

# A Novel Cooperative Ant-based Clustering Approach for Wireless Multimedia Sensor Networks

Luis Cobo \*

*Fecha de recepción: 28 de febrero 2014*  
*Fecha de aprobación: 05 de marzo 2014*  
*Pp. 7 - 23*

## ABSTRACT

This Cluster has been well received as one of the most effective solutions to enhance energy efficiency and scalability of large-scale wireless sensor networks. However, it is rather challenging to conform a cohesive group of Clusters having permanent communication among Cluster heads. In this paper, a unique look at the Cluster head election problem is taken into account, specifically concentrating on applications in which the formation and maintenance of a virtual backbone among Cluster heads is the main requirement. Our approach for Cluster-based network organization is based on the use of collective social agents in order to lead the formation of these Clusters. Our algorithm uses ants to choose good Cluster head candidates and to create a virtual backbone among these router nodes.

## KEY WORDS

Wireless Sensor Networks, Clustering Algorithms, Ant Colony Optimization.

\* *Ph. D. en Ingeniería, Universidad de los Andes y en Génie informatique, École Polytechnique de Montréal. Magister en Ingeniería de Sistemas y Computación, Universidad de los Andes.*

# UN NUEVO ENFOQUE COOPERATIVO BASADO EN EL PARADIGMA DE COLONIAS DE HORMIGAS PARA LA FORMACIÓN DE GRUPOS EN REDES DE SENSORES INALÁMBRICOS MULTIMEDIA

## RESUMEN

*El proceso de armar grupos de sensores inalámbricos (Clustering en inglés) ha demostrado ser una de las soluciones más efectivas para mejorar el consumo de energía y la escalabilidad en redes de sensores inalámbricos de gran tamaño. Sin embargo, problemas como determinar el número apropiado de Clusters y que además haya una comunicación permanente y confiable entre las cabezas de Cluster (jefes de grupo), no han sido satisfactoriamente solucionados. En este artículo, nos adentramos de forma especial al problema de elección de la cabeza de Clusters y del mantenimiento de un Backbone virtual en la red, que permita una comunicación permanente entre estas cabezas de Cluster y el Sink principal de la red de sensores. Nuestro enfoque está basado en la utilización de una serie especial de agentes (que llamamos hormigas) que permiten guiar la construcción y formación de estos Clusters. Nuestro algoritmo utiliza estas hormigas para seleccionar los mejores candidatos a cabeza de cluster (que se convertirán en una especie de enrutadores en la red) y para crear el Backbone virtual entre estos nodos enrutadores. Este algoritmo converge muy rápidamente y utiliza muy pocos recursos de los nodos que conforman la red. Las simulaciones realizadas muestran que el algoritmo logra una muy buena distribución de los Clusters y es mejor que otro algoritmo muy conocido que realiza la misma tarea.*

## PALABRAS CLAVE

*Redes Inalámbricas de Sensores, Agrupación de Algoritmos, Optimización con colonia de hormigas.*

# *Une nouvelle approche coopérative utilisant les fourmis et basée sur des capteurs sans fil connectés en réseau*

## RÉSUMÉ

*Cet article approfondit la question du choix du leader et le maintien d'une ossature virtuelle du réseau permettant une communication permanente entre les leaders et le capteur principal du réseau. L'approche est basée sur l'utilisation d'une série d'agents (fourmis) qui permettent de guider la construction et de la formation de la colonie. L'algorithme utilise ces fourmis pour sélectionner les meilleurs candidats au rôle de leader de la colonie (qui sera une sorte de routeurs sur le réseau) et créer la colonne vertébrale virtuelle entre ces routeurs.*

## MOTS CLÉS

*Réseau de capteurs multimédia sans fil, algorithmes, colonie de fourmi, optimisation.*

## ***Nova abordagem de Clustering cooperativo baseada em formigas para redes de sensores multimídia sem fios***

### **RESUMO**

*Este artigo faz ênfase no problema da escolha da cabeça de Clusters e da manutenção de um Backbone virtual na rede, que permita uma comunicação permanente entre estas cabeças de Cluster e o Sink principal da rede de sensores. A abordagem está baseada na utilização de uma série especial de agentes (formigas) que permitem guiar a construção e formação destes Clusters. O algoritmo utiliza estas formigas para seleccionar os melhores candidatos a cabeça de Cluster (que se tornarão em uma espécie de roteadores na rede) e para criar o Backbone virtual entre estes nodos roteadores.*

### **PALAVRAS-CHAVE:**

*Redes de sensores multimídia sem fios, algoritmos de Clustering, otimização por colônia de formigas.*

# 1. Introduction

---

The availability of inexpensive hardware such as CMOS cameras and microphones that are able to ubiquitously capture multimedia content from the environment has fostered the development of Wireless Multimedia Sensor Networks (WMSNs), i.e., networks of wirelessly interconnected devices that allow retrieving video and audio streams, still images, and scalar sensor data (Akyildiz, Melodia & Chowdhury, 2008). With rapid improvements and miniaturization in hardware, a single sensor device can be equipped with audio and visual information collection modules. Even though, there have been significant advances in recent years to improve these wireless sensor nodes, more energy-efficient solutions are required within the communication stack and middleware for the conservation of the battery power. Within the communication stack, an approach that is likely to succeed is the use of a hierarchical structure for routing (Akkaya & Younis, 2005).

Clustering with data aggregation is an important technique in this direction. Even though many protocols have been proposed in the literature to minimize energy dissipation on the forwarding paths, some nodes may still be drained quickly. By employing a dynamic Clustering technique, these nodes are rotated dynamically (Younis & Fahmy, 2006). Various Clustering techniques in different contexts have been proposed. Most algorithms aim at generating the minimum number of Clusters and transmission distance. But most of them forget an important issue: permanent connection between Clusters. In that way, we can obtain reliability and robustness in the routes between nodes and sink.

In this paper, we propose to biologically inspire the Clustering approach, whereby the network is Clustered around certain nodes deemed biologically fit. This work is based on the TANT algorithm (Selvakennedy, Sinnappan & Shang, 2009), but we have made improvements in order to create a virtual backbone among the CHs (Cluster heads) and to select the best nodes as CH candidates. According to this approach, when a node has a special agent, it selects itself to become a CH. Such election obviates the need to maintain many state variables, which is common in other Clustering algorithms. A fixed number of such agents are used

to ensure that a certain number of Clusters are formed throughout the network useful life, and these Clusters are selected in such a way as to maximize the network lifetime and to get a permanent connection among them.

## 2. Clustering algorithm

---

For large scale sensor networks that may have thousands of nodes, it is more realistic to have hierarchical models of network rather than flat ones. In that way, we can get scalability in our network. Nevertheless, scalability is not the only reason to Cluster the network. This process also allows improving network data aggregation mechanisms, while concentrating this activity in the CH, consequently reducing node workloads, saving energy and increasing the network lifetime. Arboleda & Nasser (2006) present other advantages of Clustering that apply to this novel protocol: the fact that only the CH transmits information out of the Cluster helps prevent collisions between the sensors inside the Cluster, as they do not have to share communication channels with nodes in other Clusters. This also promotes energy savings and avoids the black hole problem. Latency is also reduced. Although data must hop from one CH to another, they cover larger distances than when sensors use a multi-hop communication model (non-Clustered) as the one used in other protocols.

Finally, Clustering is applied in order to take advantage of the existence of nodes of different abilities inside a WMSN. Table 1 (Misra, Reisslein, & Guoliang, 2008) presents the processing performance and memory capacities among standard (TelosB) and multimedia sensors. Table 1 shows that the memory and processing capacities of multimedia sensors are superior to those of conventional sensors. That is the reason for selecting multimedia sensors to become the network CHs. This novel algorithm will be designed to favor the selection of these nodes as CHs. Our Clustering algorithm aims at achieving the following goals:

- Saving network resources by encouraging the selection of resource-rich nodes (multimedia sensor nodes) as network CHs.
- Ensuring network connectivity by forming a virtual backbone among the different CHs. Each CH is in the radio range transmission of at least one other CH. Communication between two CHs is direct (there are no relay nodes between them).

- Maximizing network lifetime by implementing a mechanism of CH rotation.

**Table 1.** Abilities of Video and Standard Sensors

	<b>Stargate</b>	<b>Samsung S3C44B0X</b>	<b>Telos B</b>
Clock Frequency	200/300/400 Mhz	66 Mhz	8 Mhz
Architecture	32 bit RISC	16/32 bit RISC	16 bit RISC
Memory	64 MB SDRAM 32 MB Flash	256 MB	10 KB 1 MB Flas
Cache	32 KB data 32 KB instruction	8 KB	Data not available
Cost (US\$)	595	500	100

**Source.** Misra, Reisslein & Guoliang (2008)

With a virtual backbone in the network, only CHs are concerned with data transportation, and other nodes are free to pursue their sensing tasks. Such task sharing improves network performance with respect to routing overhead and, moreover, a smaller number of nodes need to be alert for data transportation. This procedure reduces energy consumption, thus simultaneously maximizing network lifetime.

## 2.1 Information update phase

In the same way to the protocol T-ANT (Selvakennedy, Sinnappan & Shang, 2009), our Clustering protocol uses a collection of agents to form Clusters in a sensor network. It is completely distributed and completed in constant time. These are some of the reasons why this algorithm was selected.

As in T-ANT, in our Clustering algorithm, the Clustering operations are split into rounds. Each round comprises a Cluster setup phase and a steady phase. In the steady phase of the algorithm, data transmission takes place between sensors and the sink. A number of timers are used to control the process operations. During the Cluster setup phase, CHs are elected and Clusters are placed around them. In order to avoid the maintenance of many state variables, as one finds in numerous current Clustering proposals, a series of agents (known as Cluster-ants or CANTS) are used to control CH elections. A node with a CANT becomes a CH, whereas others choose to join the best Cluster in range.

The Cluster radius  $R_{\text{Cluster}}$  is defined as a tunable parameter that determines the minimum distance between any two CH nodes in the network. The value of this parameter always remains inferior to the sensor communication radio range (called  $r$ ). Before the Cluster setup phase, an information update phase is carried out by the sensors. Each sensor node broadcasts a HELLO packet with information regarding its ID, its Clustering pheromone value ( $\Phi_c(n)$ ) and its state to its neighbors. When a HELLO packet arrives, the node stores such information in a table, the neighborhood or the neighbor's information table. This table is then used to select Clusters, to join a Cluster and to route data packets.

The Clustering pheromone value determines whether it is appropriate for this node to become a CH. For each node, this value is calculated using the following formula:

$$\Phi_c(n) = m(n)^a \cdot e(n)^b \quad (1)$$

Where  $m(n)$  denotes the available memory in the node,  $e(n)$  is the residual ratio of the node's energy and  $a$  and  $b$  denote the importance of each component of the pheromone:  $a$  for the memory capacity and  $b$  for the energy component. Thus, the network administrator can establish which component is most important when selecting a CH, namely, memory or energy or both. The state indicates if the node is a CH, a member of a Cluster or neither. These HELLO packets are constantly broadcast by the nodes throughout their lifetime.

## 2.2 Ant release phase

After the information update phase, the sink releases a fixed number of ants (i.e. control messages) into the network. Assuming that the terrain is square  $M \times M$ , the number of ants to be released is set at:

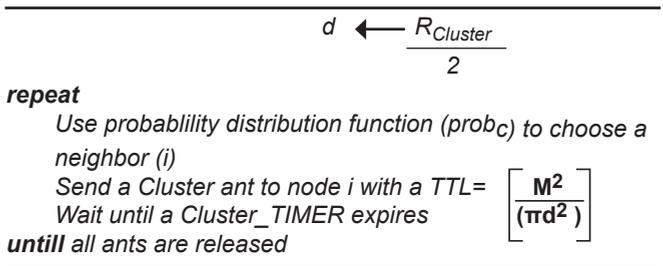
$$\left[ \frac{M^2}{(\pi d^2)} \right]$$

Where  $d$  depicts half of  $R_{Cluster}$ . The latter formula also represents the number of Clusters that make up the network. Attempts are made to obtain complete coverage of the area with this number of Clusters, where every node belongs to a Cluster and the CHs are disseminated throughout the terrain. Ants move about the network in a random fashion, as far as they can, respecting the limits imposed by their Time-To-Live (TTL) values. The TTL value equals the number of ants. Hence, an ant can visit a large number of candidate nodes to become a CH before they die. When the sink releases an ant, it chooses one of its neighbors randomly according to the following probability distribution function:

$$prob_c(j) = \frac{(\Phi_c(j))}{\sum_{(i \in N_s)} \Phi_c(i)}$$

Where  $\Phi_c(j)$  denotes the Clustering pheromone value sent by node  $j$ , as defined in equation (1), and  $N_s$  represents the set of all of the sink neighbors located at a distance of at least  $R_{Cluster}$ . Before releasing the next Cluster ant, the sink waits for a timer to expire (Cluster\_timer). Although the timer expiration is set at a random value, it always remains proportional to the delay of sending an ant from a node to a neighbor. The objective of this timer is to ensure that the ants' subsequent transmissions do not self- interfere. Aside from that, when the sink selects a neighbor, the pheromone value of that node is artificially decreased, in order to avoid choosing the same set of nodes repeatedly.

**Figure 1.** Tasks developed by the sink.



**Source.** Developed by the author.

It presents the algorithm with the tasks performed by the sink in order to start the Clustering process. When an ant arrives at a node, that node will execute the tasks.

**Figure 2.** Tasks developed by nodes distinct to the sink

---

```

if an ant arrives at node i then
  if node i is not a CH then
    if there is a CH in the radius  $R_{Cluster}$  Then
      Pick a random CH neighbor
      Send the ant to it
    else
      Store the ant [ This node is a CH]
      Broadcast a message ADV_Cluster to neighbors
      in range  $R_{Cluster}$ 
    end if
  else if node i is a CH then
    Decrement the TTL of the ant
    if TTL > 0 then
      Pick a random neighbor according to the probability
      function  $prob_C$ 
      Send the Cluster ant to it
    else
      Destroy the ant
    end if
  end if
end if

```

---

**Source.** Developed by the author.

The algorithm shows that, in order to become a CH, the selected node must have received a Cluster ant from another CH (or the sink) located at a distance  $R_{Cluster}$  from it.  $R_{Cluster}$  was previously defined as the minimal distance between two CHs. Hence, at the moment of selecting the following neighbor, the node reads its neighbors' information table and selects, with probability  $prob_C$ , a node whose distance is a minimum of  $R_{Cluster}$  in a random manner. The reason why a CH elects the next CH is to create a virtual backbone between the various CHs, a direct communication strategy between them. This backbone will facilitate the task of routing accomplished by the routing protocol which is in charge of delivering packages between nodes and the sinking in the network. When a node becomes a CH, it broadcasts an *adv\_Cluster* message to advise its neighborhood of its new condition. It also changes the value of field state of a HELLO package subsequently sent by the node. Once a regular node receives an *adv\_Cluster* message from a CH located at a distance

below  $R_{\text{Cluster}}$ , it stores the corresponding information that pertains to that CH. This information is later used to join a given Cluster. Contrary to other proposals documented in the literature, this CH election approach has a very small constant time and a low level of complexity.

An ant's TTL indicates the maximum number of hops that it can perform. The CH pulverizes an ant once its TTL reaches the value zero. This situation shows the existence of a superfluous number of Clusters in the network, and the Cluster ant is destroyed in order to avoid the creation of new Clusters that would hinder the network.

The actual Clustering process happens once another timer expires. A regular node decides to join a Cluster when its `join_timer` expires. This node chooses the nearest Cluster to join (from all of the `adv_Cluster` packages it received) by sending a join message with its ID. When a CH receives join messages, it stores such information in order to subsequently select a Cluster member as a new CH. If a regular node has never received an `adv_Cluster` package from a CH, it starts a `join_timer` once again and repeats the latter process until this timer expires. However, if in the process, it receives neither an `adv_Cluster` message nor a HELLO package from a CH, the node uses the nearest neighboring Cluster member as a bridge to reach its CH. The properties of the proposed Clustering algorithm can be highlighted as follows:

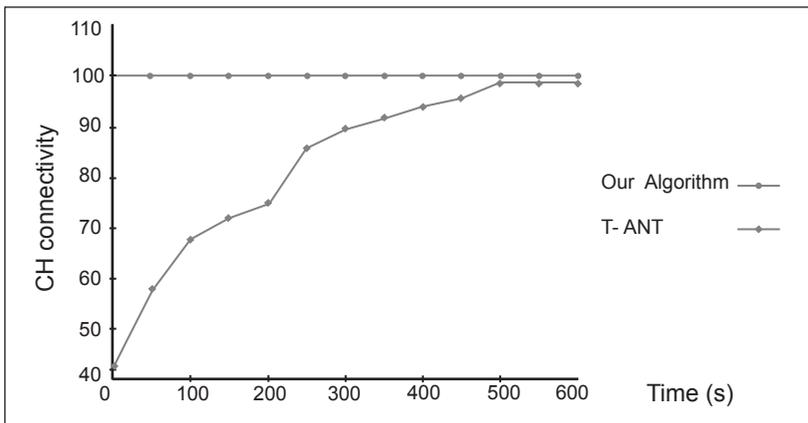
- The algorithm is completely distributed. A node locally decides to become a CH if an ant reaches it or joins a Cluster.
- Given the absence of looping statements as a function of node quantity, it is clear that the election process has an  $O(1)$  time complexity.
- The algorithm ensures the creation of a backbone among the CHs. As all CHs are connected, paths to a sink can be easily discovered.

### 3. Experimental results

In order to evaluate our Clustering process, NS-2 (VINT, 2013) was used to implement and simulate the novel algorithm. For this simulations, we consider there are two types of nodes: scalars and multimedia (with more energy and memory than scalar nodes). Half of the nodes are multimedia. The radio range of the nodes span 100 meters and the data rate equals 2 Mbit/s. At the MAC layer, a modified version of 802.11b DCF protocol was used. The modification was made in the queue politics of the MAC protocol in order to accept multi-class and multi-priority traffic.

It was assumed that 400 sensor nodes were distributed randomly over a square area of 400 m x 400 m. This scenario was executed during 600 seconds. In order to benchmark this new protocol, it was decided to compare it to T-ANT (Selvakennedy, Sinnappan & Shang, 2009), as it was the base of our Clustering protocol and also since it outperformed other well-known Clustering algorithms, such as LEACH (Heinzelman, Chandrakasan & Balakrishnan, 2002) and HEED (Younis & Fahmy, 2004). For this experiment, a  $R_{Cluster} = 60m$  is assumed, and the CH rotation scheme is the same as in T-ANT: there are multiple rounds in the network lifetime, and in each round, a CH rotation is carried out. The CH finds its Cluster member whose level of pheromone, equation (1), is the highest, before it becomes the CH for the next round.

**Figure 3.** The CH connectivity at various simulation time for our Clustering algorithm.

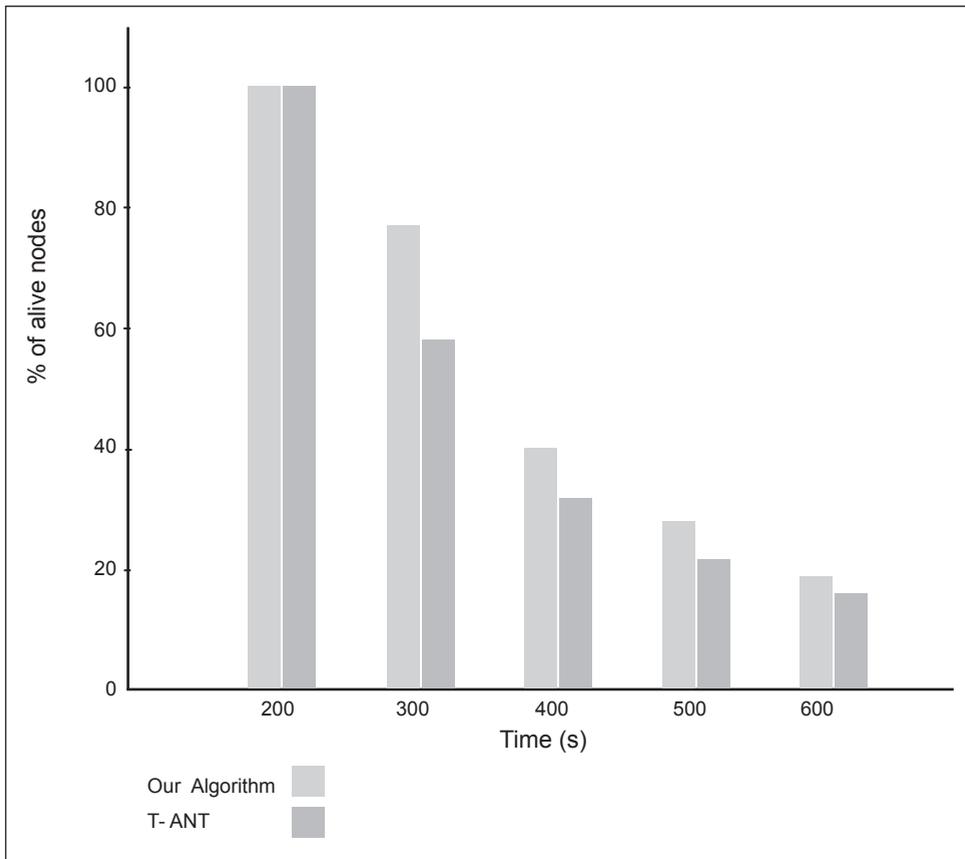


**Source.** Developed by the author.

Figure 3 depicts the CH connectivity of these Clustering algorithms at different simulation time. This property indicates if there is direct communication between the CHs of the network, meaning that no CH is isolated. This property is very important in this novel algorithm, as all of the traffic between source nodes and the sink is transported by the CH. If CHs are isolated, it is impossible to transmit information from that Cluster to the sink. In this simulation, any node can be a CH, in other words we set  $a = b = 0$  in equation (1). Observe that after only 20 rounds (one round 20s each), the connectivity of T-ANT is acceptable. Meanwhile, the connectivity of our algorithm remains at a steady 100% from the beginning of the simulation. The main design goal of our Clustering algorithm is reached with the permanent connectivity of the CHs.

In Figure 4, the improvement gained through our Clustering algorithm is further exemplified by the network lifetime graph. The network lifetime is defined as the time the first node in the network has a depleted battery. For this experiment, the memory component in the Clustering pheromone formula (parameter  $a$  in the equation (1)) was set to 0 (zero) and the energy component (parameter  $b$  in the equation (1)) was set to 1. This way, energy rich sensors have greater probabilities of becoming a CH.

Moreover, a Constant Bit Rate (CBR) traffic source was used to generate data traffic of 32-byte packets. All regular nodes sent the sink a packet/second on average. Non sending nodes fall into a sleep mode. Five simulations were carried out, where the value of the energy component varied. The initial energy of the scalar nodes was 0.1J and for the multimedia nodes, this initial energy was 0.5J in order to let the nodes disappear sooner. However, this does not change the behavior pattern of these protocols. It is clear that our algorithm exhibits the longest lifetime with all nodes remaining fully functional. Test results show that our algorithm achieves more than twice the lifetime of T-ANT, which can be explained by the fact that the CHs never sleep, making them the first candidates to get their battery depleted. As the algorithm mainly chooses energy-rich nodes as CHs, the network lifetime is longer. T-ANT does not differentiate between scalar and multimedia nodes, and that is the reason of the poor performance of this algorithm.

**Figure 4.** Network lifetime vs simulation time for T-ANT and our Clustering algorithm

**Source.** Developed by the author.

## 4. Conclusion

---

We have presented an ant-based Clustering algorithm for Wireless Multimedia Sensor Networks, with heterogeneous nodes. Our Clustering algorithm selects Cluster heads based on its pheromone value. This pheromone depends on the residual energy and the other parameters, customizable by the network administrator. Besides that, our Clustering algorithm uses special agents (ants) to guide the selection of CHs in a totally distributed manner. In comparison with T-ANT (Selvakennedy, Sinnappan & Shang, 2009), another ant-based Clustering algorithm, our Clustering process achieves a permanent CH connection with lower energy costs.

Our future work will include mechanisms for CH rotation, in order to get a bigger network lifetime.

## 5. References

---

- Akkaya, K. & Younis, M. (2005). *A survey on routing protocols for wireless sensor networks*. *Ad Hoc Networks*, 3(3), 325-349.
- Akyildiz, I. F., Melodia, T. & Chowdhury, K. R. (2008). *A survey on wireless multimedia sensor networks*. *Computer Networks*, 51(4), 921-960.
- Almalkawi, I. T., Guerrero-Zapata, M., Al-Karaki, J. N., Morillo-Pozo, J., Almalkawi, I. T., & Zapata, M. G. (2010). *Wireless multimedia sensor networks: current trends and future directions*. *Sensors*, 10(7), 6662.
- Arboleda, L. M., & Nasser, N. (2006). *Comparison of Clustering algorithms and protocols for wireless sensor networks*. *Canadian Conference on Electrical and Computer Engineering*, (pp. 1787-1797). Vancouver.
- Heinzelman, W. B., Chandrakasan, A. P. & Balakrishnan, H. (2002). *An application-specific protocol architecture for wireless microsensor networks*. *IEEE Transactions on Wireless Communications*, 1(4), 660-670.
- Misra, S., Reisslein, M. & Guoliang, X. (2008). *A survey of multimedia streaming in wireless sensor networks*. *IEEE Communications Surveys and Tutorials*, 10(4), 18-39.
- Selvakennedy, S., Sinnappan, S. & Shang, Y. (2009). *A biologically-inspired Clustering protocol for wireless sensor networks*. *Computer Communications*, 30(14-15), 2786–2801.
- VINT. (2013, May 11). *NS-2 The Network Simulator*. Retrieved from <http://www.isi.edu/nsnam/ns/>
- Younis, O. & Fahmy, S. (2004). *HEED: a hybrid, energy-efficient, distributed Clustering approach for ad hoc sensor networks*. *IEEE Transactions on Mobile Computing*, 3(4), 366-379.
- Younis, O. & Fahmy, S. (2006). *HEED: a hybrid, energy-efficient, distributed Clustering approach for ad hoc sensor networks*. *IEEE Transactions on Mobile Computing*, 3(4), 366-379.