INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 2 ISSUE 6 (2016) PAGES 1305 - 1309
RECEIVED : 29/10/2016. PUBLISHED: 07/11/2016
November 7, 2016

DYNAMIC LOAD-BALANCING PROCESS ON PUBLIC CLOUD NETWORK

[1]K.Punniya, [2]S.Pradeepa,

[1]PG Scholar, Dept Of Computer Science Engineering,Mailam Engineering College,Villupuram,

[2]Assistant Professor Of Computer Science Engineering,Mailam Engineering College,Villupuram.

Abstract:

Load-balanced flow scheduling for big data centers in clouds, in which a large amount of data needs to be transferred frequently among thousands of interconnected servers, is a key and challenging issue. The Open Flow is a promising solution to balance data flows a data center network through its programmatic traffic controller . Existing Open Flow based scheduling scheme, however, statically set up routes only at the initializaton stage of data transmissions, which suffers from dynamical flow distribution and changing network states in data centers and often results in poor system performance .we propose a novel dynamical load-balanced scheduling (DLBS) approach for maximizing the network throughput while balancing workload dynamically. We firstly formulate the DLBS problem, and then develop a set of efficient heuristic scheduling algorithms for the two typical Open Flow network models, which balance data flows time slot by time slot. Experimental results demonstrate that our DLBS approach significantly out performs other representative load-balanced scheduling algorithms Round Robin and LOBUS; and the higher imbalance degree data flows in data centers exhibit, the more improvement our DLBS approach will bring to the data centers.

Index Terms: Flow scheduling, load balancing, data center, Cloud Computing, Open Flow.

1. INTRODUCTION

Cloud computing is a type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., computer networks, servers, storage, applications and services), which can be rapidly provisioned and released with minimal management effort. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third-party data centers that may be located far from the user–ranging in distance from across a city to across the world. Cloud computing relies on sharing of resources to achieve coherence and economy of scale, similar to a utility (like the electricity grid) over an electricity network. Advocates claim that cloud computing allows companies to avoid upfront infrastructure costs (e.g., purchasing servers). As well, it enables organizations to focus on their core businesses instead of spending time and money on computer infrastructure. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and enables Information Technology (IT) teams to more rapidly adjust resources to meet fluctuating and unpredictable business demand. Cloud providers typically use a "pay as you go" model. This will lead to unexpectedly high charges if administrators do not adapt to the cloud pricing model.

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 2 ISSUE 6 (2016) PAGES 1305 - 1309
RECEIVED : 29/10/2016. PUBLISHED: 07/11/2016

Cost reductions are claimed by cloud providers. A public-cloud delivery model converts , as infrastructure is typically provided by a third party and need not be purchased for one-time or infrequent intensive computing tasks. Pricing on a utility computing basis is "fine-grained", with usage-based billing options. As well, less in-house IT skills are required for implementation of projects that use cloud computing.

2. LITERATURE SURVEY

A new technique for assigning tasks to processors is developed and the schedulability of the algorithm is analyzed for worst-case performance. We prove that, if the workload (utilization) of a given task set is less than or equal to 55.2% of the total processing capacity on m processors, then all tasks meet their deadlines. During task assignment, the total work load is regulated to the processors in such a way that a subset of the processors are guaranteed to have an individual processor load of at least 55.2%. Due to such load regulation, our algorithm can be used efficiently as an admission controller for online task scheduling. And this online algorithm is scalable with increasing number of cores in chip multiprocessors. With phenomenal growth of distributed real time control system in high speed applications, the usage of multiprocessor system has become essential. In order to obtain performance benefits from the resources available in multiprocessor system, real time task have to coordinate and share resources to their maximum use. Load balancing amongst processors of multiprocessing system is one of the most important issues for task scheduling. The processors working on balanced load perform better compared to unbalanced load in terms of response time and resource utilization. Heat dissipation on individual processor is also reduced. In this paper, we propose an algorithm named LBPSA (Load Balanced Partitioning and Scheduling Algorithm) that offers load balancing amongst processors. The Quasi-Partitioning Scheduling algorithm optimally solves the problem of scheduling a feasible set of independent implicit-deadline sporadic tasks on a symmetric multiprocessor. It iteratively combines bin-packing solutions to determine a feasible task-to-processor allocation, splitting task loads as needed along the way so that the excess computation on one processor is assigned to a paired processor. Though different in formulation, QPS belongs in the same family of schedulers as RUN, which achieve optimality using a relaxed (partitioned) version of proportionate fairness.

3. CLOUD MANAGEMENT:

Fig.1.Cloud computing

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 2 ISSUE 6 (2016) PAGES 1305 - 1309
RECEIVED : 29/10/2016. PUBLISHED: 07/11/2016
November 7, 2016

Cloud management is the management of cloud computing products and services. Public clouds are managed by public cloud service providers, which include the public cloud environment's servers, storage, networking and data center operations. Users of public cloud services can generally select from three basic categories: User self-provisioning: Customers purchase cloud services directly from the provider, typically through a web form or console interface. The customer pays on a per-transaction basis. Advance provisioning: Customers contract in advance a predetermined amount of resource, Dynamic provisioning: which are prepared in advance of service. The customer pays a flat fee or a monthly fee.The provider allocates resources when the customer needs them, then decommissions them when they are no longer needed. The customer is charged on a pay-per-use basis. Managing a private cloud requires software tools to help create a virtualized pool of compute resources, provide a self-service portal for end users and handle security, resource allocation, tracking and billing. Management tools for private clouds tend to be service driven, as opposed to resource driven, because cloud environments are typically highly virtualized and organized in terms of portable workloads. In hybrid cloud environments, compute, network and storage resources must be managed across multiple domains, so a good management strategy should start by defining what needs to be managed, and where and how to do it. Policies to help govern these domains should include configuration and installation of images, access control, and budgeting and reporting. Access control often includes the use of Single sign-on (SSO), in which a user logs in once and gains access to all systems without being prompted to log in again at each of them.

4. ASPECTS OF CLOUD MANAGEMENT SYSTEMS

A cloud management system combines software and technologies in a design for managing cloud environments. Software developers have responded to the management challenges of cloud computing manage a pool of heterogeneous compute-resources provide access to end users monitor security manage resource allocation manage tracking For composite applications, cloud management systems also encompass frameworks for workflow-mapping and management .Enterprises with large-scale cloud implementations may require more robust cloud management tools which include specific characteristics, such as the ability to manage multiple platforms from a single point of reference, or intelligent analytics to automate processes like application lifecycle manage a pool of heterogeneous compute-resources provide access to end users monitor security manage resource allocation manage tracking For composite applications, cloud management systems also encompass frameworks for workflow-mapping and management .Enterprises with large-scale cloud implementations may require more robust cloud management tools which include specific characteristics, such as the ability to manage multiple platforms from a single point of reference, or intelligent analytics to automate processes like application lifecycle management. High-end cloud management tools should also have the ability to handle system failures automatically with capabilities such as self-monitoring, an explicit notification mechanism, and include failover and self-healing capabilities.

5. PERFORMANCE MEASURES

Scalability and elasticity via dynamic ("on-demand") provisioning of resources on a fine-grained, self-service basis in near real-time (Note, the VM startup time varies by VM type, location, OS and cloud providers), without users having to engineer for peak loads. This gives the ability to scale up when the usage need increases or down if resources are not being used.

Performance is monitored by IT experts from the service provider, and consistent and loosely coupled architectures are constructed using web services as the system interface.

Productivity may be increased when multiple users can work on the same data simultaneously, rather than waiting for it to be saved and emailed. Time may be saved as information does not need to be re-entered when fields are matched, nor do users need to install application software upgrades to their computer.

CONCLUSION

In this work, we address the load-balanced scheduling problem through balancing transmission traffic dynamically and globally in cloud data centers. Aiming at two typical OpenFlow architectures: FPN and FTN, we proposed and implemented a set of efficient scheduling algorithms DLBS-FPN and DLBS-FTN respectively. Compared with existing scheduling schemes for load balancing and route selection, our algorithms have two main advantages. Firstly, our algorithms can adapt to dynamical network states and changing traffic requirements through updating load imbalance factor δ(t) and accordingly balancing the transmission load slot by slot during data transmissions. Next, our algorithms can globally balance transmission traffic in the whole network by means of evaluating link, path and In this work, we address the load-balanced scheduling problem through balancing transmission traffic dynamically and globally in cloud data centers. Aiming at two typical OpenFlow architectures: FPN and FTN, we proposed and implemented a set of efficient scheduling algorithms DLBSFPN and DLBS-FTN respectively. Compared with existing scheduling schemes for load balancing and route selection,our algorithms have two main advantages. Firstly, our algorithms can adapt to dynamical network states and changing traffic requirements through updating load imbalance factor $\delta(t)$ and accordingly balancing the transmission load slot by slot during data transmissions. Next, our algorithms can globally balance transmission traffic in the whole network.

REFERENCES

[1] T.N.Anitha, R.Balakrishna, An Efficient and Scalable Content Based Dynamic Load Balancing Using Multiparameters on Load Aware Distributed Multi-Cluster Servers, IJEST, 2008.

[2] M.Al-Fares, S.Radhakrishnan, B.Raghavan, N.Huang and A.Vahdat. Hedera: Dynamic flow scheduling for data center networks. Proc. of Networked Systems Design and Implementation (NSDI) Symposium, 2010.

[3] M.Chowdhury, Y.Zhong and I.Stoica. Efficient coflow scheduling with Varys. Proc. of the 2014 ACM conference on SIGCOMM, vol.44, No.4, pp.443-454, 2014.

[4] Q.Zhang, M.F.Zhani, Y.Yang et al. PRISM: Fine-Grained Resource-Aware Scheduling for MapReduce. IEEE Transactions on Cloud Computing, Vol.PP, No.99, 2015.

[5] Y.Shi, C.Tian, Y.Zhang, and S.Lu, BCube: A High Performance,Server-centric Network Architecture for Modular Data Center, In Proceedings of SIGCOMM 2009.

[6] Y.Yuan, D.Wang and J.Liu. Joint scheduling of MapReduce jobs with servers: Performance bounds and experiments. Proc. Of IEEE INFOCOM, 2014, pp.2175-2183.

[7] R.Kanagavelu, L.N.Mingjie, M.M.Khin et al. OpenFlow based control for re-routing with differentiated flows in Data Center Networks. Proc. of 18th IEEE International Conference on Networks (ICON), 2012, pp.228-233.

[8] A.Bianco, V.Krishnamoorthi, N.F.Li et al. OpenFlow driven Ethernet traffic analysis. Proceedings of 2014 IEEE International Conference on Communications (ICC), 2014, pp.3001-3006