2007 EVO* www.evostar.org
11-13 APRIL 2007 valencia::spain
Europe's premier co-located events on evolutionary computing

A GRASP and Branch-and-Bound Metaheurisitc for the Job-Shop Scheduling Problem

Susana Fernandes
Universidade do Algarve
Faro, Portugal
sfer@ualg.pt

Helena R. Lourenço
Universitat Pompeu Fabra
Barcelona, Spain
helena.ramalhinho@upf.edu

The work of S. Fernandes is supported by the program POCI2010 of the portuguese Fundação para a Ciência e Tecnologia. The work of Helena R. Lourenço is supported by Ministerio de Educación y Ciencia, Spain, SEC2003-01991/ECO.

2007 EVO* www.evostar.org
11-13 APRIL 2007 valencia::spain
Europe's premier co-located events on evolutionary computing

Outline of the Presentation

- The Job-Shop Scheduling Problem
- Optimized Search Heuristic – GRASP_B&B
- Computational Results
- Conclusions

The Job-Shop Scheduling Problem

- Definition
  - It considers a set of jobs to be processed on a set of distinct machines.
  - Each job is defined by an ordered set of operations.
  - Each operation is assigned to a different machine with a predefined constant processing time.
  - The order of the operations within the jobs and its correspondent machines are fixed a priori and are independent from job to job.

2007 EVO* www.evostar.org
11-13 APRIL 2007
valencia::spain
Europe's premier co-located events on evolutionary computing

The Job-Shop Scheduling Problem

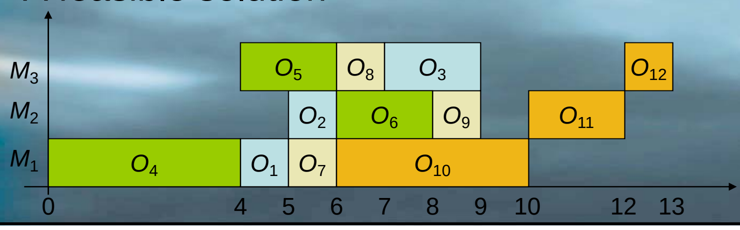
- Objective
  - Find a sequence of time slots for each operation on each machine, minimizing the maximum of the completion time of all jobs – the makespan.
- Constraints
  - Each machine can only process one operation at a time.
  - Different machines can not process the same job simultaneously.
  - Preemption is not allowed when processing operations.

The Job-Shop Scheduling Problem

- Instance

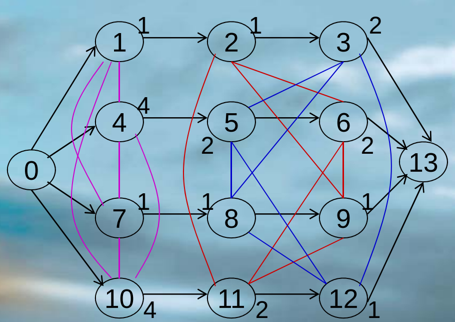
	Job 1			Job 2			Job 3			Job 4		
operations	1	2	3	4	5	6	7	8	9	10	11	12
machines	1	2	3	1	3	2	1	3	2	1	2	3
proc. time	1	1	2	4	2	2	1	1	1	4	2	1

- A feasible solution

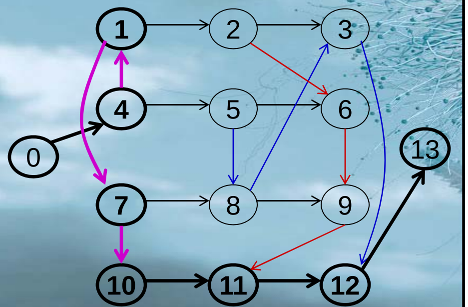


The Job-Shop Scheduling Problem

- Disjunctive Graph



Instance



A feasible solution
makespan 13

Optimized Search Heuristic – GRASP_B&B

- GRASP – applied to job shop

```
GRASP (total_runs)
for (run =1 to total_runs)
    while (solution not complete) do
        Greedy randomized building step
        Local search
```

  - Elements to join the solution - the sequence of operations at each machine.
  - Each element is evaluated by a heuristic function and incorporated (or not) in a restricted candidate list (RCL) according to its evaluation.
  - Greedy function to evaluate the elements – the makespan of each one-machine problem.

2007 EVO* www.evostar.org
11-13 APRIL 2007 VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

Optimized Search Heuristic – GRASP_B&B

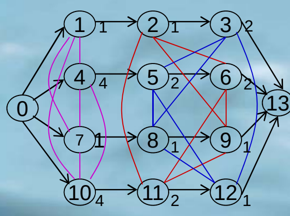
- Building step – RCL
  - Find the optimal solution for the one-machine problems for every machine not yet scheduled
    - **branch-and-bound** algorithm of Carlier (1982).
  - identify the best (lowest) and worse (biggest) makespans.
  - A machine *k* is included in the RCL if

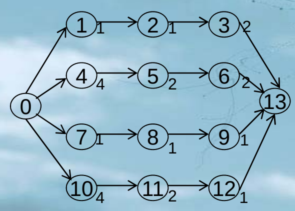
$$f(x_k) \geq \overline{f} - \alpha\left(\overline{f} - \underline{f}\right)$$

  - Machines with low values of makespan have less probability of being included in the RCL.
  - The machine to join the solution is chosen randomly from the RCL.

Optimized Search Heuristic – GRASP_B&B

- Building step – instance





M_1	O_1	O_4	O_7	O_{10}
r_i	0	0	0	0
p_i	1	4	1	4
q_i	3	4	2	3

$O_4 - O_{10} - O_1 - O_7$

makespan 12

M_2	O_2	O_6	O_9	O_{11}
r_i	1	6	2	4
p_i	1	2	1	2
q_i	2	0	0	1

$O_2 - O_9 - O_{11} - O_6$

makespan 8

M_3	O_3	O_5	O_8	O_{12}
r_i	2	4	1	6
p_i	2	2	1	1
q_i	0	2	1	0

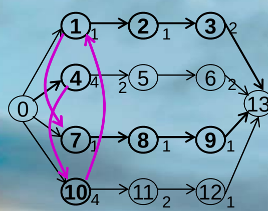
$O_8 - O_3 - O_5 - O_{12}$

makespan 8

2007 EVO* www.evostar.org
11-13 APRIL 2007
VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

Optimized Search Heuristic – GRASP_B&B

- Building step – instance



M_2	O_2	O_6	O_9	O_{11}
r_i	9	6	11	8
p_i	1	2	1	2
q_i	2	0	0	1

$O_6 - O_2 - O_{11} - O_9$

makespan 13

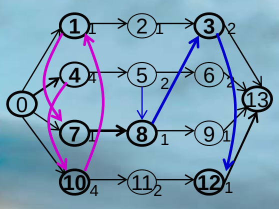
M_3	O_3	O_5	O_8	O_{12}
r_i	10	4	10	10
p_i	2	2	1	1
q_i	0	2	1	0

$O_5 - O_8 - O_3 - O_{12}$

makespan 14

Optimized Search Heuristic – GRASP_B&B

- Building step – instance

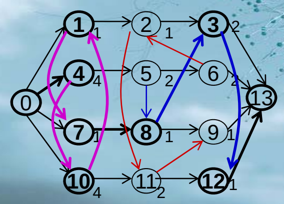


makespan 14

M_2	O_2	O_6	O_9	O_{11}
r_i	9	6	11	8
p_i	1	2	1	2
q_i	3	0	0	1

$O_6 - O_2 - O_{11} - O_9$

makespan 13



makespan 14

2007 EVO* www.evostar.org
11-13 APRIL 2007 VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

Optimized Search Heuristic – GRASP_B&B

- Local search – neighborhood structure
  - Forward moves over forward critical pairs of operations
  - Backward moves over backward critical pairs of operations
    - Block of critical operations - maximal ordered set of consecutive operations on a critical path sharing the same machine

Optimized Search Heuristic – GRASP_B&B

- Local search – neighborhood structure

  – Two operations *u* and *v* form a **forward critical pair** *(u,v)* if:
    - a) they both belong to the same block;
    - b) *v* is the last operation of the block;
    - c) the job successor of *v* also belongs to the same critical path;
    - d) the length of the critical path from *v* to the end is not less than the length of the critical path from the job successor of *u* to the end.

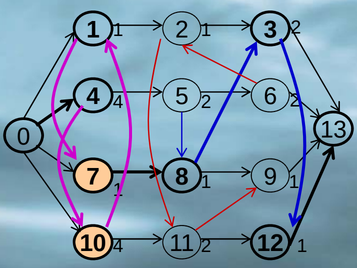
  – A forward move is executed by moving operation *u* to be processed immediately after operation *v*.

2007 EVO* www.evostar.org
11-13 APRIL 2007
VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

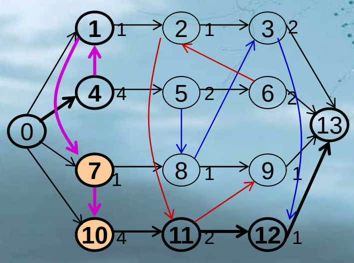
Optimized Search Heuristic – GRASP_B&B

- Local search – neighborhood structure

  – forward move – example (10,7)



makespan 14



makespan 13

2007 EVO* www.evostar.org 11-13 APRIL 2007 VALENCIA::SPAIN Europe's premier co-located events on evolutionary computing

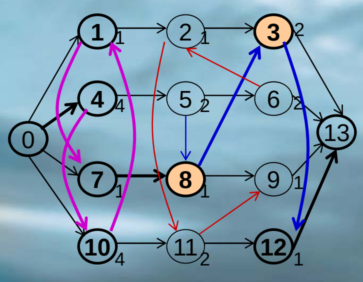
Optimized Search Heuristic – GRASP_B&B

- Local search – neighborhood structure
  - Two operations *u* and *v* form a **backward critical pair** *(u,v)* if:
    - a) they both belong to the same block;
    - b) *u* is the first operation of the block;
    - c) the job predecessor of *u* also belongs to the same critical path;
    - d) the length of the critical path from node 0 to *u*, including the processing time of *u*, is not less than the length of the critical path from node 0 to the job predecessor of *v*, including the processing time of the job predecessor of *v*.
  - A backward move is executed by moving operation *v* to be processed immediately before operation *u*.

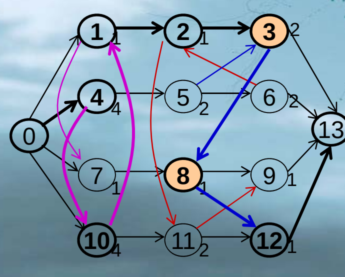
2007 EVO* www.evostar.org 11-13 APRIL 2007 VALENCIA::SPAIN Europe's premier co-located events on evolutionary computing

Optimized Search Heuristic – GRASP_B&B

- Local search – neighborhood structure
  - backward move – example (8,3)



makespan 14



makespan 14

Optimized Search Heuristic – GRASP_B&B

- Local search – inspecting the neighborhood
  - Acceptance strategy – first accept
    - Given solution *x* with M_k machines already scheduled
    - stop whenever we find a neighbor with a best evaluation value than the makespan of *x*.
  - Evaluation function
    - Compute the length of all the longest paths through the operations that were between operations *u* and *v* of the critical pair *(u,v)* on the critical path of solution *x*.
    - Use the same subroutine of Balas and Vazacopoulos (1998) algorithm.

2007 EVO* www.evostar.org 11-13 APRIL 2007 valencia::spain Europe's premier co-located events on evolutionary computing

Computational Results

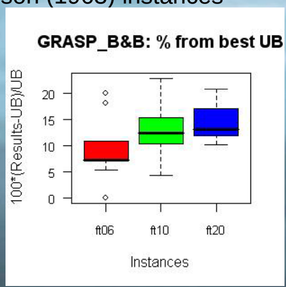
- Platform
  - Pentium 4 CPU 2.80 GHz
  - Code in Visual C
- Runs
  - 100 runs for each instance.
- Instances
  - abz5-9 (Adams et al. 1988)
  - ft6, ft10, ft20 (Fisher and Thompson 1963)
  - la01-40 (Lawrence 1984),
  - orb01-10 (Applegate and Cook 1991)
  - swv01-20 (Storer et al. 1992)
  - ta01-70 (Taillard 1993)
  - yn1-4 (Yamada and Nakano 1992)

Computational Results

- Boxplots of $$RE_{UB}(x) = 100 \times \frac{f(x) - UB}{UB}$$

Upperbounds (UB) gathered from Jain and Meeran (1999) and Nowicki and Smutnicki (1996, 2005)

  - Fisher and Thompson (1963) instances

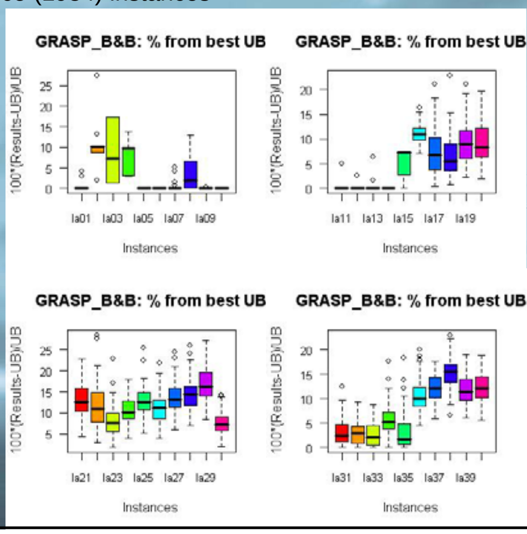


ft06: 6*6 ft10: 10*10 ft20: 20*5

2007 EVO* www.evostar.org
11-13 APRIL 2007 VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

Computational Results

- Lawrence (1984) instances



la 01 – 05: 10*5
la 06 – 10: 15*5
la 11 – 15: 20*5
la 16 – 20: 10*10
la 21 – 25: 15*10
la 26 – 30: 20*10
la 31 – 35: 30*10
la 36 – 40: 15*15

Computational Results

- Comparison to other algorithms
  - The Shifting Bottleneck Procedure of Adams, Balas and Zawack (1988)
  - The GRASP Procedure of Binato, Loewenstern and Resende (2001)

2007 EVO* www.evostar.org
11-13 APRIL 2007 VALENCIA::SPAIN
Europe's premier co-located events on evolutionary computing

Computational Results

- GRASP_B&B Comparison to other algorithms
  - Lawrence (1984) instances 31-40

name	GRASP_B&B	btime (s)	ttime (s)	GRASP	time (s)	Shifting Bottleneck	time (s)
la31	**1784**	0.0702	7.0160	1784	231290	1784	38.3
la32	**1850**	0.5612	6.2350	1850	241390	1850	29.1
la33	**1719**	1.265	7.9060	1719	241390	1719	25.6
la34	**1721**	3.8093	8.2810	1753	240380	1721	27.6
la35	**1888**	0.2844	5.6880	1888	222200	1888	21.3
la36	**1325**	0.0853	4.2650	1334	115360	1351	46.9
la37	1479	4.0295	4.7970	1457	115360	1485	61.4
la38	1274	0.7153	5.1090	1267	118720	1280	57.5
la39	1309	2.9835	4.4530	1290	115360	1321	71.8
la40	1291	3.5581	5.3910	1259	123200	1326	76.7

Conclusions

- We have designed a very simple optimized search heuristic
  - GRASP
  - Branch-and-Bound
- Although this is a very simple (and fast) algorithm, it achieves the best known upper bound in 23 of the 152 instances used in this study.
- Compared with other initial solution heuristics:
  - GRASP of Binato et al. (2001) - base for a GRASP with path-relinking of Aiex et al (2003).
    - Much faster
    - Quality of solution slightly worse in 60% of all the instances tested.
  - Shifting bottleneck of Adams et al.(1988) –base for guided local search of Balas and Vazacopoulos (1998).
    - Faster
    - Better solutions for all comparable instances (except 2).
- Good starting point for a more elaborated metaheuristic.

2007 EVO* www.evostar.org
11-13 APRIL 2007
valencia::spain
Europe's premier co-located events on evolutionary computing

References

- Adams, J., E. Balas and D. Zawack (1988). "The Shifting Bottleneck Procedure for Job Shop Scheduling." Management Science, vol. 34(3): pp. 391-401.
- Aiex, R. M., S. Binato and M. G. C. Resende (2003). "Parallel GRASP with path-relinking for job shop scheduling." Parallel Computing, vol. 29(4): pp. 393-430.
- Applegate, D. and W. Cook (1991). "A Computational Study of the Job-Shop Scheduling Problem." ORSA Journal on Computing, vol. 3(2): pp. 149-156.
- Balas, E. and A. Vazacopoulos (1998). "Guided Local Search with Shifting Bottleneck for Job Shop Scheduling." Management Science, vol. 44(2): pp. 262-275.
- Binato, S., W. J. Hery, D. M. Loewenstern and M. G. C. Resende (2001). "A GRASP for Job Shop Scheduling." In C.C. Ribeiro and P. Hansen, editors, Essays and surveys on metaheuristics, pp. 59-79. Kluwer Academic Publishers.
- Carlier, J. (1982). "The one-machine sequencing problem." European Journal of Operational Research, vol. 11: pp. 42-47.
- Fisher, H. and G. L. Thompson (1963). Probabilistic learning combinations of local job-shop scheduling rules. In J. F. Muth and G. L. Thompson eds. Industrial Scheduling. pp. 225-251. Prentice Hall, Englewood Cliffs.
- Jain, A. S. and S. Meeran (1999). "Deterministic job shop scheduling: Past, present and future." European Journal of Operational Research, vol. 133: pp. 390-434.
- Lawrence, S. (1984), "Resource Constrained Project Scheduling: an Experimental Investigation of Heuristic Scheduling techniques", Graduate School of Industrial Administration, Carnegie-Mellon University.
- Nowicki, E. and C. Smutnicki (2005). "An Advanced Tabu Search Algorithm for the Job Shop Problem." Journal of Scheduling, vol. 8: pp. 145-159.
- Nowicki, E. and C. Smutniki (1996). "A Fast Taboo Search Algorithm for the Job Shop Problem." Management Science, vol. 42(6): pp. 797-813.
- Taillard, E. D. (1993). "Benchmarks for Basic Scheduling Problems." European Journal of Operational Research, vol. 64(2): pp. 278-285.
- Taillard, É. D. (1994). "Parallel Taboo Search Techniques for the Job Shop Scheduling Problem." ORSA Journal on Computing, vol. 6(2): pp. 108-117.
- Yamada, T. and R. Nakano (1992). A genetic algorithm applicable to large-scale job-shop problems. In R. Manner and B. Manderick eds. Parallel Problem Solving from Nature 2. pp. 281-290. Elsevier Science.