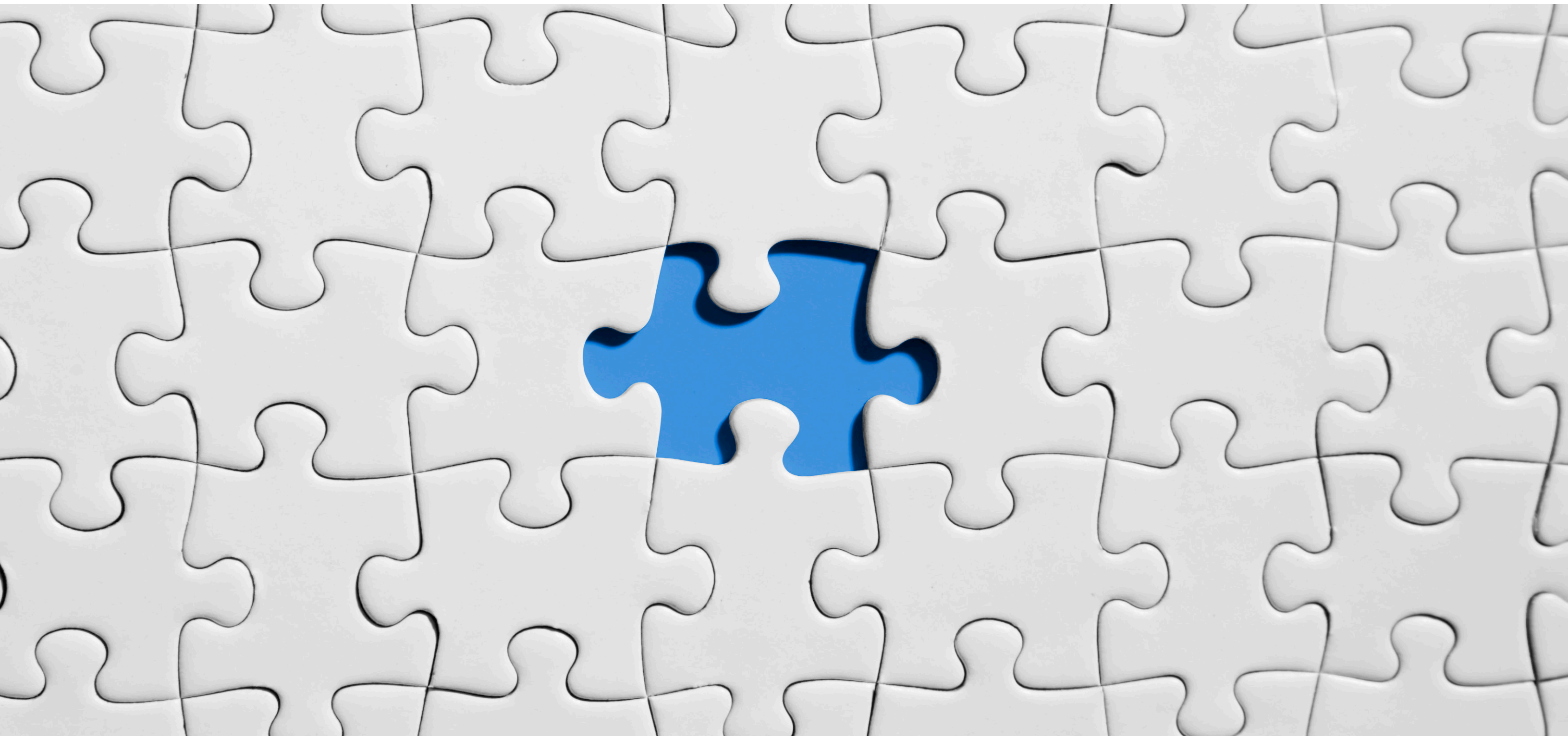
RESOURCE ALLOCATION USING REINFORCEMENT LEARNING



Chethan Nagaraj
Chethan187@gmail.com

Manu Joshi
Manu.Joshi@dell.com

Knowledge Sharing Article

Proven Professional

The Dell Technologies Proven Professional Certification program validates a wide range of skills and competencies across multiple technologies and products.

From Associate, entry-level courses to Expert-level, experience-based exams, all professionals in or looking to begin a career in IT benefit from industry-leading training and certification paths from one of the world's most trusted technology partners.

Proven Professional certifications include:

- Cloud
- Converged/Hyperconverged Infrastructure
- Data Protection
- Data Science
- Networking
- Security
- Servers
- Storage
- Enterprise Architect

Courses are offered to meet different learning styles and schedules, including self-paced On Demand, remote-based Virtual Instructor-Led and in-person Classrooms.

Whether you are an experienced IT professional or just getting started, Dell Technologies Proven Professional certifications are designed to clearly signal proficiency to colleagues and employers.

Learn more at www.dell.com/certification

Table of Contents

Abstract 4
Impediments to Migrating to Public Cloud 4
What is Dynamic Resource Allocation? 4
HPC workloads with complex communication paths on cloud platform 5
What a bare-metal cloud platform for latency-sensitive HPC applications brings to the customer 6
Trade-off Between Private and Public Cloud 6
Hybrid Cloud Storage Capacity Optimization 7
Hybrid Cloud Compute Capacity Optimization 8
How are we managing the above solutions through Reinforcement Learning? 8
Problem Solving Approach 9
Temporal Difference Learning 10
Hybrid Cloud Management Architecture based on Reinforcement Learning 10
Reinforcement Learning: Examples 11
Reinforcement Learning vs. the other ML models 12
Conclusion 12
References 12

Abstract

Cloud Computing has been a trending technology for a few years supporting computational services over internet. But ever since its adoption, cloud's consistent challenge is in its dynamic resource allocation. The existing cloud model details the online and offline algorithms used to decide the dynamic resource allocation. The goal is to have a dynamic resource allocation framework that aligns to cloud data management's objective of maximizing revenue with minimum cost. This encourages both consumers and cloud providers not only with energy-efficient power usage but also high CPU utilization.

This article discusses the impediments of migrating to Public Cloud, what is dynamic resource allocation, HPC workloads with complex communication path on cloud platform, and the benefits of bare metal platform for latency-sensitive applications. We shed light on trade-offs (compute balance) between Private and Public Cloud, how existing resources can be leveraged, reinforcement learning (RL) solutions including a study on hybrid cloud computing capacity optimization framework. Understanding RL architecture, problem solving approach, learning structure and Hybrid Cloud Management Architecture framework are also explored. Also given are a few RL implemented gaming examples on how it makes an impact. Lastly, we shall do the comparisons of RL with other Machine Learning (ML) approaches.

Impediments to Migrating to Public Cloud

Here is the list of barriers to Pubic Cloud investment by organizations.

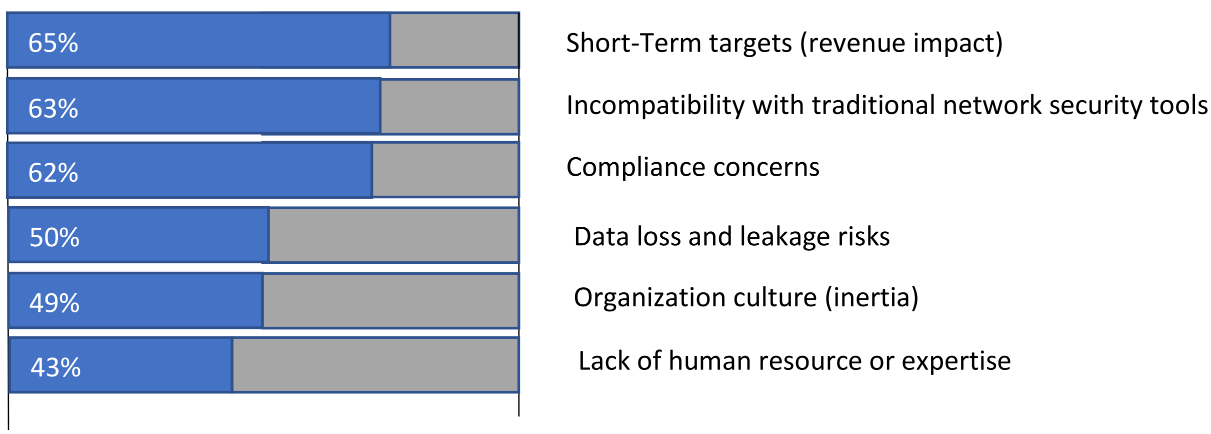


Figure 1: Barriers to investing in cloud

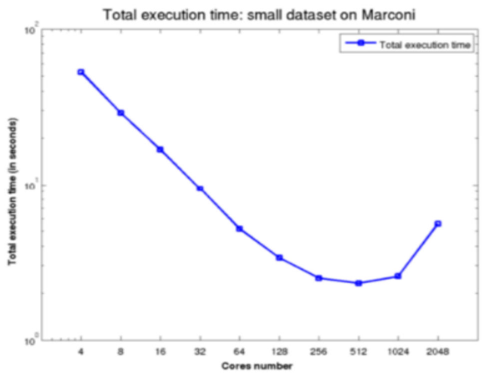
What is Dynamic Resource Allocation?

Dynamic Resource Allocation (DRA) is used in cloud computing environments. It is considered as an important optimization technique to achieve maximum resource efficiency and scalability, as well as load balancing, the effective distribution of loads among back-end virtual machines.

DRA enables the business to be more flexible and agile in resource management, particularly in provisioning resources by allowing users to scale-up and scale-down the allocation as per their needs through auto-balance methodologies. Due to the dynamic nature of cloud environments adoption of on-demand resource allocation technique is very much needed and is beneficial in terms of cost for end users and offers an efficient resource handling for cloud providers.

HPC workloads with complex communication paths on cloud platform

This example considers the **BloodFlow** tool used to monitor the blood flow in various blood vessels and to measure cardiac output.



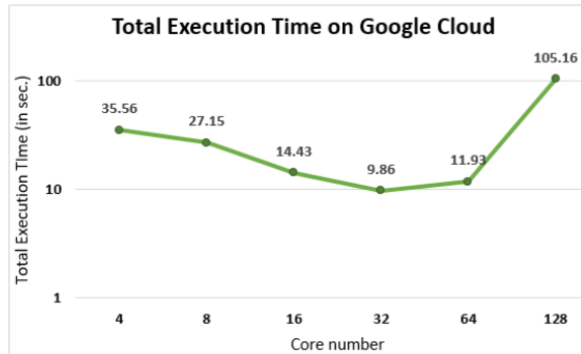


Figure 2(a): Total Execution on Marconi

Figure 2(b): Total Execution on Google Cloud

The study begins by analyzing the application stability on Marconi from Bloodflow tool data on HPC environments. HPC continues to scale up to 512 nodes whereas cloud scales to 32 nodes.

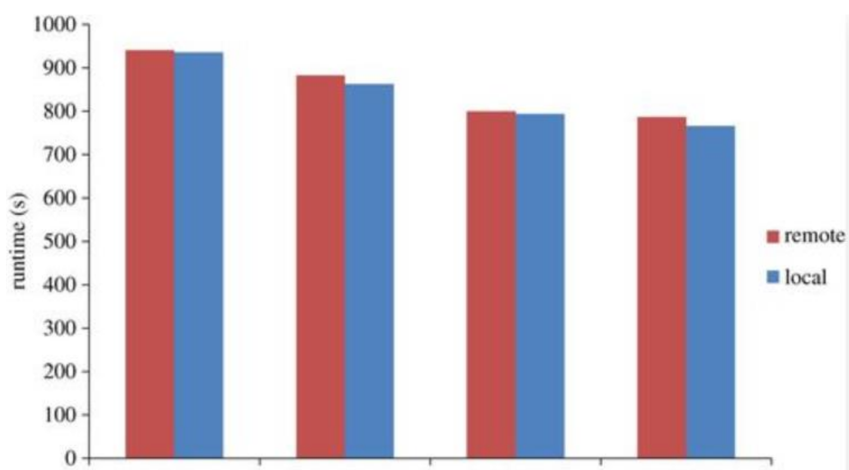


Figure (3): MPI processes on remote vs local hosts

Figure 3 shows the runtimes of a pair of MPI processes on a pair of remote, single processor nodes indicating minor overhead of network communication on application performance. This impact will be lower on the bare metal hosts supporting virtualization. Other overheads such as data transfer rates to remote cloud platforms can be an issue, hence it is recommended that for highly data-intensive tasks, local platforms are the best solution. The above representation depicts only the runtime; it does not include the provisioning part. This approach helps in comparing the computation time on cloud vs. non-cloud environments.

As our focus is on platforms supporting HPC applications, a significant amount of computation involved in provisioning part has been excluded from this discussion.

What a bare-metal cloud platform for latency-sensitive HPC applications brings to the customer

1. Customers can reduce the time to run the workloads from days to hours to minutes

2. Dedicated hardware as a service offers flexibility, scalability and efficiency without the drawbacks of a shared server

3. A bare-metal cloud model also enables on-demand usage and metered hourly billing with physical hardware which was sold earlier on a fully dedicated basis

4. Bare-metal cloud is the best fit for bursty I/O-heavy workloads. Ideal use cases include media encoding and render farms which are both periodic and data-intensive in nature.

Trade-off Between Private and Public Cloud

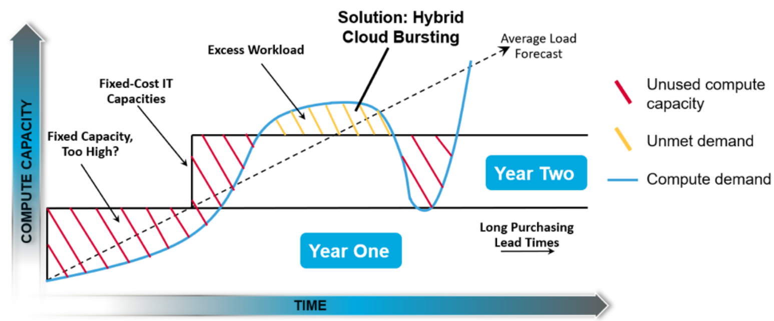
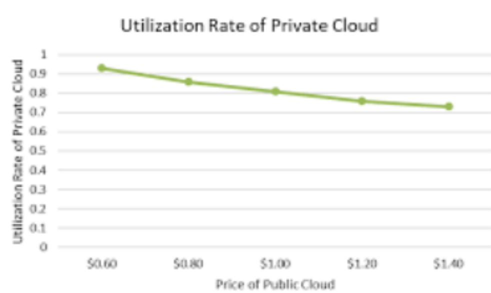


Figure (4) Compute Demand



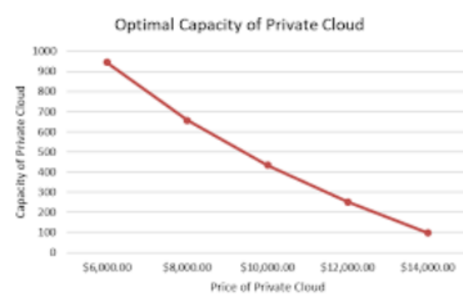


Figure (5)a: Price vs. utilization **Figure (5)b: Price vs. Optimal Capacity**

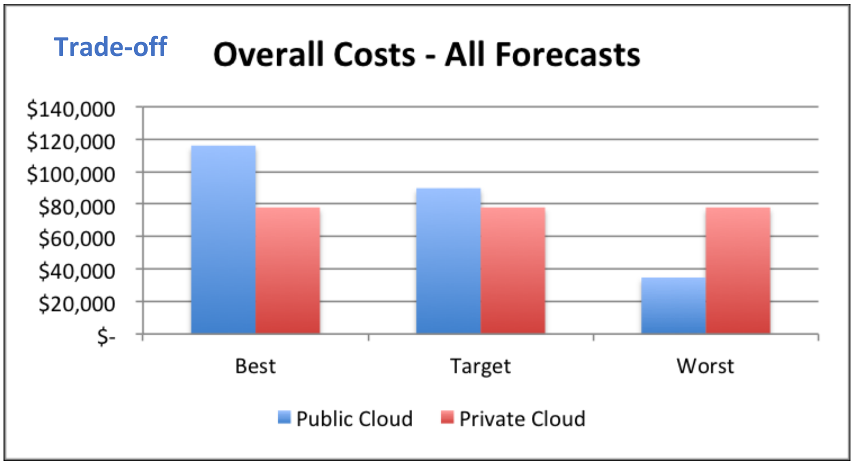


Fig (6): Price comparisons between public and private cloud

Total Cost = Private Cost + Public Cost

The best trade-off between Private and Public Cloud compute capacity is neither All Public or All Private. Figure 7 illustrates leveraging the existing resources to gain the benefits that on-premises and public cloud have to offer, while simplifying use and operations.

Hybrid Cloud Storage Capacity Optimization

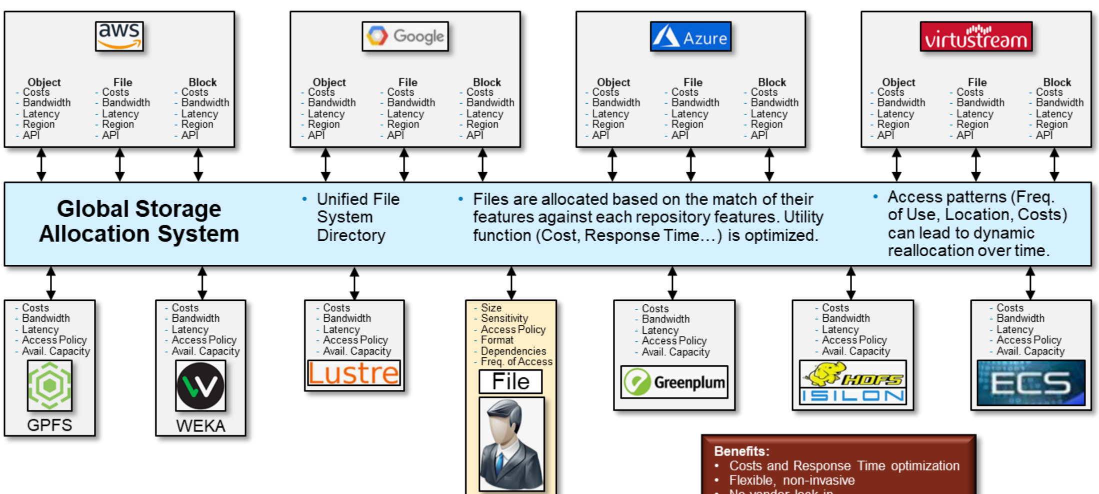


Figure (7): Leveraging existing resources

Hybrid Cloud Compute Capacity Optimization

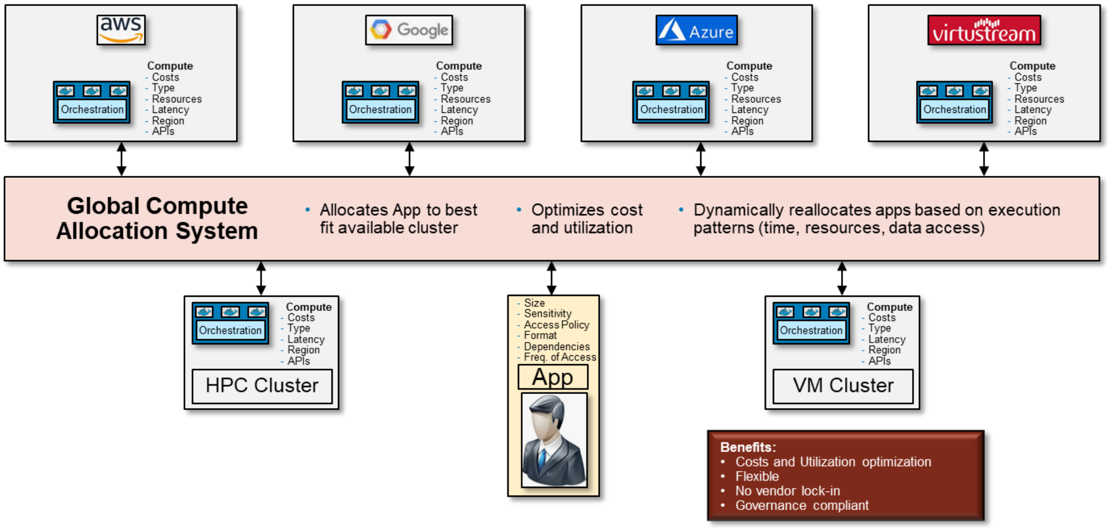


Figure (8): Compute Capacity Optimization

How are we managing the above solutions through Reinforcement Learning?

Architecture

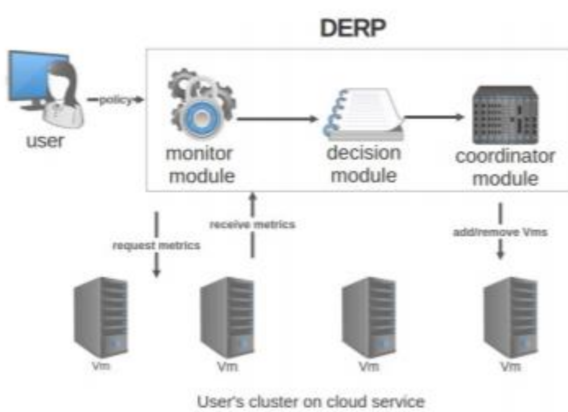


Figure (9): RL architecture overview

Algorithm 1: Simple Deep Q Learning

```
Initialize replay memory D
Initialize action-value function Q with random values θ
s = initialstates_1
for episode = 1 to M do
    Observe state s (by collecting metrics with the monitor module)
    With probability ε select a random action a_t (add/remove VM using coordinator module) otherwise select a = arg max_a' Q(s, a')
    observe reward r and new state s' (by collecting metrics with the monitor module).
    Store sequence (s, a, r, s') at the experience replay buffer
    Sample L number of past experiences < ss, aa, rr, ss' > from our memory buffer and training our agent with them, by calculating the Q targets (tt) for each minibatch transitions
```

$$tt = \begin{cases} rr \quad if \quad ss' \quad is \quad a \quad terminal \quad state \\ rr + \gamma max'_a Q(ss', aa') \\ for \quad non \quad terminal \quad ss' \end{cases}$$

```
    train the Q network using gradient descent with (tt − Q(ss, aa))^2 as loss
    s = s'
end for
```

Benefits

- Continuous improvement via Reinforced Learning
- Simultaneous allocation optimization of resources
- Future-Proof

Problem Solving Approach

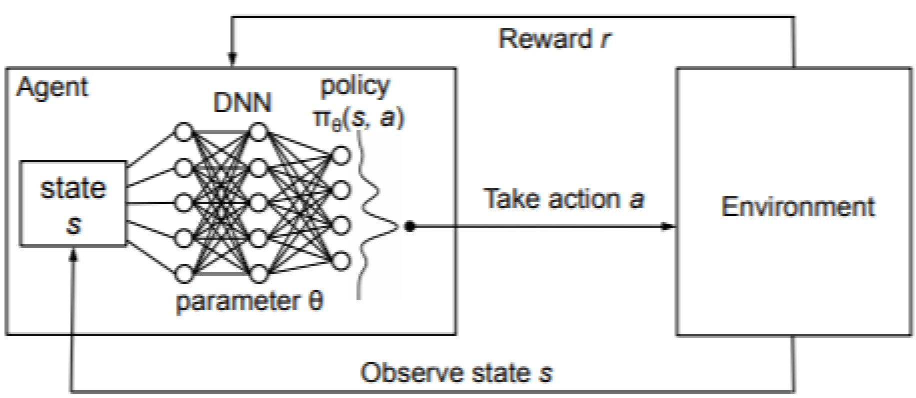


Figure (10): Reinforcement Learning with policy represented through Deep Neural Networks (DNN)

Think of DNN as a black box. Taking game-state as input returns the Q-value – like Q-learning does. However, we are now trying to recognize the patterns instead of just mapping every state to its best action. This wouldn't be possible with higher state spaces. For neural network to predict we must feed the pairs of input and output. Then neural network would train on the data to approximate the output based on the input by updating the parameters iteratively.

Temporal Difference Learning

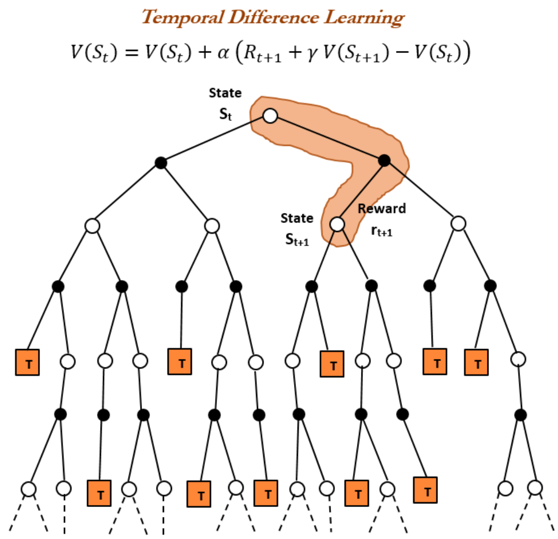


Figure (11): TD Learning structure

Temporal Difference (TD) learning is the central theme of reinforcement learning. TD learning is a combination of Monte Carlo (MC) and Dynamic Programming (DP) ideas. Like Monte Carlo methods, TD can learn directly from the experiences without prior knowledge on the domain and, as with Dynamic Programming, updates the estimates based on the learned estimates. However, unlike MC methods, TD does not wait for the outcome to update the estimates.

Hybrid Cloud Management Architecture based on Reinforcement Learning

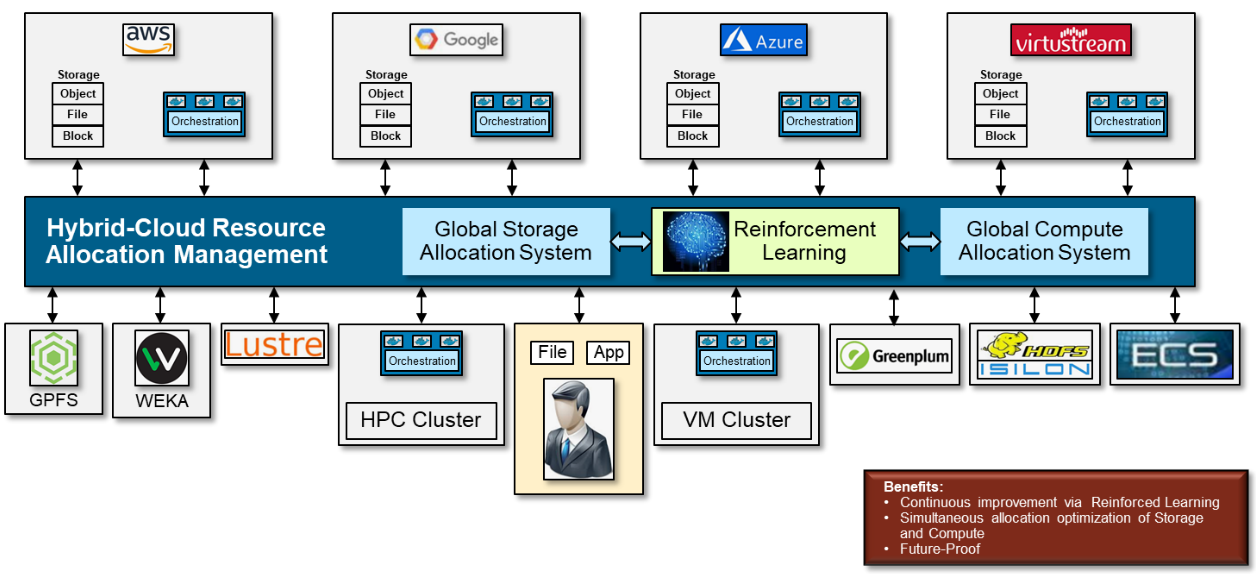
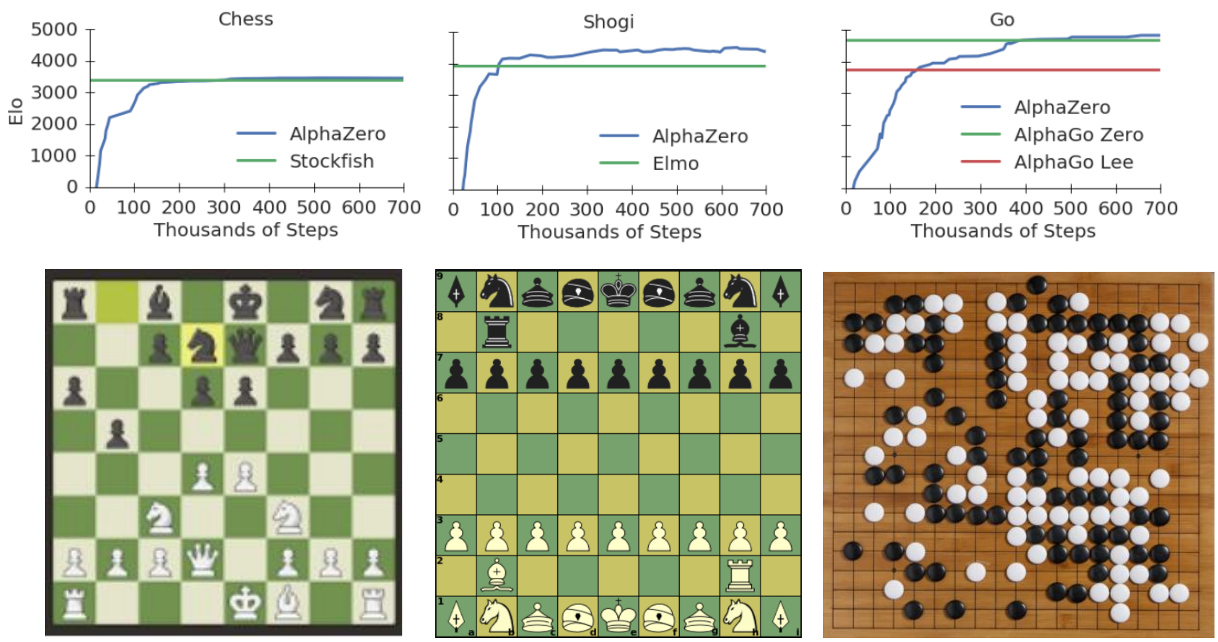


Figure (12): Hybrid Cloud Based Architecture

Reinforcement Learning: Examples

Starting from random play, given no domain knowledge except the rules of the game, Alpha zero achieved within 24 hours a superhuman level of play in the games of chess, shogi and Go and convincingly defeated a world-champion program in each play.



(a) *(b)* *(c)*

Fig 13: (a) Performance of AlphaZero in chess compared to StockFish (b) Performance of AlphaZero in Shogi when compared with Elmo (c) Performance of AlphaZero in Go when compared with AlphaGo Zero and AlphaGo Lee

Statistics of the Games

	Chess	Shogi	Go
Mini-batches	700k	700k	700k
Training Time	9h	12h	34h
Training Chess	44 million	24 million	21 million
Thinking Time	800 sins	800 sins	800 sins
	40ms	80 ms	200 ms

Table (1): Selected statistics of AlphaZero training in Chess, Shogi and Go

Reinforcement Learning vs. the other ML models

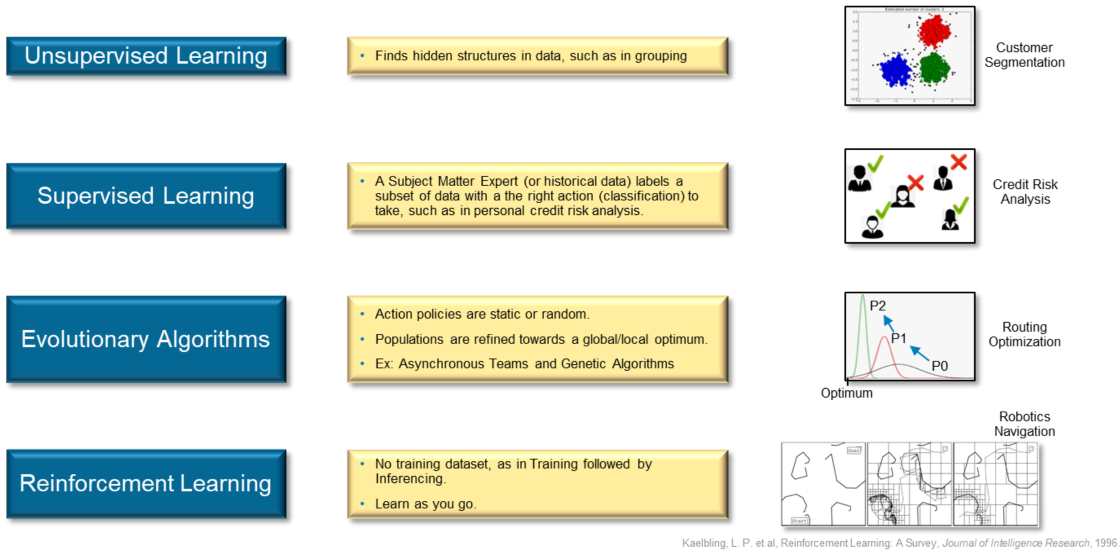


Table (2): RL and ML comparisons

Conclusion

This work explored the concept of Dynamic Resource in Cloud Computing and its benefits while understanding the trade-off between private and public clouds as there is no sole winner where the business needs to balance between the two. The research also sought an understanding about the cost of cloud services for consumers. This article will help readers gain technical knowledge and discover opportunities where business can leverage their existing resources and its benefits for cloud services. We delved into hybrid cloud's storage and compute optimization techniques and discussed implications of RL techniques into hybrid cloud management architecture that empower the business to aptly handle data center infrastructure.

References

https://www.refinitiv.com/perspectives/future-of-investing-trading/overcoming-the-barriers-to-cloud-adoption/
https://royalsocietypublishing.org/doi/full/10.1098/rsta.2012.0073
http://www.cslab.ntua.gr/~ikons/derp_proceedings.pdf
https://www.novatec-gmbh.de/en/blog/deep-q-networks/
https://lifein19x19.com/viewtopic.php?f=18&t=15289