Leader Election Algorithm in Anonymous Rings: Franklin Goes Probabilistic

Rena Bakhshi

Theoretical Computer Science Section
Department of Computer Science
Vrije Universiteit Amsterdam

Milan, September 9, 2008
Joint work with

• Wan Fokkink
  • Vrije Universiteit Amsterdam

• Jun Pang
  • Université du Luxembourg

• Jaco van de Pol
  • University of Twente

Thanks to

• Bert Lisser
  • CWI, Amsterdam
Outline

Leader Election
- Franklin’s Leader Election Algorithm
- Leader Election in Anonymous Networks
- Our Leader Election Algorithm

Verification
- Verification with $\mu$CRL and CADP
- Partial order reduction with $\mu$CRL
- Performance evaluation with PRISM
Outline

**Leader Election**
- Franklin’s Leader Election Algorithm
- Leader Election in Anonymous Networks
- Our Leader Election Algorithm

**Verification**
- Verification with $\mu$CRL and CADP
- Partial order reduction with $\mu$CRL
- Performance evaluation with PRISM
(Deterministic) Leader Election

To break symmetry in a distributed system.

Assumptions

- Each node has a unique identity
- The identities are ordered
- System is fully asynchronous

Elect a node with max (or min) identity.

Definition

- The algorithm is decentralized
- Each node has the same local algorithm
- Upon termination one node is ‘leader’ and the rest are ‘lost’.
Franklin’s Leader Election Algorithm

A undirected ring.

Each active node:
  • sends its identity to its neighbors
  • compares its identity with the nearest active neighbors ids
  • if the identity is not the largest, node becomes passive

Passive nodes pass on messages

Repeat until node with the largest id receives its own message

Worst-case message complexity: $O(n \log n)$. 
Anonymous Networks

In some cases, nodes don’t have (unique) identities:

Example

• FireWire bus
  • to send ids can be costly
• Lego Mindstorms robots
  • CPUs don’t have identities

Assumptions

• Nodes are indistinguishable
  • No unique identities
  • Execute the same local algorithm
• Asynchronous communication
Leader Election in Anonymous Networks

Impossibility results:

• The knowledge of network size is needed
• Election with deterministic algorithm is impossible


Probabilistic algorithms can be used

• Node can pick random identity
• Message with a hop counter to detect its source.
Itai-Rodeh Election Algorithm

Based on Chang-Roberts algorithm.

A directed ring. All nodes know the ring size $n$.

A node selects a random identity, and sends it out

- When $u$ receives $v$
  - if $u < v$, $u$ becomes passive and passes on $v$
  - if $u > v$, $u$ purges the message

- When $u$ receives $u$
  - if hop counter is $n$, it becomes the leader
  - otherwise, passes on the message with a 'dirty' bit

If several nodes picked the same largest identity

- they start a new election round

Round number is a part of:

- the state of a node
- message
Probabilistic Finite-State Leader Election Algorithm

Itai-Rodeh algorithm with rounds is infinite-state algorithm.
  • Avoid round numbers using FIFO channels

Our contribution
  • use undirected ring with round numbers modulo 2

Assumptions

  • An anonymous, undirected ring
  • Asynchronous communication
  • Message order is not preserved between any pair of nodes
  • Reliable channels
  • Fair scheduler for message queues
    • Sent message will eventually be processed at its destination.
  • All nodes know the ring size $n$. 
Our Algorithm

Each node is either active, passive or leader
An active node maintains two parameters:
  • identity, not necessarily unique
  • the number of the current election round (modulo 2)
All messages are
  • of the form \( \langle id, hop, bit \rangle \)
    • \( id \) is the originator identity
    • \( bit \) is the election round of the owner (modulo 2)
    • \( hop \leq n \) is hop counter, used to detect identity clashes
  • travelling in both directions
Passive nodes simply pass on messages
  • increasing hop counter by one
Initially, all nodes are active and their round number $bit = T$.

At the start of a round an active node picks a random identity
The Algorithm

and sends the message $\langle id, 1, bit \rangle$ in both directions
Upon receipt of a message \( \langle id, hop < n, bit \rangle \), an active node

- stores it, and
- waits for a message from the other direction.
Upon receipt of messages from both sides, an *active* node
- becomes passive, if any of the ids is larger than its own
- otherwise, it starts a new election round with an inverted round number and a new identity
Upon receipt of a message $\langle id, hop, bit \rangle$

- a passive node passes on a message $\langle id, hop + 1, bit \rangle$
The Algorithm

\[ \langle 0, 2, F \rangle \rightarrow 2(F) \rightarrow \langle 2, 1, F \rangle \rightarrow \langle 1, 2, F \rangle \rightarrow 1(F) \rightarrow \langle 0, 2, F \rangle \]
The Algorithm

\[ \langle 2, 1, T \rangle \xrightarrow{2(T)} \langle 2, 1, T \rangle \]
The Algorithm

\[ 2(T) \]

\[ \langle 2, 2, T \rangle \]
The Algorithm

\[ 2(T) \]

\[ \langle 2, 3, T \rangle \]

\[ \langle 2, 3, T \rangle \]
The Algorithm

\[ 2(T) \]

\[ \langle 2, 4, T \rangle \]
Upon receipt of a message $\langle \text{id}, \text{hop} = n, \text{bit} \rangle$

- an active node becomes the leader
The Algorithm

2
Why Round Numbers?

When channels are FIFO, round numbers are not needed.

For instance, shown for Itai-Rodeh leader election algorithm


But for our algorithm, the message order is not preserved between any pair of nodes
Why Round Numbers?

\[ u \rangle \langle u, 1 \rangle \]

\[ u \rangle \langle u, 1 \rangle \]

\[ v \rangle \langle v, 1 \rangle \]

\[ u > v > w \]
Why Round Numbers?

\[ \langle w, 1 \rangle \]

\[ \langle w, 1 \rangle \]

\[ \langle w, 1 \rangle \]

\[ \langle w, 1 \rangle \]

\[ \langle u, 1 \rangle \]

\[ \langle u, 1 \rangle \]

\[ \langle u, 1 \rangle \]

\[ v \]

\[ u > v > w \]
Why Round Numbers?

\[ \langle u, 1 \rangle \quad w \quad \langle w, 1 \rangle \quad w \quad \langle u, 1 \rangle \]

\[ \langle v, 1 \rangle \quad v \quad \langle v, 1 \rangle \]

\[ u > v > w \]
Why Round Numbers?

\[ \langle u, 1 \rangle \]

\[ v \]

\[ \langle u, 1 \rangle \]

\[ u > v > w \]
Why Round Numbers?

\[ u > v > w \]
Why Undirected Algorithm?

Dolev, Klawe and Rodeh (also Peterson) adapted Franklin’s idea to a directed ring.

• An active node only progresses to the next election round if its identity is larger than the identities of two left active consecutive nodes.

In our probabilistic version of DKR algorithm

• identities are picked at random
• hop counts are used to decide the leader
• labels one and two are used to identify the message originator
• a round number modulo 2

But round numbers modulo 2 are not sufficient for probabilistic DRK algorithm.
Probabilistic Dolev-Klawe-Rodeh

Message $\langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle$
Probabilistic Dolev-Klawe-Rodeh

Message \( \langle id, label, hop, round \rangle \)
Message $\langle id, label, hop, round \rangle$
Probabilistic Dolev-Klawe-Rodeh

Message \langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle
Message $\langle id, label, hop, round \rangle$
Message $\langle id, label, hop, round \rangle$
Probabilistic Dolev-Klawe-Rodeh

Message \( \langle id, label, hop, round \rangle \)
Probabilistic Dolev-Klawe-Rodeh

Message $\langle id, label, hop, round \rangle$
Message $\langle id, label, hop, round \rangle$
Message $\langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle$
Message \( \langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle \)
Probabilistic Dolev-Klawe-Rodeh

Message $\langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle$
Message $\langle \text{id}, \text{label}, \text{hop}, \text{round} \rangle$
Probabilistic Dolev-Klawe-Rodeh

Message \( \langle id, label, hop, round \rangle \)
Probabilistic Dolev-Klawe-Rodeh

Message $\langle id, label, hop, round \rangle$
Probabilistic Dolev-Klawe-Rodeh

Message \(\langle id, label, hop, round \rangle\)
Probabilistic Dolev-Klawe-Rodeh

Message $\langle id, label, hop, round \rangle$
Message $\langle id, label, hop, round \rangle$
Message $\langle id, label, hop, round \rangle$
Message \( \langle id, label, hop, round \rangle \)
The probabilistic Franklin algorithm for anonymous undirected rings terminates with probability one, and upon termination a unique leader has been elected.

Shown using model checking analysis with $\mu$CRL and CADP

- up to ring size 6
- distributed $\mu$CRL used for 6 nodes (2 identities) and 5 nodes (3 identities)
- branching bisimulation equivalence abstracts away from infinite executions
- minimized state space of the algorithm has two states and “leader” action as transition
Generated State Space Statistics

- State space for two identities

<table>
<thead>
<tr>
<th># Procs</th>
<th>States</th>
<th>Transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>657</td>
<td>1,368</td>
</tr>
<tr>
<td>3</td>
<td>15,445</td>
<td>43,968</td>
</tr>
<tr>
<td>4</td>
<td>380,609</td>
<td>1,396,512</td>
</tr>
<tr>
<td>5</td>
<td>9,819,065</td>
<td>44,242,920</td>
</tr>
<tr>
<td>6</td>
<td>260,753,105</td>
<td>1,393,967,976</td>
</tr>
</tbody>
</table>

- State space for three identities

<table>
<thead>
<tr>
<th># Procs</th>
<th>States</th>
<th>Transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1,525</td>
<td>3,564</td>
</tr>
<tr>
<td>3</td>
<td>55,009</td>
<td>168,102</td>
</tr>
<tr>
<td>4</td>
<td>2,095,777</td>
<td>8,182,092</td>
</tr>
<tr>
<td>5</td>
<td>84,381,157</td>
<td>401,681,445</td>
</tr>
</tbody>
</table>
To reduce the state space for model checking

- Each node first reads from the left and then from the right
- Static analysis
  - Elimination of constant node parameters (constelm)
- Finding and marking confluent summands (confcheck)
- Symbolic confluence reduction (confelm)
- On-the-fly $\tau$-reduction (--confluence ctau option with instantiator)
Partial Order Reduction

- State space for two identities

<table>
<thead>
<tr>
<th>Strategy</th>
<th>#</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
<td>s.</td>
<td>385</td>
<td>7,613</td>
<td>152,065</td>
<td>3,162,337</td>
<td>67,758,817</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>664</td>
<td>17,880</td>
<td>459,488</td>
<td>11,736,100</td>
<td>298,484,184</td>
</tr>
<tr>
<td>confelm</td>
<td>s.</td>
<td>205</td>
<td>2,875</td>
<td>40,881</td>
<td>606,783</td>
<td>9,280,633</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>340</td>
<td>6,342</td>
<td>114,384</td>
<td>2,069,040</td>
<td>37,381,488</td>
</tr>
<tr>
<td>on-the-fly</td>
<td>ext. s.</td>
<td>165</td>
<td>1,819</td>
<td>21,409</td>
<td>263,963</td>
<td>3,348,345</td>
</tr>
<tr>
<td></td>
<td>int. s.</td>
<td>181</td>
<td>2,343</td>
<td>30,039</td>
<td>395,723</td>
<td>5,350,021</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>276</td>
<td>4,086</td>
<td>60,576</td>
<td>902,820</td>
<td>13,449,324</td>
</tr>
</tbody>
</table>

- State space for three identities

<table>
<thead>
<tr>
<th>Strategy</th>
<th>#</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
<td>s.</td>
<td>877</td>
<td>26,299</td>
<td>802,489</td>
<td>25,919,965</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>1,680</td>
<td>65,853</td>
<td>2,560,848</td>
<td>100,868,445</td>
</tr>
<tr>
<td>confelm</td>
<td>s.</td>
<td>469</td>
<td>9,874</td>
<td>214,957</td>
<td>4,952,449</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>876</td>
<td>23,310</td>
<td>637,884</td>
<td>17,778,660</td>
</tr>
<tr>
<td>on-the-fly</td>
<td>ext. s.</td>
<td>385</td>
<td>6,400</td>
<td>116,785</td>
<td>2,242,609</td>
</tr>
<tr>
<td></td>
<td>int. s.</td>
<td>433</td>
<td>8,518</td>
<td>170,131</td>
<td>3,524,305</td>
</tr>
<tr>
<td></td>
<td>t.</td>
<td>732</td>
<td>15,570</td>
<td>353,508</td>
<td>8,137,080</td>
</tr>
</tbody>
</table>
An active node chooses a fresh identity at the start of:

- A: each election round.
- B: new election round only if an identity clash is detected.

---

**Performance Evaluation with PRISM**

---
Thank you!

Thank you for your attention!

Questions?