J. Cryptology (2005) 18: 187–189
DOI: 10.1007/s00145-005-1803-1

Journal of
CRYPTOLOGY

Preface

Ran Canetti
IBM Research,
19 Skyline Drive,
Hawthorne, NY 10532, U.S.A.
canetti@watson.ibm.com

Byzantine Agreement, i.e., reaching agreement among multiple agents in the presence of adversarial faults, is a compelling task: It is so easy to formulate and yet so hard to solve. It is also one of the first and most basic challenges in distributed computing, one that appears in many different settings and problems. In fact, Byzantine Agreement has probably shaped the field of distributed computing more than any other task.

One of the fascinating aspects of Byzantine Agreement (or "Consensus" as it is sometimes called) is the variety of approaches and techniques that are used to solve this problem, in face of the far-reaching impossibility results of Pease et al. [PSL], who also formulated this problem, and Fischer et al. [FLP]. Randomization is one approach. (In fact, Byzantine Agreement is one of the first and most prominent examples of the power of randomness in fault-tolerant distributed computing, see, e.g., [R] and [CD].) Other approaches include the *fault oracles* of Toueg [CT], set-up assumptions [DS], [PW], and computational assumptions. Furthermore, new solutions and approaches keep being proposed. In fact, two of the works in this issue propose such new approaches.

Byzantine Agreement is also a natural "meeting point" between the disciplines of distributed computing and cryptographic protocols. Indeed, reaching agreement on the outcome of the interaction is an essential part of many cryptographic protocol problems. Examples include contract signing by two or more parties, financial transactions, voting, and general multi-party ("peer-to-peer") computation. Conversely, cryptographic techniques and thinking have been used in a number of works to construct Byzantine Agreement protocols (e.g., [FM], [PW], and [CR]).

This special issue features three works that address the Byzantine Agreement problem. The works take very different approaches at tackling the problem. Presenting them side by side underlines the diversity of the research dealing with this problem.

The first paper, by Considine et al., addresses the "classic" problem of reaching agreement unconditionally by deterministic protocols with no prior set-up. In fact, in their case it may be more natural to present the task in the equivalent terms of obtaining *global broadcast*, namely the task where each party can transmit values to all other parties, and where the parties are guaranteed to obtain the same value from any transmission. In this

case it is known that if the parties have pairwise communication links then agreement (or, equivalently, global broadcast) can be reached if and only if the number of non-faulty parties is strictly more than a two-thirds of the overall number of parties. The paper extends this tight result to the case where the pairwise links are replaced by broadcast channels among subsets of limited size. Specifically, for any $b > 0$ it is shown that, in a model where each b-tuple of parties has a dedicated broadcast channel among them, global broadcast can by achieved *if and only if* the non-faulty parties are strictly more than a $2(b + 1)$-fraction of the parties. This remarkable result can be viewed either as a purely "structural" study of the task of reaching agreement, or alternatively as a "design paradigm" for agreement and broadcast protocols. (In fact, the second view was already put to use in [LLR].)

The second paper, by Cachin, Kursawe, and Shoup, employs cryptographic techniques and modeling to provide a simple and efficient instantiation of an old approach for reaching Byzantine Agreement. Ben Or [B] and Rabin [R] have shown how to reach Byzantine Agreement assuming that the parties have access to a common source of "timely randomness", i.e., a source of random bits that are generated in a timely manner and are unpredictable beforehand. This paradigm was later pursued by many works, which realize this "common coin" primitive under various assumptions. The present paper provides a very efficient way for generating such a "common coin." The idea is to have the parties hold shares of a cryptographically "unpredictable" function f, in a way that allows them to evaluate the function jointly. The values of the coin are set to be the value of f at a pre-determined sequence of points. If the analysis is carried out in the Random Oracle model then this "common coins" protocol involves only a single message by each party. To obtain a protocol that is analyzable in the standard model, more rounds are needed. In addition, the present protocol makes extensive use of threshold signatures in order to reduce considerably the communication complexity of the agreement protocol given a common coin.

The third paper, by Goldwasser and Lindell, studies the types of agreement necessary for realizing different levels of security for general cryptographic computation. Surprisingly, they show that a relatively simple form of agreement suffices for achieving most of the security goals of generic secure function evaluation and reactive secure computation. Specifically, if one relaxes the notion of security so as to allow the adversary to "abort" the computation at will, then only a relatively weak form of agreement suffices. This weak agreement is not susceptible to the impossibility result of [PSL]. Furthermore, it can be realized by a simple, deterministic "two round echo" protocol for any number of faults. This simple result is very useful, since in many cryptographic tasks the adversarial ability to "abort" the computation is inherent and unavoidable in the first place.

References

[B] M. Ben-Or. Another advantage of free choice: completely asynchronous agreement protocols. In *Proc. 2nd ACM Symposium on Principles of Distributed Computing* (*PODC*), pp. 27–30, 1983.

[CD] B. Chor and C. Dwork. Randomization in Byzantine agreement. In *Randomness and Computation* (S. Micali, ed.), vol. 5 of Advances in Computing Research, pp. 443–497. JAI Press, Greenwich, CT, 1989.

[CR] R. Canetti and T. Rabin. Fast asynchronous Byzantine agreement with optimal resilience. In *Proc. 25th Annual ACM Symposium on Theory of Computing* (*STOC*), pp. 42–51, 1993.

[CT] T. D. Chandra and S. Toueg. Unreliable failure detectors for reliable distributed systems. *Journal of the ACM*, vol. 43, no. 2, pp. 225–267, 1996.

[DS] D. Dolev and H. R. Strong. Authenticated algorithms for Byzantine Agreement. *SIAM Journal on Computing*, vol. 12, no. 4, pp. 656–666, 1983.

[FM] P. Feldman and S. Micali. An optimal probabilistic protocol for synchronous Byzantine Agreement. *SIAM Journal on Computing* vol. 26, no. 4, pp. 873–933. (1997)

[FLP] M. J. Fischer, N. A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM*, vol. 32, pp. 374–382, Apr. 1985.

[LLR] Y. Lindell, A. Lysyanskaya, and T. Rabin. On the composition of authenticated byzantine agreement. In *Proc*. 34*th Annual ACM Symposium on Theory of Computing* (*STOC*), pp. 514–523, 2002.

[PSL] M. Pease, R. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *Journal of the ACM*, vol. 27, pp. 228–234, Apr. 1980.

[PW] B. Pfitzmann and M. Waidner. Unconditional Byzantine Agreement for any number of faulty processors. In *Proc*. *Symposium on Theoretical Aspects of Computer Science* (*STACS*), pp. 339–350, 1992.

[R] M. O. Rabin. Randomized Byzantine generals. In *Proc*. 24*th IEEE Symposium on Foundations of Computer Science* (*FOCS*), pp. 403–409, 1983.