Write Markers for Probabilistic Quorum Systems

Michael G. Merideth and Michael K. Reiter

November 2007
CMU-ISRI-07-118

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Also appears as Computer Science Technical Report CMU-CS-07-165

Abstract

Probabilistic quorum systems can tolerate a larger fraction of faults than can traditional (strict) quorum systems, while guaranteeing consistency with an arbitrarily high probability for a system with enough replicas. However, they are hampered in that, like strict quorum systems, they allow for Byzantine-faulty servers to collude maximally to provide incorrect values to clients. We present a technique based on *write markers* that prevents faulty servers from colluding unless they are all also selected to be participants in the same update operations. We show that write markers increase the maximum fraction of faults that can be tolerated to $b < n/2$ from $b < n/2.62$, where n is the total number of replicas, for probabilistic masking quorum systems (compared with $b < n/4$ for strict masking quorum systems) and to $b < n/2.62$ from $b < n/3.15$ for probabilistic opaque quorum systems (compared with $b < n/5$ for strict opaque quorum systems). In addition, with write markers, probabilistic masking quorums no longer require write quorums of large or maximal size in order to tolerate the maximum fraction of faults. We describe an implementation of write markers that is effective even if Byzantine clients collude with faulty servers.

This work was partially supported by NSF grant CCF-0424422.

Keywords: Distributed systems, Byzantine fault tolerance, probabilistic quorum systems

1 Introduction

Many modern Byzantine-fault-tolerant protocols rely on Byzantine quorum systems [11] in order to tolerate the arbitrary failure of a subset of their replicas; some, e.g., [12, 4, 6], rely on masking quorum systems [11], while others, e.g., [1, 14], rely on opaque quorum systems [11]. A quorum system is simply a set of quorums (sets) of servers. Read and write operations need involve only quorums, instead of all servers. Therefore, some servers may be unaware of a recently written value. Consistency constraints dictate that all quorums intersect such that enough non-faulty servers are collectively aware of all previously written values to ensure that these values are not lost or changed.

The consistency constraints limit the maximum number of faults (b) in relation to the total number of replicas (n) that the system can tolerate. This is unfortunate because, all else being equal, it is desirable for the system to be able to tolerate a larger number of faults as such a system is therefore more resilient to faults. In addition, due to the constraints, the size of quorums is typically proportional to the number of faults that can be tolerated—quorums can be made smaller to an extent, but then the systems are restricted to tolerating fewer faults. This is again unfortunate, because, all else being equal, it is desirable for the size of quorums to be smaller, as this implies lower communication and redundant storage/computation requirements.

Probabilistic quorum systems (PQS) [13, 16] differ from traditional (strict) quorum systems in two essential ways. First, in PQS, a quorum for an operation is selected probabilistically, or, more specifically, from an access set [3] that is itself selected according to a probability distribution known as an *access strategy*. In this paper, we utilize the access strategy that selects every access set uniformly at random [16]; if an access set is the same size as a quorum, then this implies that quorums are selected uniformly at random (c.f., [13]). Second, the consistency constraints need only account for the expected intersection of quorums, where probabilities are taken with respect to the access strategy from which each access set is chosen, instead of accounting for every allowable combination of quorums. Because of this, given the access strategy, the consistency constraints are met with some probability for a given system.

PQS can guarantee consistency with what we call *compelling probability*, i.e., with a probability that the constraints are met that can be made arbitrarily high by considering a large enough n, assuming that the fraction of faults (b/n) is not too large; such a system can tolerate the same fraction—and hence a correspondingly large number—of faults. (This is described more formally in Section 3.) Application domains that could give rise to systems with large n include sensor networks and edge services. Because the consistency constraints are written for the expected case, they can be relaxed to varying degrees compared with those of strict quorum systems, yielding benefits in terms of the maximum fraction of faults and/or sizes of quorums that still guarantee consistency with compelling probability. In addition, for an access strategy in which read quorums are selected uniformly at random, the maximum fraction of faults that allows for consistency with compelling probability is independent of the size of read quorums; therefore, an arbitrarily high probability of consistency can be achieved even if read quorums are a relatively small fraction of n.

However, the benefits of probabilistic quorum systems in the Byzantine fault model [7] have been limited since, like strict quorum systems, PQS must allow for Byzantine-faulty servers to collude maximally to provide incorrect values to clients. This gives the faulty servers and clients an advantage over the non-faulty servers and clients, which are not assumed to know the identities of the other non-faulty servers or clients. One impact of this advantage is that all of the faulty servers in a read operation may vote for the same incorrect history of write operations, whereas non-faulty servers together vote for a previous write only if they were all participants in that write operation.

In this paper, we present *write markers*, a way for probabilistic quorum systems to limit the extent to which faulty servers can collude. At a high level, a write marker is a verifiable data item that is written together with the data value and verified during a read; it identifies the servers in the write access set and is valid only if the choice of this access set follows the access strategy. Since no servers outside of the write access set should be involved in the write operation, write markers can be used to prevent the faulty servers that are not selected for the write operation from colluding with those that are. As seen in Table 1, this enables us to increase the maximum fraction of faults that can be tolerated while achieving consistency with compelling probability. In addition, for masking quorum systems that use an access strategy in which every write quorum is chosen uniformly at random, this also makes this maximum fraction independent of the size of write quorums. Therefore, an arbitrarily high probability of consistency can be achieved with enough replicas while tolerating a relatively large fraction of faults, despite read and write quorums that are an arbitrarily small fraction of n.

1.1 Our Results

Our primary contributions are (i) the identification and analysis of the benefits of write markers for probabilistic quorum systems; and (ii) a proposed implementation of write markers that handles the complexities of tolerating Byzantine clients. Write markers benefit both probabilistic masking and probabilistic opaque quorum systems by increasing the maximum fraction of faults that these systems can tolerate with compelling probability. In addition, write markers allow probabilistic masking quorum systems to tolerate this fraction with compelling probability, even when write quorums are an arbitrarily small fraction of n.

Our summary below considers upper bounds on b in the case where quorums are selected uniformly at random, e.g., when quorums are the same size as access sets (which are selected uniformly at random due to the access strategy, which can be enforced with our implementation of write markers). This represents the optimal configuration for maximizing the fraction of faults that can be tolerated with compelling probability. However, our analysis later in the paper also more

Table 1: Upper bounds on number of faults (b) in terms of total replicas (n).

	write markers	probabilistic	strict
Masking	$n/2$ [here]	$n/2.62$ [16]	$n/4$ [11]
Opaque	$n/2.62$ [here]	$n/3.15$ [16]	$n/5$ [11]

generally considers cases where faulty clients have some flexibility in their choices of quorums (i.e., when access sets are larger than quorums).

Probabilistic Masking Quorums. Strict masking quorum systems can tolerate up to $b < n/4$ faults when $q = n - b$, where q is the size of each quorum [11]. Here, we show that the use of write markers allows probabilistic masking quorum systems to tolerate up to $b < n/2$ faults with compelling probability without restriction on the minimum sizes of quorums.

The previous best results are as follows. Malkhi et al. [13] showed that probabilistic masking quorum systems (without write markers) can tolerate up to $q/2$ faults with compelling probability when $q = \omega(\sqrt{n})$.[1] In the case where $q = n - b$, this requires $b < n/3$. Using other proof techniques, the maximum number of faults in this case can be improved to $b < n/2.62$ [16].

For purposes of direct comparison with write markers, we generalize these results in this paper to consider the sizes of read quorums (q_{rd}) and write quorums (q_{wt}) independently. We show that q_{rd} does not impact the number of faults that can be tolerated with compelling probability when read quorums are selected uniformly at random, and so the lower bound on the size of quorums pertains only to write quorums. However, this still means that, without write markers, probabilistic masking quorum systems require $q_{wt} = n - b$ in order to tolerate $b < n/2.62$ faults with compelling probability. As such, the use of write markers is significant for masking quorum systems in that it increases the maximum fraction of faults that can be tolerated with compelling probability to $b < n/2$, while simultaneously removing the restriction on q_{wt} in this case.

Probabilistic Opaque Quorums. Strict opaque quorum systems can tolerate up to $b < n/5$ faults when $q = n - b$, where q is the size of each quorum [11]. Here, we show that the use of write markers allows probabilistic opaque quorum systems to tolerate up to $b < n/2.62$ faults with compelling probability when $q_{wt} = n - b$, without restriction on the minimum size of read quorums.

The previous best result is that probabilistic opaque quorum systems (without write markers) can tolerate up to $b < n/3.15$ faults with compelling probability when $q_{wt} = q_{rd} = n - b$ [16].

For purposes of direct comparison with write markers, we reanalyze this result in this paper and show that q_{rd} need not impact the number of faults that can be tolerated with compelling probability, so the lower bound on the size of quorums need only pertain to write quorums. As such, the use of write markers is significant for opaque quorum systems in that it increases the maximum number of faults that can be tolerated with compelling probability to $b < n/2.62$.

2 Related Work

Probabilistic quorum systems were introduced by Malkhi et al. [13]. Lee and Welch make use of probabilistic quorum systems in randomized algorithms for distributed read-write registers [8] and shared queue data structures [9]. Previously [16], we presented probabilistic opaque quorum

[1] ω is the little-oh analog of Ω, i.e., $f(n) = \omega(g(n))$ if $f(n)/g(n) \to \infty$ as $n \to \infty$.

systems, as well as an access-restriction protocol to enforce the uniform access strategy while tolerating Byzantine-faulty clients and servers. As we describe in this paper, probabilistic masking [13] and opaque [16] quorum systems can be improved with write markers.

Another implementation of write markers was introduced by Alvisi et al. [2] for purposes different than ours. Whereas, with the goal of increasing the maximum fraction of faults that the system can tolerate with compelling probability, we use write markers to prevent some faulty servers from colluding, Alvisi et al. use write markers in order to increase accuracy in estimating the number of faults present in Byzantine quorum systems, and for identifying faulty servers that consistently return incorrect results. Because the implementation of Alvisi et al. does not prevent faulty servers from lying about the write quorums of which they are members, it cannot be used directly for our purposes. In addition, our implementation is designed to tolerate Byzantine clients, unlike theirs.

We adapt the access-restriction protocol of probabilistic opaque quorum systems [16] in order to provide a write marker protocol that tolerates Byzantine clients. The protocol forces faulty clients to follow the access strategy (i.e., to choose access sets uniformly at random); otherwise, non-faulty servers reject the operation being performed at the access set. As such, the protocol is designed to prevent faulty clients from completing operations at access sets not chosen uniformly at random. However, because faulty servers could choose to accept an operation even if the access set was not chosen at random, we extend the protocol so that non-faulty clients can also verify the choices of access sets (during reads)—thereby preventing faulty servers from lying about the operations of which they were participants.

Probabilistic dissemination quorum systems [13] assume that the authenticity of a data value can be determined by a single instance. (As such, dissemination quorum systems are limited in the types of data they can accept.) Because the faulty servers of probabilistic dissemination quorums cannot fabricate data values that conflict, such quorum systems are not improved by write markers.

3 Definitions and System Model

For explanatory purposes, we begin with a few definitions. The system consists of a universe (U) of n servers, and an arbitrary, but bounded, number of clients. There is a set $B \subset U$ that represents the b faulty servers; the composition of B is known by the faulty clients and servers, but not by the non-faulty ones. The remaining $n - b$ servers, i.e., $U \setminus B$, are non-faulty. Faulty servers and clients can behave arbitrarily (i.e., Byzantine faults [7]), but, as is typical, are computationally bound such that, e.g., they cannot subvert the cryptographic primitives used in the implementation of write markers. An access set is selected uniformly at random for each operation due to the access strategy. (The mechanism by which this is ensured is discussed later.)

Table 2 shows a terminological classification of servers in which A_{wt} and A'_{wt} are write access sets, and A_{rd} is a read access set. A write that is accepted by a server yields a *candidate* at that server. A candidate is *established* once it is accepted by all of the non-faulty servers in some write quorum. In opaque quorum systems, different non-faulty servers may have different candidates issued by concurrent writes at a given instant. (This must be prevented by the protocol if a masking quorum system is used.) Moreover, in either masking or opaque quorum systems, a faulty server may try to forge a concurrent candidate. If there are multiple concurrent candidates and one is established, the others are called *conflicting*. A server can try to *vote* for some candidate if the server is a *participant* in voting (i.e., if the server is a member of the read access set). However a server becomes *qualified* to vote for a particular candidate only if the server is a member of the write access set selected for the write operation for which it votes.

Table 2: Dynamic Classification of Servers

Class	Membership
qualified (to vote for A_{wt})	A_{wt}
qualified (to vote for A'_{wt})	A'_{wt}
non-qualified (to vote for A_{wt})	$U \setminus A_{wt}$
non-qualified (to vote for A'_{wt})	$U \setminus A'_{wt}$
participant	A_{rd}
non-faulty participant	$A_{rd} \setminus B$
faulty participant	$A_{rd} \cap B$
non-faulty qualified participant	$(A_{rd} \cap A_{wt}) \setminus B$
faulty qualified participant	$(A_{rd} \cap A_{wt}) \cap B$
non-faulty non-qualified participant	$(A_{rd} \setminus A_{wt}) \setminus B$
faulty non-qualified participant	$(A_{rd} \setminus A_{wt}) \cap B$

A *system configuration* for a probabilistic quorum system specifies an access strategy, and functions for b, a_{rd}, q_{rd}, a_{wt}, and q_{wt} (where a_{rd} and a_{wt} are the sizes of read and write access sets, respectively) in terms of n (and possibly each other), such that n is the only free variable—meaning that a numerical value for n results in a numerical value for each parameter. (E.g., these functions might specify fixed ratios of n.) A *system instance* specifies a positive integer value n and a configuration. It therefore determines the numerical values of the parameters b, a_{rd}, q_{rd}, a_{wt}, q_{wt}, as well as the probability that the constraints are met. The *error probability*, ϵ, is the probability that the constraints are not met; it is computed as a function of the system instance. Changing the value of n or the configuration that comprise the system instance will change the error probability in general. Intuitively, if a probabilistic quorum system configuration works with *compelling probability*, then we can create an instance that works with arbitrarily high probability by specifying a large enough value of n. Formally, compelling probability for a configuration implies that,

$$n \rightarrow \infty \quad \Rightarrow \quad \epsilon \rightarrow 0.$$

As we are concerned with comparisons to strict quorum systems, we restrict our attention to the maximum fraction of faults that can be tolerated with compelling probability.

We assume that faulty clients seek to maximize the error probability by following specific strategies [16]. This is a conservative assumption; a client cannot increase—but may decrease—the probability of error by failing to do so. At a high level, the strategies are as follows: a faulty client, which may be completely restricted in its choices: (i) when establishing a candidate, writes the candidate to as few non-faulty servers as possible to minimize the probability that it is observed by a non-faulty client; and (ii) writes a conflicting candidate to as many servers as will accept it (i.e., faulty servers plus, in the case of an opaque quorum system, any non-faulty server that has not accepted the established candidate) in order to maximize the probability that it is observed.

4 Analysis of Write Markers

Write markers remove the advantage enjoyed by faulty servers in strict and traditional probabilistic masking and opaque quorum systems, where any faulty participant can vote for any candidate—and therefore can collude to have an incorrect, potentially fabricated candidate chosen instead of the correct candidate. This is because, with write markers, faulty servers must be qualified for the same candidate in order to vote for it successfully (i.e., without having the client discard the votes). This aspect of write markers is summarized in Table 3, which shows the impact of write markers in terms of the abilities of faulty and non-faulty servers to vote for a given candidate.

To abstract away implementation complexities, we analyze a hypothetical operational model. We stress that this is a descriptive simplification; in Section 5, we describe an implementation with its associated complexities. In our model, a client submits an operation to the network. At this point, the network reliably chooses an access set for the operation uniformly at random; the clients and servers have no control over (or prior information about) which access set is, or will be, selected for a given operation. Furthermore, we posit that the network issues a verifiable write marker that identifies the access set it chooses for a write operation; this write marker becomes part of the candidate and cannot be forged or modified. (As such, each candidate is tied to a single access set.) The network does not deliver the candidate to servers outside of the write access set. A faulty client can instantaneously choose to restrict the delivery of the candidate further, so that some servers within the write access set do not receive the candidate (though servers are not restricted from sending the candidate to other servers).

The model guarantees the following properties:

W1. Every candidate has a write marker that identifies the access set chosen for the write;

W2. A valid write marker implies that the access set was selected according to the access strategy;

W3. Every non-faulty client can verify the validity of a write marker.

When considering a candidate, non-faulty clients and servers verify its write marker. Because of this verification, no non-faulty node will accept a vote for a candidate unless the issuing server is qualified to vote for the candidate. Since each write access set is chosen uniformly at random, the faulty servers that can vote for a candidate, i.e., the faulty qualified servers, are a random subset of the faulty servers.

Table 3: Ability of a server to vote for a given candidate: ● (traditional); ⋆ (write markers).

Type of server	Vote	Abstain
Non-faulty qualified participant	● ⋆	
Faulty qualified participant	● ⋆	
Non-faulty non-qualified participant		● ⋆
Faulty non-qualified participant	●	⋆

4.1 Consistency Constraints

In a probabilistic quorum system, there is a defined error probability for any number of faults. However, to enable consistency with compelling probability (i.e., with an error probability that can be made arbitrarily small by increasing n) there is an upper bound on b in terms of n. This is because two consistency constraints must be met in order to guarantee that operations are *observed* consistently in subsequent operations. To be more precise, we say that a write operation is observed in a read operation if it receives at least a threshold r of votes from different servers.

The constraints are as follows. First, a non-faulty client must, in expectation, observe the latest established candidate if such a candidate exists. Second, a conflicting candidate (which occurs only if there is already an established candidate for the same timestamp) is, in expectation, not observed by any client (non-faulty or faulty). Together, these two requirements also trivially imply that there is at most one established candidate at a time in expectation. Informally, because there is consistency in expectation, the probability of error goes to zero as we increase n, and so we have compelling probability.

To ensure that an established candidate is observed in expectation we require the following. Let $\mathsf{Q}_{\rm rd}$ represent a read quorum chosen uniformly at random, i.e., a random variable. (Think of this quorum as one used by a non-faulty client.) Let $\mathsf{Q}_{\rm wt}$ represent a write quorum chosen by a potentially faulty client; $\mathsf{Q}_{\rm wt}$ must be chosen from $\mathsf{A}_{\rm wt}$, an access set chosen uniformly at random. (Think of $\mathsf{Q}_{\rm wt}$ as a quorum used for an established candidate.) Then we require,

$$\mathbb{E}\left[|(\mathsf{Q}_{\rm rd} \cap \mathsf{Q}_{\rm wt}) \setminus B|\right] \geq r \tag{1}$$

The use of write markers has no impact here on $\mathbb{E}\left[|(\mathsf{Q}_{\rm rd} \cap \mathsf{Q}_{\rm wt}) \setminus B|\right]$ because faulty servers are not part of this expression. However, write markers do enable us to set r smaller, as the following shows.

Let $\mathsf{A}'_{\rm rd}$ and $\mathsf{A}'_{\rm wt}$ represent read and write access sets, respectively, chosen uniformly at random. (Think of $\mathsf{A}'_{\rm wt}$ as the access set used by a faulty client for a conflicting candidate, and of $\mathsf{A}'_{\rm rd}$ as the access set used by a faulty client for a read operation.[2])

Probabilistic Masking Quorums. In a probabilistic masking quorum system, only faulty servers may vote for a conflicting candidate (because non-faulty servers do not accept conflicting candidates). Using write markers, we require that the faulty qualified participants alone cannot produce sufficient votes for a candidate to be observed in expectation:

$$r > \mathbb{E}\left[|(\mathsf{A}'_{\rm rd} \cap \mathsf{A}'_{\rm wt}) \cap B|\right] \tag{2}$$

Contrast this with the consistency requirement for traditional probabilistic masking quorum systems [13] (adapted to consider access sets), which requires that the faulty participants (qualified or not) cannot produce sufficient votes for a candidate to be observed in expectation:

$$r > \mathbb{E}\left[|\mathsf{A}'_{\rm rd} \cap B|\right] \tag{3}$$

[2] In general, it is important for all clients to follow the access strategy even for read access sets [16], so as to enable higher-level protocols that employ repair phases within a read (e.g., [1]).

Intuitively, we can set r smaller with write markers because the right-hand side of (2) is less than the right-hand side of (3) if $a_{wt} < n$.

Probabilistic Opaque Quorums. With write markers, we have the benefit, described above for probabilistic masking quorums, in terms of the number of faulty participants that can vote for a candidate in expectation. However, opaque quorum systems must also consider the maximum number of non-faulty qualified participants that vote for the same conflicting candidate in expectation. As such, r is constrained as follows:

$$r > \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] + \mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right] \tag{4}$$

Contrast this with the consistency requirement for traditional probabilistic opaque quorums [16]:

$$r > \mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right] + \mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right] \tag{5}$$

Again, intuitively, we can set r smaller with write markers because the right-hand side of (4) is less than the right-hand side of (5) if $a_{wt} < n$.

4.2 Implied Bounds

In this subsection, we prove that probabilistic quorum systems with write markers can achieve compelling probability if b is not too large, and show that write markers allow probabilistic quorum systems to tolerate a larger b with compelling probability. We also prove that the bound on b is independent of the sizes of quorums in probabilistic masking quorums systems with write markers when access sets are no larger than quorums.

Of primary interest are Theorem 4.16 and its corollaries, which demonstrate the benefits of write markers for probabilistic masking quorum systems, and Theorem 4.21 and its corollaries, which demonstrate the benefits of write markers for probabilistic opaque quorum systems. They are based on Theorem 4.5, which presents three basic requirements that together allow a probabilistic quorum system to provide consistency with compelling probability. As a preliminary step, Theorem 4.2 provides a general tool for proving concentration of a random variable in terms of asymptotic bounds.

The following theorem is a restatement of the Molloy and Reed statement [18, p. 172] of the McDiarmid Inequality that can be used to show that a random variable computed on a series of independent permutations is concentrated about its expectation.

Theorem 4.1 ([18])**.** *Let $\mathsf{Z} = \mathbf{z}(\Pi_1, \ldots, \Pi_l)$ be a random variable that is a non-negative function of a series $\Pi_1, \ldots, \Pi_l$ of independent random variables, where each Π_i takes on a random permutation (bijection) $\pi : \{1, \ldots, |P|\} \rightarrow P$ of a finite non-empty set P. Also, for some positive constants δ and μ, let the following conditions hold (where if $\Pi_j = \pi_j$ then the* mapping $\langle i, j, m\rangle$ *indicates that $\pi_j(i) = m$):*

M1. Swapping the mappings of any two elements in a single permutation π_j (i.e., changing $\{\langle i, j, m\rangle, \langle i', j, m'\rangle\}$ to $\{\langle i', j, m\rangle, \langle i, j, m'\rangle\}$, where $i \neq i'$ and $m \neq m'$) changes the value of Z by at most δ.

M2. If $\mathsf{Z} = \mathbf{z}(\pi_1, \ldots, \pi_l) = x$, then there exists a set of at most μx distinct mappings $\{\langle i_1, j_1, m_1 \rangle, \ldots, \langle i_{\mu x}, j_{\mu x}, m_{\mu x} \rangle\}$ such that $\mathbf{z}(\pi'_1, \ldots, \pi'_l) \geq x$ for any $\pi'_1, \ldots, \pi'_l$ sharing the same set of mappings.

If $0 \leq d \leq \mathbb{E}[\mathsf{Z}]$, then:

$$\Pr(|\mathsf{Z} - \mathbb{E}[\mathsf{Z}]| \geq d + 60\delta\sqrt{\mu \mathbb{E}[\mathsf{Z}]} + 1) \leq 4/e^{\left(d^2/8\delta^2\mu\mathbb{E}[\mathsf{Z}]\right)}.$$

We simplify Theorem 4.1 to create Theorem 4.2 that deals with asymptotic bounds and that additionally assumes that $\mathbf{z}$ has finite range.

Theorem 4.2. *Let $\mathsf{Z} = \mathbf{z}(\Pi_1, \ldots, \Pi_l)$ be a random variable that is a non-negative function with finite range of a series $\Pi_1, \ldots, \Pi_l$ of independent random variables, where each Π_i takes on a random permutation (bijection) $\pi : \{1, \ldots, |P|\} \rightarrow P$ of a finite non-empty set P. If $d = \omega(\sqrt{\mathbb{E}[\mathsf{Z}]})$, then,*

$$\Pr(|\mathsf{Z} - \mathbb{E}[\mathsf{Z}]| \geq d) = 2/e^{(\omega(1))} \quad \textit{as } d \rightarrow \infty.$$

Proof. First, assume that there exist constants δ and μ that satisfy M1 and M2 in Theorem 4.1, respectively. In this case, since $d = \omega(\sqrt{\mathbb{E}[\mathsf{Z}]})$, the $60\delta\sqrt{\mu\mathbb{E}[\mathsf{Z}]} + 1$ term is negligible, and, for any constant $\beta < 1/8\delta^2\mu$ and large enough value of $\mathbb{E}[\mathsf{Z}]$, we have (c.f., [18, p. 81]),

$$\Pr(|\mathsf{Z} - \mathbb{E}[\mathsf{Z}]| \geq d) \leq 2/e^{\left(\beta d^2/\mathbb{E}[\mathsf{Z}]\right)}.$$

In other words, if $d = \omega(\sqrt{\mathbb{E}[\mathsf{Z}]})$, then,

$$\Pr(|\mathsf{Z} - \mathbb{E}[\mathsf{Z}]| \geq d) = 2/e^{(\omega(1))} \quad \text{as } d \rightarrow \infty.$$

We now show that there exist satisfactory constants δ and μ. We can satisfy M1 by setting δ to be the difference between the maximum and minimum values of the range of $\mathbf{z}$; because the range of the $\mathbf{z}$ is finite, δ can be at most this difference. Next, note that there is a finite number of mappings—specifically, if $|P| = s$, then there are ls mappings. Therefore, we can satisfy M2 by setting $\mu = ls$. □

So that we can apply Theorem 4.2 to bound ϵ, we present a method for defining $\mathsf{Q}_{\rm wt}$, $\mathsf{A}'_{\rm rd}$, $\mathsf{A}_{\rm wt}$, $\mathsf{A}'_{\rm wt}$, and $\mathsf{Q}_{\rm rd}$ (where $\mathsf{Q}_{\rm rd}$ represents a read quorum selected by a non-faulty client) in terms of independent random variables Π_1, Π_2, Π_3, and Π_4, each taking on a random permutation $\{1, \ldots, |U|\} \rightarrow U$, where U is the set of all n servers. Fix any set of b servers to constitute B. Then consider the following definitions:

- Define $\mathsf{A}_{\rm wt} = \{\Pi_1(1), \ldots, \Pi_1(a_{wt})\}$.
- Define $\mathsf{A}'_{\rm wt} = \{\Pi_2(1), \ldots, \Pi_2(a_{wt})\}$.
- Define $\mathsf{A}'_{\rm rd} = \{\Pi_3(1), \ldots, \Pi_3(a_{rd})\}$.

- Define $\mathsf{Q}_{\rm rd} = \{\Pi_4(1), \ldots, \Pi_4(q_{rd})\}$.

Because each permutation is randomly selected (independently of B), so too are $\mathsf{A}_{\rm wt}$, $\mathsf{A}'_{\rm wt}$, $\mathsf{A}'_{\rm rd}$, and $\mathsf{Q}_{\rm rd}$. We define $\mathsf{Q}_{\rm wt}$ in accordance with Section 3. Specifically, first we choose each $\Pi_1(i)$ such that $\Pi_1(i) \in \mathsf{A}_{\rm wt} \cap B$. Next, for each $j = 1..a_{wt}$, we choose $\Pi_1(j)$ if we have not yet chosen q_{wt} servers and $\Pi_1(j) \in \mathsf{A}_{\rm wt} \setminus (\mathsf{A}'_{\rm wt} \cup B)$. Finally, for each $k = 1..a_{wt}$, we choose $\Pi_1(k)$ if we have not yet chosen q_{wt} servers and $\Pi_1(k) \in \mathsf{A}_{\rm wt} \cap (\mathsf{A}'_{\rm wt} \setminus B)$.

Define $\mathsf{MinCorrect}$ to be a random variable for the number of non-faulty servers with the established value based only on the intersection, union, complement, and difference of some number of $\mathsf{Q}_{\rm rd}$, $\mathsf{Q}_{\rm wt}$, $\mathsf{A}'_{\rm rd}$, $\mathsf{A}_{\rm wt}$, $\mathsf{A}'_{\rm wt}$, and B. For the purposes of this paper, $\mathsf{MinCorrect} = |(\mathsf{Q}_{\rm rd} \cap \mathsf{Q}_{\rm wt}) \setminus B|$ as indicated in (1).

Lemma 4.3. *Let* $\mathsf{Z} = \mathsf{MinCorrect}$. *Let* $\mathbb{E}[\mathsf{Z}] > r$ *and* $\mathbb{E}[\mathsf{Z}] - r = \omega(\sqrt{\mathbb{E}[\mathsf{Z}]})$. *Then,*

$$\Pr(\mathsf{Z} \le r) = 2/e^{(\omega(1))} \quad \textit{as } \mathbb{E}[\mathsf{Z}] - r \to \infty.$$

Proof. Z can be treated as a function of independent permutations. Note that the range of Z is non-negative and finite: $[0..n]$. Let $d = \mathbb{E}[\mathsf{Z}] - r$; then, by assumption, $d = \omega(\sqrt{\mathbb{E}[\mathsf{Z}]})$. We apply Theorem 4.2, yielding,

$$\begin{aligned}
&\Pr(\mathsf{Z} \le r) \\
&= \Pr(\mathsf{Z} \le \mathbb{E}[\mathsf{Z}] - d) \\
&= \Pr(\mathbb{E}[\mathsf{Z}] - \mathsf{Z} \ge d) \\
&\le \Pr(|\mathsf{Z} - \mathbb{E}[\mathsf{Z}]\,| \ge d) \\
&= 2/e^{(\omega(1))} \quad \text{as } d \to \infty \\
&= 2/e^{(\omega(1))} \quad \text{as } \mathbb{E}[\mathsf{Z}] - r \to \infty.
\end{aligned}$$

□

Define $\mathsf{MaxConflicting}$ to be a random variable for the number of servers with a conflicting response based only on the intersection, union, complement, and difference of $\mathsf{Q}_{\rm rd}$, $\mathsf{Q}_{\rm wt}$, $\mathsf{A}'_{\rm rd}$, $\mathsf{A}_{\rm wt}$, $\mathsf{A}'_{\rm wt}$, and B. For example: due to (2), in masking quorums with write markers, $\mathsf{MaxConflicting} = |(\mathsf{A}'_{\rm rd} \cap \mathsf{A}'_{\rm wt}) \cap B|$; due to (3), in masking quorums without write markers, $\mathsf{MaxConflicting} = |\mathsf{A}'_{\rm rd} \cap B|$; due to (4), in opaque quorums with write markers, $\mathsf{MaxConflicting} = |(\mathsf{A}'_{\rm rd} \cap \mathsf{A}'_{\rm wt}) \cap B| + |\,((\mathsf{A}'_{\rm rd} \cap \mathsf{A}'_{\rm wt}) \setminus B) \setminus \mathsf{Q}_{\rm wt}|$; and, due to (5), in opaque quorums without write markers, $\mathsf{MaxConflicting} = |\mathsf{A}'_{\rm rd} \cap B| + |\,((\mathsf{A}'_{\rm rd} \cap \mathsf{A}'_{\rm wt}) \setminus B) \setminus \mathsf{Q}_{\rm wt}|$.

Lemma 4.4. *Let* $\mathsf{Z}' = \mathsf{MaxConflicting}$. *Let* $r > \mathbb{E}[\mathsf{Z}']$ *and* $r - \mathbb{E}[\mathsf{Z}'] = \omega(\sqrt{\mathbb{E}[\mathsf{Z}']})$. *Then,*

$$\Pr(\mathsf{Z}' \ge r) = 2/e^{(\omega(1))} \quad \textit{as } r - \mathbb{E}[\mathsf{Z}'] \to \infty.$$

Proof. Z' can be treated as a function of independent permutations. Note that the range of Z' is non-negative and finite: $[0..n]$. Let $d = r - \mathbb{E}[\mathsf{Z}']$; then, by assumption, $d = \omega(\sqrt{\mathbb{E}[\mathsf{Z}']})$. We apply Theorem 4.2, yielding,

$$\begin{aligned}
&\Pr(\mathsf{Z}' \geq r) \\
&= \Pr(\mathsf{Z}' \geq d + \mathbb{E}\left[\mathsf{Z}'\right]) \\
&= \Pr(\mathsf{Z}' - \mathbb{E}\left[\mathsf{Z}'\right] \geq d) \\
&\leq \Pr(|\mathsf{Z}' - \mathbb{E}\left[\mathsf{Z}'\right]| \geq d) \\
&= 2/e^{(\omega(1))} \quad \text{as } d \to \infty \\
&= 2/e^{(\omega(1))} \quad \text{as } r - \mathbb{E}\left[\mathsf{Z}'\right] \to \infty.
\end{aligned}$$

□

Theorem 4.5. *A probabilistic quorum system configuration can provide consistency with compelling probability if,*

$$\begin{aligned}
&\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] > 0, \\
&\mathbb{E}\left[\mathsf{MinCorrect}\right] = \omega(1) \quad \textit{as } n \to \infty, \quad \textit{and} \\
&\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]}).
\end{aligned}$$

Proof. Set r as follows,

$$r = \frac{\mathbb{E}\left[\mathsf{MinCorrect}\right] + \mathbb{E}\left[\mathsf{MaxConflicting}\right]}{2}.$$

Then we can apply Lemma 4.3 to $\Pr(\mathsf{MinCorrect} \leq r)$ because,

$$\begin{aligned}
&\mathbb{E}\left[\mathsf{MinCorrect}\right] > r, \text{ and} \\
&\mathbb{E}\left[\mathsf{MinCorrect}\right] - r = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]}).
\end{aligned}$$

Next, note that $r - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$. But, since $\mathbb{E}\left[\mathsf{MinCorrect}\right]$ is growing faster than $\mathbb{E}\left[\mathsf{MaxConflicting}\right]$, it is also the case that $r - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MaxConflicting}\right]})$. Therefore, we can apply Lemma 4.4 to $\Pr(\mathsf{MaxConflicting} \geq r)$ because

$$\begin{aligned}
&r > \mathbb{E}\left[\mathsf{MaxConflicting}\right], \text{ and} \\
&r - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MaxConflicting}\right]}).
\end{aligned}$$

As described in Section 4.1, it is an error if $\mathsf{MinCorrect} < r$ or $\mathsf{MaxConflicting} \geq r$. Therefore, the error probability, ϵ, goes to zero with increasing n because,

$$\begin{aligned}
\epsilon &= \Pr(\mathsf{MaxConflicting} \geq r \vee \mathsf{MinCorrect} < r) \\
&= \Pr(\mathsf{MaxConflicting} \geq r) + \Pr(\mathsf{MinCorrect} < r) - \\
&\qquad \Pr(\mathsf{MaxConflicting} \geq r \wedge \mathsf{MinCorrect} < r) \\
&\leq \Pr(\mathsf{MaxConflicting} \geq r) + \Pr(\mathsf{MinCorrect} < r) \\
&\leq \Pr(\mathsf{MaxConflicting} \geq r) + \Pr(\mathsf{MinCorrect} \leq r) \\
&= 2/e^{(\omega(1))} + 2/e^{(\omega(1))} \\
&\qquad \text{as } (\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right])/2 \to \infty \\
&= 2/e^{(\omega(1))} + 2/e^{(\omega(1))} \quad \text{as } n \to \infty.
\end{aligned}$$

Where the second-to-last line follows because,

$$(\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right])/2 = \mathbb{E}\left[\mathsf{MinCorrect}\right] - r = r - \mathbb{E}\left[\mathsf{MaxConflicting}\right].$$

And the final line follows because,

$$(\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right])/2 \to \infty \text{as } n \to \infty. \qquad \square$$

The remainder of the section is focused on determining, for each type of probabilistic quorum system, the upper bound on b for which Theorem 4.5 applies.

Lemma 4.6.

$$\mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right] = \frac{a_{rd}b}{n}. \tag{6}$$

Proof. $\mathsf{A}'_{\mathrm{rd}}$ is selected independently of B. As such, $|\mathsf{A}'_{\mathrm{rd}} \cap B|$ is a hypergeometric random variable characterized by a_{rd} draws from a population of n elements containing b successes. Therefore, we use the formula for the expected value of a hypergeometric random variable. □

Lemma 4.7.

$$\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] = \frac{a_{rd}a_{wt}b}{n^2}. \tag{7}$$

Proof. We calculate $\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right]$ directly as follows. Consider an indicator random variable Ind_u, such that $\mathsf{Ind}_u = 1$ if $u \in (\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B$, and $\mathsf{Ind}_u = 0$ otherwise. For each $u \in B$, we have $\Pr[\mathsf{Ind}_u = 1] = \frac{a_{rd}a_{wt}}{n^2}$, since A_{rd} and A'_{wt} are chosen independently. By linearity of expectation:

$$\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] = \sum_{u \in B} \Pr(\mathsf{Ind}_u = 1) = b\left(\frac{a_{rd}a_{wt}}{n^2}\right). \qquad \square$$

In the proofs of the following lemmas, we use rules of conditional expectation (c.f., [17, Section 2.3]). In particular, the following.

Definition 4.8 ([17])**.** *The expression* $\mathbb{E}\left[\mathsf{X} \mid \mathsf{Y}\right]$ *is a random variable* $f(\mathsf{Y})$ *that takes on the value* $\mathbb{E}\left[\mathsf{X} \mid \mathsf{Y} = y\right]$ *when* $\mathsf{Y} = y$.

Because $\mathbb{E}\left[\mathsf{X} \mid \mathsf{Y}\right]$ is a random variable, i.e., a function, it makes sense to consider its expectation.

Theorem 4.9 ([17, Theorem 2.7])**.**

$$\mathbb{E}\left[\mathsf{X}\right] = \mathbb{E}\left[\mathbb{E}\left[\mathsf{X} \mid \mathsf{Y}\right]\right]. \tag{8}$$

Lemma 4.10.

$$\mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right] = \frac{q_{wt}n - a_{wt}b}{n}. \tag{9}$$

Proof. Let $\mathsf{MalWrite} = |\mathsf{A}_{\mathrm{wt}} \cap B|$. Since A_{wt} is selected uniformly at random independently of B, $\mathsf{MalWrite}$ is a hypergeometric random variable, characterized by a_{wt} draws from a population of n elements containing b successes; therefore,

$$\mathbb{E}\left[\mathsf{MalWrite}\right] = \frac{a_{wt}b}{n}.$$

Recall from Section 3, that a write is established once all of the non-faulty servers in any write quorum in A_{wt} have accepted it. Therefore,

$$\mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B| \ \mid \ \mathsf{MalWrite} = m\right] = q_{wt} - m.$$

Applying Theorem 4.9 and linearity of expectation, we have that,

$$\begin{aligned}
&\mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right] \\
&\quad = \mathbb{E}\left[\mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B| \ \mid \ \mathsf{MalWrite}\right]\right] \\
&\quad = \mathbb{E}\left[q_{wt} - \mathsf{MalWrite}\right] \\
&\quad = q_{wt} - \mathbb{E}\left[\mathsf{MalWrite}\right] \\
&\quad = q_{wt} - \frac{a_{wt}b}{n}.
\end{aligned}$$

□

Lemma 4.11.

$$\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] = \frac{q_{rd}(nq_{wt} - a_{wt}b)}{n^2}. \tag{10}$$

Proof. Q_{rd} is independent of $\mathsf{Q}_{\mathrm{wt}} \setminus B$; therefore, $|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B| \ \mid \ |\mathsf{Q}_{\mathrm{wt}} \setminus B| = m$ is a conditional hypergeometric random variable characterized by q_{rd} draws from a population of n elements containing m successes, and,

$$\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B| \ \mid \ |\mathsf{Q}_{\mathrm{wt}} \setminus B| = m\right] = \frac{q_{rd}m}{n}.$$

Applying Theorem 4.9, by linearity of expectation we have that,

$$\begin{aligned}
&\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] \\
&= \mathbb{E}\left[\ \mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B| \mid |\mathsf{Q}_{\mathrm{wt}} \setminus B|\right]\ \right] \\
&= \mathbb{E}\left[\frac{q_{rd}}{n}\left(|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right)\right] \\
&= \frac{q_{rd}}{n}\mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right] \\
&= \frac{q_{rd}(nq_{wt} - a_{wt}b)}{n^2}.
\end{aligned}$$

□

Lemma 4.12.

$$\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B|\right] = a_{rd}\left(\frac{a_{wt}}{n} - \left(\frac{a_{wt}}{n}\right)\left(\frac{b}{n}\right)\right). \tag{11}$$

Proof. We calculate $\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B|\right]$ directly as follows. Consider an indicator random variable Ind_u, such that $\mathsf{Ind}_u = 1$ if $u \in (\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B$, and $\mathsf{Ind}_u = 0$ otherwise. For each $u \in U \setminus B$, we have $\Pr[\mathsf{Ind}_u = 1] = \frac{a_{rd}a_{wt}}{n^2}$, since A'_{rd} and A'_{wt} are chosen independently. By linearity of expectation:

$$\begin{aligned}\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B|\right] &= \sum_{u \in U \setminus B} \Pr(\mathsf{Ind}_u = 1) \\ &= (n-b)\left(\frac{a_{rd}a_{wt}}{n^2}\right) = a_{rd}\left(\frac{a_{wt}}{n} - \left(\frac{a_{wt}}{n}\right)\left(\frac{b}{n}\right)\right). \qquad \square\end{aligned}$$

Lemma 4.13.

$$\mathbb{E}\left[|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)|\right] \geq \frac{a_{rd}}{n}\left(q_{wt} - \frac{a_{wt}}{n}\left(b + \frac{(n-a_{wt})(n-b)}{n}\right)\right). \tag{12}$$

Proof. In calculating $\mathbb{E}\left[|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)|\right]$, because a faulty client may perform *both* a write that becomes established and another write that conflicts with the first write, we cannot assume that Q_{wt} is selected independently of $\mathsf{A}'_{\mathrm{wt}}$. Let $\mathsf{CI} = \mathsf{A}'_{\mathrm{wt}} \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)$. Then, in particular, as described in Section 3, such a client seeks to maximize $\mathbb{E}\left[\mathsf{MaxConflicting}\right]$, and therefore minimizes $\mathbb{E}\left[|\mathsf{CI}|\right]$ by choosing the servers for $\mathsf{Q}_{\mathrm{wt}} \setminus B$ from $\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B)$ first. Thus,

$$\begin{aligned}\mathbb{E}\left[|\mathsf{CI}|\right] &= \mathbf{max}(\ 0\ ,\ \mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right] - \mathbb{E}\left[|\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B)|\ \right]) \\ &\geq \mathbb{E}\left[|\mathsf{Q}_{\mathrm{wt}} \setminus B|\right] - \mathbb{E}\left[|\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B)|\right]. \end{aligned} \tag{13}$$

We calculate $\mathbb{E}\left[|\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B)|\right]$ directly as follows. For clarity, note that $\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B) = (\mathsf{A}_{\mathrm{wt}} \setminus \mathsf{A}'_{\mathrm{wt}}) \setminus B$. Consider an indicator random variable Ind_u, such that $\mathsf{Ind}_u = 1$ if $u \in (\mathsf{A}_{\mathrm{wt}} \setminus \mathsf{A}'_{\mathrm{wt}}) \setminus B$, and $\mathsf{Ind}_u = 0$ otherwise. For each $u \in U \setminus B$, we have $\Pr[\mathsf{Ind}_u = 1] = \frac{a_{wt}(n-a_{wt})}{n^2}$, since $\mathsf{A}'_{\mathrm{wt}}$ and A_{wt} are independent. By linearity of expectation:

$$\mathbb{E}\left[|\mathsf{A}_{\mathrm{wt}} \setminus (\mathsf{A}'_{\mathrm{wt}} \cup B)|\right] = \sum_{u \in U \setminus B} \Pr(\mathsf{Ind}_u = 1) = (n-b)\left(\frac{a_{wt}(n-a_{wt})}{n^2}\right) = \frac{a_{wt}}{n^2}(n-a_{wt})(n-b). \tag{14}$$

By (13), (9), and (14), we have that,

$$\mathbb{E}\left[|\mathsf{CI}|\right] \geq \left(q_{wt} - \frac{a_{wt}b}{n}\right) - \frac{a_{wt}}{n^2}(n-a_{wt})(n-b) = q_{wt} - \frac{a_{wt}}{n}\left(b + \frac{(n-a_{wt})(n-b)}{n}\right). \tag{15}$$

Since A_{rd} is independent of CI, we see that $|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)| \ \mid\ |\mathsf{CI}| = c$ is a hypergeometric random variable characterized by a_{rd} draws from a population of n elements containing c successes, and,

$$\mathbb{E}\left[|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)| \ \mid\ |\mathsf{CI}| = c\right] = \frac{a_{rd}c}{n}.$$

Applying Theorem 4.9, by linearity of expectation we have that,

$$
\begin{aligned}
&\mathbb{E}\left[|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B))|\right] \\
&\quad = \mathbb{E}\left[\, |(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)| \;\mid\; |\mathsf{CI}|\, \right] \\
&\quad = \mathbb{E}\left[\frac{a_{rd}}{n}|\mathsf{CI}|\right] = \frac{a_{rd}}{n}\mathbb{E}\left[|\mathsf{CI}|\right]. \qquad (16)
\end{aligned}
$$

Therefore, by (15) and (16) we have that,

$$
\mathbb{E}\left[|(\mathsf{A}_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B))|\right] \geq \frac{a_{rd}}{n}\left(q_{wt} - \frac{a_{wt}}{n}\left(b + \frac{(n-a_{wt})(n-b)}{n}\right)\right). \qquad \square
$$

Lemma 4.14.

$$
\mathbb{E}\left[|\,((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B) \setminus \mathsf{Q}_{\mathrm{wt}}|\right] \leq \frac{a_{rd}}{n^3}(2a_{wt}n^2 - na_{wt}b - q_{wt}n^2 - a_{wt}^2 n + a_{wt}^2 b). \qquad (17)
$$

Proof. To calculate $\mathbb{E}\left[|\,((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B) \setminus \mathsf{Q}_{\mathrm{wt}}|\right]$, first note that:

$$
\begin{aligned}
&\mathbb{E}\left[|\,((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B) \setminus \mathsf{Q}_{\mathrm{wt}}|\right] \\
&\quad = \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B|\right] - \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap (\mathsf{Q}_{\mathrm{wt}} \setminus B)|\right]. \qquad (18)
\end{aligned}
$$

Combining and simplifying equations (11), (12), and (18), we obtain (17). □

Lemma 4.15. *For a probabilistic or opaque masking quorum system configuration with or without write markers,*[3] *let the ratios of q_{rd}, q_{wt}, a_{wt}, and a_{rd} to n be fixed, and let $\mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]$. Then,*

$$
\begin{aligned}
&\mathbb{E}\left[\mathsf{MinCorrect}\right] = \Omega(n). \\
&\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]}).
\end{aligned}
$$

Proof. First, we see from (10) that for the four types of quorums,

$$
\mathbb{E}\left[\mathsf{MinCorrect}\right] = \Omega(n).
$$

Next, consider that $\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$:

- Masking without write markers:
  $\mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]$ implies $\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] - \mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right] \neq 0$, and so (10) and (6) show us that $\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$.

- Masking with write markers:
  $\mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]$ implies $\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] - \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] \neq 0$, and so (10) and (7) show us that $\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$.

[3] Though not shown here, Lemma 4.15 also applies trivially to dissemination quorum systems.

- Opaque without write markers:
  $\mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]$ implies $\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] - (\mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right]$ $+\mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right]) \neq 0$, and so (10), (6), and (17) show us that $\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$.

- Opaque with write markers:
  $\mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]$ implies $\mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] - (\mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] + \mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right]) \neq 0$, and so (10), (7), and (17) show us that $\mathbb{E}\left[\mathsf{MinCorrect}\right] - \mathbb{E}\left[\mathsf{MaxConflicting}\right] = \omega(\sqrt{\mathbb{E}\left[\mathsf{MinCorrect}\right]})$. □

Theorem 4.16. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed. Then a probabilistic masking quorum system configuration employing write markers provides consistency with compelling probability if,*

$$b < \frac{q_{rd} q_{wt} n}{q_{rd} a_{wt} + a_{rd} a_{wt}}.$$

Proof. By (10) and (7),

$$\begin{aligned} & b < \frac{q_{rd} q_{wt} n}{q_{rd} a_{wt} + a_{rd} a_{wt}} \\ \Leftrightarrow\ & \mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] > \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] \\ \Leftrightarrow\ & \mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]. \end{aligned}$$

Therefore, we can apply Lemma 4.15 and, thus, Theorem 4.5. □

Corollary 4.17. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt}$. Then a probabilistic masking quorum system configuration employing write markers provides consistency with compelling probability if,*

$$b < n/2.$$

Note that, with write markers, the size of quorums does not impact the the maximum fraction of faults that can be tolerated with compelling probability when quorums are selected uniformly at random, that is, when $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt}$.

Theorem 4.18. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed. Then a probabilistic masking quorum system configuration without write markers provides consistency with compelling probability if,*

$$b < \frac{q_{rd} q_{wt} n}{q_{rd} a_{wt} + a_{rd} n}.$$

Proof. By (10) and (6),

$$\begin{aligned} & b < \frac{q_{rd} q_{wt} n}{q_{rd} a_{wt} + a_{rd} n} \\ \Leftrightarrow\ & \mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] > \mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right] \\ \Leftrightarrow\ & \mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right]. \end{aligned}$$

Therefore, we can apply Lemma 4.15 and, thus, Theorem 4.5. □

Corollary 4.19. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt}$. Then a probabilistic masking quorum system configuration without write markers can provide consistency with compelling probability if,*

$$b < \frac{q_{wt} n}{q_{wt} + n}.$$

Corollary 4.20. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt} = n - b$. Then a probabilistic masking quorum system configuration without write markers can provide consistency with compelling probability if,*

$$b < n/2.62.$$

Theorem 4.21. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed. Then a probabilistic opaque quorum system configuration employing write markers provides consistency with compelling probability if,*

$$b < \frac{n(a_{rd}a_{wt} + a_{rd}q_{wt}n + q_{rd}q_{wt}n - 2a_{rd}a_{wt}n)}{a_{wt}(a_{rd}a_{wt} + q_{rd}n)}.$$

Proof. By (10), (7), and (17),

$$\begin{aligned}
& b < \frac{n(a_{rd}a_{wt} + a_{rd}q_{wt}n + q_{rd}q_{wt}n - 2a_{rd}a_{wt}n)}{a_{wt}(a_{rd}a_{wt} + q_{rd}n)} \\
& \Rightarrow \mathbb{E}\left[|(\mathsf{Q}_{\mathrm{rd}} \cap \mathsf{Q}_{\mathrm{wt}}) \setminus B|\right] > \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] + \mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right] \\
& \Leftrightarrow \mathbb{E}\left[\mathsf{MinCorrect}\right] > \mathbb{E}\left[\mathsf{MaxConflicting}\right].
\end{aligned}$$

Therefore, we can apply Lemma 4.15 and, thus, Theorem 4.5. □

Corollary 4.22. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt}$. Then a probabilistic opaque quorum system configuration employing write markers can provide consistency with compelling probability if,*

$$b < \frac{q_{wt} n}{q_{wt} + n}.$$

As with traditional probabilistic opaque quorum systems, the maximum fraction of faults that can be tolerated with compelling probability is independent of the size of read quorums when quorums are selected uniformly at random.

Corollary 4.23. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt} = n - b$. Then a probabilistic opaque quorum system configuration employing write markers can provide consistency with compelling probability if,*

$$b < n/2.62.$$

Theorem 4.24 ([16]). *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed. Then a probabilistic opaque quorum system configuration without write markers provides consistency with compelling probability if,*

$$b < \frac{(a_{rd}q_{wt}n - 2a_{rd}a_{wt}n + a_{wt}^2 a_{rd} + q_{rd}q_{wt}n)n}{n^2 a_{rd} - a_{rd}a_{wt}n + a_{wt}^2 a_{rd} + q_{rd}a_{wt}n}.$$

Corollary 4.25. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt}$. Then a probabilistic opaque quorum system without write markers can provide consistency with compelling probability if,*

$$b < \frac{q_{wt}^2 n}{q_{wt}^2 + n^2}.$$

Corollary 4.26. *Let the ratio of each of q_{rd}, a_{rd}, q_{wt}, a_{wt}, and b to n be fixed, and let $a_{rd} = q_{rd}$ and $a_{wt} = q_{wt} = n - b$. Then a probabilistic opaque quorum system without write markers can provide consistency with compelling probability if,*

$$b < n/3.15.$$

5 Implementation

Our implementation of write markers provides the behavior assumed in Section 4, even with Byzantine clients, without relying on the network to issue write markers. Specifically, it ensures properties W1–W3. (Though, technically, it ensures W2 only approximately in the case of opaque quorum systems, in which, as we explain below, a faulty server may be able to create a conflicting candidate using a write marker for a stale, i.e., out-of-date, access set—but to no advantage.)

Because clients may be faulty, we cannot rely on, e.g., digital signatures issued by them to implement write markers. Instead, we adapt mechanisms of our access-restriction protocol for probabilistic opaque quorum systems [16]. The access-restriction protocol is designed to ensure that all clients follow the access strategy. It already enables non-faulty *servers* to verify this before accepting a write. And, since it is the only way of which we are aware for a probabilistic quorum system to tolerate Byzantine clients when write markers are of benefit (i.e., when the sizes of write access sets are restricted), its mechanisms are appropriate.

The preexisting protocol works as follows [16]. From the servers, a client obtains a *verifiable recent value* (VRV), the value of which is unpredictable to clients and b or fewer servers prior to its creation. This VRV is used to generate a pseudorandom sequence of access sets. Since the validity of a VRV can be verified using only public information, both it and the sequence of access sets it induces can be verified by clients and servers. Non-faulty clients simply choose the next unused access set for each operation.[4] However, a faulty client is motivated to maximize the probability of error; if the use of the next access set in the sequence does not maximize the probability of error

[4] Non-faulty clients should choose a new access set for each operation to ensure independence from the decisions of faulty clients [16].

given the current state of the system (i.e., the candidates accepted by the servers), such a client may try to skip ahead some number of access sets, or, alternatively, to wait to use the next access set until the state of the system changes. If allowed to follow either strategy, such a client would circumvent the access strategy because its choice of access set would not be independent from the state of the system.

Three mechanisms are used together to coerce a faulty client to follow the access strategy. First, the client must perform exponentially increasing work in expectation in order to use later access sets. As such, a client requires exponentially increasing time in expectation in order to choose a later access set. This is implemented by requiring that the client solve a client puzzle [5] of the appropriate difficulty. The solution to the puzzle is, in expectation, difficult to find but easy to verify. Second, the VRV and sequence of access sets become invalid as the non-faulty servers accept additional candidates, or as the system otherwise progresses (e.g., as time passes). Non-faulty servers verify that an access set is still valid, i.e., not stale, before accepting it. Thus, system progress forces the client to start its work anew, and, as such, makes the work solving the puzzle for any unused access set wasted. Finally, during the time that the client is working, the established candidate propagates in the background to the non-faulty servers that are non-qualified (c.f., [10]). This decreases the window of vulnerability in which a given access set in the sequence is useful for a conflicting write by making non-qualified servers aware that there is an established candidate (so that they will not accept a conflicting candidate) and that the state of the system has progressed (so that they will invalidate the current VRV if appropriate).

The impact of these three mechanisms is that a non-faulty *server* can be confident that the choice of write access set adheres (at least approximately) to the access strategy upon having verified that the access set is valid, current, and is accompanied by an appropriate puzzle solution.

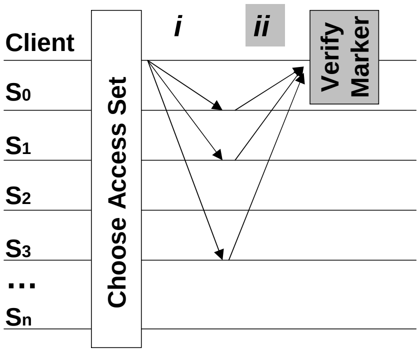

Figure 1: Read operation with write markers: messages and stages of verification of access set. (Changes in gray.)

For write markers, we extend the protocol so that, as seen in Figure 1, *clients* can also perform verification. This requires that information about the puzzle solution and access set (including the VRV used to generate it) be returned by the servers to clients. (As seen in Figure 2 and explained below, this information varies across masking and opaque quorum systems.) In the preexisting access-restriction protocol, this information is verified and discarded by each server. For write markers, this information is instead stored by each server in the verification stage as a write marker, and is sent along with the data value as part of the candidate to the client during any read operation. If the server is non-faulty—a fact that a non-faulty client cannot know—the access set used for the operation was indeed chosen according to the access strategy because the server performed verification before accepting the operation. However, because the server may be faulty, the client performs verification as well; it verifies that the server is a member of the access set, and that the write marker is valid. This allows us to guarantee points W1–W3. As such, faulty non-qualified servers are unable to vote for the candidates for which qualified servers can vote.

Figures 1, 2, 3, and 4 illustrate relevant pieces of the preexisting protocol and our modifications for write markers in the context of read and write operations in probabilistic masking and opaque quorum systems. (They ignore details [16] irrelevant to write markers such as the structure of the VRV and how a client obtains one, as well as propagation of established data values.) The figures highlight that the additions to the protocol for write markers involve saving the write markers and returning them to clients so that clients can also verify them.

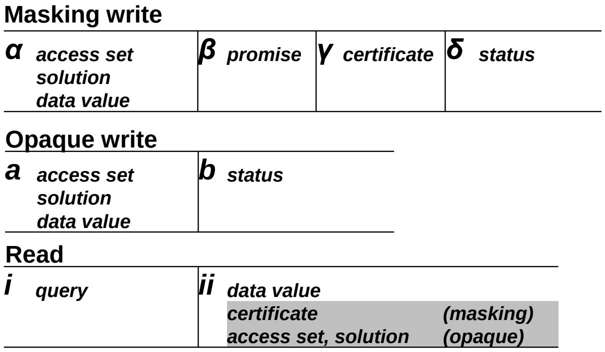

Figure 2: Message types. (Write marker emphasized with gray.)

The differences in the structure of the write marker for probabilistic opaque and masking quorum systems results in subtly different guarantees. The remainder of the section discusses these details.

5.1 Probabilistic Opaque Quorums

As seen in Figure 2 (message ii), a write marker for a probabilistic opaque quorum system consists of the write-access-set identifier (including the VRV) and the solution to the puzzle that unlocks the use of this access set. Unlike a non-faulty server that verifies the access set at the time of use, a non-faulty client cannot verify that an access set was not already stale when the access set was accepted by a faulty server. Initially, this may appear problematic because it is clear that, given sufficient time, a faulty client will eventually be able to solve the puzzle for its preferred access set to use for a conflicting write—this access set may contain all of the servers in B. In addition, the faulty client can delay the use of this access set because non-faulty clients will be unable to verify whether it was already stale when it was used.

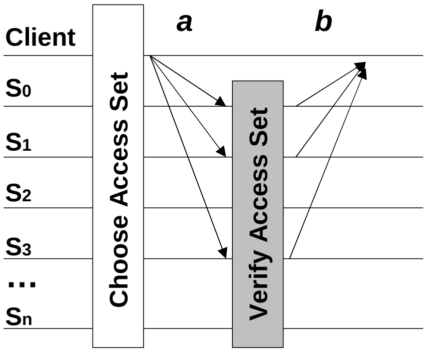

Figure 3: Write operation in opaque quorum systems: messages and stages of verification of write marker. (Changes in gray.)

Fortunately, because non-faulty servers will not accept a stale candidate (i.e., a candidate accompanied by a stale access set) during a write (Figure 3), the fact that a stale access set may be accepted by a faulty server does not impact the benefit of write markers for opaque quorum systems. In general, consistency requires (4), i.e.,

$$r > \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right] + \mathbb{E}\left[|\left((\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \setminus B\right) \setminus \mathsf{Q}_{\mathrm{wt}}|\right].$$

However, only faulty servers will accept a stale candidate. Therefore, if the candidate was stale when written to A'_{wt}, no non-faulty server would have accepted it. Thus, in this case, the consistency constraint is equivalent to,

$$r > \mathbb{E}\left[|(\mathsf{A}'_{\mathrm{rd}} \cap \mathsf{A}'_{\mathrm{wt}}) \cap B|\right].$$

Even if the access set contains all of the faulty servers, i.e., $B \subset A'_{wt}$, then this becomes,

$$r > \mathbb{E}\left[|\mathsf{A}'_{\mathrm{rd}} \cap B|\right].$$

However, this is (3), the constraint on probabilistic masking quorum systems without write markers. In effect, the client must either: (i) use a recent access set that is therefore chosen approximately uniformly at random, and be limited by (4); or (ii), use a stale access set and be limited by (3). If quorums are the sizes of access sets, both inequalities have the same upper bound on b as seen in Corollary 4.19 and Corollary 4.22; otherwise, a faulty client is disadvantaged by using a stale access set because (3) allows the system to tolerate more faults and, therefore, to achieve a lower error probability. (Compare the bounds in Theorem 4.18 and Theorem 4.21.)

5.2 Probabilistic Masking Quorums

Protocols for masking quorum systems involve an additional round of communication (an echo phase, c.f., [12] or broadcast phase, c.f., [15]) during write operations in order to tolerate Byzantine or concurrent clients. This round prevents non-faulty servers from accepting conflicting data values, as assumed by the consistency constraints for masking quorum systems. In order to write a data value, a client must first obtain a *write certificate* (a quorum of replies that together attest that the non-faulty servers will accept no conflicting data value). In contrast to optimistic protocols that use opaque quorum systems, these protocols are pessimistic.

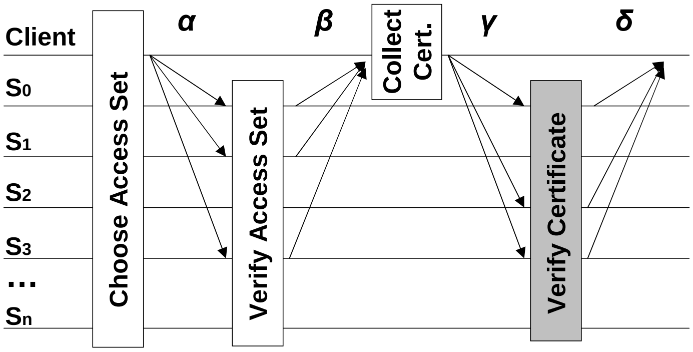

Figure 4: Write operation in masking quorum systems: messages and stages of verification of write marker. (Changes in gray.)

This additional round allows us to prevent clients from using stale access sets. Specifically, in the request to authorize a data value (message α in Figure 2 and Figure 4), the client sends the access set identifier (including the VRV), the solution to the puzzle enabling use of this access set, and the data value. We require that the certificate come from servers in the access set that is chosen for the write operation. Each server verifies the VRV and that the puzzle solution enables use of the indicated access set before returning authorization (message β in Figure 2 and Figure 4). The servers that contribute to the certificate all implicitly agree that the access set is not stale, for otherwise they would not agree to the write. This certificate (sent to each server in message γ in Figure 2 and Figure 4) is stored along with the data value as a write marker. Thus, unlike in probabilistic opaque quorum systems, a valid write marker in a probabilistic masking quorum system implies that a stale access set was not used. The reading client verifies the certificate (returned in message ii in Figure 1 and Figure 2) before accepting a vote for a candidate. Because a writing client will be unable to obtain a certificate for a stale access set, votes for such a candidate will be rejected by reading clients. Therefore, the analysis in Section 4 applies without additional complications.

6 Conclusion

We have presented write markers, a way to: (i) increase the number of faults that probabilistic quorum systems can tolerate with compelling probability; and (ii) allow probabilistic masking quorum systems to tolerate this number independent of the size of write quorums. Write markers achieve this by limiting the extent to which Byzantine-faulty servers may collude to provide incorrect values to clients. We have presented an implementation of markers that is effective even while tolerating Byzantine-faulty clients and servers.

References

[1] M. Abd-El-Malek, G. R. Ganger, G. R. Goodson, M. K. Reiter, and J. J. Wylie. Fault-scalable Byzantine fault-tolerant services. In *Symposium on Operating Systems Principles*, October 2005.

[2] L. Alvisi, D. Malkhi, E. Pierce, and M. K. Reiter. Fault detection for Byzantine quorum systems. *IEEE Transactions on Parallel and Distributed Systems*, 12(9):996–1007, 2001.

[3] R. A. Bazzi. Access cost for asynchronous Byzantine quorum systems. *Distributed Computing*, 14(1):41–48, 2001.

[4] G. R. Goodson, J. J. Wylie, G. R. Ganger, and M. K. Reiter. Efficient Byzantine-tolerant erasure-coded storage. In *International Conference on Dependable Systems and Networks*, June 2004.

[5] A. Juels and J. Brainard. Client puzzles: A cryptographic countermeasure against connection depletion attacks. In *Network and Distributed Systems Security Symposium*, pages 151–165, 1999.

[6] L. Kong, D. Manohar, A. Subbiah, M. Sun, M. Ahamad, and D. Blough. Agile store: Experience with quorum-based data replication techniques for adaptive Byzantine fault tolerance. In *IEEE Symposium on Reliable Distributed Systems*, pages 143–154, 2005.

[7] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.

[8] H. Lee and J. L. Welch. Applications of probabilistic quorums to iterative algorithms. In *International Conference on Distributed Computing Systems*, pages 21–30, April 2001.

[9] H. Lee and J. L. Welch. Randomized shared queues applied to distributed optimization algorithms. In *International Symposium on Algorithms and Computation*, December 2001.

[10] D. Malkhi, Y. Mansour, and M. K. Reiter. Diffusion without false rumors: On propagating updates in a Byzantine environment. *Theoretical Computer Science*, 299(1–3):289–306, 2003.

[11] D. Malkhi and M. Reiter. Byzantine quorum systems. *Distributed Computing*, 11(4):203–213, 1998.

[12] D. Malkhi and M. K. Reiter. An architecture for survivable coordination in large distributed systems. *IEEE Transactions on Knowledge and Data Engineering*, 12(2):187–202, 2000.

[13] D. Malkhi, M. K. Reiter, A. Wool, and R. N. Wright. Probabilistic quorum systems. *Information and Computation*, 170(2):184–206, 2001.

[14] J.-P. Martin and L. Alvisi. Fast Byzantine consensus. *IEEE Transactions on Dependable and Secure Computing*, 3(3):202–215, 2006.

[15] J.-P. Martin, L. Alvisi, and M. Dahlin. Minimal Byzantine storage. In *International Symposium on Distributed Computing*, 2002.

[16] M. G. Merideth and M. K. Reiter. Probabilistic opaque quorum systems. In *International Symposium on Distributed Computing*, 2007.

[17] M. Mitzenmacher and E. Upfal. *Probability and Computing*. Cambridge University Press, 2005.

[18] M. Molloy and B. Reed. *Graph Colouring and the Probabilistic Method*. Springer, 2002.