Embedded WiSeNts

Information Society Technologies

IST-004400

Paradigms for algorithms and interactions

EW Identifier: EW-D312-DEI-001-03-PARADIGMS

Date: 2005-10-10

Authors: Andrea Zanella[*], Michele Zorzi[*], Elena Fasolo[*]; Anibal Ollero[§], Ivan Maza[§], Antidio Viguria[§]; Marcelo Pias[†], George Coulouris[†]; Chiara Petrioli[♮]

Companies: [*]DEI: Department of Information Engineering, University of Padova, Italy; [§]AICIA: Asociación de Investigación y Cooperación Industrial de Andalucía, Spain; [†]UCAM-CL: University of Cambridge, Computer Lab, UK; [♮]CINI: Consorzio Interuniversitario Nazionale per l'Informatica, Roma

Work package/task: WP 3 / T 3.1.2

Document status: Final

Confidentiality: Public

Keywords: paradigm, interaction, cooperating, algorithm, architecture, wireless sensor network, cooperative robotics, mobile robots, embedded systems, inter vehicular communications, vehicular networks

Abstract: This document provides a survey of the most important design paradigms, algorithms and interaction patterns that characterize the systems based on Cooperating Objects. The aim of the study is to identify the areas that need further research and the most promising design approaches in the context of CO-based systems.

Table of Contents

1. Executive Summary

This document is intended to provide a survey of the algorithms and paradigms for systems based on Cooperating–objects (COs). The aim of the study is to identify the areas that need further research and the most promising design approaches in the context of CO-based systems. The document is structured in three main parts. Part I introduces the subject considered in the document and describes the thematic areas that have been considered in the analysis of the literature. Part II provides an overview of the most common and interesting solutions adopted in the thematic areas previously introduced. Finally, Part III digests the survey of the literature and identifies the critical issues and the most promising design approaches in the context of COs.

Part I.
Introduction

2. Aim of the study

Within the `Embedded WiSeNTs` consortium, a CO has been defined as a collection of:

- *sensors*,
- *controllers*,
- *actuators*,
- *cooperating objects*,

that communicate with each other and are able to achieve, more or less autonomically, a common goal.

Usually, *sensors* are devices capable of communicating information retrieved from the environment to other sensors or, more generally controllers and COs. On the contrary, *actuators* are devices capable of interacting and modifying their environment, in response of appropriate commands. *Controllers* process the information gathered by sensors and issue the appropriate commands to the actuators, in order to interact with their environment. Finally, the definition is somehow recursive, so to indicate that COs can combine their capabilities in a hierarchical way and are, therefore, able to create arbitrarily complex structures.

The generality of this model allows us to seamlessly include different fields like classic embedded systems, wireless sensors networks (WSNs), ubiquitous and pervasive computing systems, an so on. In general, such systems present some commonalities, as the hardware and software modules they are built upon, the use of wireless communication, the dynamic and unpredictable system topology, and so on. On the other hand, the wide number and variety of systems that lie in the COs set, prevent a unified approach to the subject. As a consequence, the research activity has been focused on specific aspects of specific systems, mostly ignoring the aforementioned commonalities.

In this way, however, the research effort risks to be partially wasted, since studies are duplicated in different, though similar, contexts. Even worse, solutions that might be of wide interest in the COs set could remain isolated within the borders of the specific system they were designed for.

Therefore, to boost the diffusion of COs technologies in the near future, it is necessary to change the perspective by considering the COs systems as a single, though variegated, subject. This requires, as a first step, a deep understanding of the current state of the art on the subject that will help to identify and classify the various facets of the topic.

The four studies promoted by the `Embedded WiSeNTs` consortium are aimed at covering this gap, by proving a comprehensive and detailed overview of the scenarios, paradigms, functions and system architectures dealing with Cooperating Objects. Although the four studies are focused on different aspects concerning COs, they are strictly inter–related. The Study 3.1.1, *Applications and application scenarios*, provides an overview of the several different applications scenarios that involve COs. That study will, hence, identify the application characteristics and requirements that have to be fulfilled by the solutions proposed to realize and run the CO systems. Such requirements are, hence, considered in the Study 3.1.2, presented in the following, that deals with the paradigms for the design of algorithms for cooperating objects. The aim of this study is to provide a rather comprehensive overview of the algorithmic solutions proposed for COs systems. This study shall permit to clearly identify commonalities and differences regarding the different design paradigms adopted in each specific system. Furthermore, the study proposes a classification of the algorithms according to the requirements provided by the Study 3.1.1. This classification will permit to assess the matching between the features of an algorithm and the requirements of an application, thus providing a tool to evaluate the suitability of a solution for a specific scenario. The Study 3.1.2 will also emphasize which functions require vertical integration, i.e., message passing among entities of different protocol layers. Such functions are thoroughly described and characterized in Study 3.1.3. Finally, Study 3.1.4 will provide an overview of the mechanisms and strategies that can be adopted to realize in practice the algorithms that are described by this document.

3. Organization of the Study

An exhaustive survey of the literature concerning all the possible systems belonging to the COs family would result an overwhelming task for the limited resources that can be deployed by the `Embedded WiSeNTs` partners. In order to make feasible the study, therefore, the playground has been divided in some specific *Thematic Areas* that have been selected according to the following criteria:

- each thematic area shall be representative for a number of possible application scenarios and, in particular, for the reference scenarios provided by Study 3.1.1;
- thematic areas shall differ for characteristics and requirements;
- thematic areas shall reflect the expertise of the partners involved in the study.

The Study 3.1.1 proposes the following classification of some reference application scenarios:

- *Control and Automation*. The applications that fall into this category may be used in indoor or outdoor environments and they should provide the ability to enable distributed process control with ad hoc and robust networking in challenging environments. They include robotics and artificial intelligence studies.
- *Home and Office Applications*. This class includes the applications intended to provide smart environments through the integration of sensors and actuators that allow users to interact with the surrounding environment.
- *Logistics*. This category includes the applications aimed at supervising the management stocks and supply chains.
- *Transportation*. Applications of this class aim at providing people with more comfortable and safer transportation conditions.
- *Environmental Monitoring*. Environmental monitoring applications may monitor indoor or outdoor environments. Networked microsensors make it possible to obtain localized measurements and detailed information about natural spaces where it is not possible or too expensive to do this through known methods.
- *HealthCare*. Applications in this category are intended to provide health assistance and prevention of diseases.
- *Security and Surveillance*. This class encompasses security and surveillance applications in different environments, without human intervention.
- *Tourism*. Here we include the applications that are designed to provide support to people visiting cities, museums, exhibitions, and so on.
- *Education/Training*. Under this hat we enclose the applications that merge embedded systems into the education methods.

Accordingly, the following four thematic areas have been identified:

1. **Wireless Sensor Networks for Environmental Monitoring** (WSNEM);
2. **Wireless Sensor Networks with Mobile Nodes** (WSNMN);
3. **Autonomous robotics teams for surveillance and monitoring** (ART);
4. **Inter Vehicular Networks** (IVN).

In Part II, we will first define the concepts considered in the study, in order to gain a common understanding of the covered topics. Hence, the different thematic areas will be extensively described in terms of characteristics and requirements. The issues concerning each thematic area will be analyzed together with the most interesting solutions proposed in the literature.

In Part III, we will finally provide a taxonomy of the literature according to the applications requirements identified by Study 3.1.1. From this classification, we will extract the general trends that characterize the design of algorithms for COs systems and we identify the issues that require further investigation in the next future.

Part II.
Survey on Cooperating Objects

The set of characteristics exhibited in CO applications are more diverse than the ones found in applications of traditional wireless and wired networks. Critical factors impact the architectural and protocol design of such applications. These factors also introduce some strict constraints.

The study WP3.1.1 (Applications and Application Scenarios) examined this set of characteristics. Below we briefly review the applications requirements relevant to this particular study.

Network topology: in a CO application, nodes may communicate directly provided they are geographically close to each other. Such a communication can be established in a *single hop* network topology. When nodes are located far from each other, they need to rely on third nodes to forward their data packets requiring, therefore, a *multi-hop* sensor network.

Scalability: the number of COs that may support an application can vary depending on the environment where it is deployed and on its task.

Fault tolerance: it is highly possible that some COs may fail during the operation of the network for various reasons including battery discharge and harsh environmental operation conditions. Fault tolerance is also closely related to security since node failures may be caused by attackers.

Localisation: there exist several CO applications for target tracking and physical event detection including intrusion and forest fire that require node and/or target localisation. GPS may be the natural choice for computing a node's location. These devices, however, do not work in indoor areas and are still of high cost for low-power sensor nodes.

Data traffic characteristics: The amount of data travelling inside the network determines the traffic characteristics of an application. In a particular application, the data transferred among nodes may be limited to a few bytes for simple measurements whereas heavy video-audio traffic may be conveyed in another application scenario.

Networking infrastructure: CO networks can be *infrastructured* or *infrastructureless (ad hoc)*. Even in some applications, the data can be collected by some mobile nodes when passing by the source nodes. This is an important characteristic that determines the type of system approach used (with or without supporting infrastructure) in the majority of the paradigms and algorithms surveyed in this document.

Mobility: in some applications, all physical components of the system may be static whereas in others, the architecture may contain mobile nodes. Applications which can benefit from autonomous robots for actuation may require special assistance for mobility. Adequate support for medium and high mobility in multi-hop networks is still an open issue that should be addressed in the design of future algorithms.

Node heterogeneity: the majority of CO applications include nodes that have distinct hardware and software technical specifications. In a precision agriculture application, for instance, there may exist various types of sensors such as biological and chemical. Energy may be constrained in some of the nodes.

Power Awareness: power consumption is one of the performance metrics and limiting factors almost in any CO application. Systems require prolonged network lifetime. Thus, efficient power consumption strategies must be developed including power-aware communication protocols.

Also, as more complex sensors are designed - for instance in healthcare applications - there is a growing need for tighter control on nodes' resources in order to save energy.

Real-time: the system delay requirements are very stringent in real-time applications. The broad meaning of delay in this context comprises the system data processing and network delay. For instance, in a industrial automation scenario actuation signals are required in real-time. VFs are capable of offering the required functionality to applications through cross-layer system approaches which can significantly reduce the overall system delay. Thus, resource monitoring and system adaptation achieved with cross-layer component-based interactions are important schemes that should be made available to the applications.

Reliability: end-to-end reliability guarantees that the transmitted data is properly received by the receiving-end. In some applications end-to-end reliability may be a dominating performance metric; whereas it may not be important for others. In security and surveillance applications in particular, guaranteed end-to-end delivery is of high importance.

4. Definition of concepts

This section is aimed at providing a common understanding of the different concepts that this study deals with. To this aim, in the following we propose a short definition for each one of the concepts considered.

Thematic Area An exhaustive survey of the literature on CO–based systems would result an overwhelming task for the `Embedded WiSeNTs` partners. Therefore, the playground has been divided in some rather broad *Thematic Areas* that are representative for a number of possible application scenarios and differ for characteristics and requirements. The thematic areas selected by the study are the followings.

- *WSNEM:* Wireless Sensor Networks for Environmental Monitoring are characterized by a large number of stationary sensor nodes, disseminated in a wide area and few sink nodes, designated to collect information from the sensors and act accordingly. Sensor nodes are often inaccessible, battery powered, prone to failure due to energy depletion or crashes. Furthermore, network topology can vary over the time due to the power on/off cycles that nodes go through to save energy.
- *WSNMN:* Wireless Sensor Networks with Mobile Nodes are characterized by the use of mobile nodes. Some of the advantages of using mobile nodes are: less number of nodes

to cover the same area, dynamic adaptation with the environment triggers or changes and dynamic change of the topology to optimize communications in the network. These mobile nodes can appear alone (*mobile sensor networks*) or with static nodes (*static sensor networks with mobile nodes*). There is a special interest case when mobile nodes are autonomous objects (*wireless sensor networks with autonomous mobile nodes*).

- *ART:* Autonomous Robotic Teams. Multiple-robot, or in general multiple-CO systems, can accomplish tasks that no single CO can accomplish, since ultimately a single CO is spatially limited. Autonomous robotic teams are also different from other distributed systems because of their implicit “real-world” environment, which is presumably more difficult to model and reason about than traditional COs of distributed system environments (i.e., computers, databases, networks).
- *IVN:* Inter Vehicular Networks are networks composed by an ad hoc organization of wireless-powered vehicles, each able to transmit or receive data on a highway or in a city street.

Tab. 4 shows how the reference scenarios can be associated to one or more thematic areas. As can been observed, the mapping is not one–to–one, since the scope of the thematic areas is wider than that of the reference scenarios. In fact, the thematic areas have been selected with the purpose of covering as much applications scenarios as as possible, thus giving a rather complete overview of the huge variety of applications that involve cooperating objects.

	WSNEM	WSNMN	ART	IVN
Control and Automation	✓	✓	✓	×
Home and Office Applications	✓	✓	×	×
Logistics	✓	✓	×	×
Transportation	✓	✓	×	✓
Environmental Monitoring	✓	×	×	×
HealthCare	✓	✓	×	×
Security and Surveillance	✓	✓	✓	✓
Tourism	✓	×	×	✓
Education/Training	✓	×	✓	×

Paradigm In this context, the term *paradigm* refers to the methodologies and strategies that can be followed to approach a problem and define the solution.

Algorithm An *algorithm* is the description of a step-by-step procedure for solving a problem or accomplishing some ends. Several different type of algorithms can be defined, according to the purposes they are designed for. In particular, this study deals with the following types of algorithms.

- *Medium Access Control:* MAC algorithms define the mechanisms used by the objects to share a common transmission medium.

- *Routing:* Generally speaking, a routing algorithms provides a mechanism to route the information units (usually data packets) from the source object(s) to the destination object(s).
- *Localization:* Localization algorithms are mechanisms that permit an object to determine its geographical position, either with respect to an absolute reference system or relatively to other objects in the area.
- *Data Processing:* Data processing includes both Data Aggregation & Data Fusion techniques. Data aggregation algorithms are methods to combine data coming from different (and possibly heterogeneous) sources enroute, into an accounting record that can be then forwarded, reducing the number of transmissions, overhead and energy consumption of the system. A possible example is the aggregation of temperature and pressure data produced by two different sensors located in the same area into a single compounded packet that will hence be delivered to an environmental–monitoring station.
  Data fusion algorithms are used to merge together information produced by different sources, in order to reduce redundancy or to provide a more syntectic description of the information. For example, the temperatures measured by several sensors located in a given area can be fused into a single average value for that area.
- *Synchronization:* Synchronization protocols allow nodes (or a subset of nodes that perform a common task) to synchronize their clocks, so that they all have the same time, or are aware of offsets of other nodes. Time synchronization is essential for those numerous applications where events must be time-stamped. Moreover, protocol design is eased when nodes share a common clock (time division techniques for channel access, design of sleep/awake schedules, etc.). Time can be absolute (i.e., referred to an external, well-known measure of time), or nodes can agree a common time reference. This last case is useful in those case when time is needed to compare the occurrence of events.
- *Navigation:* Robots using sensor networks opens a new research area which includes the navigation of the autonomous robots using distributed information as a relevant issue. The navigation of the robot is possible even without carrying any sensor and just using the communications with the wireless sensor network. Furthermore, it should be mentioned that these algorithms can also be used for the guidance of people with a suitable interface .

Interactions The term *Interaction:* refers to the exchange of information among objects that permits the realization of the coordination and cooperation of the objects:

- *Coordination:* is a process that arises within a system when given (either internal or external) resources, they are simultaneously required by several components of this system. In the case of autonomous robotic teams, there are two classic coordination issues to deal with: spatial and temporal coordination.
- *Cooperation:* can be defined as a joint collaborative behavior that is directed toward some goal in which there is a common interest or reward. Furthermore, a definition for *cooperative behavior* could be: given some task specified by a designer, a COs system displays

cooperative behavior if, due to some underlying mechanism (i.e., the "mechanism of co-operation"), there is an increase in the total utility of the system.

Taxonomy *taxonomy* consists in the classification of the concepts according to specific requirements and principles.

5. Wireless Sensor Networks for Environmental Monitoring

Technological advances as well as the advent of 4G communications and of pervasive and ubiquitous computing have fostered a renewed interest on multi-hop (ad hoc) communications. In particular, the interest is in self-organizing wireless multi-hop networks composed of a possibly very large number of nodes. These nodes can be either static or mobile, and are usually constrained in terms of power and computation capabilities.

A typical example of this kind of networks are the *wireless sensor networks* (WSNs) [1, 2]. In this case, the well-known paradigm of ad hoc networking specializes to consider a higher number of nodes (in the thousands and more) that are heavily resource-constrained. Rather than on mobility the emphasis is now on data transport from the sensors to other sensors, or to specific data collection nodes (*sinks*). Sensor nodes are usually irreplaceable, and become unusable after failure or energy depletion. It is thus crucial to devise protocols for topology organization and control, MAC, routing, and so on, that are energy conserving, scalable and able to prolong the overall network longevity, especially in networks with a large number of devices.

5.1. Application Scenarios

The thematic area named **Wireless Sensor Networks for Environmental Monitoring** (*WSNEM*) deals with the deployment of WSNs in static environments. For instance, WSNEM may be used in greenhouses to monitor the environmental conditions, such as air humidity and temperature, light intensity, fertilizer concentration in the soil, and so on. The information is, then, delivered to some central nodes that trace the dynamic of the environmental conditions and determine the actions to perform. The WSNEM may also be used to perform some actions, by activating specific actuators such as watering springs, sliding shatters (to darken the glasshouse), air humidifiers, and so on.

Another possible use of the WSNEM is for monitoring the integrity of buildings, bridges or, more generally, structures. In this case, sensors are displaced on strategic points of the structure and detect any significant variation in the structure form, by revealing variations of pressure, positions of landmark nodes, or relative position of the surrounding sensors. The sensors may periodically read the data and send it to a controller node, either spontaneously on in response to an explicit solicitation of the controller. The controller nodes, hence, may generate an alarm in case a significant variation in the structure is detected.

WSNEM concerns also the realization of the so-called *Ambient Intelligence* or *Smart Environments* (home/market/building/park), where the environment is equipped with sensor nodes that allow interaction with people. For instance, art exhibition area could be equipped with radio nodes and sensors that can detect the proximity of a visitor and provide (upon request or spontaneously) a set of information regarding a particular painting or, maybe, the location of the closest toilettes with facilities for disabled. Similarly, commercial malls equipped with sensors can inform the nearby clients about the on-going commercial promotions and direction to specific stores from the current position of the user. Home networking provides yet another example of a smart space. Different appliances can be interconnected, often wireless, and depending on the personal profile of the users in the home act on environment to provide personalized features such as the ideal room temperature for a specific user, or the redirection of incoming calls to the phone that is closer to the user's current position.

Therefore, the range of scenarios that can be mapped onto this thematic area is rather large. In particular, referring to the sectoral areas proposed by the Study 3.1.1 , the application areas that may be ascribed to WSNEM are the following:

- Environmental Monitoring
    - GoodFood
    - Habitat Monitoring on Great Duck Island
    - Smart Mesh Weather Forecasting
    - Waternet
- Home and Office
    - Smart Surroundings
    - Oxygen
- Logistic
    - CoBIs
    - Smart–dust inventory control
- Security and Surveillance
    - Sustainable Bridges
    - Under water acoustic sensor networks
    - FloodNet
    - Monitoring volcanic eruption with WSNs
    - Cooperative Artefact and Handling of storage of chemicals

5.2. Peculiarities of WSNs

To some extent, a Wireless Sensor Network (WSN) can be consider as a special case of an ad hoc network, for it inherits the characteristics of (quasi) random topology, multi–hop wireless communication, absence of backbone or core structure, distributed control. However, WSNs and ad hoc networks differ in some important features. Some of the specific characteristics are the following.

- The network is composed by a large number of stationary sensor nodes, disseminated in a wide area and few sink nodes, designated to collect information from the sensors and act accordingly.
- After displacement, the sensor nodes are often inaccessible, so that they need to be autonomously powered and controllable by a remote connection.
- In order to save energy, the sensors nodes alternates periods of activity and sleeping, so that the network topology is time variant.

- The connectivity of the network shall be guaranteed also when some nodes are inactive due to sleeping phase, breaking or battery depletion.
- Traffic flows are mostly unidirectional, from sensors to one or more sinks. Sink–to–sensors traffic can be generated in some cases, such as whenever the sensors are explicitly solicited for reading and communicating their data.
- The traffic generated by sensor nodes has usually very low bit rate and it can be continuous and constant (e.g., temperature monitoring) or bursty (e.g. event driven).
- Nodes close to the sinks are required to relay also the traffic generated by peripheral sensors, so that the traffic density increases in the proximity of sink nodes.
- There might be a strong correlation among data generated by sensors that lie in a common region. For instance, more than one sensor node may detect the same event, as the starting of a fire. In this case, it is not necessary that all nodes transmit the same information. Priorities can be defined in order to eliminate redundancy and to reduce traffic burst preferring, for example, that only nodes which are nearest to the source send their information. Therefore, data aggregation can be considered to reduce the energy expense of the network.

Moreover it is worth pointing out that WSNs are usually data–centric in nature, in that data is requested based on certain preferred attributes, which are characteristic to a data query. From this point of view, addressing functions in data–centric sensor networks may be performed by an attribute–value pair, so that if a station is interested in detecting movements that happen at a speed greater than 10 km/h, then it will issue a query that resembles $\{$movement–speed ≥ 10 km/h$\}$, and only the sensors that detect such a movement speed need to report their readings. This is also a way to simplify addressing and save energy.

Therefore, typical performance indexes considered in ad hoc networks, such as QoS, throughput and protocol fairness, might not apply to WSNs. On the other hand, other metrics sauch as the consumed energy per event, the event delivery ratio and so on, acquire a new important role in the WSNs performance evaluation. For this reason, although many protocols designed for ad hoc networks can be adapted to the WSN, a different approach is advisable in order to obtain higher performance.

The existing literature on protocol design for sensor networks highlights many design issues and relevant problems that are to be solved or coped with before efficient communication and management is achieved in WSNs. We recall that a sensor network works under the general concept that the network lifetime needs to be extended as much as possible meanwhile obtaining efficient information forwarding and preventing link disconnections due to node failures. As the primary source of node failure is identified in battery depletion, network connectivity degradation may be prevented by the use of effective energy conservation techniques. On the basis of previous observations, all the network layers should be carefully designed taking into account, in particular, the following issues:

- Medium access control and packet scheduling;
- Routing;

- Sensor data aggregation;
- Node discovering and localization;
- Self–hierarchical organization and/or clustering.

In the literature, all this problems are widely considered, though not all solutions are suitable for the specific scenario of WSN for environmental monitoring. Generally, two different approaches are taken into account: *layered* and *cross–layer*.

The layered approach considers a classical network architecture organized according to the OSI model, where each layer is separately developed and optimized.

In the cross–layer approach, instead, all possible interactions and integrations among layers are considered in order to optimize the entire system. Such an approach appears particularly valuable in the context of WSNs, for the high specialization of this type of solutions. Indeed, in this context, it appears reasonable to sacrifice the flexibility of the layered approach in order to better exploit the peculiarities of the systems at every layer and design more efficient solutions.

In the following, we consider separately the main issues concerning WSNs, taking into account also cross layer solutions, and we underline the protocols, algorithms and proposals that are particularly suitable for environmental monitoring.

5.3. Medium Access Control

The medium access control (MAC) mechanism defines the strategy used by the wireless node to access the common transmission resource, namely the radio channel. Ideally, the MAC mechanism should have the following properties:

- high energy efficiency;
- low access delay;
- support to different access priorities.

In a wireless network there are four fundamental causes of power wasting:

- *Collisions*: a receiver is in the reception range of two or more transmitting nodes and is unable to cleanly receive signal from either node.
- *Overhearing*: nodes sense also transmissions addressed to other nodes
- *Overhead*: nodes have to transmit/receive control traffic.
- *Idle listening*: sensing channel is also performed when the channel is idle (many measurements have shown that idle listening consume $50 - 100$ of the energy required for receiving).

In order to minimize the power consumption, an efficient random access mechanism should reduce all these factors.

Unfortunately, in many cases there is a tradeoff between energy efficiency and access delay, which is determined by many factors, such as power–saving techniques based on active/sleep cycles, medium contention, time synchronization, and so on. In general, however, the access delay is considered as a secondary issue in WSN, with respect to energy saving.

Finally, in several application scenarios for WSNs, access fairness among contending nodes is not a primary issue, due to the high data redundancy.

Many of these aspects are in contrast, so that different MAC algorithms have been defined according to the specific purposes of the network. A first, rough classification of the MAC mechanisms is in two categories:

- *random access* and
- *deterministic access*.

The random access algorithms are based on a medium contention policy and on a carrier sensing mechanism. Such mechanisms achieve good performance in the case of wide and dynamic networks, since they do not require synchronization among the nodes and they make use of local topology information only. Therefore, random access mechanisms gain in flexibility, to the detriment of the energy efficiency.

The deterministic access algorithms are based on a time division mechanism (TDMA) that permits each node to transmit on a different time slot. This approach can potentially achieve very high energy efficiency, since it makes possible to schedule the sleeping and transmission phases of the nodes in an appropriate way and to avoid collisions. On the other hand, deterministic methods are inefficient for high number of nodes, since the time frame required to allocate all the nodes becomes excessively long. Furthermore, heavy signaling is required to maintain the synchronization among the nodes.

In most of the cases, MAC protocols for WSN are a mixture of deterministic and random access.

In literature there are many examples of MAC protocols for WSN. In the following, some of the most important algorithms are briefly presented taking into account their relation with the environmental monitoring applications.

5.3.1. Random Access Protocols

◇ **CSMA** Most of the random medium access protocols are based on the Carrier Sense Multiple Access (CSMA) strategy. The CSMA requires every station to be able to sense the wireless medium before transmitting. If the station reveals energy above a given threshold, then it defers transmission to a later time (that depends on the specific CSMA scheme adopted). The CSMA mechanism, however, is not sufficient to solve the problem of collisions, for the carrier sensing is performed in the vicinity of the transmitter while collisions occur at the receiver. Indeed, the CSMA mechanism rises the so-called *hidden node problem* and *exposed node problem*. The first happens when a node (B) lies in between the transmission range of two other nodes (A and C), which are mutually hidden, i.e., which cannot sense each other transmissions [3]. In this case, collisions may occur at the intermediate node B, since node C will keep sensing an idle channel even during the transmission of node A. Hence, node C may start transmitting when B is still receiving a valid packet from A, causing severe interference on node B. The exposed node problem, instead, occurs when a station (A) is prevented to transmit by the transmission of a nearby station (B) that occupies the channel, even though no interference would be generated to the receiver (C) [3–7]. A graphical representation is given in Fig. 1.

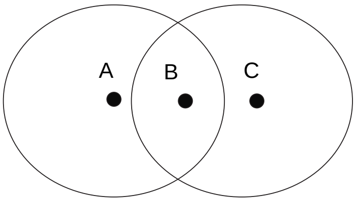


Figure 1: Example of hidden and exposed node scenarios. C is a hidden terminal when it starts transmitting while B is receiving a message from A. A is an exposed terminal when it is kept from starting a transmission cause B is transmitting to C.

To alleviate these problems, a Collision Avoidance (CA) mechanism is often adopted. CA makes use of two special control packets, namely request-to-send (RTS) and clear-to-send (CTS), which are much shorter than usual data packets. Before attempting a data packet transmission, nodes can try to reserve the channel by exchanging a pair of RTS/CTS packets. Clearly, collisions may occur, but they will be limited to the short RTS packet, thus saving time and energy. Nodes that successfully receive either the RTS or the CTS packet are required to refrain from transmitting for the time period declared in the packets header. If the RTS/CTS handshake is successful, all nodes within the coverage range of both transmitter and receiver units will keep silence for the entire duration of the data transmission. Hence, CSMA/CA mechanism helps alleviating the effect of strong interferers on data communications, on the expense of an overhead that is heavier as the data transmission rate increases. Little can be done against *weak* interferers, i.e., nodes that lie in the border of the sensing region [6, 8]. Such nodes, indeed, will have a weak probability to receive any RTS/CTS or packet header transmission, being outside of the reception region. Thus, they may transmit without any limitation, causing interference on the target nodes.

> HIGHLIGHTS
> The CSMA mechanism pursues energy saving by reducing collision probability. On the other hand, the mechanism requires idle listening to the channel and does not alleviate the problem of overhearing. The mechanism is completely distributed, robust against topology variations, and it provides low latency in case of light traffic.

◇ **MACA** *MACA* (Medium Access with Collision Avoidance) [9] is one of the first MAC protocol designed for Wireless LAN. The MACA protocol is largely based on the CSMA/CA mechanism, from which it inherits the CA strategy (based on the RTS/CTS handshake), while dropping the Carrier Sense feature (hence the acronym MA/CA). The sender first transmits a RTS packet to the intended receiver for eliciting the transmission of a CTS reply. Nodes overhearing the RTS packet do not occupy the channel during the immediate successive time period, in order to do not interfere with the reception of the CTS packet returned by the intended receiver. A specific field in the CTS packet carries the expected duration of the pending transmission. Therefore, nodes overhearing the CTS packet retain from transmission for such a time period, thus avoiding interference with the receiver. Since nodes do not perform carrier sensing, collisions are likely to occur over RTS packets. If the sender does not receive a valid CTS packet within a prefixed time after the RTS transmission, it will schedule a new transmission attempt after a random time delay, picked up in the contention window $[1, CW]$. After every failed retransmission, the contention window size, CW, is exponentially increased whereas after every success it is reset to the initial value. The backoff strategy is operated over a slotted–time domain, where each slot corresponds to the transmission time of an RTS or CTS packet (of approximately 30 bytes). Notice that, MACA does not encompass any acknowledgment (ACK) at the MAC layer, so that erroneous data packets have to be retransmitted by the upper layers.

> HIGHLIGHTS
> The MACA algorithm largely inherits the pros and cons of CSMA. The RTS/CTS handshake, though, helps alleviating the energy consumption due to collisions and overhearing.

◇ **MACAW** *MACAW* [10] is largely inspired to MACA protocol that is modified in order to alleviate some inefficiencies. In particular, the authors of [10] observe that the binary exponential backoff (BEB) strategy adopted in MACA in case of collisions may lead to an unfairness probability of medium access in the presence of heavily loaded nodes. Indeed, nodes that undergo a series of collisions, thus exponentially increasing their contention windows, have less probability of winning the contention with other, less-backed-off terminals. To alleviate this problem, in MACAW all nodes in neighborhood have the same contention window size. The contention window size is, indeed, embedded in the header of each packet transmitted over the channel. Whenever a station hears a packet, it copies that value into its own backoff counter. Furthermore, the backoff window is adjusted in a gentler way: upon a collision, the backoff interval is increased by multiplicative factor (1.5), while upon success it is linearly decreased.

In order to further improve the fairness of the channel access, MACAW suggests also the use of the multi–stream model. Basically, authors propose to keep, in each station, separate queues for each stream and, then, to run the access mechanism independently for each queue. Collisions that could potentially occur among the traffic flows of a same terminal, will be internally resolved by the terminal by choosing a winner and backing off the other colliding streams. Notice that the multi–stream model is also considered in the standard $IEEE\ 802.11e$, which aims at providing different access classes.

Finally, MACAW encompasses the use of link layer acknowledgment (ACK) packet. This permits to run an Automatic Retransmission Query (ARQ) algorithm at the MAC layer, thus increasing the reliability of the link layer transport service and reduce the inefficiency due to higher layer retransmissions. Therefore, MACAW introduces the RTS–CTS–Data–ACK access paradigm.

> HIGHLIGHTS
> MACAW has been originally proposed for ad hoc networks. It is based on CSMA/CA and it aims at providing access fairness among users, which might be self-defeating in WSN, as previously discussed.

◇ **PAMAS** *Power Aware Multi–Access* (PAMAS) [11] is a protocol for ad hoc network based on MACAW with the addition of a separate signaling channel with the aim of reducing power consumption.

The RTS/CTS message exchange takes place over the signaling channel that is completely separate from the data channel, used for data–packet transmissions. If the RTS/CTS handshake is successful, the data transmission begins in the data channel. Simultaneously, the receiving station transmits a Busy Tone over the signaling channel to inhibit any other transmission in its neighborhood. The length of the Busy Tone is greater than twice the length of a CTS, therefore, neighbors that transmit an RTS while reception is still occurring will not receive back the expected CTS packets.

The mechanism is also used to reduce the power consumption of the terminals that are not actively involved in communication or are blocked by nearby transmitting stations. To this end, if at least one neighbor of a node is transmitting (data channel is busy) or receiving (busy tone is activated), the node can turn off its radio because it cannot receive or transmit packets. Generally, after overhearing a CTS or RTS transmission, a node enters the sleep mode for the time period signaled by the control packets or the Busy Tone. However, a node that wakes up after a long sleeping period may hear an ongoing transmission without knowing its duration. In this case, a sort of binary search to determine the end of the current transmission is performed by sending short question messages over the control channel that are replayed by the transmitting nodes with longer residual transmission time.

> HIGHLIGHTS
> PAMAS is still a CSMA–based mechanism. It aims at enhancing the energy efficiency by reducing the overhearing of nodes. However, this is accomplished by means of a second radio transceiver, which might increase the complexity and cost of the devices.

◇ **SMAC** *Sensor MAC* (SMAC) [12] is a MAC protocol designed for saving energy in sensor networks. It is based on the assumption that sensor networks are often deployed in an ad hoc fashion with nodes alternating long periods of inactivity and short, aperiodic and (sometime) unpredictable periods of rush activity when an event is detected.

SMAC inherits from PAMAS the powering–off strategy during transmissions of other nodes but, unlike PAMAS, it does not require a separate signaling channel. In order to reduce energy consumption and message delivery latency, SMAC resorts to three techniques:

1. periodic listen and sleep periods;
2. collision and overhearing avoidance;
3. message passing.

The basic scheme for *periodic listen and sleep* is simple. Each node sleeps for some time and wakes up periodically to listen for any transmission request by neighboring nodes. The time duration of the sleeping and listening periods can be selected according to the constraints of the specific application scenario considered. All nodes are free to choose their own listen/sleep schedule. However, control overhead is reduced by forcing neighboring nodes to synchronize on a same schedule. The schedule is, hence, broadcasted to the immediate neighborhood so that each node can maintain a table with the information about the waking–up periods of all its known neighbors. Such a synchronization is required to permit the communication between in–range nodes. If a node receives a schedule that differs from its own, the node merges the two schedules (this event should be rather unlikely). Therefore, neighboring nodes form a sort of *virtual communication cluster*.

The *collision and overhearing avoidance* is obtained by adopting a RTS/CTS–like mechanism, including both virtual and physical carrier sense. In particular, SMAC protocol tries to reduce the energy cost caused by the reception of packets directed to other nodes, by powering off the interfering nodes, i.e., the nodes that overhear an RTS or CTS packet. Interfering nodes are, indeed, all immediate neighbors of either the sender or the receiver node, so that they are not allowed to occupy the radio channel until the ongoing transmission is over. Therefore, interfering nodes can set the so-called Network Allocation Vector (NAV) with the duration of the transmission and sleep for the entire NAV period.

Message passing is aimed at ensuring an application–level fairness instead of a per–node fairness. It is largely inspired to the IEEE 802.11 fragmentation mode that permits to transmit a burst of packets after a single, successful RTS/CTS handshake. In IEEE 802.11, however, the RTS and CTS only reserves the medium for the first data fragment and the corresponding ACK. The first data fragment and ACK, then, reserve the medium for the next fragment and so on. When a node receives a fragment or an ACK, it knows that there is one more fragment to be sent, but it cannot know the total number of still pending fragments. Furthermore, if an ACK packet is not received by the sender, it releases the channel and re-enters the contention phase. This mechanism was intended to provide fairness among the stations.

On the contrary, in SMAC protocol, RTS and CTS reserve the medium for all data fragments, so that a interfering unit can sleep for the entire message transmission. Moreover, if the owner of the channel fails to receive an ACK, it extends the reserved transmission time for one more fragment, by adjusting accordingly the duration field in the header of the following fragments and ACKs. Hence, the node that wins the contention for the channel access can use the medium until it has transmitted all its fragments. Notice that, neighboring nodes that wake up or join the network while the burst transmission is still ongoing, can set their NAV by overhearing the duration field in the header of the first fragment or ACK packet that they see.

In this way, the per–node fairness is not anymore guaranteed. However, in wireless sensor networks it is more important to provide application–level fairness.

> HIGHLIGHTS
>
> The major SMAC protocol advantage is the reduction of the energy waste obtained by the scheduling of the sleep periods. On the other hand, this time scheduling may result in high latency and it is not efficient under variable traffic load. Furthermore, SMAC is a first attempt of cross layer optimization, in that it considers both aspects of physical and application layers.

◇ **Sift** As previously remarked, nodes in sensor networks often encounter spatially-correlated contention, where multiple nodes in the same neighborhood sense the same event. In other words, multiple sensors sharing the wireless medium may all have messages to send at almost the same time in response to a common external event. In the case of environmental monitoring, it would be sufficient that just some of such nodes report the event to the controller (sink) node.

Therefore, in event–driven sensor networks, the main objective is to reduce the latency of the first event reports, rather than maximizing the network throughput.

Sift [13] is a medium access control based on a randomize CSMA which takes into account previous observations. Unlike other CSMA protocols as IEEE 802.11, it does use a fixed–size contention windows from which a node randomly chooses a transmission slot. To reduce collisions, however, nodes try to estimate the number of contending neighbors and adapt accordingly the transmission probability in each slot within the contention window.

Suppose we have a sensor network run by Sift. Every node with something to transmit competes for any slot $r \in [1, CW]$ (with $CW = 32$) based on a shared *belief* of the current population size N. The estimation of N is adjusted after each slot in which no transmission has occurred. For instance, if no node transmits in the first slot, then the estimated number of competing node is reduced, while the transmission probability for the second slot is increased, and so on for every silent slot.

In [13] results show that Sift outperforms IEEE 802.11 in terms of latency. One of the disadvantages of Sift is that it does not take into account any energy saving technique expect for collision reduction.

> HIGHLIGHTS
> Sift is based on CSMA mechanism and aims at reducing the collision probability while limiting the access delay. To this aim, the nodes determine the contention backoff on the basis of the estimation of the node density. This makes the algorithm topology–dependent, though adaptable to topology variations. No explicit energy saving mechanisms are considered.

◇ **STEM** STEM [14] is an acronym for *Sparse Topology and Energy Management*.

It is run once at the beginning of network operations to select, for each node, a sensor which is endorsed the role of *target*. This node is of paramount importance from a forwarding perspective, since all data and control messages that a node needs to send must be directed to its own target node.

In a first version of this protocol, namely STEM–B (where B stands for *Beacon*), nodes wishing to initiate data transmission, i.e., to inject a new packet into the network, explicitly send beacon or *wakeup* messages containing the *target* address until the target wakes up and is able to answer the message. When this happens, a link between the initiator node and its target node is established, so that the data packet may be forwarded from the initiator to the target. If the packet needs to be relayed further, the target node will start sending beacon messages to its own target node and so on until the final destination is reached.

Note that aggressive wake-up messages transmission is needed to avoid excess latency in link establishment, but, on the other hand, a persistent wake-up message sending may disrupt data transmission efforts, since it would generate collisions at receiving nodes with high probability. So as to cope with both problems, the authors in [14] suggest to make use of two radios operating on two separate frequencies, one for data transmission and the other one for control signaling.

Another version of the STEM protocol, namely STEM–T (where T stands for *Tone*) is devised in [14]. In this case, nodes use a tone-based signaling to wake up their target node. Since the wake-up process is initiated by the simple overhearing of the tone, all sleeping nodes in the neighborhood of the sender leave their sleeping status even if they are not important for the packet relaying phase. This type of signaling has the advantage of waking up the target sensor in an on-line manner, thus reducing the mean latency affecting data packet transmissions, but it also raises the overall network energy consumption, as it forces more nodes to wake up than are really useful.

> HIGHLIGHTS
> To some extent, STEM is a cluster–based algorithm in that each node selects a target node that is in charge for data delivering. STEM incurs in high access delay since nodes have to wait for the target node waking up. To alleviate this problem, a two–radio solution has been proposed that, however, worsens the energy efficiency of the network since a transmission wakes up all the overhearing nodes.

◇ **DB–MAC** The primary objective of *Delay Bounded Medium Access Control* (DB–MAC) [15] is to minimize the latency for delay bounded applications also considering the power consumption reduction by means of a path aggregation mechanism.
DB–MAC adopts a CSMA/CA contention scheme based on a four way RTS/CTS/DATA/ACK handshaking. It is defined for a scenario in which different sources contemporarily sense an event and they have to send their information to the sink node. Generated data flows can be dynamically aggregated in the path towards the sink giving rise to an aggregation tree. Intermediate nodes in the path may aggregate several flows into a single flow to reduce transmissions and amount of data to be sent. Also in the case in which the aggregation does not reduce the effective amount of data to be transmitted, it can reduce the overall transmission overhead (for example the contention overhead is reduced).
The MAC protocol scheme is very similar to the IEEE 802.11 RTS/CTS Access with some modifications: RTS/CTS messages are exploited to perform data aggregation and the backoff intervals are computed by taking into account the priority assigned to different transmissions. In particular each node takes advantage of transmissions from other nodes by overhearing CTSs in order to facilitate data aggregation and it gains access to the medium with higher probability if it is close to the source. This policy brings to choose the relay among such nodes that already have some packets to be transmitted.

> HIGHLIGHTS
> DB–MAC, based on CSMA/CA principle, is a cross layer solution that also encompasses data aggregation functionalities to reduce traffic and energy consumption. The algorithm, however, requires a rather complicated data processing.

5.3.2. Deterministic Access protocols

Below there are two examples of MAC protocols based on a TDMA approach rather than on random access.

◇ **Energy-aware TDMA-Based MAC** [16]The main objective of *Energy-aware TDMA-Based MAC* protocol is to extend the lifetime of the sensor through topology adjustment, energy aware routing

and MAC. Message traffic between sensors is arbitrated in time to avoid collisions and to allow turning off the unneeded sensors. The network is organized in such a way that some gateway nodes assume responsibility in their cluster for sensor organization and routing/MAC management.
This protocol is based on a time division multiple access (TDMA) whose slot assignment is managed by the gateway. It informs each node about the slots that the node can use for its transmission. This type of mechanism requires clock synchronization among all nodes within the same cluster but collisions are completely avoided if each node correctly receives its slot assignment.
The protocol consists of four main phases: data transfer, refresh, event-triggered rerouting and refresh-based rerouting phase. In refresh phase every node uses its pre assigned slots to inform the gateway about its state (energy level, state, position). Taking into account this information, the gateway decides how to allocate cluster resources, which nodes should be used as relay and so on and on the basis of some cost function assigns the time slots for the transmissions. In particular rerouting is performed when the sensor energy drops below a certain threshold, after receiving a status update from the sensors and when there is a change in sensor organization.

> HIGHLIGHTS
> Energy-aware TDMA-Based MAC aims at reducing the energy cost of nodes by superimposing a cluster structure on the network, where clusterheads are in charge for managing the channel access of the other nodes in the cluster. The algorithm is a cross–layer solution, in that it provides both MAC and routing functionalities. It suffers of typical TDMA drawbacks, such as need for nodes synchronization and tight cooperation, topology dependency and so on.

⋄ **TRAMA** The *TRraffic–Adaptive Medium Access* protocol (TRAMA) [17], a TDMA–based algorithm, is introduced for energy–efficient collision–free channel access in wireless sensor networks. TRAMA reduces energy consumption by ensuring that unicast, multicast, and broadcast transmissions have no collisions, and by allowing nodes to switch to a low-power, idle state whenever they are not transmitting or receiving. It is similar to NAMA (Node Activation Multiple Access) where for each time slot a distributed election algorithm is used to select one transmitter within two–hop neighborhood. This kind of election eliminates the hidden terminal problem and hence, ensures all nodes in the one-hop neighborhood of the transmitter will receive data without collision. However, NAMA is not energy efficient and incurs overhearing.

TRAMA assumes that time is slotted and uses a distributed election scheme based on information about the traffic at each node to determine which node can transmit at a particular time slot. TRAMA avoids the assignment of time slots to nodes with no traffic to send, and also allows nodes to determine when they can become idle and not listen to the channel using traffic information. TRAMA is shown to be fair and correct, in that no idle node is an intended receiver and no receiver suffers collisions. The performance of TRAMA is evaluated through extensive simulations [17]. Delays are found to be higher compared to contention-based protocols due to higher percentage of sleep times. On the other hand, the advantages of TRAMA are the higher percentage of sleep time compared to contention-based protocols and a less collision probability achieved especially in broadcast type of communication.

> HIGHLIGHTS
> TRAMA is a completely distributed mechanism, where the slot assignment is decided by nodes on the basis of the traffic at each node. TRAMA outperforms contention-based and scheduling-based protocols with significant energy savings.

◇ **TSMA** TSMA (Time Spreading Multiple Access) [18, 19] is a robust scheduling protocol which is unique in providing a topology transparent solution to scheduled access in multi-hop mobile radio networks using a TDMA fashion. The classical requirement that the schedule guarantees a collision free transmission in every slot is not a necessary condition. Instead, in TSMA is considered the following requirement: for each node and for each one of its neighbors, there is at least one time-slot in the frame, in which the packet can be received successfully by the neighbor. In order to satisfy this requirement, TSMA mechanism proceeds in the following way. It assigns each node one or more slots in a frame, so that while collisions at some of the neighboring nodes may occur in every slot, requirement is satisfied by the end of the frame. To obtain such a solution slot assignment is performed by using of the mathematical properties of finite (Galois) fields and it is based on an estimation of the mean neighbor number of a node. TSMA adds the main advantages of random access protocols to scheduled access. Similarly to random access it is robust in presence of mobile nodes. Unlike random access, however, it does not suffer from inherent instability, and performance deterioration due to packet collisions. Unlike current scheduled access protocols, the transmission schedules of TSMS solution are independent of topology changes, and channel access is inherently fair and traffic adaptive.

> HIGHLIGHTS
> TSMA is a TDMA–based algorithms which is topology independent and traffic adaptive. It make use of the mathematical properties of finite fields to assign each node a set of slots.

5.4. Routing and Forwarding Algorithms

Sensor nodes are known to possess very low data processing and storage capabilities. From such a perspective, the design of algorithms and protocols that are supposed to let nodes forward packets towards a final destination should obey to the following rules:

- not require complex calculations;
- rely on easy–to–compute and easy–to–store metrics;
- be easy to program;
- be designed with attention to energy–conservation techniques like the employing of on/off duty cycles.

The characteristics listed above are essential to manage efficiently the scarce memory capabilities and to avoid overwhelming the sensor node's micro–controller with too many operations, because this would lead to a fast exhausting of the available energy. Moreover, it may turn out to be difficult to align with the ISO–OSI model when designing protocols for Wireless Sensor Networks: it is easier (and sometimes more affordable) to think of the MAC and Routing layers as a monolithic entity, so that there is no need for explicit communication between the two layers, and information can be easily shared.

Following [20], we summarize here some general WSN features which are important for routing protocol design:

1. Nodes are typically deployed in very large networks. This makes the use of IP–like global addressing schemes infeasible, as maintaining a global identification may result in high energy expenditure. Further, getting the data transmitted by the sensors is sometimes more important than getting the exact ID of the node that sent the data. This is particularly of interest as sensors may not always be supervised by a centralized attending unity and need to organize themselves in an ad hoc manner, forming connections and coping with the subsequent nodes distribution.

2. Almost all applications of sensor networks involve data communications from multiple sources to a particular base station called the *sink*, even if other kinds of traffic are indeed supportable.

3. Sensor nodes are constrained in terms of energy, processing capabilities and storage capacity, as described before, and therefore need wasteless resource management.

4. Sensors in most applications are supposed to be stationary, which enormously simplifies the prediction of routing protocols' behaviors, but in some cases mobility could be allowed, even for a small amount of nodes inside the network. Routing protocols need to be aware of node mobility to properly engage forwarding decisions. We stress that if an alternation between a low power and an operating state is independently implemented at each node, then sender nodes could find a different network topology each time they want to transmit a packet, so that issues related to duty cycle and mobility management are expected to converge.

5. Sensor networks are thought as application–specific, as different applications require different configurations to meet specialized constraints. For instance, low latency requirements of tactical surveillance are different from long network life and connectivity requirements of a periodic temperature sensing system.

6. Data collection is performed on a location basis in many applications. Position awareness is thus a major issue in sensor networks, as it is difficult to obtain with hight precision. Typical approaches include the use of GPS hardware and the estimation of a node's location by measuring incoming signal strength.

7. Many data collection tasks, in particular during environmental monitoring, are originated by common phenomena, so that the amount of sensed data may be redundant. This may generate a large amount of redundant packets which could be processed by data aggregation or data fusion methods to decrease the total number of data sent thereby enhancing energy efficiency.

The main design problems incurred in creating routing protocols for use in WSNs may be summarized as follows.

Random Topology. First, nodes are deployed to form a network following a usually random displacement. This is not always true, but is indeed the most frequent situation (manual deployment is anyway more affordable for some applications: this simplifies the routing problems as fixed data forwarding paths may be preemptively setup at the beginning of network operations). With random deployment, nodes are required to form an ad hoc infrastructure in an unattended fashion. When the node position distribution is non uniform, clustering may turn out to be a valid choice to reduce the overall energy consumption in data communication. Moreover, as networks are large in size, multihop communications (i.e., communications formed by many independent transmission–reception phases that packets undergo on their way to the final destination) are to be employed and will require an efficient way to manage them.

Wireless communication. A second problem is tightly bound to the various radio communication methods. In a wireless sensor network, nodes are linked by a wireless medium that is used at typically low speeds, on the order of 1–100 kbps. The traditional problems associated with a radio channel (e.g., fading, high error rate, hidden terminals, collisions) may affect the general operations of a WSN. To this extent, routing protocols for WSNs should be appropriately designed to cope with radio impairments.

Nodes mobility. Third, a WSNs is often assumed to be formed of fixed nodes. However, in many applications both the nodes or the sinks may move [21]. Movement requires to bring into account topology stability issues. In addition, the phenomenon under tracking may be mobile in nature (e.g., a moving objective during tactical surveillance), therefore requiring proactive or periodic reporting from nodes to the base station, whereas fixed events may require a reactive (i.e., generating traffic when reporting) behavior. Dynamic nodes or phenomena also generate area coverage issues. As sensors can only cover a circumscribed area with their limited sensing capabilities, movement may leave some environment portions uncovered. Thus, coverage conservation issues must be treated in protocol design.

Scalability. Fourth, as the number of sensors in a network may vary from tens to hundreds to thousands of units, a routing protocol must be able to scale well over small to large number of nodes. It is worth highlighting that scalability may also be referred to the efficiency or quality in event reporting: when nothing particular is happening to sense, most sensor can spare energy by spending majority of the time in a power saving mode, while other active sensors provide a coarse detection quality, to be improved when needed, i.e., when an event occurs.

Application-awareness. A fifth issue stems from the data reporting methods used. These methods are strictly application dependent and are also to be selected depending on data time–criticality. We can identify three major methods of data reporting:

- *Time Driven*: to be used in applications that require periodic monitoring, like temperature and weather sensing. With this method, sensors will periodically wake up, sense the environment, generate related data and send a packet to a base station.
- *Event Driven*: to be used when events are to be sensed automatically, but not in a continuous way. In this configurations, sensors are expected to be in sleep state most of the time, and to switch on only when a significant event occurs. The rationale behind this is that

rare events such as earthquakes and bridge breaks do not require continuous sensing, but are to be continuously followed when they happen.

- *Request Driven*: to be used when events are to be sensed, but data are predominantly generated following upcoming requests from a base station. For instance, one may be tracing the level of soil humidity and other relevant environmental parameters in an agricultural context, and may be interested in sending queries regarding the parameters to sense rather than having them sensed automatically.

Quality of Service. Finally, quality of service may be a fundamental issue for some applications. If data are to be delivered within a certain amount of time from the moment they are sensed, then low latency routing protocols are to be implemented. If, on the other hand, network lifetime is a major concern, energy–aware routing protocols should be implemented so as to prolong network operativity as much as possible.

Note that hybrid configurations implementing more than one of the previous methods are also possible. The routing protocol is greatly influenced by the selected method in terms of route calculation and energy consumption. For the sake of clarity, we decide to classify the approaches taken in the design of routing protocols for WSNs into three macro–categories, namely

Location–based routing, that exploits geographical information to route data inside the network;

Data–Centric routing, where nodes are assigned essentially the same functionalities, and

Hierarchy–based routing, where nodes play different roles in the network.

Obviously, this is not the only possible classification. Routing protocols with peculiar features may be defined better by terms like *multi path–based*, *query–based*, *negotiation–based*, or *QoS–based*, for instance. Moreover, a further classification is possible into *reactive* protocols which compute the routing paths only when they are requested and *proactive* protocols which preserve all routing paths; into *adaptive* and *non adaptive* protocols, which can or cannot tune certain parameters to adapt to current network or residual energy conditions and into *cooperative* and *non cooperative*, that are or are not able to forward data to central nodes (or concentrators) which are endowed with the task to process, aggregate and possibly further process incoming packets, so that the overall amount of data packets in the network is reduced.

5.4.1. Location–based Routing

We start discussing location–based routing as algorithms of this kind are sometimes easier in concept.

◇ **Geographic Adaptive Fidelity (GAF)** This protocol represents an effort to use system-level and application-level information to steer unnecessary nodes to a sleeping status, while keeping active the only nodes that are important from a forwarding point of view.

The protocol is based on the concept of *equivalence* of forwarding nodes. Each node assumes that all sensors are distributed over a virtual planar grid subdivided into square sections: nodes inside a grid square section are all equivalent from a routing perspective, in the sense that it is not important whether the data packet is forwarded to one of the nodes or another inside the same square. This feature lowers the energy expenditure, since all nodes that lay in the same region of the selected relay may be turned off to save energy, but implies an inherent reduction factor of $\sqrt{5}$ in the maximum allowable transmission range, which is needed to allow communications between the two farthest possible nodes in two adjacent grid squares [22]. Thereby, the mean number of hops needed to reach a destination increases.

The aim of GAF is to have a single active node inside each grid square, while all other nodes are sleeping. This is obtained through periodic discovery operations and broadcasting of neighborhood information and through tuning of the sleeping time of each node.

As an additional feature, GAF explicitly takes into account node mobility through the knowledge of a system-level node speed parameter from which it measures the probability that an active node moves out of the square it is in charge of and, in turn, the mean allowable sleeping time a node can afford without losing contact with a previously known active node.

◇ **SPAN** SPAN [23] is a protocol oriented to power saving via topology information maintenance. It is based on the election of some nodes to the role of *coordinators*, so that forwarding is only allowed from the initial source to a coordinator and, then, only between coordinators up to the sink.

If the network is sufficiently dense, it is possible to discover a set of nodes which ensures total network coverage and connectivity: this set of nodes is to be periodically substituted with a totally disjoint set, so as to redistribute energy consumption.

Every node may become a coordinator: this choice is made as the sensor node wakes up and is based on the residual node energy and the perceived advantage given to the node's neighborhood if the node decides to become a coordinator.

> HIGHLIGHTS
> GAF and SPAN are topology–dependent location–based algorithms. The reduction of energy consumption is obtained by selecting a connected subset of nodes as relays. These nodes are determined by means of negotiation procedures that take into consideration neighbors position and other physical layer parameters, following a cross–layer paradigm.

◇ **MFR, DIR and GEDIR** In [24], basic geographic routing algorithms are described. The main concern here is to engage in a progressive advancement towards the final data destination through the selection of a next hop that has some desirable properties which are derived from the geographical locations of the sink, the sender and the intermediate relay. Most Forward within Radius (MFR) tries to select the relay which offers the maximum advancement towards the destination by the minimization of the dot product $\overline{DR} \cdot \overline{DS}$, where D is the sink, R is the relay, S is the source and, for instance, $\overrightarrow{DR}$ is the Euclidean distance between D and R. DIR is a "compass" method that maximizes the inner product $\overrightarrow{SR} \cdot \overrightarrow{SD}$, i.e., tries to forward the packet in a direction that is most close to the line that ideally joins S and D. Geographic DIrection Routing (GEDIR) is a greedy algorithm that, in its basic version, forwards the packet to the closest neighbor to the destination among all available neighbors. There may be situations that force a node to forward a received packet towards the same node that sent it: in this case, the algorithm fails.

We note that none of the algorithms described above is able to guarantee the delivery of the packet.

◇ **Geographic Random Forwarding (GeRaF)** With this protocol, the sensors manage MAC and routing operations in a distributed manner, basing on a contention to select the next hop for packet forwarding. The routing operations are conducted by means of geographical information, i.e., the position of the active relay node, of the final destination (the sink) and of the selected next hop. It is supposed that this information is already available at the moment a node has to make MAC/Routing operations, hence a preliminary phase of network setup, with flooding of localization messages has to be taken into account.

Since sensor nodes have a limited storage capability, it is not feasible that each sensor keeps trace of every other node, since this would lead to a waste of memory resources. In order to make nodes aware of the geographic position of all other sensors implied in a contention or forwarding phase, it is sufficient that every needed positional information (namely, of the transmitting relay, of the sink and that of the chosen next hop) is exchanged on-time by means of control messages. Since GeRaF sets up a collision avoidance mechanism to increase the global network throughput, the use of control messages to send and piggy–back positional information comes at no cost.

From an operational point of view, GeRaF tries to select the relay node that guarantees the maximum advancement towards the sink. This is done by subdividing the advancement zone into regions: nodes belonging to regions nearer to the transmitting node offer a lesser advancement towards the destination.

When a node has a packet to send, would it be to relay another node's packet or to transmit a packet generated on his own, it interrogates advancement regions using a Request-to-Send (RTS) messa e, containing the sender's geographical coordinates: only the nodes in the first advancement region (and all of them) reply to the message by a Clear-to-Send (CTS) packet, including their own coordinates. Should a single node reply to the RTS, it is automatically selected as a relay. If multiple nodes reply and a collision occurs, the sender node issues a COLLISION message to start up a collision-resolution scheme at the eligible receiving nodes: in particular, from now on nodes will reply to subsequent solicitations using a probabilistic bisection rule, that is, they send back control messages with a fixed probability of 0.5.

On the other hand, should no nodes reply to the first RTS message, the sender assumes there are no relays available in the first advancement region and transmits a CONTINUE message to solicit the nodes in the next advancement region. If no nodes reply, the sender keeps issuing CONTINUEs until the last region is reached, or until one or more nodes reply: in this case, the sender and the relays behave in the same way that has been described before when CTS transmission was cited.

Note that all the previously described operation is transparent to on/off duty cycles.

This protocol apparently needs geographical information to be available, for instance via control messages: it is worth noting that this information has to be processed to take routing decision, namely computing one or more Euclidean distances. Since these calculations imply root-square extraction, they may turn out to be infeasible for nodes with low processing capabilities, unless simplifying assumptions of some kind are made.

GeRaF has been initially designed for sensor nodes owning two separate radios to be used for a data and a control channel [25, 26], but there also exists a version for single-radio sensors [27].

HIGHLIGHTS

GeRaF is position–based algorithms, designed according to a cross–layer approach (MAC/Routing). Energy saving mechanism consists in on/off cycle. The algorithm is quasi–topology independent, since nodes are required to know their own spatial coordinates and those of the target node only. Next relay is chosen to maximize the progressive advancement of the message toward the destination. The contention mechanism used to select the relay node is robust to topology variations due to ON/OFF cycles. GeRaF is developed to work either with one radio only or two radios.

◇ **Geographic and Energy Aware Routing (GEAR)** This algorithm [28] addresses the problem of query dissemination to appropriate data regions, since often queries contain geographic information. GEAR is similar to a protocol discussed below, Directed Diffusion, in that it propagates queries on an interest basis, but only to a geographically restricted portion of the network, hence conserving more energy than Directed Diffusion. In GEAR, each node keeps an *estimated cost* and a *learned cost* which are related to the energy to spend and the distance to cover in sending the data to the destination through neighboring nodes. The learned cost also includes an estimation of the additional cost due to routing around network connectivity holes. Learned costs are propagated one hop back every time a packet is forwarded, so that the next route to the same destination could be adjusted.

A first phase of the algorithm covers the forwarding operations towards the region data are claimed from. Each node in the path to the queried region selects the neighbor closest to the destination as a next hop and performs cost learning if connectivity holes occur.

A second phase is then driven once the packet reaches the geographic region it is directed to: at this point, further query dissemination may be performed with any sort of flooding. For example, flooding as in Directed Diffusion could be employed or, more efficiently, nodes could engage recursive geographic forwarding (the region to cover would be subdivided into four smaller regions, sending a copy of the packet to each one and so forth, until regions with only one node are left).

In [28], GEAR is also compared to a non–energy–aware routing protocol, Greedy Perimeter Stateless Routing (GPSR), which also can solve network holes problems. GPSR is somehow simpler, in that it reduces the number of states a node must keep, but on the other hand was designed for general mobile ad hoc networks, and is in fact outperformed by GEAR, that offers higher packet delivery rates and lower energy consumptions under both even and uneven traffic distributions.

HIGHLIGHTS

GEAR is an application–aware algorithm, that aims at gathering information from a target region rather than a specific node. The query is forwarded to the nodes in the target region by means of a distributed algorithm, which keeps into consideration both energy cost and connectivity problems to reach the destination. The algorithm requires nodes to maintain a rather large state vector.

◇ **Adaptive Self-Configuring sEnsor Networks Topologies (ASCENT)** ASCENT is another topologically driven protocol. It is based on the distinction between *active* and *passive* nodes. An active node may forward packets while passive nodes eventually process the collected information but do not participate to any forwarding procedure.

If a node finds itself caught in the impossibility to forward a message, it sends a HELP message to neighboring passive nodes, which may in turn choose whether to become active. In the first case, they inform their neighborhood of the state transition and begin forwarding packets.

> HIGHLIGHTS
> In general location–based algorithm are completely distributed and require none or local only topology information. Energy saving is usually obtained by allowing each node to schedule on/off cycles. Most of them are designed according to a cross–layered approach using physical parameters or implementing data aggregation techniques. All of them introduce some control traffic overhead. They do not work properly in case of sparse networks.

5.4.2. Data–Centric routing

Data–Centric routing category encompasses those algorithms that are aimed at getting information about a given event or from a specific region, rather than creating a path to a specific node.
Ideas behind these algorithms come from the fact that as nodes tend to be deployed in large networks, it is inefficient to assign each node a global identifier, whereas it is better to query some regions or nodes which have access to interesting data, waiting then for the nodes to answer back.

⋄ **Directed Diffusion** Directed Diffusion (DD) [29] is a popular data routing and aggregation paradigm for WSNs which enables efficient multiple–to–single node transmission.

In DD, when a certain station is interested in harvesting data from nodes in the network, it sends out an *interest*, which describes the task to be done by the network. Each node receiving the interest forwards it to its neighbors in a broadcast fashion. As broadcasts are received at a node, it sets up *interest gradients*, i.e., vectors that describe the next hop to propagate the query results back to the requesting station. As a general idea, if a node S sends an interest which reaches node A and B and both forward the interest to node C, then node C will set up two vectors indicating that query results matching that interest should be sent back to A and/or B. The gradient modulus (or *strength*) is different towards different neighbors, which may result in different amount of information being redirected to each neighbor. Each gradient is related to the interest it has been set up for. As the gradient setup phase has been finished for a certain interest, future requests interested in the same attribute are used to reinforce the best (i.e., strongest) gradients throughout the network, so as to convey information flows through the best paths, hence avoiding future flooding. As nodes may receive related information flows from different nodes in the network, they perform a sort of data aggregation or fusion to reduce the total amount of forwarded information. This process may be done efficiently as all nodes in DD are application–aware and may then be able to use the best aggregation method. Finally, the base station will be eventually reached by the information flows.

We note that the base station must periodically re-send interests to the network, in order to avoid interest loss due to unreliable wireless transmission.

Another DD use is to spontaneously propagate important detected events to some portion of the network.

Because of the hefty gradient setup operations, DD is only suited to applications that require nearly continuous query answers, whereas it is too energy–expensive to implement if the gradients are to be set and used only rarely.

◇ **Sensor Protocols for Information via Negotiation (SPIN)** The SPIN protocol family [30, 31] runs on the concept that information must be conveyed through the network assuming that all nodes are potentially base stations. SPIN assigns each data to be routed a meta–data, i.e., a high level name that enables negotiations before sending data, and that is totally configurable depending on application needs. SPIN is also energy–aware and takes routing decisions based on residual node energy levels.

The handshake prescribed by SPIN obeys the following sequence. First, if a node has data to be shared, it sends an ADV message to advertise all neighbors of the upcoming communication. If a neighbor is interested in the data, it sends back a REQ message and then waits for DATA transmission.

A second SPIN version, namely SPIN–2, performs as described before if the available energy is abundant, but when it approaches a low threshold level then the protocol verifies first if the residual energy is sufficient to terminate an handshake without depleting. Other protocol versions are SPIN–BC (for broadcast channels), SPIN–PP (for point–to–point connections), SPIN–EC (like SPIN–PP with some additional energy heuristic) and SPIN–RL (used over lossy channels). All of these protocols are suited to applications where the sensors are mobile, since all forwarding decisions are based on local neighborhood information.

The advantages of SPIN include the meta–data negotiation, which substantially reduces the amount of redundant data, and the need for only local neighborhood information for routing. On the other hand, a strong disadvantage is that SPIN cannot guarantee data delivery, since uninterested nodes on the path to the destination will not participate in the communications.

SPIN is different form DD, since DD issues data queries and builds up interest gradients for query answers whereas in SPIN data sources are the first to initiate the communication. Moreover, all communication in DD is towards neighbors, with data aggregation features, whereas SPIN needs to maintain global topology information to route towards the sink. Indeed, SPIN is more suited than DD to applications that require continuous data reporting to a base station, since every communication is driven by nodes that have access to the requested data; moreover, matching queries in DD to the available data may require extra overhead and, thus, extra energy consumption with respect to SPIN.

◇ **Rumor Routing** Rumor Routing (RR) [32] exploits the fact that for many applications, geographic routing is infeasible, and it is not necessary to find the shortest path to the destination, but an arbitrary path is sufficient. As explained before, DD floods queries over the whole network to reach nodes that can provide data matching the query. With RR, instead, nodes detecting an event record it into a local event table and then propagate a long life packet call *agent*. When a node is interested in a certain event, it forwards a query that eventually reaches a sensor that has the event recorded in its local table. If this happens, the node knowing about the requested event can respond to the query, since a hit in the event table may happen only if the node had previously been reached by an agent that informed it of the event. This way, there is no need for global network flooding and energy may be saved.

A drawback of RR is that it is only suitable for applications where few events are generated, otherwise the network would experience saturation due to excess agent generation. Moreover, the protocol is very dependent on the heuristics used to propagate agents hop–by–hop, and is vulnerable to node mobility and sparse regions.

⋄ **Minimum Cost Forwarding Algorithm (MCFA)** MCFA [33] exploits the fact that the direction of routing is known, since it is towards a certain sink. Thus, a node need not maintain a routing table, but instead can maintain a minimum cost value, containing the estimate of the least forwarding cost to the base station. Such a minimum path–cost is enclosed in the header of each transmitted packet. Furthermore, the packet also carries the total cost that has been consumed along the path, from the source to the current intermediate node. However, no node identification fields are required to be enclosed in the forwarded message.

When a node has a message to transmit, it broadcasts the message to its neighbors. Hence, each node checks whether the sum of its own minimum cost towards the BS and the cost consumed so far by the packet (enclosed in the packet header) equals the minimum path cost declared for the packet (also enclosed in the packet header). If so, the node belongs to the transmitter's least cost path to the base station and, hence, it rebroadcasts the message. The process continues this way until the message is delivered to the base station.

To obtain the least cost estimate, the BS floods the network with a message with initial cost zero, while each node sets its cost estimate to ∞. When a node receives a cost estimate it checks if the sum of the estimate with the cost of the link it has received the packet, is less than the actual known estimate. If this is true, it updates and broadcasts the new estimate, otherwise it does nothing. A backoff algorithm is implemented to avoid collisions at nodes distant from the sink, that are more likely to receive a greater number of estimates.

⋄ **Information–Driven Sensor Querying and Constrained Anisotropic Diffusion Routing** These two protocols (IDSQ and CADR) [34] aim to be a general form of Directed Diffusion, which queries sensors in such a way that information gain is maximized, while energy consumption and latency are minimized. CADR, in particular, activates only the sensors that are close to a required event and routes basing on gradients which are formed taking into account a balancing over the information acquired and the cost spent. Moreover, IDSQ sets up a communication between the sender node and the neighbor that offers the highest information gain, but also that best balances the energy cost. As IDSQ does not provide instructions on how to forward data requests from the base station to the nodes, it may be seen more as a complementary optimization procedure.

The CADR–IDSQ approach is more efficient than Directed Diffusion, since it avoids the excess energy expenditure due to isotropic query forwarding.

◇ **COUGAR** This protocol [35] abstracts from the format used to compile a query (which is left to be application–dependent) and utilizes data aggregation to reduce the amount of transmission in the network. It differs from Directed Diffusion in that it makes nodes select a *leader* to perform data aggregation. The leader itself is in charge of data forwarding to the sink.

The base station provides a query that is compliant with the application–specific format and which incorporates a method to select leaders among the nodes. The main advantage of this procedure is that the network itself operates in a way that is independent of the application, but the architecture also has some drawbacks: first, the nodes need query processing functions which may in turn generate extra overhead and energy consumption, and second, the leaders need to be dynamically changed in order not to become bottlenecks or to deplete available energy too fast.

◇ **Routing protocols with random walks** Routing based on random walks [36] tries to achieve load balancing through multipath routing in WSNs.

It assumes that the network is made of a large number of nodes with limited or no mobility which may go to sleep mode or wake up at random times. A regular grid is assumed so that nodes fall exactly on a crossing point each, even if the topology is irregular (i.e., nodes may not cover all of the grid crossing points). To find a route towards the sink, nodes apply a distributed asynchronous version of the Bellman–Ford algorithm and select the next hop to be the one which is closer to the base station with a certain probability. By adjusting this probability of node choice, some load balancing may be obtained. Anyway, the next hop is changed any time a new packet has to be forwarded.

This protocol has its main drawback in the assumption about the topology of the nodes.

> HIGHLIGHTS
> These algorithms use flooding mechanisms to propagate the requests over the networks and to find the best routing path to deliver the queried information to the sink. They are distributed and strictly topology dependent. They are often designed according to a cross–layer approach, since data aggregation techniques are combined with path discovery to reduce data redundancy and energy consumption. Notice that, this additional data processing requires not trivial storage and computational capabilities.

5.4.3. Hierarchical–based Routing

Hierarchical protocols exploit the advantages of clustering techniques, specially under a scalability and a communication efficiency point of view. Moreover, the packet forwarding may also be more energy-sparing, since high–energy nodes may be selected to process and send information in place of low–energy nodes which may limit to sensing operations.

◇ **Low Energy Adaptive Clustering Hierarchy (LEACH)** LEACH [37] is a protocol based on clustering which elects and subsequently rotates cluster heads (CH) on a periodic basis to evenly distribute the energy consumption throughout the network. CHs set up communication with a TDMA method, making use of CDMA to reduce inter–cluster interference. Energy savings come from data compression, data fusion and the random rotation of cluster heads.

The operation of LEACH is separated into two phases: a first one where the cluster heads are selected and a second one where data transfers are performed. During the first phase, all nodes independently decide whether to become a CH: the election depends on the desired percentage of CHs, on the actual rounds and on the set of nodes that has not become a CH in the previous rounds. Elected CHs broadcast an advertisement to neighboring nodes. During the second phase, sensing is performed and data received by CHs are aggregated and sent to the sink. Periodically, CHs are re–elected.

This protocol indeed suffers from a number of problems, i.e., it leads to a fast exhaust of the energy of nodes that lead from a “hot spot” to the sink, where a hot spot is a place where multiple events take place in a rapid succession; moreover, it is not suitable for time-critical applications. Moreover, it assumes that nodes always have data to transmit and that, if needed, each node is sufficiently close to the sink to be able to communicate directly to it.

LEACH could also be combined with meta–data negotiation [37].

◇ **Power–Efficient Gathering in Sensor Information Systems (PEGASIS)** PEGASIS, [38]is derived from LEACH [37] and share its advantages and problems. In particular, it assumes that all nodes own location information of all other network nodes and are all able to transmit directly to the sink, besides being immobile. These assumptions lead PEGASIS to outperform LEACH in the sense that it does not need any more dynamic cluster formation, and only needs to equilibrate the overhead due to the communication between the leader and the sink by means of token passing.

◇ **Threshold–Sensitive Energy Efficient sensor Network protocol (TEEN)** TEEN [39]is a forwarding protocol that is most suited for time–critical applications. It forces nodes to avoid transmission if the value output from their sensing apparatus, or the variation from the last sensed value, is below some threshold. Specifically, clustered network structure is maintained as in LEACH, but the CH also broadcasts to the nodes a pair of threshold values, namely a *soft* and a *hard* threshold. Thus, as a new value is sensed, the node compares it with the hard threshold. If it passes the test (i.e., it is in the range specified by the threshold), then the variation from the last sensed value is evaluated and compared with the soft threshold before transmitting. If this test is also passed, then transmission is allowed. At cluster changes, new parameters are broadcasted.

APTEEN (Adaptive Periodic TEEN) [40] is a hybrid protocol that modifies TEEN in order to achieve more adaptability to application needs, as it allows to scale periodicity and threshold values. It engages a TDMA schedule between nodes to manage multiple transmissions in the same cluster and to let parallel transmissions by hybrid networks take place without excess interference. Its main drawbacks are the additional complexity required for maintaining the TDMA schedule.

◇ **Small Minimum Energy Communication Networks (SMECN)** The work in [41] devises a protocol (MECN) that computes an energy–efficient subnetwork that is made by neighboring nodes towards which transmission is more efficient than direct transmission to the destination of a packet. These regions may re–configure automatically to adapt to node failures or to the deployment of new sensors.

SMECN [42] is an extension to this protocol that constructs smaller subnetworks and, thus, operates on smaller graphs representing the network, helping in constructing minimum–energy paths to send messages. The algorithm, anyway, is local in that it does not compute the global minimum energy path, but builds a subnetwork in which it is guarantee to exist.

◇ **Self–Organizing Protocol (SOP)** SOP [43] provides a hierarchical architecture where each node willing to send a packet to a base station forwards it first to router nodes. Thus, only routers need to be addressed uniquely and each sensor in their coverage range is identified by the address of the router they report to. Routing is performed by a Local Markov Loop algorithm (i.e., a random loop on spanning trees) that let the network be robust towards node faults.

As nodes are reachable directly through their router address, this protocol is particularly suited to applications that interrogate single nodes. However, some overhead is required for the organization of clusters.

◇ **Virtual Grid Architecture Routing (VGA)** The work in [44] regards an energy–efficient algorithm that is best applied to static networks. A localization method that does not make use of GPS [45] is implemented to build clusters that are equivalent in shape and non overlapping. A square shape is chosen in [44]. Inside each zone, a node is optimally elected to cover the role of cluster head. Two–level aggregation is used as a form of data redundancy removal, both in a local and a global context. The optimal choice of local aggregators (LA) is NP–hard, hence some heuristics have been proposed in [44] to solve this problem. Moreover, another work [46] presented a way to reduce the energy consumption by optimally selecting global aggregators (GA) among the LAs that minimize the overall energy consumption.

These algorithms have proven to be fast and scalable for large networks, and to produce LA and/or GA selections that are non far from the optimal solution.

◇ **Hierarchical Power–Aware Routing (HPAR)** The HPAR [47] protocol divides the network into smaller groups of sensors that are clustered in a geographic zone. Each zone is left the decision about packet routing, that is performed in such a way that the energy consumption in the cluster is minimized. Routes are chosen so that the formed path has the maximum over all minimum remaining powers (the *max–min path*). In [47], this algorithm was approximated by a so–called *max–min* zP_{min} approach. As a matter of fact, the algorithm first calculates a shortest path with the Dijkstra method using power consumption as a link metric and then finds a path that maximizes the minimum residual energy in the network. This is obtained by relaxing the constraint on the P_{min} power consumption of the shortest path through the multiplication of P_{min} by a factor $z \geq 1$. Thus, the algorithm consumes at most zP_{min} while maximizing the residual power in the network.

Zone–to–zone routing is then implemented to find a path from zone to zone. Each node inside a zone participates in the routing process by estimating the zone power level.

⋄ **Two–Tier Data Dissemination (TTDD)** TTDD [21] realizes effective data delivery to multiple mobile base stations. Sensors are assumed to be stationary and location–aware. Each data source builds a grid structure that is used to disseminate data. Its formation is as follows. A future data source sends an advertisement to its four nearest grid crossing points. When the message reaches the node closest to a crossing point, the node stops it and further propagates another message to all its nearest grid crossing points except the one the advertisement had come from. Stopping nodes will then act as dissemination points.

With this structure, a base station may flood a query which will then propagate through dissemination points up to the queried node. The message is then sent back in the same fashion using the reverse path to the sink. The base stations forward trajectory information to let nodes predict their position for routing purposes.

This protocol does not compute the shortest path to the destination, but is indeed very scalable, and the authors in [21] think this advantage is worth the lack in path optimality. Furthermore, another concern about this protocol regards the maintenance of grid information and the necessity for an accurate position information.

> HIGHLIGHTS
> The main drawback of cluster–based algorithms is the resulting high cost, in terms of control traffic and energy consumption, to maintain the hierarchical architecture. On the other hand, cluster–based algorithms are suited to perform data aggregation and manage residual network energy in an efficient way. Some of them also require localization capabilities.

5.5. Sensor data aggregation

Data aggregation and in-network processing techniques have been investigated recently as efficient approaches to achieve significant energy savings in wireless sensor networks by combining data arriving from different sensor nodes at some aggregation points enroute, eliminating redundancy and minimizing the number of transmissions before forwarding data to the sink. This paradigm shifts the focus from traditional address-centric approaches for networking (finding short routes between pairs of addressable end-nodes) to a more data centric approach (finding routes from multiple sources to a single destination that allows in-network consolidation of redundant data).
This approach is particular useful in many wireless sensor networks for environmental monitoring.

In [48] there is a study of the energy savings and the delay tradeoffs involved in data aggregation and how they are affected by factor such as source-sink placements and the density of the network.

There are two different types of data aggregation:

- Data Fusion or aggregation with Size Reduction;
- Aggregation without Size Reduction.

The first form of aggregation can be used, for example, in sensor networks intended for temperature monitoring when we are only interested in averaged regional value of the temperature. In this case an intermediate node that receives two values of temperature can calculate the average and forward it to the sink. The resulting data field has the same length as the incoming packets.

Aggregation without Size Reduction, on the other hand, occurs when two packets received from different sources are merged in a single packet with a longer data field.

In both cases, the MAC layer leverages data aggregation since the overall transmission overhead can be reduced. Furthermore, since medium contention is performed at MAC layer for each packet to be transmitted, the contention overhead is reduced when a node contends only once to transmit a longer packet with respect to multiple contentions for shorter packets.

In general, it is possible to examine three schemes of data aggregation:

- *Center at nearest sources*: all sources send their data directly to the source which is nearest the sink. This source then sends the aggregated information to the sink.
- *Shortest path tree*: each source sends its information to the sink along the shortest path. If there are overlapping paths, they are combined to form aggregation tree.
- *Greedy incremental tree*: this is a sequential scheme. At the first step, the aggregation tree consists of only the shortest path between the sink and the nearest source. At each step after that, the next source closest to the current tree is connected to the tree.

One of the first approaches to the data fusion problem is LEACH [37]. The network is divided in some clusters, each having a cluster head that aggregates the data gathered by the cluster members. This scheme permits to limit the amount of data transmitted over the air, thus reducing the network energy consumption and increasing the efficiency in the channel access. In [49] an analytical study of the data aggregation problem in a multi–hop geographical routing scenario is considered. The work presents an aggregation scheme where spatially correlated data is aggregated at the node called "cluster" which, subsequently, forwards the information to the sink node through multi–hop routing. The authors show the trade–off between energy consumption and network lifetime at the varying of the cluster size. The optimal cluster size as a function of the spatial data correlation is, finally, derived.

The most common approach for data aggregation is based on aggregation trees. Some example of this approach are illustrated in [35] [50] and [51]. In each of this studies, the goal is to find the tree that produces the best performance in terms of energy consumption and time delay. The tree–based approach, though, is sensible to variations of the network topology and links failure.

Such weaknesses are partially compensated by multi–path based schemes. However, [52] demonstrated that multi–path routing often results in message duplication, which would cause a higher overhead (energy consumption). To alleviate this problem, authors have introduced an operator capable of annihilating duplicated measurements.

An evolution of [52] is presented in [53]. In this case, multi–path routing [52] is used in some regions, while tree–based schemes are used in others. The dimension of the regions is modified depending on the data aggregation requirements and on the link and node failure probability. The disadvantage of this algorithm is due to the message overhead.

In [15] a cross layer solution for routing and data aggregation is proposed.

HIGHLIGHTS
In summary data aggregation and data processing techniques can be valid instruments to reduce the amount of data sent over the network and, consequently, to reduce the energy consumption. On the other hand, they might require higher storage and computational capabilities, resulting in a higher power consumption at nodes that perform data processing. Cluster–based network architectures are particularly suitable to support data aggregation, since cluster heads are ideal candidate to perform data fusion/aggregation. According to this perspective, it is advisable the use of more powerful nodes as cluster-heads, in order to alleviate the problems that have been pointed out.

5.6. Clustering and Backbone Formation

5.6.1. Clustering for Ad Hoc networks

The notion of cluster organization has been investigated for ad hoc networks since their appearance. The first solutions aimed at partitioning the nodes into *clusters*, each with a *clusterhead* and some *ordinary nodes*, so that the clusterheads form an independent set, i.e., a set whose nodes are never neighbors among themselves. In [54] and [55], a fully distributed Linked Cluster Architecture (LCA) is introduced mainly for hierarchical routing and to demonstrate the adaptability of the network to connectivity changes. The basic concept of LCA is adopted and extended to define multi-level hierarchies for scalable ad hoc routing in [56]. With the advent of multimedia communications, the use of the cluster architecture for ad hoc network has been revisited by Gerla et al. [57–59]. In these latter works the emphasis is toward the allocation of resources, namely, bandwidth and channel, to support multimedia traffic in the ad hoc environment. These algorithms differ on the criterion for the selection of the clusterheads. For example, in [54, 55, 59] the choice of the clusterheads is based on the unique identifier (ID) associated to each node: the node with the lowest ID is selected as clusterhead, then the cluster is formed by that node and all its neighbors. The same procedure is repeated among the remaining nodes, until each node is assigned to a cluster. When the choice is based on the maximum *degree* (i.e., the maximum number of neighbors) of the nodes, the algorithm described in [58] is obtained.

The DCA algorithm generalizes these clustering protocols in that the choice of the clusterhead is performed based on a generic “weight” associated to a node. This attribute basically expresses how fit that node is to become a clusterhead. All these protocols produce a set of clusterheads that are independent, and criteria for joining them to form a connected backbone must be defined. A possible rule adopted by DCA is that defined by Theorem 1 in [60]: In order to obtain a connected backbone it is necessary (and sufficient) to join all clusterheads that are at most three hops apart via intermediate nodes (called *gateways*). Clusterheads and gateways form the backbone. Different choices and definitions for the weights and the effects of the particular choice on the DCA and similar protocols have been investigated in [61], [62] and [63]. The effects of mobility on the basic clustering produced by the DCA, as well as methods for reducing role changes (protocol overhead), has been considered in [64–67]. Clusterhead selection and backbone formation, although different from the methods used in the mentioned solutions, are also the two fundamental steps of the WAF algorithm [68]. More specifically, WAF start by selecting a node that will serve as the root of a tree. The tree construction leads to the selection of some nodes as clusterheads, which are then interconnected to form quite a slim of a backbone.

Other protocols based on constructing an independent set of clusterheads and then joining them to form a backbone are defined in [69] where the choice of the clusterheads is performed based on the normalized link failure frequency and node mobility. Two rules are defined for joining the selected clusterheads. The first rule utilizes periodic global broadcast messages generated by every node but forwarded only by the clusterheads. Non receiving re-broadcasts from all neighboring nodes via its clusterhead makes an ordinary node aware of a disconnection. The ordinary node becomes a backbone node providing connectivity. The second rule is similar to the one used by the DCA algorithm to build up a backbone. In [70] and in [71] DCA is used as basic clustering and rules are then defined to limit the size of the clusters.

Once the nodes are partitioned into clusters, techniques are described on how to maintain the cluster organization in the presence of mobility (clustering maintenance). Mobility has been the driving design parameter for some clustering algorithm, such as the ones presented in [72]. Wang and Olariu discuss the problem of clustering maintenance at length in [73], where they also present a tree-based clustering protocol based on the properties of diameter-2 graphs.

Algorithms directly concerned about building a backbone which is a connected dominating set have been presented in [74], [75] and [76–78]. The idea in this case is to seek for a dominating set and then grow it into a connected dominating set. The emphasis here is to build a routing structure, a connected *spine* that is adaptive to the mobility of the network nodes. Differently from the solutions mentioned above, which distribute and localize the greedy heuristic for finding a maximal (weight) independent set, these solutions are a distributed implementation of the Chvátal heuristic for finding a minimal set cover of the set of nodes. A similar approach is followed in [79] where a minimum set cover is built in a distributed and localized way: Nodes in the set cover are databases that contain routing information. A somewhat different approach is adopted in the WuLi protocol [80]. Instead of constructing a dominating set and then to join its nodes to make it connected, a richer connected structure is built, and then redundant nodes are pruned away to obtain a smaller CDS. Several different pruning rules are investigates in [81]. One among the most effective in removing redundant nodes has been introduced in [82]. The number of nodes in the CDS can be further reduced by using nodal degree and the nodes (GPS) coordinates, as proposed in [83].

Most of the clustering protocol mentioned so far generate clusters of diameter ≤ 2: The clusterhead always dominates its cluster members. There have been advocates for larger, possibly overlapping clusters. For instance, [84] describes routing for dynamic networks (such as ad hoc networks) which is based on overlapping k-clusters. A k-cluster is made up of a group of nodes mutually reachable by a path of length $k \geq 1$ (1-clusters are cliques). Clustering construction and maintenance in face of node mobility is presented, as well as the corresponding ad hoc routing. A clusterhead election protocol, with corresponding cluster formation is described in [85]. The focus in this paper is to efficiently build disjoint clusters in which each node is at most $d \geq 1$ hops away from its clusterhead. The network is clustered in a number of rounds which is proportional to d, which favorably compares to most of previous solutions when d is small. Finally, the mentioned clustering protocol presented in [72] produces clustering of variable diameter. In this case the diameter depends on the degree of mobility of the nodes: The more the nodes move, the smaller the clusters (easier to maintain), and vice-versa.

More recent work for clustering and backbone set up is described for networks quite different from the general ad hoc model considered in this paper. In [86] some nodes are assumed to have “backbone capabilities” such as the physical radio capacity to communicate with other backbone nodes (i.e., the network nodes are assumed to be heterogeneous). The solution proposed in [87] constructs a CDS relying on all nodes having a common clock (time is slotted and nodes are synchronized to the slot).

5.6.2. Clustering for WSNs

We now review clustering and backbone formation protocols that have been proposed explicitly for wireless sensor networks. The main problem here is that of devising energy efficient techniques to transport data from the sensors to the sink. The overall goal, in general, is to increase the network lifetime. Hierarchical solutions like those provided by clustering and backbones appear to be viable for accomplishing this task, as demonstrated in [88–90]. Rather than belonging to one of the general classes of protocols for clustering and backbone formation described above, papers on clustering for WSNs are often specific for a given scenario of application, and, as mentioned, are designed to achieve given desirable goals, such as prolonged network lifetime, improved tracking, etc.

Among the protocols that use clustering for increasing network longevity one of the first is the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol presented in [91]. LEACH uses randomized rotation of the clusterheads to evenly distribute the energy load among the sensors in the network. In addition, when possible, data are compressed at the clusterhead to reduce the number of transmissions. A limitation of this scheme is that it requires all current clusterheads to be able to transmit directly to the sink (single-hop topologies). Improvements to the basic LEACH algorithms have been proposed by [92] and [93] where multi-layer LEACH-based clustering is proposed and the optimal number of clusterheads is analytically derived that minimizes the energy consumption throughout the network. An alternative method for selecting clusterheads for the LEACH-like model is presented in [94]. Particle swarm optimization (PSO) is used for dividing the network into clusters. A clusterhead is then selected in each cluster based on its mean distance from all the other nodes in the cluster.

Localized clustering for wireless sensor networks has been proposed by Chan et al. in [95]. The Algorithm for Clustering Establishment (ACE) has the sensor nodes iteratively talking to each other until a clustering with clusterheads and followers (the ordinary nodes) is formed where the clusters are quite uniform in size and mostly non-overlapping. This minimizes the number of clusterheads. No backbone formation among the clusterheads is described. A tree-based clustering algorithm for sensor networks is presented in [96]. A selected node starts the BFS-based process of building a spanning tree of the network topology. Clusters are then formed by those nodes whose sub-tree size exceeds a certain threshold. The protocol is quite message intensive, and there is no explicit description of backbone formation.

Clustering protocols have been proposed for networks in which some nodes are capable of long-haul communications (heterogeneous networks). This is the case of the clustering proposed in [97] where special, more powerful nodes act as clusterheads for simpler sensor nodes, and transmit the sensed data directly to the sink. The authors have further explored their idea in [98–102] and [103]. Gerla and Xu [104] propose to send in swarms to collect data from the sensors, and to rely the data to the sink via intermediate swarms (multi-hop transport). Clustering for sensor networks of heterogeneous nodes is also explored in [105]. Clusters are formed dynamically, in response to the detection of specific events (e.g., acoustic sensing of a roaming target). Only a certain type of node can be clusterhead, and methods are presented for selecting the more appropriate clusterhead for target identification and reporting. A comparison among sensor networks with homogeneous nodes and with heterogeneous nodes is presented in [106], both for topologies “single-hop” (à la LEACH) and for the multi-hop case. The same authors make the case for heterogeneous sensor networks in [107]. The authors also propose M-LEACH, a variation of LEACH where intra-cluster communications are multi-hop, instead of having ordinary nodes directly accessing their clusterhead.

Finally, the construction of a backbone of sensor nodes is considered in [108] where the nodes need to know their position (such as their GPS coordinates). This work is more along the lines of sensor network topology control (GAF [109]) rather then on hierarchical organization of WSNs as considered in this study. Further references on CDS construction in sensor and ad hoc networks can be found in [110].

HIGHLIGHTS

Overall, clustering has been proven effective to solve many of the problems encountered in the management and in the deployment of ad hoc and wireless sensor networks. While paying little in terms of overhead for clustering construction and maintenance and for the natural increase in route length, recent works have demonstrated that the benefits of imposing a hierarchy in the network are multifold. These benefits include reduced route maintenance cost, reduced information maintained at the node, and corresponding overhead to update it, a natural way to identify the best nodes where to aggregate the data, or the nodes to be switched off for conserving energy, and overall, increased network lifetime. The application of clustering and backbone formation methods to large networks of resource-constrained nodes (e.g., sensor nodes) requires simple, fully distributed solutions. Recent investigations showed that fully localized approaches to clustering (e.g., the WuLi protocol described above) pays off big time, being able to significantly reduce the overhead, and results in low energy consumption while building small backbones.

5.7. Localization in Ad Hoc and Wireless Sensor Networks

One of the major challenges for sensor networks deployment is the *localization* of a sensor node. Each sensor should be able to infer its position with respect to some global (or relative) system of coordinates. This information is essential to all those applications that require spatial mapping of the sensed data for further processing, and is also useful for the overall system performance as the availability of positioning information enable the adoption of low overhead protocols such as geographic-based routing schemes. The Global Positioning System (GPS) provides an immediate solution to the problem of localizing a node. However, GPS is often not a viable solution for providing localization to sensors in many scenarios, due to its cost, the associated energy consumption, its inapplicability in indoor or heavy foliage scenarios. Therefore, GPS-free solutions for localization are of great interest.

Existing localization protocols may be classified by the information used to estimate position. In *range-based* protocols, a node uses estimates of the range or angle from its neighbors to compute its own position. When nodes have no capability for computing such estimates, the corresponding solutions are called *range free*. In the latter case, the use of beacons is required. Both categories present centralized and distributed solutions.

5.7.1. Range-Free Localization

Range-free localization enables the use of very simple network node hardware. In short, there is no need in most nodes for on-board GPS receivers, compasses, antenna arrays or signal-strength indicators. Instead, nodes localize with the transmitted coordinates from other nodes, including special nodes called *landmarks* or *beacons*, which are usually scattered randomly over the network area. These landmarks have extra features with respect to common network nodes. Among these, they know their position—via GPS or manual setup—and they have a greater coverage area than other nodes. Landmarks provide information to the nodes by *beaconing* their position over the network. A non-landmark collects this information and infers its position from this information, using a range-free algorithm.

In [111] the position is estimated using a simple *coordinate centroid method*. Once a particular node, labeled 0 here, has received coordinates from the in-range landmarks it estimates its position as

$$\hat{\mathbf{c}}_0 = \frac{1}{k}\sum_{i=1}^{k}\mathbf{c}_i$$

where $\hat{\mathbf{c}}_0$ is the estimated position of node 0, and $\mathbf{c}_i$ denotes the received coordinates from the i^{th} beacon. The accuracy of this method depends on the distribution of landmarks around the node of interest, as well as the number of landmarks a node can hear. Increasing the number of *symmetrically-placed* landmarks will increase the accuracy of the estimated position. In particular, the accuracy of the coordinate centroid depends on a uniform placement of landmarks with respect to each receiver node. In practice, the solution proposed by Bulusu et al. produces a coarse positioning of the nodes that makes the algorithm suitable only for application in which accurate localization is not required.

In [112], Niculescu and Nath propose a solution called *DV-Hop* that uses a method similar to distance routing. A certain number of landmarks flood the network with their location information. The message containing this information also contains the landmark identification and a counter that is incremented at each hop. Thus, range is estimated by the hop count. Once landmark i receives location information from landmark j, $j = 1, \ldots, N$, it computes the average one-hop (physical) distance as follows:

$$\text{AvgHopDistance} = \frac{\sum_{j=1}^{N}\|\mathbf{c}_i - \mathbf{c}_j\|_2}{\sum_{j=1}^{N} h_{i_j}}$$

where h_{i_j} is the distance, in hops, from landmark i and landmark j, and $\mathbf{c}_i$ and $\mathbf{c}_j$ are the coordinates of landmark i and j, $i \neq j$. This information is then flooded through the network. During the first phase of the algorithm, nodes store the location and distance information from each landmark. When a node receives the average one-hop distance, it uses the stored information to estimate its own coordinates by applying *multilateration*. The main drawback of this method is the network flooding, which results in a considerable waste of energy at the nodes. To reduce this energy cost, each node may decide whether or not to forward received information. While this technique performs well in uniform networks, it is unacceptable in nonuniform networks where the real distances have a high variance around the average one-hop distance value.

APIT, the localization protocol proposed by He et al. [113] is based on the intersection of areas. Unlike [111], APIT uses triangular regions to derive the area in which it is highly likely to reside. An *APIT network* is a heterogeneous network in which the landmarks are *super nodes* that are different from sensors in that they are capable of high-power transmissions and equipped with GPS receivers. Each node considers triples of landmarks and decides whether it is or not in the triangular area whose vertices are the three landmarks via a test called *Point-In-Triangulation (PIT)*. Repeating this operation for each triple, a node is able to determine a series of areas whose intersection gives the area in which the node itself lies with high probability. The main weakness of this approach is due to the reliability of the PIT test. This test requires a node to move in order to determine if it had been previously inside or outside the triangle. This is possible only if the node has some mobility and it is able to determine in which direction it is moving. For these reasons the authors provide an approximate version of the PIT test which overcomes the need for mobility and direction-sensing capabilities but has an intrinsic approximation error which depends on the number of neighbors of the node.

In a recent work by Hu and Evans [114] range-free localization is performed by applying the Monte Carlo Localization method [115]. This method is based on a first step in which a node guesses a set of possible locations based on the position distribution at time $t-1$ and its movements in $[t-1, t]$. A second step follows in which the node uses information received from the neighbors to eliminate from the set from the first step inconsistencies with neighbors' observations.

Previous work on range-free approaches also include [116–118]. All of these works employ area-based techniques for the estimation of position. In [116], each in-range constraint to neighbors is mapped to a linear matrix inequality, and a feasible region was found by semidefinite programming. The estimation technique in [116] is centralized, and the algorithm does not permit uncertainty in the coordinates of the reference nodes. Reference [117] and later [118] consider decentralized, area-based estimation of position. However, their approach employs a discrete grid, the L_∞-norm for distance, and again assumes that reference nodes have perfect knowledge of their position.

5.7.2. Range-Based Localization

The main difference between range-based and range-free methods is the availability of a range estimate between the node of interest and a transmitting node. This range estimate is provided by some physical-layer measurement, and ancillary radio hardware.

Priyantha et al. [119] proposed *Cricket*, a solution based on a heterogeneous network in which particular devices provided with ultrasonic transceivers help the nodes in the network to decide their positions based on a time-difference of arrival between a radio frequency (RF) signal and an ultrasonic signal. In [120] an improvement of cricket is proposed through the use of compasses while in the more recent [121] the presence of noise on range measurements is investigated and an algorithm working on this noisy environment and without use of beacon node is proposed and tested.

In [122] the authors propose a RF-based solution for indoor applications in which distances are measured using RSSI (Receiver Signal Strength Indicator) techniques. During an initial calibration phase, the algorithm creates a signal strength map of the indoor environment based on the RSSI received from landmarks placed in the building. Based on this map a centralized system is used to determine the position of a node based on its received signal.

In [123] Savvides et al. propose a method called *AHLoS (Ad Hoc Localization System)* in which sensor nodes (called Medusa nodes) are equipped with arrays of ultrasound microphones, which enable measurements of acoustic time of flight. In the first phase of this algorithm, node 0 estimates the distance between itself and all transmitting, in-range landmarks i, $i = 1, \ldots, N$. During the second phase the node 0 estimates its position by minimizing a mean-squared-error metric. Each node applies multilateration when it receives information from at least three landmarks. Once a node estimates its coordinates, it becomes a landmark in order to make the algorithm converge faster. While location is estimated accurately in the nominal case, it is prone to error propagation if landmark positions are inaccurate. Secondly, this method assumes an acoustic line-of-sight path, which may not be possible in some sensor network deployments. A similar method is proposed in [124] where *N-Hop Multilateration* is introduced. This method estimates the coordinates of a node in a distributed way so that a node can use information provided by landmarks not directly in range of the node.

In [125], Niculescu and Nath propose two range-based protocols alternative to DV-Hop. The first one is called *DV-Distance* and it is similar to DV-Hop but uses RSSI to measure range. Provided accurate range estimates, DV-Distance yields more accurate location estimates compared to DV-Hop. However, received power is not only a function of range, but also of path-loss exponent, shadowing, and multipath fading, and so range estimation is not necessarily accurate. The second range-based alternative to DV-Hop is based on the Euclidean distance between nodes. A node 0 computes its coordinates when it has the distance to at least two neighbors that know their distance from a landmark and between them. With this information, node 0 infers its distance from the landmark, and hence its coordinates. Simulations in [125] show that the two range-based solutions perform worse than the range-free method because of the error due to the measurements of the distance and the propagation of this error. Range-based methods do not account for range-measurement errors directly in the location estimate for node 0.

An alternative to the RSSI-based range estimate used in DV-distance algorithm is the *AoA* (Angle-of-Arrival) [126] and an improvement of this last algorithm is proposed in [127] in which an error propagation control technique based on a simplified Kalman filter is applied.

A solution similar to DV-Hop, called *Hop-TERRAIN*, is proposed by Savarese et al. in [128]. In this localization algorithm, landmarks first broadcast their information, and each node keeps a table with the distances in hops from each landmark. Once a landmark receives a packet from another landmark it calculates the average one-hop distance and flood this calculation through the network. When the distance information reaches node 0, this node 0 infers a coarse position estimate via multilateration. Node 0 then refines its position estimate iteratively through 1-hop sharing of location information, for a fixed number of iterations.

This solution has two main drawbacks. First, the algorithm does not provide a method to contain location estimate errors which propagate through the network. Moreover, the energy cost for information sharing is high due to flooding and repeated local broadcasts.

Fretzagias and Papadopouli [129] presented a voting algorithm for determining the position of nodes in a grid-based representation of the terrain. Starting from the measured distances from in-range landmarks, a node deduces the area in which it could be located. These areas are circular crowns, obtained taking into account the error on the distance. By superposition of the crowns corresponding to different landmarks, certain cells gain a relatively higher probability to host the node, and the node itself determines its position by maximizing this *a posteriori* probability. After node 0 estimates its coordinates, it broadcasts the information in order to allow other nodes to continue the process. The accuracy of this method depends on the grid resolution: a smaller grid size leads to increased accuracy. The main problem in increasing the grid resolution depends on the available hardware: small sensors node cannot support the required hardware for highly defined grids.

In [130] Chintalapudi et al. investigate the performance of the ad hoc localization schemes based on ranging and bearing information presented in [124] and [125]. They observe that information about the angle of arrival or about the bearing sector lead to good performance and accuracy of localization algorithms even in sparse networks. When only the range information is used, instead, the same accuracy is achieved only when the nodes are densely deployed.

A range-based approach in which no beacon node is used is presented in [131] for localizing sensor nodes. The protocol, termed *Anchor-Free Localization (AFL)*, is a two-step algorithm. AFL provides an anchor-free solution to the problem of localization by using a *concurrent* approach. Instead of proceeding *incrementally* every node guesses its initial coordinates based on local information. As it is typical of concurrent methods, this may lead to *false minima*, i.e., each node believe it is in the optimal (correct) position but the global configuration is incorrect. This motivates a second step, in which, as typical in these cases, the process converges to “correct” localization by applying a force-based relaxation procedure [132, 133]. Simulation results show that AFL achieve acceptable accuracy even in sparse networks, as opposed to the majority of previous approaches that achieve the same accuracy at higher network densities.

Two interesting ideas appeared in [134] and later in [135]. While these works address target tracking in sensor networks, the estimation of target position is closely related to localization. Sequential Bayesian filtering is used to update, at each time, the posterior joint density of the target’s coordinates. Confidence volumes may be easily determined by density contours, as was done in [135], or more generally by integration of the posterior density. The sequential Bayesian filtering approach in [135] requires the one-step state transition probability density function for the target’s position, as well as the conditional joint density of the observations given the target’s state. The first density requires an accurate model for target motion, and also a stochastic characterization model noise. The second density requires a stochastic model relating the target’s position to received measurements (received power, etc.), and this would require stochastic characterization of the physical medium which corrupts the measurement.

HIGHLIGHTS

Among the several proposal for localization in wireless sensor networks, it is increasingly clear that range-based approaches are able to provide a reasonable localization of a node in terms of error and of energy consumption. The challenges now are mainly to provide a increasingly accurate determination of range, and possibly angle of arrival measurements, and new methods for limiting the propagation of the errors.

6. Wireless Sensor Networks with Mobile Nodes

6.1. Introduction

From a communication perspective, WSNs are expected to be deployed as support (transport) networks in that they allow movement of data to/from the sensors from/to the sinks. For instance, data collected by a node may need to be conveyed to a number of infrastructures such as monitoring collection points, public networks, the Internet, etc., located at the *periphery* of the networked area for reliable storage and more sophisticated elaboration. Conversely, data and code can be sent from designated (query and tasking) centers to relevant sensor nodes.

The main objective of a sensor network is to support in (a) efficient and robust data transport, and (b) uninterrupted coverage of the geographic area in which the sensors have been placed arbitrarily. Given that the sensors are irreplaceable, protocols that implement (a) and (b) should be designed so that important WSN performance parameters are optimized. These parameters include: Extended network lifetime, energy consumption, data throughput, the amount of data to be collected, data fidelity and security, and data transfer delay (latency).

Most of the research on data delivery in WSN concerns networks whose nodes do not move and are irreplaceable. The sensed data is delivered to static sinks. In this scenario, it has been observed that the nodes closer to the sinks have their energy drained from data transmission more than all the others. These nodes relay data for all the other nodes in the network as well as possible packets from the sinks to the sensors. As a result, sinks are soon disconnected from the rest of the network, which determines the end of the network functionalities (network lifetime)

A trend of the research on data dissemination in WSNs has recently started trying to exploit the mobility of some of the network components in order to facilitate the delivery of the sensed data to the sinks and to enhance system's performance. For instance, some mobile agents (such as robots or mobile sinks) can move over the area covered by the static sensor nodes for many purposes, as collecting data from the peripheral sensors, replacing exhausted nodes, synchronizing the nodes, or, in general, increasing the energy efficiency and the network lifetime.

Also when sensor nodes are, in themselves, not expected to be mobile, mobility can still arise from the fact that the sensor nodes are located either in an environment that makes them move over time (e.g., underwater sensor networks), or onto moving users or objects (e.g. buses, pedestrian users and so on.)

The applications for WSNs with mobile autonomous nodes are multifold. Here are just very few examples showing why mobility has to be accounted for in wireless sensor networks.

- Wireless networks of mobile sensors, that can be dispersed in a quasi-random fashion, *e.g.*, from airplanes, intended to extend the ability of data collection, monitoring, and control of the physical environment from remote locations. For instance, submerging a network of sensors in an ocean bed to detect debris of plane crashes for recovery and identification purposes.

- Networks in support of mobile sensing/measuring devices, such as small-scale robot squads. While performing their tasks, the robots are enabled to exchange information among each other and/or to transmit the collected measures to final collection/information centers at the periphery of the inspected area.

- To collect information about the location of a user/piece of equipment, etc., in networks of small, mobile radio transmitters that enable wireless connection among heterogeneous devices (cellular telephones and laptops, printers and PDAs, etc.) which can be placed arbitrarily by the user at home, office, etc.

This section deals with Wireless Sensor Networks with Mobile Nodes. The main advantages of using mobile nodes are:

- Less number of nodes to cover the same area.
- Dynamic adaptation with the environment triggers or changes.
- Dynamic change of the topology to optimize communications in the network.

Mobile nodes require specific paradigms and algorithms. Thus, localization of the mobile nodes is an important issue. Furthermore, in order to exploit the use of these nodes, algorithms for planning the motion of the nodes could be needed.

In the following sections, we will better characterize the WSNs with mobile nodes. Furthermore, we will provide an overview of the algorithms regarding the most interesting aspects of WSNs with mobile nodes.

6.2. Types of mobile nodes and networks

The above section introduces mobile nodes without discussing the nature of these nodes. In practice, these mobile nodes may consist of static sensor nodes installed in a suitable mobile object. These objects can be people, animals, vehicles or robots.

In fact one person can carry a mobile node and eventually additional hardware for communications and data storage.

As far as the motion is concerned, the mobile nodes can be classified in:

- Nodes with uncontrolled and non predictable motion;
- Nodes with controlled and predictable motion.

The first case corresponds to nodes that are moving in a scenario without explicitly considering their role as mobile nodes of the network (e.g., carried out by animals or persons). The motion can be considered random, and the time required for approaching a given static node can be considered as a random variable too. Obviously, this time depends on the density of mobile and static nodes and the motion characteristics.

In the second one, the motion of the nodes can be controlled.

On the other hand, there are two different types of wireless sensor networks with mobile nodes:

- **Static sensor networks with mobile nodes**: this type of sensor network is composed by a large number of static nodes and few mobile nodes. It has the characteristics of a static sensor network, plus some other advantages that come from the use of mobile nodes: sensor calibration, reprogramming nodes capability, coverage extension, dynamic density, etc.

- **Mobile sensor network**. This sensor network consists of a number of mobile nodes. A special interest case arises when the nodes are autonomous objects, i.e. autonomous ground, aerial or underwater vehicles. This type of networks are also referred as **Wireless Sensor Networks with Autonomous Mobile Nodes**.

In the following sections, the characteristics of static sensor networks with mobile nodes and wireless sensor networks with autonomous nodes will be considered.

6.3. Static sensor networks with mobile nodes

The use of mobile nodes in sensor networks increases the capabilities of the network and allows dynamic adaptation with the changes of the environment. The different applications of mobile nodes are the following:

- **Collecting and storing sensor data in sensor networks**: mobile nodes are used to deploy networks with conventional nodes having short range communications. The mobile nodes can approach to distant static nodes to collect sensor data and store them. In this case, the mobile nodes should have enough storing capabilities to collect sequentially the information of the static nodes while moving to other nodes or to the base station (which could be provided of wired connections). Furthermore, notice that storage capabilities are also required in the static nodes to wait for the visit of the mobile node. Alternatively, mobile nodes may be used to alleviate the power consumption due to multi-hop data forwarding. The disadvantage is the latency required to collect data in distant locations when the mobile node has to move sequentially to approach the static nodes. Then, in general, the solution could be applied only in case of delay tolerant scenarios.

- **Sensor calibration**: static nodes could be calibrated or re-calibrated for a particular application using mobile nodes with different and eventually more accurate sensors. An example could be the calibration of temperature sensors using a mobile platform with an infrared camera or viceversa.

- **Reprogramming nodes**: the same can be done reprogramming “by air” static nodes for a particular application. The program or parameters can be downloaded to these nodes from a mobile node at a convenient distance. The functionality of the static nodes could be changed depending on the environment or any other factor. The sensor network will have more capability of adaptation increasing its utility.

- **Network repairing**: when the static nodes are failing to sense and/or to communicate, the mobile node could move to the static node location to replace it when required. Additionally, a robot could deploy some static nodes to fix the network and achieve a better communication between the static nodes. This could also be used for a particular application as it will be described later. On the other hand, a mobile node could be used as a gateway to communicate different parts of the sensor network that are far enough to be communicated directly by the static nodes.

6.3.1. Nodes with uncontrolled and non predictable motion

Considering mobility as a *blessing* rather than a curse for network performance has been widely discussed for ad hoc and sensor networks in different contexts [136–142]. The primary objective of these works is to deliver messages in disconnected ad hoc networks. Examples include wireless sensor networks for environment monitoring or for traffic monitoring.

The work by Chatzigiannakis et al. [137] explores the possibility of using the coordinated motion of a small number of users in the network to achieve efficient communication between any pair of other mobile nodes. A fraction of the network nodes acts as forwarding agents carrying packets for other nodes: the packet is exchanged when the source node and the agent are neighbors (i.e., in the radio vicinity of each other), and it is then delivered to the intended destination when the agent passes by it.

This basic idea has been introduced to WSNs by Shah et al. in their works on *data mules* [143]. Mobile nodes in the sensor field, called mules, are used as forwarding agents. The idea here is to save energy by having single-hop communication (from a sensor to the mule that is passing by) instead of the more expensive multi-hop routing (from the sensor to the sink): it is the mule that will eventually take the sensed data to the sink. The data mule architecture is effective for energy conservation in delay tolerant networks. Energy is traded off for latency, i.e., the energy needed to communicate a packet to the sink is decreased at the cost of waiting for a mule to pass nearby (and at the cost of waiting for the mule to move to the vicinity of a sink).

This approach has been further investigated by Kim et al. [144] which propose a dissemination protocol in which a tree-like communication structure is built and maintained and mobile sinks access the tree from specified sensor nodes in the tree (access nodes). The protocol, termed SEAD (Scalable Energy–Efficient Asynchronous Dissemination), demonstrates via simulation the effectiveness of deploying mobile sinks for energy saving as opposed to keeping the sinks static. SEAD is shown to be more effective for conserving energy than other solutions for data dissemination in wireless sensor networks such as directed diffusion [145], TTDD [146] and ADMR [147].

Common to all these works is that the mobility of the sink is unpredictable and *uncontrollable*. For example, in [144] sinks move according to the random waypoint model.

6.3.2. Nodes with controlled or predictable motion

On the other hand mobile nodes can use the static nodes to locate themselves or even to follow a path. The use of mobile sinks with *predictable*, or even deterministic mobility has been more recently proposed in [148–150] and [151]. In these works the sinks (airplanes) fly over the sensor field and gather the sensed data periodically. While the movement of the sink is fully controllable, it is external to the network infrastructure, i.e., the trajectories are not determined by the network components and activity. The main contribution of these papers concerns the energy-efficient transmission to the passing sink [148, 149], but no implementations with real vehicles have been provided. In [150] the authors consider heterogeneous sensor networks made up of two types of nodes, and determine the densities of each type and the battery energy needed to achieve a given network lifetime.

Inherent patterns of the sink movement are exploited in [152] for the design of robust and energy-efficient routing. This paper assumes that there is a certain degree of predictability in the sink movement, such as the routine route of a ranger patrolling a forest. Based on statistics and distributed reinforcement learning techniques, the sensor nodes learn about the sink whereabouts at given times and use this information to find routes to the mobile sink.

A model for sink movement is proposed in [151], where *observers* (i.e., the sinks) move along the same path repeatedly. The sensed data are collected while the observer traverses the network. When passing by sensor nodes, the observer wakes them up and receives their data (if any). The authors describe a prototype system developed at Rice University where the observers are carried by campus shuttles, and the sensors are spread out throughout the university property. In particular, the authors determine the transmission range needed to collect data from a predefined percentage of the sensor nodes, given the observer speed, the time required to transmit a piece of information, and the traffic pattern. The correlation among the various system parameters is investigated analytically.

An in-depth discussion on the advantages of incorporating *controllable* mobile components into the network infrastructure has been presented in [153]. The authors present an implementation of a sensor network with an autonomous mobile router (a robot) that visits the (static) sensors, collects their data, and reports them to the sink. The idea of collecting data in a single-hop fashion (i.e., when the robot approaches a sensor) is similar to that of data mules. The key difference here is that the motion of the robot is *controlled*: The movement of the robot adapts to data collection performance parameters, and it is determined by the network application priorities. The robot is part of the system. The testbed-based experimental results in this paper concern the evaluation of methods for controlling the speed of the robot for optimizing data collection. The robot traverses networks with different densities following a straight trail and collects the data that are then brought to the sink.

The controlled mobility of the sensor nodes (rather than the mobility of an autonomous router or of the sink) is explored in [154]. The idea here is to have the sensors move into positions that minimize the energy cost of reporting streams of data to the sink.

The problem of reducing the energy consumption and of maximizing the lifetime of a sensor network by exploiting controllable sink mobility within the network has been tackled with in [155] and, more recently, in [156]. In these two works, it is the sink that moves directly among the (static) sensor nodes and, while sojourning at given locations, collects data that are sent to it via multi-hop (ad hoc) routing.

The first work is mostly concerned with energy minimization. The authors present an ILP (Integer Linear Programming) model to determine the locations of multiple sinks and the routes from the sensors to the sinks. Time is divided into rounds. At the beginning of each round information on the nodes' residual energy is centrally gathered and the ILP problem is solved to determine new, feasible locations the sinks should travel to for minimizing the maximum energy consumption spent at the nodes during that round. Minimizing the energy consumption yields to increased network longevity. No constraints are enforced on the sink movements and successive location, and there is no relation between the number of the sinks and their position in subsequent rounds.

The problem of network lifetime maximization through controlled sink mobility is explicitly addressed in [156] for networks with a single sink. Via a new LP formulation, both sink locations and sink sojourn times at those locations are determined that maximize the network lifetime. The experiments performed in the paper refer to scenarios where $n = L^2$ nodes are arranged in a $L \times L$ grid. The sink can visit each node at its location. Improvements on network longevity are obtained

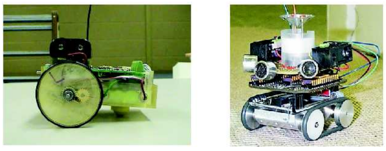


Figure 2: MICAbot [158] (Univ. of Notre Dame) left, Millibots [161] (Carnegie Mellon) right

that are almost five-fold when the sink sojourns at the nodes located at the four corner areas and in the central area of the grid.

6.4. Wireless Sensor Networks with Autonomous Mobile Nodes

In this section, WSN with mobile nodes will be treated in the particular case when the mobile nodes are robotic. However, the same algorithms can be useful for people as mobile nodes whether a Human Machine Interface (HMI) is used. An example of a HMI between a person and a sensor network is the interaction device *Flashlight* [157], used to guide a person with the information acquired from a sensor network. The device uses a pager vibrator and a led to guide the person, i.e., when the device is in the right direction the vibrator and the led turn on. Finally, other types of HMI could be used: for example a graphical interface, where an arrow on a map displayed in a Personnel Device Assistants (PDAs), shows somebody the direction to follow.

The robotic nodes of the sensor network should have some autonomous navigation capabilities such as following a path or finding a sensor source from the sensors signals.

The most popular robotic mobile nodes are low cost and small size nodes based in *Xbow* products, like the sensor node MICA2. These mobile nodes are usually applied in indoor environments.

Several types of robotic mobile nodes can be found in the literature:

- MICAbot [158]
- CostBots [159]
- Robomote [160]
- Millibots [161]

Figures 2, 3 and 4 show some of these robots.

In some applications, it is possible to use **autonomous vehicles** (usually developed for outdoor environments) with more powerful locomotion capabilities, that use a sensor node on-board to become part of the sensor network. For example, an autonomous helicopter has been integrated with a sensor network (see Figure 5). The helicopter is applied for the deployment and repairing of the sensor network [162].

It is also possible to combine autonomous nodes with other mobile nodes, like persons, animals or vehicles, in the same network.

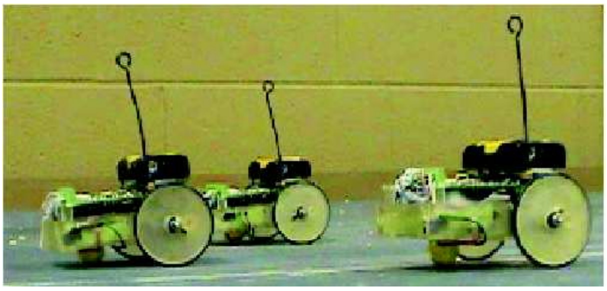


Figure 3: Group of MICAbots [158] (Univ. of Notre Dame)

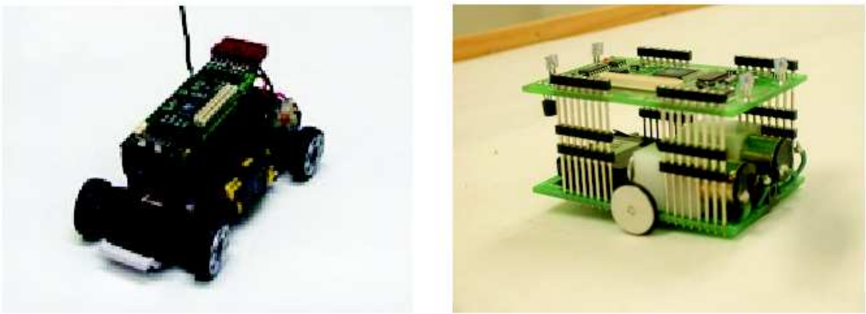


Figure 4: CostBot [159] (Berkeley) left, Robomote [160] (USC) right

Figure 5: Deployment and repair of a sensor network [162]

6.5. Algorithms

6.5.1. Localization algorithms

As it was described in Section 5.7, the localization of the nodes in a sensor network can be solved with two classes of algorithms: range-free and range-based localization. Most of the algorithms that will be presented are range-free and they are based in the use of a global positioning system such as GPS by some mobile nodes. It is not possible to use a GPS with every node of the sensor network due to the high cost and energy consumption of the GPS.

However, the use of a GPS in few mobile nodes, i.e. carried by people, animals or vehicles, makes the solution feasible in cost and energy. If the mobile nodes were autonomous nodes they could always recharge their batteries somewhere.

The next algorithms [163] compute an estimate $\hat{c}_i$ of the position of a static node from the positions $p_{ij} = (x_j, y_j)$ transmitted by the mobile nodes with a strength s_j. These algorithms could be based on the following principles:

- The static nodes take the strongest received message so far, as the best estimate of node position.

$$\begin{aligned} &\text{if } s_j > s_{max} \text{ then} \\ &\quad s_{max} = s_j \\ &\quad \hat{c} = p_j \end{aligned} \tag{1}$$

- The node takes the mean of the received positions as the computed position (see Section 5.7.1.
- The mean of the received positions is weighted with the signal strength, and the estimated position is:

$$\hat{c}_i = \frac{\sum_j s_j p_j}{\sum_j p_j} \tag{2}$$

- The median of the received positions as the estimate:

$$\hat{c}_i = \text{median}(p_{1...j}) \tag{3}$$

- Each received position is a constraint on the node position which is considered to lie within the rectangular region Q. At each step, the node is constrained to lie in the intersection of its current region, Q_{i-1}, and a square of side length $2d$ centered on the GPS transmission. The position estimate of the node is taken as the centroid of the region Q_i.

$$Q_i = Q_{i-1} \bigcap [x_i - d, x_i + d] \times [y_i - d, y_i + d] \tag{4}$$

  The parameter d should reflect the size of the radio communication region.

These algorithms do not require inter node communications, so the power consumption and network congestion is reduced. However, the localization accuracy is higher when the inter node communication is used.

6.5.2. Coverage algorithms

Some applications of WSN with mobile nodes require them to disperse throughout their environment. Exploration, surveillance, and security applications all require coverage of large areas. In this section, an algorithm [164] for dispersing a number of mobile nodes into an enclosed space or an open environment is presented. Those algorithms must take into several aspects - maintaining network connectivity, allowing for mobile node and communications failures, and providing an infrastructure for the mobile nodes to maintain their battery charge.

◇ **Uniform dispersion algorithm** This algorithm disperses mobile nodes uniformly throughout their environment. A thorough treatment of this technique is presented in [165]. Physical walls and a maximum dispersion distance between any two mobile nodes of r_{safe} are used as boundary conditions to help prevent the nodes from spreading too thin and fracturing into multiple disconnected components.

The algorithms works by moving each mobile node, away from the vector sum of the positions $p = p_1, \ldots, p_c$ of their c closest neighbors $\mathrm{nbr} = \mathrm{nbr}_1, \ldots, \mathrm{nbr}_c$. The magnitude of the velocity vector that is given to the motor controller is:

$$v = \begin{cases} -\frac{v_{max}}{c r_{safe}} \sum_{i=1}^{c} p_i & |p_i| \leq r_{safe} \\ 0 & |p_i| > r_{safe} \end{cases} \tag{5}$$

where v_{max} is the maximum allowable velocity output by this behavior. This vector directs the active mobile node away from its c nearest neighbors. The drive velocities are:

$$v_{rot} = v\cos(\mathrm{nbr}_i.\mathrm{bearing}),\ v_{trans} = v\sin(\mathrm{nbr}_i.\mathrm{bearing}) \tag{6}$$

where $\mathrm{nbr}_i.\mathrm{bearing}$ is the bearing to nbr_i.

This is a relaxation algorithm; imagine replacing a graph G with its Delaunay triangulation G', and then placing compressed springs between connected mobile nodes. This will tend to expand the network to fill the available space, but once the space is occupied, mobile nodes will position themselves to minimize the energy in the springs. Total group energy is minimized by minimizing local contributions, which happens when all the internode distances are equal. In practice, using the two closest neighbors works very well.

6.5.3. MAC algorithms

Because one of the tightest constraints in a wireless sensor network is the energy consumption, and communication is the highest energy consumption in a sensor node, the MAC algorithms deal with the mobile nodes in a special manner with regard to the static nodes. In a sensor network composed by static nodes, after the configuration of the communication in the data link layer, the waste of energy in messages that are not information messages is going to be very low because the topology of the network will be the same most of the time. Only when a sensor fails, the network will need to be configured again. However, when there are mobile nodes the density of configuration messages grows because the topology of the network changes.

One protocol that deals with mobile nodes is the SMACS-EAR (Self Organizing Medium Access Control for Sensor Networks, Eavesdrop And Register) [1, 2]. SMACS is a protocol that allows to configure the nodes of a sensor network in a distributed way and that has a flat topology for static nodes. However, EAR is an extension of the protocol for mobile nodes that attempts to offer continuous service to the mobile nodes under both mobile and static constraints. The key of the EAR protocol is to configure the communication channel between a static node and a mobile node with the least messages transmitted by the static node. Therefore, most of the energy is consumed by the mobile nodes. This policy is based on the fact that the mobile nodes can have a better autonomy or the chance to recharge their batteries in special places or with the use of alternative energy sources.

With the use of the EAR protocol, the topology is no more flat and the mobile nodes assume full responsibility for making and breaking connections with the static nodes. The algorithm consists in different steps:

1. The static node invites the other nodes to open a new connection (this step is done periodically and even if there are mobile nodes or not).
2. The mobile node responds to the invitation of the static node, when it is going to create a connection with the static node.
3. The static node sends a reply accepting or denying the connection.
4. To break the link, the mobile node only has to send a disconnection message.

Finally, the mobile nodes need to have a registry of neighbors in order to keep a constant record of neighboring activity.

6.5.4. Routing algorithms

Because the mobile nodes interact with the network, it is possible that they become involved in the routing paths calculated at the network layer. For mobile nodes working as information sources, such as data collectors, routing is not an issue since the only goal is to place the information on the network, transmitting it to the static nodes and allowing them to route the information to the required destinations. If the mobile node is a sink of the sensor network, but its speed is low enough, the routing trees could be calculated with the movement of the mobile node. To save some energy, if the mobile node moves a short distance, only the nodes around the mobile one will recompute the routing trees. However, when the mobile node moves large distances, all the nodes will recompute the routing trees.

There are also some routing algorithms which are specially designed for sensor networks with mobile nodes [166]. Relevant algorithms are SPIN and TTDD, already presented in Section 5.4 and briefly reported in the following, for the reader convenience.

- SPIN Sensor Protocols for Information via Negotiation): the SPIN family of protocols uses data negotiation and resource-adaptive algorithms. Nodes running SPIN assign a high-level name to completely describe their collected data (called meta-data) and perform meta-data negotiations before any data is transmitted. SPIN is a three stage protocol as sensor nodes use three types of messages (ADV, REQ and DATA) to communicate. ADV is used to advertise new data, REQ to request data and DATA is the actual message itself. The protocol starts when a SPIN node obtains new data that it is willing to share. It does so by broadcasting an ADV message containing meta-data. If a neighbor is interested in the data, it sends a REQ message for the DATA and finally the DATA is sent to this neighbor node. The neighbor sensor node then repeats this process with its neighbors. As a result, the entire sensor area (interested in that information) will receive a copy of the data. On the other hand, these protocols are well-suited for an environment where the sensors are mobile because their forwarding decisions are based on local neighborhood information.
- TTDD (Two-Tier Data Dissemination): this protocol provides data delivery to multiple mobile nodes working as base stations. In TTDD, each data source proactively builds a grid structure which is used to disseminate data to the mobile sinks by assuming that sensor nodes are stationary and location-aware, whereas sinks may change their locations dynamically. Using the grid, a base-station can flood a query, which will be forwarded to the nearest dissemination point (a node that stores the source information) in the local cell to receive data. Then, the query is forwarded along other dissemination points upstream to the source. The requested data then flows down in the reverse path to the sink. Trajectory forwarding is employed as the base station moves in the sensor field. Although TTDD is an efficient routing approach, there are some concerns about how the algorithm obtains location information, which is required to set up the grid structure.

6.5.5. Mobile nodes planning algorithms

In the following section, the predictable and controlled “ad-hoc” motion of the mobile nodes is considered. Different techniques can be applied to obtain the required paths and trajectories to collect and store the information of the static nodes of the network. Path planning techniques used in robotics or well known techniques from the operational research field can be applied.

This section will be mainly devoted to the algorithms that compute the optimal path to visit a group of static nodes by using a team of mobile nodes (this problem is known as the *multi Travelling Salesmen Problem*, often referred as *m-TSP*). It is an instance of the *Optimal Assignment Problem* (OAP) [167], which is a well-known problem that was originally studied in game theory and then in operations research, in the context of personnel assignment.

Problem statement. Given m mobile nodes, each visiting one static node (a more general case will be studied later) and n prioritized static nodes, each requiring one mobile node to visit it for a given purpose (collect information, share energy, etc.). Also, given for each mobile node a nonnegative parameter (i.e., utility estimate) that predicts its performance to visit a mobile node; if a mobile node is incapable of undertaking a task (visit a static node or perform a required operation after reaching a static node), then the mobile node is assigned a rating of zero for that task. The goal is to assign mobile nodes to static nodes so as to maximize overall expected performance, taking into account the priorities of the static nodes and the skill ratings of the mobile nodes.

This problem can be cast as an integral linear program [167]: find mn nonnegative integers α_{ij} that maximize

$$U = \sum_{i=1}^{m} \sum_{j=1}^{n} \alpha_{ij} U_{ij} w_j \tag{7}$$

subject to:

$$\begin{array}{l} \sum_{i=1}^{m} \alpha_{ij} = 1, \quad 1 \leq j \leq n \\ \sum_{j=1}^{n} \alpha_{ij} = 1, \quad 1 \leq i \leq m \end{array} \tag{8}$$

The sum (7) is the overall system utility (note that since α_{ij} are integers, they must all be either 0 or 1). Given an optimal solution to this problem (i.e., a set of integers α_{ij} that maximizes (7) subject to (8)), an optimal assignment is constructed by assigning mobile node i to static node j only when $\alpha_{ij} = 1$.

If the mobile nodes' utilities can be collected at one machine, then a centralized linear programming approach (e.g., Kuhn's Hungarian method [168]) will find the optimal allocation in $\mathcal{O}(mn^2)$ time. Alternatively, a distributed auction-based approach (e.g., Bertsekas's Auction algorithm [169]) will find the optimal allocation, usually requiring time proportional to the maximum utility and inversely proportional to the minimum bidding increment. The two approaches (i.e., centralized and distributed) represent a tradeoff between solution time and communication overhead. To implement a centralized assignment algorithm, n^2 messages are required to transmit the utility of each mobile node for each visit; an auction-based solution usually requires far fewer (sometimes fewer than n) messages to reach equilibrium. Moreover, the time required to transmit a message cannot be ignored, especially in wireless networks, which can induce significant latency.

In the previous problem statement, it has been assumed that it should be assigned at most one static node to each mobile node. When the system consists of more static nodes than mobile nodes, this problem is one of building a time-extended *schedule* of tasks for each robot, with the goal of minimizing total weighted cost. Using Brucker's terminology [170], this problem is an instance of the class of scheduling problems

$$M || \sum w_j C_j \tag{9}$$

That is, the mobile nodes execute tasks in parallel (M) and the optimization criterion is the weighted sum of execution costs ($\sum w_j C_j$). Problems in this class are strongly $\mathcal{NP}$-hard [171]. Even for relatively small problems, the exponential space of possible schedules precludes enumerative solutions.

A mean of treating these problems is to ignore the time-extended component. For example, given m mobile nodes and n static nodes, the following approximation algorithm can be used:

1. Optimally solve the initial $m \times n$ assignment problem.
2. Use the Greedy algorithm to assign the remaining tasks in an online fashion, as the robots become available.

The performance of this algorithm is bounded below by the normal Greedy algorithm, which is 3-competitive for online assignment. The more nodes that are assigned in the first step, the better this algorithm will perform. As the difference between the number of mobile and static nodes that are initially presented decreases, performance approaches optimality.

Another way to approach this problem is to employ an iterative task allocation system, such as [172] price-based market. The mobile nodes would opportunistically exchange static nodes to visit over time, thereby modifying their schedules.

6.5.6. Mobile nodes reactive algorithms

It could be also possible to apply reactive techniques used in robotics to guide the mobile node. In this case, the mobile platform moves reacting to sensorial stimulus of the environment. There are several techniques that have been used in wireless sensor networks such as:

- The diffusion-based path planning [173] that is applied when it is known that the quantities of interest in the system are generated via a diffusion process. This algorithm assumes that a network of mobile sensors can be commanded to collect samples of the distribution of interest. These samples are then used as constraints for a predictive model of the process. The predicted distribution from the model is then used to determine new sampling locations.
- Random walk algorithm using gradient descent [160]: this easy to implement algorithm is applied to guide the mobile node to a focus of interest. The algorithm proceeds as follows:
    1. Record the current sense value reading from the sensor $P(i)$.
    2. Move along a straight line for a constant distance.
    3. Record again the current sensor intensity reading from the sensor $P(i+1)$.
    4. If the difference $P(i+1) - P(i) > 0$, move along the same direction.
    5. If the goal is reached then stop.
    6. If the goal is not reached then rotate by a random angle and go back to step 1.

6.5.7. Network repairing algorithm

As it was described before, the use of mobile nodes allows to repair the network connectivity of a sensor network [162]. The use of mobile nodes could be important to increase the fault tolerance in a wireless sensor network.

Assumed that a manual or automatic network deployment is executed, the connectivity repair algorithm could consist on two phases:

- The mobile node measures the connection topology of the deployed network and compares it to the desired topology. If they match, none deployment is done. This phase can be run at any time, with the objective to detect the potential failure of sensor nodes and ensure sustained connectivity.

- Otherwise, the measured connectivity graph is used to compute new deployment locations that will repair the desired topology. These deployment locations can be represented as a set of waypoints.

In a real implementation [162], a simplified version of this procedure has been applied. The algorithm computes deployment locations whose connectivity graph is one connected component. Therefore, the task of the connectivity repair algorithm is to determine the number of connected components in the deployed network. The algorithm that determines the number of connected components works as follows:

1. Each static sensor broadcasts its identification number and forwards the identification numbers that it hears.
2. Each sensor keeps the largest value it has heard. The number of different values is the number of connected components in the graph.
3. The mobile node can collect this information and determines how many components there are.
4. If the network has at least two connected components, it can compute the separation region and determines how to cover it with waypoints in a way that connects the two components.

6.6. Critical issues and future research

In wireless sensor networks with mobile nodes, it is not necessary (relying on direct or multi-hop connections) to transmit directly information from the static nodes to the sink. Indeed, mobile nodes can relay the information among the static nodes and the sinks, also in absence of permanent wireless connectivity among them. The static nodes could be organized in clusters and the mobile nodes could approach to the clusters and transmit the information to others sinks or to the main sink. In the same way, there are advantages in the collection of the information, and advantages for the diffusion of the information because the path of the information is the same but with the opposite direction. It should be pointed out that in this architecture there could be some delay problems. These problems could be solved using more mobile nodes or mobile nodes with larger communication ranges. A higher energy consumption is the drawback in this case, but as it has been mentioned before, a mobile node can always recharge its batteries. Furthermore, the use of mobile nodes allows energy savings in the static nodes. For example, a mobile node can move near the locations of the static nodes reducing their energy consumption due to communications.

Mobile nodes can improve or repair the communications in the sensor network. Furthermore, they can be used to calibrate the sensors of the static nodes. On the other hand, mobile nodes can offer a better coverage among subnetworks of static nodes. A mobile node with better communication capabilities can even be used as a gateway between the sensor network and another type of network such as Internet, cell phone net, etc.

Several main trends have been identified in the literature about WSN with mobile nodes:

- Energy constraints: concerning WSN, energy constraints is one of the most relevant issues. In WSN with mobile nodes, this topic includes for example the study of the impact of mobile nodes in the energy consumption of the entire network.

- Motion planning: an important trend is related to the study of how the sensor network can compute, in a distribute way, the path that the mobile node must follow. Also, this path can be updated depending on changes of the environment or using new data collected by the sensor network.
- Cooperative perception: the sensor network can be considered an extension of the sensorial capabilities of a mobile node, and therefore, an improved model of the environment can be built with the information from the WSN for navigation purposes.

Finally, it should be pointed out that WSN with mobile nodes is a relatively new topic and it is still a non-mature field. The following topics have been found in the literature about WSN with mobile nodes: localization, communications (MAC and routing), planning and reactivity. However, more algorithms and theoretical studies are needed in those fields. Furthermore, there are some topics that have been poorly addressed, such as self-hierarchical organization and/or clustering techniques, fault tolerance and coverage techniques. Therefore, more efforts should be devoted to those aspects and their study should be promoted in next years. The following points can be suggested as relevant research trends in the next future:

- Energy considerations in: static and mobile trade-off, computation of optimal parameters (speed, delay, etc) and relation between data collection and data diffusion.
- Motion planning, taking into account communication constraints and the relation with self-hierarchical organization and/or clustering techniques.
- Cooperative perception algorithms to exploit sensorial information from the nodes of the network.
- New reliability and fault tolerant concepts.
- Integrate and exploit heterogeneity in mobile nodes.
- More field experiments with WSN and mobile nodes.

7. Autonomous robotic teams for surveillance and monitoring

7.1. Introduction

Although most mobile robotic systems involve a single robot operating alone in its environment, a number of works have considered the problems and potential advantages involved in having an environment inhabited by a group of robots which cooperate in order to complete a task. There are several reasons which could lead to the development of a multirobot system. The main reason may be that it is possible to build more robust and reliable systems by combining unreliable but redundant components. Performance is another important advantage (many hands make light work) which becomes critical in dangerous environments over a broad area such as forest fires, contaminated areas, etc. In such an environment, a robotic team could provide several services: surveillance, searching, detection, tracking, monitoring, measurement, etc. For example, The *COMETS Project* of the European Commission (IST-2001-34304) on multiple heterogeneous unmanned aerial vehicles considers the coordination and control of multiple heterogeneous UAVs for applications such as detection and monitoring (see Figure 6).

Successes in the field over the last decade seem to have shown the feasibility, effectiveness and advantages of a multirobot system for some specific robotic tasks: exploring an unknown planet [174], pushing objects [175, 176], or cleaning up toxic waste [177].

In the literature, there are not many formal models for multi-robot coordination, because research has mainly covered the construction and validation of working systems. In next section, several taxonomies (see [178] and [179]) that categorizes the bulk of existing multirobot systems along various axes are presented. Those axes include team organization (e.g., centralized vs. distributed), communication topology (e.g., broadcast vs. unicast), and team composition (e.g., homogeneous vs. heterogeneous).

7.2. A taxonomy of multirobot systems

A key “difficulty” in the design of multirobot systems is the size and complexity of the space of possible designs. In order to make principled design decisions, an understanding of the many possible system configurations is essential. Several taxonomies have been proposed providing the following aspects:

- Defining key “features” that help to identify different multirobot systems in a precise and complete way.
- Allowing to state relations, analysis and formal proofs for the groups derived from the taxonomy.
- Making it easier to compare different systems in a common and simple framework.
- Describing the extent of the space of possible designs for a multirobot system; in this way, during the development of a new system, it is possible to identify problems previously solved.

Different taxonomies are possible depending on the selection of the main axes and the subset of systems studied (i.e., a team of mobile ground robots). In the following, several taxonomies that categorize the bulk of existing multirobot systems are presented:

Figure 6: Aerial robotics team

Yuta and Premvuti [180] The following classification for multiple autonomous robot systems is proposed:

- The degenerate case – A Single Robot System.
- There is common objective or mission to be performed, and the robots work toward their purpose together – A Common Objective System.
- Each robot has its own objective or mission, but there is some interference between robots during the execution of missions – An Individual Objective System.

In case of a system with a single robot, the robot does not need to consider the idea of others or have any thoughts for a community. Suppose there are some movable objects travelling in the robots world, these objects are merely obstacles each with its own purpose.

Cao [178] This taxonomy, presented in [178] in the framework of cooperative mobile robotics, classifies different works taking into account the *cooperative behaviour* of the system, which is a subclass of collective behaviour that is characterized by cooperation. Main axes in this survey are group architecture, resource conflicts, origin of cooperation, learning and geometric problems. Within group architecture, issues such as centralization/decentralization, differentiation and communication structures are considered. Regarding resource conflict, it can arise due to physical objects, communication channels or space sharing. Spatial coordination problems have been traditionally solved by using multirobot motion planning techniques, which take into account critical issues such as deadlock and collision avoidance. Within geometric problems, formation and marching problems are also considered.

Balch [181] There are many cases, especially in multirobot systems, where task and reward should be treated separately in the framework of reinforcement learning. Autonomous agents embedded in their environment are not always able to accurately access their performance, and overall performance may also depend on other agents over which the learner has no direct control. Taxonomies of task and reward are presented, providing a framework for investigating the impact of differences in the performance metric and reward on multirobot system performance. Taxonomies of multirobot tasks are based on six axes (time, criteria, subject of action, resource limits, group movement and platform capabilities), whereas five axes are provided for rewards (source of reward, relation to performance, time, continuity and locality). In both cases, only certain values are valid for each axis.

Ali [182] This research compares the performance of different classes of mobile behavior-based multiagent telerobotics systems in relation to the kinds of tasks they are performing. The systems are compared in terms of safety, effectiveness, and ease-of-use, for applications representing classes of tasks in a newly-developed taxonomy of mobile multiagent tasks. This taxonomy categorizes the tasks in terms of the relative motion of the agents. Four different task classifications from this taxonomy were studied.

Todt [183] This taxonomy is restricted to the framework of robot motion coordination and provides five axis for the classification: coupled/decoupled coordination, coordination time, existence of coordination priorities, coordination cost evaluation and workspace representation. Not only mobile robots, but also systems composed by several manipulators are considered. Several conclusions and trends are derived from the taxonomy. For example, an evolution in the degree of abstraction of the coordination problem along the time is identified. The first works deal with the problem in the physical space, and then the formulation of the coordination problem moves to a more appropriate form. A simplification in the complexity of the methods and a trend towards decoupled methods is also identified.

Dudek [179] This classification can be applied to any multirobot system and the so called *natural dimensions* form its basis. Those parameters are related to the properties of the group of robots as a collective. Seven main axes are defined, covering several aspects of the system: the size of the collective, different parameters related to the communication system (bandwidth, range, topology, etc.), collective reconfigurability, processing capabilities of each element, and the composition of the group (in terms of hardware and software homogeneity). Within each axis, a bounded set of values is considered. By using this taxonomy, the authors classify some classical problems in multirobot systems and several existing architectures. Furthermore, for certain tasks, formal proofs are provided to show higher performances for a collective when comparing to individual robots.

In Table 1, a summary of those multirobot taxonomies is presented.

It is clear that many classifications are possible, with many different features, depending on the domain and the purpose of the system. However, there are several aspects that are common in many taxonomies:

- Differentiation between the robots in the system. A group of robots is defined *homogeneous* if the capabilities of the individual robots are identical, and *heterogeneous* otherwise. In general, heterogeneity introduces complexity since task allocation becomes more difficult, and robots have a greater need to model other individuals in the group. In general, the realities of individual

Taxonomy	Domain	Number of axes	Description
Yuta and Premvuti	Multirobot	2	Derived from objectives and decision mechanisms.
Cao	Cooperative mobile robotics	5	Based on problems and solutions related to the cooperation mechanisms.
Balch	Tasks and rewards	6/5	Useful in systems with reinforcement learning.
Ali	Mobile multiagent telerobotics systems	3	Based on the relative motion of the robots.
Todt	Multirobot	5	Limited to multirobot motion planning.
Dudek	Multirobot	7	Based on characteristics of the group of robots.

Table 1: Multirobot taxonomies.

Figure 7: Ground Robotic Teams [160] (left), Ground and aerial robotics team (right)

robot design, construction and experience will inevitably cause a multirobot system to drift to heterogeneity over time. This tendency has been recognized by experienced roboticists who have seen that several copies of the same model of robot can vary widely in capabilities due to the differences in sensor tuning, calibration, etc. This means that to employ robot teams effectively, it is important to understand diversity and predict how it will impact performance. In fact, differentiation can also be treated as an advantage in terms of complementarity between different components in order to complete a given task or mission.

- Centralized/decentralized approach in the design (see Section 7.3.2).
- Allocation: in multirobot systems, each robot is assumed to be able to perform tasks in response to tasks requests. The issue is to decide which robot should be endowed with each given task to be performed. This requires the capability to assess the interest of providing a given robot with a given task. This is a difficult issue when the decision has to be done taking into account the current individual plans of the robots as well as the tasks left to be assigned. Information needed to perform an optimal choice includes the models, each robot's plan and the current states of tasks execution for each robot.
- Communication between components (from different points of view). Some aspects are commented in Section 7.3.3.
- Motion of the components in a common environment (see Section 7.3.4).

So, it seems that those aspects are relevant and should be taken into account in the design and classification of a multirobot system.

7.3. Paradigms for coordination and cooperation

Coordination is a process that arises within a system when given (either internal or external) resources, they are simultaneously required by several components of this system. In the case of a multirobot system, there are two classic coordination issues to deal with:

- Temporal coordination: can be achieved relying on robots synchronization. Several schemes to enable incremental negotiations related to possible time intervals synchronization can be defined and implemented. As a result, a group of robots acknowledge a common time interval in which the synchronization should occur. Temporal coordination can be necessary in a wide spectrum of applications. For instance, in the case of an event monitoring, several synchronized perceptions of the event are required.
- Spatial coordination: the sharing of space between the different robots to ensure that each robot will be able to perform its plan safely and coherently regarding the plans of the other robots. Interactions models can be considered in order to reason about the interactions requirements within the joint tasks. Afterward, during plan execution, collision avoidance can be safely achieved applying several algorithms on the planned trajectories of robots in a neighborhood.

On the other hand, there are several explicit definitions of cooperation in the robotics literature. Cooperation is defined as a “joint collaborative behavior that is directed toward some goal in which there is a common interest or reward”. Furthermore, in [178] a definition for *cooperative behavior* can be found: given some task specified by a designer, a multiple-robot system displays *cooperative behavior* if, due to some underlying mechanism (i.e., the “mechanism of cooperation”), there is an increase in the total utility of the system.

The amount of research in the field of cooperative mobile robotics has grown substantially in recent years. This work can be broadly categorized into two groups [177]: swarm-type cooperation and intentional cooperation. The swarm-type approach to multi-robot cooperation deals with large numbers of homogeneous robots. This approach is useful for non-time-critical applications involving numerous repetitions of the same activity over a relatively large area. The approach to cooperative control taken in these systems is derived from the fields of neurobiology, ethology, psychophysics, and sociology. The second primary area of research in cooperative control deals with achieving intentional cooperation among a limited number of typically heterogeneous robots performing several distinct tasks. In this type of cooperative system, the robots often have to deal with some sort of efficiency constraint that requires a more directed type of cooperation than is found in the swarm approach described above. Furthermore, this second type of mobile robotic mission usually requires that several distinct tasks be performed. These missions thus usually require a smaller number of possibly heterogeneous mobile robots involved in more purposeful cooperation. Although individual robots in this approach are typically able to perform some useful task on their own, groups of such robots are often able to accomplish missions that no individual robot can accomplish on its own. Key issues in these systems include robustly determining which robot should perform which task (*task allocation*) so as to maximize the efficiency of the team and ensuring the proper coordination among team members to allow them to successfully complete their mission.

7.3.1. Paradigms in the architecture of multirobot systems

Robots individually have their own hardware/software structure, which is usually called robot architecture. Those architectures should take into account the particular characteristics of a multirobot system, because a robot architecture designed for a single robot is not necessarily valid when this robot has to interact with other robots.

Only, one of the taxonomies mentioned in Section 7.2 ([178]) proposes a definition of a multirobot system *group architecture*: it is an element which provides the infrastructure upon which collective behaviors are implemented, and determines the capabilities and limitations of the system. This is just a functional definition, so components of the architecture or integration aspects are not mentioned. In literature, relatively little work has covered those aspects. Research in multirobot systems has focused primarily on construction and validation of working systems, rather than more general analysis. As a result, one can find many architectures for multirobot coordination, but relatively few formal models.

A list of representative working architectures developed during last years could include SWARM [184], ACTRESS [185], CEBOT [186], GOFER [187], ALLIANCE [188], MARTHA [189, 190] and COMETS among others.

7.3.2. Centralized/decentralized architecture

A specific section is devoted to this issue because the most fundamental decision that is made when defining a group architecture is whether the system is centralized or decentralized. A centralized decision configuration (with a minimal distributed supervision) is compatible (at the least) and even complementary with a configuration endowed with fully distributed decision capabilities. Aspects related to decision can be developed either within a central decisional component or between several distributed components (e.g. possibly the different robots within the system). However, several trade-off should be considered, regarding decision:

- **Knowledge's scope and accessibility**: a preliminary requirement, to enable decisional aspects within a central component, is to ensure permanently the availability of (relevant) up-to-date knowledge within this central component. It requires the centralization of any decisional-related knowledge from any component of the system, and to have them updated permanently. However, assuming that this requirement can be fulfilled, it allows to perform more informed decisions, and hence to manage the mission operations in a more efficient way.

  Regarding distributed decision, the local scope of the available knowledge is a double edge issue: as far as the only available knowledge is the knowledge related to the considered component (or close to the considered component), this knowledge is usually far more easy to access and to refresh. But on the other hand, local and partial knowledge lead to decisions that may turn out to be incoherent regarding the whole system.

- **Computational power and scalability**: in a multirobot system, the amount of data to process is quite huge: the processing of this knowledge in a centralized way requires obviously powerful computational means. Moreover, such a centralized computation reaches its limits when the number of robots increases: a centralized system can not be scalable to any number of robot.

  In contrast, a distributed approach within a multirobot system can stay available when increasing the number of robots, since the complexity remains bounded: each robot still only deals with a local partial knowledge of the system, that leads to manipulate local information.

The respective inconveniences of each approach can be mitigated with respectively constraining or extending their framework: a centralized approach will be relevant if

- The computational capabilities are compatible with the amount of information to process,
- The exchange of data meets both the requirements of speed (up-to-date data) and expressivity (quality of information enabling well-informed decision-taking).

As a consequence, one way to help to satisfy these two points is to reduce the complexity of the data exchanged to a minimal level, still satisfying the decision-taking but without over-loading communications. This can be achieved with designing a task's communication protocol meeting this minimal expressivity need, and then with fitting this protocol to the particular relevant field of application.

On the other hand, a decentralized approach will be relevant if

- The available knowledge within each distributed component is sufficient to perform “coherent” decisions,
- This required amount of knowledge does not endow the distributed components with the inconveniences of a centralized system (in terms of computation power and communication bandwidth requirements).

One way to ensure that a minimal global coherence will be satisfied within the whole system is to enable communications between the robots of the system, up to a level that will warranty that the decision is globally coherent.

However, instead of definitely choose one of this extreme configurations, an alternative possibility lie in hybrid solutions, that may fit at the best the requirements of an heterogeneous system.

7.3.3. Communication between components

Network and data link layers in the communications of an autonomous robot team are similar to a MANET (Mobile Ad hoc NETwork). A mobile ad hoc network can be described as a peer-to-peer network which usually comprises tens to hundreds of meters, and aims to form and maintain a connected multihop network capable of transporting large amount of data between nodes. The main goal for a MANET like other conventional wireless networks is providing high QoS and high bandwidth efficiency when mobility exists. In contrast, in a sensor network is necessary to extend its lifetime saving energy. As a consequence, a lower performance in other aspects of operation such as QoS and bandwidth usage is assumed.

Two examples of multihop routing algorithms for MANET are: Ad Hoc On Demand Distance Vector (ADOV) routing and Temporally Ordered Routing Algorithm (TORA). Both are examples of demand-driven systems that eliminate most of the overhead associated with table updating in high mobility scenarios. However, the energy cost during route setup (path discovery) is high, so they are not used in wireless sensor networks. Another algorithm, called Power-Aware Routing, finds the minimum metric paths on two different power metrics: minimum energy per packet, and minimum cost per packet. The first metric is intuitive and produces substantial energy saving while the network retains full connectivity. However, performance degradation due to node/link failure is not accounted for. The minimum cost metric is obtained by weighting the energy consumption by the energy reserve on each node. It has the nice property of delaying failures by steering traffic away from low-energy nodes, but overhead for path maintenance could be high. As a conclusion, it is obvious that different protocols are needed for autonomous robotic teams and wireless sensor networks.

7.3.4. Path planning for multiple robot systems

As it has been mentioned, one of the main issues in multirobot coordination is the *spatial coordination*. In this section, a formal statement of this problem is presented, and main approaches to solve it are summarized.

Let assume multiple robots that share the same world, $\mathcal{W}$. A path must be computed for each one that avoids collisions with obstacles and with other robots. Superscripts will be used in this section to denote different robots. The i^{th} robot will be denoted by $\mathcal{A}^i$. Suppose there are m robots,

$\mathcal{A}^1, \mathcal{A}^2, \ldots, \mathcal{A}^m$. Each robot, $\mathcal{A}^i$, has its associated configuration space, $\mathcal{C}^i$, and its initial and goal configurations, q^i_{init} and q^i_{goal}.

A state space can be defined that considers the configurations of all of the robots simultaneously,

$$X = \mathcal{C}^1 \times \mathcal{C}^2 \times \ldots \times \mathcal{C}^m \tag{10}$$

A state $x \in X$ specifies all robot configurations, and may be expressed as $x = (q^1, q^2, \ldots, q^m)$. Let N denote the dimension of X, which is given by

$$\sum_{i=1}^{m} \dim(\mathcal{C}^i) \tag{11}$$

.

There are two sources of obstacle regions in the state space: 1) *robot-obstacle* collisions, and 2) *robot-robot* collisions. For each i such that $1 \le i \le m$, the subset of X that corresponds to robot $\mathcal{A}^i$ in collision with the obstacle region, $\mathcal{O}$, is defined as

$$X^i_{obs} = \{x \in X | \mathcal{A}^i(q^i) \cap \mathcal{O} \neq \emptyset\} \tag{12}$$

This models the robot-obstacle collisions.

For each pair, $\mathcal{A}^i$ and $\mathcal{A}^j$, of robots, the subset of X that corresponds to $\mathcal{A}^i$ in collision with $\mathcal{A}^j$ is given by

$$X^{ij}_{obs} = \{x \in X | \mathcal{A}^i(q^i) \cap \mathcal{A}^j(q^j) \neq \emptyset\} \tag{13}$$

Both (12) and (13) will be combined in (15) to yield X_{obs} as it will be described now.

Formulation (Multiple-Robot Motion Planning)

1. There are m robots, $\mathcal{A}^1, \ldots, \mathcal{A}^m$, which each may consist of one or more moving bodies.
2. Each robot, $\mathcal{A}^i$, for $1 \le i \le m$ has an associated *configuration space*, $\mathcal{C}^i$.
3. The state space, X, is defined as the Cartesian product

   $$X = \mathcal{C}^1 \times \mathcal{C}^2 \times \ldots \times \mathcal{C}^m \tag{14}$$

   The obstacle region in X is

   $$X_{obs} = (\bigcup_{i=1}^{m} X^i_{obs}) \bigcup (\bigcup_{ij, i \neq j}^{m} X^{ij}_{obs}) \tag{15}$$

   in which X^i_{obs} and X^{ij}_{obs} are the robot-obstacle and robot-robot collision states from (12) and (13), respectively.
4. A state $x_I \in X_{free}$ is designated as the *initial state*, in which $x_I = (q^1_I, \ldots, q^m_I)$. For each i such that $1 \le i \le m$, q^i_I specifies the initial configuration of $\mathcal{A}^i$.

5. A subset $x_G \in X_{free}$ is designated as the *goal state*, in which $x_G = (q_G^1, \ldots, q_G^m)$.
6. The task is to compute a continuous path, $\tau : [0,1] \rightarrow X_{free}$ such that $\tau(0) = x_{init}$ and $\tau(1) \in x_{goal}$.

X can be considered as an ordinary configuration space. The classical planning algorithms for a single robot with multiple bodies [191] may be applied without adaptation in case of centralized planning (planning that takes into account all robots). The main concern, however, is that the dimension of X grows linearly in the number of robots. Complete algorithms require time that is at least exponential in dimension, which makes them unlikely candidates for such problems. Sampling-based algorithms are more likely to scale well in practice when there many robots, but the resulting dimension might still be too high.

The motions of the robots may be decoupled in many interesting ways. This leads to several interesting methods that first develop some kind of partial plan for the robots independently, and then consider the plan interactions to produce a solution. This idea is referred to as decoupled planning. In [191], two approaches are given: (i) *prioritized planning* considers one robot at a time according to a global priority, while (ii) the *path coordination method* essentially plans paths by scheduling the configuration space-time resource.

7.4. Robots using Wireless Sensor Networks

A robot can use a sensor network to expand its capabilities, for example sensing at inaccessible locations using the information from the network. In this case, the sensors of the robot are distributed in the environment, and are not centralized on-board the robot itself. Therefore, the robot has multiple inputs from the same event, so its reactivity and performance is improved. Furthermore, a sensor network can be useful in the robot localization, navigation and tracking.

A sensor network can also be used for the guidance of robots, and as it has been explained previously (see Section 6.3), the same algorithms can be applied for other mobile nodes (such as people) with a suitable interface. This guidance can be based on the following principles:

- Follow the movement of the source to be sensed.
- Extract motion direction from the sensor network.
  - Safe path from the type of danger detected by sensors (temperature, contaminants, etc.).
  - Path to improve sensing of the source (source localization).

Particulary, gradient-based methods can be applied for local reactive navigation. Two different situations can be considered:

- Motion to increase the gradient (increasing perception of the event). In this case, the objective is to detect and sense a source, such as pollution.
- Motion to decrease the gradient (decreasing perception of the event). This case deals with a situation in which a escape path is needed.

Two examples of the use of a sensor network by a robot can be found in [192] and [163]. In the first paper, a robot without a GPS or a map can follow a path and reach a goal position, just using the information received from the sensor network. On the other hand, in the second paper, an autonomous UAV follows a path determined by the sensor network. Furthermore, the sensor network can change the path dynamically according to several environment events.

7.5. Algorithms for navigation of autonomous robots using wireless sensor networks

The navigation of autonomous robots has been studied from a centralized approach: the autonomous vehicle has different sensors (GPS, compass, ultrasonics, etc.) and can localize itself and follow a path using the information provided by those sensors. However, the use of sensor networks by robots opens a new research area which considers the navigation of the autonomous robots using distributed information. In this way, the navigation is possible even without the robot carrying any sensors, exclusively relying on the communications with the wireless sensor network.

7.5.1. Potential field guiding algorithm

A moving object, such as a robot or a person with a suitable interface, is guided across the network along a safe path, away from the type of danger that can be detected by the sensors [193]. Each sensor can sense the presence or absence of such types of danger. A danger configuration protocol runs across all the nodes of the network generating a danger map. In this map, the dangerous areas detected by the sensor network are represented as obstacles. Those obstacles will have repulsing values and the goal will have an attracting value according to some metric.

The danger map is generated by a potential field protocol that works as follows [193]:

- Each node whose sensor triggers “danger” diffuses the information about the danger to its neighbors in a message that includes its source node id, the potential value and the number of hops from the source of the message to the current node.
- When a node receives multiple messages from the same source node, it keeps only the message with the smallest number of hops (the message with the least hops is kept because that message is likely to travel along the shortest path).
- The current node computes the new potential value from the source node. The node then broadcasts a message with its potential value and number of hops to its neighbors.
- After this configuration procedure, nodes may have several potentials from multiple sources. To compute its current danger level information, each node adds all the potentials.

On the other hand, the potential field information stored at each node can be used to guide an object equipped with a sensor that can talk to the network in an on-line fashion. The next algorithm [193] can compute the safest path to the goal:

- The goal node broadcasts a message with the danger degree of the path, which is zero for the goal.

- When a sensor node receives a message, it adds its own potential value to the potential value provided in the message, and broadcasts a message updated with this new potential to its neighbors.
- If the node receives multiple messages, it selects the message with the smallest potential (corresponding to the least danger) and records the sender of the message.

Finally, there is another algorithm [193] with the navigation guiding protocol. In this algorithm the user asks the network for where to go next. The neighboring nodes reply with their current values. The users sensor chooses the best possibility from the returned values. Note that this algorithm requires the “integrated” potential computed by the first two algorithms in order to avoid getting stuck in local minima.

7.5.2. Path computation and following algorithm

Assuming a sensor network deployed with localized nodes, the algorithm presented in [163] can compute the nodes of the sensor network that are within a certain *pathwidth* distance from a path defined by a list of coordinates provided in a broadcast message. Nodes on the path will store the path segment, will rebroadcast the path message and will be activated for robot guidance. The rest of nodes use the knowledge of their location and the location of the sender (contained in the message) to determine if they are close enough to the direction vector pointing to where the path starts. If they are, they forward the message and otherwise, they remain silent.

It is interesting to note that multiple paths can be computed, stored and updated by the network to match multiple robots and goals. Furthermore, a map computation algorithm could be implemented, where the map could be constructed incrementally and adaptively such as an artificial potential field (see Section 7.5.1) using hop-by-hop communication. The “obstacles” could correspond to events and will have repulsing values whereas the goal will have an attracting value. Finally, joining the two algorithms, a distributed motion planning protocol could be computed by the sensor network where different path computation algorithms could be run as distributed protocols on top of the distributed map, updating the path dynamically according to different events in the environment.

On the other hand, this path stored in the sensor network can be used for the navigation of a robot (see Figure 8) which has communications with the nodes of the sensor network. In the same way as the path message is propagated, the process has two phases: first, the robot has to reach the location where the path starts, and then the robot is guided along the path. The first phase has the next steps:

1. One (or all) of the sensors which know that they are near to the start of the path send out a message that contains the location of the start of the path.
2. That message is forwarded throughout the sensor network.
3. The robot sends its location by a message in three different directions (120 degree dispersal angle).

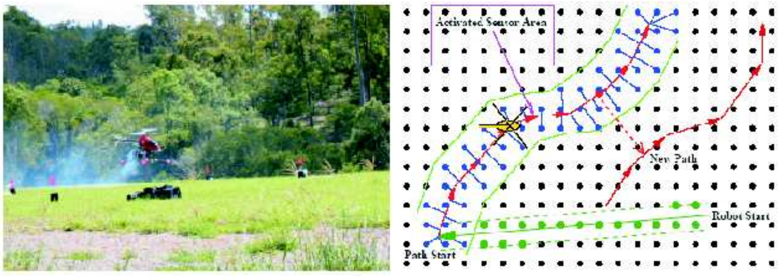


Figure 8: Following a path using a sensor network [163]

4. The sensor node that receives the message from the robot knows the start location, the robot location and the direction which the message came from. Using this information, the node can send a directional message to the robot that directs it to the start point.

After the initialization phase which places the robot on the path, the navigation guidance algorithm is used to control the motion direction of the robot. This algorithm can be summarized as follows [163]:

1. The robot starts by sending out a `QueryOnPath` message which includes the senders identification and location.
2. If it is received by a sensor on the path, this sensor replies with a `QueryAck` message which includes the path section, some consecutive way points, and an indication of where these way points fit into the path.
3. By gathering lists of segments from multiple sensors, the entire path can be assembled piece by piece as the robot moves following the way points in order.

7.5.3. Probabilistic navigation

With this algorithm [192], assuming neither a map nor a GPS are available, the robot can navigate through the environment from point A to point B communicating with a sensor network (see Figure 9).

The algorithm has two stages:

- *Planning*: when the navigation goal is specified (either the robot requests to be guided to a certain place, or a sensor node requires the robot's assistance), the node that is closest to the goal triggers the navigation field computation. During this computation, every node probabilistically determines the optimal direction in which the robot should move when in its vicinity. The computed optimal directions of all nodes in conjunction compose the navigation field. The navigation field provides the robot with the "best possible" direction that has to be followed in order to reach the goal. The navigation field is computed based on a value iteration algorithm which considers the deployed sensor network as a graph, where the sensor nodes are vertices. Assume a finite set of vertices S in the deployed network graph and a finite set of actions A the robot can take at each node. Given a subset of actions $A(s) \subseteq A$, for every two vertices $s, s' \in S$ in the deployed network graph, and an action $a \in A(s)$, the transition probabilities $P(s'|s,a)$ (probability of arriving at vertex s' given that the robot started at vertex s and commanded an action a) for all vertices are determined. The general idea behind the value iteration is to compute the utilities for every state and then pick the actions that yield a path towards the goal with maximum expected utility. The utility is incrementally computed [192]:

$$U_{t+1}(s) = C(s,a) + \max_{a \in A(s)} \sum_{s' \in S-s} P(s'|s,a) \times U_t(s') \tag{16}$$

where $C(s,a)$ is the cost associated with moving to the next vertex. Initially, the utility of the goal state is set to 1 (0 for the other states). Given the utilities, an action policy for every state s will be as follows [192]:

$$\pi(s) = \arg \max_{a \in A(s)} \sum_{s' \in S-s} P(s'|s,a) \times U(s') \tag{17}$$

Finally, the robot maintains a probabilistic transition model for the deployed network graph, and can compute the action policy at each node for any destination point.

However a much more attractive solution is to compute the action policy distributively in the deployed network. The idea is that every node in the network updates its utility and computes the optimal navigation action (for a robot in its vicinity) on its own. When the navigation goal is determined (either a robot requiring to be guided to a certain node, or a node requiring robot's assistance), the node that is closest to the goal triggers the computation by injecting a *Start Computation* packet into the network containing its *id*. Every node redirects this packet to its neighbors using flooding, and updates the utilities according to equation (16). After the utilities are computed, every node computes an optimal policy for itself according to equation (17). Neighboring nodes are queried once again for the final utility values. The computed optimal action is stored at each node and is sent as part of a suggestion packet that the robot would receive if it is in the vicinity of the node.

- *Navigation*: the algorithm explained above allows the robot to navigate through the environment between any two nodes of the deployed network. Initially, the current node is set to the node closest to the robot. Using the algorithm above, the node suggests a direction to the robot, and the robot takes that direction. Afterwards, using the signal strength, the robot can know that its closest node has changed, so it takes the new direction as the direction suggested by the new closest node. In that way, the robot is able to navigate without using neither a map nor a GPS.

7.6. Critical issues and future trends

One of the main reasons leading to the development of a multirobot system may be the fact that it is possible to build more robust and reliable systems by combining unreliable but redundant com-

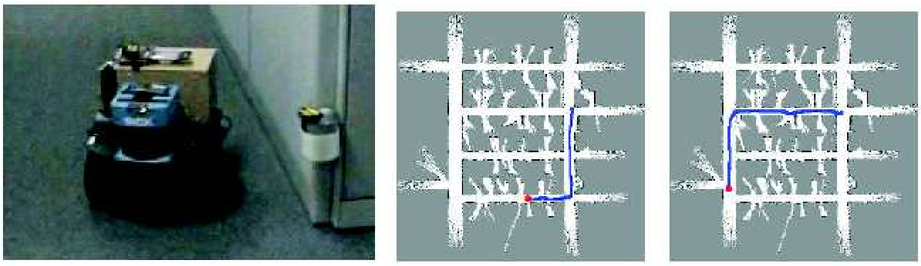


Figure 9: Navigation using a Sensor Network [192]

ponents. As it has been mentioned before, research in multi-robot systems has focused primarily on construction and validation of working systems. Therefore, in the literature there are few general analysis of problems and solutions. Furthermore, regarding the fundamental decision between a centralized or a decentralized multirobot architecture, we are not aware of any published theoretical comparison. Such a comparison would be interesting, particularly in scenarios where the team of robots is relatively small, and it is not clear whether the scaling properties of decentralization offset the coordinative advantage of centralized systems.

On the other hand, a combination of multirobot systems with wireless sensor networks can improve the overall reliability and performance of the whole system. Benefits can be identified in both directions: for example, a multirobot system could provide network repairing services to a WSN, and a WSN could provide extended sensorial capabilities to a multirobot team. Unfortunately, the interaction between a WSN and a robotic team has not been addressed in the literature.

The following issues have been identified as relevant research trends in the next future:

- Robotic teams networked with the environment: integration with wireless sensor networks and other cooperating objects should be studied in order to exploit many potential benefits and complementarities.
- Multirobot planning with reliability constraints has not been properly solved, specially in a distributed way.
- Centralized/decentralized architecture trade-off: a set of rules should be provided to allow optimal design decisions.
- Hard real-time interaction with the environment (transportation, etc.)
- Development of a set of metrics for the performance of a robotic team.
- Increase heterogeneity in the robotic team: complementarities, synergies, etc. can be exploited, but heterogeneity implies complexity in the design and the algorithms.
- More field experimental studies are definitely needed.

8. Inter-Vehicles Communication Networks

Urban areas will have dense network coverage in the near future with a large number of deployed co-operating objects to sense and actuate in the environment. Various vehicle and traffic management applications including road safety are expected to be supported by major technological penetration of Intelligent Traffic Systems (ITS).

Cooperation among vehicles is a promising approach to address critical road safety and efficiency. For instance, coordinated collision avoidance systems could significantly reduce road accidents. The road safety becomes rather problematic in countries which heavily rely on the road network for transportation of goods.

Various research initiatives aim at developing and applying advanced technological systems to increase highway capacity and safety, and to reduce road congestion and air pollution. The PATH project is a US collaboration between the California state government, the University of California, and private industry [194] which have been working towards this objective. Projects in Europe [195–197] and Japan [198] have similar long-term research goals.

Despite the fact that promising results have been achieved in some of these projects, major issues remain to be resolved. This might explain the large list of proposed traffic applications with no real deployment.

This section intends to survey the literature and discuss the issues in the area of inter-vehicle and road-vehicle communication.

8.1. Road-vehicle communication (RVC)

This system supports data exchange between vehicles and devices deployed along the roadside. The simplest form of RVC is to sense the vehicles flow at a junction and forward the raw traffic data directly (or through third vehicles) to a remote unit for human assisted data processing. A more intrusive but accurate approach is to instrument sensors in the vehicle [199]. In this case, critical fine-grained information such as the average/instantaneous speed and flow statistics could be computed. The recommendation to the drivers including countermeasures would be transmitted to vehicles on the road via tunable radio station or wireless network. In addition, the driver could receive important information (e.g. speed limits) about a road section.

This centralised approach provides more control over the data processing. However, it may introduce issues for applications that cannot tolerate high system processing time.

8.2. Inter-vehicle communication (IVC)

Inter-vehicle communication (IVC) enables vehicles to communicate with other vehicles without line-of-sight (LOS) using wireless communications. We classify the IVC approaches identified in the literature with respect to the infrastructure they are built upon. When the communication among and between vehicles relies on a deployed system infrastructure, say a cellular network, the IVC is termed *infrastructure-based*.

In contrast, vehicles may communicate where essentially no pre-planned infrastructure is present.

They usually form a wireless mobile ad-hoc network (MANET) in order to deliver data among themselves. We call this an *infrastructure-less or ad-hoc* network setup.

IVC can be supported by any of these two approaches. RVC is often an infrastructure-based scheme as ground support including pre-deployed road sensors needs to be in place.

8.3. Communication Scenario

IVC networks exhibit characteristics that are specific to scenarios of *high node mobility* [200]. We list some of these properties:

- Low-power consumption and small physical size have been some of the fundamental issues in designing software and hardware for low-cost wireless sensor nodes. However, vehicles exibit more physical space and energy consumption is not an issue. Unlike typical WSNs deployment scenarios, vehicles can be instrumented with powerful sensors and radio to achieve long transmission ranges and high quality sensor data.
- Contraints on mobility may impact the coverage of the network in cases where no ground infrastructure (e.g base stations) to support the communication between vehicles is present. Adding more nodes to the network may not improve coverage but rather affect negatively the use of available bandwidth.
- Despite the constraints on the movement of vehicles (i.e., they must stay on the lanes), the communication can be severely impaired by the high relative velocities of vehicles, even when moving in the same direction. In ad-hoc short-range communication, the network will tend to experience very rapid changes in topology.
- The mobility pattern, driver behaviour and vehicle responses are difficult to model in a simulator. Also, the current lack of information on these aspects pose a challenge in validating models created for real world applications.
- Vehicle density varies from one area to another. Urban areas tend to have higher density than, for instance, remote rural areas.
- The adoption rate of IVC/RVC technology in vehicles is expected to be low in the near future.

Although these characteristics have some impact on infrastructure-based IVNs, the major issues arise from ad-hoc network setups. Blum et al [200] pointed out that IVC networks have mobility characteristics different from the typical MANETs considered in the literature. Specifically, these characteristics cause rapid topology changes, frequent fragmentation of the network, a small effective network diameter, and limited utility from network redundancy. Such issues must be tackled from the outset since they directly influence the IVC system design at all communication layers.

8.4. IVN applications

In this section we introduce categories of IVC applications.

8.4.1. Safety

To make roads safer for passengers, data among vehicles can be exchanged to determine and avoid dangerous situations ahead of time. Proactive safety systems are at the top of the list of priorities in the current IVC research.

The co-operative assistance systems coordinate vehicles at critical points such as in highway entries, round-about and crossing without traffic lights. In the case of a traffic jam, event messages could be broadcasted to vehicles following behind just in time for a turn in the next exit (alternative path discovery).

Also, sophisticated vehicle onboard warning systems can disseminate real-time information about roadway and environmental hazards to near-by vehicles. The driver's behaviours (steering, braking and acceleration) could be determined by inspecting the controller area networks (CAN) of the vehicle. This information may be associated with data exchanged through the IVC network in order to give an early warning of dangerous driving situations [201].

Such information includes weather and road surface conditions (e.g. slippery, icy), accidents events immediately detected by sensing the air bags ignition. Zones of relevance for such events could be defined in order to make a focused and efficient hazard warning data dissemination [202].

Also, it would be useful for vehicles to receive information about cars preceding the immediate vehicle. For instance, consider a lorry driving in front of a vehicle in a two-way road. It would help if the visual field of the lorry driver could be 'passed' to the driver following behind. The lorry could send realtime video images from a camera installed in its front side. This would extend the visibility of the vehicle driver and therefore could potentially avoid frontal crashes in take-overs. This is even more relevant when the weather conditions affect the driver's visibility.

Intelligent cruise control is another safety and comfort application area. A common scenario is where the vehicle attempts to automatically keep a safe distance margin to the vehicle ahead. The distance can be measured using an accurate range finder system (e.g. milimeter radar or infrared laser). Realtime adaptation algorithms would use IVC in order to proper control the speed and acceleration of vehicles.

Besides, the vehicle can use an IVC network to exchange messages between the vehicle ahead and vehicles in adjacent lanes to perform control manoeuvres such as smoothly braking when the adjacent vehicle changes lane. This application scenario is discussed in [203].

The safety applications discussed in the literature rarely include other parties that may be involved in road accidents such as motorcyclists and pedestrians. For instance, speed limit control based on the location-awareness of vehicles requires some local knowledge of speed constraints and density of pedestrians in the section a vehicle is entering. Cooperating objects deployed on the roadside could communicate this information to the vehicle.

8.4.2. Traffic management

A route finder system avoids an otherwise convenient-looking route in favour of another less congested. Cooperating vehicles could monitor areas in order to identify traffic congestion ahead of time by communicating dynamic traffic flow information such as the density of nearby vehicles.

The envisaged system would give enough time for vehicles to opt for alternative routes and therefore avoid the traffic jams. This real-time congestion information would be frequently forwarded from one vehicle to another as long as there is still traffic congestion.

In addition, efficient traffic flow can be achieved with platooning, which is a technique that arranges two or more vehicles at a regular distance. Inside a platoon all the vehicles follow the leader with a small intra-platoon separation of usually 1 meter. The inter-platoon spacing is assumed to be large so as to isolate the platoons from each other [204]. The available capacity of the road network is expected to increase as the overall vehicle headway is significantly reduced.

GPS-based mapping and guidance are other areas of application. Current GPS-based navigational systems use limited information on the current traffic conditions since real-time data collection through sattelite is currently expensive to deploy at large scale. This can be achieved with information exchange among vehicles through wireless communications.

8.4.3. Environmental protection

The Kyoto protocol was established to set specific targets for reductions in greenhouse gas concentrations [205]. Emission restrictions targets were made to some countries, including member states of the EU.

Appropriate means for monitoring and reducing gas emissions will be required in the near future. To support this, vehicle-based sensor networks may be deployed where vehicles not only gather dynamic traffic and in-vehicle information but also sense the air quality around the car (nitric oxide, carbon monoxide, etc).

The sentient car project [206] proposed a sensor-based vehicle to collect real-time information about the car's performance and its surrounding air quality and overlays this data on a map of the area the vehicle is visiting. In this application, multiple cars locally sense levels of pollution (nitric oxides, carbon monoxides, noise level) measured with sophisticated sensing equipment placed in the tailpipe. The sensor data can be communicated to roadside devices or forwarded to a remote processing unit through a hybrid wireless infrastructure: cellular network and IEEE 802.11.

8.4.4. Traffic and vehicle information for billing

Congestion-based charging has gained interest over the past years as a means to mitigate traffic congestion in urban areas. There are different models in use today, ranging from area to time-based charging. Central London and Singapore are two cities which have implemented this charging scheme.

Other approaches discussed in the research literature rely on dynamic traffic information in order to implement a dynamic congestion-based charging scheme. The charge to pay to use a particular route varies over time according to the levels of measured congestion. This scheme has been regarded as more efficient than time or area-based charging.

To put this dynamic charging scheme into practice, an efficient distributed traffic monitoring system needs to be in place. IVC networks could assist vehicles in cooperatively detecting congested areas. Such a charging scheme could offer choices of routes to vehicles entering a congested region. The driver could opt, for instance, between inexpensive but longer alternative routes or pay more to use the current congested route.

A slightly related application is dynamic insurance policy as pointed out in [201]. The premium should change to indicate the vehicle usage and driver's behaviour on the road. Information that could be taken into account include the common routes selected, time of day, and acceleration patterns.

8.4.5. Data communication using delay-tolerant networks

The problem of providing data communications to remote and rural areas is discussed in [207]. The authors consider the approach of asynchronous messaging in order to greatly reduce the cost of connectivity.

The Wizzy Digital Courier service provides asynchronous Internet access to schools in remote areas of South Africa [208]. A courier on a motorbike or bicycle, equipped with a USB flash storage device, travels from a village school to a city carrying all the outbound email and web requests for the day. The courier may forward the data collected from distant schools to another courier through wireless communication.

A similar project in Lapland [209] aims at providing intermittent Internet connectivity to the Saami population, who live in widely dispersed communities in remote areas, and are not well served by either wired, fixed wireless, or satellite internet service. As they travel on snow vehicles from community to community, the data can be stored and forwarded through opportunistic communication between devices on these vehicles supplemented by a few solar-powered base-stations positioned on tracks in wilderness locations [210].

Because the data communication is asynchronous and it relies on mobile routers to collect, forward and deliver messages between static nodes (sensor networks, and central servers) [211], this type of application is best supported by a delay-tolerant network framework (DTN) [212]. Such networks are assumed to experience frequent, long-duration partitioning and may never have an end-to-end path.

This is a niche scenario since few applications will tolerate high delay for web access and other instant-based services.

8.4.6. Added-value services

We observe that there is a potential for using vehicle-to-vehicle communication to leverage a class of opportunistic communication services that, although non-safety critical, could be realised with an IVC network. We include in this list services such as data look-up for file sharing among moving vehicles.

The IVC/RVC would be used as an ad-hoc distributed storage mechanism which would be accessed by the in-vehicle entertainment system. Gerla et al [213] put forward a proposal for a file sharing system between cars based on the bit torrent peer-to-peer system. To be legally acceptable, however, such an application would require appropriate incentives to avoid the unauthorised distribution of copyrigthed material.

Interactive communication among vehicles is another type of service that drivers and passengers may find useful. This service could establish a voice connection to other vehicles driving in the same direction for the purposes of traffic information exchange. Interactive applications including instant messaging and multiplayer games and onboard Internet access would make the journey smoother for families driving with kids.

8.4.7. Important aspects

We consider in this section a non-exhaustive list of design issues that should be taken into account when engineering an IVC network.

- Time constraints: establish the application's relative tolerance to overall latency including network and processing delays.
- Reliability: application's tolerance to errors made visible to it. A zero tolerance threshold indicates that the applications requires 100% guaranteed data delivery.
- Scale: refers to the number of destination vehicles intended to receive a particular data item.
- Levels of infrastructure: indicates to what extent a pre-planned infrastructure is required to support the deployment of the application. In ad-hoc network setups none infrastructure is necessary and vehicles devices self-organise into an IVN.
- Security and privacy: some applications require authentication schemes for communication and degrees of anonymity of vehicles and drivers information.

According to these aspects, an IVC service may experience severe network delay and still be reliable. We envisage, however, that a significant number of services including the dissemination of warning messages require low latency and highly guaranteed data delivery. Such requirements pose interesting design challenges in pure ad-hoc inter-vehicle communications. An open question here is whether any level of infrastructure will be necessary to support reliable and timely IVC services.

In Table 8.4.7 we intersect each application category with the design issues discussed above.

These applications have been discussed in the research literature along with proposed schemes for their implementation either using infrastructure or ad-hoc network setups. The sections that follow present an overview of these schemes and issues.

Our approach is to break down the issues at various levels of system and communication protocol design. The lowest level we discuss is medium access control (MAC) layer in the next section. We then describe unicast and multicast routing issues and approaches.

8.5. MAC Layer

This communication layer should strive to maximise the packet throughput of the IVC/RVC network by minimizing the latency and packet loss rates. Thus, congestion control becomes a key design issue of MAC layer protocols.

The scale of the IVC network can significantly complicate the engineering and validation of a suitable MAC protocol. We expect that perhaps in a busy highway thousands of cars travel within a road segment. It does not necessarily mean that thousands of nodes will communicate. However, in the order of hundreds may require some form of inter-vehicle communication.

	Safety	**Environmental protection**	**Charging**	**Delay-tolerant**	**Added-value Services**
Time constraints	Real-time (bounded low latency)	High tolerance to latency	May tolerate high communication delay	Asynchronous communication, may tolerate extreme latencies - e.g. hours or days.	Interactive systems may not tolerate high delay
Reliability	High	Medium - tolerate loss of data (e.g. multiple sensor sources in a region)	High - e.g. data communication for 'pay as you drive' insurance must be reliable and accurate	High reliable communication. Similarly to the email system - guaranteed delivery and variable latency	Medium-High. For instance, voice calls between vehicles require reliable communication with guaranteed bounds on data loss.
Scale	Source of event towards multiple vehicles	Multiple sources to individual destinations	Multiple sources to multiple vehicles	Single source-single destination	Single source-multiple destination; Single source-single destination
Security	Source authentication	Source authentication	Source authentication, high degree of privacy	Authentication and privacy	Authentication and privacy

Table 2: Important aspects to IVC applications

Luo et al [214] suggest that there are two general approaches in designing MAC layers for IVC networks. One scheme uses the MAC functionality of existing wireless LAN systems including the medium-range radio IEEE 802.11. In contrast, the other approach extends the MAC layer of 3G cellular radio systems. The next sections discuss these two approaches.

8.5.1. Wireless LAN

Usually WLAN MAC protocols have limited support for ad-hoc distributed coordination to the medium access. The 802.11a is the MAC layer protocol chosen as the basis of the Dedicated Short Range Communication (DSRC) standards, which has been allocated 75MHz (5.85-5.925 GHz) of the spectrum in the USA by the FCC for any type of vehicular communication.

The motivation for this technical choice is to take advantage of the Distributed Coordination in ad-hoc mode that is currently built-in in the PHY and MAC standards specifications of these systems [215].

The difficulty arises from the high mobility characteristic of an IVC/RVC scenario which increases significantly the probability of network partioning. Thus, a MAC mechanism that explicitly allocates resources (timeslots, frequency spectrum or codes) introduces a major problem [200]. Contentions created in congested roads, for instance, may require dynamic allocation and deallocation of codes in order to optimise the network transmission throughput. Such a system overhead could potentially add an extra delay to the system. This is acceptable for non-safety applications but it is likely to be problematic for real-time data service delivery of safety-critical messages.

8.5.2. Cellular Network

An alternative to the previous approach is the use of unmodified MAC layer of current proposed cellular networks. This scheme relies on aggregation points (cells) to forward the data packets from one vehicle to another. The advantage of this is the availability of pre-deployed cellular network infrastructure. To what extent, however, the system latency (network and processing) and the limited transmission rate would impact the applications remains to be investigated.

Today, the reasonable cellular network coverage may have a positive aspect when providing reliable and timely IVC service. The problem for some applications would be the limited transmission data rate of a 3G cellular network system which is up to 144 Kbps to users in high-speed motor vehicles.

Another approach extends 3G cellular systems in order to add functionality for decentralised medium access [197]. This scheme has potential for a better control of the radio resource because of the Code Division Multiple Access (CDMA) subsystem. Consequently, this gives flexibility to adjust the data transmission rate when compared with WLAN extensions. The design challenge is how the decentralised ad-hoc coordination could be engineered.

The discussion of which approaches should be used in IVC networks is tightly coupled with the question of having an infrastructure-based or an ad-hoc system deployment. The extended WLAN approach would certainly be more suitable for ad-hoc than the 3G cellular network.

However, an infrastructure-based 3G cellular system can offer the appropriate functionality to design the MAC layer for an ad-hoc vehicular network. The ad-hoc network could be *overlayed* on the

3-G network.

A hybrid solution of these two approaches is being explored in some research projects [206]. In the Sentient Car, the researchers equipped an experimental vehicle, a Ford Transit, with two dashboard LCD displays, one for the driver and a larger one for the navigator. A PC placed in the back of the van integrated all the sensing and communication capabilities. A GSM link (9.6 Kb/s) provided low-speed data connection but with a good network coverage. The 802.11b network interface provided much higher data rate but at the expense of very limited coverage.

In hybrid approaches, an issue that arises is how the system seamlessly handles the transition from one type of wireless medium to another (vertical handover). This may introduce packet loss and latency which can severely damage the data communication. To minimise these adverse effects, network handover approaches have been discussed in the literature.

Although handover for homogenous wireless networks is a well-understood research topic, there are ongoing research efforts in developing mechanisms for hetegeneous networks. An approach is to leave the handover decisions to the network. The mobile node reports the received signal strength from various base stations to the network which then decides when to switch the node to another base station [216]. Usually buffering schemes can be used to mitigate the effects of handovers.

The problem with network-controlled handover is the lack of information on the current status of the mobile node including factors such as the applications running, processor load, physical context and so forth. This complicates the process of deciding precisely the appropriate time to handoff. Recently practical results obtained from a IPv6-based testbed composed of GPRS, WLAN (802.11b) and LAN network systems [217, 218] show that the major issue in vertical handovers is the latency. The average handover latency of a WLAN to GPRS TCP connection is 3.8 sec in upward handover (maximum of 4.4 sec) and 6.8 sec in downward handovers (maximum of 8.8 sec).

Network coverage can be significantly improved with a hybrid IVC network (infrastructured and ad-hoc). However, these results suggest that vertical handover issues in heteregenous cellular-based IVC systems should be addressed in order to offer reliable communication services to the applications.

8.5.3. Approaches

The issue of designing a MAC layer for ad-hoc IVC networks has been addressed in [219]. The authors discuss an extension to the reservation ALOHA to efficiently deal with distributed slot reservation. Vehicles rely on their neigbours to determine if their request for a slot has succeeded. As pointed out in [200], the high mobility of vehicles results in varying sets of neigbhours. It is unclear, therefore, whether the proposed distributed reservation scheme can deal efficiently with high mobile nodes by keeping low the number of packet collisions in the network.

There are other proposals based on some traditional LAN technologies such as the non or p-persistent CSMA used by DOLPHIN [220]. The contribution of this work is to show that the non-persistent CSMA outperforms the p-persistent scheme regarding packet loss in those cases usually involved in IVC. As a result, the non-persistent CSMA is adopted as the IVC protocol of the DEMO 2000 cooperative driving application [198].

FleetNet Design The FleetNet project have chosen the UMTS Terrestrial Radio Access Time Division Duplex (UTRA TDD) [202, 221] as the MAC layer. The original specification of this layer relied on a centralised coordination scheme between mobile nodes. Such a design choice may introduce a problem when extensive ad-hoc communication capabilities are needed. For instance, in cooperative driver assistance high speed vehicles may be only required to communicate with vehicles in the closest vicinity and usually for a short period of time.

To overcome the centralized control, several changes to the UTRA TDD layer have been proposed. The structure of the protocol framing was kept from the original specifications. The frame duration is 10ms and 14 slots are available in each frame in the aspired Low Chip-Rate mode (LCR).

The MAC is organised in a decentralised manner where each station is individually responsible for managing the available resources. The Reservation ALOHA (R-ALOHA) scheme is used to coordinate the distributed access to the medium. Reserved slots are used in subsequent frames as long as there are packets to be transmitted. The reservation of the next slot is indicated by piggyback signalling.

To offer high priority services, an adjustable portion of the transmission capacity in terms of number of slots per frame can be constantly reserved. The remaining part can be dynamically assigned and temporarily reserved by different stations for services with lower priority. In order to avoid the problem of near-far-probe and to keep the power control scheme simple, the MAC protocol mandates that only one station transmits in a slot at a time.

The simulation model assumed in this work that all nodes were within the same radio range and the interference was measured as a binary value of 'yes or no'. The physical layer was entirely based on the UTRA-TDD PHY layer. The simulated MAC layer assumed that nodes have at least one reserved time-slot. In practise this implies that they need to establish a circuit-switched broadcast connection which they never quit. The maintenance of a long-term connection is an issue.

An important result obtained is that as long as the traffic load is below saturation, the mean delay is almost constant. For messages of small lengths, the constant delay was 25ms. Although this was constant, it is reasonably high for the direct communication model of the simulation.

The great weaknesses of this protocol validation is the simulation model used. First only direct communication among nodes withing the same radio range has been considered. Secondly, there is no discussion on the mobility model simulated. Third, the authors intend to include detailed channel models with respect to path loss and interference conditions in future simulations. Thus, it is unclear whether the presented results can be representative of a real application scenario.

WRTP The Wireless Token Ring Protocol (WTRP) [222] addresses the distributed coordination issue by proposing the design of MAC layer protocol based on the IEEE 802.4 standards. This protocol is commonly referred as token ring and it was extensively used in the late eighties and early nineties.

In particular, WTRP addresses the single points of failure of centralised coordination mechanisms in order to support dynamic changing topologies, where nodes can be partially disconnected from other nodes. The authors made a case for using a token ring protocol for platooning applications because of the high spatial reuse that can be achieved.

A connectivity manager component in each node maintains a connectivity table that is an ordered list of stations in the ring. This table scales linearly with the length of the ring. Similarly to the original token ring protocol, the WTRP protocol offers a ring recovery mechanism that is trigerred when the monitoring node decides that its successor is unreachable.

In such a case the station recovers from the failure by reforming the ring. When looking-up in the connectivity table, a node can find the next connected node in the transmission order to send a SET-PREDECESSOR message. This mechanism allows nodes to dynamically join and leave the ring.

It is unclear, however, how the system copes with frequent joins/leaves in which case the ring needs to be frequently recreated. This process may render the system to a high maintenance overhead and *livelock* where the communication overhead (control token messages) is so high that the system cannot make any useful progress, i.e. communicate data among stations.

The results presented cannot be taken as representative of a realistic IVC network scenario. Only four nodes have been used in the simulation model where far more moving nodes would be expected to simulated increased topology changes with announced joins and leaves.

Also, the protocol is implemented on top of 802.11/DCF mode. Thus, it is difficult to assimilate the real performance of this protocol when compared to the 802.11 based approaches. However, token ring is more robust to collisions compared to the 802.11. The cost to pay is the maintenance of the ring.

MAC for DSRC Qing Xu et al [223] argue that MAC protocols that rely on centralised schemes for dynamic allocation of resources such as TDMA (time slots), FDMA (channels) or CDMA (codes) are difficult to use in highly dynamic IVC/RVC networks. Also the authors rule out protocols that use synchronous control communication schemes such as the request-to-send (RTS) and clear-to-send (CTS) schemes.

This paper discusses the design of various random access protocols for IVC and RVC in the Dedicated Short Range Communications Spectrum (DSRC). The design is optimised for safety applications, which require strict quality of service guarantees. The communication should be reliable and with low latency.

The authors pointed out that the distributed coordinated function (DCF) for MAC protocols cannot provide the QoS required by safety applications. For instance, the Enhanced DCF protocol addresses QoS by packet prioritisation. The performance, however, degrades when the number of packets of equal priority increases. This is the case in safety applications where differentiation of packets may not be suitable since it is difficult to establish the priorities among different types of safety messages.

In this scenario, receivers are specified as a geographic region (geo-cast zone) relative to the transmitter. The sender broadcasts messages to all the receivers in its communication range. The receiver applies a multicast filter to determine whether its is in the geo-cast zone of the message. Each message has an associated lifetime which is regarded as the usefulness of the message. In case the lifetime expires, more messages could be transmitted if those sent could not be delivered.

In this work, the following protocols have been studied:

- Asynchronous fixed repetition (AFR) – the radio does not listen to the channel prior to communication and the protocol randomly selects k distinct slots out of the n available slots constituting the lifetime.

- Asynchronous p-persistent repetition (APR) – the node transmits a packet in each of n available slots in lifetime with probability $p = k/n$, while with probability $1 - (k/n)$ it delays the message transmission to the next time slot. In this case, the radio does not listen to the channel before it sends out a packet. The positive integer $k \leq n$ is a design parameter of the protocol.
- Synchronous fixed repetition (SFR) – same as the AFR protocol except that all nodes are synchronised to a global clock similarly to the slotted ALOHA scheme. All the generation and transmission of messages happen at the beginning of a slot. This partially avoid overlap between packets. It does require, however, a global clock.
- Synchronous p-persistent repetition (SPR) – similar to the APR except for the global synchronisation among the nodes of packet generation and transmission.
- Asynchronous fixed repetition with carrier sensing (AFR-CS) – if a node with a new message to transmit senses the channel is busy, the message is regarded as backlogged and the node will attempt the message transmission in the next empty slot. The node repeats the attempt until the channel becomes idle.
- Asynchronous p-persistent reception with carrier sensing (APR-CS) – similar to the AFR-CS protocol with the repetition slots selected following the the p-persistent scheme.

These protocols were studied analytically and through simulation in NS2 (network simulator) and SHIFT (highway vehicle traffic simulator). Messages were generated based on events including on-board sensor measurements. The message is passed down to the MAC layer, which attempts to send a message only within the lifetime of the message.

The preliminary results presented validated their simulation. There is an optimal probability for p-persistence protocols where transmitting packets beyond this point increases the number of congestion in the channel. Also, the idea of sensing the channel before transmitting improves the performance as observed in this work but introduces issues that have not been fully discussed such as synchronisation of colliding stations.

The main conclusion is that asynchronous fixed repetition AFR with carrier sensing showed the lowest probability of reception failure around 0.0008, which was more than an order of magnitude lower that of IEEE 802.11. This indicates that repetitions with equal probabillity (fixed repetitions) helps to overcome interference by giving a transmitter more chances to transmit. The synchronous and asynchronous versions of this protocol have shown the same benefit. However, it is more practical for the radios to sense the channel before transmitting than to synchronise the transmission of all nodes. This is the reason why the AFR-CS protocol was the preferred choice by these authors.

The MAC protocols discussed here have been evaluated in simulations with limited real traffic data and mobility models. This raises the question of their efficacy to address the communication needs at link layer of real IVC/RVC applications.

8.6. Routing

Application messages need to be transmitted from a source to a destination point. When these nodes are within the same radio range (closest vicinity), then the MAC layer protocol is sufficient to forward directly these messages.

In other cases, however, the radio device of a source node might not be sufficiently powerful to reach a far away node. To overcome this, we can extend the communication range by routing packets through third nodes. This creates a multihop path between any two points.

The source node transmits its packets to destination nodes through its neighbouring nodes, which decide the next hop to forward packets using some static or dynamic routing policy information with local or global scopes. The scalability of the routing protocol depends on various factors including the average amount of status information each node must store about other nodes and the update frequency of the network status data. In large scale ad-hoc deployments the maintenance of global network status in each node introduces scalability issues. Ideally, the routing protocol should rely on local status information (neibhouring nodes) in order to make global routing decisions.

8.6.1. Traditional MANET protocols

Traditional Internet routing protocols such as RIP and OSPF are both *proactive* routing protocols. Periodic broadcast of network topology updates (e.g., distance vector or link state information) is used in order to compute the shortest path from the source to every destination. This operation consumes a reasonable amount of network bandwidth.

Although they are widely used in the Internet backbone, they cannot be used in the MANET directly because of the differences between these two types of network including the limited bandwidth in the MANET (up to 54Mbps with 802.11g compared to Gigabits in Internet backbones). Thus, the routing control overhead cannot be ignored in the MANET. The network topology in MANET is dynamic, changing at rates that depend on how nodes move around. The topology in this case for proactive protocols need to be updated at a higher frequency than a fixed network.

These are the main reasons for why several proposed proactive MANET protocols stemmed from Internet routing protocols had major issues when dealing with high mobility and limited bandwidth of MANETs. The issue to address is how we can decrease the amount of routing control traffic. Protocols in this category include the Destination-Sequenced Distance Vector Routing Protocol (DSDV) [36].

To keep the amount of routing control traffic low, reactive protocols have been proposed to notify the network on how packets should be treated by forwarding nodes in a multi-hop scenario. This may range from pre-defined routing paths that packets should follow to specific policies implemented in the forwarding nodes, for instance, to QoS-based packet scheduling. Typical MANET reactive protocols include the Ad hoc On Demand Distance Vector (AODV) and the Dynamic Source Routing (DSR). Reader is referred to [224] for further information on these protocols.

Reactive routing follows two steps. In the route discovery the source node broadcasts a packet throughout the network to find the route between itself and the destination. In the route maintenance phase, the established routes are checked for validity since the nodes along the path are free to move arbitrarily. When any failure is found along the path, the source will be notified and it may decide to restart the route discovery process.

The high mobility characteristics of IVC networks create a challenge for any type of reactive and proactive protocols. The rapid topology changes and the high probability of disconnection of some parts of the network are issues that need to be addressed in the design of routing protocols. Blum et al [200] showed through simulations that current MANET routing protocols fail to address these issues. Proactive protocols will be overwhelmed in maintaining fresh routing status with rapid topological changes. In contrast, reactive protocols attempt to discover routing paths before sending a message. However, the short lifetime of routing paths and the fragmentation of the networks make reactive routing in IVC/RVC networks extremely difficult .

8.6.2. Location-based routing

Communication messages will likely to have an *area of relevance* which defines the subset of the travelling vehicles that would be interested in receiving such messages. In intelligent cruise control, for instance, the vehicle attempts to maintain a safe distance to the other vehicles in a highway. The geographical area of interest for message exchange can be geometrically represented as a circumference with the centre being the vehicle which generates events.

Location-based routing protocols have been proposed for IVC/RVC networks. Although some of these protocols were originally proposed to mobile ad-hoc networks with low mobility they were adapted to address the issues that arise in high mobility scenarios. In this mechanism, the location of the vehicles (e.g. GPS position) are used to route messages from a source to a destination point through neighbouring nodes. An intermediate node upon receiving a message decides locally whether there are neighbours closest to the destination than the node itself. This strategy is called *greedy forwarding*.

This approach can fail when there is no neighbour available that is closer to the destination than the current forwarding node. To address this, several recovery mechanisms have been discussed in the literature including the Perimeter Mode in Greedy Perimeter Stateless Routing (GPSR) [225]. The perimeter mode constructs a planar graph from the local connectivity graph where the void space has been identified. The planar graph eliminates any redundant edge and packets are routed using the paths of this constructed graph using the *right hand rule*. This rule states that when arriving at node x from node y, the next edge traversed is the next one sequentially counter-clockwise about x from edge (x;y). The rule traverses an exterior region, in this case, the region outside the same triangle, in counter-clockwise edge order.

To address the deficiencies of the perimeter mode approach, Lochert et al [226] proposed a strategy to deal with the high mobility of nodes that uses information about the specific topological structure of a city. With this information, the routing can be assisted with data to decide ahead of time likely trajectories that vehicles may take. This is shown to overcome the major problem of ’void spaces’.

The proposed position-based routing (called Geographic Source Routing) protocol uses maps of a small part of Berlin. The authors argue that availability of maps is a valid assumption as more often vehicles are instrumented with onboard navigational systems.

Similarly to other position-based routing protocols, the GSR strategy relies on position information in order to make forwarding decisions. One of the requirements for this class of protocol is the availability of a location service that can provide the current position of a node. To send a packet to a destination node, the sender needs the fresh position of the destination so that it can be appended to the message header. The GSR proposes a location service called reactive location service (RLS) to fulfil this requirement. Location queries are flooded on the network and the reply message will contain the current geographical position of a given vehicle.

By means of simulation, this research work compares the GSR approach with non-position-based ad-hoc routing strategies such as the Dynamic Source Routing/DSR and Ad-Hoc On demand Distance Vector Routing (AODV) [224]. While the DSR's performance is severely affected by issues such as scalability and high mobility (e.g. short routes lifetimes), both AODV and the GSR position-based approaches have good performance while the position-based approach outperforming AODV.

The DSR protocol has the highest network load among all the three protocols as it generates a large amount of signalling traffic. This protocols sends large packets because of the routes appended to the packet headers, especially during the route discovery phase, which leads to a significant network overload. The second cause of failure for DSR is where mobility of vehicles cause frequent route breaks. This is more severe in highway scenarios which have higher mobility than the considered city scenarios.

On the other hand, the GSR position-based approach shows slightly higher packet delivery rate when compared to DSR and AODV. The GSR and DSR shows that there is an overhead for the first packet of a connection. As these two protocols are source-based routing a route establishment for the DSR or the location discovery for the GSR are two similar processes that need to be carried out. However, AODV presents the highest latency as it uses an expanding ring search technique.

The authors recognise the ad-hoc routing approach as a feasible mechanism compared to cellular network-based telematics. The benefits discussed are the low delay for data transport in emergency warning systems, robustness of ad-hoc because of the network's mesh structure and low costs because of unlicensed frequency bands. However, it remains to be investigated whether these benefits will happen in real deployment of this system. A review of position-based unicast protocols can be found in [227, 228].

8.7. Multicast networking in the context of wireless inter-vehicle and road networks

8.7.1. Multicast addressing and delivery

Many of the envisaged applications for vehicle-vehicle and road-vehicle communication require communication with destinations that are groups of vehicles. The group may be defined by:

- *Geocast:* group membership defined by the current locations of the destination vehicles
- *Property-based:* group membership defined by other properties of the vehicles (e.g. vehicle type: private car, taxi, heavy truck, light truck, PSV)
- *Explicit groups:* a group may consist of vehicles that have explicitly joined a named grouping, such as the subscribers to a commercial traffic alert system

From the application point of view, the purpose of multicast communication is to address some messages to a group of destination nodes without any need for the application to be concerned with:

- Group membership: the identities of the nodes in the group and dynamic changes in its membership

- Reliable delivery: guaranteed delivery to all members of the group. Approaches to this include the use of negative acknowledgements and 'gossip' protocols in which nodes keep neighbouring nodes informed about the messages they have received.
- Ordered delivery: several messages sent to a group may arrive at different nodes in the group in a different order. If the messages are all from the same source, then the problem is called source ordering and can be achieved by the use of timestamps and delayed delivery of out-of-order messages. However, for messages from different source nodes whose clocks may not be perfectly synchronised, the problem is non-trivial and has at least two sub-classes: total ordering and causal ordering.

Not all applications require reliable or ordered delivery, but it seems likely that some vehicle-related applications will. For example, safety warnings, traffic control commands and speed limit information need to be delivered reliably. Traffic control commands must be received in the same order by all vehicles and speed limit end messages must be delivered after the corresponding begin message.

The above discussion is based on established distributed systems techniques as can be found in several sources such as [229]. But the standard techniques for reliable and ordered delivery often incur significant delivery delays and many vehicle-based applications are sensitive to delay.

8.7.2. Multicast routing

Single hop Since wireless networking is inherently a multiple-access technology, all messages sent are received by all the nodes in the same cell as the source node. But multicast messages are delivered to the application layer only in nodes that are members of a destination group identified in the message header. This is achieved by filtering on the destination address in the network layer or in the network interface hardware.

Multi-hop If the members of a multicast group are in more than one cell, then the network layer must ensure that message addressed to the group are received by all members of the group. Some nodes must act as relays to transmit messages between cells. This can be done in one of the following ways:

- Flooding: the multicast messages are forwarded to all cells. Nodes in each cell select messages by filtering. This is clearly extremely inefficient and non-scalable.
- Simulated multicast: a list of group members' network addresses is supplied to every sending node and multicast messages are actually sent by unicasting to each member of the group
- Multicast-aware relaying: relay nodes maintain sufficient knowledge of the multicast tree to enable them to efficiently transmit messages to all of the cells in which members of the destination group reside. Note that this can be viewed as an optimised form of flooding in which the tree of cells to which messages are sent is pruned to cover only cells that are likely to contain group members. The pruning need not be perfect, just good enough the achieve adequate efficiency. But the maintenance of the multicast tree in the case of rapidly-changing groups such as those that are used in geocasting may be unacceptably expensive.

8.7.3. Geocasting

The message recipients in a multicast data communication are likely to exhibit some spatial correlation. To address this need, location-based multicast regarded as *geocasting* may be a suitable communication paradigm for group-based data dissemination that takes into account geographical areas.

Geocasting enable the transmission of a message from a set of sources to all nodes within a destination geographic area. Geocasting is a subclass of multicasting which can be implemented with a multicast membership defined by the current locations of the destination vehicles. Also, further multicast filtering can be applied in order to refine the received data and associate it with other groups of interest. For instance, this could be used to distinguish messages targeting vehicles and pedestrians.

In this section, we give an overview of the most important geocasting algorithms based on the descriptions presented in [230]. When appropriate we comment on the suitability of the algorithm for IVC/RVC applications.

8.7.4. Flooding-based geocasting

Pure Flooding The simplest form for implementing geocasting is to flood the network with packets targeted to a geographical destination region. The receiver checks whether its position is within this region upon the packet arrival. The major drawback of this scheme is the high number of generated packets in the network which significantly increase the overhead as the number of participating vehicles increase.

Location-based multicast (LBM) An approach to minimise such an overhead is to give some orientation to the flooding process by establishing a forwarding zone. Nodes discard packets when they are outside this zone and forward them otherwise. In [7,8] the authors proposed two LBM schemes which differ in the way these zones are defined.

In one of them, a zone consists of at least the destination region and a path between the sender and this region. The zone size can be adjusted by a system parameter in order to increase the probability for message reception at the expense of an overhead increase.

The second scheme specifies the zone by the coordinates of the sender, the destination region, and the distance of a node to the center of this region. When a node receives a geocast packet, it determines whether it belongs to the forwarding zone specified in the packet by computing its own geographic distance to the center of the destination region. If such a distance is less than the one recorded in the packet, the node forward this packet to its neighbours.

Voronoi diagrams When the forwarding zone is empty or partitioned the LBM protocol can perform poorly. To address this, neighbours that belong to a zone are determined using Voronoi diagrams, which partition the network in n regions, where n is the number of neighbours. The voronoi region of a neighbour consists of all nodes that are closer to this neighbour than to any other neighbour.

When a node receives a geocast packet it first computes the voronoi diagram The voronoi partitions intersecting with the geocast destination region are used as the forwarding zone. Inside the destination region, flooding is often used.

Although Voronoi diagrams minimise the problem of empty forwarding zones through network partition the overhead for flooding packets in the network is still high. Also, in scenarios of high mobility the computation of voronoi regions can be a major issue as the node distribution in the network will vary over time.

Mesh Mesh algorithms [13,14] discover redundant paths between the source and destination regions to address issues that arise from node link failures. In such schemes, the forwarding zone approach of LBM is used to create the mesh network. Once this is established, packets are forwarded using the routing paths of the mesh. The overhead of this type of geocasting routing is mainly associated with the setup of the mesh and its maintenance over time. This approach may work reasonably in low mobility scenarios but unlikely to succeed in high mobility IVC/RVC networks. It may be necessary to update or even reconstruct an established mesh because of the rapid rate of changes in topology.

8.7.5. Routing without flooding

GeoNode Imielinski and Navas [20-22] considered the problem of geographic multi-point to multipoint routing in fixed networks. Their assumption is that the network has a cellular architecture with a GeoNode assigned to each cell. This results in two-level routing, one between a sender and the GeoNode and the other between a GeoNode and the destination region. This approach may be suitable for fixed networks but it seems unrealistic for IVC/RVC networks.

GeoTORA GeoTORA [24,55] is based on TORA (Temporally Ordered Routing Algorithm), which is an unicast routing protocol for ad-hoc networks. In TORA a directed acyclic graph (DAG) is maintained for each destination. It represents for each node the direction to the destination node. Thus, this approach can be used for forwarding a packet to a destination starting at any node.

As nodes are moving in IVC/RVC scenarios, the DAG needs frequent updates which can lead to instability. It also introduces a maintenance problem since relevant information about the network topology needs to be gathered to construct each node's DAG.

8.7.6. Summary of simulation results

The authors in [230] carried out some simulation with NS2. The simulated 802.11 network was configured with a 250m wireless transmission range and the number of nodes vary between 100 and 1000.

The first observation is that none of the protocols previously discussed have guaranteed delivery. Redundancy as in the flooding-based approaches may not provide high delivery success rates as expected. One reason is because redundant routes in highly dynamic scenario, for example vehicles travelling at high relative speeds, exhibit very short lifetime [200]. The delivery success rates for the protocols described above were between 95% and 100% for an edge length between two nodes of 1000 meters and this rate drops to between 10% and 30% with edges length of 3000 meters.

8.8. Time Synchronisation

IVC/RVC applications need a mechanism to establish a common sense of time among the vehicles on the road. Generated events should be associated with a timestamp at the source in order to determine the reliability and accuracy of received travel and traffic information - a recent event data will usually be more accurate [199].

We list two time frameworks that could suit these applications:

- *Relative time:* the sense of time is established with respect to an agreed time reference which could be an elected vehicle among a specific group. This is applicable to scenarios where preserving the order of events is the only required function from a time synchronisation service.
- *Absolute time:* in other cases preserving the order of events is crucial but it is not the only functionality needed. Applications might require event data to be timestamped when they occurred with an absolute time value with respect to a true time standard such as the universal coordinated time (UTC).

MAC protocols based on a slotted TDMA structure require time synchronization to align nodes to the commonly used slot structure. The authors in [231] proposed and analysed a decentralized time syncrhonisation protocol. The protocol avoids systematic timing drift of the nodes, in a steady state. Results show that all nodes are kept synchronous within a time interval of twice the propagation delay between them. It is unclear, however, how the protocol deals with rapid changes in network topology which is a characteristic of IVC/RVC networks.

The question that remains to be explored is to what extent decentralised time synchronisation protocols are really a requirement when we assume the majority (at least those which have wireless communication) will be equipped with GPS receivers. Depending on the type of receiver, a time with an error below 1 microsec can be achieved.

8.9. Simulation: more real world models

Many applications require a model of the real world. The question is how such a model can be designed and validated. Traffic simulation for Intelligent Transport Systems can be classified in two categories: (a) microscopic modelling - suitable for group communication as the applications are often concerned with local behaviours of vehicles (e.g. instantaneous velocity and position); (b) macroscopy modelling - the mobility pattern is defined by four parameters: average car speed v (m/s), traffic density (in vehicles/km), traffic flow (in vehicles/second), and net time gap in second. Some of these parameters are assigined according to normal or uniform distribution.

The mobility patterns are quite different from the random waypoint model that is extensively used for ad- hoc network simulations. There are a few simulators built and used for the simulations in the literature. We refer to the PATH/CORSIM and GloMoSim that can simulate a vehicle network in addition to NS2 that is commonly used to simulate the network communication protocols. Other research have used MatLab and Simulink in order to simulate the vehicular network [203].

Although there are research work that have used realistic data such as the study discussed in [211] where GPS data collected from actual buses in San Francisco were used, there is a lack of realistic data in the majority of the simulated work that includes real road/traffic conditions and also realistic vehicle models. The research community should attempt to incorporate more GIS data with dynamic information of the traffic systems.

We argue that complete and realistic simulators can only be built when we take a multidisciplinary re- search approach that includes simulations from areas such as GIS data, network/protocol, road transport and health/safety.

Part III.
Comparative study of algorithms and paradigms

This section will provide a reasoned classification of the literature collected by the partners. The aim is to clearly identify the most interesting issues regarding CO–based systems together with a classification of the most promising design paradigms, algorithms and communication patterns in this context. Finally, considerations on the possibility to define a common framework for the paradigms to be followed in designing solutions for Cooperating Objects are proposed.

9. Classification of the concepts

This section provides a taxonomy of the concepts that have been dealt with in this study, according to the applications requirements and characteristics defined by Study 3.1.1 and summarized in the following table.

REQUIREMENTS & CHARACTERISTICS	
Topology	Scalability
Fault Tolerance	Localization
Data Traffic Characteristics	Networking infrastructure
Mobility	Node heterogeneity
Power Consumption	Real Time
Reliability	

Applications requirements and characteristics defined by Study 3.1.1

9.1. Classification of the Thematic Areas

The four thematic areas have been selected pursuing the aim of covering most of the aspects that concern the various types of CO–based systems. In the following, each thematic area is analyzed according to the list of aforementioned requirements.

9.1.1. Wireless Sensor Networks for Environmental Monitoring (WSNEM)

A WSNEM is characterized by a large number of stationary sensor nodes, disseminated in a wide area, designated to collect information from the sensors and act accordingly. For example, a WSNEM can be used in greenhouses to monitor the environmental conditions (air humidity/temperature, light intensity, fertilizer concentration in the soil, and so on) and deliver the sensed data to a central controller that determines the actions to take (activating watering springs, sliding shatters, air humidifiers, and so on). Also, WSNEMs may be used for monitoring the integrity of buildings, bridges or, more generally, structures. Sensors inserted in the structure can detect any significant variations of pressure, positions of landmarks, or relative position of the surrounding sensors and send data to a controller node, either spontaneously on in response to an explicit solicitation of the controller. Another possible example is the so–called Smart Environments. The environment is equipped with sensor nodes that allow interaction with people. In particular, a WSN can track the movement of objects or persons in a given (also wide) area, thus providing the basic functionalities for the development of surveillance applications.

- *Topology.* The topology can be: pre–planned, semi–random or random. In the first case, nodes are accurately placed in the field for different purposes, as for monitoring the deformation of rigid structures or the environment in a specific area (greenhouses). A semi–random topology is obtained when most of the nodes are randomly scattered over the area, but some specific nodes are placed in specific positions, for instance to guarantee connectivity or to act as beacons for the other nodes. Finally, a random topology is obtained when nodes are scattered in the area without any plan, as in the case of airplane dissemination of sensors over a forest or a contaminated area. Although nodes are static, topology changes can still occur because of the ON/OFF cycles that the nodes go through to spare energy and because of the dynamic nature of the wireless medium.
- *Scalability.* Such networks can be composed by thousands of nodes. Therefore, the scalability is a primary issue that has to be dealt with.
- *Fault Tolerance.* Nodes are prone to failure due to energy depletion or physical crashing.
- *Localization.* Localization can play a important role in case of random or quasi–random topologies. Furthermore, WSNEM can also be used to trace the motion of (non cooperative) objects in the area, as the motion of enemy troops in a battlefield, or wild animals in a forest, and so on. Notice that, the case of cooperative moving objects can be referred to the scenario WSNMN.
- *Data Traffic Characteristics.* Traffic patterns are rather peculiar and differ from classic all–to–all paradigm considered in classic ad hoc networks. Flows are mostly unidirectional, from sensors to one or more sinks and vice versa (greenhouse). Traffic is usually light (very low average bit rate), though some future applications may be requiring the transmission of heavy data bursts (e.g., image transfer). Data may show strong spatial correlation. For instance, the readings of temperature sensors that are placed in close proximity each other will be strongly correlated.
- *Networking infrastructure.* WSNEMs do not rely upon any network infrastructure. However, interest is being devoted to the connection of WSNEMs with other networks (e.g., WLANs).
- *Mobility.* Nodes are normally static. In some cases, nodes position may undergo small variation due to external causes (e.g., wind, quake, vibrations, and so on).
- *Node heterogeneity.* Nodes can differ in their sensing, computational and transmission capabilities. However, the literature is mainly focused on homogeneous networks.

- *Power Consumption.* Since nodes are usually battery powered and not (easily) rechargeable, power consumption is a primary issue.
- *Real Time.* Real time can be a requirement for WSNEM when the detection of some events (fire, intrusions, and so on) has to be notified to the sink/controller in a strictly limited time period.
- *Reliability.* Reliability of data transport in WSNEM may or may not be a requirement, depending on the specific type of application considered. Clearly, each scenario that involves safety, surveillance or, in general, monitoring of potentially hazardous events, data reliability is a primary issue. However, many other application scenarios for WSNEM may exploit data redundancy to increase their robustness to data corruptions (e.g., climate monitoring in greenhouses, temperature controllers, smart environments and so on).

9.1.2. Wireless Sensor Network with Mobile Node (WSNMN)

A WSNMN is characterized by the use of mobile nodes in a WSN, ranging from a network with only mobile nodes to a network with a trade-off between static and mobile nodes. The use of mobile nodes in sensor networks increases the capabilities of the network and allows dynamic adaptation with the changes of the environment. Some applications of mobile nodes could be: *collecting and storing sensor data in sensor networks* reducing the power consumption due to multi-hop data forwarding, *sensor calibration* using mobile nodes with different and eventually more accurate sensors, *reprogramming nodes* “by air” for a particular application and *network repairing* when the static nodes are failing to sense and/or to communicate.

- *Topology.* The topology can be pre-planned, semi–random or random. The first case appears when the static nodes are deployed in known places and mobile nodes have a controlled and predicted motion. The second case can be found when either the static nodes are randomly placed or the mobile nodes have uncontrolled and non predicted motion. Finally, a random topology is obtained when the static nodes are placed without any plan and the mobile nodes move uncontrolled and with a non predicted motion. On the other hand, the communication can be single-hop because the mobile nodes can approach to the different static sensor to collect data. However, multi-hop communication can be used in other cases, for example when the number of static nodes is much higher than the number of mobile nodes. Finally, it should be mentioned that the topology is dynamic due to the presence of mobile nodes .
- *Scalability.* As it has been stated in the section about WSN with static nodes, scalability is a primary issue. Mobile nodes can improve and make easier the scalability of the network, because those nodes can manage different regions with a fixed number of static nodes, allowing communications between them.
- *Fault Tolerance.* The use of mobile nodes can increase the fault tolerance of a WSN, because a mobile node can be used to replace a non-working node or to calibrate the different sensors of the static node.

- *Localization.* This is an important issue due to the use of mobile nodes. Furthermore, the mobile nodes can be used to localize the static nodes with random or quasi-random topologies.
- *Data Traffic Characteristics.* In WSNMN, the mobile nodes can be used as sinks that send the information to a central station. Usually, the traffic between static and mobile nodes is low, whereas the traffic among mobile nodes and the central station is high. Finally, the mobile nodes can also process the information gathered from the static nodes in order to reduce the data traffic.
- *Networking infrastructure.* WSNMN does not need any networking infrastructure. However, it could be interesting to connect the WSNMNs (or at least some mobile nodes) to other networks, such as Internet.
- *Mobility.* WSNMN considers a range of configurations, from WSN composed only by mobile nodes to WSN composed mainly by static nodes and only a few mobile nodes.
- *Node heterogeneity.* Nodes can be static or mobile. Within these two groups, nodes can be heterogeneous in terms of sensing capabilities, batteries, locomotion system, etc.
- *Power Consumption.* Mobile nodes can reduce the power consumption due to multi-hop data forwarding. Furthermore, it is possible to have places where the mobile nodes can recharge their batteries when a low energy level is detected. In any case, power consumption is also a primary issue in WSNMN.
- *Real Time.* When a mobile node has to move sequentially to approach different static nodes in distant locations, an important disadvantage is the latency required. Then, in general, this strategy can be applied only in case of delay tolerant scenarios.
- *Reliability.* Mobile nodes can increase the reliability of the network because they can be used to repair or increase the connectivity of the network deploying new static nodes.

9.1.3. Autonomous Robotics Team (ART)

One of the main reasons that could lead to the development of a multirobot system may be that it is possible to build more robust and reliable systems by combining unreliable but redundant components. Research in multi-robot systems has focused primarily on construction and validation of working systems, rather than more general analysis of problems and solutions. As a result, in the literature, one can find many architectures for multi-robot coordination, but relatively few formal models. Those models can only be found when addressing specific aspects of a multi-robot system, such as path planning or task allocation for example, but not for the whole architecture. On the other hand, a combination of multirobot systems with wireless sensor networks can improve the overall reliability and performance of the whole system. Benefits can be identified in both directions: for example, a multirobot system could provide network repairing services to a WSN, and a WSN could provide extended sensorial capabilities to a multirobot team.

- *Topology.* In ART, topology can be pre–planned because the motion of the robots is always in some extent controlled and predictable.
- *Scalability.* In a multirobot system, the amount of data to process is quite huge: the processing of this knowledge in a centralized way requires obviously powerful computational means. Moreover, such a centralized computation reaches its limits when the number of robots increases (for example in swarm robotics): a centralized system can not be scalable to any number of robot. In contrast, a distributed approach can stay available when increasing the number of robots, since each robot still only deals with a local partial knowledge of the system, that leads to manipulate local information.
- *Fault Tolerance.* Here, by fault tolerance, we mean the ability of the robot team to respond to individual robot failures or failures in communication that may occur at any time during a mission.
- *Localization.* Many techniques have been developed in robotics in the last decades in the research field of localization. Each robot can localize itself by using a satellite global positioning system (GPS) in outdoor scenarios, beacons or landmarks in the environment, etc. A perception system involving several sensors such as cameras (visual and infrared), ultrasonics, laser, etc. can be also used in combination with odometry techniques and those systems to localize the robot. In ART, new techniques can be applied to exploit the information from other robots in terms of localization improvement.
- *Data Traffic Characteristics.* The data traffic among the robots is usually higher (images, telemetry, etc.) than the traffic among the nodes of a WSN. Moreover, traffic patterns are more similar to the classic all–to–all paradigm considered in ad hoc networks.
- *Networking infrastructure.* In ART, a networking infrastructure is not usually required. However, if a standard wireless communication system is used (i.e., WiFi), some kind of infrastructure such as an access point will be needed. In any case, it is usually interesting to connect the robots to an external network such as Internet for monitoring, remote control and supervision of the tasks execution.
- *Mobility.* In ART, mobility is implicit since the interactions with the environment usually requires mobile robots or robotic manipulators.
- *Node heterogeneity.* In general, heterogeneity introduces complexity since task allocation becomes more difficult, and robots have a greater need to model other individuals in the group. But, differentiation can also be treated as an advantage in terms of complementarity between different components in order to complete a given task or mission.
- *Power Consumption.* Robots can autonomously recharge their batteries when a low level of energy is detected. Furthermore, mixed solutions including solar panels are also possible.
- *Real Time.* Real time can be a requirement for ART depending on the particular application considered.

- *Reliability.* This requirement should be addressed during the design of the ART in order to provide mechanisms and algorithms to increase the reliability during the operation of the system, allowing to detect and overcome various sources of system failure, like the reduction in the number of visible GPS satellites or communication breakdowns. Perception techniques and special planning functions can be used to detect and monitor a faulty robot allowing to increase reliability in the execution of a mission.

9.1.4. Inter Vehicular Networks (IVN)

Cooperation among vehicles is a promising approach to address critical road safety and efficiency in IVN scenarios. For instance, coordinated collision avoidance systems could significantly reduce road accidents. The road safety becomes rather problematic in countries which heavily rely on the road network for transportation of goods.

Despite the fact that promising results have been achieved in some of the research projects in this area, major issues remain to be resolved. This might explain the large list of proposed traffic applications with no real deployment. Inter Vehicular Communications (IVC) exhibit characteristics that are specific to scenarios of *high node mobility*:

- *Topology.* Despite the constraints on the movement of vehicles (i.e., they must stay on the lanes), the network will tend to experience very rapid changes in topology.
- *Scalability.* Such networks can be formed by thousands of vehicles that can interact to each other. Thus, scalability is major issue that needs to be addressed.
- *Fault Tolerance.* It is important to understand how IVC/RVC networks can detect and recover from faults. Fault tolerance is more easily achieved with a hybrid of ad-hoc approaches for direct inter-vehicle communication and infrastructure-based schemes. The latter could introduce fallback mechanisms through a pre-deployed network backbone.
- *Localization.* Location is a key characteristic in IVN. The design choices for localization schemes in this scenario range from centralised but inexpensive satellite positioning systems (e.g. GPS) to decentralised systems. An important aspect to consider is the accuracy of localization systems and its implications to the applications, especially safety applications. For instance, routing strategies need to consider an error region while making the decisions to forward packets.
- *Data Traffic Characteristics.* Some of the IVN applications require communication with destinations that are groups of vehicles. Thus, the type of traffic should be predominantly multicast, in particular geocasting with physical areas of coverage.
- *Networking infrastructure.* Some of the IVC approaches assume a pre-deployed infrastructure, say a cellular network. Others rely on ad-hoc communication when vehicles may communicate where essentially no pre-planned infrastructure is present. IVC can be supported by any of these two approaches. However, road-to-vehicle communication is often an infrastructure-based scheme as ground support including pre-deployed road sensors needs to be in place.

- *Mobility.* The high mobility of vehicles poses issues that need to be addressed. The communication can be severely impaired by the high relative velocities of vehicles, even when moving in the same direction. In ad-hoc short-range communication, the network will tend to experience very rapid changes in topology. Also, constraints on mobility may impact the coverage of the network in cases where no ground infrastructure (e.g base stations) to support the communication between vehicles is present. Adding more nodes to the network may not improve coverage but rather affect negatively the use of available bandwidth.

- *Node heterogeneity.* Unlike typical WSNs deployment scenarios, vehicles can be instrumented with powerful sensors and radio to achieve long transmission ranges and high quality sensor data. Interoperability between different types of devices is a major issue.

- *Power Consumption.* Low-power consumption and small physical size have been some of the fundamental issues in designing software and hardware for low-cost wireless sensor nodes. However, vehicles exibit more physical space and energy consumption is not an issue.

- *Real Time.* Safety applications will not tolerate high overall latency including network and processing delays.

- *Reliability.* In this scenario, reliability is the application's tolerance to errors made visible to it. A zero tolerance threshold indicates that the applications requires 100% guaranteed data delivery.

Although these characteristics have some impact on infrastructure-based IVNs, the major issues arise from ad-hoc network setups. Blum et al [200] pointed out that IVC networks have mobility characteristics different from the typical MANETs considered in the literature. Specifically, these characteristics cause rapid topology changes, frequent fragmentation of the network, a small effective network diameter, and limited utility from network redundancy. Such issues must be tackled from the outset since they directly influence the IVC system design at all communication layers.

The thematic areas taxonomy is summarized in the following table.

Taxonomy of the Thematic Areas				
	WSNEM	WSNMN	ART	IVN
Topology	Typically multi–hop Slow dynamic due to on/off duty cycles	Single–hop Medium dynamic due to node mobility	Multi–hop High dynamic due to nodes mobility	Multi–hop High dynamic due to nodes mobility
Scalability	Primary Issue	Primary Issue	Primary Issue	Primary Issue
Fault Tolerance	Prone to nodes failure	Partially resilient to nodes failure	Prone to nodes failure	Partially resilient to failures
Localization	Relevant	Primary Issue	Primary Issue	Primary Issue
Data Traffic Characteristics	Generally from the sensors to one or few sinks	Generally from the sensor to the mobile nodes	Robot–to–robot and environment–to–robot	Road–to–vehicle or vehicle–to–vehicle
Networking infrastructure	None	None	Possible interaction with access points	Possible interaction with cellular network or fixed access points
Mobility	None	Medium and limited to mobile nodes	Medium-High	High
Node heterogeneity	Possible	Always	Possible	Possible
Power Consumption	Energy efficient algorithms On/off duty cycles	Rechargeable mobile nodes	Rechargeable robots	Irrelevant
Real Time	Secondary Issue (depending on application)	Secondary Issue (depending on application)	Secondary Issue (depending on application)	Primary Issue
Reliability	Secondary Issue (depending on application)	Secondary Issue (depending on application)	Primary Issue	Relevant Issue (depending on application)

9.2. Classification of the algorithms

In this section we will mainly refer to the algorithms covered by the Study on Paradigms for Algorithms & Interactions. For further details on the algorithms described, please refer to the specific sections of the study.

9.2.1. MAC algorithms

- *Topology.* MAC algorithms can be classified on the basis of the topology information they need to operate. Topology independent MAC algorithms, as those based on CSMA (MACA, MACAW, PAMAS) or SMAC and DBMAC, do not require any knowledge of the network topology. Other protocols, like SIFT and STEM, require nodes to have local information only, i.e., information regarding the nodes in their proximity. Finally, topology dependent protocols, such as TRAMA, assume nodes are aware of the entire network topology.

- *Scalability.* Generally, the performance of MAC protocols, in terms of medium access delay, is affected by the number of contending users. Contention–based access protocols, such as MACA, MACAW, PAMAS and so on, scale rather well with the number of nodes, when the traffic offered to the network is low. On the contrary, with high traffic loads, random protocols performance (in terms of medium access delay) worsens rather rapidly as the number of nodes increases. Contention–free MAC algorithms (e.g., time–division based algorithms) scale better with high traffic loads, while for low traffic such solutions may incur in longer access delay than random algorithms.

- *Fault Tolerance.* In general, MAC algorithms are not affected by nodes failure, even though a certain performance loss may be experienced in case of topology dependent algorithms.

- *Localization.* MAC algorithms can be classified in location–aware and location–independent. Location–aware solutions usually follow a cross–layer approach, since the location information is used both to manage the access to the medium and the forwarding of the information towards the intended destination (see GeRaF, Smart Broadcast). Some algorithms assume only that each node is acquainted (in some way) with its own spatial coordinates, others require the knowledge of the positions of the surrounding nodes only or of all the nodes in the network. Pure medium access algorithms are generally location independent. Location–aware algorithms are usually much more efficient that location–independent algorithms. However, they may turn out to be excessively sensitive to localization errors. These aspects have not been sufficiently covered in the literature yet.

- *Data Traffic Characteristics.* Traffic characteristics may have a strong impact in MAC algorithms performance. Contention–based protocols usually show better performance in case of sporadic traffic bursts, while deterministic access mechanisms are more suited for handling periodic traffic generation patterns. At the state of the art, MAC algorithms do not consider the traffic flow patterns, i.e., the set of nodes that exchange data. An exception is represented by cross layer solutions that provide a integrated mechanism for both MAC and routing and are sometime designed according to the specific traffic flow pattern expected in the system.

- *Networking infrastructure.* Generally, the presence of a network infrastructure permits to resort to contention–free MAC protocols based on polling strategies or resource reservation. However, most of the protocols covered by this study can be operated in absence of any networking infrastructure.

- *Mobility.* Mobility might represent an issue for MAC protocols for two reasons. First, mobility involves topology variations that may affect algorithms that need to tune some parameters according to the density of nodes in the contention area (SIFT, TRAMA, TSMA, MACAW). Second, MAC algorithms based on medium reservation mechanisms (MACA, MACAW) may fail in case of mobility, since the reservation procedures usually assume static nodes. For instance, algorithms based on the RTS/CTS handshake to reserve the medium may fail because either the corresponding nodes move outside the mutual coverage range after the handshake or external nodes get into the contention area and start transmitting without being aware of the medium reservation.
  Nevertheless, many MAC algorithms considered in this study are capable of self–adapting to the topology variations in case of nodes mobility. Algorithms like TRAMA, TSMA and SMACS-EAR can still adapt to topology variations, but at the expense of the energy efficiency and the access delay.

- *Node heterogeneity.* MAC algorithms for heterogeneous network have not been yet investigated in the literature. Algorithms based on channel sensing (CSMA–based) provide some resilience to interference produce by other radio interfaces operating in the same frequency band and, hence, can be adopted in heterogeneous system. However, this solutions would not leverage on the nodes diversity. This topic is, indeed, still to be investigated in the literature.

- *Power Consumption.* Energy efficiency is considered in several MAC protocols, in particular in the case of wireless sensor networks. A typical method to reduce energy consumption is to let nodes alternate periods of activity and sleeping. Notice that such on/off cycles may be either managed independently of the MAC protocol or be part of it. For instance, CSMA, MACA, MACAW protocols do not explicitly consider the presence of such on/off cycles. Nevertheless, CSMA behaving is not affected by on/off cycles, while MACA and MACAW may fail since they assume nodes are always notified of the channel state. Protocols like PAMAS and SMAC, on the contrary, take into account the sleeping periods of the nodes, thus permitting a more efficient power management of the system. Usually, this is obtained at the cost of a higher complexity of the MAC protocol.

- *Real Time.* Contention–based MAC protocols cannot usually provide any real time guarantee. Conversely, contention–free algorithms, such as TSMA or TRAMA, are able to guarantee a given maximum access delay, which depends on the number of competing nodes.

- *Reliability.* Almost all the MAC algorithms considered in the study require explicit acknowledgment (ACK) of correct data reception from the receiver. Usually, in case of missing or negative ACK, the data link layer entity retransmits the data unit. However, the process is stopped when a given number of retransmissions is reached. In this case, the data unit is discarded. Hence, in general, MAC protocols can provide only limited reliability. Notice that, contention–based

MAC algorithms are prone to transmission errors due to collisions, events that, on the contrary, never occur in contention–free algorithms. Therefore, contention-based algorithms are typically less reliable that contention–free ones.

The MAC algorithms taxonomy is summarized in the following table.

Taxonomy of the MAC algorithms

	CSMA	MACA MACAW PA-MAS	SMACS-EAR	Sift	STEM	DB–MAC	TRAMA	TSMA	Energy–aware TDMA
Topology	Indep.	Indep.	Indep.	Only local	Only local	Indep.	Compl. top. knowledge	Only local	Compl. top. knowledge
Scalability	Partial (low traffic)	Partial (low traffic)	Partial (low traffic)	medium (low traffic)	medium (low traffic)	Partial (low traffic)	Good (high traffic)	Good (high traffic)	Good (high traffic)
Fault Tolerance	Resilient	Resilient	Resilient	Resilient	Resilient	Resilient	Partially Res.	Partially Res.	Partially Res.
Localiz.	Not required	Not required	Not required	Not required	Not required	Not required	Not required	Not required	Not required
Data Traffic Characteristics	Better for sporadic traffic	Better for sporadic traffic	Better for sporadic traffic	Better for sporadic traffic	Better for sporadic traffic	Better for sporadic traffic	Better for periodic traffic	Better for periodic traffic	Better for periodic traffic
Netw. infrastruct.	None	None	None	None	None	None	None	None	None
Mobility	High resilience	Medium resilience	Medium resilience	Medium resilience	Low resilience	Medium resilience	Low resilience	Low resilience	Low resilience
Node heterogeneity	None	None	None	None	None	None	None	None	None
Power Consumption	High	High	Medium (on/off)	High	Low (sleep)	Medium (Data aggr.)	Low (sleep)	Low (sleep)	Low (sleep)
Real Time	Partial	Partial	Partial	Partial	No	Partial	Yes	Yes	Yes
Reliab.	Partial	Partial	Partial	Partial	Partial	Medium	High	High	High

9.2.2. Routing algorithms

- *Topology.* Routing algorithms can be differentiated on the base of the routing topology they realize. Usually, table–based algorithms create tree topologies, so that each node is the root of a routing tree towards each other node in the network.
  On–demand routing algorithms, on the contrary, realize point to point routing topologies, where a path from a node to its destination is created when needed (GedRaF, GAF, GEDIR). Cluster–based routing algorithms construct a hierarchical topology, where some nodes are elected as cluster–heads (or coordinators) and forward data collected from their neighbors towards the final destination (LEACH, PEGASIS, TEEN). Request–driven routing algorithms aim at defining a path from possible multiple sources to the node that issues a specific data request. These algorithms lead to a star–shaped routing topology, where many paths originating from the source nodes converge to the destination node. Examples are Rumor Routing, Direct Diffusion routing.

- *Scalability.* Scalability is a important issue for routing protocols. Table–based protocols usually show scalability problems when the number of nodes (and, consequently, routing table entries) grows, in particular for systems with limited storing and computational capabilities (typically sensors networks). To alleviate this problem, many protocols resort to clustering techniques that, in turn, bring forth some control overhead (LEACH, TEEN, PEGASIS). State–less algorithms have been introduced to cope with scarce storing capabilities, while maintaining good scaling properties. Typical examples are location–based algorithms, such as GeRaF, GAF, GEDIR, where nodes need to maintain the information regarding their own location and that of the destination node. Notwithstanding, the literature does not consider in detail the issue of distributing and maintaining the location information over the network.
  Algorithms that make use of broadcast packets to gather and/or diffuse topological information usually show scalability problems in large network due to the broadcast storm problem, unless broadcasting is obtained by means of specific broadcast–diffusion algorithm (Direct–diffusion, Rumor routing).

- *Fault Tolerance.* Usually, routing algorithms can adapt to topology variations due to nodes failure. However, the reaction to a topology variation due to nodes failure may require some time and, hence, bring some performance degradation. During this time, data can be delayed, duplicated or lost.

- *Localization.* As seen for the MAC algorithms, also routing algorithms can be classified in location–aware and location–independent. Location–aware routing algorithms include the cross–layer solutions discussed in the classification of MAC algorithms, as GeRaF, and other pure routing algorithms, such as GAF, GEDIR, GEAR. Usually, location–aware routing algorithms assume that each node is acquainted (in some way) with its own spatial coordinates and those of the intended destination node. Hence, the next hop is determined in order to move the packet towards the destination.
  Data centric routing algorithms, such as Direct-diffusion, Rumor routing, SPIN, make use of broadcast techniques to disseminate and gather routing information and, therefore, do not require any localization feature.

Hierarchical routing algorithms, in general, are based on topological information, though do not require exact node localization (LEACH, TEEN). Nevertheless, localization may help the process of creating the cluster structure, thus resulting in better performance (VGA, TTDD).

- *Data Traffic Characteristics.* Data traffic characteristics may affect routing algorithms design. In WSN, for instance, spatial correlation among data generated by nodes in close proximity is exploited by cross layer solutions that merge routing and data processing functionalities (TEEN, VGA, COUGAR). Specific routing algorithms have been proposed for centralized traffic patterns, where information flows to and from a single central node (e.g., a sink node in WSN) and several peripheral nodes. Examples are SOP and MCFA.
- *Networking infrastructure.* Most of the routing algorithms for cooperating objects are designed according to an ad hoc paradigm. Therefore, solutions are completely distributed and do not require any backbone infrastructure.
- *Mobility.* Generally speaking, all the routing algorithms considered are able to cope with topology dynamic due to nodes mobility. However, most of them react to topology variations by dropping the broken paths and computing new ones from scratch, thus incurring in performance degradations. In particular, mobility may strongly affect cluster–based algorithms, due to the cost for maintaining the cluster–architecture over a set of mobile nodes. Routing algorithms specifically designed for networks with slow–mobile nodes are, for example, GAF and TTDD, which attempt to estimate the nodes trajectories. Other protocols that are well-suited for an environment where the sensors are mobile are the SPIN family of protocols because their forwarding decisions are based on local neighborhood information.
- *Node heterogeneity.* Node heterogeneity can be a winning feature to develop efficient routing algorithms, in particular for WSNs with mobile nodes. Notwithstanding, the literature still lacks in solutions that leverage on nodes heterogeneity to enhance the routing process.
- *Power Consumption.* Power consumption is typically a very important issue in the design of routing protocols, since many cooperating–objects systems involve battery–powered units. Accordingly, several energy–efficient routing algorithms have been presented in the literature, in particular for WSNs. The simplest manner to reduce power consumption is to allow each node to schedule sleeping periods. Therefore, routing protocols have to be designed to work also in the presence of ON–OFF duty cycles (GAF, ASCENT).

  Other protocols to reduce the amount of information improve the energy efficiency of the system is Moreover, other techniques such as data aggregation, overhead reduction, cluster–heads rotation and so on can be used to reduce the energy wasting (LEACH, GEAR, PEGASIS, TEEN, HPAR).
- *Real Time.* Routing algorithms that can provide tight constraint on the packet delivery time are rather seldom.
- *Reliability.* Most of the considered routing protocols cannot guarantee data reliability, especially when the network is rarely populated. Some routing algorithms, such as GeRaF, GAF, GEDIR,

SPIN, may fail to discover a path in case of connectivity holes within a connected network. Other algorithms can, instead, guarantee delivery if source and destination nodes are connected (GOAFR, SPAN, LEACH, PEGASIS). Broadcasting–based algorithms, such as Rumor routing and Direct diffusion, generally offer high reliability thanks to the capillary diffusion of the routing control packets.

The routing–algorithm taxonomy is summarized in the following table.

Taxonomy of the Routing algorithms

	GeRaF	GAF SPAN	SPIN	LEACH PEGASIS	TEEN	HPAR	Rumor Direct–Diffusion
Topology	Point–to–point	Point–to–point	Star	Hierar.	Hierar.	Hierar.	Star
Scalability	Good	Good	Good	High	High	High	Low
Fault Tolerance	High	High	High	Medium	Medium	Medium	High
Localization	Required	Required	Not required	May help	May help	May help	Not required
Data Traffic Characteristics	Irrelevant	Irrelevant	Relevant	Relevant	Relevant	Relevant	Relevant
Networking infrastructure	None	None	None	None	None	None	None
Mobility	High resilience	High resilience	High resilience	Medium resilience	Medium resilience	Medium resilience	High resilience
Node heterogeneity	None	None	None	None	None	None	None
Power Consumption	Low (on-off)	Low (sleep)	Medium (data aggr.)	Low (clust.)	Low (clust.)	Low (clust.)	High
Real Time	No	No	No	No	No	No	No
Reliability	Partial	Partial	Medium	Medium	Medium	Medium	High

9.2.3. Localization algorithms

- *Topology.* Localization algorithms are used to infer the geographical position of a node by elaborating the signals received from position–aware nodes (beacons/landmarks). The precision of the estimation is usually strictly dependent upon the placement of the beacons. Therefore, the network topology may have effects on the performance of most localization algorithms. For example, in case of range–free approaches, inhomogeneous nodes density may lead to incorrect distance estimate (Centroid, DV-Hop).
- *Scalability.* Localization algorithms are usually scalable with the network population. However, if the geographical extension of the network increases, a higher number of beacons may have to be deployed (APIT). Even if a multilateration approach is adopted, relaxing the need for direct beacons visibility, an increasing of the average number of hops from the beacons leads to localization errors accumulation (DV-Host, DV-Dist). The complexity of the localization algorithms, as well as the preciseness of the estimation, usually increase with the number of beacons (AHLoS). To conclude, the localization algorithms, in general, might show scalability problem with the number of nodes that populate the network.
- *Fault Tolerance.* The localization algorithms are usually tolerant to the dead of some nodes, given that they are not beacons. Failure of beacons is, instead, particularly critical for localization algorithms performance (Centroid, DV-Hop, DV-DIST). Also, malfunctioning nodes, for instance nodes with defecting HW, may have an impact on localization errors and on localization error propagation (DV-Hop, N-Hop TERRAIN, AHLoS).
- *Localization.* Localization algorithms require, in general, a suitable disposition of the beacon nodes in the network area. Estimation may also be refined by using the positioning information estimated by the surrounding nodes.
- *Data Traffic Characteristics.* To estimate the position, sensor nodes use the control packets sent by beacons that contribute to the network load (DV-Hop, DVB-Distance, N-Hop Terrain). In the case of networks with mobile nodes, furthermore, the position estimation might be improved by increasing the beacons frequency. Hence, a tradeoff between localization accuracy and network load can arise. Furthermore, some localization algorithms might make use of data packets sent by position–aware nodes to adjust their position estimation. In this case, regular or periodic data traffic exchange involving position–aware nodes can improve the performance of the localization mechanisms without introducing extra control traffic.
- *Networking infrastructure.* In general, localization algorithms make use of infrastructures. In the specific, satellite–based positioning mechanisms obviously require a complex satellite network infrastructure. More generally, localization algorithms require the presence of a network infrastructure that hosts beacon nodes, whose positioning information is disseminated over the network. Localization algorithms that aim at providing only relative positions of a node in a network, on the contrary, do not require any settled infrastructure.
- *Mobility.* In general, nodes mobility increases the localization error. However, in some context, mobile nodes capable of accurate position estimation might be used to disseminate positioning

information over a network of elementary static nodes.

- *Node heterogeneity.* The use of the satellite–based positioning systems is not always possible, for it increases the cost of the nodes and the power consumption. The heterogeneity of nodes play a fundamental role in these scenarios, since a bunch of localization–enabled nodes might be exploited by the other nodes of a network to derive an estimation of their position (APIT).
- *Power Consumption.* Localization schemes increase the power consumption. In particular, the use of satellite–based schemes is very expensive in terms of power consumption in some contexts (such as WSN). This cost might be reduced by installing a limited number of such devices in the network (beacons) and by using localization algorithms to estimate the position of the other nodes in the network. Clearly, in this case the energy consumption is due to the control packets exchange (DV-Hop). Other localization strategies encompass the use of ultrasonic transceivers (Cricket, AHLos) that, however, determine further energy consumption.
- *Real Time.* In general, satellite–based positioning system are capable to provide quasi real time localization service. On the contrary, localization algorithms that are based on the elaboration of beacon signals are not suitable for strictly real time applications, since, in general, they require the reception of several control packets to reduce the estimation error. On the other hand the problem of whether a real time application can be supported or not becomes an issue only for mobile networks of cooperating objects. In many application scenarios in which nodes are instead static the localization process can be performed at the network set-up, reducing its costs and allowing to use the different types of algorithms independently of the real-time constraints of the application.
- *Reliability.* Reliability of the localization algorithms depends of the number and position of the beacon nodes, possibly the number of hops over which the localization error propagates, the presence of malfunctioning or malicious nodes. Malicious nodes are nodes whose purpose is to compromise the correct operation of the network. Such nodes can provide for example wrong ranging estimates or wrong information on their own position to other nodes, affecting the other nodes localization accuracy or the ranging estimate accuracy (DV-Hop, N-Hop TERRAIN, AHLoS). Ways to detect and filter the information provided by malfunctioning or malicious nodes have to be provided.

The localization–algorithm taxonomy is summarized in the following table.

	Range–free				Range–Based		
	Centroid	DV-Hop	APIT	Monte Carlo	AHLoS, N–Hop Multilat.	DV-DIST, HOP-TER.	AFL
Topology	Symmetric	Uniform	Generic	Generic	Generic	Generic	Generic
Scalability	Good	Limited	Good	Good	Good	Limited	Very Good
Fault Tolerance	Partial	Partial	Good	Good	Partial	Partial	Good
Localization	Coarse	Good/Coarse	Good/Coarse	Good	Good	Coarse	Good
Data Traffic Characteristics	Local	Flooding	Local	Local	Local	Flooding	Local
Networking infr.	Required	Required	Required	Required	Required	Required	Not required
Mobility	Fragile	Fragile	Required	Robust	Fragile	Fragile	Partially robust
Node heterogeneity	Landmarks	Landmarks	Landmarks	None	None	Landmarks	None
Power Cons.	Low	High	Medium	Medium	High	High	Low
Real Time	No	No	No	Partial	No	No	No
Reliability	Partial	Partial	Partial	Partial	Partial	Partial	Medium

9.2.4. Data Processing

- *Topology.* The network topology might play an important role on the design of specific data processing. The perfect knowledge of the network topology, for instance, can be used to determine the position of the better collector nodes. Moreover, if the topology is pre–planned, nodes with more computational capabilities can be displaced in strategic position. On the contrary, in case of random topology placed, the choice of more suitable aggregation points have to be taken in a distributed manner and can be less efficient. Regarding the organization of the network structure, the data processing techniques can lie on different communication topologies. Most known algorithms (Direct Diffusion, LEACH, PEGASIS, TAG and TiNA) run over tree–based or hierarchical structures. Differently, other schemes such as Synopsis Diffusion and Tributaries and Deltas organize the network in a concentric ring structure.

- *Scalability.* Scalability is an important goal in the design of efficient data processing techniques especially in large and dynamic networks. Existing data processing techniques based on the construction of some aggregation tree are less scalable than the multipath schemes due to the high cost to maintain the organization of the network. This characteristic is accentuated in large or dynamic networks where adding or removing some nodes from the tree structure heavily impact in the performance of the algorithms. On the contrary, multipath solutions offer a good scalability especially due to the local and distributed functionalities.

- *Fault Tolerance.* Data processing, in general, is performed in order to reduce the intrinsic data redundancy that might characterize some cooperating–objects scenarios (e.g.,WSNEM). On the other hand, data redundancy may assure a higher reliability in case of sensor failure, connectivity holes and so on. Hence, a tradeoff between fault tolerance and redundancy reduction has to be cut. More in detail, in case of low packet loss probability, tree–based algorithms achieve better performance because they are able to minimizing the number of transmissions to deliver data reducing as much as possible the redundancy. On the contrary, the multipath schemes preserve some data redundancy so that perform better in case of high packet loss probability. There are also some hybrid approach such as Tributaries and Deltas which are able to tune their behavior according to the link conditions.
- *Localization.* In some case data aggregation techniques require information about the location of nodes. Nevertheless almost all data processing techniques do not require any type of localization methods.
- *Data Traffic Characteristics.* The design of data processing techniques is strongly correlated to the specific considered application. In some cases, for instance in WSNEM, may be useful to perform data aggregation as near as possible to the data sources due to the high redundancy among data collected in the same spatial region. In other cases, data processing can be performed along the path, for instance to merge information flows directed to a common destination.
- *Networking infrastructure.* Data processing can take advantages from the presence of networking infrastructure. For instance, access points can play the role of data collectors and perform any type of simple data processing before forwarding information to the end destination. But at this time, none of the proposed algorithms make use of existing network infrastructure.
- *Mobility.* Mobility might improve the efficiency of data aggregation techniques. For instance, nodes with controlled or predictable motion can be driven all over the network to collect, process and store data generated by static nodes. On the contrary, mobility can affect the performance of the data processing schemes based on the aggregation tree.
- *Node heterogeneity.* Data processing may exploit objects with higher storage and computational capabilities as aggregation centers, in order to convey and process data from less–powerful objects displaced in a common region. Moreover, it is necessary to take into account the different node capabilities when the aggregation functions or the data structures are designed. For instance the proposed Q–digest structure can be used to store data with a different degree of precision according to the storage capabilities of the nodes.
- *Power Consumption.* Data processing techniques are, in general, implemented to limit the energy consumption by reducing the amount of transmitted data or the network overhead. Nevertheless, the required processing power contributes to deplete the energy resources of

collector nodes. This aspect, however, is rarely considered in the literature and needs further investigation. At this time, aggregation functions implemented by algorithms such as Direct Diffusion, LEACH, TiNA, TAG are very simple (in general they are statistical function) and they do not require additional power consumption.

- *Real Time.* Data processing techniques usually involve time delay for gathering and processing many data units from different sources. Consequently, such techniques might not be guarantee real–time requirements. This drawback is independent by the algorithm because it derives form the need to collect more than one packet before aggregating data and sending a new packet
- *Reliability.* Data processing techniques are usually reliable, though, as mentioned, they might incur in large delays that could affect the utility of the delivered data for the final node. Also in this cases, multipath strategies could guarantee a higher reliability than the tree–based schemes.

The data processing algorithm taxonomy is summarized in the following table.

	Direct Diffusion	LEACH, PEGASIS	TAG, Cougar, TiNa	Synopsis Diffusion	Tributaries and Deltas
Topology	Tree–based	Tree–based	Tree–based	Ring	Hybrid
Scalability	Low	Low	Low	High	Medium
Fault Tolerance	Low	Low	Low	High	High
Localization	None	None	None	None	None
Data Traffic Characteristics	Relevant	Relevant	Relevant	Relevant	Relevant
Networking infr.	None	None	None	None	None
Mobility	High resilience	High resilience	High resilience	Low resilience	Medium resilience
Node heterogeneity	None	None	None	None	None
Power Cons.	Low	Low	Low	High	Medium
Real Time	No	No	No	No	No
Reliability	Medium	Medium	Medium	High	High

9.2.5. Navigation algorithms

- *Topology.* Information about the topology can be used to improve the accuracy of the localization algorithms, providing a better performance of the navigation algorithm. The localization has a more significant impact on the performance of the path computation and following algorithm (PACFA) when comparing with others. Therefore, information about topology can improve the accuracy in the navigation in general, and especially with PACFA.

- *Scalability.* The navigation algorithms presented in the study use local information provided by the WSN and these algorithms have only been tested with one mobile node. Then, the scalability would depend on the capability of the communication protocol to support it. Furthermore, the use of a team of mobile nodes would involve other considerations such as the coordination among them for optimal covering of an area or collision avoidance for example. Those aspects could have a significant impact in the amount of information exchanged.

- *Fault Tolerance.* The information provided by WSNs used in the navigation algorithms improve the fault tolerance w.r.t. mobile nodes that only use the information from sensors installed on board.
- *Localization.* Localization of both the static nodes of the WSN and the mobile nodes is required for most of the navigation algorithms. The performance of the potential field guiding (POFA) and probabilistic navigation (PRONA) algorithms is more robust to localization errors than PACFA. If the localization of the nodes is not provided “a priori”, the navigation algorithm should also involve a position estimation.

- *Data Traffic Characteristics.* The navigation algorithms involve the exchange of a large amount of data among the static and the mobile nodes. These data should be updated at a rate which depends on the speed of the mobile node. PRONA does not require a “explicit” computation of the path and therefore a lower amount of data is involved. Finally, the increasing in the information exchanged due to the navigation algorithm can not exceed the capacity of the sensor network.

- *Networking infrastructure.* The navigation algorithms found in the literature only use local information from the nodes close to the mobile node. Then, there is not any special requirement regarding the networking infrastructure.
- *Mobility.* This is an intrinsic characteristic of these algorithms, that can be applied to guide a robot or a person with a suitable interface in a given environment.

- *Node heterogeneity.* It is also an intrinsic characteristic of these algorithms due to the fact that both static and mobile nodes are present. Even among the mobile nodes different characteristics, such as the locomotion system, are possible.

- *Power Consumption.* Power consumption of the nodes is increased due to the higher information exchange rate required during navigation. PACFA and POFA involve a first stage to compute the path, so a higher power consumption is required.
- *Real Time.* Real time requirements mainly depend on the speed of the mobile node. On the other hand, the navigation algorithms found in the literature are designed considering negligible delays in the information exchange among the nodes.
- *Reliability.* Besides the general reliability issues in WSNs, the reliability of the mobile platform itself must be also considered.

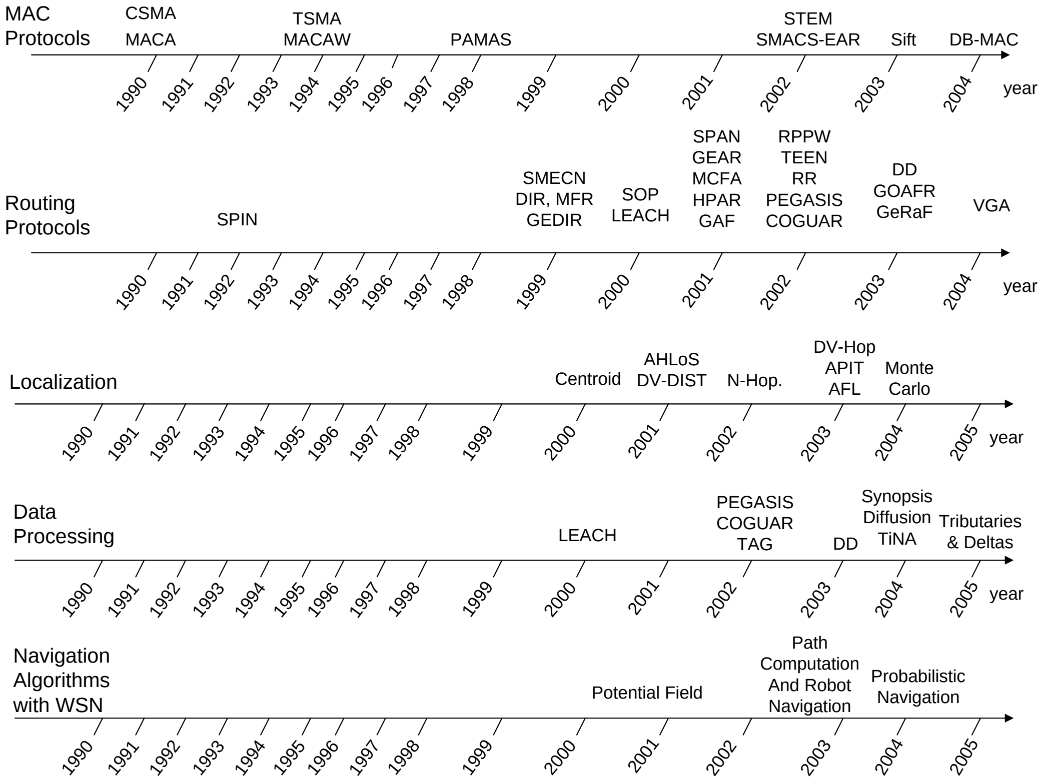
The navigation algorithm taxonomy is summarized in the following table.

Taxonomy of the Navigation algorithms

	POFA	PACFA	PRONA
Topology	Non-relevant	Relevant	Non-relevant
Scalability	Not considered	Not considered	Not considered
Fault Tolerance	High	High	High
Localization	Less required	More required	Less required
Data Traffic Characteristics	Very high rate	Very high rate	High rate
Networking infrastructure	None	None	None
Mobility	Intrinsic	Intrinsic	Intrinsic
Node heterogeneity	Intrinsic	Intrinsic	Intrinsic
Power Consumption	Very high	Very high	High
Real Time	Relevant	Relevant	Relevant
Reliability	High	High	High

9.2.6. Timetable of the literature

The following figure presents the different concepts that have been classified in this document according to the year of their appearance in the literature. The figure offers a one–look view of the trends evolution.



Timetable of the literature

10. Critical issues and research gaps

Localization The localization in wireless sensor network has been, and still is, an interesting research topic. Indeed, localization is both an interesting service that can be provided by the network for different purposes (object tracking, mapping, automatic driving, and so on) and a very powerful tool that can be exploited in the design of algorithms to run the networks itself. Many localization methods have been proposed, including the distance estimation based on the strength of the radio signals received by surrounding nodes, measurement of the time of flight of light and pressure impulses, broadcasting of geographical coordinates of beacon nodes, and so on. Unfortunately, despite the great interest that this topic has focused in the research community, the solutions proposed in the literature often reveal their limits when realized in practice. Indeed, first experiments in this direction have revealed that signal strength measurement does not allow a fine localization, even after long and precise calibration. This is due essentially to the unpredictability of the radio channel dynamic. Furthermore, energy level has impact on the transmission power of a node, so that the calibration process becomes rapidly loose when nodes progressively discharge their batteries. Mechanisms based on the comparison of the time of flight of impulses with different propagation speeds usually require either cumbersome equipment, or are energy expending or, else, prone to errors due to environmental noise. Therefore, the topic requires further investigation, in particular through experimental campaigns in various environmental conditions. Furthermore, the sensitivity of the location–based algorithm to the tolerance in the location information has not been sufficiently covered in the literature yet and deserves further investigation.

Sensor transmission range The sensor transmission range depends on the residual energy resource and the power depletion is different from node to node. Therefore, further research on the effect of this phenomenon on the protocols performance is necessary. Another promising scenario that deserves further investigation concerns heterogeneous networks, i.e., networks with nodes capable of larger transmission range.

WSN with multiple sinks Almost all the WSNs solutions considered in this document take into account networks with only one sink. However, the application scenarios suggest the possibility that WSNs can be characterized by the presence of multiple sinks. In this context, it is necessary, for example, to design and develop new routing protocols, new addressing mechanisms, more efficient localization methods and so on. These aspects need further investigation, both from a theoretical and practical perspective.

Characteristics of commercial sensors The capability of the hardware is a fundamental aspect in the design of efficient algorithms for WSN. For instance, some of the proposed solutions consider sensors equipped with two radios, which can be used to optimize routing and MAC procedures. It is important to notice that, although many protocols and algorithms have been proposed, most of the commercialized sensors are very simple and do not allow for complex solutions.

Experiments In the next future the research in WSN should be more experimental–oriented, in order to verify the limits of the theoretical analysis and to reveal the issues that inevitably arise when a system is really deployed in the field. This knowledge can, hence, be used to refine existing solutions and develop novel approaches.

QoS requirements QoS in WSNs has not been considered as a primary issues till now. However, emerging applications may require QoS-guarantees, thus making QoS support over WSN a primary issue.

WSN with mobile nodes It should be pointed out that WSN with mobile nodes is a relatively new topic and it is still a non-mature field. The following topics have been found in the literature about WSN with mobile nodes: localization, communications (MAC and routing), planning and reactivity. However, more algorithms and theoretical studies are needed in those fields. Furthermore, there are some topics that have been poorly addressed, such as self-hierarchical organization and/or clustering techniques, fault tolerance and coverage techniques. Therefore, more efforts should be devoted to those aspects and their study should be promoted in next years.

Energy constraints Concerning WSN, energy constraints is one of the most relevant issues and it should be interesting to take into account energy considerations in: static and mobile trade-off, computation of optimal parameters (speed, delay, etc) and relation between data collection and data diffusion.

Motion planning An important trend is related to the study of how the sensor network can compute, in a distribute way, the path that the mobile node must follow. Also, this path can be updated depending on changes of the environment or using new data collected by the sensor network. Also, it should be considered communication constraints and the relation with self-hierarchical organization and/or clustering techniques.

Cooperative perception The sensor network can be considered an extension of the sensorial capabilities of a mobile node, and therefore, an improved model of the environment can be built with the information from the WSN for navigation purposes.

Robotic teams networked with the environment Integration with wireless sensor networks and other cooperating objects should be studied in order to exploit many potential benefits and complementarities.

Multirobot Planning with reliability constraints has not been properly solved, specially in a distributed way. Hard real-time interaction with the environment (transportation, etc.) and the development of a set of metrics for the performance of a robotic team can be exploited.

Experiments More field experiments in all the thematic areas are definitely needed.

Heterogeneity Integrate and exploit heterogeneity in the different kind of nodes of a WSN should be taken into account, although heterogeneity implies complexity in the design and in the algorithms.

Evaluation of data processing costs Most of the described algorithms make use of some data processing techniques to perform data fusion, data aggregation, routing and so on, in order to maximize the efficiency of the network functionalities and reduce the power consumption spent in the transmission procedure. However, the computational and energy cost of these processing is rarely considered. Since, many cross–layer approaches gain advantage from data processing, the cost of these utilities and their feasibility in sensor networks should be addressed.

Technology contamination Very recently, the possibility to integrate the RFID technology in WSN, for instance to wake up nodes on demand, has been considered. Also, the perspective of integrating the WSN with other networks, such as the cellular system or the Internet, has been recently proposed, though the topic has not been yet addressed in the literature. This perspective of *Technology contamination* opens the way to several interesting possibilities that should be investigated in the next future.

11. Conclusions

This last section is intended to wrap up the study by making some considerations on the definition of a unified framework for the design of algorithms for the different CO–systems.

Study 3.1.2 was conceived to provide an in–depth analysis of the literature regarding cooperating–object systems and, hence, to identify a set of algorithms and architectures that could form a common framework for the next generation of CO–based systems. According to such a purpose, this document has been organized in three different parts. The first part introduces the study and states its purposes. The second part offers an overview of the literature on the paradigms for the design of algorithms and interaction patterns in CO–based systems. The third part, finally, provides a reasoned classification of the collected literature and draws some conclusions on the most interesting aspects of the subject, research trends and gaps to be filled in the next future.

Unfortunately, the heterogeneity and vastity of the subject make rather difficult to identify a common framework for the design of algorithms and communication patterns for CO–based systems. The Cooperating Object umbrella, indeed, encompasses systems with static and energy–limited nodes, very low duty cycles and very flexible delay constraints, as well as systems with autonomous mobile nodes, no energy supply problems, and strict requirements in terms of communication delay and reliability, as clearly arises from the analysis of the reference thematic areas considered in the study.

A first conclusion that can be drawn from this study, hence, is that Cooperating Objects may present irreconcilable discrepancies, which make hardly feasible the definition of a unified approach for the design of algorithms and solutions for this type of systems. Nonetheless, it is possible to identify some common *trends* that are transversal to the plethora of different design approaches.

Cross-layer

Two main approaches can be followed in the definition of algorithms for embedded systems: the *layered approach* and the *cross–layered approach*. The layered approach keeps a clear separation among the layers that realize the different network functionalities, thus simplifying the implementation and update of the algorithms. On the contrary, a cross–layered approach requires a joined design of the various network layers. This method usually permits to define more effective algorithms, by taking into consideration the interaction among the mechanisms that act at the different layers. In the case of WSN, for example, some physical parameters or application information can be exploited to define energy–efficient MAC or routing protocols. This performance improvement, though, is payed in terms of flexibility of the software structure, which becomes more difficult to update and maintain. Therefore, there is a tradeoff between the two approaches that has to be cut weighting the performance advantage obtained by adopting a cross layer approach against the difficulty to develop and maintain the software stack. The general trend is to prefer a cross layer approach, since the gain in performance that can be potentially achieved with a cross layer approach largely overcomes the drawbacks of a more complex protocol structure.

Distributed & Centralized architectures

From an architectural point of view, we can identify two general approaches: *Distributed* and *Centralized*. The first approach seems to be more suitable for WSNs due to the high number of nodes, the geographical extension of the network and the unpredictable network topology, which make rather difficult the realization of a centralized structure. Nevertheless, the literature shows that, while preserving a basic distributed approach, the realization of dynamic clustering structures permits a more efficient handling of the sensor resources. The trend is, hence, to follow a distributed approach in the design of algorithms for WSN, with the possibility to organize the nodes in a dynamic hierarchical architecture, which permits a more efficient management of the network resources. This advantage is particularly evident in the case of heterogeneous networks, i.e., networks with some more powerful nodes that are capable of longer battery autonomy, longer transmission range, bigger storing and computational capability.

Location–based solutions

Many algorithms for WSN assume the availability of location information on the nodes. Some algorithms assume only that each node is acquainted (in some way) with its own spatial coordinates, others require the knowledge of the positions of the surrounding nodes only or of all the nodes in the network. Algorithms that make use of localization information for managing the medium access or the routing are usually much more efficient that location–independent algorithms. However, they may turn out to be excessively sensitive to localization errors. These aspects have not been sufficiently covered in the literature yet.

Synchronous & Asynchronous paradigms

Algorithms that are designed to work upon a network with synchronous nodes are usually more efficient that asynchronous algorithms. However, keeping the synchronization in a large network of nodes is a demanding task, both in terms of energy and control overhead. Indeed, cheap hardware leads to large clock drift among the nodes, which have to exchange periodic information to maintain the synchronization. Furthermore, synchronous network are fragile with respect to long–term topology variations (addition/elimination of nodes), which, though, can be considered rather seldom events in static WSNs. On the other hand, asynchronous algorithms are much more flexible and easy to realize. If the cost to keep synchronization could be neglected, hence synchronous network would definitely overcome asynchronous solutions. However, the general trend, at least when considering protocols that are implemented in real networks, is to go for asynchronous solutions.

Wireless & Wired systems

Although the topic covered by this study refers exclusively to wireless networks, it is worth bearing in mind that many existing sensor networks make use of wired communication media. However, the trend is definitely towards the wireless solution which opens the way to a large variety of new applications, most of which in safety, surveillance and monitoring area.

References

[1] Y. S. I. F. Akyildiz, W. Su and E. Cayirci, "Wireless sensor networks: A survey," pp. 393–422, March. 2002.

[2] J. E. H. Callaway, *Wireless Sensor Networks: Architectures and Protocols*. Auerbach Publications, Boca Raton, FL, August 2003.

[3] A. K. Sumit Khurana and A. P. Jayasumana, "Effect of hidden terminals on the performance of IEEE 802.11 mac protocol." in *Proceedings of the 23rd Annual Conference on Local Computer Networks*. Institute of Electrical and Electronics Engineers, 1998.

[4] F. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II– the hidden terminal problem in carrier sense multiple access modes and the busy–tone solution," *IEEE Trans. on Communications*, vol. 23, pp. 1417–1433, Dec. 1975.

[5] C.-K. Toh, "Ad–hoc mobile wireless networks," *ACM Computer Communications Review*, July 1998.

[6] X. Kaixin and M. Gerla, "Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 107–123, September 2003.

[7] H. S. Chhaya and S. Gupta, "Performance modeling of asynchronous data transfer methods of IEEE 802.11 mac protocol," *Wireless Networks*, vol. 3, no. 3, pp. 217–234, 1997.

[8] M. Borgo, A. Zanella, P. Bisaglia, and S. Merlin, "Analysis of the hidden terminal effect in multirate IEEE 802.11b networks," in *Proceedings of WPMC04*, vol. 3, Abano Terme (Padova), Italy, 12-15 Sep. 2004, pp. 6–10.

[9] P. Karn, "MACA - A New Channel Access Method for Packet Radio," *AARRL/CRRL Amateur Radio 9th Computer Networking Conference*, Sep. 22 1990.

[10] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a media access protocol for wireless LANs," *ACM SIGCOMM Computer Communication Review*, pp. 212–225, Aug. 1994.

[11] S. Singh and C. Raghavendra, "PAMAS: power aware multi–access protocol with signalling for ad hoc networks," *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 3, pp. 5–26, July 1998.

[12] W. Ye, J. Heidemann, and D. Estrin, "An energy–efficient MAC protocol for wireless sensor networks," *Proc. 21st Int'l. Joint Conf. IEEE Comp. Commun. Soc. (Infocom 2002)*, vol. 3, pp. 1567–1576, June 2002.

[13] K. Jamieson, H. Balakrishnan, and Y. C. Tay, "Sift: a MAC protocol for event driven wireless sensor networks," MIT, Tech. Rep. LCS-TR-894, May 2003.

[14] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing Sensor Networks in the Energy–Latency–Density Design Space," *IEEE Transaction on Mobile Computing*, vol. 1, no. 1, pp. 70–80, January 2002.

[15] G. di Bacco, T. Melodia, and F. Cuomo, “A MAC protocol for delay–bounded applications in wireless sensor networks,” *Proc. 3rd Annual Mediterranean Ad Hoc Networking Workshop (Med Hoc Net)*, pp. 208–220, June 2004.

[16] M. Y. K. Arisha, M. Youssef, “Energy–aware tdma–based MAC for sensor networks.” in *IMPACCT 2002*, May 2002.

[17] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, “Energy–efficient collision–free medium access control for wireless sensor networks,” *Proceedings of the 1st ACM International Conference on Embedded Networked Sensor Systems*, pp. 181–192, 2003.

[18] A. F. Imrich Chlamtac, “Making transmission schedules immune to topology changes in multihop packet radio networks,” *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 23–29, February 1994.

[19] H. Z. Imrich Chlamtac, András Faragó, “Time–spread multiple–access (TSMA) protocols for multihop mobile radio networks,” *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 804–812, December 1997.

[20] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6–28, Dec 2004.

[21] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, “A two–tier data dissemination model for large–scale wireless sensor networks,” *Proc. 8th ACM Int’l. Conf. Mob. Comp. and Net. (MOBICOM)*, pp. 148–159, Sept. 2002.

[22] Y. Xu, J. Heidemann, and D. Estrin, “Geography–informed energy conservation for ad hoc routing,” *Proc. 7th ACM Int’l. Conf. Mob. Comp. and Net. (MOBICOM)*, pp. 70–84, 2001.

[23] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, “Span: an energy–efficient coordination algorithm for topology maintenance in ad hoc wireless networks,” *Proc. 7th Int’l. Conf. Mob. Comp. Net. (MOBICOM)*, pp. 85–96, July 2001.

[24] I. Stojmenovic and X. Lin, “GEDIR: loop–free location based routing in wireless networks,” *Int’l. Conf. Parallel and Distrib. Comp. and Net.* , Nov. 1999.

[25] M. Zorzi and R. R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance,” *IEEE Trans. on Mobile Computing*, vol. 2, no. 4, pp. 349–365, 2003.

[26] ——, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance,” *IEEE Trans. on Mobile Computing*, vol. 2, no. 4, pp. 337–348, 2003.

[27] M. Zorzi, “A new contention–based MAC protocol for geographic forwarding in ad hoc and sensor networks,” *Proc. IEEE Int’l. Conf. Commun. (ICC)*, vol. 6, pp. 3481–3485, June 2004.

[28] Y. Yu, D. Estrin, and R. Govindan, “Geographical and energy–aware routing: a recursive data dissemination protocol for wireless sensor networks,” UCLA Comp. Sci. Dept., Tech. Rep. 010023, May 2001.

[29] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed diffusion for wireless sensor networking,” *ACM/IEEE Transactions on Networking*, vol. 11, no. 1, pp. 2–16, February 2003.

[30] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” *Proc. 5th ACM/IEEE Int'l. Conf. Mob. Comp. and Net. (MOBICOM)*, pp. 174–185, Aug. 1999.

[31] J. Kulik, W. R. Heinzelman, and H. Balakrishnan, “Negotiation based protocols for disseminating information in wireless sensor networks,” *Wireless Networks*, vol. 8, pp. 169–185, 2002.

[32] D. Braginsky and D. Estrin, “Rumor Routing Algorithm for Sensor Networks,” *Proc. of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, 2002.

[33] F. Ye, A. Chen, L. Songwu, and Z. Lixia, “A scalable solution to minimum cost forwarding in large sensor networks,” *Proceedings of the 10th International Conference on Computer Communications and Networks*, vol. 8, pp. 304–309, Oct. 2001.

[34] M. Chu, H. Haussecker, and F. Zhao, “Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks,” *Int'l. J. High Performance Computing Applications*, vol. 8, Fall 2002.

[35] Y. Yao and J. Gehrke, “The COUGAR approach internetwrok query processing in sensor networks,” *SIGMOD Record*, Sept. 2002.

[36] S. Servetto and G. Barrenechea, “Constrained random walks on random graphs: routing algorithms for large scale wireless sensor networks,” *Proc. 1th ACM Int'l. Wksp. Wireless Sensor Networks and Apps.*, 2002.

[37] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy–efficient communication protocol for wireless microsensor networks,” *Proc. 33rd Hawaii Int'l. Conf. Sys. Sci.*, pp. 3005–3014, Jan. 2000.

[38] S. Lindsey and C. Raghavendra, “PEGASIS: power–efficient gatering in sensor information systems,” *IEEE Aerospace Conference Proceedings*, vol. 3, pp. 1125–1130, July 2002.

[39] A. Manjeshwar and D. Agrawal, “TEEN: a routing protocol for enhanced efficiency in wireless sensor networks,” *Proc. of the 15th International Parallel an Distributed Processing Symposium*, pp. 2009–2015, Apr. 2001.

[40] A. Manjeshwar and D. P. Agrawal, “APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks,” *Proc. 16th Int'l. Parallel and Distrib. Proc. Symp.*, pp. 195–202, 2002.

[41] V. Rodoplu and T. H. Meng, “Minimum energy mobile wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1344, August 1999.

[42] L. Li and J. Y. Halpern, “Minimum energy mobile wireless networks revisited,” *Proc. IEEE Int'l. Conf. Commun. (ICC)*, vol. 1, pp. 278–283, June 2001.

[43] L. Subramanian and R. H. Katz, “An architecture for building self–configurable systems,” *Proc. 1st ACM/IEEE Wksp. Mob. Comp. and Ad Hoc Net. (MOBIHOC)*, pp. 63–73, Aug. 2000.

[44] J. N. Al-Karaki, R. Ul-Mustafa, and A. E. Kamal, “Data aggregation in wireless sensor networks – exact and approximate algorithms,” *IEEE Wksp. on High Perf. Switching and Routing*, pp. 241–245, Apr. 2004.

[45] A. Savvides, C. Han, and M. B. Strivastava, “Dynamic fine–grained localization in ad hoc networks of sensors,” *Proc. 7th ACM Int'l. Conf. Mob. Comp. and Net.*, pp. 166–179, July 2001.

[46] J. N. Al-Karaki and A. E. Kamal, “On the correlated data gathering problem in wireless sensor networks,” *Proc. 9th IEEE Symp. Comp. and Commun.*, vol. 1, pp. 226–231, July 2004.

[47] Q. Li, J. Aslam, and D. Rus, “Hierarchical power–aware routing in sensor networks,” *Proc. DIMACS Wksp. Pervasive Networking*, May 2001.

[48] B. Krishnamachari, D. Estrin, and S. Wicker, “The impact of data aggregation in wireless sensor networks,” *International Workshop on Distributed Event-Based Systems*, 2002.

[49] M. Lotfinezhad and B. Liang, “Effect of partially correlated data on clustering in wireless sensor networks,” in *IEEE SECON 2004*, Santa Clara, CA, US, Oct. 2004.

[50] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, “TAG: a tiny aggregation service for ad-hoc sensor networks,” in *OSDI 2003*, 2003.

[51] A. Sharaf, J. Beaver, A. Labrinidis, and K. Chrysanthis, “Balancing energy efficiency and quality of aggregate data in sensor networks,” *The VLDB Journal, The International Journal on Very Large Data Bases*, vol. 13, no. 4, Dec. 2004.

[52] S. Nath, P. B. Gibbons, S. Seshan, and Z. R. Anderson, “Aggregation: Synopsis diffusion for robust aggregation in sensor networks,” in *ACM SenSys 2004*, Baltimore, Maryland, US, Nov. 2004.

[53] A. Manjhi, S. Nath, and P. B. Gibbons, “Tributaries and deltas: efficient and robust aggregation in sensor network streams,” in *ACM SIGMOD 2005*, Baltimore, Maryland, US, June 2005.

[54] A. E. D. J. Baker and J. A. Flynn., “The design and simulation of a mobile radio network with distributed control.” *IEEE Journal on Selected Areas in Communications SAC-2*, pp. 226–237, January 1984.

[55] J. E. W. A. Ephremides and D. J. Baker, “A design concept for reliable mobile radio networks with frequency hopping signaling.” in *Proceedings of the IEEE 75*, vol. 1, January 1987, pp. 56–73.

[56] E. M. Belding-Royer, “Multi-level hierarchies for scalable ad hoc routing.” *ACM/Kluwer Wireless Networks 9*, vol. 5, pp. 461–478, September 2003.

[57] M. Gerla and C. R. Lin., “Multimedia transport in multihop dynamic packet radio networks.” in *Proceedings of International Conference on Network Protocols*, Tokyo, Japan, 7–10 November 1995, pp. 209–216.

[58] M. Gerla and J. T.-C. Tsai, *Multicluster, mobile, multimedia radio network.*, 1995.

[59] C. R. Lin and M. Gerla, “Adaptive clustering for mobile wireless networks.” *Journal on Selected Areas in Communications 15*, pp. 1265–1275, September 1997.

[60] I. Chlamtac and A. Faragó, “A new approach to the design and analysis of peer-to-peer mobile networks.” *Wireless Networks 5*, pp. 149–156, May 1999.

[61] C. Bettstetter and R. Krauser, “Scenario-based stability analysis of the distributed mobility-adaptive clustering (DMAC) algorithm.” in *Proceedings of the* 2*nd ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2001*, Long Beach, CA, October 4–5 2001, pp. 232–241.

[62] S. K. D. M. Chatterjee and D. Turgut, “An on-demand weighted clustering algorithm (WCA) for ad hoc networks.” in *Proceedings of IEEE Globecom 2000*, vol. 3, San Francisco, CA, November 27–December 1 2000, pp. 1697–1701.

[63] G. Chen, F. Nocetti, J. Gonzalez, and I. Stojmenovic, “Connectivity-based k-hop clustering in wireless networks,” in *HICSS ’02: Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS’02)-Volume 7*. Washington, DC, USA: IEEE Computer Society, 2002, p. 188.3.

[64] S. Basagni, “Distributed clustering for ad hoc networks.” in *Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN’99)*, Perth/Fremantle, Australia, June 23–25 1999, pp. 310–315.

[65] ——, “Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks.” in *Proceedings of the IEEE* 50*th International Vehicular Technology Conference, VTC 1999-Fall*, vol. 2, Amsterdam, The Netherlands, September 19–22 1999, pp. 889–893.

[66] C. Bettstetter, “The cluster density of a distributed clutering algorithm in ad hoc networks.” in *Proceeding of the IEEE International Conference on Communications, ICC 2004*, vol. 7, Paris, France, June 20–24 2004, pp. 4336–4340.

[67] C. Bettstetter and B. Friedrich, “Time and message complexity of the generalized distributed mobility adaptive clustering (GDMAC) algorithm in wireless multihop networks.” in *Proceeding of the* 57*th IEEE Semiannual Vehicular Technology Conference, VTC 2003-Spring*, vol. 1, Jeju, Korea, April 22–25 2003, pp. 176–180.

[68] K. M. A. P.-J. Wan and O. Frieder, “Distributed construction of connected dominating sets in wireless ad hoc networks.” *ACM/Kluwer Mobile Networks and Applications, MONET 9*, pp. 141–149, April 2004.

[69] B. R. U. C. Kozat, G. Kondylis and M. K. Marina, “Virtual dynamic backbone for mobile ad hoc networks.” in *Proceedings of the IEEE International Conference on Communications (ICC 2001)*, vol. 1, Helsinki, Finland, June 11–14 2001, pp. 250–255.

[70] P. H. T. Nieberg and J. Hurink, “Size-controlled dynamic clutering in mobile wireless sensor networks.” in *Proceedings of the 2004 Communication Networks and Distributed Systems Modeling and Simulation Conference, (CSDN’04)*, San Diego, CA, January 18–21 2004.

[71] D. T. S. Basagni and S. K. Das, “Mobility-adaptive protocols for managing large ad hoc networks.” in *Proceedings of the IEEE International Conference on Communications, ICC 2001*, vol. 5, Helsinki, Finland, June 11–14 2001, pp. 1539–1543.

[72] A. B. McDonald and T. Znati, “A mobility-based framework for adaptive clustering in wireless ad hoc networks.” *IEEE Journal on Selected Areas in Communications, Special Issue on Wireless Ad Hoc Networks 17*, pp. 1466–1487, August 1999.

[73] L. Wang and S. Olariu, “A unifying look at clustering in mobile ad hoc networks.” *Wireless Communications and Mobile Computing 4*, pp. 623–637, September 2004.

[74] B. Das and V. Bharghavan, “Routing in ad-hoc networks using minimum connected dominating sets.” *IEEE International Conference on Communications. ICC’97.*, pp. 376–380, June 8–12 1997.

[75] R. S. P. Sinha and V. Bharghavan, “Enhancing ad hoc routing with dynamic virtual infrastructures.” in *Proceedings of IEEE Infocom 2001*, vol. 3, Anchorage, AK, April 22-26 2001, pp. 1763–1762.

[76] B. D. R. Sivakumar and V. Bharghavan, “The clade vertebrata: Spines and routing in ad hoc networks.” in *Prooceedings of the IEEE Symposium on Computer Communications(ISCC’98)*, Athens, Greece, June 30–July 2 1998.

[77] ——, “Spine-based routing in ad hoc networks.” November 1998, pp. 237–248.

[78] P. S. R. Sivakumar and V. Bharghavan, “CEDAR: A core-extraction distributed ad hoc routing algorithm.” *IEEE Journal on Selected Areas in Communications 17*, pp. 1454–1465, August 1999.

[79] B. Liang and Z. Haas, “Virtual backbone generation and maintenance in ad hoc network mobility management.” in *Proceedings of the* $19th$ *IEEE Infocom*, vol. 3, Tel Aviv, Israel, March 26–30 2000, pp. 1293–1302.

[80] J. Wu and H. Li, “On calculating connected dominating sets for efficient routing in ad hoc wireless networks,” *Telecommunication Systems, Special Issue on Mobile Computing and Wireless Networks*, vol. 18, no. 1/3, pp. 13–26, September 2001.

[81] M. G. J. Wu, F. Dai and I. Stojmenovic, “On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks.” *Journal of Communications and Networks 4*, pp. 1–12, March 2002.

[82] F. Dai and J. Wu, “An extended localized algorithms for connected dominating set formation in ad hoc wireless networks.” *IEEE Transactions on Parallel and Distributed Systems 15*, October 2004.

[83] M. S. I. Stojmenovic and J. Zunic, “Dominating sets and neighbors elimination-based broadcasting algorithms in wireless networks.” *IEEE Transactions on Parallel and Distributed Systems 13*, pp. 14–25, January 2002.

[84] M. C. P. Krishna, N. H. Vaidya and D. K. Pradhan, “A cluster-based approach for routing in dynamic networks.” *ACM SIGCOMM Computer Communication Review 27*, pp. 49–64, 2 April 1997.

[85] T. H. P. V. A. D. Amis, R. Prakash and D. T. Huynh, “Max-min d-cluster formation in wireless ad hoc networks.” in *Proceedings of IEEE Infocom 2000*, vol. 1, Tel Aviv, Israel, March 26–30 2000, pp. 32–41.

[86] X. H. K. Xu and M. Gerla, “An ad hoc network with mobile backbones.” in *Proceedings of the IEEE International Conference on Communications (ICC 2002)*, vol. 5, New York, NY, April 28–May 2 2002, pp. 3138–3143.

[87] L. Bao and J. J. Garcia-Luna-Aceves, “Topology management in ad hoc networks.” in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2003*, Annapolis, MD, June 1–3 2003, pp. 129–140.

[88] A. C. S. Basagni and C. Petrioli, “Sensor-DMAC: Dynamic topology control for wireless sensor network.” in *Proceedings of the 60th IEEE Vehicular Technology Conference, VTC 2004 Fall*, Los Angles, CA, September 26–29 2004.

[89] M. E. S. Basagni and R. Ghosh, “ViBES: Virtual backbone for energy saving in wireless sensor networks.” in *Proceedings of the IEEE Military Communication Conference, MILCOM 2004*, Monterey, CA, October 31–November 3 2004.

[90] C. P. A. Marcucci, M. Nati and A. Vitaletti, “Directed diffusion light: Low overhead data dissemination in wireless sensor networks.” in *Proceedings of IEEE VTC 2005 Spring*, Stockholm, Sweden, May 29-June 1 2005.

[91] A. C. W. R. Heinzelman and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks.” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, HICSS 2000*, Maui, HA, January 4–7 2000, pp. 3005–3014.

[92] S. Bandyopadhyay and E. J. Coyle, “An energy efficient hierarchical clustering algorithm for wireless sensor networks.” in *Proceedings of the 22ns IEEE Infocom 2003*, vol. 3, San Francisco, March 31–April 3 2003, pp. 1713–1723.

[93] ——, “Minimizing communication costs in hierarchically clustered networks of wireless sensors.” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2003)*, vol. 2, New Orleans, LA, March 16–20 2003, pp. 1274–1279.

[94] F. S. J. Tillett, R. Rao and T. M. Rao, “Cluster-head identification in ad hoc sensor networks using particle swarm optimization,” in *Proceedings of the IEEE International Conference on Personal Wireless Communications, ICPWC 2002*, New Delhi, India, December 15–17 2002, pp. 201–205.

[95] H. Chan and A. Perrig, “Ace: An emergent algorithm for highly uniform cluster formation.” in *Proceedings of the 1st IEEE European Workshop on Wireless Sensor Networks, EWSN 2004*, A. W. H. Karl and A. Wolisz, Eds., Berlin, Germany, January 19–21 2004, pp. 154–171.

[96] S. Banerjee and S. Khuller, “A clustering scheme for hierarchical control in multi-hop wireless networks.” in *Proceedings of the 20th IEEE Infocom 2001*, vol. 2, Anchorage, AK, April 22–26 2001, pp. 1028–1037.

[97] G. Gupta and M. Younis, “Fault-tolerant clustering of wireless sensor networks.” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2003)*, vol. 3, New Orleans, LA, March 16–20 2003, pp. 1579–1584.

[98] ——, “Performance evaluation of load-balanced clustering of wireless sensor networks.” in *Proceedings of the 10th International Conference on Telecommunications (ICT 2003)*, vol. 2, Papeete, French Polynesia, February 23–March 1 2003, pp. 1577–1583.

[99] P. K. G. Jolly, M. C. Kuscu and M. Younis, “A low-energy key management protocol for wireless sensor networks.” in *Proceedings of the 8th IEEE International Symposium on Computers and Communications (ISCC 2003)*, Kemer-Antalya, Turkey, June 30–July 3 2003, pp. 335–340.

[100] G. Gupta and M. Younis, “Load-balanced clustering of wireless sensor networks.” in *Proceedings of the IEEE International Conference on Communications (ICC 2003)*, vol. 3, Anchorage, AK, May 11–15 2003, pp. 1848–1852.

[101] K. A. M. Younis and A. Kunjithapatham, “Optimization of task allocation in cluster-based sensor networks.” in *Proceedings of the 8th IEEE International Symposium on Computers and Communications (ISCC 2003)*, Kemer-Antalya, Turkey, June 30–July 3 2003, pp. 329–334.

[102] M. Y. M. Younis and K. Arisha, “Energy-aware routing in cluster-based sensor networks.” in *Proceedings of the 10th IEEE International Symposium on Modeling, Analysis and Simulations of Computer and Telecommunication Systems (MASCOTS 2002)*, Fort Worth, TX, October 11–16 2002, pp. 129–136.

[103] K. Akkaya and M. Younis, “An energy-aware qos routing protocol for wireless sensor networks.” in *Proceedings of the 23rd International Conference on Distributed Computing Systems Workshop, (ICDCSW 2003)*, Providence, RI, May 19–22 2003, pp. 710–715.

[104] M. Gerla and K.Xu, “Multimedia streaming in large-scale sensor networks with mobile swarms,” *SIGMOD Rec.*, vol. 32, no. 4, pp. 72–76, 2003.

[105] J. C. H. W.-P. Chen and L. Sha, “Dynamic clustering for acoustic target tracking in wireless sensor networks.” *IEEE Transactions on Mobile Computing 3*, vol. 3, pp. 258–271, July 2004.

[106] V. Mhatre and C. Rosenberg, "Homogeneous vs. heterogeneous clustered sensor networks: A comparative study." in *Proceedings of the 2004 IEEE International Conference on Communications, ICC 2004*, vol. 6, Paris, France, June 20–24 2004, pp. 3646–3651.

[107] ——, "Design guidelines for wireless sensor networks: communication, clustering and aggregation." *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.

[108] P. Santi and J. Simon, "Silence is golden with high probability: Maintaining a connected backbone in wireless sensor networks." in *Proceedings of the* 1*st IEEE European Workshop on Wireless Sensor Networks, EWSN 2004*, A. W. H. Karl and A. Wolisz, Eds., Berlin, Germany, January 19–21 2004, pp. 106–212.

[109] J. H. Y. Hu and D. Estrin, "Geography-informed energy conservation for ad hoc routing." in *Proceedings of the* 7*th ACM Annual International Conference on Mobile Computing and Networking*, Rome, Italy, July 16–21 2001, pp. 70–84.

[110] A. T. J. Blum, M. Ding and X. Cheng, *Connected dominating sets in sensor networks and MANETs.*, D.-Z. Du and P. M. Pardalos, Eds. Kluwer Academic Publisher, 2004, vol. 1.

[111] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," University of Southern California, Tech. Rep. Technical Report 00-729, April 2000. [Online]. Available: URL: http://citeseer.nj.nec.com/bulusu00gpsless.html

[112] D. Niculescu and B. Nath, "Dv based positioing in ad hoc networks," *Telecommunication Systems*, vol. 22, no. 1–4, pp. 267–280, 2003. [Online]. Available: URL: http://paul.rutgers.edu/ dnicules/research/aps/aps-jrn.pdf

[113] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *Proc. of 9th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2003)*. ACM, Sept 2003. [Online]. Available: URL: http://citeseer.nj.nec.com/he03rangefree.html

[114] L. Hu and D. Evans, "Localization for mobile sensor networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*. ACM Press, 2004, pp. 45–57.

[115] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.

[116] L. Doherty, L. Ghaoui, and K. Pister, "Convex position estimation in wireless sensor networks," in *Proc. of IEEE INFOCOM '01*, vol. 3. IEEE, April 2001, pp. 1655–1663.

[117] S. Simic and S. Sastry, "Distributed localization in wireless ad hoc networks," UC Berkeley, Tech. Rep. UCB/ERL M02/26, 2002.

[118] G. Stupp and M. Sidi, "The expected uncertainty of range free localization protocols in sensor networks," in *Algorithmic Aspects of Wireless Sensor Networks: First International Workshop, ALGOSENSORS 2004, Turku, Finland, July 16, 2004. Proceedings*, ser. Lecture Notes in Computer Science, vol. 3121. Springer, 2004.

[119] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the 6th annual international conference on Mobile computing and networking*. ACM Press, 2000, pp. 32–43.

[120] N. B. Priyantha, A. K. Miu, H. Balakrishnan, and S. Teller, "The cricket compass for context-aware mobile applications," in *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM Press, 2001, pp. 1–14.

[121] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization wit noisy range measurements," in *SenSys '04: In Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems*, Baltimore, MD, November 2004.

[122] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM (2)*, 2000, pp. 775–784. [Online]. Available: URL: http://citeseer.ist.psu.edu/bahl00radar.html

[123] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine–grained localization in ad-hoc networks of sensors," in *Proceedings of the seventh annual international conference on Mobile computing and networking*. ACM Press, July 2001, pp. 166–179. [Online]. Available: URL: http://doi.acm.org/10.1145/381677.381693

[124] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of the first ACM international workshop on Wireless sensor networks and applications*. ACM Press, September 28 2002, pp. 112–121. [Online]. Available: URL: http://doi.acm.org/10.1145/570738.570755

[125] D. Niculescu and B. Nath, "Ad hoc positioning system (aps)," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM 2001)*, vol. 1. GLOBECOM, November 2001, pp. 2926–2931. [Online]. Available: URL: http://citeseer.nj.nec.com/519054.html

[126] ——, "Ad hoc positioning system (aps) using aoa," in *in Proc. of 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, vol. 3. San Francisco, CA: IEEE, March 2003, pp. 1734–1743. [Online]. Available: URL: http://citeseer.nj.nec.com/niculescu03ad.html

[127] ——, "Error characteristics of ad hoc positioning systems," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2004*, Roppongi Hills, Tokyo, Japan, 2004, pp. 20–30.

[128] C. Savarese, J. M. Rabaey, and K. Langendoen, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*. USENIX Association, 2002, pp. 317–327. [Online]. Available: URL: http://rama.pds.twi.tudelft.nl/ koen/papers/robust-positioning.pdf

[129] C. Fretzagias and M. Papadopouli, "Cooperative location sensing for wireless networks," in *Second IEEE International conference on Pervasive Computing and Communications*, Orlando, Florida, March 2004. [Online]. Available: URL: http://www.cs.unc.edu/ maria/percom04.pdf

[130] K. K. Chintalapudi, A. Dhariwal, R. Govindan, and G. Sukhatme, “Ad-hoc localization using ranging and sectoring,” in *IEEE INFOCOM ’04: The Conference on Computer Communications*, vol. 1, March 2004, pp. 2662–2672.

[131] E. D. Nissanka B. Priyantha, Hari Balakrishnan and S. Teller, “Anchor-free distributed localization in sensor networks,” in *in Proc. of the 1st International Conference on Embedded Networked Sensor Systems (SenSys 2003)*, November 5-7 2003, pp. 340–341.

[132] T. Fruchterman and E. Reingold, “Graph drwaing by force-directed placement,” *Software - Practice and Experience (SPE)*, vol. 21, no. 11, pp. 1129–1164, November 1991.

[133] A. Howard, M. Mataric, and G. Sukhatme, “Relaxation on a mesh: A formalism for generalized localization,” in *In Proceedings of IEEE/RSJ Intl. Conference On Intelligent Robots and Systems (IROS)*, Wailea, Hawaii, October 2001.

[134] F. Zhao, J. Shin, and J. Reich, “Information-driven dynamic sensor collaboration,” *IEEE Signal Processing Mag.*, vol. 19, pp. 61–72, Mar. 2002.

[135] F. Zhao, J. Liu, J. Liu, L. Guibas, and J. Reich, “Collaborative signal and information processing: An information-directed approach,” *Proc. of the IEEE*, vol. 91, no. 8, pp. 1199–1209, Aug. 2003.

[136] S. N. I. Chatzigiannakis and P. Spirakis., “An efficient communication strategy for ad-hoc mobile networks.” in *Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing, PODC 2001, Newport, RI*, August 26–29 2001, pp. 320–332.

[137] I. Chatzigiannakis and S. Nikoletseas., “An adaptive compulsory protocol for basic communications in highly changing ad-hoc mobile networks.” in *Proceedings of the International Parallel and Distributed Processing Symposium, IPDPS 2002, Fort Lauderdale, FL*, April 15–19 2002, pp. 193–202.

[138] Y. W. M. M. L.-S. P. P. Juang, H. Oki and D. Rubenstein., “Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet.” in *Proceedings of the* 10*th International Conference on Architecural Support for Programming Languages and Operating Systems, ASPLOS-X*, October 5–9 2002, pp. 96–107.

[139] Q. Li and D. Rus., “Sending messages to mobile users in disconnected ad-hoc wireless networks.” in *Proceedings of* 6*th ACM Annual International Conference on Mobile Computing and Networking, MobiCom 2000*, Boston, MA, August 6–11 2000, pp. 44–55.

[140] M. Grossglauser and D. N. C. Tse., “Mobility increases the capacity of ad-hoc wireless networks.” *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, pp. 477–486, August 2002.

[141] M. A. W. Zhao and E. W. Zegura., “A message ferrying approach for data delivery in sparse mobile ad hoc networks.” in *Proceedings of the* 5*th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2004*, Roppongi Hills, Tokyo, Japan, May 24–26 2004, pp. 187–198.

[142] W. B. G. B. S. Jain, R. C. Shah and S. Roy., "Exploiting mobility of energy-efficient data collection in sensor networks." in *Proceedings of the IEEE Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, WiOpt 2004*, Cambridge, UK, March 24–26 2004.

[143] S. J. R. C. Shah, S. Roy and W. Brunette., "Data MULEs: Modeling a three-tier architecture for sparse sensor networks." in *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, SNPA 2003*, Anchorage, AK, May 11 2003, pp. 30–41.

[144] T. F. A. H. S. Kim and W. H. Kwon., "Minimum energy asynchronous dissemination to mobile sinks in wireless sensor networks," in *Proceedings of the First International Conference on Embedded Networked Sensor Systems, SenSys 2003*, Los Angeles, CA, November, 5–7 2003, pp. 193–204.

[145] D. E. J. H. C. Intanagonwiwat, R. Govindan and F. Silva., "Directed diffusion for wireless sensor networking." *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 2–16, February 2003.

[146] J. C. S. L. F. Ye, H. Luo and L. Zhang., "A two-tier data dissemination model for large scale wireless sensor networks." in *Proceedings of the 8th ACM Annual International Conference on Mobile Computing and Networking, MobiCom 2002*, Atlanta, GA, September 23–28 2002, pp. 148–159.

[147] J. G. Jetcheva and D. B. Johnson., "Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks." in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing, MobiHoc 2001*, Long Beach, CA, October 4–5 2001, pp. 33–44.

[148] Q. Z. L. Tong and S. Adireddy., "Sensor networks with mobile agents." in *Proceedings of the IEEE Military Communication Conference, MILCOM 2003*, vol. 1, Boston, MA, October 13–16 2003, pp. 705–710.

[149] S. A. P. Venkitasubramaniam and L. Tong., "Sensor networks with mobile agents: Optimal random access and coding." *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1058–1068, August 2004.

[150] D. K. R. M. V. P. Mhatre, C. Rosenberg and N. Shroff., "A minimum cost heterogeneous sensor network with a lifetime contraint." *IEEE transactions on Mobile Computing*, vol. 4, no. 1, pp. 4–15, January/February 2005.

[151] A. S. A. Chakrabarti and B. Aazhang., "Using predictable observer mobility for power efficient design of sensor networks." in *Proceedings of the Second International Workshop on Information Processing in Sensor Networks, IPSN 2003*, F. Zhao and L. Guibas, Eds., Palo Alto, CA, April 22–23 2003, pp. 129–145.

[152] R. U. P. Baruah and B. Krishnamachari., "Learning-enforced time domain routing to mobile sinks in wireless sensor fields." in *Proceeding of the 29th Annual IEEE International Conference on Local Computer Networks, LCN 2004*, Tampa, FL, November 16–18 2004, pp. 525–532.

[153] D. D. J. M. B. S. D. E. A. Kansal, A. A. Somasundara, “Intelligent fluid infrastructure for embedded networks.” in *Proceedings of the 2nd ACM/SIGMOBILE International Conference on Mobile Systems, Applications, and Services, MobySys 2004*, Boston, MA, June 6–9 2004, pp. 111–124.

[154] A. S. M. B. E. R. D. K. Goldenberg, J. Lin and Y. R. Yang., “Towards mobility as a network control primitive.” in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2004*, Roppongi Hills, Tokyo, Japan, May 24–26 2004, pp. 163–174.

[155] R. P. S. R. Gandham, M. Dawande and S. Venkatesan., “Energy efficient schemes for wireless sensor networks with multiple mobile base stations.” in *Proceedings of IEEE Globecom 2003*, vol. 1, San Francisco, CA, December 1–5 2003, pp. 377–381.

[156] E. M. Z. M. Wang, S. Basagni and C. Petrioli., “Exploiting sink mobility for maximizing sensor networks lifetime.” in *Proceedings of the 38th Hawaii International Conference on System Sciences, Big Island, Hawaii*, January 3–6 2005.

[157] R. Peterson and D. Rus, “Interacting with sensor networks,” in *Proc. of the 2004 IEEE Intl. Conference on Robotics & Automation*, April 2004, pp. 180–186.

[158] M. B. McMickell, B. Goodwine, and L. A. Montestruque, “Micabot: A robotic platform for large-scale distributed robotics,” in *Proc. of the 2003 IEEE, Intl. Conference on Robotics & Automation*, September 2003, pp. 1600–1605.

[159] S. Bergbreiter and K. S. J. Pister, “Costbots: An off-theshelf platform for distribuited robotics,” in *Proc. of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, October 2003, pp. 1632–1637.

[160] D. K., M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. Sukhatme, “Robomote: Enabling mobility in sensor networks,” Dept. of Computer Science, University of Southern California, Los Angeles, California, Tech. Rep. CRES-04-006, 2004.

[161] L. Navarro-Serment, R. Grabowski, C. J. Paredis, and P. K. Khosla, “Millibots,” *IEEE Robotics & Automation Magazine*, pp. 31–40, December 2002.

[162] P. Corke, S. Hrabar, R. Peterson, D. Rus, S. Saripalli, and G. Sukhatme, “Autonomous deployment and repair of a sensor network using an unmanned aerial vehicle,” in *Proc. of the 2004 IEEE Intl. Conference on Robotics & Automation*, April 2004, pp. 3602–3608.

[163] P. Corke, R. Peterson, and D. Rus, “Networked robots: Flying robot navigation using a sensor net,” in *(International Symposium of Robotic Research (ISRR)*, 2003.

[164] J. McLurkin and J. Smith, “Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots,” in *Proc. of the 7th International Symposium on Distributed Autonomous Robotic Systems*, Toulouse, France, June 2004, pp. 381–390.

[165] J. Cortes, S. Martinez, S. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*. Arlington, VA: IEEE, 2002, pp. 1327–1332.

[166] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *IEEE Wireless Communications Magazine*, pp. 6–28, December 2004.

[167] D. Gale, *The Theory of Linear Economic Models*. Nueva York, EEUU: McGraw-Hill Book Company, Inc., 1960.

[168] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2(1), pp. 83–97, 1955.

[169] D. P. Bertsekas, "The auction algorithm for assignment and other network flow problems: A tutorial," *Interfaces*, vol. 20(4), pp. 133–149, 1990.

[170] P. Brucker, *Scheduling Algorithms*. Berlin: Springer-Verlag, 1998.

[171] J. L. Bruno, E. G. Coffmann, and R. Sethi, "Scheduling independent tasks to reduce mean finishing time," *Communications of the ACM*, vol. 17(7), pp. 382–387, 1974.

[172] M. B. Dias and A. Stentz, "A market approach to multirobot coordination," The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Tech. Rep. CMU-RI-TR-01-26, 2001.

[173] K. L. Moore, Y. Chen, and Z. Song, "Diffusion-based path planning in mobile actuator-sensor networks (mas-net): Some preliminary results," in *Proceedings of SPIE*, April 2004.

[174] R. S. Aylett and D. P. Barnes, "A multi-robot architecture for planetary rovers," in *Proc. 5th ESA Workshop on Space Robotics, ASTRA'98*, Noordwijk: European Space Agency, 1998.

[175] M. Mataric, M. Nilsson, and K. T. Simsarian, "Cooperative multi-robot box-pushing," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3. Los Alamitos, CA: IEEE Computer Society Press, 1995, pp. 556–561.

[176] D. Rus, B. Donald, and J. Jennings, "Moving furniture with teams of autonomous robots," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1. Pittsburgh, PA: IEEE, 1995, pp. 235–242.

[177] L. E. Parker, "Alliance: An architecture for fault-tolerant multi-robot cooperation," *IEEE Transactions on Robotics and Automation*, vol. 14(2), pp. 220–240, 1998.

[178] Y. U. Cao, A. S. Fukunaga, and A. Kahng, "Cooperative mobile robotics: Antecedents and directions," *Autonomous Robots*, vol. 4(1), pp. 7–27, 1997.

[179] G. Dudek, M. Jenkin, and E. Milios, *Robot Teams: From Diversity to Polymorphism*. Natick, MA, EEUU: A.K. Peters, 2002, ch. A taxonomy of multirobot systems, pp. 3–22.

[180] S. Yuta and S. Premvuti, “Coordinating autonomous and centralized decision making to achieve cooperative behaviors between multiple mobile robots,” in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Raleigh, North Carolina, 1992, pp. 1566–1574.

[181] T. Balch, “Taxonomies of multirobot task and reward,” Carnegie Mellon University, Pittsburgh, Pennsylvania, Tech. Rep., 1998.

[182] K. Ali, “Multiagent telerobotics: matching systems to tasks,” Ph.D. dissertation, Georgia Institute of Technology, 1999.

[183] E. Todt, G. Raush, and R. Suarez, “Analysis and classification of multiple robot coordination methods,” in *Proceedings IEEE International Conference on Robotics and Automation*, San Francisco, California, 2000.

[184] G. Beni, “The concept of cellular robotic system,” in *Proceedings of the IEEE International Symposium on Intelligent Control*, 1988, pp. 57–62.

[185] H. Asama, A. Matsumoto, and Y. Ishida, “Design of an autonomous and distributed robot system: Actress,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1989, pp. 283–290.

[186] T. Fukuda, Y. Kawauchi, and H. Asama, “Analysis and evaluation of cellular robotics (cebot) as a distributed intelligent system by communication amount,” in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1990, pp. 827–834.

[187] P. Caloud, W. Choi, J. C. Latombe, C. Le Pape, and M. Yin, “Indoor automation with many mobile robots,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1990, pp. 67–72.

[188] L. E. Parker, “Alliance: an architecture for fault tolerant, cooperative control of heterogeneous mobile robots,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1994, pp. 776–783.

[189] R. Alami, S. Fleury, M. Herrb, F. Ingrand, and S. Qutub, “Operating a large fleet of mobile robots using the plan-merging paradigm,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico, April 1997, pp. 2312–2317.

[190] R. Alami, S. Fleury, M. Herrb, F. Ingrand, and F. Robert, “Multi robot cooperation in the martha project,” *IEEE Robotics and Automation Magazine*, vol. 5(1), pp. 36–47, March 1998.

[191] J. C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer Academic Publishers, 1991.

[192] M. Batalin, G. S. Sukhatme, and M. Hatting, “Mobile robot navigation using a sensor network,” in *Proc. of the 2004 IEEE Intl. Conference on Robotics & Automation*, April 2004, pp. 636–641.

[193] Q. Li, M. DeRosa, and D. Rus, “Distributed algorithms for guiding navigation across a sensor network,” 2003. [Online]. Available: URL: citeseer.ist.psu.edu/li03distributed.html

[194] U. Berkeley, “Path Project,” URL: http://www.path.berkeley.edu/, Apr. 2005.

[195] D. Reichardt, M. Miglietta, L. Moretti, P. Morsink, and W. Schulz, “CarTALK 2000 - Safe and Comfortable Driving Based Upon Inter-Vehicle-Communication,” in *In IEEE Intelligent Vehicle Symposium*, Versailles, France, June 2002.

[196] N. et al, “FleetNet Project,” URL: http://www.ccrle.nec.de/Projects/fleetnet.htm, Apr. 2005.

[197] L. Xu, R. Tonjes, T. Paila, W. Hansmann, M. Frank, and M. Albrecht, “DRiVE-ing to the Internet: Dynamic Radio for IP Services in Vehicular Environments,” in *The 25th Annual IEEE Conference on Local Computer Networks (LCN'00)*, Tampa, Florida, USA, Nov. 2000, http://www.ist-drive.org/papers/Lcn2000/LCN2000.pdf.

[198] S. Tsugawa, K. Tokuda, S. Kato, T. Matsui, and H. Fujii, “An overview on demo 2000 cooperative driving,” in *IEEE Intelligent Vehicle Symposium (IV01)*, Tokyo, Japan, May 2001, p. 327332.

[199] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, “Adaptive Broadcast for Travel and Traffic Information Distribution Based on Inter-Vehicle Communication,” in *IEEE Intelligent Vehicles Symposium (IV 2003)*, Columbus, Ohio, USA, June 2003, http://www.et2.tu-harburg.de/Mitarbeiter/Wischhof/IV2003 098.pdf.

[200] J. Blum, A. Eskandarian, and L. Hoffman, “Challenges of Intervehicle Ad Hoc Networks,” *IEEE Transaction on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 347–351, Dec. 2004.

[201] D. Cottingham, “Research Directions on Inter-vehicle Communication,” URL: http://www.cl.cam.ac.uk/users/dnc25/references.html, Dec. 2004.

[202] M. Rudack, M. Meincke, K. Jobmann, and M. Lott, “On traffic dynamical aspects intervehicle communication (IVC),” in *57th IEEE Semiannual Vehicular Technology Conference (VTC03 Spring)*, Jeju, South Korea, Apr. 2003, http://portal.acm.org/citation.cfm?id=778434.

[203] Q. Xu, K. Hedrick, R. Sengupta, and J. VanderWerf, “Effects of Vehicle-vehicle / roadside-vehicle Communication on Adaptive Cruise Controlled Highway Systems,” in *IEEE VTC Fall 2002*, Vancouver, Canada, Sept. 2002, http://path.berkeley.edu/dsrc/pub/vtc2002f.pdf.

[204] D. N. Godbole and J. Lygeros, “Longitudinal Control of the Lead Car of a Platoon,” University of California at Berkeley, URL: http://www.path.berkeley.edu/PATH/Publications/PDF/TECHMEMOS/TECHMEMO-93-07.pdf, Tech. Rep. PATH Tech Report 93-7, Nov. 1993.

[205] U. Nations, “A Summary of the Kyoto Protocol,” URL: http://unfccc.int/essential_background/feeling_the_heat/items/2879.php.

[206] P. Vidales and F. Stajano, “The Sentient Car: Context-Aware Automotive Telematics.” in *First EE Workshop on Location Based Services*, London, UK, Sept. 2002, http://www-lce.eng.cam.ac.uk/ pav25/publications/lbs-2002(abstract).pdf.

[207] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant networking,” in *ACM SIGCOMM Technical Conference*, Portland, OR, USA, Aug. 2004, http://www.acm.org/sigs/sigcomm/sigcomm2004/papers/p299-jain111111.pdf.

[208] “Wizzy Project,” URL: http://www.wizzy.org.za/.

[209] “Sami Network Connectivity Project (SNC),” URL: http://www.snc.sapmi.net/.

[210] A. Lindgreny, A. Doria, and O. Scheln, “Probablistic routing in Intermittently Connected Networks,” in *The First International Workshop on Service Assurance with Partial and Intermittent Resources SAPIR 2004. In conjunction with ICT 2004*, Fortaleza, Brazil, Aug. 2004.

[211] J. LeBrun, C. N. Chuah, D. Ghosal, and H. M. Zhang, “Knowledge-Based Opportunistic Forwarding in Vehicular Wireless Ad Hoc Networks,” in *IEEE Vehicular Technology Conference (VTC’05) - Spring*, Stockholm, Sweden, May 2005, http://www.ece.ucdavis.edu/ chuah/paper/2005/vtc05-move.pdf.

[212] K. Fall, “A delay-tolerant network architecture for challenged internets,” in *ACM SIGCOMM Technical Conference*, Karlsruhe, Germany, Aug. 2003.

[213] S. Das, A. Nandan, G. Pau, M. Sanadidi, and M. Gerla, “SPAWN: A Swarming Protocol for Vehicular Ad Hoc Networks,” in *Proc. 1st ACM VANET*, Philadelphia, PA, USA, Oct. 2004.

[214] J. Luo and J.-P. Hubaux, “A Survey of Inter-Vehicle Communication,” EPFL, Tech. Rep. Tech report IC/2004/04, Mar. 2004, http://icwww.epfl.ch/publications/abstract.php?ID=200424.

[215] J. Zhu and S. Roy, “MAC for dedicated short range communications in Intelligent Transport Systems,” *IEEE Communications Magazine*, vol. 41, pp. 60–67, Jan. 2003.

[216] P. Vidales, L. Patanapongpibul, G. Mapp, and A. Hopper, “Experiences with Heterogeneous Wireless Networks Unveiling the Challenges,” in *Second International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs)*, West Yorkshire, UK, July 2004.

[217] P. Vidales, J. Baliosian, J. Serrat, G. Mapp, F. Stajano, and A. Hopper, “A Practical Approach for 4G Systems: Deployment of Overlay Networks,” in *First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities*. IEEE Computer Society Press, Feb. 2005.

[218] ——, “Autonomic Systems for Mobility Support in 4G Networks,” *Journal on Selected Areas in Communications (J-SAC), Special Issue in Autonomic Communications*, Nov. 2005.

[219] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, “ADHOC MAC: A new, flexible and reliable MAC architecture for ad-hoc networks,” in *IEEE Wireless Communications and Networking Conference (WCNC03)*, New Orleans, Louisiana, USA, Mar. 2003, http://www.elet.polimi.it/upload/cesana/papers/WCNC2003.pdf.

[220] K. Tokuda, M. Akiyama, and H. Fujii, "DOLPHIN for Inter-vehicle Communications System," in *IEEE Intelligent Vehicle Symposium (IV00)*, Dearborn, MI, USA, Oct. 2000, pp. 327–332, http://ieeexplore.ieee.org/iel5/7217/19432/00898395.pdf?arnumber=898395.

[221] M. Lott, R. Halfmann, E. Schulz, and M. Radimirsch, "Medium access and radio resource management for ad hoc networks based on UTRA TDD," in *ACM SIGMOBILE Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc01)*, Long Beach, California, USA, Oct. 2001, http://portal.acm.org/.

[222] D. Lee, R. Attias, A. Puri, R. Sengupta, S. Tripakis, and P. Varaiya, "A Wireless Token Ring Protocol For Intelligent Transportation Systems," in *IEEE 4th International Conference on Intelligent Transportation Systems*, Aug. 2001.

[223] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "MAC Protocol Design for Vehicle Safety Communications in Dedicated Short Range Communications Spectrum," in *IEEE ITSC 2004*, Washington, D.C., USA, Oct. 2004, http://path.berkeley.edu/dsrc/pub/ITSC04.pdf.

[224] C. Perkins, *Ad Hoc Networking*. Addison-Wesley, 2001.

[225] B. Karp and H. Kung, "Greedy Perimeter Stateless Routing for Wireless Networks," in *Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, Boston, MA, Aug. 2000, pp. 243–254.

[226] C. Lochert, H. Hartenstein, J. Tian, H. Fubler, D. Herrmann, and M. Mauve, "A Routing Strategy for Vehicular Ad Hoc Networks in City Environments," in *IEEE Intelligent Vehicles Symposium*, Columbus, Ohio, USA, June 2003.

[227] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad-hoc networks," *IEEE Network Magazine*, vol. 15, no. 6, pp. 30–39, Nov. 2001.

[228] S. Giordano and I. Stojmenovic, *Position based Routing Algorithms for Ad Hoc Networks: A taxonomy*. Kluwer, 2004, pp. 103–136.

[229] G. Coulouris, J. Dollimore, and T. Kindberg, *Distributed Systems, Concepts and Design*, 3rd ed. Addison-Wesley, 2001.

[230] C. Maihfer, "A Survey on Geocast Routing Protocols," *IEEE Communications Surveys and Tutorials*, vol. 6, no. 2, June 2004, http://www.comsoc.org/livepubs/surveys/public/2004/apr/maihofer.html.

[231] A. Ebner, L. Wischhof, and H. Rohling, "Aspects of Decentralized Time Synchronization in Vehicular Ad hoc Networks," in *1st International Workshop on Intelligent Transportation (WIT 2004)*, Hamburg, Germany, Mar. 2004, http://www.et2.tu-harburg.de/Mitarbeiter/Wischhof/Ebner WIT2004.pdf.