INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

ParadisEO-MOEO for a Bi-objective Flow-Shop Scheduling Problem

July 2007

E.-G. Talbi and the ParadisEO team

http://paradiseo.gforge.inria.fr
paradiseo-help@lists.gforge.inria.fr

Framework and Tutorial Application

- A framework for the design of metaheuristics for multi-objective optimization (mainly evolutionary algorithms)

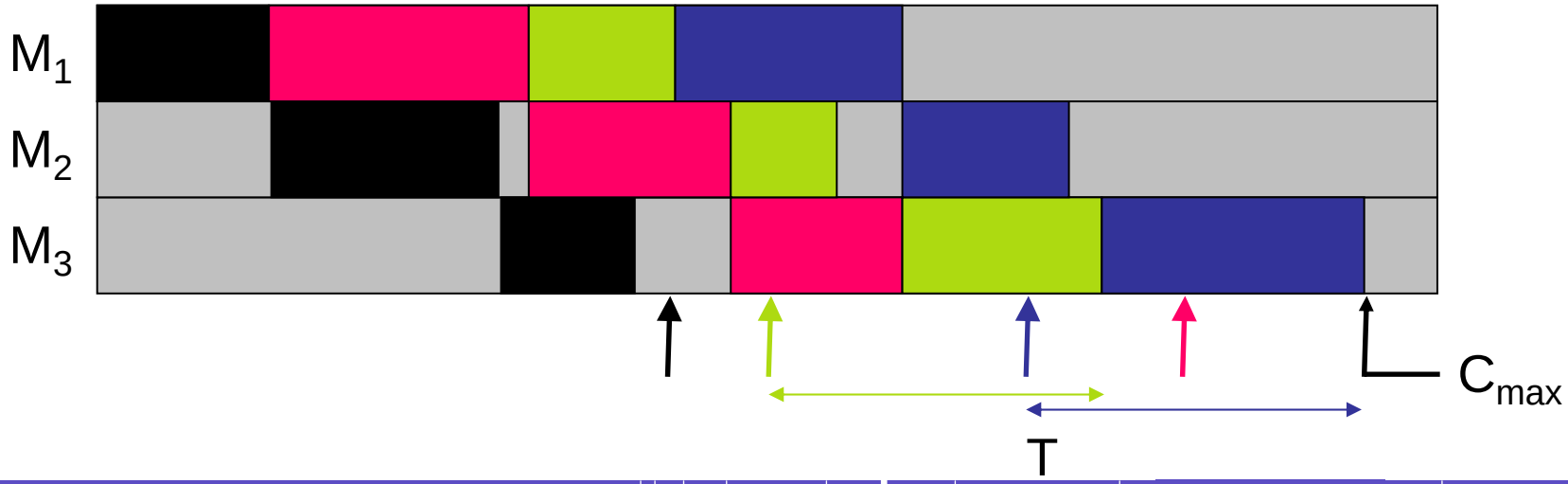
➔ ParadisEO-MOEO (Multi-Objective Evolving Objects)

- Tutorial application

➔ A bi-objective flow-shop scheduling problem

Flow-shop Scheduling Problem

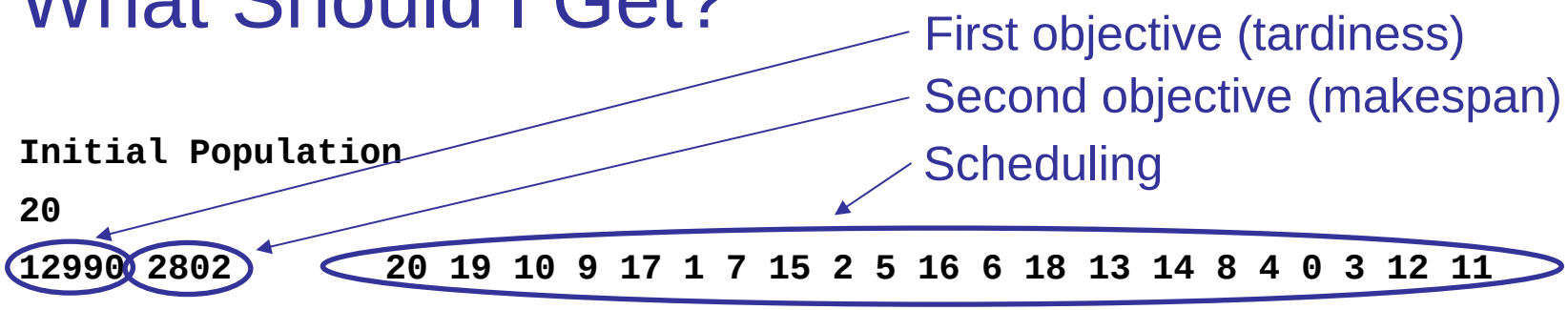
- N jobs to schedule on M machines
- Machines are critical resources
- 2 objectives to optimize (minimize)
  - Makespan (C_{max})
  - Total tardiness (T)



Getting Started

- Go to the first ParadisEO-MOEO's lessons directory
  - > cd paradiseo-moeo/tutorial/Lesson1/
  - *FlowShopEA.cpp*
    - Main file
  - > cd paradiseo-moeo/examples/flowshop/
  - Component files (**.h, *.cpp*)
  - *benchs/**
    - Benchmark instances
- Run the EA
  - > cd paradiseo-moeo/build/tutorial/Lesson1/
  - > ./FlowShopEA @FlowShopEA.param

What Should I Get?




```
Initial Population
20
12990 2802        20 19 10 9 17 1 7 15 2 5 16 6 18 13 14 8 4 0 3 12 11
10496 2732        20 19 2 11 17 14 12 18 5 7 8 1 6 3 10 9 4 0 13 15 16
…
Final Population
20
7184 2453         20 2 15 13 6 8 5 12 0 17 14 19 3 16 9 18 10 1 4 11 7
7191 2482         20 2 15 13 6 8 5 12 0 17 14 19 3 16 9 4 18 10 1 11 7
…
Final Archive
5
7137 2495         20 2 15 13 8 5 6 11 12 0 17 14 19 3 16 9 4 18 10 1 7
7184 2453         20 2 15 13 6 8 5 12 0 17 14 19 3 16 9 18 10 1 4 11 7
…
```

Benchmarks for Flow-shop

- Filename: N_M_i.txt
  - N
    - Number of jobs
  - M
    - Number of machines
  - i
    - index

- www.lifl.fr/~liefooga/benchmarks/

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Representation

- In the objective space (objective vector)

➔ FlowShopObjectiveVectorTraits

  - Defines the number of objectives and if they have to be minimized or maximized

➔ FlowShopObjectiveVector

```
typedef moeoRealObjectiveVector<FlowShopObjectiveVectorTraits>
  FlowShopObjectiveVector;
```

- In the decision space

➔ FlowShop

```
class FlowShop:
  public moeoVector<FlowShopObjectiveVector,double,double,unsigned int>
```

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization ➔ FlowShopInit
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Evaluation Function

➔ FlowShopEval

- Evaluation in the objective space

```
// creation of an objective vector object
FlowShopObjectiveVector objVector;
// computation of objective values
objVector[0] = tardiness(_eo);
objVector[1] = makespan(_eo);
// setting of the objective vector for the solution under consideration
_eo.objectiveVector(objVector);
```

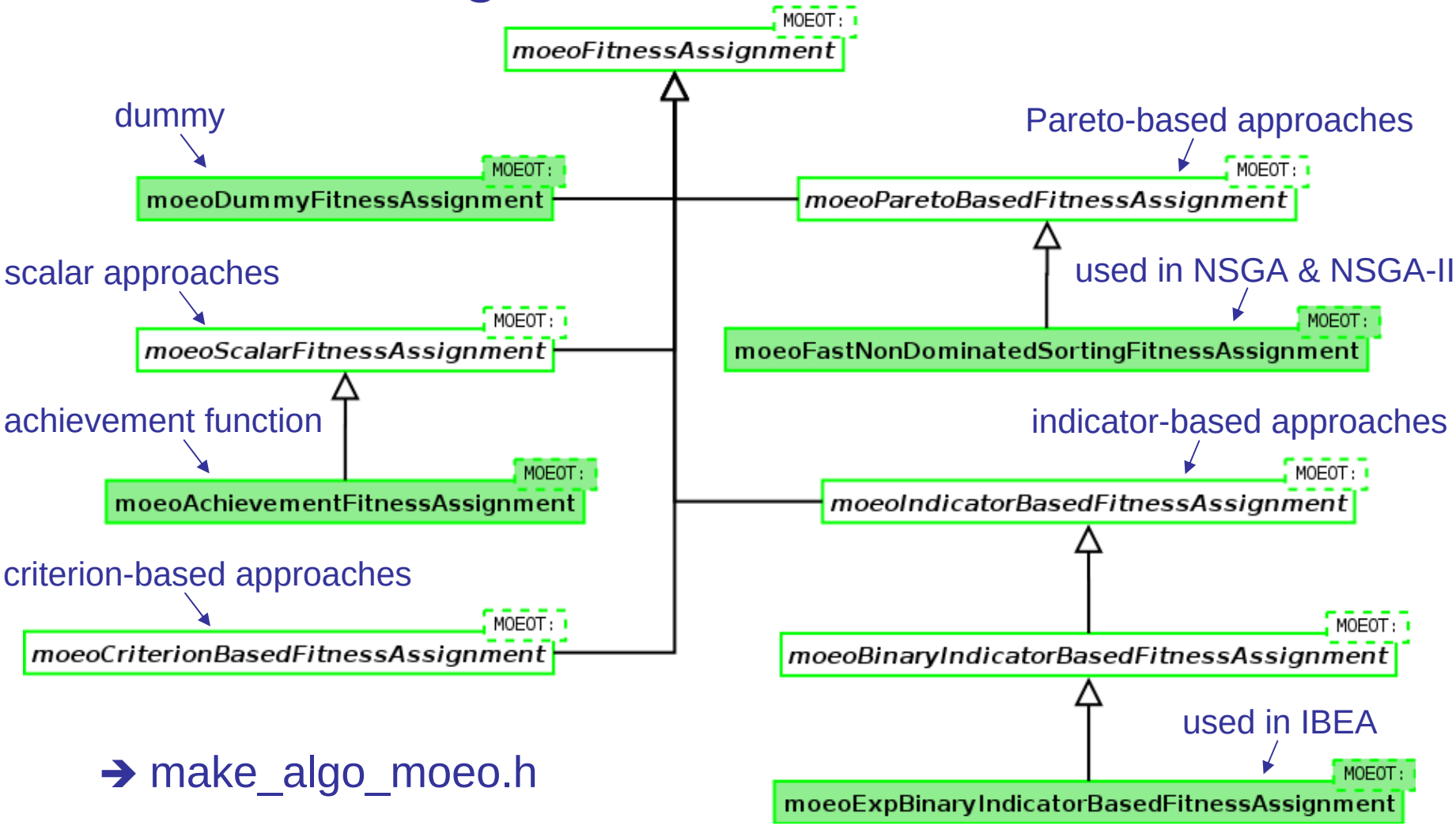
A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators ➔ FlowShopOp*
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Fitness Assignment: Core Classes



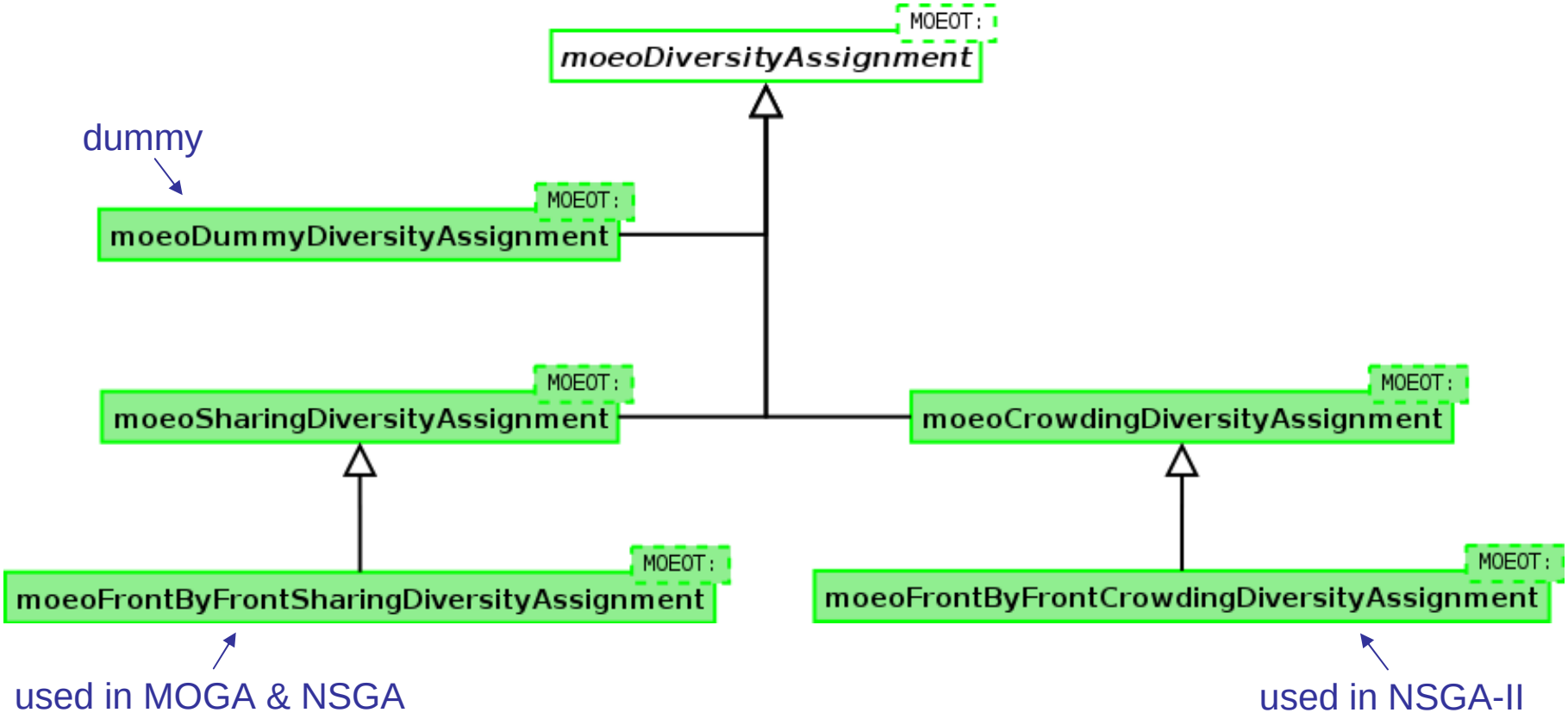
➔ make_algo_moeo.h

A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Diversity Assignment: Core Classes

➔ make_algo_moeo.h

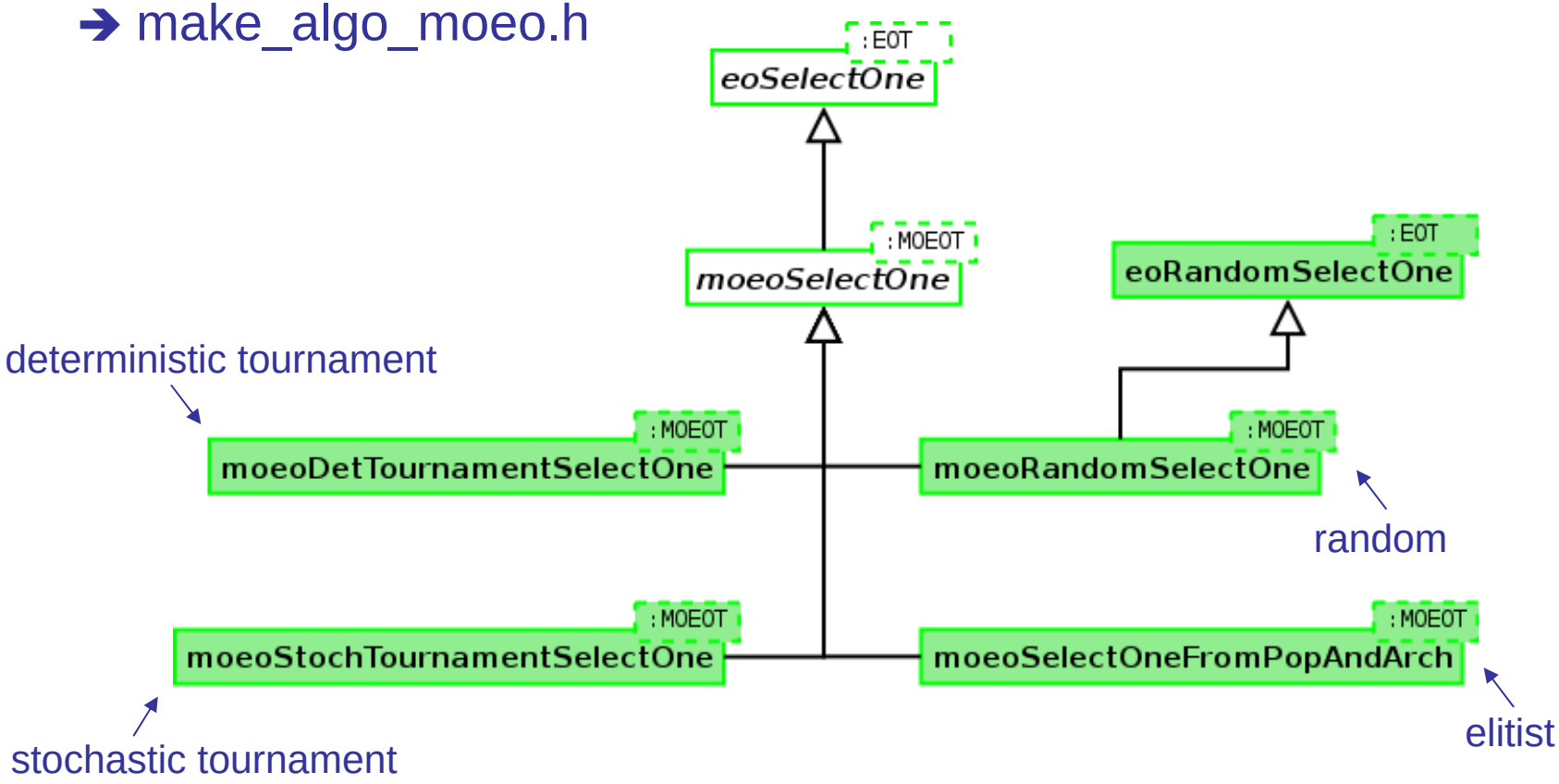


A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Selection: Core Classes

➔ make_algo_moeo.h

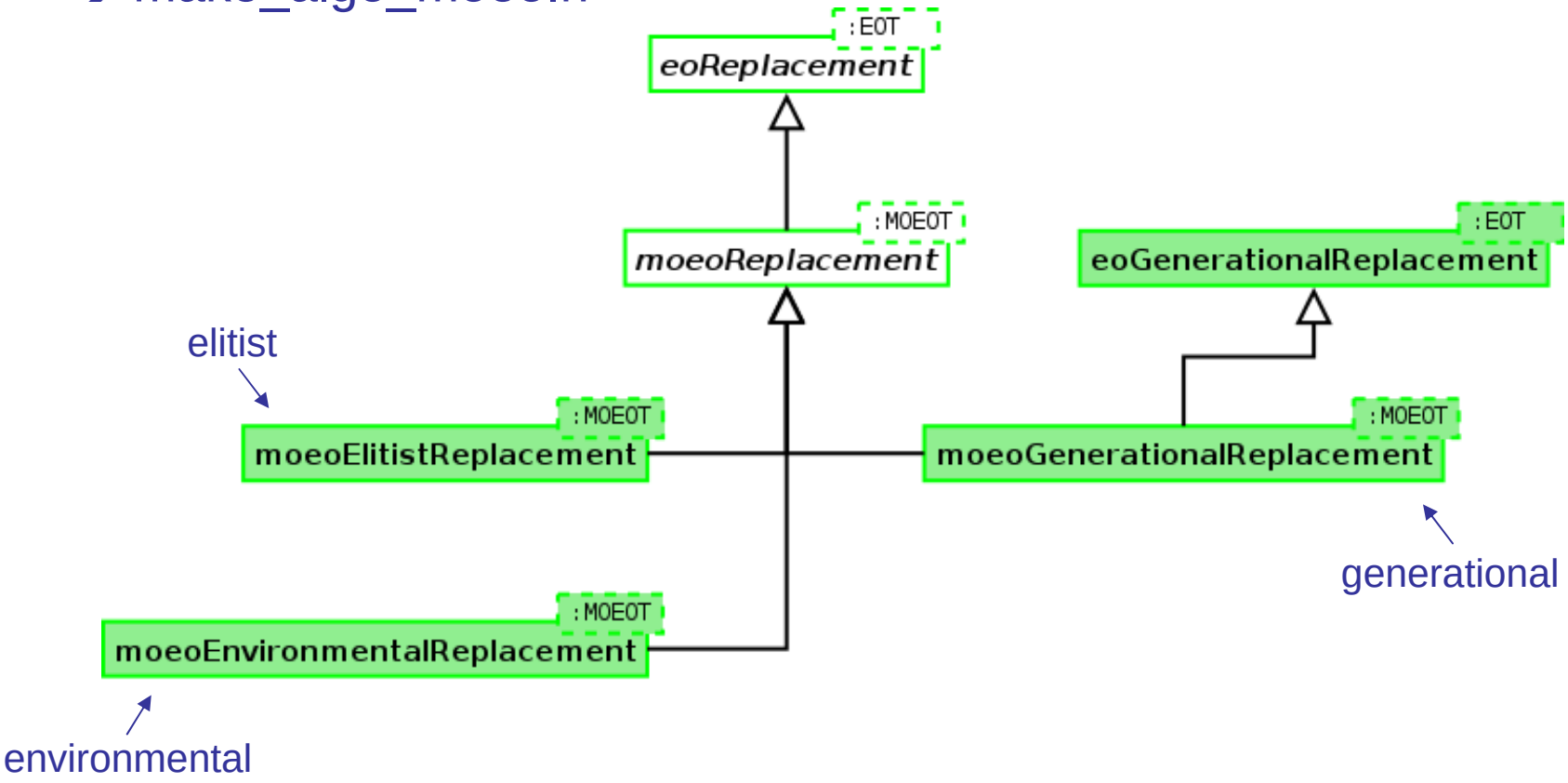


A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity assignment
7. Selection
8. Replacement
9. Stopping criteria

Replacement: Core Classes

➔ make_algo_moeo.h



A Multi-Objective Evolutionary Algorithm

1. Representation
2. Initialization
3. Evaluation
4. Variation operators
5. Fitness assignment
6. Diversity Assignment
7. Selection
8. Replacement
9. Stopping criteria ➔ make_continue_moeo.h

Statistical Tools

➔ make_checkpoint_moeo.h

Testing

• Computing Pareto set approximations obtained by two different algorithms on the benchmark 020_05_01.txt (by modifying some parameters)

Testing

- Edit the parameters file FlowShopEA.param (don't forget to delete the "#")

  > gedit FlowShopEA.param

- Run it

  > ./FlowShopEA @FlowShopEA.param

- Save the Pareto set approximation for ALGO-1

  > mv N.front ALGO-1.front

- Modify what you want in the parameters file

  > gedit FlowShopEA.param

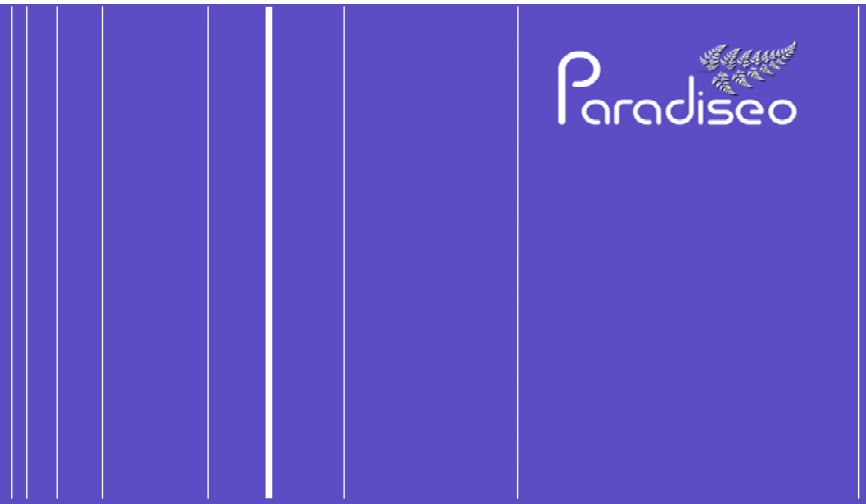
- Run it

  > ./FlowShopEA @FlowShopEA.param

- Save the Pareto set approximation for ALGO-2

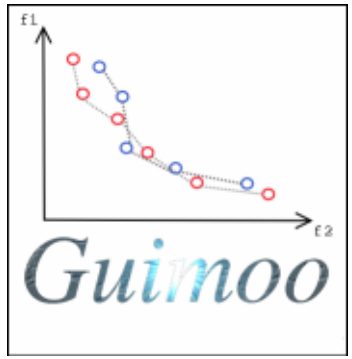
  > mv N.front ALGO-2.front

How to compare these two fronts?



GUIMOO

a Graphical User Interface for Multi-objective Optimization



http://guimoo.gforge.inria.fr

Paradiseo

GUIMOO: Testing

- Create your own project for Flow-shop
  - Name
  - Objectives

- Add the 2 fronts obtained using ParadisEO-MOEO

- Visualize the 2 fronts in 2D

- Compute the Contribution and the S metrics

Conclusion

- Thanks to ParadisEO-MOEO, you designed your first multi-objective evolutionary algorithm for the flow-shop scheduling problem
- Thanks to GUIMOO, you analyzed the output of your algorithm for different parameters

- ParadisEO-MOEO is only another step to
  - Hybridization and parallelism using ParadisEO-PEO

- http://paradiseo.gforge.inria.fr
- paradiseo-help@lists.gforge.inria.fr