

ExPERT: Pareto-efficient task replication on grids and a cloud

Orna Agmon Ben-Yehuda¹ Assaf Schuster¹
Artyom Sharov¹ Mark Silberstein¹ Alexandru Iosup²

¹Department of Computer Science
Technion — Israel Institute of Technology

²Faculty of Engineering, Mathematics and Computer Science (EWI)
TU Delft

IPDPS, May 2012

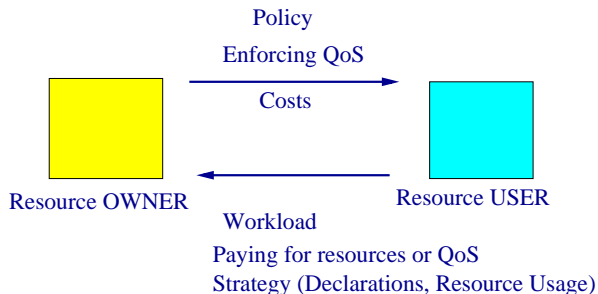
The Shared Resource Game — Players and Goals

Owner goals – minimize:

- *Operational costs (energy)
- *Effective load

User goals – minimize:

- *Makespan
- *Cost



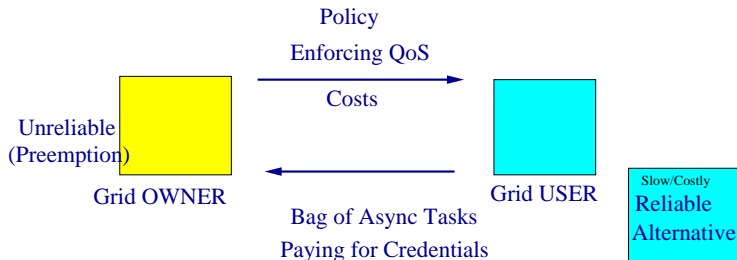
The Unreliable Shared Resource Game

Owner goals – minimize:

- *operational costs (energy)
- *effective load

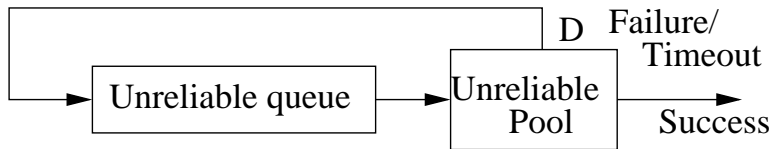
User goals – minimize:

- *Makespan
- *Cost



An environment of uncertainty: Will the task fail on the unreliable resource? Which system to use?

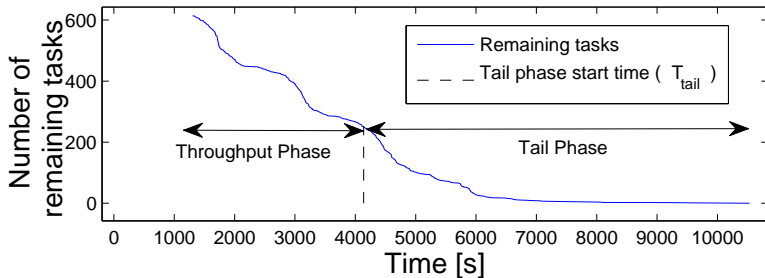
In the Beginning...



- $\#machines < \#unfinished\ tasks.$
- D - instance deadline.
- No replication (replication is inefficient).

Using the Same Strategy After the Tail Starts

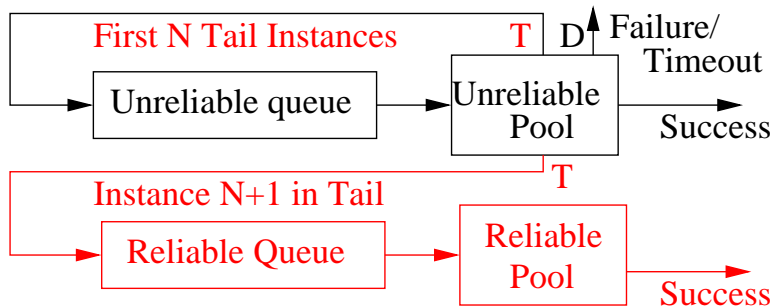
#machines > #unfinished tasks



The tail is wagging the dog...

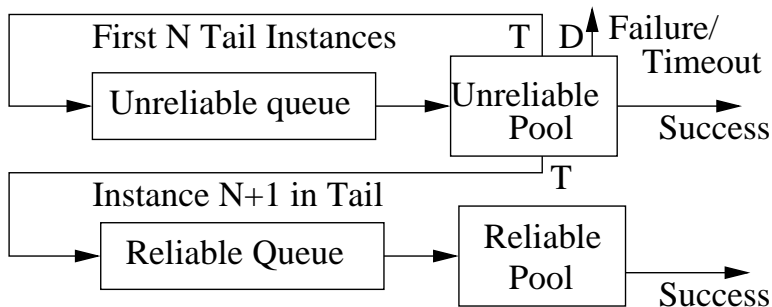


Replication - the User's Bank of $NTDM_r$ Strategies



- D - instance deadline, T - replication time
- Reliable machine used to ensure task completion
- N tail instances at most on unreliable resources
- M_r - max ratio of reliable to unreliable resources

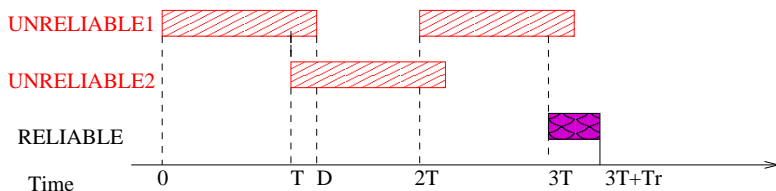
Replication - the User's Bank of $NTDM_r$ Strategies



- D - instance deadline, T - replication time
- Reliable machine used to ensure task completion
- N tail instances at most on unreliable resources
- M_r - max ratio of reliable to unreliable resources

Replication - the User's Bank of $NTDM_r$ Strategies

- Example: Number of unreliable instances $N = 3$



- Replication wastes work!

The User's Problem: Optimization of...

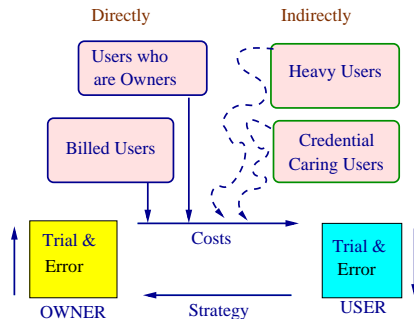
The user cares about multi-objective optimization:

- $\langle \text{Cost} \rangle$ - Mean $\frac{\text{cost}}{\text{task}}$ or $\frac{\text{tail}-\text{cost}}{\text{tail}-\text{task}}$
- $\langle MS \rangle$ - Mean makespan or tail makespan.

Each user may have her own objective, pending on those values:

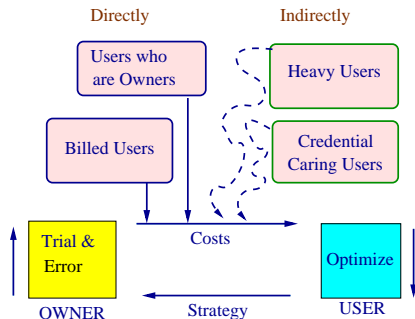
- Below minimal makespan: $\langle MS \rangle < \text{Const}$
- As fast as possible: $\min \langle MS \rangle$
- Below max budget: $\langle \text{Cost} \rangle < \text{Const}$
- As cheap as possible: $\min \langle \text{Cost} \rangle$
- Best price for the goods: $\min \langle \text{Cost} \rangle \langle MS \rangle$
- Any other function of means: $\langle \text{Cost} \rangle, \langle MS \rangle, \dots$

The Feedback Loop



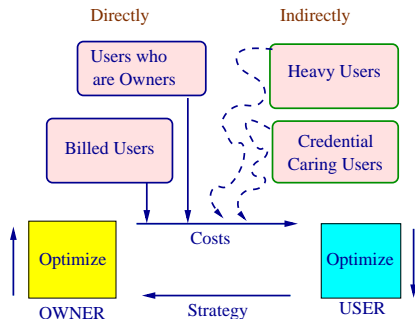
Users who do not optimize well behave irrationally and are hard to predict.

The Feedback Loop — Our Contribution



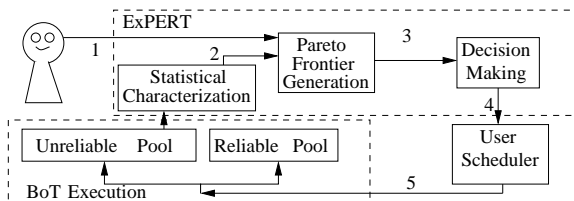
Rational users can optimize general utility function.

The Feedback Loop - Lookout



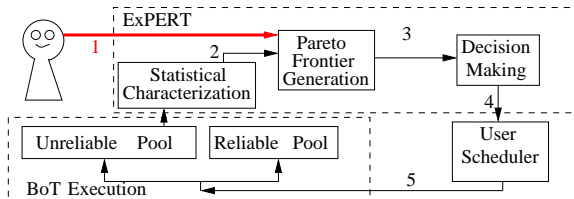
Towards the final goal of manipulating users to save energy

Solution Concept

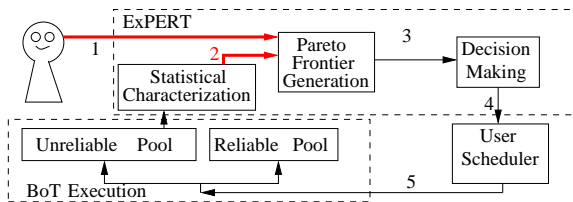


Solution Concept - Step 1

Get **user additional data** (costs, reliable pool times).

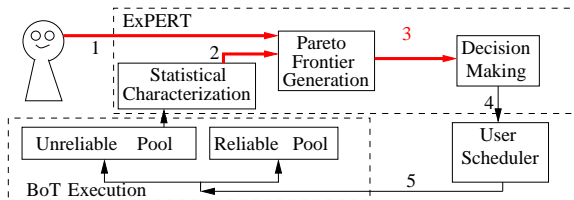


Get unreliable resource statistics (trace analysis).



Solution Concept - Step 3

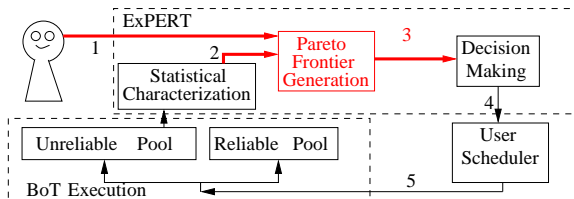
Compute a **Pareto frontier** for $\langle \text{Cost} \rangle, \langle \text{MS} \rangle$.



Solution Concept - Step 3

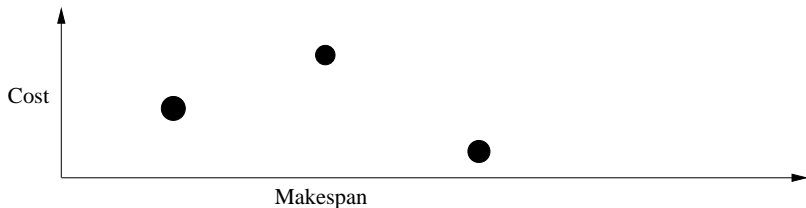
Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space:

- For every working point, the *ExPERT Estimator* computes several random realizations on the basis of the statistic characterization.
- The average makespan and cost over these realizations are used as the expectation values $\langle Cost \rangle, \langle MS \rangle$.



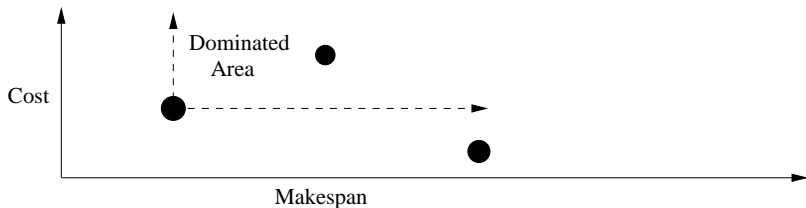
Solution Concept - Step 3

- Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space.
- Filter out dominated strategies.
- Keep frontier composed of non-dominated strategies.



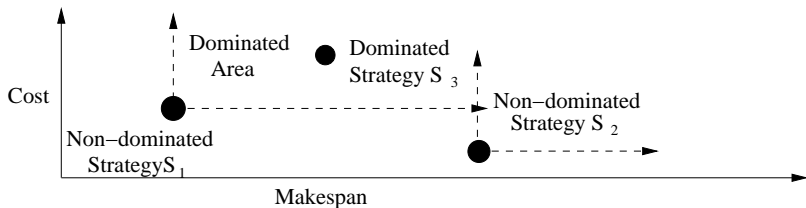
Solution Concept - Step 3

- Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space.
- Filter out dominated strategies.
- Keep frontier composed of non-dominated strategies.



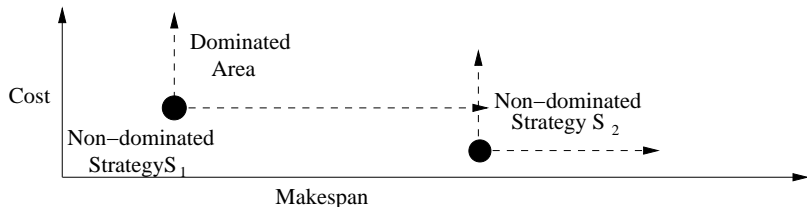
Solution Concept - Step 3

- Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space.
- Filter out dominated strategies.
- Keep frontier composed of non-dominated strategies.



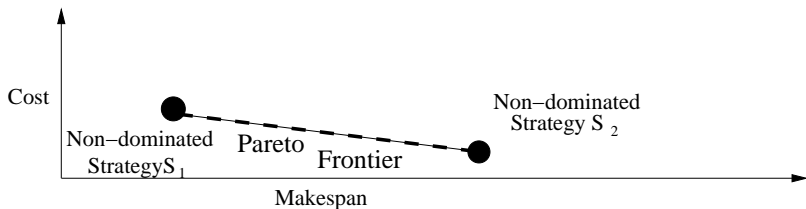
Solution Concept - Step 3

- Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space.
- Filter out dominated strategies.
- Keep frontier composed of non-dominated strategies.



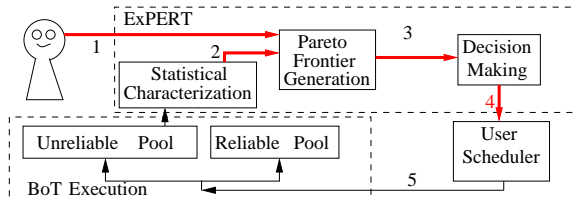
Solution Concept - Step 3

- Estimate $\langle Cost \rangle, \langle MS \rangle$ for each strategy in the search space.
- Filter out dominated strategies.
- Keep frontier composed of non-dominated strategies.



Solution Concept - Step 4

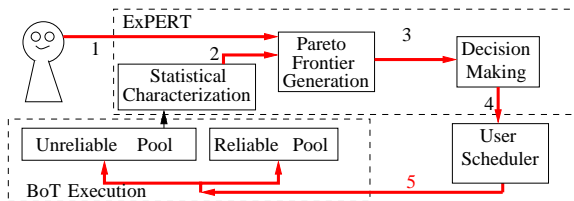
Choose **optimal** strategy according to user utility (by **expectation value**).



Get N , T , D , M_r params for the desired strategy.

Solution Concept - Step 5

Apply strategy: Feed N , T , D , M_r params as input to the user scheduler and deploy tasks on the resource pools.



Example - Based on a GridBoT Trace on UW-M

GridBoT:

- Supplies a unified front-end to multiple grids and clouds using their local resource management infrastructure.
- Employs dynamic run-time scheduling and replication strategies to execute BoTs in multiple environments simultaneously.

A BoT trace holds a line per task with the following fields:

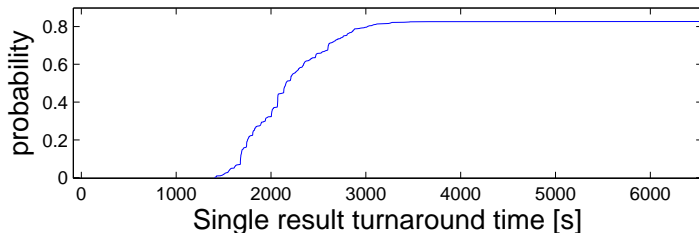
- Status (failed/succeeded)
- Runtime : only for successful tasks.
- Wait time : from submitting to starting running. May be unavailable for failed tasks.

Result time= Runtime+Wait time

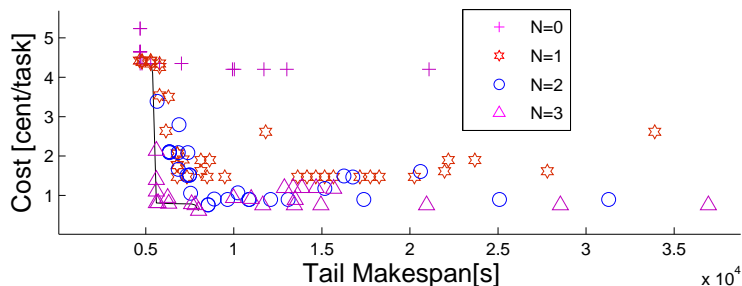
UW-M: Condor cluster of University of Wisconsin-Madison.

Characterize the Unreliable Resource

- $\#ur$: the *effective* size of the unreliable pool.
- $F(t, t') = \text{reliability}(t') \cdot F_s(t)$, CDF of result turnaround time, on the basis of:
 - $F_s(t)$: the measured CDF of result turnaround time (t) of *successful* tasks.
 - $\text{reliability}(t')$: the fraction of successful tasks as a function of the time since the BoT started t' .

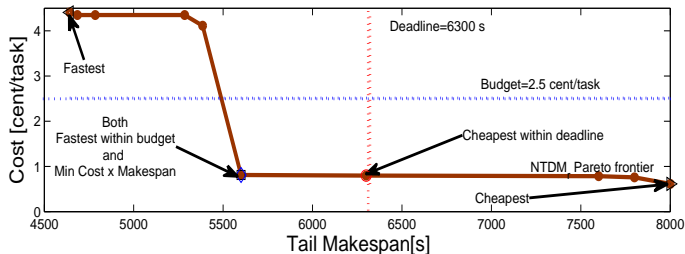


Pareto Frontier and Working Points



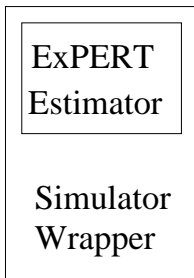
- Local optimization is not trivial.
- Unoptimized strategies are wasteful.

Optimizing a General User Utility Function along the Pareto Frontier

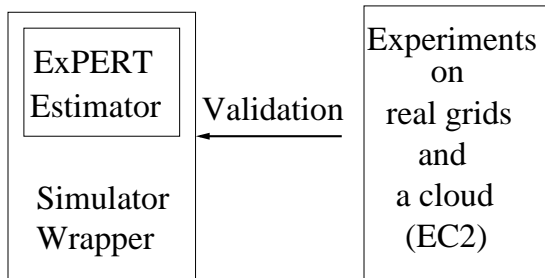


Utility function choice can be postponed till the user is aware of the cost-makespan tradeoff. It does not have to be expressed.

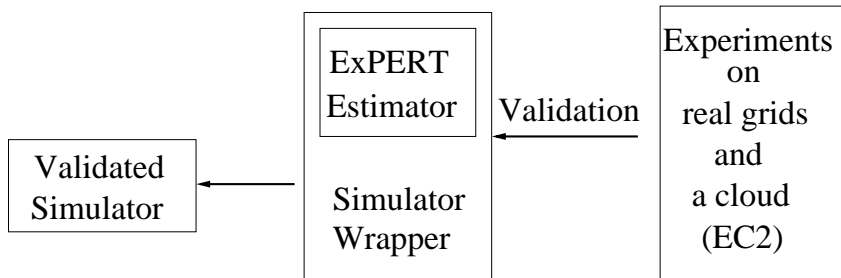
ExPERT
Estimator



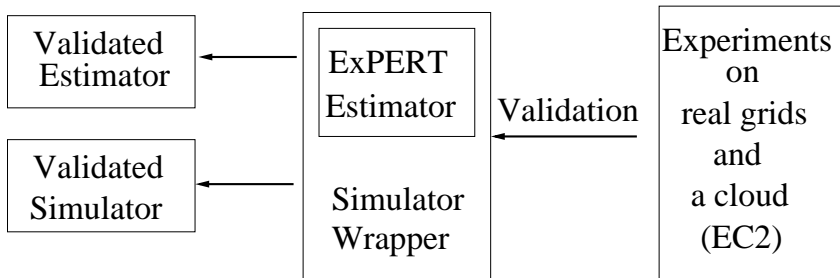
Validation and Evaluation



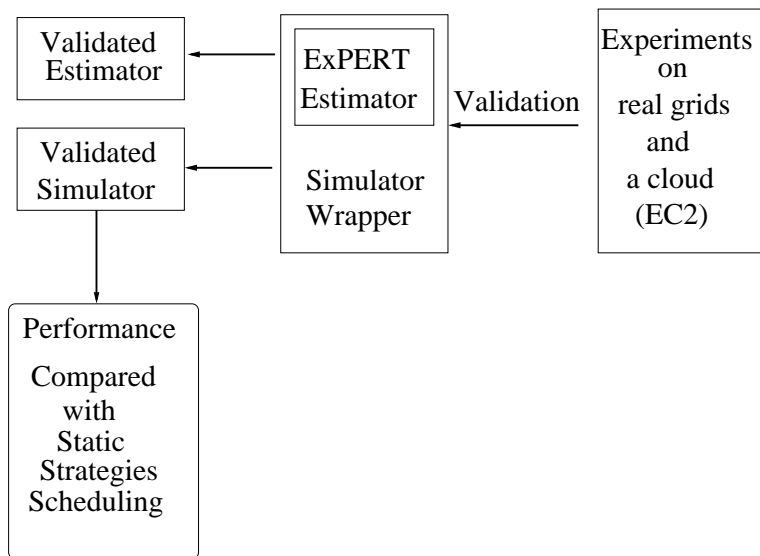
Validation and Evaluation



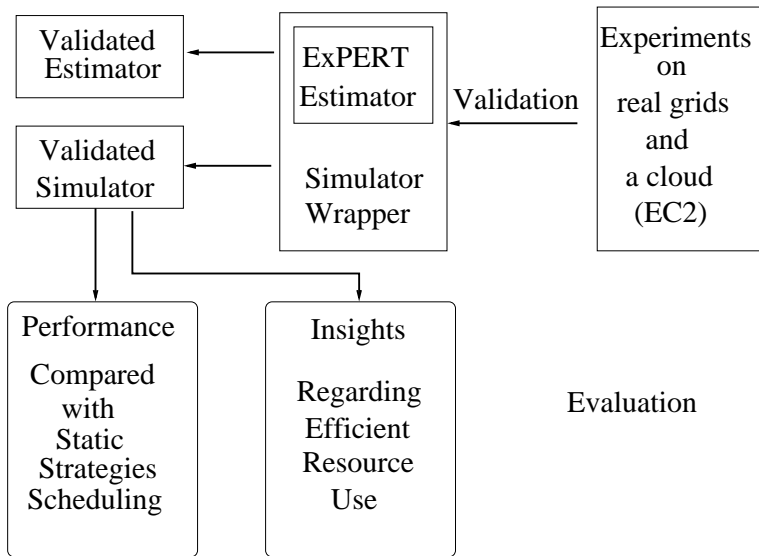
Validation and Evaluation



Validation and Evaluation



Validation and Evaluation

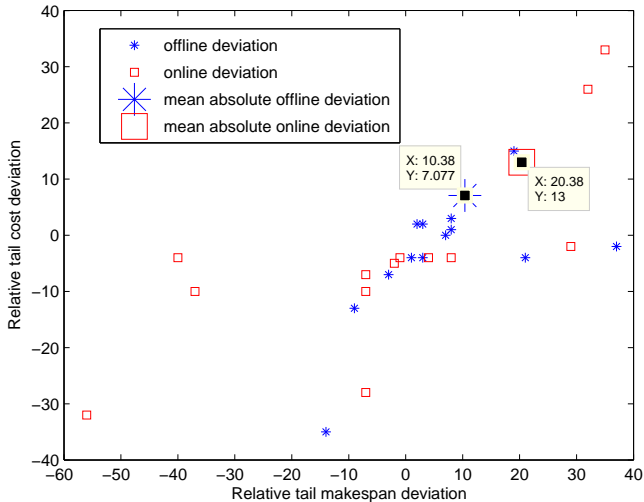


Evaluation

Table: Resource Pools

Reliable Pool	Properties
Technion EC2	20 self-owned CPUs in the Technion. 20 large EC2 cloud instances.
Unreliable Pool	Properties
UW-M	UW-Madison Condor pool (preempts).
OSG	Open Science Grid (no preemption).
UW-M + OSG	Combined: half μ_{ur} from each pool.
UW-M + EC2	Combined: 200 UW-M, 20 EC2.
UW-M + Technion	Combined: 200 UW-M, 20 Technion.

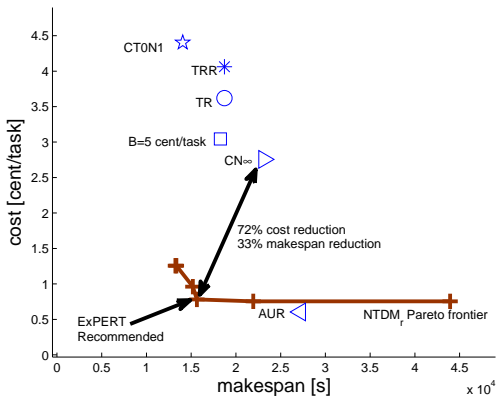
Validation: Prediction Deviation



Augmenting the Estimator with Static Strategies

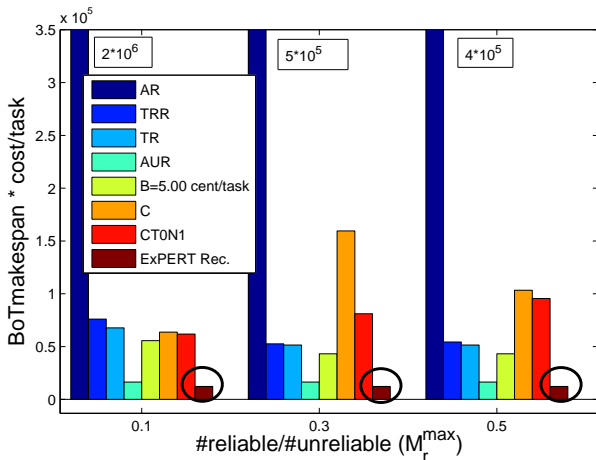
- 1 *AR: All to Reliable*
- 2 *TRR: all Tail Replicated to Reliable ($N=0, T=0$)*
- 3 *TR: all Tail to Reliable ($N = 0, T = D$)*
- 4 *AUR: All to UnReliable, no replication*
- 5 *B: Budget of 1\$ for a BoT of 150 tasks (on average,
 $\frac{2 \text{ cent}}{3 \text{ BoTtask}}$)*
- 6 *CN ∞ : Combine resources, no replication*
- 7 *CT0N1: Combine resources, replicate at tail with $N = 1,$
 $T = 0$*

Bi-Objective Performance, $M_r^{max} = 0.1$



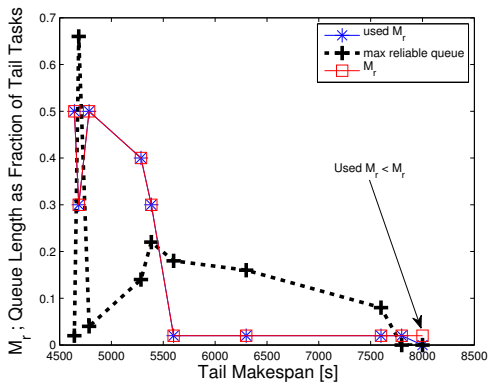
ExPert's Pareto frontier dominates most static strategies.

Performance: makespan \times cost, 3 M_r^{max} values



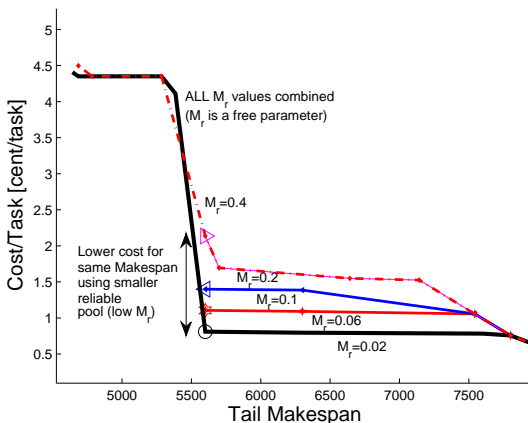
Smaller is better. Cost \times makespan ExPERT's recommended strategy is 25% lower than second-best and at least 72% lower than third -best.

Insight: Efficient Reliable Resource Use Includes a Queue



On the Pareto frontier, usually all reliable resources are used at some point, and a significant queue is built for them.

Insight: the Importance of M_r as a Free Parameter



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.

- The NTM_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.

Contact us at:

- Orna Agmon Ben-Yehuda ladypine@cs.technion.ac.il
- Assaf Schuster assaf@cs.technion.ac.il
- Artyom Sharov sharov@cs.technion.ac.il
- Mark Silberstein marks@cs.technion.ac.il
- Alexandru Iosup A.iosup@tudelft.nl

Thank You!

夫人杉木

