Battery Allocation for Wireless Sensor Network Lifetime Maximization Under Cost Constraints

Hengyu Long† Yongpan Liu† Yiqun Wang† Robert P. Dick‡ Huazhong Yang†

†TNList, EE Dept.
Tsinghua University
Beijing, 100084, China
{ypliu,yanghz}@tsinghua.edu.cn

‡ EECS Dept.
University of Michigan
Ann Arbor, MI 48109, U.S.A.
dickrp@eecs.umich.edu

ABSTRACT

Wireless sensor networks hold the potential to open new domains to distributed data acquisition. However, such networks are prone to premature failure because some nodes deplete their batteries more rapidly than others due to workload variations, non-uniform communication, and heterogeneous hardware. Many-to-one traffic patterns are common in sensor networks, further increasing node power consumption heterogeneity. Most previous sensor network lifetime enhancement techniques focused on balancing power distribution, based on the assumption of uniform battery capacity allocation among homogeneous nodes.

This paper gives a formulation and solution to the cost-constrained lifetime-aware battery allocation problem for sensor networks with arbitrary topologies and heterogeneous power distributions. An integer nonlinear programming formulation is given. Based on an energy–cost battery pack model and optimal node partitioning algorithm, a rapid battery pack selection heuristic is developed and its deviation from optimality is quantified. Experimental results indicate that the proposed technique achieves network lifetime improvements ranging from 3–11× compared to uniform battery allocation, with no more than 10 battery pack energy levels. The proposed technique achieves 2–5 orders of magnitude speedup compared to a general-purpose commercial nonlinear program solver, solution quality improves, and little approximation error is observed.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Wireless Communication, Resource Allocation

Keywords

Wireless Sensor Network, Lifetime, Battery Allocation

ICCAD'09, November 2–5, 2009, San Jose, California, USA.

1. INTRODUCTION

Wireless Sensor Networks (WSN) are distributed data acquisition systems consisting of numerous wireless sensor nodes. They have the potential to allow sensing in applications and environments where it was previously impossible or prohibitively expensive. For example, WSNs may be used in weather monitoring, security, tactical surveillance, disaster management, and intelligent traffic control applications [1]. Infrastructure-free operation is one of their primary advantages. However, this beneficial attribute exacts a penalty. Distributed infrastructure-free operation in remote locations makes replacing batteries expensive. Energy constraints are therefore extremely tight.

WSN lifetime depends on the distribution of power among nodes in addition to average power consumption. There is a large body of work on reducing node power consumption. The most frequently described techniques are power state control [2], [3], hardware-software co-design [4], [5], as well as clustering and compression [6]–[8].

Past work has also attempted to balance power consumption among nodes. Much of this work exploits spatial and temporal data correlation. Work attempting to spatially balance power consumption allocates sensing, computation, and communication tasks more evenly among sensor nodes, e.g., via power-balanced routing [9]–[11] or non-uniform node deployment [12]. Work on balancing power consumption via temporal changes multiplexes the assignment of tasks to nodes, e.g., by periodically adjusting cluster heads [13] or mobile or multiple sinks [14], [15].

Past work on lifetime extension has focused on distributing power consumption evenly among sensor nodes based on the assumption that homogenous nodes with equal battery capacities are used. However, real WSN deployments may use heterogenous sensor nodes. For example, Hou et al. [16] deployed heterogeneous sensor nodes to construct a two-tier infrastructure to prolong WSN lifetime. Furthermore, the locations of sensor nodes are often carefully controlled in real deployments in order to reduce the cost. It is feasible to equip nodes with battery packs with different capacities, based on their computation and communication requirements.

Distributed battery configuration has the potential for both cost and energy efficiency in WSNs with heterogeneous spatial power consumption distributions. Conventional power balancing can be inconsistent with energy efficiency because some tasks and communication events really are spatially heterogeneous: there is a cost to balancing power by moving tasks and communication events from locations that were otherwise optimal. Heterogeneous battery allocation has the potential to reduce cost by reducing battery capacity for lightly loaded nodes and to increase WSN lifespan by allocating more energy to heavily loaded nodes. Battery allocation might also be used in future ad-hoc networks composed of sensor nodes and other computing devices such as notebooks and mobile phones. In summary, heterogeneous battery allocation has the potential to improve network lifespan and cost.

Sichitiu and Dutta [17] were the first to report that equipping sensor nodes with different battery capacities can prolong WSN lifespan. However, their battery allocation formulation is specific to circular communication topology WSNs with homogenous nodes. Their formulation minimizes batter energy instead of cost. Since battery volume is discrete and battery cost is highly non-linear in energy capacity, a total energy based cost function has drawbacks for real-world applications.

This paper makes the following contributions.

1. Given a set of battery types with different energy capacities and costs, we solve the cost-constrained battery allocation problem for a WSN with arbitrary topology and spatial power consumption distribution. An integer nonlinear programming (INLP) formulation is given, which allows the problem to be solved by general optimization packages.

2. We provide and prove an upper bound on WSN lifetime improvement via energy allocation and indicate the properties of node partitions resulting in maximal lifetime. Furthermore, we propose an approximate method to obtain the node partitioning. We also provide an energy–cost model for battery packs based on real data, which permits us to calculate the corresponding energy under different battery pack configurations given a specific budget.

3. Based on the optimal node partitioning and energy-cost model for battery packs, we proposed a heuristic to solve the cost-constrained WSN energy allocation problem. Experimental results indicate that the proposed technique can prolong lifetime by 4–13× with no more than 10 battery pack levels, compared with the uniform battery allocation approach. Furthermore, the heuristic achieves 2–5 orders of magnitudes speedup and better results than a general-purpose commercial INLP solver. Solution quality remains within 1% of optimal.

After we discuss the motivation of this work in the following section, the cost-constrained WSN battery allocation problem is formulated as an INLP problem in Section 3. The problem is solved in Section 4 and the errors of the proposed method are analyzed in Section 5. Section 6 presents the experimental results. We conclude the work in Section 7.

2. MOTIVATION

This section first discusses the power distribution characteristics of WSNs and defines network lifetime. It then explains the motivation of the proposed battery allocation method.

2.1 Power Distribution and Lifetime in WSN

Even in homogenous-node WSNs, the spatial heterogeneity of power consumption due to gathering, processing, and communicating data leads to unequal power consumptions for different WSN nodes. The flow of data from many leaf nodes to a sink node results in heterogeneous data transmission rates and therefore power consumptions. In heterogenous WSN, the power imbalance is further increased by dramatic differences in workloads and communication for different node types. Substantial work has been done to balance power distribution in WSNs [9]–[11], [13]–[15]. However, energy balancing techniques often impose substantial costs by forcing data to take longer routes or moving data to redistribute computation. The many-to-one traffic patterns in WSNs also complicate energy balancing [12]. Complementary techniques, such as heterogenous battery allocation, have the potential to efficiently extend WSN lifetimes.

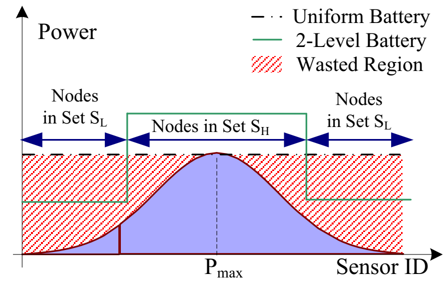

Figure 1: Uniform vs 2-level battery allocation.

Gandham et al. describe several WSN lifetime metrics [14]. In this paper, we will use the *time to first node battery depletion* lifetime metric. It should be noted that the proposed method is not limited to this metric. For example, by assigning various weights to the power consumptions of different sensor nodes, it can be easily extended to deal with a WSN lifetime metric based on the expiration of a set of critical nodes.

2.2 Battery Allocation in WSN

This section describes the motivation of lifetime-aware battery allocation for WSNs with arbitrary topologies, node configurations, and power distributions. Figure 1 shows a typical WSN power distribution in which compression and clustering techniques are used [18], assuming that the total energy is denoted as E_{tot} and it is assigned to all n sensor nodes. Given uniform battery allocation, each sensor node is equipped with a battery containing E_{tot}/n energy. The first sensor node failure time T_{life} is $T_{life} = E_{tot}/(n \cdot p_{max})$, where p_{max} is the maximum power consumption of any sensor node. The doted line indicates the power level at which the maximum-power (p_{max}) sensor node depletes its battery. The batteries of the sensor nodes located in the *wasted region* have unused energy when the network fails.

Using 2-level battery allocation, it is possible to divide the total energy into two parts: E_{Hv} and E_{Lv} ($E_{tot} = E_{Hv} + E_{Lv}$). n_H sensor nodes use high-volume batteries, each of which has a capacity of E_{Hv}/n_H. The other n_L sensor nodes have low-volume batteries wit E_{Lv}/n_L capacity. We denote the sets composed of sensor nodes with high- and low-volume batteries as S_H and S_L. The lifetime for this system is $T_{life} = \min\{E_{Hv}/(n_H \cdot p_H), E_{Lv}/(n_L \cdot p_L)\}$, where p_H and p_L are the maximum power consumption value among the sensor nodes in set S_H and S_L, respectively. The 2-level battery allocation approach increases network lifetime by moving energy capacity from low power consumption nodes to high power consumption nodes.

Figure 2 shows an example of energy allocation using different battery packs. Node A is allocated more energy because it transmits more data. Leaf nodes C and E are al-

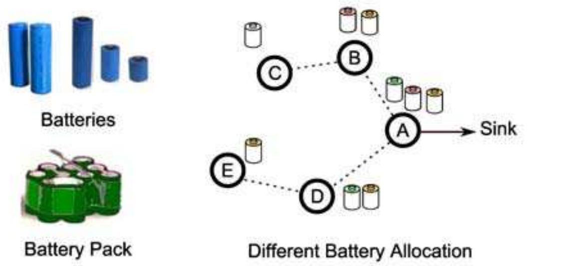

Figure 2: Battery allocation to extend the lifetime of WSN.

located less energy. This example motivates the primary research questions addressed in this paper: *Given a budget of energy cost for a WSN with arbitrary topology and power distribution, how should battery energy be assigned to sensor nodes to maximize the network lifetime given a constraint on total cost and on the number of different battery pack types?* The solution should include the lifetime-aware node partitioning to enable battery pack capacities and battery allocations to be determined.

3. PRACTICAL BATTERY ALLOCATION

This section describes battery allocation methods for realistic situation when several battery pack energy levels and their costs are given. An integer nonlinear program (INLP) problem formulation is given.

3.1 General INLP Formulation

Although batteries can be combined into packs in numerous ways, fabricating and deploying many different types of battery packs is complex and expensive. In practice, it is expected that a few battery pack types will be used, and that this will be sufficient to significantly improve WSN lifetime. Therefore, we formulate the battery allocation problem with a constraint on the number of pack levels in addition to a cost constraint. The objective is to achieve a maximum network lifetime under these constraints.

Let $P = (p_1, p_2, \ldots, p_n)$ be the power distribution of the $n-$node network. There are m types of batteries with energy capacities $E_1, E_2, \ldots, E_m$ and costs $C_1, C_2, \ldots, C_m$. The relationship between energy and cost is represented as follows: $C_i = f(E_i), i = 1, \ldots, m$. By combining battery units into packs, M types of battery pack levels $E_{pk\{1\}}, E_{pk\{2\}}, \ldots, E_{pk\{M\}}$ are achieved. If $\omega(i,k)$ denotes the number of battery units k assigned to battery pack i, then $E_{pk\{i\}} = \sum_{k=1}^{m} \omega(i,k) \cdot E_k$. Each node is equipped with one battery pack. The problem is formulated as follows: given a cost constraint $C_{total} \leq C_{cons}$ and the number of battery pack levels M, determine the numbers of the battery units in the M levels allocated to different nodes to maximize the network lifetime. To do the allocation, the sensor nodes are divided into M sets: $L_1, L_2, \ldots, L_M$. Each L_i has N_i nodes, and each node in L_i is assigned a battery pack $E_{pk\{i\}}$, $i = 1, 2, \ldots, M$.

The working time of node set L_i is

$$T_i = \frac{E_{pk\{i\}}}{g_i} = \frac{\sum_{k=1}^{m} \omega(i,k) \cdot E_k}{g_i}, \ i = 1, \ldots, M \qquad (1)$$

where g_i is the power of the most consuming node in L_i. The lifetime of the system is therefore

$$T = \min_{i=1}^{M} T_i \qquad (2)$$

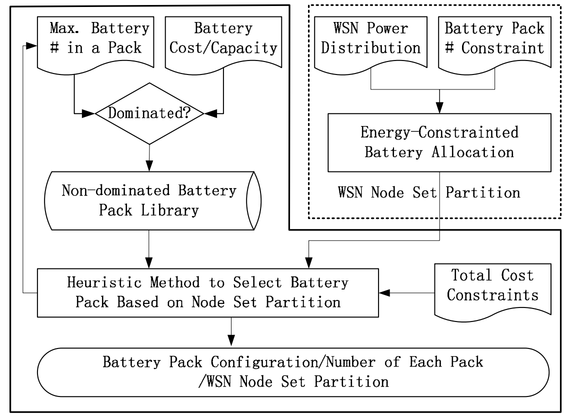

Figure 3: Proposed Method Flow: Optimal Partition and CLPS Algorithm.

The total cost can be represented as

$$C_{total} = \sum_{i=1}^{M} \left(N_i \cdot \sum_{k=1}^{m} \omega(i,k) \cdot f(E_k) \right) \qquad (3)$$

The optimization objective is now formulated as follows:

$$T = \min_{i=1}^{M} \left(\frac{\sum_{k=1}^{m} \omega(i,k) \cdot E_k}{g_i} \right) \ \rightarrow \ \max \qquad (4)$$

subject to the following constraints:

1. $\sum_{i=1}^{M} \left(N_i \cdot \sum_{k=1}^{m} \omega(i,k) \cdot f(E_k) \right) \leq C_{cons}$ and
2. $\omega(i,k)$ is a nonnegative integer, $i = 1, \ldots, n$ and $k = 1, \ldots, M$.

Given a capacity-price function $C_i = f(E_i), i = 1, \ldots, m$, the maximum working time optimization problem is formulated as an INLP that can be solved by standard INLP solver, such as LINGO.

3.2 Complexity Analysis

The time cost of running a general INLP solver on large problem instance of the battery energy allocation problem is excessive. Any node can be equipped with a mix of batteries of different types, making search space large. For a test case with n nodes and m battery types, the size of the search space is at least $O((Dim + 1)^{m \times n})$, where Dim is a constant specifying the maximum number of batteries of each type assigned to each battery pack. For example, if $Dim = 3$, 0–3 batteries may be assigned to each pack. The time complexity could therefore be as high as $\left(C_{4^m}^{M}\right)^n$. In the case when $m = 5$, $M = 10$, and $n = 100$, the search space could have 10^{2300} points. Although intelligent search algorithms can reduce the visited search space, our experimental results in Section 6 indicate that a general-purpose commercial solver fails to solve the formulation in 24 hours for networks with more than 64 nodes. Section 4 therefore describes an efficient heuristic.

4. PROPOSED ALGORITHM

This section proposes a fast heuristic for the battery pack energy assignment problem.

4.1 Overview of Heuristic

Figure 3 illustrates the proposed algorithm. The first step is shown in the dotted block, in which the sensor nodes are partitioned into M sets to allow the lifetime of the wireless sensor network to be optimized in the second step. Based on the partitioning, the second step presents a Cost Limited Pack Select (CLPS) heuristic to choose proper battery pack configurations for each set. Finally, the node set partition, the corresponding battery pack configuration, and the lifetime of wireless sensor network are produced.

4.2 Optimal Partition for Maximal Lifetime

This section describes a technique to divide the sensor nodes into M sets to achieve the maximal lifetime improvement. It should be noted that given the spatial power distribution and a bound on the number of energy levels, Theorem 1 shows that the optimal node partition is independent on the total energy E_{cons}. As the capacity–cost model in Section 4.3.1 has shown, there is a nearly linear relationship between cost and energy for battery packs. It will permit us to find the optimal node partition independently under given cost constraints.

We now present Theorem 1 to bound the network lifetime extension for a given node partition based on the energy constraint and the condition to achieve the optimal node partition.

THEOREM 1. *Given an energy constraint E_{cons} and the power distribution $p_1, p_2, \ldots, p_n$, if E_{cons} can be continuously allocated, the network lifetime under any node partition of M sets will at most be*

$$T = \frac{E_{cons}}{\sum_{i=1}^{M}(g_i \cdot N_i)}, \tag{5}$$

where g_i is the maximum value of the power consumption in set L_i, and N_i is the number of nodes in L_i.

Theorem 1 is proven in [19]. It gives the maximum lifetime of a given node partition under ideal energy allocation, assuming that continuous energy allocation is permitted. This assumption provides the lifetime upper bound in Section 6, however it can not be met in most real cases with discrete battery capacities. According to Equation 5, the maximum lifetime is obtained when the appropriate g_i and N_i in the optimal node partition minimize $\sum_{i=1}^{M}(g_i \cdot N_i)$. It is obvious that the partition is independent on the total energy E_{cons}. Next, we will formulate the optimal partition problem as an INLP problem and state how to solve it.

By sorting $p_1, p_2, \ldots, p_n$ in ascending order as $q_1, q_2, \ldots, q_n$, the optimization objective is

$$V_{disc} = \sum_{i=2}^{M+1}(q_{x_i} \times (x_i - x_{i-1})) \rightarrow \min, \tag{6}$$

where x_i is the objective variable and $i = 1, 2, \ldots, M+1$ represent locations for the partitioning points in the sorted power distribution $q_1, q_2, \ldots, q_n$. The constraint conditions follow:

1. x_i is a nonnegative integer, $i = 1, \ldots, M+1$,
2. $x_1 = 0$, $x_{M+1} = n$, and
3. $x_i > x_{i-1}$, $i = 2, \ldots, M+1$.

This INLP problem differs from traditional nonlinear programming because the objective variables are the subscripts of a discrete mapping. The objective function (Equation 6) cannot be expressed as an elementary function, preventing standard solution techniques. It should be noted that the power consumption sequence $q_1, q_2, \ldots, q_n$ is monotonically increasing. If the discrete mapping is represented as a piecewise-continuous function $\mathfrak{q}(x)$, without the integer constraint on objective parameters, the problem can be solved by a standard INLP solver. The regressive function $\mathfrak{q}(x)$ is defined as:

$$\mathfrak{q}(x) = q_{\lfloor x \rfloor} + (q_{\lfloor x \rfloor + 1} - q_{\lfloor x \rfloor})(x - \lfloor x \rfloor), \tag{7}$$

where $\lfloor x \rfloor$ is the floor of x. The optimization objective is therefore

$$V_{cont} = \sum_{i=2}^{M+1}(\mathfrak{q}(x_i) \times (x_i - x_{i-1})) \rightarrow \min \tag{8}$$

subject to

1. $x_1 = 0$, $x_{M+1} = n$ and
2. $x_i - x_{i-1} \geq 1$, $i = 2, \ldots, M+1$.

Now that the optimization is described as a standard *non-linear programming* problem, it is amenable to classical algorithms and tools, e.g., LINGO and MATLAB.

After obtaining the optimal x_i of the continuous objective function, the near-optimal discrete solution is produced by rounding each x_i to $[x_i]$. The rounded solution is defined as:

$$V_{round} = \sum_{i=2}^{M+1}(\mathfrak{q}([x_i]) \times ([x_i] - [x_{i-1}])) \rightarrow \min, \tag{9}$$

where x_is are the solution of the continuous problem (Equation 8), and $[x_i]$ is the rounded values of x_i. We use the solution of Equation 9 to approximate that of Equation 6. Rounding error is discussed in Section 5.

The above statement provides a method of obtaining the energy-constrained node partition, which we used to approximate the cost-constrained partition. An example of the partition result is shown in Figure 4, with a 100 node network and 4 energy levels.

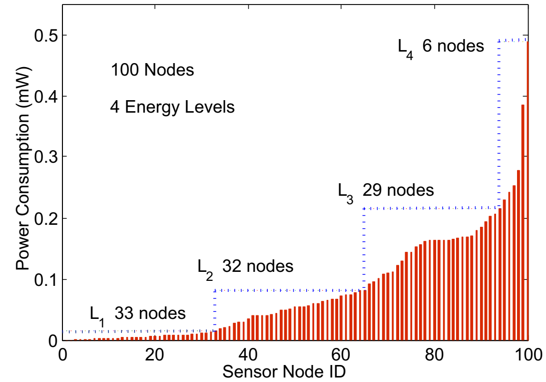

Figure 4: Optimal 4-level Energy Allocation for a 100-node Network.

It should be noted that the above INLP node partitioning problem can be solved much faster than the battery allocation problem in Section 3 because (a) the search space of the node partitioning is much smaller and (b) the proper initial value of node partition can be easy given by average node sets, while that of the battery allocation is hard to determine. Section 6 will show the execution time of the battery allocation by a commercial solver is extremely high.

4.3 Heuristic Method to Select Battery Pack

The previous section provided an algorithm to obtain the optimal node partition and the relative energy partition ratio for each set. This section will present a Cost Limited Pack Selection (CLPS) algorithm to transform the energy-constrained solution to a cost-constrained battery pack allocation. This section is organized as follows. First, we present a cost model for battery packs is built based on data from battery vendors. Second, we present a method to allocate battery packs to node sets based on the energy ratios described in the previous section. Finally, the CLPS algorithm is designed based on the battery pack cost model and energy assignment procedure.

4.3.1 Energy-Cost Model for Battery Pack

Compared with manufacturing a specific battery with customer specified capacity, battery packs provide a less expensive way to acquire batteries with various volumes under a cost constraint. Many standard batteries, such as alkaline, lithium, and NiMH batteries are available. By using standard batteries, battery packs with different capacities can be obtained. In order to build an energy–cost model for battery pack, a battery capacity–price list may be used. In this paper, we adopted a real capacity–price model for NiMH AAA batteries from the website of PowerStream [20], which is listed in Table 1. Note that our modeling method also applies to other battery packs consisting of alkaline or lithium batteries. We observed similar capacity-price trends in the alkaline or lithium battery packs [20].

Algorithm 1 BatComb

Input: *Dim*, *BatUni*, *CostUni*
Output: *PackLev*, *Cost*, *Comb*
1: $PackLev = null$ (empty set)
2: $Cost = null$
3: **for** $i_1 = 1$ to *Dim* **do**
4:   **for** $i_2 = 1$ to *Dim* **do**
5:     ...
6:     **for** $i_m = 1$ to *Dim* **do**
7:       Pick $(i_1, i_2, \ldots, i_m)$ batteries of each unit type from *BatUni*, calculate its cost.
8:       Add $(i_1, i_2, \ldots, i_m)$ into *Comb*, along with its energy into *PackLev* and its cost into *Cost*.
9:     **end for**
10:     ...
11:   **end for**
12: **end for**
13: Sort *PackLev* and *Cost* in increasing order.
14: Remove dominated combinations.

Next, we propose an algorithm to build the energy-cost model for the battery pack. Algorithm 1 is designed to find all non-dominated battery combinations for all possible battery packs. A battery combination for a pack is non-dominated if no other battery combination has both the same or lower price and the same or higher energy capacity. The input *Dim* denotes the maximum number of each kind of battery type in one pack. If $Dim = 3$, the number of each type of battery in a pack can range from 0–3. Line 3–12 show the process of enumeration of all possible combination given *Dim*. After achieving all possible levels, the dominated battery combinations are removed (Line 14). The Pareto curve of energy-cost relationship for all battery combinations with 6 battery types and $Dim = 3$ is shown in Figure 5.

As Figure 5 has shown, the Pareto energy–cost relationship for battery pack is near-linear in most ranges. The battery with the lowest price per energy unit is used as much as possible, while other types of batteries are seldom used. Therefore, the energy–cost Pareto curve can be approximated by a linear function $C = a + b \cdot E$. The fitting error is analyzed and evaluated in Section 5 and Section 6, which validates the accuracy of this approach. This property greatly simplifies optimization, allowing the cost constraint to be transformed into an energy constraint.

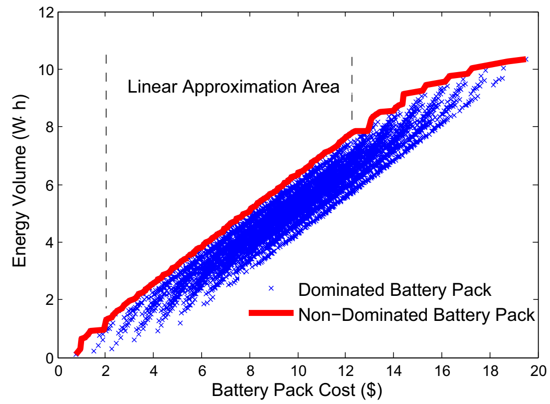

Figure 5: Non-Dominated Battery Pack Selection from the Battery Unit Combinational Usage.

4.3.2 Energy Assignment and Quantization

Algorithm 2 PackAssign

Input: *Ref*, *PackLev*, *Cost*, *G*
Output: *T*, *TotCost*, *Alloc*
1: Assign $PackLev(Ref)$ to each node in set L_M.
2: **for** each node set $L_i \neq L_M$ **do**
3:   Assign $\lceil \frac{G(i)}{G(M)} \cdot PackLev(Ref) \rceil$ to each node in L_i.
4: **end for**
5: Calculate T and *TotCost*, record the allocation *Alloc*.

Given the maximum battery pack and relative energy ratio from node partitioning, the energy assignment procedure allocates proper battery packs with various capacities to each node set. The sensor nodes in each set are equipped with battery packs having the same capacity. The input of Algorithm 2 contains the combinational energy capacity vector *PackLev*, the corresponding price vector *Cost* from Algorithm 1, the reference index *Ref*, and the power vector G, where each $G(i)$ refers to the maximum power consumption of sensor node in separate node sets L_i, $i = 1, 2, \ldots, M$. In Line 1, the reference level $PackLev(Ref)$ is assigned to the most power consuming node set L_M. For each other node set, their energy capacity values are given out by multiplying the energy ratio $\frac{G(i)}{G(M)}$ given by the energy-constrained partition (Section 4.2). In order to map those energy capacity values to real battery packs, the pack with the nearest energy capacity from *PackLev* is chosen. The output of the algorithm includes the network lifetime T, the total battery price *TotCost*, and the battery allocation result *Alloc*.

4.3.3 Cost Limited Pack Selection

Based on the energy-cost model and energy assignment procedure, CLPS (Algorithm 3) solves the entire battery allocation problem to achieve the maximum network lifetime and limits the total cost to C_{cons}. The first phase of Algorithm 3 determines the maximum number of each type of battery in one pack *Dim* (Line 1–11). The algorithm begin its search from $Dim = 1$. By calling Algorithm 1, the combinational energy capacity vector *PackLev* and the corresponding price vector *Cost* are obtained. Those vectors are given to Algorithm 2, which generates a battery pack allocation. If the cost of the allocation is less than C_{cons}, *Dim* is increased. The second phase determines the final allocation (Line 12–15). By reducing *Ref* incrementally, a cost just below and close to the cost constraint C_{cons} is achieved.

Table 1: Capacity-Price Relationship of PowerStream NiMH AAA Battery

Capacity (Amp-Hours)		0.10	0.12	0.30	0.65	0.70	0.80	0.90
Unit Price ($)	*Quantity* = 100	0.90	0.94	1.12	1.22	1.36	1.54	1.68
	Quantity = 1000	0.76	0.79	0.94	1.01	1.13	1.28	1.39

The battery energy allocation *Alloc* and the battery combination *Comb* are produced.

Algorithm 3 CLPS

$Dim \leftarrow 1$.
while 1 **do**
  ($PackLev$, $Cost$, $Comb$) $\leftarrow$ BatComb (Dim, $BatUni$, $CostUni$).
  $Ref \leftarrow$ index of the Maximum $PackLev$.
  (T, $TotCost$, $Alloc$) $\leftarrow$ PackAssign (Ref, G, $PackLev$, $Cost$).
  if $TotCost > C_{cons}$ **then**
    Break.
  else
    $Dim \leftarrow Dim + 1$.
  end if
end while
while $TotCost > C_{cons}$ **do**
  $Ref \leftarrow Ref - 1$.
  (T, $TotCost$, $Alloc$) $\leftarrow$ PackAssign (Ref, G, $PackLev$, $Cost$).
end while
Output *Alloc* and *Comb*.

5. ERROR ANALYSIS

In order to control execution time, the proposed method makes three approximation that may introduce errors.

5.1 Node Partition Error Bound

When we interpolate the power distribution vector to do node partitioning, the relative error has an upper bound.

THEOREM 2. *For the battery allocation problem defined as Equation 6, the minimum value of the objective function satisfies the following constraint:*

$$|\frac{V_{round} - V_{disc}}{V_{round}}| \leq |\frac{V_{round} - V_{cont}}{V_{round}}|, \tag{10}$$

where V_{cont} and V_{round} are defined in Equation 8 and Equation 9.

Theorem 2 is proven in [19]. It provides an error bound for the proposed node partitioning method. Experimental results in Section 6 show that the rounding error for the continuous solution is small. Therefore, Equation 9 well approximates Equation 6.

5.2 Linear Fitting Error

The battery pack modeling phase is subject to error as a result of energy–cost linear fitting. Simulations show that most battery packs are found in the linear region of Figure 5. This is also validated in Section 6. Note that this might not hold for some sets of batteries.

5.3 Energy Level Quantization Error Bound

The optimal node and energy partition makes the lifetime of each node set equivalent. However, in practice designers are limited to discrete battery pack levels. The quantization of energy level thus introduces error. The lifetime can be compared with the (ideal) continuous-energy lifetime, yielding the following expression:

$$\mathcal{R} = \frac{|T_{real} - T_{cont}|}{T_{cont}}, \tag{11}$$

where T_{real} and T_{cont} denote the network lifetime for the discrete and continuous cases. Based on Equations 1 and 2, T_{real} can be represented as $\min_{i=1}^{M} E_{pk\{i\}}/g_i$. By assuming that the first failure node is in set L_t, the relative error can be rewritten as

$$\mathcal{R} = \frac{|E_{pk\{t\}}/g_t - E'_{pk\{t\}}/g_t|}{E'_{pk\{t\}}/g_t} = \frac{|E_{pk\{t\}} - E'_{pk\{t\}}|}{E'_{pk\{t\}}}, \tag{12}$$

where $E'_{pk\{t\}}$ is the energy assigned to every node in set L_t in the continuous case. The numerator of the fraction is the quantization error, which could not be larger than the energy of the minimum battery pack difference E_{dmin}. T_{real} should be no more than T_{cont}, so $E_{pk\{t\}} \leq E'_{pk\{t\}}$, the relative error is therefore bounded as

$$\mathcal{R} \leq \frac{E_{dmin}}{E_{pk\{t\}}}, \tag{13}$$

where E_{dmin} is the minimum battery pack difference, $E_{pk\{t\}}$ is the energy assigned to the first failed node.

6. EVALUATION

This section describes our experimental setup and compares the battery allocation approach with uniform battery approach. The proposed algorithm is then compared with a standard INLP solver and deviation from optimality for the CLPS algorithm is evaluated.

6.1 Experimental Setup

To evaluate the proposed CLPS algorithm, we assumed a µAMPS-1 node [21] based WSN platform [18] when determining sensor node power distribution and network model. The number of sensor nodes ranges from 36–900. The average node-to-node distance d_0 is 20 m, and the transmission parameters are extracted from a real hardware platform [21]. Adaptive clustering and compression techniques are considered and the data aggregation model is obtained from differential compression and real data [22]. Based on these configurations, we obtained the power distribution of the WSN, $P = (p_1, p_2, \cdots, p_n)$. For the WSN containing 100 nodes, the difference between the maximum and minimum single node power consumptions varies by 2–3 orders of magnitude in our experiments, which can be observed in Figure 4. The energy–cost model for battery packs is built based on the data in Table 1. Six type batteries are considered in our experiments. The proposed CLPS method is implemented in MATLAB running on a PC with a 2.67 Ghz Intel processor with 2 GB RAM.

6.2 Network Lifetime Impact

This section discusses the impact of the proposed technique on network lifetime. The cost budget is converted to an energy budget as described in Section 6.4.2 and the reference strategy is defined as the continuous energy allocation with the continuous node partition (Equation 8), which is unreachable in reality. The reference solution obtains equal

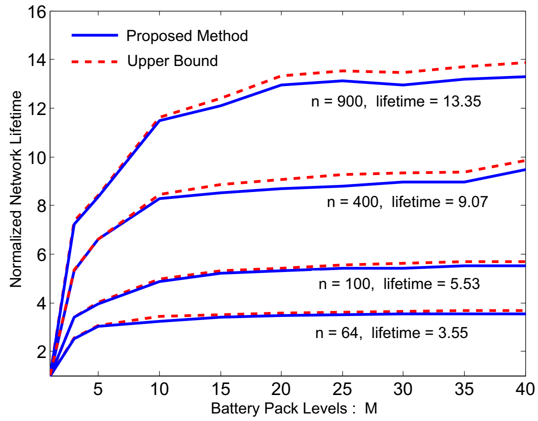

Figure 6: Normalized lifetime for different battery pack levels and network sizes.

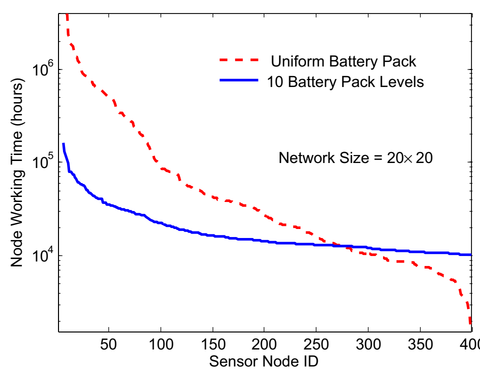

Figure 7: Lifetime difference in different nodes of uniform and 10 battery pack levels in a 400-node network.

lifetimes for each node set. Simulations are run on networks with different sizes and numbers of battery pack levels.

Figure 6 illustrates the impact of battery pack level count (1–40) on the network lifetime compared with the traditional uniform battery allocation. Network sizes ranging from 64 to 900 nodes are considered. Given a cost budget, the network lifetime (normalized by that of uniform-level battery) increased by up to 13×. The benefit increases with network size. The normalized lifetime increase for a WSN ranges from 3.55× (for 64 nodes) to 13.35× (for 900 nodes). This results from the greater node power consumption heterogeneity in larger networks.

The differences between the discrete and continuous methods in Figure 6 are small: discretization leads to only small deviations from optimality. Increasing the number of battery pack levels M initially has an large effect on lifetime but this effect saturates. In our experiments, 10 battery pack levels were sufficient to improve lifetime from 3–11×, depending on network size. It implies that a reasonable number of battery pack level would be enough for real sensor deployment.

Theorem 1 indicates that the ideal battery allocation would lead to the equal lifetime for each node set. Figure 7 shows the lifetime difference of each sensor node in a 20×20 network with 10 battery pack levels. Lifetime differs by nearly 7 orders of magnitude from node to node if only a single battery pack levels is used. The proposed method with 10 battery pack levels largely balances the node lifetimes.

Table 2: Time & Performance Comparisons between CLPS and LINGO

Network Size	Normalized Lifetime			Execution Time (s)	
	CLPS	LINGO	Ref	CLPS	LINGO
36	3.22	1.93	3.26	0.004	125
49	3.31	1.99	3.38	0.004	405
64	3.59	3.13	3.70	0.015	1.06×10^5
100	5.63	N/A	5.80	0.010	N/A
400	9.58	N/A	10.43	0.014	N/A
900	15.11	N/A	15.59	0.067	N/A

6.3 Performance Comparisons with LINGO

As stated in Section 3.2, the time to solve the problem using a general-purpose INLP solver was excessive. We now compare the proposed algorithm with LINGO, a popular solver for linear and nonlinear programming. The best-case continuous-level results are also provided to bound deviation from optimality. Since LINGO could not handle node partitioning, we do not limit the number of the battery pack types in the comparison. A branch-and-bound solver and a default iteration number are used in LINGO. The results are listed in Table 2.

For the settings in Table 2, the proposed CLPS method gains an up to 2–5 orders of magnitude speedup over LINGO. Furthermore, our approach can solve the battery energy allocation problem for WSNs with 400 nodes in than 0.02 seconds while LINGO fails to find the local optimal solutions after 24 hours. In the small cases, our approach gives even better solutions than the locally optimal solutions by LINGO; LINGO does not necessarily provide the globally optimal solutions. Our approach considers the characteristics of solution to reduce the search space. It may be possible to provide LINGO other configurations to get a better result using more execution time. However, this comparison provides evidence that straight-forward use of a general-purpose INLP solver is inappropriate for the WSN battery energy allocation problem, and that the proposed solution rapidly produces high-quality results.

6.4 Error Analysis

In this section, we experimentally quantify the sources of suboptimality for the proposed technique, which are described in Section 5.

6.4.1 Error Bound in Node Partition

Theorem 2 states that the relative error of the rounded solution V_{round} and the discrete solution V_{disc} is no more than that between the rounded solution V_{round} and the continuous solution V_{cont}. We can therefore bound the error of the approximate method to between V_{round} and V_{cont}. The relative errors in a wide range of network sizes and energy levels are listed in Table 3. The less than 2‰ error implies that the approximate partitioning method is accurate enough to solve the INLP problem.

6.4.2 Error of Battery Pack Energy-Cost Fitting

The relationship between battery packs and their costs are fitted by a linear function $C = a + b \cdot E$, permitting some error. A least squared error fitting algorithm is used [23]. The resulting correlation coefficients are listed in Table 4, where the maximum number of each battery type Dim ranges from

Table 3: Relative Error Bound of Partition(‰)

Network Size		36	49	64	100	400	900
Energy Level M	5	0.66	0.15	0.83	0.18	0.14	0.40
	10	0.93	0.56	0.45	0.47	0.18	1.25
	15	0.23	0.19	0.11	0.35	0.34	0.82
	20	0.24	0.58	0.10	0.71	0.13	0.23

Table 4: Energy-Cost Model Parameters and Linear Correlation Coefficient

Dim	3	4	5	6
a	0.0077	-0.0018	-0.0161	0.0319
b	1.8983	1.8925	1.8906	1.8777
Coef. r	0.9994	0.9999	0.9998	0.9999

3 to 6. The linear fitting relative coefficients are close to 1, which means characterizing the energy–cost relationship as a linear function is reasonable.

6.4.3 Error Bound in Discrete Battery Pack Level Selection

The error bound between the discrete battery pack levels and continuous energy levels is discussed in Section 5. From Figure 6, we found that the lifetime difference between the proposed method and the continuous (ideal case) method is much less than the error bound given in Equation 13. The error bounds and the errors in simulations are shown in Table 5.

7. CONCLUSIONS

Energy budgets are tight for low-cost battery-powered WSN nodes. Unbalanced power distributions due to the intrinsic many-to-one traffic in WSNs results in uneven battery depletion and short WSN lifetimes. Previous energy balancing techniques considered homogeneous node WSNs with uniform battery capacities. This paper formulates the cost-constrained heterogenous WSN battery allocation problem as an INLP and provides a fast heuristic that produces near-optimal solutions. Experimental results show that the proposed techniques can provide 3–11× lifetime improvement with no more than 10 battery pack levels compared with uniform battery allocation. Compared with popular INLP solver, the proposed method not only gains a 2–5 orders of magnitude speed improvement and produces better results.

Acknowledgement

This work was supported in part by the NSFC under awards 60976032, in part by the 863 Program under award 2009AA01 Z130 and in part by the NSF under awards CNS-0721978 and CNS-0347941.

8. REFERENCES

Table 5: Error Bound of Quantization (%)

Network Size	36	49	64	100	400	900
Error Bound	5.45	6.50	12.0	7.11	5.10	4.90
Real Error	0.72	0.51	0.66	0.78	0.73	0.71

[1] C. Y. Chong and S. P. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proc. of the IEEE*, vol. 91, pp. 1247–1256, 2003.

[2] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *"Proc. Int. Conf. Computer Communications*, vol. 3, 2002.

[3] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proc. International Workshop on Wireless Sensor Networks and Applications*, 2002, pp. 88–97.

[4] M. Hempstead, N. Tripathi, P. Mauro, G. Wei, and D. Brooks, "An Ultra Low Power System Architecture for Sensor Network Applications," *ACM SIGARCH Computer Architecture News*, vol. 33, no. 2, pp. 208–219, 2005.

[5] M. Sheets, F. Burghardt, T. Karalar, J. Ammer, Y. Chee, and J. Rabaey, "A Power-Managed Protocol Processor for Wireless Sensor Networks," in *Proc. IEEE Symp. VLSI Circuits*, 2006, pp. 262–263.

[6] S. Pattem, B. Krishnamachari, and R. Govindan, "The impact of spatial correlation on routing with compression in wireless sensor networks," in *Proc. Int. Symp. Information Processing in Sensor Networks*, Apr. 2004, pp. 28–35.

[7] L. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *Proc. Distributed Computing Systems Workshops*, 2002, pp. 575–578.

[8] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. Hawaii Int. Conf. on System Sciences*, Jan. 2000, pp. 3005–3014.

[9] G. Gupta and M. Younis, "Performance evaluation of load-balanced clustering of wireless sensor networks," in *Proc. Telecommunications*, vol. 2, 2003.

[10] C. Huang, H. Lee, and Y. Tseng, "A Two-Tier Heterogeneous Mobile Ad Hoc Network Architecture and Its Load-Balance Routing Problem," *Mobile Networks and Applications*, vol. 9, no. 4, pp. 379–391, 2004.

[11] J. Luo and J. Hubaux, "Joint mobility and routing for lifetime elongation in wireless sensor networks," in *"Proc. Int. Conf. Computer Communications*, vol. 3, 2005.

[12] X. Wu, G. Chen, and S. K. Das, "Avoiding energy holes in wireless sensor networks with nonuniform node distribution," *IEEE TRANS. on Parallel and Distributed Systems*, vol. 19, no. 5, pp. 710–720, 2008.

[13] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proc. Aerospace Conf.*, vol. 3, Apr. 2002, pp. 1125–1130.

[14] S. R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," in *Proc. Global Telecommunications Conf*, vol. 1, 2003.

[15] E. Oyman and C. Ersoy, "Multiple sink network design problem in large scale wireless sensor networks," in *Proc. International Communications Conf*, vol. 6, 2004.

[16] Y. Hou, Y. Shi, H. Sherali, and S. Midkiff, "Prolonging sensor network lifetime with energy provisioning and relay node placement," in *Sensor and Ad Hoc Communications and Networks*, 2005.

[17] M. Sichitiu and R. Dutta, "Benefits of multiple battery levels for the lifetime of large wireless sensor networks," *Proceedings of Networking*, pp. 1440–1444, 2005.

[18] H. Long, Y. Liu, X. Fan, R. Dick, and H. Yang, "Energy-efficient spatially-adaptive clustering and routing in wireless sensor networks," in *Proc. Design, Automation & Test in Europe Conf*, 2009.

[19] H. Long, Y. Wang, and Y. Liu, "Wsn lifetime optimization methodology," NICS Group, Tsinghua University, Tech. Rep., 2009.

[20] "Engineering guidelines for designing battery packs," http://www.powerstream.com/BPD.htm.

[21] R. Min and A. Chandrakasan, "A framework for energy-scalable communication in high-density wireless networks," in *Proc. Int. Symp. Low Power Electronics & Design*, Aug. 2002, pp. 36–41.

[22] "Tropical atmosphere ocean project," http:// www. pmel. noaa. gov/ tao/.

[23] R. W. Farebrother, *Linear least squares computations*. CRC Press, 1988.