ADAPTIVE AND POWER-AWARE FAULT TOLERANCE FOR FUTURE EXTREME-SCALE COMPUTING

by

Xiaolong Cui

Bachelor of Engineering

Xi'an Jiaotong University

2012

Submitted to the Graduate Faculty of
the Kenneth P. Dietrich School of
Arts and Sciences in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2016

UNIVERSITY OF PITTSBURGH

COMPUTER SCIENCE DEPARTMENT

This proposal was presented

by

Xiaolong Cui

Dr. Taieb Znati, Department of Computer Science, with joint appointment in Telecommunication Program, University of Pittsburgh

Dr. Rami Melhem, Department of Computer Science, University of Pittsburgh

Dr. John Lange, Department of Computer Science, University of Pittsburgh

Dr. Esteban Meneses, School of Computing, Costa Rica Institute of Technology

Dissertation Advisors: Dr. Taieb Znati, Department of Computer Science, with joint appointment in Telecommunication Program, University of Pittsburgh,

Dr. Rami Melhem, Department of Computer Science, University of Pittsburgh

ADAPTIVE AND POWER-AWARE FAULT TOLERANCE FOR FUTURE EXTREME-SCALE COMPUTING

Xiaolong Cui, PhD

University of Pittsburgh, 2016

As the demand for computing power continue to increase, both HPC community and Cloud service provides are building larger computing platforms to take advantage of the power and economies of scale. On the HPC side, several of the most powerful countries are competing for developing the next generation supercomputer–exascale computing machines to accelerate scientific discoveries, big data analytics, etc. On the Cloud side, large IT companies are all expanding large-scale datecenters, for both private usage and public services. However, aside from the benefits, several daunting challenges will appear when it comes to extreme-scale.

This thesis aims at simultaneously solving two major challenges, i.e., power consumption and fault tolerance, for future extreme-scale computing systems. We come up with a novel computational model, referred to as Lazy Shadowing, as a power-aware and scalable approach to achieve high-levels of resilience, through forward progress, in extreme-scale, failure-prone computing environments. Two approaches have been proposed to realize this idea. Accordingly, precise analytical models and optimization framework have been developed to quantify and optimize the improvement in system efficiency and energy savings, respectively.

In this work, I propose to continue the research in three aspects. Firstly, I propose to develop a MPI-based prototype to prove the viability of Lazy Shadowing in real environment. Using the prototype, I will run benchmarks and real applications to measure its performance compared to state-of-the-art approaches. Then, I propose to study the problem of mapping main and shadow processes to physical cores, with the consideration of hardware, architec-

ture, environment, etc. Last but not least, I propose to further explore the potential of Lazy Shadowing and improve its efficiency. Based on the specific system configuration, application characteristics, and QoS requirement, I will study the viability of partial shadowing in three dimensions, i.e., time, space, and workload.

TABLE OF CONTENTS

1.0 INTRODUCTION

As our reliance on IT continues to increase, the complexity and urgency of the problems our society will face in the future drives us to build more powerful and accessible computer systems. Among the different types of computer systems, High Performance Computing (HPC) and Cloud Computing systems are the two most powerful ones. For both of them, the compute power attributes to the massive amount of parallelism, which is supported by the massive amount of CPU cores, memory modules, communication devices, storage components, etc.

Since CPU frequency flatten out in early 2000s, parallelism has become the "golden rule" of boosting performance. In HPC, Terascale performance was achieved in the late 90s with fewer than 10,000 heavyweight single-core processors. A decade later, petascale performance required about ten times processors with multiple cores on each processor. Nowadays, a race is underway to build the world's first exascale machine to accelerate scientific discoveries and breakthroughs. It is projected that an exascale machine will achieve billion-way parallelism by using one million sockets each supporting 1,000 cores.

Similar trend is happening in Cloud Computing. As the demand for Cloud Computing accelerates, cloud service providers will be faced with the need to expand their underlying infrastructure to ensure the expected levels of performance, reliability and cost-effectiveness. As a result, lots of large-scale data centers have been built and are being built by IT companies to exploit the power and econies of scale. For example, Microsoft, Google, Facebook, and Rackspace have hundreds of thousands of web servers in dedicated data centers to support their business.

Unfortunately, several challenging issues come with the increase in system scale. As today's HPC and Cloud Computing systems grow to meet tomorrow's compute power demand, the behavior of the systems will be increasingly difficult to specify, predict and manage. This upward trend, in terms of scale and complexity, has a direct negative effect on the overall system reliability. Even with the expected improvement in the reliability of future computing technology, the rate of system level failures will dramatically increase with the number of components, possibly by several orders of magnitude. At the same time, the rapid growing power consumption, as a result of the increase in system components, is another major concern. At future extreme-scale, failure would become a norm rather than an exception, driving the system to significantly lower efficiency with unprecedented amount of power consumption.

1.1 PROBLEM STATEMENT

The system scale needed to address our future computing needs will come at the cost of increasing complexity, unpredictability, and operating expenses. As we approach future extreme-scale, two of the biggest challenges will be system resilience and power consumption, both being direct consequences of the increase in the number of components.

Regardless of the reliability of individual component, the system level reliability will continue to decrease as the number of components increases. It is projected that the Mean Time Between Failures (MTBF) of future extreme-scale systems will be at the order of hours, meaning that many failures will occur every day. Without an efficient fault tolerance mechenism, faults will be so frequent that the applications running on the systems will be continuously interrupted, requiring the execution to be restarted every time there is a failure.

Also thanks to the continuous growth in system components, there has been a steady rise in power consumption in large-scale distributed systems. In 2005, the peak power consumption of supercomputers was 3.2 Megawatts. This number was doubled 5 years later, and reached 17.8 Megawatts with a total of 3,120,000 cores in 2013. Recognizing this rapid upward trend, the U.S. Department of Energy has set 20 megawatts as the power limit for future exascale systems, challenging the research community to provide a 1000x improvement in performance with only a 10x increase in power. This huge imbalance makes system power a leading design constraint on the path to exascale.

Today, two approaches exist for fault tolerance. The first approach is rollback recovery, which rolls back and restarts the execution every time there is a failure. This approach is often equiped with checkpointing to periodically save the execution state to a stable storage so that execution can be restarted from a recent checkpoint in the case of a failure. Although checkpointing is the most widely used technique in today's HPC systems, it is strongly believed that it may not scale to future extreme-scale systems. Given the anticipated increase in system level failure rates and the time to checkpoint large-scale compute-intensive and data-intensive applications, it is predicted that the time required to periodically checkpoint an application and restart its execution will approach the systems MTBF. Consequently, applications will make little forward progress, thereby reducing considerably the overall system efficiency.

The second approach, referred to as process replication, exploits hardware redundancy and executes multiple instances of the same task in parallel to overcome failure and guarantee that at least one task reaches completion. Although this approach is extensively used to deal with failures in Cloud Computing and mission critical systems, it has never been used in any HPC system due to its low system efficiency. To replicate each process, process replication requires at least double the amount of compute nodes, which also increases the power consumption proportionally.

Previous studies show that neither of the two approaches is efficient for future extreme-scale systems. And unfortunately, neither of them addresses the power cap issue. Achieving high resilience to failures under strict power constraints is a daunting and critical challenge that requires new computational models with scalability, adaptability, and power-awareness.

1.2 RESEARCH OVERVIEW

There is a delicate interplay between fault tolerance and power consumption. Checkpointing and process replication require additional power to achieve fault tolerance. Conversely, it has been shown that lowering supply voltages, a commonly used technique to conserve power, increases the probability of transient faults. The trade-off between fault free operation and optimal power consumption has been explored in the literature. Limited insights have emerged, however, with respect to how adherence to applications desired QoS requirements affects and is affected by the fault tolerance and power consumption dichotomy. In addition, abrupt and unpredictable changes in system behavior may lead to unexpected fluctuations in performance, which can be detrimental to applications QoS requirements. The inherent instability of extreme-scale computing systems, in terms of the envisioned high-rate and diversity of faults, together with the demanding power constraints under which these systems will be designed to operate, calls for a reconsideration of the fault tolerance problem.

In this thesis, our research objective is to simultaneously address the power and resilience challenges for future extreme-scale systems so that both system efficiency and application QoS are guaranteed. To this end, we propose an adaptive and power-aware computational model, referred to as Lazy Shadowing, as an efficient and scalable alternative to achieve high-levels of resilience, through forward progress, in extreme-scale, failure-prone computing environments.

Previously, we have formally defined the computational model, studied possible techniques to realize and optimize the idea, and built analytical models for performance evaluation. Next, I propose to continue the study of Lazy Shadowing in two aspects.

1.2.1 Achieve Fast Rows via Reorganization (Completed)

1.2.2 Refresh-aware Partial Restore (Completed)

1.2.3 Explore Restoring in Extended Scenarios (Future)

1.3 CONTRIBUTIONS

This thesis makes the following contributions:

- We perform pioneering study on DRAM restoring in deep sub-micron scaling. We built models to simulate restoring behaviors and then generate DRAM devices to faithfully repeat the manufacturing process and perform architectural-level studies.

- Targeting at restoring issues, we propose schemes from different perspectives. On device and architectural levels, we apply chunk remapping and chip clustering techniques to achieve fast memory access; on system level, we maximizing performance improvement by allocating hot pages of the running workloads to fast regions.
- Going further, we integrate restoring variation characteristics with approximate computing to strike a good balance among performance, energy and accuracy. We then explore restoring issues in extended scenarios including information leakage and 3D-stacked memory.

1.4 OUTLINE

The rest of this proposal is organized as follow: Chapter **??** introduces the DRAM structures, operations and scaling issues. In Chapter **??**, we build models to study restoring effects, and then propose a series of techniques to shorten restoring timing values. In Chapter **??**, we explore the correlation between restoring and refresh, and seek the opportunities to early terminate restore operations. Further restoring explorations are discussed in Chapter **??**. Chapter **??** and **??** lists the timeline and concludes the proposal, respectively.

BIBLIOGRAPHY