Wireless Sensor Network, **2012, 4, 8-17**
http://dx.doi.org/10.4236/wsn.2012.41002 Published Online January 2012 (http://www.SciRP.org/journal/wsn)

Optimal Stop Points for Data Gathering in Sensor Networks with Mobile Sinks

Junyoung Park[1], Kyoungjin Moon[2], Sungjoo Yoo[1], Sunggu Lee[1]
[1]Department of Electronic and Electrical Engineering, POSTECH, Pohang, South Korea
[2]Samsung Electronics Co., Ltd., Seoul, South Korea
Email: {reinhard, sungjoo.yoo, slee}@postech.ac.kr, k_jin.moon@samsung.com

Received October 18, 2011; revised November 27, 2011; accepted December 9, 2011

ABSTRACT

Given a wireless sensor network (WSN) in which a mobile sink is used to collect data from the sensor nodes, this paper addresses the problem of selecting a set of stop points that results in low energy usage by the sensor nodes. This paper assumes an approach in which a mobile sink travels along a fixed path and uses a stop-and-collect protocol since this has previously been shown to be an efficient WSN data collection method. The problem of selecting an optimal set of stop points is shown to be an *NP-hard* problem. Then, an Integer Linear Programming (ILP) formulation is used to derive an optimal algorithm that can be used for small problem instances. Next, a polynomial-time Tabu-search-based heuristic algorithm is proposed. Simulations are used to compare the energy consumption values, computation times and expected network lifetimes when using the optimal ILP algorithm, the proposed heuristic algorithm and several other possible heuristic algorithms. The results show that the proposed heuristic algorithm results in near-optimal energy usage values with low computation times, thereby making it suitable for large-sized WSNs.

Keywords: Wireless Sensor Network (WSN); Mobile Sink; Energy Efficiency; Optimization; Tabu Search Heuristic

1. Introduction

In recent studies, sink mobility has been considered as a promising approach to reducing energy consumption in wireless sensor network (WSN). The main idea is that a mobile sink (MS) may move around an area and collect data from sensor nodes using short-range communication. The energy consumption of sensor nodes is reduced due to a reduced data relay requirement in the WSN.

The MS can collect data using collect-on-move or stop-and-collect. In the collect-on-move mode, the MS travels along a path and collects data from nodes without stopping at any location. In the stop-and-collect mode, the MS stops at certain fixed locations to collect data. In [1], the authors propose two protocols, Adaptive Speed Control (ASC) and Stop Collect Data (SCD). In ASC, the MS travels slowly in regions where data collection conditions are moderately poor, and stops in regions where data loss is severe. In SCD, the MS stops at locations where predetermined rendezvous nodes are found. In another proposed protocol based on SCD [2], the MS uses adaptive stop-times to collect data. Compared to collect-on-move, stop-and-collect algorithms are shown to reduce data collection latency by up to 70% [2] and increase data delivery rates by up to 400% [1,2].

To fully exploit the advantages of the stop-and-collect protocol, the most important task is the selection of stop points for the MS that result in reduced energy consumption by sensor nodes while maintaining low latency and high data delivery rates; this is referred to as the Path Stop Point (PSP) problem. By selecting a set of "good" stop points for the MS, energy consumption can be decreased by reducing the amount of energy required to communicate sensed data from the sensor nodes to the MS. Based on the stop-and-collect protocol, the main contributions of this paper can be summarized as follows: 1) a formal system model is developed to model the PSP problem and to show that it is an NP-hard problem; 2) an Integer Linear Programming (ILP) approach is used to derive an optimal solution for the PSP problem; 3) a Tabu-search-based heuristic PSP algorithm and a stop points estimation method are developed and 4) extensive simulation studies are used to show that our Tabu-search heuristic algorithm executes in a reasonable amount of time and produces PSP solutions that are within a few percent of the optimal solution and consistently outperform other possible heuristic approaches to the PSP problem.

2. Related Works

This section briefly surveys previous work on how to exploit sink mobility energy conservation in WSNs. To limit the sensor data latency in a WSN (while conserving energy), the works in [3,4] formulate the "sink tour problem" as a Traveling Salesman Problem (TSP). The sink meets all sensor nodes exactly once during the tour. In similar methods presented in [5,6], the authors model the sink tour as a "label covering tour" that relaxes the requirement of exact one-time-visits of the sink to each sensor's neighborhood.

Previous works include rendezvous-based [7,8], waypoiont-based [9] and other types of methods [10-14].

A common observation that can be made from all of the previous works is that "sink-stop" methods, which require the sink to pause when collecting data (e.g., stop-and-collect), give better results than methods in which the sink collects data while in motion (e.g., collect-on-move). In "sink-stop" methods, the selection of the stop points is crucial to minimizing energy consumption. This is the problem addressed by this work.

3. System Model and Problem Definition

3.1. System Model

This paper considers the situation in which there are a large number of sensor nodes deployed in a $W \times W$ target area with the purpose of monitoring and checking for certain activities or events of interest. To gather the event data accumulated in sensor nodes, the MS moves along a fixed path *P*, with length *L*, periodically at an average speed *v* while stopping for short periods of time at several stop points selected from a predetermined set of "candidate stop points." The MS collects data from rendezvous nodes, which are chosen to be 1-hop neighbors of the stop points. All non-rendezvous nodes should periodically send their data to rendezvous nodes using shortest-length routes. A path *P* can be circular, rectilinear or any arbitrary shape, and candidate stop points are located on the path *P*.

Each sensor node in the network generates a fixed amount of data between consecutive visits by the MS. Data generated by sensor nodes are sent to the rendezvous nodes of their respective stop points. When the MS reaches a stop point, it sends a beacon message to rendezvous nodes to collect data. Since the energy required for data transmission and reception is the dominant component of a node's energy usage, energy consumption required for data gathering is proportional to the average hop count from sensor nodes to their stop points. Thus, as the length of the path increases and more stop points are used, the average number of hops to sensor nodes decreases and the energy consumption required for data gathering decreases. However, since receiving beacon messages from the MS for connection establishment also consumes the energy of sensor nodes, careful selection of stop points is required.

3.2. Problem Definition and Complexity Analysis

Given the system model presented in the previous subsection, this paper considers the problem of selecting the stop points, for the MS, from among the set of candidate stop points on a given fixed data collection path. Stop points need to be determined for the MS such that the energy required for data collection from the sensor nodes is minimized. This problem is formalized as the *Path Stop Point* (*PSP*) problem.

***Problem* 1 (*PSP*):** Given a set of sensor nodes *S* and a fixed path *L* with a set *P* of candidate stop points, determine a set of stop points $S_L \subseteq P$ that result in the minimum total energy consumption for data gathering in the WSN. To balance energy consumption of each node, sensor node has energy limitation to prevent a heavy relay data.

In this subsection, it will be shown that the PSP problem is *NP-hard*. This result will be proven by showing that a restricted form of the PSP problem is equivalent to the *Uncapacitated Facility Location* (*UFL*) problem [15], which is a known *NP-hard* problem. In the UFL problem, there are a set of facilities providing a product or service and the objective is to determine a minimum-cost subset of those facilities to open, taking into account the sum of the distances from each demand point to its nearest facility and the opening cost for each facility.

***Problem* 2 (*UFL*):** This problem can be formally presented using a problem instance statement and a search question [15].

***Instance*:** The set of facilities *F*, the set of clients *C* and the distance metric. For each $j \in F$, the cost of opening facility *j*, denoted by f_j, is also given.

***Theorem* 1:** The UFL problem is *NP-hard* [15].

Although there are many similarities between these two problems, the PSP problem is quite different from the UFL problem because the energy usage and the energy capacity of all relay nodes must be considered when selecting a set of stop points in the PSP problem. Taking this into consideration, the following optimization problem can be defined and used to analyze the complexity of the PSP problem.

***Problem* 3 (*Unlimited Energy PSP*):**

***Instance*:** A set *S* consisting of *n* sensor nodes that transmit data and a set *P* consisting of *m* candidate stop points where the sink node (MS) can stop and collect data. For each sensor node $i \in S$ and each candidate stop point $j \in P$, let $e_{i,j}$ denote the energy required to send data from *i* to *j* over a least-energy path (which should be a path with the least number of hops, given that each packet transfer incurs the same energy usage). For each $j \in P$, the combined energy required to establish communication links with all neighboring rendezvous nodes using beacon packets is denoted as c_j.

***Question*:** Does a set $S_L \subseteq P$ exist such that $\sum_{j \in S_L} c_j + \sum_{j \in S_L} \sum_{i \in S} e_{i,j}$ is minimized?

***Proposition* 1:** The Unlimited Energy PSP problem is *NP-hard*.

***Proof*:** The Unlimited Energy PSP problem is equivalent to the UFL problem since the parameters *S*, *P*, $e_{i,j}$, *t* and S_L in the former problem can be converted to the parameters *C*, *F*, $dist_{i,j}$, *t* and S_F in the latter (and vice versa).

The PSP problem is a search problem extension of the Unlimited Energy PSP problem with the additional requirement that no sensor node may exceed its energy allocation while sending its own data packet or relaying packets from other sensor nodes. Since instances of the PSP problem with arbitrarily large initial energy allocations could be created, the following corollary holds.

***Corollary* 1:** The PSP problem is *NP-hard*.

4. Integer Linear Programming Formulation

Based on the definition of the Unlimited Energy PSP problem given above and the additional constraints imposed by the PSP problem, an integer linear programming formulation of the PSP problem is proposed. Once the PSP problem has been formulated as an integer linear programming problem, existing techniques for solving integer linear programming problems can be utilized to solve the PSP problem in an optimal manner. Although such a solution will have exponential time complexity in the worst case, it will nevertheless be useful for small problem instances and as a benchmark for evaluating alternative heuristic solutions.

To model the PSP problem using integer linear programming, the following notations and constraints will be defined. Let $S=\{1,\cdots,N\}$ be the set of sensor nodes (in short, sensors). Let $P=\{1,\cdots,M\}$ be the set of candidate stop points (in short, candidates). Then, $\forall i,k\in S$, $\forall j\in P$, let us define the following.

h: Maximum number of hops required for any sensor node to transmit its data to a nearby candidate.

$$H_{i,j}=\begin{cases} h_t & \text{if sensor } i \text{ reaches candidate } j \text{ in } h_t\text{<h hops} \\ h_{unreachable} & \text{else,}\end{cases}$$

$$G_{i,k}=\begin{cases} h_t & \text{if sensor } i \text{ reaches sensor } k \text{ in } h_n\text{<h hops} \\ h_{unreachable} & \text{else,}\end{cases}$$

C_j: Energy consumption by sensor nodes for connection establishment when the MS stops at point *j*

N_{total}: Number of all types of nodes in the *WSN*

N_j: Number of 1-hop neighbor sensors of candidate *j*

f_i: Number of selected stop points near sensor node *i*

g: Amount of data, in units of packets, generated at each sensor node between two consecutive visits by the MS

e_{beacon}: Energy required to receive a beacon message from the MS

e_{limit}: Energy limit per sensor node (identical for all nodes) in one round

e_{tx}: Energy required to transmit one data packet

e_{rx}: Energy required to receive one data packet

Binary decision variables are defined as follows.

$\forall i,k\in S$, $\forall j\in P$

$$x_{i,j}=\begin{cases} 1 & \text{if sensor } i \text{ is covered by candidate stop point } j \\ 0 & \text{else,}\end{cases}$$

$$y_i=\begin{cases} 1 & \text{if candidate } j \text{ is selected as a stop point} \\ 0 & \text{else,}\end{cases}$$

$$z^k_{i,j}=\begin{cases} 1 & \text{if sensor } i \text{ to point } j \text{ route goes through sensor k} \\ 0 & \text{else,}\end{cases}$$

The energy consumed for data collection from sensor node *i* while the MS travels once through the path *P* to collect data is computed as

$$E_i = g\cdot\left(\sum_{j\in P}\sum_{k\in S} z^k_{i,j}\right)\cdot\left(e_{tx}+e_{rx}\right)+f_i e_{beacon} \tag{1}$$

The first term in **Equation (1)** represents the energy consumption for relaying data generated from a sensor node *i* to the stop points. The second term is the energy required to receive beacon messages from the MS. It can also be represented as

$$\sum_{i\in S} f_i e_{beacon}=\sum_{j\in P} y_i N_j e_{beacon}=\sum_{j\in P} y_i C_j \tag{2}$$

The number of beacon messages generated by the MS is proportional to the number of stops. Also, the number of receiving nodes is proportional to the 1-hop neighbors of each stop point. Thus, the total energy consumption for connection establishment is dependent on the number of stop points and their 1-hop neighbors.

The relationship between the total amount of data received by sensor nodes and the sum of the number of hops for all nodes is represented as

$$\sum_{i\in S}\sum_{j\in P}\sum_{k\in S} z^k_{i,j}=\sum_{i\in S}\sum_{j\in P} H_{i,j} \tag{3}$$

where $H_{i,j}$ is the shortest hop count from sensor node *i* to its destination candidate stop point *j*.

According to Equations (1) to (3), the total energy consumption can be expressed as the sum of the energy consumed for data transmission from sensor nodes to stop points and the energy required to receive beacon messages. Thus,

$$\begin{aligned} E_{total} &= \sum_{i\in S} E_i \\ &= \sum_{i\in S}\left(g\cdot\left(\sum_{j\in P}\sum_{k\in S} z^k_{i,j}\right)\cdot\left(e_{tx}+e_{rx}\right)+e_{beacon}f_i\right) \\ &= g\cdot\sum_{i\in S}\sum_{j\in P}\sum_{k\in S} z^k_{i,j}\cdot\left(e_{tx}+e_{rx}\right)+\sum_{i\in S} e_{beacon}f_i \\ &= g\cdot\sum_{i\in S}\sum_{j\in P} H_{i,j}\cdot\left(e_{tx}+e_{rx}\right)+\sum_{j\in P} y_i C_j \end{aligned} \tag{4}$$

The PSP problem can then be mathematically formulated as shown below. The optimal algorithm based on the solution of this *ILP* problem will be referred to as the *ILP-PSP* algorithm.

***Integer Linear Programming* (*ILP*) *Formulation*:**

Minimize
$$g\cdot\sum_{i\in S}\sum_{j\in P}H_{i,j}x_{i,j}\cdot\left(e_{tx}+e_{rx}\right)+\sum_{j\in P}y_iC_j \quad (5)$$

Subject to:

$$\forall i\in S, j\in P \quad y_j\ge x_{i,j} \quad (6)$$

$$\forall i\in S \quad \sum_{j\in P}x_{i,j}=1 \quad (7)$$

$$\forall i,k\in S, j\in P \quad z_{i.j}^{k}\le x_{i,j} \quad (8)$$

$$\forall i\in S, j\in P \quad x_{i,j}\le\sum_{k\in S}z_{i,j}^{k} \quad (9)$$

$$\forall i\in S \quad \sum_{j\in P}z_{i.j}^{i}=1 \quad (10)$$

$$\forall i\in S \quad \sum_{j\in P}\sum_{\substack{k\in S\\ H_{j,k}=1}}z_{i.j}^{k}=1 \quad (11)$$

$$\forall i\in S, j\in P \quad \sum_{k\in S}z_{i.j}^{k}\ge H_{i,j}x_{i,j}-1 \quad (12)$$

$$\forall i\in S, j\in P \quad \sum_{k\in S}z_{i.j}^{k}\le H_{i,j}x_{i,j} \quad (13)$$

$$\forall i,k\in S, j\in P \quad H_{i.j}\ge\left(G_{i,k}+H_{k,j}\right)\cdot z_{i.j}^{k} \quad (14)$$

$$\forall i\in S \quad \sum_{j\in P}\sum_{k\in S}\left(z_{i.j}^{k}e_{tx}+z_{k.j}^{i}e_{rx}\right)\cdot g\le e_{limit} \quad (15)$$

$$\forall i,k\in S, j\in P \quad \begin{array}{l} x_{i,j}\in\{0,1\}, y_j\in\{0,1\},\\ z_{i.j}^{k}\in\{0,1\}\end{array} \quad (16)$$

The objective Equation (5) states that the objective of this problem is the minimization of the energy consumed in the WSN. Equations (6) and (7) state that each sensor node should be assigned to a single stop point. Routing paths for sensor nodes need to be found while considering energy limitations, as stated in Equations (8) through (16). Equations (8) and (9) state that a flow from sensor node *i* to candidate stop point *j* via node *k* exists only if sensor node *i* sends data to candidate stop point *j*. Equation (10) simply notes that each node *i* has an outward link in the transmission route. Equation (11) states that each data packet should eventually arrive at a stop point. Equations (12) and (13) represent constraints on the number of possible relay nodes. Equation (14) dictates that we should select a node *k* as a relay node if it results in a shortest-path route. Equation (15) bounds the energy consumption within the energy limit. Finally, Equation (16) is required since *x, y*, and *z* are binary decision variables.

5. Estimate of the Optimal Number of Stop Points

It is possible to estimate the number of stop points in an optimal solution by using a simple geometric method. First, Equation (5) can be rewritten as

$$E_{total}=g\cdot\left(\bar{H}N_{total}\right)\cdot\left(e_{tx}+e_{rx}\right)+\bar{N}ne_{beacon} \quad (17)$$

where $\bar{H}$ is the average hop count from sensor nodes to their destination stop points, $\bar{N}$ is the average rendezvous nodes per stop point and *n* is the number of stop points used. If we assume nodes and stop points are uniformly distributed in the $W\times W$ target area, $\bar{H}$ and $\bar{N}$ can be approximated as

$$\begin{aligned}\bar{H}&=\sum_{1}^{H_{\max}}k\cdot\frac{\pi(rk)^2-\pi\left(r(k-1)\right)^2}{\pi\left(rH_{\max}\right)^2}\\ &=\frac{4H_{\max}+3-1/H_{\max}}{6}\approx\frac{4H_{\max}+3}{6}\end{aligned} \quad (18)$$

$$\bar{N}=N_{total}\cdot\frac{\pi r^2}{W^2} \quad (19)$$

where $H_{\max}$ is the maximum hop count from any sensor node to its destination stop point. Since we assumed stop points are uniformly distributed, the target area is divided equally by the *n* stop points, and $H_{\max}$ is thus

$$H_{\max}=\begin{cases}\dfrac{W}{r\sqrt{n\pi}} & \text{if } n<\dfrac{W^2}{\pi r^2}\\ 1 & \text{else,}\end{cases} \quad (20)$$

Using Equations (18), (19) and (20), we can rewrite Equation (17) as

$$\begin{aligned}E_{total}(n)=&\frac{2gWN_{total}}{3r\sqrt{\pi}}\cdot\left(\frac{1}{\sqrt{n}}\right)\cdot\left(e_{tx}+e_{rx}\right)\\ &+\frac{gN_{total}}{2}\cdot\left(e_{tx}+e_{rx}\right)+\frac{\pi r^2N_{total}e_{beacon}}{W^2}\cdot n\end{aligned} \quad (21)$$

where $E_{total}(n)$ is a convex function since $\mathrm{d}^2E_{total}(n)/\mathrm{d}n^2>0$. Thus, the value of n_0 satisfying $\mathrm{d}E_{total}(n_0)/\mathrm{d}n=0$ is the number of stop points that minimizes total energy consumption. The optimal number of stop points, n_0, is

$$n_0=\begin{cases}\dfrac{gW^3\left(e_{tx}+e_{rx}\right)}{3\pi r^3e_{beacon}} & \text{if } \dfrac{gW^3\left(e_{tx}+e_{rx}\right)}{3\pi r^3e_{beacon}}<\dfrac{W^2}{\pi r^2}\\ \dfrac{W^2}{\pi r^2} & \text{else,}\end{cases} \quad (22)$$

This equation gives us a simple way to determine how the MS should move, particularly if there's no initial path given. With n_0 stop points arranged uniformly in the target area, a path passing through all those stop points should result in minimal total energy consumption.

6. Proposed Heuristic Method

Since the PSP problem is *NP-hard*, a heuristic algorithm based on Tabu search [16,17] is proposed. The complete pseudocode solution and a more detailed description are provided below.

Algorithm 1 (TABU-PSP): This is a Tabu search algorithm for path stop point selection that jointly considers the transmission routes used by sensor nodes and the MS stop frequency overhead.

Initial solution: The TABU-PSP algorithm starts with a configuration in which all candidate stop points are selected as MS stop-points. This configuration must be admissible in order for a problem instance to have a solution. The current, best, and *move* (next solution to be considered) are each represented by a vector $\langle y_{0,} y_1, y_2, \cdots, y_{n-1} \rangle$, where each y_i is a binary decision variable that denotes whether candidate stop point *i* is selected (as in the ILP formulation).

Algorithm 1: Proposed TABU-PSP algorithm

```
Input : S, P, h,  $e_{limit}$ , I, definedN
Output: x, y, z
1  current = $\langle 1,1,\cdots 1 \rangle$  // set point to be feasible
2  best = current
3  noImprove = 0
4  while I-- > 0 do // I is the number of iterations left
5    tabuIndex = 0
6    Generate redundant points set  $R \subset P$
7    move = $j \in R$  with the lowest cost, not in tabuList
8    tabu = move; tabuIndex++; j to tabuList;
9    y = current – move // drop a redundant stop point
10   current = y
11   if Score(best) < Score (current) then
12     noImprove++
13   else if TSsub(current) is feasible do
14     best = current
15     noImprove = 0
16   else
17     noImprove++
18   end-if
19   if noImprove > definedN then
20     I = -1 //stop iterating
21   end-if
22 end-while
23 Subroutine TSsub (current):
24   Construct initial solution $x_{i,j}$
25   flag: = true
26   for i in set S do
27     if $H_{i,j}$ = 1 then
28       $currentz_{i,j}^{i}$  = 1
29     else
30       Hop2Go = $H_{i,j}$
31       while Hop2Go  $\neq$  1 do
32         Construct  $z_{i,j}^{k}$  considering  $G_{i,k}$ ,  $H_{k,j}$
33         Generate k` considering  $G_{i,k}$ ,  $H_{k,j}$
34           if hop( $currentz_{i,j}^{k}$ ) > hop( $z_{i,j}^{k'}$ ) then
35             $currentz_{i,j}^{k}$  =  $z_{i,j}^{k'}$ ; Hop2Go =  $H_{k'j}$ ;
36           end-if
37       end-while
38     end-if
39    flag = checkElimit ( ) // check Equation (15)
40   end-for
41   if flag = true return feasible;
42   else return infeasible; //node energy spent > $e_{limit}$
43 End of Subroutine
```

Admissible configuration: A configuration *P* is defined by the set of routes from sensor nodes to stop points. To find a shortest-length transmission route, we assign each node to a stop point using the hop count factor in Line 24. To find a relay node *k*, we search for a candidate relay node that is at a hop-count distance of 1 and is closer to a stop point ($G_{i,k} = 1$ and $H_{k,i} < H_{i,j}$, i.e., which satisfies Equation (14)). Only feasible configurations that cover all nodes given the energy limitations for those nodes *(i.e.*, which satisfy Equations (11)-(16)) are considered.

Cost function: A configuration *P* is evaluated using the scoring function given by Equation (5).

Neighborhood investigation: A search movement consists of dropping a stop point location *j*, where *j* is selected by considering data generation rates, sensor node locations and the energy limitations of sensor nodes. In Line 6, when searching for a redundant stop point, only a fraction of the candidate stop point list is considered (in this implementation, only 15% is searched). Then, the selected redundant point is dropped if it satisfies the cost function (leads to a lower cost solution).

Aspiration criterion: Tabu movements are allowed when the score of the resulting configuration is lower than the score of the best solution *P** found thus far during the search process.

Stopping criterion: The algorithm stops if a predefined number of iterations are reached or if no improvement is observed for a predefined number of iterations.

***Theorem* 2:** The time complexity of TABU-PSP is $\left(I|S|^3\right)$, where *I* is the maximum number of iterations used.

***Proof*:** Follows from the pseudocode, the overall running time is $\left(I\left(t|P|\log\left(|P|\right)+|S|n_{\max}h\right)\right)$. The variables $n_{\max}$, *h* and $|P|$ are each smaller than $|S|$. Thus, the total running time for all iterations is $\left(I|S|^3\right)$.

7. Simulation Results

The performance of the TABU-PSP algorithm was evaluated using numerous simulations. In order to provide a comparison benchmark, the ILP-PSP algorithm, an optimal solution to the PSP problem based on the integer linear programming formulation provided in Section III, was used. ILP-PSP was implemented using XPRESS-MP [18], which is a mathematical programming and optimization tool for solving linear problems. All algorithms used in our comparison study were executed using a computer with an Intel Core i7 920 2.67 GHz CPU and 6 GB of RAM.

In these simulations, we used a fixed rectilinear path *P* with *m* bends into the interior of the target area, which is a loop version of the snake-like route used in [12]. The target area $W \times W$ is partitioned into squares of the same size and the path *P* crosses the central points of each of the squares. For this type of path *P*, exemplified in **Figure 1**, the path length *L* can be approximated as

$$L \approx 4(W-hr)+2m(W-2hr) \qquad (23)$$

7.1. Comparison with Optimal Algorithm: TABU-PSP Versus ILP_PSP

To evaluate the quality of our heuristic, we evaluated our solution with 80 sensor nodes deployed randomly using a *Beta* distribution [19] in a $60 \text{ m} \times 60 \text{ m}$ area. A *Beta* distribution is a probability distribution with two parameters, α and β that determine the degree to which randomly placed nodes will tend to "cluster" together. In a realistic node deployment, sensor nodes will tend to be deployed in a clustered manner because sensor nodes may be more densely deployed near potential target points or because there may be obstacles such as wooded areas. The MS moved along a fixed path and stopped to collect data. The initial energy of each sensor node was set to 5 J and energy usage values were set to e_{tx} = 1.6 μJ/byte and e_{rx} = 1.8 μJ/byte [20]. Each sensor node generated the same length data packet in each MS round. The size of each packet was set to 5 bytes. The sensor node transmission range was 15 m. Simulations were conducted using path lengths from 40 m to 240 m. Candidate stop points were placed along the path at five meter intervals, similar to scale length $\Delta l = 5 \text{ m}$ in [11]. The ILP-PSP and TABU-PSP algorithms were compared with different path lengths in **Figure 2**. TABU-PSP had total energy consumption values very close to ILP-PSP. The optimality gap of TABU-PSP was less than 1.5%.

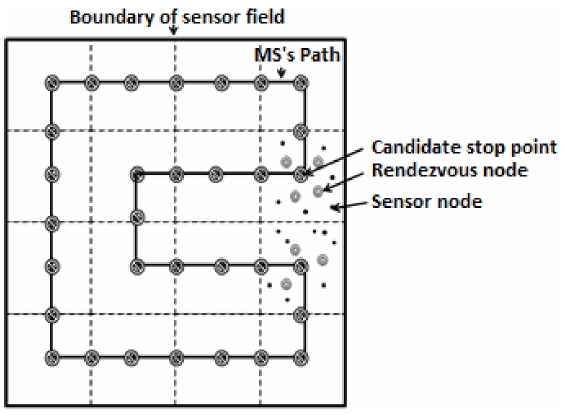


Figure 1. System model: MS path and candidate stop points.

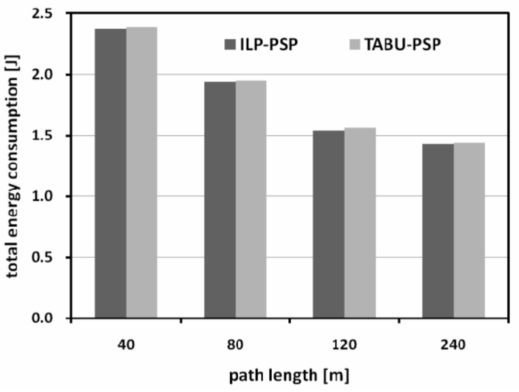


Figure 2. Comparison of the ILP-PSP and TABU-PSP.

The actual computation times of the two algorithms were also compared. Since the time complexity of TABU-PSP was determined to be $\left(I|S|^3\right)$, in Theorem 2, the actual execution time of TABU-PSP was evaluated using a 3rd-order polynomial regression line (**Figure 3**). The resulting r^2 value was greater than 0.94, which indicated that this curve closely followed the regression line trend. Also, the constant used in the third-order polynomial regression line was very small—on the order of about 10^{-7}. As a result, TABU-PSP was found to be a highly scalable algorithm that always computed its solution within a few seconds in all of our simulation experiments.

7.2. Comparison with Other Heuristic Approaches: ILP-PSP, TABU-PSP, HD-PSP, LD-PSP and Static Sink

Since TABU-PSP is a heuristic algorithm for selecting stop points, there is no guarantee that another heuristic algorithm will not perform better. Thus, four other heuristic algorithms were implemented for comparison purposes: *UNI-PSP*, *HD-PSP*, *LD-PSP* and *Static Sink*.

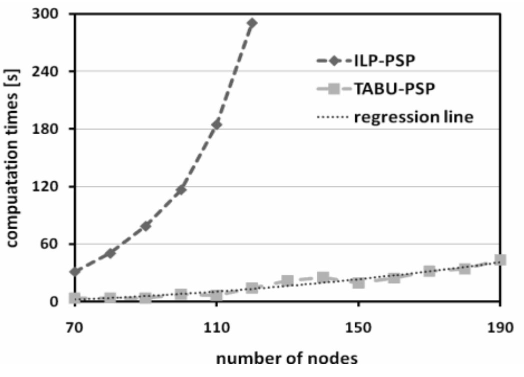
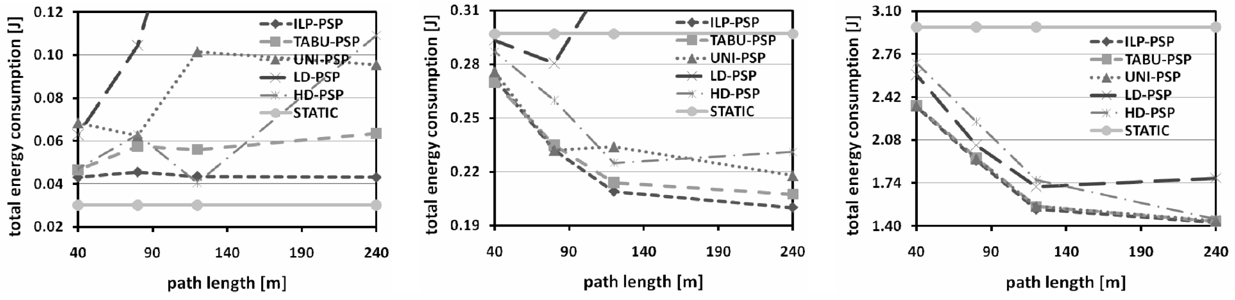


Figure 3. Computation times of ILP-PSP and TABU-PSP and a regression line analysis of TABU-PSP: path length 80 m, regression line: y = 2 × 10^{-7} x^3 – 1.4 × 10^{-4} x^2 + 0.04 x – 1.34, r^2 = 0.9434.

In UNI-PSP, stop points in the path were selected uniformly considering multi-hop factors such that stop points are selected at the centers of partitioned unit areas. Selecting uniform stop points is a good policy for WSNs in which sensor nodes are deployed uniformly [12]. In HD-PSP, stop points in the path were selected in a greedy manner, with higher preference given to candidate stop points in regions with high densities of sensor nodes. LD-PSP was an alternative greedy algorithm that selected stop points with preference given to regions with low densities of sensor nodes. In Static Sink, the sink was located in the center of the target area and did not move.

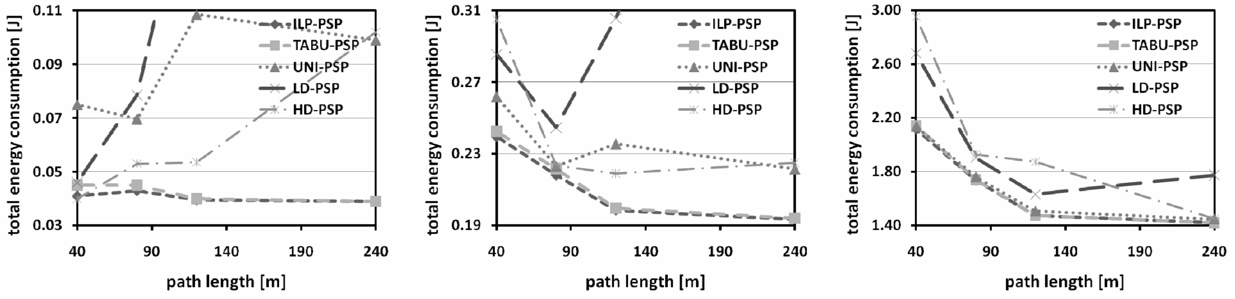
Each algorithm was compared by measuring the total energy consumed in the WSN. The data delivery deadline may change due to user demands. To consider this realistic scenario, we measured energy consumption with path lengths varying from 40 m to 240 m – given that the MS moves at a constant speed when it is moving and stops at a similar number of stop points in the various algorithms simulated, a situation with a long data delivery deadline can be modeled by using a long total path length. All algorithms were simulated in different node deployment conditions such as uniform (Beta distribution with a low degree of clustering, **Figure 4**) and highly clustered (Beta distribution with a high degree of clustering, **Figure 5**) node distributions.

The simulation results with a low-clustered (almost uniform) node deployment are shown in **Figure 4**. In all simulation scenarios, TABU-PSP was found to perform very close to ILP-PSP, the optimal solution. There were only a very few data points at which any of the other heuristic solutions outperformed TABU-PSP—the best heuristic solutions in those rare cases varied. The Static Sink solution maintained constant energy consumption levels regardless of the path length because MS did not move. In one data point instance (**Figure 4(a)**), the Static Sink solution consumed the minimum amount of energy because all other algorithms required generation of beacon messages due to MS movement (even the ILP-PSP algorithm performed worse—but this was not a contradiction as the ILP-PSP algorithm was meant to the optimal "mobile sink" solution under the conditions outlined in our system model). However, as the data generation rate was increased (**Figures 4(b)** and **4(c)**), data gathering using a MS required lower total energy consumption than a Static Sink solution since the former methods sent a lot of data over *short* data transmission routes. Of the MS methods, ILP-PSP performed the best since stop points were optimally selected while taking into consideration network conditions such as data generation rates and the beacon packet overhead for establishing communication with the MS.



(a) (b) (c)

Figure 4. Comparison of algorithms with respect to the total energy consumption in the WSN in a low-clustered node deployment with (a) 1 data packet generated per round per node, (b) 10 data packets generated per round per node and (c) 100 data packets generated per round per node.



(a) (b) (c)

Figure 5. Comparison of algorithms with respect to the total energy consumption in the WSN in a highly-clustered node deployment with (a) 1 data packet generated per round per node, (b) 10 data packets generated per round per node and (c) 100 data packets generated per round per node.

Figure 5 shows analogous performance results with a highly-clustered node deployment. In this scenario, the Static Sink solution performed so poorly (and even failed to produce valid solutions in many cases) in comparison to the other methods that it was left out in our compareson analysis. For all data generation cases and node distributions (**Figures 4** and **5**), when the path length increased, the total energy consumption decreased. This was due to the fact that longer path lengths resulted in shorter data transmission routes.

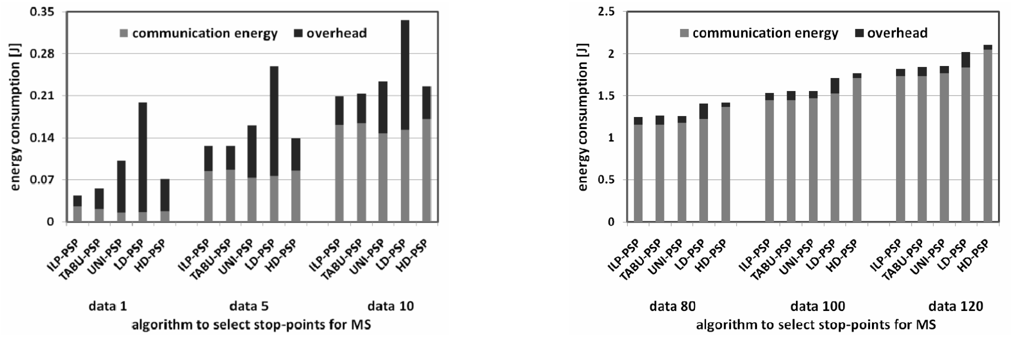
7.3. Overhead Comparison: ILP-PSP, TABU-PSP, HD-PSP and LD-PSP

The energy consumed in the WSN when using a PSP solution is composed of two components: energy used for data communication and overhead (the energy used for connection establishment with the MS). In order to analyze *how* energy was consumed in each of the algorithms simulated, plots were made of the types of energy used with each algorithm. As shown in **Figures 6(a)** and **7(a)**, when the data generation rate was low, overhead energy constituted a significant portion of the overall energy consumed. ILP-PSP, TABU-PSP and HD-PSP required relatively little overhead energy since these algorithms tended to select small numbers of stop points. When data generation rate was high (**Figures 6(b)** and **7(b)**), energy consumption due to data transmission was much larger than overhead energy. ILP-PSP and TABU-PSP both tended to select larger numbers of stop points in order to reduce the number of hops from sensor nodes to stop points, which in turn resulted in higher overhead energy at the expense of lower data transmission energy.

We also compared the number of stop points computed using our geometric estimation method (Section 5) with the optimal solution (ILP-PSP) in order to check the accuracy of Equation (22). Though the size of target area and the communication range limit the maximum number of stop points when the amount of data is larger than 5 packets, ILP-PSP with the low-clustered network showed similar results as our estimates (**Table 1**). On the other hand, the optimal solutions with the highly-clustered distribution differ significantly from our estimates, which is not unexpected since we assumed a uniformly distributed network when deriving Equation (22).

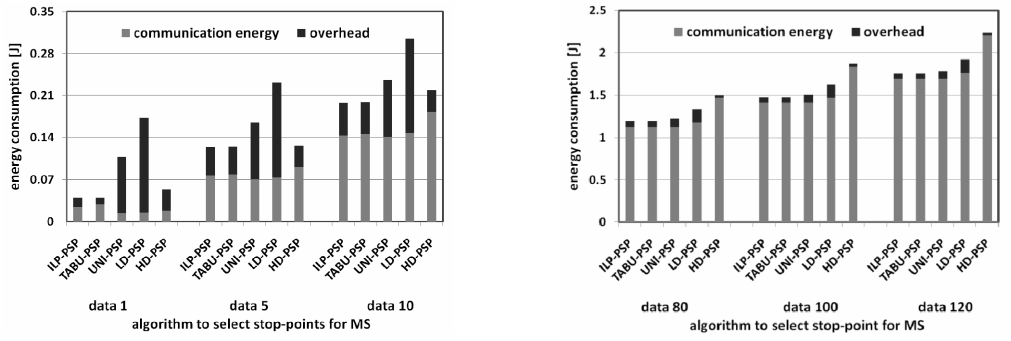
Table 1. The number of stop points (L = 240 m).

	Amount of data from each sensor node			
	1	5	10	20 ~ 120
Estimated solution	2.2	5.1	5.1	5.1
Low-clustered	2	5	6	6.5 ~ 7
Highly-clustered	8	8	9	9



(a) (b)

Figure 6. Comparison of algorithms with respect to the total energy consumption in the WSN with a 120 m path for the MS and a low-clustered node deployment: (a) low data generation rate per round, (b) high data generation rate per round.



(a) (b)

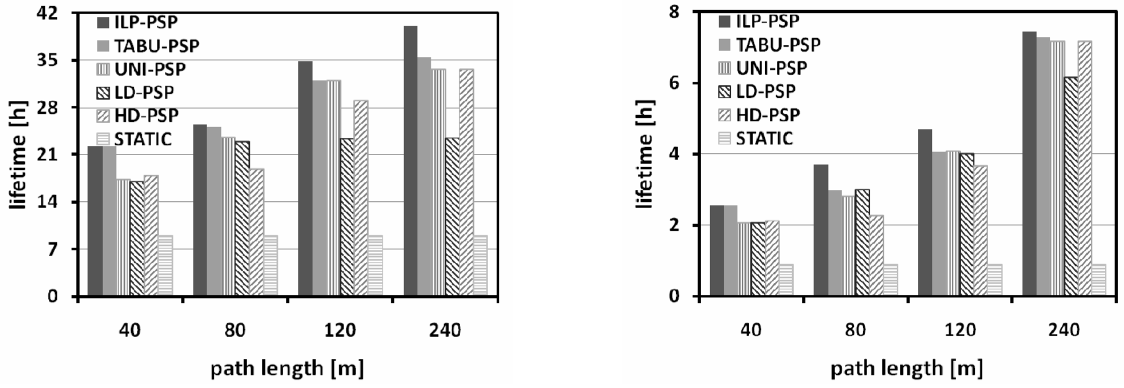
Figure 7. Comparison of algorithms with respect to the total energy consumption in the WSN with a 120 m path for the MS and a highly-clustered node deployment: (a) low data generation rate per round, (b) high data generation rate per round.

7.4. Network Lifetime Comparison

The various algorithms simulated were compared with respect to *network lifetime* values in **Figures 8** and **9**. The network lifetime is the length of time before the first sensor node dies in a WSN. When sensor nodes were deployed in a uniform manner (**Figure 8**), the network lifetimes achieved with a MS were always 2 to 7 times longer than with a static sink. When sensor nodes were deployed in a highly-clustered manner (**Figure 9**), TABU-PSP achieved approximately 3 times longer network lifetimes than the other MS heuristic methods in many cases. Since a highly-clustered node distribution tended to induce unbalanced energy consumption, selecting "good" stop points and data transmission routes was more important than with a uniform distribution. In all cases, the proposed TABU-PSP algorithm achieved longer network lifetime values than any other heuristic solution. This was because TABU-PSP selected stop points that resulted in balanced node energy consumption values—these stop points produced transmission routes (from sensor nodes to stop points) that utilized intermediate nodes with large levels of remaining energy.

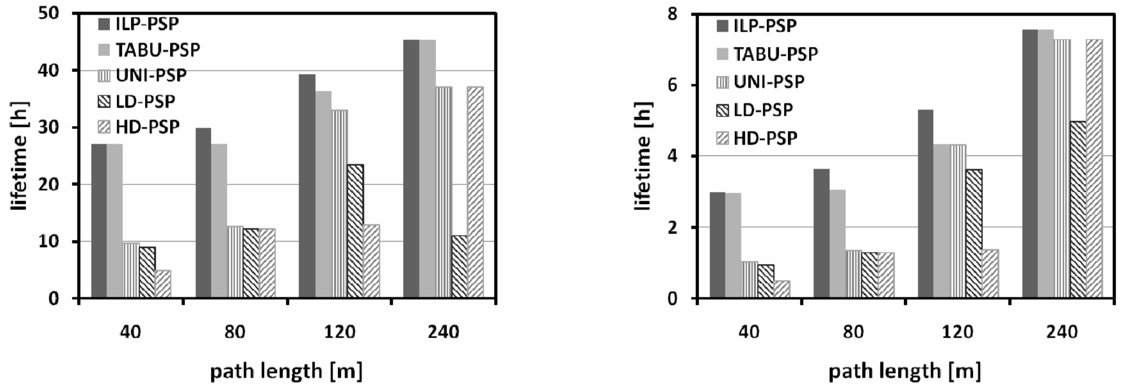
8. Conclusion

This paper has considered the problem of selecting stop points that result in minimal energy consumption usage in a wireless sensor network with a mobile sink. This problem has been formally modeled as an integer linear programming problem that is aimed at selecting an optimal set of stop points while allocating minimal-hop-length data transmission routes from sensor nodes to stop points. After showing that this optimization problem was *NP-hard*, a Tabu search algorithm (TABU-PSP) was proposed as a viable heuristic solution. Extensive simulation studies were conducted with the proposed TABU-PSP algorithm, an optimal algorithm, and various other possible heuristic solutions. The simulation results showed that TABU-PSP achieved near-optimal energy-usage results while completing execution in a reasonable amount of time (within a few seconds) even on large wireless sensor networks. Simulations under various conditions (uniform and clustered sensor node deployments, low and



(a) (b)

Figure 8. Comparison of the ILP-PSP and TABU-PSP algorithms with a variety of path lengths under a low-clustered node distribution: (a) 10 data packets generated per round per node, (b) 100 data packets generated per round per node.



(a) (b)

Figure 9. Comparison of the ILP-PSP and TABU-PSP algorithms with a variety of path lengths under a highly-clustered node distribution: (a) 10 data packets generated per round per node, (b) 100 data packets generated per round per node.

high data rates, short and long mobile sink paths) showed that TABU-PSP significantly outperformed all other heuristic solutions considered with respect to total energy consumption and expected network lifetime. Thus, it is claimed that the proposed TABU-PSP algorithm is a viable and scalable stop point selection algorithm for data gathering in wireless sensor networks with mobile sinks.

REFERENCES

[1] A. Kansal, A. A. Somasundara, D. D. Jea, M. B. Srivastaba and D. Estring, "Intelligent Fluid Infrastructure for embedded Networks," *Proceedings of MobiSys*, 2004, pp. 111-124.

[2] A. Kinalis, S. Nikoletseas, D. Patroumpa and J. Rolim, "Biased Sink Mobility with Adaptive Stop Times for Low Latency Data Collection in Sensor Networks," *Proceedings of Globecom*, Vol. 30, 2009, pp. 1-6.

[3] W. Aioffi, G. Mateus and F. Quintao, "Optimization Issues and Algorithms for Wireless Sensor Networks with Mobile Sinks," *Proceedings of the International Network Optimization Conference*, Paris, 2007.

[4] Y. Gu, D. Bozdag, E. Ekici, F. Ozguner and C. G. Lee, "Partitioning Based Mobile Element Scheduling in Wireless Sensor Networks," *Proceedings of the Sensor and Ad-Hoc Communication and Networks*, Santa Clara, 2005.

[5] R. Sugihara and R. K. Gupta, "Scheduling under Location and Time Constraints for Data Collection in Sensor Networks," *Proceedings of the* 28*th IEEE Real-Time Systems Symposium* (*RTSS*), *Work in Progress Session*, 2007.

[6] R. Sugihara and R. K. Gupta, "Improving the Data Delivery Latency in Sensor Networks with Controlled Mobility," *Proceedings of the* 4*th IEEE International Conference on Distributed Computing in Sensor Systems* (*DCOSS*), Vol. 5067, 2008, pp. 386-399.

[7] G. Xing, T. Wang, Z. Xie and W. Jia, "Rendezvous Planning in Mobility-Assisted Wireless Sensor Net-works," *Proceedings of the* 28*th IEEE International Real-Time Systems Symposium* (*RTSS*), 2007, pp. 311-320. doi:10.1109/RTSS.2007.44

[8] G. Xing, T. Wang, W. Jia and M. Li "Rendezvous Design Algorithm for Wireless Sensor Networks with a Mobile Base Station," *Proceedings of the* 9*th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2008, pp. 231-239. doi:10.1145/1374618.1374650

[9] J. Rao and S. Biswas, "Network-Assisted Sink Navigation for Distributed Data Gathering: Stabilityand Delay-Energy Trade-Offs," *Computer Communications*, Vol. 33, No. 2, 2010, pp. 160-175. doi:10.1016/j.comcom.2009.08.009

[10] J. Luo, J. Panchard, M. Piorkowski, M. Grossglauser and J. P. Hubaux, "MobiRoute: Routing Towards a Mobile Sink for Improving Lifetime in Sensor Networks," *Proceedings of the* 4*th IEEE International Conference on Distributed Computing in Sensor Systems* (*DCOSS*), 2006, pp. 480-497.

[11] S. Gao and H. K. Zhang, "Energy Efficient Path-Constrained Sink Navigation in Delay-Guaranteed Wireless Sensor Networks," *Networks*, Vol. 5, No. 6, 2010, pp. 658-665. doi:10.4304/jnw.5.6.658-665

[12] A. Giannakos, G. Karagiorgos and I. Stavrakakis, "A Message-Optimal Sink Mobility Model for Wireless Sensor Networks," *Proceedings of the Eighth International Conference on Networks*, 2009. doi:10.1109/ICN.2009.53

[13] T. Shi and Y. Thomas Hou, "Theoretical Results on Base Station Movement Problem for Sensor Network," 27*th Conference on Computer Communications IEEE*, 2008, pp. 13-18.

[14] R. C. Shah, S. Roy, S. Jain and W. Brunette, "Data MULEs: Modeling and Analysis of a Three-Tier Architecture for Sparse Sensor Networks," *Sensor Network Protocols and Applications*, Vol. 1, No. 2-3, 2003, pp. 215-223. doi:0.1109/SNPA.2003.1203354

[15] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala and V. Pandit, "Local Search Heuristic for K-Median and Facility Location Problems," *Proceedings of the ACM Symposium on Theory of Computing*, 2001, pp. 21-29.

[16] F. Glover and M. Laguna, "Tabu Search," Kluwer Academic, Boston, 1997.

[17] F. Glover and M. Laguna, "Tabu Search," In: C. R. Reeves, Ed., *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publications, Oxford, 1992, pp. 70-150.

[18] Xpressmp. http://www.dash.co.uk

[19] M. Evans, N. Hastings and B. Peacock, "Statistical Distributions," 3rd Edition, Wiley Press, New York, 2000, pp. 34-42.

[20] S. L. Wu and Y. C. Tseng, "Wireless Ad Hoc Networking," Auerbach Publications, Boca Raton, 2007. doi:10.1201/9781420013825