Byzantine k-Anonymous Broadcast in O(Nf^2) Messages

Bryan Turner
bryan.turner@pobox.com
August, 2006

Abstract

Anonymous Broadcast protocols based on Dining Cryptographers are inefficient in message complexity, requiring O(N^2) messages per round. We analyze a k-Anonymous protocol which remains live against Byzantine adversaries. We achieve message complexity of O(Nf^2) messages per round against f Byzantine adversaries.

Introduction

The Dining Cryptographers problem was introduced by David Chaum as a technique for sender and recipient untraceability [1]. The original formulation requires messages quadratic in the number of participants. k-Anonymous protocols compartmentalize groups of participants to reduce the message complexity.

The Byzantine Generals problem was named by Leslie Lamport [2] to describe the problem of arriving at consensus amidst the presence of traitors. The lower bounds for participants in a network exchanging non-verifiable messages is 3*f*+1, where *f* is the number of traitors [2].

In [3], we introduced a technique for implementing Dining Cryptographers in the presence of Byzantine adversaries. We extend our previous work to reduce message complexity and extend the analysis of adversarial tactics.

Network Model

We operate in a network where each participant may exchange secure, authenticated messages with any other. Messages sent are never lost or corrupted, and all messages arrive in a timely fashion. While this model is overly simplistic, we believe the techniques presented here may be extended to more realistic models without loss of generality.

Adversary Model

We allow an omniscient adversary to control some participants in the network, but may not alter communications in which it is not the sender or receiver. In all other respects, the adversary may act arbitrarily; sending gibberish, misleading or incorrect messages, and non-participation.

We assume the adversary may control strictly less than 1/3 of the participants, as defined by the lower bounds for Byzantine adversaries exchanging non-verifiable messages [2].

Linearly Independent Groups

In [3] we describe a technique which divides a group of N participants into N sub-groups of (N-1) participants such that each participant is excluded from some sub-group. The linearly-independent nature of this division is leveraged to reveal the identity of the byzantine adversary. The sub-group from which the adversary is excluded is unaffected by his behavior.

We also introduced an anonymous reservation protocol, which may be generalized to allow arbitrarily small chance of collision in any given round. Header bits are appended to the front of the protocol messages and each round participants select one bit at random. Collisions occur if two participants reserve the same bit. Without loss of generality, the number of header bits may be selected to be arbitrarily large to afford a range of collision rates, as needed by the application.

Analysis of Adversary Tactics

Jamming

Jamming occurs when a participant broadcasts in a slot which it did not reserve. An adversary which performs jamming on all sub-channels is trivially revealed and ejected from the anonymity group.

Selective Jamming

An adversary which jams a single sub-channel reveals his presence but remains anonymous within that sub-channel. While this does not cause complete failure of the protocol, it is inefficient for the honest members which must continue to bear the burden of communication for the useless sub-channel.

Selective Non-Participation

An adversary which chooses not to participate in some message exchanges is equivalent to one which sends gibberish for those exchanges. His presence is revealed to the honest members, but he remains anonymous within the sub-channel.

Targeting Reservations

Adversaries which target the reservation bits achieve no advantage. The number of expected reservations is known and any additional reservations will be immediately recognized by honest members. This case is equivalent to selective jamming.

Analysis of Message Exchanges

The standard protocol for Secure Multiparty Sums requires two phases of messages broadcast to every other participant. This is quadratic in the number of participants, requiring a total of 2N(N-1) messages per round.

The techniques in [3] enable equivalent message complexity over the number of sub-channels in the group. While efficient in achieving Byzantine tolerance, it remains quadratic in the number of participants.

In order to reduce the message complexity, we first examine a complete message exchange for the 4-party scenario and analyze the results.

4-Party Message Exchange

Participants A, B, C, D join a Byzantine k-Anonymous group, and form 4 sub-channels containing 3 participants each, as illustrated in figure 1. Each sub-channel proceeds as independent multiparty sums.

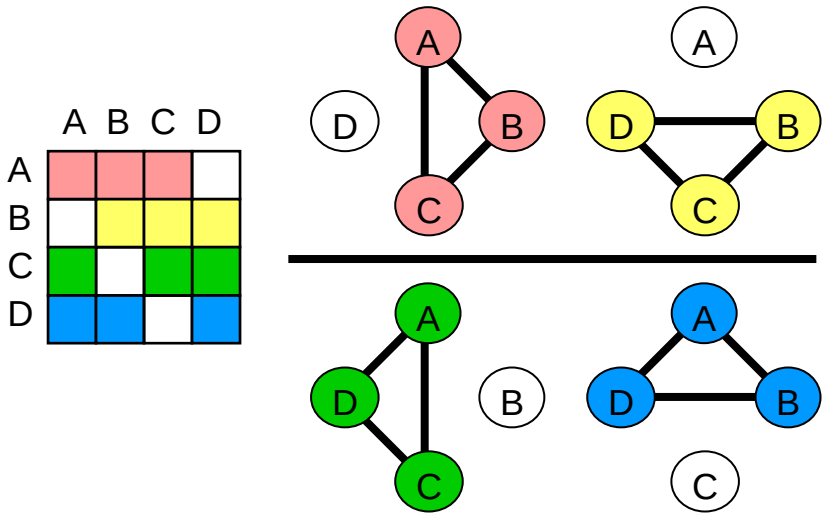

Figure 1. Left: Connection matrix for 4-party scenario. Right: Connection graph.

Let us examine the message exchanges for a single sub-channel in detail. In Phase 1, each participant selects a message to broadcast and generates random shares which combine to reveal the message. Each participant sends one share to each other participant, and receives a share in return (Figure 2).

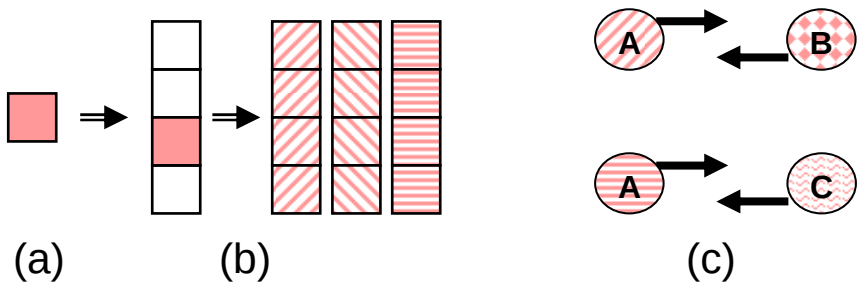

Figure 2; a) Select message; b) Divide into shares; c) Exchange shares.

In Phase 2, these shares are combined into a partial sum which is broadcast to all other members. Finally, the partial sums are combined and the protocol output is revealed (Figure 3).

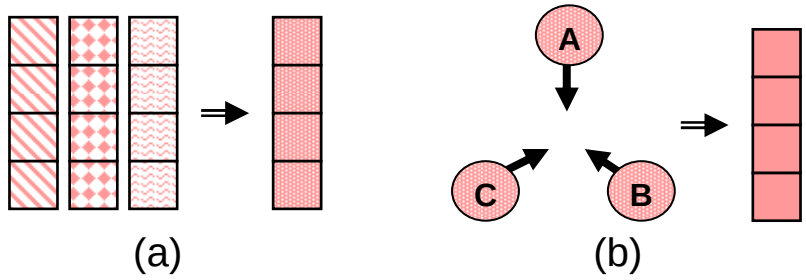

Figure 3; a) Combine shares into partial sum; b) Broadcast partial sums and compute output.

Eliminating Phase 1 Messages

Without loss of security, the messages exchanged in Phase 1 may be replaced by a single setup round in which seeds for cryptographically-secure random number generators are exchanged. Phase 1 messages are computed by all parties using the knowledge of the generator held by the intended recipients. This allows correctly calculating the cyphertext for Phase 1, without the need to exchange random strings.

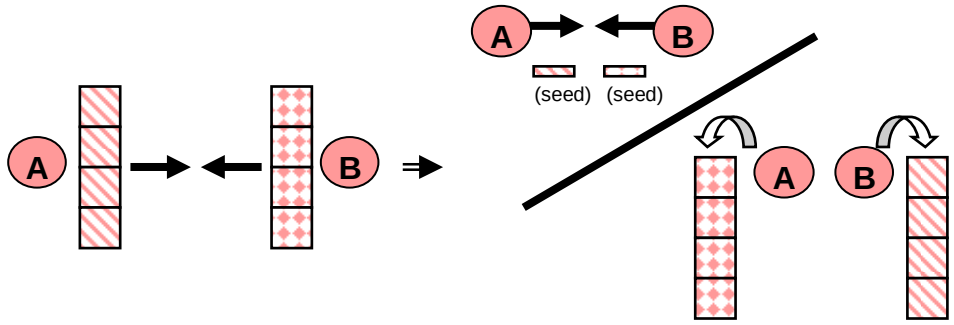

Figure 4: Eliminating Phase 1 Messages

The security of the protocol now relies on the security of the random number generator. However, this is a small price to pay for cutting the message complexity by half.

Coordinator-Assisted Broadcast[1]

Intuitively, if all participants were honest, one could be elected as a coordinator. The coordinator would receive the complete set of Phase 2 messages (N-1), combine them, then broadcast the result (N-1), for a total of 2(N-1) messages.

In our model, the adversary controls some fraction of the members. If enough coordinators are elected such that at least 1 honest participant is included, then the protocol would guarantee progress. Given f traitors in a group, a total of (f+1) participants must be elected coordinator to guarantee at least 1 honest coordinator.

Phase 2 becomes a two-part exchange. Each non-coordinator (N-f-1) sends their Phase 2 message to each coordinator (f+1). Also, the coordinators (f+1) send a message to each other coordinator (f). The coordinators combine all phase 2 messages, then re-broadcast. Since each coordinator is privy to the final result, coordinators do not need to learn each others results. This leaves each coordinator (f+1) sending a message to each non-coordinator (N-f-1).

The total exchange is 2(N-f-1)(f+1)+(f)(f+1) messages, achieving O(Nf^2).

Special Case: Reducing Message Size

In a long running anonymity group it is likely that one or more participants will abstain from broadcasting in some rounds. The naïve protocol requires a complete set of slot exchanges only to find some message slots empty.

The anonymous reservation protocol allows a simple optimization in these cases - only exchange as many slots as have been reserved. Nominally, in an N-participant group, each sub-channel will exchange messages of size O(N-1). In cases where fewer slots have been reserved, all participants of the sub-channel simply send smaller messages, corresponding to the number of reserved slots.

Practical Implementation Considerations

Many small details remain to be worked out before practical deployment of these algorithms. Specifically; choosing appropriate group sizes, selecting coordinators, and how much anonymity to aim for.

Coordinators may be fixed as part of the connection matrix, such that each sub-channel's coordinators are predetermined for any given group size. This is easily seen in the example connection matrices (Figure 5).

[1] In a byzantine group, this section's N is the number of participants in each sub-group, not the full group size.

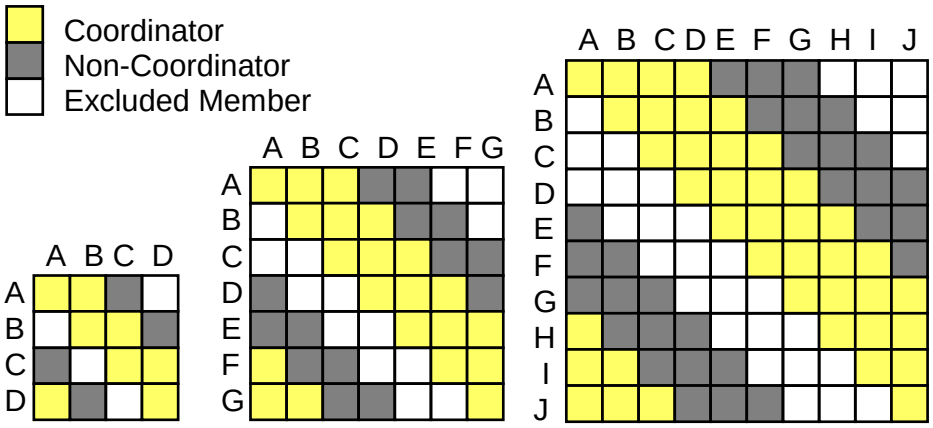

Figure 5: Connection matrix, with sub-channel coordinators marked.

The choice of anonymity level is entirely arbitrary, however some lower bounds are a good place to start. In legal cases within the US, 2-anonymity is enough to establish "reasonable doubt" in criminal trials, while 3-anonymity is enough to invalidate civil charges [4]. It is therefore suggested that a lower bound of 3-anonymity be used.

Given a large pool of participants, and an adversary which may control strictly less than 1/3 of them, we may use the formula from [4] to calculate the group size, such that any random subset of M participants will contain at least k honest members.

k = Desired level of k-anonymity
M = Group size
B = Fraction of participants adversary may control

$M = 2k / (1 - B)$

Substituting k=3, B=1/3, we obtain N=9. Since we desire byzantine failure tolerance, we must select the smallest f such that $M < 3f+1$. This occurs with f=3, and gives us a final byzantine group size N=10, with 3-anonymity, and security against 3 traitors (Figure 5).

Finally, a message generated in an anonymous broadcast group may require delivery to a member of another broadcast group. As with previous work [4,5], we adopt the technique of including a destination group identifier in the output message slot. Each member of the local broadcast group sends messages targeted to a remote group to every member of the remote broadcast group.

Considering the analysis in prior sections, a simple optimization of this inter-group exchange involves sub-group coordinators (f+1) sending each remotely-targeted message of their sub-group to the (f+1) coordinators of the remote group, who in-turn broadcast to non-coordinators of their group (N-f-1). For a total exchange of $(f+1)^2+(N-f-1)$ messages per round.

Conclusion

We have analyzed a k-Anonymous Broadcast Protocol secure against Byzantine Adversaries. The analysis and security assumptions have revealed techniques for reducing the message complexity of the protocol, achieving $O(Nf^2)$ messages per round. This scaling is significantly better than existing practice and brings these techniques within practical application.

References:

[1] **The Dining Cryptographers Problem**
David Chaum
http://www.ece.cmu.edu/~adrian/731-sp04/readings/dcnets.html

[2] **The Byzantine Generals Problem**
L. Lamport, et. al.
http://research.microsoft.com/users/lamport/pubs/pubs.html#byz

[3] **Efficient Byzantine k-Anonymous Broadcast**
Bryan Turner
http://brturn.googlepages.com

[4] **k-Anonymous Message Transmission**
Luis von Ahn, et. al.
http://crypto.stanford.edu/~abortz/work/k-anon-final.html

[5] **A New k-Anonymous Message Transmission Protocol**
Gang Yao, Dengguo Feng
http://dasan.sejong.ac.kr/~wisa04/ppt/9A2.pdf