On the Impact of Sense of Direction on Communication Complexity*

(*Revised Version*)

Paola Flocchini† Bernard Mans‡ Nicola Santoro†

Abstract

In this paper, we prove a general result on the impact of Sense of Direction. We show that, in arbitrary graphs, *any* Sense of Direction has a dramatic effect on the communication complexity of several important distributed problems: *Broadcast*, *Depth-First Traversal*, *Election*, and *Spanning Tree Construction*.
In systems with n nodes and e communication links, the solution for the the Depth First Traversal and the Broadcast problems require $\Omega(e)$ messages without labeling; we show that, with *any* Sense of Direction, they can be solved exchanging only $\Theta(n)$ messages, even if the system is *anonymous*. The problems of Election and of Spanning-Tree Construction require $\Omega(e + n \log n)$ messages in absence of labeling; on the other hand, with Sense of Direction we show that they can be solved with $\Theta(n)$ messages
The results presented here completely explain and generalize the existing results which now follow as corollaries for specific labelings.

Key Words Distributed algorithms, Complexity, Graph algorithms.

1 Introduction

A *distributed system* is a collection of processing entities (e.g., entities) 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 *communication topology* of the system can be viewed as a graph $G(V, E)$ where nodes correspond to entities and edges correspond to direct bidirectional communication links between entities. Let $E(x)$ denote the set of edges adjacent to node x.

---

*Research supported in part by N.S.E.R.C, grant #A2415, by a Carleton University Research Achievement Award, by a CNR International Fellowship, and by a Merit Research Allocation (MRA) from James Cook University of North Queensland, Australia.

†Center for Parallel and Distributed Computing, School of Computer Science, Carleton University, Ottawa K1S 5B6 Canada. {flocchin,santoro}@scs.carleton.ca

‡Dept. of Computer Science, James Cook University of North Queensland, Townsville, Qld 4811, Australia. bernard@cs.jcu.edu.au

Every node x has a local label (also called *port number*) associated to each of its incident edges; let $\lambda_x(\langle x, z\rangle)$ denote the label associated by x to $\langle x, z\rangle \in E(x)$. In other words, every link has two labels, one for each of its incident nodes. A classical example is a ring network where each arc is labeled "right" at an incident node and "left" at the other.

Thus the system structure is a labeled graph (G, λ), where $\lambda = \{\lambda_x : x \in V\}$ is the labeling function.

A very important fact is that some labelings λ have properties which can be exploited in the solution of distributed problems.

This fact has been made explicit by the result of [13]: the *chordal* 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 [11]) to $O(n)$ messages, where n is the number of entities in the system (see also [15, 21]).

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 *chordal rings* without labeling, the Election process requires $\Omega(n \log n)$ messages; with the *chordal* labeling there exist algorithms whose complexity depends on the chord structure and can be linear [2, 9, 17]. In these graphs, the properties of this labeling has been efficiently used for the Weak Unison problem [8] and for Fault-Tolerant Election (e.g., [16]).

Similarly, $O(n)$ Election algorithms exist for an *hypercube* with the traditional *dimensional* labeling (e.g., [4, 18]); without labeling, the best known complexity is $O(n \log n)$ which is probably optimal. An even simpler $O(n)$ technique has been found if the hypercube has the *chordal* labeling [4].

In systems of unknown topology (the so-called *arbitrary graph* case), the availability of the *neighboring* labeling reduces the complexity of the Election problem from $\Omega(e + n \log n)$ (for the unlabeled case [1, 19]) to $O(e)$ messages. With the same labeling, the message complexity of *Depth First Traversal* drops from $\Omega(e)$ to $O(n)$ [20]. The availability of a "group" labeling [22] reduces the complexity of the Broadcast problem from $\Omega(e)$ to $\Omega(n)$. The same reductions for the Election, the Broadcast, and Depth First Traversal problems can be obtained also with the simpler *chordal* labeling [5, 14].

All these labelings differ greatly from each other: the "chordal" labeling in chordal rings, the "dimensional" labeling in hypercubes, the "neighboring" and the "group" labeling in arbitrary graphs, etc. What they have in common is that they satisfy the same set of global consistency constraints, called *Sense of Direction* [6]. In other words, all existing results can be seen as evidence of the impact that specific instances of Sense of Direction has on the communication complexity of distributed problems (in specific graphs).

In this paper, we prove a general result on the impact of Sense of Direction. We show that, in arbitrary graphs, *any* Sense of Direction has a dramatic effect on the communication complexity of several important distributed problems: *Broadcasting*, *Depth-First Traversal*, *Election*, and *Spanning Tree Construction*.

In particular, both Broadcasting and Depth First Traversal require $\Omega(e)$ messages without labelings, even if each entity has a unique global identifier. At the same time, we show that with any Sense of Direction it is possible to solve those problems exchanging only $O(n)$ messages; this results holds even if the system is *anonymous* (i.e., the entities do not have any identity).

Both the problems of Election and of Spanning-Tree Construction require $\Omega(e + n \log n)$ messages in absence of labeling. On the other hand, with Sense of Direction we show that we can construct Election protocols which require only $O(n \log n)$ messages. Because of the equivalence between the two problems, this results implies a similar bound for Spanning Tree Construction.

The results presented here completely explain and generalize the results of [12, 14, 20] which now follow as corollaries for specific labelings.

The paper is organized as follows. The framework is described in Section 2 where the basic definitions and properties of $\mathcal{SD}$ are given. We prove lower-bounds on the communication complexity in unlabeled networks in Section 3; the upper-bounds obtainable with sense of direction are described in Section 4.

2 The Framework

2.1 Labelings and Sense of Direction

Let $G(V, E)$ be the graph describing the communication topology of a distributed system, where nodes correspond to entities and edges correspond to direct bidirectional communication links between entities. Let $E(x)$ denote the set of edges adjacent to node x. Every node x has a label (also called *port number*) associated to each of its incident edges; let λ_x be the *local edge-labeling* (or labeling) function for $x \in V$ which associates a label l to each edge $e \in E(x)$: $\lambda_x(\langle x, z\rangle)$ denotes the label associated by x to $\langle x, z\rangle \in E(x)$. Each node x refers to the other nodes using local labels called *names*. The set of local names used by x is called the *Local View* of x; these local names are *not* necessarily identities (i.e., unique global identifiers), in fact, the system could be anonymous. Let β_x be the local *node-labeling (or naming) function* for $x \in V$ which associates a name to each of the other nodes of the system in such a way that: $\forall y, z \in V \;\; \beta_x(y) = \beta_x(z)$ iff $y = z$. let $\beta_x(y)$ be the name associated by x to y.

The entire system can thus be described by the triple (G, λ, β), where the *labeling* $\lambda = \{\lambda_x : x \in V\}$ is the set of local labeling functions, and *naming* function $\beta = \{\beta_x : x \in V\}$ is the set of local naming functions.

Consider a system (G, λ, β). Let $\mathcal{L}$ and $\mathcal{N}$ denote the set of edge and of node labels, respectively; without loss of generality, let $\mathcal{N} \subset \mathcal{L}^*$. The labeling λ is a *local orientation* when each node can distinguish among its incident edges; that is $\forall x \in V$, $\forall e_1, e_2 \in E(x)$, $\lambda_x(e_1) = \lambda_x(e_2)$ iff $e_1 = e_2$. The set of paths from x to y is denoted by $P[x, y]$, and $P[x]$ denotes the set of paths starting from x.

A *consistent coding function* f of a graph (G, λ, β) is a function which maps the sequences of labels $\Lambda_x(\pi)$ associated to any paths π from x to y to the local name $\beta_x(y)$ used by x to refer to y, more precisely, a coding function f is *consistent* in (G, λ, β) iff $\forall x, y \in V$, $\pi \in P[x, y]$, $f(\Lambda_x(\pi)) = \beta_x(y)$.

A *consistent decoding function* h for f is a function which associates a name to a given name and a label, and allows a node to translate the local views of its neighbors. More

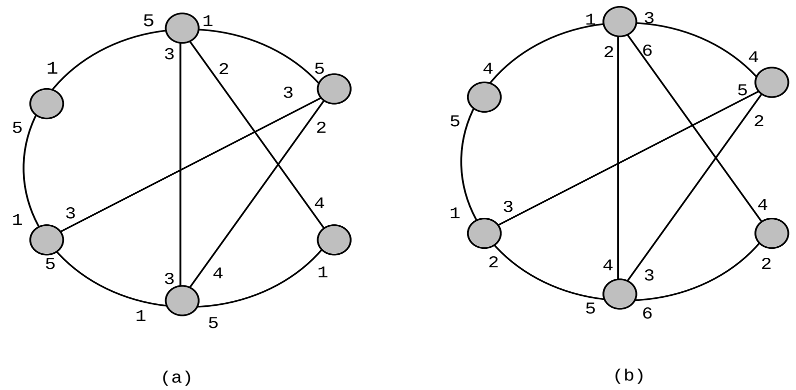

Figure 1: Graph with (a) Chordal and with (b) Neighboring labeling

precisely, given a consistent coding function f, a decoding function h for f is consistent iff $\forall\langle x, y\rangle \in E(x), \pi \in P[y, z]$, $h(\lambda_x(\langle x, y\rangle), f(\Lambda_y(\pi))) = \beta_x(z)$.

Given (G, λ, β), λ is a *Sense of Direction* ($\mathcal{SD}$) iff the following conditions hold [6]:
(1) λ is a Local Orientation,
(2) there exists a consistent coding function f,
(3) there exists a consistent decoding function h for f.

2.2 Instances: Chordal and Neighboring $\mathcal{SD}$

To illustrate some of these concepts, we will describe here the only two types of labeling known (in the literature) to have an impact on communication complexity in arbitrary graphs: *neighboring* and *distance*.

A *chordal* (or *distance*) 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 $\langle x, y\rangle$ of x by the distance $\delta(x, y)$ in the above cycle (see Figure 1).

Let λ be a chordal labeling and $\forall x, y$ let $\beta_x(y) = \delta(x, y)$, where $\delta(x, y)$ is the distance from x to y in the cyclic ordering; note that $\mathcal{N} = \mathcal{L} = \mathbf{Z}_n^+$. This labeling is a Sense of Direction in (G, λ, β) [6]. For example, a consistent coding function f is the following: for any path $\pi = (\langle x_0, x_1\rangle, \langle x_1, x_2\rangle, \ldots, \langle x_{m-1}, x_m\rangle) \in P[x_0]$

$$f(\Lambda_{x_0}(\pi)) = \sum_{i=0}^{m-1} \lambda_{x_i}(\langle x_i, x_{i+1}\rangle) \bmod n$$

It is easy to verify that $f(\Lambda_{x_0}(\pi)) = \delta(x_0, x_m) = \beta_{x_0}(x_m)$, and thus, f is consistent.

The corresponding consistent decoding function is the following: $\forall\langle x_0, y_0\rangle \in E(x_0)$, $\forall\pi \in P[y_0]$

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

$$= \lambda_{x_0}(\langle x_0, y_0\rangle) + \sum_{i=1}^{m-1} \lambda_{y_i}(\langle y_i, y_{i+1}\rangle) \bmod n = \beta_{x_0}(x_m)$$

Sometimes called *Distance* $\mathcal{SD}$, its impact on the Election problem in arbitrary graphs has been studied in [14], where a $O(n \log n)$ solution algorithm has bee shown.

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 1). If $\beta_x(y) = l_{(y)}$ $\forall x, y \in V$, then λ is a $\mathcal{SD}$ in (G, λ, β) [6]. In fact, the coding function $\forall \pi \in P[x_0]$, $\pi = (\langle x_0, x_1 \rangle, \ldots, \langle x_{m-1}, x_m \rangle)$

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

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

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

are consistent.

This type of labeling is available for example in systems where every node has a distinct global identifier (the *identity*) and knows the identities of all its neighbours (hence the name "neighbouring"). Since distinct global identifiers are easily recoverable from a Neighbouring labeling, systems with such labelings are *not* anonymous [6].

It has been shown that, in arbitrary graphs with Neighboring $\mathcal{SD}$, the *Election* problem can be solved with $O(n \log n)$ messages [12], and Depth-First Traversal can be done in $O(n)$ messages [20].

2.3 The Power of Sense of Direction

In Sense of Direction, the existence of a consistent decoding function is clearly the crucial property which allows the nodes to solve global problems while working solely and truly in a local mode. In particular, it provides the ability to *consistently decode* the information received from another node.

As an example, consider the situation of a node x wanting to send to its neighbor z information about a node y. 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 called $\beta_z(y)$. In other words, when a labeling is a *Sense of Direction*, each node can consistently *translate the local views* of its neighbors.

It is by exploiting the existence of this property in the neighboring, the group, and chordal labelings that [12, 20], [22], and [14], respectively, where able to obtain the reduction in complexity in arbitrary graphs. In fact, as we show in this paper, it is possible to obtain the same results with *any* sense of direction.

3 Lower Bounds

3.1 Problems

In this paper we consider a variety of interrelated distributed processes, briefly described below. Each process is a transformation of system configurations; an entity will spontaneously and independently start the process if in state *initiator*.

The *Election* or *Leader Finding* ($\mathcal{L}$) is the process to transform an initial system configuration, where one or more of the entities are in state *initiator* and the others in a different state (say, *asleep*), into a final system configuration where exactly one entity is in a *leader* state and all the other entities are in state *lost*. Notice that there are no a priori restrictions on which entity can become *leader*, nor on the number of *initiators*. The Election problem is *solvable* (i.e., there exists a deterministic protocol which will always correctly solve the problem) if and only if each entity has a unique identity (e.g., [1]). In this case, the Election problem is computationally equivalent to the problems of *Spanning-Tree Construction* ($\mathcal{SPT}$) and *Minimum Finding* ($\mathcal{M}$) with an arbitrary number of initiators (e.g., [19]).

The *Depth-First Traversal* ($\mathcal{DFT}$) problem consists to arrive from an initial configuration where all the entities except one (the *initiator*) are in the same state to a final configuration in which a depth-first spanning-tree of the graph rooted at the initiator has been constructed.

The *Broadcast* ($\mathcal{B}$) problem consists to arrive from an initial configuration where exactly one entity (the *initiator*) has an information, to a configuration where all the entities have the information. The Broadcast problem is a particular instance of the *Wake-up* ($\mathcal{W}$) problem which consists to arrive from an initial configuration where some entities are in the same state (say *awake*) while the others are in a different state (say *asleep*), to a configuration where all the entities are *awake*. Namely, a Broadcast is a Wake-up with only one entity initially *awake*.

3.2 Complexity in Unlabeled Graphs

Given a distributed problem P, let $\mathcal{C}(P)$ denote the message complexity of P in unlabeled graphs.

Lemma 1 *[19] If the entities have unique identities,* $\mathcal{C}(\mathcal{L}) = \mathcal{C}(\mathcal{SPT}) = \mathcal{C}(\mathcal{M})$

Lemma 2 $\mathcal{C}(\mathcal{DFT}) \geq \mathcal{C}(\mathcal{B})$

Proof Trivially, since any traversal performs a broadcast. □

Lemma 3 $\mathcal{C}(\mathcal{L}) \geq \mathcal{C}(\mathcal{B})$

Proof Any Election protocol solves the Wake-up problem: the *initiators* are the initially *awake* entities, and both *leader* and *lost* entities are *awake*; thus, $\mathcal{C}(\mathcal{L}) \geq \mathcal{C}(\mathcal{W})$. Since a broadcast is just a wake-up with only one initiator, we have $\mathcal{C}(\mathcal{W}) \geq \mathcal{C}(\mathcal{B})$. It follows that $\mathcal{C}(\mathcal{L}) \geq \mathcal{C}(\mathcal{B})$. □

Theorem 1 *In arbitrary networks of unknown size* $\mathcal{C}(B) \geq \Omega(|E|)$.

Proof By contradiction. Assume that there is an execution of a broadcast protocol in $G = (V, E)$ such that some node x did not send to nor receive from a neighbour y. Consider the graph $G' = (V', E')$ such that $V' = V \cup \{z\}$, where $z \notin V$, and $E' = E \cup \{\langle x, z\rangle, \langle y, z\rangle\} - \{\langle x, y\rangle\}$ (see Figure 2).

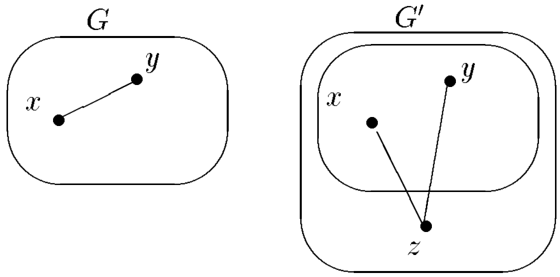

Figure 2: Proof of Theorem 1

.

Since the network size is unknown to any solution protocol by hypothesis, and since x and y did not communicate, the protocol cannot distinguish between G and G'; that is, the same execution is possible in G'. But, in this execution, z will not receive any message, thus the broadcast will never be completed. $\square$

This establishes an $\Omega(e)$ lower bound on the message complexity of $\mathcal{L}, \mathcal{SPT}, \mathcal{M}, \mathcal{DFT}$ problems mentioned above. The proof of the theorem relies on the fact that $|V|$ is not known to the entities.

We now consider the case in which the size of the network is *common knowledge* to all the entities. Given a positive integer n, any integer m such that $n-1 \leq m \leq \frac{n(n-1)}{2}$ is said to be an *edge-value* for n (or, simply, edge-value).

Theorem 2 *For any n and edge-value m, and any problem $P \in \{\mathcal{L}, \mathcal{SPT}, \mathcal{M}, \mathcal{DFT}\}$, there exists a graph $G(V,E)$ on n nodes and e edges, $m \leq e \leq m+n-2$, such that $\mathcal{C}(P)[G] \geq e$ even if n is common knowledge.*

Proof By Lemmas 1,2, and 3,it is sufficient to show that the theorem holds for $P = \mathcal{B}$. Let n be even. Let G' be a k-regular graph on $n' = n-2$ nodes, where $k = \left\lceil \frac{2(f(n)-1)}{n-2} \right\rceil$; clearly, G' has $e' = \frac{k(n-2)}{2}$ edges, Consider now the graph$G$ obtained from G' by adding two new nodes u and v, removing an edge (x,y), and adding the new edges (x,u) and (y,v). The new graph G has n nodes and $e = e'+1 = \frac{k(n-2)}{2} + 1$ links. Notice that, because in the ceiling operator in the definition of k, we have $m \leq e \leq m+n-2$. It is easy to see that in G, broadcasting requires $O(e)$ messages even if n is common knowledge: the two leaf nodes must receive the message, and every internal node has the same degree and is incapable of distinguishing whether it is one of the two nodes connected to a leaf. A similar construction and argument holds when n is odd. $\square$

That is, even if the size of the network is common knowledge, for any given size there exist graphs of almost that size which require $\Omega(|E|)$ messages.

4 Impact of Sense of Direction

4.1 Distributed Depth First Traversal

In this Section, we consider the distributed problem of constructing a Depth First spanning tree of the graph G representing the system.

In unlabeled graphs, this problem requires $\Omega(|E|)$ messages (see Lemma 2) even if the system is not anonymous; this bound can be easily achieved (e.g., [3]). An improvement in the communication complexity has been shown to exists if each entity has a distinct identity *and* knows the identities of all its neighbours; in this case, this problem can be solved with $O(n)$ messages [20]. This result implies that, in presence of *Neighboring* Sense of Direction (discussed in Section 2.2), $\mathcal{DFT}$ can be performed in $O(n)$ messages. Recall that graphs with neighboring labelings are *not* anonymous.

In the following we show that Depth First Traversal can be performed in $O(n)$ messages with *any* sense of direction, even if the graph is *anonymous*.

Algorithm DFT

The algorithm is token-based; at any point in time, at most one node has the "token". Initially, the token is held by the initiator node. During each step of the algorithm, the node holding the "token" transmits it to one of its "untraversed" neighbors; if all the neighbours have been already traversed, the node "backtracks" the token to the node from which the token was first received. The algorithm terminates when the owner of the "token" detects that all the nodes have been traversed. With respect to a DF-traversal of G, the edges of G can be partitioned into "tree edges" and "back edges". To achieve the desired complexity, the "token" must be transmitted only along the tree edges. However, a node does not know a priori which of its *unused* edges (i.e., on which no message has been transmitted or received) is a "tree edge" or a "back edge". The properties of Sense of Direction are used to determine exactly this, without message transmissions on "back edges".

Let (G, λ, β) be a system with Sense of Direction. Given a set $S \subseteq V$ of nodes, let $V_x(S) = \{\beta_x(s) : s \in S\}$ be S as viewed by node x; given a neighbour y of x, let $H_y(S) = \{h(\lambda_y(< y, x >), q) : q \in V_x(S)\}$, where h is the consistent decoding function.

Property 1 *For every $S \subseteq V, x \in V, y \in E(x)$, $H_y(S) = V_y(S)$.*

In other words, if a node x transmits its view (i.e., its local names) of a set S of nodes to a neighbour y, the neighbour is capable of correctly constructing its view of the same set. This implies that, even in absence of distinct identities, if a node sends a set of local names, the receiver node can understand to which nodes the sender is referring to.

Furthermore, $F(x) = \{f(\lambda_x < x, y >) : y \in E(x)\} = \{\beta_x(y) : y \in E(x)\}$, where f is the consistent coding function. Thus, for any $V' \subseteq V_x$ subset of the local view of $x \in V$ it is possible to correctly determine whether or not a neighbour is in V'.

Summarizing, if a node transmits its view of a set of nodes, the receiver is capable of correctly determining whether or not an incident link is connected to a node in that set. The algorithm is shown in Figure 3.

Theorem 3 *In an anonymous system with Sense of Direction, Depth First Traversal can be performed using at most* $2n - 2$ *messages and time.*

Proof Using Property 1, messages in the algorithm described above are transmitted only along the tree edges. In particular, a "token" traverses each tree edge twice; the first time when the edge in "untraversed", the second time in the backtracking phase. □

It is crucial to note that this bound can be reached even in absence of distinct node identities. The algorithm of [20] assumed distinct node identities as well as a priori knowledge at each entity of the identity. Furthermore, the size of the messages in the two algorithms is the same.

4.2 Broadcast

Without labeling, the complexity of the broadcast problem, denoted by $\mathcal{C}(B)$, is $\Omega(|E|)$ when the size is unknown (by Theorem 1). If the size of the network is common knowledge, for any given size there exists graphs which require $\Omega(|E|)$ messages, (by Theorem 2).

When the labeling is a chordal sense of direction, there exists an algorithm for solving the Broadcast problem in $O(n)$ [14] and the same complexity is achievable when the labeling is a group sense of direction [22]. We can now generalize the existing results which hold for specific sense of directions.

Theorem 4 *In an anonymous network with Sense of Direction, the broadcast can be performed using at most* $2n - 2$ *messages.*

Proof Any traversal algorithm solves the broadcast problem; thus, the complexity follows from Theorem 3. □

Notice that the size of the messages used is the same as the one in the algorithms for the specific senses of direction of [14, 22].

4.3 Election and Spanning Tree Construction

Consider now non-anonymous systems; i.e., where each entity has a unique value (or identity). It is well known that in arbitrary unlabeled networks the Election problem requires $\Omega(e + n \log n)$ messages (e.g., [19]) and such a bound is achievable (e.g., [7]).

With Chordal $\mathcal{SD}$ [14] and Neighboring $\mathcal{SD}$ [12, 14], this problem can be solved in $O(n \log n)$ messages. In the following, we prove that the availability of *any* $\mathcal{SD}$ allows to achieve the same bound.

Lemma 4 *[10] The Election problem can be solved using* $(f(n) + n)(\log n + 1)$ *or* $(f(e) + n)$ *messages where* $f(n)$ *is the message complexity of graph traversal .*

Theorem 5 *With Sense of Direction, the Election problem can be solved using* $3\,n \log n + O(n)$ *messages.*

Local information:
•$Unused_x$: list of labels of edges at node x, initialized to $\{\lambda_x(\langle x,y\rangle) : y \in E(x)\ \}$.
•$Tree$: list of local node names which represents the set of nodes reached by now.
•$Parent_x$: label of the edge to the parent node in the rooted tree being constructed.

```
procedure INITIATE
begin
/* entity initiator */
        Tree := {x}
        SEND (TOKEN,Tree) on edge λx(⟨x,y⟩) ∈ Unusedx
        Unusedx := Unusedx − {λx(⟨x,y⟩)}
end INITIATE

Upon RECEIPT of (TOKEN,Tree) on edge labeled λx(⟨x,y⟩) by node x
        Parentx := λx(⟨x,y⟩)
        Unusedx := Unusedx − {λx(⟨x,y⟩)}
        DECODE(Tree, λx(⟨x,y⟩))
        Tree := Tree ∪ {x}
        for each edge λx(⟨x,z⟩) ∈ Unusedx
                if βx(z) ∈ Tree then
                    Unusedx := Unusedx − {λx(⟨x,z⟩)}
        if Unusedx ≠ ∅ then
          SEND (TOKEN,Tree) on an edge labeled λx(⟨x,z⟩) ∈ Unusedx
          Unusedx := Unusedx − {λx(⟨x,z⟩)}
        else /* backtrack */
          if |Tree| = n then
                    Terminate
          else
                    SEND (BACKTRACK,Tree) on edge labeled Parentx
          fi
        fi
end TOKEN

Upon RECEIPT of (BACKTRACK,Tree) on edge labeled λx(⟨x,y⟩) by node x
        DECODE(Tree, λx(⟨x,y⟩))
        for each edge λx(⟨x,z⟩) ∈ Unusedx
          if βx(z) ∈ Tree then
                    Unusedx := Unusedx − {λx(⟨x,z⟩)}
        if Unusedx ≠ ∅ then
          SEND (TOKEN,Tree) on an edge labeled λx(⟨x,z⟩) ∈ Unusedx
          Unusedx := Unusedx − {λx(⟨x,z⟩)}
        else /* backtrack again */
          SEND (BACKTRACK,Tree) on edge labeled Parentx
        fi
end BACKTRACK

procedure DECODE(Tree, λx(⟨x,y⟩))
begin
        foreach r = βy(z) ∈ Tree do r := h(λx(⟨x,y⟩), βy(z))
end DECODE
```

Figure 3: DFT Algorithm

Proof Combining theorem 1 and Lemma 4, where the message complexity of the election algorithm is bounded by $(f(n)+n)(\log n+1)$ and f(n) is the message complexity of traversing the nodes of the network. More specifically, with the algorithm presented above, an algorithm with a $3\,n\,log\,n\ +\ O(n)$ message complexity can be shown.

□

Because of Lemma 1, the same bounds hold also for $\mathcal{SPT}$ and for $\mathcal{M}$. Notice that the size of the messages is the same as in [12, 14].

References

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

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

[3] T.Y. Cheung. Graph traversal techniques and the maximum flow problem in distributed computation. *I.E.E.E. Transactions on Software Engineering*, 9:504–512, 1983.

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

[5] P. Flocchini, B. Mans, and N. Santoro. Distributed traversal and broadcasting in arbitrary network with distance sense of direction. In *Proc. of 9th International Symposium on Computer and Information Sciences*, pages 196–203, 1994.

[6] P. Flocchini, B. Mans, and N. Santoro. Sense of direction: formal definition and properties. In *Proc. of 1st Colloquium on Structural Information and Communication Complexity*, pages 9–34, Ottawa, Canada, 1994.

[7] 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):66–77, 1983.

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

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

[10] E. Korach, Kutten, and S. Moran. A modular technique for the design of efficient distributed leader finding algorithms. *A.C.M. Transactions on Programming Languages and Systems*, 12(1):84–101, January 1990.

[11] 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*, pages 199–207, Vancouver, 1984.

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

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

[14] 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*, pages 258–265, Poznan, 1994.

[15] 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*, volume 3, pages 1627–1632, 1991.

[16] 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):11–22, 1991.

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

[18] S. Robbins and K.A. Robbins. Choosing a leader on a hypercube. In N. Rishe, S. Najathe, and D. Tal, editors, *Proc. of International Conference on Databases, Parallel Architectures and their Applications*, pages 469–471, Miami Beach, 1990.

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

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

[21] G. Singh. Leader election in complete networks. In *Proc. of 11th Symposium on Principles of Distributed Computing*, pages 179–190, 1992.

[22] G. Tel. Sense of direction in processor networks. In Miroslav Bartošek, Jan Staudek, and Jiři Wiedermann, editors, *SOFSEM'95: Theory and Practice of Informatics*, volume 1012 of *Lecture Notes in Computer Science*, pages 50–82. Springer-Verlag, 1995.