Algorithm – Ballot Leader Elector
Algorithm 1 Gossip Leader Election

Implements:

BallotLeaderElector, instance ble.

Uses:

PerfectPointToPointLinks, instance pp2p.

1: upon event ⟨ Init ⟩ do
2:   round := 0
3:   ballots := ∅
4:   ballot := (0, pid)
5:   leader := ⊥
6:   ballot_max := ballot
7:   delay := ∆
8:   startTimer(delay)
9: function checkLeader
10:   top := (topProcess, topBallot) := MaxByBallot(ballots ∪ {(self, ballot)})
11:   if topBallot < ballot_max then
12:      while ballot <= ballot_max do
13:         ballot := Increment(ballot)
14:      leader := ⊥
15:   else
16:      if top ≠ leader then
17:         ballot_max := topBallot
18:         leader := top
19:         trigger ⟨ ble, Leader | topProcess, topBallot ⟩
20: upon event ⟨ Timeout ⟩ do
21:   if ballots + 1 ≥ [Π 2 ] then
22:      checkLeader( )
23:   ballots := ∅
24:   round := round + 1
25:   for all p ∈ Π do
26:      if p ≠ self then
27:         trigger ⟨ pp2p, Send | p, [HeartbeatRequest, round, ballot_max] ⟩
28:      startTimer(delay)
29: upon event ⟨ pp2p, Deliver | p, [HeartbeatRequest, r, bmax] ⟩ do
30:   if bmax > ballot_max then
31:      ballot_max := bmax
32:      trigger ⟨ pp2p, Send | p, [HeartbeatReply, r, ballot] ⟩
33: upon event ⟨ pp2p, Deliver | p, [HeartbeatReply, r, b] ⟩ do
34:   if r = round then
35:      ballots := ballots ∪ {(p, b)}
36:   else
37:      delay := delay + ∆