GEORGETOWN UNIVERSITY

Nitin H Vaidya <nv198@georgetown.edu>

[DISC 2020] Accepted submission 86: "Improved Extension Protocols for..."

1 message

DISC 2020 HotCRP <noreply@disc2020.hotcrp.com> Fri, Jul 17, 2020 at 11:57 AM
Reply-To: hagit@cs.technion.ac.il
To: Nitin Vaidya <nitin.vaidya@georgetown.edu>
Cc: Hagit Attiya <hagit@cs.technion.ac.il>

Dear Nitin Vaidya,

The 34th International Symposium on DIStributed Computing (DISC) (DISC 2020) program committee is delighted to inform you that your submission 86 has been accepted to appear as a regular paper in the conference.

Title: Improved Extension Protocols for Byzantine Broadcast and Agreement
Site: https://disc2020.hotcrp.com/paper/86

39 of 170 regular submissions were accepted. Congratulations!

Visit the submission site for reviews, comments, and related information. Reviews and comments are also included below.

The final version of your submission should be uploaded by *August 10, 2020*.
Instructions for uploading will follow soon.
Your paper should be prepared in the same LIPIcs format used for submission.
There is a page limit of 15 pages, excluding the bibliography; appendixes are not allowed.

Due to the Coronavirus, the conference will be held virtually, during the dates originally planned for the conference (the week of October 12-16, 2020). Details about the format will follow.
Remember that we require at least one author to register for the conference.

Contact me with any questions or concerns.

Hagit Attiya <hagit@cs.technion.ac.il>
DISC 2020 Program committee chair

Review #86A
==========

Overall merit
---
4. Weak accept

Paper summary

The paper considers extension protocols for Byzantine agreement (BA) and broadcast (BB) reducing protocols for long input messages to protocols for short messages with the goal to minimize the dependency of the overall communication complexity on the message size. The protocols assume a trusted cryptographic setup (similar to a PKI).

The paper proposes improvements over previous protocols and observes some lower bounds for different adversarial threshold cases and both, the synchronous and the asynchronous network model.

The protocols nicely combine previous building blocks to achieve the results. The paper is well-written.

Comments for the author(s)

Some details:
- The trusted setup is based on multisignatures and an accumulator. For completeness, it would be nice to see a short discussion of minimal cryptographic standard assumptions
under which the given protocols are achievable.
- Table 1: although [9] does not qualify as an extension protocol, it would still be telling (and fair) to have a comparison to [9].
- Proof of Theorem 7: Steps 1 and 4 seem to be overestimated.

* *

Review #86B
==========

Overall merit
---
4. Weak accept

Paper summary

In the quest of the best and most performant algorithms to solve Byzantine agreement and Byzantine broadcast, two of the most fundamental building blocks in distributed computing, the biggest effort was focused on to the basic case where the exchanged value are minimal (1 bit). The role of extension algorithms aim is to use the basic elementary blocks (for values of size 1) to solve the same problems for l bits. A simple repetition of the basic block l times leads to a bit/communication complexity that is l times bigger.

Several papers try to design more sophisticated extensions that can "multiplex some synchronization in order to get a complexity as close as possible to the optimal Omega(nl+n^2) as n^2 is known to be optimal for both BB and BA and nl is obviously a lower bound as a value (l bits) has to reach all the parties.

Not that much papers studied this problem. The present one has the merit to sum up all the results and fill all the grades (synch/asynch, BB/BA, w/wo authnetication). Each time either a first extension protocol is proposed or a if one already exists, better one is proposed.

Comments for the author(s)

This paper completed all the grade of Table 1. It is is technically sound even though the writing and structuring of the paper does not always make the reading easy. This paper needs some effort from the reader.

The proposed algorithms are not totally original, but they are quiet simple and allow to make the point.

There is an important point that has not been discussed in the paper: time complexity. Indeed, the authors try to minimize the size of exchanged data, but say noting about the number of communication steps needed for executing

the extension protocol. For example, we know that synchronous consensus needs in the worst case O(n) communication steps (rounds) to converge for the best algorithm (for a linear number of faulty processes). How about the proposed algorithms? Of course if the basic building block needs to pay the n rounds, one can expect the extension protocol to have a constant time complexity! Indeed if the extension protocol has a time complexity linear in n, this means that the stacking of the protocols (extension+basic BB and BA building blocks) will be quadratic in n which contradicts the affirmation of the authors to fit modern applications such as blockchains. The authors cannot say that this is out of the scope of the paper. It is important to say something about this even if it is not very positive. These are two facets of the same coin.

In a same vein, I appreciate the discussion in Section 4.3 on the optimality of the proposed result. They open new research direction.

* *

Review #86C
==========

Overall merit
---
3. Weak reject

Paper summary

The paper explores extension protocols for Byzantine agreement (BA) and Byzantine Broadcast (BB), first in networks with synchrony and secondly also with asynchrony. Such extension protocols communicate (for broadcast or agreement) large messages by relying on primitives for the same problem but with shorter messages. By necessity, the reductions are cryptographic (not unconditionally secure).

Comments for the author(s)

The paper achieves improvements in a number of cases, mostly by applying known techniques such as erasure-coding and cryptographic authentication, but in combination with somewhat novel protocol constructions. One new element not seen elsewhere directly is the use of cryptographic accumulators.

The results appear mildly interesting, and these ideas should eventually be published. The current paper, however, has mistakes and gaps and must not be accepted. It seems infeasible to correct this during a short period.

For instance, the definition of the accumulator is wrong. Lemma 3 (which stands for a proper definition) allows the adversary to produce (ak, D, z, d', w'). But this is trivial to break, as a little thought shows, when one would implement the abstract accumulator with the Strong-RSA based accumulator construction. Proper cryptographic definitions appear, e.g., in a manuscript by Fazio and Nicolosi (Cryptographic Accumulators: Definitions, Constructions and Applications) or in a paper by Derler et al. (Revisiting Cryptographic Accumulators...). The analysis of the scheme should be constructed in a more formal way, such that issues like these become evident.

For protocol n/2-BA (Fig.2), when the 1-bit agreement outputs 0, it means simply that not all honest parties have input 1 to this agreement oracle. This follows from the (strong) validity used here. But it may be that only one honest party has z_i = z from the step before and this caused the binary agreement to return 1. Then the conclusion in the text below the figure "if above agreement outputs 0, then no honest party has a message corresponding to an accumulation value..." appears wrong.

The lower bound in theorem 8 is not precise. Impossibility results must be formulated carefully to have a meaning. Some of the cited lower bounds for BA hold only for deterministic protocols. But since the presented definitions use probabilistic conditions, the connection is missing, since A() and B() are probabilistic protocols elsewhere in the paper.

In practice, the problem with using accumulators like this is key generation. There needs to be a trusted setup that

goes beyond a PKI because of the parameter generation for the accumulator public key, A PKI only authenticates the public keys of the parties, this seems much weaker in practice.

* *

Review #86D
==========

Overall merit
---
5. Accept

Paper summary

The paper studies the problem of Byzantine Agreement (BA) and Broadcast (BB) extension in various settings and corruption thresholds: strong honest majority, honest majority, (a weakening of) dishonest majority. Extension protocols are protocols for messages of any length while only calling the underlying seed protocol for one bit a small number of times where the goal is to extend the domain of BA and BB in a non-trivial manner that achieves better communication complexity than the trivial approach of running many binary BAs or BBs in parallel. For n parties, the best possible solution to the extension problem given a BA/BB oracle for small number of bits that can be called, is n*\ell communication complexity. This is as opposed to n^2*\ell given by the trivial approach. While there exist optimal communication complexity extension protocols in literature, the presented protocols provide improvements with respect of state of the art in terms of lower order terms and the length of message for which optimality is achieved.

Comments for the author(s)

The paper studies n-party BB/BA protocols of long \ell-bit strings with asymptotically optimal communication complexity. The paper considers the settings settings: (1) t<n/2, (2) t< (1-\eps) n for constant \eps >0. The length of the message \ell at which optimality is achieved is also smaller than existing protocols resulting in O(\ell n) complexity protocols for broader range of input sizes.

The improved protocol for the authenticated synchronous case t<n/2 setting lowers the size of \ell to achieve an asymptotic behaviour of n*\ell communication by a factor of n.

The synchronous BC extension protocol for t<(1-\eps)n (for eps<1) improves upon the state of the art by a factor of n^3*log(n). This protocol works for constant-fraction dishonest majority (but not for t arbitrarily close to n. See my remark below for related comments.)

The paper also shows how to transfer the techniques to the asynchronous regime.

The key idea that leads to improvement in all protocols is the use of tools from coding theory; in particular erasure codes and a cryptographic accumulator. An accumulator allows a constant-sized representation of set of elements together with efficient proofs of membership.

The first protocol is a computationally secure BA for t<n/2 with communication complexity O(\ell n + A(k) + k n^2). The previous best in this setting achieved communication complexity O(\ell n + B(k) + A(1) + k n^2) where A(m) is the communication cost of m-bit BA oracle, and B(m) is the communication cost of m-bit BB oracle. The idea here is for each party to encode its message using Reed-Solomon code, distribute the accumulation value of the coded values, and run an instance of k-bit BA with the accumulation value as the input.

The second protocol is a computational BB protocol for t< (1-\eps) n with communication complexity O(\ell n + B(k) + k n^2 + n^3). This also beats the previous best communication complexity of O(\ell n + B(nk) + n^2), at the cost of reduced corruption threshold compared to t<n. The sender encodes the message and computes the accumulation value of the coded values. A k-bit BB oracle is used to broadcast the accumulation value. Now, in a spirit similar to previous works, the protocol runs in iterations upto t+1, in each round trying to reconstruct the correct message. Here,

the protocol does some careful budgeting at each step to ensure some steps are executed only once. This is crucial to argue communication complexity.

The protocol for t<n/3 achieves communication complexity O(n^3) compared to O(n^4) of state-of-the-art. The starting point is the protocol of [17,18]. The observation that leads to the improvement is that a step in the protocol that had all parties broadcast a vector can instead be replaced with parties ending it to each other via point-to-point channels and address the resulting potential inconsistency.

The downside is the price to pay for the improvements, which is the assumption of a trusted setup. For the suggested bilinear accumulator, if the setup is compromised, there exists an explicit trapdoor that can be used to violate the accumulator property. This could result in breaking the BB properties.

Overall, I think that the paper studies an important question. Extension protocols are useful to send long strings rather than bits in many use cases of broadcast and BA. While the results seem incremental (and two of the protocols build on techniques of [17,18]), I think they are useful, nevertheless. The techniques used are interesting. The idea of using cryptographic accumulators to compress communication is very nice.

I have some comments below, but I think they are not too serious and I recommend acceptance.

-- The protocol in section 5 is for t< (1-\eps) n for constant \eps. The construction uses RS codes with n the number of parties, and b the number of honest parties. b is set to be equal to n − t.
When t < n, it appears like there will not be enough codewords for correct reconstruction. But the proof does not seem to use the fact that t < (1-\eps)n. If I understand correctly, the claim is that security holds even when t<n, but the claimed communication complexity depends on t being < (1-\eps)n. It is not clear how, since b depends on (1-\eps)n.

-- The lower bound in 4.3 is rather unclear to me. It also notes that further improvements to the upper bound to match the lower bound seem challenging. However, the Dolev-Reischuk bound of n^2 messages does not apply to randomized protocols. So I don't see why doing better than o(kn^2) should have to "follow a very particular paradigm".

-- There exists a straight-forward reduction from BA to BB with the same complexity. So what is the point of the alternative construction in Fig. 6? It doesn't seem to be better in complexity than one obtained using the standard reduction..