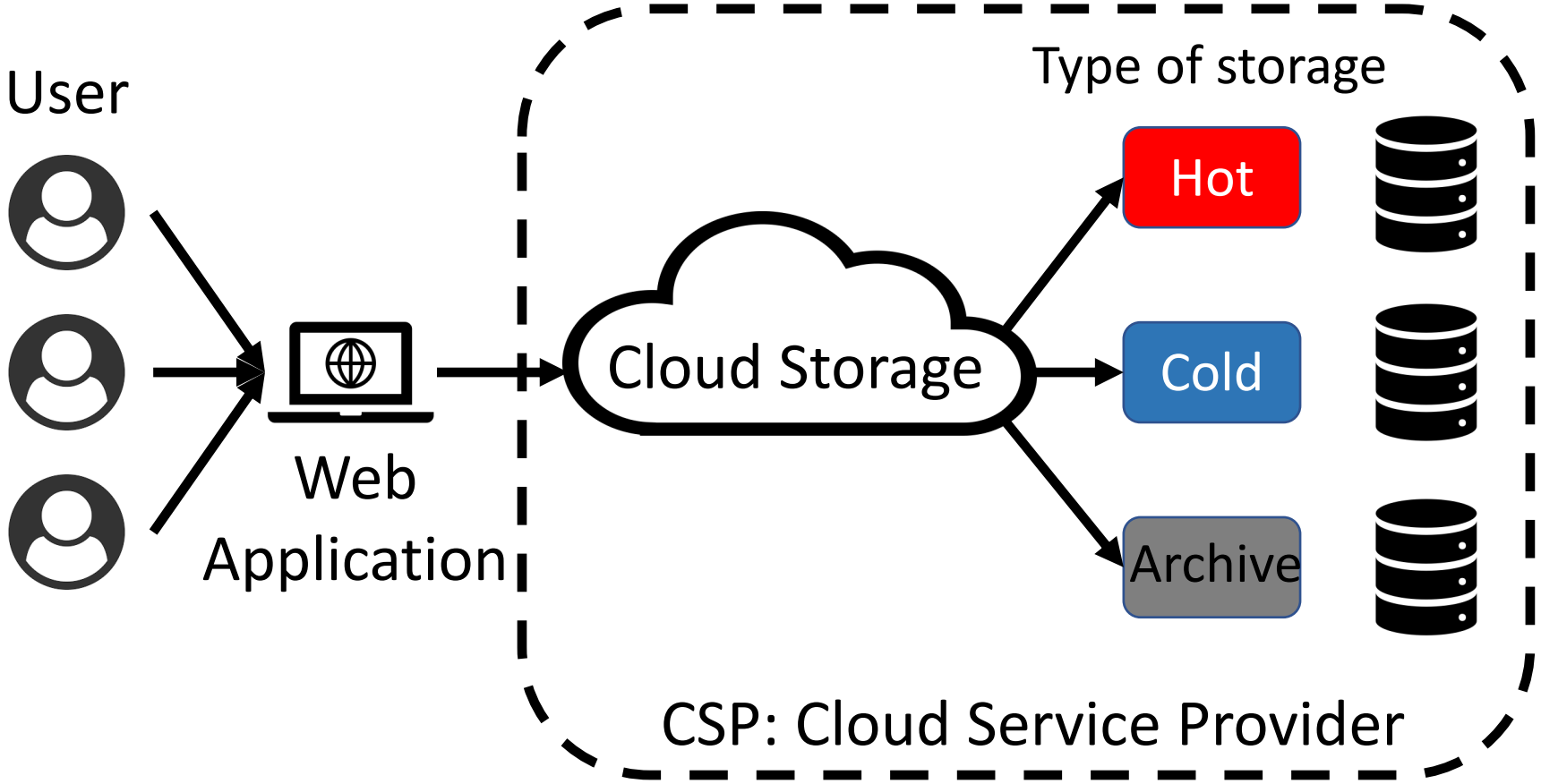
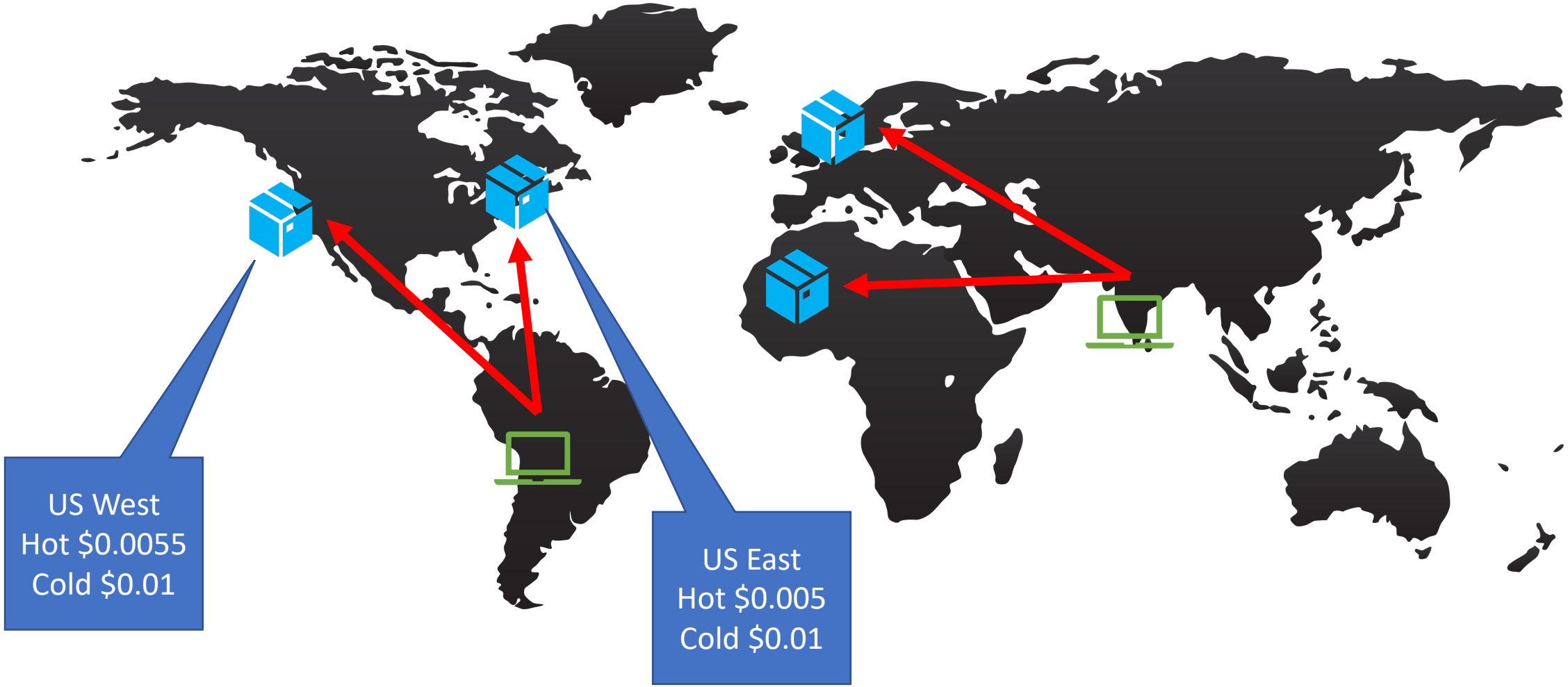
A Reinforcement Learning Based System for Minimizing Cloud Storage Service Cost

Haoyu Wang[1], Haiying Shen[1], Qi Liu[1], Kevin Zheng[1], and Jie Xu[2]
[1]University of Virginia and [2]George Mason University

ICPP2020 Online presentation

Web application:


User
Web
Application
Cloud Storage
Type of storage
Hot
Cold
Archive
CSP: Cloud Service Provider


US West
Hot $0.0055
Cold $0.01
US East
Hot $0.005
Cold $0.01

Outline

- How to minimize storage monetary cost
- Related work
- Wikipedia trace analysis
- Markov decision process problem formulation
- Main design
- Performance evaluation
- Conclusion

Minimize storage monetary cost

Different price is determined by:

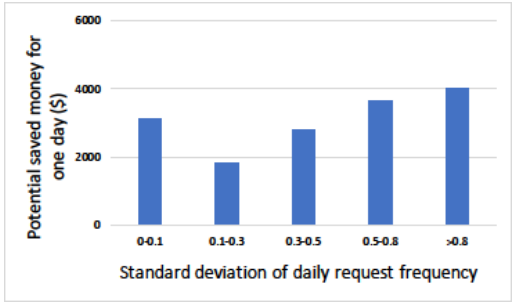
- Storage type
- Read/write operation frequencies
- Storage period

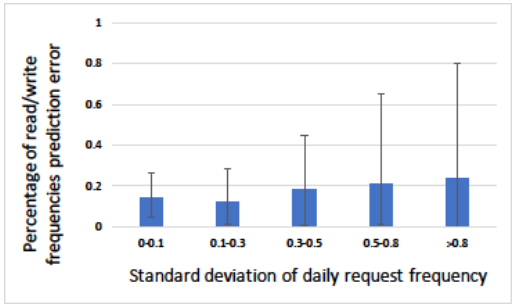
Related work

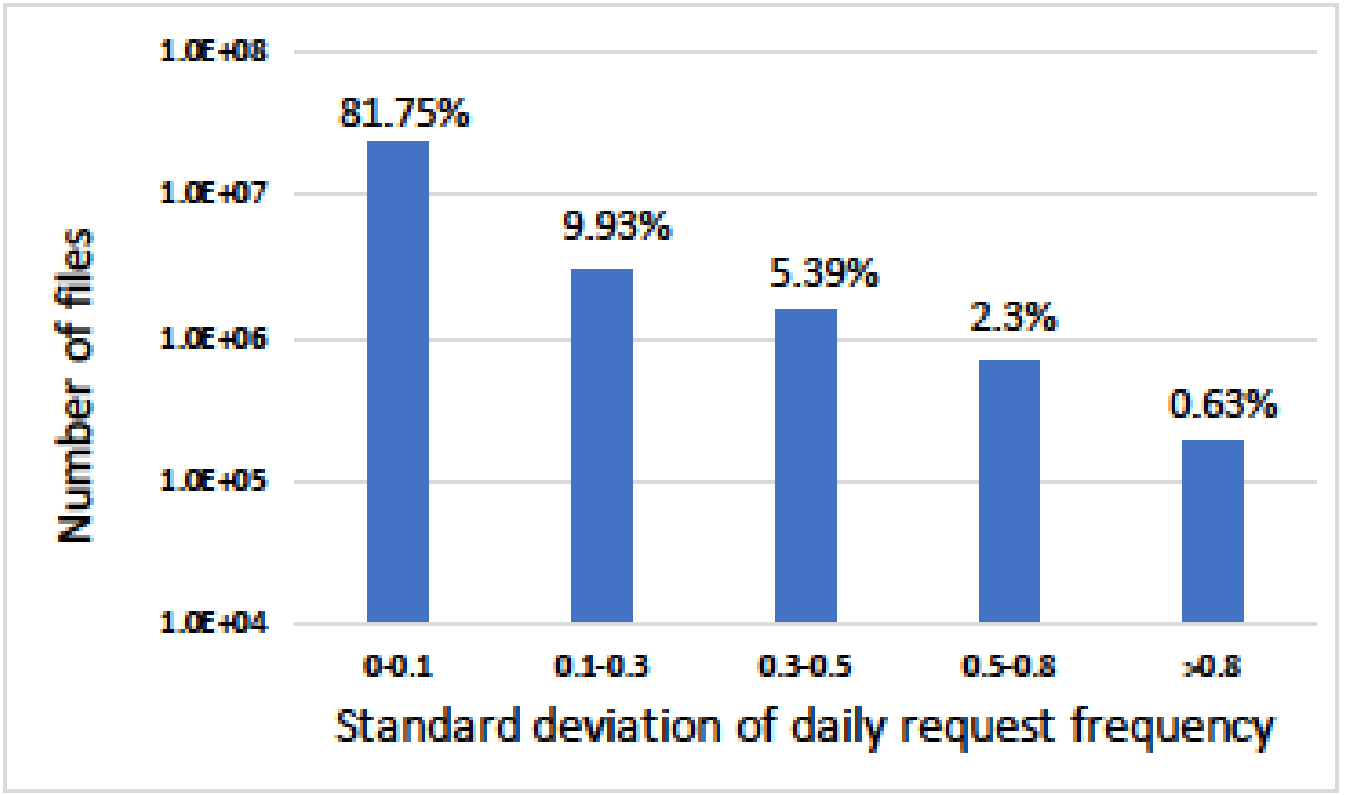
- Cloud storage payment minimization
- Cloud resource pricing
- Combining cloud providers

Unlike the above methods, the goal of our method is to minimize the total payment a cloud storage service customer made to a CSP by leveraging the different types of storage provided by the CSP.

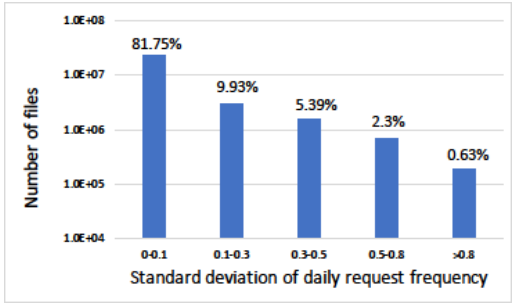
Trace analysis

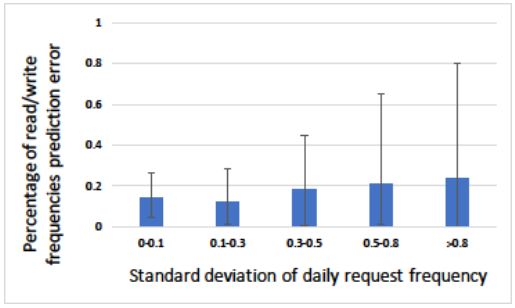


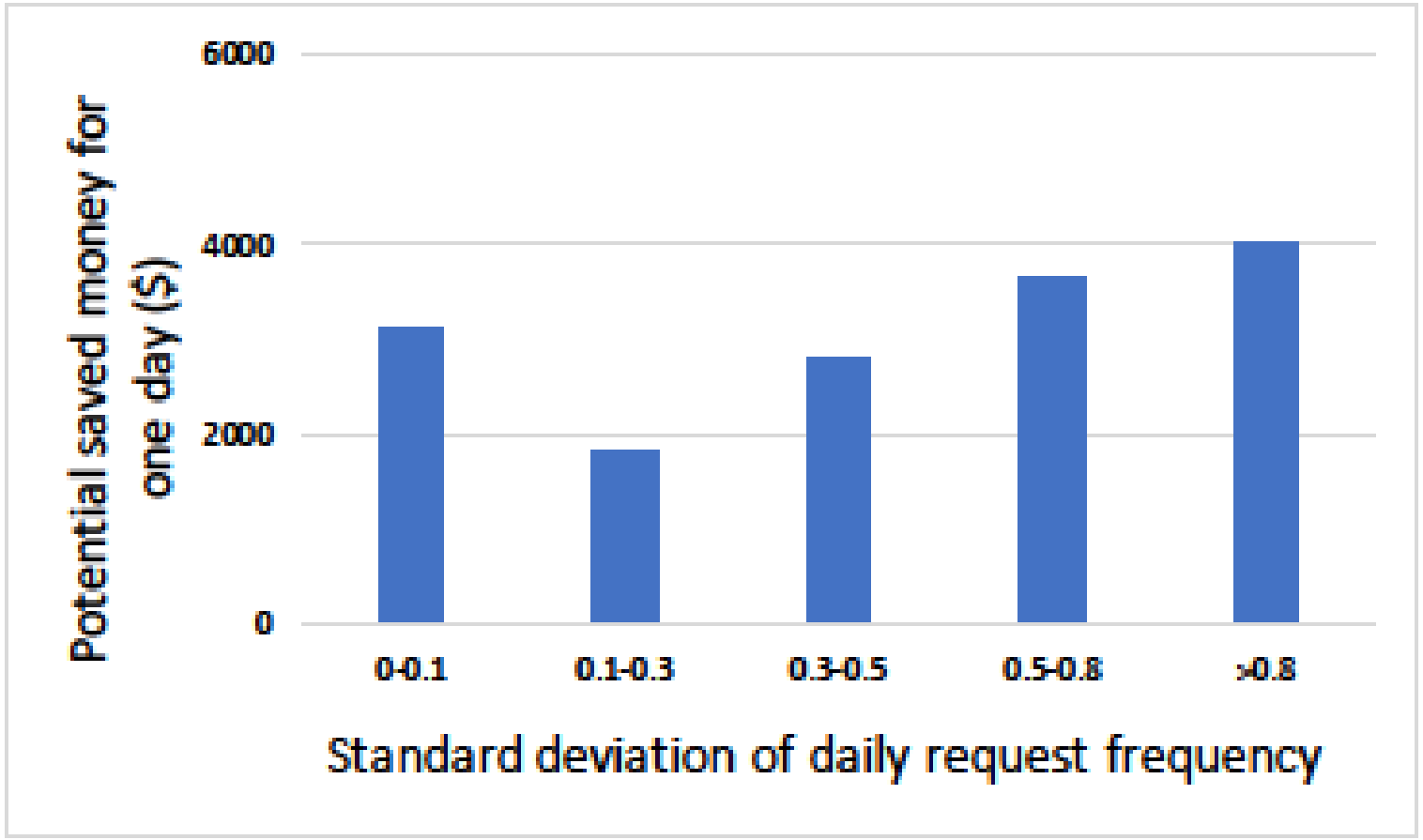




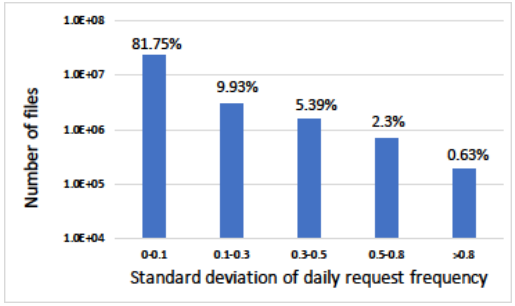
Trace analysis

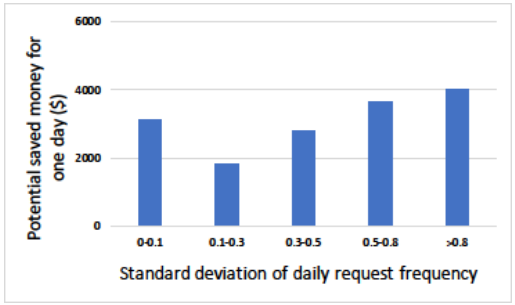


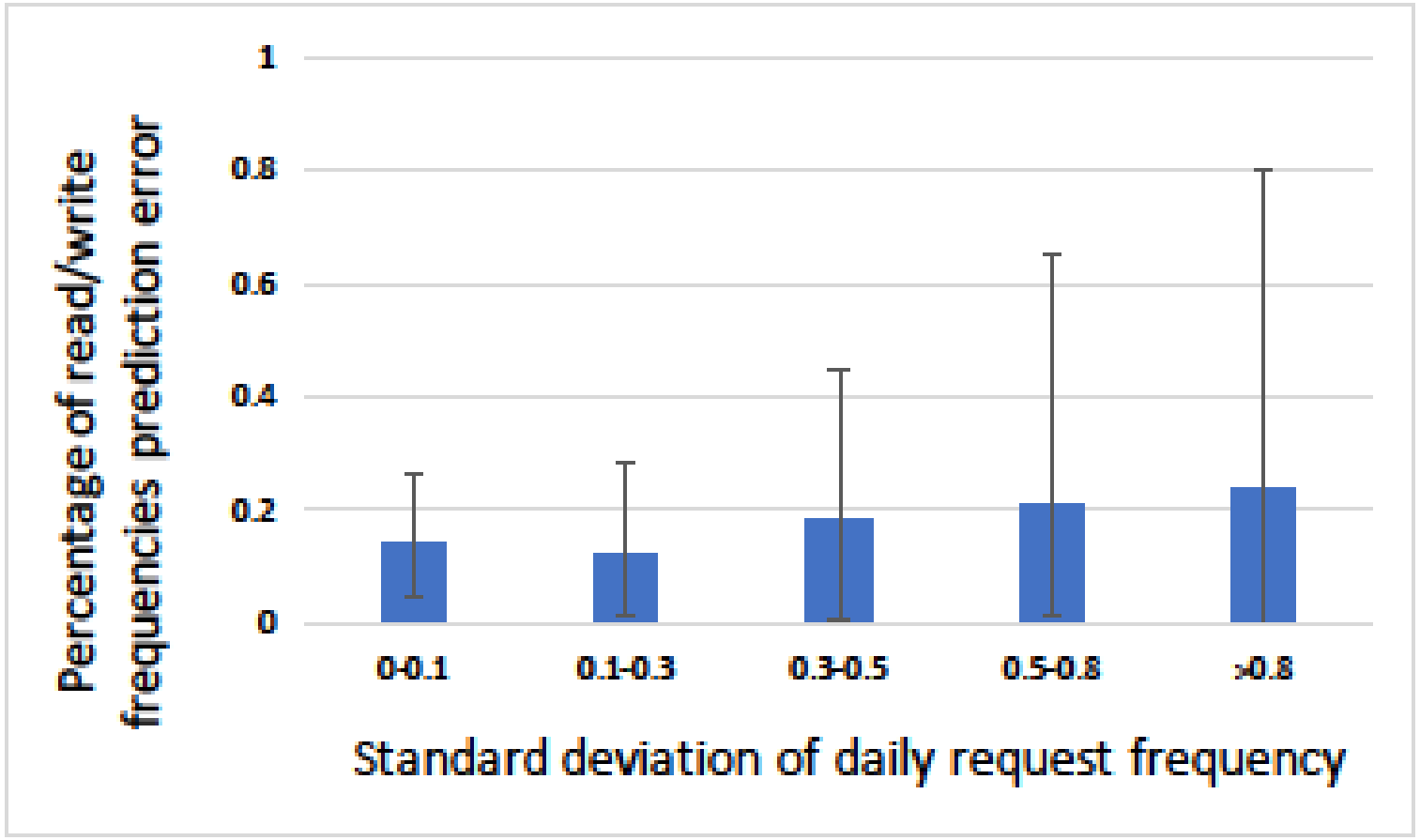




Trace analysis







Problem formulation

Markov Decision Process

M=(S,A,P,R)

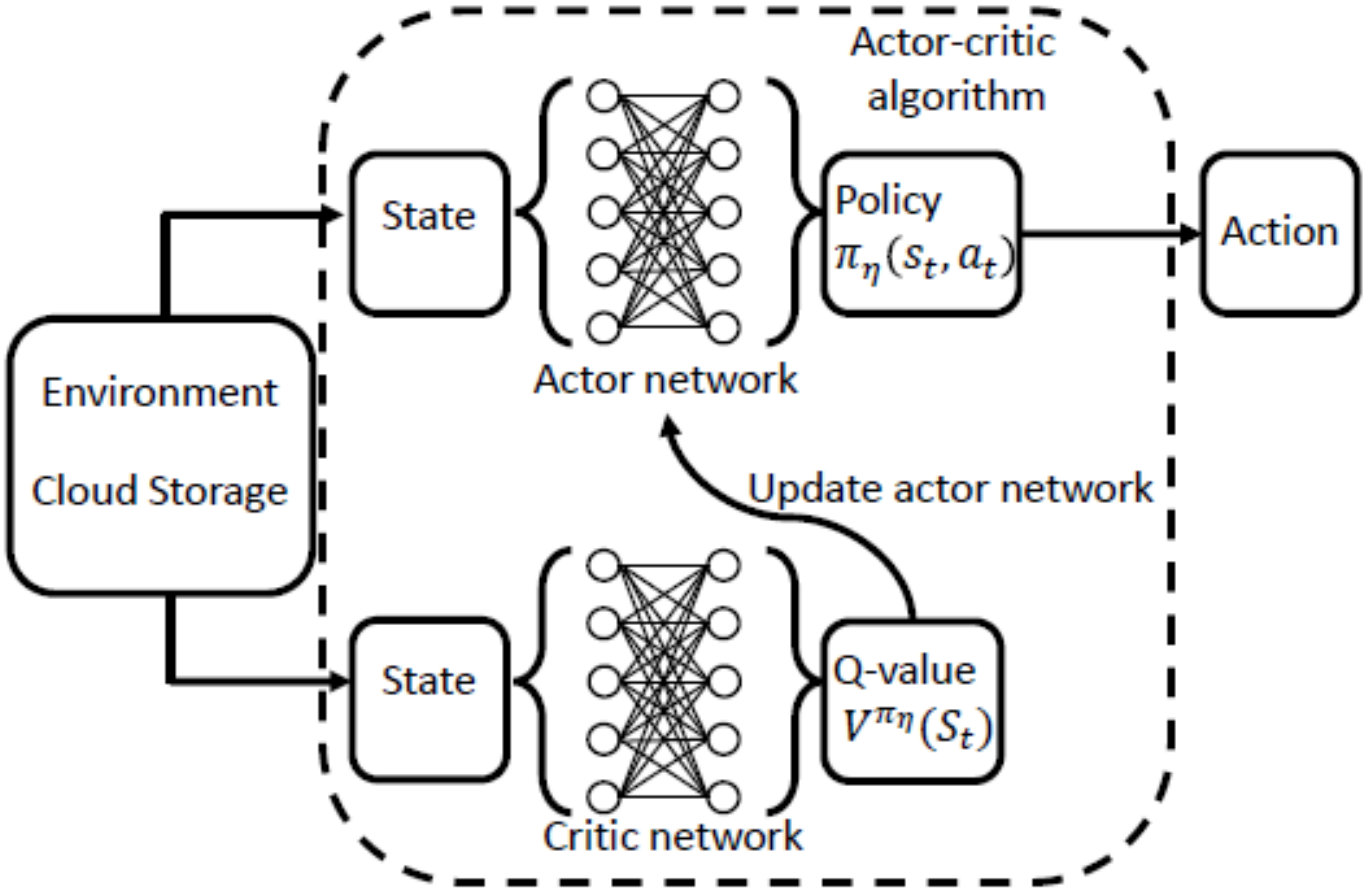
State space: $$S = \{s = (F_r, F_w, D, \Gamma)\}$$

Action space: $$A = \{a = (a_0 ... a_N) | a_i \in \{1, ..., \Gamma\}, i = 1, \ldots, N\}$$

Reward: $$R(s_t, a_t) = \frac{\alpha}{C(s_t, a_t)} + \Delta$$

Main design

1. A3C algorithm used in MiniCost



Main design

2. Concurrent requested data files aggregation

Algorithm 2: Pseducode of the concurrent requested data files aggregation algorithm.

```
for Collect the concurrent requests information of all the data files;
do
    for each group of data files;
    do
        Calculate the data file aggregation coefficient Ω according to Equation 16);
    Sort the group of data files in descending order according to Ω;
    Select top Ψ groups of data files to generate the aggregated data files;
    if Ω of one group of data files is smaller than 0 then
        Delete the aggregated data file related to this group
    end for
end for
```

Performance evaluation

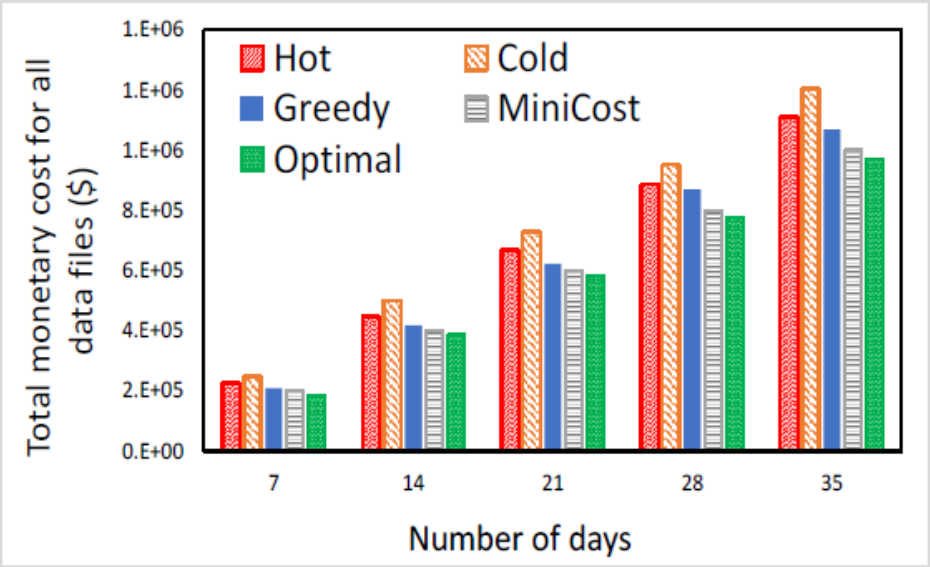


Figure 7: Comparison of total costs.

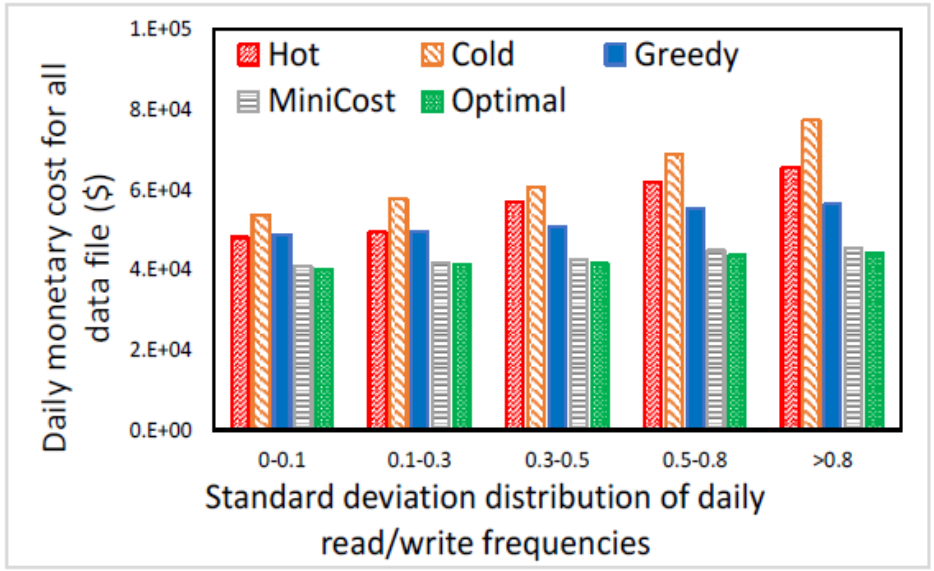


Figure 8: Cost per data file by standard deviations of daily request frequencies.

Performance evaluation

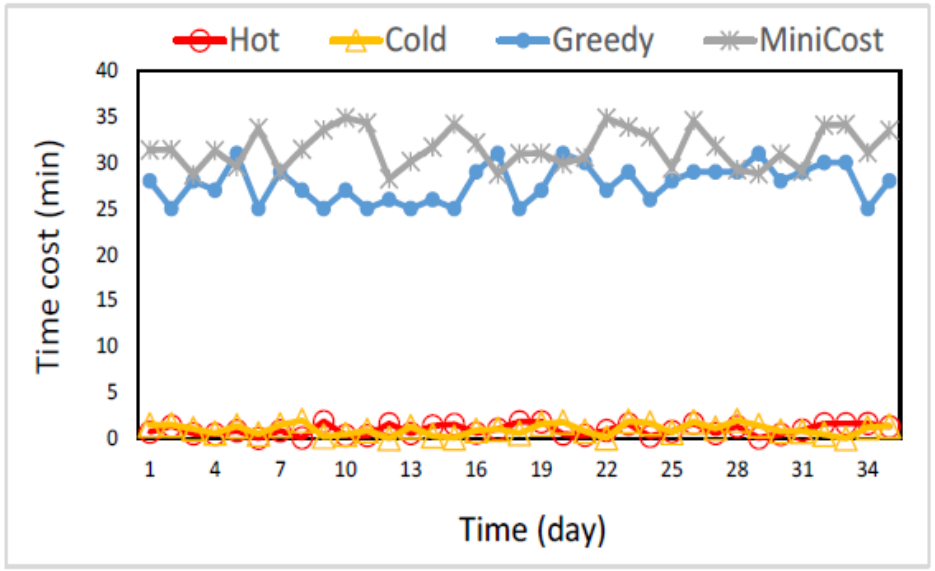


Figure 12: Overhead.

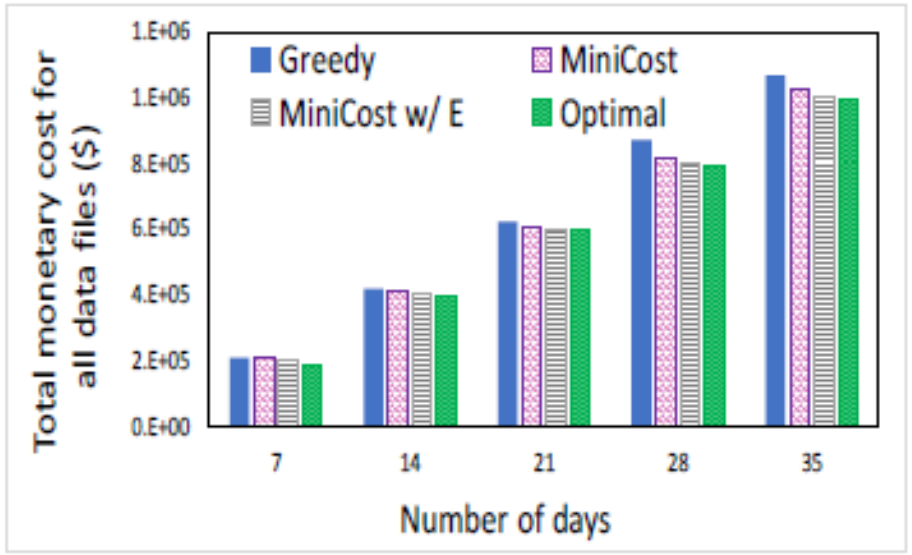


Figure 13: The performance with and without data file aggregation.

Conclusion

- Analysis on the Wikipedia trace demonstrates that the substantial request frequency variabilities may make it cost-inefficient for cloud storage service customer.
- An RL based data storage types assignment algorithm that generates data storage types assignment plans periodically can save monetary cost significantly.
- Trace-driven experiment shows that our online RL based method can achieve significant cost savings.

Thank you!

Questions?