

Recoverable Computing

Panagiota Fatourou ✉

Institute of Computer Science, Foundation for Research and Technology-Hellas, Heraklion, Greece
Department of Computer Science, University of Crete, Heraklion, Greece

Abstract

Non-Volatile Memory (NVM) is an emerging memory technology which aims to address the high computational demands of modern applications and support recovery from crashes. Recovery ensures that after a crash every executed operation is able to recover and return a correct response. This talk will shed light on different aspects of the question “*How does concurrent computing change in systems with NVM and what will the impact of persistent memory be on the way we compute?*”. Specifically, this talk addresses the following four main challenges in NVM computing.

- **Challenge 1:** *How to appropriately model and abstract fundamental aspects of NVM computing?* The talk will provide an overview of the theoretical framework for NVM computing, including a discussion of correctness conditions, progress guarantees, failure types, etc.
- **Challenge 2:** *How to compute in a recoverable way at no significant cost?* The talk will summarize state-of-the-art generic approaches for deriving recoverable synchronization algorithms, as well as recoverable implementations of many widely-used concurrent data structures on top of them. The collection of data structures includes fundamental structures, such as stacks and queues, but also more complex structures that implement sets, such as linked-lists and trees.
- **Challenge 3:** *How to analyze the cost of recoverable algorithms?* The talk will present a way of analyzing the cost of persistence instructions, not by simply counting them but by separating them into categories based on the impact they have on the performance. This analysis reveals that understanding the actual persistence cost of an algorithm in machines with NVM, is more complicated than previously thought, and requires a thorough evaluation, since the performance impact of different persistence instructions may greatly vary.
- **Challenge 4:** *When is Recoverable Consensus Harder Than Consensus?* The talk will briefly discuss the ability of different shared object types to solve recoverable consensus using NVM when processes crash and recover, and it will compare the difficulty of solving recoverable consensus to the difficulty of solving the standard consensus problem in a system with halting failures.

For each of the above challenges, the talk will present main results, provide some of the details of the best-performing techniques, and discuss open problems and directions for further research. Some of the results that will be discussed in detail have appeared in [1, 2, 3].

2012 ACM Subject Classification Theory of computation → Distributed computing models; Theory of computation → Concurrent algorithms; Theory of computation → Data structures design and analysis

Keywords and phrases non-volatile memory, persistence, detectability, durability, recoverable algorithms, recoverable data structures, persistent objects, stacks, queues, heaps, synchronization, universal constructions, software combining, lock-freedom, wait-freedom, persistence cost analysis

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2022.2

Category Invited Talk

Supplementary Material *Text (Slides of the Talk):* <https://sites.uclouvain.be/OPODIS2022/>



© Panagiota Fatourou;
licensed under Creative Commons License CC-BY 4.0

26th International Conference on Principles of Distributed Systems (OPODIS 2022).

Editors: Eshcar Hillel, Roberto Palmieri, and Etienne Rivière; Article No. 2; pp. 2:1–2:2



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

References

- 1 H. Attiya, O. Ben-Baruch, P. Fatourou, D. Hendler, and E. Kosmas. Detectable recovery of lock-free data structures. In *Proc. of the 2022 PPOPP*, pages 262–277, 2022.
- 2 C. Delporte-Gallet, P. Fatourou, H. Fauconnier, and E. Ruppert. When is recoverable consensus harder than consensus? In *Proc. of the 2022 PODC*, pages 198–208, 2022.
- 3 P. Fatourou, N. D. Kallimanis, and E. Kosmas. The performance power of software combining in persistence. In *Proc. of the 2022 PPOPP*, pages 337–352, 2022 (Best Paper Award).