J. Parallel Distrib. Comput. 74 (2014) 2860–2871

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

journal homepage: www.elsevier.com/locate/jpdc

Approximate Byzantine consensus in sparse, mobile ad-hoc networks☆

Chuanyou Li [a], Michel Hurfin [b,*], Yun Wang [a]

[a] School of Computer Science and Engineering, Southeast University, Key Lab of Computer Network & Information Integration, Ministry of Education, Nanjing, 211189, China

[b] INRIA Rennes Bretagne Atlantique Research Center, Campus de Beaulieu, 35042, Rennes, France

HIGHLIGHTS

- A new condition for the convergence of an approximate Byzantine Consensus protocol.
- The condition focuses on the two extreme values owned by some correct nodes.
- Connectivity requirements have to be satisfied by a minimal number of nodes.
- Alternative functions are proposed to manage the data structures (logs).
- Specific notions are used to prove the sufficiency of the value-based condition.

ARTICLE INFO

Article history:
Received 14 June 2013
Received in revised form
3 April 2014
Accepted 20 May 2014
Available online 2 June 2014

Keywords:
Approximate consensus
Mobility
Ad-hoc network

ABSTRACT

We consider the problem of approximate consensus in mobile ad-hoc networks in the presence of Byzantine nodes. Due to nodes' mobility, the topology is dynamic. We propose a protocol based on the linear iteration method. The nodes collect information during several consecutive rounds: moving gives them the opportunity to gather progressively enough values. A novel sufficient and necessary condition guarantees the final convergence: from time to time only the correct nodes that own a value equal to (or very close to) either the minimum or the maximum value have to receive enough messages (quantity constraint) with either higher or lower values (quality constraint). Of course, nodes' motion should not prevent this requirement to be fulfilled. New concepts are introduced to prove the correctness of the protocol. Based on particular mobility scenarios, simulations are conducted to analyze the impact of some parameters on three variants of the protocol.

1. Introduction

We consider a distributed system made of mobile nodes connected by wireless links. These nodes form an ad-hoc network characterized by a dynamic topology. The system is unreliable. Nodes may suffer from Byzantine faults and messages may be lost. A Byzantine node, also called a malicious node, may stop its activity or execute an arbitrary code. In particular, it may send messages with fake values. Nodes that are not malicious are said to be correct.

Consensus is recognized as a basic paradigm for fault-tolerance in distributed systems. According to the application's needs, several variants of the consensus problem have been proposed. Among these agreement abstractions, one is called the *approximate consensus* problem and has been presented for the first time in [5]. In this particular problem, each node begins its participation by providing a real value called its initial value. Then eventually all correct nodes must obtain final values that are different from each other within a maximum gap denoted as ϵ (convergence property) and must be in the range of initial values proposed by the correct nodes (validity property). Whatever the semantic associated to the value (an instant of time, a reputation score, a positioning in the space, ...), this generic convergence mechanism ensures that the correct nodes adjust their values to obtain eventually equivalent values: by definition, all the values are eventually contained in a limited range which is itself in the range of correct initial values.

☆ This work is partially supported by (1) Natural Science Foundation, China, under Grant 60973122, by National 863 Hi-Tech Program, China, under Grant 2011AA040502, and (2) by the ANR French national program for Security and Informatics (Grant #ANR-11-INSE-010, project AMORES).

* Corresponding author.
E-mail address: Michel.Hurfin@inria.fr (M. Hurfin).

http://dx.doi.org/10.1016/j.jpdc.2014.05.005
0743-7315/

Several protocols have been proposed to solve this problem in the presence of Byzantine nodes. Some protocols [5,1] assume that the network is fully connected: during the whole execution, a correct node should be able to communicate by message passing with any other correct node. Obviously, this property is not satisfied in our context. Other protocols [14,15,11] consider partially connected networks but require an additional constraint: any correct node must know the whole topology. Again, such global information is impossible to obtain in our context. Based on the linear iterative consensus strategy [13], recent protocols [17,18, 10] also assume that the network is partially connected but do not require any global information. At each iteration, a correct node broadcasts its value, gathers values from its neighborhood and updates its own value. Its new value is an average of its own previous value and those of some of its neighbors. In order to achieve convergence, the proposed solutions rely on additional conditions that have to be satisfied by the topology. In [17,10], the proposed conditions are proved to be sufficient and necessary in the case of an arbitrary directed graph.

The solution presented in this paper addresses the approximate Byzantine consensus problem in mobile ad-hoc networks. It follows the general strategy proposed in [17,18,10]. However, this work differs from these previous works for two main reasons. First we modify the iterative protocol to cope more efficiently with mobility. Each node still follows an iteration scheme and repeatedly executes rounds. Yet a round is now decomposed into two parts: a moving step followed by a computing step. During the moving step, a node may change its location. As in other solutions, during the computing step, a node broadcasts its own value, gathers values from its neighbors and updates its own value. However, in our solution, the values used to compute a new value are not only those received during the current round. In other words, a correct node can now take into account values contained in messages sent during several consecutive rounds. An integer parameter (denoted as Δ hereafter) is used to define the maximal number of rounds during which gathered values can be locally stored and used. Thanks to this flexibility, a node can use its ability to travel to collect progressively enough values. Through the definition of two different resetting strategies and two different selecting strategies, we propose a few variants of the same basic algorithm. For each of them, the subset of gathered values used during the computation may differ.

The second difference is the most important contribution. It does not concern the protocol itself but the condition on which the proof of convergence is based. Obviously, two different values can remain forever in the system if the nodes that own these values belong to two disconnected parts of the system. A condition aims at defining some minimal connectivity requirements. While the solutions proposed in [17,18,10] define conditions that refer only to the topology, we present a condition that considers also the values proposed by the correct nodes. To understand the interest of our approach, let us consider the following example. One correct node p_i proposes an initial value v_a while all the other correct nodes propose an initial value v_b (different from v_a). In this particular scenario, if the node p_i can receive values from a sufficient number of correct neighbors, approximate consensus can be reached even if all the other nodes are isolated and receive no message. This particular example suggests that the location of the (minimal and maximal) values is just as important as the network topology. In [17,10], constraints on the topology ensure that each node has enough neighbors. These constraints are "universal" because they apply to all nodes of the network. The above example shows that universality is not always necessary: p_i is the only node which must have enough neighbors; the others can remain disconnected and keep their value v_b. More generally, a universal constraint is not well suited to cope with a mobile environment, where a node is sometimes isolated from (or insufficiently connected to) the rest of the network. In this paper, we present a sufficient and necessary condition where both the topology and the values proposed by correct nodes are taken into account. The condition affects only a subset of nodes that can change from one round to another. More precisely, the condition focuses only on the correct nodes that own a value equal (or very close) to either the maximum or the minimum value. The other nodes have no obligation. To achieve consensus, from time to time, these particular nodes must receive enough messages (quantity requirement) with values that are sufficiently different from their current value (quality requirement). As the proposed condition has to be satisfied by just a subset of nodes, it is not universal and therefore it is weaker than those already proposed in related works. The fact that a different condition is considered and the fact that collected values are stored for several rounds both have an impact on the correctness proof of the protocol.

The rest of this paper is organized as follows. Section 2 introduces the model and provides a formal definition of the approximate consensus problem. Section 3 sketches out some related works. In Section 4, we present our protocol based on linear iteration. The proof of the validity property is provided in Section 5. Section 6 focuses on the convergence property and presents a sufficient and necessary condition suitable for mobile environments. In Section 7, we analyze the performances of three variants of the protocol under two different mobility models. Section 8 brings our concluding remarks.

2. Model and problem definition

2.1. Model

We consider a mobile distributed system composed of n nodes. The set of nodes V is denoted as $\{p_1, p_2, \ldots, p_i, \ldots, p_n\}$. Each node p_i can move towards any direction and at any speed within a limited geographical area. A node communicates only with its close neighbors by exchanging messages. Therefore the topology (*i.e.*, the communication graph) is dynamic. Interferences occur frequently in a wireless environment. Thus messages can be duplicated or lost during their transmission. If a message reaches its destination, the message passing delay depends on many factors (the channel characteristics, the task scheduling strategies, the arrival rate, *etc.*). Works on the propagation delay in ad hoc networks [19,6] show that the message passing delay between two neighbors is often of the order of a few milliseconds when the network is not heavily loaded. Consequently, if a node communicates infinitely often with the same neighbors, most of its messages are received by the one-hop stable nodes shortly after their sending. Communication channels are not symmetric. Therefore, during a round r (*i.e.*, an iteration of the protocol), a *simple directed graph* $G_r(V, E_r)$ is usually used to model the dynamic topology.

Nodes of V are divided into two static subsets denoted as C and F. The set C contains the correct nodes. By definition, they always follow the protocol's specification. The nodes of the set F are Byzantine nodes. They may behave arbitrarily and can collude together. In particular, each of them can stop its computation or send simultaneously messages with different fake values to different neighbors. Of course, no correct node knows the splitting of the nodes into the subsets C and F. Only two assumptions restrict the power of the faulty nodes. First, the total number of Byzantine nodes is limited by f. Therefore $|F| \leq f$. The number of correct nodes is denoted as $c = |C|$. Consequently, $n - f \leq c \leq n$. Second, a node is always able to identify the real sender of any received message. In other words, a faulty node cannot use fake identities to communicate with other nodes. In the proposed solution, each