Fair Energy Consumption in Wireless Sensor Networks

Niels Kasch*
Email: nkasch1@umbc.edu

Dave Feltenberger*
Email: dfelten1@umbc.edu

Fatih Senel*
Email: fsenel1@umbc.edu

*Department of Computer Science and
Electrical Engineering
University of Maryland, Baltimore County
Baltimore, MD 21250
May 8th, 2009

Abstract—We introduce the notion of *fairness* in power consumption of wireless sensor networks (WSNs). Herein, *fairness* in power consumption refers to an approximately equal power consumption rate among all nodes in a WSN. This topic has been largely overlooked by the literature, but has a significant impact in the design of WSNs with fixed sensor nodes and resource scarce flexible relay nodes.

We propose an approximation algorithm with the aim to introduce fairness to power consumption in WSNs. While overall power consumption has been studied previously, our algorithm considers fairness of power consumption rates a priority in WSN design. Our algorithm introduces additional, limited resources (i.e. relay nodes) to a fixed network. These resources may be moved to achieve minimal overall power consumption and maximum fairness. Hence, we give a multi-objective optimization algorithm for (1) maximum fairness in power consumption, (2) minimal additional resources and (3) minimal power consumption of WSNs.

Experimental analysis of randomly generated WSNs show that the introduction of a small numbers of relay nodes (as compared to sensor nodes) results in a significant increase in fairness of power consumption. However, as further relay nodes are added, we encounter a diminishing return in the increase of fairness. Hence, we pose the question of optimality considering the trade off between fairness and relay nodes and give a discussion for the experimental determination of such optimality.

We briefly illustrate our WSN fairness simulation software that utilizes the algorithm and visualizes the process of optimizing a WSN for fairness.

Index Terms—Wireless Sensor Network, Energy Conservation, Fairness, Energy Consumption

I. INTRODUCTION

Over the past decade, Wireless Sensor Networks (WSN) have been employed in a variety of domains ranging from macroscopic applications such a weather monitoring and traffic control to microscopic applications in medical screening and biomedication. WSNs consist of a collection of independent devices (nodes) that are connected wirelessly in an ad hoc fashion. Individual nodes are equipped with sensors collecting information from the environment. The types of sensors employed by a node depend on the purpose of the WSN and may include active and passive sensors. The size of WSNs (i.e. the number of nodes in the network) also depends on the intended purpose of the network as well as other factors such as the networks resolution in terms of sampling frequency across space and transmission range, to name a few.

As wireless transmission ranges are limited, multi-hop networking plays an integral role in ensuring the connectivity of the network as a whole. Nodes transmit collected information to a centralized collection point, which in turn may distribute instructions or management data to nodes. Therefore, it is essential that segments of the network do not become disconnected due to wireless transmission range limitations. Section II reviews past and current research of connectivity in WSNs.

A second important criterion concerns the energy efficiency of WSNs. Consider, for example, a remotely located earth quake sensing station or a patient with implanted heart rate and blood gas sensors. Due to geographical (non-availability of power line infrastructure), economic (prohibitive cost of expanding infrastructure)

and/or medical (infeasibility of permanent power connections) reasons, among others, it often is impractical or impossible to power the nodes of a WSN using existing energy grids. For those reason, nodes are often battery powered, and as such, their lifetime is subject to the life of their batteries. Energy efficiency of nodes is therefore a primary concern for extending the lifetime of individual nodes and the network as a whole.

Furthermore, in this paper we introduce the notion of fairness with regard to power consumption. The aim of fair power consumption is to equalize the power consumption rates of nodes across an entire WSN. Fair power consumption has an immediate impact on economical and practical considerations for WSNs. Fair power consumption enables precise predictability of battery lifetimes of all the nodes in a network. Such predictability can be used to optimize node replacement schedules such that (1) all nodes fail (due to battery exhaustion) simultaneously, (2) groups of nodes fail simultaneously, or (3) nodes fail in a predetermined order. For example, it is desirable to replace implanted biomedical sensor nodes of a patient all at once at the largest possible intervals in order to minimize frequent multiple invasive procedures. Therefore, it is desirable to maximize the fair battery lifetime of all nodes in the WSN.

Wireless radio transmissions drop off at an exponential rate due to ground reflection of radio signals. Since the signal attenuation rate is exponential with distance, it requires a significant amount of energy to transmit a signal only a small distance further. As such, it is often desirable to introduce relay nodes. Relay nodes differ in that their primary task is not sensing their environment but to ensure connectivity of the network. For the purposes of this paper, relay nodes will also be utilized to approximately equalize power consumption rates. Relay nodes are introduced in areas (as long as resource limitations permit) where power consumption is highest.

In this work, we propose an algorithm with the aim to introduce fairness to power consumption in WSNs. While overall power consumption has been studied previously, our algorithm considers fairness of power consumption rates a priority in WSN design. As seen in Section III, our algorithm may introduce additional, limited resources (i.e. relay nodes) to a fixed network. These resources may be moved to achieve minimal overall power consumption and maximum fairness. Hence, we introduce an approximation algorithm for three dimensions in WSNs: (1) (1) maximum fairness in power

consumption, (2) minimal additional resources and (3) minimal power consumption.

In Section II we present previous work relevant to our research followed by a precise problem definition, our approach and resulting algorithm in Section III. Section IV gives a formal analysis of our algorithm. We conclude our discussion in Section VII.

II. RELATED WORK

Construction and design of wireless sensor networks often heavily relies on common graph algorithms such as Prim and Kruskal's *Minimum Spanning Tree (MST)* [1] to ensure connectivity while minimizing the total length of all edges (i.e. wireless connections). The *Steiner Tree* problem extends this notion by introducing additional vertices to a graph in order to reduce the length of the spanning tree. The Steiner Tree problem is of particular interest to WSNs, in that additional vertices called relay nodes are often introduced to ensure connectivity. Estrin et al. [2] focus on connectivity establishment and maintainability using relay nodes in WSNs. They note that WSNs are subject to wireless transmission range constraints which in turn affect the interconnections between nodes. As the energy required to transmit data on wireless links is directly proportional to the distance between source and destination nodes, a significant amount of additional energy is required to establish and maintain a connection with more distant nodes. Hence, the most prominent source of energy consumption in wireless sensor networks is message transmission. Estrin et al. utilize relay node placement to combat increased energy requirements by placing nodes along wireless links of maximum distance in the network.

Lloyd [3] discusses two strategies for ensuring a WSN is fully connected. The first is a single-tiered node placement strategy in which every node is connected via some path consisting of either sensor nodes or relay nodes. The length between a sensor node and any other node must be $\leq r$, while the length between two relay nodes can be of distance R, where R is defined as $R \geq r$. Node placement is achieved using Steiner Tree nodes. The second strategy discussed is what they call a two-tiered node placement strategy, in which any two nodes of the WSN are fully connected by all relay nodes. That is, between any two sensor nodes in the WSN there exists a path by which every intermediary node is a relay node.

Gao et al. [4] introduce the notion of reducing the radio range of nodes and employing a collaborating scheme between nodes to forward data to a base station or sink in order to conserve energy.

Song et al. [5] address the *energy hole* problem - the problem of depleting the energy reserves of high load nodes faster than non-high load nodes - by proposing an NP hard multi-objective optimization problem (MOP). They propose a centralized algorithm as well as a distributed algorithm for assigning the transmission ranges of sensors in order to maximize the network lifetime. Notice that increasing the network lifetime also increases the lifetime the shortest lived nodes. We differ in our work in that we aim to equalized the power consumption rates of all nodes while simultaneously increasing the network lifetime.

Cheng [6] proposes a relay node placement algorithm using the minimum number of relay nodes, so that the distance between each hop is less than or equal to the common transmission range. This problem is similar to the Steiner Minimum Tree with minimum number of Steiner Points and bounded edge-length (SMT-MSPBEL). They propose a 3-approximation algorithm as well as a 2.5 approximation algorithm. The 2.5 approximation algorithm follows a randomized strategy whose performance is faster than 3-approximation algorithm.

Gandham et al. [7] investigate energy efficiency in wireless sensor networks using multiple base stations. In order to prolong the lifetime of the sensor network, multiple base stations are employed that cover the entire network area. The lifetime of the sensor network is separated into equal periods of time called *rounds* and base stations are relocated at the start of a round. They propose an algorithm for base station placement at the beginning of each round which maximizes network lifetime.

Tang et al. [8] study relay node placement problem in large scale wireless sensor networks such that sensor nodes are connected to at least one relay node and all relay nodes are connected amongst each other. To provide relay node fault tolerance they define the *2-Connected Relay Node Double Cover* (2CRNDC) problem and present a polynomial time approximation algorithms to solve the 2CRNSC.

Nguyen et al. [9] address the problem of unbalanced energy consumption among sensor nodes. This paper touches on concepts directly related to our work. They notice that unbalanced energy consumption among sensor nodes in home network domains is the result of employing only a single base station to collect sensor data. As sensor nodes further from the base station have to utilize more energy to transmit data across further distances, they propose a sensor network architecture with 2 base stations (and their corresponding communication protocol). While this solution approximately equalizes power consumption in relatively small networks, we are interested in a more general network topology.

III. PROPOSED APPROACH

A. System Model

The model and algorithms described in this paper make the following assumptions:

- Each sensor/node sends an equal amount of data per time unit.
- Each node can receive an arbitrary amount of data per time unit.
- Each node can store an arbitrary amount of data without energy penalties. This assumption enables this model to ignore data transmission bottlenecks.
- The transmission range of each sensor node does not exceed the maximum transmission range T.
- Only nodes within transmission range of each other are connected (i.e. have an edge between each other in the network).
- A node's energy dissipation per bit transmitted (according to the first order radio model [10]).
- Transmitting a bit over distance d requires energy d^2.
- The power consumption rate PCR for a node n is defined as $PCR(n) = \frac{\max_{v \in N} d(v)^2}{\text{unit time}}$, where N is the set of neighbors of a node.
- Sensor nodes have a fixed location. Their location may not change during the lifetime of the network.
- Relay nodes are movable. Their location is flexible during the lifetime of the network.
- Relay nodes can be added to the original network.

B. Problem Formulation

We formulate the Minimal Fair Energy Consumption with Minimal Additional Resources (MFEC-MAR) problem as follows: Given a set of fixed sensor nodes S, find a minimal set of relay sensor nodes R such that the power consumption rate PCR is equal for each node $v \in \{S \cup R\}$, that is:

$$\forall_{i=0}^{n=|S \cup R|} PCR(v_i) = PCR(v_2) = ... = PCR(v_n).$$

The induced graph $G = (S \cup R, E)$ must be connected, where an edge $(u, v) \in E$ if u is within transmission range T_v of v and v is within transmission range T_u of u. A node u is within transmission range of v if $PCR(u) \leq distance(u, v)^2$. Formally, MFEC-MAR is defined as follows:

- **Input:** A set of fixed sensor nodes S in Euclidean space.

- **Output:** A connected network $G = (S \cup R, E)$ such that:
    - R is the set of introduced relay nodes such that:
        * $|R|$ is minimal
        * $\forall_{i=0}^{n=|S \cup R|} PCR(v_i) = PCR(v_2) = ... = PCR(v_n)$
    - G is connected
    - $\sum_{v \in |S \cup R|} PCR(v)$ is minimal

The Minimal Fair Energy Consumption with Minimal Additional Resources Approximation (MFEC-MAR-Approx) problem aims to find an approximate solution to the MFEC-MAR problem. In the latter part of this paper, we will give an algorithm for MFEC-MAR-Approx. The MFEC-MAR-Approx problem is defined as follows:

Given a set of fixed sensor nodes S, an allowable standard deviation α and a maximum number of relay nodes k, find a set of relay sensor nodes R, where $|R| \leq k$, such that the power consumption rates PCR for all nodes is within α. The induced graph $G = (S \cup R, E)$ must be connected, where an edge $(u, v) \in E$ if u is within transmission range T_v of v and v is within transmission range T_u of u. A node u is within transmission range of v if $PCR(u) \leq distance(u, v)^2$. Formally, MFEC-MAR-Approx is defined as follows:

- **Input:** A set of fixed sensor nodes S in Euclidean space, a standard deviation α of power consumption rates and a maximum number of relay nodes k.
- **Output:** A connected network $G = (S \cup R, E)$ such that:
    - R is the set of introduced relay nodes such that:
        * $|R| \leq k$
        * $\sigma(PCR) \leq \alpha$
    - G is connected

For the remainder of this paper, whenever we refer to fairness of power consumption rates (PCR), we denote this fairness measure α as the maximum allowable standard deviation of power consumption rates.

IV. ALGORITHM AND ANALYSIS

A. Algorithms for MFEC-MAR (Fairness in WSNs)

In this section we will explain our two step approach to solving the fairness problem in WSNs.

1) Connecting Nodes: Given a set T of n terminals which are deployed in an Euclidean plane, and a positive constant R, our aim is to find a Steiner Minimum Tree τ with minimum number of Steiner points (SMT-MSP). In 1999, Lin *et al.* [11] showed that SMT-MSP is NP-Hard. In 2008, Cheng *et al.* [6], presented a $O(n^3)$-time

approximations with performance ratio is at most 3. In this project we implemented Ratio-3 approximation presented in [6], to connect initially disconnected terminals. The idea behind the algorithm is simple: First we assume a fully connected undirected graph $G = (V, E)$ where $V =$ set of terminals, and $(u, v) \in E \ \forall u, v \in V$ and $u \neq v$. Then we sort all $\frac{n \cdot (n-1)}{2}$ edges. For each subset of 3 terminals a, b and c respectively in three connected components, if there exists such a point s within distance R, we put a 3-star s which includes edges (s, a), (s, b), and (s, c). Then for each edge where $|e_i| > R$ and e_i connects two connected component, we move the steinerized e_i into τ. Thus we establish the connectivity of terminals.

---

1 FairSMT(*G, α, k, R*)

Input: $G = (T, E)$ such that T is the set of terminals and E is the initial set of edges, α is the target Standard Deviation, k is the maximum number of relay nodes, and R is the positive constant which indicates the radio range of a sensor

Output: $G' = (V, E')$ with $V = T \cup S$ where S is the set of relay nodes and E' is the new set of edges

2 begin

3 $G' \leftarrow Ratio - 3 - SMT(G)$

4 $G' \leftarrow MakeFair(G', \alpha, k)$

5   return G'

6 end

Algorithm 1: FairSMT

---

2) Optimizing Node Location: SMT-MSP provides connectivity using minimum number of Steiner points. In some cases, the relay nodes in the resulting topology, may be too close or too far from each other. However this is not desirable, since it violates fairness of power consumption rates.

In this section we explain an extension to the SMT problem to include a fairness measure across the network of nodes. The explanation will include the moving of Steiner Points to optimal locations as well as the introduction of additional resources (nodes) to achieve optimality. The discussion includes an overview of the trade-off between additional resources and fairness and illustrates results of the optimization operations across multiple dimensions (such as minimizing the deployment of additional resources, maximizing fairness while minimizing power consumption).

1 MakeFair(G', α, k)
Input: $G = (V, E)$ with $V = T \cup S$ where T is the set of terminals and S is set of nonterminals returned by Ratio-3 approximation, α is the target Standard Deviation, k is the maximum number of relay nodes
Output: $(G' = (V', E'), achieved)$ where $V' = V \cup S'$ where S' is the set of newly added relay nodes and E' is the new set of edges, and achieved is boolean variable that indicates if target Standart Deviation is achieved or not

2 begin
3   // build a max-heap with the PCR of each steiner node
4 $heap \leftarrow Max - Heap(S)$
5   **while** $|S| < k$ **do**
6 $\beta \leftarrow$ STDEVofPCRs()
7 $dec \leftarrow true$
8   **while** $dec = true$ **do**
9 $MoveRelayNode - geometric(G)$
10   **if** *STDEVofPCRs() $< \beta$* **then**
11 $dec \leftarrow true$
12 $\beta \leftarrow$ STDEVofPCRs()
13   **else**
14 $dec \leftarrow false$
15   **end**
16   **end**
17 $AddRelayNode(G)$
18 $MoveRelayNodes(G)$
19 **end**
20 $G' \leftarrow G$
21 $\beta \leftarrow$ STDEVofPCRs()
22 **if** $\beta \leq \alpha$ **then**
23   return $(G', true)$
24 **else**
25   return $(G', false)$
26 **end**
27 end
Algorithm 2: Pseudo-code of fairness approximation

1 MoveRelayNodes-geometric(G)
Input: $G = (V, E)$ with $V = T \cup S$ where T is the set of terminals and S is set of nonterminals
Output: $G = (V, E)$ same graph with different edge lengths

2 begin
3   **for** $s_i \in S$ **do**
4   **if** *s_i has 3 neighbors* **then**
5   **if** *neighbors of s_i is on a circle* **then**
6   move s_i to the center of the circle
7   **end**
8   **end**
9   **end**
10 end
Algorithm 3: Detailed description of MoveNodes-geometric function

1 AddRelayNodes(G)
Input: $G = (V, E)$ with $V = T \cup S$ where T is the set of terminals and S is set of nonterminals
Output: $G' = (V', E')$ where $V' = V \cup u$ and u is the newly added relay node and E' is the new set of edges, and achieved is boolean variable that indicates if target Standart Deviation is achieved or not

2 begin
3 $s \leftarrow$ heap.extractMax()
4 $t \leftarrow$ s.getFarthestNeighbor()
5   remove (s, t) from the steiner tree.
6   deploy new steiner node u
7 $p \leftarrow$ middle point (s, t)
8 u.setCoordinates(p)
9   add (s, u) and (u, t) to the tree
10 heap.insert(u)
11 heap.insert(s)
12 heapify()
13 end
Algorithm 4: Pseudo-code of AddSteinerNodes

Initially, our fairness approximation algorithm calculates the power consumption rates of each relay node (PCR) and inserts the nodes to a max heap. The main loop of the MakeFair method iteratively moves and adds relay nodes until the max number relay nodes allowed, is reached. The inner while loop of Algorithm 2 (lines 8-16), invokes MoveRelayNodes-geometric function, which is described in Algorithm 3. MoveRelayNodes-geometric takes each 3-stars as defined in [6], and moves the relay nodes to the center of circle that circumscribes the 3-neighbors. Here, our aim is to provide local fairness. After moving each relay node to the center of 3-star, we extract the node which has the highest PCR, and find its farthest neighbor (see Algorithm 4). Since the longest edge determines the PCR of the node, we add an additional relay node to

```
1  MoveRelayNodes(G)
   Input: G = (V, E) with V = T ∪ S where T is the
          set of terminals and S is set of nonterminals
   Output: G = (V, E) same graph with different
          edge lengths
2  begin
3     while heap is not empty do
4        s ← heap.extractMax()
5        t ← s.getFarthestNeighbor()
6        // calculateLocation finds the best location p
         which is in between s and t, where p
         minimizes the PCR of s.
7        p ← calculateLocation(s, t)
8        move(s, p)
9        heap.insert(s)
10       heapify()
11    end
12 end
```

Algorithm 5: Pseudo-code of MoveRelayNodes

the middle point the longest edge. Then we add newly deployed node along with the extracted node to the heap. Deploying new node will definitely change PCRs of some set of nodes. We move each relay node to the local optimal positions. In Algorithm 5, it is seen that, the optimal location for a node u is in between itself and its farthest neighbor. We calculate this location using some trigonometric functions. Then we insert u to the heap and continue iteration until $|S| = k$ where S is the set of relay nodes and k is the maximum number of relay nodes which can be deployed.

B. Theoretical Analysis

In Fairness Approximation we first create a heap and insert the relay nodes to the heap. Assuming that FIB-HEAP structure is used to implement the MAX-HEAP. Since amortized cost single insert operation is $O(1)$, the cost of inserting k elements into the heap is $O(k)$ where k is maximum number of relay nodes which can be deployed. In the worst case, the outer while loop of Algorithm iterates k times and inner while loop iterates β times where β is the initial standard deviation of PCRs. The time complexity of MoveRelayNodes-geometric is $O(k)$. So the total cost of inner while loop (lines 8-16 of Algorithm 2) is $O(k\beta)$

In Algorithm 4, we extract the node with highest PCR from the heap, and add a relay node to middle of the longest edge. The cost of extracting from the heap

is $O(\lg k)$. In Algorithm 5, we move each and every relay node to their local optimal locations. The time complexity of MoveRelayNodes function is $O(k \lg k)$.

The overall time complexity is $O(n^3)+O(k)(O(k\beta)+O(\lg k) + O(k \lg k)) = O(n^3) + O(k^2\beta) + O(k^2 \lg k)$

C. Experimental Analysis

The theoretical analysis will be verified by an experimental analysis.

V. RESULTS

A. Experimental Results

To summarize succinctly: maximizing fairness with restrictions on the number of relay nodes is a very difficult problem. Maximizing fairness is particularly difficult with an arbitrary topology in a large surface area. Large surface areas are problematic because a topology may have very diverse distances between nodes, and with restrictions on the number of relay nodes there are limits on fairness. This is natural, however, as one cannot expect to add an arbitrary number of relay sensors to a sensor network on the human body, for example: there are practical limits to the number of relay sensors, and by proxy, the amount of fairness that can be reasonably expected. This section will discuss the experimental findings of this paper.

B. Methodology

Tests were performed using the Java programming language and a freely available implementation of the Steiner Minimum Tree (SMT) algorithm found on the Internet [12]. This implementation was extended to include support for fairness. Figure 1 shows a sample of what a graph looks like. Notice the key at the top, showing the standard deviation of power consumption (i.e. fairness), as well as various other important measurements. White points denote "terminal", i.e. non-movable, nodes in the sensor network. Red points are Steiner-or, relay-nodes. Relay nodes are added to the graph during the creation of the SMT as well as during the procedure to approximately equalize fairness. Relay nodes can be moved in any direction.

A number of configurations were used to collect data. Varied data elements include every combination of the folllowing:

- Canvas size, from 100x100 pixels to 600x600 pixels, increased by increments of 100
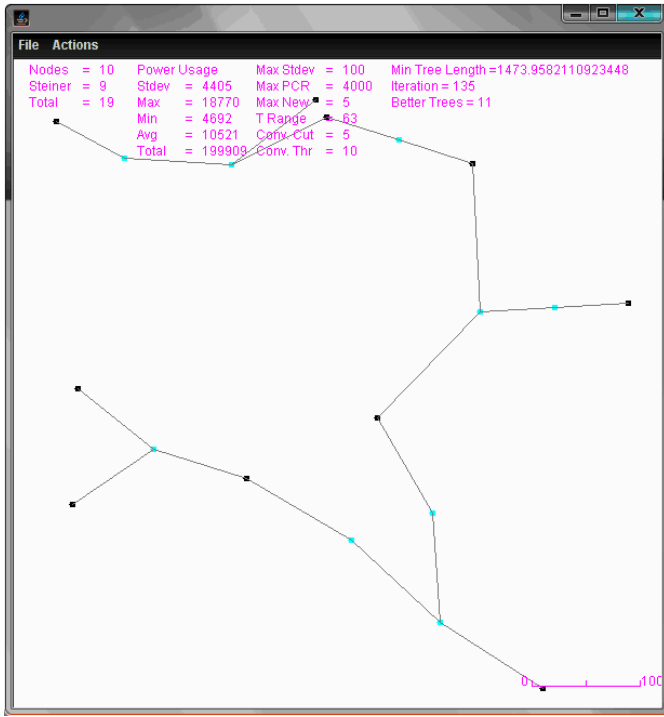- Maximum k value, from 1 to 20 increased by increments of 1

Fig. 1. Steiner Minimum Tree Simulation.

- Number of terminal nodes, from 10 to 50, increased by increments of 10

Tests were run 10 times each and the results were averaged for more accurate results. Following are graphs depicting the results, and the group's interpretation of the results. Although it was mentioned earlier, it is worth repeating that fairness is determined by the standard deviation of the difference between power consumption rates for each node. Therefore, the more fair a network, the lower the standard deviation.

C. Results

Within each graph, the results are separated according to the size of the canvas on which fairness was measured. The differences are dramatic as the canvas size increases because the distance between nodes can be much greater.

D. Fairness With Varying Number of Terminal Nodes

The following three graphs show the total power consumption for different numbers of terminal nodes. In all three cases, the graphs confirm intuition: as the number of relay nodes increases, the graphs become more fair. Fairness increases because there are more opportunities to minimize the difference in distances.

Fairness is significantly different for large canvas sizes compared to smaller ones, so in order to better illustrate the trend towards fairness as k increases on small

canvases, Figure 6 shows a single canvas size-100x100 pixels-graphed against an increasing k. As expected, it depicts increasing fairness as the number of relay nodes increases.

E. Power Consumption

Another interesting metric is what happens to the total power consumption as surface area increases. As explained in an earlier section, the PCR increases as distance between nodes increases. Power consumption, therefore, decreases based on two factors: as k increases, and as the surface area decreases. Figure 7 illustrates this.

Total power consumption, however, is less important in the context of this paper. More important is fairness. To illustrate fairness in power consumption, Figure 8 shows the power consumption per node on a 300x300 pixel canvas, with 20 terminal nodes, graphed as k increases. When k is low, there is little to be done to make the network fair. By contrast, as k increases, the maximum, average, and minimum power consumption per node start to converge.

F. Interpretation of Results

The authors of this paper could not find literature studying fairness. It is difficult to compare the findings of this paper against some previous standard. This section, therefore, will not compare its results with other papers. The results of experimenting, however, do meet expectations: as relay nodes are added, fairness increases. This is little help ultimately, as it naively implies that k should increase indefinitely. Clearly this cannot happen: if k approaches some very large number, the sensor network might as well be wired! The problem, then, is to find a good cutoff for k.

G. Optimal k Value

Optimal k values change based on the number of terminal nodes and the surface area of the sensor network. The graphs above and the rest of the data from experimentation show that there are very significant gains in fairness as k increases from 0 regardless of the number of terminal nodes and surface area. For networks in a small surface area, for example 100x100 to 300x300, fairness shows little gain after k reaches 8 to 12 relay nodes, depending on the number of terminal nodes. The graphs start to level off earlier when the number of terminal nodes is high, while fewer terminal nodes result in more relay nodes to reach approximate fairness. Larger surface areas exhibit the same behavior, but require larger k values before leveling off. Although there isn't a single

optimal k value that fits all applications, the important aspect of the fairness is that results tend to be consistent: as k increases, fairness increases, and as surface area decreases, fairness also increases. The optimal k value, therefore, should be found experimentally, given the practical constraints of the network and application. For instance, if the surface area of the sensor network is very small, the optimal k value will very likely be lower than a network with double the surface area.

VI. SOFTWARE SIMULATION

The algorithms as described in section IV-A1 and IV-A2 have been implemented and augmented with a graphical user interface (GUI). The GUI allows for the plotting of sensor (shown as white points) and relay nodes (shown as red points) and their corresponding edges in 2D-space. Figure 2 illustrates the GUI showing a graph of 11 sensor nodes and 10 relay nodes. The graph has been optimized to maximize energy consumption fairness. The GUI supports the loading and saving of nodes. Once a dataset of sensor nodes is loaded, the interface connects the nodes using the Steiner Minimum Tree approximation described in section IV-A1. The *Action* menu item enables control over the fairness optimization algorithms described in section IV-A2. The top part of the interface displays statistical information such as total power consumption and fairness in power consumption of the sensor network.

VII. CONCLUSION

Approximating fairness in sensor networks has had little to no attention in the literature. This is perhaps because in many cases, only total power consumption matters. When dealing with sensor networks where changing, repairing, or working on sensors is invasive-for instance a sensor network on a person, or a network in a difficult to reach destination-fairness becomes important. As discussed throughout the paper, the reason fairness is important is to ensure node failures occur approximately at the same time. With approximate fairness, this becomes possible with regard to battery failure, as nodes will deplete power reserves at roughly the same rate. With node failures occurring at the same time, it allows a network to operate under better-understood and more predictable conditions, as well as helping to avoid two undesirable properties. First, invasive node replacement procedures do not occur as frequently as if single nodes were allowed to consume power at significantly different rates: with unfair conditions, certain nodes may fail
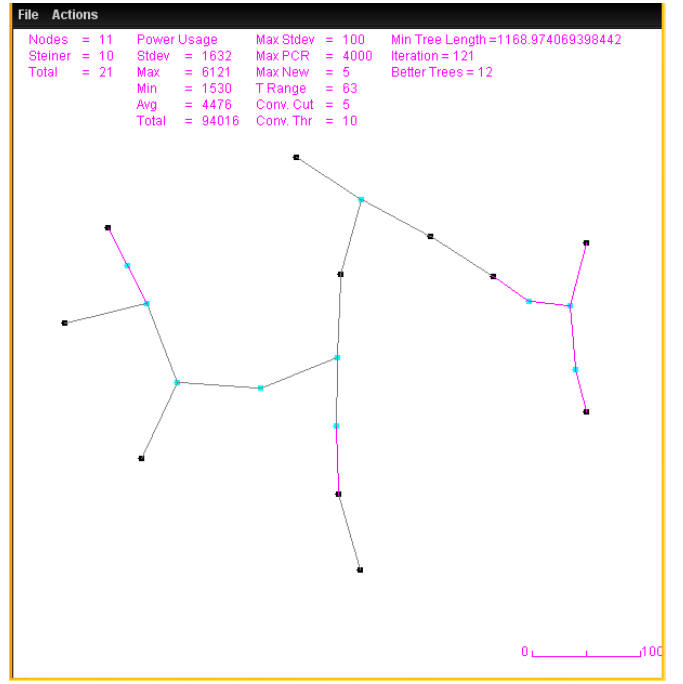

Fig. 2. The graphical user interface (GUI) for creating *fair* sensor networks. 11 sensor (white) and 10 Relay (red) nodes are plotted in 2D-space. The image shows the resulting sensor network after optimizing energy consumption fairness.

significantly sooner than others, thereby requiring replacement procedures more frequently. Second, because the procedure is invasive, it is desirable to replace all nodes during the same procedure; under this model, fairness means less waste, since there will be fewer nodes with significant battery power left, while unfair networks potentially waste nodes with significant battery power if all nodes are replaced at the same time.

The fairness approximation procedure discussed above is difficult to measure in any absolute sense. The reason is twofold: first, there are no baseline comparisons to be made, as the authors could not find fairness studies in the literature; second, and more importantly, the context of the application changes characteristics significantly. Larger surface areas require more relay nodes to achieve fairness. Similarly, when the number of terminal nodes is low, more relay nodes are needed to achieve fairness. Conversely, smaller surface area or more terminal nodes tend to require fewer relay nodes. These observations are both intuitive and experimentally shown. This is a significant advantage of the fairness procedure described in this paper: consistent results that follow intuition. To evaluate the number of relay nodes necessary, it is advisable, therefore, to run simulations and find a k value that best meets the requirements of the application.

VIII. Future Work

There are a number of areas touched on in this paper that could be studied more extensively in future work. The most important consideration that was ignored, which in practical applications will almost definitely be an important measure, is reducing overall power consumption. The two-fold optimization of optimizing fairness and power consumption would prove very useful in the use cases outlined in this paper, too.

A second future area to consider is connecting all nodes within range and including them in fairness and power measurements. Currently the algorithm works on a tree without cycles for simplicity, but in any real application of the sensor networks, nodes would need to consider all other nodes within range instead of just one.

Acknowledgment

References

[1] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*. McGraw-Hill Higher Education, 2001.

[2] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 4, pp. 2033–2036 vol.4, 2001.

[3] E. L. Lloyd, "Relay node placement in wireless sensor networks," *IEEE Trans. Comput.*, vol. 56, no. 1, pp. 134–138, 2007, senior Member-Xue,, Guoliang.

[4] Q. Gao, K. Blow, D. Holding, I. Marshall, and X. Peng, "Radio range adjustment for energy efficient wireless sensor networks," *Ad Hoc Networks*, vol. 4, no. 1, pp. 75 – 82, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/B7576-4CKBS1W-1/2/2187b27cbc399de3e1997217d9c0615d

[5] C. Song, M. Liu, J. Cao, Y. Zheng, H. Gong, and G. Chen, "Maximizing network lifetime based on transmission range adjustment in wireless sensor networks," *Computer Communications*, vol. In Press, Corrected Proof, pp. –, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6TYP-4VKP3YY-2/2/1e6382fb40bc30f6f8325508af770ac7

[6] X. Cheng, D.-Z. Du, L. Wang, and B. Xu, "Relay sensor placement in wireless sensor networks," *Wirel. Netw.*, vol. 14, no. 3, pp. 347–355, 2008.

[7] S. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," *Globecom*, 2003.

[8] J. Tang, B. Hao, and A. Sen, "Relay node placement in large scale wireless sensor networks," *Computer Communications*, vol. 29, no. 4, pp. 490 – 501, 2006, current areas of interest in wireless sensor networks designs. [Online]. Available: http://www.sciencedirect.com/science/article/B6TYP-4FG484B-3/2/99a1085f745e7339e6537931c7c32ba3

[9] B. N. Nguyen and D. Choi, "Fair energy consumption protocol with two base stations for wireless sensor networks," Sept. 2007, pp. 398–403.

[10] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8*. Washington, DC, USA: IEEE Computer Society, 2000, p. 8020.

[11] G.-H. Lin and G. Xue, "Steiner tree problem with minimum number of steiner points and bounded edge-length," *Inf. Process. Lett.*, vol. 69, no. 2, pp. 53–57, 1999.

[12] T. Ohyama, "Steiner tree problem," http://www.nirarebakun.com.
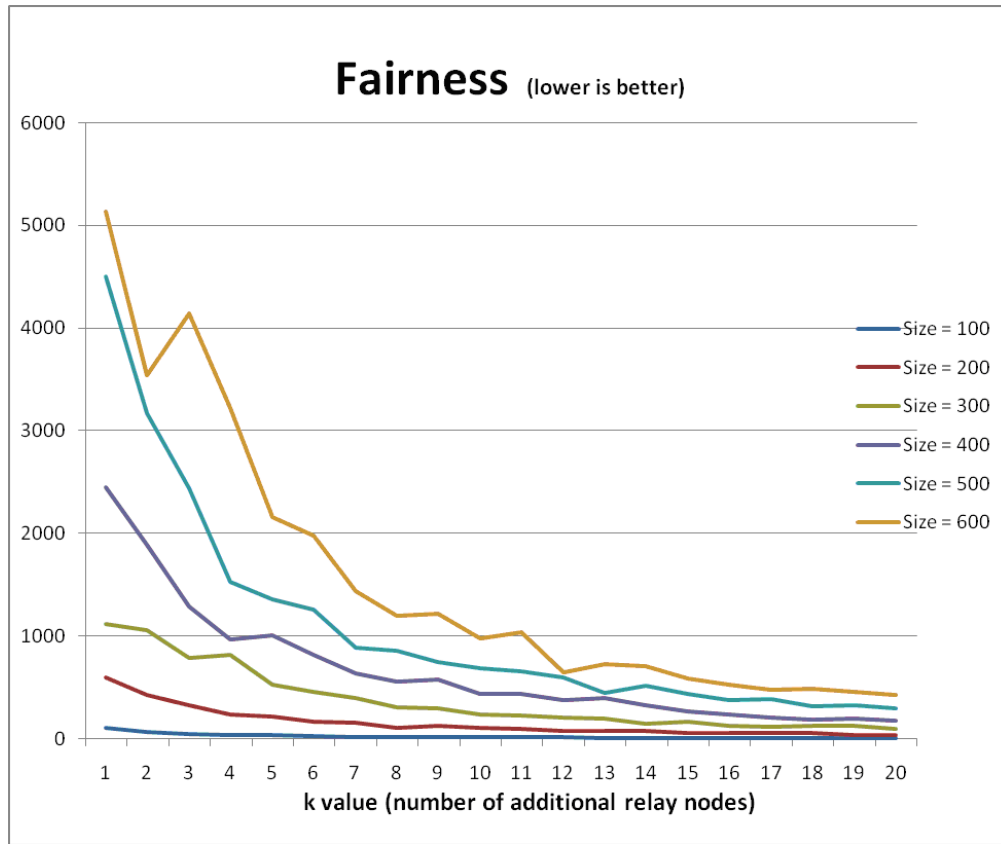
Fig. 3.   Fairness as k increases; 10 terminal nodes
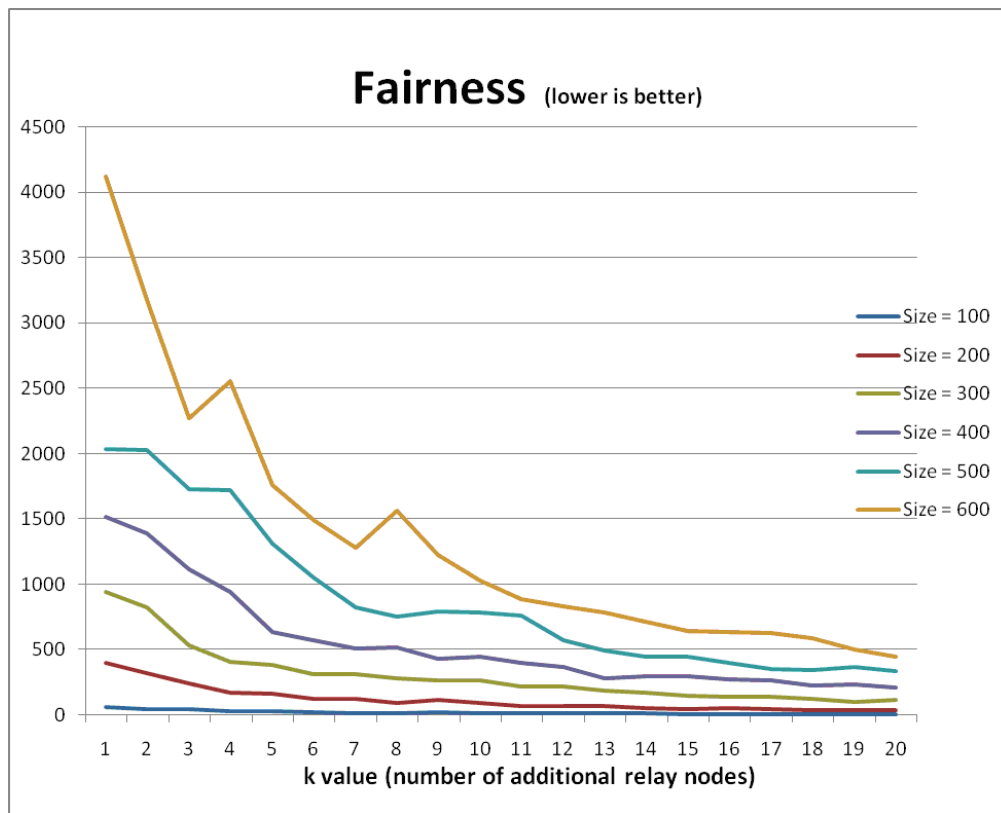
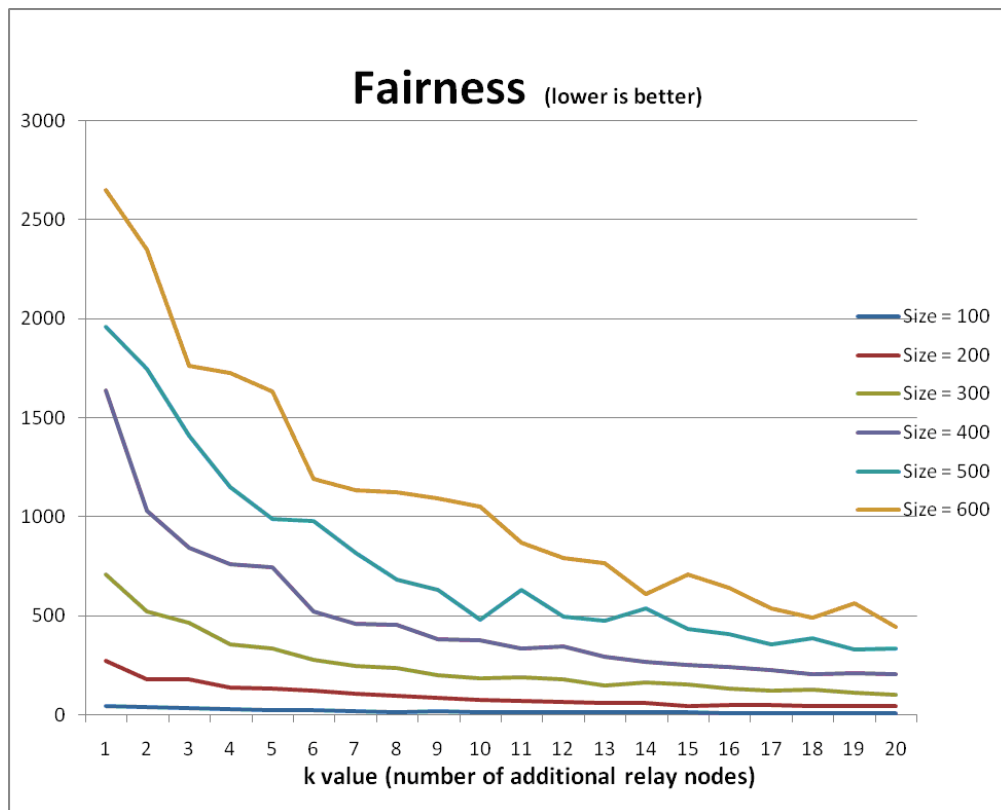Fig. 4. Fairness as k increases; 20 terminal nodes

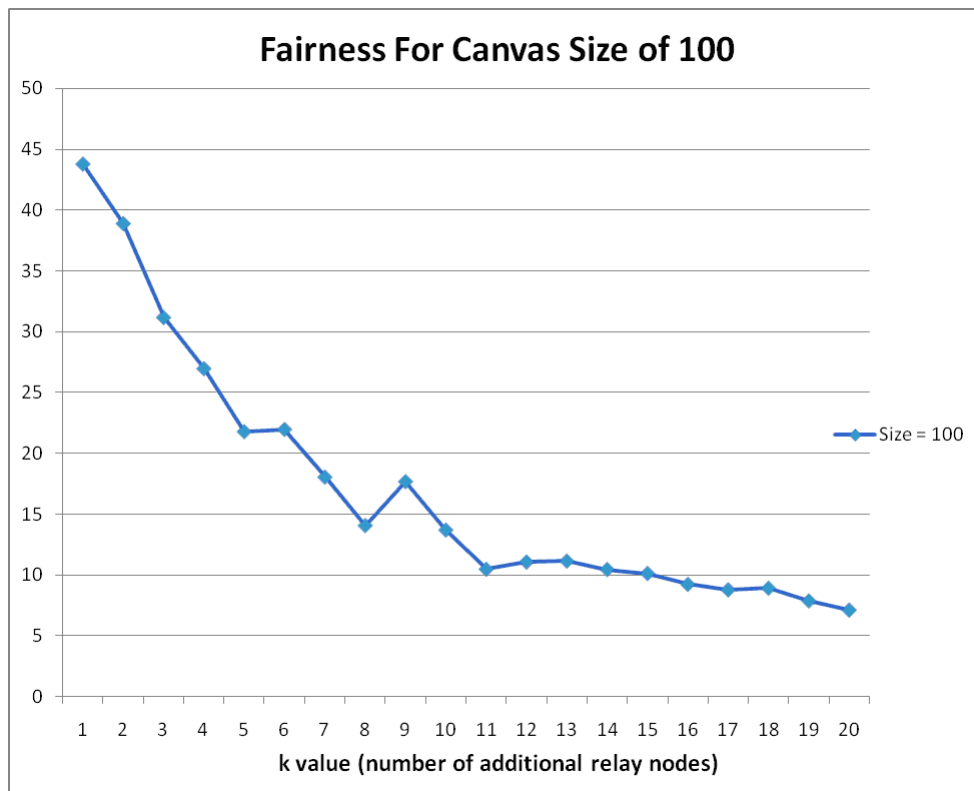Fig. 5. Fairness as k increases; 30 terminal nodes

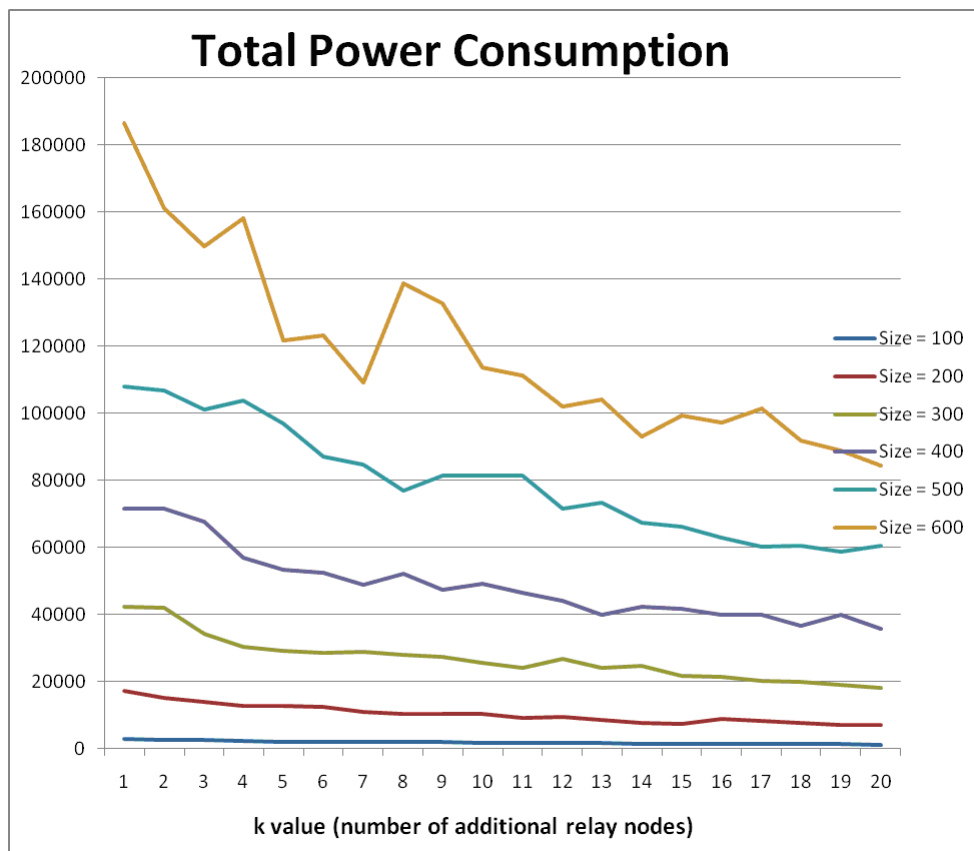Fig. 6. Fairness as k increases; 30 terminal nodes, and a canvas size of 100x100 pixels.

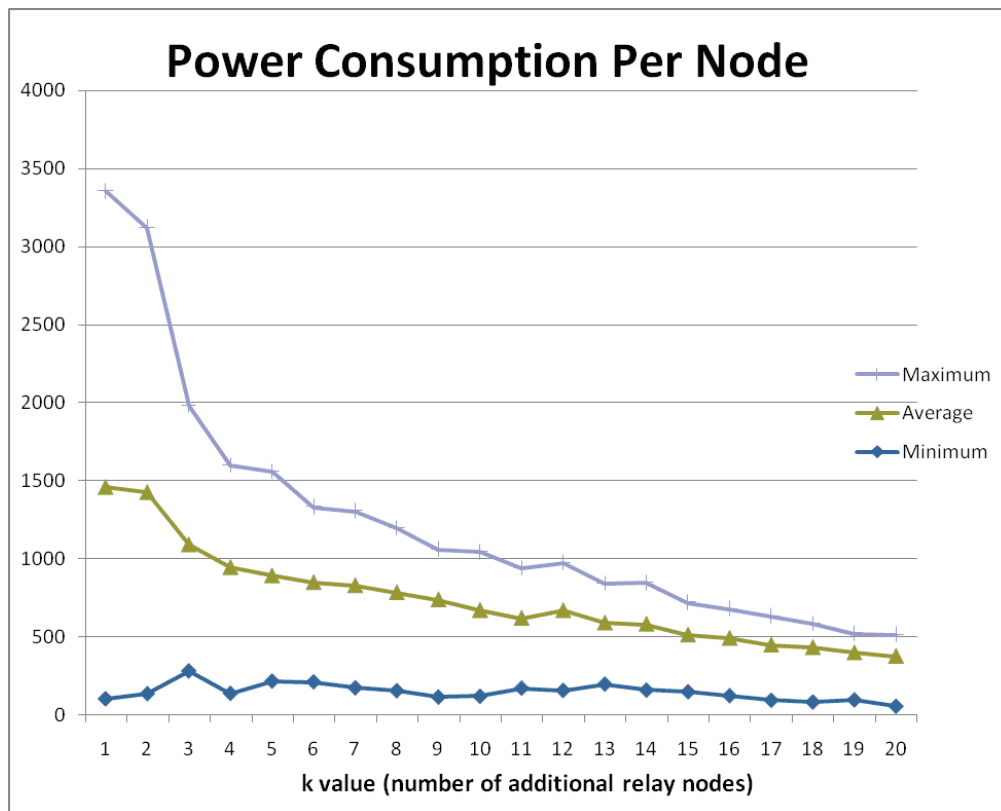Fig. 7. Total power consumption as k increases; 20 terminal nodes.

Fig. 8. Power consumption per node on a 300x300 pixel canvas; 20 terminal nodes.