

## Algorithm – Eventually Perfect Failure Detector

---

**Algorithm 1** Increasing Timeout

---

**Implements:**

EventuallyPerfectFailureDetector, **instance**  $\diamond P$ .

**Uses:**

PerfectPointToPointLinks, **instance**  $pp2p$ .

```
1: upon event  $\langle \text{Init} \rangle$  do
2:    $alive := \Pi$ 
3:    $suspected := \emptyset$ 
4:    $delay := \Delta$ 
5:   STARTTIMER( $delay$ )
6: upon event  $\langle \text{Timeout} \rangle$  do
7:   if  $alive \cap suspected \neq \emptyset$  then
8:      $delay := delay + \Delta$ 
9:   for all  $p \in \Pi$  do
10:    if  $(p \notin alive) \wedge (p \notin suspected)$  then
11:       $suspected := suspected \cup \{p\}$ 
12:      trigger  $\langle \diamond P, Suspect \mid p \rangle$ 
13:    else if  $(p \in alive) \wedge (p \in suspected)$  then
14:       $suspected := suspected \setminus \{p\}$ 
15:      trigger  $\langle \diamond P, Restore \mid p \rangle$ 
16:    trigger  $\langle pp2p, Send \mid p, [\text{HEARTBEATREQUEST}] \rangle$ 
17:    $alive := \emptyset$ 
18:   STARTTIMER( $delay$ )
19: upon event  $\langle pp2p, Deliver \mid p, [\text{HEARTBEATREQUEST}] \rangle$  do
20:   trigger  $\langle pp2p, Send \mid p, [\text{HEARTBEATREPLY}] \rangle$ 
21: upon event  $\langle pp2p, Deliver \mid p, [\text{HEARTBEATREPLY}] \rangle$  do
22:    $alive := alive \cup \{p\}$ 
```

---

---

**Algorithm 2** Increasing Timeout with sequence numbers

---

**Implements:**EventuallyPerfectFailureDetector, **instance**  $\diamond P$ .**Uses:**PerfectPointToPointLinks, **instance**  $pp2p$ .

```
1: upon event  $\langle \text{Init} \rangle$  do
2:    $seqnum := 0$ 
3:    $alive := \Pi$ 
4:    $suspected := \emptyset$ 
5:    $delay := \Delta$ 
6:   STARTTIMER( $delay$ )
7: upon event  $\langle \text{Timeout} \rangle$  do
8:   if  $alive \cap suspected \neq \emptyset$  then
9:      $delay := delay + \Delta$ 
10:   $seqnum := seqnum + 1$ 
11:  for all  $p \in \Pi$  do
12:    if  $(p \notin alive) \wedge (p \notin suspected)$  then
13:       $suspected := suspected \cup \{p\}$ 
14:      trigger  $\langle \diamond P, \text{Suspect} \mid p \rangle$ 
15:    else if  $(p \in alive) \wedge (p \in suspected)$  then
16:       $suspected := suspected \setminus \{p\}$ 
17:      trigger  $\langle \diamond P, \text{Restore} \mid p \rangle$ 
18:    trigger  $\langle pp2p, \text{Send} \mid p, [\text{HEARTBEATREQUEST}, seqnum] \rangle$ 
19:   $alive := \emptyset$ 
20:  STARTTIMER( $delay$ )
21: upon event  $\langle pp2p, \text{Deliver} \mid p, [\text{HEARTBEATREQUEST}, n] \rangle$  do
22:   trigger  $\langle pp2p, \text{Send} \mid p, [\text{HEARTBEATREPLY}, n] \rangle$ 
23: upon event  $\langle pp2p, \text{Deliver} \mid p, [\text{HEARTBEATREPLY}, n] \rangle$  do
24:   if  $n = seqnum \vee p \in suspected$  then
25:      $alive := alive \cup \{p\}$ 
```

---