Adaptive and Power-aware Resilience for Extreme-scale Systems

Xiaolong Cui
mclarencui@cs.pitt.edu
Advisor: Dr. Taieb Znati and Dr. Rami Melhem

1 Introduction

As our reliance on IT continues to increase, the complexity and urgency of the problems our society will face in the future will increase much faster than are our abilities to understand and deal with them. Future IT systems are likely to exhibit a level of interconnected complexity that makes it prone to failure and exceptional behaviors. The high risk of relying on IT systems that are failure-prone calls for new approaches to enhance their performance and resiliency to failure. My research focuses on efficient and power-aware computational models that tolerate failures for extreme-scale distributed systems, including both HPC supercomputers and Cloud data centers.

Current fault-tolerance approaches are designed to deal with fail-stop errors, and rely exclusively on either time or hardware redundancy for recovery. Rollback recovery, which exploits time redundancy, requires full or partial re-execution when failure occurs. Such an approach can incur a significant delay, and high energy costs due to extended execution time. On the other hand, Process Replication relies on hardware redundancy and executes multiple instances of the same task in parallel to guarantee completion without delay. This solution, however, requires a significant increase in hardware resources and increases the power consumption proportionally.

My proposed approaches to resiliency go beyond adapting or optimizing well known and proven techniques, and explore radical methodologies to fault tolerance that scale to extreme-scale computing infrastructures. The proposed solutions differ in the type of faults they manage, their design, and the fault tolerance protocols they use. It is not just a scale up of "point" solutions, but an exploration of innovative and scalable fault tolerance frameworks. When integrated, it will lead to efficient solutions for a "tunable" resiliency that takes into consideration the nature of the data and the requirements of the application.

2 Methodology

The basic tenet of the proposed fault tolerance framework, referred to as Leaping Shadows, is the concept of shadowing, whereby each main process is guarded by a suite of "shadow" processes. To finish a task, the shadow processes will execute the same code as its associated main process, and the task will be successfully completed as long as one of the processes can finish. The novelty of Leaping Shadows lies in its dynamic and differential adjustment of the execution rates. Specifically, it executes the main processes at the rate required for response time constraint, while slowing down the shadows to reduce hardware and power requirement. Upon failure of a main process, its associated shadow increases its rate to complete the task, thereby reducing the delay to the progress of other tasks.

Leaping Shadows is adaptive in four dimensions. Firstly, the number of shadow processes to use for each main process can be decided to balance the trade-off between the criticality of the application, and hardware and power budget. Secondly, the execution rates of the main and shadow processes are tunable performance determinants. Thirdly, if compute nodes exhibit different "health" status, partial shadowing,

where only processes running on "weak" nodes are shadowed, can be applied to minimize resource usage. Last but not least, instead of slowing down, the shadows may execute reduced code, e.g. with less resolution, to generate less precise results in the case of failure.

2.1 Execution rate control

One challenge in Leaping Shadows is how to control the execution rates. So far, we have explored two methods to differentiate the execution rates. The first approach is to use Dynamic Voltage and Frequency Scaling (DVFS), of which the CPU frequency scales linearly with the supply voltage [1, 2]. The second approach is to use time sharing to reduce the effective execution rate while keeping the compute nodes running at maximal frequency. Since this approach collocates multiple processes on a same node, it simultaneously reduces the number of machines to be used and the power consumption [3].

2.2 Modeling and Simulation

The other challenge resides in determining jointly the execution rates of all processes, with the objective to minimize power consumption while satisfying QoS requirements. To achieve this, I propose two analytical models, corresponding to the two rate-control approaches above. The models consider the time and power needed for a job, under different system specifics and failure distributions. From the analytical models an optimization problem is formulated and solved to derive the optimal execution rates.

To verify the correctness of the analytical models, I build an event-driven simulator that simulates the behaviors of Leaping Shadows under different system specifics, task characteristics, and failure distributions. It can report all necessary statistics, such as number of failures encountered, time to completion, and energy consumption. The statistics can then be used to compare with the results from the analytical model.

2.3 Implementation

I am actively implementing Leaping Shadows as a runtime for Message Passing Interface (MPI), which is the de facto programming paradigm for HPC. Instead of a full-feature MPI implementation, the runtime is designed to be a separate layer between MPI and user application, in order to take advantage of existing MPI performance optimizations that numerous researches have spent years on. The runtime spawns the shadow processes, manages the coordination between main and shadow processes, and guarantees consistency for messages and non-deterministic events.

2.4 Results

Several important factors are identified that impact the performance of Leaping Shadows. Correspondingly, I conduct a series of sensitivity studies where Leaping Shadows is compared to state-of-the-art approaches. The results from both the analytical model and simulator show that Leaping Shadows can achieve significant power/energy savings, without violating the QoS constraints.

3 Conclusion

My current research is enabling new insights into the multi-faceted and challenging resiliency problem in extreme-scale distributed systems. The goal is to investigate radical approaches to the design of scalable and efficient fault tolerant approaches that go beyond state-of-the-art algorithms. Throughout my design, the interplay between resiliency, performance and power consumption is analyzed carefully to determine the required levels of endurance and redundancy, in order to achieve a desired level of both QoS and fault tolerance while minimizing resource usage.

Acknowledgments

The author would like to thank Dr. Bryan Mills for his work in developing the initial analytical model (modeling of a single task) for Shadow Computing with DVFS. For clarification, the author performed all the other work presented in this paper.

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.

[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 *Proceedings of the 4th International Conference on Cloud Computing and Services Science*, ser. CLOSER '14, Barcelona, Spain, 2014.

[3] X. Cui, R. Melhem, and T. Znati, "Adaptive and power-aware resilience for extreme-scale computing." in *Proceedings of the 16th IEEE International Conference on Scalable Computing and Communications*, ser. ScalCom '16, Toulous, France [under review], 2016.