Brief Announcement: Byzantine Geoconsensus

Joseph Oglio, Kendric Hood, Gokarna Sharma(✉), and Mikhail Nesterenko

Department of Computer Science, Kent State University, Kent, OH 44242, USA
{joglio,khood}@kent.edu, {sharma,mikhail}@cs.kent.edu

Abstract. We define and investigate the consensus problem for a set of N processes embedded on the d-dimensional plane, $d \geq 2$, which we call the *geoconsensus* problem. The processes have unique coordinates and can communicate with each other through oral messages. In contrast to the literature where processes are individually considered Byzantine, it is considered that all processes covered by a finite-size convex fault area F are Byzantine and there may be one or more processes in a fault area. Similarly as in the literature where correct processes do not know which processes are Byzantine, it is assumed that the fault area location is not known to the correct processes.

In this paper, we first prove that the geoconsensus is impossible if all processes may be covered by at most three areas where one is a fault area. We then prove the following results on the constructive side considering the 2-dimensional embedding. For $M \geq 1$ fault areas F of arbitrary shape with diameter D, we present a consensus algorithm that tolerates $f \leq N-(2M+1)$ Byzantine processes provided that there are $9M+3$ processes with pairwise distance between them greater than D. For square F with side ℓ, we provide a consensus algorithm that lifts this pairwise distance requirement and tolerates $f \leq N - 15M$ Byzantine processes given that all processes are covered by at least $22M$ axis aligned squares of the same size as F. For a circular F of diameter ℓ, this algorithm tolerates $f \leq N - 57M$ Byzantine processes if all processes are covered by at least $85M$ circles. Finally, we extend these results to various size combinations of fault and non-fault areas as well as d-dimensional process embeddings, $d \geq 3$.

1 Introduction

The problem of *Byzantine consensus* [10,14] has been attracting extensive attention from researchers and engineers in distributed systems. It has applications in distributed storage [1,2,4,5,9], secure communication [6], safety-critical systems [16], blockchain [12,17,19], and Internet of Things (IoT) [11].

Consider a set of N processes with unique IDs that can communicate with each other. Assume that f processes out of these N processes are Byzantine. Assume also that which process is Byzantine is not known to correct processes,

S. Devismes and N. Mittal (Eds.): SSS 2020, LNCS 12514, pp. 199–204, 2020.
https://doi.org/10.1007/978-3-030-64348-5_15

except possibly the size f of Byzantine processes. The Byzantine consensus problem here requires the $N - f$ correct processes to reach to an agreement tolerating arbitrary behaviors of the f Byzantine processes.

Pease *et al.* [14] showed that the maximum possible number of faults f that can be tolerated depends on the way how the (correct) processes communicate: through oral messages or through unforgable written messages (also called signatures). An *oral* message is completely under the control of the sender, therefore, if the sender is Byzantine, then it can transmit any possible message. This is not true for a signed, written message. Pease *et al.* [14] showed that the consensus is solvable only if $f < N/3$ when communication between processes is through oral messages. For signed, written messages, they showed that the consensus is possible tolerating any number of faulty processes $f \leq N$.

The Byzantine consensus problem discussed above assumes nothing about the locations of the processes, except that they have unique IDs. Since each process can communicate with each other, it can be assumed that the N processes work under a complete graph (i.e., clique) topology consisting of N vertices and $N(N-1)/2$ edges. Byzantine consensus has also been studied in arbitrary graphs [14,18] and in wireless networks [13], relaxing the complete graph topology requirement so that a process may not be able to communicate with all other $N - 1$ processes. The goal in these studies is to establish necessary and sufficient conditions for consensus to be solvable. For example, Pease *et al.* [14] showed that the consensus is solvable through oral messages tolerating f Byzantine processes if the communication topology is $3f$-regular. Furthermore, there is a number of studies on a related problem of *Byzantine broadcast* when the communication topology is not a complete graph topology, see for example [8,15]. Byzantine broadcast becomes fairly simple for a complete graph topology.

Recently, motivated by IoT-blockchain applications, Lao *et al.* [11] proposed a consensus protocol, which they call Geographic-PBFT or simply G-PBFT, that extends the well-known PBFT consensus protocol by Castro and Liskov [4] to the geographic setting. The authors considered the case of fixed IoT devices embedded on geographical locations for data collection and processing. The location data can be obtained through recording location information at the installation time or can also be obtained using low-cost GPS receivers or location estimation algorithms [3,7]. They argued that the fixed IoT devices have more computational power than other mobile IoT devices (e.g.., mobile phones and sensors) and are less likely to become malicious nodes. They then exploited (geographical) location information of fixed IoT devices to reach consensus. They argued that G-PBFT avoids Sybil attacks, reduces the overhead for validating and recording transactions, and achieves high consensus efficiency and low traffic intensity. However, G-PBFT is validated only experimentally and no formal analysis is given.

In this paper, we formally define and study the Byzantine consensus problem when processes are embedded on the geographical locations in fixed unique coordinates, which we call the *Byzantine geoconsensus* problem. If fault locations are not constrained, the geoconsensus problem differs little from the Byzantine consensus. This is because the unique locations serve as IDs of the processes and same set of results can be established depending on whether communication between processes is through oral messages or unforgable written messages. Therefore, we relate the fault locations to the geometry of the problem, assuming that the faults are limited to a *fault area* F (going beyond the limitation of mapping Byzantine behavior to individual processes). In other words, the fault area lifts the restriction of mapping Byzantine behavior to individual processes in the classic setting and now maps the Byzantine behavior to all the processors within a certain area in the geographical setting. Applying the classic approaches of Byzantine consensus may not exploit the collective Byzantine behavior of the processes in the fault area and hence they may not provide benefits in the geographical setting. Furthermore, we are not aware of prior work in Byzantine consensus where processes are embedded in a geometric plane while faulty processes are located in a fixed area.

In light of the recent development on location-based consensus protocols, such as G-PBFT [11], discussed above, we believe that our setting deserves a formal study. In this paper we consider the Byzantine geoconsensus problem in case the processes are embedded in a d-dimensional plane, $d \geq 2$. Formally, we define the problem as follows. Consider the binary consensus where every correct process is input a value $v \in \{0, 1\}$ and must output an irrevocable decision with the following three properties.

Agreement – no two correct processes decide differently;
Validity – if all the correct processes input the same value v, then every correct process decides v;
Termination – every correct process eventually decides.

Definition 1 (Byzantine Geoconsensus). *An algorithm solves* the Byzantine geoconsensus Problem *(or* geoconsensus *for short) for fault area set $\mathcal{F}$, if every computation produced by this algorithm satisfies the three consensus properties.*

We study the possibility and bounds for a solution to geoconsensus. We demonstrate that geoconsensus allows quite robust solutions: all but a fixed number of processes may be Byzantine. We discuss in detail our contributions below.

Contributions. Let N denotes the number of processes, M denotes the number of fault areas F, D denotes the diameter of F, and f denotes the number of faulty processes. Assume that each process can communicate with all other $N-1$ processes and the communication is through oral messages. Assume that all the processes covered by a faulty area F are Byzantine. The correct processes know the size of each faulty area (such as its diameter, number of edges, area, etc.) and the total number M of them but do not know their exact location.

In this paper, we made the following five contributions:

(i) An impossibility result that geoconsensus is not solvable if all N processes may be covered by 3 equal size areas F and one of them may be fault area.

This extends to the case of N processes being covered by $3M$ areas F with M areas being faulty.

(ii) The algorithm *BASIC* that solves geoconsensus tolerating $f \leq N-(2M+1)$ Byzantine processes, provided that there are $9M+3$ processes with pairwise distance between them greater than D.

(iii) The algorithm *GENERIC* that solves geoconsensus tolerating $f \leq N-15M$ Byzantine processes, provided that all N processes are covered by $22M$ axis-aligned squares of the same size as the fault area F, removing the pairwise distance assumption in the algorithm *BASIC*.

(iv) An extension of the *GENERIC* algorithm to circular F tolerating $f \leq N-57M$ Byzantine processes if all N processes are covered by $85M$ circles of same size as F.

(v) Extensions of the results (iii) and (iv) to various size combinations of fault and non-fault areas as well as to d-dimensional process embeddings, $d \geq 3$.

Our results are interesting as they provide trade-offs among N, M, and f, which is in contrast to the trade-off provided only between N and f in the Byzantine consensus literature. For example, the results in Byzantine consensus show that only $f < N/3$ Byzantine processes can be tolerated, whereas our results show that as many as $f \leq N-\alpha M$, Byzantine processes can be tolerated provided that the processes are placed on the geographical locations so that at least βM areas (same size as F) are needed to cover them. Here α and β are both integers with $\beta \geq c \cdot \alpha$ for some constant c.

Furthermore, our geoconsensus algorithms reduce the message and space complexity in solving consensus. In the Byzantine consensus literature, every process sends communication with every other process in each round. Therefore, in one round there are $O(N^2)$ messages exchanged in total. As the consensus algorithm runs for $O(f)$ rounds, in total $O(f \cdot N^2)$ messages are exchanged in the worst-case. In our algorithms, let N processes are covered by X areas of size the same as fault area F. Then in a round only $O(X^2)$ messages are exchanged. Since the algorithm runs for $O(M)$ rounds to reach geoconsensus, in total $O(M \cdot X^2)$ messages are exchanged in the worst-case. Therefore, our geoconsensus algorithms are message (equivalently communication) efficient. The improvement on space complexity can also be argued analogously.

Finally, Pease *et al.* [14] showed that it is impossible to solve consensus through oral messages when $N = 3f$ but there is a solution when $N \geq 3f+1$. That is, there is no gap on the impossibility result and a solution. We can only show that it is impossible to solve consensus when all N processes are covered by $3M$ areas that are the same size as F but there is a solution when all N processes are covered by at least $22M$ areas (for the axis-aligned squares case). Therefore, there is a general gap between the condition for impossibility and the condition for a solution.

Techniques. Our first contribution is established extending the impossibility proof technique of Pease *et al.* [14] for Byzantine consensus to the geoconsensus setting. The algorithm *BASIC* is established first through a leader selection to compute a set of leaders so that they are pairwise more than distance D away from each other and then running carefully the Byzantine consensus algorithm of Pease *et al.* [14] on those leaders.

For the algorithm *GENERIC*, we start by covering processes by axis-aligned squares and studying how these squares may intersect with fault areas of various shapes and sizes. Determining optimal axis-aligned square coverage is NP-hard. We provide constant-ratio approximation algorithms. We also discuss how to cover processes by circular areas. Then, we use these ideas to construct algorithm *GENERIC* for fault areas that are either square or circular, which does not need the pairwise distance requirement of *BASIC* but requires the bound on the number of areas in the cover area set. Finally, we extend these ideas to develop covering techniques for higher dimensions. These covering techniques then provide tolerance bounds for Byzantine consensus in higher dimensions.

Future Work. Our results show the dependency of the tolerance guarantees on the shapes and sizes of the fault areas. Therefore, for future work, it would also be interesting to consider fault area F shapes beyond circles and squares that we studied; to investigate process coverage by non-identical squares, circles or other shapes to see whether better bounds on the set $\mathcal{A}$ and fault-tolerance guarantee f can be obtained. It would also be interesting to close or reduce the gap between the condition for impossibility and a solution (as discussed in Contributions).

References

1. Abd-El-Malek, M., Ganger, G.R., Goodson, G.R., Reiter, M.K., Wylie, J.J.: Fault-scalable byzantine fault-tolerant services. ACM SIGOPS Operating Syst. Rev. **39**(5), 59–74 (2005)
2. Adya, A., et al.: FARSITE: federated, available, and reliable storage for an incompletely trusted environment. ACM SIGOPS Operating Syst. Rev. **36**(SI), 1–14 (2002)
3. Bulusu, N., Heidemann, J., Estrin, D., Tran, T.: Self-configuring localization systems: design and experimental evaluation. ACM Trans. Embed. Comput. Syst. (TECS) **3**(1), 24–60 (2004)
4. Castro, M., Liskov, B.: Practical byzantine fault tolerance and proactive recovery. ACM Trans. Comput. Syst. (TOCS) **20**(4), 398–461 (2002)
5. Castro, M., Rodrigues, R., Liskov, B.: BASE: using abstraction to improve fault tolerance. ACM Trans. Comput. Syst. (TOCS) **21**(3), 236–269 (2003)
6. Cramer, R., Gennaro, R., Schoenmakers, B.: A secure and optimally efficient multi-authority election scheme. Eur. Trans. Telecommun. **8**(5), 481–490 (1997)
7. Hightower, J., Borriello, G.: Location systems for ubiquitous computing. Computer **34**(8), 57–66 (2001)
8. Koo, C.Y.: Broadcast in radio networks tolerating byzantine adversarial behavior. In: PODC, pp. 275–282 (2004)
9. Kubiatowicz, J., et al.: OceanStore: an architecture for global-scale persistent storage. ACM SIGOPS Operating Syst. Rev. **34**(5), 190–201 (2000)
10. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Trans. Programm. Lang. Syst. **4**(3), 382–401 (1982)
11. Lao, L., Dai, X., Xiao, B., Guo, S.: G-PBFT: a location-based and scalable consensus protocol for IOT-Blockchain applications. In: IPDPS, pp. 664–673 (2020)

12. Miller, A., Xia, Y., Croman, K., Shi, E., Song, D.: The honey badger of BFT protocols. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 31–42 (2016)
13. Moniz, H., Neves, N.F., Correia, M.: Byzantine fault-tolerant consensus in wireless Ad Hoc networks. IEEE Trans. Mobile Comput. **12**(12), 2441–2454 (2012)
14. Pease, M., Shostak, R., Lamport, L.: Reaching agreement in the presence of faults. J. ACM **27**(2), 228–234 (1980), https://doi.org/10.1145/322186.322188
15. Pelc, A., Peleg, D.: Broadcasting with locally bounded byzantine faults. Inf. Process. Lett. **93**(3), 109–115 (2005)
16. Rushby, J.: Bus architectures for safety-critical embedded systems. In: Henzinger, T.A., Kirsch, C.M. (eds.) EMSOFT 2001. LNCS, vol. 2211, pp. 306–323. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45449-7_22
17. Sousa, J., Bessani, A., Vukolic, M.: A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In: DSN, pp. 51–58. IEEE (2018)
18. Vaidya, N.H., Tseng, L., Liang, G.: Iterative approximate byzantine consensus in arbitrary directed graphs. In: PODC, pp. 365–374 (2012)
19. Zamani, M., Movahedi, M., Raykova, M.: RapidChain: scaling blockchain via full sharding. In: CCS, pp. 931–948 (2018)