## Flexible job shop scheduling problem

What is JSSP and what variant will this paper be about? JSSP you have *n* jobs which have a set of operations and *m* machines. You want to minimize the makespan (the total length of the schedule) with the constraints all operations must be in order and any operation must be performed on a machine. This paper also minimizes energy consumption and instability (Small change in input doesn't cause a big change in output).

This paper uses the backtracking search algorithm. This algorithm is an evolutionary algorithm, which means that a new and old population provides a direction of where the solution could be. It uses mutation and crossover to generate new populations.

The solution represented as two vectors: Machine assignment and Operations sequence vector. The factor *a* changes the speed of the mutation of the algorithm. The crossover procedure exchanges information between subsequent populations in an effective manner. In the literature there is a widely used order-based crossover procedure, but this is less effective in later stages. A different crossover procedure is based on similarity. Partial slack is defined as the maximum time interval by which the start time can be delayed without delaying subsequent operations.

Computational dstudies: Prerformance metrics, effectivness of the proposed SPox comparison of rescheduling strategies, comparison with other algorithms. Benchmark instance : 20 instances, small, medium, large. Number of machines between 3-30 and number of new jobs 2-10. Performance metrics.

- Set coverage C(A,B):-
  - the measure of dominance between two sets of pareto solutions.
  - the percentage of solutions in B that are dominated by solutions in A
- Generational distance

- Measure of average distance between solutions on the obtained front and the optimal front

- Smaller value indicates better convergence performance

They made a table of all the result of all benchmarks. All of them work better with the proposed crossover instead of the literature..

Comparison of rescheduling strategies

- Insert new jobs at the end of schedule
- Reschedule every procedure job to get optimal remaining schedule
- Explore all available time intervals first, then aplly 1

• The proposed rescheduling strategy The result of all strategies where put in a table. In some cases the strategy of the authors is not optimal.

Comparison with other algorithms

All the numbers of the proposed algorithms than other algorithms. The authors call it absolute supremacy.

Conclusions

- Proposed algorithms work better than others
- Mutation and SPOX perform well Future work:
- Machine breakdown
- Tool deterioration
- Worker Flexibility
- integration of local search operators
- More research into modeling of energy consumption My opinion
- Well written
- Nice informative ex and fig
- Headers more informative
- No criticism of own results
- Very extensive comp. study
- Real world study was done, but not compared to anythin