Broadcast Using Certified Propagation Algorithm in Presence of Byzantine Faults *

Lewis Tseng[1,3], Nitin Vaidya[2,3], and Vartika Bhandari[4]

[1] Department of Computer Science,
[2] Department of Electrical and Computer Engineering,
[3] Coordinated Science Laboratory
University of Illinois at Urbana-Champaign,
and [4] Google, Inc.

Email: {ltseng3, nhv}@illinois.edu, and vartika@google.com

Technical Report

September 20, 2012

Abstract

We explore the correctness of the Certified Propagation Algorithm (CPA) [5, 1, 7, 4] in solving broadcast with locally bounded Byzantine faults. CPA allows the nodes to use only local information regarding the network topology. We provide a *tight* necessary and sufficient condition on the network topology for the correctness of CPA. We also present some simple extensions of this result.

*This research is supported in part by Army Research Office grant W-911-NF-0710287. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies or the U.S. government.

1 Introduction

In this work, we explore fault-tolerant broadcast with locally bounded Byzantine faults in synchronous point-to-point networks. We assume a *f-locally bounded model*, in which at most f Byzantine faults occur in the neighborhood of every *fault-free* node [5]. In particular, we are interested in the necessary and sufficient condition on the network topology for the correctness of the Certified Propagation Algorithm (CPA) – the CPA algorithm has been analyzed in prior work [5, 1, 7, 4].

Problem Formulation. Consider a network of n nodes. One node in the network, called the *source* (s), is given an initial input, which the source node needs to transmit to all the other nodes. The source s is assumed to be *fault-free*. We say that CPA is *correct*, if it satisfies the following properties, where x_s denotes the input at source node s:

- **Termination:** every fault-free node i eventually decides on an output value y_i.
- **Validity:** for every fault-free node i, its output value y_i equals the fault-free source's input, i.e., $y_i = x_s$.

Related Work: Several researchers have addressed problems similar to the above problem. [5] studied the problem in an infinite grid. [1] developed a sufficient condition in the context of arbitrary topologies, but the sufficient condition is not necessary. [7] provided necessary and sufficient conditions, but the two conditions are not identical (not tight). [4] provided another condition that can approximate (within a factor of 2) the largest f for which CPA is correct in a given graph.

2 System Model

The system is assumed to be *synchronous*. The synchronous communication network consisting of n nodes including source node s is modeled as a simple *directed* graph $G(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of n nodes, and $\mathcal{E}$ is the set of directed edges between the nodes in $\mathcal{V}$. We assume that $n \geq 2$, since the problem for $n = 1$ is trivial. Node i can transmit messages to another node j if and only if the directed edge (i, j) is in $\mathcal{E}$. Each node can transmit messages to itself as well; however, for convenience, we exclude self-loops from set $\mathcal{E}$. That is, $(i, i) \notin \mathcal{E}$ for $i \in \mathcal{V}$. All the links (i.e., communication channels) are assumed to be reliable. With a slight abuse of terminology, we will use the terms *edge* and *link* interchangeably.

For each node i, let N_i^- be the set of nodes from which i has incoming edges. That is, $N_i^- = \{ j \mid (j, i) \in \mathcal{E} \}$. Similarly, define N_i^+ as the set of nodes to which node i has outgoing edges. That is, $N_i^+ = \{ j \mid (i, j) \in \mathcal{E} \}$. Nodes in N_i^- and N_i^+ are, respectively, said to be incoming and outgoing neighbors of node i. Since we exclude self-loops from $\mathcal{E}$, $i \notin N_i^-$ and $i \notin N_i^+$. However, we note again that each node can indeed transmit messages to itself.

We consider the f-local fault model, with at most f incoming neighbors of any fault-free node becoming Byzantine faulty.

3 Certified Propagation Algorithm (CPA)

In this section, we describe the Certified Propagation Algorithm (CPA) from [5] formally. Note that the faulty nodes may deviate from this specification.

Source node s commits to its input x_s at the start of the algorithm, i.e., sets its output equal to x_s. The source node is said to have committed to x_s in round 0. The algorithm for each round r, $r > 0$, is as follows:

1. Each node that commits in round $r-1$ to some value x, transmits message x to all its outgoing neighbors, and then terminates.
2. If any node receives message x directly from source s, it commits to output x.
3. Through round r, if a node has received messages containing value x from at least $f+1$ distinct incoming neighbors, then it commits to output x.

3.1 The Necessary Condition

For CPA to be correct, the network graph $G(\mathcal{V}, \mathcal{E})$ must satisfy the necessary condition proved in this section. We borrow two relations $\Rightarrow$ and $\not\Rightarrow$ from our previous paper [9].

Definition 1 *For non-empty disjoint sets of nodes A and B,*

- *$A \Rightarrow B$ iff there exists a node $v \in B$ that has at least $f+1$ distinct incoming neighbors in A, i.e., $|N_v^- \cap A| > f$.*
- *$A \not\Rightarrow B$ iff $A \Rightarrow B$ is not true.*

Definition 2 *Set $F \subseteq \mathcal{V}$ is said to be a feasible f-local fault set, if for each node $v \notin F$, F contains at most f incoming neighbors of node v. That is, for every $v \in \mathcal{V} - F, |N_v^- \cap F| \leq f$.*

Theorem 1 *Suppose that CPA is correct in graph $G(\mathcal{V}, \mathcal{E})$ under the f-local fault model. Let sets F, L, R form a partition[1] of $\mathcal{V}$, such that (i) source $s \in L$, (ii) R is non-empty, and (iii) F is a feasible f-local fault set, then*

- *$L \Rightarrow R$, or*
- *R contains an outgoing neighbor of s, i.e., $N_s^+ \cap R \neq \emptyset$.*

Proof: Consider any partition F, L, R such that $s \in L$, R is non-empty, and F is a feasible f-local fault set. Suppose that the input at s is x_s. Consider any single execution of the CPA algorithm such that the nodes in F behave as if they have crashed.

By assumption, CPA is correct in the given network and faulty behavior. Thus, all the fault-free nodes commit their output to x_s. Given any execution of CPA under the above behavior by the nodes in F, consider a node $v \in R$ that commits its output to x_s, such that no other node in R

[1] Sets $X_1, X_2, X_3, ..., X_p$ are said to form a partition of set X provided that (i) $\cup_{1 \leq i \leq p} X_i = X$, and (ii) $X_i \cap X_j = \Phi$ if $i \neq j$.

commits its output in a round prior to v's commit. Due to the correctness of CPA, such a node v must exist. For v to be able to commit, either it should receive the message x_s directly from s, or node v must have $f+1$ distinct incoming neighbors that have committed to x_s. By assumption, nodes that have committed to x_s prior to v must all be in set L. Thus, either $(v, s) \in \mathcal{E}$, or node v has at least $f+1$ distinct incoming neighbors in set L. □

3.2 Sufficiency

We now show that the condition in Theorem 1 is also sufficient.

Theorem 2 *If $G(\mathcal{V}, \mathcal{E})$ satisfies the condition in Theorem 1, then CPA is correct in $G(\mathcal{V}, \mathcal{E})$ under the f-local fault model.*

Proof: Suppose that $G(\mathcal{V}, \mathcal{E})$ satisfies the condition in Theorem 1. Let F' be the set of faulty nodes. By assumption, F' is a feasible local fault set. Let x_s be the input at source node s. We will show that, (i) fault-free nodes do not commit to any value other than x_s, and, (ii) until all the fault-free nodes have committed, in each round of CPA, at least one additional fault-free node commits to value x_s. The proof is by induction.

Induction basis: Source node s commits in round 0 to output equal to its input x_s. No other fault-free nodes commit in round 0.

Induction: Suppose that L is the set of fault-free nodes that have committed to x_s through round r, $r \geq 0$. Thus, $s \in L$. Define $R = \mathcal{V} - L - F'$. If $R = \emptyset$, then the proof is complete. Let us now assume that $R \neq \emptyset$.

Now consider round $r+1$.

Consider any fault-free node u that has not committed prior to round $r+1$ (i.e., $u \in R$). All the nodes in L have committed to x_s by the end of round r. Thus, in round $r+1$ or earlier, node u may receive messages containing values different from x_s only from nodes in F'. Since there are at most f incoming neighbors of u in F', node u cannot commit to any value different from x_s in round $r+1$.

By the condition in Theorem 1, there exists a node w in R such that (i) node w has an incoming link from s, or (ii) node w has incoming links from $f+1$ nodes in L. In case (i), node w will commit to x_s on receiving x_s from node s in round $r+1$ (in fact, $r+1$ in this case must be 1). In case (ii), since all the nodes in L from whom node w has incoming links have committed to x_s (by definition of L), node w will be able to commit to x_s after receiving messages from at least $f+1$ incoming neighbors in L, since all nodes in L have committed to x_s by the end of round r by the definition of L.[2] Thus, node w will commit to x_s in round $r+1$.

This completes the proof. □

4 Discussion

This section discusses some extensions on the result presented above.

[2]Since node w did not commit prior to round $r+1$, it follows that at least one node in L must have committed in round r.

4.1 Generalized Fault Model

In this subsection, we briefly discuss how to extend the above results under a generalized fault model. The generalized fault model [8] is characterized using *fault domain* $\mathcal{F} \subseteq 2^{\mathcal{V}}$ as follows: Nodes in set F may fail during an execution of the algorithm only if there exists set $F^* \in \mathcal{F}$ such that $F \subseteq F^*$. Set F is then said to be a *feasible* fault set.

Definition 3 *Set $F \subseteq \mathcal{V}$ is said to be a feasible fault set, if there exists $F^* \in \mathcal{F}$ such that $F \subseteq F^*$.*

Please refer to our previous work [8] for more discussion on generalized fault model.

For a set of nodes B, define $N^-(B) = \{i \mid (i,j) \in \mathcal{E},\ i \notin B,\ j \in B\}$, the set of incoming neighbors of B.

Definition 4 *Given $\mathcal{F}$, for disjoint sets of nodes A and B, where B is non-empty.*

- $A \stackrel{g}{\Rightarrow} B$ *iff for every* $F^* \in \mathcal{F}$, $N^-(B) \cap A \not\subseteq F^*$.
- $A \stackrel{g}{\not\Rightarrow} B$ *iff* $A \stackrel{g}{\Rightarrow} B$ *is* not *true.*

Under the generalized fault model, step 3 of CPA needs to be modified as follows. Let us call the modified algorithm CPA-G.

3. Through round r, if a node has received messages containing value x from a set M, where M is not a feasible fault set, then the node commits to value x.

It is easy to show that a modified version of Theorem 1 stated below holds for the generalized fault model.

Theorem 3 *Suppose that CPA-G is correct in graph $G(\mathcal{V}, \mathcal{E})$ under the generalized fault model. Let sets F, L, R form a partition of $\mathcal{V}$, such that source (i) $s \in L$, (ii) R is non-empty, and (iii) F is a feasible fault set, then*

- $L \Rightarrow R$, *or*
- R *contains an outgoing neighbor of s, i.e., $N_s^+ \cap R \neq \emptyset$.*

4.2 Broadcast Channel

We have so far assumed that the underlying network is a point-to-point network. The results, however, can be easily extended to the *broadcast* or *radio model* [5, 1] as well. In the *broadcast model*, when a node transmits a value, all of its outgoing neighbors receive this value identically. Thus, no node can transmit mismatching values to different outgoing neighbors. Then, it is easy to see that the same condition as the point-to-point network can be shown to be necessary and sufficient for of CPA under the broadcast model as well.

Now consider the following variation of the CPA algorithm: if the outgoing neighbors of source s do not receive a message from s in round 1, the message value is assumed to be some default value. With this modification, the condition in Theorem 1 can also be shown to be necessary and sufficient to perform Byzantine Broadcast [6] under the broadcast model, while satisfying the following three conditions (allowing s to be faulty):

- **Termination:** every fault-free node i eventually decides on an output value y_i.
- **Agreement:** the output values of all the fault-free nodes are equal, i.e., there exists y such that, for every fault-free node $i, y_i = y$.
- **Validity:** if the source node is fault-free, then for every fault-free node i, the output value equals the source's input, i.e., $y = x_s$.

The proof follows from the proof of Theorem 1 and the observation that all the outgoing neighbors of s receive identical value from s, which equals its input x_s when s is fault-free.

4.3 Asynchronous Network

In our analysis so far, we have assumed that the system is synchronous. For a point-to-point network with fault-free source s, it should be easy to see that the condition in Theorem 1 is also necessary and sufficient to achieve agreement using a CPA-like under the asynchronous model [2] as well. In this case, the algorithm may not proceed in rounds, but a node still commits to value x either on receiving the value directly from s, or from $f + 1$ nodes.

This claim may seem to contradict the FLP result [3]. However, our claim assumes that the source node is fault-free, unlike [3].

5 Conclusion

In this paper, we explore broadcast using the CPA algorithm in presence of Byzantine faults. In particular, we provide a *tight* necessary and sufficient condition for the correctness of CPA.

References

[1] V. Bhandari and N. H. Vaidya. On reliable broadcast in a radio network: A simplified characterization. Technical report, University of Illinois at Urbana-Champaign, 2005.

[2] D. Dolev, N. A. Lynch, S. S. Pinter, E. W. Stark, and W. E. Weihl. Reaching approximate agreement in the presence of faults. *J. ACM*, 33:499–516, May 1986.

[3] M. J. Fischer, N. A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *J. ACM*, 32:374–382, April 1985.

[4] A. Ichimura and M. Shigeno. A new parameter for a broadcast algorithm with locally bounded byzantine faults. *Inf. Process. Lett.*, 110(12-13):514–517, June 2010.

[5] C.-Y. Koo. Broadcast in radio networks tolerating byzantine adversarial behavior. In *Proc. 23rd Annual ACM Symp. on Principles of Distributed COmputing (PODC' 04)*, 2004.

[6] M. Pease, R. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *J. ACM*, 27(2):228–234, Apr. 1980.

[7] A. Pelc and D. Peleg. Broadcasting with locally bounded byzantine faults. *Inf. Process. Lett.*, 93(3):109–115, Feb. 2005.

[8] L. Tseng and N. H. Vaidya. Iterative approximate byzantine consensus under a generalized fault model. Technical report, CSL, UIUC, 2012.

[9] N. H. Vaidya, L. Tseng, and G. Liang. Iterative approximate byzantine consensus in arbitrary directed graphs. In *Proceedings of the thirty-first annual ACM symposium on Principles of distributed computing*, PODC '12. ACM, 2012.