Bikash Sharma

Research Statement

www.cse.psu.edu/~bus145

Optimizing Performance in Cloud Platforms

Cloud computing has emerged as one of the pivotal future compute platforms. It is the most versatile form of utility computing, where computing infrastructure and applications are delivered as services over Internet. Cloud systems differ from traditional compute clusters like grids and data centers both in terms of the services and features provided, and the underlying platform. The important differentiating features include elasticity, multi-tenancy, shared resource pooling and self-organization. There are various dimensions of cloud computing, that relates to the different challenges associated with this pay-as-you-go computing paradigm. In this research work, I have made contributions towards addressing one such important and vital facet – *optimizing performance in clouds*. I have pursued this aspect in three different but inter-related perspectives. These are (a) modeling and characterization of cloud workloads; (b) resource management of MapReduce cloud clusters; and (c) performance diagnosis in cloud platforms.

My research methodology combines developing simulation framework for representing large scale systems like multi-tier data centers, prototyping real systems, experimentation with working systems and real workloads and benchmarks, modeling, and statistical analysis. These methodologies are geared for addressing specific challenges towards realizing the full potentials of cloud computing. My work has extensively used modeling and characterization techniques to better understand the workload and infrastructure behaviors of the cloud systems. The algorithms deployed in my projects are designed with the underlying requirements of high scalability and implementation efficiency, which are critical aspects of cloud systems. In all my research projects, I have tried to follow my philosophy that a successful project is the right blend of innovative research, well thought architecture and fine-grained efficient implementation of the target system, and widespread acceptance in terms of its utility, performance and scalability.

Characterization and Modeling of Cloud Workloads

The first attribute of cloud performance is related to the proper understanding of the workloads that drive the cloud systems. Evaluating the functionality of large data centers and clouds requires performance benchmarks with representative workloads to gauge the performance impact of system changes, assess changes in application codes, machine configurations, and scheduling algorithms. To achieve this objective, it is quintessential to construct workload characterizations from which realistic and representative performance benchmarks can be derived. Workload specific performance insights are essential to better manage, provision and optimize these large systems. There are different dimensions of cloud workloads which are related to the job arrival rate, resource limit, reservation and usage of jobs, and completion time of jobs. Each of these characteristics in turn manifests into different observed performance impacts. Amongst these, there exists a particular workload characteristic, called *task placement constraint* that tends to have significant impact on scheduling performance, as quantified by task pending delay and cluster resource utilization.

In this research project, I have analyzed the performance impacts of task placement constraints [3] in large compute clusters like those in Google cloud back-end, and developed methodologies to incorporate them into existing performance benchmarks to make them more realistic and cloud representative. This work involved design and implementation of large-scale simulation tools that run production Google scheduler code; analysis of scheduling traces across three large Google compute clusters; and development of well validated benchmarking algorithms. The most important impacts of this work include (i) techniques that allows existing general purpose benchmarks to incorporate the notion of task constraints to make them representative of real cloud workloads; (ii) mathematical model to quantify the impact of constraints on scheduling performance; (iii) support (in terms of real cluster traces statistics and methodology) to make existing resource scheduling algorithms more effective by taking into consideration the underlying infrastructure heterogeneity.

Resource Management of MapReduce Cloud Clusters

Cloud computing has become analogous to large scale data intensive computing, where huge amount of data needs to be processed and analyzed within the constraints of efficiency and economy of scale. The second dimension of cloud performance relates to the effective management of the resources of cloud clusters running representative large data intensive applications like Hadoop MapReduce. In typical Hadoop MapReduce clusters, applications from multiple users share the same set of resources, and efficient management of these resources is an important design choice both with respect to application performance and cluster resource utilization standpoints. The key challenge in such shared Hadoop MapReduce clusters is the ability to automatically manage and optimally control the allocation of resources to multiple applications, within performance goals. This translates into important performance implications like reduced latency, better throughput and increased resource utilization.

Inspired by these, in this research, I have addressed the problem of how to efficiently manage the resources in cloud clusters running large data intensive applications like MapReduce. Specifically, I have identified three potential drawbacks of current slot-based resource allocation in Hadoop MapReduce. These are related to the static, coarse-grained and insufficient isolation based resource scheduling. Towards this, I have designed and implemented *MROrchestrator* [2] *,* which is a resource management framework for MapReduce clusters. MROrchestrator builds a run-time performance model of constituent map/reduce tasks based on their progress and resource footprints. Its distributed master-slave architecture identifies both resource hogging and resource straggling tasks, and makes coordinated run-time resource allocation/adjustment decision. Evaluation on both 24-node native and virtualized Hadoop MapReduce clusters demonstrates the efficiency of MROrchestrator in terms of achieving 25% improvement in cluster resource utilization and around 38% reduction in job completion times. MROrchestrator is in the process of being released as an open source patch to Apache Hadoop incubator.

Problem Determination in Shared Dynamic Clouds

The third facet of cloud performance relates to providing efficient reliability and problem determination in virtualized cloud systems. This research direction explores identifying the various performance anomalies or faults in virtualized consolidated cloud environments, which manifest into poor performance of the system and the hosted applications. Previous literatures including commercial performance analysis tools have thoroughly addressed problem determination and diagnosis in traditional distributed systems like Grids and data centers. Here, the focus is on application related anomalies. However, faults that manifest as artifacts of virtualization technology in cloud environment have not been studied in detail before. The shared virtualized cloud infrastructure makes it difficult to pinpoint the location, identify the root

cause, and isolate the faults in the system. The impact of failures is also magnified as a failure of a single physical server could affect all the virtual machines running on it. The dynamism in the application workload, virtual machines movement across the hosts and cloud activities (VMs can be resized dynamically and can be migrated from one server to another), makes it difficult to monitor and differentiate between actual performance anomalies and workload changes. The various cloud related activities such as resizing, migration and reconfiguration can themselves cause performance problems, if not performed smartly. Distinguishing such cloud related anomalies from other application related anomalies is non-trivial and has never been attempted yet. The exascale nature of such systems demands that any problem determination framework needs to be quick and efficient monitoring huge amounts of data, and in accurately determining and diagnosing faults. The main challenges faced by any problem determination framework for cloud environments are the shared virtualized infrastructure, dynamism, and exascale. These challenges make traditional fault management tools (both commercial and noncommercial) unsuitable for virtualized cloud environments.

In this work, I have addressed this missing piece by proposing an end-to-end automated, scalable and workload agnostic fault detection and diagnosis framework, called *CloudPD* [3], that is geared towards faults that arise due to the various virtualization related cloud activities. CloudPD uses a three-stage methodology to effectively identify and diagnose the various kinds of faults occurring in clouds. CloudPD is evaluated on IBM cloud testbed running cloud representative benchmarks (Hadoop, Olio and RUBiS) and real enterprise traces. Evaluation results demonstrate the efficiency of CloudPD in terms of being able to achieve around 88% accuracy with low false positive rate, fast and calibration/analysis time of less than 30 seconds. Some of the key characteristics of CloudPD include high scalability, lightweight, agnostic to applications, and platform generic framework.

Datacenter Management

Datacenter management is an integral part towards understanding and maintaining the efficiency of the data centers. One critical component in this is to explore the “what if” analysis. For example, what happens when you replace your 1GB Ethernet interconnect by 100GB interconnect. How does this replacement translates into performance (latency, throughput, etc). To better understand and be able to answer these kinds of scenarios, simulation framework comes very handy because running real experiments is not always feasible, and modeling these complex systems is non-trivial. Besides simulation provides further benefits like fast turnaround time, flexibility, generality, etc. Inspired by this, I have collaborated in the design and implementation of *MDCSim* [5], which is a multi-tier data center simulation platform. MDCSim is a comprehensible, flexible and modular simulator that simulates kernel, user-level communication and application level characteristics of a multi-tier data center. My main contribution was to design and implement the power model, that estimates the total power consumption across the data center under various workload settings, coupled with conducting analysis case-study for different data center interconnects and data center configurations.

I have also collaborated in the research focusing on the energy management of multi-tier data centers [4] through a hybrid methodology involving dynamic server provisioning, dynamic voltage frequency scaling and dynamic power management of server systems.

Future Work

The emergence of cloud computing has left very interesting unsolved problems in different fronts. My ongoing and future research work will focus on making contributions towards providing novel and practical solutions to some of these problems. Some specific directions/ plans are described below:

(a) Develop new scheduling algorithms that explicitly leverage the demand of tasks for specific constraints and the supply of machines with matching properties. Following which, design a simulation platform to conduct such scheduling/QoS/performance trade-off studies. Another related focus is to develop an initial version of a mathematical (control theoretic) scheduling framework that incorporates the notion of task placement constraints and machine properties.
(b) Extend MROrchestrator with control of other shared resources like disk and network bandwidth, besides CPU and memory. This will be followed by the design of a resource scheduling disciplines for Hadoop MapReduce clusters that augments MROrchestrator for better resource scheduling, and harness the heterogeneity of the underlying infrastructure. I plan to leverage here the task placement constraints to better match the demands/preferences of tasks for certain machine configurations with available machine properties, besides their resource requirements.
(c) Extend CloudPD with more diverse and extensive types of anomalies that occurs in cloud environment. This will be followed by the integration of CloudPD with a dynamic cloud reconfiguration manager, with the objective to provide an end-to-end dynamic, flexible and robust consolidation and fault diagnosis framework for virtualized cloud systems.
(d) Optimize resource scheduling in hybrid clusters consisting of both native and virtualized machines, running both transactional and batch workloads like MapReduce [6]. The challenge here is to fine-tune the placement of big data workloads on virtualized environment to provide best-effort delivery, while maintaining the strict service level objectives of interactive workloads.
(e) I am also interested to explore the benefits and associated challenges involved in the usage of solid state devices (SSDs) for enhancing the storage efficiency of Hadoop stack.

References

1. *Bikash Sharma*, Praveen Jayachandran, Akshat Verma, and Chita R. Das, CloudPD: Problem Determination and Diagnosis in Shared Dynamic Clouds, CSE Technical Report, Pennsylvania State University.
2. *Bikash Sharma*, Ramya Prabhakar, Seung-Hwan Lim, Mahmut T. Kandemir, and Chita R. Das, MROrchestrator: A Fine-Grained Resource Orchestration Framework for MapReduce Clusters, 5th IEEE International Conference on Cloud Computing (CLOUD) 2012.
3. *Bikash Sharma*, Victor Chudnovsky, Joseph L. Hellerstein, Rasekh Rifaat, and Chita R. Das, Modeling and Synthesizing Task Placement Constraints in Google Compute Clusters, 2nd ACM Symposium on Cloud Computing (SOCC) 2011.
4. Seung-Hwan Lim, *Bikash Sharma*, Byung Chul Tak, and Chita R. Das, A Dynamic Energy Management In Multi-tier Data Centers, IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS) 2011.
5. Seung-Hwan Lim, *Bikash Sharma*, Gunwoo Nam, Eun Kyoung Kim, Chita R. Das, MDCSim: A Multi-tier Data Center Simulation Platform, IEEE Cluster 2009.

6. *Bikash Sharma,* Timothy Wood, Chita R. Das, “HybridMR: A Hierarchical MapReduce Scheduler for Hybrid Clouds”, under preparation for submission to IEEE ICDCS 2013.