Hierarchical Leader Election Algorithm With Remoteness Constraint

Mohamed Tbarka

October 20, 2019

Outline

Outline

What's Distributed Systems ?

A distributed system is a network that consists of autonomous computers that are connected using a distribution middleware. They help in sharing different resources and capabilities to provide users with a single and integrated coherent network.

The Bully Algorithm

As a first example, consider the bully algorithm devised by Garcia-Molina (1982). When any process notices that the coordinator is no longer responding to requests, it initiates an election. A process, P, holds an election as follows:

The Bully Algorithm

As a first example, consider the bully algorithm devised by Garcia-Molina (1982). When any process notices that the coordinator is no longer responding to requests, it initiates an election. A process, P, holds an election as follows:

- *P* sends an *ELECTION* message to all processes with higher numbers.

The Bully Algorithm

As a first example, consider the bully algorithm devised by Garcia-Molina (1982). When any process notices that the coordinator is no longer responding to requests, it initiates an election. A process, P, holds an election as follows:

- *P* sends an *ELECTION* message to all processes with higher numbers.
- If no one responds, P wins the election and becomes coordinator.

The Bully Algorithm

As a first example, consider the bully algorithm devised by Garcia-Molina (1982). When any process notices that the coordinator is no longer responding to requests, it initiates an election. A process, P, holds an election as follows:

- *P* sends an *ELECTION* message to all processes with higher numbers.
- If no one responds, P wins the election and becomes coordinator.
- If one of the higher-ups answers, it takes over. P's job is done.

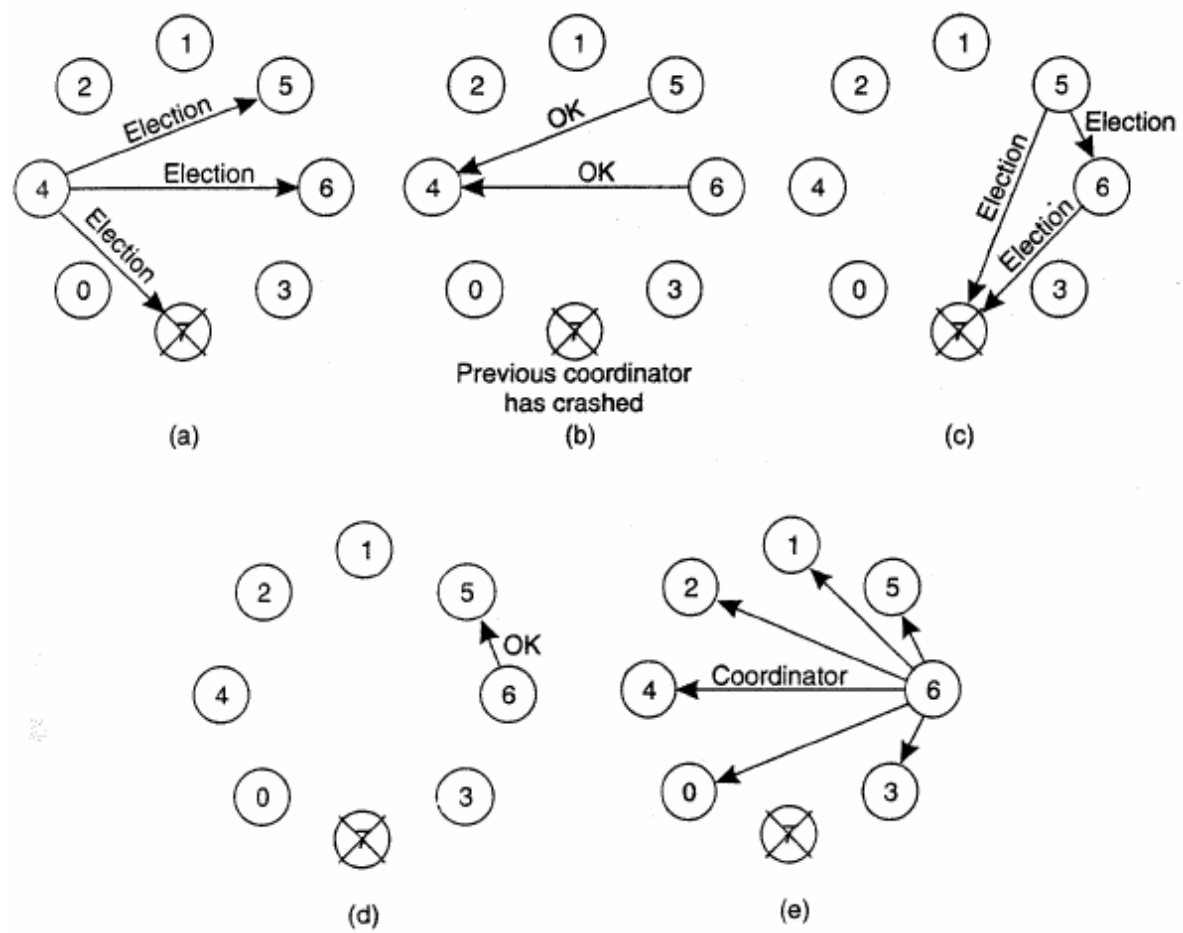

Figure: Bully Algorithm

Outline

System Model

- We assume a system consisting of a set P of computing nodes and a set χ of directed communication channels from one node to another node. χ consists of one channel.

System Model

- We assume a system consisting of a set P of computing nodes and a set χ of directed communication channels from one node to another node. χ consists of one channel.
- We model the whole system as a set of (infinite) state machines that interact through shared events (a specialization of the IOA model [17]).

Asynchronous Dynamic Links' Model

The state of Channel(u, v), which models the communication channel from node u to node v, consists of:

- a $status_{uv}$ variable;

Asynchronous Dynamic Links' Model

The state of Channel(u, v), which models the communication channel from node u to node v, consists of:

- a $status_{uv}$ variable;
- and a queue $mqueue_{uv}$ of messages.

Configurations and Executions

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

Outline

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.

Heights

The height for each node is a 7-tuple of integers $((\tau, oid, r), \delta, (nlts, lid), id)$, where the first three components are referred to as the reference level (RL) and the fifth and sixth.

- τ, a non-negative timestamp which is either 0 or the value of the causal clock time when the current search for an alternate path to the leader was initiated.
- oid, is a non-negative value that is either 0 or the id of the node that started the current search (we assume node ids are positive integers).
- r, a bit that is set to 0 when the current search is initiated and set to 1 when the current search hits a dead end.
- δ

The Code triggered by Update Message

When node u receives $Update(h)$ from node $v \in forming \cup N$:

```
      // if v is in neither forming nor N, message is ignored
 1.   height[v] := h
 2.   forming := forming \ {v}
 3.   N := N ∪ {v}
 4.   myOldHeight := height[u]
 5.   if ((nlts^u, lid^u) = (nlts^v, lid^v)) // leader pairs are the same
 6.       if (SINK)
 7.           if (∃ (τ,oid,r) | (τ^w, oid^w, r^w) = (τ,oid,r) ∀ w ∈ N)
 8.               if ((τ > 0) and (r = 0))
 9.                   REFLECTREFLEVEL
10.               else if ((τ > 0) and (r = 1) and (oid = u))
11.                   ELECTSELF
12.               else // (τ = 0) or (τ > 0 and r = 1 and oid ≠ u)
13.                   STARTNEWREFLEVEL
14.               end if
15.           else // neighbors have different ref levels
16.               PROPAGATELARGESTREFLEVEL
17.           end if
          // else not sink, do nothing
18.       end if
19.   else // leader pairs are different
20.       ADOPTLPIFPRIORITY(v)
21.   end if
22.   if (myOldHeight ≠ height[u])
```

Subroutines

ELECTSELF
1. $height[u] := (0,0,0,0,-\mathscr{T}_u,u,u)$

REFLECTREFLEVEL
1. $height[u] := (\tau,oid,1,0,nlts^u,lid^u,u)$

PROPAGATELARGESTREFLEVEL
1. $(\tau^u,oid^u,r^u) := max\{(\tau^w,oid^w,r^w) |\ w \in N\}$
2. $\delta^u := min\{\ \delta^w\ |\ w \in N$ **and** $(\tau^u,oid^u,r^u) = (\tau^w,oid^w,r^w)\} - 1$

STARTNEWREFLEVEL
1. $height[u] := (\mathscr{T}_u,u,0,0,nlts^u,lid^u,u)$

ADOPTLPIFPRIORITY(v)
1. **if** $((nlts^v < nlts^u)$ **or** $((nlts^v = nlts^u)$ **and** $(lid^v < lid^u)))$
2. $\quad height[u] := (\tau^v,oid^v,r^v,\delta^v+1,nlts^v,lid^v,u)$
3. **else send Update**$(height[u])$ **to** v
4. **end if**

Outline

Outline

What's JBotSim ?

Outline

Questions and Answers

Want to know more?

- Browse `http://web.mit.edu/smoot/history.htm`.

Questions and Answers

Want to know more?

- Browse `http://web.mit.edu/smoot/history.htm`.
- Smoot's Legacy `http://alum.mit.edu/news/AlumniNews/Archive/smoots_legacy`.

Questions and Answers

Want to know more?

- Browse `http://web.mit.edu/smoot/history.htm`.
- Smoot's Legacy `http://alum.mit.edu/news/AlumniNews/Archive/smoots_legacy`.
- Smoot Salute! `http://web.mit.edu/spotlight/smoot-salute`.