Research Statement

Xiaolong Cui
Department of Computer Science
University of Pittsburgh
xic51@pitt.edu

I. Introduction

Cloud Computing has emerged as an attractive platform for increasingly diverse compute- and data-intensive applications, as it allows for low-entry costs, on demand resource provisioning and reduced cost of IT infrastructure maintenance. Cloud Computing will continue to grow and attract attention from commercial and public markets. Recent studies predict an annual growth rate of 17.7% by 2016, making cloud computing the fastest growing segment in the IT industry.

In its basic form, a cloud computing infrastructure is a large cluster of interconnected servers hosted in a datacenter that delivers on-demand, "pay-as-you-go" services and resources to customers. As the demand for cloud computing accelerates, cloud service providers will need to expand their infrastructure to ensure the expected levels of performance, reliability and cost-effectiveness, resulting in a multifold increase in the number of computing, storage and communication components in their datacenters.

Service level agreement (SLA) is a critical aspect for a sustainable Cloud Computing business. Basically, SLA is a contract between the cloud service provider and consumer that specifies the terms and conditions under which the service is to be provided, including expected response time and reliability. Failure to deliver the service as specified in the SLA not only subjects the cloud service providers to penalties, but also impacts customers' confidence in using Cloud Computing in the future. Unfortunately, the direct implication of expanded datacenters is its increased propensity to failure. While the likelihood of a server failure is very small, the sheer number of computing, storage and communication components that can fail is daunting. In addition, datacenters are fast becoming a major source of global energy consumption, exacerbating the impact of CO_2 emission on the environment. It is reported that energy costs alone account for 23-50% of the Cloud Computing expenses and this mounts up to $30 billion worldwide. Altogether the question of how fault tolerance might impact energy consumption, the profit of Cloud Computing business, and the environment, becomes critical.

Current fault tolerance approaches rely upon either time or hardware redundancy in order to tolerate failure. The first approach, which uses time redundancy, requires the re-execution of the failed task after the failure is detected. Although this can further be optimized by the use of checkpointing, such an approach can result in a significant delay subjecting cloud service providers to penalties, when SLA terms are violated, and high energy costs due to re-execution of failed tasks. The second approach exploits hardware redundancy and executes multiple instances of the same task in parallel to overcome failure and guarantee that at least one task completes without delay. This approach, which has been used extensively to deal with failure in time-critical applications, is currently used in Cloud Computing to provide fault tolerance while hiding the delay of re-execution. This solution, however, requires additional hardware resources and increases the energy consumption for a given service, which in turn might outweigh the profit gained by providing the service. The trade-off between profit and fault-tolerance calls for new frameworks to take into account both SLA requirements and energy consumption in dealing with failures.

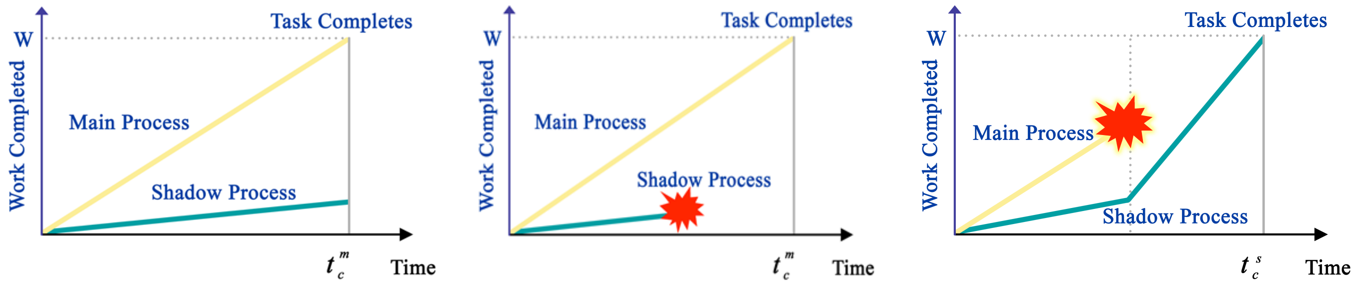
To this end, we propose an energy-aware, SLA-based fault tolerance framework, referred to as Shadow Replication, for profit maximization and energy reduction in Cloud Computing. Similar to the second approach above, Shadow Replication ensures successful task completion by simultaneously running multiple instances. However, Shadow Replication is distinctive in that it differentiates the execution rates of the instances. Specifically, it executes the main instance of the task at the rate required for response time constraint, while slowing down the replicas for energy saving, thereby enabling a parameterized trade-off between response time and energy consumption. This allows cloud service providers to maximize the expected profit by accounting for income, potential penalties, and energy cost, ultimately promoting environment-friendly and sustainable Cloud Computing.

II. Current progress

So far, we have accomplished the following goals:

- A formal definition of the Shadow Replication fault tolerance framework;
- Development of a profit-based analytical model to explore the applicability of Shadow Replication to Cloud Computing, and to determine the optimal execution rates of all task instances to maximize profit as well as to reduce energy consumption;
- Development of an event-driven simulator to verify the above analytical model;
- A comprehensive evaluation using both the analytical model and simulator to analyze the profit and energy savings achievable by Shadow Replication, compared to existing fault tolerance approaches.

We have submitted three papers on this work. [1] is published in Energies 7, no. 8 (2014) and [2] is accepted to CLOSER 2014. Our third paper is currently under review.



(a) No Failure (b) Shadow Process Failure (c) Main Process Failure

Fig. 1: Shadow Replication with a single shadow process.

A. Shadow Replication

The basic tenet of Shadow Replication is to associate with each main instance a suite of "shadows" whose size depends on the criticality of the application and its performance requirements, as defined by the SLA. Each instance is executed with a process. To overcome potential failures, each process executes on a separate computing node.

Formally, we define the Shadow Replication fault tolerance framework as follows:

- A main process, $P_m(W, \sigma_m)$, that executes at the rate of σ_m to complete a task of size W;
- A suite of shadow processes, $P_s(W, \sigma_b^s, \sigma_a^s)$ $(1 \leq s \leq S)$, where S is the size of the suite. The shadow processes execute on separate nodes, and start execution simultaneously with the main process at rate σ_b^s. Upon failure of the main process, all shadows switch their rates to σ_a^s, with one shadow process designated as the new main process. This continues until the completion of the task.

There are several ways to slow down the execution rates of the processes. The one that we are studying is Dynamic Voltage and Frequency Scaling (DVFS), as it is the most straightforward solution. When DVFS is enabled, the CPU frequency (which translates to execution rate) scales proportionally to the voltage. Therefore, decreasing voltage not only reduces power and energy consumption, but also effectively slows down the running processes.

To illustrate the behavior of Shadow Replication, we use only one shadow process and consider the scenarios depicted in Figure 1, assuming at most one process failure. Figure 1(a) represents the case when neither the main nor the shadow fails. The main process, executing at a higher rate, completes the task at time t_c^m. At this time, the shadow process, progressing at a lower rate, stops execution immediately. Figure 1(b) represents the case when the shadow fails. This failure, however, has no impact on the progress of the main process, which can still complete the task at t_c^m. Figure 1(c) depicts the case when the main process fails while the shadow is in progress. After detecting the failure of the main process, the shadow begins executing at a higher rate, completing the task at time t_c^s. Given that the failure rate of an individual node is much lower than the aggregate system failure, it is very likely that the main process will always complete its execution successfully, thereby achieving fault tolerance at a significantly reduced cost of energy consumed by the shadow.

A closer look at the model reveals that Shadow Replication is a generalization of existing fault tolerance approaches, namely re-execution and process replication. If the SLA specification allows for flexible completion time, Shadow Replication would take advantage of the delay laxity to trade time redundancy for energy savings. It is clear, therefore, that for a large response time, Shadow Replication converges to re-execution, as the shadow remains idle during the execution of the main process and only starts execution upon failure. If the target response time is stringent, however, Shadow Replication converges to process replication, as the shadow must execute simultaneously with the main at the same rate. The flexibility of the Shadow Replication provides the basis for the design of a fault tolerance strategy that strikes a balance between task completion time and energy saving, thereby maximizing profit.

B. Analytical model and simulator

One challenge of Shadow Replication resides in determining jointly the execution rates of all processes, both before and after a failure occurs, with the objective to minimize energy and maximize profit. To achieve this, we propose a reward-based analytical model. In the model, we consider the completion time and energy consumption of executing a cloud job, which is composed of multiple parallel tasks, under different system specifics and failure distributions. The profit for cloud service providers is modeled as the difference between the payment from customers, and expenses for running the cloud job. Afterwards, an optimization problem is formulated to derive the optimal execution rates. For more details of the analytical model, please refer to [1].

To verify the correctness of the analytical model, we build an event-driven simulator using CSim that simulates the behaviors of Shadow Replication under different configurations and with different failure distributions, and then report statistics, such as number of failures encountered, time to completion, and energy consumption.

C. Preliminary results

With the analytical model and simulator, we identify several important parameters that impact the energy consumption and profit gains of Shadow Replication, and conduct a series of sensitivity studies correspondingly. The results using both the analytical model and simulation show that Shadow Replication can achieve significant energy savings and profit gains compared to traditional process replication and re-execution, without violating the SLA constraints.

We first study the sensitivity to static power/dynamic power ratio. In this study, we considered modern systems with static power ratio from 40% to 70%. Within this range, Shadow Replication can achieve, on average, 19.3% more profit than traditional replication, and 28.8% more than re-execution. In terms of energy, the saving is 15%-30%. The second study is to the targeted job completion time. The results show that targeted job completion time influences the execution strategies of Shadow Replication to a large extent, and the reason is that Shadow Replication would strive to maintain the completion time constraint. When time is critical, Shadow Replication uses both a main and a shadow from the very beginning, in the same manner as traditional replication, to guarantee that task can be completed on time; when time is not critical, it mimics re-execution and starts its shadow only after a failure. The profit gains by Shadow Replication can be as much as 52.8%. In the next study, we vary the number of tasks from 100 to 10,000,000. On average, Shadow Replication achieves 59.3% and 18.4% more profits than process replication and re-execution, respectively. Our last study is to assess the sensitivity to failure vulnerability, where we find that increasing the failure vulnerability has the same effect as increasing the number of tasks.

To summarize, Shadow Replication can achieve 15%-30% energy savings and 20%-30% more profit on modern systems [2]. Furthermore, Shadow Replication would converge to process replication, when target response time is stringent, and to re-execution when target response time is relaxed or when failure is unlikely.

III. Future directions

Current results reveal that Shadow Computing is promising for significant energy saving while satisfying SLA requirements. The direct benefits include profit gains for cloud service providers, and reduced CO_2 emission that makes Cloud Computing more environment-friendly and more sustainable. Inspired by that, we have several research directions to explore in the future.

The first plan is to further improve the efficiency of Shadow Replication for tightly-coupled jobs. In a tightly-coupled job, the parallel tasks need to synchronize with each other frequently. If one task fails, others have to stop and wait for it to catch up. Shadow Replication is able to reduce the catch-up time, because it runs multiple instances of each task, and if one fails, others can speed up to catch up soon. However, the idle time during the catch-up is still a waste, resulting in a delay in the job completion time. In order to minimize this effect and further improve performance, we plan to explore a new technique, referred to as "Shadow Leaping". The idea is to take advantage of the idle time and align the execution state of the slow shadow processes with their faster main processes to achieve forward progress. Remote Direct Memory Access (RDMA) is a potential way to implement Shadow Leaping.

The next research direction is to evaluate the possibility of using process collocation to implement Shadow Replication for Cloud Computing. Process collocation is an alternative approach to slowing down the process execution rate. By collocating multiple processes on the same computing node, the execution rate of each process is proportional to its time share on the node. For Cloud Computing, process collocation may be a better choice than DVFS as the computing nodes in the cloud datacenters are usually time shared by multiple virtual machines.

The two alternatives are equivalent in terms of completion time, since they have the same effect on the controlling process execution rate. In terms of energy, however, each of them has its advantage. Process collocation requires less hardware resources and this reduce the energy linearly, while DVFS uses more hardwares but can reduce energy superlinearly. It needs further analysis to determine which alternative consumes less overall energy.

The last step is to build a prototype, in order to experimentally evaluate the performance of Shadow Replication using real life applications. This effort is to design and implement a software library to support the main components of Shadow Replication, including process collocation, required consistency protocols, message logging and message forwarding protocols, and execution state update operations in support of Shadow Leaping.

IV. Conclusion

The main motivation of this work stems from the observation that, as Cloud Computing continues to grow, both the failure rate and energy consumption are highly-likely to increase significantly. Hence, understanding the interplay between fault-tolerance, energy consumption and profit maximization is critical for the viability of Cloud Computing. To this end, we propose Shadow Replication as a novel energy-aware, SLA-based fault tolerance framework to minimize energy and maximize profit.

Our preliminary results predict that Shadow Replication is able to achieve significant energy saving and profit gains while satisfying SLA requirements. We will continue to explore and optimize Shadow Replication in support of green and sustainable Cloud Computing.

References

[1] X. Cui, B. Mills, T. Znati, and R. Melhem, "Shadow replication: An energy-aware, fault-tolerant computational model for green cloud computing," *Energies*, vol. 7, no. 8, pp. 5151–5176, 2014. [Online]. Available: http://www.mdpi.com/1996-1073/7/8/5151

[2] X. Cui, B. Mills, R. Melhem, and T. Znati, "Shadows on the cloud: An energy-aware, profit maximizing resilience framework for cloud computing." in *4th International Conference on Cloud Computing and Services Science*, 2014.