Providing SLOs for Resource-Harvesting VMs in Cloud Platforms

李超

2021/12/14

Reference: Ambati’s Slides

目录

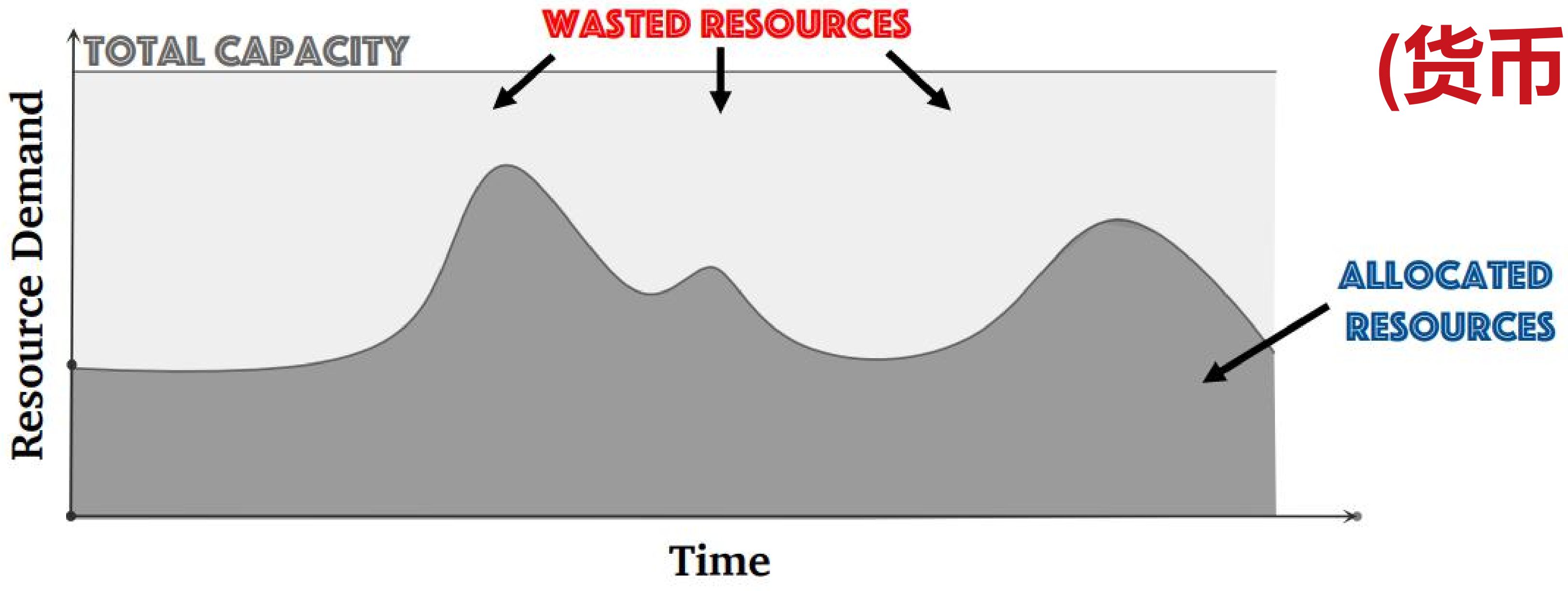
01 Evictable VMs	02 Characterization	03 Harvest VMs
04 SLOs for Harvest VMs	05 Harvest Hadoop	06 Lessons and Conclusion

Public Cloud Platforms (Azure)

- Cloud Platforms offer compute resources as **virtual machines (VMs)**
  - Users can keep the VMs from seconds to years and request more VMs
  - Cloud platforms provide **illusion of infinite scalability** (无限拓展的假象)

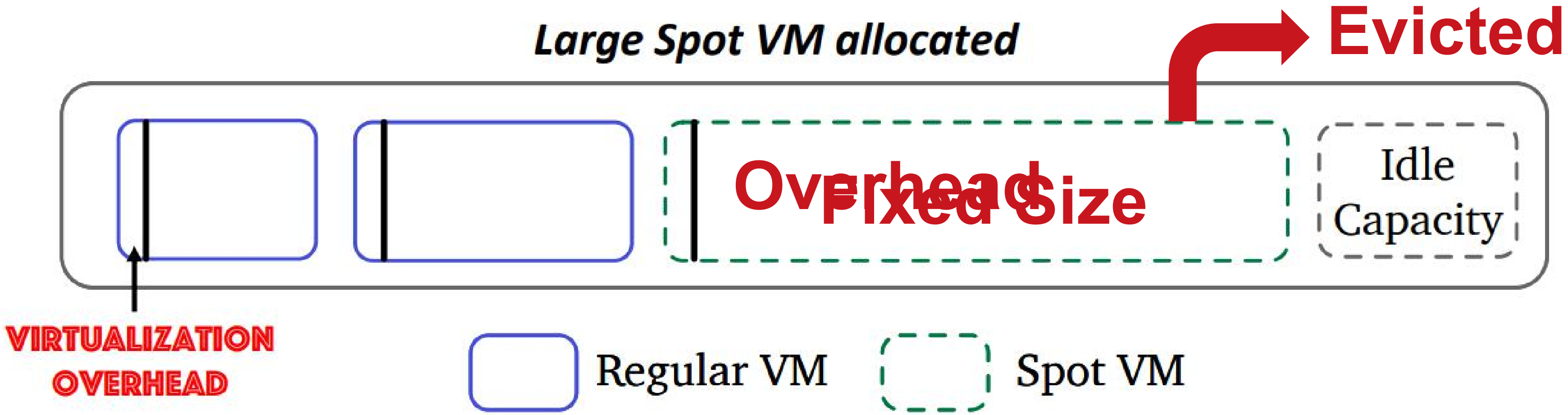
    to allow user growth, handle hardware failures etc.

Hard to monetize (货币化) capacity



Evictable VMs

- **Unallocated resources** are leveraged as spot VMs with relaxed Service Level Objectives (SLOs)
  - Spot VMs can be **revoked** (evicted, 撤回) anytime for regular-priority VMs
  - They cost 50%~90% less than regular-priority VMs



Proposal: Harvest VMs and SLOs for Them

- Harvest VM - a new class of **evictable** VMs

  - Allocated with a minimum size (physical resources)

  - Dynamically **grows and shrinks** to harvest unallocated resources

  - Only evicted if its **minimum size** is needed for a regular VM

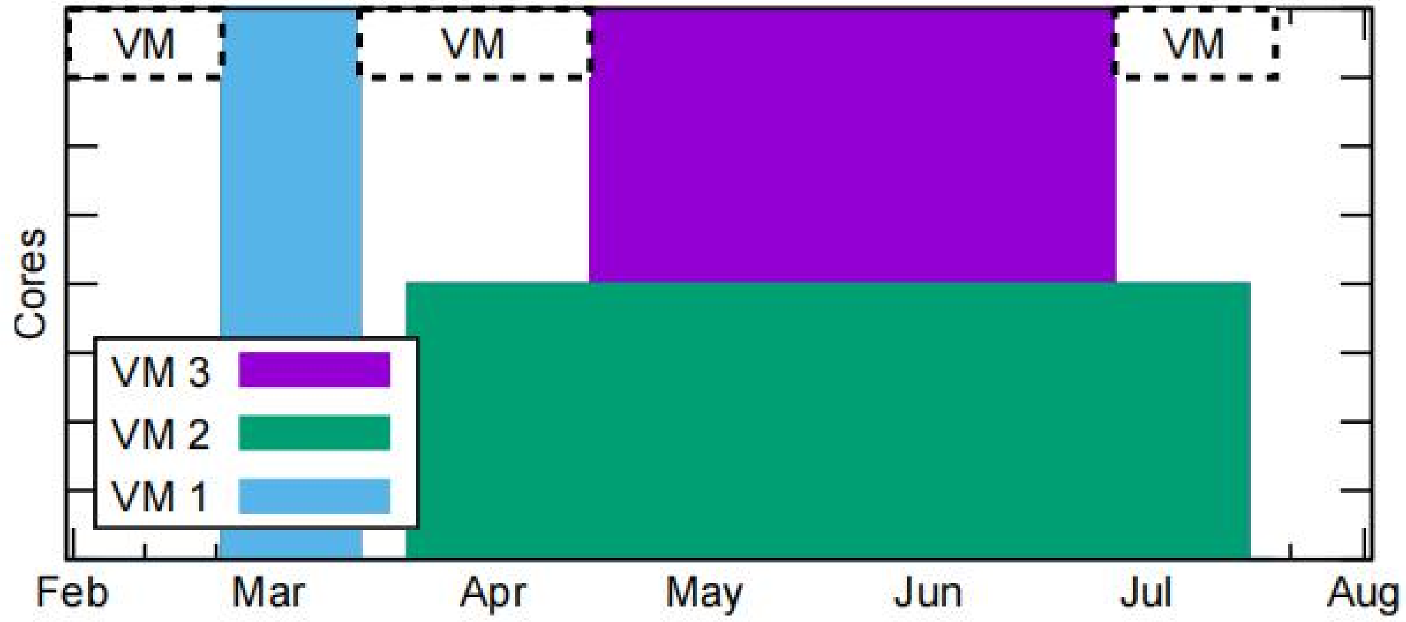
- different from Burstable VMs

  - Burstable VMs **only burst for brief time** up to their max size after accumulating credits

  - Harvest VMs grow to **consume all unallocated resources at all times**

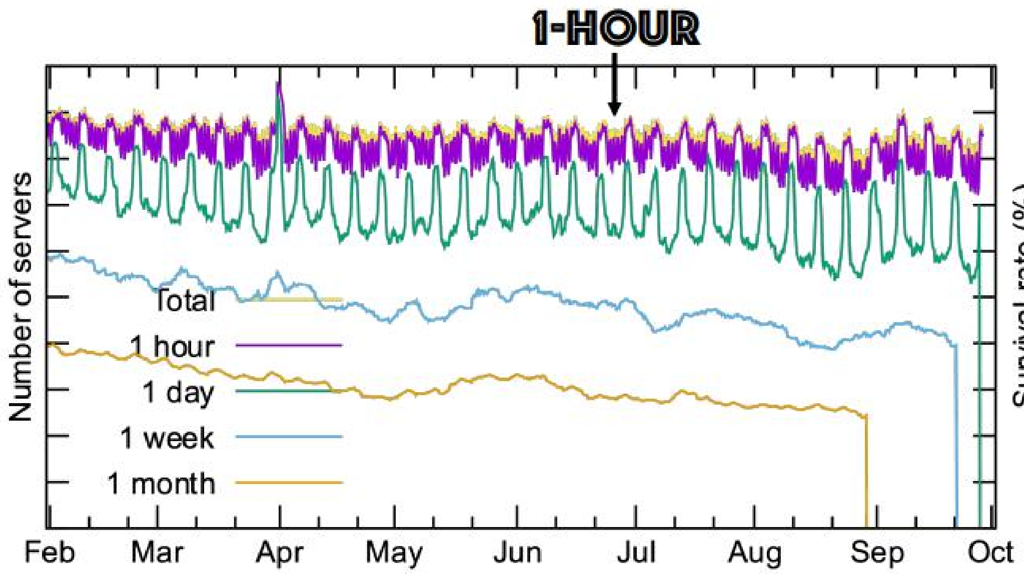
Characterizing Unallocated Resources

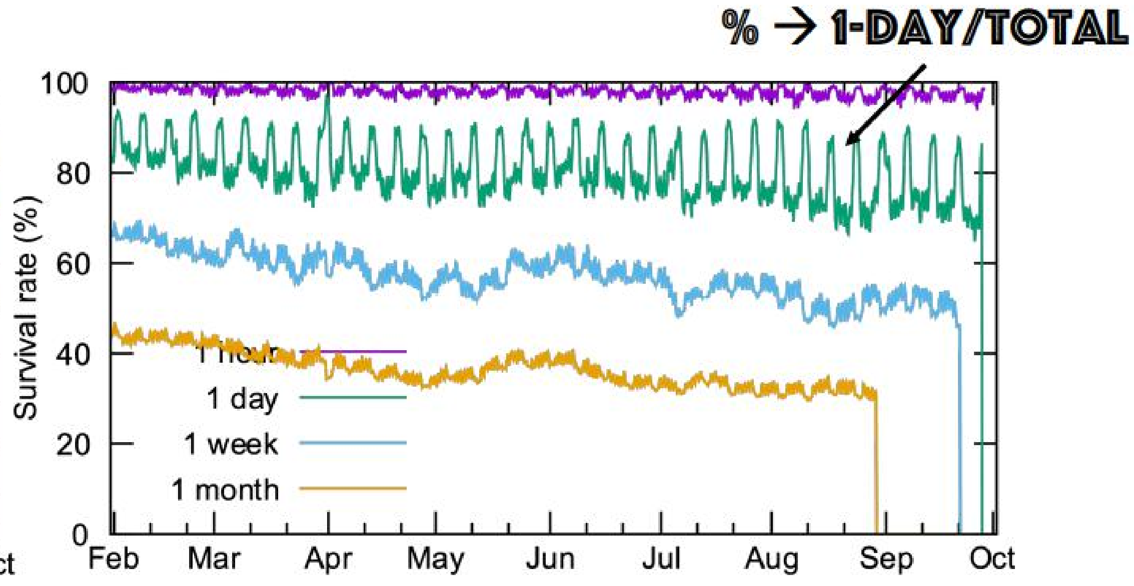
- Allocate hypothetical (假设的) VMs with idle resources
  - Characterized from **Microsoft Azure**
  - Compute unallocated resources for each host server



Characterization: Temporal patterns

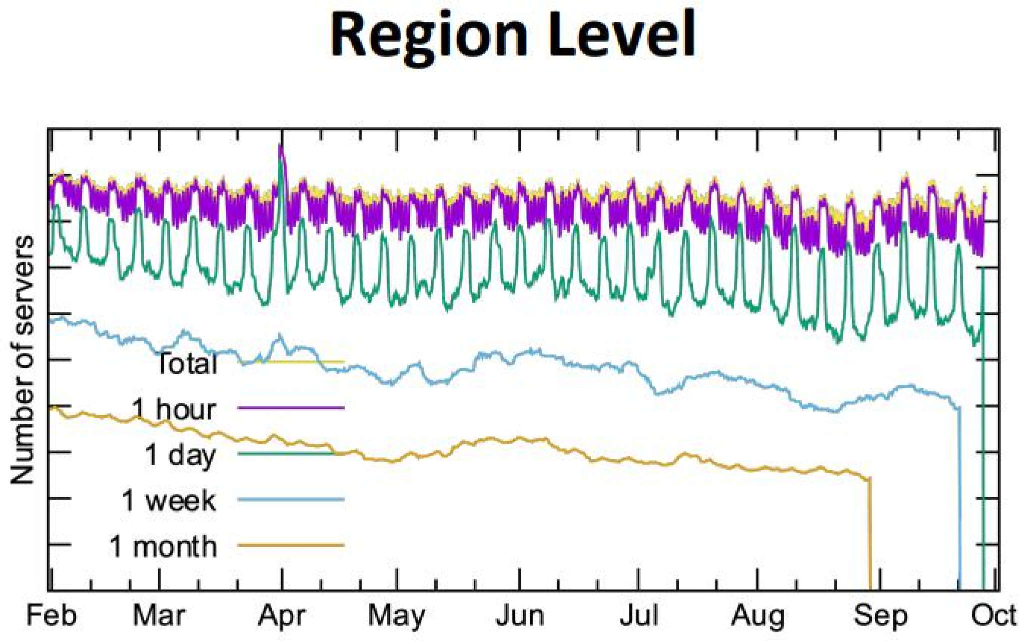
- Unallocated resources for a region
  - 1-hour shows diurnal pattern (nights have more)
  - 1-day shows weekly patterns (weekends have more)
  - Fewer servers have enough unallocated capacity over longer horizon

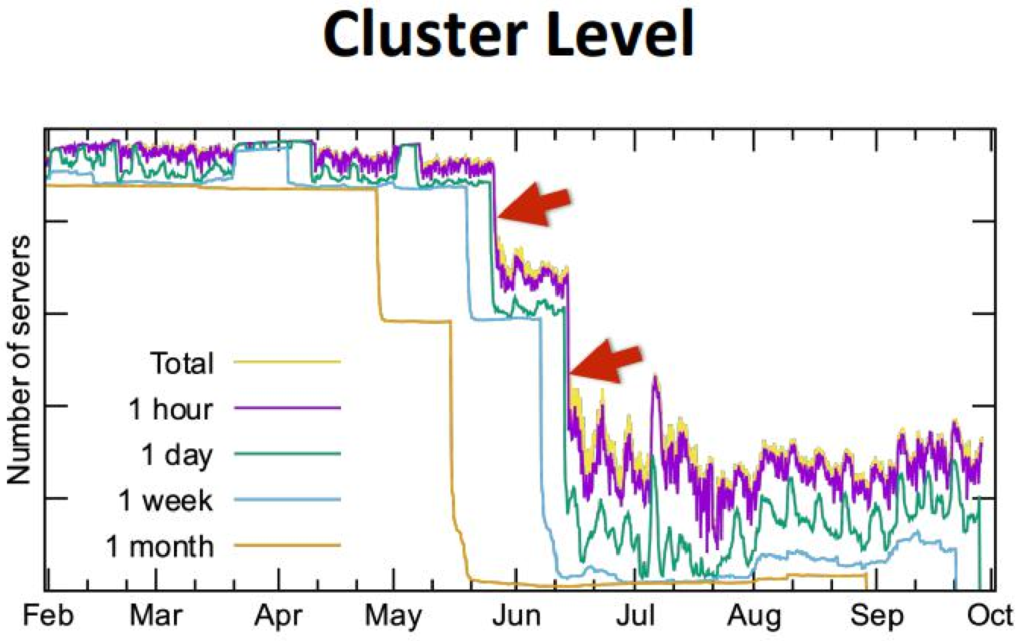




Characterization: Cluster behaviors

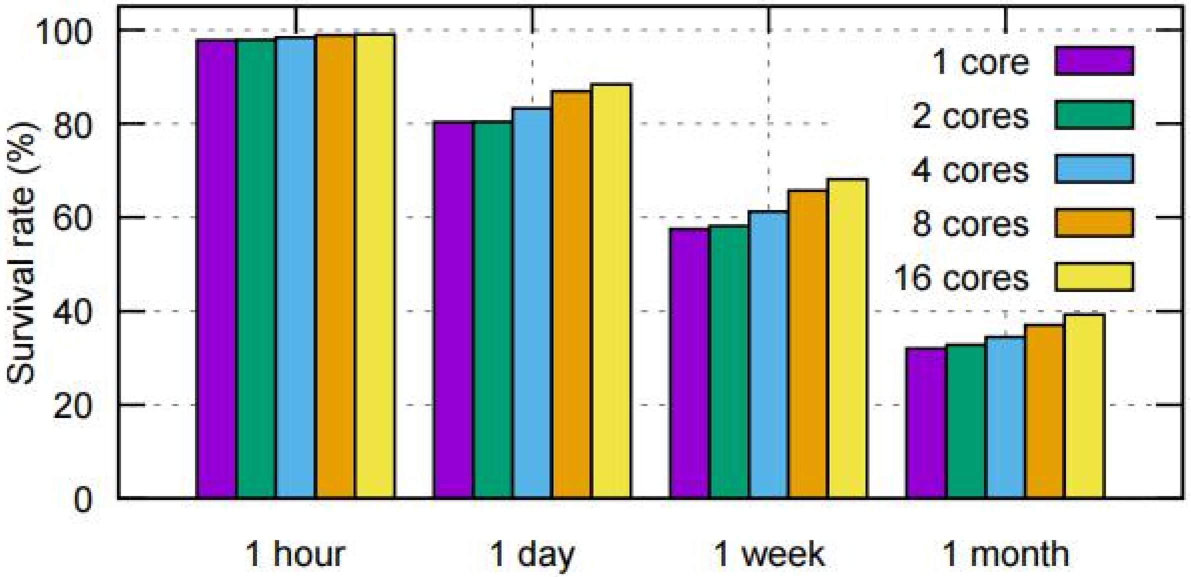
- Unallocated resources at **region level** are **stable**
- Unallocated resources at **cluster level** can **change abruptly**





Characterization: Servival Rate

- Smaller amount of resources more widely available
- Larger amounts of resources may last longer



Characterization: Key Takeaways

- Many unallocated resources are available for harvesting
  - Dynamic temporal (时间的) and spatial (空间的) behaviors
- Unallocated resources **not evenly** distributed **across clusters**
- Many **additional unallocated resources** beyond spot VMs size
- Filling with spot VMs takes many more VMs (and many more evictions)

Harvest VMs

- User picks minimun/maximum size
- **Harvest unallocated resources dynamically**

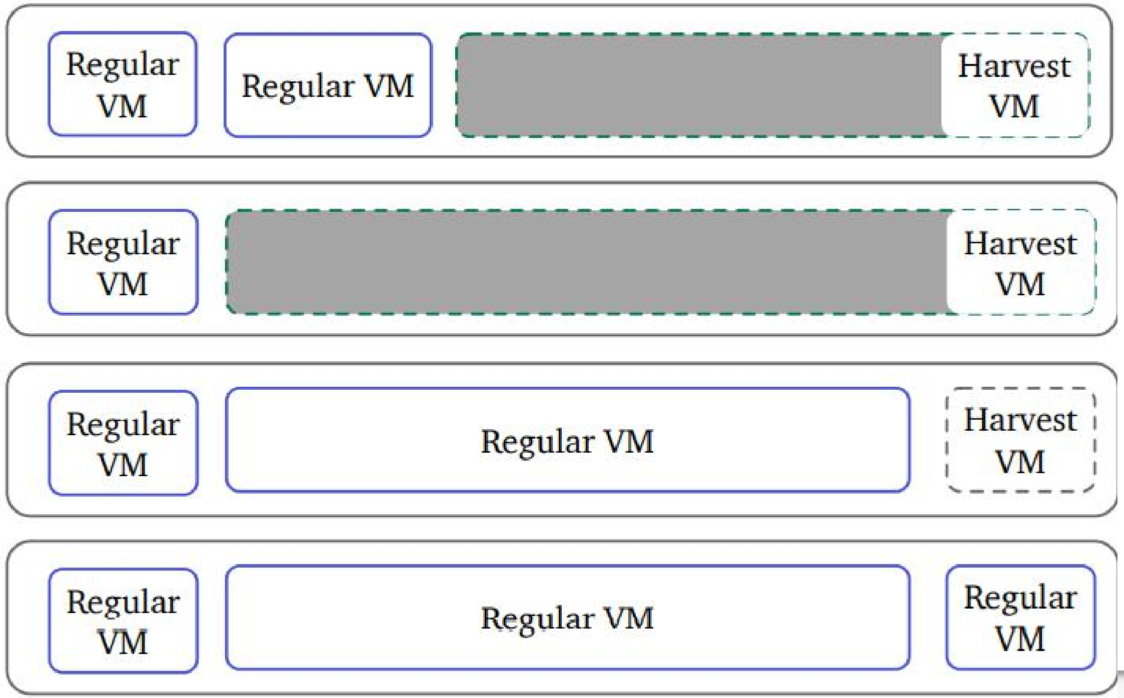
  - Dynamic physical cores & Fixed numbers of virtual cores

T_0: All unallocated first

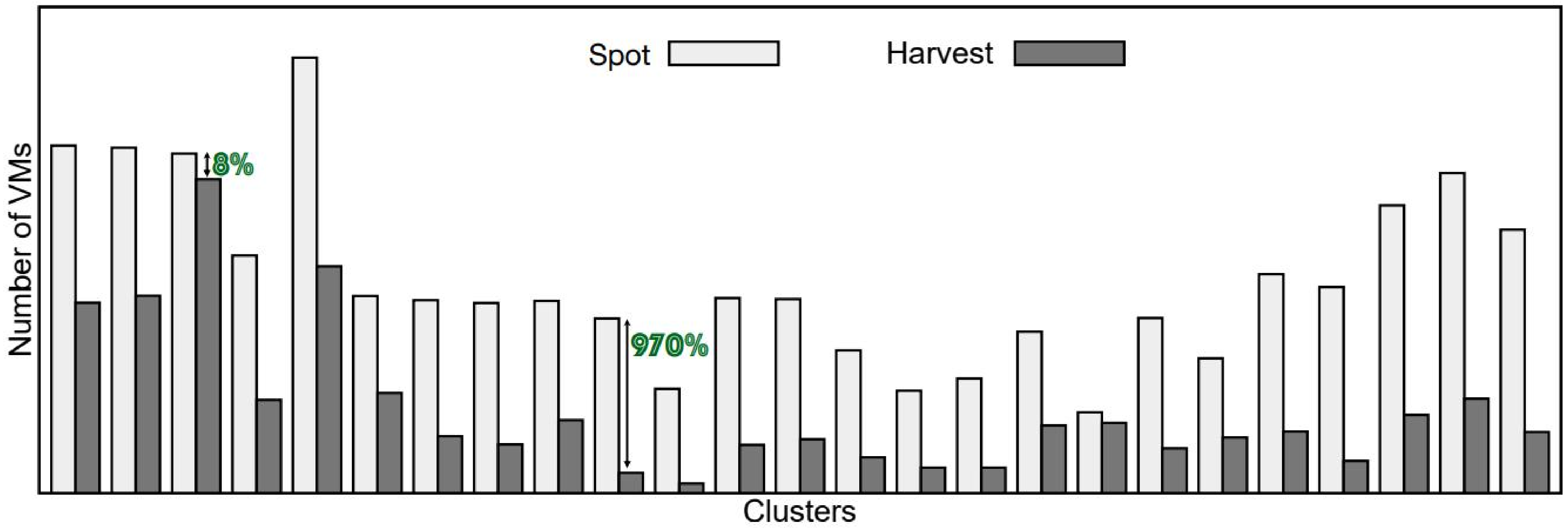
T_1: *Grow* when VM leaves

T_2: *Shrink* when new VM lands in host

T_3: *Evicted* if providers needs minimum

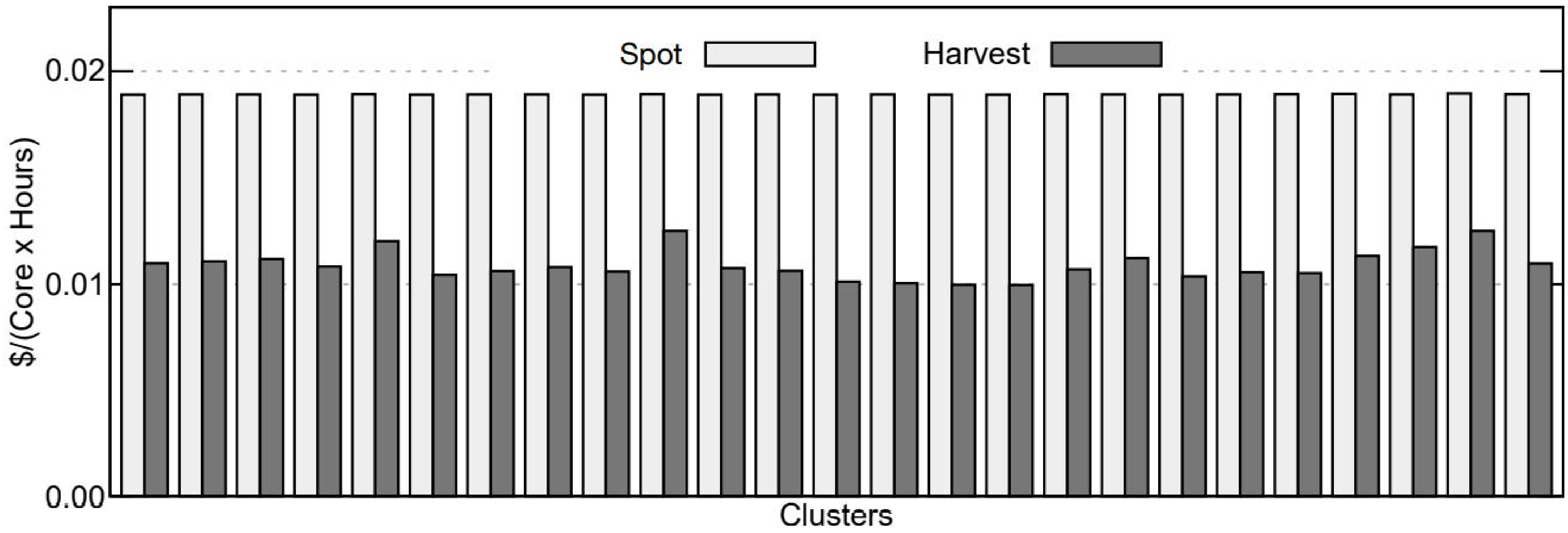


Evaluation: Spot VMs vs Harvest VMs



Requires around 3.7x more evictable VMs on average than Harvest VMs to fill unallocated capacity across all clusters

Evaluation: Cost Comparison



Harvest VMs 91% cheaper than regular VMs and 45% cheaper than spot VMs

SLOs for Harvest VMs

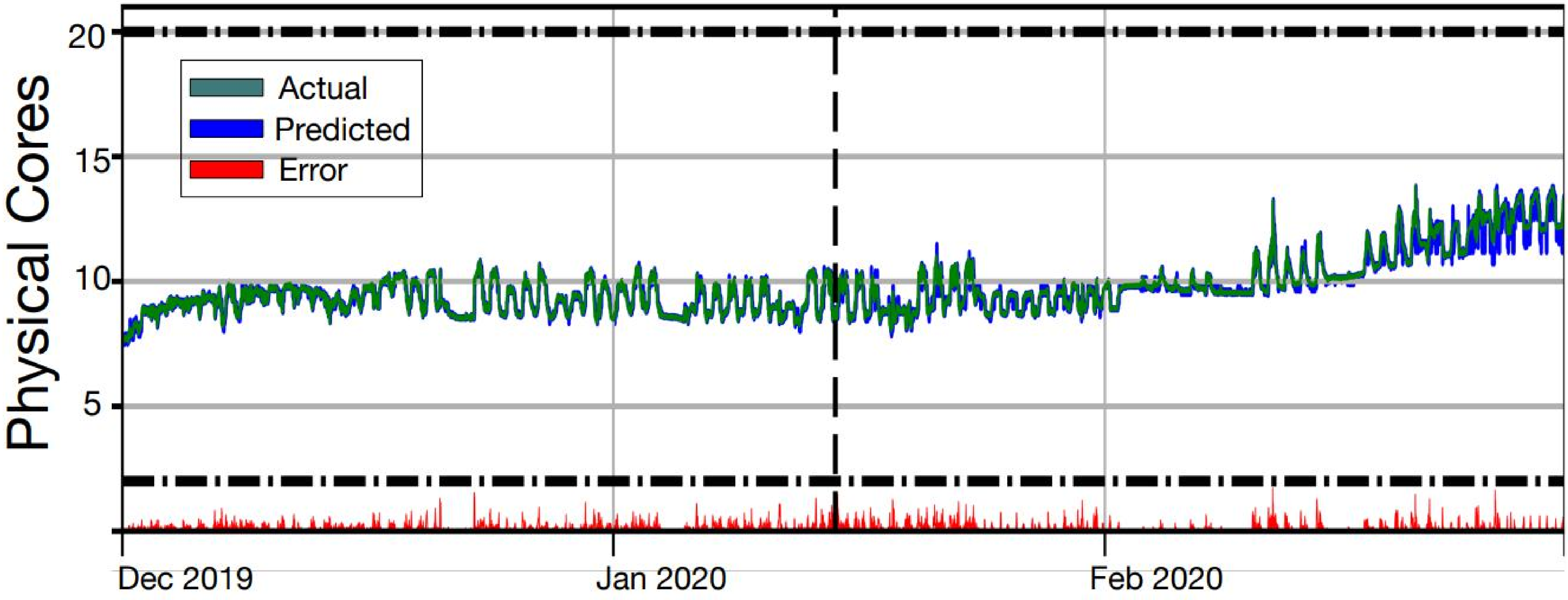
- Hard to provision just enough VMs with variable resources

  - **Key:** VMs survival rate

- Example SLO: User requests 100 Harvest VMs in Shanghai

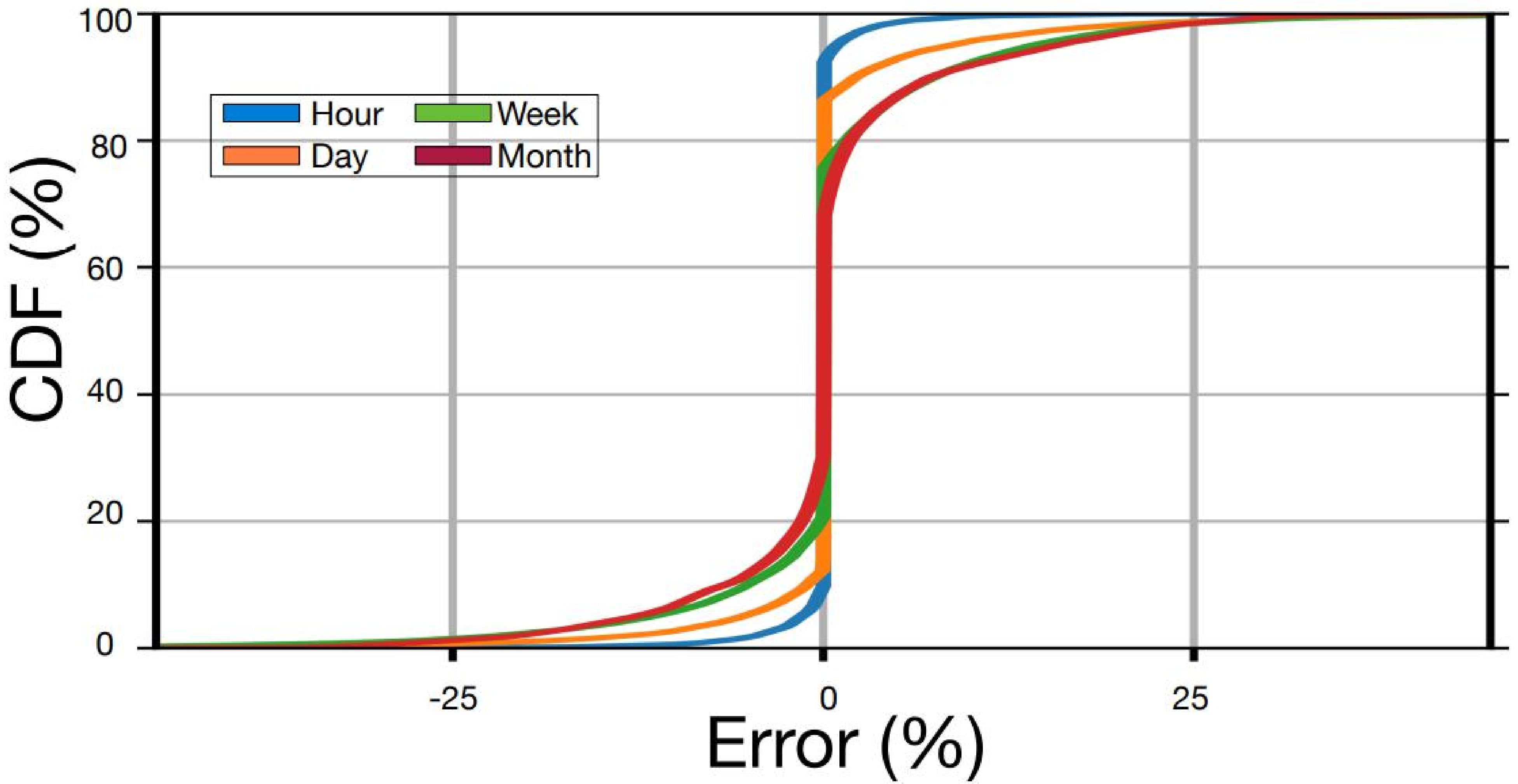
  - 85% of them survive for 1 hour
  - 30% of them survive for 1 month
  - An average of 8.5 cores in each host server (each Harvest VM)
  - 95% confidence intervals (80-90% survive for 1-hour)

Evaluation: SLOs for Harvested Cores



Prediction accuracy is very high i.e. average cores SLO would be accurate

Evaluation: SLOs for Survival Rate



Short-term predictions have an avg error < 2% and < 6% for longer terms

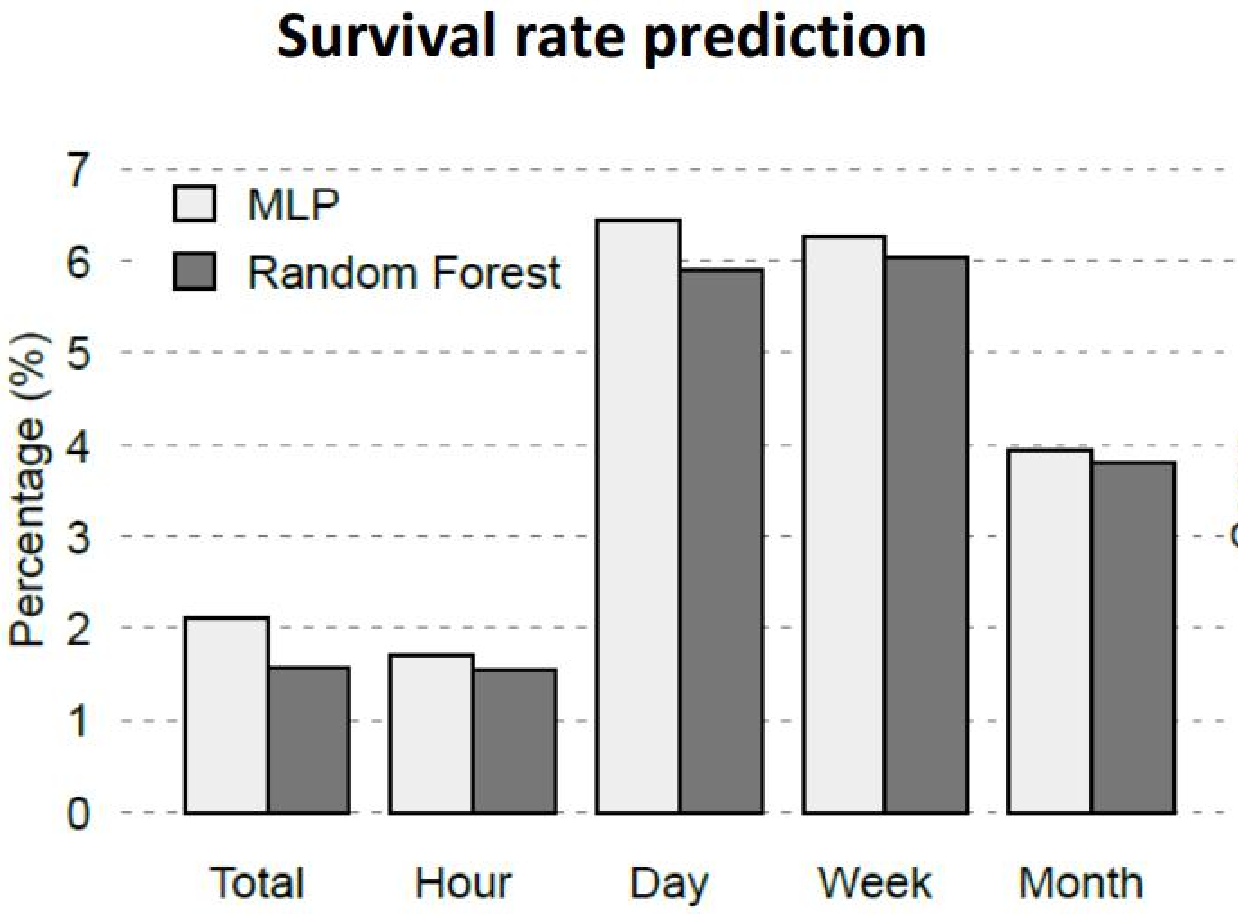
SLO Predictor Features

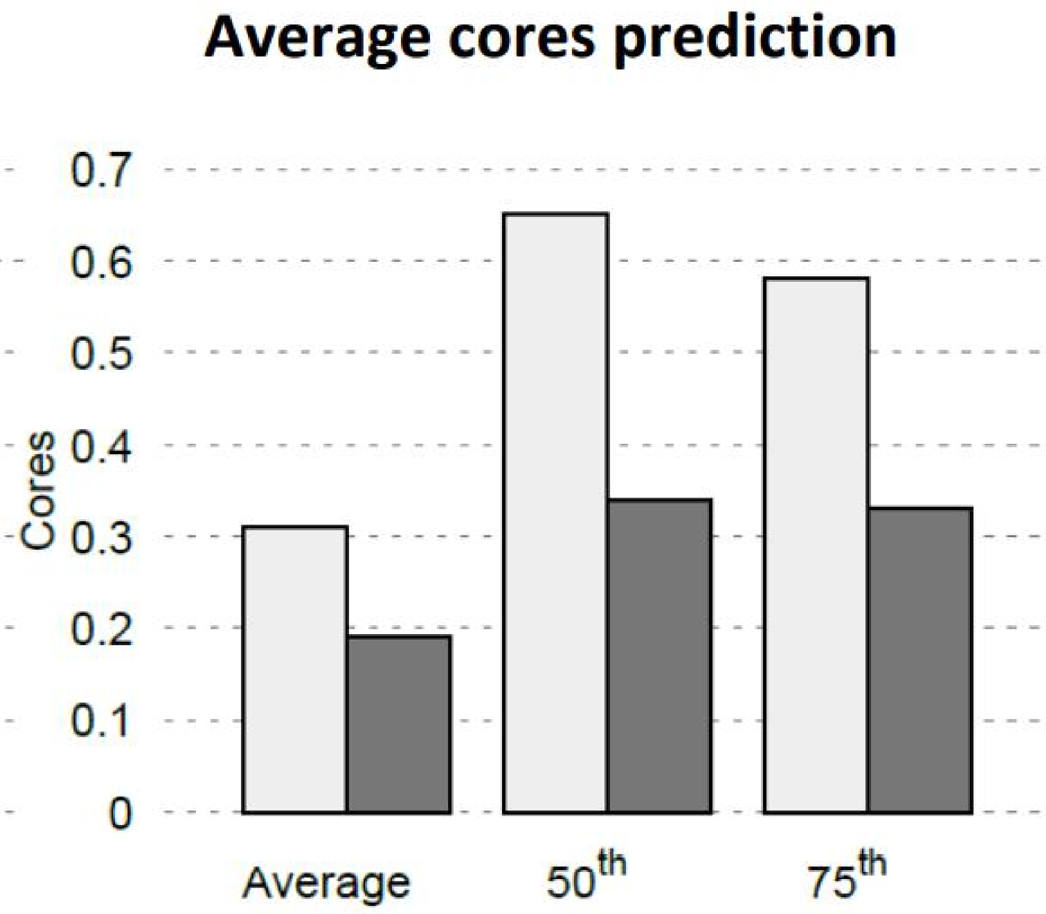
- Random Forest Regressor

- Features

  - Total VMs in the cluster
  - Total cores/memory allocated and available
  - Cluster characteristics (generation, number of racks,…)
  - Auto-regressive (e.g., values 1 day ago)
  - Moving average (average values for the last week)

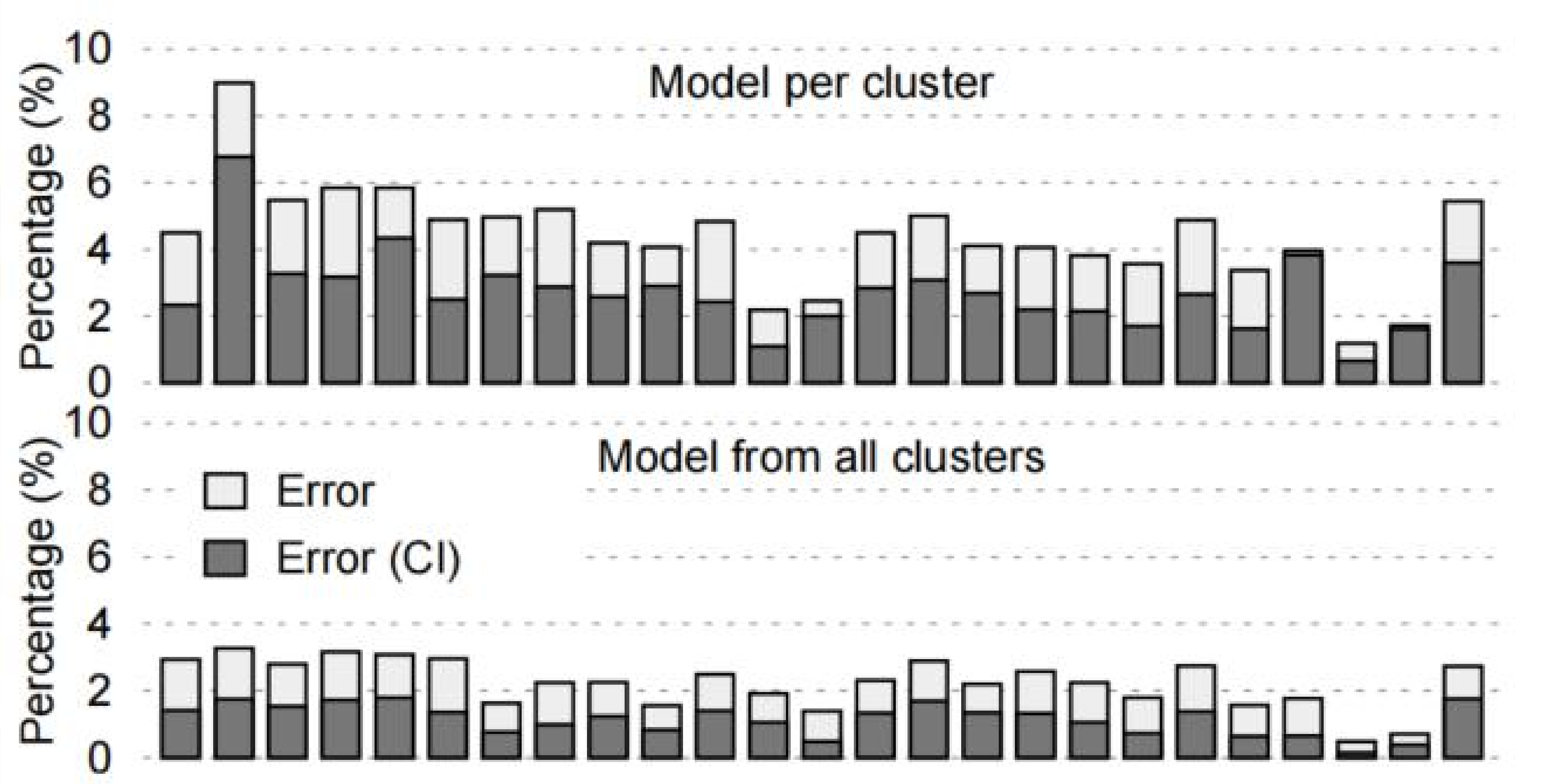
Evaluation: Random Forest vs MLP





Random Forest yields an (overall) accuracy of ~98% with mean error of ~0.2 cores

Evaluation: Model for Region or Cluster



SLO Predictor Features

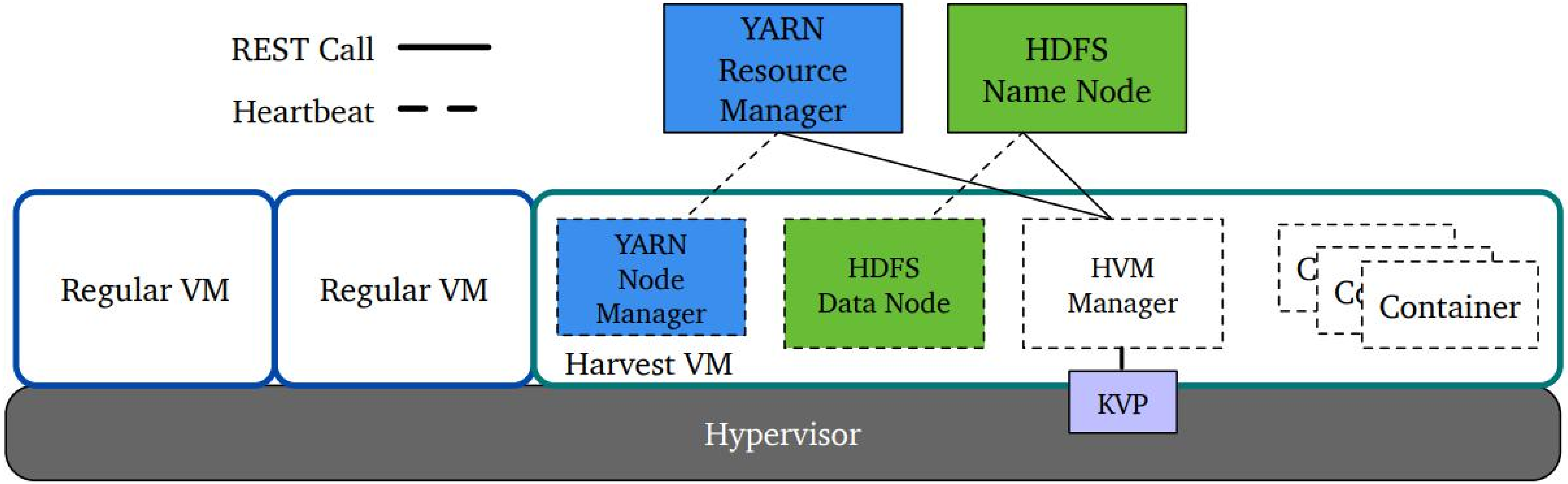
- Random Forest Regressor

- Features

  - Total VMs in the cluster
  - Total cores/memory allocated and available
  - Cluster characteristics (generation, number of racks,…)
  - Auto-regressive (e.g., values 1 day ago)
  - Moving average (average values for the last week)

Building Application on Harvest VMs

- Applications can naively use Harvest VMs
  - Leverage fault tolerance for evictions
  - Run using minimum resources available and change dynamically
- **Extend Hadoop** - run many applications (Spark, MapReduce,…)



Lessons

- Adapting applications is the main blocker
- When a Harvest VM gets 40 virtual cores, it becomes unbalanced
  - 2 cores/16GB of memory ==> 40 cores/16GB of memory
- Allowing multiple Harvest VMs per server
  - Add the maximum size of each Harvest VM
- Impact to regular VMs
  - Optimization to reduce impact in creation time
- New VM family

Conclusion

- Characterization shows many unallocated resources for harvesting
  - Dynamic temporal and spatial behaviors
- Harvest VMs successful at leveraging unallocated resources
- We provide SLOs for the availability of harvested resources
  - Our prediction models show high accuracy (~98%)
- Harvest Hadoop can adjust to changing harvested resources
- Harvest VMs and Harvest Hadoop running in production in Azure
  - 91% cheaper than regular VMs
  - 45% cheaper than spot VMs and with 73% fewer evictions

上海交通大學

SHANGHAI JIAO TONG UNIVERSITY

Thanks

饮 水 思 源　爱 国 荣 校