Sense of Direction: Definitions, Properties and Classes *

Paola Flocchini[†] Bernard Mans[‡] Nicola Santoro[§]

Abstract

An extensive body of evidence exists of the impact that specific edge labelings have on the communication complexity of distributed problems. It has been long suspected that these very different labelings share a common property, named *Sense of Direction*. In spite of the large amount of investigations, and of the obvious practical importance, a formal characterization of this property did not exist.

In this paper, we finally provide a formal definition of sense of direction, making explicit the very specific relationship between three factors: the labeling, the topological structure, and the local view that an entity has of the system. In a way, sense of direction is the capability of a node in the system to use the labeling to translate the local view of its neighbors into its own. Using the formal definition as an observational platform, we describe several properties which allow the translation process to be possible beyond the immediate neighborhood. Finally, we identify four general classes of labelings and analyze their properties; these classes include all the labelings used in the literature.

*A preliminary version of this paper appeared in the proceedings of the *1st Colloquium on Structural Information and Communication Complexity*, Ottawa, May 6-8, 1994. This work has been supported in part by N.S.E.R.C under grant #A2415, by a Macquarie University grant, and by a N.A.T.O. Advanced Research Fellowship.

[†]Dept. d'Informatique et de Recherche Opérationelle, Université de Montréal, Montréal, Canada H3C 3J7. (flocchini@iro.umontreal.ca)

[‡]Department of Computing, School of MPCE, Macquarie University, Sydney, NSW 2109, Australia. (bmans@mpce.mq.edu.au)

[§]Center for Parallel and Distributed Computing, School of Computer Science, Carleton University, Ottawa K1S 5B6, Canada. (santoro@scs.carleton.ca)

1 Introduction

A distributed system is a collection of autonomous entities (e.g., processors) connected by a communication network, where each entity has a local non-shared memory and can communicate by sending messages to and receiving messages from its neighbors. The entire system can be viewed as a graph where each node corresponds to a system entity, and each edge corresponds to a direct communication link between two entities. Every entity has a distinct label (e.g., port number) associated to each of its incident links; every edge has, thus, two labels, one for each of its incident nodes. A classical example is a ring network where each edge is labeled "right" at an incident node and "left" at the other.

A very important fact is that some assignments of labels (or, labelings) have a dramatic effect on the communication complexity of distributed problems.

This fact has been made explicit by the surprising result of [29]: the "distance" labeling of a complete graph allows the message complexity of the election process to be reduced from $\Omega(n \log n)$ (for the unlabeled case [22]) to $O(n)$, where n is the number of entities in the system (see also [34, 46]). Since this first result, the evidence of the impact of specific labelings in particular graphs has been accumulating in the recent years. For example, in particular chordal rings without labeling, the election process requires $\Omega(n \log n)$ messages; with a "distance" labeling there exist algorithms whose complexity depends on the chord structure and can be linear [2, 21, 33, 38, 39, 50]. Similarly, $O(n)$ election algorithms exist for an hypercube with the traditional "dimensional" labeling [12, 41, 49, 51]; without labeling, the best known complexity is $O(n \log \log n)$ [10]. An even simpler $O(n)$ technique has been found if the hypercube has a particular "distance" labeling [12]. In systems of unknown topology (the so-called *arbitrary graph* case), the availability of the "neighboring" labeling (or, equivalently, knowledge of the identifiers of the neighbours) reduces the complexity of the election problem from $\Omega(e + n \log n)$ (for the unlabeled case [17, 42]) to $O(n \log n)$ messages (where e is the number of communication links). With the same labeling, the message complexity of depth first traversal drops from $\Omega(e)$ to $O(n)$ [45]. The same reductions for both the election and depth first traversal problems can be obtained also with the simpler "distance" labeling [32].

The properties of some of these labelings have been intensely studied and applied. For example, the "distance" labeling in chordal rings and complete graphs has been used for the weak unison problem [20] and for fault-tolerant election [31, 36]. The "dimensional" labeling in hypercubes has been investigated for its impact on computability when the system is anonymous and possibly faulty [24, 26]. The complexity of constructing the traditional "left-right" labeling of a ring has been studied [3, 19, 47], and lower bounds for the election problem in presence of such a labeling have been established [6]. The construction of the traditional labelings of the well-known topologies (hypercubes, tori, etc.) has been the object of extensive study [48].

Incomplete labelings of specific topologies have also been the object of investigations (e.g., [2, 18, 44]). It has been shown that even in this case there is an impact on complexity. For example, it is possible to elect a leader in a complete graph with $O(n)$ messages [29] (instead of $\Omega(n \log n)$) even if each node has the "distance" labeling on only $O(1)$ appropriate incident arcs [2]. In unlabeled torus and chordal rings (with one chord of length approximately $\sqrt{N}$) linear communication cost can be achieved [30, 40].

All these labelings differ greatly from each other: the "distance" labeling in chordal rings, the "dimensional" labeling in hypercubes, the "neighboring" labeling in arbitrary graphs, etc. Still, the way they impact on the complexity (i.e., the manner in which the solution algorithms exploit the labelings) is similar, hinting the presence of a *common* extremely useful property. This property has been named *Sense of Direction* [43], and is generally described as the presence of some "global consistency" of the labeling.

The knowledge acquired by most investigations concentrates on specific problems in particular topologies with particular labeling: it provides information on *instances* of sense of direction. Other information on sense of direction is given implicitly by the related investigations on the impact of network structure in anonymous systems [3, 4, 15, 25, 37, 52], on the difference between labeled and unlabeled anonymous systems [23], on the relationship between graph symmetry and labelings [14, 52], and, to some extent, by the investigations on *Implicit Routing* (see [28] for a survey).

Unfortunately, in spite of the large amount of investigations, of the extensive body of knowledge, and of the evident practical importance, a formal definition of sense of direction did not exist. Actually, even an understanding of what "global consistency" is and why it works in a context larger than the single instances has been missing.

In this paper, we provide a formal definition of sense of direction. In particular, we define those properties which make it possible to reduce the communication complexity, a task which was not exploited in the previous investigations. This is achieved by identifying the mechanisms which operate in the reduction, and determining the conditions for the existence of those mechanisms.

From the definition, it emerges that in sense of direction there is a very specific relationship between three factors: the *labeling*, the *topological structure*, and the *local view* that an entity has of the system. In a way, sense of direction is the capability of a node in the system to use the labeling to *translate* the local view of its neighbors into its own.

Using the formal definition as an observational platform, we derive several previously unknown properties of sense of direction as well as properties implied by having sense of direction in a system.

Based on the formal definition, four general classes of labelings are identified and defined. These classes include all the labelings used in the field. We have shown in [13] that all the existing results for general graphs follow as simple applications of the definition or of the derived properties.

The paper is organized as follows. In the next Section we give an informal description of sense of direction. In Section 3 we introduce the notion of local edge- and node-labelings and, on the basis of these notions, we formally define sense of direction.

In Section 4, we discuss properties of sense of direction which allow a node to derive information about the local view of other nodes in the system. In Section 5, we introduce several instances of sense of direction, we group them in four general classes, and we analyze their properties. Finally, in Section 6, we discuss some open problems.

2 An Informal Description

In this Section, we will provide an intuitive description of the three fundamental properties which characterize the notion of sense of direction. These properties are (sometimes obscurely) implicit in the previous (topology-dependent) results.

First of all, in sense of direction, there is a relationship between labeling and capability of distinguishing among walks. Each node x has a unique label associated to each of its incident edges; let $\lambda_x(\langle x, z\rangle)$ be the label associated by x to the edge $\langle x, z\rangle$.

Intuitively, when the labeling is a sense of direction it is possible to understand, from the labels associated to the edges, whether different walks from any given node x end in the same node or in different nodes.

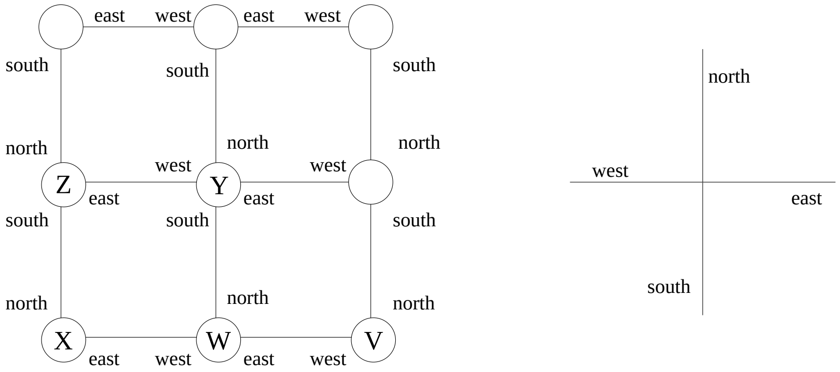

Figure 1: Labeling which is a $\mathcal{SD}$.

For example, consider the system depicted in Figure 1: the communication topology is a 2-dimensional mesh where the edge labels are from the set $\{north, south, east, west\}$ and are assigned in the natural globally consistent way. This labeling is a sense of direction (for an appropriate choice of the node names, as discussed later). Consider for instance the three walks, starting from X, whose associated labels are $c_1 = [north, north, east, south]$, $c_2 = [\ east, east, north, west]$, and $c_3 = [east, east]$. Using the rules of the globally consistent labeling, it is trivial to deduce that the two walks corresponding to c_1 and c_2 will end in the same node Y, while the one corresponding to c_3 will end in a different node V.

In other words, when there is sense of direction, it must be possible to distinguish for each pair of nodes, x and y, in the set of walks starting from x, the ones which terminate in y.

The second property, which is a consequence of the first, is the existence of a relationship between edge labelings and local node names. Each node x refers to the other nodes using local names. Let us stress that these local names are *not* necessarily identities (i.e., unique global identifiers): the system could be anonymous. The set of local names given by x will be called the *local view* of x. Let $\beta_x(y)$ be the local name under which y is known at x. Obviously, this name is the one which will be used whenever x wants to refer to y.

Intuitively, in a labeled graph with sense of direction, there is a function which maps the sequences of labels associated to the walks from x to y to the local name $\beta_x(y)$ used by x to refer to y.

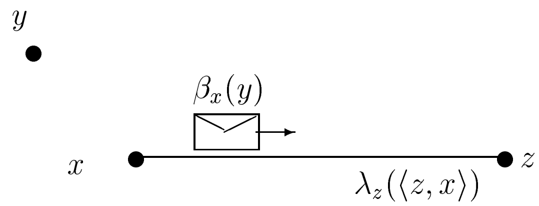

Figure 2: Communication of information about node y from node x to z.

The third property is, perhaps, the most important property of sense of direction and refers to the "translation" capability of a node. This property can be described by an example.

Consider the situation of node x sending to its neighbor z information about a node y (see Figure 2). Node y is known at x as $\beta_x(y)$, thus, the message sent by x will contain information about a node called "$\beta_x(y)$". Suppose that this information is received by z along the edge locally labeled with $\lambda_z(\langle z, x\rangle)$.

Informally, if there is sense of direction, node z, based on the label $\lambda_z(\langle z, x\rangle)$ and on the name $\beta_x(y)$, can deduce that the received information is about the node locally known as $\beta_z(y)$.

In other words, when a labeling is a sense of direction, the decoding function must be global: each node can consistently *translate the local views* of its neighbors.

Together these three properties indicate that, in a labeled graph with sense of direction, there is a very specific relationship between edge labels, local names and walks. The nature of this relationship, and thus, the definition of sense of direction, will be formally described in the following Sections.

3 Sense of Direction

Let $G = (V, E)$ be a connected graph where nodes correspond to entities and edges correspond to direct bidirectional communication links between entities. Let $E(x)$ denote the set of edges incident to node x.

A *walk* in G is a sequence of edges $[\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$, $\langle x_i, x_{i+1}\rangle \in E(x_i)$, in which the endpoint of one edge is the starting point of the next edge. A walk is a *cycle* if the starting point x_0 coincides with the ending point x_m. A *path* is a walk where all vertices are distinct.

Let $P[x]$ denote the set of all the walks with $x \in V$ as a starting point, and let $P[x, y]$ denote the set of walks starting from node $x \in V$ and ending in node $y \in V$.

3.1 Local Edge-Labelings, Local Node-Labeling and Local View

Each node $x \in V$ associates a label to each incident edge $e \in E(x)$. Given a graph $G = (V, E)$ and a set $\mathcal{L}$ of labels, a local *edge-labeling* (or labeling) function for $x \in V$ is any function $\lambda_x : E(x) \to \mathcal{L}$.

The *labeling* λ of G is the set of local labeling functions, that is $\lambda = \{\lambda_x : x \in V\}$. The resulting *labeled graph* will be denoted by (G, λ).

We now extend the definition of the labeling function from edges to walks. Given a labeling λ and a node $x \in V$, let $\Lambda_x : P[x] \to \mathcal{L}^+$ be the *walk-labeling function* defined as follows: for every walk $\pi \in P[x]$,

$$\Lambda_x(\pi) = (\lambda_x(\langle x, x_1\rangle), \lambda_{x_1}(\langle x_1, x_2\rangle), \ldots, \lambda_{x_{m-1}}(\langle x_{m-1}, x_m\rangle))$$

where $\pi = [\langle x, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$.

If a labeling λ is injective, the labeling is called a *Local Orientation*. A labeling λ has *Edge Symmetry* if there exists a bijection $\psi : \mathcal{L} \to \mathcal{L}$, such that, $\forall \langle x, y\rangle \in E$, $\lambda_x(\langle x, y\rangle) = \psi(\lambda_y(\langle y, x\rangle))$; that is, knowing the label at one side allows to derive the label on the other side of the edge. We say that a labeling is a *Locally Symmetric Orientation* when it is a local orientation with edge symmetry.

Let λ be a locally symmetric orientation, let $\alpha = [\alpha_1, \ldots, \alpha_m]$ be a sequence of labels corresponding to a walk in G, and let $\Psi : \mathcal{L}^+ \to \mathcal{L}^+$ be defined as follows: $\Psi(\alpha) = [\psi(\alpha_m), \ldots, \psi(\alpha_1)]$. By definition of Ψ, we have that:

Property 1 *$\forall \pi \in P[x, y]$, $\Lambda_x(\pi) = \Psi(\Lambda_y(\pi^R))$, where $\pi^R \in P[y, x]$ is the reverse of walk π.*

Each node x refers to the other nodes using local names. Let us stress that these local names are *not* necessarily identities (i.e., unique global identifiers). In fact, the system could be anonymous. A local *node-labeling (or naming) function* for $x \in V$ is an injective function $\beta_x : V \to \mathcal{N}$, where $\mathcal{N}$ is a set of names, such that $\forall y, z \in V$ $\beta_x(y) = \beta_x(z)$ iff $y = z$.

The *naming* function β for (G, λ) is the set of local naming functions, that is $\beta = \{\beta_x : x \in V\}$. We shall denote by (G, λ, β) the labeled graph with naming β.

Each node has a *local view* of the system. The local view $W(x)$ of node x consists of the set of names $\{\beta_x(y) : y \in V\}$ used by x.

A naming function is said to have *Name Symmetry* when, for any two nodes x and y, there exists a relationship between the name that x associates to y and the name that y associates to x, i.e., if there exists a bijection $\mu : \mathcal{N} \to \mathcal{N}$, such that, $\forall x, y \in V$, $\beta_x(y) = \mu(\beta_y(x))$. The naming function β is *homonymous* if $\forall x, y \in V$, $\beta_x(x) = \beta_y(y)$, i.e., it assigns the same "self" name to every vertex.

In the following, for clarity, names refer to node-labels whereas labels refer to edge labels.

3.2 Sense of Direction

We will now introduce the formal definition of sense of direction.

A *coding function* f of a graph (G, λ, β) is a function that associates names to sequences of labels of walks in G.

Definition 1 Coding Function
Given a labeling λ, a coding function *for λ is any function $f : \mathcal{L}^+ \to \mathcal{N} \cup \{\star\}$, where $\star \notin \mathcal{N}$ is a distinguished element called the* null name*, such that:*

$$f(\alpha) \in \mathcal{N} \text{ iff } \exists x \in V,\, \pi \in P[x] : \alpha = \Lambda_x(\pi)$$

Definition 2 Consistent Coding Function
A coding function f is consistent *in (G, λ, β) iff $\forall x, y \in V$, $\pi \in P[x, y]$,*

$$f(\Lambda_x(\pi)) = \beta_x(y)$$

Intuitively, the coding function is consistent if it allows the node to understand, from the labels associated to the edges, whether different walks from any given node x end in the same node or in different nodes.
By the above definition,

Property 2 *If f is consistent, then: $\forall x, y, z \in V$, $\forall \pi_1 \in P[x, y]$, $\pi_2 \in P[x, z]$*

$$f(\Lambda_x(\pi_1)) = f(\Lambda_x(\pi_2)) \quad \text{iff} \quad y = z$$

In other words, if a coding function is consistent, then walks originating from the same node are mapped to the same name if and only if they end in the same node.

Notice that, if β is homonymous, any consistent coding function must map to the same name all sequences of labels associated to cycles. Let us stress that consistency is *not* related with homonymy. In fact, for example, in the labeled square of Figure 3, consistent coding functions exist *only* for non-homonymous namings β. This is due to the fact that any consistent coding function f must be such that: $\beta_x(x) = f(a \cdot a) \neq f(b \cdot b)$, while $\beta_y(y) = f(b \cdot b)$; that is, nodes x and y must use different "self" names for consistency to exist.

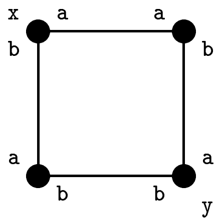

Figure 3: Graph where consistency exists only for non-homonymous namings.

A *decoding function* h for f in a graph (G, λ, β) is a function which associates a name to a given name and a label.

Definition 3 Decoding Function
Given a coding function f, a decoding function *h for f is any function $h : \mathcal{L} \times \mathcal{N} \to \mathcal{N} \cup \{\star\}$, where $\star \notin \mathcal{N}$ is a distinguished element called the* null name*, such that:*

$$h(l, q) \in \mathcal{N} \text{ iff } \exists \langle x, y\rangle \in E(x),\, \pi \in P[y] : \, l = \lambda_x(\langle x, y\rangle) \text{ and } q = f(\Lambda_y(\pi))$$

To guarantee a consistent "translation" mechanism, a decoding function requires an additional property called *Consistent Local Decoding.*

Definition 4 Consistent Decoding Function
Given a consistent coding function f, a decoding function h for f is consistent *iff $\forall\langle x, y\rangle \in E(x), \pi \in P[y, z]$*

$$h(\lambda_x(\langle x, y\rangle), f(\Lambda_y(\pi))) = \beta_x(z)$$

The existence of a consistent decoding function is clearly a crucial property since it would allow the nodes to solve global problems while working solely and truly in a local mode.

Definition 5 $\mathcal{SD}$ - Sense of Direction
Given (G, λ, β), λ is a Sense of Direction *($\mathcal{SD}$) iff the following conditions hold:*
1) there exists a consistent coding function f for λ,
2) there exists a consistent decoding function h for f.

Note that condition 1 implies that λ must be a local orientation; however, it does not imply condition 2 (see Example 3).

Example 1 - Labeling which is a $\mathcal{SD}$
Consider a system (G, λ, β) where

G is a 2-dimensional mesh;

λ is the natural "compass" assignment of the labels $\mathcal{L} = \{north,\ south,\ east,\ west\}$ (see Figure 1). The labeling clearly has edge symmetry, e.g. $north = \psi(south)$.

β is the following function: $\forall x, y \in V$, if $x \neq y$ $\beta_x(y)$ is the (lexicographically ordered) sequence of labels corresponding to the shortest path between x and y; if $x = y$, $\beta_x(x)$ is the empty string. For example, in Figure 1, $\beta_x(y) = [east, north]$

Note that, in this system, $\mathcal{N} \subset \mathcal{L}^*$.

We will now show that this labeling λ is a $\mathcal{SD}$. Given a sequence α of labels, let $\bar{\alpha}$ be the sequence obtained from α by deleting every pair of labels l, l' such that $l = \psi(l')$ and lexicographically sorting the resulting sequence. To show that λ is a $\mathcal{SD}$ we show that there exists a consistent coding function f in (G, λ, β). Consider, for example, the function $f : \mathcal{L}^+ \rightarrow \mathcal{N} \cup \{\star\}$ such that

$$f(\alpha) = \begin{cases} \bar{\alpha} & \text{if } \exists x \in V,\ \pi \in P[x] : \alpha = \Lambda_x(\pi) \\ \star & \text{otherwise} \end{cases}$$

It is easy to verify that function f, applied to any walk between x and y, coincides with $\beta_x(y)$. Note that the name does not necessarily correspond to an existing walk. In the example of Figure 1, we have $f([north,\ east,\ north,\ south]) = [east, north] = \beta_x(y)$ and $f([east, west, north, east]) = [east, north] = \beta_x(y)$. It is easy to see that the function $h(l, n) = f(l \circ n)$, where $\circ$ is the concatenation operator, is a consistent decoding function for f. This labeling is an instance of *Contracted Sense of Direction*, which will be discussed in Section 5.3.

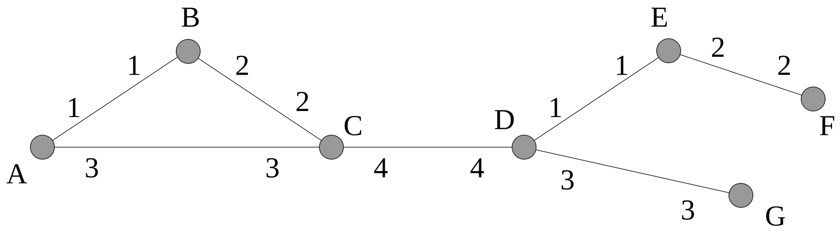

Figure 4: Labeling which is not a $\mathcal{SD}$.

Example 2 - Labeling which is not a $\mathcal{SD}$

Consider the system (G, λ) of Figure 4. To see that λ cannot be a $\mathcal{SD}$ for any choice of β, consider the four paths $\pi_1 = [\langle A, B\rangle,\ \langle B, C\rangle]$, $\pi_2 = [\langle A, C\rangle]$, $\pi_3 = [\langle D, E\rangle,\ \langle E, F\rangle]$ and $\pi_4 = [\langle D, G\rangle]$. For these walks we should have $f(\Lambda_A(\pi_1)) = f(\Lambda_A(\pi_2)) = \beta_A(C)$ and $f(\Lambda_D(\pi_3)) \neq f(\Lambda_D(\pi_4))$. On the other hand, we have $\Lambda_A(\pi_1) = [1, 2] = \Lambda_D(\pi_3)$, and $\Lambda_A(\pi_2) = [3] = \Lambda_D(\pi_4)$, which proves that the coding function is not consistent regardless of β.

3.3 Weak Sense of Direction and Associativity

A weaker form of sense of direction in a system (G, λ, β) is represented by a labeling λ such that there exists a consistent coding function f, but not necessarily a consistent decoding function h for f.

Definition 6 $\mathcal{WSD}$ - Weak Sense of Direction
Given (G, λ, β), λ *is a* Weak Sense of Direction *($\mathcal{WSD}$) iff there exists a consistent coding function* f.

It is important to note that not every $\mathcal{WSD}$ is a $\mathcal{SD}$, that is, there are systems where no consistent coding function can be consistently decoded. Consider the following example:

Example 3
Consider the labeled graph (G, λ) shown in Figure 5. It is simple to find a naming function β and a consistent coding function such that λ is a $\mathcal{WSD}$ in (G, λ, β), for example by using the recognition algorithm of [9]. However, for any choice of consistent coding function f, there exists no corresponding decoding function: λ is not a $\mathcal{SD}$. The proof can be found in [9]; here we give a sketch of it.
Let f be a consistent coding function for λ. From node X, we must have $f([a, b]) = f([c, d])$, and from node W, $f([e, g]) = f([c, d])$, it follows that $f([a, b]) = f([e, g])$. Let h be a consistent decoding function. From node Y, we would have $h(m, f([a, b])) = f(p)$, thus,

$$h(m, f([e, g])) = f(p) \tag{1}$$

Consider now node Z; $h(m, f([e, g])) = f(q)$. By (1), it follows that $f(p) = f(q)$ but this contradicts the fact that, from node R, $f(p) \neq f(q)$. This proves that λ is a $\mathcal{WSD}$ but not a $\mathcal{SD}$.

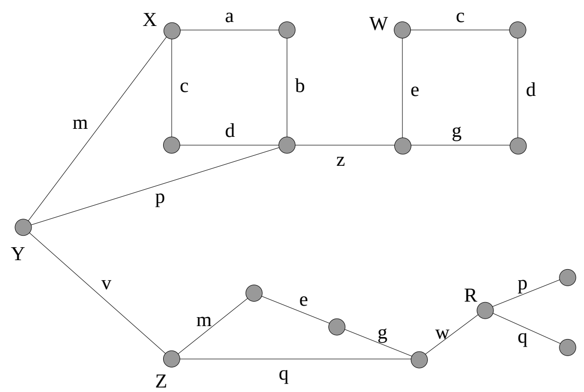

Figure 5: A labeling which is $\mathcal{WSD}$ but not $\mathcal{SD}$

We now see a condition which guarantees the existence of a consistent decoding function.

Definition 7 *Let $\mathcal{N} \subseteq \mathcal{L}^+$. A coding function f is* associative *iff,* $\forall\langle x,y\rangle \in E(x)$, $\forall\pi \in P[y]$,

$$f(\lambda_x(\langle x,y\rangle) \circ f(\Lambda_y(\pi))) = f(\lambda_x(\langle x,y\rangle)) \circ \Lambda_y(\pi))$$

where $\circ$ is the concatenation operator.

Theorem 1 *Let $\mathcal{N} \subseteq \mathcal{L}^+$ and λ be a Weak sense of direction. If the corresponding coding function f is associative, λ is a sense of direction.*

Proof Consider the following decoding function $h(l,n) = f(l \circ n)$, where $l \in \mathcal{L}$, $n \in \mathcal{N}$ and $\circ$ is the concatenation operator. Note that $\mathcal{N} \subseteq \mathcal{L}^+$, thus $[l \circ n] \in \mathcal{L}^+$. We have that $\forall\langle x,y\rangle$, $\forall\pi \in P[y,z]$,

$$h(\lambda_x(\langle x,y\rangle), f(\Lambda_y(\pi))) = f(\lambda_x(\langle x,y\rangle) \circ f(\Lambda_y(\pi))$$

By definition of associative coding function, we have that:

$$f(\lambda_x(\langle x,y\rangle)) \circ f(\Lambda_y(\pi)) = f(\lambda_x(\langle x,y\rangle) \circ \Lambda_y(\pi))$$

But

$$f(\lambda_x(\langle x,y\rangle) \circ \Lambda_y(\pi)) = \beta_x(z)$$

Thus, $h(\lambda_x(\langle x,y\rangle), f(\Lambda_y(\pi))) = \beta_x(z)$, and the decoding function $h(l,n) = f(l \circ n)$ is consistent for f in (G, λ, β). □

4 Translation of Local Views

In the previous Section, we have seen that the availability of sense of direction in (G, λ, β) implies the existence of a decoding function h which allows any node x to translate the local view of its neighbors into its own local view.

In this Section, we consider a system (G, λ, β), where λ is a $\mathcal{SD}$, and focus on what other knowledge of the system can be derived from the viewpoint of a node x. We discuss properties of $\mathcal{SD}$ which allow x to derive information about the local views of other nodes. In particular, we show that, under certain conditions, the availability of $\mathcal{SD}$ allows the translation process to be possible beyond the immediate neighborhood.

4.1 Translation of Incident Walks

The following properties of sense of direction express what knowledge can be derived by x from the sequence of labels corresponding to a walk in $P[x]$.

Given a system (G, λ, β), where λ is a $\mathcal{SD}$, let $\pi = [\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$ and let $\alpha = \Lambda_{x_0}(\pi)$ be the corresponding sequence of labels. If node x_0 knows the sequence α and the consistent coding function f, then the following properties hold.

Property 3 *Node x_0 can derive the local names of all the nodes on the walk, that is, $\{\beta_{x_0}(x_i) : \ i = 1, 2, \ldots, m\}$.*

Proof By definition of consistent coding function, to derive $\beta_{x_0}(x_i)$, it suffices to compute $f(\alpha_i)$ where $\alpha_i = \Lambda_{x_0}([\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{i-1}, x_i\rangle])$. □

That is, x_0 can translate into its local view the names of all the nodes on the walks. It can actually do more, as expressed by the following:

Property 4 *For any x_i, x_j in the walk, $i < j$, node x_0 can derive $\{\beta_{x_i}(x_j) : \ j = 1, 2, \ldots, m;\ i = 1, 2, \ldots, j\}$.*

Proof To derive the name that x_i associates to x_j, with $i < j$, it suffices to use the coding function. In fact, by definition of consistent coding function, $\beta_{x_i}(x_j) = f(\alpha_{i,j})$, where $\alpha_{i,j} = \Lambda_{x_i}([\langle x_i, x_{i+1}\rangle, \langle x_{i+1}, x_{i+2}\rangle, \ldots, \langle x_{j-1}, x_j\rangle])$ □

In other words, x_0 can translate into the local view of x_i the names of the nodes following x_i in the walk.

If there is also edge symmetry, the translation capabilities of x_0 increase, as shown by the following.

Property 5 *If λ has edge symmetry, node x_0 can derive $\{\beta_{x_i}(x_j) : \ i, j = 1, 2, \ldots, m\}$ for any x_i, x_j in the walk.*

Proof Let λ have edge symmetry, and let ψ be the edge symmetry function. Consider now any x_i, x_j in the walk. If $i \leq j$, the Property holds by Property 4. Let $i > j$. By definition of consistent coding function, $\beta_{x_i}(x_j) = f(\alpha_{i,j})$, where $\alpha_{i,j} = \Lambda_{x_i}([\langle x_i, x_{i-1}\rangle, \ldots, \langle x_{j+1}, x_j\rangle]) = [\lambda_{x_i}(\langle x_i, x_{i-1}\rangle), \ldots, \lambda_{x_{j+1}}(\langle x_{j+1}, x_j\rangle)]$. The sequence $\alpha_{i,j}$ is clearly derivable since, by definition of edge symmetry function, $\lambda_{x_l}(\langle x_l, x_{l-1}\rangle) = \psi(\lambda_{x_{l-1}}(\langle x_{l-1}, x_l\rangle))$, for any $x_l \in \pi$. □

4.2 Translation of Remote Walks

The properties of the previous Section refer to the knowledge which can be derived by a node x from the labels of a walk incident on x. We consider now what x can derive from the labels of a walk which might not contain x.

Property 6 *Given a system (G, λ, β), where λ is a $\mathcal{SD}$, let $\pi = [\langle y_0, y_1\rangle, \langle y_1, y_2\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$, and $\alpha = \Lambda_{y_0}(\pi)$. If node x knows the sequence α and the consistent coding function f, we have that:*
1) For any $y_j \in \pi$, node x can derive $\{\beta_{y_0}(y_j) : \ j = 1, 2, \ldots, m\}$;
2) for any $y_i, y_j \in \pi$, $i < j$, node x can derive $\{\beta_{y_i}(y_j)\}$;
3) if λ has edge symmetry, node x can derive $\{\beta_{y_i}(y_j)\}$, for any $y_i, y_j \in \pi$.

Proof Analogous to the proofs of Properties 3, 4, and 5. □

That is, x can derive the name of any node in the walk in the local view of the origin of the walk. Moreover, it can derive how a node y_i refers to the nodes following it in the walk and, in presence of edge symmetry it can derive the name of any node in the view of any other node.

Note that all the translation of remote walks described above hold without requiring x to know the local name of any node in the walk.

Further note that, unless x is in the walk (which is the case covered by Section 4.1), it cannot in general translate those names in *its* own local view.

Knowledge of the local name of the origin of the walk does not seem to make any difference for the systems considered here. In the next Section we will consider the impact of such a knowledge in stronger systems.

4.3 Translation with Symmetric Sense of Direction

In this Section we will show that, in systems with both edge and name symmetry, knowledge of the origin of a walk has an impact on the translation capabilities of a node outside the walk.

Definition 8 $\mathcal{SSD}$ - Symmetric Sense of Direction
Given (G, λ, β)*,* λ *is a* Symmetric Sense of Direction *if* λ *is a* $\mathcal{SD}$ *with edge symmetry and* β *has name symmetry.*

Example 4 Consider the system (G, λ, β) of Example 1. Recall that $\beta_x(y)$ is the (lexicographically ordered) sequence of labels corresponding to the shortest path between x and y. We can easily see that β has name symmetry and, thus, λ is a Symmetric Sense of Direction. In fact, for every $x \in V$, $\beta_x(x)$ is the empty string; and for $x \neq y$:

$$\Psi(\beta_x(y)) = [\psi(l_k), \psi(l_{k-1}), \ldots, \psi(l_0)] = \beta_y(x)$$

where $\beta_x(y) = [l_0, l_1, \ldots, l_k]$ and Ψ is the walk-symmetry function. Note that, due to the particular choice of labels, it is not necessary to re-order the sequence of labels to obtain the name that y associates to x.

We will now show the impact of $\mathcal{SSD}$ on the translation of local views. Given a system (G, λ, β), where λ is a $\mathcal{SSD}$, let $\pi = [\langle y_0, y_1\rangle, \langle y_1, y_2\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$, and $\alpha = \Lambda_{y_0}(\pi)$. Let node x know the sequence α, the consistent decoding function h, and the name $\beta_x(y_0)$ that corresponds to the origin of π.

Property 7 *Node* x *can derive the local name* $\beta_x(y_1)$ *of* y_1*.*

Proof Let ψ and μ be the edge and the name symmetry function, respectively. By definition of name symmetry function,

$$\beta_{y_0}(x) = \mu(\beta_x(y_0)) \text{ and } \beta_x(y_1) = \mu(\beta_{y_1}(x))$$

By definition of consistent decoding function

$$\beta_{y_1}(x) = h(\lambda_{y_1}(\langle y_1, y_0 \rangle), \beta_{y_0}(x))$$

By definition of edge symmetry function,

$$\lambda_{y_1}(\langle y_1, y_0 \rangle) = \psi(\lambda_{y_0}(\langle y_0, y_1 \rangle))$$

It follows that

$$\beta_x(y_1) = \mu(\beta_{y_1}(x)) = \mu(h(\psi(\lambda_{y_0}(\langle y_0, y_1 \rangle)), \mu(\beta_x(y_0))))$$

$\square$

The previous property shows that, when there is a symmetric sense of direction and when x knows α, x can translate into its local view the name of the immediate neighbor y_1 of y_0. We can now obtain the following:

Property 8 *Node x can derive the local names $\beta_x(y_i)$ of all $y_i \in \pi$.*

Proof Let ψ be the edge symmetry function and let μ be the name symmetry function. By Property 7, x can derive the name $\beta_x(y_1)$. By induction, assume that x can derive $\beta_x(y_i)$, and consider $\beta_x(y_{i+1})$. By definition of consistent decoding function,

$$\beta_x(y_{i+1}) = h(\lambda_{y_{i+1}}(\langle y_{i+1}, y_i \rangle), \beta_x(y_i))$$

By definition of edge symmetry function,

$$\lambda_{y_{i+1}}(\langle y_{i+1}, y_i \rangle) = \psi(\lambda_{y_i}(\langle y_i, y_{i+1} \rangle))$$

and by the induction hypothesis $\beta_x(y_i)$ is derivable. The property follows. $\square$

5 Classes of Sense of Direction

In this Section, several instances of $\mathcal{SD}$ are introduced. These instances include all the labelings used in the literature on senses of direction, and are grouped in four general classes: *Cartographic*, *Chordal*, *Contracted*, and *Neighboring* $\mathcal{SD}$s.

5.1 Cartographic Sense of Direction

A *Cartographic Sense of Direction* is any $\mathcal{SD}$ which uses properties of an embedding of $G = (V, E)$ in the plane. Instances of cartographic $\mathcal{SD}$s are the following.

5.1.1 Coordinate $\mathcal{SD}$

A coordinate labeling is one which labels the edge $\langle u, v\rangle$ at u by the relative coordinates of v (see Figure 6).

Definition 9 Coordinate labeling
Given an embedding of G *in the plane,* λ *is a* Coordinate labeling *iff:*

$$\forall\langle u, v\rangle \in E[u] \quad \lambda_u(\langle u, v\rangle) = (x_1 - x_0, y_1 - y_0)$$

where (x_0, y_0) *and* (x_1, y_1) *are the coordinates of* u *and* v*, respectively, in the embedding.*

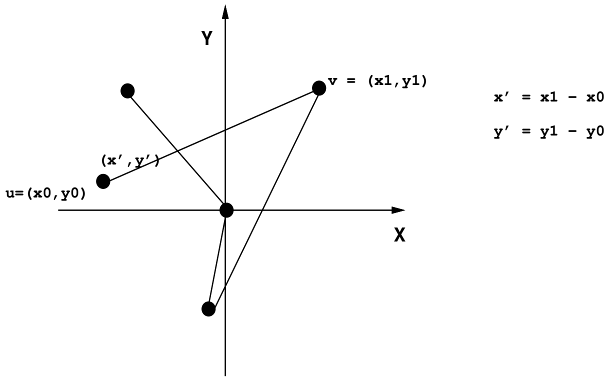

Figure 6: Example of coordinate $\mathcal{SD}$.

Note that the labels are elements of $\mathbf{R}^2$. When the local names of the nodes are the appropriate relative coordinates, the coordinate labeling is a $\mathcal{SD}$.

Theorem 2 *Let* λ *be a coordinate labeling and* $\forall u, v \in V$*, let* $\beta_u(v) = (x_1 - x_0, y_1 - y_0)$*, where* $u = (x_0, y_0)$ *and* $v = (x_1, y_1)$*. Then* λ *is a* $\mathcal{SD}$*.*

Proof To verify that λ is a $\mathcal{SD}$, consider the coding function f defined as follows: $\forall \pi = [\langle u_0, u_1\rangle, \ldots, \langle u_{m-1}, u_m\rangle] \in P[u_0, u_m]$, where $u_i = (x_i, y_i)$

$$\begin{aligned} f(\Lambda_{u_0}(\pi)) &= f([(x_1 - x_0, y_1 - y_0), \ldots, (x_m - x_{m-1}, y_m - y_{m-1})]) \\ &= \left(\sum_{i=1}^{m} x_i - x_{i-1}, \sum_{i=1}^{m} y_i - y_{i-1}\right) \end{aligned}$$

It follows that

$$f(\Lambda_{u_0}(\pi)) = (x_m - x_0, y_m - y_0) = \beta_{u_0}(u_m)$$

Thus, f is consistent in (G, λ, β).

Consider now the following decoding function h:
$\forall\langle u_0, u_1\rangle \in E(u_0)$, $\forall\pi \in P[u_1]$, $\pi = [\langle u_1, u_2\rangle, \ldots, \langle u_{m-1}, u_m\rangle]$ where $u_i = (x_i, y_i)$

$$h((x_1 - x_0, y_1 - y_0), (x_m - x_1, y_m - y_1)) = (x_m - x_0, y_m - y_0)$$

The decoding function is consistent. In fact,

$$(x_m - x_0, y_m - y_0) = \beta_{u_0}(u_m)$$

Thus, λ is a $\mathcal{SD}$. □

Note that the set of names and the set of labels coincide: $\mathcal{L} = \mathcal{N} = \mathbf{R}^2$. We shall call this labeling a *Coordinate* $\mathcal{SD}$.

Theorem 3 *Coordinate sense of direction is symmetric.*

Proof Let λ be a coordinate $\mathcal{SD}$ in (G, λ, β). To prove the Theorem, we must show that λ and β have edge and name symmetry, respectively. Consider the function $\psi : \mathbf{R}^2 \to \mathbf{R}^2$, $\psi((x, y)) = -(x, y)$. Function ψ is an edge symmetry function: let $u = (x_0, y_0)$ and $v = (x_1, y_1)$ be neighbors, then,

$$\psi(\lambda_u(\langle u, v\rangle)) = \psi(x_1 - x_0, y_1 - y_0) = -(x_1 - x_0, y_1 - y_0)$$

It follows that

$$\psi(\lambda_u(\langle u, v\rangle)) = (x_0 - x_1, y_0 - y_1) = \lambda_v(\langle v, u\rangle)$$

Thus, λ has edge symmetry. Since $\beta_u(v) = -\beta_v(u)$, it follows that ψ is also a name symmetry function. Hence, β has name symmetry and λ is a symmetric sense of direction. □

5.1.2 Polar Sense of Direction

A particular class of embeddings of G is obtained by placing the nodes on the unit circle centered in the origin, and by connecting each pair of incident nodes by a straight line. (See Figure 7). Any embedding of this type will be called a *polar representation* of G.

Definition 10 Polar labeling
Given a graph (G, λ) *in polar representation,* λ *is a* Polar labeling *iff:*

$$\forall\langle x, y\rangle \in E[x] \quad \lambda_x(\langle x, y\rangle) = \alpha_{xy}$$

where α_{xy} *is the angle under the arc* $\langle x, y\rangle$.

When the local names of the nodes are the appropriate angles, the polar labeling is a $\mathcal{SD}$.

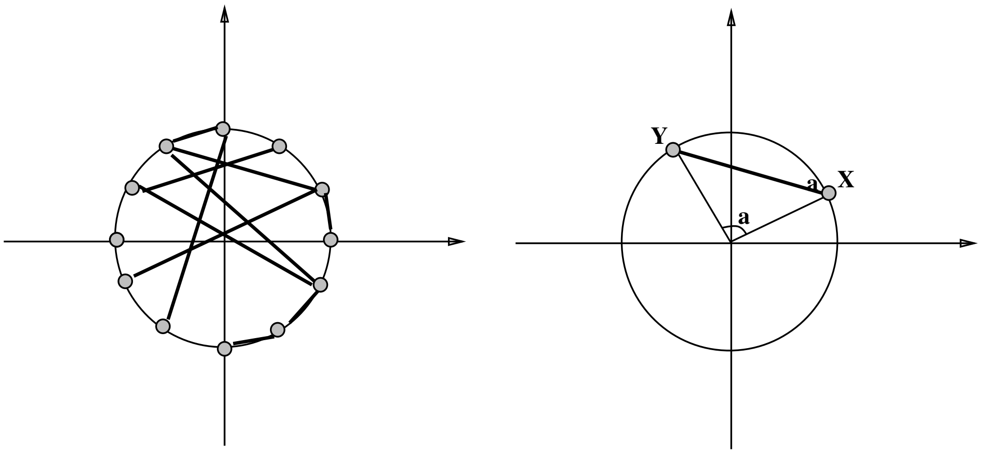

Figure 7: Polar representation of a graph and polar $\mathcal{SD}$.

Theorem 4 *Let λ be a polar labeling and $\forall x, y \in V$ let $\beta_x(y) = \alpha_{xy}$. Then λ is a $\mathcal{SD}$.*

Proof The proof is similar to the one of Theorem 2 considering the following coding function f: $\forall \pi \in P[x_0]$, $\pi = [\langle x_0, x_1\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$

$$f(\Lambda_x(\pi)) = f(\alpha_{x_0x_1}, \alpha_{x_1x_2}, \ldots, \alpha_{x_{m-1}x_m}) = \sum_{i=0}^{m-1} \alpha_{x_ix_{i+1}} \bmod 2\pi$$

And the following decoding function h:
$\forall\langle x_0, y_0\rangle \in E(x_0),\ \forall \pi \in P[y_0]$, $\pi = [\langle y_0, y_1\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$

$$h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) = \alpha_{x_0y_0} + f(\Lambda_y(\pi)) \bmod 2\pi$$

$$= \alpha_{x_0y_0} + \alpha_{y_0y_m} \bmod 2\pi = \alpha_{x_0y_m}$$

□

We shall call the above labeling a *Polar $\mathcal{SD}$*.

Theorem 5 *Polar sense of direction is symmetric.*

Proof The proof is similar to the one of Theorem 3 where the edge symmetry function is: $\psi : \mathcal{L} \to \mathcal{L}$, $\psi(\alpha) = 2 \cdot \pi - \alpha$ and the name symmetry function coincides with the edge symmetry function.

□

5.2 Chordal Sense of Direction

A *Chordal* labeling of a graph $G = (V, E)$, with $|V| = n$, is defined by fixing a cyclic ordering of the nodes and labeling each incident link by the distance in the above cycle.

Definition 11 *Let $\gamma : V \to V$ be a successor function defining a cyclic ordering of the nodes of (G, λ) and let $\gamma^k(x) = \gamma^{k-1}(\gamma(x))$ for $k > 0$. Let $\delta : V \times V \to \{0, \ldots, n-1\}$ be the corresponding distance function, i.e., $\delta(x, y)$ is the smallest k such that $\gamma^k(x) = y$. The labeling λ is a* Chordal *labeling iff, $\forall \langle x, y\rangle \in E(x)$:*

$$\lambda_x(\langle x, y\rangle) = \delta(x, y)$$

Note that γ is the function defining the cyclic ordering of the nodes, and different chordal labelings arise from different γs. Further note that, if the link between p and q is labeled by d at node p, it is labeled by $n - d$ at node q (see Figure 8).

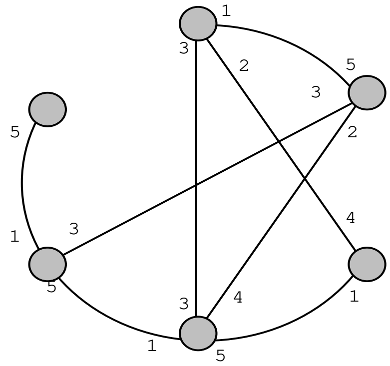

Figure 8: A Graph with chordal sense of direction.

Also when the local names of the nodes are relative distances in the cyclic ordering, the chordal labeling is a $\mathcal{SD}$.

Theorem 6 *Let λ be a chordal labeling and $\forall x, y$ let $\beta_x(y) = \delta(x, y)$. Then λ is a $\mathcal{SD}$.*

Proof The proof is similar to the one of Theorem 2, considering the coding function f defined as follows: $\forall \pi \in P[x_0],\ \pi = (\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle)$

$$\begin{aligned} f(\Lambda_{x_0}(\pi)) &= f(\lambda_{x_0}(\langle x_0, x_1\rangle), \lambda_{x_1}(\langle x_1, x_2\rangle), \ldots, \lambda_{x_{m-1}}(\langle x_{m-1}, x_m\rangle)) \\ &= \sum_{i=0}^{m-1} \lambda_{x_i}(\langle x_i, x_{i+1}\rangle) \bmod n \end{aligned}$$

and considering the following decoding function h:
$\forall \langle x_0, y_0\rangle \in E(x_0),\ \forall \pi \in P[y_0]$

$$h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) = \lambda_{x_0}(\langle x_0, y_0\rangle) + f(\Lambda_{y_0}(\pi)) \bmod n$$

□

Note that the set of names and the set of labels coincide: $\mathcal{L} = \mathcal{N} = \mathbf{Z}_n^+$. We shall call this labeling *Chordal* $\mathcal{SD}$.

Theorem 7 *Chordal sense of direction is symmetric.*

Proof The proof is similar to the one of Theorem 3 with the following edge symmetry function: $\psi : \mathbf{Z}_n^+ \to \mathbf{Z}_n^+$, $\psi(d) = n - d$, (where $n = |V|$ and $d \in \mathbf{Z}_n^+$). It is easy to see that ψ is also a name symmetry function.

$\square$

The chordal labeling is the natural labeling for *Chordal Rings* (also called *Circulant Graphs* [7], or *Loop Networks* [5]) from which it takes the name. It can obviously be defined for any graph. In the literature, the chordal $\mathcal{SD}$ has been extensively investigated in specific topologies. Sometimes called *Distance* $\mathcal{SD}$, it has been studied and exploited in complete graphs [20, 29, 34, 35, 36, 46] and chordal rings [2, 21, 38, 39]. Its impact has been also investigated in hypercubes [12], as well as in systems of unknown topology (the *Arbitrary Graph* case) [32].

5.3 Contracted Sense of Direction

In this Section, we will analyze a rather general class of $\mathcal{SD}$ based on labelings with locally symmetric orientation (i.e., with both local orientation and edge symmetry). As we will see, this class contains the traditional labelings for meshes, tori, and hypercubes, among others.

5.3.1 Simple Contraction

Let λ be a labeling with locally symmetric orientation, and let ψ be the corresponding edge symmetry function.

Definition 12 Contraction
Given a sequence $\alpha \in \mathcal{L}^+$, the contraction of α is the sequence $\overline{\alpha}$ of labels obtained from α by deleting every pair of labels l and l' such that $l = \psi(l')$, and lexicographically sorting the resulting sequence.

Definition 13 Contracted Labeling
A labeling λ with edge symmetry is contracted *iff $\forall x, y \in V$, $\forall \pi_1, \pi_2 \in P[x, y]$*

$$\overline{\Lambda_x(\pi_1)} = \overline{\Lambda_x(\pi_2)}$$

That is, if λ is *contracted*, then all the sequences of all the walks from x to y have the same contraction, which we shall denote by $\overline{\Lambda_{x,y}}$.

When the local names of the nodes are the appropriate contractions, the contracted labeling is a $\mathcal{SD}$.

Theorem 8 *Let λ be a contracted labeling and $\forall x, y \in V$ let $\beta_x(y) = \overline{\Lambda_{x,y}}$. Then λ is a $\mathcal{SD}$.*

Proof To verify that it is a $\mathcal{SD}$, consider the coding function f defined as follows: $\forall \pi \in P[x_0]$, $\pi = [\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$,

$$f(\Lambda_{x_0}(\pi)) = \overline{\Lambda_{x_0,x_m}}$$

It follows that

$$f(\Lambda_{x_0}(\pi)) = \beta_{x_0}(x_m)$$

Thus, f is consistent.

Consider now the following decoding function h: $\forall \langle x_0, y_0\rangle \in E(x_0)$, $\forall \pi \in P[y_0]$, $\pi = [\langle y_0, y_1\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$,

$$h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) = \overline{\lambda_{x_0}(\langle x_0, y_0\rangle) \circ f(\Lambda_{y_0}(\pi))}$$

where $\circ$ is the concatenation operator. It follows that

$$\begin{aligned} h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) &= \overline{\lambda_{x_0}(\langle x_0, y_0\rangle) \circ \overline{\Lambda_{y_0 y_m}}} \\ &= \overline{\Lambda_{x_0 y_m}} = \beta_{x_0}(y_m) \end{aligned}$$

Thus, h is consistent and λ is a $\mathcal{SD}$. □

We shall call this labeling *Contracted $\mathcal{SD}$*.

Theorem 9 *Contracted Sense of Direction is Symmetric.*

Proof Let λ be a Contracted $\mathcal{SD}$ in (G, λ, β). To prove that it is a Symmetric Sense of Direction, we have to show that λ and β have edge and name symmetry, respectively. The labeling λ has edge symmetry by definition. To show that β has name symmetry, first observe that it is, by definition, homonymous; in fact, $\beta_x(x)$ is the empty string for any $x \in V$. Consider now $x \neq y$, and let ψ and Ψ be the edge and the walk symmetry function, respectively. By Property 1, $\forall \pi \in P[x, y]$, $\Lambda_x(\pi) = \Psi(\Lambda_y(\pi))$. Since $x \neq y$, $\overline{\Lambda_x(\pi)}$ is not the empty string and, thus, $\Psi(\overline{\Lambda_y(\pi)})$ is defined. By definition of β and since λ is a contracted labeling, it follows that

$$\beta_x(y) = \overline{\Lambda_x(\pi)} = \overline{\Psi(\Lambda_y(\pi))} = \Psi(\overline{\Lambda_y(\pi)}) = \Psi(\beta_y(x))$$

Thus, β has name symmetry follows that λ is a Symmetric Sense of Direction. □

Example 5 *Dimensional $\mathcal{SD}$- contraction in hypercubes*
The traditional labeling of a $d-dimensional$ hypercube, shown in Figure 9 for $d = 3$, is an instance of contracted $\mathcal{SD}$ where the local name $\beta_x(y)$ is the (sorted) sequence of labels (dimensions) on the shortest path between x and y. In fact, it is a locally symmetric orientation where the edge symmetry function ψ is the identity function. It is easy to verify that, in the hypercube, this labeling is a contracted labeling. Consider, for example, the two walks π_1 and π_2 from x to y in Figure 9 with $\Lambda_x(\pi_1) = [3, 2, 3, 1, 3]$ and $\Lambda_x(\pi_2) = [1, 2, 3]$, in this case we have $\overline{\Lambda_x(\pi_1)} = [1, 2, 3] = \overline{\Lambda_x(\pi_2)}$, and $\beta_x(y) = \beta_y(x) = [1, 2, 3]$.

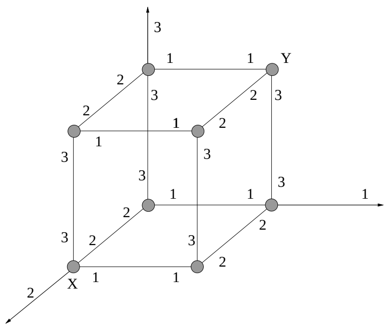

Figure 9: Hypercube with dimensional sense of direction.

Example 6 *Compass $\mathcal{SD}$ - contraction in meshes.*
Each type of d-dimensional orthogonal mesh has a natural labeling which forms a particular case of contracted $\mathcal{SD}$ where the local name $\beta_x(y)$ is the (sorted) sequence of labels on the shortest path between x and y.

Consider, for example, the traditional labeling of a d-dimensional quadrilateral mesh, shown in Figure 1 for $d = 2$. This labeling is a locally symmetric orientation where the edge symmetry function ψ is such that $\psi(north) = south$, $\psi(east) = west$, and so on. It is easy to verify that this labeling is contracted. Consider, for example, the two walks π_1 and π_2 from X to Y in Figure 1 with $\Lambda_X(\pi_1) = [north,\ east,\ north,\ south]$ and $\Lambda_X(\pi_2) = [east,\ north,\ east,\ west]$, we have $\overline{\Lambda_X(\pi_1)} = [east, north] = \overline{\Lambda_X(\pi_2)}$. In this case, $\beta_X(Y) = [east, north]$, while $\beta_Y(X) = \mu(\beta_X(Y)) = [south, west]$, where μ is the name symmetry function.

In the literature, the impact of contracted labelings has been extensively studied (e.g., in [12, 24, 26, 30, 40, 41, 48, 49, 51]).

5.3.2 Contraction with Wraparound

An immediate generalization of the contracted $\mathcal{SD}$ is the one which applies to topologies with *wraparound* (e.g., rings, tori, etc.). In this case, the sequences associated to walks are transformed using only a subset of the labels (termed "allowed directions") and taking into account the structure of the wraparound.

Let λ be a labeling with locally symmetric orientation, and let ψ be the corresponding edge symmetry function.

Definition 14 Contraction with Wraparound
Let $L = \{l_1, \ldots, l_m\}$ in $\mathcal{L}^m$ where $l_i \neq \psi(l_j)$ for $i \neq j$, and let $W = \{w_1, \ldots, w_m\} \in \mathbf{Z}^m$. Given a sequence of labels $\alpha \in \mathcal{L}^+$, the contraction with Wraparound W and Allowed Directions L *(shortly, LW-*contraction*) of α is the sequence $\widehat{\alpha}$ of labels obtained from the contraction $\overline{\alpha}$ by*
1. replacing any subsequence of k consecutive l_is with a subsequence of $w_i - k$ consecutive

$\psi(l_i)$, where $k \geq 0$, $l_i \in L$, $w_i \in W$ (note that $w_i > k$ by definition); and
2. lexicographically sorting the resulting sequence.

Examples of contractions with wrap-around are given in Examples 7 and 8.

Using this operation, the notions of contracted labeling and contracted $\mathcal{SD}$ are extended as follows.

Definition 15 *LW*-contracted labeling
*A labeling λ with edge symmetry is LW-*contracted *iff $\forall x, y,\ \forall \pi_1, \pi_2 \in P[x, y]$*

$$\widehat{\Lambda_x(\pi_1)} = \widehat{\Lambda_x(\pi_2)}$$

That is, if λ is LW-contracted, then all the sequences of all the walks from x to y have the same LW-contraction, which we shall denote by $\widehat{\Lambda_{x,y}}$.

Theorem 10 *Let λ be a LW-contracted labeling and $\forall x, y$ let $\beta_x(y) = \widehat{\Lambda_{x,y}}$. Then λ is a $\mathcal{SD}$.*

The proof follows the same lines as the one of Theorem 8. Similarly, we can prove the following.

Theorem 11 *LW-contracted sense of direction is symmetric.*

Each d-dimensional torus has a natural labeling which forms a particular case of LW-contracted $\mathcal{SD}$ where the local name $\beta_x(y)$ is the (sorted) sequence of labels on the shortest path between x and y using only the allowed directions.

Example 7 - *Contraction in rings*
Consider a Ring (i.e., a 1-dimensional torus) of size n with the traditional labeling with $\mathcal{L} = \{left,\ right\}$ and with edge symmetry function ψ: $right = \psi(left)$. This labeling is LW contracted where the wraparound is $W = \{n\}$ and the direction is, for example, $L = \{left\}$. Consider, for example, the two walks π_1 and π_2 from x to y, in a Ring of size $n = 7$, with $\alpha_1 = \Lambda_x(\pi_1) = [left, left, left, left]$ and $\alpha_2 = \Lambda_x(\pi_2) = [right, right, left, right, right]$. The corresponding LW-contractions are $\widehat{\alpha_1} = [right, right, right] = \widehat{\alpha_2}$. In this case $\beta_x(y) = [right, right, right]$.

Example 8 - *Contraction in tori: compass $\mathcal{SD}$*
Consider the 2-dimensional torus of size $n_1 \times n_2$ with the traditional "compass" assignment of the labels $\mathcal{L} = \{north,\ south,\ east,\ west\}$ (see Figure 10) and edge symmetry function ψ: $north = \psi(south)$, $east = \psi(west)$. Clearly, the set of wraparounds is $W = \{n_1, n_2\}$, the corresponding set of allowed directions is, for example, $L = \{north, west\}$. The labeling λ is a LW-contracted labeling. Consider, for example, the two walks π_1 and π_2 from X to Y, in Figure 10 with $\alpha_1 = \Lambda_X(\pi_1) = [east, south, west, west, west]$ and $\alpha_2 = \Lambda_X(\pi_2) = [north,\ north,\ east,\ north,\ north,\ east,\ east]$.

The contractions of α_1 and α_2 are $\overline{\alpha_1} = [south,\ west,\ west]$ and $\overline{\alpha_2} = [east,\ east,\ east,\ north,\ north,\ north,\ north]$. The corresponding LW-contractions are $\widehat{\alpha_1} = [east,\ east,\ east,\ south] = \widehat{\alpha_2}$. In this case, $\beta_X(Y) = [east,\ east,\ east,\ south]$.

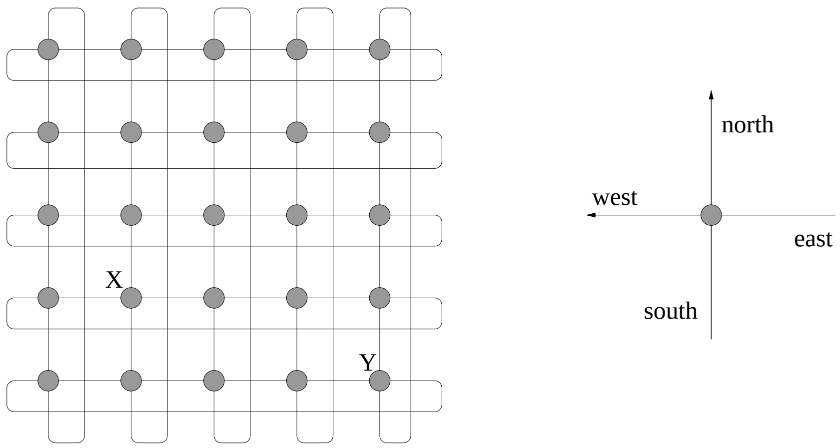

Figure 10: A torus with LW-contracted sense of direction.

The study of the impact of contracted labelings with wraparound is extensive, especially for the ring (see, e.g., [3, 4, 6, 19, 30, 47, 48]).

5.4 Neighboring Sense of Direction

In this Section, we describe a class of labelings, which we will show are very powerful ones.

Definition 16 *Given a graph* (G, λ)*,* λ *is a* Neighboring labeling *iff:* $\forall\langle x, y\rangle \in E[x]$, $\langle z, w\rangle \in E[z]$,

$$\lambda_x(\langle x, y\rangle) = \lambda_z(\langle z, w\rangle) \text{ iff } y = w$$

That is, in a neighboring labeling, all the links ending in the same node x are labeled with the same label, which we shall denote by $l_{(x)}$ (see Figure 11).

Theorem 12 *Let* λ *be a neighboring labeling, and,* $\forall x, y \in V$*, let* $\beta_x(y) = l_{(y)}$*. Then* λ *is a* $\mathcal{SD}$*.*

Proof To verify that it is a $\mathcal{SD}$, consider the coding function f with $\mathcal{N} = \mathcal{L}$, defined as follows: $\forall\pi \in P[x_0],\ \pi = [\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle]$

$$f(\Lambda_x(\pi)) = \lambda_{x_{m-1}}(\langle x_{m-1}, x_m\rangle)$$

Since, by definition of neighboring labeling, $\lambda_{x_{m-1}}(\langle x_{m-1}, x_m\rangle) = l_{(x_m)}$ it follows that

$$f(\Lambda_x(\pi)) = \beta_{x_0}(x_m)$$

Thus, f is consistent.

Consider now the following decoding function:
$\forall\langle x_0, y_0\rangle \in E(x_0),\ \forall\pi \in P[y_0],\ \pi = [\langle y_0, y_1\rangle, \langle y_1, y_2\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$

$$h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) = f(\Lambda_{y_0}(\pi))$$

h is consistent, in fact:

$$h(\lambda_{x_0}(\langle x_0, y_0\rangle), f(\Lambda_{y_0}(\pi))) = \lambda_{y_{m-1}}(\langle y_{m-1}, y_m\rangle) = l_{(y_m)} = \beta_{x_0}(y_m)$$

Thus, λ is a $\mathcal{SD}$. □

We shall call this labeling a *Neighboring* $\mathcal{SD}$.

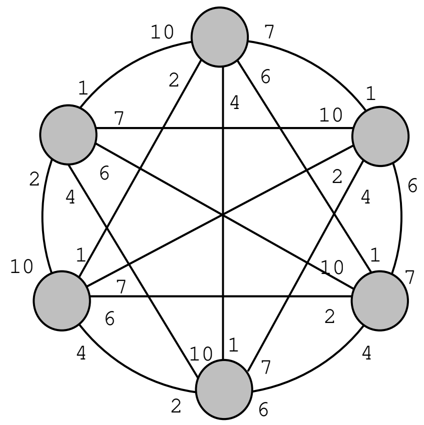

Figure 11: A complete network with neighboring $\mathcal{SD}$.

Let us observe that, unlike all the previous classes of $\mathcal{SD}$s, the neighboring sense of direction is *not* symmetric. This implies that we cannot apply the properties of Section 4.3. Let us recall that, if a labeling is symmetric, knowledge of the origin of a walk has an impact on the translation capabilities of a node outside the walk.

However, we will now show that the neighboring $\mathcal{SD}$ has actually a very strong property. In fact, with such a labeling, the translation capabilities of a node outside the walk are the same as for a symmetric labeling, *even* without knowledge of the origin of the walk.

Given a system (G, λ, β), where λ is a neighboring $\mathcal{SD}$, let $\pi = [\langle y_0, y_1\rangle, \langle y_1, y_2\rangle, \ldots, \langle y_{m-1}, y_m\rangle]$, and let $\alpha = \Lambda_{y_0}(\pi)$. Let node x know the sequence α.

Property 9 *Node x can derive the local names $\beta_x(y_i)$ of all $y_i \in \pi$.*

Proof It trivially follows since $\beta_x(y_i) = l_{(y_i)} = \lambda_{y_{i-1}}(\langle y_{i-1}, y_i\rangle)$. □

The above Property shows an aspect of the *strength* of the neighboring $\mathcal{SD}$, which sets it apart from the other classes of $\mathcal{SD}$s. Another, even more startling proof of this strength is given by the following:

Property 10 *Given an anonymous system* (G, λ, β)*, if* λ *is a neighboring* $\mathcal{SD}$*, then the* Election Problem *is solvable in* G*.*

Proof Let λ be a neighboring $\mathcal{SD}$. Then each node x can acquire a unique global identifier, e.g., by asking an arbitrary neighbor for the label of the link connecting them and assuming such a label as its identifier. In presence of a unique global identifier for each node, the election problem can be solved using any of the existing algorithms. □

To fully appreciate this result, recall that the election problem is *unsolvable* in an anonymous systems with local orientation alone [1], and that similar results do not exist for the other classes of $\mathcal{SD}$s described above.
Note that this strength of the neighboring $\mathcal{SD}$ is also its *weakness.* In fact, exactly because the election problem is unsolvable in anonymous graphs, it follows that *the neighboring* $\mathcal{SD}$ *cannot be constructed in anonymous systems.*

In the literature, the neighboring $\mathcal{SD}$ has been studied solely in systems of unknown topology [27, 45].

6 Conclusions and Open Problems

In this paper, we have provided a formal definition of sense of direction. In particular, we have identified the properties whose presence in a labeling make possible the reduction in communication complexity uncovered by the previous investigations.

Using the formal definition as an observational platform, we have derived previously unknown properties of sense of direction as well as properties implied by having sense of direction in a system. Based on the formal definition, we have identified and defined four general classes of labelings which include all the labelings used in the field.

A major contribution of this paper is to provide researchers with a firm starting point as well as a powerful formal tool. From this point and using this tool, many intriguing questions are now open, all of them of immediate practical relevance. Here are just a few:

What is the nature of the relationship between graph topology and sense of direction?

Which topological properties guarantee the existence of a "minimal" sense of direction (i.e., with smallest number of labels)?

What is the complexity of deciding if a labeling is a sense of direction?

What is the impact of sense of direction on computability ?

Investigations in these directions have already started, providing the first partial results [9, 8, 11, 15, 16].

An interesting problem is to find simpler ways to describe (some of the) classes of $\mathcal{SD}$; a step in this direction has been recently taken by [50]. A more interesting and important problem is to find an equivalent but simpler definition of sense of direction; the one proposed in [50] unfortunately only captures a small subset of classes (as proven in [14]).

Another important research area is the application of sense of direction to distributed problems. We have shown in [13] that all the existing results for general graphs follow as simple applications of the definition or of the derived properties. That is, in arbitrary graphs, the complexity improvements obtained with very specific labelings can be obtained with *any* sense of direction. As for specific topologies, no such a result exists. The open problem is thus to understand how *topology-dependent* distributed algorithms (e.g., election protocols for hypercubes) can be constructed which would be efficient with *any* sense of direction.

Finally, a very interesting research direction now open is the study of the interplay between *Implicit Routing* (e.g, [28]) and sense of direction.

Acknowledgments. We would like to acknowledge the many lively discussions which have forced us to clarify in words what was clear in thoughts. In particular, we would like to thank, for their suggestions and comments: Paolo Boldi, Nancy Norris, Alessandro Roncato, Sebastiano Vigna, and all the participants to SIROCCO'94. We also would like to thank the anonymous referees whose recommendations have contributed to greatly improve the presentation of the results.

References

[1] D. Angluin. Local and global properties in networks of processors. In *Proc. of 12th A.C.M. Symposium on Theory of Computing* (1980), pp. 82–93.

[2] H. Attiya, J. van Leeuwen, N. Santoro, and S. Zaks. Efficient elections in chordal ring networks. *Algorithmica 4* (1989), 437–446.

[3] H. Attiya, M. Snir, and M.K. Warmuth. Computing on an anonymous ring. *Journal of the A.C.M. 35*, 4 (1988), 845–875.

[4] P.W. Beame and H.L. Bodlaender. Distributed computing on transitive networks: the torus. In *Proc. of 6th Symposium on Theoretical Aspects of Computer Science* (1989), pp. 294–303.

[5] J.-C. Bermond, F. Comellas, and D.F. Hsu. Distributed loop computer networks: a survey. *Journal of Parallel and Distributed Computing 24* (1995), 2–10.

[6] H.L. Bodlaender. New lower bound techniques for distributed leader finding and other problems on rings of processors. *Theoretical Computer Science 81* (1991), 237–256.

[7] F.T. Boesch and R. Tindell. Circulants and their connectivity. *Journal of Graph Theory 8* (1984), 487–499.

[8] P. Boldi and S. Vigna. Sense of direction and decision problems for Cayley graphs. *Information Processing Letters*. To appear.

[9] P. Boldi and S. Vigna. On the complexity of deciding sense of direction. In *Proc. of 2nd Colloquium on Structural Information and Communication Complexity* (Olympia, 1995), pp. 39–52.

[10] S. Dobrev and P. Ruzicka. Linear broadcasting and $n \log \log n$ election in unoriented hypercubes. In *Proc. of the 4th International Colloquium on Structural Information and Communication Complexity,* (1997). To appear.

[11] P. Flocchini. Minimal sense of direction in regular networks. *Information Processing Letters 61* (1997), 331–338.

[12] P. Flocchini and B. Mans. Optimal election in labeled hypercubes. *Journal of Parallel and Distributed Computing 33*, 1 (1996), 76–83.

[13] P. Flocchini, B. Mans, and N. Santoro. On the impact of sense of direction on message complexity. *Information Processing Letters 63*, 1 (1997), 23–31.

[14] P. Flocchini, A. Roncato, and N. Santoro. Complete symmetries and minimal sense of direction in labeled graphs. *Discrete Applied Mathematics*. to appear. Preliminary version in *Proc. of 27th SE Conference on Combinatorics, Graph Theory and Computing*; Congressus Numerantium 121, 3-18, 1996.

[15] P. Flocchini, A. Roncato, and N. Santoro. Computing on anonymous networks with sense of direction. Tech. Rep. TR-96-13, School of Computer Science, Carleton University, Ottawa, K1S 5B6 Canada, 1996. (Preliminary version in *Proc. of 3rd Colloquium on Structural Information and Communication Complexity*, 32–46, 1996).

[16] P. Flocchini and N. Santoro. Topological constraints for sense of direction. *International Journal on Foundations of Computer Science*. To appear. Preliminary version in *Proc. of 2nd Colloquium on Structural Information and Communication Complexity*, 27-38, 1995.

[17] R.G. Gallager, P.A. Humblet, and P.M. Spira. A distributed algorithm for minimum spanning tree. *A.C.M. Transactions on Programming Languages and Systems 5*, 1 (1983), 66–77.

[18] Z. Harpaz and S. Zaks. Sense of direction in complete distributed networks. Tech. Rep. TR-453, Department of Computer Science, Technion, Haifa 32000, Israel, April 1987. extended abstract in Proc. of 23rd Allerton Conference on Communication, Control and Computing, October 1985.

[19] A. Israeli and M. Jalfon. Uniform self-stabilizing ring orientation. *Information and Computation 104*, 2 (1993), 175–196.

[20] A. Israeli, E. Kranakis, D. Krizanc, and N. Santoro. Time-message trade-offs for the weak unison problem. In *Conference on Algorithms and Complexity* (1994), Lecture Notes in Computer Science 778, Springer-Verlag, pp. 167–178.

[21] T.Z. Kalamboukis and S.L. Mantzaris. Towards optimal distributed election on chordal rings. *Information Processing Letters 38* (1991), 265–270.

[22] E. Korach, S. Moran, and S. Zaks. Tight lower and upper bounds for a class of distributed algorithms for a complete network of processors. In *Proc. of 3rd Symposium on Principles of Distributed Computing* (Vancouver, 1984), pp. 199–207.

[23] E. Kranakis and D. Krizanc. Labeled versus unlabeled distributed Cayley networks. *Discrete Applied Mathematics 63*, 3 (1995), 223–236.

[24] E. Kranakis and D. Krizanc. Distributed computing on anonymous hypercubes. *Journal of Algorithms 23* (1997), 32–50.

[25] E. Kranakis, D. Krizanc, and J. van den Berg. Computing boolean functions on anonymous networks. In *Proc. of 17th International Colloquium on Automata, Languages, and Programming* (1990), Lecture Notes in Computer Science 443, Springer-Verlag, pp. 254–267.

[26] E. Kranakis and N. Santoro. Distributed computing on anonymous hypercubes with faulty components. In *Proc. of 6th International Workshop of Distributed Algorithms* (1992), Lecture Notes in Computer Science 647, Springer-Verlag, pp. 253–263.

[27] I. Lavallée and G. Roucairol. A fully distributed (minimal) spanning tree algorithm. *Information Processing Letters 23* (Aug 1986), 55–62.

[28] J. van Leeuwen and R. Tan. Compact routing: a survey. In *Proc. of 1st Colloquium on Structural Information and Communication Complexity* (Ottawa, 1994), pp. 99–110.

[29] M.C. Loui, T.A. Matsushita, and D.B. West. Election in complete networks with a sense of direction. *Information Processing Letters 22* (1986), 185–187. see also Information Processing Letters, vol.28, p.327, 1988.

[30] B. Mans. Optimal distributed algorithms in unlabeled tori and chordal rings. *Journal on Parallel and Distributed Computing* (1997). To appear. Preliminary version in *Proc. of the 3rd International Colloquium on Structural Information and Communication Complexity*, 17-31,1996.

[31] B. Mans and N. Santoro. Optimal fault-tolerant leader election in chordal rings. *IEEE Transactions on Computers*. To appear.

[32] B. Mans and N. Santoro. On the impact of sense of direction in arbitrary networks. In *Proc. of 14th International Conference on Distributed Computing Systems* (Poznan, 1994), pp. 258–265.

[33] S.L. Mantzaris. Almost optimal election in chordal rings. In *Proc. of Conference on Parallel and Distributed Computing in Engineering Systems* (1991), pp. 459–464.

[34] G.H. Masapati and H. Ural. Effect of preprocessing on election in a complete network with a sense of direction. In *Proc. of I.E.E.E. International Conference on Systems, Man and Cybernetics* (1991), vol. 3, pp. 1627–1632.

[35] T. Masuzawa, N. Nishikawa, K. Hagihara, and N. Tokura. Optimal fault-tolerant distributed algorithms for election in complete networks with a global sense of direction. In *Proc. of 3rd International Workshop on Distributed Algorithms* (1989), Lecture Notes in Computer Science 392, Springer-Verlag, pp. 171–182.

[36] T. Masuzawa, N. Nishikawa, K. Hagihara, and N. Tokura. A fault-tolerant algorithm for election in complete networks with a sense of direction. *Systems and Computers in Japan 22*, 12 (1991), 11–22.

[37] N. Norris. Classifying anonymous networks: when can two networks compute the same vector-valued functions ? In *Proc. of 1st Colloquium on Structural Information and Communication Complexity* (Ottawa, 1994), pp. 83–98.

[38] Yi Pan. An improved election algorithm in chordal ring networks. *International Journal of Computer Mathematics 40*, 3-4 (1991), 191–200.

[39] Yi Pan. A near-optimal multi-stage distributed algorithm for finding leaders in clustered chordal rings. *Information Sciences 76*, 1-2 (1994), 131–140.

[40] G.L. Peterson. Efficient algorithms for elections in meshes and complete networks. Tech. Rep. TR-140, Dept. of Computer Science, Univ. of Rochester, Rochester, NY-14627, 1985.

[41] S. Robbins and K.A. Robbins. Choosing a leader on a hypercube. In *Proc. of International Conference on Databases, Parallel Architectures and their Applications* (Miami Beach, 1990), pp. 469–471.

[42] N. Santoro. On the message complexity of distributed problems. *Journal of Computing and Information Science 13* (1984), 131–147.

[43] N. Santoro. Sense of direction, topological awareness and communication complexity. *SIGACT NEWS 2*, 16 (Summer 1984), 50–56.

[44] N. Santoro, J. Urrutia, and S. Zaks. Sense of direction and communication complexity in distributed networks. In *Proc. of the 1st International Workshop on Distributed Algorithms* (Ottawa, 1985), Carleton University Press, pp. 123–132.

[45] M.B. Sharma, S.S. Iyengar, and N.K. Mandyam. An efficient distributed depth-first-search algorithm. *Information Processing Letters 32* (September 1989), 183–186. see also Information Processing Letters, vol.35, p.55, 1990.

[46] G. Singh. Efficient leader election using sense of direction. *Distributed Computing 10* (1997), 159–165.

[47] V. Syrotiuk and J. Pachl. A distributed ring orientation algorithm. In *Proc. of the 2nd International Workshop on Distributed Algorithms* (Amsterdam, 1987), Lecture Notes in Computer Science 312, Springer-Verlag, pp. 332–336.

[48] G. Tel. Network orientation. *International Journal of Foundations of Computer Science 5*, 1 (1994), 1–41.

[49] G. Tel. Linear election in hypercubes. *Parallel Processing Letters 5*, 1 (1995), 357–366.

[50] G. Tel. Sense of direction in processor networks. In *Proc. of Conference on Theory and Practice of Informatics* (1995), Lecture Notes in Computer Science 1012, Springer-Verlag, pp. 50–82.

[51] A.M. Verweij. Linear-message election in hypercubes. Manuscript.

[52] M. Yamashita and T. Kameda. Computing on anonymous networks, part I: characterizing the solvable cases. *IEEE Transaction on Parallel and Distributed Computing 7*, 1 (1996), 69–89.