

## 3 Quorum Systems

### 3.1 Introduction

**Motivation.** Replicate an object on  $n$  servers in order to tolerate faulty servers. The servers communicate by sending messages over an asynchronous point-to-point network. Define quorums of servers such that a client needs only talk to a quorum of servers for accessing the object.

**Definition 3.1 (Quorum System).** Let  $\mathcal{P} = \{P_1, \dots, P_n\}$  be a set of servers. A *quorum system*  $\mathcal{Q} \subset 2^{\mathcal{P}}$  is a set of subsets of  $\mathcal{P}$  such that every two subsets intersect. Each  $Q \in \mathcal{Q}$  is called a *quorum*.

W.l.o.g., quorum systems considered here are minimal, i.e., for any  $Q, Q' \in \mathcal{Q} : Q \not\subseteq Q'$ .

**Algorithm 3.2 (Replicated Read-Write Register).** A variable  $x$  is stored in the system; clients are a reader and a writer who maintains a timestamp  $\tau$ . Every server  $P_i$  stores local copies  $x_i$  and  $\tau_i$ .

- To write  $x$ , the writer picks a quorum  $Q$ , increments  $\tau$ , and sends `(write, x,  $\tau$ )` to all  $P_i \in Q$ ; upon receiving `(write, x,  $\tau$ )` with  $\tau > \tau_i$ , server  $P_i$  sets  $(x_i, \tau_i) \leftarrow (x, \tau)$  and returns an `ack` message; the writer waits for `ack` from all servers in  $Q$  before terminating the write.
- To read  $x$ , the reader picks a quorum  $Q$  and sends a `read` message to all  $P_i \in Q$ ; upon receiving the `read` query,  $P_i$  returns  $(\text{value}, x_i, \tau_i)$ , the reader waits for values from all servers in  $Q$  and selects the one with the highest timestamp.

The algorithm works only for a single reader and for a single writer, which must not fail (why?). The message complexity of every operation is  $2|Q|$ .

**Lemma 3.3.** *A quorum system implements a single-reader single-writer [atomic] read/write register in an asynchronous network.*

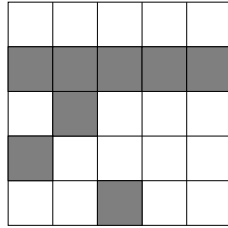
*Proof.* The quorum used by the writer has non-empty intersection with the quorum used by the reader. □

### 3.2 Example Quorum Systems

**Singleton.**  $\mathcal{Q} = \{\{P_1\}\}$ .

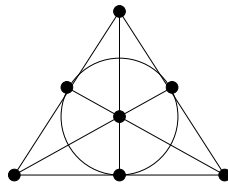
**Majority.**  $\mathcal{P} = \{P_1, \dots, P_n\}$  and  $\mathcal{Q} = \{Q \subset \mathcal{P} \mid |Q| = \lceil \frac{n+1}{2} \rceil\}$ ; tolerates  $t < \frac{n}{2}$  faulty servers. Generalization to *weighted majority* by assigning multiple “votes” to some servers.

**Grid.** Suppose  $n = k \cdot k$  and arrange the servers in a square; a quorum is a full row and one element from each row below the full row.



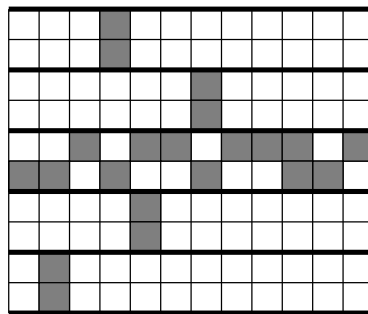
A grid quorum on  $n = 25$  elements.

**Finite Projective Planes (FPP) [Mae85].** Suppose  $n = q^2 + q + 1$  for a prime power  $q$ ; then there is a finite projective plane on  $n$  elements, which consists of a set of subsets from  $\mathcal{P}$  such that every subset has exactly  $q+1$  elements, every element is contained in exactly  $q+1$  subsets, and every two subsets intersect in exactly one element.



The finite projective plane of order 2 (“Fano plane”).

**B-Grid [NW98].** Suppose  $n = dhr$  and arrange the elements in a grid with  $d$  columns and  $h \cdot r$  rows. Call every group of  $r$  rows a *band* and call  $r$  elements in a column restricted to a band a *mini-column*. A quorum consists of one mini-column in every band and one element from each mini-column of one band; thus, every quorum has  $d + hr - 1$  elements.



The B-Grid quorum system over  $n = 120$  elements with  $d = 12$  columns,  $h = 5$  bands, and  $r = 2$  rows per band.

### 3.3 Measures on Quorum Systems

The load is a property of the quorum system; it is defined as the probability that the *busiest* server is in use under an *optimal* strategy of accessing the servers.

**Definition 3.4 (Load).** An *access strategy*  $W$  is a random variable on a quorum system  $\mathcal{Q}$ , i.e.,  $\sum_{Q \in \mathcal{Q}} P_W(Q) = 1$ . The *load induced by  $W$  on a server  $P_i$*  is

$$\ell_W(i) = \sum_{Q \in \mathcal{Q}: i \in Q} P_W(Q).$$

The *load induced by  $W$  on  $\mathcal{Q}$*  is

$$L_W(\mathcal{Q}) = \max_{P_i \in \mathcal{P}} \ell_W(i).$$

The *[system] load of  $\mathcal{Q}$*  is

$$L(\mathcal{Q}) = \min_W L_W(\mathcal{Q}).$$

Let  $c(\mathcal{Q})$  denote the size of the smallest quorum of a quorum system  $\mathcal{Q}$ .

**Theorem 3.5 ([NW98]).**  $L(\mathcal{Q}) \geq \max\{\frac{1}{c(\mathcal{Q})}, \frac{c(\mathcal{Q})}{n}\}$ . Consequently,  $L(\mathcal{Q}) \geq \frac{1}{\sqrt{n}}$ .

Resilience is a worst-case measure for the fault-tolerance of a quorum system, defined as the maximum number of faulty servers that the quorum system can tolerate.

**Definition 3.6 (Resilience).** The *resilience*  $R(\mathcal{Q})$  of a quorum system is the largest  $f$  such that for all sets  $F \subset \mathcal{P}$  of cardinality  $f$ , there is at least one quorum  $Q \in \mathcal{Q}$  which has no common element with  $F$ .

Clearly, the resilience is at most  $c(\mathcal{Q}) - 1$ .

An average-case measure for fault-tolerance is the failure probability. Assume that every server  $P_i$  fails independently with probability  $p$ ; let  $FAIL(i)$  denote the event that  $P_i$  fails.

**Definition 3.7 (Failure probability).** The *failure probability* of a quorum system  $\mathcal{Q}$  is the probability that at least one server of every quorum fails, i.e.,

$$F_p(\mathcal{Q}) = \Pr[\forall Q \in \mathcal{Q} : \exists P_i \in Q \text{ such that } FAIL(i)].$$

**Definition 3.8 ( $s$ -uniform).** A quorum system  $\mathcal{Q}$  is  *$s$ -uniform* if every quorum in  $\mathcal{Q}$  has exactly  $s$  elements.

**Definition 3.9 (balanced).** An access strategy  $W$  for a quorum system  $\mathcal{Q}$  is *balanced* if it satisfies  $\ell_W(i) = L$  for all  $P_i \in \mathcal{P}$ .

**Lemma 3.10.** An  $s$ -uniform quorum system  $\mathcal{Q}$  with a balanced access strategy has load  $L(\mathcal{Q}) = L = \frac{s}{n}$ . Moreover, this load is optimal.

*Proof.* [NW98, Proposition 4.8]. □

**Lemma 3.11.** *The B-Grid quorum system has load  $\frac{d+hr-1}{dhr}$ , resilience  $\max\{d-1, hr-1\}$ , and failure probability at most  $(dp^r)^h + h(1 - (1-p)^r)^d$ . For  $d = \sqrt{n}$ ,  $r = \lfloor \ln d \rfloor$ , and  $0 \leq p \leq \frac{1}{3}$ , we have  $L(\text{B-Grid}) = O(\frac{1}{\sqrt{n}})$ ,  $R(\text{B-Grid}) = O(\sqrt{n})$ , and  $F_p(\text{B-Grid}) = O(e^{-\frac{n^{1/4}}{2}})$ .*

*Proof.* Load follows from Lemma 3.10 since B-Grid is  $(d + hr - 1)$ -uniform. Define  $\mathcal{E}_1$  to be the event that *in every band, all elements of some mini-column fail*, and  $\mathcal{E}_2$  the event that *in some band, at least one element of every mini-column fails*. Clearly, the system fails when  $\mathcal{E}_1 \vee \mathcal{E}_2$  and thus

$$F_p(\text{B-Grid}) \leq \Pr[\mathcal{E}_1] + \Pr[\mathcal{E}_2] = (dp^r)^h + h(1 - (1-p)^r)^d.$$

The last expression is bounded by  $O(e^{-h} + e^{-\frac{\sqrt{d}}{2}})$ , which is bounded by  $O(e^{-\frac{n^{1/4}}{2}})$  for sufficiently large  $n$ .  $\square$

### Comparison.

$\mathcal{Q}$	$L(\mathcal{Q})$	$R(\mathcal{Q})$	$F_p(\mathcal{Q})$
Singleton	1	0	$p$
Majority	$\frac{1}{2}$	$\lfloor \frac{n-1}{2} \rfloor$	$e^{-\Omega(n)}$
Grid	$O(\frac{1}{\sqrt{n}})$	$\sqrt{n} - 1$	$\approx 1^*$
FPP	$O(\frac{1}{\sqrt{n}})$	$q$	$\approx 1^*$
B-Grid <sup>+</sup>	$O(\frac{1}{\sqrt{n}})$	$O(\sqrt{n})$	$O(e^{-\frac{n^{1/4}}{2}})$

\* for large  $n$ .

<sup>+</sup> for  $d = \sqrt{n}$ ,  $r = \lfloor \ln d \rfloor$ , and  $0 \leq p \leq \frac{1}{3}$ .

Only the B-Grid quorum system achieves optimal and close-to-optimal values of all three measures.

### References

- [Mae85] M. Maekawa, *A  $\sqrt{N}$  algorithm for mutual exclusion in distributed systems*, ACM Transactions on Computer Systems **3** (1985), no. 2, 145–159.
- [NW98] M. Naor and A. Wool, *The load, capacity and availability of quorum systems*, SIAM Journal on Computing **27** (1998), no. 2, 423–447.