Chapter 3
Design of Low Data-Rate Environmental Monitoring Applications

Agnelo Rocha da Silva, M. Moghaddam and M. Liu

Abstract The majority of low-cost and off-the-shelf Wireless Sensor Networks (WSNs) solutions cannot adequately address issues related to an unattended deployment in a harsh environment, especially if the network needs to scale and achieve high density or high coverage or both. This is usually the case in environmental applications. In this chapter, this problem is investigated and extensive discussion on the pros and cons of a specific WSN design is presented. However, before moving from generic and well-established WSN solutions to customization, a detailed analysis of the gains of having a tailored design is necessary. Accordingly, a case study involving sparse deployments in outdoors is used to illustrate the process.

The majority of existing WSN deployments are for low data-rate monitoring applications, and some of the first WSNs were designed for environmental monitoring. However, many challenges remain to be fully addressed before low-cost and large-scale outdoor WSNs can become a more common reality [13, 15, 23, 24]. In this chapter, the rationale supporting this statement is presented and some guidelines for the design of specialized WSNs for such a scenario are proposed. For this purpose, a real-world environmental monitoring application is used as a case study along with a specific WSN design. The taxonomy given in Chap. 2 is used to characterize the most relevant aspects of the design. More specifically, for each design dimension, the pros and cons of the proposed solution are discussed. We also highlight the pros and cons of designing a specialized WSN as opposed to adopting generic off-the-shelf WSN technologies.

A. R. da Silva · M. Moghaddam
Electrical Engineering–Electrophysics, University of Southern California,
Los Angeles, CA, USA
e-mail: agnelosi@usc.edu

M. Moghaddam
e-mail: mahta@usc.edu

M. Liu (✉)
EECS Department, University of Michigan, Ann Arbor, MI, USA
e-mail: mingyan@eecs.umich.edu

H. M. Ammari (ed.), *The Art of Wireless Sensor Networks*,
Signals and Communication Technology, DOI: 10.1007/978-3-642-40009-4_3,

In the first section of the chapter, the challenges of environmental monitoring are discussed. We then present the requirements of an illustrative project called NatureMONITOR. A specialized WSN solution is proposed along with a discussion of its main advantages and limitations. In the third part of the chapter, the importance of properly applying taxonomy to a WSN project is highlighted. We also make the observation that this task is made easier when a solution is highly specialized. Each design option of the NatureMONITOR project is explained in more detail and alternative design options are also discussed. In the last section, lessons learned are highlighted with a conclusion that advocates WSN designs tailored to application-specific characteristics [33].

1 Challenges in Environmental Monitoring

One of the main challenges in typical environmental monitoring applications is actually the cost involved in deploying and maintaining a significant number of sensors in the area of interest. For instance, although commercial radio, cellular, and satellite telemetry solutions have been available for a long time [13], the cost associated with deploying and supporting such solutions is relatively high, even when only a dozen nodes are deployed. When larger areas (e.g., >1 km^2) are involved, the cost becomes a major factor in the feasibility of the project. About 10 years ago when WSNs were presented as a feasible low-cost solution to large-scale deployments, there was great expectation that these networks would be the key answer to indoor/outdoor environmental monitoring [1].

However, a careful investigation of recent WSN work in this area, such as the references provided in Chap. 2 shows that we have not achieved an *effective* solution. From a business (or project sponsor's) viewpoint, the effectiveness is measured by a good balance between the expected functionality of such a network and acceptable levels of cost, reliability, and scalability. We next discuss these 3 fundamental metrics, as illustrated in Fig. 1.

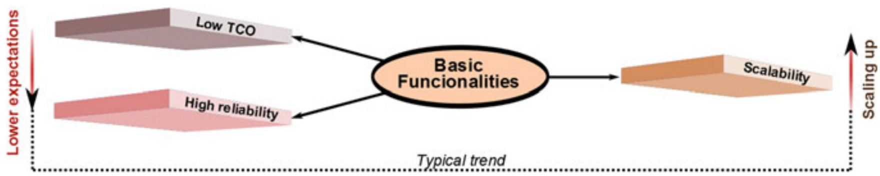

Fig. 1 The main challenge of designing a large outdoor WSN is to achieve a balanced solution involving TCO (effective cost), reliability, and scalability

1.1 Controlling the Total Cost of Ownership

While the terms *large-scale*, *long-term*, and *low-cost* are frequently used in the WSN literature, a certain proposed solution often falls short of meeting the requirement of an application. As an example, assume a sensor node carries a unitary cost of $50 and it employs a "perpetual" energy harvesting system using a solar panel. This does not automatically mean that the solution is truly low-cost. The true cost must be evaluated systemically, while sensors constitute only one part of the system. In particular, it is important to remember that the regular maintenance costs of a WSN are *not exclusively* related to the power modules. For instance, if periodic sensor calibrations are also necessary, then such costs must be included in the total cost of ownership (TCO) of the sensor node. Moreover, in an outdoor setting regular maintenance may be unavoidable due to a variety of reasons. Besides theft and vandalism, the following are some of the potential issues found in a typical outdoor deployment [27]:

- Negative effect of low-temperature on rechargeable batteries.
- Obstruction of the solar panel caused by dirt left by birds.
- Obstruction of the solar panel caused by shadow of trees (seasonal).
- Efficiency loss of the solar panel caused by long-term exposition to the environment (e.g., >1 year).
- Damage to the components of the sensor node, including the sensor itself, caused by insects and animals.
- Water infiltration.
- Condensing water inside waterproof enclosures.
- Communication issues due to new sources of radio frequency (RF) interference at the site.

Therefore, we argue that the key to controlling the total cost lies in reducing maintenance needs and operational cost. Best practices in the industry include preventive maintenance, which has long been used as a way to minimize the probability of failures of the system. During a *scheduled* maintenance, many components of the system are replaced simply because they are close to their expected lifetime.

Similarly, in the context of outdoor WSNs, it is important to determine whether there are tasks or situations that require regular human intervention; such needs may dictate other design aspects of the system. For instance, in a case where sensors must be physically inspected, cleaned, or calibrated every 6 months, the pros and cons of designing the lifetime of a sensor node for 2 years (in terms of its energy solution) instead of a value closer to 6 months must be re-evaluated. Observe that even a node with a perpetual energy lifetime will require such intervention. One way to minimize outdoor maintenance cost is to coordinate multiple maintenance tasks, e.g., scheduling them on the same, pre-defined dates. Knowing such a schedule affects the design consideration for the rest of the system. For instance, such maintenance cycle will ultimately define the expected lifetime of the node in terms of energy.

1.2 Reliability and Performance Metrics

In order to properly define maintenance dates for an outdoor WSN without compromising its functionality and costs, a *high degree of reliability* of system components along with a set of sound performance metrics are required. Observe that such statement does not follow the traditional assumption that WSNs comprise inexpensive nodes with a relative high failure probability. Component redundancy is a typical solution to achieve high availability, although it is not always feasible to have replicas of all components, not to mention that a complete self-healing solution is in general complex and expensive. An alternate approach is to develop smart strategies for the most vulnerable components. Observe that both approaches require a qualitative understanding of the reliability level of system components. Usually for mature solutions, this is known prior to implementation. However, early adopters of a new technology must use or develop measurement tools to acquire such knowledge. This is typically the case with large outdoor WSNs, where researchers and developers must determine what to measure to assess the reliability level of system components. The following is a list that exemplifies some of the expected performance metrics for this scenario:

1. Remaining energy level (real-time) of each node.
2. Ratio between the number of packets correctly received and that of packets transmitted.
3. Ratio between the number of measurements received by the sink node and the number actually taken by the sensor nodes.
4. Number of power shortages (or reset) on each node.
5. Average data-latency over the entire WSN.
6. Measurement health (or detection reliability) of each sensor module in each node.

We have now argued that low operating cost is key to making WSNs affordable for environmental monitoring applications, and that to lower maintenance cost it is important to identify the weakest components. However, to realize such analysis, at least basic network management functionalities must be available as briefly listed above. However, for the design of a large WSN, it is essential a clear understanding of how scalability affects both performance and cost, as highlighted in Fig. 1. Accordingly, in the next section, additional metrics related to the terms *large* and *sparse* are established.

1.3 Network Metrics for Large Network Design

The term *large-scale* in the WSN literature usually refers to a large number of sensor nodes over a relatively confined area (thus high node-density) [1, 23, 24]. However, we note that for outdoor WSNs sometimes large-scale is more related to large spatial coverage than the number of sensors. As an example, consider a 4 km-long beach that

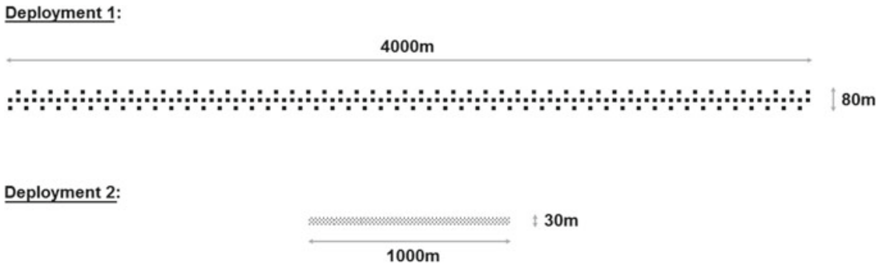

Fig. 2 *Two* possible deployment options for the beach example involving 200 sensor nodes

must be monitored for possible pollution, as shown in Fig. 2. If the budget constraint limits the number of nodes to 200, it may be more desirable to have these sensors covering all 4 km in a sparse deployment (Deployment 1) than to densely deploy these sensors at a few selected locations while ignoring the rest (Deployment 2). The technical issues underlying both approaches are distinctly different. While a solution may be scalable in terms of node-density, this does *not imply* that it is also scalable in terms of spatial coverage.

The above illustrates two types of scaling (at two extremes), one with increasing density within a fixed area (density scaling) and the other with increasing area with a fixed number of nodes (area scaling). In practice, a particular large-scale deployment may be some combination of the two with one being more dominant. Area scaling notably gives rise to the question of whether the network can remain connected as we stretch the area without adding more nodes. But the difference between the two also goes beyond the communication range. Depending on how the network scales, data-rate, duty-cycle, data-latency, communication errors, and energy consumption all vary in ways determined by the architecture, hardware, and protocols in use, and not all of this is well understood.

For the above example, if we assume a typical communication range of 50 m, the average number of one-hop neighbors for a node is 2 (Deployment 1) or 16.6 (Deployment 2). Therefore, the denser deployment (2) has the clear advantage of a plurality of data paths which is usually associated with a higher degree of data transfer reliability for the network. Similarly, the ratio between the communication range of the nodes and the average inter-node distance between a node and its 3 closest neighbors is 1.1 and 3.3 for deployments 1 and 2, respectively. Again, the probability of communication errors for the first case is significantly higher compared with the second one [43]. In short, for the same number of nodes, a sparser deployment is typically more critical.

1.3.1 Motivation for Metrics Toward Large Networks

The previous discussion highlights the importance of carefully considering the area scaling aspect when the scalability of a WSN solution is under analysis. Unfortunately, such information is typically ignored or not mentioned in WSN designs. For instance, many WSN papers discuss the scalability of networking protocols considering density scaling, but it is much less clear how the same architecture and protocols behave under area scaling. At the same time, cost concerns (both deployment and maintenance) suggest that area scaling is often the more relevant one, as shown in the preceding beach example. In this way, as the networks scales (area), it becomes more difficult to achieve the initial goals related to TCO and reliability, as illustrated in Fig. 1. In fact, rarely a large-scale deployment is reported as having a longer lifetime (e.g., >1 year).

The majority of the reported large-scale WSN studies are for scenarios where the average inter-node distance among immediate neighbors is still relatively small compared to the communication range of the nodes, implying a high-density regime [32, 34]. Consequently, the emphasis has been on handling interference in dense deployments and on examining the performance of collaborative and multi-hopping protocols for this scenario. When typical WSN solutions suggested for wider physical coverage area are considered [1, 17], multi-hopping usually works due to a plurality of paths available. For instance, ZigBee specifications allow deep multi-hopping (e.g., 30 hops under ZigBee PRO [10, 14, 16]) and ZigBee-based networks are being successfully deployed especially indoors. This type of solution obviously requires the deployment of an increasing number of nodes to cover a larger area so as to maintain a similar degree of density and this is the case for deployments inside buildings and industrial plants.

We strongly believe that the success of using wireless sensors in environmental monitoring applications, especially outdoor WSN deployments, heavily lies in the proper support for *sparse* deployments. This is the case because the goal of any environmental monitoring application is to capture certain underlying phenomenon of interest. The quality of our observation is determined by both spatial resolution of data (node-density) as well as the spatial diversity of the data (coverage area), sometimes one more than the other. Ideally if we can have both then we have the best information quality. However, if both cannot be afforded simultaneously due to high cost, then one must prioritize. For some applications it makes sense to first focus on a few select, small areas and get high-resolution data limited to those areas. For many other applications it may be far more sensible to start with a wider coverage first and then gradually increase the node-density over time if necessary and when more resources become available. This observation is also relevant to many envisioned infrastructure monitoring applications, such as the detection of leakages in oil pipelines, structural failures of bridges, and the detection of landslides in roads. Precision agriculture is another example [39]. Note that in all these examples high spatial coverage is usually a requirement.

When the number of sensor nodes is limited, node placement becomes a critical issue in the design process. Consider a deployment with 100 sensor nodes over an area of 500×500 m. If we assume a square grid deployment, the average inter-node distance among immediate neighbors is around 56 m. Many low-power radio transceivers actually offer good communication performance for distances higher than this one. Therefore, we can realistically assume that one node in this network will have more than 6 neighbors within its communication range of 100 m. In this case, we expect multi-hop protocols to work properly, e.g., in [21] a routing protocol was shown via simulation to achieve satisfactory results for 100 nodes non-uniformly spread in a 100×100 m area. Moreover, because of the high density, this network is more robust to the actual placement of sensors, i.e., changes in nodes' locations (as long as not too drastic as to alter connectivity) do not severely impact the performance of the network.

Now consider a second scenario with a sparser deployment: 100 nodes over an area of 3000×3000 m. The average inter-node distance among immediate neighbors is higher than 330 m. Although some WSN transceivers at their maximum transmit power levels can operate for this distance, the potential number of neighbors of a node is drastically reduced. In fact, to the best of our knowledge, there is no published work reporting a satisfactory multi-hop WSN solution operating in this scenario.

The connectivity challenge becomes more complicated when non-uniform deployments, obstacles, and environmental issues are considered. In this case, well-known multi-hop and collaborative protocols may not work properly in even relatively small areas. It is also clear that, when the number of sensor nodes is limited, the quality of observation is highly affected by the placement of the nodes. Due to the resource constraint, a good placement may mean spatially varying node densities over the target area.

We will consider hereafter a deployment with 100 nodes and a coverage area of 1000×1000 m. While not a particularly very large network, this example will help illustrate the problem of finding WSN solutions for realistic sparse deployments. In Fig. 3a, b we show a node placement in a square grid and a more realistic shape for the 1000×1000 m target area, respectively. In Sect. 2, a WSN solution to placement in the area shown in Fig. 3b is presented. The proposed architecture can be applied in different scenarios that also require high coverage area, or high spatial scalability.

When comparing Fig. 3a with b, it is clear that the latter case has more associated challenges. However, we still have to find a way to quantitatively express the challenges for this scenario. Moreover, similar to the standard processing benchmarks used by the microprocessor and computer industries, there is a real need of a way to compare the performance of WSN protocols for distinct and strategic cases involving large networks. Such cases can be real-world examples and/or artificial scenarios of network placements involving different number of nodes and coverage areas. The first step toward this goal is the formalization of metrics that better translate the challenges involving communication range, number of neighbor nodes, and distance to these nodes. Accordingly, in the next section we propose one of such metrics, although it is not complete.

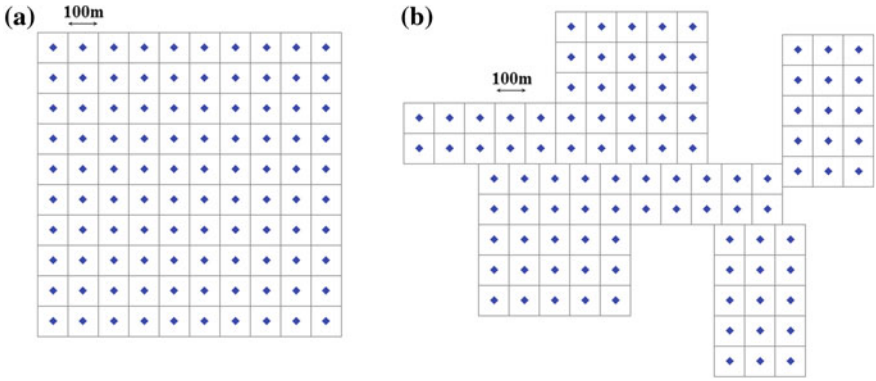

Fig. 3 **a** Grid-based node placement in a 1000 × 1000 m area. **b** An example of a realistic 1000 × 1000 m deployment area

1.3.2 AIND & ANON Metrics

In this section, it is introduced a novel network metrics, called AIND-ANON. AIND stands for Average Inter-Node Distance among immediate neighbors and ANON stands for Average Number of One-hop Neighbors. The metrics cannot be used in its current form as a network performance tool with the objective of comparing network solutions, although we acknowledge that such tool is potentially a first step toward this goal. The fact that the tool is not a complete one is revealed by the observation that the metrics does not capture the location and number of sinks (i.e., end-destinations) at the node topology. Without this information, it is not possible to calculate the average number of hops necessary to conclude an end-to-end communication transaction. Moreover, the sensing range of a node is assumed to be the same as the communication range, which cause distortions in the analysis of channel contention and interference [43]. Similarly, the tool does not capture aspects related to network congestion, data-latency, or network throughput.

Therefore, it is natural to question the effectiveness of the AIND-ANON metrics at the ongoing discussion in this chapter. The main compelling reason behind the development of this metrics is *simplicity*, that is, a simple way to formalize the concept of network sparsity. To this end, it is important to maintain the technical discussion in a higher level without involving too many details related to underlying protocols and deployment environment. Moreover, we are interested to understand how sparsity is related to network performance, at least in relation to the average success for delivering a packet. If we assume that such delivery success in a *multi-hop* network is strongly associated with the success of a node in communicating with its closest neighbors, we significantly reduce the complexity of the analysis. However, we will see that such assumption may not be realistic under certain circumstances.

The key input parameters of this high-level analysis are two: (a) the node topology in 2D (or 2D-equivalent from a 3D scenario) and (b) the expected communication range of the node. The rationale is simple: calculate the average distance of a node to its 3 closest one-hop neighbors and also calculate the total number of one-hop neighbors. By averaging these values for all nodes in the network, it is possible to have qualitative indicators related to the expected challenges for this specific topology when multi-hopping is used. This goal is achieved even without informing the location of the sink(s). If, on the average, the ratio between the communication range of a node and the distance to its closest neighbors are close to the unity, the probability of communication errors increase because the received signal strength is expected to be smaller. However, even for this worst scenario, if the number of one-hop neighbors of this node is relatively high, the likelihood of communication success increases. As expected, having multiple neighbors close to a node (dense network) is the ideal case for this analysis. At the contrary direction, having a scenario with few neighbors located at the limits of the communication range is the worst case (sparse network). Therefore, by using the AIND-ANON metrics, it is possible to get the sparsity degree of a network and infer about possible communication issues.

Observe that so far the use of the AIND-ANON metrics is always associated with the expressions *neighbors* and *multi-hopping*. It suggests that the analysis under this metrics will potentially degrade or fail when any form of segmentation or in-network aggregation is applied to the network thus decreasing the use of multi-hopping in that network. More specifically, assume that after analyzing a node topology using AIND-ANON we conclude that the network is very sparse and it has high probability of communication errors and low packet delivery rate. Although such deployment is potentially sparse from the physical point of view, it does not necessarily imply that the performance of the network is as critical as indicated by the metrics. The initial pessimistic conclusion about the network is potentially valid if the average number of hops used for the message delivery is high; otherwise, the conclusion can be invalid. The cases where the analysis with this metrics can be distorted are the ones where the average number of hops is close or equal to the unity and they are the following: (a) small networks and (b) any network that employs a form of segmentation or in-network aggregation.

Opportunistically, the above latter observation (b) is very important for our analysis. More specifically, if a multi-hopping network is analyzed under AIND-ANON as sparse (with high probability of network issues), any form of segmentation and aggregation can potentially mitigate the performance issues in that scenario. Although the AIND-ANON metrics cannot capture such improvement (there is no associated input parameter), it is not difficult to get the intuition behind the previous statement. While the metrics captures values on the average for the *entire* network, when aggregation and/or segmentation are used, the success of the message delivery is now constrained to a *smaller* physical region. For instance, in many cluster-based WSNs, the majority of nodes exchange their messages by means of a single hop to a cluster-head. Therefore, even if the average distance between a node and its immediate neighbors is high (compared to its communication range), the success of the solution is mainly associated with the average performance of the links involving nodes and their cluster-heads, not the links among regular nodes. Similarly, when one large

WSN employs a significant number of sinks, the same rationale applies because on the average a smaller number of hops is required.

We summarize this discussion as follows. The AIND-ANON metrics can be used as a simply tool to evaluate the sparsity level of a network with a specific node topology. Moreover, the metrics can be used as a qualitative tool to evaluate the expected challenges associated with the use of generic multi-hopping protocols assuming that no aggregation or segmentation effort is being used. If the metrics indicates that a network is very sparse, network performance issues are potentially expected. In this case, aggregation or segmentation efforts can be adopted to improve the performance of the network. Following this direction, it is possible to reach the extreme case where the communications at the entire network are based on one-hop links only. If these links have good performance, the overall network achieves good performance and easily scales. We will see that this rationale will be later applied for the design of the case study in this chapter.

Another alternative for enhancing performance in sparse networks without segmenting the network is the simple addition of extra nodes, mainly serving as relay-nodes. Typically, this is a very usual practice, although it comes with the penalty of a higher cost to install and maintain the solution. As expected, with more nodes, the network becomes denser and the existing WSN multi-hop protocols can potentially achieve better performance. If the AIND-ANON metrics are evaluated again for the new node topology, the values will indicate that the network density in fact increased, as expected.

Next, we will see how the values for the AIND and ANON metrics are produced by means of an algorithm. After that, we will study some cases of real-world deployments (large networks) and calculate the AIND and ANON values for each case in addition to brief discussions.

Before proceeding with the presentation of the AIND-ANON algorithm, two expressions must be properly explained. We define *immediate neighbors* as the closest one-hop neighbors of a node. In our proposed algorithm, we limited the number of these immediate neighbors to 3, meaning that AIND reflects the average inter-node distance in relation to the closest 3 one-hop neighbors. The reason to this constraint is simple: because AIND is ultimately related to the probability of communication errors due to high inter-node distances, we are interested to limit the analysis to the closest neighbors that have better potential to successfully perform the communication task. In relation to ANON, the term *neighbors* is related to all one-hop neighbors that are located at the communication range of a node. Each node is evaluated individually and the average values of AIND and ANON for the network are calculated.

Algorithm 1 AIND-ANON

Requires:	the network is, at least, weakly connected
Requires:	*n*: number of nodes, (n>1)
Requires:	*distance(a,b)*: inter-node distance between nodes *a* and *b*
Requires:	maximum communication range (MCR) of each node
Returns:	*AIND*: average inter-node distance among immediate neighbors
Returns:	*ANON*: average number of one-hop neighbors

```
For Each node i in the network
    AIND_i ← 0, ANON_i ← 0
    For Each node j in the network (j≠i)
        If node j is in the communication range (MCR) of node i
            AIND_i ← AIND_i + distance(i,j)
            ANON_i ← ANON_i + 1
        End If
    Next j
    If ANON_i >3
        For All node j in the network (j≠i)
            AIND_i ← first_smallest(distance(i,j))
            AIND_i ← AIND_i + second_smallest(distance(i,j))
            AIND_i ← AIND_i + third_smallest(distance(i,j))
        AIND_i ← AIND_i / 3
    Else If ANON_i > 0
            AIND_i ← AIND_i / ANON_i
    End if
Next i
```

$AIND = {}^{1}/_{n} \cdot \sum_{k=1}^{n} AIND_k$

$ANON = {}^{1}/_{n} \cdot \sum_{k=1}^{n} ANON_k$

In conjunction[1] with AIND and ANON, the maximum communication range of the nodes (MCR) will form the set of parameters that we want to consider. Note that the number of nodes is indirectly included when the node topology is analyzed and it is also used when the values for AIND and ANON are calculated. Such parameters (AIND, ANON, MCR) give an indication (although without accuracy) of how sparse the network is and the potentiality of reliability and connectivity issues when multi-hopping is intensively used. Using such metrics, we will eventually conclude that some scenarios involving distinct number of nodes can have the same *sparsity* degree and potentially share some of the network problems[2] associated with this fact. In other words, scalability issues are closely related to the sparsity level of the network. The AIND-ANON algorithm is presented in Algorithm 1.

The calculated ANON value represents the expected number of one-hop neighbors of a node in a network. A higher value for ANON typically is associated with a denser network and a smaller probability of issues related to the data transfer. In fact, with

[1] Contextual definition for a *weakly* connected network: considering the communication range of the nodes, there is at least one path that connects all nodes.

[2] The number of nodes, application/network duty-cycles, network topology, and selected WSN protocols will also be associated with potential bandwidth, contention, and similar issues. However, another metrics besides AIND-ANON must be developed to address these needs.

Table 1 AIND-ANON metrics: extreme cases interpretation

ANON value	MCR/AIND value	Extreme cases interpretation
Small	Small	Very sparse network (critical scenario): small number of data paths and high probability of communications errors
Small	High	Not a large network
High	Small	Rare case: an efficient deployment geometry (e.g., hexagonal tessellation) covering huge areas
High	High	Very dense network: issues related to the data transfer reliability and communication errors are significantly smaller

a higher number of neighbors, the potential existence of multiple paths for the same message increases the likelihood of having this message properly delivered.

The calculated value for AIND represents the expected average distance between a node and its closest one-hop neighbors. A smaller value for AIND in general is associated with a smaller probability of communication problems due to a higher signal-to-noise ratio (SNR) level. Naturally, the numerical analysis of AIND only makes sense if the maximum communication range (MCR) of a node is also included. Therefore, the ratio MCR/AIND is the ultimate metric of interest besides ANON. Although the Algorithm 1 supports heterogeneous MCRs, we will hereafter assume a single MCR for all nodes. An initial interpretation of the values of ANON and MCR/AIND is shown in Table 1.

In the next section, 13 real-world large[3] WSN cases will be investigated under the AIND-ANON metrics. Because only the node topology and the MCR is captured for each case study (no additional detail in relation to the protocols and location of the sink), the comparison between the cases are mainly to understand the network sparsity level for each case. Nonetheless, as already mentioned, the sparser the network is, the higher are the challenges to the deal with the performance issues if no segmentation or aggregation techniques are employed.

The focus of this chapter is related to the cases similar to the first scenario shown in Table 1, that is, large *and* sparse networks. In the next section, we will discover that few cases of the real-world deployments of large WSNs really lie on the first scenario. Nonetheless, important applications for outdoor WSNs require specific solutions for this scenario, as discussed in Sect. 2.

1.3.3 AIND & ANON Metrics Applied to Real-World Deployments

Chapter 2 listed 62 real-world deployments of WSNs and, from that list, only 10 are related to deployments of more than 30 nodes. On the other hand, there are deployments with fewer nodes that cover impressive large areas. In this section,

[3] The term large WSN in this context is associated to a high number of nodes, or to a large coverage area, or both.

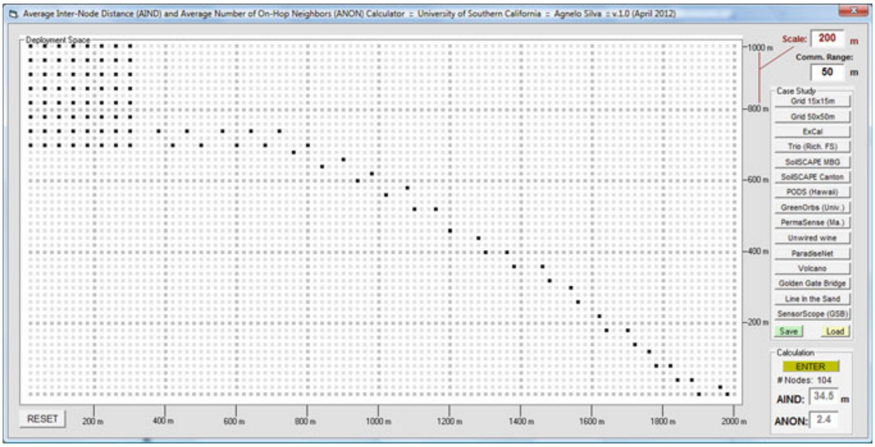

Fig. 4 AIND-ANON analysis for the PODS case [8]

some of these cases and additional ones are evaluated in relation to their AIND and ANON values. To this end, we developed a program (Microsoft Visual Basic) that implements the Algorithm 1. The interface of this program is shown in Fig. 4.

Due to the lack of precise topology information for some of the 13 selected cases, some of the distances are estimated based on additional reports and pictures provided by the authors or associated websites. Similarly, while in some cases the MCR value is explicitly provided in the related work, there are cases where we must derive the value for MCR based on the characteristics of the radio transceiver, antenna height, and related information. Usually, the adopted MCR value is significantly smaller than the communication range specified in the data sheets of the radio transceivers. It is explained by the fact that the MCR is associated with the *connected region* [43], that is, with the region related to good quality bi-communication. In particular for outdoor scenarios, the connected region is reduced due to the characteristics of the environment, such as vegetation and topography.

Next, the case studies are presented in chronological order and each figure shows the use of the AIND-ANON calculator software for each case.

Case 1: PODS [8]

This is a large deployment in both senses: number of nodes and coverage area. If the deployment was only restricted to the dense grid area (top-left of Fig. 4), the value of ANON would be higher. However, the network challenges significantly increase with the introduction of the non-regular part of the network. Consequently, the ratio MCR/AIND becomes significantly smaller. Therefore, the combination of a small ANON (2.4) with a small MCR/AIND (1.4) indicates that this network is sparse and critical network challenges are associated.

The details of the hardware are not extensively presented in [8] and, in particular network performance results are missing. However, based on the information about

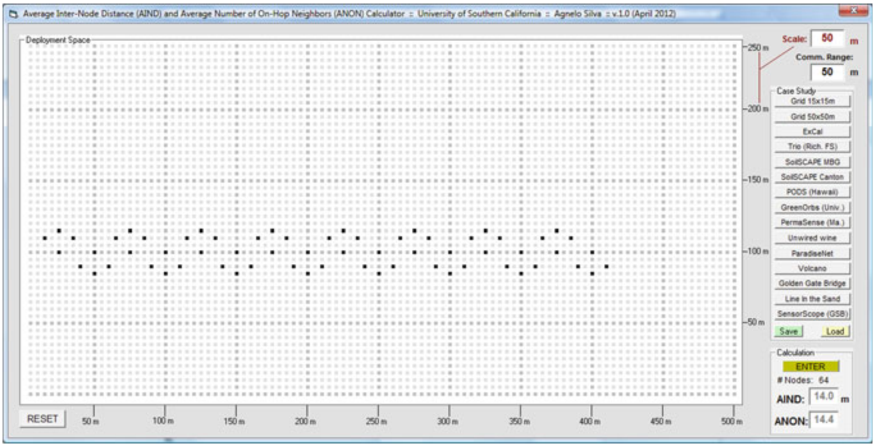

Fig. 5 AIND-ANON analysis for the Unwired Wine case [6]

the Tephranet node (900 MHz, +4.5 dBm transmit power), it is possible that the effective MCR (for a good quality communication) is higher than used in the calculation shown in Fig. 4. If this is the case, the network challenges can be potentially less critical than expected for this deployment. Again, the lack of network performance data in this case (the network operated for few weeks), impact a deeper analysis of the case.

Case 2: Unwired Wine [6]
The node topology for this case study was not provided in [6] and Fig. 5 represents the sparser configuration based on our interpretation based after reading the paper and researching associated work. The relatively high values for ANON (14.4) and MCR/AIND (3.6) indicate that generic WSN protocols would work properly for this scenario due to the high number of additional paths for the data transfer and potentially smaller communication error probability.

Case 3: A Line in the Sand [4]
This is a large network only in terms of the number of sensors. The Fig. 6 (observe the 2 m-scale) quickly reveals how dense this network is. Note that due to the lack of information for the antenna height, we adopted a conservative value for MCR: 20 m. Nonetheless, the very high values for ANON (77) and MCR/AIND (15.4) indicate that generic WSN protocols would potentially fit for this scenario. Nonetheless, there are other issues not revealed by the AIND-ANON metrics.

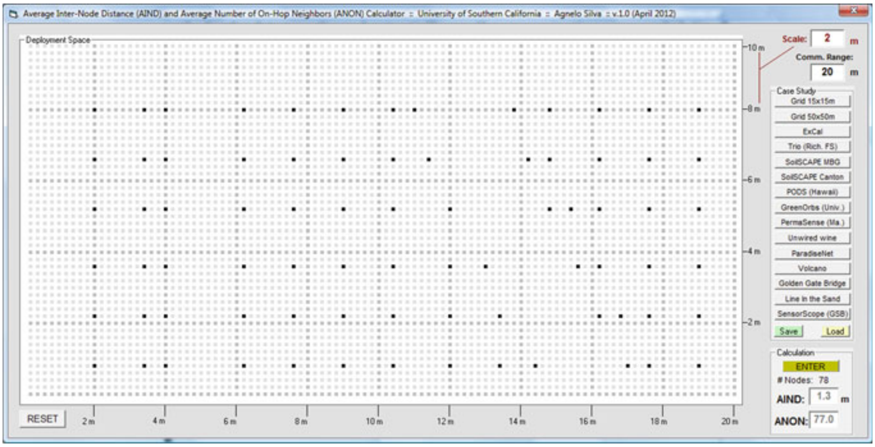

Fig. 6 AIND-ANON analysis for the A Line in the Sand case [4]

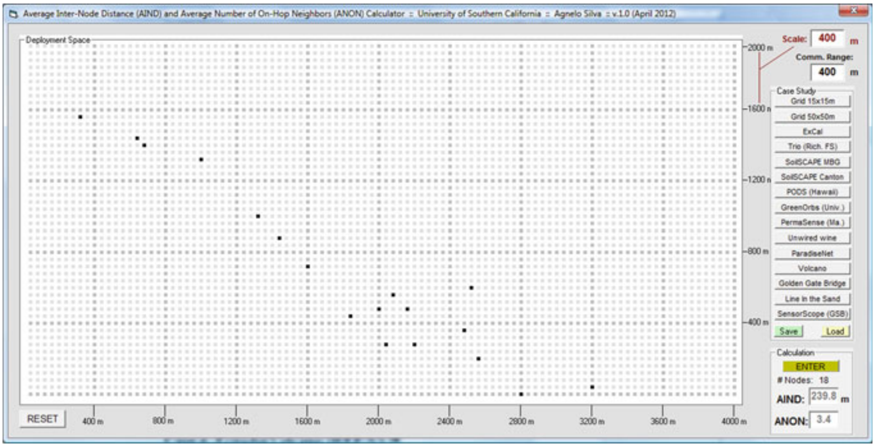

Fig. 7 AIND-ANON analysis for the Volcano (Ecuador) case [41]

Case 4: Volcano (Ecuador) [41]

This is a large network only in terms of coverage area. The proper choice for the antenna and radio transceivers allows the node to have a MCR of around 400 m. It is easy to observe how sparse this network is in the Fig. 7. Such degree of sparsity is also revealed by the small values in the AIND-ANON metrics: 3.4 for ANON and 1.7 for MCR/AIND. Similar to the first case (PODS), this is other case of a critical network with the potential of having high packet loss rate. In fact, the Fig. 4 in [41] indicates significant packet loss rate.

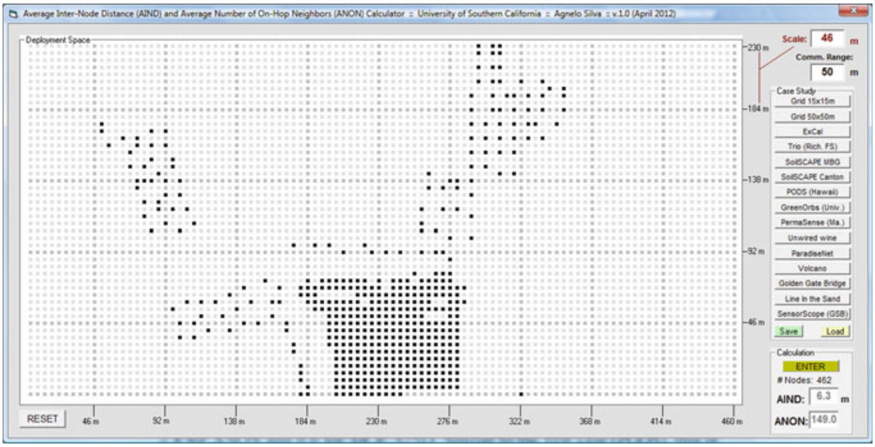

Fig. 8 AIND-ANON analysis for the Trio Testbed (Richmond Field Station) case [12]

Case 5: Trio Testbed [12]
The analysis of this case is partial: only 462 of the reported 557 nodes are actually shown in Fig. 8. Nonetheless, the additional nodes probably would not significantly change this metrics: 149 for ANON and 7.9 for MCR/AIND. Although the coverage area is not small, the network is mainly considered large due to the number of nodes. In fact, it represents one of the densest outdoor WSNs so far deployed. Similar to the Line in the Sand case, excluding contention and bandwidth issues, the network can potentially have good performance for the majority of the current WSN protocols, if energy and contention-related issues are disregarded.

Case 6: ExScal [3]
The AIND-ANON analysis in this case is also partial (now due to limitations of our program): only 824 of 1200+ nodes are actually shown in Fig. 9. Similar to the previous case, a high-dense network is identified: ANON = 35 and MCR/AIND = 3.4. Due to the existence of regions with sparser deployment, the expected network performance is a little bit more critical compared to the Line in the Sand and Trio cases.

Case 7: Golden Gate Bridge [19]
Due to the lack of 3D support of our program, the layout shown in Fig. 10 is an approximation of the real network. Nonetheless, the analysis of this layout must reveal some similarities with the Ecuador Volcano case, another very sparse network that has a large network section with a linear topology. In fact, ANON (2.3) and MCR/AIND (1.6) are small values and pretty similar to the values for the Volcano case. Unfortunately, there is no additional information about the network performance for this 3-week deployment, except the use of an efficient patch antenna in order to increase communication range.

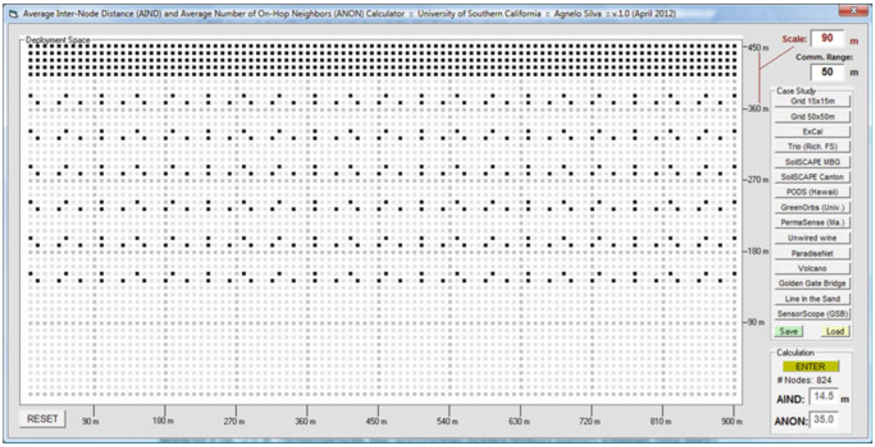

Fig. 9 AIND-ANON analysis for the ExScal case [3]

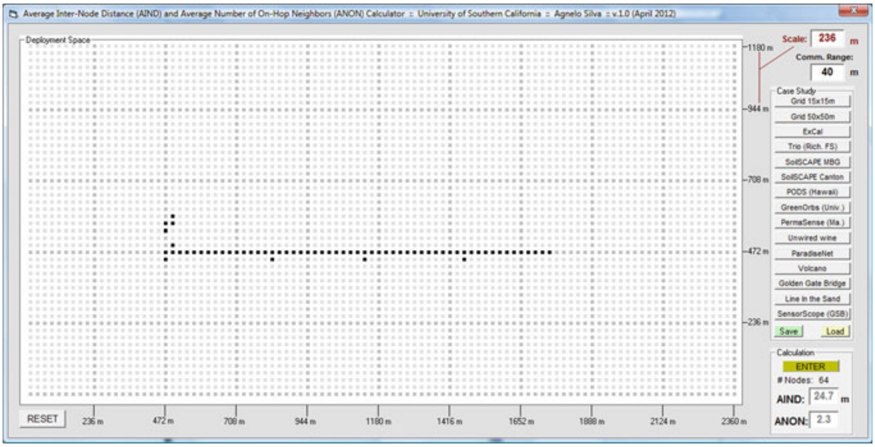

Fig. 10 AIND-ANON analysis for the Golden Gate Bridge case [19]

Case 8: SensorScope (Grand Saint Bernard) [5]

Due to the lack of precise information related to the node distances in [5] and also due to the lack of 3D support of our tool, the layout in Fig. 11 is an approximation of the actual network. Nonetheless, similar to the previous case, we still expect to have AIND-ANON metrics that reveal some degree of sparsity. In fact, MCR/AIND (2.2) is a relative small value. Although ANON (5.4) is not small, if we take into account that the each set of nodes is deployed in a distinct mountain, the real value of ANON can potentially be half of the calculated one. Therefore, the sparsity degree in this case is medium-to-large.

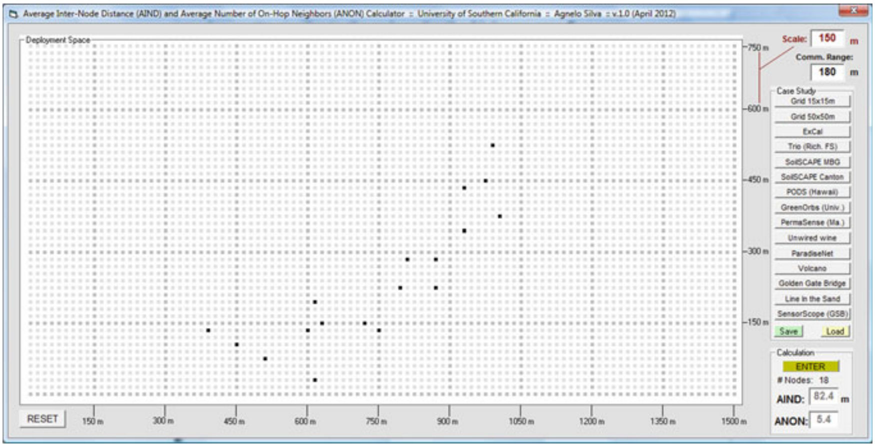

Fig. 11 AIND-ANON analysis for the SensorScope (Grand Saint Bernard) case [5]

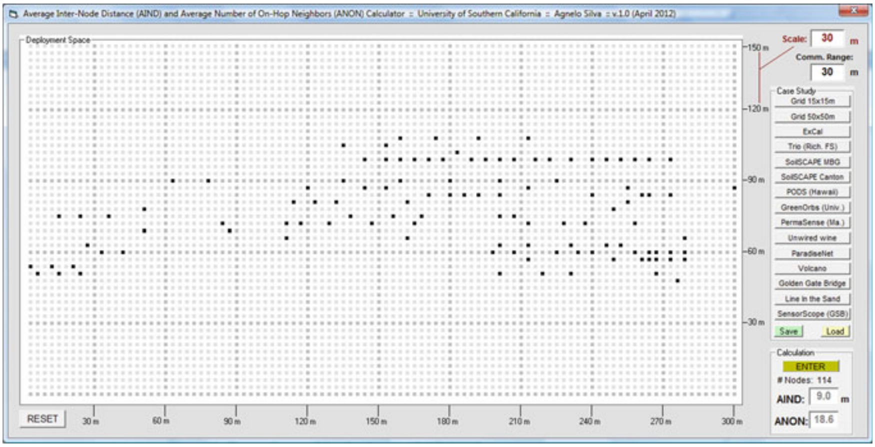

Fig. 12 AIND-ANON analysis for the ParadiseNet case [29, 30]

Case 9: ParadiseNet [29, 30]
The MCR in this case is strongly reduced due to the existence of large high-voltage transformers among the nodes. The Fig. 12 quickly reveals a very dense network. Accordingly, ANON is calculated as 18.6 and MCR/AIND as 3.3 which are relatively high values in our metrics.

Case 10: GreenOrbs (University Site) [24]
Another example of a dense network as shown in Fig. 13 : ANON $=$ 9.5 and MCR/AIND $= 2.1$. Due to the relatively small MCR/AIND, this network can present a higher packet error rate. However, due to the possibility of multiple data paths, the

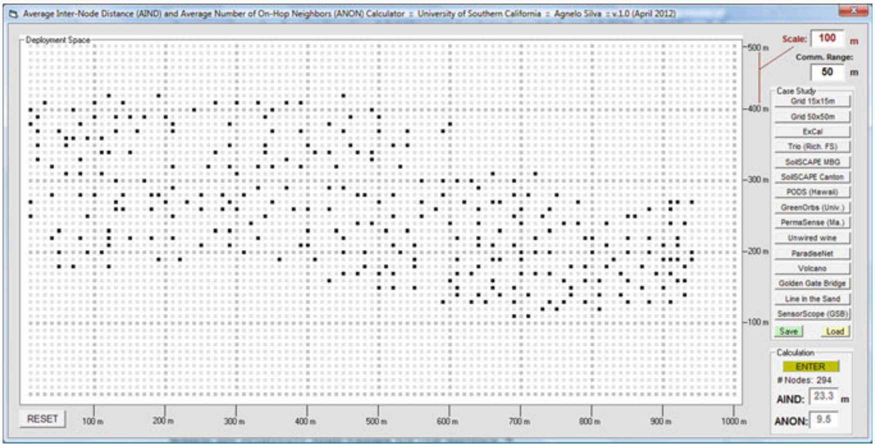

Fig. 13 AIND-ANON analysis for the GreenOrbs (University Site) case [24]

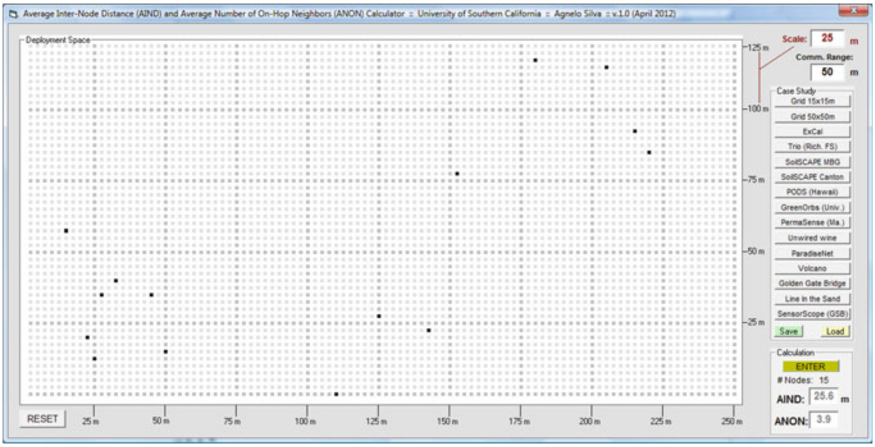

Fig. 14 AIND-ANON analysis for the PermaSense (Mattherhorn) case [7]

effective data transfer reliability for the network can be significantly superior. Therefore, such case study is another medium-to-large sparsity degree case.

Case 11: PermaSense (Mattherhorn) [7]
Although not involving a significant number of nodes, this is a large network in terms of coverage area. As shown in Fig. 14, this is definitely a sparse network: ANON = 3.9 and MCR/AIND = 1.9. The MCR was strongly reduced in this case due to existence of mountains in this scenario.

Case 12: SoilSCAPE I (Matthaei Botanical Gardens) [27, 28]
Involving an area of around $200 \times 300\,\text{m}^2$, this network is not very large but it is relatively sparse if we consider the number of nodes: only 27. With ANON = 6.4

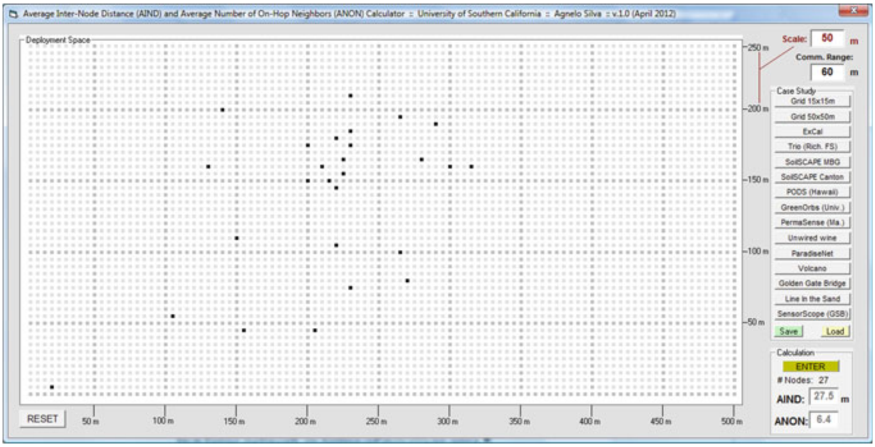

Fig. 15 AIND-ANON analysis for SoiSCAPE I case [27, 28]

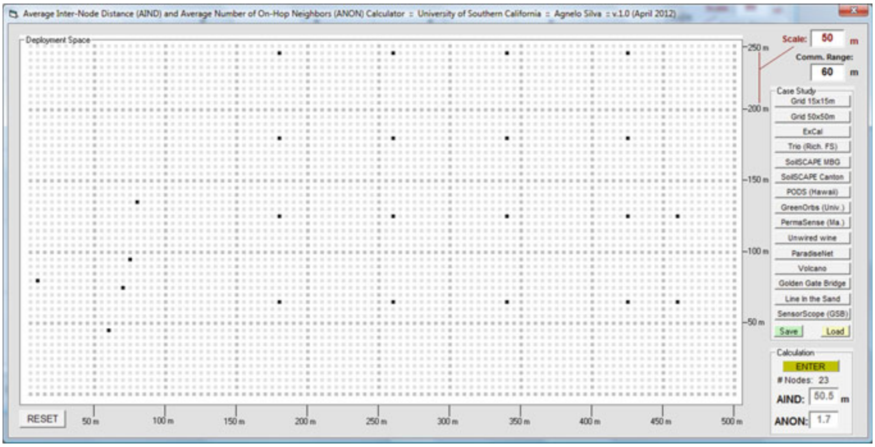

Fig. 16 AIND-ANON analysis for SoiSCAPE II case [28]

and MCR/AIND = 2.2, it has a medium-to-large sparsity degree. The majority of the WSN solutions will have good performance for this scenario.

Case 13: SoilSCAPE II (Canton Farm) [28]

Comparing Fig. 15 with Fig. 16, one can observe that SoilSCAPE II basically doubles the coverage area with a smaller number of nodes, i.e., a sparser network is achieved. In this particular scenario, the existence of obstacles (vegetation, trees, etc.) and differences at the topography make some nodes work better than others. Although MCR of around 400 m was achieved in ideal conditions, on the average 60m is the MCR for the nodes with their standard antennas [28]. The small values of ANON (1.7) and MCR/AIND (1.2) clearly indicate that this network is very sparse and the

Table 2 Estimated values of AIND and ANON for large-scale WSN deployments

Deployment case	#Nodes	MCR (m)	AIND (m)	MCR/AIND	ANON
Environment monitoring: PODS [8]	104	50	34.5	1.4	02.4
Vineyard monitoring: Unwired wine [6], approx. topology	64	50	14	3.6	14.4
Intrusion detection: A Line in the Sand [4]	78	20	01.3	15.4	77
Volcano monitoring (Ecuador) [41]	18	400	239.8	1.7	03.4
Outdoor Testbed: Trio (Richmond F. S.) [12], approx. topology	462	50	06.3	7.9	149
Intrusion detection: ExScal [3], based on a partial deployment	824	50	14.5	3.4	35
Structure monitoring: Golden Gate Bridge [19]	64	40	24.7	1.6	02.3
Environment monitoring: SensorScope (G. Saint Bernard) [5]	18	180	82.4	2.2	05.4
Substation monitoring: ParadiseNet [30, 29]	114	30	09	3.3	18.6
Environment. monitoring: GreenOrbs (Univ. woodland) [24]	294	50	23.3	2.1	09.5
Environment monitoring: PermaSense (Mattherhorn) [7]	15	50	25.6	1.9	03.9
Environment monitoring: SoilSCAPE I (Matthaei B. G.) [27, 28]	27	60	27.5	2.2	06.4
Environment monitoring: SoilSCAPE II (Canton farm) [28]	23	60	50.5	1.2	01.7

MCR: maximum communication range (on the average for all nodes). The values for MCR are approximated ones; they are based on the information provided by the related work and also on the characteristics of the environment, height of the antenna, and radio transceiver.

communication performance can be strongly impacted. Without a careful network design, this scenario is highly associated with high packet error loss and associated higher energy consumption. In fact, as reported in [28], the network only achieved certain degree of stability when 4 additional nodes (ZigBee routers) where added to the network.

The results of these 13 case studies are summarized in Table 2. Also, in Fig. 17, the values of MCR/AIND and ANON for these cases are plotted. It is clear that moving from the left-bottom part of Fig. 17 to the upper-right, the network becomes denser and many performance problems are potentially avoided in this way. However, due to the realistic budget constraints, more and more outdoor WSNs can be potentially located at the *sparser* area of the picture(left-bottom corner). In fact, this is the

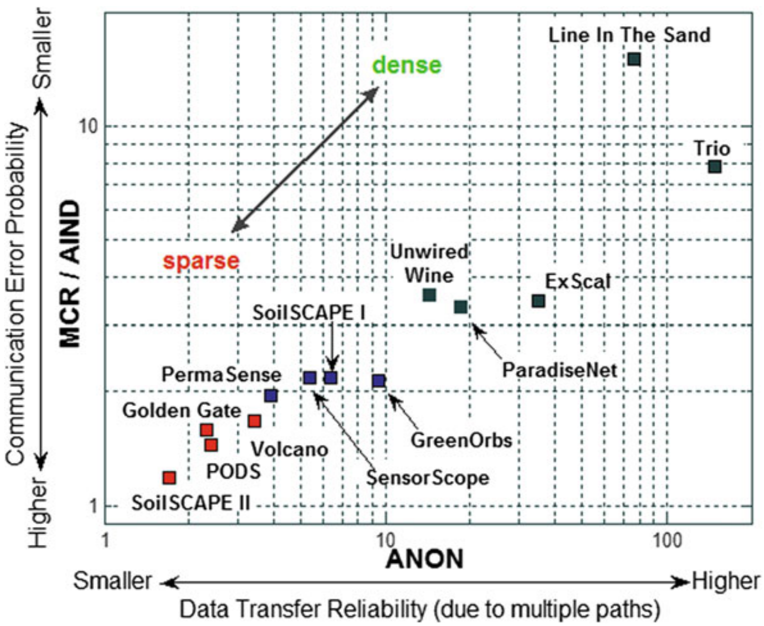

Fig. 17 AIND-ANON metrics applied to 13 large-scale WSN deployments

challenge behind the case study of this chapter, NatureMONITOR, as considered in the next section.

2 NatureMONITOR Case Study

In this section we introduce an illustrative project called NatureMONITOR to highlight the advantages of having a custom WSN design when off-the-shelf solutions cannot provide a complete answer. In this case study, the functional requirements are very strict (though realistic) such that the final solution is rather non-traditional compared to that typically found in the literature. Specifically, an application-centric approach is adopted in the design and many well-established principles are ignored. The main advantages and limitations of the proposed solution are presented. In the next sections more extensive discussion is provided where we also consider other, including more traditional, options. The final conclusions are not definitive ones but they stimulate discussions around the design of low data-rate environment monitoring and similar applications.

2.1 NatureMONITOR Project Specifications

Objective: 400 environmental sensors (humidity, solar radiance, soil moisture, wind spend, temperature, etc.) must be deployed and maintained over multiple years in an 1 km^2 area (hereafter called "cell") to provide in-situ measurements for a variety of environmental studies. Multiple cells will be deployed.

Business Requirements:

1. The deployment must follow the guidelines of the environmental scientists, the main users of the application, in that observations must be of sufficient spatial, topographical, and geological diversity across the 1 km^2 region. This could mean that some areas may need to have more sensors than others, so that the ensemble of data properly represents the region (e.g., its heterogeneity in soil makeup and so on).
2. At each node location, 4 environmental sensors are installed. Therefore, each cell has a maximum of 100 sensor nodes each one with up 4 sensors. While taking measurements, each environmental sensor consumes a maximum of 100 mW.
3. The default sampling rate is once every 15 min for all sensors. However, the system needs to be able to dynamically change this parameter for each location from a central web application.
4. The cell has roughly 1 km^2 of area but can have different geometrical shapes. It can be anywhere in the world and no assumption may be made about the topography and environmental parameters. In particular, support for extreme weather conditions must be provided.
5. The total material cost of a cell cannot exceed US$25,000, excluding the environmental sensors.
6. The total deployment cost of a cell cannot exceed US$15,000.
7. The annual operating cost of a cell (excluding the costs associated with the central data server and also the connection to it) must be smaller than $15,000. This value assumes the default sampling rate of 15 min.
8. The additional maintenance costs due to the usage of a more frequent sampling rate at some nodes for a certain period of time must be known a priori before adopting a higher sampling rate.
9. The maximum allowed latency between the moment when the measurement is performed and the moment when the data is stored in an existing central database (with Internet connectivity) is 24 h.
10. The sensing data must be time-stamped with real local time and the maximum allowed timing error is 1 s. It must also be stamped with the location and depth/elevation of the measurement.
11. When the number of environmental sensors that simultaneously experience problems in a cell exceeds 30, the network cell is considered "unavailable." Such problems may be related to both communication and sensory devices, e.g., defective measurements. The availability of the network cell must be higher than 95 %

over a period of 1 year. If the network cell is unavailable due to maintenance, these periods of time effectively count *against* the network availability metric.

12. The maximum number of lost measurements over a period of 1 month in a cell cannot exceed 90 % of the expected value (i.e., the amount of measurements scheduled to take place).

2.2 Functional and Non-functional Requirements

Observe that at the above business requirements list there is no mention to the term WSN. However, the project budget quickly rules out more expensive connectivity solutions like satellite or cellular connections for each location. We will therefore seek a WSN solution. The first step for the network designer is to evaluate the feasibility of the project, and then to use incremental deployments to test parts of the design under real-world scenarios due to the lack of a more systematic guideline [13]. However, it is expected that outdoor WSN deployments will eventually achieve the same level of standardization and out-of-the-box solution that some indoor applications are starting to experiment in recent years [10, 14, 16].

One approach to the feasibility study of this project is to fill and analyze a list of 29 strategic technical aspects of WSNs, as shown in Table 3. Each technical aspect will be marked either “Required” or “Not Required.” The term *Required* must be understood in this context as “must have” rather than “desired” (or “can be supported”). For instance, in our case study some form of time-synchronization is clearly required. However, “multi-hopping” or “reduced size of a node” are not explicit requirements and they are marked *Not Required* (even if desired). The main point here is to separate real functional requirements from features that are typically *expected* to be included in the design of a WSN solution. Each time a non-required feature is included in the design, the design becomes more generic and significant cost and performance penalties can be hidden in that decision.

Analyzing Table 3, we see for this case study that the majority of the *essential* characteristics of a typical WSN are marked as “not required.” This suggests that a simpler WSN tailored to the application may be the right answer for this project; many checks in the “Required” column would be an indication that a generic WSN may be a better design direction. In short, the main constraints of the NatureMONITOR project are:

- High reliability;
- Spatial scalability with low-density: large and sparse network;
- Unattended operation at harsh environmental conditions;
- Dynamic sampling scheduling;
- Real-time-based time-stamp for the measurements;
- High degree of accuracy of energy prediction at the level of a single node;
- Critical budget for deployment and ongoing operation.

Table 3 High-level business/functional requirements

Feature	Required	Not required
Bi-directional communication	✓	
Unicast communication		✓
Multi-hopping communication		✓
Multi-cast communication		✓
Broadcast communication		✓
Heterogeneous network (nodes with different profiles)		✓
Real-time communication ($< 1\ s$ data-latency)		✓
Node-to-node communication		✓
Continuous network connectivity		✓
Node mobility		✓
Well-planned node location	✓	
Support for random node deployment		✓
High-accurate localization of mobile nodes		✓
High reliability	✓	
Spatial scalability without high node-density	✓	
Ability to withstand harsh environmental conditions	✓	
Dynamic sampling scheduling	✓	
Remote reconfiguration (exclude sampling scheduling)		✓
Remote reprogramming		✓
Small size of the node		✓
High data-rate		✓
In-network processing		✓
Persistent data storage at the nodes		✓
Localization and timestamp for the measurements	✓	
Time synchronization among nodes		✓
Authentication		✓
Data encryption		✓
Remaining available energy prediction	✓	
Unattended operation	✓	

Harsh environment in this chapter refers to the scenario where the nodes are exposed to extreme weather conditions (sun light, wind, humidity, temperature, etc.) and the action of insects and animals. In some cases, the area is also one with very difficult access.

Considering the availability and characteristics of off-the-shelf hardware and software solutions and the costs associated with the development of fully customized solutions, the design team in this case study finally decides on a balanced solution, a non-traditional mix of telemetry [13], short-range wireless solutions [10, 14, 16], WSN technologies, and some degree of customization. The project proposal (in fact, this is the initial formal feedback to the project sponsors) is described as follows, as an extension and/or adjustment to the original business requirements:

Overview:

To cover a network cell with $1\,\text{km}^2$ area, an open, asynchronous, and hybrid wireless sensor network (WSN) is proposed. The cell is divided into multiple physical *segments*, each one with up to 30 sensor nodes. Four environmental sensors are attached to each sensor node. The nodes in a segment communicate with a special node called *master* of that segment. On the second layer of the hierarchy, master nodes communicate with the main gateway of the WSN. The wireless technologies used in these 2 layers are not necessarily the same. Therefore, this network solution is potentially a *hybrid* one.

Sensor nodes in the same segment are deployed up to 300 m away from the master provided that the communication performance is acceptable over that distance under different environmental conditions. No peer-to-peer communication or collaboration among nodes is provided. Communication between nodes and the master is based on commercial radio transceivers typically used in WSNs. The link(s) between the master(s) and the main gateway is (are) realized using point-to-point radio technologies capable of supporting distances of up to 5 km. However, short-range radios (including WSN-based links) can also be implemented depending on the distance.

The solution is considered *open* because both network tiers (sensor node-to-master, master-to-gateway) can adopt any current and future wireless technology that supports point-to-point connection. Possibilities include, but are not limited to, IEEE 802.15.4 [10, 14, 16], IEEE 802.11, Bluetooth [10, 16], Z-Wave [10], DASH7 [11], GPRS, VHF/UHF wireless modem, etc. The solution is also insensitive to the existence of low-level protocols that deal with medium access, reliability, and synchronization. All these features are actually implemented at the application layer (layover design), which facilitates the future change of wireless technologies if necessary.

All nodes, except the main gateway, are powered by non-rechargeable batteries. Assuming sensor measurements every 20 min, the expected lifetime of a sensor node (including the master) is 13 months. Thus every 12 months, human inspection is expected to replace all batteries and also to perform additional preventive maintenance tasks. This scheme is possible because all regular nodes are expected to have similar energy consumption. The nodes send data according to the sampling schedule provided by the gateway via the master. They follow a TDMA-like protocol implemented at the application layer in order to avoid medium contention [35]. Data collected by the master is stored in its persistent memory. Sometime later (e.g., minutes or hours), the master forwards the data to the gateway. These features outline an *asynchronous* behavior of the proposed network architecture.

In order to achieve the budget goals, the default sampling duration has been revised from 15 to 20 min. For many environmental monitoring applications we expect that such modification will not significantly impact the results, though this must be confirmed by the end user. Due to temporal and spatial correlation inherent in many environmental measurements, scientists routinely use subsampling and round-robin subsetting [23] techniques to make up for measurement losses (either in time or in space). However, to apply either technique, dynamic and individual measurement schedules must be supported by a sensor node. Incidentally, for this project such provision is also a business requirement. Therefore, it is possible to maintain the same energy budget (in terms of battery lifetime) and to have a default sampling schedule of 20 min while variations in this scheduling are supported for some of the nodes.

Besides the original functional requirements, the following functional and non-functional requirements are added to the project proposal:

- The maximum number of sensor nodes per network segment is 30, excluding the master itself which is also a sensor node (i.e., a cell has a maximum of 31 nodes).
- Adjacent network segments must use different wireless technologies or, alternatively, distinct radio channels.
- All nodes in the same segment must use the same wireless technology.
- The location of the master in a segment must be carefully chosen: each sensor node in that segment must communicate with (and only with) that master through a reliable communication link in order to satisfy the performance metrics previously mentioned.
- Every master has a persistent (non-volatile) memory and it must store at least 10 days worth of measurements from all sensors of the segment.
- The battery level of a node must be sent to the master along with sensor measurements. Such information must eventually reach the data server.
- Only non-rechargeable batteries are used on the sensor nodes (including the master) and such batteries must work from −40 to 70 °C. Many Li-$SOCL_2$ models support this range.
- Assuming that measurements from the 4 environmental sensors take place every 20 min, the lifetime of the non-rechargeable batteries must be at least 13 months considering the temperature range previously mentioned.
- A new sampling rate or schedule sent to a node must be applied within 1 h.
- The sensor node enclosure must be weatherproof and have IP67 or similar/superior rating.
- All nodes including masters and the gateway must have external watchdog circuitry in order to reset the device in case of a continuous (e.g., lasting more than 4 h) non-functional state.
- Nodes must synchronize their internal clocks with respect to the master at least every hour.
- The masters must synchronize their internal clocks with respect to the gateway at least every 24 h.
- The gateway must synchronize its internal clock with respect to the global real time clock at least every 24 h.
- The installation or presence of the sensor node (processor and radio modules) cannot interfere with the measurements taken by the environmental sensors at the same location.
- The system administrator or the end user must have a way to verify the health status (energy and communication performance) of each sensor node and the network as a whole.
- The system administrator or the end user must have a way to forecast the energy costs incurred by changes in the sampling scheduling of a node or a group of nodes.

As the design process progresses, some technical constraints emerges and some features that were not previously required become required. In the case of this project, the modifications are as follows:

- Unicast communication: **Required**.
- Heterogeneous network (nodes with different profiles): **Required**.
- Persistent data storage at the nodes : **Required for Masters**.
- Time synchronization among nodes: **Required**.

We next turn our attention to the issue of node placement. As already mentioned, the limited (and small) number of nodes over a large coverage area implies low node-density or areas with no nodes at all. Assume that, in this case study, the end user (i.e., the environmental scientists) indicate the areas that must be populated with sensors as shown in Fig. 18a. This placement is planned considering the topography, landscape, soil composition, and other science-related factors.

Using the Fig. 18a as a starting point, the WSN designer has now the task of considering available technologies and proposing the final node placement plan, an example of which is shown in Fig. 18b. In this case, it is clear that not all areas have an ideal representation due to the limited number of nodes, but all areas of *interest* are represented. The overall network architecture of the proposed solution is shown in Fig. 18c. As expected, this is a large and sparse network according to the AIND-ANON metrics, as shown in Fig. 19. In this case, assuming MCR = 100 m,[4] ANON is 2.9 and MCR/AIND = 1.4. We can easily anticipate that traditional WSN protocols will potentially face problems in this scenario.

The proposed system is a collection of 2-tier WSNs, each network segment with a star topology. For the lower tier (nodes-master), a customized overlay network solution is placed on top of a single-hop WSN. The master of each segment is usually close to the center of the segment. Note that the disc-shaped communication ranges are shown here for simplicity of illustration; it has been reported that such shapes are much more complex and irregular in practice [43]. We have used a smaller circle compared to the real communication range. While such circles include parts of the so called transitional region [43], the high-quality of the node-to-master links is achieved because the topology, transmit power level, antennas, hardware, data-rate, and protocols have been taken into account and verification tests are performed. In practice, only the most critical links of a network segment due to distance, topography, or existence of obstacles must be carefully tested and validated before the actual deployment.

Note that, in this case study, node placement is very carefully planned and some preliminary tests are even performed prior to the actual deployment. This is in stark contrast to a "random" placement that can be proposed for environment monitoring studies. As argued earlier, the cost constraint typically prohibits the adoption of a random or ad hoc placement. Accordingly, the domain knowledge that environmental

[4] Although the worst-case to be supported is 300 m, the AIND for this specific case is 69.2 m. Therefore, we want to consider a more realistic case where MCR = 100 m (rather than 300 m) considering the capabilities of existing WSN radios. The exceptions can be potentially solved with higher transmit power levels, special antennas, or the use of additional intermediate nodes as repeaters. However, for this preliminary analysis we want to see how critical would be the typical WSN solution (collaborative protocols) assuming the use of typical hardware and neglecting the worst scenarios.

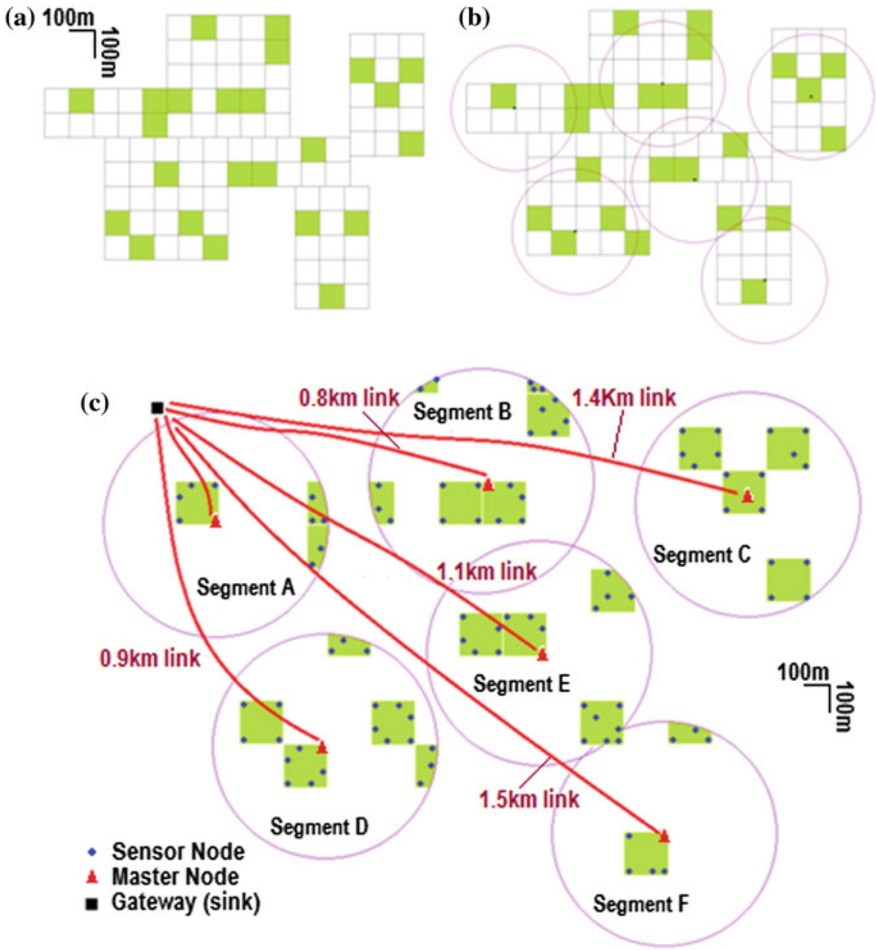

Fig. 18 **a** Areas to be covered by sensors as required by environmental scientists. **b** *Circles* representing the communication coverage of the sensor-to-master link; each *circle* is a segment. **c** Final placement plan for a cell (100 sensor nodes) based on a two-tier network

scientists have plays a fundamental role in the placement plan, as exemplified here.

The advantage of the 2-tier, non-collaborative, and asynchronous approach in the architecture for NatureMONITOR is the predictable network behavior which is highly deterministic. In addition, issues within one segment do not propagated to another. As a result, higher scalability is achieved both in terms of number of nodes and in terms of spatial coverage. It is clear that this architecture shares similarities with the existing LANs and WLANs: the scalability challenge is addressed by segmenting the network into small groups through the use of hubs/switches/access points and gateways/routers.

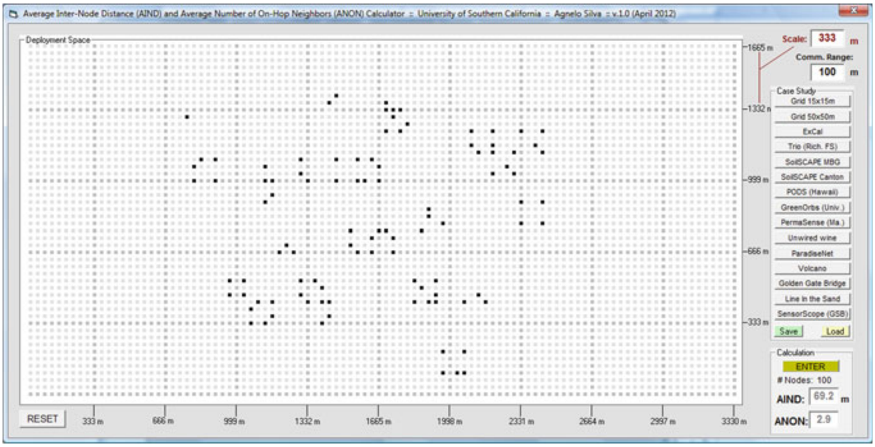

Fig. 19 AIND-ANON analysis for the NatureMONITOR case

In general, a planned deployment involving network segments or clusters has a special advantage rarely mentioned in the literature: the issues related to the communication of a regular node (end device, slave, etc.) and a cluster-head (coordinator, master, router, sink, etc.) can be solved case-by-case. For instance, a different antenna height or the adoption of a directional antenna can be investigated for the most critical cases involving a link of a regular node and its master. In ad hoc deployments and also in typical collaborative WSNs, such approach normally cannot be employed due to the plurality of neighbors of a given node. This fact is evidenced by the regular use of disc-shaped communication models for WSNs.

Another special feature of the proposed solution is the abstraction in relation to the underlying wireless technologies. There is no explicit tie between the architecture and a specific solution, such as IEEE 802.15.4, TinyOS-based, or similar low-range technology: any of these may be a proper answer. More importantly, a mix of these solutions is supported provided that a single segment shares the same technology. In other words, in a cell with 100 nodes, some segments may use 802.15.4 PHY 2.4 GHz standard, while one specific segment inside a dense forest adopts ISM 900 MHz links for the nodes. Such an approach is extremely flexible in controlling choices and costs. Moreover, the risks of having an outdated technology are significantly reduced. As an example, the design team may opt to use 802.15.4 transceivers as much as possible due to its high performance, current low cost, and high market availability. However, if after few years significant issues come to the scene due to over-utilization of the 2.4 GHz bands by devices nearby the deployment area, the radio transceivers can be simply changed by other solution (e.g., 900 MHz) that has similar or better performance.

In the next section we evaluate the proposed architecture underlying the NatureMONITOR project comparing it with more traditional WSN designs.

3 Why Applying Taxonomy to a WSN Project?

In this section we apply the taxonomy introduced in Chap. 2 to the NatureMONITOR project and its proposed architecture. More details about this architecture are provided and the advantages of applying such taxonomy are highlighted. By classifying our design, we have an opportunity to discuss different alternatives and be better prepared to identify the pros and cons of the solution [25].

Goal. NatureMONITOR is clearly a *sense-only* tool which is the case for the majority of WSN applications. Although the proposed architecture does provide some level of support for “sense-and-react,” it is not a functional requirement of the project. Note that cost, energy, and data-latency constraints make the implementation of a sense-and-react tool relatively difficult to be achieved in this scenario.

Time. The solution which is being proposed is clearly a *periodic* data collection one. In order to take advantage of (a) the pre-defined static node placement, (b) the low data-rate, and (c) the limited number of nodes per segment (30 sensor nodes plus the master), a TDMA-like MAC scheme is adopted in NatureMONITOR for each segment. A sensor node wakes up according to its measurement schedule and the master node wakes up when one or more nodes of its segment become active. Contrast this approach with a typical ZigBee network configuration where a router device, in contrast with our master node, must be always active [10, 14, 16, 27]. Because each node in a segment has its own assigned slot-time, medium access contention is kept at a minimum. Moreover, the energy consumption of a node is almost entirely determined by its measurement scheduling, making it highly deterministic, homogeneous among the nodes, and can be accurately predicted.

On the other hand, the simplicity of this time-division scheme comes with the drawback of a weak support for *event-triggered/driven* measurements because neither the nodes nor the master are continuously active. This is a potential constraint for some applications and the architecture proposed for NatureMONITOR is thus tailored for applications with a ***sense-send-sleep*** data pattern. This solution is also not suited for critical real-time applications, such as security monitoring/surveillance, due to its significant high data-latency.

One may argue that many existing event-driven WSN solutions could also be adopted for this project. Indeed, many MAC protocols allow a sensor node to sleep more than 99 % of the time while quickly waking up for fractions of a second. During this small active time period, events can still be detected and measurements conducted. However, the tradeoff here is the energy consumption [20]. Considering a traditional WSN hardware platform, in order to achieve very quick wake-up times, many parts of the hardware cannot be simply turned off and on again. Instead, the modules are usually put into some form of sleeping or standby mode. With this approach the latency to wake up the hardware can be as low as a few microseconds. However, because the modules (e.g., radio transceivers and voltage regulators) cannot be completely shutdown, the required sleeping power consumption can be significant as discussed next.

Even with the recent advance in low-power technology, the difference between digitally switching off such hardware devices and putting them in standby mode is between 1 to 2 orders of magnitude. Again, for medium to high duty-cycle networks (e.g., >1 %), this difference is insignificant as the energy consumption is dominated by the active power, not the sleeping power. However, for low duty-cycle applications, such as NatureMONITOR, the sleeping power becomes critical and a barrier to achieve significant energy savings.

Since there is nothing at requirements of the NatureMONITOR project that points to a future need for critical real-time support and the average sampling rate is once every 20 min, many parts of the hardware can be simply turned off. This approach is possible because the activation delay when the module is completely shutdown (e.g., hundreds of milliseconds) is still very small compared to the 20 min-period. By doing so, the design achieves energy consumption far below current WSN solutions.

Sensed Phenomenon. NatureMONITOR deals with *multiple* | *distributed* sensed phenomenon, while uses a large number of sensors. In this regard the proposed architecture is highly scalable as discussed in the previous section. A detailed investigation of current software and hardware WSN solutions suggests that very few existing solutions can achieve a reliable and low-cost solution for a scenario similar to the one in Fig. 18a: 100 nodes non-uniformly deployed over an irregular area of $1\,\text{km}^2$.

To understand how difficult it is to achieve spatial scalability, consider a larger area of $9\,\text{km}^2$ instead of $1\,\text{km}^2$, e.g., $3000 \times 3000\,\text{m}$. It is not difficult to see that the proposed architecture is ready to address this new scenario by the simply addition of more network segments. In essence, the segmentation and the two relatively independent communication layers (sensor node-master, master-gateway/data server) are the mechanisms that allow the architecture in NatureMONITOR to be highly scalable. By contrast, it is not clear how many large-scale solutions reported in the existing WSN literature would spatially scale without having to add infrastructure nodes, such as repeaters or relay-only nodes. Nonetheless, one notable solution for large and sparse WSNs is the use of mobile nodes [2, 36, 40]. Unfortunately, such infrastructure does not fit with the characteristics or requirements of NatureMONITOR.

A ZigBee-based solution has also been directly applied to a relative sparse network in [27], with 3 router nodes and 21 end device nodes. This is the scenario for the SoilSCAPE I case study. It is reported that when one or more of these routers failed, the network quickly became overwhelmed with excessive retransmissions and ZigBee-level traffic. Energy issues also emerged at the end devices due to damages to solar panels. Using a network analyzer, it was observed that the duty-cycle exceeded 10 %, far beyond the original application requirement ($\sim$0.3 %). This example shows the need of a significant network infrastructure (e.g., routers) to properly support the outdoor deployment based on ZigBee: 1 router per 7 nodes in this case [27]. Similarly, in [9] a large outdoor deployment with ZigBee uses 1 router per 8 nodes. Besides ZigBee, it is expected that many typical collaboration and multi-hopping WSN solutions also present network performance issues in the NatureMONITOR scenarios. A typical strategy in this case is the addition of *infrastructure nodes*, devices that are deployed only to maintain the network connectivity and better performance. As expected, this simple solution comes with a significant cost penalty.

Even if one artificially increases the network density (that is, extra nodes are not actually required by the application) and adopts a multi-hopping solution for the NatureMONITOR project without segmenting the cell, data-latency can be aggravated if a strong sleep policy for the nodes is maintained. In short, the higher is the number of hops in conjunction to longer sleeping periods of the nodes, the higher is the data-latency. For instance, for the deployment plan shown in Fig. 18c, assume the adoption of WSN nodes with transmit power between 0 and 17 dBm and frequency 2.4 GHz. In this case, the average number of hops required for one node to reach the sink of the cell is around 5. For the worst case, a node would have more than 10 hops toward the sink and the data-latency becomes significantly high.

Data-Rate. NatureMONITOR has *low data-rate* and the proposed architecture takes this as both an assumption and a feature to be exploited. If the data-rate requirement increases, even if each network segment has enough bandwidth to support this increase, two problems can potentially occur. First, the probability of network errors would significantly increase due to the combination of higher data-rates and high distances between the sensor nodes and the master. As a result, the proposed architecture may not achieve the expected reliability metric.

A second problem related to a higher data-rate is that long-distance links used to connect the master node to the gateway may not afford the bandwidth increase. Usually these links have smaller bandwidth compared to the short-range link used within a network segment. Therefore, some form of data aggregation and in-network processing would be necessary. In short, the lack of support for high data-rate applications is another drawback of the architecture proposed in NatureMONITOR.

Heterogeneity. NatureMONITOR uses an *architecture* that comprises 3 distinct: sensor node, master, and gateway, forming a heterogeneous WSN. Specifically, for each segment, the master node is in charge of collecting the data from all sensor nodes of that segment. In general, using another more powerful radio transceiver, the master node communicates with the gateway via a long-distance link. Due to the need of storing the measurements data in case of failure of this long-distance link, the master also requires some form of non-volatile memory. As a result, the communication, processing, storage, and power capabilities of the master are distinctly different from that of a sensor node.

The main weakness of this heterogeneous approach is the network unavailability in case of failure of the master node. If one regular sensor node stops working, the overall application does not suffer very much. However, if a master fails, its entire network segment becomes disconnected. If sensor nodes do not have any persistent storage, then all the sensing data of a segment will be lost in the meantime. One approach used in many WSNs that also employ special nodes is to apply redundancy for these nodes. Another approach is to increase the hardware/software reliability level of the special node (the master in this case) and to perform emergency maintenance for this specific node if necessary.

One of the resources commonly used to achieve high availability in embedded systems is the use of “watch-dog timers” (WDTs). Such optional device provides some form of initialization of the system when the latter freezes due to a software or hardware problem. For the master node used in this project, two independent WDTs are used. This solution obviously does not solve all the problems; thus long-term tests must be performed in order to verify the failure probability rate for the master node in particular.

Mobility. NatureMONITOR does not have mobility support, as it is not a functional requirement. It is also not hard to see that it would be quite difficult to modify the proposed architecture to support mobility: the proper operation and low energy consumption of the current solution heavily rely on the static nature of the system by means of a fixed node placement and a priori known sampling scheduling. On the other hand, the introduction of mobility brings randomness to the network topology and generic ad hoc WSN solutions surely are more effective. This observation highlights the fact that customized designs are adequate when the underlying fundamental assumptions are clearly understood and not broken. Otherwise, generic solutions are potentially in advantage.

For instance, one can argue that mobile data collectors can be a better solution for large and sparse networks [2, 36]. The design team of NatureMONITOR investigated this option, but because the sensor nodes are left unattended in harsh environment, no practical way to have a mobile data collector for this scenario was figured out. Nonetheless, there are scenarios, including environmental monitoring, that are suitable for mobile data collectors, such as the precision agriculture application in [39].

Connectivity. NatureMONITOR has an *intermittent* connectivity and this feature significantly simplifies the network design. Note that there is no peer-to-peer communication provision in this architecture and a fully connected network is not required. Moreover, the data over the most critical links (master-to-gateway) is protected by storage provision at the master side. The main weaknesses of having intermittent connectivity in NatureMONITOR are (a) lower reliability due to the lack of multiple data paths and (b) a higher data-latency. As already discussed, the sensor-to-master communication occurs over a one-hop link. Since a node is not logically connected to another in the same segment (even if both are within the communication range of each other), there is no other way for the data of one sensor node to reach the gateway or data server except passing through the master. Based on the requirements for NatureMONITOR, the total number of lost messages can be as high as 10 % of the total. For this particular application, due to the spatio-temporal correlation among measurements, such relaxed metrics are usually acceptable and the network architecture *exploits* this fact. However, other applications may require higher reliability on the data transfer.

NatureMONITOR adopts an acknowledge mechanism to guarantee the proper data delivery. However, the retransmissions are allowed in the same time slot. After the end of the assigned time slot, the sensor node simply discards the measurements for that cycle. Although more reliable transport protocols are available, it is clear that the simply approach under NatureMONITOR has a very small overhead while it still can satisfy the relatively relaxed reliability requirements of the project.

Message redundancy is another technique used to mitigate the reliability issue without compromising the deterministic behavior of the solution and this mechanism is also available for NatureMONITOR.

Besides weaker data transfer reliability, another drawback of having intermittent connectivity is the higher data-latency. In fact, the architecture behind NatureMONITOR does not properly support critical real-time and very time sensitivity applications, such as intruder detection. Two aspects of the architecture affect data-latency. First, the master collects the data from the nodes in its segment and only transmits that data to the gateway after receiving the data packet from the last node of a giving sampling cycle. Depending on the length of the node's time-slot and the number of active nodes, this delay can be on the order of seconds. A second source of data-latency is due to the transmission of data from the master to the gateway or data server. Although, it is possible that such data transfer occurs immediately after the conclusion of the sensor-master communication, the master-gateway data transfer can be delayed on the order of minutes or hours if energy or communication issues at the master node are considered.

Processing. The master nodes in the NatureMONITOR solution perform some form of *filtering and compression* in order to optimize the usage of time while transmitting data to the gateway. This is an important provision due to the high energy costs associated with the typical high-power transceivers and long-range links used for the master-gateway communication. However, due to the simplicity of this data collection application, there is no need for data processing at the sensor node itself, that is, a small message is enough for the transmission of the measurements.

On the other hand, some applications require significant in-network processing, particularly those associated with high data-rates. As previously mentioned, the architecture used in NatureMONITOR is not appropriate for these scenarios. In particular, typical in-network processing involves a high level of collaboration among nodes, which is not supported by this architecture.

Storage. The master node in NatureMONITOR has *persistent* data storage provision through the use of non-volatile memory and/or an SD Card. Thus, the measurements data are properly saved in case of energy issues at the master or communication problems in the master-gateway link. However, there is no similar provision for the sensor nodes. This is a decision aligned with the expected reliability metrics for the sensor nodes and the fact that communication range tests are conducted at the deployment site before the actual installation of the sensor nodes.

The main problem associated with storing data at the sensor node itself, an approach not used in NatureMONITOR, is the additional energy consumed by the operation. Moreover, under the current architecture, one time slot is only sufficient for the transmission of the measurements data collected by the 4 environmental sensors (also a single re-transmission in case of failure). The current slot structure does not allow transmission of more data, i.e., previous measurements that could not be transmitted. In short, there is no data queue at the sensor node's side.

Services. NatureMONITOR offers two network infrastructure services: ***time synchronization*** and a very basic form of node ***reconfiguration***. The former service is actually a requirement for the project because the measurements must be accompanied by a time stamp based on the real clock at the gateway. Even with cheap clock systems, the sensor and master nodes do not suffer significant clock skew effects in the proposed architecture. This is the case because every time a sensor node sends measurements to the master, it receives scheduling information that allows its clock to be automatically adjusted on the order of milliseconds. Therefore, only very large data sampling schedules, such as >15 h, may lead to time stamp errors beyond the project specification. Similarly, the master also adjusts its clock according to the gateway's clock every time it sends data to the gateway. Finally, the same process also occurs between the gateway and the central data server.

Node reconfiguration is a second requirement of the project. However, only the measurement scheduling is remotely configurable. In some cases, it may be desirable to implement full re-configurability of the sensor node without local intervention. The same is also true in relation to remotely upload a new version of the program that runs at the sensor node. NatureMONITOR does not fully support reconfiguration and reprogramming in order to maintain more deterministic network traffic, which is a key characteristic of the architecture. On the other hand, services that have regular and predictable behavior (fixed and regular bandwidth) are relatively easy to be implemented under the proposed architecture. Two examples are encryption and authentication services. They are not used in NatureMONITOR as they are not required, but their usage is feasible although resulting in additional network overhead. Finally, because NatureMONITOR is based on a static topology and a planned node placement, there is no need of localization techniques. A simple addressing scheme for the sensor nodes is adequate for this objective.

After applying the taxonomy to the NatureMONITOR project, we conclude that the proposed architecture is a highly specialized/customized WSN with the following characteristics: sense-only, periodic sampling, multiple sensors for a distributed phenomenon, low data-rate, heterogeneous and static nodes, intermittent connectivity, filtering and compression features at some special nodes, persistent storage at these nodes, time synchronization, and dynamic measurement scheduling.

We conclude this section with an important discussion related to the node size and battery choices. Note that small size is not a project requirement under NatureMONITOR. In fact, while size is critical for some WSN applications [32], it is usually not so important for outdoor scenarios compared to the cost of a sensor node. Also note that the antenna of a sensor node is usually placed more than 1m above the soil surface in order to increase the wireless channel quality. Therefore, cylindrical structures longer than 1m in length and 5–10 cm in diameter can be potentially used. There is no doubt that such form factor is far beyond the typical match-size of WSN nodes. However, for the NatureMONITOR project, there is no need to have a tiny sensor node. Also, the resulting gain in terms of available room at the node's enclosure can be exploited in favor of the adoption of **larger batteries**.

Despites the increasing popularity of energy-harvesting systems based on supercapacitors and/or rechargeable batteries [18], the design team in NatureMONITOR project opted for non-rechargeable batteries. This is because unknown weather conditions (recall that the system may be deployed anywhere) imply that the design must take into consideration extreme temperatures, which can significantly affect the performance of energy-harvesting solutions. Moreover, it was observed in [15] that the canopy would impact the efficient usage of solar panels. Snow, pollution, and dirt caused by animals are also potential sources of problems for these devices.

There are 3 additional factors that favor non-rechargeable batteries despite the fact that they must be replaced from time to time. First, non-rechargeable batteries have the highest energy-density in comparison with any other form of low-cost energy source for WSNs. Specifically, for the same physical volume the energy stored in a non-rechargeable battery can be 2 or 3 times that stored in a rechargeable battery. Second, more accurate methods to determine the end of life of these batteries are available in comparing with rechargeable cells. More specifically, when non-rechargeable batteries are used with a well-known discharging behavior, it is possible to predict the remaining lifetime of the node with a high degree of accuracy. Note that this aspect is actually a requirement for NatureMONITOR. Finally, non-rechargeable batteries have superior performance under extreme temperature variations. For instance, in [27] many problems with the nodes occurred when the rechargeable batteries stopped charging due to low temperatures ($< 0\,^{\circ}$C).

Therefore, in some outdoor scenarios it may be more economical to replace the batteries following a pre-determined schedule than having to deal with the uncertainty of maintenance of some energy-harvesting solutions. However, such a guideline is usually valid only when (a) the energy consumption is relatively small due to low duty-cycles and low data-rates and (b) the system has very stable network traffic. Because these aspects, in particular the latter one, are very difficult to be achieved in large-scale outdoors WSNs, the majority of these solutions employ rechargeable batteries usually associated to solar panels or other form of energy harvesting system.

4 Discussion and Conclusions

In this chapter, the advantages of designing a WSN according to specific application needs instead of going toward a generic WSN are highlighted. The filling practice involving the Table 3 allows the project manager and network designer to have a better understanding of the *actual* requirements that are behind a WSN project. It is highlighted that the fact that a certain feature is available and can be potentially supported in our design does not make it a *requirement* for the project. By removing such feature(s) from the design, it is possible to evaluate the option of having a customized WSN solution. In particular, low data-rate environmental monitoring applications can potentially follow this track.

However, before moving from generic and well-established WSN solutions to any form of customized solution, it is highly recommended to analyze the gains of having a tailored design. In fact, this project management principle must govern any project, in particular involving technologies that are quickly evolving. In this chapter, it is analyzed why sparse networks in outdoors deserve a special attention in their design. It is shown that, for some scenarios, the current WSN solutions may not provide the proper answer or, at least, the effective cost of the project may be significantly higher than initially planned. At the end of the chapter, a discussion about the pros and cons of adopting customized vs. typical WSN solutions is provided. The arguments are somewhat controversial because each WSN project has a significant number of design aspects to be considered and what is presented here as a proper solution for some particular scenarios may not satisfy the requirements of an ongoing project which a WSN designer is involved.

4.1 NatureMONITOR Project: Discussion

With the help of an illustrative project called NatureMONITOR, it is highlighted some of the advantages of tailoring the WSN design to the environmental monitoring application in comparison of simply adopting traditional WSN options. Nonetheless, there are tradeoffs to be considered. We summarize this high-level comparison in Table 4, where the architecture behind NatureMONITOR is compared with a commercial, ZigBee, and TinyOS-based solutions.

An interesting network metrics, called AIND-ANON, is presented in this chapter. The metrics can be used as a qualitative tool to evaluate the expected challenges associated with the use of multi-hopping protocols giving the topology of the nodes and their expected communication range. The proposed metrics are not complete but provide a formal way to define sparsity level in WSNs. Moreover, when a network is found to be sparser in this metrics, it is expected significant network issues. In the case of environmental monitoring systems, three solutions are suggested: (1) simply increase the network density by adding more nodes typically functioning as relay-nodes, (2) apply some sort of in-network aggregation in order to reduce the average number of hops required by the network, or (3) apply some of network segmentation. For the options (2) and (3), one can propose certain level of customization of the WSN design.

When a node topology for the NatureMONITOR project is analyzed under the AIND-ANON metrics, it is revealed that the network is very sparse and potential network issues are expected with traditional WSN multi-hopping solution are employed without a careful study. For this scenario, the network segmentation seems to be a good option. Moreover, considering the strict requirements of the project, a discussion about customization of the solution evolves. The idea of customizing a WSN design is not a novel approach. For instance, WSNs to be used in a human body are routinely tailored to the specific requirements of body-area networks (BANs). Additional examples are wireless underwater sensor networks [31] and wireless underground sensor network [39], both considered in this book.

For our illustrative large-scale and sparse monitoring application (NatureMONITOR), the proposed solution is highly flexible to future changes requested by environmental scientists or end users. For instance, based on data analysis from the previous year, an end user may decide to reduce the number of sensor nodes in certain areas with historically very similar results and spread them in areas where no

Table 4 Customized architecture (NatureMONITOR) compared to traditional WSN solutions

Aspect	NatureMONITOR	ZigBee PRO [9, 10, 16, 27]
Expected functionality	Appropriate for the project	Potential feasibility for multiple topologies assuming a significant number of routers (e.g., >1 per 10 nodes)
Cost	Appropriate for the project due to the network segmentation and open use of different wireless technologies	High deployment and operating costs. Just one network protocol is used and long-range communication must be achieved by multi-hopping and a high number of router devices
Reliability (and feasibility of accurate performance metrics)	Appropriate for the project assuming high-degree of reliability for the master nodes	*Uncertainty* due to inexistency of similar scenario in real-world
	The performance metrics control is achieved	Many of the requested performance metrics cannot be achieved with high accuracy
Scalability	Highly scalable due to the network segmentation and asynchronous behavior	Very good node-density scalability. *Reasonable* spatial scalability. Problems occur in very sparse networks
Aspect	TelosB+TinyOS [24, 43]	eKo solution [26]
Expected functionality	Potential *unfeasibility* for some topologies if protocols and design do not consider a sparser WSN. Potential use of multiple sink nodes	Potential *unfeasibility* for some topologies (maximum hop-distance is 5 for this solution). Multiple replicas of the solution are expected
Cost	Potential *unfeasibility* for some topologies if protocols and design do not consider a sparser WSN. Potential use of multiple sink nodes	Deployment cost far beyond the budget. Operating costs similar to TelosB case. In case of areas without sufficient solar radiation for extended periods of time, the rechargeable batteries must be changed frequently
Reliability (and feasibility of accurate performance metrics)	*Uncertainty* (same as the ZigBee case)	*Uncertainty* (same as the ZigBee case)
Scalability	Very good node-density scalability. *Uncertainty* related to the spatial scalability	Very good node-density scalability. *Uncertainty* related to the spatial scalability

sensor was previously deployed or in areas that require more resolution by means of a higher node-density. NatureMONITOR adapts and scales (number of nodes and space) nicely in this scenario. However, we still believe that this solution is not the unique solution. Different architectures can still satisfy the requirements while giving more emphasis on one of the three main design challenges discussed in this chapter: cost, reliability, or scalability.

It is not rare to see network projects (not only related to WSN) that start with a certain target related to the coverage area. Nonetheless, as the network grows, issues potentially rise and the deployment stops when the project reaches its budget. It is possible that the project never reaches the initial target in terms of scalability. In short, this is the problem with this design challenge (scalability): it is usually taken for granted. However, if the scalability aspect is analyzed since the beginning of the project, many constraints are properly imposed to the project. For instance, the designer can figure out that cheaper WSN platforms will not scale. Other more expensive platform claims to properly scale, but there is no evidence that it can achieve the level of expected reliability and tests must be realized. Even if the new solution scales and is reliable, its energy consumption can still be too high implying the need of frequent battery exchanges (higher cost). One can suggest the use of solar panels (or other form of energy harvester) but new problems can potentially come to the scene. Therefore, the real hidden challenge in designing large environmental monitoring systems is to achieve a balance involving functionality, cost, reliability, and scalability. In some cases, the solution comes in the form of a customized design as with the Nature-MONITOR case study. However, other techniques can be employed as discussed next.

Among the architectural aspects of the environmental monitoring application discussed in this chapter, dynamic scheduling was mentioned as one of the project requirements of the case study. In fact, the potential advantages of using this mechanism can significantly improve the energy performance of large and sparse WSN networks. For instance, an adaptive scheduling has been proposed as a solution for similar scenario [15, 22, 27]. Similarly, aggregation and compressive sensing [42] can be properly exploited in the NatureMONITOR project. Therefore, instead of following the segmentation and customization approaches, one can propose some of the mentioned techniques to highly mitigate the expected issues in large and sparse deployments.

4.2 NatureMONITOR Project: Preliminary Results

Although the NatureMONITOR project was introduced in this chapter as a hypothetical one, it was actually implemented in real-world deployments by means of an architecture called Ripple-2 [38]. The requirements of Ripple-2 are, in fact, a subset of the NatureMONITOR project but with a particular focus on soil moisture and temperature. We are still working on this project to implement all the strict network management requirements mentioned in this chapter. Nonetheless, the majority of the discussions related to NatureMONITOR also correspond to Ripple-2. For instance, the numbers behind the discussion involving the lifetime of the nodes and application duty-cycle in Sect. 2.2 come from our work with Ripple-2 [38].

The networks for the cases SoilSCAPE I and SoilSCAPE II, analyzed at Sect. 1.3.2, were finally converted to the Ripple-2 architecture. It is important to highlight that SoilSCAPE II was initially deployed favoring the segmentation approach (ZigBee network). However, it still presented scalability issues even with few hops involved. Moreover, a significant number of ZigBee routers are necessary to cover the area. Because these special nodes cannot sleep, their energy requirements are a challenge. In addition, we had many issues related to solar panels and rechargeable batteries associated with extreme temperatures, in particular subzero temperatures [37].

The Ripple-2 architecture gives a strong emphasis on non-rechargeable batteries and, accordingly, we work on hardware and software solutions to enhance the lifetime of such batteries in one more folds in relation to the current technology [37]. Therefore, we definitely followed the customization track discussed in this chapter. The solution comes as a software and hardware overlay [38], meaning that it can work on top of many WSN platforms. Because environmental monitoring systems typically have a very low duty-cycle, the concept of hibernation can be exploited. During this state, the inactive modules in the node are not simply in sleep or standby mode, they are turned-off. However, to implement this solution, we had to develop a cross-layer protocol. The effort paid off when we confirmed that, on the average, the additional effective network overhead is smaller than 1 %. For low duty-cycled applications, this result is very important in terms of energy efficiency [38]. Moreover, the effective packet loss rate is consistently below 2 % in all sites.

Up to the date, we deployed 105 nodes in 5 different sites with an average of 1 node per 5,000 km^2. No *extra* nodes, such as repeaters or routers, were used in these sites and the master node (Ripple-2 Local Coordinator) is actually also a sensor node which can also hibernate. So far, all SoilSCAPE nodes are based on 802.15.4 (2.4 GHz) transceivers, 17 dBm transmit power. The energy and network performance associated with the solution are potentially above the average in comparison with current WSN protocols [38, 42]. We are currently working to integrate the solution to ISM 900 MHz radio modules and a well-known WSN platform. The drawbacks of the architecture are exactly the ones discussed in this chapter: high data-latency and low throughput. In short, the network performance is traded for excellent energy performance. However, the low-cost, relative high reliability, and high scalability goals are achieved.

4.3 Conclusions

Although the expressions ah-doc, collaboration, and multi-hopping are constantly mentioned in the WSN literature, we believe that these features are systematically overvalued. Despites their high value in military applications and academic researches, a significant number of real-world applications can be properly addressed with basic networking functionalities. The recent industrial trend toward 802.15.4 in star-topology and ZigBee standards is strong evidence that simple solutions can be enough for many scenarios.

The existence of a significant number of solutions related to WSN is an indicative that hybrid solutions mixing state-of-the-art technology with some level of customization can be a promising direction for many applications, in particular low data-rate environmental monitoring systems. Nonetheless, as time goes by, new technological options and standards allow us to return to a more conservative, flexible, and generic solution avoiding the extra cost (and risks) of customizing a solution. However, as new challenges eventually appear according to the current demand, the specialization of the solution becomes again an appealing option. And this cycle repeats and not only for WSNs.

References

1. I.F. Akyildiz et al., A survey on sensor networks. IEEE Commun. Mag. **40**(8), 102–114 (2002)
2. G. Anastasi et al., Reliable and energy-efficient data collection in sparse sensor networks with mobile elements. Perform. Eval. **66**(12), 791–810 (2009)
3. A. Arora et al., Exscal: Elements of an extreme scale wireless sensor network. in *Proceedings of IEEE RTCSA*, 2005, pp. 102–108
4. A. Arora et al., A line in the sand: a wireless sensor network for target detection, classification, and tracking. Comput. Netw. **46**(5), 605–634 (2004)
5. G. Barrenetxea et al., Sensorscope: Out-of-the-box environmental monitoring. in *Proceedings of IEEE IPSN*, 2008, pp. 332–343
6. R. Beckwith et al., Unwired wine: sensor networks in vineyards. in *Proceedings of IEEE Sensors*, 2004, pp. 561–564
7. J. Beutel et al., PermaDAQ: A scientific instrument for precision sensing and data recovery in environmental extremes. in *Proceedings of IEEE IPSN'09*, 2009, pp. 265–276
8. E.S. Biagioni, K.W. Bridges, The application of remote sensor technology to assist the recovery of rare and endangered species. Int. J. High Perform. Comput. Appl. **16**(3), 315–324(2002)
9. H.R. Bogena et al., Potential of wireless sensor networks for measuring soil water content variability. Vadose Zone J. **9**, 1002–1013 (2010)
10. C. Buratti et al., An overview on wireless sensor networks technology and evolution. Sensors **9**, 6869–6896 (2009)
11. DASH7 Alliance, http://www.dash7.org Accessed 23 April 2012
12. P. Dutta et al., Trio enabling sustainable and scalable outdoor wireless sensor network deployments. in *Proceedings of IEEE IPSN'06*, 2006, pp. 407–415
13. S.A. Ellwood et al., in *Key Topics in Conservation Biology*, ed. by D. Macdonald, K. Service. Technology in Conservation: A Boon but with Small Print (Blackwell Publishing, Oxford, 2007), pp. 105–119
14. S. Farahani, *ZigBee Wireless Networks and Transceivers* (Elsevier, Burlington, 2008)
15. D. Ganesan et al., Networking issues in wireless sensor networks. J. Parallel Distrib. Comput. **64**, 799–814 (2003)
16. N. Hunn, *Essentials of Short-Range Wireless* (The Cambridge University Press, Cambridge, UK, 2010)
17. J.M. Kahn et al., Next century challenges: mobile networking for smart dust. in *Proceedings of MobiCom '99*, (Seattle, WA, 1999)
18. A. Kansal et al., Power management in energy harvesting sensor networks. ACM Trans. Embedded Comput. Syst. **6**(4), 1–38 (2007)

19. S. Kim et al., Health monitoring of civil infrastructures using wireless sensor networks. in *Proceedings of IEEE IPSN*, 2007, pp. 254–263
20. R. Kuntz et al., Medium access control facing the reality of WSN deployments. ACM Comput. Commun. Rev. **39**(3) 2009
21. X.F. Li et al., A differential evolution-based routing algorithm for environmental monitoring wireless sensor networks. Sens. J. **10**(6), 5425–5442 (2010)
22. J.C. Lim, C. Bleakley, Adaptive WSN scheduling for lifetime extension in environmental monitoring applications. Int. J. Distrib. Sens. Netw. (2012). doi:10.1155/2012/286981
23. J.C. Lim, C.J. Bleakley, Adaptive WSN scheduling for lifetime extension in environmental monitoring applications. Int. J. Distrib. Sens. Netw. **2012** (2012) article ID 286981
24. Y. Liu et al., Does wireless sensor network scale? a measurement study on greenOrbs. in *Proceedings of IEEE INFOCOM '11*, Shanghai, China, 2011
25. K. Lu et al., Wireless sensor networks for environmental monitoring applications: a design framework. in *Proceedings of GLOBECOM*, 2007, pp. 1108–1112
26. Memsic Corp. ēKo Pro series system, http://www.memsic.com/products/wireless-sensor-networks/environmental-systems.html. Accessed 23 April 2012
27. M. Moghaddam et al., A wireless soil moisture smart sensor web using physics-based optimal control: concept and initial demonstration. IEEE-JSTARS **3**(4), 522–535 (2010)
28. M. Moghaddam et al., Ground network design and dynamic operation for validation of space-borne soil moisture measurements: initial developments and results. in *Proceedings of ESTF-2010*, 2010
29. A. Nasipuri et al., Design considerations for a large-scale wireless sensor network for substation monitoring. in *SenseApp*, 2010, pp. 882–889
30. A. Nasipuri et al., Wireless sensor network for substation monitoring: Design and deployment. in *Proceedings of SenSys, Demo Session*, 2008, pp. 365–366
31. D. Pompili, T. Melodia, An architecture for ocean bottom underWater acoustic sensor networks. Poster Presentation, in *Proceedings of Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, (Bodrum, Turkey, 2004)
32. V. Rajendran et al., Energy-efficient collision-free medium access control for wireless sensor networks. in *Proceedings of 1st International Conference on Embedded Networked Sensor Systems*, 2003, pp. 22–27. doi:10.1145/958491.958513
33. B. Raman, K. Chebrolu, Censor networks: a critique of sensor networks from a systems perspective. ACM SIGCOMM Comput. Commun. Rev. (2008), pp. 22–27. doi:10.1145/1384609.1384618
34. I. Rhee et al., Z-MAC: a hybrid MAC for wireless sensor networks. in *Proceedings of 3rd International Conference on Embedded Networked Sensor Systems*, (2004). doi:10.1145/1098918.1098929
35. L. Selavo et al., LUSTER: wireless sensor network for environmental research. in *Proceedings of 5th International Conference on Embedded Networked Sensor Systems*, 2007. doi:10.1145/1322263.1322274
36. R.C. Shah et al., Data MULEs: Modeling a three-tier architecture for sparse sensor networks. in *Proceedings of IEEE SNPA*, 2003, pp. 30–40
37. A. Silva, M. Liu, M. Moghaddam, Power management techniques for WSNs and similar low-power communication devices based on non-rechargeable batteries. J. Comput. Netw. Commun. 1–10 (2012)
38. A. Silva, M. Liu, M. Moghaddam, Ripple-2: a non-collaborative; asynchronous; and open architecture for highly-scalable and low duty-cycle WSNs. in *Proceedings of ACM MiSeNet*, (Istanbul, Turkey, 2012), pp. 39–44
39. A.R. Silva, M.C. Vuran, $(CPS)^2$: integration of center pivot systems with wireless underground sensor networks for autonomous precision agriculture. in *Proceedings of ACM/IEEE International Conference on Cyber-physical Systems (ICCPS '10)*, (Stockholm, Sweden, 2010)
40. A. Somasundara et al., Controllably mobile infrastructure for low energy embedded networks. IEEE Trans. Mob. Comput. **6**(8), 958–973 (2006)

41. G. Werner-Allen et al., Fidelity and yield in a volcano monitoring sensor network. in *Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2006, pp. 381–396
42. X. Wu, M. Liu, In-Situ soil moisture sensing: measurement scheduling and estimation using compressive sensing, in *Proceedings of IEEE/ACM IPSN*, Beijing, China, April 2012
43. M. Zuniga, B. Krishnamachari, Analyzing the transitional region in low power wireless links. in *Proceedings of IEEE SECON '04*, vol. 2012 (Santa Clara, CA, 2004)