Preface

Objective

The main objective of this monograph is to provide an overview of *Fault-Tolerance Techniques for High-Performance Computing* (HPC). Resilience has already become a prominent issue on current large-scale platforms. The advent of exascale computers with millions of cores and billion-parallelism is only going to worsen the scenario. The capacity to deal with errors and faults will be a critical factor for HPC applications to be deployed efficiently.

While there are many research papers available on this hot and important topic, there was no comprehensive and easy-to-access reference available in the literature. The purpose of this monograph is to fill the gap, and to provide a detailed presentation and analysis of the various fault tolerance methods for HPC applications.

The first part of the book is made of a single survey chapter that introduces checkpoint protocols and scheduling algorithms, prediction, replication, silent error detection, and correction, together with some application-specific techniques such as Algorithm-Based Fault Tolerance (ABFT). A key feature of this survey chapter is the importance given to analytical performance models. As future extreme-scale platforms are not yet available (by definition!), a refined (and publicly available) performance model is the key to assess any resilience technique without bias nor a-priori. Various scenarios can be instantiated through selecting one's preferred model parameters, and further explored through simulations. The emphasis given to performance models explains the unusual amount of mathematical equations in the chapter, but let the reader be comforted: (i) every method is first described informally; (ii) the mathematical derivations are detailed and complemented with examples; and (iii) it is always possible to skip some proof and be back to it during a second reading.

The second part of the book is composed of four chapters, each dedicated to further investigating one topic. Chapter 2 surveys the various sources for error and faults in real large-scale systems, details their characteristics, and focuses on detection and prediction. Chapter 3 presents the spectrum of techniques that can be

applied to design a fault-tolerant MPI, i.e., to enable MPI application recovery using either fully automatic or completely user-driven techniques. Chapter 4 investigates replication (coupled with checkpointing) and compares two approaches. In the first approach, entire application instances are replicated, while in the second one, each process in a single application instance is (transparently) replicated. Finally, Chap. 5 addresses the challenge of energy consumption related to fault tolerance in extreme-scale systems, and proposes a methodology to estimate the energy consumption of fault-tolerant protocols used in HPC.

The best way to read the book is to start with the overview chapter in Part I, and then to move on to the more specialized chapters of Part II. However, experienced readers may want to read a single specific chapter in Part II. To ease this approach, we have made each chapter independent of the others, at the price of some redundant information throughout the book. Cross-references between related sections of different chapters and index terms have been provided to help navigate across chapters whenever needed.

Thanks

This monograph is the follow-up of a tutorial that we gave at ICS'13. We were approached by Springer Verlag and invited to write this monograph, a task that we eventually succeeded to complete after … some delay.

We would like to thank all chapter authors for their contribution. All of them are colleagues and Ph.D. students with whom we worked on various topics during the past few years, and all of us share many ideas on resilience for HPC. Hopefully, the reader will sense a common perspective while reading the monograph!

The tutorial that George Bosilca, Aurélien Bouteiller, and the two of us gave at SC'14 came one year after the one given at ISC'13. At that point the monograph was far from ready, and intense discussions when preparing the SC'14 tutorial have greatly influenced the overview chapter. We thank them for this.

Finally, we would like to thank Jack Dongarra and ICL for providing a unique place to collaborate and work on HPC-related research, from linear algebra to resilience and more.

Knoxville — Thomas Herault
Lyon — Yves Robert
April 2015

http://www.springer.com/978-3-319-20942-5

Fault-Tolerance Techniques for High-Performance Computing

Herault, Th.; Robert, Y. (Eds.)
2015, IX, 320 p. 113 illus., Hardcover
ISBN: 978-3-319-20942-5