

Integrated Risk Analysis for a Commercial Computing Service

Chee Shin Yeo and Rajkumar Buyya



Grid Computing and **D**istributed **S**ystems (GRIDS) Lab.
Dept. of Computer Science and Software Engineering
The University of Melbourne, Australia

<http://www.gridbus.org>

Problem/Motivation: Commercial Computing Service

- Towards utility computing
 - Service market thru dynamic service delivery
- Commercial computing service
 - Different from non-commercial computing service
 - What objectives to achieve
 - How to identify suitable resource management policies

Related Work

- Cluster Resource Management System (RMS)
 - Condor, LoadLeveler, LSF, PBS, Sun Grid Engine
- Managing risk in computing jobs
 - [Kleban04]: Job delay
 - [Irwin04][Popovici05]: Penalty for job delay
 - [Xiao05]: Loss of profit for conservative providers
- Our work
 - Identify essential objectives for a commercial computing service
 - Evaluate whether these objectives are achieved

Commercial Computing Service: Objectives

■ Service Level Agreement (SLA)

- Different user needs and requirements

$$SLA = \frac{n_{SLA}}{m} * 100$$

■ Reliability

- Guarantee of required service

$$reliability = \frac{n_{SLA}}{n} * 100$$

■ Profit

- Monetary performance

$$profit = \frac{\sum_{i=1}^n utility_i}{\sum_{i=1}^m budget_i} * 100$$

Commercial Computing Service: Risk Analysis

- Separate risk analysis

$$\text{performance, } \mu_{sep} = \frac{\sum_{i=1}^n \text{result}_i}{n}$$

$$\text{volatility, } \sigma_{sep} = \sqrt{\frac{\sum_{i=1}^n (\text{result}_i)^2}{n} - (\mu_{sep})^2}$$

- Integrated risk analysis

$$\text{performance, } \mu_{int} = \sum_{i=1}^n w_i * \mu_{sep,i}$$

$$\text{volatility, } \sigma_{int} = \sum_{i=1}^n w_i * \sigma_{sep,i}$$

Performance Evaluation: Simulation

- GridSim toolkit: Simulated scheduling in a cluster computing environment
(<http://www.gridbus.org/gridsim>)
- Feitelson's Parallel Workload Archive
(<http://www.cs.huji.ac.il/labs/parallel/workload>)
 - Last 5000 jobs in SDSC SP2 trace (3.75 mths)
 - Average inter arrival time: 1969 s (32.8 mins)
 - Average run time: 8671 s (2.4 hrs)
 - Average number of requested processors: 17
- SDSC SP2
 - Number of computation nodes: 128

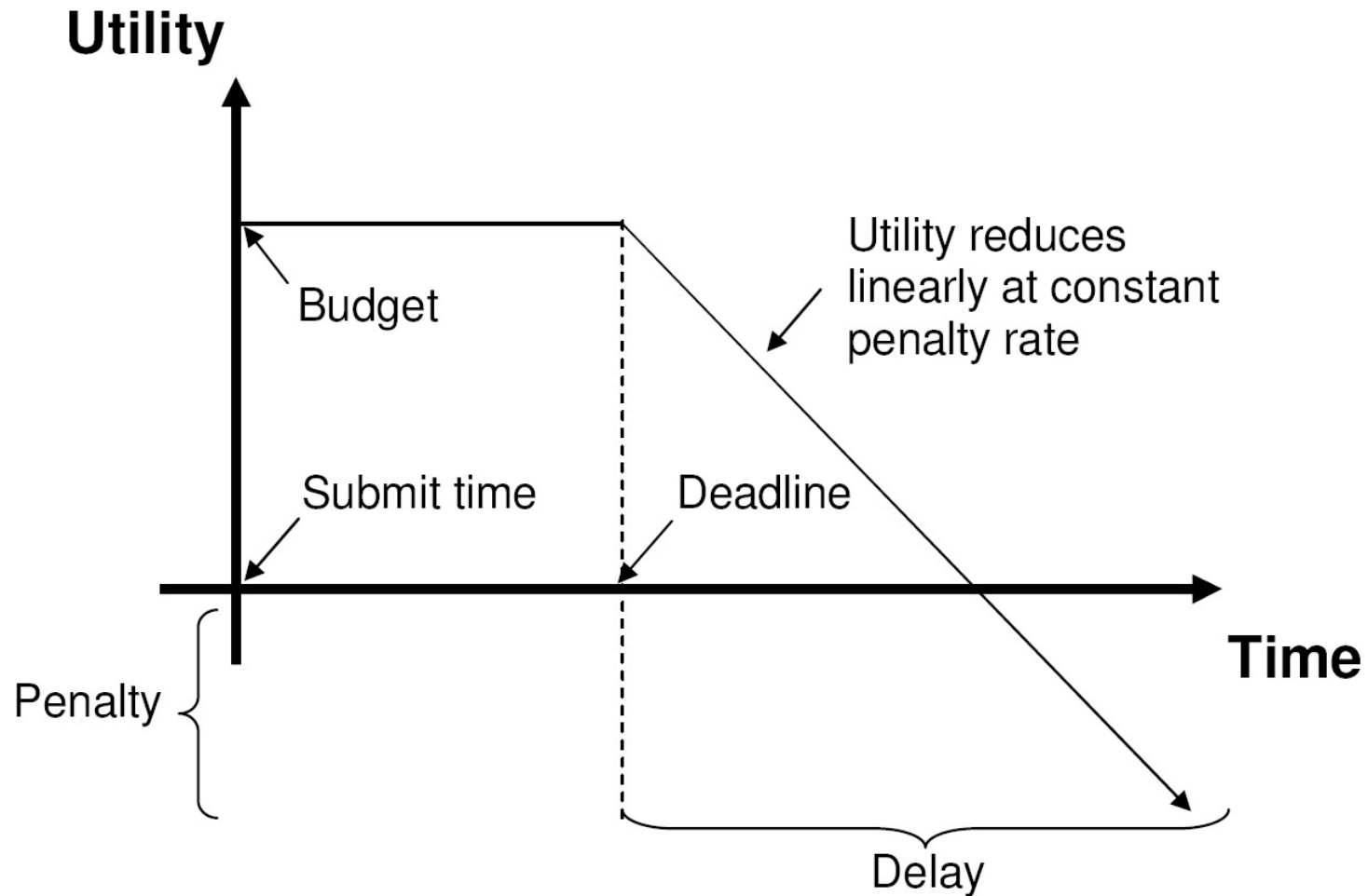
Performance Evaluation: Simulation Settings

- Modeling deadline, budget, penalty QoS [Irwin04]
- High urgency jobs
 - LOW deadline/runtime, HIGH budget/runtime, HIGH penalty/runtime
 - Values normally distributed in each HIGH & LOW set
 - Randomly distributed in arrival sequence
- High:Low ratio
 - Ratio of means for HIGH and LOW deadline/runtime, budget/runtime, penalty/runtime

Performance Evaluation: Simulation Settings

- Bias parameter
 - Deadline, budget, penalty not always set as a larger factor of runtime.
- Arrival delay factor
 - Model cluster workload thru job inter arrival time
- Actual runtime estimates from trace
 - Inaccurate

Performance Evaluation: Simulation Settings



Performance Evaluation: Policies

- First Come First Serve Backfilling (FCFS-BF)
Earliest Deadline First Backfilling (EDF-BF)
 - Space-shared with EASY backfilling
 - FCFS (arrival time), EDF (deadline)
 - Admission control reject job only prior to execution (not submission)
- FirstReward [Irwin04]
 - Space-shared
 - Reward based on possible future earnings & opportunity cost penalties (thru weighting function)
 - Admission control based on slack threshold – high avoids future commitments with possible penalties
 - Accurate runtime estimates & no backfilling

Performance Evaluation: Policies

- **Libra [Sherwani04]**
 - Time-shared (Deadline-based proportional processor share)
 - Suitable node if deadline of all jobs met
 - Best fit strategy (least available processor time after accepting new job)
 - Accurate runtime estimates
- **LibraRisk**
 - Libra's Deadline-based proportional share
 - Suitable node if zero risk of deadline delay for all jobs
 - Inaccurate runtime estimates

Performance Evaluation: Scenarios & Metrics

Parameter	Default value	
	Set A	Set B
% of high urgency jobs	20	same
% of low urgency jobs	80	same
Deadline bias	1	14
Deadline high:low ratio	4	same
Deadline low mean	4	same
Budget bias	1	same
Budget high:low ratio	4	same
Budget low mean	4	same
Penalty bias	1	same
Penalty high:low ratio	4	same
Penalty low mean	4	same
Arrival delay factor	1	same

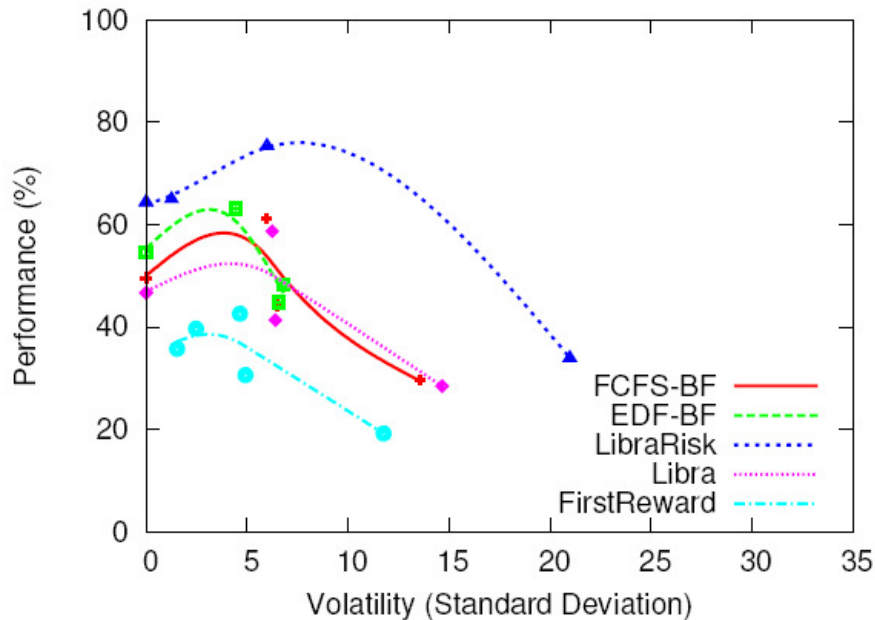
Scenario	Varying value	
	Set A	Set B
Deadline bias	1	10
	2	12
	4	14
	6	16
	8	18
	10	20

$$SLA = \frac{n_{SLA}}{m} * 100$$

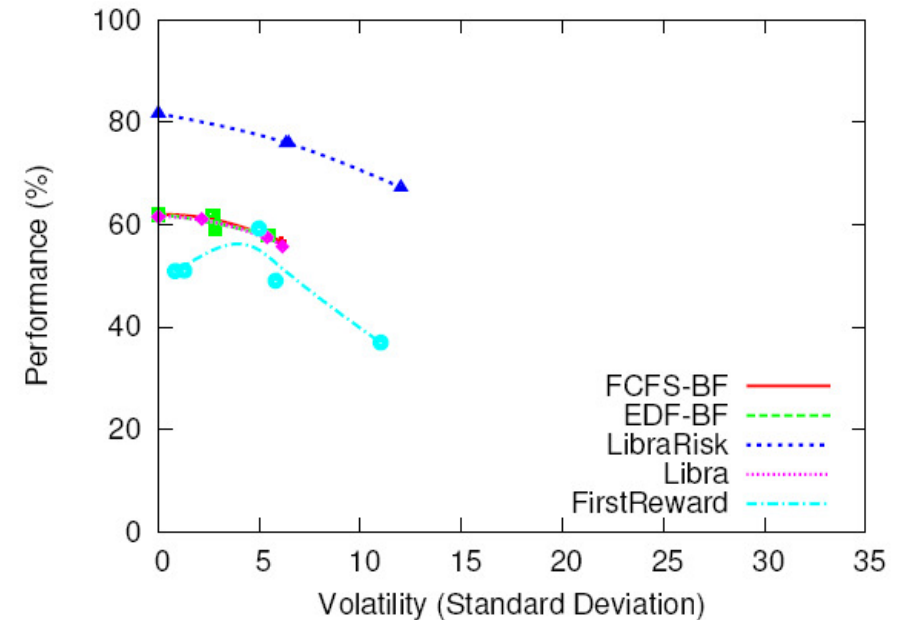
$$reliability = \frac{n_{SLA}}{n} * 100$$

$$profit = \frac{\sum_{i=1}^n utility_i}{\sum_{i=1}^m budget_i} * 100$$

Separate Risk Analysis of 1 Objective: SLA



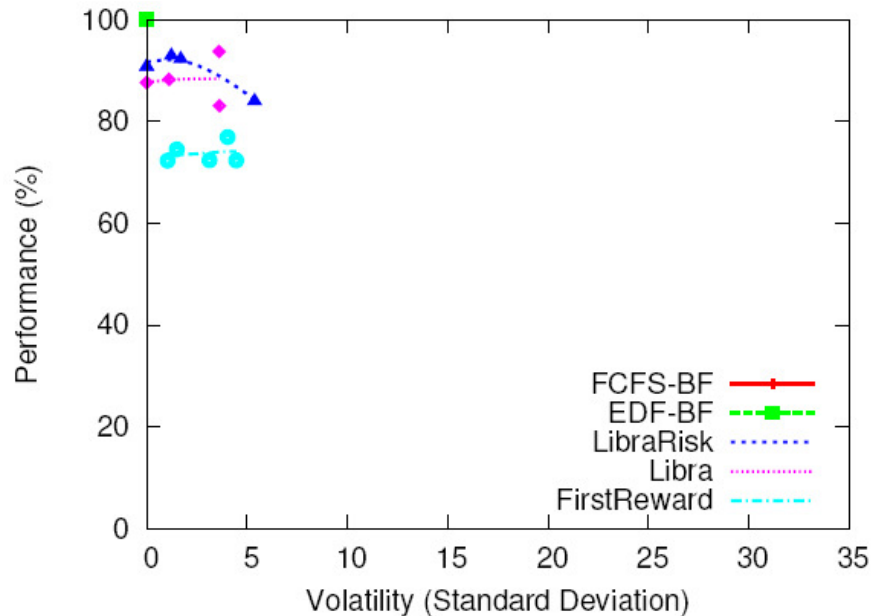
(a) Set A: SLA



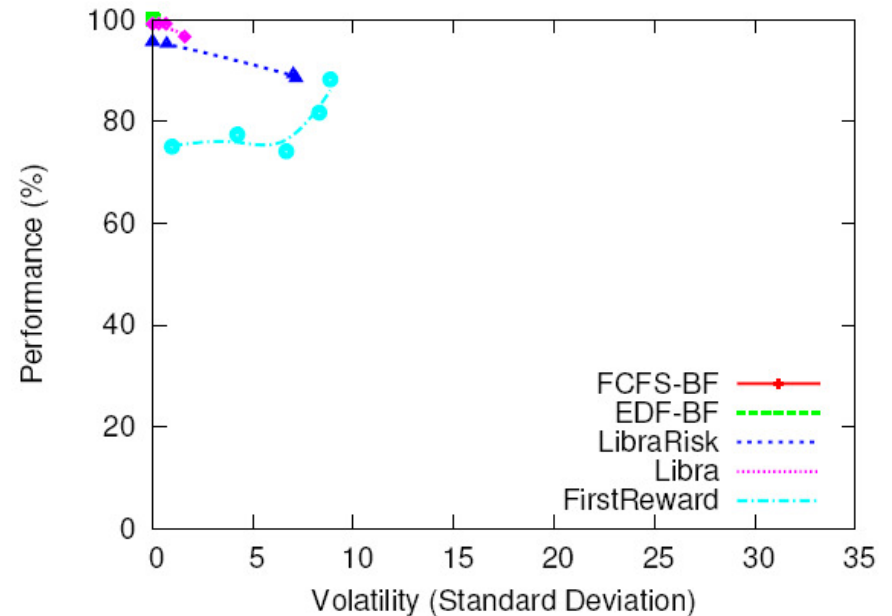
(b) Set B: SLA

- FCFS-BF & EDF-BF: Deadline bias
- LibraRisk: Highest performance & volatility
- Libra & LibraRisk: Exploit changes in deadlines

Separate Risk Analysis of 1 Objective: Reliability



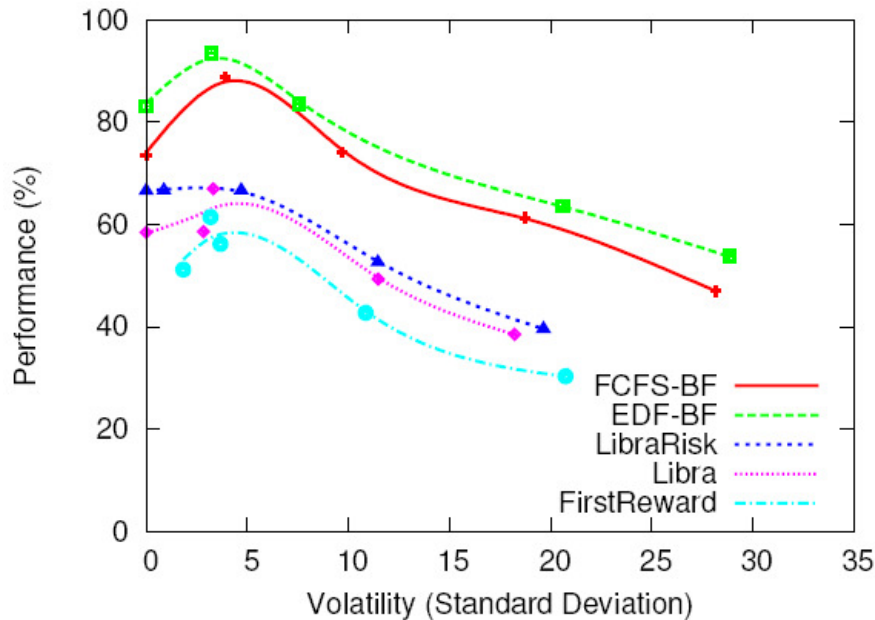
(c) Set A: Reliability



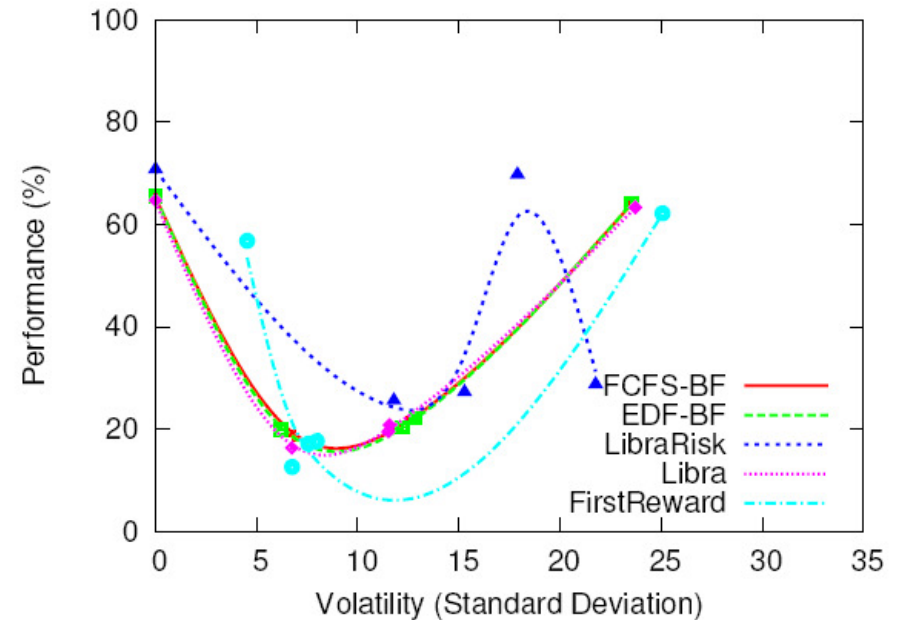
(d) Set B: Reliability

- FCFS-BF & EDF-BF: Generous admission control
- FirstReward: More jobs delayed with lower penalty

Separate Risk Analysis of 1 Objective: Profit



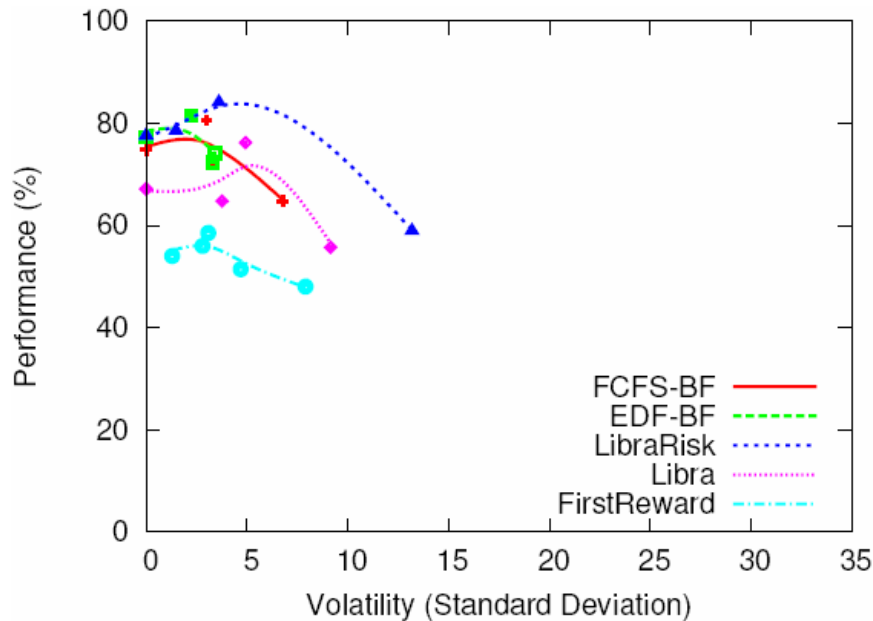
(e) Set A: Profit



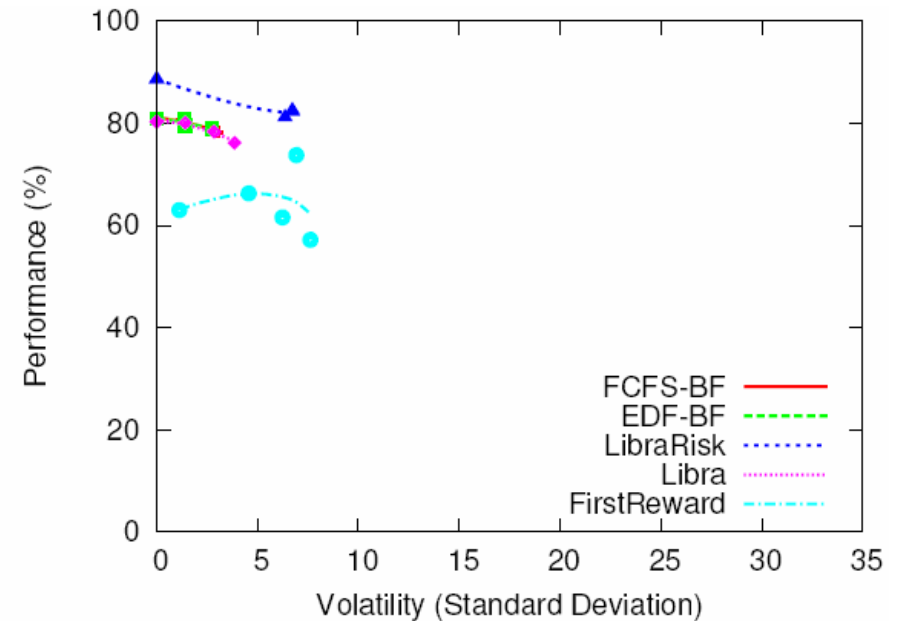
(f) Set B: Profit

- FCFS-BF & EDF-BF: Better without deadline bias
- LibraRisk: Better than Libra for high deadline bias
- FirstReward: No backfilling

Integrated Risk Analysis of 2 Objectives: SLA + Reliability



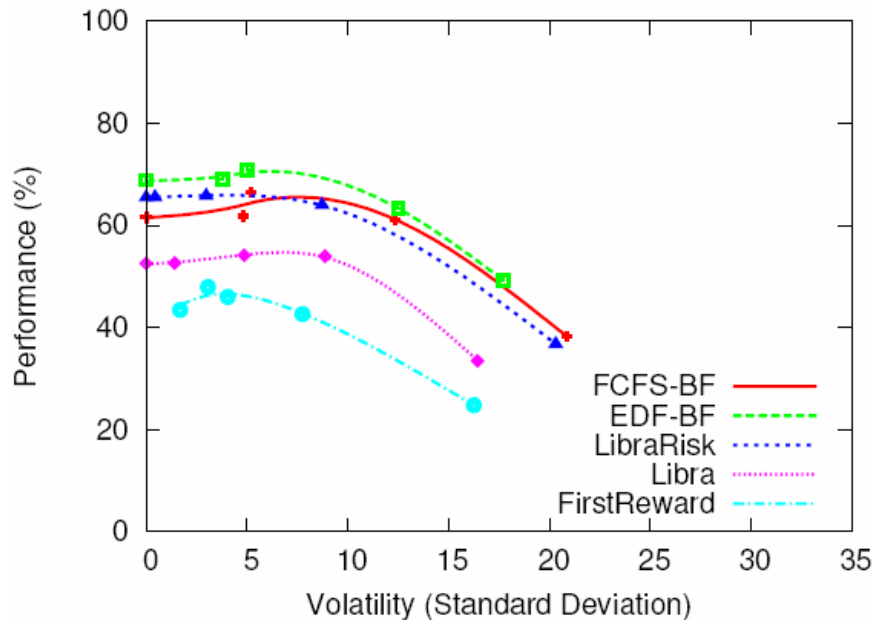
(a) Set A: SLA + Reliability



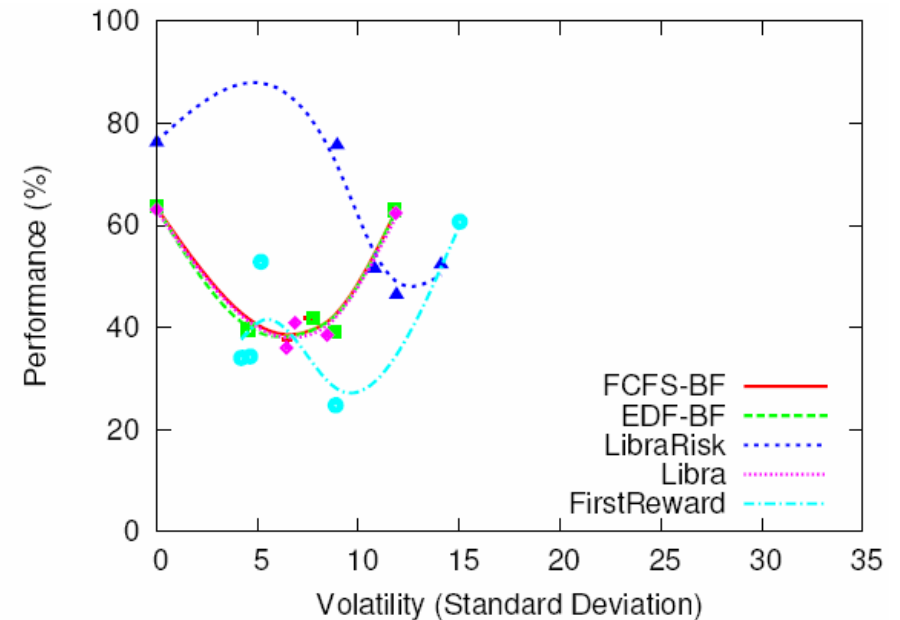
(b) Set B: SLA + Reliability

- LibraRisk: Highest performance & volatility
- FCFS-BF, EDF-BF & Libra: Similar

Integrated Risk Analysis of 2 Objectives: SLA + Profit



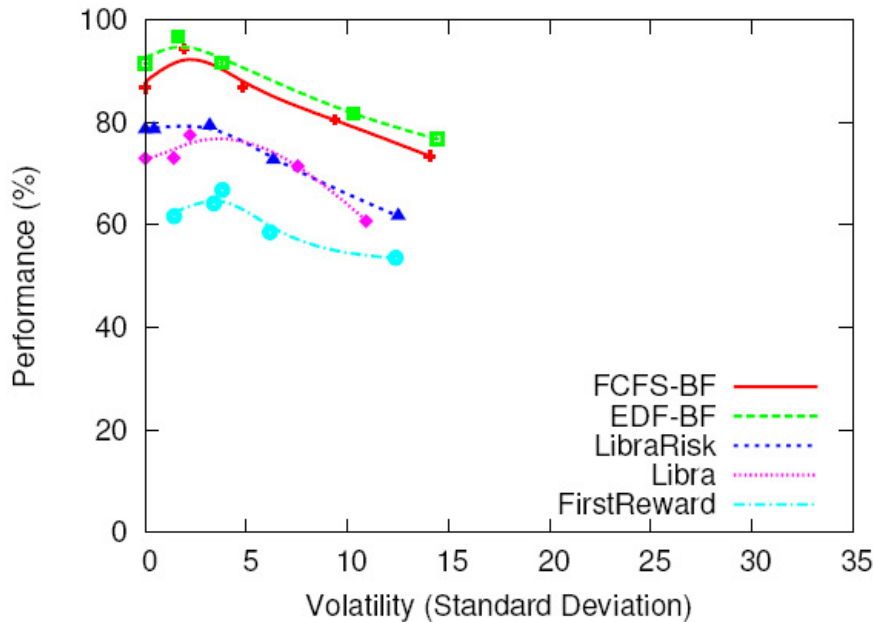
(c) Set A: SLA + Profit



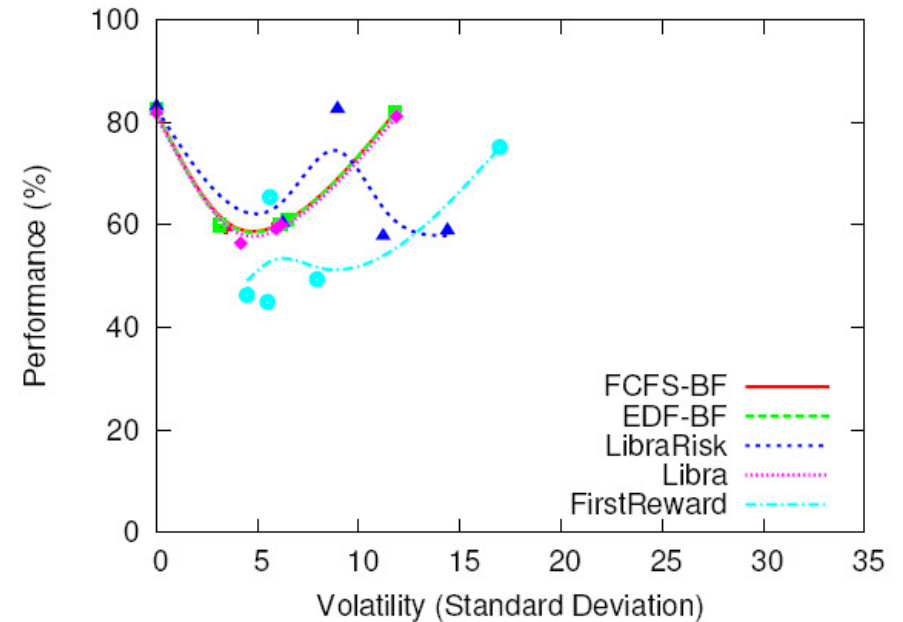
(d) Set B: SLA + Profit

- **LibraRisk: Better performance due to high SLA**
- **Others: Worse performance for high deadline bias**

Integrated Risk Analysis of 2 Objectives: Reliability + Profit



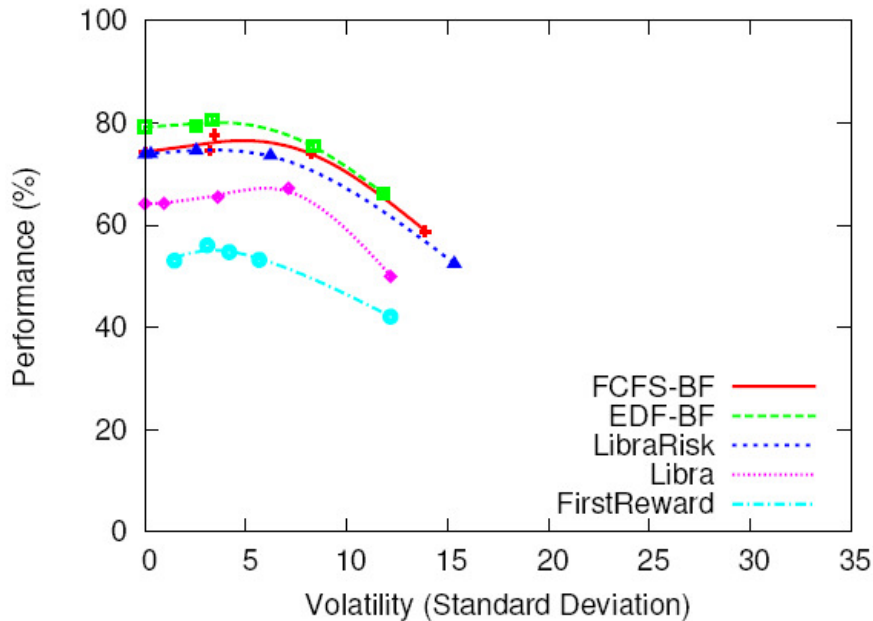
(e) Set A: Reliability + Profit



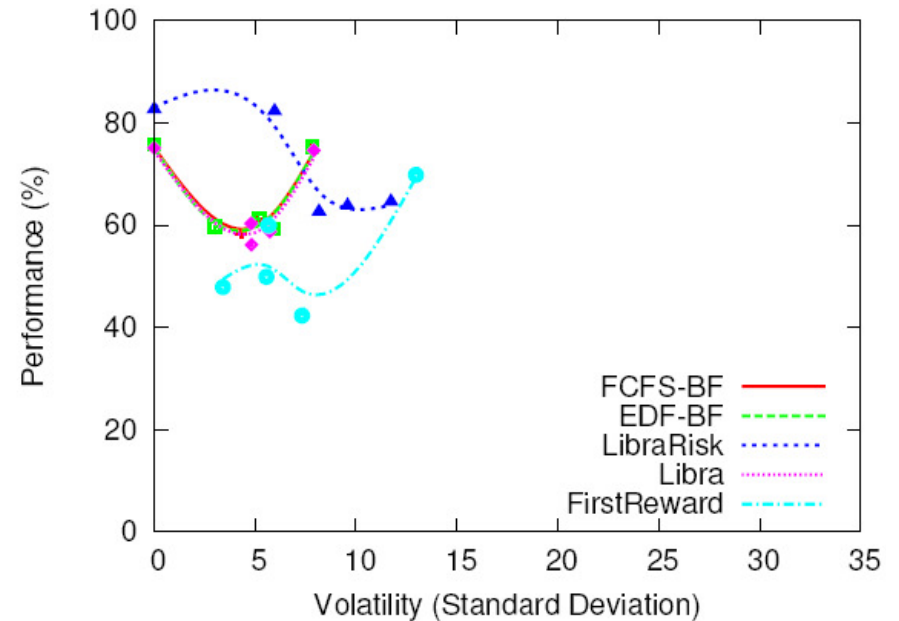
(f) Set B: Reliability + Profit

- FCFS-BF & EDF-BF: Best without deadline bias
- LibraRisk & FirstReward: Higher volatility with high deadline bias

Integrated Risk Analysis of 3 Objectives: SLA + Reliability + Profit



(a) Set A: SLA + Reliability + Profit



(b) Set B: SLA + Reliability + Profit

- FCFS-BF & EDF-BF: Best without deadline bias
- LibraRisk: Better than Libra thru risk of deadline delay & best with deadline bias

Conclusion

- 3 essential objectives
 - SLA, reliability & profit
- Evaluation of policies
 - Separate & integrated risk analysis
- Importance of identifying and analyzing achievement of objectives
 - Impact by under-achieved objectives

End of Presentation



Questions ?