High level communication functionalities for wireless sensor networks

Carme Àlvarez, Josep Díaz, Jordi Petit, José Rolim, Maria Serna

A R T I C L E   I N F O

Keywords:
Sensor networks
Directional antennas
MAC protocols
Virtual networks
Mesh of stars
Communication primitives

A B S T R A C T

In this paper we show how to establish a reliable and efficient high level communication system in a randomly deployed network of sensors equipped with directional antennas. This high level communication system enables the programming of the sensor network using high level communication functionalities without the burden of taking care of their physical capacities (low range, unidirectional links, single frequency, presence of collisions, etc.). The high level communication functionalities we offer include point-to-point communication, point-to-area communication, and one-to-all communication. The basic idea to implement this system is to simulate a virtual network that emerges from the ad-hoc network using self-organization, self-discovery and collaborative methods. We also analyse the efficiency, scalability and robustness of the proposed protocols.

1. Introduction

Current development in micro-electrical-mechanical systems and wireless communication have opened a fast development of small, low-power, low-cost sensor devices that integrate sensing, computation and communication. Sensor devices are very small but contain a computing unit, radio, sensors and a power supply, and they are designed to transmit data about location, temperature, vibration, light, sound or airflow. A very large number of these devices can be spread to form wireless sensor networks. Applications of sensor networks include fire or flood detection, disaster relief, smart environments, health control, etc. In recent years, many papers have been devoted to network of sensors, see for example the surveys [4,1].

Wireless sensor networks present a new challenge to the design of communication protocols: Sensor devices are supposed to be deployed in hostile environments, and therefore must use self-organization and collaborative methods to form a network in an ad-hoc way, taking into account that energy efficiency becomes a prime concern.

Radio frequency is the most widespread choice of communication for sensor networks. Ad-hoc networks are typically assumed to be equipped with omnidirectional antennas. However, in networks of sensors, directional antennas may have multiple advantages over omni-directional antennas [3,14,13,17]. In this paper we cope with several models of antennas that have been proposed in the literature: omni-directional antennas, also known as isotropic antennas, directional antennas, orientable directional antennas, directional antennas with unicast and broadcast modes, antennas with unicast, multicast and broadcast modes. Besides the previously cited references the basic characteristics of the different types of antennas are presented in [15,16] for particular applications to sensor networks. Note that directional antennas involve unidirectional received.
Fig. 1. 4 Levels architecture.

links, which makes the design of communication protocols much more complex. Our approach to deal with such different hardware is to present first our protocols for the weakest models and then discuss how to adapt them to stronger models.

In this paper, we consider the general setting of sensor networks in which a large number of sensor devices are scattered on a terrain. We assume that such devices are static. Moreover, to model the whole system, we assume that the terrain fits inside a square and therefore random deployment of sensors on the terrain is equivalent to random deployment on the square. Our goal is to present a way to establish a reliable and efficient high level communication system in a network of sensors. Such a system will enable the programming of sensor networks using high level communication functionalities without the burden of taking care of their physical capacities, including different types of antennas, low range, unidirectional links, use of a single frequency and presence of collisions among others. In this paper we concentrate in point-to-point communication (device $u$ sends a message to device $v$ for arbitrary $u$ and $v$), point-to-area communication (device $u$ sends a message to one or to all devices in some area), and one-to-many (device $u$ sends a message to several other devices).

Because the network of sensors is an ad-hoc and randomly deployed system, such a high level communication system cannot be precomputed and encoded in the devices, it must rather emerge from the system itself. Our design for such a system is made up of four layers (see Fig. 1): On the upper layer lie the applications, which use the high level communication functionalities. These high level communication functionalities are implemented on top of the virtual network that is simulated by the physical network. The configuration of this virtual network involves an initialization phase where the devices perform self-discovery and auto-organization tasks and compute routing and scheduling tables.

The paper is organized as follows. In Section 2, we propose a formal model to study networks of sensors. In Section 3, we present a protocol for initialization, that is, to establish a virtual network. This protocol involves clustering and leader election simultaneously in each cluster, together with the computation of routing and scheduling tables. In Section 4 we describe how to implement several high level communication functionalities. In Section 5 we discuss some applications which can use our functionalities for their own goals. We conclude the paper with a summary of our conclusions and possible extensions.

2. Models

In this section we present the models of sensor networks and antennas that we will consider throughout the paper. We also present some basic results.

2.1. Antenna patterns

Most sensor networks use radio frequency systems in order to communicate. The field of ad-hoc networks usually assumes the use of omnidirectional antennas. In this case, all the devices are equipped with an omnidirectional antenna that can emit in all directions, up to some range. Specifically, any receiver inside a disc of radius $r$ centered at the transmitter position can potentially receive the message. This leads to a geometric undirected graph modelization, where the vertices of the graph correspond to the devices of the network, and where there is an edge $\{u, v\}$ in the graph whenever the distance from $u$ to $v$ is at most $r$.

While omnidirectional antennas offer an easy setting because of the existence of bidirectional communication, it has been pointed out that directional antennas offer higher capacity, require less energy consumption, have less fading area and reduce channel interference [3,14,17]. Real modeling of directional antennas with precise values is a non-trivial task. For the sake of simplicity, the pattern of a directional antenna consists in a main lobe and diverse side lobes. The main lobe is a cone of uniform gain, and the side lobes form together a single “bulb” at the base of the cone (see Fig. 2). For sake of simplicity, we further simplify this model and consider only the main lobe of the antenna, which can be assimilated to a
Several variations of directional antennas have been proposed in the literature. Most of them are combinations of different directional elements that enable different modes of operation. Specifically, in this paper we deal with the following types of antennas, whose behavior are sketched in Fig. 3.

**Omnidirectional antenna:** Devices are equipped with a radio frequency system with an omnidirectional antenna. As a consequence they can emit in a disc of radius $r$.

**Directional antenna:** Devices are equipped with a radio frequency system with a directional antenna that can emit in a sector of $\alpha$ degrees up to a distance $r$.

**Orientable directional antenna:** Devices are equipped with a radio frequency system with a directional antenna that can emit in a sector of $\alpha$ degrees up to a distance $r$. The directional antenna is located in an orientable platform that allows rotations at fixed intervals of $\rho$ degrees, $\rho \leq \alpha$. By moving around the platform the sensor is able to send information to any device in the disk of radius $r$ around it, but not to all of them at the same time step.
![Sector](image)

**Fig. 4.** Sector associated to a device located at \(x\) and angle \(\beta\) with respect to the horizontal axis. Its transceiver can reach the sector \(S = S(x, r, \alpha, \beta)\), centered at \(x\), with angle \(\alpha\) and range \(r\).

### Antenna with unicast and broadcast modes

Devices are equipped with a radio frequency system that combines the capabilities of an omnidirectional antenna and of a directional antenna. In broadcast mode it can emit in a disc of radius \(r\) and in unicast mode it can emit in a sector of \(\alpha\) degrees to a distance \(r\). Modes can be switched at any time with negligible latency but energy consumption in broadcast mode is larger than in directional mode.

### Antenna with unicast, multicast and broadcast modes

Here the devices are equipped with a radio frequency system with an orientable antenna that can emit through any of the disjoint sector subdivisions of \(\alpha\) degrees of the complete disk. We assume that \(2\pi\) is a multiple of \(\alpha\). In all the modes the radius of emission is the same and equal to \(r\). In the unicast mode, only one of the sector subdivisions is selected, in the multicast mode, some of the subdivisions are selected, and in the broadcast mode they cover the entire disk.

Because sensor networks use a single frequency, collisions can arise. Consider that at, some time step, two devices \(u_1\) and \(u_2\) are transmitting information simultaneously. Consider also that some device \(v\) is located in the area of emission of \(u_1\) and \(u_2\). In this case, because of interference, \(v\) cannot decode any of the messages sent by \(u_1\) and \(u_2\). In this paper we assume that a receiving device can differentiate between silence (no message is being emitted over it), reception (a single message is being emitted over it) and noise (two or more messages are being emitted over it). A related line of research is to minimize the number of channels assigned in order to effectively improve performance, which is out of the scope of this paper (see, for example, [12]).

#### 2.2. Random sector graphs

We present a specification of the underlying graph model of the sensor networks that we consider in this paper. Clearly directed antennas are the weakest kind of antennas described in the previous subsection, we concentrate on providing a model for this case. It is straightforward to extend the graph model to any of the other cases. Let us mention that the weakness of the directed antennas translates into a graph property. Assuming the same distribution of devices, the model corresponding to a network using directional antennas is a subgraph of the graph modeling any of the other kind of antennas previously presented.

We model the potential communication of the network of sensors by means of a **random sector graph** introduced in [6]. Let us mention that during the execution of a protocol, at each time step, the real network will correspond to a subgraph of the underlying graph.

In the following, we denote by \(S(x, r, \alpha, \beta)\) the sector centered at position \(x\) with radius \(r\), amplitude \(\alpha\) and elevation \(\beta\), see Fig. 4.

We assume that we have \(n\) devices that are scattered uniformly at random on a square area. For sake of simplicity, we assume that this area is the unit square \([0, 1]^2\). The position of a device \(u\) is represented by \(x[u]\). When a device is deployed, the shift angle between the basis of its sector and the horizontal axis is considered to be uniformly random distributed. We denote by \(\beta[u]\) such an angle for device \(u\). Each device is equipped with a radio frequency system with a directional antenna that can emit in a sector of \(\alpha\) degrees, we assume that the amplitude is a constant, up to a distance \(r_n\). Due to the scaling into the unit square we assume that all the devices have the same transmission range and that this range is a function of the number of nodes. This description leads to a random directed graph modelization:

**Definition 1.** Assume that \(\alpha\) is a fixed angle. Let \((x_i)_{i \geq 1}\) be a sequence of independently and uniformly distributed (i.u.d.) random variables giving the coordinates of points in \([0, 1]^2\), let \((b_i)_{i \geq 1}\) be a sequence of i.u.d. angles and let \((r_i)_{i \geq 1}\) be a sequence of numbers in \([0, 1]\). For any natural \(n\), we write \(X_n = (x_1, \ldots, x_n)\) and \(B_n = (b_1, \ldots, b_n)\). We call \(\mathbb{G}_\alpha(X_n, B_n, r_n)\) the **random sector graph** on \(n\) nodes \(\{1, \ldots, n\}\), where there is an arc from \(i\) to \(j\), if and only if \(x_j \in S_i = S(x_i, r_n, \alpha, \beta_i)\).

In the following, to diminish boundary effects, we focus our attention to **interior devices**, that is, devices whose distance from the boundaries of the terrain is greater than \(r_n\). We also assume that the transmission range, as function of the number of nodes, is of the form

\[ r_n = \sqrt{\frac{a_n}{n}}, \]
where the function \( a_n \) is any selected function that guarantees \( a_n = o(\log n) \) and \( r_n = o(1) \). This selection ensures that the underlying random sector graph is sparse but yet strongly connected when restricted to the interior devices (see [6] for the details).

We consider that devices always communicate in a synchronous way using send and receive primitives that take one unit of time (pulse). Any device \( v \) that falls inside the sector \( S_u \) of a device \( u \) can potentially receive the signals emitted by \( u \). However, due to the use of a single frequency, devices can only receive messages if they get no collision, that is, if only one device is emitting over their coordinates. In terms of graphs, this means that if two devices \( u \) and \( v \) are emitting at the same time and that there is a vertex \( t \) so that \( (u, t) \) and \( (v, t) \) are both arcs in the corresponding sector graph, then \( t \) does not get any message.

We dissect the unit square in a grid of \( \kappa_s \times \kappa_s \) cells, each of size \( s_n = r_n / \sqrt{8} \), so that \( \kappa_n = 1/s_n \). Cells in the grid are numbered from top to bottom and from left to right starting from 0. Two cells \( c_1 \) and \( c_2 \) are called mutually dangerous if the distance between their centers is smaller than or equal to \( 3s_n \). This means that a message originated by some device in \( c_1 \) could potentially be received by some device in \( c_2 \).

This grid serves as reference position for the devices, so that the data they report refers to the cell in the grid that contains them. In this paper we assume that each device knows which cell it belongs to. One could do this by installing a GPS on each device, but this can be considered in some applications to be too expensive in terms of size and power consumption. Several localization algorithms have been proposed, but this problem is far from being completely solved [2]. In any case, we consider the weaker assumption that each device knows an approximation to its coordinates, which from now on will be denoted as its row and column (of the grid).

2.3. Basic results

Let us now present some results that will be useful in the rest of the paper. The first result provides sharp bounds on the number of devices that fall in a cell; its proof is based on the use of Chernoff’s bounds and Boole’s inequality.

**Lemma 1.** Let \( \epsilon > 0 \) and let \( M_+ = M_n = \frac{1}{8}(1 + \epsilon) a_n \) and \( M_- = \frac{1}{8}(1 - \epsilon) a_n \). Then, with high probability, no cell contains more than \( M_+ \) devices or less than \( M_- \) devices, where as stated before \( r_n = \sqrt{\frac{a_n}{n}} \), and \( a_n \) is any function such that \( a_n = o(\log n) \) and \( r_n = o(1) \).

The following result, provable with standard probabilistic tools, is a variation of the birthday paradox:

**Lemma 2.** Consider that we have \( m \) balls thrown to \( m^5 \) bins. Then, the probability that some bin contains two or more balls is \( o(1/m^4) \).

Our final result is about connectivity properties in random sector graphs:

**Lemma 3** ([6]). Consider a random sector graph under the hypotheses stated in this section with \( n \) nodes. Then, with high probability, for any two interior operative devices \( i \) and \( j \) with \( d(x[i], x[j]) \leq r_n \), there is a directed path of length at most 4 from \( i \) to \( j \).

**Fig. 5** sketches the proof given in [6]. The basic idea is that the graph is sufficiently dense to always have some antenna directed to the suitable direction. The directed path of length at most 4 between devices \( i \) and \( j \) is called a simulating path for \( i \) to \( j \).

3. Virtual network establishment

In this section we present an initialization protocol to establish a virtual network on top of the physical network.

**Basic idea.** Our main idea is to establish a virtual mesh of stars network (see **Fig. 6**) in order to provide a high level communication system. In this virtual network, we have a node of the mesh for each cell (the central node). The stars are centered around this node and include all the devices in its cell (we call peripheric the non-central nodes). This network is
virtual in the sense that not all its links do exist in the physical network, they are simulated by the underlying logical system.
Communication on the virtual network is achieved through the repetition of the following three phases: peripheric nodes in the stars send to its central node, central nodes send to their neighbors in the mesh, central nodes send to their peripheric nodes (of course no emission is performed if there is nothing to transmit). As said, because these virtual links may not exist or, in the case they exist, their simultaneous use can lead to collision, efficient and reliable routing and scheduling must be configured by the system. High level communication functionalities are finally defined on top of this virtual network.

**A safe medium access control.** In order to present our initialization algorithm we need a reliable (even if slow) method to guarantee that any device can send information without interferences from others.

Our first task is to assign to each device a unique identifier in its cell. We do so with the following randomized algorithm: Assign to each device $u$ a random integer $\text{rid}[u]$ uniformly selected between $0$ and $M_n^5 - 1$. According to Lemma 2, with high probability, all devices in the same cell will have distinct identifiers.

Our second task is to coordinate the emission of messages between different cells. To do so, let $\text{cx}[u]$ and $\text{cy}[u]$ be the column and the row of the cell to which a device $u$ belongs; then, define $s[u] := 7 \cdot (\text{cx}[u] \mod 7) + \text{cy}[u] \mod 7$. We multiplex time in $a$-phases. Each $a$-phase is made of 49 pulses. During the execution of our initialization protocol, at the $t$th pulse, only devices $u$ satisfying $s[u] = t \mod 49$ will be allowed to send information. Observe that because of the definition of $s[u]$ and the dimensions of the cells, this scheme allows the communication in one cell while avoids the communication in $48 = (3 + 1 + 3)^2 - 1$ dangerous cells in a round robin way. During each $a$-phase, a mote aggregates all the messages in which it is the next intermediate node and forwards this aggregated information.

Since the previous scheme does still allow collisions inside a cell, our third task is to coordinate the emission of messages in a given cell. To do so, we further multiplex time in $b$-phases. Each $b$-phase is made of $M_n^5 a$-phases (and hence of $49M_n^5$ pulses). During the execution of our initialization protocol, at the $t$-th $b$-phase, only devices $u$ satisfying $t \mod M_n^5 = \text{rid}[u]$ will be allowed to send information.

As, by Lemma 1, $M_n$ is an upper bound on the number of devices per cell and identifiers are unique, this time division multiplexing scheme guarantees that only one device can emit a message at the same pulse in a cell, and that no two devices can emit a message at the same pulse in mutually dangerous cells. By the discussion above, we obtain the following.

**Lemma 4.** The above transmission scheduling protocol excludes collisions and allows every device to send a message per round.

Consequently, we can describe now our protocol in terms of these safe primitives, which follow the indicated schedule.

**A protocol for the self-organization of the virtual network.** In order to self-organize the system, we ensure that each device knows the topology of the directed graph restricted to its own cell and its eight neighboring cells. Notice that this information is not that big: it contains $9M_n = \Theta(a_n)$ vertices and $O(a_n^2)$ arcs.

By Lemma 3, there is a path of length 4 between any pair of devices in neighboring cells. Therefore, in order to have a complete knowledge of the directed graph restricted to nine boxes, it suffices to repeat 7 times the broadcasting of all the acquired information, that is, the arcs that $u$ discovers by receiving from $v$. At each emission, the acquired information is increased by the new obtained information. Note that 7 steps suffice: by pointer jumping, after time 1, all devices know the arcs at distance 1, after time 2, all devices know the arcs at distance 2, after time 3, all devices know the arcs at distance 4; then 4 time steps are necessary to ensure this information is broadcast to the rest of the devices in its cell and its neighboring cells. Observe that no two simultaneous emissions can cause collisions using the above mentioned scheduling.

Once each device has the knowledge of the local connections, each cell chooses the device with the lowest identifier as the central node of its corresponding star. As the central node of a cell also knows all the identifiers from its adjacent cells, it can also know their central nodes. Then identifiers of the rest of devices in each cell are ranked. In this way, identifiers will be in the $[0..M_n - 1]$ range rather than in the $[0..M_n^5 - 1]$ range. At this point, to improve communication speed, we redefine the $b$-phases of the communication schedule so that they contain $M_n a$-phases rather than $M_n^5$. 
Next, the system must set the simulating paths. To do so, each device randomly chooses a simulating path of minimal length to communicate with its central node. Central nodes do the same to communicate with the central nodes of their four adjacent boxes. Observe that these paths have at most length 4 and can use devices from neighboring cells.

Each device proceeds now to inform other devices about its path. This is done by a round of 4 b-phases (a total of of $4M_n$ a-phases), started in turn according to the ranked identifiers. Each 4 phases started by a device consist in forwarding its message to the central node. Many devices can hear the messages, but only the unique device that must follow the path will forward it. In this way, the system learns the routing and the scheduling to simulate the virtual links. As a result, we have obtained a way to send information from each non-central node to its central node and from each central node to its four adjacent central nodes. From this point on, we consider the initialization protocol finished and work with the virtual links using the resulting routing and scheduling.

Observe that the initialization protocol we have described needs $7 \cdot 49 \cdot M_n^5 + 4 \cdot M_n$ pulses to finalize. Thus we get the following result.

**Theorem 1.** The initialization protocol we have described needs $\Theta(a_n^5)$ pulses to finalize and the expected number of times a device performs a message emission is 11.

This time is sub-linear with respect to $n$, but possibly not optimal. However, taking into account that this initialization protocol is just executed once, this is not an important concern. In fact, the small number of emissions shows its energetic efficiency.

### 4. High level communication functionalities

In this section we briefly present how the high level communication functionalities can be implemented on top of the virtual network.

We describe communication on the virtual network implicitly assuming that we are using the emulating paths and the emission schedules described in the previous section. Communication on the virtual network is achieved through the repetition of the following three phases:

- **Phase 1:** Peripheral nodes in the stars send to their central node.
- **Phase 2:** Repeat $k$ times: Central nodes send to their four neighbors in the mesh (left, right, up and down in turn).
- **Phase 3:** Central nodes send to their peripheral nodes.

Of course, no emission is done if there is nothing to transmit and the radio receiver can be turned off when the scheduling indicates there is nothing to receive.

Observe that phases 1 and 3 need $\Theta(M_n)$ pulses but phase 2 needs only $\Theta(k)$ pulses. Therefore, the value of the $k$ parameter gives the potential to tune the priority one wants to give between the communication inside cells and the communication between cells. For instance, taking $k = 1$ gives to peripheral nodes the possibility to achieve a high throughput, while long distance communication speed is reduced, whilst taking $k = M_n$ fixes a low throughput to peripheral nodes but speeds up the long distance communication.

**Point-to-point communication.** Point-to-point communication in the network of sensors is easy to achieve on top of the virtual network. We can send a message from device $u$ to device $v$ as follows: First $u$ sends the message to the device in the center of the star of its cell, then this device forwards it to the node in the center of the star of the cell of $u$ using the mesh links, finally this node forwards it to the destination node $v$ using the star.

Forwarding in the mesh is relatively easy, but in the case that congestion could be a matter, a long literature on the problem exists [10]. However, we do not think this would be the case: almost all the devices will be sleeping most of the time. Also, one could increase $k$ to reduce congestion.

Of course, the implementation of this point-to-point functionality must take into account that several different point-to-point transmissions can coexist at the same time in the system (other high level communication functionalities can also coexist). This means that devices must aggregate messages that follow the same direction. Also devices must separate messages that stop following the same direction. In the case that the size of the messages becomes too large, devices will have to queue them (again, we think this would be a rare situation in applications of sensor networks).

**Point-to-area communication.** We foresee that point-to-area communication can exist in two variations: a device wishes to send a message to all the devices in some area or a device wishes to send a message to a single but arbitrary device in some area. Both variations have similar solutions: The message, which contains some header with its destination area is sent from the originator device to its central node. Then, it is forwarded through the mesh to the central nodes or a central node of the reception area, from which it is forwarded to all or one of the peripheral nodes. Since each device has knowledge on the coordinates of its cell, this is easy to do. Observe that one-to-all communication is just the same as point-to-area communication when the area coincides with the full square.

**One-to-many communication.** One-to-many communication is similar to point-to-point communication. However we can gain efficiency when the message can be routed as through common paths. The topology of the virtual network enables an easy implementation for such common routing paths.
5. Applications

One of the most common scenarios for randomly deployed sensor networks involves the continuous monitoring of an extended geographic area at relatively low data rates [1]. The sensors may either be set up to report at regular intervals on some quantity of interest, such as temperature, humidity, or concentration of some chemical substance, or be tuned to respond to some sporadic events of interest occurring within their observation range [9, 8, 11, 1]. The main task in those scenarios is to route all the relevant information to a base station for further processing. The collecting points can be located in a particular position [5], some randomly selected network points [6], or a unique designed node [7]. Our high level communication functionalities provide the means to easily solve the above mentioned problems using them directly, applying standard mesh communication protocols [10] or using other specific protocols for meshes of sensors [7].

Another meaningful application of the virtual network is the detection of damaged areas. Each cell can detect whether some of its components have detected damage. By comparison with neighboring cells the border of a damage area can be easily detected and forwarded to the authority. This will cover applications like fire detection or other environmental applications.

The mesh structure also allows a quick forwarding of an event detection. Using the shortest path to the central authority the task can be performed efficiently, in all the reasonable positioning of the authority.

6. Conclusions

In this paper we have considered how to organize a randomly deployed sensor network in order to offer high level communication functionalities. These functionalities make the programming of such networks much easier, as one concentrates on the logical properties of the devices rather than their rather limited physical properties. In such sense, we have provided a useful middle-ware level between the applications, the hardware and the transmission medium.

Our system is efficient and reliable in the sense that the resulting routing and scheduling allows the devices to turn down their receiver or transmitter when not used. Moreover, the initialization protocol we have described runs in sub-linear time. Our system is also tunable, in the sense that one can adjust the balance between the throughput at which sensors emit messages and the speed these are transmitted through the network.

In the case that the application for which the network is used demands a higher communication between leaders, we can easily implement a cyclic leader replacement that guarantees an equitative amount of energy consumption. To do so we use the ranking computed in the second step of the self-organization protocol. The only additional cost is the computation of the shortest path from each mote to all the other devices in its cell. However this can be computed quickly as all the devices are within distance 4 using only local computation.

We have focused on the weaker model of antenna, but the results in the paper can be easily extended to other models of antenna: the omnidirectional, the orientable—directional, the broadcast—unicast and the unicast—multicast—broadcast.

There still exist some limitations to our system: On the one hand, fault-tolerance is not considered because we perform a self-organization that remains static. We think that periodic re-organizations of the system could cope with these failures. These re-organizations would also help to alleviate the higher power consumption of the central nodes. On the other hand, our model does not capture mobility at all. We leave these issues as future work.

References