Cooperation between exact methods and metaheuristics : application on a bicriteria permutation flowshop

C. Dhaenens, J. Lemesre, E.G. Talbi

O.P.A.C. group (LIFL), University of LILLE 1, France

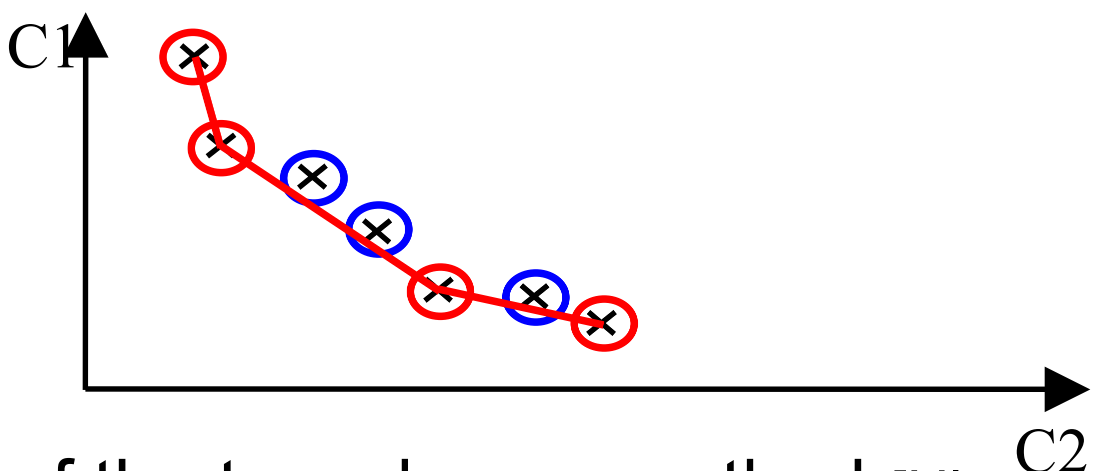
Overview

- Properties of the pareto front
- An exact method for bicriteria problems
  1. Two phases method
  2. Flowshop problem
  3. Application (Branch-and-Bound)
- Improvements of the original method
- Hybridization with a metaheuristic
- A parallel model
- Cooperation

The pareto front - properties

Description of pareto fronts :

- ***Supported solutions*** ***(aggregation : Geoffrion's theorem)***
- ***Non-supported solutions***

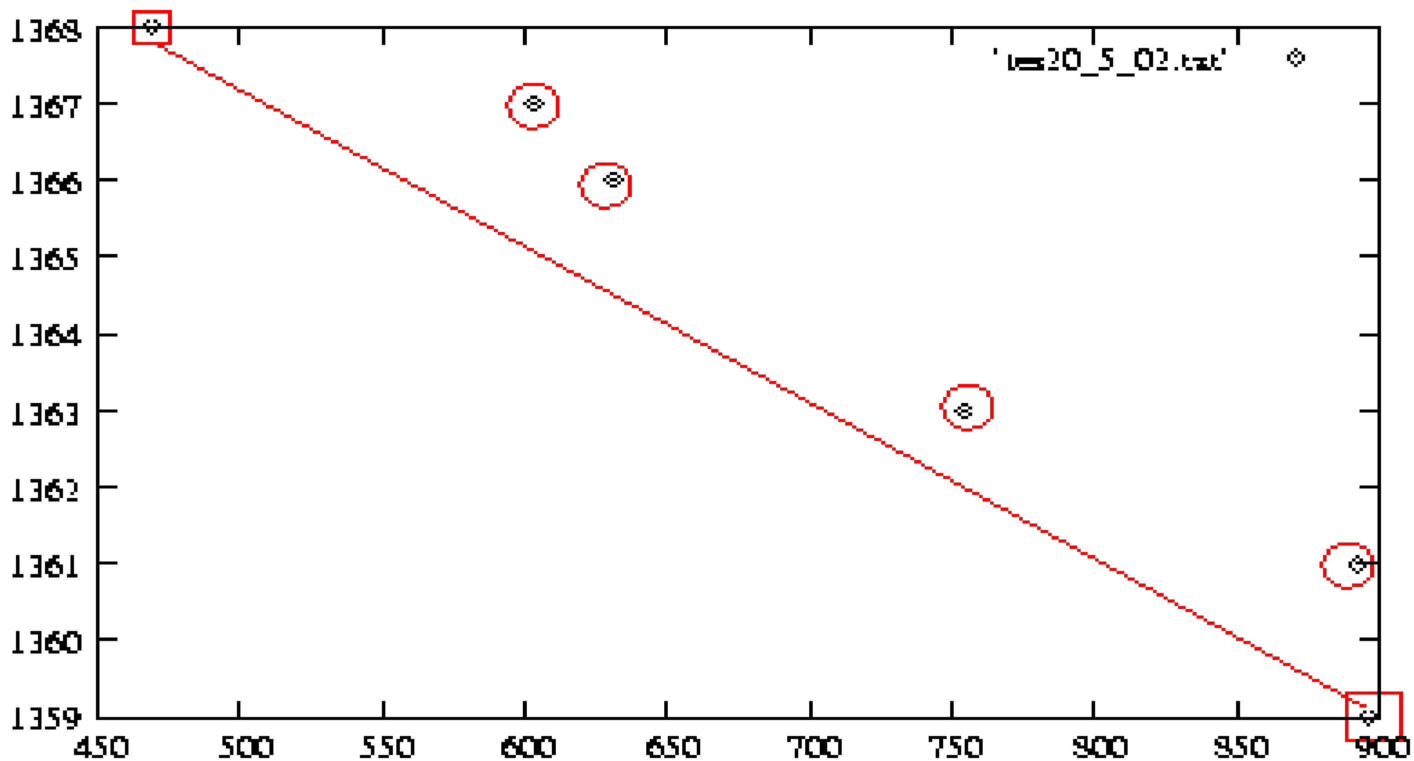


Principle of the two phases method [Ulungu and Teghem 1995]

- Applications : assignment and knapsack problems

Importance of non-supported solutions

Example (problem FS 20*5 No.2) :



Presentation of the two phases method

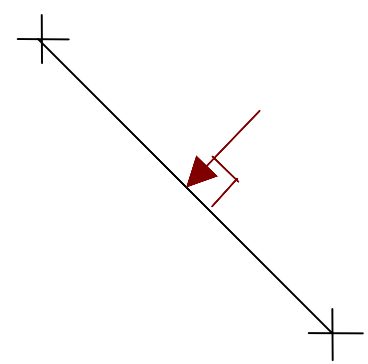
First phase (1)

Search of supported solutions :

1. The extremes
2. Two solutions : $z_1^{(r)} < z_1^{(s)}$ *and* $z_2^{(r)} > z_2^{(s)}$
3. Aggregation (Geoffrion’s theorem)
4. Search direction :

$$\lambda_1 = z_2^{(r)} - z_2^{(s)}$$

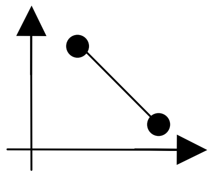
$$\lambda_2 = z_1^{(s)} - z_1^{(r)}$$



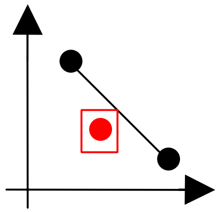
First phase (2)

Three possible configurations

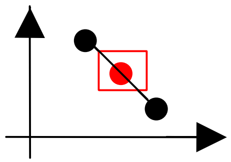
1. No new supported solution :



2. One new extreme supported solution:

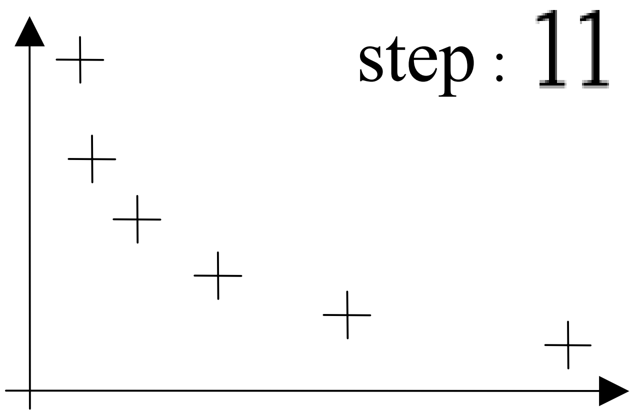


3. One new non-extreme supported solution:



First phase (3)

- Search in every intervals
- One solution $\longrightarrow$ Two new searches
- Stop when no new supported solution

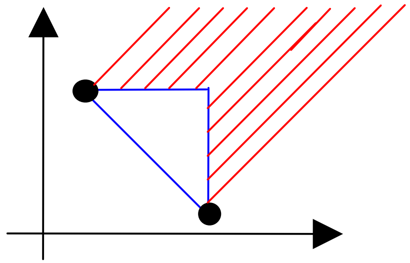


Second phase (1)

- Between supported solutions r et s.
- For all non-supported solution u :

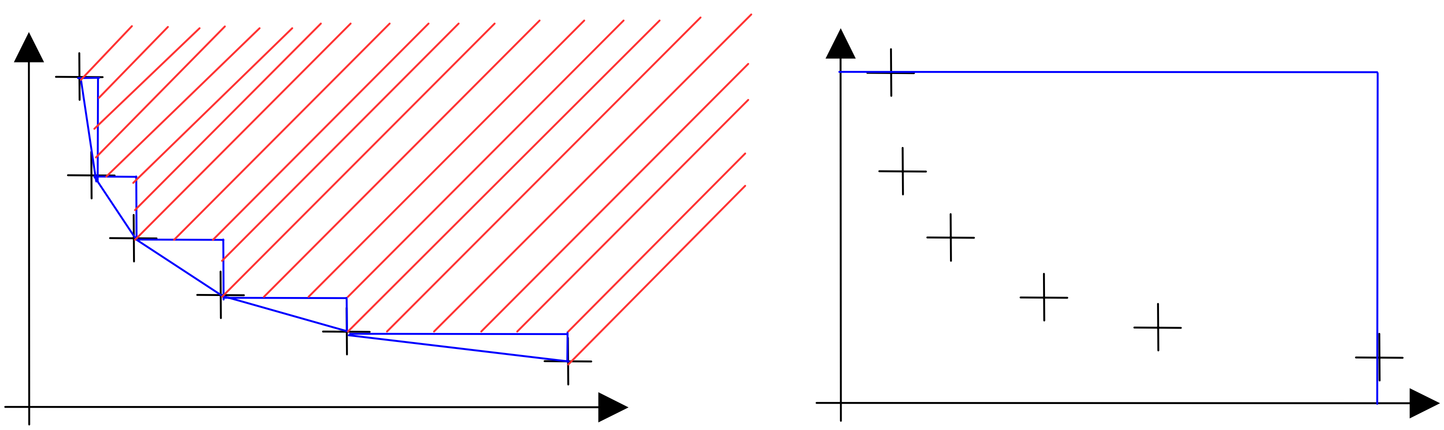
$$z_1^{(r)} < z_1^{(u)} < z_1^{(s)}$$
$$z_2^{(r)} > z_2^{(u)} > z_2^{(s)}.$$

- Non-supported solutions are in the interior of the triangle :



Second phase (2)

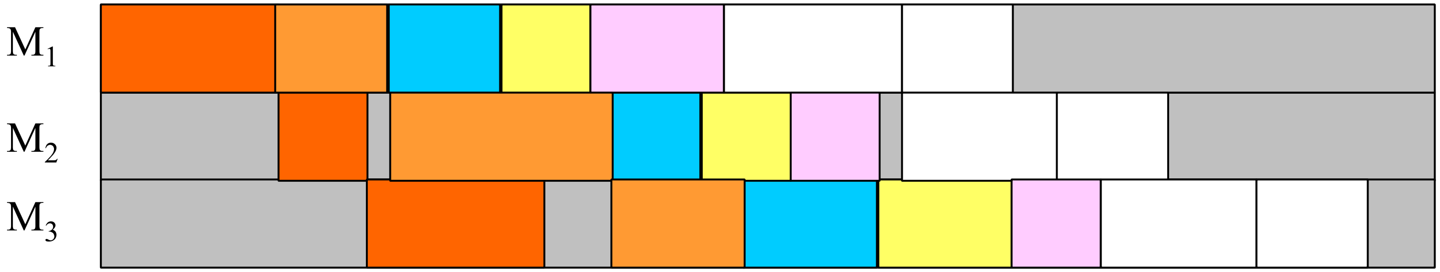
- We search in all the triangles
- All the searches are independent
- The search space is reduced



The permutation flowshop

The permutation Flowshop

- Scheduling problem (*n* jobs, *m* machines)
- Sequences of jobs are the same on every machine (n! possibilities)
- Criteria :
  1. Makespan (completion time)
  2. Total tardiness



Applying the two phases method : a branch-and-bound approach

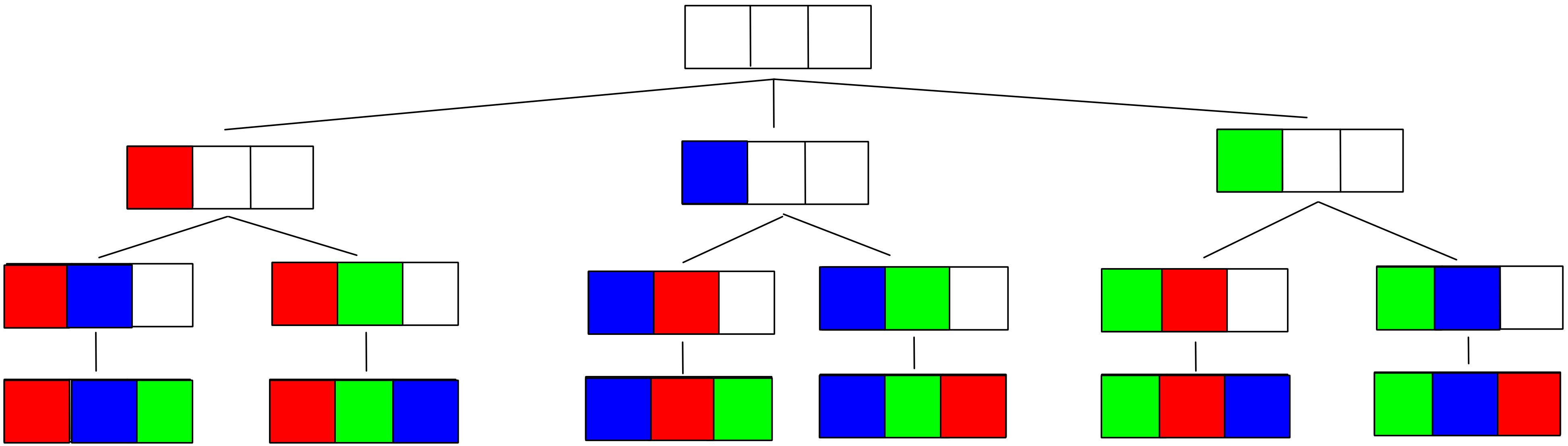
Applying of two phases method

- Requires an exact monocriterion method
- Aggregation of criteria
- Flowshop
- Best method : Branch and Bound

Branch and bound (1)

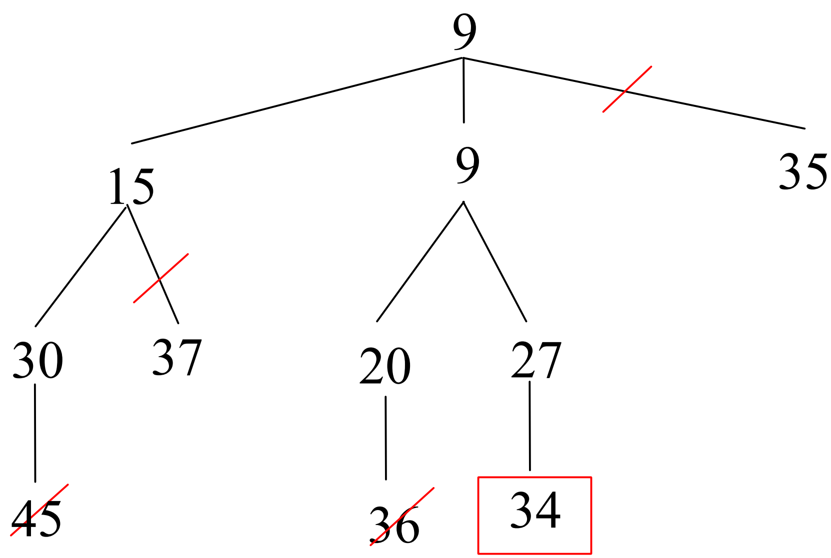
Principle :

1. Building solutions by enumeration
2. Lower bounds
3. Cut useless nodes



Branch and bound (2)

Example (best first search) :



Values are given by partial sequences and lower bounds (for jobs not in the partial sequence)

Lower bounds

Lower bounds

2 bounds :

1. For the makespan
2. For the total tardiness

1. For the makespan :

   Bound of Lageweg, Lenstra and Rinnooy kan

2. For the total tardiness :

   Schedule jobs at the beginning and at the end

Lower bound (makespan)

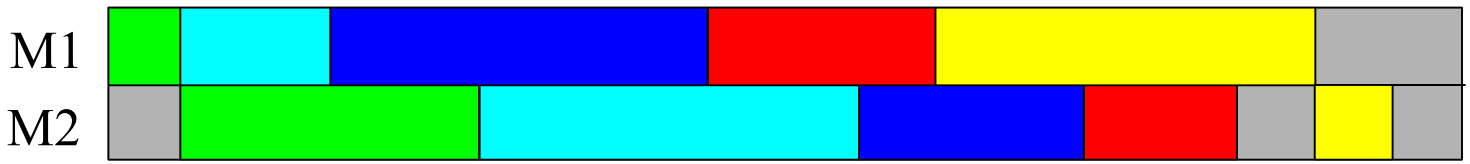
[Lageweg, Lenstra and Rinnooy kan 1978]

On two machines, problem in O(n*log(n)), Johnson's theorem :

- Jobs with smaller time on the first machine in increasing order.
- Jobs with smaller time on the second machine in decreasing order.

Example :

	M1	M2
J1	1	4
J2	3	2
J3	5	3
J4	2	5
J5	5	1



Lower bound (makespan)

Lag : carriage time between two machines.

On two machines, solve : $(p_{i,1} + l_i, l_i + p_{i,2})$

Let's take $M_k \, and \, M_l \, (with \, k<l)$, consider the problem :

$$p_{j,1} = p_{j,k}; l_j = \sum_{k<f<l} p_{j,f}; p_{j,2} = p_{j,l}.$$

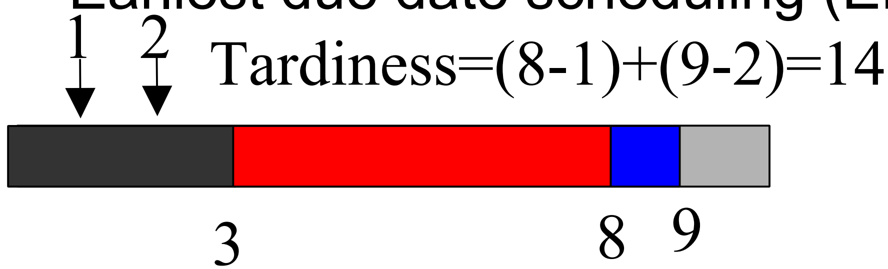
Add the time to go on the machine *k* and time to end.

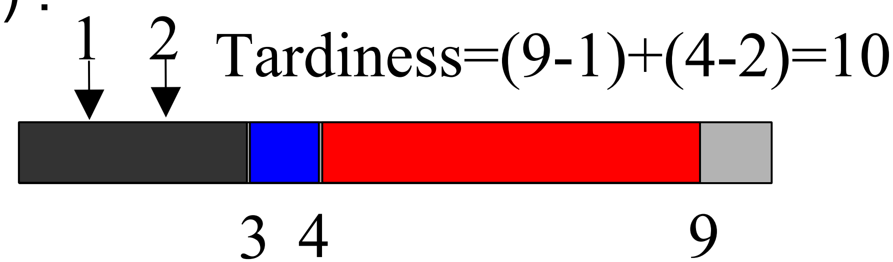
$$LB(J) = \max_{(1<k<l\leq m)} P^*_{Ja}(J, M_k, M_l) + \min_{((i,j)^2, i\neq j)} (Arr_{i,k} + Dep_{j,l})$$

Calculation in O(m²n*log(n)), reduced in O(m²n).

Lower bound (tardiness)

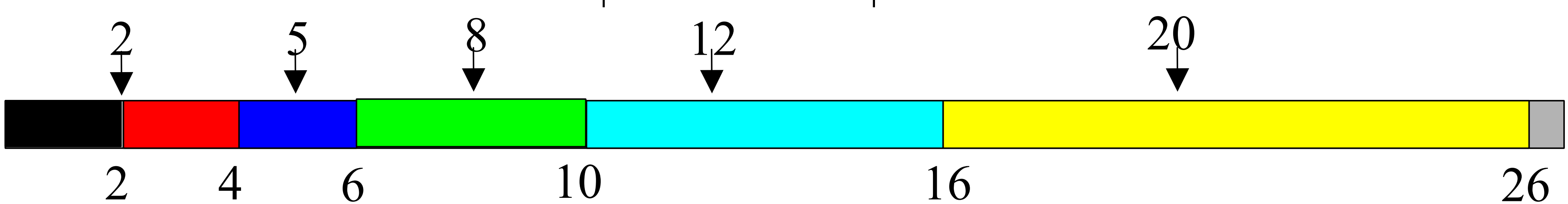
Earliest due date scheduling (EDD) :





Separation between due dates and processing times :
[Yeong-Dae Kim 1995]

	Due date	Processing time
J1	5	4
J2	2	2
J3	12	2
J4	8	6
J5	20	10



Tardiness=(4-2)+(6-5) +(10-8)+(16-12)+(26-20)=15

Lower bound (tardiness)

O(i) : sum of processing times of the *i* smallest jobs

S_1 : Due dates in increasing order

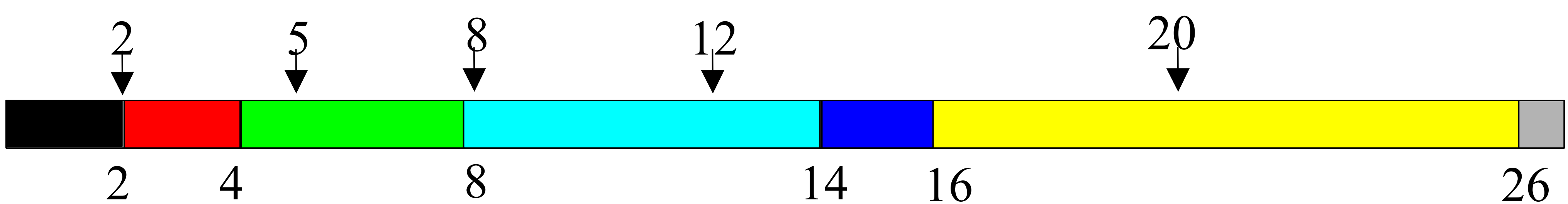
First bound :

$$LB_1 = \sum_{i=1}^{n} (O(i) - d_{S_1(i)})^+$$

Lower bound (tardiness)

Second bound : earliest due date sequence with decrease of tardiness

	Due date	Processing time
J1	5	4
J2	2	2
J3	12	2
J4	8	6
J5	20	10



Tardiness=Min((4-2);2)+Min((8-5);4)+Min((14-8);6)+Min((16-12);2)+Min((26-20);10) =19

Lower bound (tardiness)

r(i) is the tardiness of the job *i* in an EDD schedule

Second bound :

$$LB_2 = \sum_{i=1}^{n} [\min(r(i); p_i)^+]$$

Duration of passage from machine *k* to machine *l* :

Final bound :

$$LB(J) = \max(LB_1(J); LB_2(J))$$

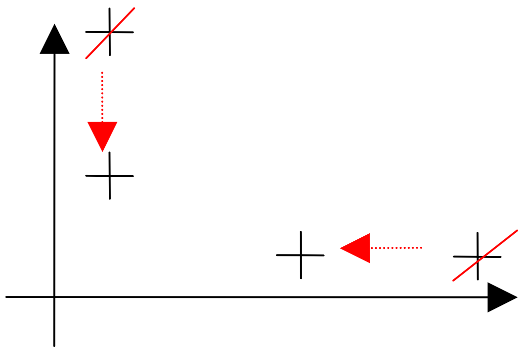
Improvements

Improvements (1)

Problems (scheduling):

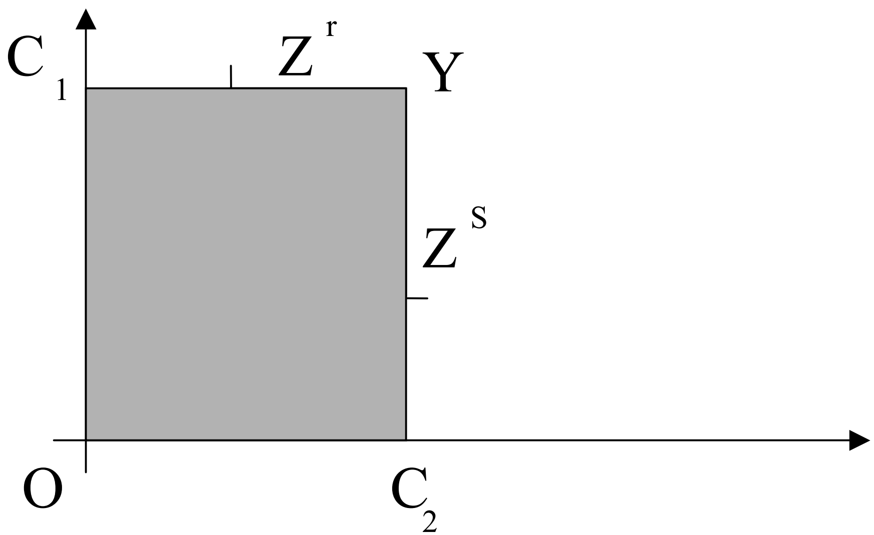
- lot of solutions with same value,
- lot of supported solutions.

We search the extremes for one criterion, after we adjust the second :



Improvements (2)

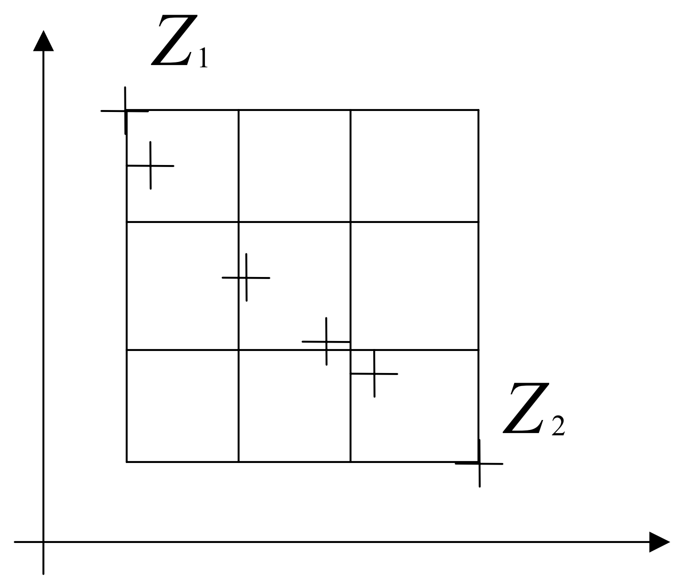
- Second phase : search not in triangle $Z^r Y Z^s$



- Search in rectangle $Y C_1 O C_2$
  - No need of upper bound
  - May find supported solutions

Improvements (3)

- No first phase between two nearby supported solutions
- Example :



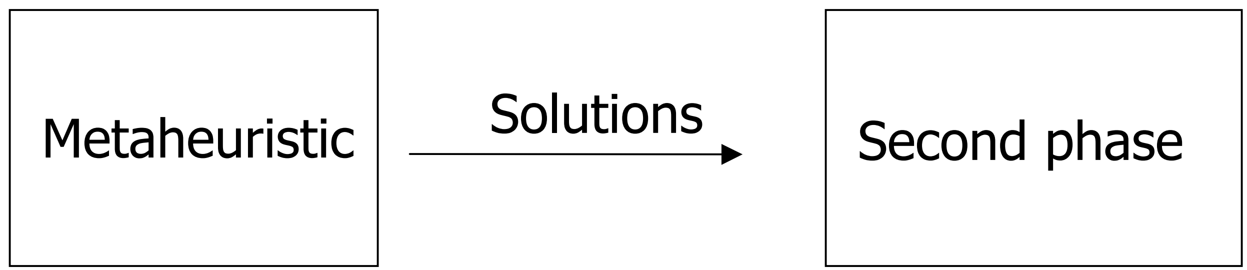
- Problem : two solutions nearby not in same box
- Minimal distance for first phase

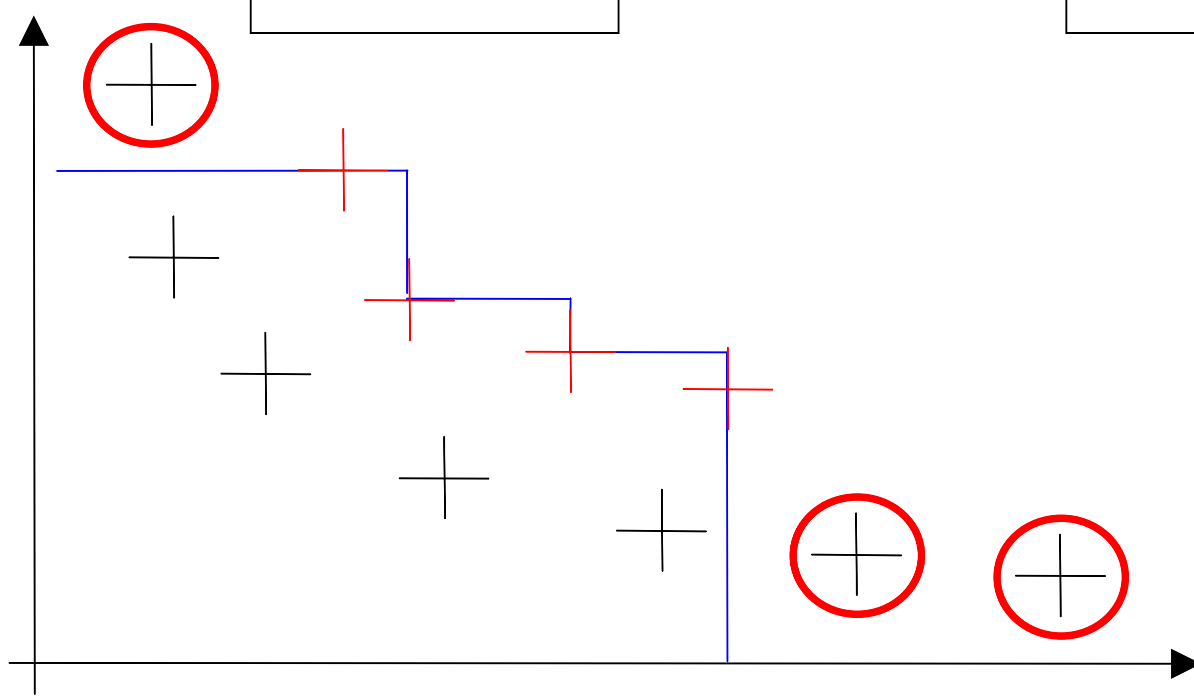
Hybridizations

Hybridization with a metaheuristic (1)

Hybridization 1 : second phase

Outline :



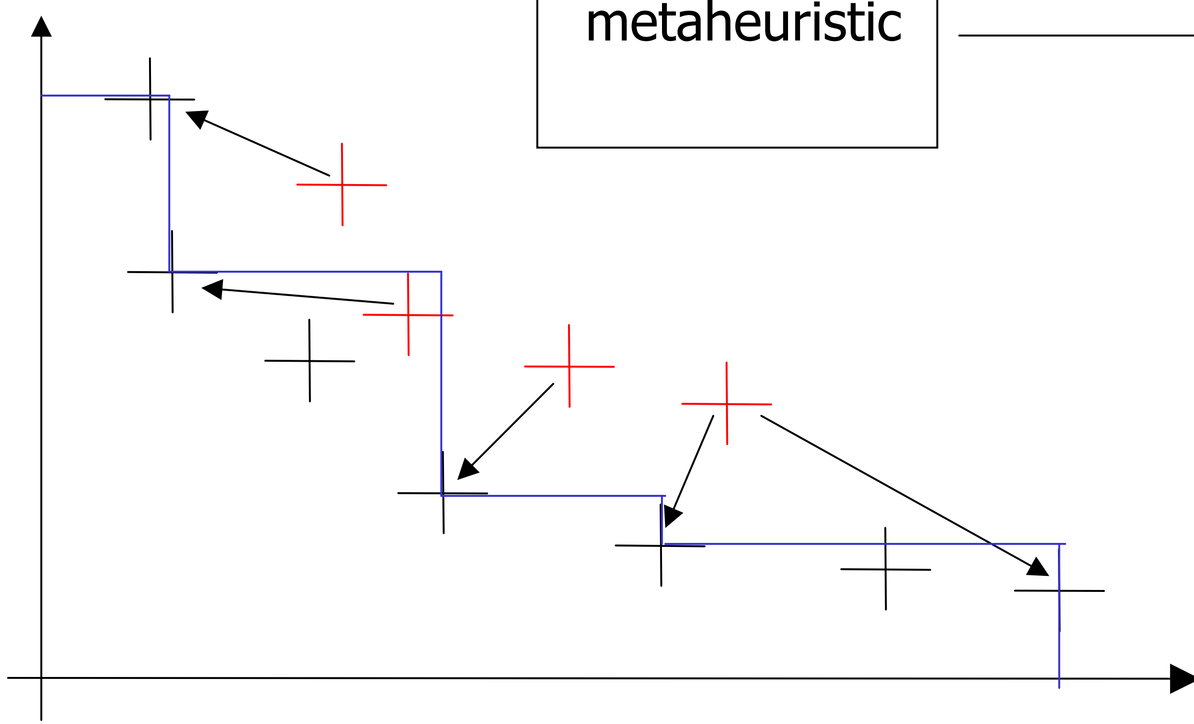


Hybridization with a metaheuristic (2)

Hybridization 2 : use solutions as initial values

Outline :

metaheuristic $\xrightarrow{\text{Solutions}}$ Two-phases method with start values for Branch-and-Bound

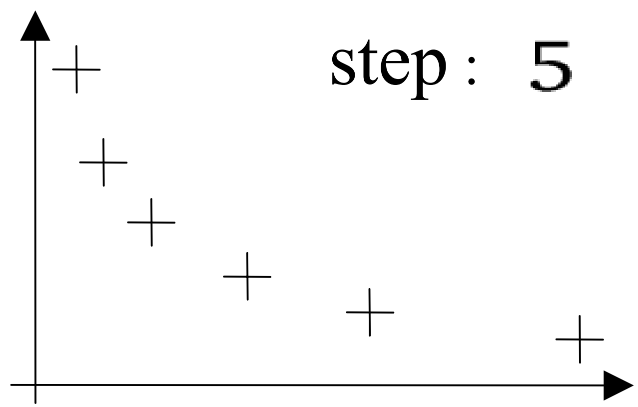


We find the whole front

Parallel model

Parallel model

- Implementation :
  - MPI
  - Master-slave model
- Parallelization of the first phase (sequential : 11 steps)

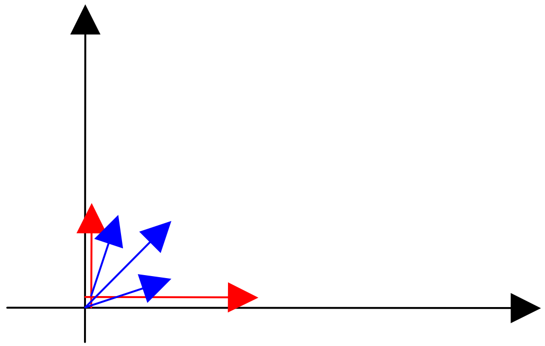


- Triangles independent in the second phase
- Problems :
  - Beginning of the search
  - Branch-and-bound=unbalanced tree

Optimization

Optimization of the beginning of the search

- Search solutions uniformly in search space with free processors
- Example : (6 processors : first step)



Covering of the two phases

- Start the second phase before the end of the first
- Priority list with a greater priority on the first phase

Results

Monocriterion results

Problems of [Taillard] extended for bicriteria problems.

For the makespan : (10 benchmarks per instance)

Instancies	Time
20*5	1 s
20*10	2 m
20*20	3 h – 1 d
50*5	10 s
50*10	1 m – 1 d
100*5	30 s

For the tardiness : (2 benchmarks per instance)

Instancies	Time
20*5	10 m
20*10	2 h
20*20	5 d
50*5	1 week

Bicriteria results

- Results of the two phases method :

Instancies	Time
20*5 (1)	30 s
20*5 (2)	30 m
20*10 (1)	1 week
20*10 (2)	1 week

- Results after the modifications :

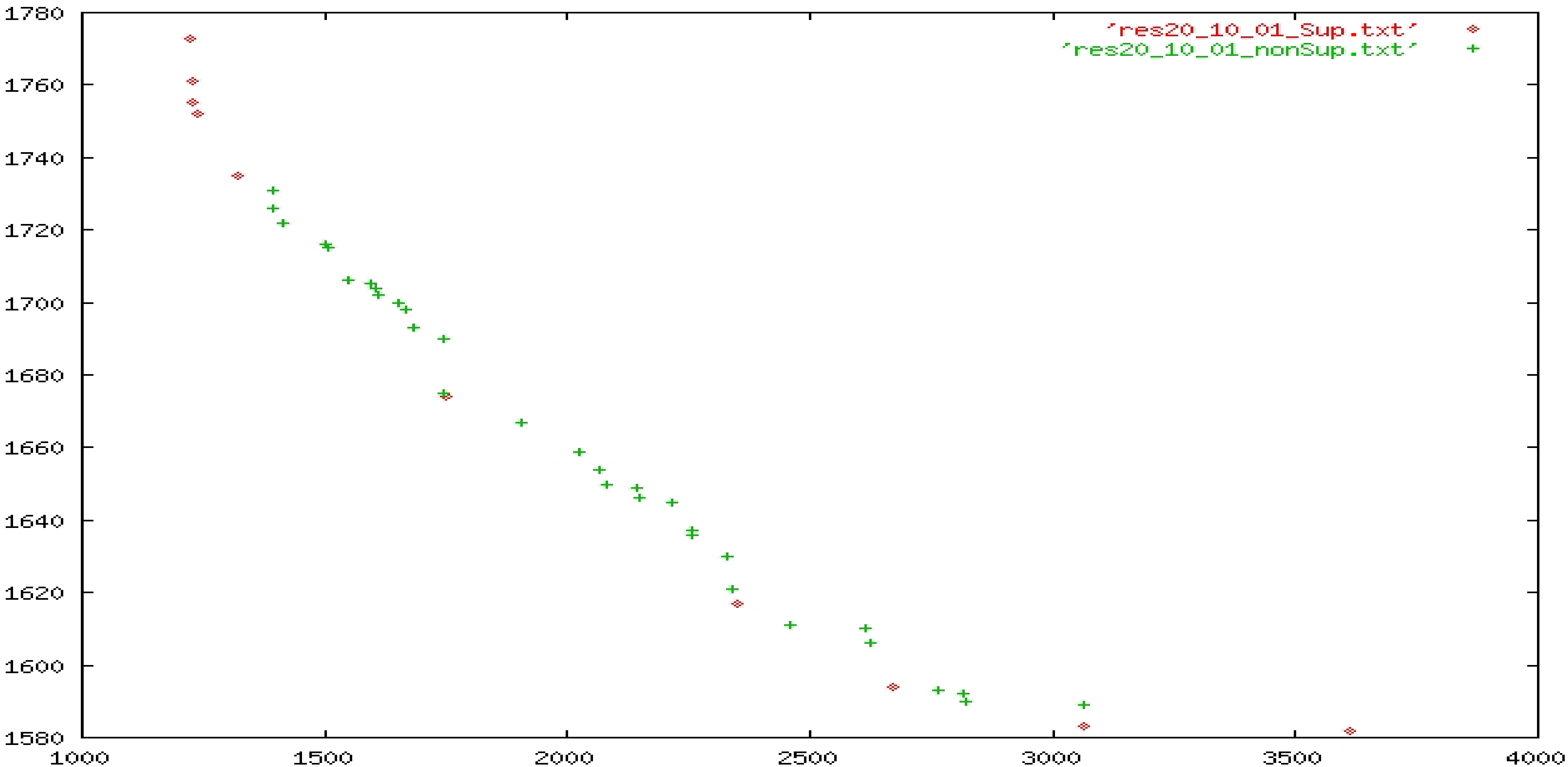
Instancies	Time
20*5 (1)	17 s
20*5 (2)	14 m
20*10 (1)	2 d 13 h
20*10 (2)	1 d 22 h

- Results with parallelization :

Instancies	Time
20*10 (1)	1 day
20*10 (2)	1 day
20*20	Solved

Bicriteria results

The pareto optimal front for 20*10 (1):



Conclusion and perspectives

- Conclusion
  - New bound for the tardiness
  - Parallel multicriteria model
  - Obtention of optimal fronts
  - Hybridization with a metaheuristic
    - Resolution of medium size problems (20 * 10)
    - Confirmation of fronts
    - Important time to prove optimality of solutions

Conclusion and perspectives

- Perspectives
  - Use of the parallel model and the hybridation simultaneously
  - Cooperation between metaheuristic and exact method [M Basseur]
  - Parallelization of the branch-and-bound : BOB (PRISM – Versailles)
  - Extension to more than two criteria ?

Cooperation

Cooperation

- Given a solution of the metaheuristic
- Decomposition of the solution :

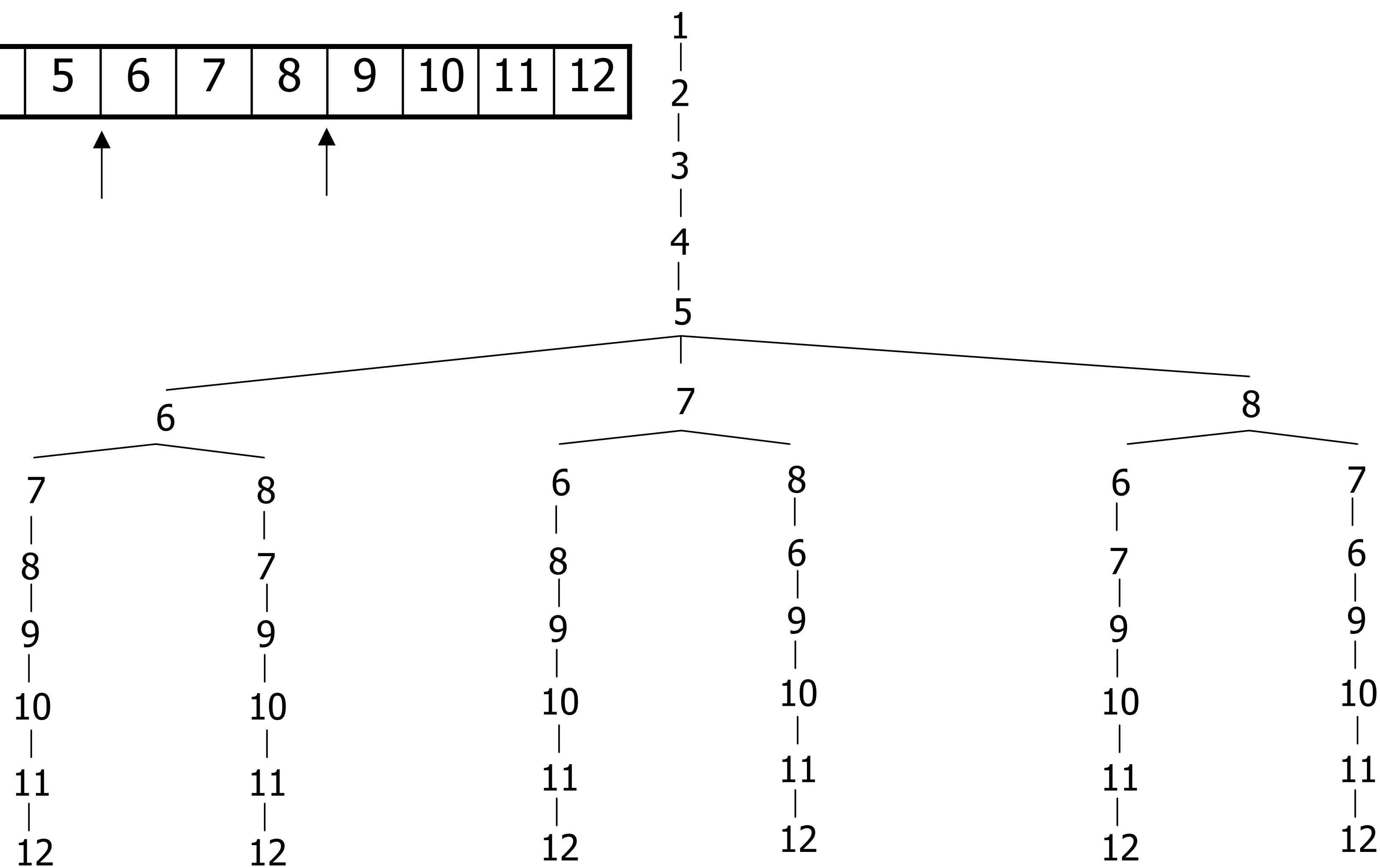
3	1	2	4	5	6	7	8	9	10	11	12

3	1	5	4	2	6	7	8	9	10	11	12

- For all solutions of the front
- Best solutions are kept : new front

Cooperation

1	2	3	4	5	6	7	8	9	10	11	12



Cooperation

- Preliminary results
  - Size of problems solved :

    Since (50*10) until (200*20)

  - Improvement of fronts especially for large problems

Improvement of fronts 100*10