Research and Reports Week 4

Name: Bingying Liang

ID: 48999397

CS7346 Cloud Computing

June 24 2023

1. Read the Netflix Challenge - data-center edition paper to understand the relationship between the Netflix challenge and the cluster resource allocation problem

Netflix challenge is its hundreds of thousands of servers in modern warehouse-scale systems make performance and efficiency optimizations pressing design challenges. These systems are traditionally considered homogeneous. However, that is not typically the case. Multiple server generations compose a heterogeneous environment, whose performance opportunities have not been fully explored since techniques that account for platform heterogeneity typically do not scale to the tens of thousands of applications hosted in large-scale cloud providers.[2]. The Netflix Challenge was a competition held by Netflix, the movie rental company, to improve its movie recommendation system. The competition was designed to spur innovation in the field of machine learning and attract the best minds to work on the Netflix movie recommendation problem.

The Cluster Resource Allocation problem deals with the efficient distribution and management of resources in a cluster of machines or servers to optimize performance, cost, and energy efficiency. ADSM, a scalable and efficient reecommendation system for application-to-server mapping in large-scale datacenters (DCs) that is QoS-aware. ADSM overcomes the drawbacks of previous techniques, by leveraging robust and computationally efficient analytical methods to scale to tens of thousands of applications with minimal overheads. It is also QoS-aware, mapping applications to platforms while enforcing strict QoS guarantees. ADSM is derived from validated analytical models, has low and bounded prediction errors, is simple to implement and scales to thousands of applications without significant changes to the system. They use two 40-machine clusters; a ”homogeneous” production cluster, with ten server configurations and a heterogeneous cluster with ten high-end and low-power machine types[2]. The ADSM can identifies similarities between different applications and then make efficient schedules. It can use different server configurations and and different end power machine types for different applications to place, which can improves performance and energy efficiency.

For their relationship, one is Scalability. To handle the large dataset provided by Netflix, it is necessary to use a cluster of computers. Efficiently allocating resources within this cluster would have been crucial for processing the data in a reasonable amount of time. Second is Parallel Computing: Large scale machine learning task, like the ones used in the Netflix Prize, often benefit from parallel computing, where tasks are divided among multiple processors to run simultaneously. Efficient resource allocation is essential to effectively manage these tasks and maximize parallelism, and so on.

2. Quasar classifies resource allocation for scale-up, scale-out, heterogeneity and interference. Why are classification criteria important, and how are they applied?

Classification criteria are important because the classification results are used to determine the right amount and specific set of resources assigned to the workload. Hence, Quasar avoids overprovisioning workloads that are currently at low load and can compensate for increased interference or the unavailability of high-end servers by assigning fewer or lower-quality resources to them.[1]. Scale-up, Scale-out, heteogeneity and interference provide a framework for effectively managing and optimizing resources in distributed computing environments. These criteria help determine how resources should be allocated and utilized to achieve better performance, scalability, and efficiency.

These classification criteria are applied dynamically and adaptively by Quasar based on the current system state, workload characteristics, and performance metrics. The framework continuously monitors the system, analyzes resource utilization, and applies appropriate resource allocation strategies to meet the changing demands of the distributed computing environment. By considering the specific requirements and challenges associated with scale-up, scale-out, heterogeneity, and interference, Quasar optimizes resource utilization and improves the overall performance of the system.

3. What are stragglers and how does Quasar deal with them?

Stragglers are tasks which in frameworks like Hadoop or Spark, individual tasks may take much longer to complete for reasons that range from poor work partitioning to network interference and machine instability.[1], which means stragglers refer to the slowest nodes or tasks that significantly delay the overall execution time of a computation. These stragglers can negatively impact the performance and efficiency of distributed systems.

These straggling tasks are typically identified and relaunched by the framework to ensure timely job completion [1, 3, 4, 17, 23, 37, 70] which are improved in detection in Hadoop in the following manner. Quasar calls the TaskTracker API in Hadoop and checks for under performing tasks (at least 50% slower than the median). For such tasks, Quasar injects two contentious microbenchmarks in the corresponding servers and reclassifies them with respect to interference caused and tolerated. If the results of the in-place classification differ from the original by more than 20%, we signal the task as a straggler and notify the Hadoop JobTracker to relaunch it on a newly assigned server. This allows Quasar to detect stragglers 19% earlier than Hadoop, and 8% earlier than LATE for the Hadoop applications described in the first scenario.[1]. Quasar is based on several innovative ideas. One of them regards reservation systems and the realization that users are seldom able to accurately predict the resource needs of their applications. Moreover, performance isolation, though highly desirable, is hard to implement. As a result, the execution time and the resources used by an application can be affected by other applications sharing the same physical platform(s)[3].

Overall, Quasar employs a combination of speculative execution, adaptive task scheduling, load balancing, and fine-grained monitoring to handle stragglers in distributed computing systems. These techniques help improve the efficiency and performance of data-parallel frameworks by minimizing the impact of slow nodes or tasks on overall computation time.

References

[1] Christina Delimitrou and Christos Kozyrakis. "Quasar: Resource-Efficient and QoS-Aware Cluster Management". In: ASPLOS '14 (Mar. 2014), pp. 127–144. DOI: 10.1145/2541940.2541941.

[2] Christina Delimitrou and Christos Kozyrakis. "The Netflix Challenge: Datacenter Edition". In: *IEEE Computer Architecture Letters* 12.1 (1905), pp. 29–32. DOI: https://doi.org/10.1109/L-CA.2012.10..

[3] Dan C Marinescu. *Cloud Computing: Theory and Practice. Third edition*. Cambridge, MA: Morgan Kaufmann is an imprint of Elsevier, 2023.