Leader Election

MPRI

Lélia Blin

lelia.blin@irif.fr

2023
Leader Election (LE)

leader election is the process of designating a single node as the organizer of some task distributed among several nodes.
Self-stabilization MPRI
Self-stabilization MPRI
Specifications problem
Small memory specification

**Definition problem:** The leader election specification sequence consists in a single specification configuration where a unique node maps to \( l_v = true \), and every other node \( u \neq v \) maps to \( l_u = false \).

<mark style="background: #FF5582A6;">Remark</mark>:
- Nobody knows the identifier of the elected node
- The size of the memory required for satisfying the election is \( O(1) \) bits per nodes
Best knowledge specification

**Definition problem**: The leader election specification sequence consists in a single specification configuration where a unique node maps to \( \ell_v = Id_v \), and every other node \( u \neq v \) maps to \( \ell_u = Id_v \).

<mark style="background: #FF5526;">Remark</mark>:
- Everybody knows the identifier of the elected node
- The size of the memory required for satisfying the election is \( O(\log n) \) bits per nodes
Impossibility result

LE is not possible in anonymous regular network
## State of art: silent

<table>
<thead>
<tr>
<th>Article</th>
<th>Scheduler</th>
<th>Knowledge</th>
<th>Rounds</th>
<th>Steps</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>AG90</td>
<td>weaklyfair</td>
<td>$N$</td>
<td>$O(N)$</td>
<td>?</td>
<td>$\Theta(\log N)$</td>
</tr>
<tr>
<td>DH97</td>
<td>Fair</td>
<td>$N$</td>
<td>$O(D)$</td>
<td>?</td>
<td>$O(N \log N)$</td>
</tr>
<tr>
<td>DLP10</td>
<td>Unfair</td>
<td>$O(n)$</td>
<td>?</td>
<td></td>
<td>unbounded</td>
</tr>
<tr>
<td>DLV11X2</td>
<td>Unfair</td>
<td>$O(n)$</td>
<td>?</td>
<td></td>
<td>$\Theta(\log n)$</td>
</tr>
<tr>
<td>KK13</td>
<td>Synchrone</td>
<td>$O(D)$</td>
<td>?</td>
<td></td>
<td>$\Theta(\log n)$</td>
</tr>
<tr>
<td>ACDDP17</td>
<td>Unfair</td>
<td>$O(n)$</td>
<td>$O(n^3)$</td>
<td>$\theta(\log n)$</td>
<td></td>
</tr>
</tbody>
</table>


A. Arora and M. Gouda, Distributed Reset (Extended Abstract):

Variables

- $r_v$ identifier of the root (the leader)
- $p_v$ identifier of the parent
- $d_v$ distance from the root
**Algorithm**

\[ R_{\text{root}} : (r_v < v) \lor (p_v = v \land (r_v \neq v \lor d_v \neq 0) \lor (p_v \notin N(v) \cup \{v\} \lor d_v \geq n) \]
\[ \rightarrow r_v = v, p_v = v, d_v = 0; \]

\[ R_{\text{correct}} : p_v \in N(v) \land d_v < n \land (r_v \neq r_{p_v} \lor d_v \neq d_{p_v} + 1) \]
\[ \rightarrow r_v = r_{p_v}, d_v = d_{p_v} + 1; \]

\[ R_{\text{parent}} : (\exists u \in N(v)|r_v < r_{p_v} \land d_v < n) \lor (\exists u \in N(v)|r_v = r_{p_v} \land d_v + 1 < d_v) \]
\[ \rightarrow r_v = r_j, p_v = j, d_v = d_j + 1; \]
Convergence
Silent or not?
K. Altisen, A. Cournier, S. Devismes, A. Durand, F. Petit,

Self-stabilizing leader election in polynomial steps.

Time Complexity: Analyse
Solution: Freeze before remove

- additional variable: $State_v \in \{C, EB, EF\}$
  - $C$ means "not involved in a tree removal"
    - Only process of status $C$ can join a tree and
    - only by choosing a process of status $C$ as parent
  - $EB$: Error Broadcast
  - $EF$: Error Feedback
Leader Election (Non Silent)

Goal: Sub-logarithmic memory size
# Non silent LE

<table>
<thead>
<tr>
<th>Art.</th>
<th>Topology</th>
<th>P/D</th>
<th>Model</th>
<th>Sched.</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOOY92</td>
<td>Ring</td>
<td>Proba.</td>
<td>Msg</td>
<td></td>
<td>$O(1)$</td>
</tr>
<tr>
<td>AO94</td>
<td>Ring</td>
<td>Proba.</td>
<td>link regist.</td>
<td></td>
<td>$O(\log^* n)/ed$</td>
</tr>
<tr>
<td>IL94</td>
<td>Ring</td>
<td>Proba.</td>
<td>State*</td>
<td></td>
<td>$O(1)$</td>
</tr>
<tr>
<td>ILS95</td>
<td>Ring**</td>
<td>Deter</td>
<td>State</td>
<td></td>
<td>$O(1)$</td>
</tr>
<tr>
<td>BGJ99</td>
<td>Ring++</td>
<td>Deter</td>
<td>State</td>
<td></td>
<td>$O(1)$</td>
</tr>
<tr>
<td>BT18</td>
<td>Ring</td>
<td>Deter.</td>
<td>State</td>
<td>unfair</td>
<td>$O(\log \log n)$</td>
</tr>
<tr>
<td>BT20</td>
<td>Graph</td>
<td>Deter.</td>
<td>State</td>
<td>unfair</td>
<td>$O(\log \log n + \log \Delta)$</td>
</tr>
</tbody>
</table>
* Augmented State Model ** Size of the ring is prime
++ \( n \)-node rings are bounded from above by \( n + k \), where \( k \) is a small constant.


Leader Election Bounds

**ILS95**: There exists a self-stabilizing leader election algorithm using $O(1)$ bits of memory per node in rings whose size is a prime number.

**BGJ99**: Any self-stabilizing leader election algorithm requires $\omega(1)$ bits of memory per node in rings whose size is not a prime number.

**DGS99**: Any silent self-stabilizing leader election algorithm requires at least $\Omega(\log n)$ bits of memory per node.
BT20: There exists a self-stabilizing leader election algorithm in any graph, using $O(\log \log n + \log \Delta)$ bits of memory per node.

BFL23: The leader election problem requires $\Omega(\log \log n)$ bits per node.

Compact memory [BT18]

- Compact identifier: decomposition bit per bit of the identifier
- Breaking symmetry: Identifier
- $\text{Bit}_v(i)$ function:
  - returns the position of the $i$-th most significant bit equal to 1 in $Id_v$
  - Exemple: 10→1010

$$\text{Bit}_v(i) := \begin{cases} 
4 & \text{if} \quad i = 1 \\
2 & \text{if} \quad i = 2 \\
-1 & \text{if} \quad i > 2
\end{cases}$$
Bitwise Comparison between identifiers

- $u=10 \ (1010)$ and $v=12 \ (1100)$
- $Bit_u(1) = Bit_v(1) = 4$
- $Bit_u(2) = 2 < Bit_v(2) = 3$
- The identifier of $u$ is greater than that of $v$
Main ingredients of BT20

1. Pointers: silent self-stabilizing distance-2 coloring
3. Fake root: silent self-stabilizing cycle and illegitimate sub spanning tree destruction [10, 13]
Pointers: silent self-stabilizing distance-2 coloring:

\[ O(\log \log n + \log \Delta) \]
Breaking cycle

Using the uniqueness of the identifiers

Fig. a

Fig. b
Silent self-stabilizing cycle and illegitimate sub spanning tree destruction
Rooted Spanning tree

Leader: Node with the maximum degree, the maximum color, the maximum identifier.
Memory Lower bound

BFL23: The leader election problem requires $\Omega(\log \log n)$ bits per node
Main Idea of BLF23

Idea

- **Algorithm $\mathcal{A}$**: $[n^c] \times \{0, 1\}^{3f(n)} \rightarrow \{0, 1\}^{f(n)}$
- $\forall id \in [1, n^c], \exists \delta_{id} : \{0, 1\}^{3f(n)} \rightarrow \{0, 1\}^{f(n)}$
- $|\mathcal{F}| = |\{\delta_{id}\}| = 2^{f(n)2^{3f(n)}}$
\[ f(n) \in o(\log \log n) \]

\[ ID \in [n^c] \]

\[ \delta \in \mathcal{F} \]
Challenges

**Silent LE**: Improve the number of steps

**LE**: Achieve the lower bound memory for graph with a non constant degree.

**Open question**: It is possible to achieve LE without constructing a spanning tree?