Zyzzyva

Speculative Byzantine Fault Tolerance

Giovanni Mormone, Fabian Aspee Encina

Dipartimento Ingegneria e Scienze Informatiche
Università di Bologna

23 febbraio 2021

Overview

1. Problema del consenso e fallimenti bizantini

2. Zyzzyva

Problema del consenso

Consenso

Il problema del consenso è uno dei problemi fondamentali della computazione distribuita, consiste nel fare in modo che tutti i processi corretti che compongono il sistema raggiungano un accordo su un valore, questo però non è risolvibile in reti asincrone, in presenza di anche un singolo fallimento inaspettato di un processo [Fischer, 1985].

I processi che costituiscono il sistema possono essere divisi in due gruppi:

- Processi corretti, che sono quindi in possesso dei giusti valori riguardo al sistema e sono raggiungibili e disponibili a comunicare questi valori.
- Processi faulty, che possono avere dei valori errati, possono essere irraggiungibili oppure possono assumere comportamenti arbitrari in cui forniscono deliberatamente valori errati.

Problema del consenso

In base a queste definizioni, in un sistema in cui i processi si scambiano messaggi per proporre dei valori, valori su cui dovranno raggiungere un consenso, devono essere sempre valide le seguenti proprietà:

- Terminazione, prima o poi un processo corretto prenderà una decisione.
- Accordo, la decisione presa da ogni processo corretto deve essere la stessa.
- Integrità, se tutti i processi corretti hanno proposto lo stesso valore, allora tutti i processi corretti hanno scelto quel valore.

Fallimenti bizantini

I possibili fallimenti che possiamo avere in un sistema distribuito sono:

- Fail-stop: il processore p esegue correttamente, ma può interrompersi in qualsiasi momento.
- Omission: un processore faulty p esegue il suo protocollo correttamente, ma in seguito ad una Send(m,p_j), quando viene eseguito per pi potrebbe non mettere m nel buffer di p_j.
- Authenticated Byzantine: comportamento arbitrario, però i messaggi possono essere firmati col nome del processore che li manda.
- Byzantine: comportamento arbitrario ma non ci sono meccanismi di firma.

Fallimenti bizantini

La soluzione proposta da Lamport et al. dice che, avendo $3f + 1$ processori in un sistema, in presenza di fallimenti bizantini il sistema può continuare a funzionare in maniera corretta, sempre che i processori faulty non superino f.

Per poter ottenere affidabilità, vengono usate delle firme sui messaggi in modo tale che una replica non riceva dei messaggi falsificati.

Zyzzyva

A differenza di altri algoritmi utilizzati per gestire fallimenti bizantini, Zyzzyva si basa sulla speculazione per ridurre il costo ed aumentare le prestazioni del sistema. In Zyzzyva le repliche eseguono le richieste proposte da un Primary in maniera speculativa senza effettuare una costosa fase di accordo. Grazie alla speculazione, Zyzzyva permette di completare una richiesta, nel caso migliore, in due fasi del protocollo invece delle solite tre fasi utilizzate negli altri algoritmi di BFT.

Zyzzyva

Zyzzyva è composto da tre sottoprotocolli[Kotla, 2007] principali:

- Agreement subprotocol
- Viewchange subprotocol
- Checkpoint subprotocol

Problemi di safety

A distanza di anni, gli stessi autori dell'algoritmo hanno messo in luce alcuni problemi riguardo la safety di Zyzzyva [Abraham, 2017].

References

Kotla, Ramakrishna and Alvisi, Lorenzo and Dahlin, Mike and Clement, Allen and Wong, Edmund

Zyzzyva: speculative byzantine fault tolerance

Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles 45–58.

Abraham, Ittai and Gueta, Guy and Malkhi, Dahlia and Alvisi, Lorenzo and Kotla, Rama and Martin, Jean-Philippe

Revisiting fast practical byzantine fault tolerance

arXiv preprint arXiv:1712.01367.

Fischer, Michael J and Lynch, Nancy A and Paterson, Michael S

Impossibility of distributed consensus with one faulty process

*Journal of the ACM (JACM)*374–382.