Computing on a Partially Eponymous Ring*

Marios Mavronicolas[1], Loizos Michael[2], and Paul Spirakis[3]

[1] Department of Computer Science, University of Cyprus, Nicosia CY-1678, Cyprus, and Faculty of Computer Science, Electrical Engineering and Mathematics, University of Paderborn, 33102 Paderborn, Germany. mavronic@cs.ucy.ac.cy

[2] Division of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, U.S.A. loizos@eecs.harvard.edu

[3] Department of Computer Engineering and Informatics, University of Patras, 265 00 Patras, Greece, and Research and Academic Computer Technology Institute, 265 00 Patras, Greece. spirakis@cti.gr

Abstract. We study the *partially eponymous* model of distributed computation, which simultaneously generalizes the *anonymous* and the *eponymous* models. In this model, processors have *identities*, which are neither necessarily all identical, nor necessarily unique; processors receive *inputs* and must reach *outputs* that respect a *relation*. We focus on the *partially eponymous ring* R, and we are interested in the computation of *circularly symmetric* relations on it.

• We distinguish between solvability and computability: in *solvability*, processors must *always* reach outputs that respect the relation; in *computability*, they must reach outputs that respect the relation whenever possible, and report impossibility otherwise.

- We provide an *efficient* characterization of solvability of an arbitrary (circularly symmetric) relation on an arbitrary set of rings. The characterization is topological and can be expressed as a number-theoretic property that can be checked efficiently.
- We present a *universal* distributed algorithm for computing any arbitrary (circularly symmetric) relation on any set of rings.

• Towards obtaining *message complexity* bounds, we derive a distributed algorithm for a natural generalization of *Leader Election*, in which a (nonzero) number of leaders are elected. We use this algorithm as a subroutine of our universal algorithm for collecting views; hence, we prove, as our main result, an upper bound on the message complexity of this particular instantiation of our universal algorithm to compute an *arbitrary* (circularly symmetric) relation on an *arbitrary* set of rings. The shown upper bound demonstrates a graceful degradation with the *Least Minimum Base*, a parameter indicating the degree of topological compatibility between the relation and the set of rings. We employ this universal upper bound to identify two interesting cases where an arbitrary relation can be computed with an efficient number of $O(|R| \cdot \lg |R|)$ messages: The set of rings is *universal* (which allows the solvability of Leader Election), or *logarithmic* (where each identity appears at most $\lg |R|$ times).

* This work has been partially supported by the IST Program of the European Union under contract number IST-15964 (AEOLUS).

1 Introduction

Motivation and Framework. Two of the best studied models in Distributed Computing Theory are the *eponymous* model and the *anonymous* model. In both models, *processors* may receive *inputs* and must reach *outputs* that are related to the inputs according to some (*recursive*) *relation*.

• In the eponymous model, processors have unique *identities*. This availability enables the computability of all relations: processors first solve *Leader Election* [11] to elect a leader among them; then, the leader undertakes computation of the relation and communicates the solution to the other processors.

• In the anonymous model, processors have *identical* identities and they run the same local algorithm. The impossibility of breaking this initial symmetry retains many relations unsolvable in the anonymous model; the prime example is the impossibility of solving *Leader Election* on an anonymous ring [1].

This long-known separation between the eponymous model and the anonymous model invites the investigation of an intermediate model, where there are identities available to the processors, but these are neither necessarily unique, nor necessarily all identical. Call this intermediate model the *partially eponymous* model. We consider a particular case of the partially eponymous model, that of the (*asynchronous*) *partially eponymous ring* R with *bidirectional* communication and *orientation*. Which relations are solvable on the partially eponymous ring? For *which message complexity* can an arbitrary relation be computed? (*Bit complexity* remains beyond the scope of this work.)

We focus on *circularly symmetric relations*, the broadest class of relations that are natural to consider for rings. Roughly speaking, in a circularly symmetric relation, shifting any *output vector* for a given *input vector* must yield a correct output vector for the correspondingly shifted input vector.

An essential attribute of most previous work on anonymous networks has been the requirement that the distributed algorithm for a particular relation runs on *all* networks and occasionally reports impossibility (exactly, of course, when it is impossible to return *admissible* outputs — ones that respect the relation). This concept will be called *computability* in this work. An orthogonal viewpoint is to actually isolate the subclass of networks on which it is always possible to return admissible outputs, in order to obtain tailored algorithms that are possibly more *efficient* (in terms of message complexity) than those running on *all* networks. This motivation leads to the concept of *solvability*: a relation is *solvable* on a set of networks if there is a distributed algorithm which, when run on any network in the set, leads all processors to reach admissible outputs (and *never* report impossibility).

Previous Work. Computation on anonymous networks was first studied in the seminal work of Angluin [1], where the impossibility of solving Leader Election was first established. Yamashita and Kameda [12, 13] have considered the solvability of several representative distributed computing problems on anonymous networks and characterized the class of (anonymous) networks on which each problem is solvable under different assumptions on the network attributes (e.g., size, topology, etc.) that are made available to the processors.

The more general model of an arbitrary partially eponymous network has been first considered by Yamashita and Kameda [15]. They focused on Leader Election and provided a graph-theoretic characterization of its solvability under different assumptions on the communication mode and the available (a)synchrony. Further work on the partially eponymous ring has been carried out in [6, 7]. Chalopin *et al.* [6] very recently considered some specific generalizations of Leader Election in an arbitrary partially eponymous network. Under the assumption that processors have an approximate knowledge of the ring size, Dobrev and Pelc [7] presented both lower and upper bounds on message complexity for both the synchronous and the asynchronous cases of a partially eponymous ring.

Boldi and Vigna [3] have considered the more general solvability problem for an arbitrary relation on an arbitrary network and for any level of knowledge and anonymity (or *eponymity*) of the processors.

Attiya *et al.* [2] initiated the study of computing functions on the asynchronous anonymous ring. Flocchini *et al.* [8] consider the problems of Leader Election, *Edge Election* and *Multiset Sorting* on the asynchronous anonymous ring R where processors are distinguished by *input values* that are not necessarily distinct. So, input values are treated in the partially eponymous model of Flocchini *et al.* [8] as either identities or as inputs. We emphasize that the partially eponymous model of Flocchini *et al.* [8] does not simultaneously consider identities and inputs. Under the assumptions that input values are binary and $|R|$ is prime, Flocchini *et al.* [8, Theorems 4.1 and 4.2] provide lower and upper bounds on message complexity for these three problems. The lower and upper bounds are $\Omega(\sum_j (z_j^2 + t_j^2))$ and $O(\sum_j (z_j^2 + t_j^2) + |R| \cdot \lg |R|)$, respectively, where z_j and t_j are the lengths of consecutive blocks of 1's and 0's, respectively, in the vector of binary inputs.

The first efficient algorithm for Leader Election in the eponymous ring is based on the intuitive idea of domination in neighborhoods with progressively doubling size, which is due to Hirschberg and Sinclair [9]; it achieves an $O(|R| \cdot \lg |R|)$ upper bound on message complexity. A corresponding lower bound of $\Omega(|R| \cdot \lg |R|)$ has been established in [5].

Contribution. We start by studying solvability and computability.

• We discover (Theorem 1) that solvability is equivalent to *compatibility*, a new abstract, topological concept we introduce to capture the *possibility* that symmetries present in the initial configuration, comprised of the identities and the inputs, persist to the reached outputs.

To measure the initial symmetry, we use the *period* of a vector consisting of the identities and the inputs; the smaller the symmetry, the longer the period, and we call it the *Minimum Base*. It turns out that Minimum Base enjoys an elegant number-theoretic expression allowing for its efficient evaluation. Similarly, we measure the symmetry in an output vector using its period. The possibility of persistence of initial symmetries to the final symmetries amounts to demanding that the period of *some* admissible output vector divides that of the initial vector, and this is our definition of compatibility. Compatibility can be checked efficiently (since it reduces to a number-theoretic property that can be checked efficiently); hence, our characterization of solvability is an *efficient* one.

• We present a *universal* algorithm for computing any arbitrary (circularly symmetric) relation on any set of rings (Theorem 2). This algorithm is comprised of any distributed algorithm for collecting *views* at each processor, followed by local steps specific to the particular relation.

• As an application of our characterization of solvability, we derive a particular characterization of solvability for (circularly symmetric) *aperiodic* relations (Theorem 3); Leader Election is an example of such relations. So, this determines a topological characterization of the class of relations that are equivalent to Leader Election with respect to solvability.

We then study message complexity (with respect to computability).

• As our chief algorithmic instrument, we present a distributed algorithm for a natural generalization of Leader Election in which a (non-zero) number of leaders must be elected; call it *Multiple Leader Election* (Proposition 2). This algorithm works correctly on a given configuration when advised with a lower bound k on the Minimum Base for that configuration. This distributed algorithm exploits the idea of *doubling neighborhoods* from the distributed algorithm of Hirschberg and Sinclair [9] for solving Leader Election on the eponymous ring; it achieves message complexity $O(|R| \cdot \lg k)$ (Proposition 1) for any advice k.

• In turn, we use the distributed algorithm for Multiple Leader Election to construct a *universal* algorithm to compute an arbitrary (circularly symmetric) relation Ψ on a set of rings $\mathbf{ID}$ (consisting of all rings with the same, arbitrarily chosen, identity multiplicities). The universal algorithm has message complexity $O\left((n^2/\mathsf{LMB}(\mathbf{ID}, \Psi)) + n \cdot \lg \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$, where $\mathsf{LMB}(\mathbf{ID}, \Psi)$ is the *Least Minimum Base* — the least value of Minimum Base over all configurations with rings coming from $\mathbf{ID}$ and input vectors coming from the domain of Ψ (Theorem 4). Here, $\mathsf{LMB}(\mathbf{ID}, \Psi)$ is used as the advice k for the distributed algorithm to solve Multiple Leader Election. (Note that this is permissible when designing a distributed algorithm to compute the relation Ψ on the set of rings $\mathbf{ID}$.) Interestingly, the established upper bound demonstrates that the message complexity on rings of size n degrades gracefully with the Least Minimum Base, ranging from $O(n \cdot \lg n)$ for the eponymous ring to $O(n^2)$ for the anonymous ring. So, our universal upper bound is *tight* for these two extreme models.

• We are finally interested in determining sets of rings on which the universal upper bound on message complexity from Theorem 4 is low. In particular, on which sets of rings (of size n) is an upper bound of $O(n \cdot \lg n)$ possible? We identify two such (*incomparable*) classes of sets:

- Say that a set of rings is *universal* if Leader Election is solvable on it. So, every relation is solvable on such a universal set. We prove that a relation is computable with $O(n \cdot \lg n)$ messages on a universal set of rings (Theorem 5). Hence, surprisingly, Leader Election is either unsolvable on a given set of rings, or efficiently computable on the given set with $O(n \cdot \lg n)$ messages.
- Say that a set of rings is *logarithmic* if each identity appears at most $\lg n$ times. We prove that a relation is computable with $O(n \cdot \lg n)$ messages on a

logarithmic set of rings (Theorem 6); note that this holds even if the relation is *not* solvable on that set of rings.

Comparison to Directly Related Work. Whereas Boldi and Vigna [3] provide an **effective** characterization of anonymous solvability for any arbitrary relation on any arbitrary network, our work provides the *first* **efficient** characterization of partially eponymous solvability for any arbitrary relation on the ring (Theorem 4.1). It is not evident how the effective graph-theoretic characterization from [3] (involving *graph coverings* and *graph fibrations*) could yield an efficient characterization for the special case of the partially eponymous ring. In fact, our main goal has been to derive a *direct* solvability characterization for the particular case of the partially eponymous ring that bypasses the complex framework of graph coverings and graph fibrations developed in [3] for the general case of an arbitrary network. Although the work in [3] invests a great effort in translating concepts of Distributed Computing into some complex graph-theoretic form, our proof techniques for the solvability characterization are elementary.

Theorem 4 improves [8, Theorem 4.2] in three fronts. First, it works for an *arbitrary* ring size $|R|$, while [8, Theorem 4.2] assumes that $|R|$ is prime. Second, [8, Theorem 4.2] assumes binary inputs, while Theorem 4 makes no assumption on either inputs or identities. Third, and most important, Theorem 4 applies to *any* arbitrary relation, while [8, Theorem 4.2] is tailored to three specific relations (Leader Election, Edge Election and Multiset Sorting). We remark, however, that the worst-case message complexity in both Theorem 4 and [8, Theorem 4.2] is $\Theta(|R|^2)$. Note also that [8, Theorem 5.1] is the special case of Theorem 3 where Ψ is the Leader Election Relation.

Our definition for Least Minimum Base is built on top of *Minimum Base*, originally defined in [4, 10] and used in [3, 6] to obtain characterizations of solvability in anonymous networks. However, we exploit the very simple structure of the ring network to derive and use a particularly simple version of Minimum Base. For the case of the anonymous ring, Attiya *et al.* [2] defined the *Symmetry Index* to measure the symmetry in an initial configuration (containing only inputs); in contrast, (Least) Minimum Base measures asymmetry, while also taking identities into account.

Dobrev and Pelc [7, Theorem 3.1] prove an $\Omega(M \cdot n)$ lower bound on message complexity for the computability of Leader Election on the partially eponymous ring, where M is an upper bound on the ring size known to the processors; this implies a corresponding $\Omega(n^2)$ lower bound when the ring size is known exactly. This lower bound applies to the class of all rings; hence, it does not contradict the upper bound in Theorem 4, which applies to a set of rings **ID**.

2 Mathematical Preliminaries

Denote $\mathbb{N} = \{0, 1, 2, \ldots\}$, $\mathbb{Z}^+ = \{1, 2, 3, \ldots\}$, and $[n] = \{0, 1, \ldots, n-1\}$ for each integer $n \geq 1$. Denote GCD and LCM the functions mapping a set of integers to their *Greatest Common Divisor* and *Least Common Multiple*, respectively. We assume a global, possibly infinite set Σ (containing 0 and 1), and we consider a *vector* $\mathbf{x} = \langle x_0, x_1, \ldots, x_{n-1}\rangle \in \Sigma^n$. λ denotes the empty vector, while $\mathbf{x} \diamond \mathbf{y}$ denotes the concatenation of vectors $\mathbf{x}$ and $\mathbf{y}$. With each vector $\mathbf{x}$, we associate a multiset $\mathsf{M}(\mathbf{x})$ with the multiplicities of the entries of $\mathbf{x}$. We use the function M to partition Σ^n into equivalence classes, where all vectors in an equivalence class have the same image under M. Denote $\mathbf{X}$ the equivalence class containing the vector $\mathbf{x}$. By abuse of notation, $\mathsf{M}(\mathbf{X})$ will denote $\mathsf{M}(\mathbf{x})$ for any $\mathbf{x} \in \mathbf{X}$.

For any integer $k \in [n]$, the ***(cyclic) shift*** $\sigma_k(\mathbf{x})$ of vector $\mathbf{x}$ is the vector $\langle x_k, x_{k+1}, \ldots, x_{k+n-1}\rangle$, with indices taken modulo n; so, σ_k shifts $\mathbf{x}$, k places anti-clockwise. The definition is extended to all integers k in the natural way.

The ***period*** $\mathsf{T}(\mathbf{x})$ of vector $\mathbf{x}$ is the *least* integer $k, 0 < k \leq n$, such that $\sigma_k(\mathbf{x}) = \mathbf{x}$. Say that $\mathbf{x}$ is $\mathsf{T}(\mathbf{x})$-periodic; $\mathbf{x}$ is ***aperiodic*** if $\mathsf{T}(\mathbf{x}) = n$, and $\mathbf{x}$ is ***uniperiodic*** if $\mathsf{T}(\mathbf{x}) = 1$. Say that $\mathbf{x}$ is ***eponymous*** if each entry of $\mathbf{x}$ is unique; clearly, an eponymous vector is aperiodic, but not vice versa. Say that $\mathbf{x}$ is ***anonymous*** if all entries of $\mathbf{x}$ are identical; so, a vector is anonymous if and only if it is uniperiodic. Clearly, the period of a vector is invariant under shifting. So, the period captures the degree of circular asymmetry of a vector: the smaller the period, the more circular symmetries the vector has. We prove:

Lemma 1. *For each vector $\mathbf{x} \in \Sigma^n$, and $l, m \in \mathbb{N}$, $\sigma_l(\mathbf{x}) = \sigma_m(\mathbf{x})$ if and only if $l \equiv m \pmod{\mathsf{T}(\mathbf{x})}$.*

Call a vector $\widetilde{\mathbf{x}} \in \mathbf{X}$ a ***min-period*** vector of $\mathbf{X}$ if it minimizes period among all vectors in $\mathbf{X}$. We prove:

Lemma 2. *For each equivalence class $\mathbf{X} \subseteq \Sigma^n$, (1) $\mathsf{T}(\widetilde{\mathbf{x}}) = n/\mathsf{GCD}(\mathsf{M}(\mathbf{X}))$, and (2) for each vector $\mathbf{x} \in \mathbf{X}$, $\mathsf{T}(\widetilde{\mathbf{x}})$ divides $\mathsf{T}(\mathbf{x})$.*

Say that $\mathbf{X}$ is ***aperiodic*** if each vector $\mathbf{x} \in \mathbf{X}$ is aperiodic; say that $\mathbf{X}$ is ***uniperiodic*** if each vector $\mathbf{x} \in \mathbf{X}$ is uniperiodic. Say that $\mathbf{X}$ is ***k-periodic*** if the min-period vector $\widetilde{\mathbf{x}}$ of $\mathbf{X}$ is k-periodic. (So, aperiodic and uniperiodic are identified with n-periodic and 1-periodic, respectively.) Clearly, Lemma 2 (condition (1)) implies that $\mathbf{X}$ is $(n/\mathsf{GCD}(\mathsf{M}(\mathbf{X})))$-periodic. Hence, $\mathbf{X}$ is aperiodic if and only if $\mathsf{GCD}(\mathsf{M}(\mathbf{X})) = 1$, and $\mathbf{X}$ is uniperiodic if and only if $\mathsf{GCD}(\mathsf{M}(\mathbf{X})) = n$. Say that $\mathbf{X}$ is ***anonymous*** if all vectors $\mathbf{x} \in \mathbf{X}$ are anonymous; say that $\mathbf{X}$ is ***eponymous*** if all vectors $\mathbf{x} \in \mathbf{X}$ are eponymous.

We use the standard lexicographical ordering $\preceq$ on Σ^n. We write $\mathbf{x} \prec \mathbf{y}$ to mean $\mathbf{x} \preceq \mathbf{y}$ and $\mathbf{x} \neq \mathbf{y}$. For each $k \in [n+1]$, the ***prefix*** of order k of $\mathbf{x}$, denoted $P_k(\mathbf{x})$, is given by $P_k(\mathbf{x}) = \langle x_0, x_1, \ldots, x_{k-1}\rangle$, with $P_0(\mathbf{x}) = \lambda$. Clearly, for $k < l$, $P_k(\mathbf{x}) \prec P_k(\mathbf{y})$ implies $P_l(\mathbf{x}) \prec P_l(\mathbf{y})$ (and, in particular, $\mathbf{x} \prec \mathbf{y}$). The ***shuffle*** of two vectors $\mathbf{x} = \langle x_0, x_1, \ldots, x_{n-1}\rangle$ and $\mathbf{y} = \langle y_0, y_1, \ldots, y_{n-1}\rangle$, denoted by $\mathbf{x} \parallel \mathbf{y}$, is the vector $\mathbf{x} \parallel \mathbf{y} = \langle (x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1})\rangle$. We observe:

Lemma 3. *For each pair of vectors $\mathbf{x}, \mathbf{y} \in \Sigma^n$, $\mathsf{T}(\mathbf{x} \parallel \mathbf{y}) = \mathsf{LCM}(\mathsf{T}(\mathbf{x}), \mathsf{T}(\mathbf{y}))$.*

A (recursive) *relation* is a subset $\Psi \subseteq \Sigma^n \times \Sigma^n$. For a vector $\mathbf{x} \in \Sigma^n$, $\Psi(\mathbf{x}) = \{\mathbf{y} \mid (\mathbf{x}, \mathbf{y}) \in \Psi\}$; every vector $\mathbf{y} \in \Psi(\mathbf{x})$ is an *image* of $\mathbf{x}$ under Ψ.

The set $\mathsf{Dom}(\Psi)$ of all vectors $\mathbf{x} \in \Sigma^n$ with at least one image under Ψ is the *domain* of Ψ; the set of all images of all vectors $\mathbf{x} \in \Sigma^n$ is the *image* of Ψ, denoted as $\mathsf{Im}(\Psi)$. The relation Ψ is *total* if $\mathsf{Dom}(\Psi) = \Sigma^n$. Given two relations $\Psi_1, \Psi_2 \subseteq \Sigma^n \times \Sigma^n$, their *composition* is the relation $\Psi_1 \circ \Psi_2 = \{(\mathbf{x}, \mathbf{y}) \mid (\mathbf{x}, \mathbf{z}) \in \Psi_2 \text{ and } (\mathbf{z}, \mathbf{y}) \in \Psi_1 \text{ for some } \mathbf{z} \in \Sigma^n\}$. For a relation $\Psi \subseteq \Sigma^n \times \Sigma^n$, note that $\sigma_1 \circ \Psi = \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{y} = \sigma_1(\mathbf{z}) \text{ for some } \mathbf{z} \in \Psi(\mathbf{x})\}$; note also that $\Psi \circ \sigma_1 = \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{y} \in \Psi(\mathbf{z}) \text{ where } \mathbf{z} = \sigma_1(\mathbf{x})\}$. In other words, for a vector $\mathbf{x}$, $\sigma_1 \circ \Psi(\mathbf{x}) = \{\mathbf{y} \mid \mathbf{y} = \sigma_1(\mathbf{z}) \text{ for some } \mathbf{z} \in \Psi(\mathbf{x})\}$ and $\Psi \circ \sigma_1(\mathbf{x}) = \{\mathbf{y} \mid \mathbf{y} \in \Psi(\sigma_1(\mathbf{x}))\}$. Thus, $\sigma_1 \circ \Psi$ maps inputs to shifts of their images, while $\Psi \circ \sigma_1$ maps inputs to images of their shifts. The relation Ψ is ***circularly symmetric*** if $\sigma_1 \circ \Psi \subseteq \Psi \circ \sigma_1$. Intuitively, in a circularly symmetric relation, shifts of images are always images of shifts. A direct induction implies that $\sigma_k \circ \Psi \subseteq \Psi \circ \sigma_k$ for any circularly symmetric relation Ψ and for all integers $k \in \mathbb{N}$.

The relation $\Psi \subseteq \Sigma^n \times \Sigma^n$ is ***aperiodic*** if each vector in $\mathsf{Im}(\Psi)$ is aperiodic; so, each image under Ψ has no circular symmetries. On the other extreme, the relation $\Psi \subseteq \Sigma^n \times \Sigma^n$ is ***uniperiodic*** if each vector in $\mathsf{Im}(\Psi)$ is uniperiodic; so, each image under Ψ is constant. In the middle, the relation $\Psi \subseteq \Sigma^n \times \Sigma^n$ is ***k-periodic*** if each vectors in $\mathsf{Im}(\Psi)$ is k-periodic. Thus, n-periodic and 1-periodic relations are precisely the aperiodic and uniperiodic relations, respectively.

In the ***Leader Election Relation*** $LE \subseteq \Sigma^n \times \Sigma^n$, the set of images of an input vector $\mathbf{x}$ is the set of all vectors with exactly one 1 and $n-1$ 0's; 1 and 0 correspond to "elected" and "non-elected", respectively. Clearly, the Leader Election Relation is both circularly symmetric and aperiodic.

We now discuss a generalization of the Leader Election Relation. Consider a function $\Phi : \Sigma^n \rightarrow \mathbb{Z}^+$. In the ***$\Phi$-Leader Election Relation*** Φ-$LE \subseteq \Sigma^n \times \Sigma^n$, the set of images of an input vector $\mathbf{x}$ is the set of all binary output vectors with the number of 1's ranging from 1 to $\Phi(\mathbf{x})$ (both inclusive). The special case where $\Phi(\mathbf{x}) = 1$ for all vectors $\mathbf{x} \in \Sigma^n$ is precisely the Leader Election Relation. A ***Multiple Leader Election Relation*** is a Φ-Leader Election Relation for some such function Φ.

3 The Partially Eponymous Ring

General. We start with the standard model of an *asynchronous*, *anonymous ring* as studied, for example, in [2, 8]. We assume that the ring is *oriented* and *bidirectional*. We augment this model so that processors have ***identities*** that are neither necessarily all identical, nor necessarily unique. Call it a ***partially eponymous ring***. In the ***anonymous ring*** identities are all identical, while in the ***eponymous ring*** identities are unique.

A ***ring*** R is a cyclic arrangement of $|R|$ identical processors $0, 1, \ldots, |R|-1$. Processor j has an identity id_j and receives an input in_j. The ***identity vector*** is $\mathbf{id} = \langle id_0, id_1, \ldots, id_{|R|-1} \rangle$; the ***input vector*** is $\mathbf{in} = \langle in_0, in_1, \ldots, in_{|R|-1} \rangle$. Note that for the anonymous ring, $\mathsf{T}(\mathbf{id}) = 1$; for the eponymous ring, $\mathsf{T}(\mathbf{id}) = |R|$. The ***(initial) configuration*** of the ring R is the pair $\langle \mathbf{id}, \mathbf{in} \rangle$. Each processor must reach an output out_j by running a local algorithm and communicating with its two neighbors. The ***output vector*** is $\mathbf{out} = \langle out_0, out_1, \ldots, out_{|R|-1} \rangle$.

There is a *single* local *algorithm* A run by all processors; A is represented as a state machine. Each computation step of A at processor j is dependent on the current state of j, the messages currently received at j and on the local identity id_j and input in_j. A ***distributed algorithm*** $\mathcal{A}$ is a collection of local algorithms, one for each processor. We restrict our attention to ***non-uniform*** distributed algorithms, where the size of the ring is "hard-wired" into the single local algorithm. So, we consider rings of a certain size n. The distributed algorithm $\mathcal{A}$ induces a set of (*asynchronous*) *executions*.

Each identity vector $\mathbf{id} \in \Sigma^n$ specifies a single ring; by abuse of notation, denote as $\mathbf{id}$ the specified ring. An equivalence class $\mathbf{ID} \subseteq \Sigma^n$ induces a set of rings, each corresponding to some particular identity vector $\mathbf{id} \in \mathbf{ID}$; by abuse of notation, denote as $\mathbf{ID}$ the induced set.

Solvability and Computability. Consider a configuration $\langle \mathbf{id}, \mathbf{in} \rangle$. Say that the distributed algorithm $\mathcal{A}$ ***solves the set of output vectors*** $\mathcal{OUT}$ ***on the configuration*** $\langle \mathbf{id}, \mathbf{in} \rangle$ if each execution of $\mathcal{A}$ on the ring $\mathbf{id}$ with input $\mathbf{in}$ results to an output vector $\mathbf{out} \in \mathcal{OUT}$. Say that the set of output vectors $\mathcal{OUT}$ is ***solvable on the configuration*** $\langle \mathbf{id}, \mathbf{in} \rangle$ if there is a distributed algorithm $\mathcal{A}$ that solves $\mathcal{OUT}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. Say that the relation Ψ is ***solvable on the set of rings*** $\mathcal{R}$ if there is a distributed algorithm $\mathcal{A}$ such that for each configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$, $\mathcal{A}$ solves $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$.

Say that the distributed algorithm $\mathcal{A}$ ***computes the set of output vectors*** $\mathcal{OUT}$ ***on the configuration*** $\langle \mathbf{id}, \mathbf{in} \rangle$ if the following holds: if $\mathcal{OUT}$ is solvable on the configuration $\langle \mathbf{id}, \mathbf{in} \rangle$, then $\mathcal{A}$ solves $\mathcal{OUT}$ in $\langle \mathbf{id}, \mathbf{in} \rangle$; else $\mathcal{A}$ solves $\{\perp^n\}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$ (an *unsolvability* output). We now develop the notion of a distributed algorithm working for a set of rings and on the entire domain of the relation Ψ; intuitively, the set of rings represents the "knowledge" that the algorithm requires. Formally, the distributed algorithm $\mathcal{A}$ ***computes the relation*** Ψ ***on a set of rings*** $\mathcal{R}$ ***with*** $g(n)$ ***messages*** if for each configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$, $\mathcal{A}$ computes $\Psi(\mathbf{in})$ on the configuration $\langle \mathbf{id}, \mathbf{in} \rangle$. The relation Ψ is ***computable on a set of rings*** $\mathcal{R}$ ***with*** $g(n)$ ***messages*** if there is a distributed algorithm $\mathcal{A}$ that computes Ψ on $\mathbf{ID}$ with $g(n)$ messages. Note that solvability of a relation Ψ on a set of rings $\mathbf{ID}$ implies computability of Ψ on $\mathbf{ID}$ (with some number of messages). However, the inverse does not necessarily hold.

The Least Minimum Base. The ***Minimum Base*** $\mathsf{MB}(\mathbf{id}, \mathbf{in})$ of a configuration $\langle \mathbf{id}, \mathbf{in} \rangle$ is defined by $\mathsf{MB}(\mathbf{id}, \mathbf{in}) = \mathsf{T}(\mathbf{id} \parallel \mathbf{in})$ (cf. [4, 10]). For a set of rings $\mathbf{ID}$ with a common input vector $\mathbf{in}$, the ***Min-Period Minimum Base*** $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$ is defined by $\mathsf{MB}(\mathbf{ID}, \mathbf{in}) = \mathsf{MB}(\widetilde{\mathbf{id}}, \mathbf{in})$, where $\widetilde{\mathbf{id}}$ is the min-period vector of $\mathbf{ID}$. Recall that $\mathsf{T}(\widetilde{\mathbf{id}}) = n/\mathsf{GCD}(\mathsf{M}(\mathbf{ID}))$. Hence, Lemma 3 implies that $\mathsf{MB}(\mathbf{ID}, \mathbf{in}) = \mathsf{LCM}(\mathsf{T}(\widetilde{\mathbf{id}}), \mathsf{T}(\mathbf{in})) = \mathsf{LCM}\,(n/\mathsf{GCD}(\mathsf{M}(\mathbf{ID})), \mathsf{T}(\mathbf{in}))$. By Lemma 2 (condition (2)), $\mathsf{T}(\widetilde{\mathbf{id}})$ divides $\mathsf{T}(\mathbf{id})$ for each ring $\mathbf{id} \in \mathbf{ID}$. Thus, it follows that $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$ divides $\mathsf{LCM}(\mathsf{T}(\mathbf{id}), \mathsf{T}(\mathbf{in}))$ for each ring $\mathbf{id} \in \mathbf{ID}$. Note that if $\mathbf{ID}$ is the set of anonymous rings, then $\mathsf{MB}(\mathbf{ID}, \mathbf{in}) = \mathsf{LCM}(1, \mathsf{T}(\mathbf{in})) = \mathsf{T}(\mathbf{in})$; if $\mathbf{ID}$

is the set of eponymous rings, then $\mathsf{MB}(\mathbf{ID}, \mathbf{in}) = \mathsf{LCM}(\mathsf{T}(\widetilde{\mathbf{id}}), \mathsf{T}(\mathbf{in})) \geq \mathsf{T}(\mathbf{in})$. So, intuitively, the Minimum Base is an indicator of computability. The ***Least Minimum Base*** $\mathsf{LMB}(\mathbf{ID}, \Psi)$ of a set of rings $\mathbf{ID}$ and a relation Ψ is defined by $\mathsf{LMB}(\mathbf{ID}, \Psi) = \min\{\mathsf{MB}(\mathbf{id}, \mathbf{in}) \mid \langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)\}$.

Views. We conclude with a definition that extends one in [15, Section 3] to a ring where processors receive inputs. Given a ring **id** with input vector **in**, the ***view*** of processor j is $view_j(\mathbf{id}, \mathbf{in}) = \sigma_j(\mathbf{id} \parallel \mathbf{in}) = \sigma_j(\mathbf{id}) \parallel \sigma_j(\mathbf{in})$. Clearly, views of processors are cyclic shifts of each other. There is a direct, non-uniform distributed algorithm $\mathcal{A}_{\mathrm{CV}}$ with message complexity $\Theta(|R|^2)$ that allows each processor to construct its own view on a ring R. It is simple to prove:

Lemma 4. *Consider a ring* **id** *with input* **in** *and two processors j and k with* $view_j(\mathbf{id}, \mathbf{in}) = view_k(\mathbf{id}, \mathbf{in})$. *Then, in a synchronous execution of a distributed algorithm with output vector* **out**, $out_j = out_k$.

4 Solvability and Computability

Preliminaries. We provide a definition of compatibility between a set of output vectors $\mathcal{OUT}$ and a configuration $\langle \mathbf{id}, \mathbf{in} \rangle$. The set of output vectors $\mathcal{OUT}$ is ***compatible*** with the configuration $\langle \mathbf{id}, \mathbf{in} \rangle$ if there is an output vector $\mathbf{out} \in \mathcal{OUT}$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$. We prove:

Lemma 5. *Assume that a set of output vectors $\mathcal{OUT}$ is solvable on a configuration $\langle \mathbf{id}, \mathbf{in} \rangle$. Then, $\mathcal{OUT}$ is compatible with $\langle \mathbf{id}, \mathbf{in} \rangle$.*

Proof (sketch). By assumption, there is a distributed algorithm $\mathcal{A}$ that solves $\mathcal{OUT}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. Fix a synchronous execution of $\mathcal{A}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$, and consider the associated vector $\mathbf{out} \in \mathcal{OUT}$. Recall that $view_j(\mathbf{id}, \mathbf{in}) = \sigma_j(\mathbf{id} \parallel \mathbf{in})$, so that $view_{j+\mathsf{T}(\mathbf{id}\|\mathbf{in})}(\mathbf{id}, \mathbf{in}) = \sigma_{j+\mathsf{T}(\mathbf{id}\|\mathbf{in})}(\mathbf{id} \parallel \mathbf{in})$. By Lemma 1, it holds that $\sigma_j(\mathbf{id} \parallel \mathbf{in}) = \sigma_{j+\mathsf{T}(\mathbf{id}\|\mathbf{in})}(\mathbf{id} \parallel \mathbf{in})$. So $view_j(\mathbf{id}, \mathbf{in}) = view_{j+\mathsf{T}(\mathbf{id}\|\mathbf{in})}(\mathbf{id}, \mathbf{in})$, and Lemma 4 implies that $out_j = out_{j+\mathsf{T}(\mathbf{id}\|\mathbf{in})}$. Since j was chosen arbitrarily, this implies that $\sigma_{\mathsf{T}(\mathbf{id}\|\mathbf{in})}(\mathbf{out}) = \mathbf{out}$. By definition of period, Lemma 1 implies that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{T}(\mathbf{id} \parallel \mathbf{in}) = \mathsf{MB}(\mathbf{id}, \mathbf{in})$. $\square$

We now introduce the distributed (non-uniform) algorithm $\mathcal{A}_\Psi$ associated with an arbitrary circularly symmetric relation $\Psi \in \Sigma^n \times \Sigma^n$; the algorithm is described in Figure 1. Note that the distributed algorithm $\mathcal{A}_\Psi$ does *not* specify how the views are constructed in *Step* 2. The views can be constructed by invoking, for example, the distributed algorithm $\mathcal{A}_{\mathrm{CV}}$ which collects the identities and inputs of processors using n^2 messages. All remaining steps are local. *Steps* 3–6 enable processors to choose a common output vector, while *Step* 6 enables processor j to output its individual coordinate in this vector. The set *Choices* contains all candidates for the common output vector; in *Step* 6, processors use a common function (e.g., min) to single out one of the candidates. We prove:

Lemma 6. *For a configuration $\langle \mathbf{id}, \mathbf{in} \rangle$, either $\mathcal{A}_\Psi$ solves $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$, or $\mathcal{A}_\Psi$ solves $\{\perp^n\}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$.*

$\mathcal{A}_\Psi$: CODE FOR PROCESSOR j WITH IDENTITY id_j AND INPUT in_j

1: Upon receiving message $\langle wake \rangle$ **do**
2: Construct $view_j(\mathbf{id}, \mathbf{in}) = vid_j \,\|\, vin_j$. $\backslash * \; vid_j = \sigma_j(\mathbf{id}),\; vin_j = \sigma_j(\mathbf{in}) \; * \backslash$
3: $Views_j[i] := \langle \sigma_i(vid_j), \sigma_i(vin_j) \rangle$ for each $i \in [|R|]$.
4: $Choices := \{(\mathbf{x}, \mathbf{y}, \mathbf{z}) \mid \langle \mathbf{x}, \mathbf{y} \rangle \in Views_j;\; \mathbf{z} \in \Psi(\mathbf{y});\; \mathsf{T}(\mathbf{z}) \text{ divides } \mathsf{MB}(\mathbf{x}, \mathbf{y})\}$.
5: **If** $Choices = \emptyset$ **then** $out_j := \bot$ and terminate.
6: (a) $(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}) := \mathsf{min}(Choices)$; (b) $k_j := \mathsf{min}\{i \in [|R|] \mid Views_j[i] = \langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle\}$.
7: Set out_j to be the first entry of $\sigma_{-k_j}(\underline{\mathbf{z}})$ and terminate.

Figure 1. Algorithm $\mathcal{A}_\Psi$: code for processor j.

The proof of Lemma 6 requires Ψ to be circularly symmetric. It is not evident whether this assumption is *essential* for the partially eponymous ring, although it is known to be so for computing functions on the anonymous ring [2].

Main Results. Our first result concerns the solvability of an arbitrary (circularly symmetric) relation on an arbitrary set of rings $\mathbf{ID}$. We provide a definition of compatibility between a relation $\Psi \subseteq \Sigma^n \times \Sigma^n$ and a set of rings $\mathbf{ID} \subseteq \Sigma^n$. The relation Ψ is ***compatible*** with the set of rings $\mathbf{ID}$ if for each input vector $\mathbf{in} \in \mathsf{Dom}(\Psi)$, there is some output vector $\mathbf{out} \in \Psi(\mathbf{in})$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$. Recall that $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$ can be computed efficiently; thus, compatibility can be checked efficiently. We prove:

Theorem 1 (Partially Eponymous Solvability Theorem). *A circularly symmetric relation Ψ is solvable on a set of rings $\mathbf{ID}$ if and only if Ψ is compatible with $\mathbf{ID}$.*

Proof (sketch). Assume that Ψ is solvable on $\mathbf{ID}$. By definition of solvability, there is a distributed algorithm $\mathcal{A}$ such that for each configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$, $\mathcal{A}$ solves the set of vectors $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. So, fix the configuration $\langle \widetilde{\mathbf{id}}, \mathbf{in} \rangle$ for an arbitrary vector $\mathbf{in} \in \mathsf{Dom}(\Psi)$. If follows that the set of vectors $\Psi(\mathbf{in})$ is solvable on $\langle \widetilde{\mathbf{id}}, \mathbf{in} \rangle$. By Lemma 5, this implies that $\Psi(\mathbf{in})$ is compatible with $\langle \widetilde{\mathbf{id}}, \mathbf{in} \rangle$. By definition of compatibility between a set of output vectors and a configuration, it follows that there is some output vector $\mathbf{out} \in \Psi(\mathbf{in})$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\widetilde{\mathbf{id}}, \mathbf{in}) = \mathsf{MB}(\mathbf{ID}, \mathbf{in})$ (by definition of Min-Period Minimum Base). By definition of compatibility, the claim follows.

Assume now that Ψ is compatible with $\mathbf{ID}$. By definition of compatibility, for each input vector $\mathbf{in} \in \mathsf{Dom}(\Psi)$, there is an output vector $\mathbf{out} \in \Psi(\mathbf{in})$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$. Fix an arbitrary configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$. Recall that $\mathsf{MB}(\mathbf{ID}, \mathbf{in})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$. It follows that there is some output vector $\mathbf{out} \in \Psi(\mathbf{in})$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$. Recall the distributed algorithm $\mathcal{A}_\Psi$. Clearly, $\langle \mathbf{id}, \mathbf{in} \rangle$ is an entry of $Views_j$ for each processor $j \in [n]$. Since $\mathbf{out} \in \Psi(\mathbf{in})$ and $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$, it follows that $(\mathbf{id}, \mathbf{in}, \mathbf{out}) \in Choices$; so $Choices \neq \emptyset$. By the algorithm $\mathcal{A}_\Psi$ (*Step* 5), it follows that the algorithm does not solve $\{\bot^n\}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. Hence, Lemma 6 implies that $\mathcal{A}_{\Psi}$ solves $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. Since the configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$ was chosen arbitrarily, it follows that Ψ is solvable on $\mathbf{ID}$. $\square$

Since compatibility is efficiently checkable, Theorem 1 provides an efficient characterization of solvability for the partially eponymous ring. We now prove:

Theorem 2 (Partially Eponymous Computability Theorem). *Algorithm $\mathcal{A}_{\Psi}$ computes the circularly symmetric relation Ψ on a set of rings* $\mathbf{ID}$.

Proof (sketch). Fix an arbitrary configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(\Psi)$. We will prove that $\mathcal{A}_{\Psi}$ computes $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. We proceed by case analysis: Assume first that $\Psi(\mathbf{in})$ is solvable on $\langle \mathbf{id}, \mathbf{in} \rangle$. By Lemma 5 it follows that $\Psi(\mathbf{in})$ is compatible with $\langle \mathbf{id}, \mathbf{in} \rangle$. By definition of compatibility of a set of output vectors with a configuration, this implies that there is some output vector $\mathbf{out} \in \Psi(\mathbf{in})$ such that $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$. From the distributed algorithm $\mathcal{A}_{\Psi}$, $\langle \mathbf{id}, \mathbf{in} \rangle$ is an entry of $Views_j$ for each processor $j \in [n]$. Since, $\mathbf{out} \in \Psi(\mathbf{in})$ and $\mathsf{T}(\mathbf{out})$ divides $\mathsf{MB}(\mathbf{id}, \mathbf{in})$, it follows that $(\mathbf{id}, \mathbf{in}, \mathbf{out}) \in Choices$; so $Choices \neq \emptyset$. By the algorithm $\mathcal{A}_{\Psi}$ (*Step* 5), it follows that $\mathcal{A}_{\Psi}$ does not solve $\{\perp^n\}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. Hence, Lemma 6 implies that $\mathcal{A}_{\Psi}$ solves $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$, as needed. Assume now that $\Psi(\mathbf{in})$ is not solvable on $\langle \mathbf{id}, \mathbf{in} \rangle$. This implies that $\mathcal{A}_{\Psi}$ does not solve $\Psi(\mathbf{in})$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. By Lemma 6, $\mathcal{A}_{\Psi}$ solves $\{\perp^n\}$ on $\langle \mathbf{id}, \mathbf{in} \rangle$. $\square$

Applications. For uniperiodic relations, Theorem 1 immediately implies that every circularly symmetric, uniperiodic relation is solvable on any set if rings $\mathbf{ID}$. As a natural application of Theorem 1 on aperiodic relations, we prove:

Theorem 3 (Solvability of Aperiodic Relations). *A total, circularly symmetric, aperiodic relation Ψ is solvable on a set of rings* $\mathbf{ID}$ *if and only if* $\mathbf{ID}$ *is aperiodic.*

Actually, Theorem 3 applies more generally to a *non-total*, circularly symmetric, aperiodic relation Ψ, as long as there is at least one constant vector in $\mathsf{Dom}(\Psi)$. Many relations from the literature assume no inputs, so that their domain consists of a single, constant input vector; Theorem 3 applies to all such relations. Finally, note that Theorem 3 can be further generalized to prove that a circularly symmetric, k-periodic relation Ψ, with at least one constant vector in $\mathsf{Dom}(\Psi)$, is solvable on a set of rings $\mathbf{ID}$ if and only if $\mathbf{ID}$ is l-periodic, and k divides l.

5 Message Complexity

Multiple Leader Election as a Tool. We present an asynchronous distributed algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ with advice k. Here, k is an integer that is available to each processor (e.g., it is "hard-wired" into its local algorithm much in the same way the ring size is in a non-uniform distributed algorithm). k will act as a parameter to $\mathcal{A}_{\mathrm{MLE}}(k)$: $\mathcal{A}_{\mathrm{MLE}}(k)$ will satisfy a particular correctness property for certain specific advices k; furthermore, the message complexity of $\mathcal{A}_{\mathrm{MLE}}(k)$ will depend on k. The algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ is similar in spirit to the well-known (*neighborhood doubling*) asynchronous distributed algorithm of Hirschberg and Sinclair [9] that computes Leader Election on the eponymous ring R. (Recall that the algorithm of Hirschberg and Sinclair uses $O(|R| \cdot \lg |R|)$ messages.) So, each processor explores neighborhoods around it whose size doubles in each *phase*; in phase r, the processor collects identities of other processors in the neighborhood that are 2^r to the left (counter-clockwise) of it, or $2^{r+1} - 1$ to the right (clockwise) of it. It then uses these identities to locally compute the prefixes of length 2^r of the views of all processors that are 2^r to the left or to the right of it. Then, the processor compares the prefix of length 2^r of its own view to those prefixes it has computed; it survives the phase (so that it can proceed to the next phase) if and only if its own prefix is *the lexicographically least* among all 2^{r+1} prefixes of the views of its neighbors that it has computed. Note that by *Step* 21, $r \leq \lceil \lg k \rceil - 1$; thus, k determines the size of the largest neighborhood that each processor will explore before terminating. Also, note that the major difference between our algorithm for the partially eponymous ring and the classical algorithm of Hirschberg and Sinclair to compute Leader Election on the eponymous ring is that our algorithm awards processors to proceed to the next phase on the basis of the *computed prefixes* of processors' views (which change across phases), as opposed to processors' identities (that remain constant across phases). Note that the lexicographic ordering provides the property that views and their corresponding prefixes are consistently ordered. Hence, the comparison of prefixes in *Step* 20 is essentially a comparison of views (hence, an efficient one since it avoids the full construction of views). This is an essential feature of our algorithm, and its achieved message efficiency is due to this feature. The distributed algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ appears in pseudocode in Figure 2. We proceed to prove certain message complexity and correctness properties of $\mathcal{A}_{\mathrm{MLE}}(k)$.

Using an analysis similar to the analysis of the message complexity for the algorithm of Hirschberg and Sinclair [9], we prove an upper bound on the message complexity of the distributed algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$. Since k determines the size of the largest neighborhood that each processor will explore before terminating, the message complexity of the distributed algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ increases with k.

Proposition 1. *Algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ uses $O(|R| \cdot \lg k)$ messages on the ring R.*

We continue with a correctness property of the algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ for specific advices k. For any k that divides n consider the function $\Phi_k : \Sigma^n \rightarrow \mathbb{Z}^+$ such that $\Phi_k(\mathbf{in}) = 2n/k$ for each input vector $\mathbf{in} \in \Sigma^n$. We continue to prove:

Proposition 2. *Algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ with advice $k, 1 \leq k \leq \mathsf{MB}(\mathbf{id}, \mathbf{in})$ solves the set of output vectors Φ_k-$LE(\mathbf{in})$ on the configuration $\langle \mathbf{id}, \mathbf{in} \rangle$.*

Intuitively, we wish to elect as few leaders as possible, since each leader will be subsequently asked to undertake an additional (message intensive) distributed computation. Proposition 2 establishes that the larger the advice k is, the less leaders are elected; on the other hand, k cannot be chosen to be arbitrarily large.

$\mathcal{A}_{\mathrm{MLE}}(k)$: CODE FOR PROCESSOR j WITH IDENTITY id_j AND INPUT in_j

Initially, $label_j = segment_j = left\_segment_j = right\_segment_j = \lambda$.

Upon receiving message $\langle wake \rangle$ **do**
  Send message $\langle probe, 0, 1 \rangle$ to left.

Upon receiving message $\langle probe, r, d \rangle$ from right **do**
  If $d < 2^r$ **then** send message $\langle probe, r, d+1 \rangle$ to left.
  If $d = 2^r$ **then** send message $\langle reply, \langle (id_j, in_j) \rangle, r, 1 \rangle$ to right.

Upon receiving message $\langle reply, s, r, d \rangle$ from left **do**
  If $d < 2^r$ **then** send message $\langle reply, s \diamond \langle (id_j, in_j) \rangle, r, d+1 \rangle$ to right.
  If $d = 2^r$ **then do**
    $left\_segment_j := s$.
    Send message $\langle probe, r, 1 \rangle$ to right.

Upon receiving message $\langle probe, r, d \rangle$ from left **do**
  If $d < 2^{r+1} - 1$ **then** send message $\langle probe, r, d+1 \rangle$ to right.
  If $d = 2^{r+1} - 1$ **then** send message $\langle reply, \langle (id_j, in_j) \rangle, r, 1 \rangle$ to left.

Upon receiving message $\langle reply, s, r, d \rangle$ from right **do**
  If $d < 2^{r+1} - 1$ **then** send message $\langle reply, \langle (id_j, in_j) \rangle \diamond s, r, d+1 \rangle$ to left.
  If $d = 2^{r+1} - 1$ **then do**
    $right\_segment_j := s$.
    $segment_j := left\_segment_j \diamond \langle (id_j, in_j) \rangle \diamond right\_segment_j$.
    $label_j := P_{2^r}(\sigma_{2^r}(segment_j))$.
    If $label_j \prec P_{2^r}(\sigma_i(segment_j))$ and $label_j \preceq P_{2^r}(\sigma_{2^r+i+1}(segment_j))$, for all $i \in [2^r]$ **then do**
      If $2^{r+1} \geq k$ **then** terminate as a leader.
      Else send message $\langle probe, r+1, 1 \rangle$ to left.
    Else terminate as a non-leader.

Figure 2. Algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$: code for processor j.

A Universal Upper Bound on Message Complexity. Recall that we have established correctness guarantees for the algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ only when $k \leq \mathsf{MB}(\mathbf{id}, \mathbf{in})$ on configuration $\langle \mathbf{id}, \mathbf{in} \rangle$. Recall also that we wish to maximize k, so as to minimize the number of elected leaders. Furthermore, if algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ is to be used for electing a number of leaders, the value of k must be "known" to the processors. This raises the natural question of what such an appropriate value for k is. In what follows, we choose $k = \mathsf{LMB}(\mathbf{ID}, \Psi)$, the least upper bound imposed on k by Proposition 2, across all configurations $\langle \mathbf{id}, \mathbf{in} \rangle$. We next ask how this advice k relates to the message complexity of computing an arbitrary (circularly symmetric) relation $\Psi \subseteq \Sigma^n \times \Sigma^n$ on an arbitrary set of rings $\mathbf{ID} \subseteq \Sigma^n$; recall that Ψ is computable on $\mathbf{ID}$ with $O(n^2)$ messages (Theorem 2). We prove:

Theorem 4 (Partially Eponymous Message Complexity Theorem). *The circularly symmetric relation Ψ is computable on a set of rings $\mathbf{ID}$ with $O\left((n^2/\mathsf{LMB}(\mathbf{ID}, \Psi)) + n \cdot \lg \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$ messages.*

Proof (sketch). Here is a distributed algorithm $\mathcal{A}$ (which is an instantiation of $\mathcal{A}_{\Psi}$) to compute Ψ on $\mathbf{ID}$ with that many messages. Consider an arbitrary configuration $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \Sigma^n$. $\mathcal{A}$ proceeds as follows:

• On top, the distributed algorithm $\mathcal{A}_{\Psi}$ (see Figure 1) is invoked to compute Ψ on $\mathbf{ID}$. *Step* 2 is implemented by the following steps:
- First, the processors run the distributed algorithm $\mathcal{A}_{\mathrm{MLE}}(k)$ with advice $k = \mathsf{LMB}(\mathbf{ID}, \Psi) \leq \mathsf{MB}(\mathbf{id}, \mathbf{in})$ using $O\left(n \cdot \lg \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$ messages (by Proposition 1); by Proposition 2, there are now elected at least 1 and at most $O\left(n / \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$ leaders.
- All elected leaders run the algorithm $\mathcal{A}_{\mathrm{CV}}$ (see Section 4) to collect their views, for a total of $O\left((n/\mathsf{LMB}(\mathbf{ID}, \Psi)) \cdot n\right) = O\left(n^2 / \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$ messages. Then, the leaders communicate the collected views to all processors for a total of $O\left((n/\mathsf{LMB}(\mathbf{ID}, \Psi)) \cdot n\right) = O\left(n^2 / \mathsf{LMB}(\mathbf{ID}, \Psi)\right)$ messages. Now, each processor locally derives its view, which it returns to top.

So, the message complexity of $\mathcal{A}$ is as claimed. □

Applications. For which sets of rings $\mathbf{ID}$ is the upper bound on message complexity from Theorem 4 low? We identify two such classes of sets.

Say that a set of rings $\mathbf{ID} \subseteq \Sigma^n$ is ***universal*** if Leader Election is solvable on $\mathbf{ID}$. Clearly, *every* (circularly symmetric) relation is solvable on a universal set of rings. Recall that Leader Election is both circularly symmetric and aperiodic. Hence, Theorem 3 implies that $\mathbf{ID}$ is aperiodic. Thus, for each $\langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(LE)$, $\mathsf{MB}(\mathbf{id}, \mathbf{in}) = n$. Hence, $\mathsf{LMB}(\mathbf{ID}, LE) = \min\{\mathsf{MB}(\mathbf{id}, \mathbf{in}) \mid \langle \mathbf{id}, \mathbf{in} \rangle \in \mathbf{ID} \times \mathsf{Dom}(LE)\} = n$. Theorem 4 now immediately implies:

Theorem 5 (Message Complexity on Universal Set of Rings). *A circularly symmetric relation is computable on a universal set of rings* $\mathbf{ID}$ *with* $O(n \cdot \lg n)$ *messages.*

Consider an arbitrary circularly symmetric relation Ψ. Consider a ring $\mathbf{id} \in \Sigma^n$ where each identity has multiplicity at most $\lg n$; call it a ***logarithmic*** ring. The corresponding set of rings $\mathbf{ID}$ will be called a ***logarithmic*** set of rings. Lemma 2 (condition (1)) implies that $\mathsf{T}(\widetilde{\mathbf{id}}) = n / \mathsf{GCD}(\mathsf{M}(\mathbf{ID})) \geq n / \lg n$. So, for each input vector $\mathbf{in} \in \mathsf{Dom}(\Psi)$, $\mathsf{MB}(\widetilde{\mathbf{id}}, \mathbf{in}) = \mathsf{LCM}(\mathsf{T}(\widetilde{\mathbf{id}}), \mathsf{T}(\mathbf{in})) \geq n / \lg n$. This implies that $\mathsf{LMB}(\mathbf{ID}, \Psi) \geq n / \lg n$. Since also $\mathsf{LMB}(\mathbf{ID}, \Psi) \leq n$, Theorem 4 immediately implies:

Theorem 6 (Message Complexity on Logarithmic Set of Rings). *A circularly symmetric relation is computable on a logarithmic set of rings* $\mathbf{ID}$ *with* $O(n \cdot \lg n)$ *messages.*

We can obtain additional upper bounds on the message complexity of computing a circularly symmetric relation by further generalizing Theorem 6. Towards this end, we generalize the definition of a logarithmic set of rings to a set of rings with an upper bound m on the multiplicity of each identity in any ring from the set. The corresponding upper bound on message complexity is $O(n \cdot \max\{m, \lg n\})$.

6 Epilogue

We presented a comprehensive study of solvability, computability and message complexity for the partially eponymous ring. Several interesting questions remain. For example, is there a *matching* lower bound to the universal upper bound on message complexity from Theorem 4? Can we characterize the class of sets of rings of size n for which this (universal) upper bound becomes $O(n \cdot \lg n)$? Finally, a challenging task is to extend our theory for the partially eponymous ring to other network architectures (such as *hypercubes* and *tori*).

References

1. D. Angluin, "Local and Global Properties in Networks of Processors," *Proceedings of the 12th Annual ACM Symposium on Theory of Computing,* pp. 82–93, 1980.
2. H. Attiya, M. Snir and M. Warmuth, "Computing on an Anonymous Ring," *Journal of the ACM,* Vol. 35, No. 4, pp. 845–875, 1988.
3. P. Boldi and S. Vigna, "An Effective Characterization of Computability in Anonymous Networks," *Proceedings of the 15th International Symposium on Distributed Computing,* Vol. 2180, pp. 33–47, LNCS, Springer-Verlag, 2001.
4. P. Boldi and S. Vigna, "Fibrations of Graphs," *Discrete Mathematics,* Vol. 243, pp. 21–66, 2002.
5. J. E. Burns, "A Formal Model for Message Passing Systems," Technical Report TR-91, Department of Computer Science, Indiana University, 1980.
6. J. Chalopin, S. Das and N. Santoro, "Groupings and Pairings in Anonymous Networks," *Proceedings of the 20th International Symposium on Distributed Computing,* Vol. 4167, pp. 105–119, LNCS, Springer-Verlag, 2006.
7. S. Dobrev and A. Pelc, "Leader Election in Rings with Nonunique Labels," *Fundamenta Informaticae,* Vol. 59, No. 4, pp. 333–347, 2004.
8. P. Flocchini, E. Kranakis, D. Krizanc, F. L. Luccio and N. Santoro, "Sorting and Election in Anonymous Asynchronous Rings," *Journal of Parallel and Distributed Computing,* Vol. 64, No. 2, pp. 254–265, 2004.
9. D. Hirschberg and J. B. Sinclair, "Decentralized Extrema-Finding in Circular Configurations of Processes," *Communications of the ACM,* Vol. 23, No. 11, pp. 627–628, 1980.
10. F. T. Leighton, "Finite Common Coverings of Graphs," *Journal of Combinatorial Theory (Series B),* Vol. 33, pp. 231–238, 1982.
11. G. LeLann, "Distributed Systems - Towards a Formal Approach," *Information Processing 77,* Proceedings of the IFIP Congress, pp. 155–160, 1977.
12. M. Yamashita and T. Kameda, "Computing on Anonymous Networks, Part I: Characterizing the Solvable Cases," *IEEE Transactions on Parallel and Distributed Systems,* Vol. 7, No. 1, pp. 69–89, 1996.
13. M. Yamashita and T. Kameda, "Computing on Anonymous Networks, Part II: Decision and Membership Problems," *IEEE Transactions on Parallel and Distributed Systems,* Vol. 7, No. 1, pp. 90–96, 1996.
14. M. Yamashita and T. Kameda, "Computing Functions on Asynchronous Anonymous Networks," *Mathematical Systems Theory,* Vol. 29, No. 4, pp. 331–356, 1998. Erratum in *Theory of Computing Systems,* Vol. 31, No. 1, pp. 109, 1998.
15. M. Yamashita and T. Kameda, "Leader Election Problem on Networks in Which Processor Identity Numbers are not Distinct," *IEEE Transactions on Parallel and Distributed Systems,* Vol. 10, No. 9, pp. 878–887, 1999.