A New Multi-Objective Optimization Scheme for Grid Resource Allocation

Monday 1. November 2008

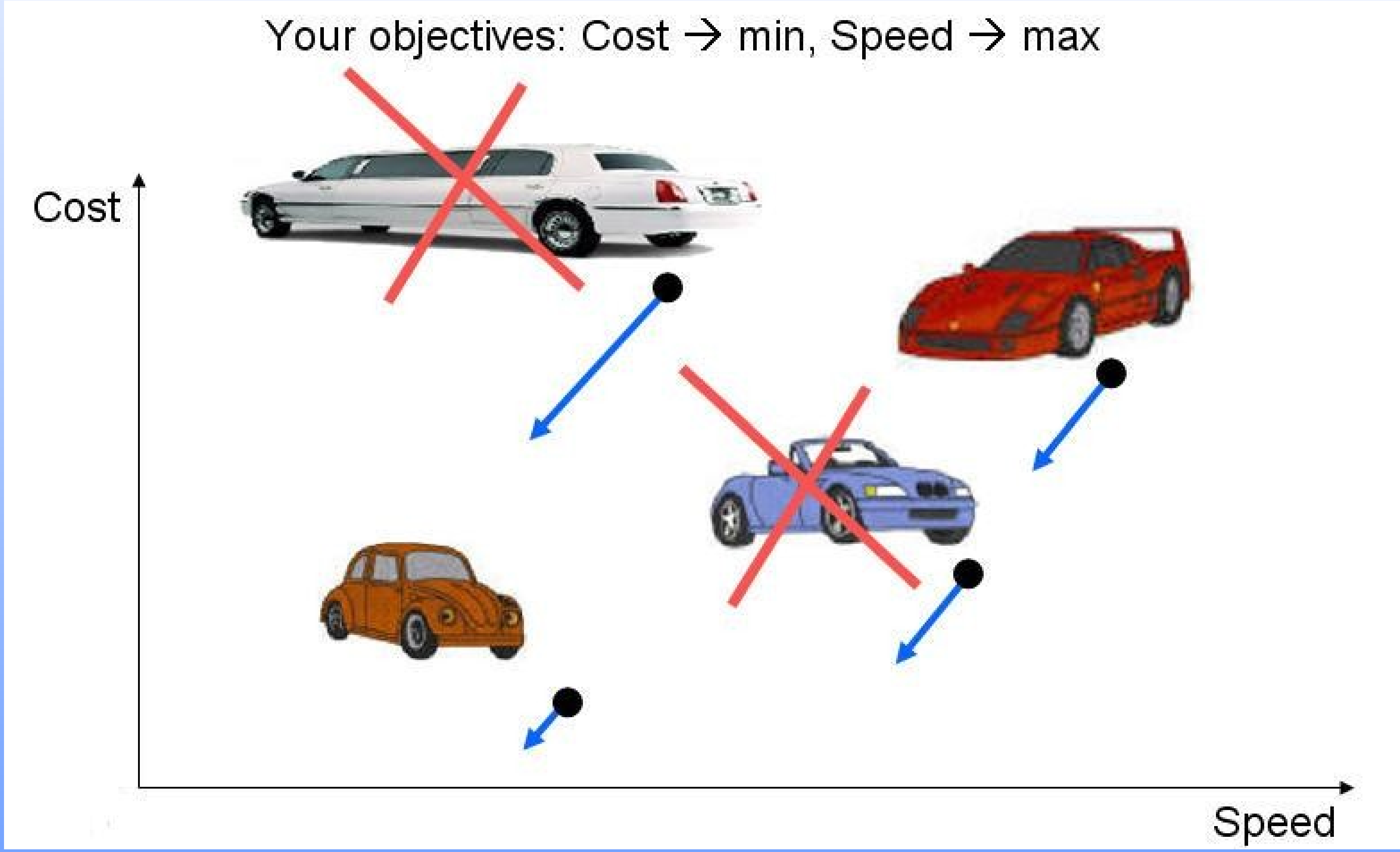
Alexander van der Kuijl
Michael Emmerich
Hui Li

LIACS – Leiden Institute for Advanced Computer Science

Contents

- Multi-Objective Optimization
- Grid Scheduler
- Experiments and Results
- Conclusion and Discussion
- Questions

Multi-Objective Optimization
Introduction



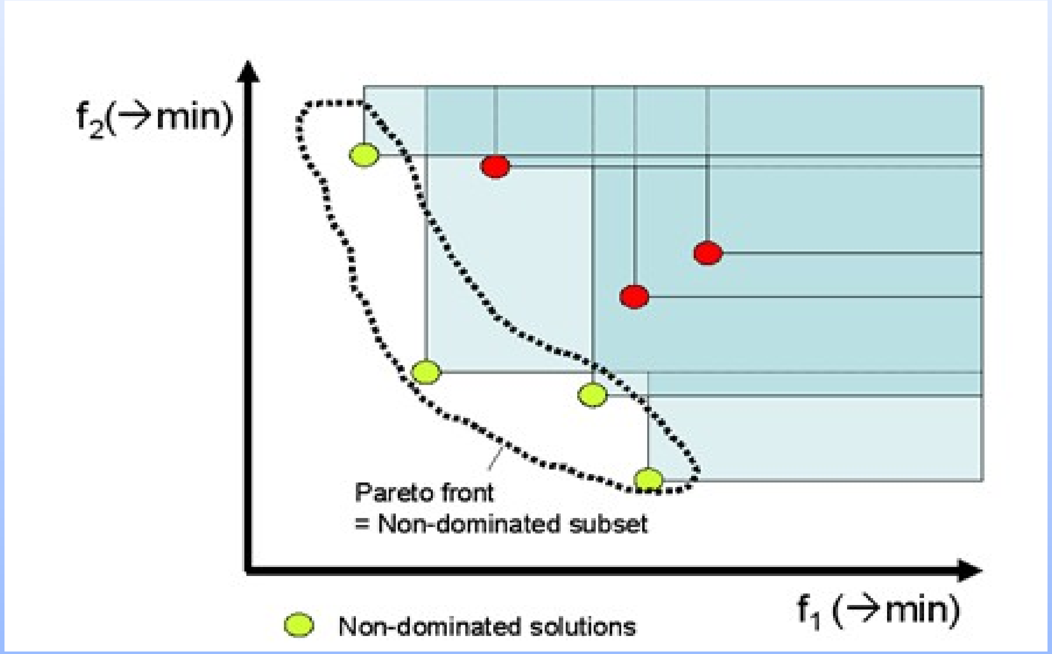
Multi-Objective Optimization
Domination

- Concept of Domination very important

- y dominates y’:

$$y \prec y' \text{ if and only if } \forall\, i \in \{1,...,m\} : y_i \leq y'_i$$
$$\textit{and } y \neq y'$$

Multi-Objective Optimization
Pareto Front



- Pareto Front: set of non-dominated solutions:

$$Y_N = \{y \in Y \mid \neg \exists y' \in Y : y' \prec y\}$$

Multi-Objective Optimization

Optimization

- Goal: To find an as close as possible approximatino of the Pareto Front

- Done with Evolutionary Algorithms due to the typical large amount of possibilities
  - Individuals represent solutions
  - Population of individuals denote set of found solutions

Multi-Objective Optimization SMS-EMOA

- SMS-EMOA uses the dominated hypervolume as selection criterium
  - Favours solutions closer to the Pareto Front
  - Leads to diversity in population

- Translates approximation to Pareto Front to maximizing dominated hypervolume of the population

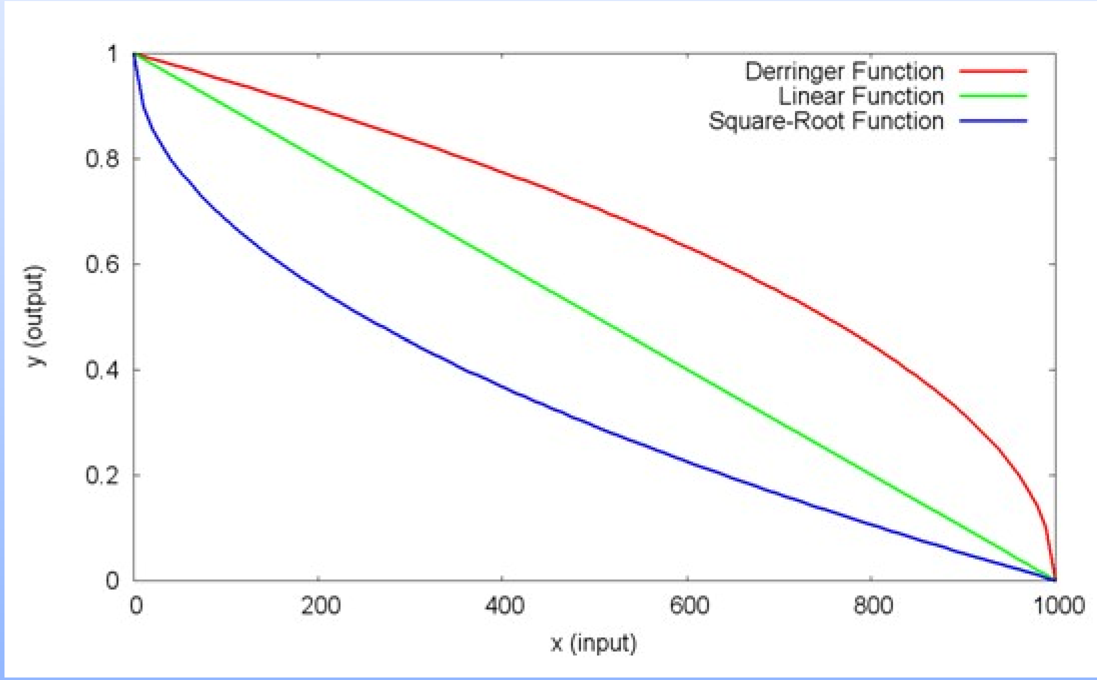
Grid Scheduling General

- Task: to assign a resource to each submitted job to the Grid
- For each submitted job, find most “appropriate” resource
- “Appropriate depends on preferences of for example Virtual Organization in which the Grid Scheduler resides

Grid Scheduling
General

- Grid Scheduler bases decisions on available Performance Indicators (PI's)
  - Response Time
  - Cost
- Translate PI's to appropriateness
- Proposed algorithm:
  use Multi-Attribute Utility Theory MAUT)
  for this translation
- PI's as utilities

Grid Scheduling Utility Functions



- Translates the incoming value to a *Satisfaction Level*, between 0 and 1

Grid Scheduling
MAUT

- Function to combine scoring functions:

$$U(x) = \beta \sum_{i}^{m} (w_i U_i(f_i(x)) + (1 - \beta) \min_{1 \le i \le l}(U_i(x)) \rightarrow \max$$

- First part optimizes total quality
- Second part avoids extreme solutions
- Several parameters to adjust behaviour

- MAUT Scheduling Algorithm

Grid Scheduling
Comparison Algorithm

- Deadline-and-Budget Constrained (DBC) scheduling algorithm used for comparison
  - Already established
  - Multiple objectives

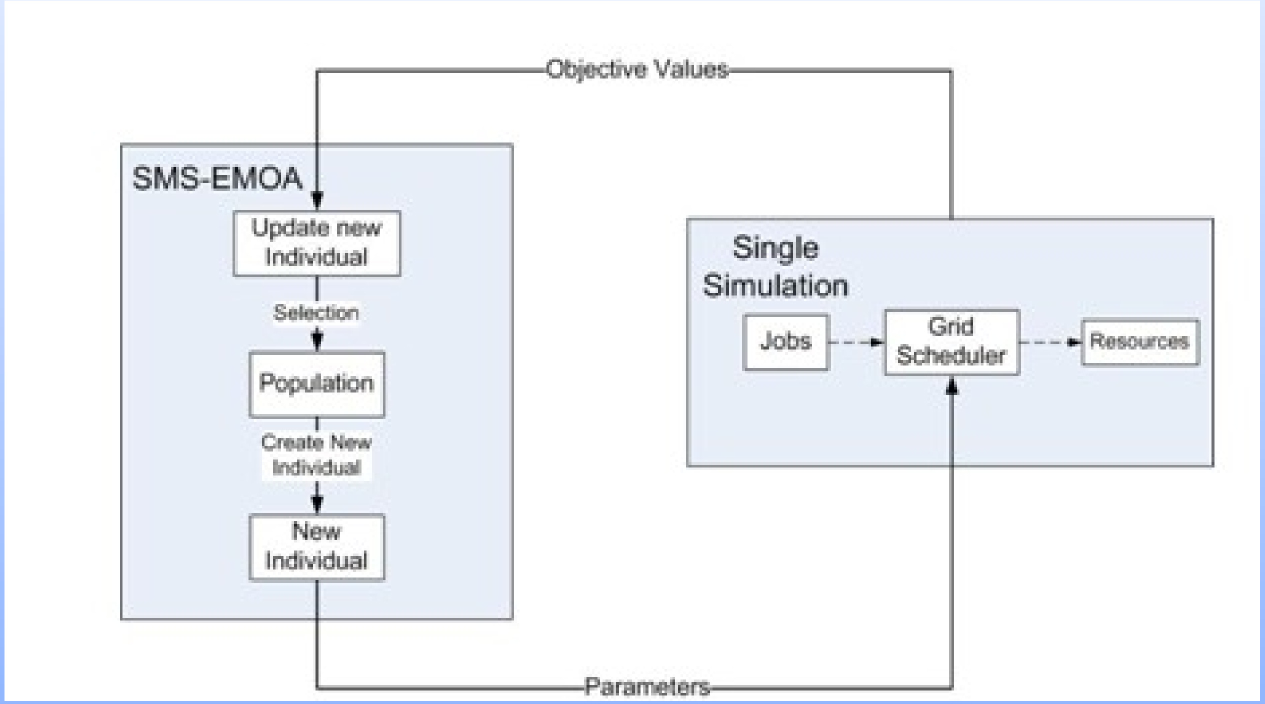
- Deadline for each Job
- Budget for each User

- Select the resource with the lowest associated cost and the response time of the job within the deadline

Grid Scheduling
Comparison Algorithm

- With DBC algorithm it can be the case that not all jobs are executed
- For for comparison with MAUT, mechanism to compensate:
  - Chance Of Execution (Parameter)
- Also applied for DBC for fair comparison

Grid Scheduling
Optimization Process

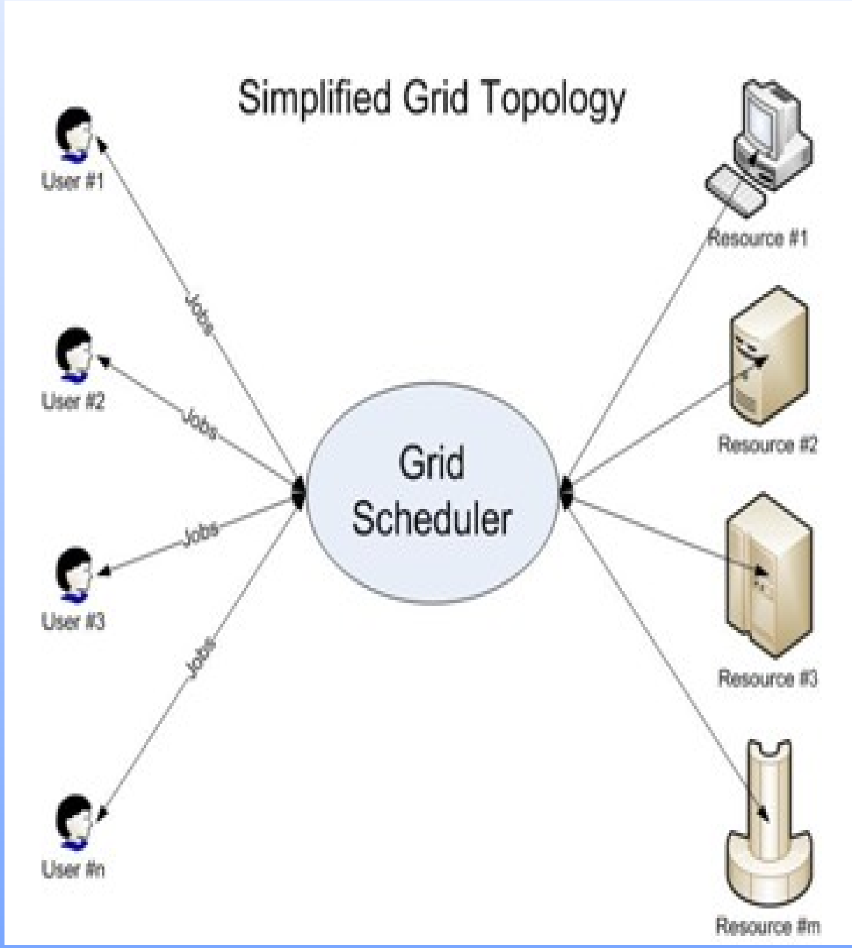


Experiments
General

- Comparison between DBC and MAUT Scheduling algorithm
- As good as possible approximation of the Pareto Front
- For performance measures:
  - Average Response Time (per 100000 MI)
  - Average Cost (per 100000 MI)
  - Total Throughput (MI)

Experiments
Grid Topology

- Central Scheduler
- Space-Shared Resources
- Full knowledge by Scheduler
- Different arrival times of submitted jobs



Experiments
Grid Simulation

- GridSim used as Grid Simulation Tool
  - Open-Source, Java-Based
  - Extendible
- Complete Topology implemented in GridSim
- Scheduler with DBC and MAUT scheduling algorithms implemented

Experiments
Workload

- Workload trace obtained from LHC Computing Grid (LCG)
- Translated to a set of jobs
  - Job Lengths
  - Inter-arrival times

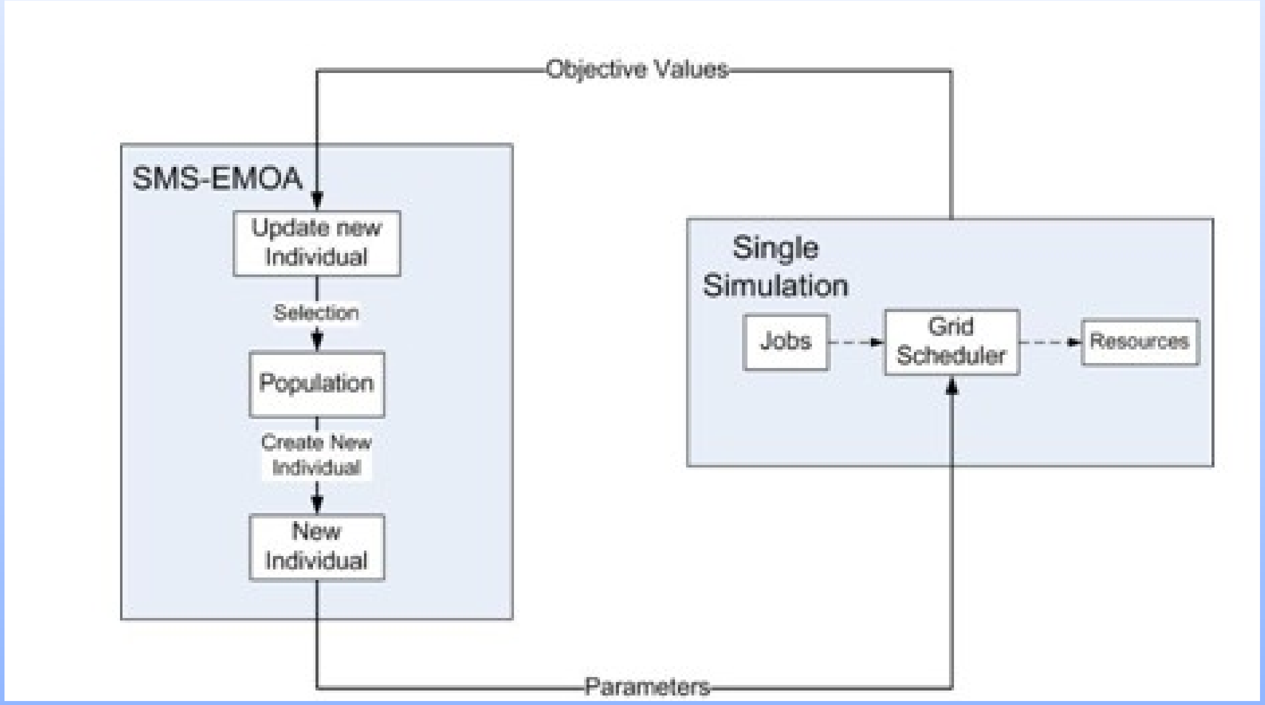
Experiments
Resources

- Based on real-world processors
  - Benchmark
  - Average retail-prices of CPU's
- Added randomly resource to the set until the average utilization was not too high or too low (approx. 25%)
- Same resource set used throughout all experiments

Experiments
Parameters

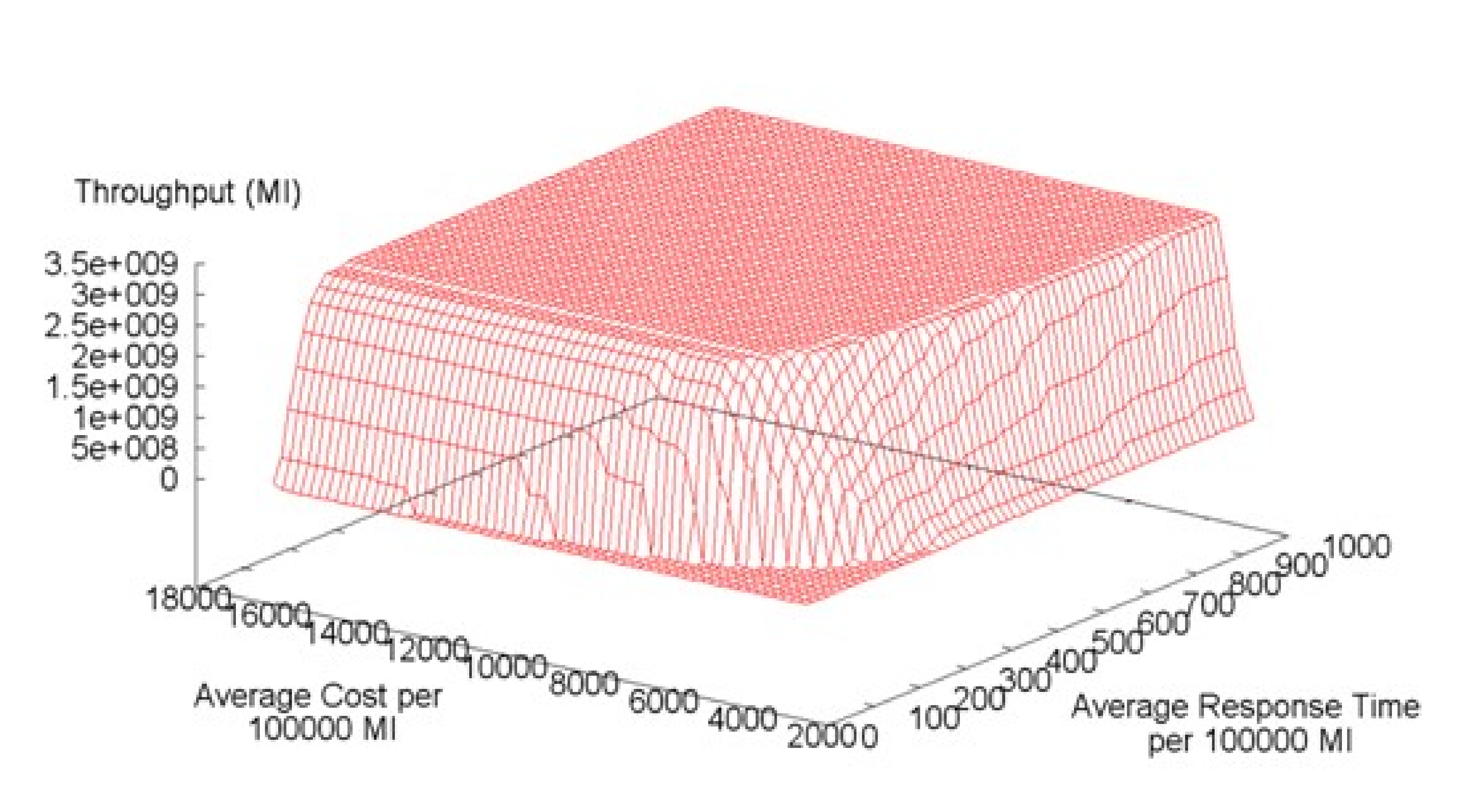
- 5 Parameters for the MAUT algorithm:
  - deadlineMultiplier and maxPrice multiplier
  - 2 Weights to combine utilities
  - ChanceOfExcecution Parameter
- 3 Parameters for the DBC algorithm:
  - deadlineMultiplier
  - budgetMultiplier
  - ChanceOfExecution Parameter

Experiments
Optimization Procerss

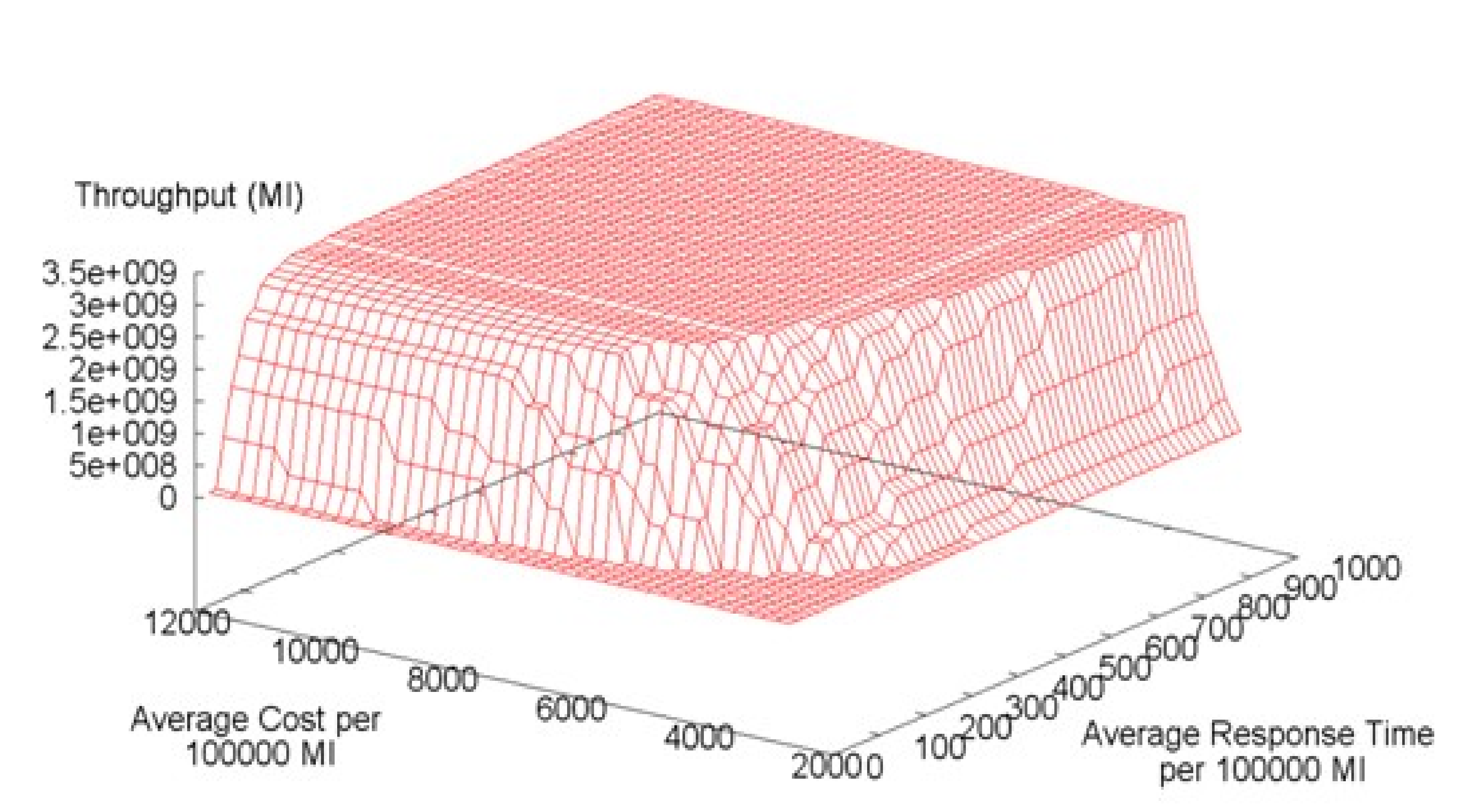


– 5000 Iterations
– 200 individuals in the population

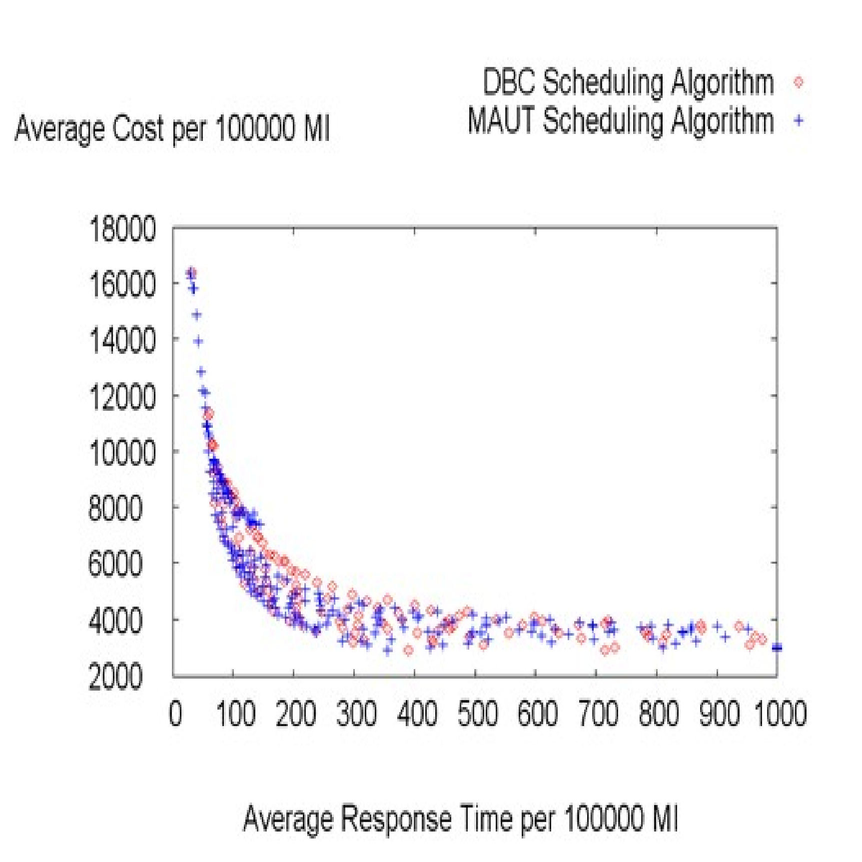
Experiments
Results MAUT

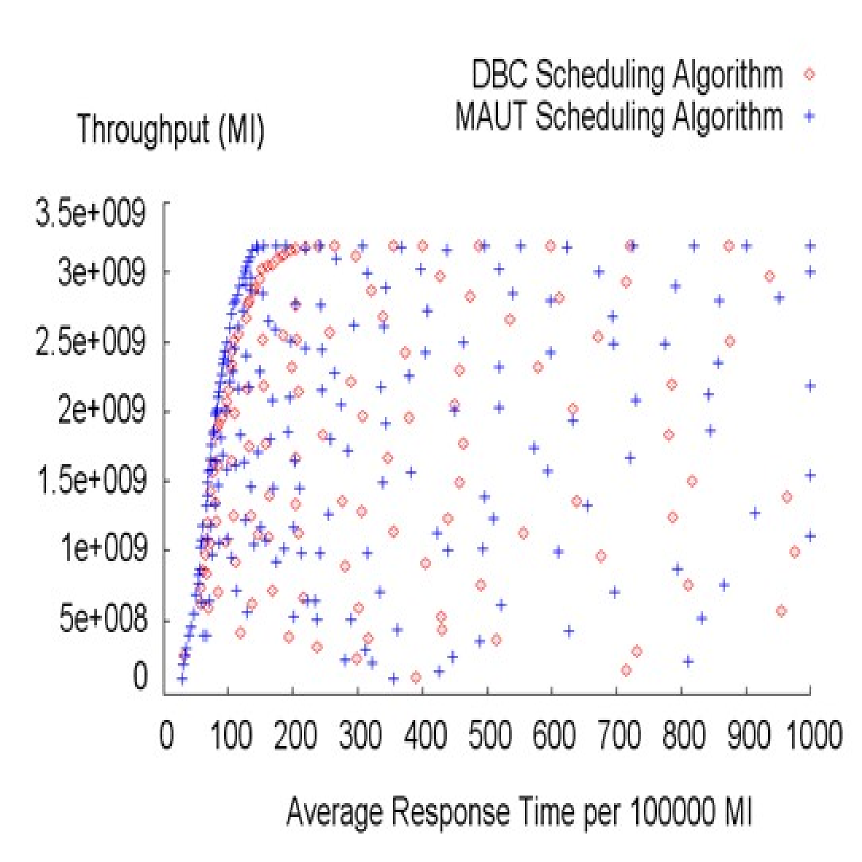


Experiments
Results DBC



Experiments
2-D Projections





Conclusion

- Results MAUT Scheduling are encouraging:
  - Solutions cover larger part of objective space than DBC Scheduling Algorithm
  - Solutions of better quality
- Good trade-off solutions possible

Discussion

- Comparison to other scheduling algorithms
- Applying MAUT Scheduling algorithm to real-world Grid
- Performance under Uncertainty

Questions?