EE 382V: Parallel Algorithms **Fall 2017**

Lecture 15: Consensus in Asynchronous Systems

Lecturer: Vijay Garg *Scribe: Dale Pedzinski*

15.1 Consensus in Asynchronous Systems

In this lecture, we discussed the Fischer, Lynch and Paterson result, which explains that it is impossible for an asynchronous system to reach a consensus, if the system experiences a fault.

15.2 Definitions

- Fischer, Lynch and Paterson(FLP) : there is no protocol that satisfies agreement, non-triviality, and termination in an asynchronous system in presence of one fault.
- Fault: any failure in the system like hardware failure. Does not include a delayed response
- Crash Model: a model in how the system fails, either unannounced or announced. This section focuses on unannounced
- Omission Model: a model where a node omits or ignores some messages.
- Byzantine Model: a model where a node acts as a lone wolf in the system
- Asynchronous: a process or message delivery that takes an arbitrary but finite time to complete or be delivered

15.3 Fischer, Lynch and Paterson

The theorem of Fischer, Lynch and Paterson is there is no deterministic algorithm to solve binary consensus in an asynchronous system with 2 or more processes in presence of even one unannounced crash.

15.3.1 Assumptions of System

Throughout the theorem, we have a set of assumptions about the environment:

- Initial Independence: Every process choses their input so every input vector is possible.
- Disjoint Events: Disjoint events that commute will have the same global state.
- Asynchronous of events: Events in the environment are asynchronous

15.3.2 Requirements

The model of this problem are a faulty process is one that eventually will complete and if at most one process is faulty. However, since the system is reliable all messages sent to the non-faulty processes are eventually delivered.

Lastly, the requirements of this problem can be summarized from

- Agreement: All correct process decide on the same value or the book defines it as: "Two non-faulty processes cannot commit on different values."
- Validity: Value decided by the steam must be proposed by someone
- Non-triviality: Both 0 and 1 possible outcomes of the protocol or the book defines it as: "Both values 0 and 1 should be possible outcomes. This requirement eliminates protocols that return a fixed value 0 or 1 independent of the initial input."
- Termination: a non-faulty process decides in finite time or the book defines it as: "A non-faulty process decides in finite time."

15.4 Binary Consensus Problem

In class we discussed two types of environments with regards to the Binary Consensus Problem, basically all types discussed that have one figure will not work. The first example discussed was around the Democratic based system, where it's failure will break while the system waits on a consensus. The other system Leader or dictatorship, will work if any node other than the leader failures but since we can't control the failures. This also fails.

The basis of the problem is any environment thought of will fail if there is one failure.

15.4.1 Every Protocol has at least one initial bi-valent global state

Proof: If they only differ in only one process there exists two neighboring global states G0 and G1 where G0 is 0-valent and G1 is 1-valent. Look at Figure 15.2 in the book for visual description

15.4.2 There Exists a strategy for the adversary to keep the protocol in a bi-valent state

Proof: For any protocol there exists a critical state for it to go from bi-valent to univalent Look at Figure 15.2 in the book for visual description

References

Referenced the Class book Chapter 15