ExPERT: Pareto-efficient task replication on grids and clouds

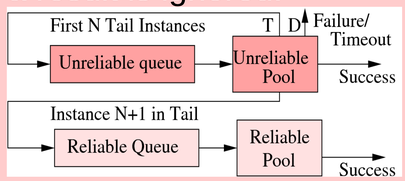
Orna Agmon Ben-Yehuda, Assaf Schuster, Artyom Sharov, Mark Silberstein and Alexsandru Iosup
(**ladypine**,assaf,sharov,marks) @cs.technion.ac.il and a.iosup@tudelft.nl, Technion CS Technical Report CS-2011-03

Grid+Cloud BoT Deployment

BoTs (bags of asynchronous tasks) are often run on unreliable grids. To cope with unreliability, users replicate tasks and use expensive reliable resources: clouds. Different users want to optimize different utility functions: cost, makespan (time needed to complete an entire BoT), or **any** function of the two. ExPERT finds **efficient** task replication strategies which are the best for **any** user utility function.

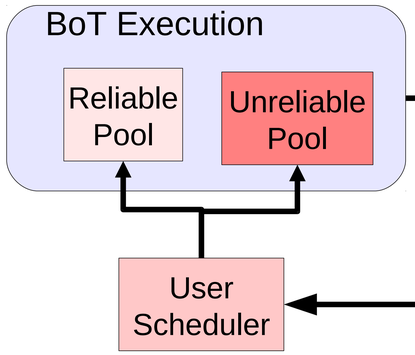
The NTDM$_r$ Strategy Space

N – Number of task instances sent to the grid. A final instance (N+1) is sent to the cloud.
T – Timeout (min wait time between instances)
D – Deadline after which a result is ignored.
M$_r$ – Reliable pool's Max concurrent size (fraction of the unreliable pool's size) .



User parameters

- Cost per time on reliable, unreliable pools.
- Average task runtimes.
- M_r^{max} - Hard limit on pool size ratio (cloud/grid).



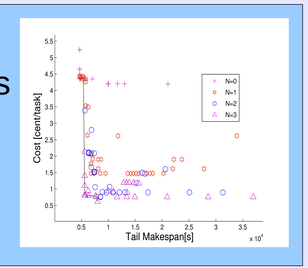
ExPERT

Statistical Characterization

The final ("tail") tasks, for which optimization is crucial, are characterized by statistical data collected during execution of the previous tasks. Grid reliability and CDF of successful tasks turnaround time are extracted from this data and used to simulate strategies for the "tail" tasks.

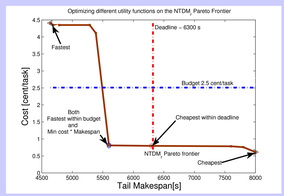
Pareto Frontier Generation

NTDM$_r$ space is sampled. Each strategy is Monte-Carlo simulated to estimate cost and makespan. Inefficient strategies are filtered out.



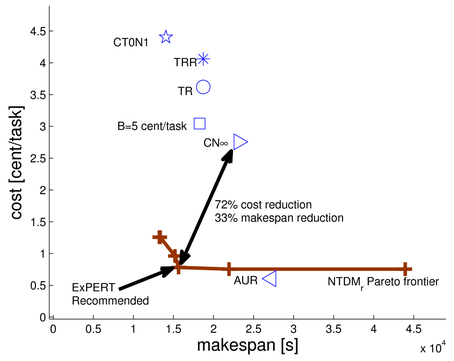
Decision Making

The best strategy on the Pareto frontier is chosen according to **any** user preference.



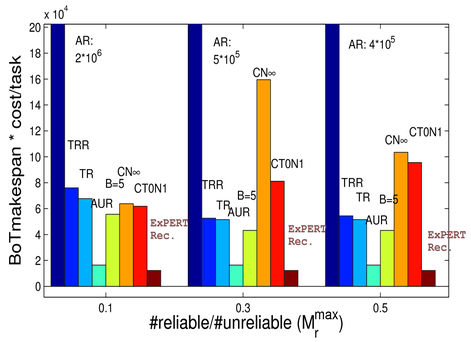
Pareto Frontier vs. Static Strategies

NTDM$_r$ Pareto frontier is more efficient than most strategies.



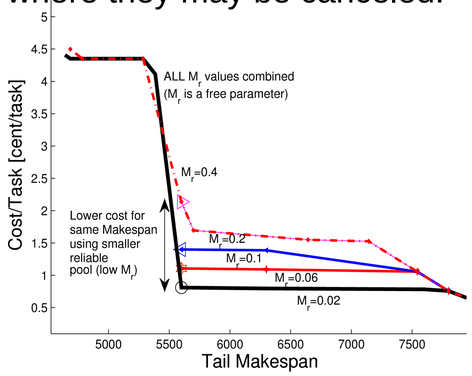
Cost X Time

Cost X time of ExPERT-recomended strategy is 25% lower than second-best and at least 72% lower than third-best.



Limiting Clouds

The free parameter M$_r$ enables efficient strategies with lower costs for the same makespan. It makes tasks wait in a queue, where they may be canceled.



Conclusion

- The NTDM_r strategy space is vast enough to provide user preference flexibility.
- ExPERT recommended strategies finish in two-thirds of the time and cost a quarter of commonly-used static strategies.

- **Using ExPERT means you do not waste time or money, and you optimize your own utility function!**