Distributed Computing Theory: Algorithms, Impossibility Results, Models, and Proofs

Knuth Prize Lecture
Symposium on Theory of Computing
June, 2007

Nancy Lynch
Massachusetts Institute of Technology
CSAIL, 32 Vassar Street
Cambridge, MA, USA
lynch@theory.csail.mit.edu

ABSTRACT

In this talk, I will recount my history of working in the area of Distributed Computing Theory, starting nearly 30 years ago. I will describe the main contributions my collaborators and I have made, with some history and perspective. These contributions include a few algorithms, many impossibility results, a strong dose of modeling and proof methods, and a variety of applications of the theory.

I will highlight one specific technical result—the well-known "Fischer, Lynch, Paterson (FLP)" result that asserts impossibility of reaching consensus in a distributed system, in the presence of failures. I will explain what it says, why it is true, and what its significance has been.

Because distributed algorithms must contend with many subtleties that do not arise in simpler settings, modeling and proof methods have come to be an important part of Distributed Computing Theory. I will describe the modeling frameworks that my collaborators and I have developed to support work on distributed algorithms. These are mainly interacting state-machine models like I/O Automata, Timed I/O Automata, and Hybrid and Probabilistic I/O Automata.

One thing led to another: Having good modeling and proof methods for distributed algorithms, we found ourselves applying them to practical case studies, in diverse areas including Internet communication protocols, mobile wireless network protocols, controlled robotics and vehicle systems, security protocols, and even cellular-level biological systems.

Finally, I will describe some new directions that my research group is currently pursuing. These include new interacting state-machine models that combine timed, hybrid, and probabilistic features in one framework, as well as a formal language (Tempo) and computer-supported tools for describing and analyzing systems modeled as Timed I/O Automata. And we are still working on distributed algorithms and impossibility results, but now we are focusing on new kinds of distributed network settings: highly dynamic networks such as mobile ad hoc networks. I believe that much interesting research remains to be done to develop a new Distributed Computing Theory for these new settings.

Categories and Subject Descriptors

F.1 [**Computation by Abstract Devices**]; F.1.1 [**Models of Computation**]; F.2 [**Analysis of Algorithms and Problem Complexity**]; F.3 [**Logics and Meanings of Programs**]; F.3.1 [**Specifying and Verifying and Reasoning About Programs**]

General Terms

Theory

Keywords

Distributed Computing Theory

STOC'07, June 11–13, 2007, San Diego, California, USA.
ACM 978-1-59593-631-8/07/0006.