

# Energy-aware Dynamic Resource Allocation in Virtual Sensor Networks

Carmen Delgado\*, María Canales\*, Jorge Ortín\*<sup>†</sup>, José Ramón Gállego\*,  
Alessandro Redondi<sup>‡</sup>, Sonda Bousnina<sup>‡</sup>, Matteo Cesana<sup>‡</sup>

\*Aragón Institute of Engineering Research. Universidad de Zaragoza, Zaragoza, Spain. {cdelga, jrgalleg, mcanales}@unizar.es

<sup>†</sup>Centro Universitario de la Defensa Zaragoza. Academia General Militar, Zaragoza, Spain. jortin@unizar.es

<sup>‡</sup>Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano, Italy.  
{alessandroenrico.redondi, sonda.bousnina, matteo.cesana}@polimi.it

**Abstract**—Sensor network virtualization enables the possibility of sharing common physical resources to multiple stakeholder applications. This paper focuses on addressing the dynamic adaptation of already assigned virtual sensor network resources to respond to time varying application demands. We propose an optimization framework that dynamically allocate applications into sensor nodes while accounting for the characteristics and limitations of the wireless sensor environment. It takes also into account the additional energy consumption related to activating new nodes and/or moving already active applications. Different objective functions related to the available energy in the nodes are analyzed. The proposed framework is evaluated by simulation considering realistic parameters from actual sensor nodes and deployed applications to assess the efficiency of the proposals.

## I. INTRODUCTION

The Internet of Things (IoT) paradigm considers that real world objects can be equipped with sensing capabilities to gather information on their environment and then to deliver it to the Internet. This data delivery can be done through wireless multi-hop paths leveraging the cooperation of other smart objects for traffic relaying and making up networks, which are often known as Wireless Sensor Networks (WSN). Typically, the hardware and network resources in WSNs are designed and deployed to the specific application requirements. While this paradigm allows to have “optimal” performance on the specific application, it prevents other applications from reusing the already deployed hardware and software resources, leading to the proliferation of redundant WSNs deployments.

In this context, virtualization is a promising technique to achieve an efficient reuse of general purpose wireless sensor networks to dynamically support multiple applications and services [1]. The key idea behind this approach, which often goes under the names of Virtual Sensor Networks (VSN), is to abstract away “physical resources” including node processing/storage capabilities, available communication bandwidth and routing protocols, which can then be “composed” at a logical level to support usage by multiple independent users and even by multiple concurrent applications. This new paradigm has stimulated research efforts in the field of novel programming abstractions at the node level and management framework at the network level to support multiple applications over a shared physical infrastructure [1], [2].

Nevertheless, comprehensive solutions for dynamic resource allocation that cope with the specific limitations of WSNs

still need to be found. In [3] an optimization framework for environmental monitoring applications is proposed that aims to perform an application-to-sensors assignment which minimizes the variance of the sensed data. The authors of [4] propose an optimization framework to prolong network lifetime by properly scheduling the tasks in a shared/virtual sensor network. In previous works [5], [6] we have proposed a mathematical programming framework to optimally allocate in a static scenario the shared physical resources of the general purpose WSN to multiple concurrent applications. In these works the whole set of applications was known in advance and constant during the analysis time period. Now we focus on addressing the dynamic adaptation of the allocated resources when applications demands are time varying. To respond to increasing network demands, new resources may be allocated or already used resources may have to be reallocated. This implies additional energy consumption related to activating new nodes and/or moving already active applications that must be considered. Due to the energy limitations of sensor networks, we tackle the problem of dynamic resource allocation trying to optimize the use of network energy. Numerical results are then obtained by applying the proposed framework to realistic WSN instances to assess the efficiency of the different proposals.

## II. SYSTEM MODEL AND OPTIMIZATION FRAMEWORK

Let  $S = \{s_1, s_2, \dots, s_l\}$  be a set of sensor nodes scattered in a reference area. Each sensor node  $s_i$  has an energy budget  $E_i$ . Let  $T = \{t_1, \dots, t_n\}$  be a set of test points in the reference area, which are physical locations where some parameters must be measured. Let  $A = \{a_1, a_2, \dots, a_m\}$  be a set of applications to be deployed in the system. Each application  $a_j$  arrives at time  $\tau_j$  and has a lifetime  $\epsilon_j$ . In the following, we will use the subscript index  $i$  (or  $h$ ) to refer to a sensor node  $s_i$  (or  $s_h$ ), the subscript index  $j$  to refer to an application  $a_j$  and the subscript index  $k$  to refer to a test point  $t_k$ .

Each application  $j$  requires to sense a given set of test points  $T_j \subseteq T$ . Formally, the application  $j$  has to be deployed in a subset of the sensor node set  $S$ , such that all the test points in  $T_j$  are sensed. A test point is covered by a sensor node  $i$  if it is within its sensing range,  $R_i^s$ . Thus, given a test point, a set of sensor nodes can *cover* it, but only one sensor node will *sense* it. Let  $S_{jk}$  be the set of sensor nodes which cover the test point  $k$ , with  $k \in T_j$ . A necessary condition for an

application  $j$  to be successfully deployed is that all the test points in its target set  $T_j$  must be sensed during its whole lifetime  $\epsilon_j$ . Each application  $j$  in  $A$  is further characterized by a characteristic vector  $r_j = \{c_j, m_j, l_j\}$  which specifies the required source rate, memory and processing load demanded by the application when it is deployed on a sensor node. Each sensor node  $i$  in  $S$  is characterized by a resource vector  $o_i = \{C_i, M_i, L_i, E_i\}$ , which specifies its available bandwidth, storage capabilities, processing power and energy. A protocol interference model with power control is used to characterize the wireless communications. The maximum transmission power is  $P_{max}$ . Given a directional link between a pair of nodes  $(i, h)$ , the channel gain from  $i$  to  $h$  is  $g_{ih} = g_0 \cdot d_{ih}^{-\gamma}$ , being  $d_{ih}$  the distance,  $\gamma$  the path loss index and  $g_0$  a constant dependent on antenna parameters. If  $p_i$  is the transmission power assigned to node  $i$ , a transmission towards  $h$  is successful if  $p_i \cdot g_{ih} > \alpha$  and interference at other node is non-negligible if  $p_i \cdot g_{ih} > \beta$ , being  $\alpha$  and  $\beta$  the receiver and interference sensitivities. Thus, the transmission range for node  $i$  can be obtained as  $R_i^T(p_i) = (p_i \cdot g_0 / \alpha)^{1/\gamma}$  and the interference range is  $R_i^I(p_i) = (p_i \cdot g_0 / \beta)^{1/\gamma}$ .

Next, we define the optimization problem to be solved every time a new application arrives in the system. Let  $y_{ijk}$  be a binary variable indicating if sensor node  $i$  is sensing test point  $k$  of application  $j$  and  $x_i$  a binary variable indicating if sensor node  $i$  is active in the network. Let  $\tau^*$  be the time instant at which a new application arrives to the system and  $A^*$  the set of applications that are running in the network at  $\tau^*$ , including the new application, i.e.  $A^* = \{a_j | \tau^* \in [\tau_j, \tau_j + \epsilon_j)\}$ . The following sets of restrictions force all the applications in  $A^*$  to be deployed. The problem may be unfeasible. If so, the system is left as it is to ensure that the current applications are not rejected. Constraint (1) forces all the applications in  $A^*$  sense all their test points. Eq. (2) ensures that if a sensor  $i$  does not cover a test point  $k$  of an application  $j$ , then it can not sense it. Eq. (3) assures that  $N_{ij}$  (maximum number of test points of the same application  $j$  that a sensor  $i$  can sense) is not exceeded. Eqs. (4)-(5) are budget-type constraints for the available storage and processing load of the nodes.

$$\sum_{i \in S_{jk}} y_{ijk} = 1 \quad \forall j \in A^*, \forall k \in T_j \quad (1)$$

$$y_{ijk} = 0 \quad \forall i \notin S_{jk}, \forall j \in A^*, \forall k \in T_j \quad (2)$$

$$\sum_{k \in T_j} y_{ijk} \leq N_{ij} \quad \forall i \in S, \forall j \in A^* \quad (3)$$

$$\sum_{j \in A^*} \sum_{k \in T_j} m_j y_{ijk} \leq M_i \quad \forall i \in S \quad (4)$$

$$\sum_{j \in A^*} \sum_{k \in T_j} l_j y_{ijk} \leq L_i \quad \forall i \in S \quad (5)$$

Deployed applications will most likely require that data generated locally are delivered remotely to collection points (sink nodes) through multihop paths. By resorting to a fluid model, it should be ensured that all the data produced by the sensors are received by the sink nodes. This fact can be expressed

using constraints (6)-(8). Constraint set (9) enforces that if a sensor node is either running an application or receiving data, then it must be active in the network. Constraints (10)-(12) ensure that all the traffic flowing out of a sensor has only one possible route to a sink, as it is typical in WSNs routes.

$$f_{ih} = \sum_{j \in A^*} f_{ihj} \quad \forall i, h \in S \quad (6)$$

$$\sum_{\substack{h \in S \\ i \neq h}} f_{hi} - \sum_{\substack{h \in S \\ h \neq i}} f_{ihj} + \sum_{k \in T_j} c_j y_{ijk} = 0, \forall j \in A^*, \forall i \in S' \quad (7)$$

$$\sum_{j \in A^*} |T_j| c_j = \sum_{h \in S \setminus S'} \left( \sum_{\substack{i \in S \\ i \neq h}} f_{ih} + \sum_{j \in A^*} \sum_{k \in T_j} c_j y_{hjk} \right) \quad (8)$$

$$\sum_{\substack{h \in S \\ h \neq i}} f_{hi} + \sum_{j \in A^*} \sum_{k \in T_j} c_j y_{ijk} \leq K x_i \quad \forall i \in S \quad (9)$$

$$b_{ih} \leq l_{ih} \quad \forall i, h \in S \quad (10)$$

$$\sum_{h \in S} b_{ih} \leq 1 \quad \forall i \in S \quad (11)$$

$$f_{ih} \leq K b_{ih} \quad \forall i, h \in S \quad (12)$$

where  $S'$  is the set of nodes that are not sinks (a subset of  $S$ ),  $f_{ihj}$  is a variable representing the flow of data of application  $j$  in bps transmitted from node  $i$  to node  $h$ ,  $f_{ih}$  is a variable representing the flow of data in bps transmitted from node  $i$  to node  $h$  and  $K$  is a constant higher than the maximum transmission rate of a node.  $b_{ih}$  is a binary variable which indicates if data are transmitted from node  $i$  to node  $h$ , and  $l_{ih}$  is a constant that indicates if there is a viable link between  $i$  and  $h$ , i.e., if the distance between both nodes is less than the maximum transmission range,  $l_{ih} = 1$  and  $l_{ih} = 0$  otherwise.

The available bandwidth in the network is limited and must be shared among sensor nodes. We assume that a fair medium access control scheme orchestrates the access. Given a directional link between a pair of nodes  $(i, h)$ , let the capacity of the link be defined as  $C_{ih} = \min(C_i, C_h)$ . According to the considered protocol interference model, for each link in the network it must be ensured that the fraction of time used by the link plus all its interferences is less or equal to 1:

$$\frac{f_{ih}}{C_{ih}} + \sum_{\substack{g \in S \\ g \neq h}} \frac{f_{ig}}{C_{ig}} + \sum_{g \in S} \frac{f_{gi}}{C_{gi}} + \sum_{\substack{g \in S \\ g \neq i}} \frac{f_{hg}}{C_{hg}} + \sum_{\substack{g \in S \\ g \neq i}} \frac{f_{gh}}{C_{gh}} + \sum_{\substack{g, t \in S \\ d_{it} < R_i^I(p_i)}} \frac{f_{gt}}{C_{gt}} + \sum_{\substack{g, t \in S \\ d_{gh} < R_g^I(p_g)}} \frac{f_{gt}}{C_{gt}} \leq 1 \quad \forall i, h \in S \quad (13)$$

The energy budget of each node  $i$ ,  $E_i$ , is limited and it decreases when an application is deployed or the node has to forward data from other nodes. The power dissipation for the application  $j$  at the radio transmitter  $P_{ij}^t$  or at the radio receiver  $P_{ij}^r$  of each node  $i$  can be modeled as [7]:

$$P_{ij}^t = \sum_{h \in S, h \neq i} (\beta_1 + \beta_2 d_{ih}^\gamma) f_{ihj} \quad \forall i \in S, \forall j \in A^* \quad (14)$$

$$P_{ij}^r = \rho \sum_{h \in S, h \neq i} f_{hij} \quad \forall i \in S, \forall j \in A^* \quad (15)$$

Typical values for  $\beta_1$ ,  $\beta_2$  and  $\rho$  are  $\beta_1 = \rho = 50$  nJ/bit and  $\beta_2 = 0.0013$  pJ/bit/m<sup>4</sup>, with  $\gamma = 4$  the path loss index.

The estimation of the power dissipation due to the processing load, which can not be neglected in multimedia applications [8], depends on factors such as the hardware architecture or the specific application implementation. Because of this, in Eq. (16) it is left as a function  $f$  of the processing loads  $l_j$  of the applications. Additionally, we also consider that there is a cost  $\varphi$  incurred every time a node is activated. This cost is related to the amount of energy that the node needs to wake up from the sleep mode. We also allow moving applications from one node to another as long as all the restrictions described previously are fulfilled. Nevertheless, we assume that moving an application has a cost  $\delta$  due to the impact of receiving the application bytecode in the new node. With this, the energy constraints that must be ensured in every node are:

$$\begin{aligned} & \varphi \cdot (x_i - X_i) \cdot x_i + \delta \sum_{j \in A} \sum_{k \in T_j} (y_{ijk} - Y_{ijk}) \cdot y_{ijk} + \\ & \sum_{j \in A} P_{ij}^t \Delta \tau_j + \sum_{j \in A} P_{ij}^r \Delta \tau_j + f \left( \sum_{j \in A} \sum_{k \in T_j} y_{ijk} l_j \right) \Delta \tau_j + \lambda_i \\ & = E_i(\tau^*) \quad \lambda_i \geq 0 \quad \forall i \in S' \end{aligned} \quad (16)$$

where  $X_i$  is a constant equal to 1 if the node  $i$  was active before the arrival of the new application, and 0 otherwise.  $Y_{ijk}$  is a constant equal to 1 if the test point  $k$  of the application  $j$  was being sensed in the node  $i$  just before the arrival of the new application, and 0 otherwise.  $\Delta \tau_j$  is the remaining lifetime of application  $j$  at time instant  $\tau^*$  ( $\Delta \tau_j = \tau_j + \epsilon_j - \tau^*$ ),  $E_i(\tau^*)$  is the remaining energy that node  $i$  has at  $\tau^*$ , and  $\lambda_i$  is a variable indicating the residual energy that node  $i$  would have once the lifetime of the applications deployed in it or forwarded by it expires. We assume the sinks do not have energy constraints since they can be plugged directly into the grid.

If the solution space described by those restrictions is null, then the new application cannot be deployed ensuring the presence of the previous applications and therefore the system rejects it. If the solution space contains several feasible solutions, we should select the solution that maximizes the capacity of the system of accepting future applications. To do so, we proposed three possible objective functions for the optimization problem: The first one, denoted as *Total* is to maximize the total residual energy of the network (17). The second one, denoted as *Max-min* is to maximize the residual energy of the node with the lowest energy (18). Finally, we also consider a weighted sum of the two previous alternatives (19), denoted as *Mixed*:

$$\max \sum_{i \in S'} \lambda_i \quad (17)$$

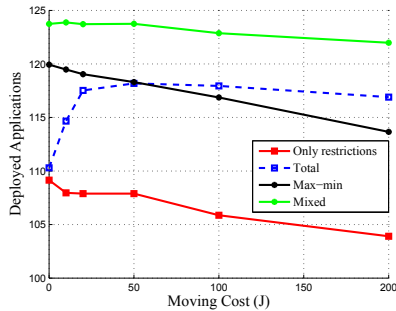
$$\max \lambda \quad \lambda \leq \lambda_i \quad \forall i \in S' \quad (18)$$

$$\max \left( \lambda + \frac{1}{|S'|} \sum_{i \in S'} \lambda_i \right) \quad (19)$$

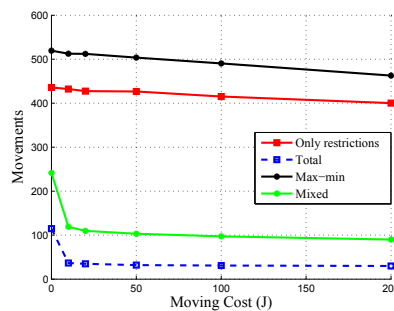
### III. PERFORMANCE EVALUATION

As a reference, we have focused on *multimedia* applications, which require the sensing, processing and delivery of multimedia content, specifically, in visual sensor networks, i.e. WSNs designed to perform visual analysis [8]. Based on the characterization of that work, the requirements vector  $r_j = \{12 \text{ kb/s}, 842 \text{ KB}, 69.23 \text{ MIPS}\}$  is used to represent the applications and the associated power dissipation (function  $f$  in eq. (16)) is 0.2 W. To support these applications, we consider *high-level* sensor node hardware. The parameters have been derived by taking as a reference BeagleBone platforms [9] equipped with an IEEE 802.15.4 radio and a low-power USB camera. The resource vector is  $o_i = \{250 \text{ kb/s}, 256 \text{ MB}, 720 \text{ MIPS}, 32400 \text{ J}\}$  assuming 2 AA batteries for all the nodes except sinks, which can be plugged directly into the grid. Results have been obtained by solving the optimization model with CPLEX software [10], averaging the outcome over 100 random instances of the scenario. In each instance, 200 visual applications are generated in a scenario with 36 BeagleBone nodes according to a Poisson process with rate of 1 application per hour and a constant lifetime  $\epsilon_j$  of 5 hours. The number of test points per application is 3,  $N_{i,j} = 1$  and there are 2 sink nodes. Nodes are deployed in a  $141 \times 141$  m scenario. The sensing range is  $R_i^s = 40$  m. A path loss model with  $\gamma = 4$  and  $g_0 = 8.1 \cdot 10^{-3}$  is considered.  $P_{max} = -10$  dBm,  $\alpha = -92$  dBm and  $\beta = -104$  dBm.

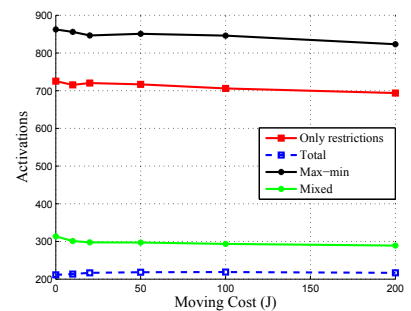
In addition to the three proposed objective functions, results are also presented for the case where no objective function is considered (*Only restrictions*). Fig. 1(a) shows that the total number of deployed applications is always higher for the *Mixed* approach; as expected the worst results are obtained with the *Only restrictions* approach and finally depending on the value of the moving cost, *Total* and *Max-min* strategies outperform each other. For the *Total* strategy, initially, as the moving cost increases, the number of deployed applications rises. This can be explained as follows: when the moving cost is low and a new application arrives at the system, the model prefers to move one current application from one active node to another, rather than activating a new node. This makes new applications tend to be located in the already activated nodes, leading to these nodes running out of energy faster. In the end, this makes the network disjoint and reduces the number of deployed applications. Figs. 1(b) and 1(c) show that *Only restrictions* and *Max-Min* are the strategies with more movements and activations. This is straightforward for *Only restrictions*, since applications are deployed without any additional objective rather than fulfilling the constraints, and therefore the nodes where the applications are deployed are more randomly chosen. A similar explanation can be applied to the *Max-Min* strategy: eq. (18) only takes care of the node with the lowest energy, so the remaining nodes can be activated or receive an application without any penalty in the objective function. Finally, it must be noted (Fig. 1(c)) that the number of activations remains almost constant (and not rises) when



(a)

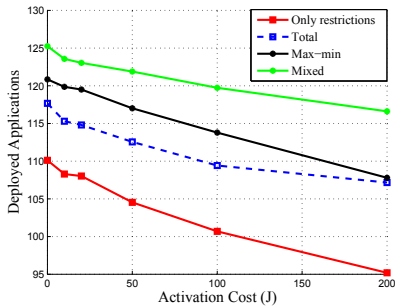


(b)

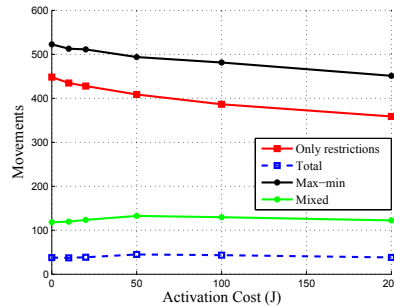


(c)

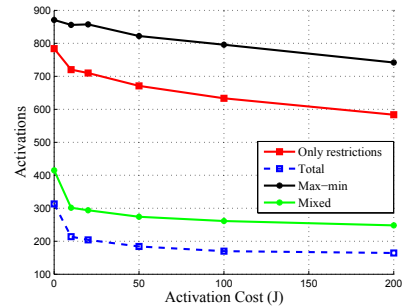
Fig. 1: Impact of moving cost  $\delta$ . a) Deployed applications. b) Number of movements c) Number of activations.  $\varphi = 10$  J.



(a)



(b)



(c)

Fig. 2: Impact of activation cost  $\varphi$ . a) Deployed applications. b) Number of movements c) Number of activations.  $\delta = 10$  J.

the movement cost increases. This is because an activation of a new node also implies that this node has to receive the bytecode of the application, so the higher moving cost cannot be compensated by activating more nodes. Fig. 2(a) shows that the *Mixed* strategy keeps providing the best performance in terms of deployed applications for different values of the activation cost. Again, for the same reasons explained above, *Only restrictions* and *Max-Min* are the strategies that have more movements and activations (Figs 2(b) and 2(c)). In addition, it is worth noting that for *Mixed* and *Total* approaches, the number of movements increases as the activation cost rises to minimize the energy consumption in the nodes.

#### IV. CONCLUSION

In this paper we have analyzed how to dynamically allocate the resources of a shared sensor network to multiple applications. Namely, the proposed optimization framework accounts for constraints on the sensor nodes capabilities and network limitations, including additional energy consumption related to resource re-allocation. Different alternatives related to the residual node energy have been analyzed as objective function: total residual energy, max-min of node energy and a mix of both metrics. The results obtained for realistic network scenarios show that the mixed metric provides the best performance in terms of number of deployable applications.

#### ACKNOWLEDGMENT

This work has been supported by the Spanish Government through the grant TEC2014-52969-R from the Ministerio de Ciencia e Innovación (MICINN), Gobierno de Aragón (research group T98), the European Social Fund (ESF), Centro

Universitario de la Defensa through project CUD2013-05, Universidad de Zaragoza, Fundación Bancaria Ibercaja and Fundación CAI (IT 2/15). This work has also been partially supported by the Italian Ministry for Education, University and Research (MIUR) through the national cluster project SHELL, Smart Living technologies (grant number: CTN01 00128 111357).

#### REFERENCES

- [1] I. Khan, F. Belqasmi, R. Glioth, N. Crespi, M. Morrow, and P. Polakos, "Wireless sensor network virtualization: Early architecture and research perspectives," *IEEE Network*, vol. 29, pp. 104 – 112, may-jun. 2015.
- [2] S. Madria, V. Kumar, and R. Dalvi, "Sensor cloud: A cloud of virtual sensors," *IEEE Software*, vol. 31, no. 2, pp. 70 – 77, mar.-apr. 2014.
- [3] Y. Xu, A. Saifullah, Y. Chen, C. Lu, and S. Bhattacharya, "Near optimal multi-application allocation in shared sensor networks," in *Proceedings of the Eleventh ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '10, 2010, pp. 181–190.
- [4] D. Zeng, P. Li, S. Guo, T. Miyazaki, J. Hu, and Y. Xiang, "Energy minimization in multi-task software-defined sensor networks," *Computers, IEEE Transactions on*, vol. 64, no. 11, pp. 3128–3139, Nov 2015.
- [5] C. Delgado, J. R. Gállego, M. Canales, J. Ortín, S. Bousnina, and M. Cesana, "An optimization framework for resource allocation in virtual sensor networks," in *IEEE GLOBECOM 2015, San Diego, CA, USA, December 6-10, 2015*, pp. 1–7.
- [6] C. Delgado, J. R. Gállego, M. Canales, J. Ortín, S. Bousnina, and M. Cesana, "On optimal resource allocation in virtual sensor networks," *Ad Hoc Networks*, vol. 50, pp. 23–40, 2016.
- [7] T. Hou, Y. Shi, and H. Sherali, "Rate allocation and network lifetime problems for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, pp. 321 – 334, 2008.
- [8] A. Redondi, L. Baroffio, L. Bianchi, M. Cesana, and M. Tagliasacchi, "Compress-then-analyze vs analyze-then-compress: what is best in visual sensor networks?" *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, pp. 1–1, 2016.
- [9] G. Coley, *Beaglebone rev a6 system reference manual*, 2012.
- [10] "ILOG CPLEX," <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>, 2015.