



HAL
open science

Using dynamic programming to solve the Wireless Sensor Network Configuration Problem

Ada Gogu, Dritan Nace, Enrico Natalizio, Yacine Challal

► **To cite this version:**

Ada Gogu, Dritan Nace, Enrico Natalizio, Yacine Challal. Using dynamic programming to solve the Wireless Sensor Network Configuration Problem. *Journal of Network and Computer Applications* (JNCA), 2017, 83, pp.140-154. 10.1016/j.jnca.2017.01.022 . hal-01469355

HAL Id: hal-01469355

<https://hal.science/hal-01469355>

Submitted on 21 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Using dynamic programming to solve the Wireless Sensor Network Configuration Problem

Ada Gogu, Dritan Nace, Enrico Natalizio and Yacine Challal

Abstract—This work studies the problem of network configuration for Wireless Sensor Networks (WSN), consisting of two interdependent problems: sensor placement and topology control, by taking into consideration both the traffic pattern and the transmission range assignment. The design objectives are i) reducing the overall energy consumption and ii) ensuring node energy consumption fairness between the sensors. First, the problem of placing the sensors in the optimal positions is studied and then a power control scheme is put in place to manage the topology of the network. For both the two sub-problems, the linear network case is treated and then extended to the two-dimensional case. The two sub-problems are treated within a unifying mathematical framework based on dynamic programming, in order to guarantee the optimality of the solution. The method can easily be adapted to solve the problem for discrete values of transmission range. The method presented in this work shows a low computational complexity in comparison to other methods, and, due to its implementation simplicity, it will be of great help to network designers in the planning phase of WSN deployment.

Index Terms—WSN, Dynamic programming, Sensor Placement, Network configuration;

I. INTRODUCTION

Wireless Sensor Networks (WSN) is a disruptive technology with potentially a wide range spectrum of applications such as remote health surveillance, precision agriculture, air and water pollution detection and containment, home automation, supply chain monitoring, etc. [?]. WSN development is the culmination of recent advances in microelectromechanical systems (MEMS) and wireless communication technologies. A typical WSN is composed of a set of tiny motes that sense the environment and transmit sensed information, hop-by-hop, to a processing workstation through the Base Station (BS). A simple WSN is presented in Fig. 1. Each mote comprises sensors, a processing unit, memory, radio transceiver and a battery.

Faced with stringent constraints that affect energy, bandwidth and memory use among others, WSN

Ada Gogu is with Polytechnic University of Tirana, Mother Teresa Square, 4, Tirana, Albania. D. Nace, E. Natalizio and Y. Challal are with the Heudiasyc Laboratory (UMR CNRS 7253), Université de Technologie de Compiègne, Compiègne 60205, France. Y. Challal is also with Laboratoire de Méthodes de Conception de Systèmes at Ecole nationale Supérieure d Informatique, Algeria E-mail: ada.goguhds.utc.fr, dnacehds.utc.fr, enataliziohds.utc.fr, ychallalhds.utc.fr

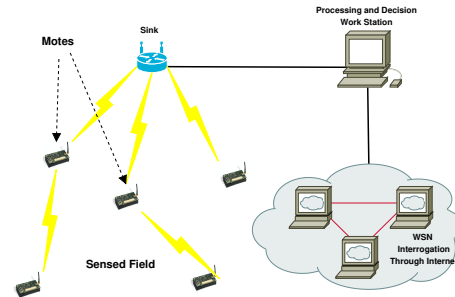


Figure 1: Wireless sensor network

technology needs to be carefully managed in order to meet the requirements of applications. Optimization techniques and strategies are applied at the physical, access control, network and application layers to improve their performance. A primary concern in WSN is the energy constraint. A carefully designed network can be a very effective means to conserve energy and therefore extend the lifetime of the network.

In this work, we focus on the network configuration problem, which consists in assessing two main aspects of WSN:

- 1) **Transmission structure:** The transmission structure determines the organization between the sensors such that their data flows may reach the Base Station (BS) through a many-to-one traffic paradigm. For an energy constrained technology, the design of a data transmission structure is strategic. The design process can be even more difficult if other application-dependent issues like data aggregation need to be taken into account.
- 2) **Deployment and range assignment:** A many-to-one traffic pattern tends to overuse the nodes closest to the BS for relaying the traffic generated by other nodes, thus causing an unbalanced distribution of the traffic load throughout the network known as the energy hole problem. As remarked in different works [? ? ?] these nodes will be more prone to failures, and their failure may culminate in disconnections, network partitioning, and even complete paralysis of the network.

Two possible ways of overcoming the issues

raised by the mentioned aspects are¹:

- Energy-aware deployment. This aims at finding a deployment scheme that distributes load fairly over all the sensors in the network.
- Transmission range assignment. An effective transmission range assignment will balance the energy consumption between the sensors in the network. Obviously, the transmission range assignment should take into account the corresponding sensor load. Moreover, the transmission range is closely related to the distance from the BS which comes to take into consideration the number of hops needed to reach it. Trade-offs between the load distribution and the number of hops complicate the problem of transmission range assignment under energy constraints.

In this study, we focus on the design of optimal strategies in terms of a given objective (i.e.: energy efficiency) for WSN. We examine two sub-problems, which are derived by the aforementioned main aspects of the network configuration problem, namely: i) sensor placement and ii) topology control based on power control. Both problems seek to extend the lifetime of the network by finding the best combination between a feasible transmission structure and a particular scheme (a deployment scheme in the case of the sensor placement problem and a range assignment scheme in the case of the topology control problem). Nonetheless, the context of these two problems, related to the sensor deployment assumption, is not the same. Sensor placement might be seen as the *pre*-deployment energy-aware network configuration problem, whereas topology control corresponds to the *post*-deployment energy-aware network configuration problem. For each of the problems we begin our investigation with a simple case, i.e. a linear network, and then extend it to the two-dimensional network case.

The contribution of this paper is twofold and can be summarized as follows:

- We solve the sensor placement problem and the topology control one at the optimality. Furthermore, we provide a complete study as we consider several scenarios involving traffic aggregation and equitable expenditure of energy, and provide comparisons between discrete and continuous power assignment scenarios.
- We propose a unifying mathematical framework to deal with the two sub-problems, which is based on dynamic programming. Our approach has a low complexity compared with nonlinear programming approaches proposed in the literature, and we believe it provides a powerful tool to deal with the high combinatorics of such problems.

The rest of the paper is organized as follows.

Section II presents a brief state of the art for different variants of the problem and position this work in the body of existing literature. In Section III, we examine the sensor placement problem and propose dynamic programming approaches for both linear and two-dimensional networks. Section IV is devoted to the network configuration problem. Apart the solution approaches for several scenarios, we provide comparative computational tests. Finally, section V concludes the paper.

II. PROBLEM DESCRIPTION AND RELATED WORK

This section gives an overview of works related to the network configuration problem. Specifically, we investigate the state of the art of the two sub-problems treated in this paper: sensor placement and topology control based on power control. In literature, only one work has investigated these two sub-problems within a unifying framework [?]. However, the authors of the mentioned work propose heuristic solutions for both the sub-problems, and they validate the results through simulations, without ensuring the optimality of their solution. Our contribution aims at building an analysis framework based on a more rigorous mathematical approach. As, at the best of our knowledge, no other works have investigated the two sub-problems by using a unifying solution methodology, we will present the literature split into two parts.

A. Sensor placement

In WSN literature the sensor placement problem may group different optimization problems depending on the constraints and objective function. For an extended state of art of the sensor placement problem, the reader is referred to [?]. Some formulations of this problem aims at deterministically placing the sensors in order to meet some requirement such as minimizing energy consumption [? ?], fault tolerance [?], or maximizing lifetime under physical attacks [?]. Other works [? ?] consider dynamic environments in which the objectives and tasks can change over the time and the sensor placement is reconfigured frequently. In [? ?] the authors determine the optimal placement of sensor nodes along the straight line between a source and a destination node. They model the energy consumed for packet transmission in a mono-directional/bi-directional flow between source and destination within a mathematical framework using non-linear programming. The proposed mathematical model allows the authors to find the optimal positions for the nodes in a closed formula. In this work, we use a more general framework for the sensor placement problem, which generalize the mentioned works, as we also consider traffic aggregation at the intermediate relays. The problem discussed in [?] consists in deciding how to place N sensors in a linear network topology, such that all the sensors expend the same amount of energy. This

¹Load balancing at the maintenance phase is beyond the scope of this work.

Table I: Related works

Scheme	Sensor placement	Topology control	Objective(s)	Methodology
[?]	●	○	Energy consumption	Multi-variable nonlinear programming
[? ?]	●	○	Energy consumption	Nonlinear programming
[?]	●	○	Energy consumption	Dynamic Programming
[?]	●	○	Fault tolerance in terms of connectivity	Heuristic based on Steiner Tree (STP-MSP)
[?]	●	○	Resistance to physical attacks	Heuristic based on binary search
[?]	●	○	Dynamic	Distributed heuristic
[?]	●	○	Dynamic	Heuristic based on Dijkstra
[?]	●	○	Multiobjective	Genetic algorithm
[?]	○	●	Multiobjective	Genetic algorithm
[?]	○	●	Load balancing	Distributed annealing algorithm
[?]	○	●	Energy consumption	Heuristic based on spanning tree
[?]	○	●	Energy consumption	Analytical method
[?]	○	●	Energy consumption	Heuristic based on clustering&energy harvesting
[?]	○	●	Coverage&Energy consumption	Heuristics based on clustering
[?]	●	●	Energy consumption	Heuristic based on virtual tree

work assumes a linear transmission structure where each node has to relay all the collected and generated data towards the immediate one-hop neighbor. The problem is formulated as a multi-variable nonlinear programming problem and solved to optimality, but without considering the idle and receiving energy. A more general formulation is given by [?], where the problem is to find the optimal number of sensors and their respective positions. The optimization problem is solved in two steps. First, the sensor position problem for a given number of sensors N is formulated as a nonlinear programming problem similar to that of [?]. In the second step, the goal is to optimize the number of sensors such that the ratio of network lifetime to the number of sensors N is maximized. Notice that the network lifetime is highly dependent on the number of sensors N calculated in the first step. This ratio represents a new metric for evaluating trade-offs between network lifetime and the cost of sensors. In contrast to sensor placement, the density control problem seeks to distribute load uniformly over the monitoring area so that traffic per node is balanced over the whole area. Density strategies [? ?] are defined as part of the pre-deployment phase and provide a non-uniform random deployment in which the density increases closer to the *BS*. In this study we do not consider this problem but the density changes may be extrapolated from the results of the sensor placement problem. Another solution approach used in literature for the sensor placement problem is the multi-objective optimization. In fact, the sensor placement can serve simultaneously to reduce the energy consumption, increase the connectivity and the coverage. In [?], the authors propose a multi-objective evolutionary algorithm that uses a decomposition approach for converting the problem of approximation of the Pareto fronts (PF) into a number of single-objective optimization problems. Even if the multi-objective approach seems promising, in this work, we focus on the energy efficiency of the WSN, as our main contribution is the unified solution methodology for

the two sub-problems. However, our work could be easily extended to treat multiple objectives.

B. Topology control based on power control

The topology control problem is a very important problem in WSN as it includes several sub-problems, which have been grouped in literature according to their main objective and to the approach used to reach this objective. Specifically, In [?] and in [?] the authors classify topology control schemes according to different attainable objectives: network coverage, network connectivity, network lifetime extension. For each of these categories they present protocols and techniques with a focus on blanket coverage, barrier coverage, sweep coverage, power management, and power control. According to the classification proposed in these papers, our work would fall in the category that concerns power control approaches for network lifetime extension. Originally, this approach has been proposed by Santi et al [?]. In this paper, we take into account only the topology control based on power control, and we redefine the problem as it follows. We divide the network into coronas or grids of different lengths such that the energy is balanced in each of them. This problem assumes a uniform node distribution and hence may be defined in a post-deployment phase. To take account of the energy hole problem, a corona-based model is usually employed, the reader is referred to [?] for an extensive survey on corona-based deployment strategies. In this model the network area is assumed to be circular, and it is divided into concentric circles forming coronas. Just like in a linear network topology, nodes belonging to a corona will forward the data generated by the nodes themselves together with data generated by higher-level coronas. In the case of linear or rectangular networks, a grid is generally employed, corresponding to a unit division of the network. The network configuration problem is discussed in [?], which proposes a method for dividing the network into coronas, then the coronas into subcoronas and finally the subcoronas into zones

such that the load in every zone is balanced. In this model, there is always a mapping between the sensors of a given zone belonging to a corona, and the sensors of another zone, belonging to an upstream corona. The problem of finding the optimal number of coronas is modeled as an optimization problem and a simulated annealing algorithm is proposed. The problem of dividing the coronas into subcoronas and zones is solved iteratively. This network division scheme assumes that the coronas and the subcoronas have the same width and that the number of zones is the same for each of them. In [?], on the other hand, the number of coronas and their respective lengths are assumed to be known, and the transmission range must be assigned for each corona so as to maximize network lifetime. The transmission range will actually determine the next hop corona. The algorithm proposed for this problem builds a graph in which the nodes represent the coronas and the edges represent the feasible wireless communication links between any two coronas. The algorithm builds a set of spanning trees for each node, and those having the best performances in terms of network lifetime are selected. Among the spanning trees that include the most distant corona and the BS, the one with the longest lifetime is retained as the solution to the problem. In [?] the authors discuss the same problem, however their transmission structure is different than ours. According to [?] sensors in a corona transmit their data to any of sensors in the downstream corona. In our model, the sensors are grouped in clusters and it is only the cluster head that transmit the data. By considering only the transmitting energy, in [?], the authors prove that the total energy consumption minimization is achieved when the coronas have the same length. They also propose an iterative algorithm to consider the uneven energy depletion of energy and avoid the energy holes. Even for the topology control based on power control there are some works that deal with multiple objectives. For example, in [?] the authors assume eight predefined transmission range levels for each sensor, and propose a genetic algorithm for optimizing coverage, connectivity and lifetime of the sensor network. As already mentioned for the sensor placement sub-problem, our approach focuses on a single objective in order to offer a more performing solution.

In order to simplify the further reading of the paper, we present below the table of notations.

III. SENSOR PLACEMENT PROBLEM

In this section the goal is to find the network dimensioning where the sensors consume the least energy in transmitting their data to the destination. **Given a number of source nodes at a distance d from the BS, the sensor placement problem is to determine the number of sensors and their respective positions from the BS such that the total energy**

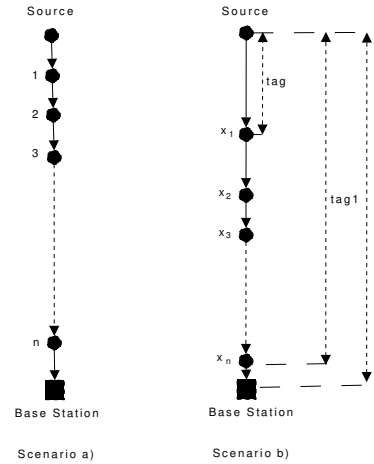


Figure 2: The multihop transmission for linear network

consumption is minimized. In Section III-A we solve the problem for a linear topology. This type of topology has applications in border surveillance, highway traffic monitoring and oil pipelines. Subsequently, in Section III-B we generalize the problem to the two dimensional network in which a given number of source nodes are placed in the surveying area.

A. Linear network

Here we assume that the sensors will be deployed in a linear network as in Fig. (2).

The energy model that we use to estimate the energy consumed by them is proposed in [?]. E_{TX} denotes the energy used for transmitting and E_{RX} the energy used for receiving, as in equation (1).

$$\begin{cases} E_{TX} = (E_{elec} + E_{amp}d^\gamma) \cdot B \\ E_{RX} = E_{rec} \cdot B \end{cases} \quad (1)$$

In these equation, d corresponds to the distance of communication. However, practically the sensor range of communication is upper limited by Tx_{max} . Besides the energy spent for receiving and transmitting the data, sensors consume energy also in the idle mode. At this stage, we assume that the MAC protocols guarantee perfect synchronization and successful transmissions between sensors, therefore we do not take the idle energy into consideration. For this problem we examine two scenarios: (a) the source sensor node has to send data to the BS and all the intermediate sensor nodes act simply as relaying nodes and (b) the intermediate sensors add their own information before relaying. Fig. 2 illustrates these two cases for the linear network. The solution for scenario (a) places the sensors equidistantly (see [?]). While obtaining an optimal solution for this first scenario is fairly straightforward, solving the second is more complex. The problem corresponding to scenario (b) is formulated as follows:

Table II: Table of notations

Type	Parameter	Description
Network	d	Network length.
	R	Network radius.
	x	Network length from the source.
	$k_i = \frac{x_i}{d}$	Ratio between the sensors' distances.
	n	Number of hops, deployment problem.
		Number of cells, network configuration problem.
	γ_0	The angle between two nodes in the uppermost circle.
	r	Ratio between the nodes in two consecutive circles.
	l_i	Transmission range for nodes in the i_{th} circle.
Application	N_i	Number of nodes placed in the i_{th} circle, deployment problem.
		Number of zones in the i_{th} corona, network configuration problem.
	β	Data generation rate in bps
	α	Data traffic per node in Erlang
	ϵ	A very small constant
	m	Compression ratio
	c	Coefficient $c = \alpha \cdot \beta$
	$f_n(d)$	The total energy function for a network with a length (d) and n hops.
Energy	$\delta y = 1$	Distance discretion
	E_{elec}	Energy/bit consumed by the transmitter electronics.
	E_{amp}	Energy/bit consumed by the amplifier.
	E_{rec}	Energy/bit consumption of the receiving circuitry
	E_{idle}	Energy consumption during the idle mode.
	B	Number of bits
	Tx_{max}	Maximal transmission range
	γ	Path loss exponent [2 – 6]
	$L = \{l_1, l_2, \dots, l_m\}$	Discrete distance transmission of a sensor
	$E = \{e_1, e_2, \dots, e_m\}$	Discrete energy transmission of a sensor

Problem Definition P1.

Input : A source node at distance d from the BS has to transmit its own information to the BS across n intermediate nodes such that:

- every intermediate node (sensor) transmits to the downstream neighbor its own information as well as the information received from the upstream neighbor;
- the information is not aggregated.

Objective : Minimize total energy consumption.

Output : The number n of intermediate nodes and their respective distances d_i $1 \leq i \leq n$ from the source node.

We remark that the total energy consumption function is convex (the proof is omitted on this paper), therefore it exists a global minimum of energy. In contrast to the first scenario, in this problem each intermediate node adds a given amount of traffic

(see Fig. 2 scenario (b)). Two parameters have to be calculated, namely the number of relaying sensor nodes and the distances between them, in order to attain the minimum of the associated energy function. The difficulty and the combinatorics of this problem lie in the interdependence of two terms: the number of intermediate nodes and the respective distances between them. Nonetheless, there is a nice property regarding these parameters.

Let us suppose that x_n is the optimal location of the last relay node in a linear network of length d where n relay nodes are employed. Given the energy formula (1), it can be shown that the value $k_n = x_n/d$ does not depend on the distance d . The following proposition holds:

Proposition 1. k_i values depend only on the number n and not on the distance d .

Proof: This result can be proved by mathematical

induction on the number n . Let us take $n = 1$. Clearly, the energy function associated with the transmission through an intermediate sensor placed at x_1 is at a minimum when the derivative is 0. The energy function is of the form

$$\alpha_1(x_1)^\gamma + \alpha_2(d - x_1)^\gamma + \zeta = \alpha_1 d^\gamma (k_1)^\gamma + \alpha_2 d^\gamma (1 - k_1)^\gamma + \zeta \quad (2)$$

where α , β and ζ are constants. The derivative gives $\alpha_1 d^\gamma \gamma (k_1)^{\gamma-1} - \alpha_2 d^\gamma \gamma (1 - k_1)^{\gamma-1}$. Simple calculations show that this function attains 0 for some $k_1 = \frac{1}{(\alpha_1/\alpha_2)^{1-\gamma} + 1}$ that depends only on the values of α_1 , α_2 and γ . The same reasoning can be applied to any higher n , provided that the result holds for $n - 1$. Then, from the recurrence assumption we can deduce that also for some n the energy function is similar to the energy function given in (2) (apart from using other constants). Let explain this in details below. We suppose that n sensors are placed between the source and the BS and x_i give the distance of sensor i from source. Then, the energy formula has the following form:

$$\begin{aligned} & \alpha_1(x_1)^\gamma + \alpha_2(x_2 - x_1)^\gamma + \alpha_3(x_3 - x_2)^\gamma \\ & + \dots + \alpha_{n-1}(x_{n-1} - x_{n-2})^\gamma \\ & + \alpha_n(x_n - x_{n-1})^\gamma + \beta_n(d - x_n)^\gamma + \zeta_n \end{aligned} \quad (3)$$

As we are looking for minimum energy formula, the placement of x_1 to x_{n-1} is done optimally and the recurrence hypothesis applies. Hence, we have $x_{n-1} = k_{n-1} \cdot x$, $x_{n-2} = k_{n-2} \cdot k_{n-1} \cdot x$, ..., $x_1 = k_1 \cdot k_2 \cdot \dots \cdot k_{n-2} \cdot k_{n-1} \cdot x$. Replacing x_i in formula (3), we obtain an energy formula similar to (2):

$$\alpha'(x_n)^\gamma + \beta'(d - x_n)^\gamma + \zeta_n = \alpha' d^\gamma (k_n)^\gamma + \beta' d^\gamma (1 - k_n)^\gamma + \zeta_n \quad (4)$$

Then we can therefore apply the same schema as for $n = 1$ which ends the proof. ■

We will consider in the following the case with $\gamma = 2$. The objective function representing the overall energy consumption of the network is given by equation (5). It is obtained by summing the energy expended by all sensors according to equation (1). Regarding the sensors, we assume that n sensors are placed respectively at distances d_1, d_2, \dots, d_n and each of them generates one unit traffic.

$$E_{(n,d_1,d_2,\dots,d_n)} = (E_{elec} + E_{amp}d_1^2) + E_{rec} + 2(E_{elec} + E_{amp}d_2^2) + \dots + nE_{rec} + (n+1)(E_{elec} + E_{amp}d_n^2) \quad (5)$$

We solve this problem using a dynamic programming approach. Let us first briefly recall the principle of Dynamic Programming (DP). DP is a sequential approach, proposed by Bellman [?] for optimizing a given objective function. The problem is thus broken down into stages and the aim at every stage is to select the optimal decision so that the objective is optimized over the current number of stages. Hence, in each stage we solve only once the corresponding

subproblem. The results of each stage are stored and later used to backtrack the optimal values. The most typical example of a DP implementation is the shortest path. If the path ($A - > B - > C$) is the shortest path between the points A and C that passes through B, then AB is evidently the shortest path between the points A and B.

More particularly, our problem is separated into stages determined by the number of intermediate sensor nodes (or by the number of hops). At some stage n we determine the minimum energy necessary for transmitting the information if n sensor nodes are placed between the source node and the base station, using the energy values obtained from the previous stage. The optimal solution for an instance of a problem with n hops and a given distance d can be seen as equivalent to the optimal solution for some distance $x < d$ and $n - 1$ hops, plus a final hop from this point (x) to the BS. Let $f_n(d)$ denote the minimum energy for transmitting from the source node to the BS where the information passes through n intermediate relaying nodes that add their own information to the flow (Figure 2 scenario (b)). Applying this principle to our problem we obtain the following energy recurrence formula:

$$f_n(d) = \min_{\substack{0 \leq x \leq d \\ (d-x) \leq Tx_{max}}} \left\{ f_{n-1}(x) + (n-1)E_{rec} + n \cdot (E_{elec} + E_{amp}(d-x)^2) \right\} \quad (6)$$

where $f_{n-1}(x)$ is the minimum energy used over a distance x from the source, passing through $n - 1$ intermediate sensors, and the remainder of the formula represents the energy added by the n^{th} sensor placed at x . We note that at each step n we only need to know the minimum value for the energy function $f_{n-1}(x), 0 \leq x \leq d$. Moreover, the distance d should be updated ($d = d - Tx_{max}$) if the distance $(d - x)$ becomes bigger than Tx_{max} . In the following, we will analyze the problem for a distance d and assuming that in each step the the transmission range will not overpass the maximal transmission range. Hence, we start with $\{f_1(x) | 0 \leq x \leq d\}$, and build the energy function recursively for the upper levels. In our case the problem can be solved analytically. We begin by expressing the function $f_{n-1}(x)$ included in the right-hand term of equation (6) through $f_{n-2}(x)$, and so on. Then, we replace $f_{n-1}(x)$ by the obtained recursive expression in (6) and derive the right-hand term, which allows us to find $f_n(d)$. Let us assume that we have $0 \leq x_{n-1} \leq x$ such that:

$$\begin{aligned} f_{n-1}(x) = & f_{n-2}(x_{n-1}) + n \cdot E_{amp}(x - x_{n-1})^2 \\ & + nE_{elec} + (n-1)E_{rec}. \end{aligned}$$

Let k_{n-1} denote $\frac{x_{n-1}}{x}$. We have:

$$f_n(d) = \min_{0 \leq x \leq d} \{ f_{n-2}(x \cdot k_{n-1}) + nE_{amp}(1 - k_{n-1})^2 x^2 + nE_{elec} + (n-1)E_{rec} + (n+1)E_{amp}(d-x)^2 + (n+1)E_{elec} + nE_{rec} \}.$$

By developing further we obtain the following recursive formula for $f_n(d)$:

$$f_n(d) = \min_{0 \leq x \leq d} \{ \sum_{l=0}^{n-1} ((l+1)E_{amp}(1 - k_l)^2 \cdot \prod_{p=l+1}^{n-1} k_p^2 \cdot x^2) + (n+1)E_{amp}(d-x)^2 + \sum_{k=1}^n ((k+1)E_{elec} + kE_{rec}) \}. \quad (7)$$

Considering the derivative function of (7) with respect to x and using the result of Proposition 1, which proves the independence of k_p with respect to x , we obtain:

$$\left(\sum_{l=0}^{n-1} (l+1) \cdot (1 - k_l)^2 \prod_{p=l+1}^{n-1} k_p^2 \right) \cdot x + (n+1)x = (n+1)d. \quad (8)$$

From (8) we find that the minimum of the right hand term of (7) is attained for the value of $x = k_n \cdot d$ where

$$k_n = \frac{n+1}{\sum_{l=0}^{n-1} (l+1) \cdot (1 - k_l)^2 (\prod_{p=l+1}^{n-1} k_p^2) + n+1} \quad (9)$$

Hence, we deduce:

$$\begin{aligned} f_n(d) &= \sum_{l=0}^{n-1} (E_{amp}(l+1)(1 - k_l)^2 \cdot \prod_{p=l+1}^{n-1} k_p^2 \cdot (k_n d)^2) \\ &\quad + (n+1)(1 - k_n)^2 d^2 + E_{elec}(n+1)(n+2)/2 \\ &\quad + E_{rec}n(n+1)/2 \\ &= E_{amp}d^2 \sum_{l=0}^{n-1} (l+1)(1 - k_l)^2 (\prod_{p=l+1}^{n-1} k_p^2) \\ &\quad + E_{elec}(n+1)(n+2)/2 + E_{rec}n(n+1)/2. \end{aligned} \quad (10)$$

This formula generalizes the work of [?] aimed at finding the most energy efficient position for a sensor. Values k_p for any p do not depend on the distance d (Prop. 1) and can be computed beforehand like in formula (11).

$$k_p = \frac{j+1}{\sum_{l=0}^{j-1} (l+1) \cdot (1 - k_l)^2 (\prod_{j=l+1}^{j-1} k_j^2) + j+1} \quad (11)$$

For scenario (b), these values are $k_0 = 0$, $k_1 = 2/3 = 0.66$, $k_2 = 9/11 = 0.82$, $k_3 = 0.88$, and so on. Then, given the distance d between the source node and the BS we can compute the distance from the source node for each intermediate node (see also Fig. 1, scenario (b)). Finally, from Formula (10), we deduce the minimum amount of energy used for problem P_1 . Given the above, the following proposition holds:

Proposition 2. For any distance d and number of intermediate nodes n for problem P_1 , the total energy consumption is minimized when each intermediate node i is placed at distance x_i , calculated by formula $x_i = \prod_{p=i}^n k_p d$, with coefficients k_p computed according to (11).

We simply need to calculate $f_k(d)$ iteratively for $k > 0$ and stop for n such that $f_{n+1}(d) \geq f_n(d)$.

In Fig. 3 we show the behavior of the energy function for $d = 100$ and $250m$ when the number of intermediate sensors is made to vary. In this work, the numerical results are calculated using Matlab 7.11.0. In Fig. 4 the

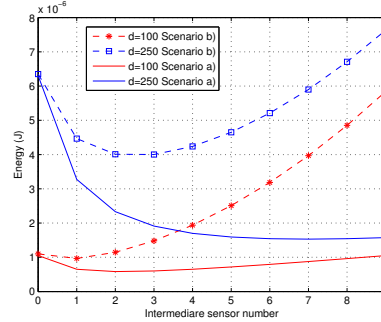


Figure 3: Energy versus the number of relay sensors for $d = 100$ and $d = 250$ meters

graphics present the optimal number of sensors versus distance for the two scenarios (a) and (b). The number

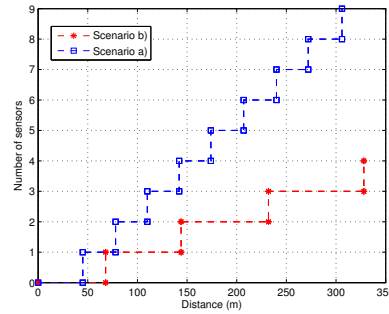


Figure 4: Optimal number of sensors

of hops obtained for scenario (a) is larger than for (b). This is because each additional hop for scenario (b) increases the traffic to be transmitted and hence the energy function. In other words, in contrast to scenario (a), decreasing the number of hops for scenario (b) may lead to energy conservation at a certain level. The main interest of scenario (b) is that for minimum energy consumption and a given number of hops n we know the distances between the relaying nodes exactly, as shown in Fig. 1.b). Furthermore, the distances from the base station are proportional to a constant sequence of n values determined by k_1, k_2, \dots, k_n .

B. Two-dimensional network

In this section, we extend the study to a two-dimensional network. Here, we assume that the nodes will be uniformly placed in circles. The general problem is defined as follows:

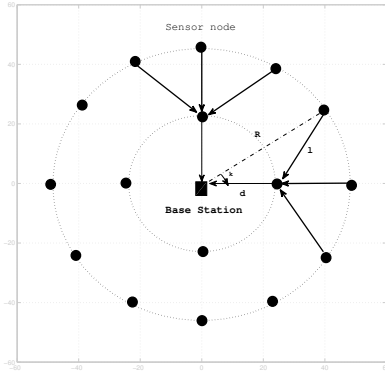


Figure 5: Sensor placement for the case with traffic aggregation

Problem Definition P2.

Input : We have N_0 sensor nodes uniformly deployed according to a circle with radius R from the BS, which need to transmit their information to the BS. We assume that:

- the information is transmitted through several relaying nodes placed on intermediate circles. Each intermediate node transmits the traffic received from the upstream neighbor;
- all sensors placed in the same circle will transmit with the same transmission range;

Objective : Minimize the total energy consumption.

Output : The distance d_i from the BS, for all intermediate circles i and the number of nodes in each circle.

For the above problem, we use an aggregation model proposed in [?], which has the form $y = mx + c$ where x and y are the node input and output information quantities respectively, m varies in $[0, 1]$ and c is a constant. For the sake of simplicity, we assume that each node will generate an unit traffic.

The total energy consumption for a scheme with N_0 sensor nodes placed in the outer circle and N_1 other nodes placed in an intermediate circle is given in equation (12).

$$E = E_{outer} + E_{inner} \quad \text{with :} \quad (12)$$

$$E_{outer} = N_0 (E_{elec} + E_{amp} \cdot l^2)$$

$$E_{inner} = N_0 E_{rec} + N_1 (E_{elec} + E_{amp} d^2)$$

where l is the transmission range of the sensors belonging to the outer circle (path loss exponent is equal to 2). Hence, E_{outer} is the energy expended by N_0 nodes of the outer circle, and E_{inner} is the energy expended by $N_1 = \frac{N_0}{r}$ nodes placed in the inner circle. To calculate the transmission range l (Fig. 5) we apply the cosine formula (13) on the triangle delimited by sides of length R , d and l :

$$l^2 = \left(d^2 + R^2 - 2Rd \cos\left(\frac{r-1}{2}\gamma_0\right) \right) \quad (13)$$

in which γ_0 gives the angle between two successive sensors in the outer circle and the BS (i.e. $\gamma_0 = (2 \cdot \pi)/N_0$). As it can be observed, we have r nodes of the outer circle that will transmit to a node in the inner one. This is why the variable r has to take integer values. The equation (13) is true for every even or odd value of r . Moreover, as we assume that the transmission range of the nodes in the outer circle is the same for all the nodes, the distance l is computed as the maximal distance between a sensor in the outer circle and his respective receiver in the inner circle.

Using the same reasoning as in the linear case, and considering the energy function in (12), we find that the minimum energy consumption is attained for some r such that:

$$r^2 \sin\left(\frac{r-1}{2}\gamma_0\right) = \frac{E_{elec} + E_{amp}d^2}{dRE_{amp}\gamma_0}.$$

Unlike the linear network case, the problem with traffic aggregation is difficult to solve analytically by the DP method. We opt for a method based on parameter discretization of the distance and the number of sensor nodes associated with the distance from the BS respectively, and apply the DP principle. Hence, for a given number of sensors N placed uniformly at distance d from the base station, with n intermediate levels up to the BS, we compute the energy function using the following recursive formula:

$$E_{d,N}^n = \min_{x \leq d, P \leq N} \left\{ \left(\frac{m \cdot N}{P} + 1\right) \cdot E_{x,P}^{n-1} + NE_{elec} + NE_{rec} + NE_{amp}(d^2 + x^2 - 2dx \cos\left(\frac{N-1}{2}\gamma_0\right)) \right\} \quad (14)$$

Here the variables x and P correspond respectively to the distance and the the number of sensors in the level $n - 1$. Based on the DP algorithm, the energy of the $n - 1$ level, $E_{x,P}^{n-1}$, is optimal. We add the energy added by the n th hop which consists of the transmitting and receiving energy of N sensors placed at the distance d to P sensors in the distance x . The coefficient m determines the aggregation data ratio, where the two extreme cases are with aggregation ($m = 0$) and without aggregation ($m = 1$). In each step (the step is determined by the number of hops) we build a matrix containing the energy values for an interval of distances $x \leq d$ and number of sensor nodes $P \leq N$. Hence, we begin by computing the first matrix E^1 containing values of $\{E_{d,N}^1 | d \leq R, N \leq N_0\}$. This corresponds to the case when nodes transmit directly to the BS. In the next step, from the recursive formula (14) for which $n = 2$ (that is, a single intermediate circle), we calculate all $\{E_{d,N}^2 | d \leq R, N \leq N_0\}$ values. We continue like this with $\{E_{d,N}^n | d \leq R, N \leq N_0\}$ for subsequent values of parameter n . We stop the calculations when we reach some n such that $E_{R,N_0}^{n-1} \leq E_{R,N_0}^n$ as shown in Algorithm 1.

At the end of the computations we obtain the number of intermediate levels (n), or the number

Algorithm 1: 2D network

Input: The number of nodes N_0 , Radius R , Max transmission range Tx_{max} ;

Output: Number of intermediate circles n , Minimal energy E ;

Initialization : $\{E_{d,N}^1 | d \leq R, N \leq N_0\}$, $n=1$;

repeat

 Calculate $\{E_{d,N}^{n+1} | d \leq R, N \leq N_0\}$ matrix:

$$E_{d,N}^{n+1} = \min_{x \leq d, p \leq N} \left\{ \left(\frac{m-N}{p} + 1 \right) \cdot E_{x,p}^n + N E_{elec} + N E_{rec} + N E_{amp} (d^2 + x^2 - 2dx \cos(\frac{N}{2} \gamma_0)) \right\}$$

 Calculate the transmission range l_k according to 15;

if ($l_k > Tx_{max}$) **then**

$l_k = Tx_{max}$

$n \leftarrow n+1$;

until $E_{R,N_0}^{n-1} \leq E_{R,N_0}^n$;

 display n, E .

of hops necessary to achieve a minimum energy scheme. From this we can derive as well the number of nodes (N_k) and the distance (d_k) from the BS for each k intermediate level. These values are stored in the matrix $E_{d,N}^k$ of the Algorithm 1. The associated transmission range can now be calculated for nodes in each k intermediate circle using formula (15):

$$l_k = \sqrt{d_k^2 + d_{k-1}^2 - 2d_k d_{k-1} \cos\left(\frac{N_k}{N_{k-1}} - 1\right) \gamma_0} \quad (15)$$

The following proposition holds:

Proposition 3. *For any given number of sensors N_0 placed at a distance d from the BS, problem P2 can be solved through dynamic programming using the recurrence formula (14).*

The case in which the nodes act only as relaying nodes² is easier. The nodes will transmit to the BS via the shortest path. The number of nodes on each intermediate circle will be the same as for the outer circle (N_0) and the distances from the BS are the same as with scenario (a) for the linear network.

1) *Numerical results for the 2D Network:* Numerical results are provided in table III to illustrate the application for both algorithms with a radius $R = 100m$ and the number of nodes $N_0 = 50$ and $N_0 = 60$.

Regarding the first algorithm, we stopped at the fourth iteration for the scenario with $N_0 = 50$. We remark that in this case only four hops are necessary to transmit the information to the BS with minimum energy, whereas for $N_0 = 60$, one additional hop is needed. On the other hand there are less hops for Algorithm 2 applied to the same scenarios. This is

²The sensor will neither generate nor aggregate the data traffic but simply forward it.

Table III: Energy values (in $10^{-4}J$) for each n

	With Aggregation		Without Aggregation	
	$N_0 = 50$	$N_0 = 60$	$N_0 = 50$	$N_0 = 60$
$n = 1$	0.5250	0.6300	0.5250	0.6300
$n = 2$	0.1641	0.1855	0.3669	0.4351
$n = 3$	0.1280	0.1414	0.3895	0.4587
$n = 4$	0.1249	0.1334	-	-
$n = 5$	0.1253	0.1326	-	-
$n = 6$	-	0.1337	-	-

obvious as the intermediate nodes increase the traffic load.

However, the above model remains essentially theoretical, because in real world applications a node may receive and transmit a certain quantity of traffic based on a network traffic pattern or the nodes can be already deployed and one needs to take decisions only for the range assignment. Nevertheless, it may be useful to calculate the lower bounds of energy consumption. This work gives a solid basis for tackling the topology control problem, which is discussed in the following section.

IV. TOPOLOGY CONTROL PROBLEM BASED ON POWER CONTROL

In some WSN applications the sensors may be already deployed in the monitoring area, the requirement being to transmit the data as energy-efficiently as possible. Here the problem is how to configure the network so as to meet this objective. Configuring a linear network involves creating a feasible division of the network into cells and determining their respective sizes (see Fig. 6). In the case of a two-dimensional network, the problem involves not only dividing the network into coronas of different lengths, but also subdividing these coronas into different zones (see Fig. 9). Regarding these objectives, we consider two strategies: (i) minimizing the overall energy consumption in the network (which we shall call Strategy 1) and (ii) guaranteeing a fair allocation of energy consumption between the sensors (which we shall call Strategy 2).

In this section we adapt the mathematical model used for the sensor placement problem to solve the topology control problem. We propose a method for optimally solving this problem under different conditions of traffic load. Finally, we show how to adapt the method to different energy consumption models without increasing the computational complexity of the solution.

A. Linear network

We consider a linear network with a length d from the BS, where the nodes are **uniformly distributed**, as in Fig. 6. The network will be divided into cells. The transmission structure is quite similar to the cluster-based scheme. For any cell there will be only one node,

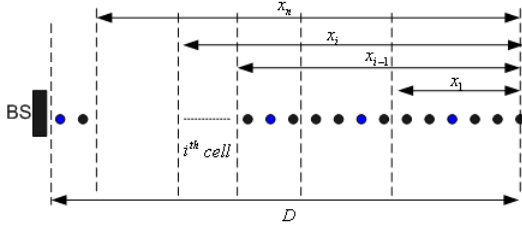


Figure 6: Network model

known as the cluster head, which receives the information from the other nodes in the same cell and from the cluster head of the upstream cell. The transmission range of a sensor corresponds to the length of the respective cell. Communication between sensors can be guaranteed as the sensors deployment is dense and the length of each cell can not exceed the maximal transmission range Tx_{max} . The received data are then transmitted to the cluster head of the downstream cell without performing any type of aggregation. The network model considered here is proposed in [?]. We assume that each node has α Erlang of traffic, and the radio data rate is β bps. Regarding the data aggregation model (see section III-B), we take $c = \alpha \cdot \beta$ and $m = 1$. The nodes are uniformly distributed with a density n_d . The i^{th} cell in the network (Figure 6) will consume a quantity of energy equal to E_{TX} when transmitting the information (see equation (16)).

$$E_{TX}(i) = (E_{elec} + E_{amp} \cdot (x_i - x_{i-1})^\gamma) \cdot x_i n_d \alpha \beta \quad (16)$$

where x_i and x_{i-1} are the distances of the i^{th} and $(i-1)^{th}$ cells respectively and γ as before presents the path loss exponent. As regards the reception of information, the energy consumed by the i^{th} cell when receiving is expressed by equation (17).

$$E_{RX}(i) = E_{rec} \cdot x_{i-1} n_d \alpha \beta \quad (17)$$

Besides the energy dissipation for receiving and transmitting, we take into account the energy spent during the idle state noted as E_{idle} . Based on the network model presented in [?], the idle energy for a cluster head of the i^{th} cell is calculated as follows:

$$E_{idle}(i) = E_{idle} \cdot (1 - 2x_{i-1} n_d \alpha) \cdot \beta \quad (18)$$

Here it is assumed that each node requires a time window equal with (δt) to transmit its own data. For receiving all the data, a cluster head of the i^{th} cell needs a time³ equal to $(x_{i-1} n_d \delta t)$. If T is the time of a round for the data collection, then the time the cluster head passes in the idle mode is $(T - 2x_{i-1} n_d \delta t)$ and the division with T leads to $(1 - 2x_{i-1} n_d \alpha)$. The network configuration problem consists in determining the optimal number of cells and the length of each cell such that (i) the total energy consumption is minimized

³The cluster head will spend the same time either for receiving or transmitting the data.

(Strategy 1) and (ii) *energy fairness* is obtained by ensuring that every cell in the network consumes the same amount of energy (Strategy 2). So far we have assumed that the node transmission energy depends on the communication distance. Since real sensors have discrete levels of power transmission we focus on minimizing the total energy consumption of nodes in the linear network using only discrete energy values. More specifically, given a linear network, we need to find the optimal number of cells, together with their respective distances, that guarantees not only a lower bound of energy consumption, but also energy balancing between cells or the minimizing of energy consumption for discrete energy levels (in IV-A1). As we can see, for any of these problems the number of possible combinations is exponential, and therefore we propose a method based on dynamic programming.

1) Optimization criteria for the linear network:

a) *Total energy minimization*: Here the objective is to minimize the total energy consumption for the linear network presented in IV-A. The total energy consumption function is convex, therefore it exists a global minimum of energy. We have proved the convexity of the function but for reason of space we have omitted it in this paper. As in the previous scenarios, the problem can be separated into a number of stages depending on the number of cells. At some stage n we determine, using the energy values obtained from the previous stage, the minimum energy necessary to transmit the data. Hence, the optimal solution for an instance of a problem with n cells and a given distance d can be seen as equivalent to the optimal solution for some distance $x < d$ and $n-1$ hops, plus a final hop from this point (x) to the BS. In the n^{th} iteration, the distance $(d-x)$ represents the length of the n^{th} cell. The Algorithm 2 begins with $n=1$ which assumes that all the nodes transmit with a range d . Next, n is incremented and for each distance y ($1 \leq y \leq d$) the best division is depicted by x . Importantly, when the optimal n is found out, the x value is identified as well. We backtrack all the x_i values for each stage n_i to construct the whole solution and the difference $(x_i - x_{i-1})$ represents the length of each cell. The energy added by the last hop $E(y-x)$ is calculated according to formula (19).

$$E(y-x) = e1 + e2 + e3 + e4 \quad (19)$$

where:

$$e1 = E_{rec} \cdot x n_d \alpha \beta$$

is the energy spent for receiving the data of the upstream cell,

$$e2 = (E_{elec} + E_{amp}(y-x)^\gamma) \cdot y n_d \alpha \beta$$

gives the energy dissipated by the cluster head to transmit all the collected data,

$$e3 = (E_{elec} + E_{amp}(y-x)^\gamma + E_{rec}) \cdot (y-x) n_d \alpha \beta$$

Algorithm 2: Strategy 1

Input: Distance d , node density n_d ;
Output: Optimal number of network division n ,
Minimal energy E ;
Initialization : $\{E_y^1 | y \leq d\}$, $n = 1$;

repeat
 for $y \leftarrow 1$ **to** d **do**
 $E_y^{n+1} = \infty$;
 for $x \leftarrow 1$ **to** $y - \delta y$ **do**
 $E_y^{n+1} \leftarrow \min \{E_y^{n+1}, E_x^n + E(y-x)\}$
 $n \leftarrow n+1$;
 until $E_d^{n-1} \leq E_d^n$;
 display n, E .

corresponds to the energy consumed for transmitting and receiving the data of the other sensors deployed in the cluster,

$$e4 = E_{idle} \cdot (1 - 2xn_d\alpha) \cdot \beta$$

and finally the $e4$ is the energy spent during the idle state. Note that E_y^n represents the total energy consumption where the network length is y and divided into n cells. The distances x and y are discretized using a distance discretization step δy . The initialization phase of the algorithm requires $O(y/\delta y)$ computations to construct the vector E_y^n . The calculation procedure costs at most $O((y/\delta y)^2)$ and it will be repeated n times. Finally, the computation complexity of the algorithm is $O(n \cdot (y/\delta y)^2)$.

Remark 1. Energy minimization with discrete transmission levels

In the above strategy while using the energy model described in (1), we estimate the sensor transmission energy according to the exact distance of transmission. However, the transmission power of sensor nodes is discrete and therefore it takes only some given values. In this section, we provide Algorithm 3, a modified version of Algorithm 2, which addresses this problem.

In the Algorithm 3, the term $(E \lceil y - x \rceil)$ is computed using equation (19) while the formula of the transmission energy is modified. Hence, $(\lceil y - x \rceil)$ is matched with the closest element⁴ of the list L . We associate to each distance l_i from the list l a power vale e_i belonging to list e . Then, the terms $e2$ and $e3$ of the equation (19) will be computed using the e_i power values as following:

$$e2 = e_i \cdot yn_d\alpha\beta$$

and

$$e3 = (e_i + E_{rec}) \cdot (y - x)n_d\alpha\beta$$

⁴We use the ceiling value of the distance $(y - x)$ to ensure that the closest matched element in the list L is not less than $(y-x)$.

Algorithm 3: Energy minimization with discrete transmission levels

Input: Distance d , node density n_d , the list of energy power levels $e=\{e_1, e_2, \dots, e_n\}$, the corresponding list of the respective distances $L=\{l_1, l_2, \dots, l_m\}$;
Output: Optimal number of network division n ,
Optimal cell sizes x , Minimal energy E ;
Initialization : $\{E_y^1 | y \leq d\}$, $n = 1$;

repeat
 for $y \leftarrow 1$ **to** d **do**
 $E_y^{n+1} \leftarrow \min_{\substack{0 < x < y \\ \lceil y-x \rceil \in L}} \{E_x^n + E \lceil y - x \rceil\}$
 $n \leftarrow n+1$;
 until $E_d^{n-1} \leq E_d^n$;
 display n, E .

b) *Energy fairness:* In order to avoid the energy hole problem, we need to guarantee that each node in the network will consume the same amount of energy, independently of the quantity of received and generated data. This is the motivation behind our second strategy which seeks to ensure the equitable expenditure of energy in the network. One way to solve this problem is to divide the network into k cells and write a system of k equations, where for each cell the total energy consumption E_i is given by equation (20).

$$E_i = (E_{elec} + E_{amp}(x_i - x_{i-1})^\gamma) \cdot x_i n_d \alpha \beta \\ + (E_{elec} + E_{amp}(x_i - x_{i-1})^\gamma) \cdot (x_i - x_{i-1}) n_d \alpha \beta \quad (20) \\ + E_{rec} \cdot n_d \alpha \beta x_{i-1} + E_{idle} \cdot (1 - 2x_{i-1} \alpha n_d) \cdot \beta$$

The equation $\sum_{i=1}^k (x_i - x_{i-1}) = d$ enables us to have k equations and to find all the variables x_i for all $i \in 1 \dots k$ and E . However, this method proposed in [?] needs to solve a system of k nonlinear equations for k cells. Instead, we propose Algorithm 4, also based on the dynamic programming method.

Here E_y^n represents the total energy consumption while $\frac{E_y^n}{y \cdot n_d}$ is the average energy consumption of a sensor in a cell⁵ when the network of length y is divided into n cells. $E(y - x)$ is the energy added only by the last hop (see equation (19)). The ratio $\frac{E(y-x)}{(y-x) \cdot n_d}$ calculates the sensor node energy consumption for the last cell. The IF condition in the algorithm identifies the solutions which guarantee a fairness dissipation of energy with a given threshold. The algorithm works as follows: it finds the network division (x_i values) for a given n such that energy is balanced. Next, it repeats this procedure till it reaches the n which minimizes the total energy consumption. At the end of execution,

⁵The sensors in all cells will consume an identical amount of energy.

Algorithm 4: Strategy 2

Input: Distance d , Node density n_d , Coefficient ϵ ;

Output: Optimal number of network division n ,
Minimal energy E ;

Initialization : $\{E_y^1 | y \leq d\}$, $n = 1$;

repeat

 for $y \leftarrow 1$ **to** d **do**

 $E_y^{n+1} = \infty$;

 for $x \leftarrow 1$ **to** $y - \delta y$ **do**

 if $(\frac{E_x^n}{x \cdot n_d} - \frac{E(y-x)}{(y-x) \cdot n_d}) < \epsilon \cdot \frac{E(y-x)}{(y-x) \cdot n_d}$ **then**

 $E_y^{n+1} \leftarrow \min\{E_y^{n+1}, E_x^n + E(y-x)\}$;

 $n \leftarrow n + 1$;

until $E_d^{n-1} \leq E_d^n$;

 display n, E .

the algorithm displays the number of cells and the total energy consumption while the length of each cell is found by backtracking the x_i values.

2) *Numerical results:* We tested our algorithms with respect to optimization strategies 1 and 2 and compared their results with the best case of uniform cell division. Uniform cell division means that the network is divided into n equal size cells, and the best case corresponds to the n that minimizes the energy value. The results of total energy consumption for the three cases with respect to the distance are shown in Fig 7. For this simulation we set the simulation parameters as shown in Table IV.

Table IV: Simulation parameters for the linear network

Type	Parameter	Value
Network	Network Length	200 ~ 2000m
	Node distribution	Uniformly
	Node density	0.4
Application	Data generation rate β	485 bps
	Data traffic per node α	0.003 Erlang
	ϵ	0.07
	Compression ratio	$m = 1 \mid c = \alpha \cdot \beta$
Energy	E_{elec}	50nJ/bit
	E_{amp}	100pJ/bit/m ²
	E_{rec}	50nJ/bit
	E_{idle}	40nJ/bit
	γ	2
	$L = \{l_1, l_2, \dots, l_m\}$	[20, 40, 60, 80] in meters
	$E = \{e_1, e_2, \dots, e_m\}$	$[0.09, \dots, 0.69] \cdot 10^{-6}$ /bit
Method	Distance discretization	$\delta y = 1$

We notice that strategies 1 and 2 outperform the best case of uniform cell division considering the overall consumption energy. If we zoom out the results for

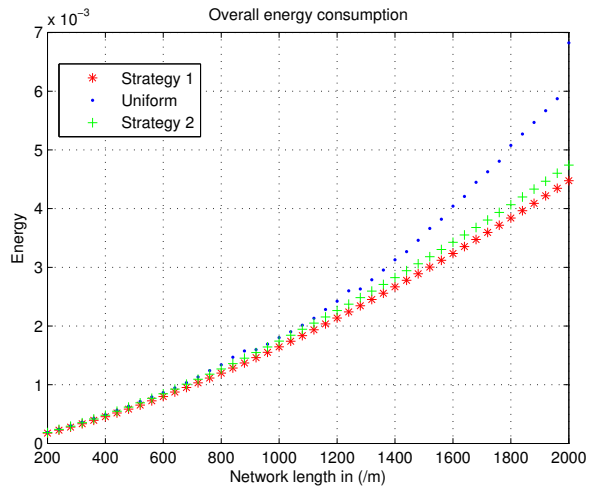


Figure 7: Energy consumption

the extremity points taken in our simulation (200m and 2000m), the strategy 1 consumes 15% (resp. 36.2%) less energy than uniform cell division for 200m (resp. 2000m). On the other hand, strategies 1 and 2 divide the network such that the energy consumption is more balanced compared to the uniform one. Strategy 1 gives a lower bound for total energy consumption, but remains comparable to Strategy 2. For a network with a distance of 200m (resp. 2000m) the difference is around 4% (resp. 4.5%) in energy saving compared to Strategy 2.

Table V gives the cell divisions, beginning with the cell closest to the Base Station, generated by strategies 1 and 2. We notice that these strategies yield different network scenarios, even where they both have the same optimal number of cells. As the distance increases, the configuration scenarios different more obviously from each other. The radio parameters also strongly influence this behavior.

Network length (250 m)	Network Division		
	Strategy 1	Strategy 2	Uniform
cell 1	35	36	41.66
cell 2	37	37	41.66
cell 3	39	40	41.66
cell 4	42	42	41.66
cell 5	47	45	41.66
cell 6	50	50	41.66

Table V: Linear network configuration

As expected, for the energy discretization problem the sensors consume more energy. For this simulation we assume that each node can achieve the distances $L = l_1, l_2, \dots, l_m$ using the respective energies $E = e_1, e_2, \dots, e_m$. Fig. 8 shows that the continuous energy value still remains an acceptable approximation of energy behavior (the gap between the discrete and the

continuous case reaches 6% for 400m). Energy does not diverge between the discrete and continuous cases, regardless of the distance, given that they both use the same energy model.

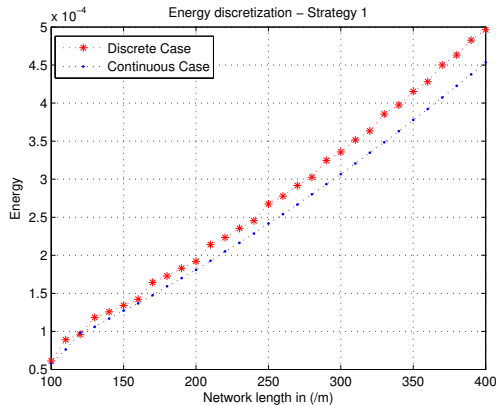


Figure 8: Energy discretization

B. Two-dimensional sensor network

The second problem considered here is the two-dimensional sensor network configuration problem. We assume that the sensors are uniformly distributed in a circular monitoring area. There is only one sink, positioned at the center of the zone.

The sensors will transmit their information according to a many-to-one traffic pattern using a multihop scheme. Total energy consumption will be minimized by optimally dividing the area into concentric coronas C_1, C_2, \dots, C_n centered at the sink and then subdividing each corona into a given number of zones. Fig. 9 gives an example of network division with 3 coronas containing respectively 1, 4 and 8 zones. Moreover, the length of every corona can be varied. For each zone there is only one sensor, the cluster head, that is able to receive information from other sensors in the zone, aggregate this information and transmit it to the downstream zone closer to the BS. We assume that any sensor in the zone is a potential cluster head. Notice that we can easily shift from this model to a specific-sensor placement model in which the cluster head position must be explicitly determined by adapting the radio transmission range allocated to each zone. In accordance with the aggregation model introduced in Section III-B, here we analyze the case with $m = 0$ in which a node will totally aggregate the information and transmit only a fixed amount of information. If we assume that no aggregation of sensor information will be performed by the cluster heads then the problem is reduced to the linear network case, because sensors will choose the shortest path when transmitting information. Finally, for the network model as described above, the questions to be answered are:

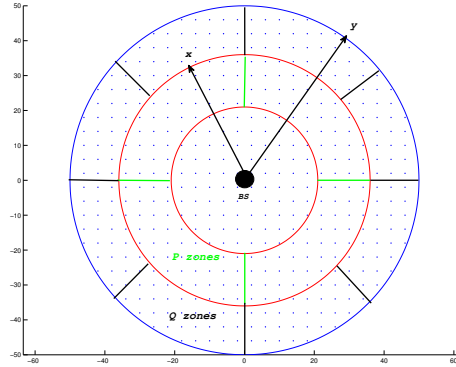


Figure 9: Two-dimensional network

- What is the optimal number of coronas that minimizes energy across the whole network? What are their respective radii?
- What is the optimal number of zones that a corona is divided in?

Here we assume that each zone has a cluster head that will aggregate the traffic and transmit it towards the cluster head of the downstream zone closer to the BS. For the sake of the simplicity, the idle energy is not considered for this problem.

1) Optimization criteria:

a) *Minimal energy consumption - Strategy 1:* The problem for a two-dimensional network with traffic aggregation is more complex than the linear case. We opt once again for a method based on parameter discretization of the *radius* and the *number of zones for each corona* respectively, and apply the dynamic programming principle. We assume that the sensor network has a uniform node distribution n_d and each sensor generates a constant traffic $\alpha \cdot \beta$. Hence, for a such network with radius R and number of zones N , we apply Algorithm 5.

In the initialization phase, the algorithm computes the energy spent in a simple cell ($N = 1$) in which all the sensors transmit their information directly to the BS using the same transmission range y . The energy values for the first vector $E_{y,1}^1$ are computed according to formula (21).

$$E_{y,1}^1 = (E_{elec} + E_{amp} \cdot y^\gamma) \cdot n_d \pi y^2 \alpha \beta \quad (21)$$

In the next step, the number of coronas is two ($n = 2$) and the algorithm computes the matrix $\{E_{y,Q}^2 | (y \leq R, Q \leq N)\}$. For each n coronas' number, a new matrix $\{E_{y,Q}^n | (y \leq R, Q \leq N)\}$ is generated where N gives the maximal number of the zone divisions of the last corona. The elements of each matrix for a given n are computed based on the dynamic programming principle where the last hop term is represented by $E(y - x, Q - P)$ and computed according to formula (22). Here $(y - x)$ gives the length of the last corona and Q (resp. P) gives the number of zones in the last

Algorithm 5: Network area configuration - Strategy 1

Input: Node density n_d , Network area radius R ,
Maximal number of zone divisions N ,
Generated traffic per node $\alpha\beta$;

Output: Minimal energy E , Optimal number of coronas n ;

Initialization : $\{E_{y,1}^1 | y \leq R\}$, $n=1$;

repeat

for $((y \leftarrow 1$ to $R)$ and $(Q \leftarrow 2$ to $N))$ **do**

$E_{y,Q}^{n+1} = \infty$;

for $((x \leftarrow 1$ to $y)$ and $(P \leftarrow 2$ to $Q))$ **do**

$E_{y,Q}^{n+1} \leftarrow \min \{E_{y,Q}^{n+1}, E_{x,P}^n + E(y-x, Q-P)\}$

$n \leftarrow n+1$;

$E_y^n \leftarrow \min_{Q \leq N} \{E_{y,Q}^n\}$;

until $E_R^{n-1} \leq E_R^n$;

display n, E .

(resp. last but one) corona. A simple illustration is given in Fig. 9.

$$E(y-x, Q-P) = Q \cdot f_1(y, x, P) + f_2(y, x, Q) \quad (22)$$

where:

$$f_1(y, x, P) = (E_{elec} + E_{amp} \cdot (R_t(y, x, P))^\gamma + E_{rec}) \cdot \alpha\beta \quad (23)$$

with

$$R_t(y, x, P) = \left(y^2 + x^2 - 2yx \cos\left(\frac{2\pi}{P}\right) \right)^{\frac{1}{2}} \quad (24)$$

Equation (23) gives the energy used by the cluster head of each zone in the n^{th} corona to transmit the information towards the next cluster head closer to the BS while using a transmission range defined by the $R_t(y, x, P)$ function. As in the sensor placement problem, the transmission range is calculated based on the cosine formula. However, in this problem we do not know the cluster head placement, therefore we consider the maximal angle for computing the transmission range which depends only on the number of zones P of the precedent level $n-1$.

$$f_2(y, x, Q) = \left(E_{elec} + E_{amp} (R_t(y, x, Q))^\gamma + E_{rec} \right) \cdot n_d \pi (y^2 - x^2) \alpha\beta \quad (25)$$

Equation (25) gives the energy consumed by all the sensors in a corona for performing intra-zone transmissions⁶. Regarding algorithm 5, we calculate the matrix $\{E_{y,Q}^n | (y \leq R, Q \leq N)\}$ for subsequent values of parameter n . Next, we find the E_R^n which is the minimal value of the last row of the matrix $\{E_{R,N}^n\}$. We stop the calculations when we reach some n such that $E_R^n \leq E_R^{n+1}$. During the execution of the algorithm we

⁶Intra-zone transmissions include the receiving and transmitting operations inside a zone between the cluster head and the other sensors in the zone.

keep track of those values (x_i, P_i) that allow to achieve the minimum of energy at the current iteration. The solution can then easily be built by backtracking these stored values. At the end of the algorithm we obtain the optimal number of coronas for minimum energy consumption, as well as the number of zones for each intermediate corona and the corresponding radius.

For the simulations, the nodes are uniformly deployed in the circular area and generate a constant amount of traffic. The cluster head of each zone aggregates all the information that it receives and transmits a constant amount of traffic to the closest cluster head. Table VI lists the system configuration parameters in detail whereas the values of the energy model are given in Table IV. For these parameters, some experimental results for Algorithm 5 are presented in Table VII. As we can observe, the optimal number of coronas is 9, and the algorithm determines the radius of each corona and the number of zones, both of which decrease uniformly.

Table VI: Simulation parameters

Type	Parameter	Value
Network	Area radius	50m - 300m
	Node distribution	Uniformly
	Node density	0.02
Application	Data generation rate β	100 bps
	Data traffic per node α	0.003 Erlang
	ϵ	0.07
	Compression ratio	$m = 0 \mid c = \alpha \cdot \beta$

Cell sizes	Algorithm results	
	Radius of each corona from the BS	Number of zones
corona 9	100	27
corona 8	90	24
corona 7	80	21
corona 6	70	18
corona 5	60	15
corona 4	50	12
corona 3	40	9
corona 2	30	6
corona 1	22	1

Table VII: Optimal configuration for overall energy consumption, network radius $R = 100m$

The energy consumptions of the whole network for different network radii and node densities are presented in Fig. 10.

b) *Energy fairness - Strategy 2:* Assuring energy fairness between the network clusters is an alternative objective to minimizing total energy consumption.

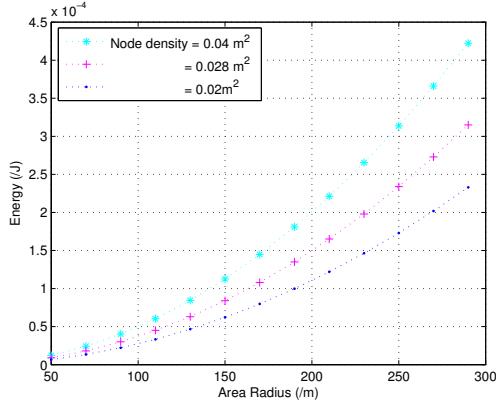


Figure 10: Energy versus network area radius

Here the optimal solution to the problem gives a network configuration such that differences in the energy consumed by the sensors of different cells and the overall energy consumed are minimized. To solve this problem, we propose the Algorithm 6. Here

Algorithm 6: Network area configuration - Strategy 2

Input: Node density n_d , Network area radius R ,
Maximal number of zone divisions N ,
Generated traffic per node $\alpha\beta$, Coefficient ϵ
;

Output: Minimal energy E , Optimal number of coronas n ;

Initialization : $\{E_{y,1}^1 | y \leq R\}$, $n=1$;

repeat

for $((y \leftarrow 1$ to $R)$ and $(Q \leftarrow 2$ to $N))$ **do**

$E_{y,Q}^{n+1} = \infty$;

for $((x \leftarrow 1$ to $y)$ and $(P \leftarrow 2$ to $Q))$ **do**

if $(\frac{E_{x,P}^n}{N1} - (\frac{f_1(y,x,P)+f_2(y,x,Q)}{N2}) < \epsilon \cdot \frac{E_{x,P}^n}{N1})$ **then**

$E_{y,Q}^{n+1} \leftarrow \min\{E_{y,Q}^{n+1}, E_{x,P}^n + E(y-x, Q-P)\}$

$n \leftarrow n+1$;

$E_y^n \leftarrow \min_{Q \leq N} \{E_{y,Q}^n\}$;

until $E_R^{n-1} \leq E_R^n$;

 display n, E .

$E_{y,Q}^n$ represents the total energy consumption while $\frac{E_{y,Q}^n}{N1}$ is the average energy consumed by a sensor when the network's radii is y and its n^{th} corona is divided into Q zones. The term $N1$ is the number of the sensors deployed in the area with a radii y . In the initialization phase, matrix $E_{y,1}^1$ is computed according to equation (21). The $E(y-x, Q-P)$, $f_1(y, x, P)$, $R_f(y, x, P)$ and $f_2(y, x, Q)$ functions are given by the corresponding equations (22), (23), (24) and

(25). The ratio $\frac{f_1(y,x,P)+f_2(y,x,Q)}{N2}$ gives the sensor's energy in the respective zone while $N2 = \frac{(y^2-x^2) \cdot 2\pi \cdot n_d}{Q}$ is the corresponding number of the sensors in this zone.

At first, the algorithm finds the network division (x_i, P_i) values for a given n which guarantees a fairness consumption of energy for each sensor in the network with a coefficient ϵ . Second, by iterating this procedure for different n , it seeks the minimal value of total energy consumption. This value is reached for some n which satisfies the inequality $E_R^{n-1} \leq E_R^n$. Algorithm 6 is implemented for the scenario described in Section IV-B. The application and parameter values are defined in Table VI where the energy parameters are given in Table IV. The results are shown in Table VIII. For application needs, the number of zones in one corona should be a multiple of this in the downstream corona. However, this can be easily taken into account by assuming that the number Q of zones, in algorithm 6, is a multiple of P .

Cell sizes	Algorithm results	
	Radius of each corona from the BS	Number of zones
corona 8	100	30
corona 7	86	27
corona 6	72	21
corona 5	60	15
corona 4	48	12
corona 3	36	9
corona 2	24	6
corona 1	12	1

Table VIII: Optimal configuration for energy fairness, network radius $R = 100m$

2) *Numerical results:* We remark that Strategy 1 builds a multi-hop tree rooted at the BS which minimizes the total energy consumption for the communications. One basic assumption of this strategy is full aggregation, meaning that each node will transmit only one packet to its parent regardless of the number of received data packets. The work in [?] considers the same constraints and objective function to build a multi-hop tree and propose a protocol PEDAP (Power Efficient Data Gathering and Aggregation Protocol). The algorithm behind the PEDAP protocol builds a near optimum spanning tree and is designed for data gathering in sensor networks. One basic assumption of this algorithm is full aggregation, meaning that each node will transmit only one packet to its parent regardless of the number of received data packets. In both cases, we assume that nodes are uniformly distributed in a circular area. To calculate the energy consumed by PEDAP we construct a spanning tree that includes all the nodes. In the spanning tree we differentiate between terminal and in-network nodes.

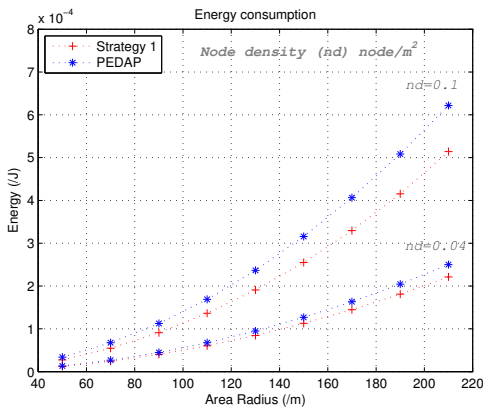


Figure 11: PEDAP and Strategy 1

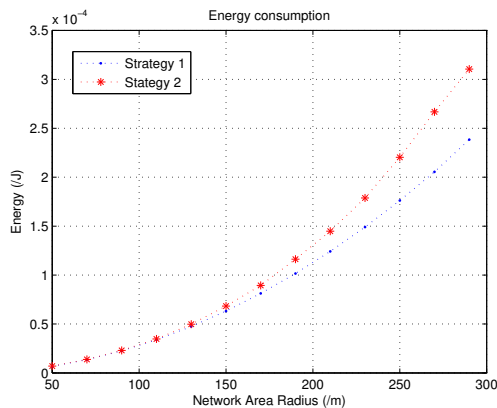


Figure 12: Energy consumption

The terminal nodes use energy only for transmitting, while the others use energy for transmitting and receiving, in accordance with equations (1). In both scenarios each node has α Erlang of traffic, and the radio data rate is β bps. The results of the comparison between Strategy 1 and PEDAP are given in Fig. 11. Our algorithm outperforms PEDAP, and more so in the case of large dense networks, which shows that it is quite scalable. The overall energy consumption for Strategies 1 and 2 is given in Fig. 12.

Through simulation we also remarked that the behavior of energy function toward the node density changes is similar for both strategies.

For the topology control problem we have looked at two strategies which seek respectively to i) minimize the energy consumption and ii) ensure a fair energy distribution while minimizing the total energy consumption. The trade-offs between the two strategies are in terms of global energy, fairness and complexity. The results show that Strategy 1 presents a lower bound of network energy consumption, but also that Strategy 2 is very close to this bound (see Fig. 12).

V. CONCLUSIONS

In this paper we analyzed the network configuration problem for a many-to-one WSN. For a given

network where the nodes are uniformly distributed, the problem is to determine the optimal number of sensors, their positions and their transmission range such that some energy objectives are met. We showed that an appropriate configuration of the WSN, addressing the sensor deployment and the respective power assignments, leads to considerable savings in energy. Our algorithms, whose aims are to minimize the total energy consumption and to ensure fairness in the energy consumption by different nodes, are based on the dynamic programming method. We showed how this method can be adapted even to the discrete case of the energy consumption minimization problem. Then, we extended the linear network case to the case of a two-dimensional network. Our overall observation is that dynamic programming is an effective method for handling trade-offs between the parameters for a number of variants of the network configuration problem. It provides an optimal solution to the problem for a given objective function and reduces the computational complexity in comparison with other methods proposed in the literature. Moreover, the method can be adapted to different energy consumption models without increasing the computational complexity of the solution. In the future we shall be looking at the effects of combining different routing protocols with solutions obtained for the network configuration problem. We think that this sort of combination will be of interest at the maintenance phase and for implementation in real world applications.