A Collaborative and Energy Efficient Data Routing Algorithm for Wireless Sensor Networks Through the Use of Game Theory

Alexandros Schillings Kun Yang
Department of Electronic Systems Engineering, University of Essex

Abstract – The proposed algorithm induces an energy-aware and efficient collaborative behaviour to the nodes using sensor centric information, by making them aware of their interdependency, without compromising the main purpose of the network - the collection of information.

1. Introduction

A Wireless Sensor Network (WSN) is composed of a collection of wireless nodes that are designed to monitor, store and report phenomena, usually, with minimal human interaction [1]. They are usually deployed in as part of a set-and-forget strategy where an operator is only needed to collect the data. Energy-aware data (including the initial data query) routing is essential to any WSN and has attracted enormous efforts from the community. Inspired by the similar nature of games as WSNs, this paper proposes the use of game theory to tackle this challenge. Game theory is a framework that allows the modelling of multiparty decision problems and is increasingly attracting more attention as a mechanism to solve various problems in wireless networks [2][3]. Rajgopal Kannan *et al* [2] propose an algorithm that induces the formation of a maximally reliable data aggregation tree, while A. Urpi *et al* [3] model the collaboration of selfish wireless nodes. Both of these approaches aim to the development of a model in which nodes cooperate in an energy aware environment.

This paper proposes an approach in which nodes are the players and their data and energy represent the resources over which they compete. Nodes are aware that their actions and choices affect nodes that are located upstream and also understand that those upstream nodes are linking them with the sink. This creates a drive to preserve those nodes as without them, a downstream node will be unable to perform its task. For this, we utilise a query driven methodology to accomplish the extraction of data from the network. The query is broadcast from the sink to the direction of the area we are interested in (or is simply restricted by a Time To Live (TTL) field) and is then expected to induce the creation of a tree in which the nodes will cooperate to route information.

The rest of this paper is organised as follows. Section 2 will introduce the game theoretic concepts that are used for this algorithm and will formulate the problem. Section 3 contains the description of the algorithms functionality along with their corresponding Game Theoretic formulas while Section 4 discusses the modelling results. Finally Section 5 contains the conclusion and possible future work.

2. Preliminaries

In this paper, the following is assumed for the WSN: *a)* The sensor nodes are homogeneous and all have a limited power supply. *b)* The sink is for simplicity assumed to have infinite power supply. *c)* The transmission range is fixed and is small enough so that most nodes will be unable to reach the sink without hoping at least once. *d)* The nodes have a general idea of their position in the network and the position of the sink.

In Game Theory, games are strategic situations that are defined and formulated as mathematical objects. A game is formed when a set of players formulates a set of possible moves (known as strategies) along with a number of functions that are used to motivate the players (payoff functions) [4]. Thus the problem can be formally formulated as follows: given a normal form game $G=\{S_i, ... S_n; u_{1,}.., u_n\}$ representing the network nodes that are reached by a query, where S_n are the strategy spaces that each node can select and u_n then corresponding payoff functions, we need to formulate the payoff functions in a way that will help the node n select a strategy s_n that represents the best response to the strategies selected by the other $n-1$ nodes. The resulting strategy profile $s=(s_{1,}..s_n)$ places the nodes responding to the query in a Nash Equilibrium [4]. The nature of the payoff functions and the subsequent selection of strategies will lead to the formation of s and also gives rise to the creation of a tree T rooted at the node originating the query (the sink n_q). The tree T is a subgraph of $F(N, L)$ (the super graph that contains all possible edges for all nodes (vertices) in the network) where each edge l_{ij} connects two nodes only if node n_i and n_j are within each others transmission range. As the network nodes are assumed to be homogeneous we can assume that if n_i can reach n_j the opposite is also true. The formation of T through the selected strategies should happen in a way that intelligently judges the energy consumption in the paths and takes notice of nodes with low remaining energy. Furthermore, the choices should be affected by the relative value of information of the ancestors of a given node, and the results of past strategies played by all relevant nodes.

3. Proposed Algorithm

Nodes are the basic building blocks that a WSN is composed of and each plays the dual role of both sensing

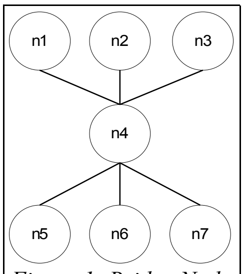

Figure 1: Bridge Node

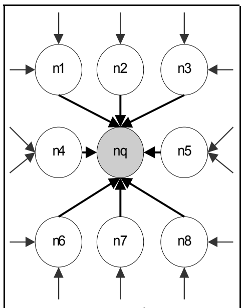

Figure 2: Sink Hotspot

and conveying information to the sink from other nodes. In order to fulfil these tasks, they need to stay alive for as long as possible by conserving energy. On the other hand, in order to convey the information, the nodes will need to expend energy and will normally need other nodes to act as liaisons between themselves and the sink, thus jeopardising their survival. This conflict of interest between the two main drives of a node (survival and purpose) makes Game Theory suitable for WSNs.

Nodes normally make decisions only concerning their next hop. This can cause problems as upstream nodes that will be contacted during the data upload are potentially more important to the network due to their position and place in the topology. This is illustrated in Figures 1 and 2 where nodes n_4 and n_1 to n_8 ,respectively, are bridging nodes that might lead to network segmentation if they fail. As such, theses nodes should be avoided unless necessary but due to the limited amount of control data that are transferred in a WSN, nodes might be unable to directly identify them.

In our algorithm, nodes are modelled having mixed strategy profiles which is a probability distribution $p_i=(p_{i1},\dots p_{iK})$ for node n_i where $0\leqslant p_{ik}\leqslant 1$ for $k=1,\dots,K$ and $p_{i1}+\dots+p_{iK}=1$. K is the number of pure strategies the node can select from ($S_i=s_{i1},\dots,s_{iK}$) and p_{ik} represents the probability that node n_i will elect to play strategy s_{ik} . This can be represented as a $m\,x\,2$ matrix l_i where m is the number of neighbours node n_i has. Each tuple contains a neighbouring node and the probability that it will be chosen for the next hop. It is assumed that the sensor node will want to transmit to only one neighbour. Also, in order to avoid routing loops a node is forbidden from routing to its ancestors. By not selecting the most probable strategy directly we avoid imposing a continuous drain to optimal paths.

Assuming that time is discrete and split into frames: $(t_1,\dots,t_n)$. At time $t_k\in\{t_1,\dots,t_n\}$, node n knows the following about the network [3]: *a)* It has a number of neighbour nodes equal to $M_n(t_k)$ which will remain fixed for this time frame. *b)* It has remaining energy equal to $B_n(t_k)$. *c)* $T_n^j(t_k)\forall j\in M_n(t_k)$ is the amount of traffic node n has generated in the frame and have been sent to node j . *d)* $F_n^j(t_k-1)\forall j\in M_n(t_k-1)$ is the number of packets node j forwarded for n in the previous time frame. During data transmission, node n_i will choose $S_i^j(t_k)$ packets to forward to node n_j , while $F_i(t_k-1)$ are the number of packets that node n_i has received in the previous time frame and will have to forward them now. The pay-off for this is shown in (1). This function is used to control and balance the drive for the node to transmit and forward packets through the network. In order to implement an energy aware protocol, another function is needed in order to ensure that the node will not overtax itself and will not commit to a course of action that will utilise the network's energy without providing sufficient benefits. This function is presented as (2). w is a weighting sent with the query from the sink to bias the node's decision in case the sink is especially interested in the results if the query and wants them at all costs. V represents the information value that the node is asked to relay upstream. In this paper we assume that V is additive (3). P are the ancestors of the node [2]. N_i^{crit} is a metric that represents the criticality of the node based on its distance to the sink (Figure 2) [5].

$$W_i(t_k)=\begin{cases}\dfrac{\sum_{(j\in M_i(t_k))}(F_j^i)}{(S_i(t_k-1)+F_i(t_k-1))}, if\ S_i(t_k-1)+F_i(t_k-1)>0\\ 0, otherwise\end{cases}\qquad(1)$$

$$O_i(t_k)=\begin{cases}\left(\left(B_i(t_k)+N_i^{crit}\right)*V\right)+w\,, if\ n_i\in T\\ 0, otherwise\end{cases}\qquad(2)$$

$$V_i=v_i+\sum_{j\in P(i)}V_j\qquad(3)$$

$$B_n(t_{proj})=m_n t_{proj}+b_n\qquad(4)$$

$$m_n=\frac{z(\sum t_z B_n(t_z))-(\sum t_z)(\sum B_n(t_z))}{z(\sum t_z^2)-(\sum t_z)^2}\qquad(5)$$

$$b_n=\frac{(\sum B_n(t_z))-m_n(\sum t_z)}{z}\qquad(6)$$

Finally, in order to preserve the nodes that that have limited energy, and provide feedback to the nodes in order to re-evaluate their strategies, we use the following linear regression functions in order to predict the average energy along with the average minimum energy (4) of a given path. Another reason for using regression is to keep track of the energy expenditure that happens due to other queries that might not reach a node as this should upset the expected values. As linear regression is based on past values, it will not apply till z samples have been taken and due to memory and processing constrains, only the last z samples will be kept.

Of course, although the energy expenditure of the nodes is not fully linear as it is affected by factors such as the data size, unpredictable node failure or introduction and battery physics, the regression analysis will provide a node with a rough estimate of both the efficiency of their strategies and the actions it will need to perform to compensate for the failing network. A different way of solving this problem of potential non-linearity is to calculate the coefficient γ_n and set a threshold above which the regression will begin to affect the strategies. The actual operation of the algorithm can be broken down to three steps:

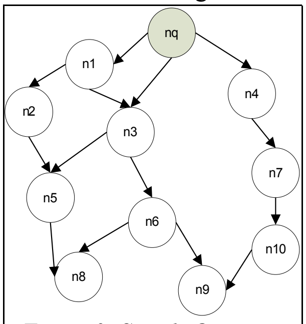

Figure 3: Step 1, Querying

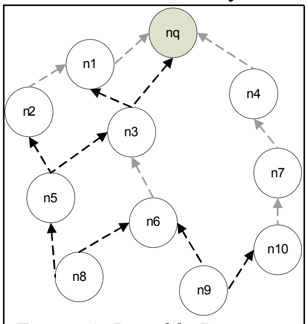

Figure 4: Possible Decisions

Step 1: The sink (n_q) initiates the process by transmitting a query towards the nodes that it is interested in. The query packet contains (along with the actual query) any weightings the sink will want to provide in order to bias the decisions of the nodes, an optional TTL value and the values needed for the payoff functions.(Figure 3). **Step 2:** Each node receiving the query waits a predetermined amount of time (in order to receive copies of the same query, travelling through different paths), updates the fields accordingly and transmits the updated query to all downstream neighbours that are closer to the target nodes/ area. **Step 3:** Once the target area is reached (or $TTL=0$) the nodes that received the query last are able to calculate the possible outcomes of choosing each of the possible upstream branches and they can estimate the values of the average and worst case upstream node. This will allow each node to calculate its strategy space S_i assign a probability p_{ik} to each strategy and finally make a selection (Figure 4). This procedure is repeated on each upstream node that will actually participate (thus forming T). In order to improve the energy awareness, the nodes keeps track of the energy changes that occur for each path through the use of regression (4). If a path seems to deplete its energy faster than others, then the probability p_{ik} that will lead to the selection of this path is adjusted accordingly.

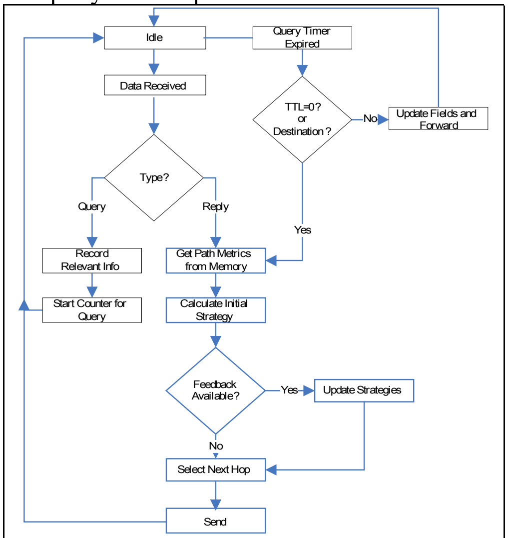

Figure 5: Simplified Protocol Flowchart

4. Performance Evaluation

For the modelling we used MATLAB 7.0 [7]. The area of the experiments was a square "arena" which was divided into LxL cells that contain n uniformly distributed sensor nodes (Figure 6). Each node n_i is assigned a random initial energy ($1 \leqslant E_i^{Init} \leqslant 100$) and a random information value ($1 \leqslant V_i \leqslant 100$), while all nodes are considered stationary. We consider t_{proj} for the nodes to be equal to the average of the time difference between the z samples taken, while the model used to calculate the energy consumption for transmission is the Heinzelman model proposed in [7]. Finally, each data point in the graphs represents the average of 30 simulations run with random seeds. Before each query, V is randomly reset in order to ensure that a different tree is formed.

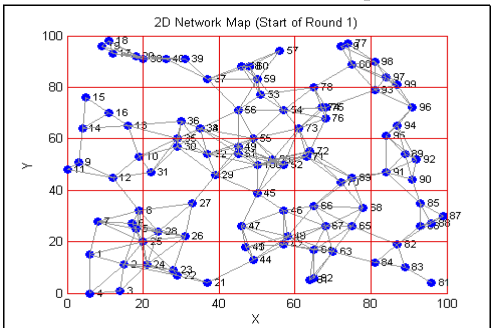

Figure 6: Sample 5x5 Network composed of 100 nodes. n_q is node number 100 at the centre.

Figure 7 shows a comparison between the proposed algorithm (GT), the Minimum Transmission Energy protocol (MTE), the Direct Connect (DC) approach and finally LEACH [7]. In LEACH, only Clusterheads are allowed to transmit and as they are elected in a semi-random way, it was expected that it would outperform the other algorithms. On the other hand, although LEACH is a considerably efficient algorithm, its assumption that all nodes are capable of communicating with the Sink directly, along with the inability of a user to request data, limit its application to very specific scenarios and is included only as a high-end benchmark. DC and MTE represent the two quintessential strategies in data reporting and although they are by their design quite limited in their scope, they are a common performance benchmark. As we can see from Figures 7, 8 and 9, both DC and MTE deplete their energy supplies quite early in the simulations. This is especially true for DC where although, it run for an average of 120 rounds without a node depleting its energy, from round 120 and onwards an average of 1 node per round failed. MTE lasted on average 60% longer than DC and had a less steep node death curve. Due to the

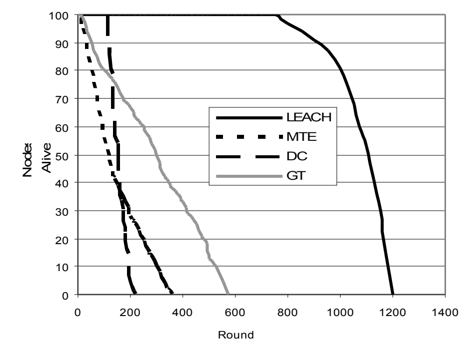

Figure 7: Total Network Lifetime

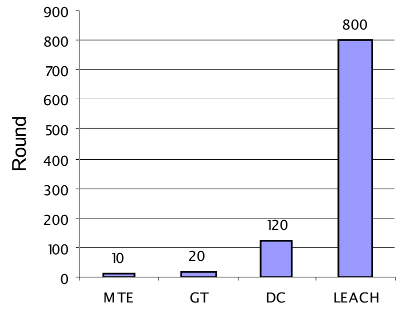

Figure 8: Round of First Node Death

fact that MTE is a multihop protocol, the less steep node death rate of 0.29% means that more paths will be open for the data to flow to the sink. Because of the intelligent selection of the next hop the proposed algorithm (GT) as a result of recognising network trends, it has a lifetime that is about 60% and 160% longer than MTE and DC respectively. This advantage can potentially be increased by optimising the selection of the t_{proj} value.

5. Conclusions and Future Work

In this paper we introduce an energy aware Game Theoretic algorithm that induces collaboration in WSNs. In order to achieve that, nodes are made to understand that if either their next hop neighbours or links in the paths behind those neighbours perish, then the nodes will not be able to perform their main purpose. Thus they are forced to rotate the selection of their next hop in a pseudo-random way by utilising the payoff functions in order to select one that will extend their functional life time, while looking after bridging nodes. Through the use of linear regression, the nodes can predict the results of their strategies and amend their strategy space accordingly so that the network can compensate for failing nodes or inefficient strategies. Currently the algorithm has no direct way of dealing with malicious or misbehaving nodes and that is something that needs to be implemented. Furthermore, the algorithm should be ported to a network simulator (such as ns2 [8]) in order to more closely examine and test its functionality in an environment more closely resembling a real world deployment so that its practical feasibility can be assessed.

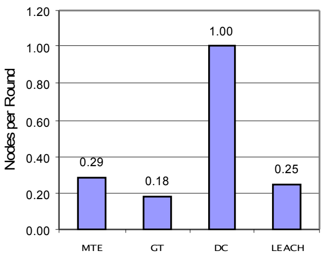

Figure 9: Average Rate of Node Death after the First Failure

References

[1] I. Akyildiz, W. Su, and Y. Sankarasubramaniam, '*A Survey on Sensor Networks*', IEEE Communication Magazine, August 2002, pp. 102-114

[2] Rajgopal Kannan, S. Sitharama Iyengar, *“Game Theoretic Models for Reliable Path-Length and Energy Constrained Routing with Data Aggregation in Wireless Sensor Networks”*. IEEE Journal on Selected Areas in Communication, Vol 22 No , August 2004

[3] A. Urpi, M. Bonuccelli, S. Giordano, *"Modelling cooperation in mobile ad hoc networks: a formal description of selfishness"*. Proceedings of WiOpt Workshop, 2003

[4] Game Theory for Applied Economists, *"Game Theory For Applied Economists"*. Princeton University Press , Princeton New Jersey. ISBN0-691-00395-5

[5] Kewei Sha, Weisong Shi, *"Modeling the Lifetime of Wireless Sensor Networks"*. Sensor Letters Vol.3, 1–10, 2005

[6] Mathwork's Home Page: http://www.mathworks.com/

[7] Wendi Rabiner Heinzelman, Anantha Chandrakasan, Hari Balakrishnan, *"Energy Efficient Communication Protocol for Wireless Microsensor Networks"*. Proceedings of the Hawaii International Conference on System Sciences, Maui, Hawaii, January 2000

[8] The NS-2 Network Simulator: http://www.isi.edu/nsnam/ns/