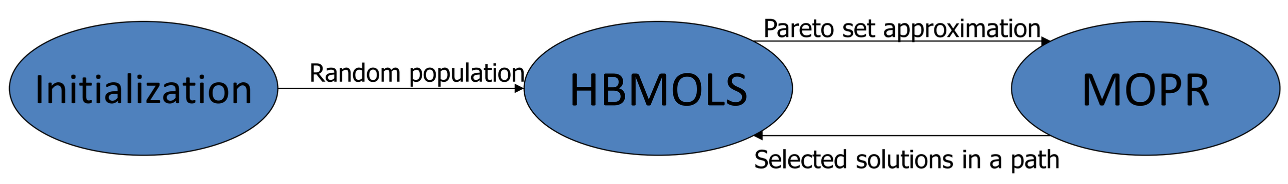
Experiments on Path Relinking Methods for bi-objective FlowShop problem

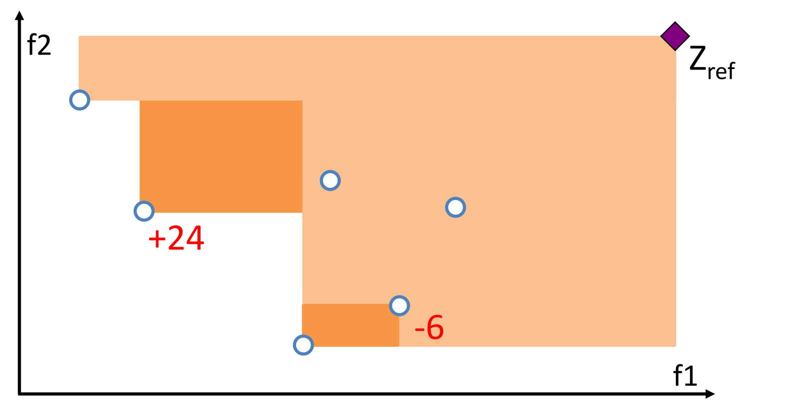
Ron-Qiang Zeng, Matthieu Basseur and Jin-Kao Hao. *University of Angers, France.*

Hybrid Path Relinking algorithm (HPR)

Cooperation of 2 algorithms:

- Hypervolume-Based Multi-Objective Local Search (HBMOLS)
- Multi-Objective Path Relinking (MOPR)





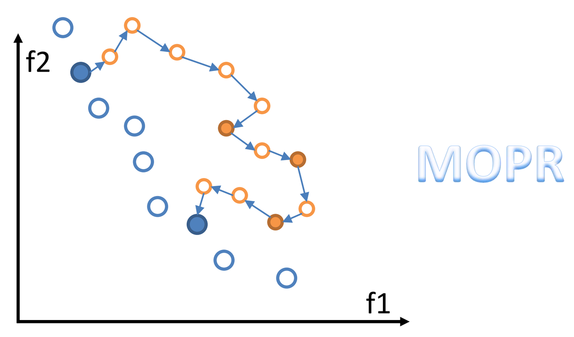
Hypervolume contribution indicator

MOPR

Step 1: randomly choose an initial and a guiding solution from a Pareto set approximation

Step 2: generate a path (set of solutions) linking the initial solution to the guiding solution

Step 3: return a subset of the path (for intensification)



- Non-dominated set of solutions
- *Initial* and *guiding* solutions
- Solutions of the path
- Subset of the path returned for intensification

HBMOLS

Repeat

Step 1: assign a fitness value to each solution in the population according to the hypervolume contribution indicator

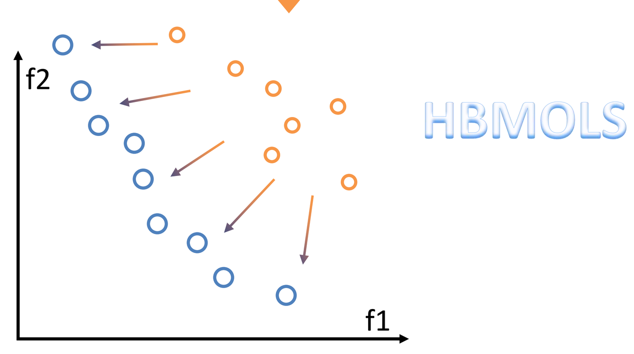
Step 2: generate a new solution from the neighborhood of a solution of the population

Step 3: evaluating the new solution using the hypervolume contribution indicator

Step 4: delete the solution with the worst fitness value from the population

Until all neighborhood explored without any improvement

Return the population



- Initial solution
- Pareto set returned (local optima)

Path Relinking Strategies

Initial and guiding solution selection

- Random
- Similar
- Different

Path generation strategies

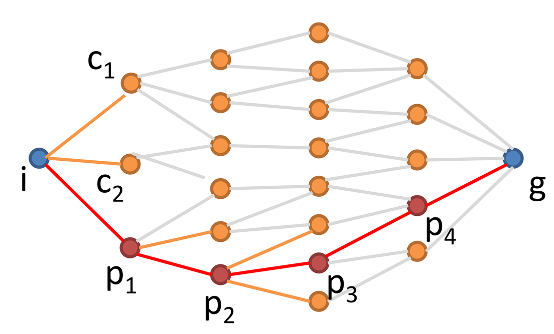
- Without comparison
  - First/Last : First/Last move reducing the distance to the guiding solution
  - Random: Random candidate move
- With comparison : Generate and evaluate all candidate moves
  - Pareto-Based: select randomly a *non-dominated* solution
  - Hypervolume-Based: select the solution with the largest hypervolume

Susbset selection

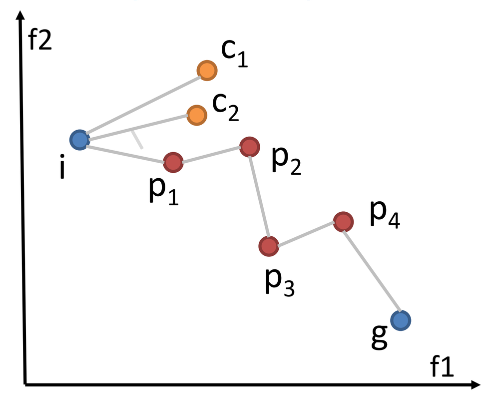
To be returned for intensification

- Without comparison
  - All: The entire path
  - Middle: The solution located in the « center » of the path
  - k-middle: A set of solutions located in the middle of the path
- With comparison
  - Best: The set of non-dominated solutions of the path

Decision space



Objective space



i: Initial solution

g: Guiding solution

: Neighbors

c_1, c_2, p_1: First candidates for path generation

p_i: Path candidates to be returned

<u>Path generation:</u> iteratively build a path by choosing among candidates c_i.

- First: c_1
- Last: p_1
- Random: p_1 or p_2 or p_3
- Pareto: p_1
- Hypervolume: p_1

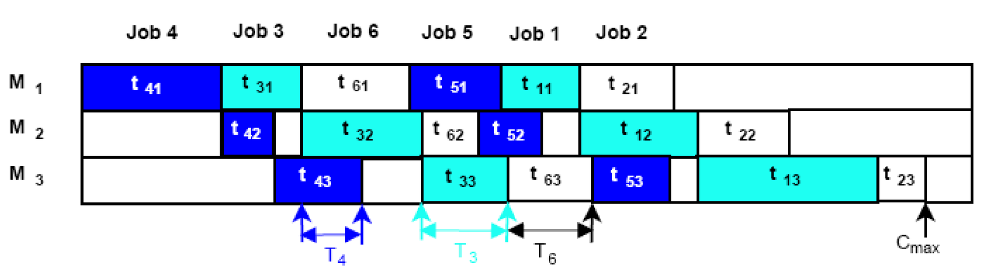
<u>Subset selection:</u>

- All: p_1, p_2, p_3 and p_4
- Middle: p_2 or p_3
- k-middle: p_2 and p_3
- Best: p_1 and p_3

Experiments

Permutation biobjective flowshop

- N jobs to schedule on M machines
- Jobs and machines are critical resources
- Jobs are treated on a defined order of machines
- 2 objectives functions : Maximal completion time (C_{max}) and Total tardiness (T_{sum})



Insertion operator: minimal path generation using the corresponding distance measure

Initial solution

16	14	7	18	13	0	1	5	8	6	3	11	9	12	15	10	4	2	17	19

Longest common subsequence: 11 jobs distance=9

14	18	0	1	5	8	3	17	2	15	7	11	16	6	4	9	13	10	12	19

Guiding solution

Results

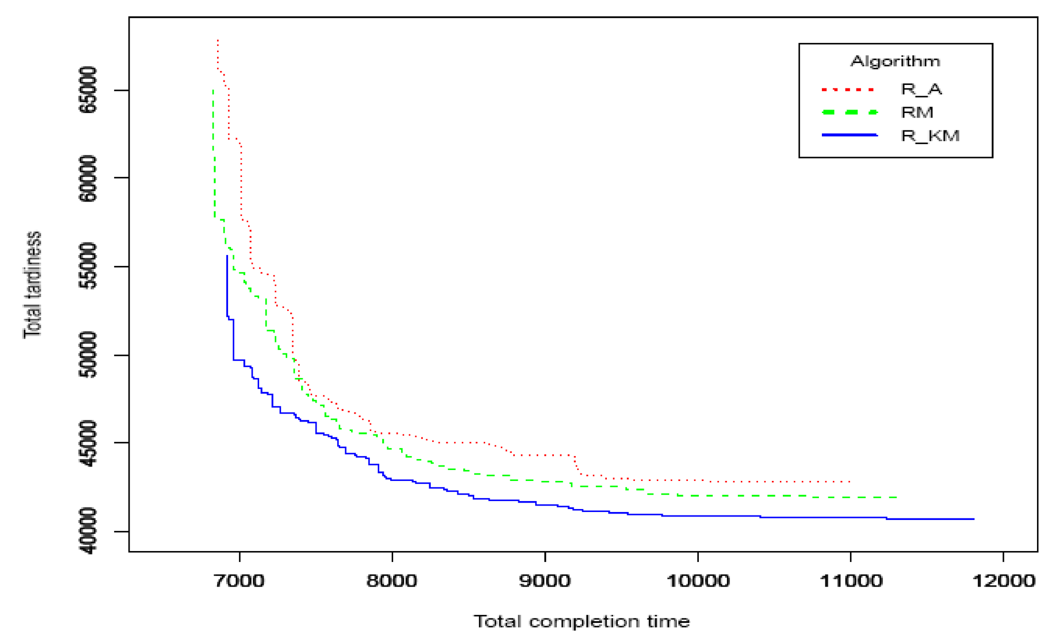
Significant differences obtained between subset selection strategies →

- PR_A: All
- PR_B: Best
- PR_M: Middle
- PR_KM: k-middle
- RM: random mutations
- CO: crossover operator

Instance	Algorithm					
	PR_A	PR_B	PR_M	PR_KM	RM	CO
20_05_01_ta001	0.050496	0.076627	0.093801	0.067028	**0.000260**	0.005152
20_10_01_ta011	0.023355	0.055498	0.048349	0.034595	**0.000739**	0.027353
20_15_01	0.032433	0.073174	0.070448	0.037654	**0.002330**	0.037131
20_20_01_ta021	0.009737	0.034508	0.024761	0.010079	**0.000077**	0.044826
30_05_01	0.049260	0.081154	0.099705	0.040607	**0.011844**	0.062030
30_10_01	0.100098	0.200979	0.176367	0.088794	**0.041814**	0.116553
30_15_01	0.052479	0.096203	0.105293	0.048227	**0.028186**	0.054050
30_20_01	0.048423	0.064844	0.071167	***0.040580***	**0.035835**	0.051028
50_05_01_ta031	0.031220	0.083466	0.090345	**0.022628**	0.041017	0.056559
50_10_01_ta041	0.103891	0.149919	0.132192	**0.079505**	0.089703	0.116051
50_15_01	0.131563	0.173639	0.156972	**0.091552**	0.114880	0.131505
50_20_01_ta051	0.129671	0.176523	0.146388	**0.093540**	0.117150	0.141695
70_05_01	0.110650	0.191452	0.152058	***0.096111***	**0.084047**	0.146741
70_10_01	0.131195	0.177933	0.157369	**0.119054**	0.146445	0.172327
70_15_01	0.149831	0.174514	0.164179	**0.134607**	0.156965	0.178769
70_20_01	0.139377	0.183869	0.147617	**0.102067**	0.135491	0.137697
100_05_01_ta061	0.199309	0.359023	0.236139	**0.157834**	***0.169815***	***0.175162***
100_10_01_ta071	0.093883	0.121682	0.104086	**0.071063**	***0.080287***	0.086577
100_15_01	0.187296	0.205879	0.175943	**0.128876**	0.163312	0.174849
100_20_01_ta081	0.205930	0.220908	0.187275	**0.131843**	***0.137246***	0.180406

Empirical attainment function

Instance 100_15_01 (100 jobs, 15 machines)



Conclusions and Perspectives

Conclusion

Proposition of a generic approach combining path-relinking and local search in a MO context

The use of path relinking offer a good alternative to RM and CO to iterate local searches

No significant observation concerning the path generation method

The solutions located in the middle of the path should be selected for intensification (k-middle)

Perspectives

Application to other multi-objective optimization problems

Evaluate methods selecting the initial and guiding solutions

Toward advanced path-relinking algorithms (path between more than two solutions, generate several paths simultaneously…)

université angers