

Algorithm – Reliable Broadcast

Algorithm 1 Lazy Reliable Broadcast

Implements:

ReliableBroadcast, **instance** rb .

Uses:

BestEffortBroadcast, **instance** beb .

PerfectFailureDetector, **instance** P .

```
1: upon event  $\langle Init \rangle$  do
2:    $delivered := \emptyset$ 
3:    $correct := \Pi$ 
4:   for all  $q \in \Pi$  do
5:      $from[q] := \emptyset$ 
6: upon event  $\langle rb, Broadcast \mid m \rangle$  do
7:   trigger  $\langle beb, Broadcast \mid [DATA, self, m] \rangle$ 
8: upon event  $\langle beb, Deliver \mid p, [DATA, s, m] \rangle$  do
9:   if  $m \notin from[s]$  then
10:    trigger  $\langle rb, Deliver \mid s, m \rangle$ 
11:     $from := from \cup \{m\}$ 
12:    if  $s \notin correct$  then
13:      trigger  $\langle beb, Broadcast \mid [DATA, s, m] \rangle$ 
14: upon event  $\langle P, Crash \mid p \rangle$  do
15:    $correct := correct \setminus \{p\}$ 
16:   for all  $m \in from[p]$  do
17:     trigger  $\langle beb, Broadcast \mid [DATA, p, m] \rangle$ 
```

Algorithm 2 Eager Reliable Broadcast

Implements:ReliableBroadcast, **instance** *rb*.**Uses:**BestEffortBroadcast, **instance** *beb*.

```
1: upon event  $\langle \textit{Init} \rangle$  do
2:   delivered :=  $\emptyset$ 
3: upon event  $\langle \textit{rb}, \textit{Broadcast} \mid m \rangle$  do
4:   trigger  $\langle \textit{beb}, \textit{Broadcast} \mid [\textit{DATA}, \textit{self}, m] \rangle$ 
5: upon event  $\langle \textit{beb}, \textit{Deliver} \mid p, [\textit{DATA}, s, m] \rangle$  do
6:   if  $m \notin \textit{delivered}$  then
7:     delivered := delivered  $\cup \{m\}$ 
8:     trigger  $\langle \textit{rb}, \textit{Deliver} \mid s, m \rangle$ 
9:     trigger  $\langle \textit{beb}, \textit{Broadcast} \mid [\textit{DATA}, s, m] \rangle$ 
```
