We study the problem of Byzantine agreement in synchronous systems where malicious agents can move from one process to another and try to corrupt them. This model is known as mobile Byzantine faults. In a previous result [10], Garay has shown that \( n > 6t \) (\( n \) is the total number of processes, and \( t \) is the number of mobile faults) is sufficient to solve this problem even in the presence of strong agents. These agents can move at full speed (in the sense that each agent can take a movement in every round) and can make corrupted processes forget that they run the algorithm (as a result, after recovery a process must learn the current state of computation including the code from other processes). Many following results [3] have improved the above result but with some additional assumptions such as a corrupted process must recover and learn the current state of computation before another process can fail instead of it. The question, whether the result of Garay can be improved without any additional assumption, remains open. In this paper, we answer this question by providing an algorithm MBA that works with \( n > 4t \).
An Improved Byzantine Agreement Algorithm for Synchronous Systems with Mobile Faults

Index Terms

Computer Science

Keywords
Synchronous Systems Agreement(consensus) Problem Mobile Byzantine Adversary