Models of Distributed Algorithms

Emulation of the distance-k model under central daemon into the asynchronous distance-1 model

Karine Altisen Stéphane Devismes

Verimag Lab

Contacts: {Karine.Altisen, Stephane.Devismes}@univ-grenoble.fr

Keywords: distributed algorithms, self-stabilization, model translation, locally shared memory model.

1 Model Translation in Distributed Computing

Theoretical researches on distributed computing mainly consist of the design, proof, and complexity analysis of distributed algorithms. In this context, modeling is a crucial issue. High-level models allow to simplify the proofs, but make the algorithmic design far from real deployment. Conversely, low-level models are closer to the reality, but design and proofs in those models are increasingly more complex. An interesting line of research is then the design of general *translation methods* from an high-level model to a lower one.

2 Self-Stabilization

Self-stabilization [Dij74] is a versatile technique to withstand transient faults in a distributed system. Indeed, a distributed algorithm is said to be self-stabilizing if, after any finite sequence of transient faults, it will eventually recover, without external (*e.g.*, human) intervention, provided that those faults did not corrupt its code. In particular, a self-stabilizing algorithm, starting from an arbitrary configuration, eventually behaves correctly.

3 Locally Shared Memory Model

The *locally shared memory model (with composite atomicity)* is an abstraction of the message-passing model introduced by Dijkstra [Dij74] which is commonly used in self-stabilization. In this model, each process is able to directly read the states of all its neighbors, *i.e.*, each process can read at distance 1.

Several methods preserving self-stabilization, *e.g.*, [KP93, AKM$^+$07, DDT06], have been proposed to translate algorithms written in this model into the message-passing model, which is the lowest level model.

Reading at distance 2. Now, even in the (high-level) locally shared memory model, proofs may be complex and subtle. Hence, some higher level models have been studied. For example, Gairing *et al* [GGH$^+$04] proposed a stronger version of locally shared memory model, where (1) processes can read at distance two and (2) no two processes can change their state simultaneously. In particular, they designed a method (preserving self-stabilization) to translate an algorithm written in that distance-2 model into the (fully) asynchronous distance-1 model.

Reading at distance k. We propose here to generalize this latter result to the distance-k model, with $k \geq 2$. Hence, the expected contribution is a method to translate self-stabilizing algorithms written in the sequential distance-k model to algorithms that are self-stabilizing (for the same specification) into the (fully) asynchronous distance-1 model.

4 Work To Be Done

Precisely, several contributions are expected:

- The design of the translation technique,
- its proof of correctness,
- the evaluation of the overhead (both in space and time),
- and the application of the method to an interesting case-study (optional).

5 Working context.

The internship is part of ANR project ESTATE[1]. The student will be integrated in the lab Verimag[2].

Possible extension into a PhD thesis.

2019-2020

References

[AKM$^+$07] Baruch Awerbuch, Shay Kutten, Yishay Mansour, Boaz Patt-Shamir, and George Varghese. A time-optimal self-stabilizing synchronizer using a phase clock. *IEEE Trans. Dependable Sec. Comput.*, 4(3):180–190, 2007.

[DDT06] Sylvie Delaët, Bertrand Ducourthial, and Sébastien Tixeuil. Self-stabilization with r-operators revisited. *Journal of Aerospace Computing, Information, and Communication (JACIC)*, 3(10):498–514, 2006.

[Dij74] Edsger W. Dijkstra. Self-stabilizing systems in spite of distributed control. *Commun. ACM*, 17(11):643–644, 1974.

[GGH$^+$04] Martin Gairing, Wayne Goddard, Stephen T. Hedetniemi, Petter Kristiansen, and Alice A. McRae. Distance-two information in self-stabilizing algorithms. *Parallel Processing Letters*, 14(3-4):387–398, 2004.

[KP93] Shmuel Katz and Kenneth J. Perry. Self-stabilizing extensions for message-passing systems. *Distributed Computing*, 7(1):17–26, 1993.

[1] https://wp-systeme.lip6.fr/estate/
[2] http://www-verimag.imag.fr/