	6.0e-04	6.3e-04	8.1e-04
Batch 2 and 3 on 3 run simultaneously	8.6e-04	8.6e-04	9.1e-04
	8.3e-04	8.4e-04	8.9e-04

Dimension Batch	10	20
Batch 1 on 3	8.3e-04	1.1e-03
Batch 2 and 3 on 3 run simultaneously	1.1e-03	1.3e-03
	1.0e-03	1.3e-03

Dimension Batch	40
Whole test suite	4.2e-03

Results

Results analysis

- Comparatively better in higher dimensions
- Results globally good for an EMOA

- More budget could have given better results
- A better initialization of population could lead to a sharper increase at the beginning

NSGA 2

Random Search

In our experiments, we majorly focused on the comparison with NSGA-II and Random search:

IBEA VS Random search: IBEA outperformed Random search, a relatively good Pareto set approximation was given by IBEA.
IBEA VS NSGA-II: IBEA performed worse than NSGA-II.

Choosing a representation of the problem addressed, an initial population, a method of selection, a crossover operator, mutation operator, the probabilities of crossover and mutation, and the insertion method creates a variant of MOEAs algorithms.

Non-exhaustive bibliography