A Study on the Bottleneck Power-Aware Many-to-One Routing Problem for Wireless Sensor Networks

Pi-Rong Sheu Yu-Ting Li Hung-Yuan Tsai Chu-Yi Ho
Department of Electrical Engineering, National Yunlin University of Science & Technology
Touliu, Yunlin 640, Taiwan, R.O.C.

Abstract

Since the operations of sensors in a wireless sensor network mainly rely on battery power, power consumption becomes an important issue. To extend the network's lifetime, many power-aware routing protocols have been designed to evenly distribute packet-relaying loads among nodes to prevent the overuse and abuse of battery power of any node. In this paper, we will consider the problem of searching for multiple routing paths between multiple source sensors and the single sink in a wireless sensor network such that the maximum of all the node's power consumptions is as small as possible. We will first show such a problem to be NP-complete. Next, based on Dijkstra's algorithm, an efficient heuristic algorithm is developed for the difficult problem. Computer simulations verify that the suboptimal solutions generated by our heuristic algorithm are very close to the optimal ones obtained by linear programming techniques.

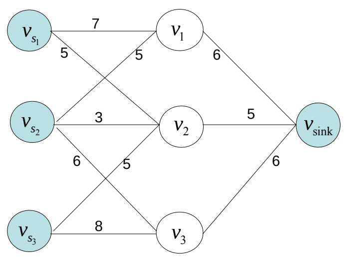
Keywords: NP-Complete, Power-Aware, Routing Algorithm, Sensor Network.

1. Introduction

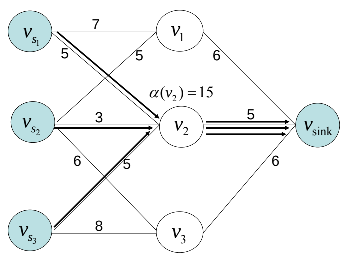
A wireless sensor network (WSNET) is formed by a large number of tiny sensing devices (or called *sensors*) [8] [14]. A sensor in a WSNET can generate as well as forward data, which are gathered from every sensor's vicinity and will be delivered to the single remote base station (or called *the sink*). Two sensors in such a network can communicate directly with each other through a single-hop routing path in the shared wireless media if their positions are close enough. Otherwise, they need a multi-hop routing path to finish their communications. In a multi-hop routing path, the data packets sent by a source sensor are relayed by several intermediate sensors before reaching the sink. WSNETs are useful in a broad range of environmental sensing applications such as vehicle tracking, seismic data, and so on.

The research of WSNETs has attracted a lot of attention recently [8] [14]. In particular, since WSNETs are characterized by their limited battery-supplied power, extensive research efforts have been devoted to the design of power-aware routing protocols [8] [14]. At present, several efficient power-aware routing protocols have been developed, such as LEACH [5], Directed Diffusion [2], Energy Aware Routing [10], and MLDA & MLDR [6]. Most of the existing power-aware routing protocols are designed primarily to maximize the lifetime of WSNETs. (One of the common definitions of the network lifetime is the time period from the beginning of the network operation to when one of the nodes exhausts its battery power).To achieve the goal, the key mechanism adopted by many current power-aware routing protocols is either to evenly distribute packet-relaying loads to each sensor to prevent the battery power of any sensor from being overused [9] [12] or to minimize the total transmission power consumption of the entire WSNET [8] [14]. Let us use Figure 1 to illustrate the difference between the two strategies. In Figure 1(a), each node is supposed to possess the same battery power of 100. The number next to each link represents the power to be consumed when one data packet is delivered through the link. There are three source sensors (v_{s_1}, v_{s_2}, and v_{s_3}) and one sink node v_{sink} in the network. Every source sensor wants to send one data packet to the sink. Now, if we use the shortest path strategy, as shown in Figure 1(b), the network's total transmission power consumption will be $(5+5)+(3+5)+(5+5)=28$ and the network's lifetime will be $\min\{\left\lfloor\frac{100}{5}\right\rfloor,\left\lfloor\frac{100}{3}\right\rfloor,\left\lfloor\frac{100}{5}\right\rfloor,\left\lfloor\frac{100}{15}\right\rfloor\}$ $=\min\{20,33,20,6\}=6$. If we use another strategy, as shown in Figure 1(c), the network's total transmission power consumption will be $(7+6)+(3+5)+(8+6)=35$ and the network's lifetime will be $\min\{\left\lfloor\frac{100}{7}\right\rfloor,\left\lfloor\frac{100}{3}\right\rfloor,\left\lfloor\frac{100}{8}\right\rfloor,\left\lfloor\frac{100}{6}\right\rfloor,\left\lfloor\frac{100}{5}\right\rfloor,\left\lfloor\frac{100}{6}\right\rfloor\}$ $=\min\{14,33,12,16,20,16\}=12$. In the two situations, it is easy to observe that a smaller total transmission power does not imply a longer network lifetime, exemplified by the first strategy. On the other hand, we discover that when the power consumptions of nodes even out, the associated network's lifetime will be longer, as the later strategy shows. In this paper, we will take the approach of leveling each node's power consumptions as a starting point.

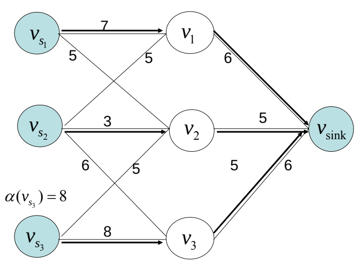
In WSNETs, the basic operation is the periodic gathering and transmission of sensed data packets to the sink for further processing [8] [14]. To be more specific, during a period of time (called *a round*), the sink first broadcasts a query for its interested data, then the sensors which posses the appropriate data (called *the source sensors*) deliver their data packets to the sink. Obviously, it is possible that a lot of source sensors want to communicate with the single sink simultaneously. In this paper, we will consider the problem that given multiple source sensors and the single sink in a WSNET, find a routing path between each source sensor and the single sink such that the maximum of all the node's power consumption is as small as possible, namely, the power consumption of each node is as even as possible. Undoubtedly, the resulted lifetime of network is extended. We will name the problem as *the bottleneck power-aware many-to-one routing (BPAMOR) problem.* In this paper, we will first show the *BPAMOR* problem to be NP-complete. Then, based on Dijkstra's algorithm, an efficient heuristic algorithm with low time complexity is developed for the difficult BPAMOR problem. Finally, by computer simulations, we verify that the suboptimal solutions generated by our heuristic algorithm are very close to the optimal ones obtained by linear programming techniques.



(a)



(b)



(c)

Figure 1: An example to illustrate the BPAMOR problem.

The rest of the paper is organized as follows. In Section 2, a formal definition of the BPAMOR problem is given. In Section 3, the BPAMOR problem is shown to be NP-complete. In Section 4, an efficient heuristic algorithm for the BPAMOR problem is proposed. In Section 5, the performance of our heuristic algorithm is evaluated through simulations and compared to the optimal solutions. Lastly, Section 6 concludes the whole research.

2. The Definition of our BPAMOR Problem

In this section, we will introduce our some assumptions, notations, and definitions. A formal definition of our problem in terms of these notations and definitions will also been stated. In the following, the term "node" is synonymous with the term "sensor" and the term "link" is synonymous with the term "communication channel".

Assumptions

The following states some important assumptions used in our research.

(1) We assume that the wireless sensor network's topology would not change, i.e. no sensor gets move [1] [7] [13].

(2) We only consider the transmission power and ignore the reception power [3].

(3) We assume that the required transmission power to establish a communication channel between any two sensors x and y is the same. In other words, $c(<v_x, v_y>) = c(<v_y, v_x>)$, where $c(<v_x, v_y>)$ and $c(<v_y, v_x>)$ denote the minimal transmission power required by sensors x and y to establish communication channels $<v_x, v_y>$ and $<v_y, v_x>$, respectively [3].

(4) We assume that when a data packet passes through a link, the transmission power consumption associated with the link can be an arbitrary value, i.e., can be independent of the Euclidean length of link [1] [3].

Traffic Model and Data Aggregation

In the following, we will define *a round* as a period of time in which the sink first broadcasts a queue for its interested data, then the source sensors, which posses the related data, deliver their data packets to the sink. We assume that during each round, each source sensor generates one data packets to be transmitted to the sink.

Data aggregation has been recognized as a useful routing paradigm in WSNETs [14]. The main idea is to combine data packets from different sensors to eliminate redundant messages and to reduce the number of transmissions such that the total transmission power of network is saved. One of the simplistic data aggregations is that an intermediate sensor always aggregates multiple incoming data packets into a single outgoing data packet. In the following, for simplicity, we will deal with our BPAMOR problem without data aggregation. In fact, as discussed in [6], data aggregation is not applicable in all sensing environments.

Problem Formulation

We represent a WSNET by a weighted graph $G = (V, E)$, where V denotes the set of sensors and the sink, and E denotes the set of communication links connecting the sensors or the sink. For E, we define a *transmission power consumption function* $\beta: \ell \rightarrow R^{+}$ that assigns a nonnegative weight to each link in the network. The value $\beta(i, j)$ associated with link $(v_i, v_j) \in E$ represents the transmission power that one data packet will consume on that link. For E, we define *a data packet flow function* $f: E \rightarrow I^{+}$. The value $f(v_i, v_j)$ denotes the number of data packets passing through link (v_i, v_j). For V, we define *a node power consumption function* $\alpha: V \rightarrow R^{+}$. Thus, $\alpha(v_i) = \sum_{(v_i, v_j) \in E} f(v_i, v_j) \times \beta(v_i, v_j)$ represents the total ransmission power that node v_i will consume during a round.

Under the BPAMOR problem we are considering, q routing paths originating from q *source sensors* $v_{s_i} \in V$ in the WSNET have to be connected to the sink $v_{sink} \in V$. Our target is to find the q routing paths such that the maximum of all the node's transmission power consumptions in G is minimized.

Based on these notations and definitions, we can now formally describe the BPAMOR problem in our paper: given a weighted graph $G=(V,E)$, *q source sensors* v_{s_i} and the sink v_{sink}, $v_{s_i}, v_{sink} \in V$, *a transmission power consumption function* $\beta: E \rightarrow R^{+}$, find a set of q routing paths such that the maximum of all the node's transmission power consumptions in G is minimized, i.e., $\max_{v_i \in G} \{\alpha(v_i)\}$ is minimized.

As an illustration of the above definitions and notations, let us consider the example shown in Figure 1. In Figure 1(a), let sensors v_{s_1}, v_{s_2}, and v_{s_3} be the source sensors and v_{sink} be the sink, respectively. The number next to a link represents the transmission power consumption of the link. If we use the shortest path strategy, we can see the result shown in Figure 1(b), and the node with maximal transmission power is v_2 and the value is 15. If we use some strategies instead of the shortest path strategy, as shown in Figure 1(c), we can see that the node with maximal transmission power is v_{s_3} and the value is 8. Therefore, we aim at designing efficient routing algorithms to minimize the maximum of all the node's transmission power consumptions. Basically, this implies a longer network lifetime can be obtained.

3. The Complexity of our BPAMOR Problem

In this section, we will show that our BPAMOR problem is NP-complete.

To prove our BPAMOR problem to be NP-complete, first let us restate it in its decision version as follows: given a weighted graph $G=(V,E)$, *q source sensors* v_{s_i} and the sink v_{sink}, $v_{s_i}, v_{sink} \in V$, *a transmission power consumption function* $\beta: E \rightarrow R^{+}$, a constant c_p, find a set of q routing paths such that the maximum of node's transmission power consumption in G is less than or equal to c_p, i.e., $\max_{v_i \in G} \{\alpha(v_i)\} \leq c_p$.

For simplicity, in what follows, we will not distinguish the decision version and the optimal version of the BPAMOR problem when no ambiguity arises.

Next, let us introduce the *3-Dimensional Matching* (3DM) problem [4].

Instance: A set $M \subseteq W \times X \times Y$, where W, X, and Y are disjoint sets having the same number q of elements.

Question: Does M contain a *matching*, that is, a subset $M' \subseteq M$ such that $|M'| = q$ and no two elements of M' agree in any coordinate?

This problem was shown to be NP-complete by Karp [4]. Now, we will use it to prove the following theorem.

Theorem 1. The BPAMOR problem is NP-complete.

Proof. First, the BPAMOR problem can be easily seen to be in the class NP. We next transform the 3DM problem to the BPAMOR problem in polynomial time. Let the sets W, X, Y, with $|W| = |X| = |Y| = q$, and $M \subseteq W \times X \times Y$ be an arbitrary instance of 3DM. Let the elements of these sets be denoted by $W = \{w_1, w_2, \cdots, w_q\}$, $X = \{x_1, x_2, \cdots, x_q\}$, $Y = \{y_1, y_2, \cdots, y_q\}$ and $M = \{m_1, m_2, \cdots, m_k\}$, where $k = |M|$.

We construct an instance of the BPAMOR problem as follows: For each element w_i of W, the corresponding weighted graph $G = (V, E)$ has a source sensor v_{w_i} $(1 \leq i \leq q)$. Similarly, for each x_i and y_i, G has sensors v_{x_i} and v_{y_i}, respectively. In addition, the sink node v_{sink} is put into G. Thus, $V = \{ v_{w_1}, v_{w_2}, \ldots, v_{w_q} \} \cup \{ v_{x_1}, v_{x_2}, \ldots, v_{x_q} \} \cup \{ v_{y_1}, v_{y_2}, \ldots, v_{y_q} \} \cup \{ v_{\text{sink}} \}$. If $(w_i, x_j, y_k) \in M$, then there exist one edge $< v_{w_i}, v_{x_j} >$ between nodes v_{w_i} and v_{x_j}, and one edge $< v_{x_j}, v_{y_k} >$ between nodes v_{x_j} and v_{y_k}. Furthermore, we connect each node v_{y_k} to the sink v_{sink}. Thus, the edge set $E = \{< v_{w_i}, v_{x_j} >: \textit{if } (w_i, x_j, y_k) \in M\} \cup \{< v_{x_j}, v_{y_k} >: \textit{if } (w_i, x_j, y_k) \in M\} \cup \{< v_{y_k}, v_{\text{sink}} >: k = 1, 2, \ldots, q\}$. Each sensor node v_i is assumed to have an amount $\alpha(v_i) = 1$ of residual power capacity. All the edges have a transmission power consumption of 1 when one data packet traverses it. Finally, let $c_p = 1$. The constructed G is illustrated in Figure 2. It is easy to see that this transformation can be finished in polynomial time.

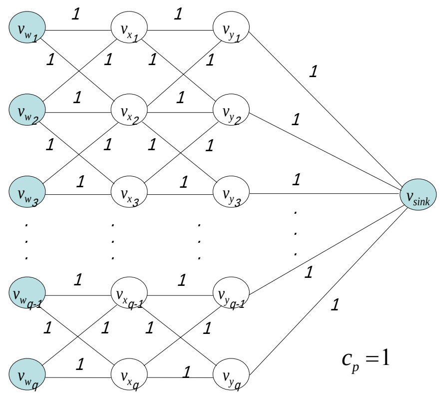


Figure 2: An illustration of Theorem 1.

We next show that a feasible solution for the BPAMOR problem exists in G if and only if the set M contain a matching M'. First, suppose M contain a matching, that is, a subset $M' \subseteq M$ such that $|M'| = q$ and no two elements of M' agree in any coordinate. If $(w'_i, x'_j, y'_k) \in M'$, then we let $(v_{w'_i}, v_{x'_j}, v_{y'_k}, v_{\text{sink}})$ be a routing path in G for source sensor $v_{w'_i}$. Since $|M'| = q$, there are q routing paths each of which is for a different source sensor v_{w_i}. Since no two elements of M' agree in any coordinate, these routing paths are pairwise node-disjoint except their common sink node. Because they are pairwise node-disjoint, any link (v_i, v_j) belongs to at most one of these q paths. Therefore, for each link $(v_i, v_j) \in E$, $f(v_i, v_j) \le 1$. Similarly, any node v_i belongs to at most one of these q paths. So, for any $v_i \in V$, $\sum_{(v_i,v_j)\in E} f(v_i, v_j) \le 1$. Then, for any $v_i \in V$, we have $\alpha(v_i) = \sum_{(v_i,v_j)\in E} \beta(i,j) \times f(v_i,v_j) = \sum_{(v_i,v_j)\in E} 1 \times f(v_i,v_j) \le 1$. As a result $\max_{v_i \in V}\{\alpha(v_i)\} \le 1 = c_p$. Thus, these q routing paths form a set of feasible routing paths for the corresponding general BPAMOR problem in G.

Next, suppose we have a solution for the general BPAMOR problem in the weighted graph G. Let $\overline{P}_1, \overline{P}_2, \cdots, \overline{P}_q$ be one of the possible solutions in G. Because $\max_{v_i \in V}\{\alpha(v_i)\} \le 1 = c_p$, $\alpha(v_i) \le 1$ for each node v_i. Furthermore, each link has a transmission power consumption of 1, and each node v_i belongs to at most one routing path $\overline{P}_l$ (otherwise, $\alpha(v_i) > 1$). Thus routing paths $\overline{P}_1, \overline{P}_2, \cdots, \overline{P}_q$ are pairwise node-disjoint except their common sink node. Let $\overline{P}_t \in (\overline{P}_1, \overline{P}_2, \cdots, \overline{P}_q)$ and $\overline{P}_t = (v_{\overline{w}_{t_i}}, v_{\overline{x}_{t_j}}, v_{\overline{y}_{t_k}}, v_{\text{sink}})$, then $(\overline{w}_{t_i}, \overline{x}_{t_j}, \overline{y}_{t_k}) \in M'$. Clearly $|M'| = q$ and no two elements of M' agree in any coordinate. Thus, M' is a matching and $M' \subseteq M$. This completes our proof of NP-completeness. ▓

When a problem is proved to be NP-complete, the follow-up quest will be to search for various heuristic algorithms and evaluate them by computer simulations. In the next section, we will design an efficient heuristic algorithm for our BPAMOR problem.

4. An Efficient Heuristic Algorithm for the BPAMOR Problem

In this section, we will propose a heuristic algorithm for the BPAMOR problem. This algorithm is based on the shortest-path concept, and we call it the BPAMOR algorithm. The basic idea of our BPAMOR algorithm is as follows: we improve the shortest-path concept by attempting to adjust each routing paths to reduce the maximum of node's transmission power consumption. To be more specific, we firstly find a set of shortest paths from each source sensor to the sink by using Dijkstra's algorithm. Then we modify each routing path by removing the sensor node with maximal transmission power consumption in the network if the required condition is matched. Figure 3 shows the pseudocode of our BPAMOR algorithm.

In line 2, A is the set of source sensors and the sink. $\max_\alpha$ denotes the maximum of node's transmission power consumption in the network. Initially, we set the value of $\max_\alpha$, $\alpha(v_i)$, and $\alpha'(v_i)$ to be zero, where $\alpha(v_i)$ is the transmission power consumption of each sensor v_i and $\alpha'(v_i)$ is its temporary value. Line 4 to line 6 are to find the shortest path r_i from each source sensor v_{s_i} to the sink by using Dijkstra's algorithm. Line 7 to line 9 compute the transmission power consumption $\alpha(v_i)$ for each node v_i. Line 10 and line 11 select the node v_z with maximal transmission power consumption in the network and set $\max_\alpha$ to be the maximum value. Line 12 to line 30 are the process of adjusting each routing path. During the process of adjustment, we first select the sensor node v_z (here we set $v_y = v_z$), which has the maximal transmission power consumption, and remove it form the network temporarily. Next we check whether this sensor node v_z is on the routing path of $(v_{s_i}, v_{\text{sink}})$ or not. If the removed sensor node v_z is on the routing path of $(v_{s_i}, v_{\text{sink}})$, we will re-find another routing path r_i' from v_{s_i} to v_{sink} in G'.

The checking operations are shown from line 15 to line 17. After finishing the checking, from line 18 to line 22, we will re-computer the transmission power consumption $\alpha'(v_i)$ for each sensor node v_i. Line 25 compares the present maximum of the transmission power consumption $\alpha(v_{z'})$ with the

Input: G = (V , E), a residual power capacity function $\alpha : V \rightarrow R^{+}$, a transmission power consumption function β : $E \rightarrow R^{+}$, q source sensors v_{s_i} and the sink v_{sink} , $v_{s_i}, v_{\text{sink}} \in V$
Output: $\max_{\alpha}$ is the maximum of node's transmission power consumption in the netwrok $r_i, i \in 1,2,...,q$ is the routing path for each source sensor v_{s_i} respectively.

1. begin
2. $A = \{(v_{s_1}, v_{\text{sink}})(v_{s_2}, v_{\text{sink}}),......,(v_{s_q}, v_{\text{sink}})\}$; $\max_{\alpha} = 0$;
3. for each $v_i \in V$ do { $\alpha(v_i) = \alpha`(v_i) = 0$;} end of for loop
4. for each $(v_{s_i}, v_{\text{sink}}) \in A$ do
5. find the routing path r_i with the smallest total transmission power between v_{s_i} and v_{sink} by Dijkstra`s algorithm;
6. end of for loop
7. for each r_i do
8. for each $v_j \in V$ do {if $(v_j, v_k) \in r_i$ then $\alpha(v_j) = \alpha(v_j) + \beta(v_j, v_k)$;} end of for loop
9. end of for loop
10. find the node v_z whose $\alpha(v_z) = \max_{v_i \in V}\{\alpha(v_i)\}$;
11. $\max_{\alpha} = \alpha(v_z)$; $G' = G$;
12. do
13. $G' = G' - v_z$; $y = z$;
14. for each $(v_{s_i}, v_{\text{sink}}) \in A$ do
15. if (r_i includes v_z)
16. {
17. find the routing path r_i' in $G`$ with the smallest total transmission power between $v_{s_i}$ and $v_{\text{sink}}$ by Dijkstra`s algorithm;
18. for each $v_j \in V$ do
19. $\alpha'(v_j) = \alpha(v_j)$;
20. if $(v_j, v_{k'}) \in r_i'$ then $\alpha'(v_j) = \alpha'(v_j) + \beta(v_j, v_{k'})$;
21. if $(v_j, v_k) \in r_i$ then $\alpha'(v_j) = \alpha'(v_j) - \beta(v_i, v_k)$;
22. end of for loop
23. find the node $v_{z'}$ whose $\alpha(v_{z'}) = \max_{v_i \in V}\{\alpha'(v_i)\}$;
24. $\max'_{\alpha} = \alpha(v_{z'})$;
25. if $(\max'_{\alpha} < \max_{\alpha})$ { $r_i = r_i'$; $\alpha(v_i) = \alpha'(v_i)$ }end if
26. }
27. end if
28. end of for loop
29. find the node v_z whose $\alpha(v_z) = \max_{v_i \in V}\{\alpha(v_i)\}$; $\max_{\alpha} = \alpha(v_z)$;
30. while($y \neq z$)
31. output $\alpha, r_i, i = 1,2,...,q$;
32. end of the algorithm

Figure 3: The pseudocode of our BPAMOR algorithm.

previous one $\alpha(v_z)$. If the present one is smaller than the previous one, we will replace the old routing path r_i by the new routing path r_i' and replace the $\alpha(v_i)$ by $\alpha'(v_i)$; Otherwise, nothing to do. After finishing the checking process for each routing path r_i of $(v_{s_i}, v_{\text{sink}})$, we search a new sensor node v_z with maximal transmission power consumption in the present status. All the above steps are repeated until the sensor node v_y is equal to the new sensor node v_z. Because as the new sensor node v_z is equal to the original sensor node v_y, nothing will happen in the following adaptation loops. Finally, we output the transmission power consumption $\alpha(v_i)$ of each sensor node v_i and each routing path r_i, where $i = 1,2,...,q$.

Example

We will explain the operation of our BPAMOR algorithm by using Figure 4. In Figure 4(a), let nodes v_{s_1}, v_{s_2}, and v_{s_3} be the source sensor nodes and nodes v_{sink} is the sink node. The number next to each link represents the power to be consumed when one data packet bypasses this link. Firstly, we execute the Dijkstra's algorithm from each source sensor to the sink to find its shortest path, as shown in Figure 4 (b). From Figure 4 (b), we can notice that the node with maximal transmission power consumption is v_2 and we set $v_z = v_2$ (here we also set $v_y = v_z$). Next, we will remove the v_z from the network temporarily and do the process of adjustment. Because the sensor node v_z is on the routing path r_1, we have to modify the routing path r_1. The new routing path r_1' decreases the maximum of node's transmission power consumption in the network, therefore we replace the routing path r_1 by the new routing path r_1' which is shown in Figure 4(c). The processes of modification for routing path r_2 and routing path r_3 will follow the same principle, which are shown in Figure 4(d) and Figure 4(e). From Figure 4(d), we notice that the maximum of node's transmission power consumption increases due to the replacement of new routing path r_2', we will use the old routing path r_2 instead of new routing path r_2'. The modification of r_3 is similar with r_1. After the process of modification, we can get a new sensor node v_z with the maximal transmission power consumption in the network. In

Figure 4(f), because the new sensor node v_z $(=v_{s_3})$ is not equal to v_y $(=v_{s_2})$, we will continue our line 12 to line 30 loop and reset the node v_y. At this time, the node v_2 has been removed temporarily last round, we can't do any modification in this round, and therefore end our algorithm. From Figure 4(f), the node with maximal transmission power consumption is v_{s_3} and the value $\max_\alpha$ is 8. Recall that, if we use the Dijkstra's algorithm directly, the result is shown in Figure 4(b), and the node with maximal transmission power consumption is v_2 and the value $\max_\alpha$ is 15.

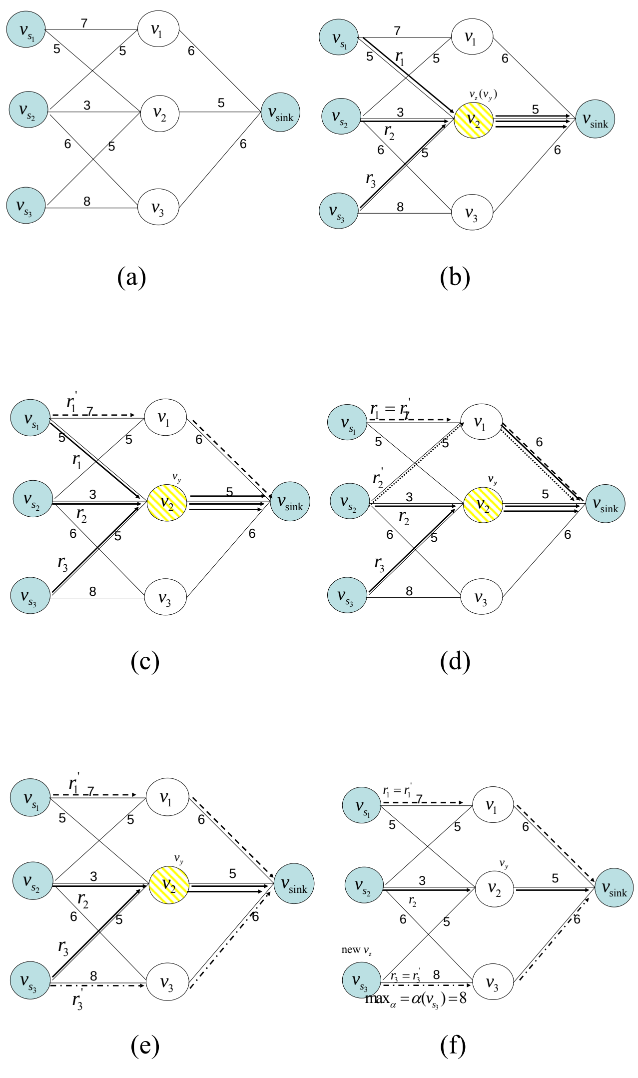


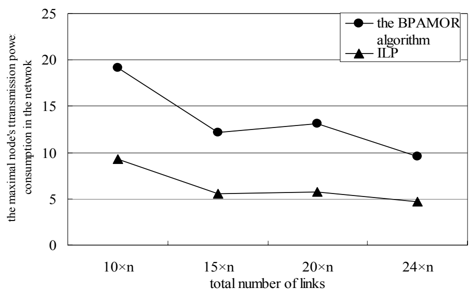
Figure 4: An illustration of our BPAMOR algorithm.

5. Computer Simulations

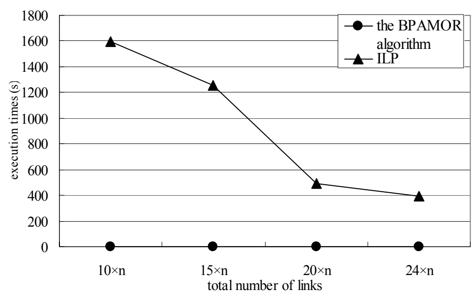
In this section, we will conduct computer simulations to evaluate the performance of our ILP formula [11] and our BPAMOR algorithm in terms of maximal node's transmission power in the network and the execution time.

Our computer simulations will be divided into three different simulation environments. In each simulation environment, we will observe the impact of one of the three parameters: the number of total links, the size of WSNET, and the structure of WSNET, on performance of our BPAMOR algorithm.

In our first simulation environment, the total number of links will be varied and its influences will be observed. The environments in our first simulation are set as follows: the WSNET consists of 50 sensor nodes located in a 100×100 m^2 area randomly. The transmission power consumption of each links is set from 1 to 20. The number of source sensors is set to 20. We will vary the total number of link from 10×*n*, 15×*n*, 20×*n*, to 24×*n* (where *n* is the total number of sensor nodes and 24×*n* means that the network is a complete graph). Notice that we neglect the 5×*n* because the execution time of ILP is too long (at least 3600 seconds each experiment). From Figure 5(a), we see that the maximal node's transmission power consumption is decreasing with the increasing of the total number of links. This is because the more links exist in the network, the more routing path can be used for different source sensors. Figure 5(b) shows the execution times. The result of Figure 5 reveals that although ILP can get the optimal solution, the long execution time is its disadvantage.



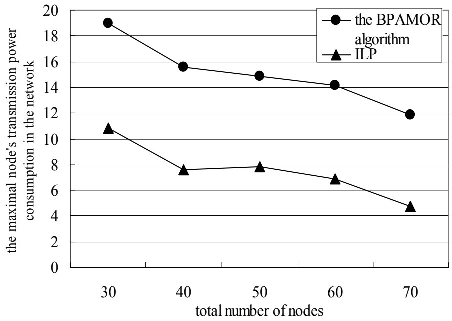
(a) The effect of the total number of links on the maximal node's transmission power consumption in the network



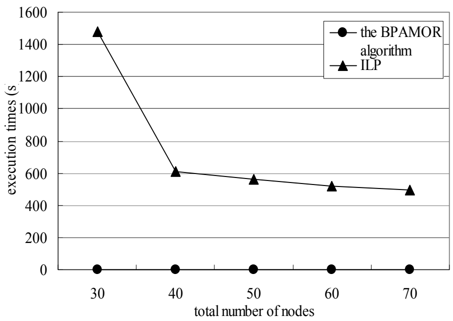
(b) The effect of the total number of links on execution times

Figure 5：The first computer simulations.

In our second simulation environment, the total number of nodes will be varied and its influences will be observed. The environments in our second simulation are set as follows: the WSNET consists of n sensor nodes located in a 100×100 m^2 area randomly. The transmission power consumption of each link is set from 1 to 20. The number of source sensors is set to 20. The total number of links is set $N \times (N-1)/4$, where n is the total number of sensor nodes. We will vary the total number of sensor nodes from 30, 40, 50, 60, to 70. From Figure 6(a), we see that the maximal node's transmission power consumption is decreasing with the increasing of the total number of sensor nodes. The impact of total number of sensor nodes is similar with that of total number of links. Figure 6(b) shows the execution times. We find that ILP needs more execution time to obtain the better performance.



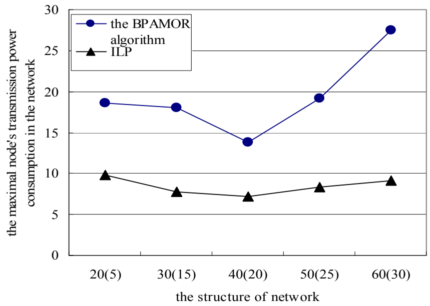
(a) The effect of the size of network on the maximal node's transmission power consumption in the network



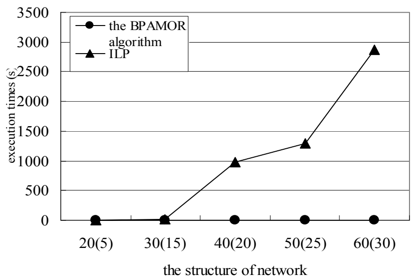
(b) The effect of the size of network on execution times

Figure 6：The second computer simulations.

In our third simulation environment, the structure of WSNET will be varied and its influences will be observed. We will change the structure from 20(10), 30(15), 40(20), 50(25), to 60(30) (where 20(10) means there are 20 sensor nodes and 10 source sensors in the WSNET). The transmission power consumption of each link is set from 1 to 20. The total number of links is set $N \times (N-1)/4$, where n is the total number of sensor nodes. We observe that the performance of our BPAMOR algorithm keep static with the different structures of WSNET in Figure 7(a). That is, our BPAMOR can get a stable effect in any structure of WSNET. Figure 7(b) is the execution times.



(a) The effect of the structure of network on the maximal node's transmission power consumption in the network



(b) The effect of the structure of network on execution times

Figure 7：The third computer simulations.

6. Conclusions

In this paper, we have discussed the BPAMOR problem. We have shown the BPAMOR problem to be NP-complete. Based on Dijkstra's algorithm, an efficient heuristic algorithm with low time complexity has been proposed to solve the difficult BPAMOR problem. Finally, by computer simulations, we have verified that the suboptimal solutions generated by our heuristic algorithm are very close to the optimal ones found by an optimal ILP program.

Acknowledgements

This work was supported by the National Science Council of the Republic of China under Grant # NSC 93-2213-E-224-023

References

[1] M. Cagalj, J. P. Hubaux, and C. Enz, "Minimum-Energy Broadcast in All-Wireless Networks: NP-Completeness and Distribution Issues," *Proceedings of the Eighth ACM*

International Conference on Mobile Computing and Networking (Mobicom 2002), Sep. 2002.
[2] I. Chalermek, G. Ramesh, E. Deborah, and H. John, “Directed Diffusion for Wireless Sensor Networking,” *IEEE/ACM Transactions on Networking,* vol. 11, no. 1, Feb. 2003.
[3] W. T. Chen and N. F. Huang, "The Strongly Connecting Problem on Multihop Packet Radio Networks," *IEEE Transactions on Communications*, vol. 37, no. 3, pp. 293-295, Mar. 1989.
[4] M. R. Garey and D. S. Johnson, *Computers and Intractability: a Guide to the Theory of NP-Completeness*, San Francisco: W. H. Freeman & Company, Publishers, 1979.
[5] W. R. Heizelman, A. Chandrakasan, and H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Micro Sensor Networks,” *IEEE Proceedings of the Hawaii International Conference on System Sciences*, pp. 1-10, Jan. 2000.
[6] K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks,” *Proceedings of the 2002 IEEE International Conference on Networking* ,Atlanta, Georgia, Aug. 2002.
[7] A. Misra and S. Banerjee, "MRPC: Maximizing Network Lifetime for Reliable Routing in Wireless Environments," *IEEE Wireless Communications and Networking Conference,* vol. 2, pp. 800-806, Mar. 2002.
[8] C. S. R. Murthy and B. S. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall, 2004.
[9] C. E. Perkins, *Ad Hoc Networking*, MA: Addison-Wesley, 2001.
[10] R. C. Shan and H. M. Rabaey, “Energy Aware Routing for Low Energy Ad Hoc Sensor Networks,” *IEEE Wireless Communications and Networking Conference*, 2002.
[11] P. R. Sheu, Y. T. Li, H. Y. Tsai, and C. Y. Ho, "A Study on the Bottleneck Power-Aware Many-to-One Routing Problem for Wireless Sensor Networks," Technical Report, Department of Electrical Engineering, National Yunlin University of Science & Technology, Taiwan, R.O.C., 2005.
[12] C. K. Toh, “Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks,” *IEEE Communications Magazine*, vol. 39, pp. 138-147, Jun. 2001.
[13] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Network,” *Proceedings of the 19th IEEE INFOCOM*, pp. 585-594, 2000.
[14] F. Zhao and L. Guibas, *Wireless Sensor Networks: an Information Processing Approach*, Boston: Elsevier-Morgan Kaufmann, 2004.