

## On the restricted use of concurrent objects

**Key words:** distributed algorithms, concurrent computing, multiprocessors shared memory, shared objects, fault-tolerance, complexity

**Direction:** Colette Johnen et Alessia Milani (milani@labri.fr)

**Laboratory:** LaBRI

**Team:** Combinatorial & Algorithm, distributed algorithms

### Abstract of thesis topic :

Concurrent objects play a key role in the design of concurrent applications. The current revolution of multi-core architectures demands to gain better understanding of the complexity of concurrent objects implementations. We are interested in fault-tolerant implementations, which allow a process that does not prematurely halt to complete its operations despite the behaviour of the other processes in the system. These implementations are called wait-free.

It is known that in an asynchronous read-write shared memory system several fundamental concurrent objects cannot be implemented in a deterministic and fault-tolerant way (e.g. consensus). The difficulty stems from handling executions where processes concurrently operate on the object.

One way to circumvent this impossibility is to only guarantee progress (using reads and writes) in the absence of contention. These implementations are called obstruction-free. Alternatively, a process is guaranteed to complete, but stronger (and more expensive) synchronization primitives can be applied in the presence of contention.

Today, several bounds have been proved on the complexity of obstruction-free implementations for concurrent objects (e.g.[1,2,3]). On the other hand, the understanding of the complexity of concurrent objects implementations that adapt to contention is still premature (e.g.[4]).

The aim of this thesis is to study the space and time complexity of wait-free implementations that adapt to contention. This study has the intention to understand if the difficulty of fault-tolerant concurrent objects implementations is due to the detection of contention or to the fact of managing it.

[1] L. Zhu. A Tight Space Bound for Consensus. STOC 2016.

[2] H. Attiya, R. Guerraoui, D. Hendler, and P. Kuznetsov. The complexity of obstruction-free implementations. J. ACM, 56(4), 2009.

[3] J. Aspnes, H. Attiya, K. Censor-Hillel and D. Hendler. Lower Bounds for Restricted-Use Objects. SPAA 2012.

[4] V. Luchangco, M. Moir, and N. Shavit. On the uncontended complexity of consensus. Distributed Computing, 2003.

[5] C. Capdevielle, C. Johnen, P. Kuznetsov and, A. Milani. On the Uncontended Complexity of Anonymous Agreement. OPODIS 2015.