



Principles of Distributed Computing

Exercise 6

1 Linear Arrow

In Theorem 7.7 of the lecture notes, it is sketched that the cost of the Arrow protocol (Algorithm 7.4) for a set S of simultaneous requests is by a factor $O(\log |S|)$ larger than the cost (i.e. message complexity) of an optimal algorithm. In this exercise, we consider this problem for the simpler case where the tree is a linear list. Show that the approximation factor of Arrow is $O(1)$ if a set S of simultaneous requests is processed on a synchronous array (linear list).

2 Concurrent Arrow

In Theorem 7.9 it was shown that the Read/Write Caching Algorithm 7.8 is 3-competitive (with respect to message complexity) for sequential access to some global variable. Now we want to investigate what happens in the case of concurrent access.

- a) If a read is implemented as in the lecture, what happens to the competitive ratio if we now allow multiple reads at the same time?
- b) Improve the algorithm so that it is 3-competitive for concurrent readers.
- c) Devise an algorithm that also allows one write concurrently with the reads.
- d) Is your algorithm above linearizable? If not, devise one. What is its competitive ratio?
- e) Finally, improve the algorithm so that it handles any number of concurrent operations (reads, writes) correctly and is linearizable.
Hint: Do not worry about the competitive ratio of your algorithm.