University of Texas at Austin

CS380L Advanced Operating Systems

Porting Fast Paxos

Authors:
Ankit Goyal
Cheng Fu
Gurbinder Gill

Supervisor:
Dr. Emmett Witchel

November 29, 2014

Abstract

Paxos is a protocol for solving consensus in an asynchronous environment that admits crash failures. *Consensus* is a process of agreeing on one result among group of participants. *Classic Paxos*[**?**] proceeds over several rounds to decide on a sequence of commands. In this paper, we port an existing implementation[**?**] variant of *classic paxos* called *Fast Paxos*[**?**]. *Fast Paxos* reduces the number of messages between client request and response by 2 [1].

1 Introduction

The problem of agreeing on a sequence of operations or values proposed by different processes is known as the *Distributed Consensus* problem. From *FLP* result[**?**], it is impossible to solve consensus in an asynchronous distributed system even if a single process fails by permanently stopping or if a distributed system suffers a *Partial Failure*, in which processes may stop and recover later. Paxos is an algorithm that gets around this problem by making sure that the system doesn't violate the safety requirements during periods when system behaves asynchronously and is certain to make progress (liveliness) if the system behaves partially synchronously for periods long enough to satisfy the progress requirements.

Classic Paxos and Fast Paxos are most widely studied algorithmic solutions to the problem of distributed consensus. Fast Paxos has smaller theoretical latency, therefore is faster, but Classic Paxos is more resilient and hence can tolerate more failures. In this project we ported the Fast Paxos algorithm on the simulator to study its behavior in the presence of different failure scenarios.

2 Protocal Overview

2.1 Paxos Roles

In our implementation of Fast Paxos, each server can take following roles:

[1]in case of no conflicts

1. **Proposers** receives request from clients, associate it with the current slot number and send accept $\langle req, slot_num, ballot \rangle$.

2. **Acceptors** that accept proposals

3. **Leader** A distinct proposer assumers the role of leader.

A distinct proposer assumes the role of leader. In Classic Paxos, the leader is responsible for serializing the commands in global order by assigning an unique timestamp or slot number when proposing a new value. While in Fast Paxos, the leader is responsible for proposing values when a conflict occurs, that is, when two proposers are trying to propose different values to the same command slot. Leader is checking the progress of the protocol periodically and doing arbitration if it sees a conflict. To be able to detect the conflicts, the leader must also be a learner.

2.2 Message Flows

The normal-case communication pattern in Paxos protocol is proposer $\rightarrow$ leader $\rightarrow$ acceptors $\rightarrow$ learners. In Fast Paxos, the proposer send its proposal to acceptors directly. bypassing the leader and saving one message delay. So the message flow is proposer $\rightarrow$ acceptors $\rightarrow$ learners. However.

Use section and subsection commands to organize your document. LaTeX handles all the formatting and numbering automatically. Use ref and label commands for cross-references.

2.3 Comments

Comments can be added to the margins of the document using the todo command, as shown in the example on the right. You can also add inline comments too:

This is an inline comment.

Here's a comment in the margin!

2.4 Tables and Figures

Use the table and tabular commands for basic tables — see Table **??**, for example. You can upload a figure (JPEG, PNG or PDF) using the files

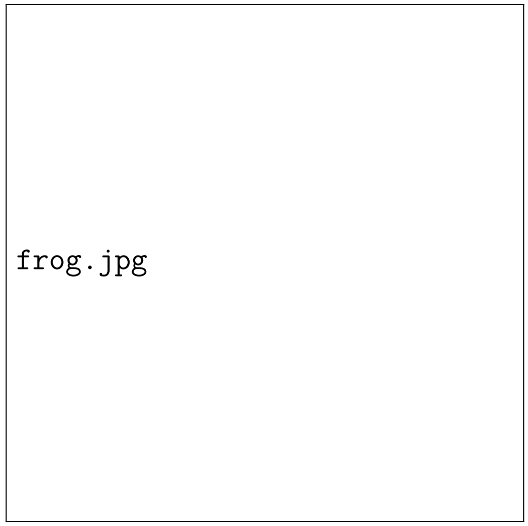

Figure 1: This is a figure caption.

Item	Quantity
Widgets	42
Gadgets	13

Table 1: An example table.

menu. To include it in your document, use the includegraphics command as in the code for Figure **??** below.

2.5 Mathematics

LaTeX is great at typesetting mathematics. Let $X_1, X_2, \ldots, X_n$ be a sequence of independent and identically distributed random variables with $\mathrm{E}[X_i] = \mu$ and $\mathrm{Var}[X_i] = \sigma^2 < \infty$, and let

$$S_n = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n}\sum_i^n X_i$$

denote their mean. Then as n approaches infinity, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $\mathcal{N}(0, \sigma^2)$.

2.6 Lists

You can make lists with automatic numbering . . .

1. Like this,
2. and like this.

. . . or bullet points . . .

- Like this,
- and like this.

We hope you find writeLaTeX useful, and please let us know if you have any feedback using the help menu above.

References

[1] Lamport, Leslie (May 1998). "The Part-Time Parliament" *ACM Transactions on Computer Systems 16 (2): 133–169*

[2] Marco Primi and Daniele Sciascia. "libfastpaxos" *http://libpaxos.sourceforge.net/paxos_projects.php#libfastpaxos*

[3] Lamport, Leslie (July 2005). "Fast Paxos"

[4] Fischer, Michael J; Nancy A. Lynch; Michael S. Paterson (April 1985) "Impossibility of distributed consensus with one faulty process". *Journal of the ACM 32*