

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/260673783>

Energy efficient wireless sensor network communications based on computational intelligent data fusion for environmental monitoring

Article in IET Communications · September 2012

DOI: 10.1049/iet-com.2011.0809

CITATIONS

40

READS

291

6 authors, including:



Diego Francisco Larios

Universidad de Sevilla

32 PUBLICATIONS 268 CITATIONS

[SEE PROFILE](#)



Julio Barbancho

Universidad de Sevilla

66 PUBLICATIONS 709 CITATIONS

[SEE PROFILE](#)



Gustavo Rodríguez

Universidad de Sevilla

3 PUBLICATIONS 59 CITATIONS

[SEE PROFILE](#)



Jose Luis Sevillano

Universidad de Sevilla

117 PUBLICATIONS 1,264 CITATIONS

[SEE PROFILE](#)

Energy efficient wireless sensor network communications based on computational intelligent data fusion for environmental monitoring

D.F. Larios¹ J. Barbancho¹ G. Rodríguez¹ J.L. Sevillano² F.J. Molina¹ C. León¹

¹Department of Electronics Technology, Universidad de Sevilla, Seville, Spain

²Department of Computer Technology and Architecture, University of Seville, Seville, Spain

E-mail: sevi@atc.us.es

Abstract: The study presents a novel computational intelligence algorithm designed to optimise energy consumption in an environmental monitoring process: specifically, water level measurements in flooded areas. This algorithm aims to obtain a trade-off between accuracy and power consumption. The implementation constitutes a data aggregation and fusion in itself. A harsh environment can make the direct measurement of flood levels a difficult task. This study proposes a flood level estimation, inferred through the measurement of other common environmental variables. The benefit of this algorithm is tested both with simulations and real experiments conducted in Doñana, a national park in southern Spain where flood level measurements have traditionally been done manually.

1 Introduction

A wireless sensor network (WSN) consists of many small devices deployed in a physical environment. Each device, called a node, has special capabilities such as communications with its neighbours, sensing and data storage and processing [1]. The nodes make mesh networks of devices that can collaborate among themselves allowing the implementation of distributed solutions to solve complex problems.

Although a sensor networks have many applications [2], environmental monitoring is an area where the potential impact is huge, as described in [3]. Owing to this, these devices permit monitoring an area at a low cost and with little need of human presence. They are normally used in environmental monitoring to collect information via sensors incorporated into each node; pooling this information in a special device called a Base Station [4]. The Base Station generally acts as a gateway, allowing the user to access the collected data through an infrastructured network, such as Internet.

The technical requirements of a WSN applied to environmental monitoring include:

- *Autonomy:* Batteries must be able to power the nodes during the whole network lifetime. As the radio transceiver accounts for most of the power consumption in a node, the network has to reduce data traffic as much as possible, as well as the number of hops required to send a message.
- *Robustness:* In this kind of application, human maintenance is usually difficult because of the hardness of

the terrain. Therefore it is important to design robust software and hardware that can be adapted to any incident.

- *Flexibility:* The network must be able to add, move or remove nodes to meet the application requirements. The network must automatically detect the changes, organising the communications in consequence.

One of the most important constraints for this application concerns energy consumption. The batteries that provide power supply to these devices have, in general, a short life. To overcome this problem, this study presents an aggregation and data-fusion technique, that allows us to reduce the network's power consumption.

The parameter considered in this study is the water level in flooded zones. We studied it using a simulator to determine the advantages of local processing and data fusion to reduce power consumption. The data fusion is based on a local Self-Organised Map (SOM) [5] distributed in each node of the WSN.

The results obtained by simulation have been compared with the real deployment of a WSN. The experimental results in real scenarios demonstrate the performance of the system and validate the results obtained by simulation. Since the start of the installation we have collected partial information about the mounted sensors and their reliability. These results are discussed in Section 6.

The rest of this study is organised as follows: Section 2 describes the application scenario of the proposed method; Section 3 summarises the state-of-the-art about aggregation and data fusion in WSN; Section 4 describes the proposed aggregation methods with outcome of this method

developed in Section 5 and finally, Section 7 presents the concluding remarks and discusses future studies.

2 Application scenario

The proposed system has been designed to be applied in the National Park of Doñana. The Park is located in the south of Spain and covers a huge area of about 542 Km². Doñana is a wildlife reserve protected by the Spanish Government; it is an area that has seen little human interference through history. As a result, it is a privileged scenario for studying populations of wild animals and plants.

The international scientific community has been conducting research in Doñana for over 50 years. Biologists have traditionally deployed different kinds of sensors to gather information about natural phenomena such as vegetation stress, water levels of the aquiferous system, etc. Recent deployments of WSNs have increased the number of information sources significantly. Consequently data fusion and aggregation techniques are needed to manage such a huge amount of data [6].

Doñana is a harsh environment that does not allow sensors to measure the flood level directly. In this study we propose a data-fusion algorithm, to predict the flood level using other more easily suitable environmental variables, such as temperature or rainfall. To test the system, a WSN called ICARO [7] has been deployed in Doñana. This network measures environmental variables on the edges of flooded areas of the Park making it possible to execute the proposed flood level estimation algorithm.

The current prototype consists of nine measuring nodes and a Base Station. This deployment is shown in Fig. 1. The installation started operation in September 2010, and it is currently fully functional.

The developed network communicates using wireless technology based on the IEEE 802.15.4 standard, with a tree topology extending radio coverage. Many routing protocols have been proposed for WSN applications [8]. Some of them have been designed to reduce power energy consumption [9] in the network layer. Table 1 shows the

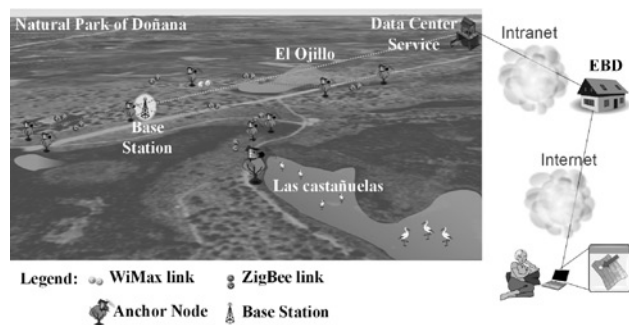


Fig. 1 ICARO network

Table 1 Protocol stack

Layer		Protocol
application	–	sCollection
transport	–	ARTICA-APP
network	–	CTP
link	LLC	AM
link	MAC	IEEE 802.15.4
physical	IEEE 802.15.4	–

protocol stack implemented in each node of the ICARO network. This stack is based on an implementation of a Collection tree protocol (CTP [10]) over the TinyOS operating system. CTP makes it possible to compute anycast routes to a single sink in the WSN from the rest of the nodes.

To do this, these classes of protocols implement two different services: one maintaining route information and the other transmitting information. In this implementation, every node receives an ACK at transport layer when the next node of the route receives the information, but no ACK is received in the source node when the sink receives the information. This is to reduce energy consumption. For the transmission service this protocol implements a best effort algorithm.

Collection protocols [11] commonly try to improve the performance or data quality of a network. The application we are proposing here aims to reduce power network consumption. To do this, in the ARTICA-APP protocol a hibernation mechanism is implemented in the transport layer. We will discuss this mechanism in Section 4.

The sCollection protocol is implemented on the top level of the protocol stack. For our applications, it offers high-level functions that allow information to be sent to the Base Station. The number of variables sent can change depending on the application. In the initial phase of the deployment, 47 bytes are sent in each transmission to the Base Station. These messages contain all the environmental information gathered by the systems' sensors.

In the initial development of the ICARO network, all sensor information is sent to a Base Station, where all the data is gathered and processed. This is done with a refresh rate of 1 min. Our aim is to use this prototype to test data-fusion techniques based on SOMs. The main objective is to reduce the traffic and thus the power consumption of the network. This would eventually enable us to drastically increase the number of measurement nodes, maintaining the low cost and small size of the devices.

The proposed SOM is executed locally in every node. This neuronal network combines the information acquired from multiple sensors and the output provided by the SOM is evaluated to determine whether the information is important enough to be sent. Our proposed approach is applicable to the study of other systems, with the same features (multiple sensors) and constraints (low-power consumption and low-data rate). The authors are currently studying other applications using a similar focus. For example, the estimation of temperature change from global warming, the evolution of the weight of birds, tracking the movement of animals, etc.

As mentioned before, a collection algorithm requires two classes of devices: a sink and many sources. In the ICARO network, the sink is the Base Station and the sources are the measurement nodes. In the following subsections we describe these devices.

2.1 Measurement nodes

Measurement nodes collect environmental variables such as temperature, rainfall, humidity, etc. The proposed data-fusion algorithm (to be described later in Section 4) is implemented in these nodes, which report their estimation of the flood level to the Base Station, as a function of the measured environmental variables. Each node sends its information to the Base Station once per minute when the data-fusion algorithm is not used.

These nodes consist of a small weather station, a processing unit based on a low-power microcontroller and a wireless radio transceiver (more specifically platform TelosB nodes [12]). In addition, the measurement nodes also incorporate a solar power system, designed to feed the system for several months even in complete darkness.

2.2 Base Station

The Base Station consists of a microcontroller and a radio transceiver, identical to those of the measurement nodes, but with a different firmware. In addition, it is connected to an industrial PC, which acts as a gateway between the WSN and the network of the Doñana Biological Station. As a result, the information gathered from the environment is accessible through the Internet. This node is also responsible for maintaining the network topology and organising communications between nodes.

3 Aggregation and data fusion in WSN

It is well known that using systems with multiple sensors presents many advantages, including:

- Increasing the robustness of the measurement, by obtaining multiple correlated measurements of the same process.
- Increasing robustness and reliability of the system in the event of sensor failure.
- In many cases, as with most environmental monitoring applications, a high number of low precision measurements over an area is preferable to a lower number of higher-resolution measurements.
- Reducing uncertainty.

Nevertheless, increasing the number of sensors has a cost in terms of the volume of message exchanges and power consumption. In addition, increasing the number of sensors also increases the amount of measurements gathered from the environment, making it more difficult to manage the greater amount of data. The trade-off between number of sensors and amount of information collected for an application is, in general, a complex problem.

WSN nodes have to manage their power consumption carefully as their power is usually supplied by batteries. This effect becomes more important when data has to be transported from a source to a sink through multiple hops. In this task, the nodes usually have to assume the role of gateways.

Nodes are usually deployed so that the information collected by one is often correlated with the information collected by its neighbours. Therefore, data from nodes deployed in the same area can be aggregated. Data aggregation and fusion (the process that combines different information into a more relevant and manageable form) reduce the amount of information gathered, minimising useless and redundant data and reducing traffic and power consumption at the same time. For these reasons, and because of the special constraints of WSN nodes [13], data aggregation and fusion have become key issue in WSN research [14].

Data fusion first appeared in the literature in the 1960s, as a mathematical model for data manipulation [15]. Nowadays there are many frameworks for designing data-fusion systems. One of the most widely used today was defined by the Joint Directors of Laboratories (JDL [16]), an organism

that belongs to the Department of Defense of the USA. This framework was developed to aid the design of military applications. It consists of the following steps [6]:

Level 0, preprocessing: On this level the data acquisition hardware associated with the sensor preprocesses the information collected by the sensor. This preprocessing may consist of data filtering, scale changing, unit conversion, etc.

Level 1, object refinement: Its goal is to identify objects and data association. To do this, the framework proposes fusing the attributes in entities from multiple sources, such as sensors or other entities. The procedures that could implement this step are data alignment, predictions of states, associations between data and entities and refinement of these entities.

Level 2, situation refinement: This step attempts to extract conclusions from incomplete information provided by the object refinement.

Level 3, impact assessment: The phenomenon is studied with the acquired data. This study intends to predict the performance of the phenomenon in the future.

Level 4, process refinement: In this step the system tries to increase the performance. It is necessary to take any trade-off with the resources used.

Level 5, cognitive refinement: On this last level, the human interprets the results obtained by the data-fusion processing. This way possible errors can be detected.

Data fusion can be a centralised process or a distributed process. Centralised data fusion takes advantage of having more information about the process. It provides a better picture about the state of the system, but it also increases the number and cost of the communications and increases storage needs. This is why systems with limited energy availability, such as WSNs with distributed data fusion (with almost all data locally preprocessed) represent a better approach.

Current research about data fusion can be split into three areas:

- *Research into the fusion function:* The main goals are to suppress redundancy and estimate a system parameter [17].
- *Research into system architecture:* The main goal is to determine how many data aggregators must be used in the WSN, and where [18].
- *Research into trade-offs and resources:* This research focuses on different trade-offs, such as energy against estimation accuracy, energy against aggregation latency, and bandwidth against aggregation [19].

The application presented in this study is an example of data fusion that can fit all three categories, as we shall see in Section 4.

The use of data aggregation over WSN is not new. Many data aggregation algorithms have been proposed in the literature [20, 21]. However, these algorithms have focused mainly on reducing the bandwidth used instead of reducing power consumption. These techniques try to reduce redundant information in a multi-hop network, where sensor measurements are acquired from several nodes.

In our approach, we execute a local data aggregation in every node of the network. To do this, we use the measurement of the sensors present in a particular node. This allows us to reduce power consumption and save bandwidth, as the amount of information sent through the network is minimised.

Among the many mathematical techniques for obtaining conclusions from sensor information, some of the most

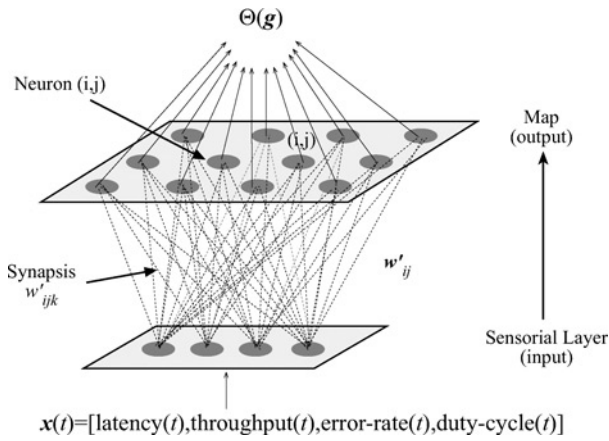


Fig. 2 SOM architecture

widely used are neural networks. One example is the Kohonen network or Self-Organising Map (SOM) [5], which is the technique used in our approach.

3.1 Self-organising map

SOM is an unsupervised neural network [22]. The neurons are organised in a unidirectional, two-layer architecture (Fig. 2). The first is the input or sensorial layer, formed by m neurons, one per each input variable. The input is formed by stochastic samples $\mathbf{x}(t) \in \mathcal{R}^m$ from the sensorial space. The second layer is formed by a grid with $n \times n' \times y$ neurons. Each neuron (i, j) is represented by an m -dimensional weight or reference vector called *synapsis*, $\mathbf{w}'_{ij} = [w'_{ij1}, w'_{ij2}, \dots, w'_{ijm}]$, where m is the dimension of the input vector $\mathbf{x}(t)$. The neurons in the output layer (the competitive Kohonen layer) are fully connected to the neurons in the input layer.

In SOM we can distinguish two phases:

Learning phase: In the learning phase, neurons from the second layer compete for the privilege of learning among each other, while the correct answer(s) is (are) unknown.

The effect of the unsupervised learning algorithm – also considered to be a weight update algorithm – is to distribute the neurons throughout the area of the n -dimensional space populated by the current training set according to the distribution of the data. This learning algorithm is detailed in Table 2.

Execution phase: In this phase the weights are declared fixed.

SOM gives an output denoted by F_{Level} that represents the flood level. This value is returned by a function Θ depending

on the winning neuron: $F_{\text{Level}} = \Theta(g)$. The function for this application is defined in Section 4.1.

4 Description of the proposed method

In the design of WSNs a trade-off is required between the number of message exchanges, number of hops and energy consumption. In Section 5.1 we evaluate this relationship for our network. Aggregation and data fusion would help in the design of the system to achieve this goal (Table 3).

The estimation of the node's power consumption is central to the strategy design of the network's behaviour. In this sense, the power consumption of the TelosB nodes can be detailed as follows: idle mode, 15 μ W; transmission mode, 38 mW; and reception mode, 41 mW [12]. Clearly, reception has a slightly higher power consumption than transmission, and both consume considerably more than in idle state. Therefore, to reduce power consumption one needs to reduce the number of message exchanges, and also stop all node activity enabling low-power modes by switching off the radio transceiver. A suitable activity manager with hibernation periods is required for low-power consumption systems. Several protocols for hibernation have been proposed [23, 24] to reduce power consumption, but the main problem lies in setting clock synchronisation in each node [25].

Our system uses a hibernation mechanism that has been implemented over the ARTICA-APP protocol in the transport layer (Table 2). This mechanism sends a wake-up beacon cyclically from the Base Station, to all the network nodes. These beacons announce the start of communications and provide information about the required on/off periods, T_{on} and T_{off} respectively. The measurement nodes use the reception of this beacon to synchronise a local timer on the basis of the received T_{on} and T_{off} information.

To receive the beacon, all measurement nodes must have their radio transceiver on. To do this, and considering that the local clocks may have an offset among the nodes, the measurement nodes start their radio transceiver a certain time before the Base Station sends its beacon (T_{delay}). In the ICARO network, the values $T_{\text{on}} = 30$ s and $T_{\text{delay}} = 10$ s have been used. In the initial deployment of the ICARO network, we set $T_{\text{off}} = 30$ s. This allows information to be gathered from the whole network every minute but, as discussed in Section 5.1, it involves high-power consumption. The goal of the proposed SOM is to reduce the power consumption while maintaining the usability of the system.

Our aim is to use artificial intelligence in the ICARO network to reduce data traffic, and consequently power

Table 2 Learning algorithm

Step 1:	synaptic-weight-vectors setup. In time $t = 0$, we can start with different configurations: $w'_{ijk} = 0$ or random values
Step 2:	in each iteration, take samples of $\mathbf{x}(t)$
Step 3:	for each neuron (i, j) , calculate the similarity between its synaptic-weight-vector, \mathbf{w}'_{ij} and the current input, $\mathbf{x}(t)$. A useful similarity criterion is the Euclidean distance:
—	$d^2(\mathbf{w}'_{ij}, \mathbf{x}) = \sum_{k=1}^m (w'_{ijk} - x_k)^2 \quad (1)$
Step 4:	declare the winning neuron: $\mathbf{g} = (g_1, g_2)$, which performs the minimum distance
—	synaptic-weight-vector update
Step 5:	end if the number of iterations has reached the maximum number of iterations or, if not, back to step 2

Table 3 Implementation of the winning neuron election in C++

```

int WinnerNeuron(float *x)
{
    float d2 = 0; % distance ^ 2
    float d[12]; % distance between input and every neuron weight
    for (int m = 0; m < 12; m++)
        d[m] = 0;
    for (int i = 0; i < 12; i++)
    {
        d2 = 0;
        for (int j = 0; j < 4; j++)
        {
            float aux = IW[i][j]-x[j]; % IW[i][j] is the input weights matrix,
            %obtained in the learning phase
            d2 += aux*aux;
        }
        d[i] = sqrt(d2);
    }
    float aux = d[0];
    int neuron = 1;
    for (int n = 0; n < 12; n++)
    {
        if (aux > d[n])
        {
            aux = d[n];
            neuron = n + 1;
        }
    }
    return neuron;
}

```

consumption, by filtering information. This network consists of ten nodes and was initially designed to send environmental information with a refresh period of 1 min. With the algorithm presented in this study, the data rate can be reduced, increasing the system's energy efficiency.

This project aims to obtain information in real time about the flood level in the natural environment described in Section 2. Until now, flood level estimation has been done manually, but muddy, boggy areas can make it difficult to get to the specific points. In some areas it is virtually impossible to resume the flood level.

In our approach each node executes an artificial neuronal network to estimate the water level of flooded zones. This execution is done once a day. The neural network is a SOM trained with historical data of the environmental variables obtained by the Biological Station of Doñana (EBD) over several years, as described in Section 4.1. The SOM receives data from environmental sensors installed in the node (level 0, source preprocessing according to the JDL definition of fusion described in Section 3) and produces an output that identifies the state of the water level with a neuron at the output layer of the SOM (level 1, object refinement). This procedure constitutes a data-fusion technique that suppresses the redundancy in the estimation of the system parameter: that is, the water level in a specific zone (level 3, impact assessment, and level 4, process refinement). Once the information of the flood level reaches the Data Center, it can be interpreted by the biologist (level 5, cognitive refinement).

All nodes process the information in the following way: if the state is different from the state estimated the day before, the node notifies a Base Station; if the state is equal to the

state obtained the day before the node sends no information. This way the nodes waste neither energy nor bandwidth. This situation could be categorised as data fusion based on trade-offs and resources, in this case of energy against estimation accuracy (see Section 3).

Since the nodes are deployed over a flood zone, and each node produces its own estimation, we can assume that there will be several estimations spread over the sensor network. Some nodes may change their estimation with a higher frequency than others, with the former producing more traffic than the latter. Since the information is sent to a Base Station using multi-hop protocols based on the IEEE 802.15.4 standard, this information should be aggregated.

At this point a second data-fusion technique is implemented to determine how many data aggregators must be used in the WSN and where. The use of aggregators enhances the speed of data processing (level 2, situations refinement, again following the JDL framework). When the Base Station receives the aggregated data, it sends it to the EBD through its Intranet network (Wi-Fi and Internet). At the EBD the biologists interpret the results obtained by the sensor networks, and evaluate the accuracy (level 5, cognitive refinement).

4.1 SOM design

We have considered a SOM design based on 12 output neurons (arranged in a 3×4 matrix). The SOM used a first layer made up of five input neurons, one for each input variable (shown in Table 4). These variables were chosen according to their relevance in the estimation of the flood level, and their values were obtained from the historic information gathered by the EBD. Other information, such as wind direction and wind strength, was not considered because historic information of these parameters was not available. Using these variables and the SOM, the flood level in a marsh it can be estimated, using a Θ function (see Section 3.1).

The SOM has been trained according to the algorithm described in Section 3.1, executing 50 000 training cycles.

Fig. 3 depicts the winning neuron from the SOM, considering one sample every 5 days from the 2002–2003 hydrological year (73 samples in total). This figure represents the SOM, with the big circles being the possible winning neuron; they are surrounded by smaller circles that depict each considered sample. The inside of the circles shows the date of the input that originates these winning neurons. The grayness of these circles varies depending on the month's rainfall. This figure shows that the winning neurons are polarised according to the flood level in two groups, one with a low flood level (corresponding to summertime, with low rainfall) and another with a high flood level (corresponding to winter, with high rainfall). This figure shows that the SOM acts as a dendrogram,

Table 4 Inputs of the SOM

Name	Symbol	Units
rain level	R_i	centimeters
maximum temperature	T_{\max}	degrees centigrades
minimum temperature	T_{\min}	degrees centigrades
average temperature	T_{avg}	degrees centigrades
frosty day	F	binary: 0 or 1

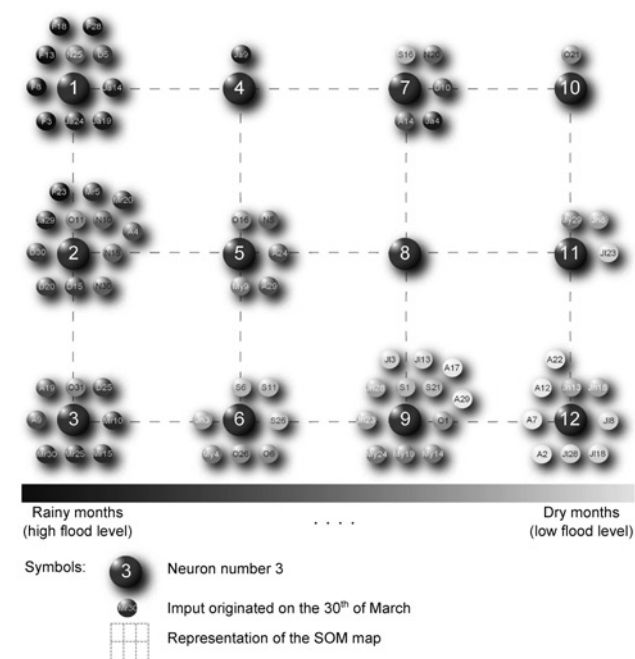


Fig. 3 SOM obtained for the training year

grouping the results according to the weather information of the inputs.

After the winning neuron is selected, we use the output function Θ to assign a flood level; this value is denoted as F . The method for defining Θ is based on a data aggregation of the current flood level obtained from the historic data. That is, for each winning neuron, all the historic values that activate this winning neuron are considered. Two different criteria have been studied:

- *Mode* (F_{Mod}): As the flood level for each winning neuron, we consider the most frequent flood level of the input that activates a winning neuron.
- *Average* (F_{Avg}): As the flood level associated to each neuron, we consider the average of all the flood levels over the days that activate a winning neuron. It can be obtained as:

$$F_{Avg} = \frac{1}{n} \sum_{i=1}^n F_{Real,i} \tag{2}$$

where $F_{Real,i}$ are the real flood levels of the historic values for each winning neuron.

Table 5 Assigned flood level

Neuron	F_{Mod} cm	F_{Avg} cm
N1	72	59
N2	72	53
N3	70	52
N4	65	58
N5	35	37
N6	15	21
N7	60	50
N8	0	17
N9	0	15
N10	0	16
N11	0	18
N12	0	13

The flood levels considered for each neuron are summarised in Table 5.

For this application, we used the Θ function based on the F_{Mod} mode because in our tests it was more accurate.

5 Simulation, tests and results

To test the proposed flood algorithm and compare it with the classical approach (i.e. sending all the information cyclically, whether it is useful or not to the Base Station, without local preprocessing, as described in Section 2), we designed to consider two metrics: power consumption and throughput. This *ad hoc* simulator is written in C++. With this simulator the user can test the consumption of the system versus the variation in the measurement obtained from the sensors and obtain the winning neuron of the SOM. This simulator calls a library, written in C, that implements the execution phase of the SOM described in Section 3.1.

The simulator is divided into three parts. The first one shows a representation of the topology of the ICARO network in the background consisting of nine nodes. This is the network considered for the power consumption estimation, on the basis of the power consumption of the TelosB nodes deployed within it (see the beginning of Section 4). As input for the simulator, the operator can introduce all the measurement inputs for each node. After that, the user can execute a simulation by pressing the button ‘Execute a day’. When the operator does this, the console shows information about the energy consumption of each node in the network and the total energy consumption of the network. It also shows the value of winner neurons for all nodes that changes their state and sends a new message to the Base Station.

This information allows us to compare the advantages of the proposed solution compared with other methods.

Fig. 4 shows the efficiency of our algorithm comparing the real flood level, and the estimated flood level (with both criteria: mode and average). As a conclusion, the design SOM offers good performance in the flood level estimation.

5.1 Estimation of power consumption

As discussed before, power consumption is one of the most important constraints of the WSN, because of the energy cost of messages exchanges.

The proposed simulator considers the TelosB power consumption as the basis for obtaining entire network consumption. In addition to the consumption of each device, the simulator also offers information about the path

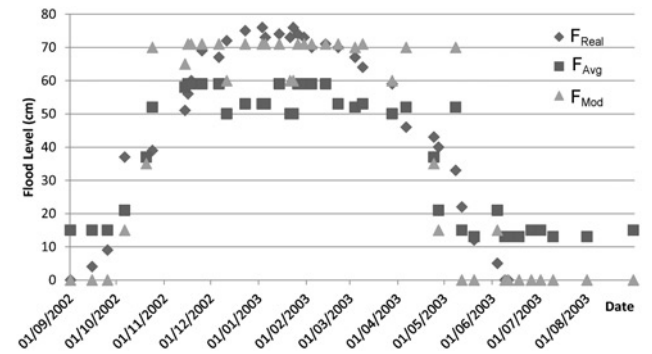


Fig. 4 Estimations of the flooding level for the 2002–2003 year

Table 6 Number of Hops of each node

Node ID	Hops
node 1	0
node 2	1
node 3	2
node 4	0
node 5	1
node 6	2
node 7	0
node 8	1
node 9	0

followed by the message from a node that changes its state to the Base Station, including the number of hops.

To obtain the power consumption of the SOM, the number of cycles required by the microcontroller to execute all the operations is calculated. This information is obtained from the IAR Systems' compiler for MSP430 microcontrollers [26], which are the ones we use in our nodes, enabling the optimisations for fast code execution. IAR offers information, using an emulator, about the number of cycles required to execute a function. With these conditions, the SOM function needs a total of 7324 cycles for the SOM.

Knowing that the TelosB works at 4 MHz, and that the power consumption of the microcontroller in execution is 3 mW [12], the total power consumption to process the algorithm is 5.49 μ J for the proposed 3×4 SOM.

To obtain the power consumption, the simulator takes into account the hibernation mechanism implemented over the ICARO networks, as described in Section 4. This consists of keeping the measurement nodes' radio transceiver on (T_{on}) for 30 s. The simulator makes it possible to adjust the time the radio transceiver is in stand-by mode (T_{off}). These two times determine the period between consecutive transmissions. Thus, the daily energy consumption in a node ($E_{node,i}$) can be obtained with $E_{node,i} = E_{Tx,i} + E_{Rx,i} + E_{StandBy,i}$. Where $E_{Tx,i}$ is the energy required to send a message; $E_{Rx,i}$ is the energy spent with the radio in receive mode; and $E_{StandBy,i}$ is the energy used in the microcontroller when the radio is asleep.

$E_{Tx,i}$ can be obtained with $E_{Tx,i} = T_{Tx} \cdot P_{Tx} \cdot (N_{hops} + 1)$, where T_{Tx} is the time needed to send a message (for this application, 4 ms in the worst case), P_{Tx} is the power consumed in transmission mode (38 mW [12]), N_{hops} represents the number of hops required for a message to reach the Base Station from a source node. For the ICARO network, this information is summarised in Table 6.

$E_{Rx,i}$ can be obtained as $E_{Rx,i} = (T_{on} - T_{Tx}) \cdot P_{Rx}$, where P_{Tx} is the power consumed in reception mode (41 mW [12]).

$E_{StandBy,i}$ can be obtained as $86\,400 \text{ s/day} - (N_{messages} \cdot T_{on})$, where $N_{messages}$ is the number of transmissions made in a day.

$E_{Tx,i}$ can be obtained as $E_{Tx,i} = (T_{on} - T_{Tx}) \cdot P_{Tx}$.

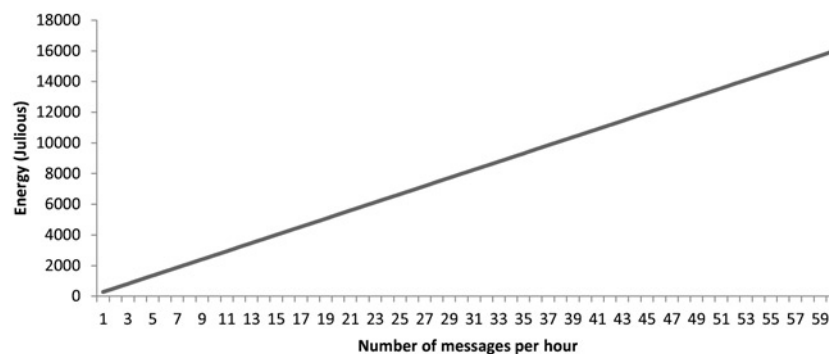
The energy consumption of the network can be obtained by adding together the energy consumption of each node, using the aforementioned equations. The simulator considers all this information and makes it possible to obtain the power consumption of the whole network. It offers a result of energy consumption as a function of the number of messages ($N_{messages}$) transmitted in a day in the [1–60] messages per hour range, with or without the use of the SOM. The information in Fig. 5 represents power consumption without using the SOM. It shows that energy consumption increases practically linearly with the number of message transmissions. These results seem to indicate that most of the energy is spent with the radio transceiver ON (transmit or receive mode). Sending and receiving a message consumes roughly the same amount of power. Furthermore, power consumption with the radio transceiver in OFF is negligible compared with the radio transceiver in transmission or reception. The estimation of power consumption could therefore be simplified, by only considering the time the radio transceiver is in receive mode: $E_{node,i} \cong T_{on} \cdot P_{Rx}$.

$T_{off} \gg T_{on}$ represents a low duty cycle scheme. This is the best situation for low-power consumption systems. This goal is obtained with the use of the SOM.

Table 7 shows the results of energy consumption of the whole network when considering sending information from 1 to 5 times per day using the SOM. These results are obtained with the simulator. The first column indicates the number of retransmissions executed per day. The second column shows the energy consumption. Finally, the third column details the energy saving obtained when the information is sent using the proposed method (SOM), compared with sending every minute all the information acquired by the sensors to the Base Station. Sending a

Table 7 Energy consumption

Number of retransmission	Energy, J	Energy saving, %
1	22.73	99.86
2	33.80	99.79
3	44.86	99.72
4	55.23	99.65
5	66.99	99.58

**Fig. 5** Estimation of the energy consumption of the system without the SOM

message every minute was the situation described before, when the estimation of the flood level was done at the Base Station. The power consumption when the proposed data fusion and aggregation algorithm is used is much lower.

Note that in any case the consumption of the radio transceiver is much higher than the consumption of the SOM algorithm. Clearly, the proposed system drastically reduces power consumption compared with the previous situation, with the classical focus of WSNs using nodes to gathering information at the Base Station, without any preprocessing or treatment of the information. The proposed SOM has shown itself to be a good data fusion and aggregation method for reducing power consumption, enabling an increase in the number of devices, keeping the cost low and reducing the size and weight of the power system.

Therefore, this SOM demonstrates that it represents a good trade-off between computational requisites and accuracy in the flood level estimation.

6 Robustness of the system

After the simulated results, the system was implemented over the ICARO network.

Initially, the measurement accuracy was tested, comparing the quality of the information gathered from the sensors with the information obtained from high accuracy devices of the Doñana Biological Station (EBD). Measurements obtained with the ICARO network were similar to those obtained with the EBD's equipment, several times more expensive than the proposed solution.

Since the start of the installation, partial information has been gathered about the sensors and their reliability. Initially, the system stored for 1 year, all environmental information from the nodes with a refresh frequency of 1 min. During this time, the system has shown its robustness even on days of high rainfall or high temperatures (with temperatures above 45°C).

As an example, Fig. 6 shows the throughput for node 10, understanding throughput as the percentage of successful transmissions. The dashed line depicts the average throughput of the entire network during 1 year.

This node (10) offers an average throughput of 56.57%. The network offers an average throughput up to $R = 56.4\%$. Because of this, to reduce the risk of information loss, the proposed system sends information three times. In this case, we obtain a system reliability (R_s) of:

$$R_s = 1 - \prod_{i=1}^n (1 - R_i) = 1 - (1 - R)^3 = 92\% \quad (3)$$

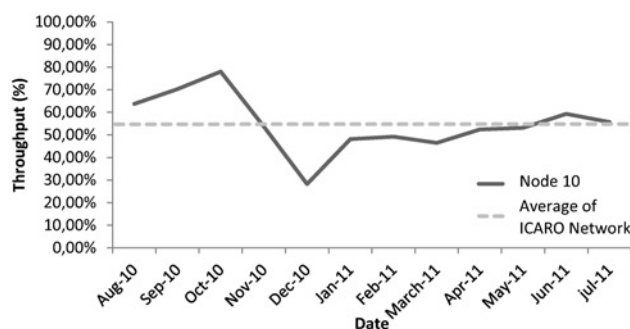


Fig. 6 Throughput of the ICARO network

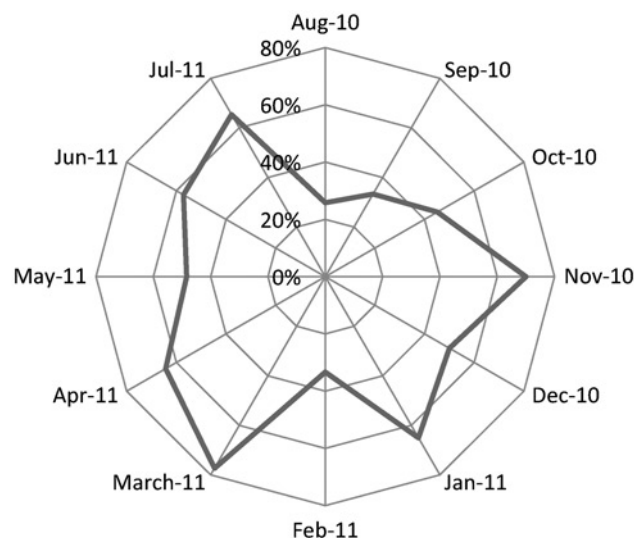


Fig. 7 Changes of the winning neuron per month in the 2010–2011 year

We assume that this value is good enough to ensure that the system will work correctly, even with packet loss.

Fig. 7 shows the frequency of change of the winning neuron over time. On average, data information changes once every 2 days (average of 53%).

On the basis of this information, an additional aggregation can be implemented. As the flood level does not change continuously (neither do the input variables), additional energy would be saved if the system only sent information to the Base Station when the winning neuron changed.

To do this, one needs to define clusterheads. These clusterheads determine whether there are relevant changes in the input variables (i.e. if there are changes in the estimation of the winning neuron). If so, it collects the information of the winning neurons of its neighbours. If not, the radio transceiver is switched off, reducing energy consumption.

Currently, the ICARO network is programmed to send the winning neuron of each node three times a day.

7 Conclusions and future work

WSNs have shown their value as a solution for environmental monitoring, allowing low cost installation with high flexibility, in contrast to wired deployments, such as the traditional installations in the National Park of Doñana.

The main issue with this technology is to obtain a satisfactory trade-off between the amount of transmitted data and energy consumption. For these applications, aggregation and data fusion are good solutions.

The proposed method consists of an example of data fusion and aggregations, based on an SOM applied to a WSN, allowing improvements to the lifetime of the system.

With the help of a simulator, we proved that to increase the network lifetime, the radio transceiver needs to be kept off for as long as possible. On the contrary, it is clear that multi-hop can increase the coverage area although it also drastically increases power consumption. Because of this, the use of data fusion and aggregation, such as SOM networks, represents a good trade-off between power consumption and accuracy, especially in extended networks.

This study does not just present theoretical results. The proposed system has been deployed in a challenging

scenario to overcome the very real problems of measuring in the National Park of Doñana, because of the difficulty of accessing flood zones where measurement has always been done by hand.

With this system a real-time estimation of the flood level can be obtained instead of a few data per year which was the previous situation. The system has worked correctly since its deployment, demonstrating its reliability and robustness.

The authors are currently working on further reducing power consumption, on the basis of new fusion information methods among nodes and considering low-level changes among winning neurons on consecutive days.

8 Acknowledgments

This research has been supported by the ‘Consejería de Innovación, Ciencia y Empresa’, ‘Junta de Andalucía’, Spain, through the excellence project ARTICA (reference number: P07-TIC-02476) and by the ‘Cátedra de Telefónica, Inteligencia en la Red’, Seville, Spain, through the ICARO project. The authors would like to thank the Doñana Biological Station – EBD (Spanish Council for Scientific Research CSIC) and the Doñana Biological Reserve ICTS (Doñana Natural Space, Almonte, Huelva) for their collaboration and support.

9 References

- Chong, C.-Y., Kumar, S.P.: ‘Sensor networks: evolution, opportunities, and challenges’, *Proc. IEEE*, 2003, **91**, (8), pp. 1247–1256
- Akyildiz, I.F., Su, W., Sankarasubramanian, Y., Çayirci, E.: ‘Wireless sensor networks: a survey’ (Computer Networks, Elsevier, 2002, vol. 38), pp. 393–422
- Ruiz-Garcia, L., Lunadei, L., Barreiro, P., Robla, I.: ‘A review of wireless sensor technologies and applications in agriculture and food industry: state of the art and current trends’, *Sensors*, 2009, **9**, (6), pp. 4728–4750
- Machado, R., Ansari, N., Wang, G., Tekinay, S.: ‘Adaptive density control in heterogeneous wireless sensor networks with and without power management’, *IET Commun.*, 2010, **4**, (7), pp. 758–767
- Kohonen, T.: ‘The self-organizing map’, *Proc. IEEE*, 1990, **78**, (9), pp. 1464–1480
- Obaidat, M.S., Misra, S.: ‘Introduction’, in ‘Cooperative Networking’ (Chichester: John Wiley and Sons, Ltd, 2011), pp. 1–5
- Mora, J.M., Molina, F.J., Larios, D.F., Rodriguez, G., Barbancho, J., León, C.: ‘Architecture for environmental data access’. Proc. Int. Conf. on Data Communication Networking. Int. Joint Conf. on e-Business and Telecommunications (ICETE), Spain, 2011, pp. 102–106
- Vijayanand, S., Suresh, R.M.: ‘An overlook on routing techniques in wireless sensor networks’. Int. Conf. on Information and Communication Technology in Electrical Sciences (ICTES), Chennai, Tamil Nadu, India, 2007, vol. 3719, pp. 940–945
- Zabin, F., Misra, S., Woungang, I., Rashvand, H.F., Ma, N.-W., Ahsan Ali, M.: ‘REEP: data-centric, energy-efficient and reliable routing protocol for wireless sensor networks’, *Commun. IET*, 2008, **2**, (8), pp. 995–1008
- Gnawali, O., Fonseca, R., Jamieson, K., Moss, D., Levis, P.: ‘Collection Tree Protocol’. SenSys, Berkeley, CA, USA, 2009
- Deng, H., Zhang, B., Zheng, J.: ‘Rate-constrained uniform data collection in wireless sensor networks’, *Commun. IET*, 2011, **10**, (5), pp. 1343–1350
- Polastre, J., Szewczyk, R., Culler, D.: ‘Telos: Enabling ultra-low power wireless research’. Fourth Int. Symp. on Information Processing in Sensor Networks (IPSN), Los Angeles, CA, April 2005, pp. 364–369
- Yonghe, L., Das, S.K.: ‘Information-intensive wireless sensor networks: potential and challenges’, *IEEE Commun. Mag.*, 2006, **44**, (11), pp. 142–147
- Lin, J., Xiong, N., Vasilakos, A.V., Chen, G., Guo, W.: ‘Evolutionary game-based data aggregation model for wireless sensor networks’, *Commun. IET*, 2011, **5**, (12), pp. 1691–1697
- Esteban, J.a., Starr, A.a., Willetts, R.a., Hannah, P.a., Bryanston-Cross, P.b.: ‘A review of data fusion models and architectures: Towards engineering guidelines’, *Neural Comput. Appl.*, 2005, **14**, (4), pp. 273–281
- Steinberg, A.N., Bowman, C.L., White, F.E.: ‘Time synchronization in sensor networks: A survey’. Proc. of SPIE – Int. Soc. Opt. Eng., 1999, vol. 3719, pp. 430–441
- Rabbat, M., Nowak, R.: ‘Distributed optimization in sensor networks’. Proc. of the 3rd Int. Symp. on Information Sensor Networks (IPSN), Berkeley, CA, USA, April 2004
- Yen, H.-H., Lin, C.-L.: ‘Integrated channel assignment and data aggregation routing problem in wireless sensor networks’, *Commun. IET*, 2009, **3**, (5), pp. 784–793
- Scaglione, A., Servetto, S.D.: ‘On interdependence of routing and data compression in multi-hop sensor networks’. Proc. 8th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking (MobiCom 02), Atlanta, Georgia, 2002
- Huang, Q., Liu, C., Guo, C.: ‘Reliable aggregation routing for wireless sensor networks based on game theory’, Int. Conf. on Innovative Technologies, Prague, Czech Republic, InTech 2010. ISBN 978-953-307-132-9
- Guo, Y., Hong, F., Guo, Z., Jin, Z., Feng, Y.: ‘EDA: Event-oriented data aggregation in sensor networks’. IEEE 28th Int. Performance Conf. on Computing and Communications (IPCCC), December 2009, ISSN 1097-2641, pp. 25–32
- Barbancho, J., León, C., Molina, F.J., Barbancho, A.: ‘A new QoS routing algorithm based on self-organizing maps for wireless sensor networks’, *Telecommun. Syst.*, 2007, **36**, (1–3), pp. 73–83
- Ye, W., Heidemann, J., Estrin, D.: ‘Medium access control With coordinated adaptive sleeping for wireless sensor networks’, *IEEE/ACM Transactions on Networking*, 2004, **12**, (3)
- Ma, L., Xing, T., Wang, X., Song, J.: ‘IEEE 802.11b-based access scheme with coordinated sleeping for wireless sensor networks’. Twentieth Int. Symp. on Personal, Indoor and Mobile Radio Communications, Tokyo, Japan, pp. 2106–2110
- Sivrikaya, F.a., Yener, B.b.: ‘Time synchronization in sensor networks: a survey’, *IEEE Network*, 2004, **18**, (4), pp. 45–50
- <http://www.iar.com/en/Products/IAR-Embedded-Workbench/TI-MSP430>, accessed October 2011