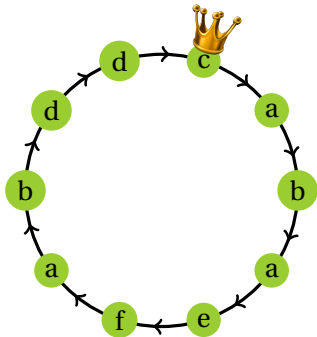


Leader Election in Rings with Bounded Multiplicity (Short Paper)

Karine Altisen, Ajoy K. Datta, Stéphane Devismes, **Anaïs Durand**, and
Lawrence L. Larmore

November 8, 2016

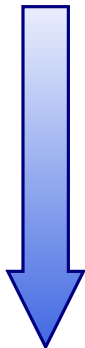


- Leader election
- Unidirectional rings
- Homonym processes
- Deterministic algorithm
- Message-passing model
- Process-terminating algorithm

Leader Election in Rings

- **Anonymous processes:**
 - ▶ Deterministic solution: Impossible [Angluin, 80], [Lynch, 96]

Leader Election in Rings



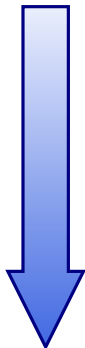
■ **Anonymous processes:**

- ▶ Deterministic solution: Impossible [Angluin, 80], [Lynch, 96]
- ▶ Probabilistic solution: [Xu and Srimani, 06], [Kutten *et al.*, 13]

■ **Identified processes:**

- ▶ Deterministic solution: [LeLann, 77], [Chang and Roberts, 79], [Petersen, 82] ...

Leader Election in Rings



■ **Anonymous processes:**

- ▶ Deterministic solution: Impossible [Angluin, 80], [Lynch, 96]
- ▶ Probabilistic solution: [Xu and Srimani, 06], [Kutten *et al.*, 13]

■ **Homonym processes**

■ **Identified processes:**

- ▶ Deterministic solution: [LeLann, 77], [Chang and Roberts, 79], [Petersen, 82] ...

Leader Election in Rings of Homonym Processes

	PT/MT	Asynch.	Unidir./Bidir.	Know	Ring Class	# Msg	Time
[Delporte <i>et al.</i> , 14]	MT	✓	Bidir.		# labels > greatest proper divisor of n	?	?
	PT	✓		n		$O(n \log n)$?
[Dobrev and Pelc, 04]	PT	✗	Bidir. + unidir.	$m \leq n$	Decide if inputs are unambiguous	$O(n \log n)$	$O(M)$
		✓	Bidir.	$M \geq n$		$O(nM)$?
[Flocchini <i>et al.</i> , 04]	PT	✓	Bidir.	n	Prime n , 2 labels, Asymmetric ring	?	?

- MT = Message-terminating: Processes do not terminate but only a finite number of messages are exchanged.
- PT = Process-terminating: Every process eventually halts.

■ Ring classes:

- ▶ \mathcal{U}^* : at least one unique label
- ▶ \mathcal{K}_k : multiplicity of labels bounded by k

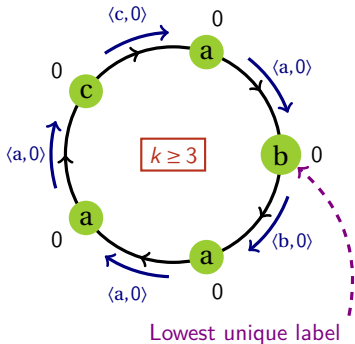
■ Message-terminating leader election:

- ▶ Impossible in \mathcal{K}_k
- ▶ Impossible in \mathcal{U}^* (work under submission)

■ Process-terminating leader election algorithm for $\mathcal{U}^* \cap \mathcal{K}_k$:

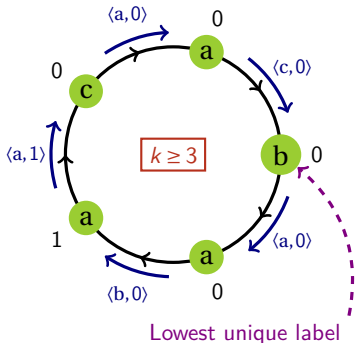
- ▶ Time complexity: at most $n(k+2)$
- ▶ # messages: $O(n^2 + kn)$
- ▶ Memory requirement: $\lceil \log(k+1) \rceil + \log(n) + 4$

Algorithm



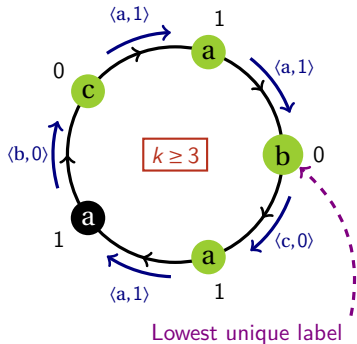
- Counter = rough estimation of the multiplicity

Algorithm



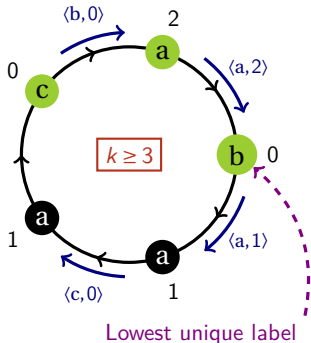
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique

Algorithm



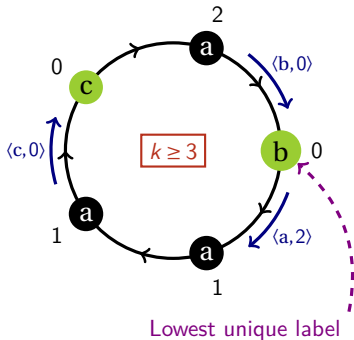
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



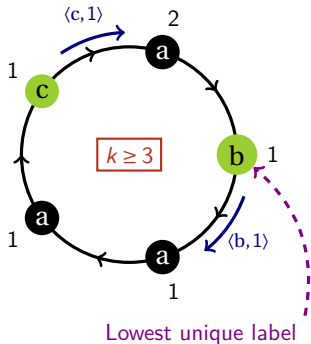
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



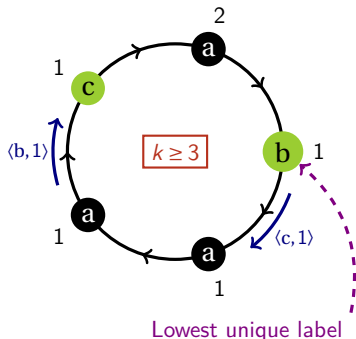
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



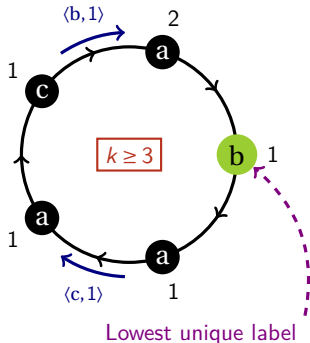
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



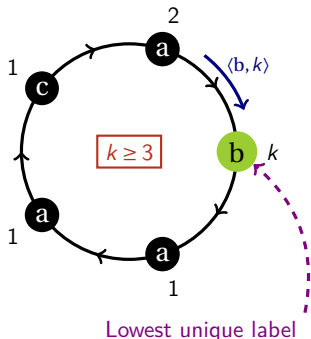
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
 - ▶ Same counter $\neq 0$, lower ID \rightarrow not lowest unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



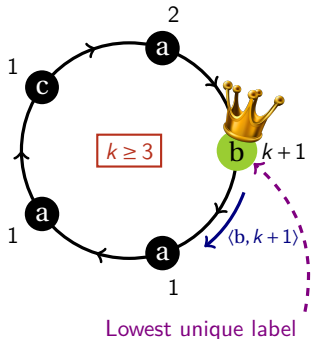
- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
 - ▶ Same counter $\neq 0$, lower ID \rightarrow not lowest unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant

Algorithm



- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
 - ▶ Same counter $\neq 0$, lower ID \rightarrow not lowest unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant
- Election detection: receiving $\langle id, k \rangle$

Algorithm



- Counter = rough estimation of the multiplicity
- Process elimination:
 - ▶ Lower counter, \neq ID \rightarrow not unique
 - ▶ Same counter $\neq 0$, lower ID \rightarrow not lowest unique
- Message elimination:
 - ▶ Passive, same ID \rightarrow not relevant
- Election detection: receiving $\langle id, k \rangle$

Process-terminating leader election algorithm for $\mathcal{U}^* \cap \mathcal{K}_k$

- **Time complexity:** at most $n(k+2)$
Asymptotically optimal (work under submission)
- **# messages:** $O(n^2 + kn)$
- **Memory requirement:** $\lceil \log(k+1) \rceil + \log(n) + 4$

Thank you for your attention.



Do you have any questions ?