Thesis Overview:

SEDAR: Soft Error Detection and Automatic Recovery in High Performance Computing Systems

Diego Miguel Montezanti
III-LIDI, Universidad Nacional de La Plata, Argentina
PhD in Computer Science
Advisors: Dolores Rexachs –Armando De Giusti
Co-advisors: Emilio Luque – Marcelo Naiouf
{dmontezanti,degiusti,mnaiouf}@lidi.info.unlp.edu.ar
{dolores.rexachs,Emilio.luque}@uab.es

Motivation

Reliability and fault tolerance have become aspects of growing relevance in the field of HPC, due to the increased probability that faults of different kinds will occur in these systems. This is fundamentally due to the increasing complexity of the processors, in the search to improve performance, which leads to a rise in the scale of integration and in the number of components that work near their technological limits, being increasingly prone to failures. Another factor that affects is the growth in the size of parallel systems to obtain greater computational power, in terms of number of cores and processing nodes.

As applications demand longer uninterrupted computation times, the impact of faults grows, due to the cost of relaunching an execution that was aborted due to the occurrence of a fault or concluded with erroneous results. Consequently, it is necessary to run these applications on highly available and reliable systems, requiring strategies capable of providing detection, protection and recovery against faults.

In the next years it is planned to reach Exa-scale, in which there will be supercomputers with millions of processing cores, capable of performing on the order of 1018 operations per second. This is a great window of opportunity for HPC applications, but it also increases the risk that they will not complete their executions. Recent studies show that, as systems continue to include more processors, the Mean Time Between Errors decreases, resulting in higher failure rates and increased risk of corrupted results; large parallel applications are expected to deal with errors that occur every few minutes, requiring external help to progress efficiently. Silent Data Corruptions are the most dangerous errors that can occur, since they can generate incorrect results in programs that appear to execute correctly. Scientific applications and large-scale simulations are the most affected, making silent error handling the main challenge towards resilience in HPC. In message passing applications, a silent error, affecting a single task, can produce a pattern of corruption that spreads to all communicating processes; in the worst case scenario, the erroneous final results cannot be detected at the end of the execution and will be taken as correct.

Since scientific applications have execution times of the order of hours or even days, it is essential to find strategies that allow applications to reach correct solutions in a bounded time, despite the underlying failures. These strategies also prevent energy consumption from skyrocketing, since if they are not used, the executions should be launched again from the beginning. However, the most popular parallel programming models used in supercomputers lack support for fault tolerance.

Objectives

In the context of high error rates, unreliable results and high verification costs, the aim of this thesis is to help scientists and programmers of parallel applications to provide reliability to their results, within a predictable time.

To accomplish this goal, we have designed and developed the SEDAR (Soft Error Detection and Automatic Recovery) methodology, which provides tolerance to transient faults in systems consisting in message passing applications that run in multicore clusters. SEDAR is based on process replication and monitoring of messages to be sent and of local computation, taking advantage of the intrinsic hardware redundancy of the multicores.

SEDAR provides three variants: detection and automatic relaunch from the beginning; automatic recovery, based on the storage of multiple system-level checkpoints (periodic or synchronized with events); and automatic recovery, based on a single safe application-level checkpoint. The main goal is the design of the methodology and the functional validation of its effectiveness to detect transient faults and automatically recover executions, using an analytical verification model; a SEDAR prototype is also implemented. From the tests carried out with this prototype, the temporal behavior is characterized, i.e. the overhead introduced by each variant. The flexibility to dynamically choose the most convenient alternative to adapt to system requirements (such as maximum allowed overhead or completion time) is also evidenced, showing that SEDAR is a viable and effective methodology to tolerate transient faults in HPC. Unlike specific strategies, which provide partial resilience for certain applications, at the cost of modifying them, SEDAR is essentially transparent and agnostic regarding the protected algorithm.

Contributions

The main contributions of this thesis are:

- The development of a functionally valid fault tolerance methodology that integrates duplication (for detection) with the checkpoint & restart that is used to guarantee recovery from permanent failures, thus obtaining a strategy that ensures both completion and reliability of the results.
- The description and verification of the functional behavior, using a model that considers all the possible failure scenarios, demonstrating the effectiveness of detection and the recovery mechanism based on multiple system-level checkpoints.
- The empirical verification of the model's predictions, through controlled fault-injection experiments.
- The implementation of prototype of an automatic tool that is capable of recovering without user intervention, which integrates the detection mechanism with the recovery mechanism based on multiple checkpoints.
- The detail of the experimental work carried out to attach SEDAR to parallel applications.
- The determination of the amount of necessary resources, together with the temporal characterization and the evaluation of the overheads of each of the three alternatives. Thus the benefits obtained both in runtime and in reliability are shown
- The evidence of SEDAR's flexibility to adapt to a particular compromise between the cost and the obtained performance.

Conclusions

In this Thesis, we have developed an analytical model that allows us to validate the fault tolerance strategy, and procedure to implement it. Therefore, we have arrived to a complete methodology, which itself contemplates all possible fault scenarios. On the other hand, experiments have shown us that standard software tools (i.e. existing well-known technologies) can be integrated to reach an implementation of our proposal. We have arrived at a prototype of an automatic system that provides a certain quality of service, being able to store checkpoints and recover without user intervention, thus guaranteeing the reliability of the results within a time that can be limited. Therefore SEDAR is a feasible and effective methodology to tolerate transient faults in HPC.

Future Lines of Work

- Extending experimental validation, using the recovery algorithm based on application-level checkpoints.
- Calculating the optimal checkpoint interval, in order to minimize both the execution overhead and the rework to be done, quantifying the relationship between detection latency and the communications pattern.
- Optimally support the occurrence of multiple non-related faults, with the recovery strategy based on multiple checkpoints, and predict the temporal response.
- Implementing a dynamic adaptation of the recovery mechanism, and auxiliary tools to provide reports and statistics to the user for subsequent analysis.

- Integrating SEDAR with architectures that tolerate permanent faults, to support both types of faults with a single functional tool for Exa-scale, considering the impact of energy consumption on resilience.

Published Research Related with the Thesis

- Montezanti, D., Frati, F. E., Rexachs, D., Luque, E., Naiouf, M., & De Giusti, A. (2012). Smcv: a methodology for detecting transient faults in multicore clusters. CLEI Electronic Journal, 15(3), 5-5.
- Montezanti, D., Rucci, E., del Rosario, D. I. R., Luque, E., Naiouf, M., & De Giusti, A. (2014). A tool for detecting transient faults in execution of parallel scientific applications on multicore clusters. Journal of Computer Science and Technology, 14(1), 32-38.
- Characterizing a detection strategy for transient faults in HPC, Computer Science Technology Series. XXI Argentine Congress of Computer Science. Selected papers, 77-90. EDULP, 2016.
- Montezanti, D., De Giusti, A., Naiouf, M., Villamayor, J., Rexachs, D., & Luque, E. (2017, July). A methodology for soft errors detection and automatic recovery. In 2017 International Conference on High Performance Computing & Simulation (HPCS) (pp. 434-441). IEEE.
- Montezanti, D. M., Rucci, E., Rexachs del Rosario, D., Luque Fadón, E., Naiouf, M., & De Giusti, A. E. (2019). SEDAR: Detectando y recuperando fallos transitorios en aplicaciones de HPC. In XXV Congreso Argentino de Ciencias de la Computación (CACIC)(Universidad Nacional de Río Cuarto, Córdoba, 14 al 18 de octubre de 2019).
- Montezanti, D., Rucci, E., De Giusti, A., Naiouf, M., Rexachs, D., & Luque, E. (2020). Soft errors detection and automatic recovery based on replication combined with different levels of checkpointing. Future Generation Computer Systems, 113, 240-254.

Citation: D.M. Montezanti. *SEDAR: Soft Error Detection and Automatic Recovery in High Performance Computing Systems.* Journal of Computer Science & Technology, vol. 20, no. 2, pp. 119-121, 2020.
DOI: 10.24215/16666038.20.e14