

Cooperative Computing in Dynamic Environments

MIT9904-12

Progress Report: July 1, 2000-December 31, 2000

Nancy Lynch and Idit Keidar

Project Overview

We are working on developing models and analysis methods for distributed systems, with a focus on cooperative group activities in networks. Such group activities range from human social activities in cyber communities to powerful distributed applications involving data sharing and cooperative work. These activities are often supported by agent communication services, which provide distributed intelligence, or by group communication services, which manage group membership and guarantee coherent communication. The environments in which such activities take place are highly dynamic: participants come and go (and change location), network topology changes, and components fail and recover. Coping with such difficult environments leads to complex implementations, which are difficult to build, understand, and analyze.

This project addresses these problems using formal modeling and verification techniques, in particular, a combination of Input/Output automaton methods used at MIT and process algebraic and knowledge-based methods used at NTT. This involves extensions to the existing techniques, for example, extending I/O automata to allow dynamic process creation and destruction. As the basic framework is developed, it is being applied to a collection of typical examples from cooperative computing applications, including computer-supported cooperative work, e-commerce, and distributed databases. Other issues being studied include analysis of performance and fault-tolerance properties, and connecting the formal models with actual runnable code.

Progress Through December 2000

Group Communication

In the past six months we have continued our efforts in the area of group communication systems, focusing on group communication systems for Wide Area Networks (WANs) and on performance evaluation. We have published in three new conferences papers related to this research area, and have finalized journal versions of three others.

We have completed running extensive experiments with the membership service of [KSMD00] over the Internet. After undergoing performance tuning, the service was run continuously for a couple of weeks at several locations around the world, spanning the US, Israel, and Taiwan. The results of this experiment show that the algorithm performs extremely well over the Internet, and that the assumptions made in the service design hold. We have finalized a journal version of the paper [KSMD00] including the new performance results.

We have finalized journal versions of two additional papers:

- 1) [KK00], where we describe a novel design for a novel Virtually Synchronous group communication service targeted for WANs; and
- 2) [KKLS00], where we present a novel inheritance-based modeling and verification technique. This technique was used in the verification of the group communication service of [KK00].

In [FK01] we present a framework for building highly available services. The framework uses group communication to coordinate a collection of servers. Our framework is configurable, in that one can adjust parameters such as the number of servers and the extent to which they are synchronized. We analyze the scenarios that can lead to the service availability being temporarily compromised, and we discuss the tradeoffs that govern the choice of parameters.

We have published a new paper [BKAL00] presenting an algorithm for totally ordered multicast which preserves Quality of Service (QoS) guarantees. The paper assumes a QoS reservation model in which the network allows for reservation of variable bandwidth, specified by the average transmission rate and the maximum burst. As long as the application sends at the reserved rate, the network guarantees to deliver messages with bounded delays. For this model, the paper presents a totally ordered multicast algorithm that preserves the bandwidth and latency reserved by the application within certain additive constants that do not depend on the number of processes participating. This is an improvement over previous work, which gave latency bounds proportional to the number of processes. Furthermore, the presented algorithm allows for dynamic joining and leaving of processes while still preserving the QoS guarantees.

We have also published a new availability study [IK01] of several different dynamic voting algorithms. The paper uses simulations to evaluate the effect of interruptions on the availability of dynamic voting algorithms. We study four dynamic voting algorithms, and identify two important characteristics that impact an algorithm's availability in runs with frequent connectivity changes. First, we show that the number of processes that need to be present in order to avoid blocking impacts the availability, especially during long runs with numerous connectivity changes. Second, more surprisingly, we show that the number of communication rounds exchanged in an algorithm plays a significant role in the availability achieved, especially in the degradation of availability as connectivity changes become more frequent. These two factors were ignored by previous analyses of dynamic voting algorithms.

Dynamic Systems

Attie and Lynch revised and simplified the Dynamic I/O Automaton (DIOA) model considerably. A new classification for action signatures was added, the transition semantics was redone using this new classification, and the theorems and proofs were generalized and simplified. Also, "clone" automata were excluded from the model, and the numbers of different kinds of actions were reduced. A new conference paper on the new model was submitted [Attie-Lynch]. A full technical report version is nearly completed [Attie-Lynch-tr].

In July, 2000, Attie and Lynch visited NTT to work with Kogure, Mano, Araragi, and Kawabe. They worked on the DIOA model and its application to agent-based computing; on defining the semantics of NePi2 in terms of IOA and DIOA; on scheduling algorithms for NePi2; and on general issues of modeling and verifying distributed algorithms and how these apply to the Erdos language.

Plans for the Next Six Months

In the next six months we intend to continue our efforts on performance evaluation of group communication services and other algorithms.

We intend to carefully analyze the performance of algorithms that we have previously developed, e.g., the algorithm of [KK00].

We are continuing our efforts on developing services that preserve QoS guarantees. We are working to develop a fault-tolerant algorithm for an *atomic broadcast service* with a *dynamic* set of participants; that is, reliable totally ordered multicast for dynamic groups. The algorithm will preserve QoS guarantees. We are conducting a detailed theoretical study of the QoS guarantees of the developed algorithm under different circumstances. In particular, we would like to be able to show that in periods with no failures, the latency for the ordered multicast is within a constant of the latency of the underlying network (independently of the number of participants). This would be an improvement over the latency exhibited by previous algorithms. When failures do occur, we strive to keep the latency linear in the number of processes that fail within a bounded time interval, as dictated by a lower bound. Unlike most group communication systems providing similar services, we are designing our algorithm to allow processes to join and leave without introducing delays in the communication between active participants. A major challenge is avoiding communication delays for active processes when joins occur, while at the same time preserving consistency if failures occur near the time of a join.

In the next six months, we plan to finish work on the basic DIOA model and complete the tech report. Then we will consider how to use DIOA to build a range of structures, for example, (a) typical structures for agent computing (like locations and mobility), and (b) typical structures for object-oriented programming. A key issue in the latter work is how to model typical data-sharing relationships among the dynamically created objects.

We will subsequently also investigate methods for proving liveness and timing properties of dynamic systems expressed in DIOA.

References

[KK00] Idit Keidar and Roger Khazan. A Client-Server Approach to Virtually Synchronous Group Multicast: Specifications and Algorithms. 20th International Conference on Distributed Computing Systems (ICDCS), pages 344—355, April 2000. MIT Lab. for Computer Science Tech. Report MIT-LCS-TR-794.

[KSMD00] I. Keidar and J. Sussman and K. Marzullo and D. Dolev. A Client-Server Oriented Algorithm for Virtually Synchronous Group Membership in WANs. 20th International Conference on Distributed Computing Systems (ICDCS), pages 356-365, April 2000. Full version: MIT Technical Memorandum MIT-LCS-TM-593a, June 1999, revised September 2000.

[KKLS00] Idit Keidar and Roger Khazan and Nancy Lynch and Alex Shvartsman. An Inheritance-Based Technique for Building Simulation Proofs Incrementally. 22nd International Conference on Software Engineering (ICSE), Limerick, Ireland, pages 478-487, June 2000.

[BKAL00] Ziv Bar-Joseph and Idit Keidar and Tal Anker and Nancy Lynch. QoS Preserving Totally Ordered Multicast. 5th International Conference On Principles Of Distributed Systems (OPODIS)}, pages 143—162, Paris, France, December 2000. Special issue of *Studia Informatica Universalis*. Editor Franck Butelle.

[IK01] Kyle W. Ingols and Idit Keidar. Availability Study of Dynamic Voting Algorithms. 21st International Conference on Distributed Computing Systems (ICDCS) April, 2001. To appear.

[FK01] Alan Fekete and Idit Keidar. A Framework for Highly Available Services Based on Group Communication. IEEE 21st International Conference on Distributed Computing Systems Workshops (ICDCS-21W 2001)); the International Workshop on Applied Reliable Group Communication (WARGC)}, April, 2001. To appear.

[Attie-Lynch] Paul Attie and Nancy Lynch. Dynamic Input/Output Automata: A Formal Model for Dynamic Systems. Submitted for publication.

[Attie-Lynch-tr] Paul Attie and Nancy Lynch. Dynamic Input/Output Automata: A Formal Model for Dynamic Systems. MIT LCS Technical Report to appear.