INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

Data storage and Precision of Dynamism-Efficient Fault-Tolerant in Mobile Cloud

B. Naseema Begum#1, S. Prasanna*1

Mailam Engineering College, Mailam#1 *1

naseemabegumbabu@gmail.com

Abstract- Although the technology improvement in hardware for hand-held mobile devices, video and image storage devices requires large calculation or computation and storage capacity. Recent research has implemented to address these issues by deploying remote servers, such as clouds and peer mobile devices. Here the mobile devices has implemented on dynamic networks with frequent topology changes and mobility in mobile cloud. In existing work, challenges of reliability and energy efficiency are not unaddressed. In this proposed system, we provide the solution for addressing these issues. Here we implement a combined approach of k-out-of-n computing for both data storage and processing in mobile cloud. In our system, mobile devices successfully save or process data, in the most energy-efficient way, as long as k out of n remote servers are processed. From this proposed system we have also achieved feasibility of the system. Additionally this system performs the fault tolerance and energy efficiency performance in large scale networks.

Keywords- K out of n, cloud, Topology, Peer

I. Related Work

Although the technology improvement in hardware for hand-held mobile devices, video and image storage devices requires large calculation or computation and storage capacity. In existing work, challenges of reliability and energy efficiency are not unaddressed.

1.1 Drawback

In existing system provides less efficiency and reliability when access the system.

II. Proposed Method

In this proposed system, we provide the solution for addressing these issues. Here we implement a combined approach of k-out-of-n computing for both data storage and processing in mobile cloud. In our system, mobile devices successfully save or process data, in the most energy-efficient way, as long as k out of n remote servers are processed.

2.1 Benefits

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

In this proposed system, it provides enhanced efficiency and high reliability. Additionally this system performs the fault tolerance and energy efficiency performance in large scale networks.

III. Introduction

Personal mobile devices have gained enormous popularity in recent years. Due to their limited resources (e.g., computation, memory, energy), however, executing sophisticated applications (e.g., video and image storage and processing, or map-reduce type) on mobile devices remains challenging. As a result, many applications rely on offloading all or part of their works to “remote servers” such as clouds and peer mobile devices. For instance, applications such as Google Goggle and Siri process the locally collected data on clouds. Going beyond the traditional cloud based scheme, recent research has proposed to offload processes on mobile devices by migrating a Virtual Machine (VM) overlay to nearby infrastructures. This strategy essentially allows offloading any process or application, but it requires a complicated VM mechanism and a stable network connection. Some systems (e.g., Serendipity) even leverage peer mobile devices as remote servers to complete computation intensive job. In dynamic networks, e.g., mobile cloud for disaster response or military operations, when selecting remote servers, energy consumption for accessing them must be minimized while taking into account the dynamically changing topology. Serendipity and other VM-based solutions considered the energy cost for processing a task on mobile devices and offloading a task to the remote servers, but they did not consider the scenario in a multi-hop and dynamic network where the energy cost for relaying/transmitting packets is significant. Furthermore, remote servers are often inaccessible because of node failures, unstable links, or node-mobility, raising a reliability issue. Although Serendipity considers intermittent connections, node failures are not taken into account; the VM-based solution considers only static networks and is difficult to deploy in dynamic environments. In this article, we propose the first framework to support fault-tolerant and energy-efficient remote storage & processing under a dynamic network topology, i.e., mobile cloud. Our framework aims for applications that require energy-efficient and reliable distributed data storage & processing in dynamic network E.g., military operation or disaster response. We integrate the k-out of-n reliability mechanism into distributed computing in mobile cloud formed by only mobile devices. K-out-of n, a well-studied topic in reliability control, ensures that a system of n components operates correctly as long as k or more components work. More specifically, we investigate how to store data as well as process the stored data in mobile cloud with k-out-of-n reliability such that: 1) the energy consumption for retrieving distributed data is minimized; 2) the energy

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

consumption for processing the distributed data is minimized; and 3) data and processing are distributed considering dynamic topology changes. In our proposed framework, a data object is encoded and partitioned into n fragments, and then stored on n different nodes. As long as k or more of the n nodes are available, the data object can be successfully recovered. Similarly, another set of n nodes are assigned tasks for processing the stored data and all tasks can be completed as long as k or more of the n processing nodes finish the assigned tasks. The parameters k and n determine the degree of reliability and different (k, n) pairs may be assigned to data storage and data processing. System administrators select these parameters based on their reliability requirements. The contributions of this article are as follows: • It presents a mathematical model for both optimizing energy consumption and meeting the fault tolerance requirements of data storage and processing under a dynamic network topology. • It presents an efficient algorithm for estimating the communication cost in a mobile cloud, where nodes fail or move, joining/leaving the network. • It presents the first process scheduling algorithm that is both fault-tolerant and energy efficient. • It presents a distributed protocol for continually monitoring the network topology, without requiring additional packet transmissions. • It presents the evaluation of our proposed framework through a real hardware implementation and large scale simulations.

IV. Literature Review

[6] Describes about, Mobile applications are becoming increasingly ubiquitous and provide ever richer functionality on mobile devices. At the same time, such devices often enjoy strong connectivity with more powerful machines ranging from laptops and desktops to commercial clouds. This paper presents the design and implementation of Clone Cloud, a system that automatically transforms mobile applications to benefit from the cloud. The system is a flexible application partitioned and execution runtime that enables unmodified mobile applications running in an application-level virtual machine to seamlessly off-load part of their execution from mobile devices onto device clones operating in a computational cloud. Clone Cloud uses a combination of static analysis and dynamic profiling to partition applications automatically at a fine granularity while optimizing execution time and energy use for a target computation and communication environment. At runtime, the application partitioning is effected by migrating a thread from the mobile device at a chosen point to the clone in the cloud, executing there for the remainder of the partition, and re-integrating the migrated thread back to the mobile device. Our evaluation shows that Clone Cloud can adapt application partitioning to different environments, and can help some applications achieve as much as a 20x execution speed-up and a 20-fold decrease of energy spent on the mobile device. [22]

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES
ISSN 2454-9924 Volume: 1 Issue: 1(2015)

Describes about Smartphones have exploded in popularity in recent years, becoming ever more sophisticated and capable. As a result, developers worldwide are building increasingly complex applications that require ever increasing amounts of computational power and energy. In this paper we propose Think Air, a framework that makes it simple for developers to migrate their smartphone applications to the cloud. Think Air exploits the concept of smartphone virtualization in the cloud and provides method-level computation offloading. Advancing on previous work, it focuses on the elasticity and scalability of the cloud and enhances the power of mobile cloud computing by parallelizing method execution using multiple virtual machine (VM) images. We implement Think Air and evaluate it with a range of benchmarks starting from simple micro-benchmarks to more complex applications. First, we show that the execution time and energy consumption decrease two orders of magnitude for a N-queens puzzle application and one order of magnitude for a face detection and a virus scan application. We then show that a parallelizable application can invoke multiple VMs to execute in the cloud in a seamless and on-demand manner such as to achieve greater reduction on execution time and energy consumption. We finally use a memory hungry image combiner tool to demonstrate that applications can dynamically request VMs with more computational power in order to meet their computational requirements. [8] Describes about Mobile devices are increasingly being relied on for services that go beyond simple connectivity and require more complex processing. Fortunately, a mobile device encounters, possibly intermittently, many entities capable of lending it computational resources. At one extreme is the traditional cloud-computing context where a mobile device is connected to remote cloud resources maintained by a service provider with which it has an established relationship. In this paper we consider the other extreme, where a mobile device's contacts are only with other mobile devices, where both the computation initiator and the remote computational resources are mobile, and where intermittent connectivity among these entities is the norm. We present the design and implementation of a system, Serendipity that enables a mobile computation initiator to use remote computational resources available in other mobile systems in its environment to speedup computing and conserve energy. We propose a simple but powerful job structure that is suitable for such a system. Serendipity relies on the collaboration among mobile devices for task allocation and task progress monitoring functions. We develop algorithms that are designed to disseminate tasks among mobile devices by accounting for the specific properties of the available connectivity. We also undertake an extensive evaluation of our system, including experience with a prototype that demonstrates Serendipity's performance. [4] Describes about Cloud computing offers

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

utility-oriented IT services to users worldwide. Based on a pay-as-you-go model, it enables hosting of pervasive applications from consumer, scientific, and business domains. However, data centers hosting Cloud applications consume huge amounts of electrical energy, contributing to high operational costs and carbon footprints to the environment. Therefore, we need Green Cloud computing solutions that can not only minimize operational costs but also reduce the environmental impact. In this paper, we define an architectural framework and principles for energy-efficient Cloud computing. Based on this architecture, we present our vision, open research challenges, and resource provisioning and allocation algorithms for energy-efficient management of Cloud computing environments. The proposed energy-aware allocation heuristics provision data center resources to client applications in a way that improves energy efficiency of the data center, while delivering the negotiated Quality of Service (QoS). In particular, in this paper we conduct a survey of research in energy-efficient computing and propose: (a) architectural principles for energy-efficient management of Clouds; (b) energy-efficient resource allocation policies and scheduling algorithms considering QoS expectations and power usage characteristics of the devices; and (c) a number of open research challenges, addressing which can bring substantial benefits to both resource providers and consumers. We have validated our approach by conducting a performance evaluation study using the Cloud Sim toolkit. The results demonstrate that Cloud computing model has immense potential as it offers significant cost savings and demonstrates high potential for the improvement of energy efficiency under dynamic workload scenarios. [23] Describes about A system reliability optimization approach with multiple k-out-of-n subsystems connected in series is presented. The design objective is to select multiple components to maximize system reliability while satisfying system requirement constraints such as cost and weight. Mixing of non-identical component types is allowed in each subsystem as well as at system level. In practice, different design configurations can be associated with different values of k. Thus k is considered a decision variable that can be selected as part of the design process. Previously, k was generally considered to be one to maximize the reliability of a series-parallel system. In this paper, k >; 1 is considered. The approach is demonstrated on a well-known test problem with interesting results. A Genetic algorithm is effectively applied to solve the optimization problem. [5] Describes about Cloud computing offers utility-oriented IT services to users worldwide. Based on a pay-as-you-go model, it enables hosting of pervasive applications from consumer, scientific, and business domains. However, data centers hosting Cloud applications consume huge amounts of electrical energy, contributing to high operational costs and carbon footprints to the environment. Therefore, we need

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

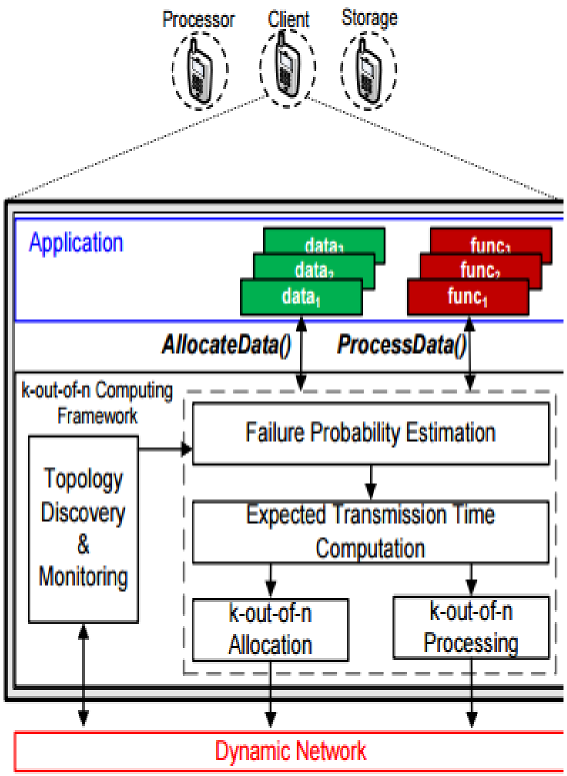
Green Cloud computing solutions that can not only minimize operational costs but also reduce the environmental impact. In this paper, we define an architectural framework and principles for energy-efficient Cloud computing. Based on this architecture, we present our vision, open research challenges, and resource provisioning and allocation algorithms for energy-efficient management of Cloud computing environments. The proposed energy-aware allocation heuristics provision data center resources to client applications in a way that improves energy efficiency of the data center, while delivering the negotiated Quality of Service (QoS). In particular, in this paper we conduct a survey of research in energy-efficient computing and propose: (a) architectural principles for energy-efficient management of Clouds; (b) energy-efficient resource allocation policies and scheduling algorithms considering QoS expectations and power usage characteristics of the devices; and (c) a number of open research challenges, addressing which can bring substantial benefits to both resource providers and consumers. We have validated our approach by conducting a performance evaluation study using the Cloud Sim toolkit. The results demonstrate that Cloud computing model has immense potential as it offers significant cost savings and demonstrates high potential for the improvement of energy efficiency under dynamic workload scenarios.

V. Architecture

An overview of our proposed framework is depicted in Figure 1. The framework, running on all mobile nodes, provides services to applications that aim to: (1) store data in mobile cloud reliably such that the energy consumption for retrieving the data is minimized (k-out-of-n data allocation problem); and (2) reliably process the stored data such that energy consumption for processing the data is minimized (k-out-of-n data processing problem). As an example, an application running in a mobile ad-hoc network may generate a large amount of media files and these files must be stored reliably such that they are recoverable even if certain nodes fail. At later time, the application may make queries to files for information such as the number of times an object appears in a set of images. Without loss of generality, we assume a data object is stored once, but will be retrieved or accessed for processing multiple times later.

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)



Fragments; the fragments are distributed to a set of storage nodes. In order to process the data, applications provide functions that take the stored data as inputs. Each function is instantiated as multiple tasks that process the data simultaneously on different nodes. Nodes executing tasks are processor nodes; we call a set of tasks instantiated from one function a job. Client nodes are the nodes requesting data allocation or processing operations. A node can have any combination of roles from: storage node, processor node, or client node, and any node can retrieve data from storage nodes. Topology Discovery and Monitoring, Failure Probability Estimation, Expected Transmission Time (ETT) Computation, k-out-of-n Data Allocation and k-out-of-n Data Processing. When a request for data allocation or processing is received from applications, the Topology Discovery and Monitoring component provides network topology information and failure probabilities of nodes. The failure probability is estimated by the Failure Probability component on each node. Based on the retrieved failure probabilities and network topology, the ETT Computation component computes the ETT matrix, which represents the expected energy consumption for communication between any pair of node. Given the ETT matrix, our framework finds the locations for storing fragments or executing tasks. The k-out-of-n Data Storage component partitions data into n fragments by an erasure code algorithm and stores these fragments in the network such that the energy consumption for retrieving k fragments by any node is minimized. k is the minimal number of fragments required to recover a data.

VI. Modules

6.1 Geology Determination

Geology Determination is executed during the network initialization phase or whenever a significant change of the network geology is detected (as detected by the Topology Monitoring component). During Geology Determination, one delegated node floods a request packet throughout the network. Upon receiving the request packet, nodes reply with their neighbor tables and failure

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

probabilities. Consequently, the delegated node obtains global connectivity information and failure probabilities of all nodes. This topology information can later be queried by any node.

6.2 Calculation of Failure Possibility

We assume a fault model in which faults caused only by node failures and a node is inaccessible and cannot provide any service once it fails. The failure probability of a node estimated at time t is the probability that the node fails by time t þ T, where T is a time interval during which the estimated failure probability is effective. A node estimates its failure probability based on the following events/causes: energy depletion, temporary disconnection from a network (e.g., due to mobility), and application-specific factors. We assume that these events happen independently. Let f be the event that node i fails and let f Bi;f Ci; and f be the events that node i fails due to energy depletion, temporary disconnection from a network, and application-specific factors respectively.

6.3 Calculation of Expected Transmission Time

It is known that a path with minimal hop-count does not necessarily have minimal end-to-end delay because a path with lower hop-count may have noisy links, resulting in higher end-to-end delay. Longer delay implies higher transmission energy. As a result, when distributing data or processing the distributed data, we consider the most energy efficient paths—paths with minimal transmission time. When we say path p is the shortest path from node i to node j, we imply that path p has the lowest transmission time (equivalently, lowest energy consumption) for transmitting a packet from node i to node j. The shortest distance then implies the lowest transmission time.

VII. Conclusion

In this paper, Data storage refinement of Dynamism-Efficient Fault-Tolerant in Mobile Cloud has implemented. It assigns data fragments to nodes such that other nodes retrieve data reliably with minimal energy consumption. It also allows nodes to process distributed data such that the energy consumption for processing the data is minimized. From this proposed system we have also achieved feasibility of the system. Additionally this system performs the fault tolerance and energy efficiency performance in large scale networks.

References

[1] A. Beloglazov, J. Abawajy, and R. Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing,” Future Generation Comput. Syst., vol. 28, no. 5, pp. 755–768, 2012.

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

[2] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Su, "A survey on network codes for distributed storage," Proc. IEEE, vol. 99, no. 3, pp. 476–489, Mar. 2011.

[3] A. Leon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering. Englewood Cliffs, NJ, USA: Prentice Hall, 2008

[4]Anton Beloglazova , Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, 2011

[5]Anton Beloglazov, Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, Anton Beloglazov, 2012.

[6]Byung-Gon Chun, Clone Cloud: Elastic Execution between Mobile Device and Cloud, Byung-Gon Chun

[7] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in Proc. 6th Conf. Comput. Syst., 2011, pp. 301–314.

[8]Cong Shi, Serendipity: Enabling Remote Computing among Intermittently Connected Mobile Devices, 2012

[9] C. A. Chen, M. Won, R. Stoleru, and G. Xie, "Energy-efficient fault-tolerant data storage and processing in dynamic network," in Proc. 14th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2013, pp. 281–286.

[10] C. Chen, M. Won, R. Stoleru, and G. Xie, "Resource allocation for energy efficient k-out-of-n system in mobile ad hoc networks," in Proc. 22nd Int. Conf. Comput.r Commun. Netw., 2013, pp. 1–9.

[11] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocation for high reliability," in Proc. IEEE Int. Conf. Commun., 2010,

[12] D. S. J. D. Couto, "High-throughput routing for multi-hop wireless networks," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., MIT, Cambridge, MA, USA, 2004.

[13] C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura,

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

"Serendipity: Enabling remote computing among intermittently connected mobile devices," in Proc. 13th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2012, pp. 145–154.

[14] D. W. Coit and J. Liu, "System reliability optimization with k-outof-n subsystems," Int. J. Rel., Quality Safety Eng., vol. 7, no. 2, pp. 129–142, 2000. pp. 1–6.

[15] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocation for high reliability," in Proc. IEEE Int. Conf. Commun., 2010, pp. 1–6.

[16] D. Shires, B. Henz, S. Park, and J. Clarke, "Cloudlet seeding: Spatial deployment for high performance tactical clouds," in Proc. World Congr. Comput. Sci., Comput. Eng., Applied Comput., 2012, pp. 1–7.

[17] D. Huang, X. Zhang, M. Kang, and J. Luo, "MobiCloud: Building secure cloud framework for mobile computing andcommunication," in Proc. IEEE 5th Int. Symp. Serv. Oriented Syst. Eng., 2010, pp. 27–34.

[18] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocation for high reliability," in Proc. IEEE Int. Conf. Commun., 2010, pp. 1–6.

[19] D. Neumann, C. Bodenstein, O. F. Rana, and R. Krishnaswamy "STACEE: Enhancing storage clouds using edge devices," in Proc. 1st ACM/IEEE Workshop Autonomic Comput. Economics, 2011, pp. 19–26.

[20] Y. Wen, R. Wolski, and C. Krintz, "Online prediction of battery lifetime for embedded and mobile devices," in Proc. 3rd Int. Conf. Power-Aware Comput. Syst., 2005, pp. 57–72.

[21] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu,

R. Chandra, and P. Bahl, "MAUI: Making smartphones last longer with code offload," in Proc. 8th Int. Conf. Mobile Syst., Appl., Serv., 2010, pp. 49–62.

[22] M. Aguilera, R. Janakiraman, and L. Xu, "Using erasure codes efficiently for storage in a distributed system," in Proc. Int. Conf. Dependable Syst. Netw., 2005, pp. 336–345.

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN 2454-9924 Volume: 1 Issue: 1(2015)

[23] M. Alicherry and T. Lakshman, "Network aware resource allocation in distributed clouds," in Proc. IEEE Conf. Comput. Commun., 2012, pp. 963–971. x.

[24] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," IEEE Pervasive Comput., vol. 8, no. 4, pp. 14–23, Oct.-Dec. 2009

[20] P. Stuedi, I. Mohomed, and D. Terry, "WhereStore: Location- based data storage for mobile devices interacting with the cloud," in Proc. 1st ACM Workshop Mobile Cloud Comput. Serv.: Soc. Netw. Beyond, 2010, pp. 1:1–1:8.

[21] S. Sobti, N. Garg, F. Zheng, J. Lai, Y. Shao, C. Zhang, E. Ziskind, A.

Krishnamurthy, and R. Y. Wang, "Segank: A distributed mobile storage system," in Proc. 3rd USENIX Conf. File Storage Technol., 2004, pp. 239–252.

[22] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in Proc. IEEE Conf. Comput. Commun., 2012, pp. 945–953.

[23]Sooktip, T, System reliability optimization with k-out-of-n subsystems and changing k, 2011