Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 483581, 12 pages
http://dx.doi.org/10.1155/2013/483581

*Research Article*

# A Novel *Physarum*-Inspired Routing Protocol for Wireless Sensor Networks

**Mingchuan Zhang,[1,2] Changqiao Xu,[1,3,4] Jianfeng Guan,[1,4] Ruijuan Zheng,[2] Qingtao Wu,[2] and Hongke Zhang[1,4]**

[1] *State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2] *Information Engineering College, Henan University of Science and Technology, Luoyang 471023, China*
[3] *Institute of Sensing Technology and Business, Beijing University of Posts and Telecommunications, Wuxi 214135, China*
[4] *National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing 100876, China*

Correspondence should be addressed to Mingchuan Zhang; zmc@bupt.edu.cn

Received 31 January 2013; Accepted 7 June 2013

Academic Editor: Shuai Li



There is a tradeoff between routing efficiency and energy equilibrium for sensor nodes in wireless sensor networks (WSNs). Inspired by the large and single-celled amoeboid organism, *slime mold Physarum polycephalum*, this paper presents a novel *Physarum*-inspired routing protocol (P-iRP) for WSNs to address the above issue. In P-iRP, a sensor node can choose the proper next hop by using a proposed *Physarum*-inspired selecting next hop model (P-iSNH), which comprehensively considers the distance, energy residue, and location of the next hop. As a result, the P-iRP can get a rather low algorithm complexity of $O(\sqrt{n})$, which greatly reduces the processing delay and saves the energy of sensors. Moreover, by theoretical analysis, the P-iSNH always has an equilibrium solution for multiple next hop candidates, which is vital factor to the stability of routing protocol. Finally, simulation results show that P-iRP can perform better in many scenarios and achieve the effective tradeoff between routing efficiency and energy equilibrium compared to other famous algorithms.

## 1. Introduction

With the development of communication, electron, and sensor technologies, wireless sensor networks (WSNs) have attracted wide concern of both researchers and application providers. WSNs consist of large numbers of sensor nodes deployed over a certain region. Each sensor node is a low-cost, short range wireless transceiver typically equipped with a low-computation processor and a battery operated power supply. Under many scenarios, the sensor nodes need to operate without battery replacement for several years. Thus, there are two questions need to be considered. One is how to achieve energy balance of these nodes to avoid the emergence of energy holes which commonly take place around the sink, since the data traffic follows a many-to-one communication pattern and nodes nearer the sink have to take heavier traffic load. The other is how to obtain high routing efficiency under multihop transmission circumstance, since WSNs can contain hundreds of such low-cost sensor nodes. Therefore, designing such networks should primarily focus on both routing efficiency and energy equilibrium in terms of trade-off.

Location-aware routing protocols seem to possess high routing efficiency, where GPS, phone, or other techniques are used for positioning nodes [1]. However, there are two extremes in location-aware routing—the greedy strategy and the robust strategy. Greedy strategies may suffer failures to route packets to destination, while robust strategies need very high flooding rates to ensure reliability and rapid delivery of data. Thus, many location-aware routing protocols are mostly to propose methods to overcome the mentioned drawbacks [2–4]. The energy-aware routing attracts more researchers'

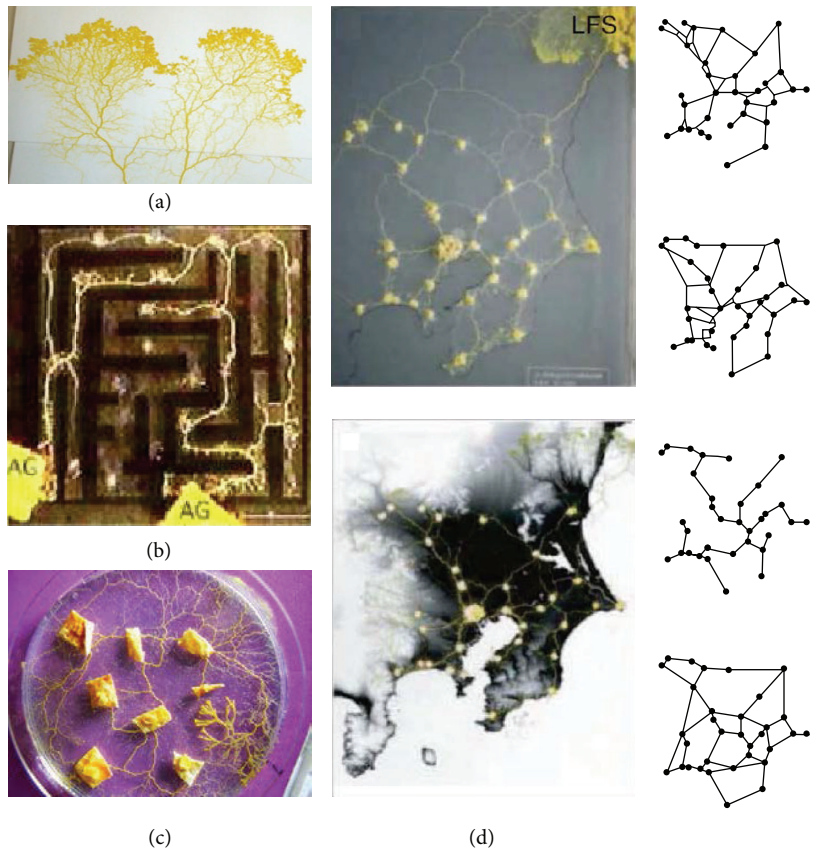



Figure 1: Photographs of *Physarum*. (a) *Physarum* is able to make complex comparisons between two food options based on the quality difference of the food and riskiness of the feeding environment, which comes from http://sydney.edu.au/news/sobs/1699.html?newsstoryid=4576. (b) Example of maze solving by *Physarum*, which comes from paper [25]. (c) Example of connecting path in a uniformly illuminated field which comes from http://cmr.soc.plymouth.ac.uk/research.htm. (d) Comparison of the *Physarum* networks with Tokyo rail network, which comes from paper [26].

attentions than that of location-aware routing for the significance of energy. There are many results relating to energy-aware routing recently [5–8] to save energy or prolong WSNs' lifetime, where energy harvesting [9] is shown to be a promising technique.

Unsurprisingly, the combination between location-aware routing and energy-aware routing becomes another researchers' focus to balance energy and efficiency for WSNs' routing protocol [10, 11]. In addition, some researchers focus on other aspects of WSNs' routing protocol, for example, the distributed characteristic [12, 13] and the trade-off of other two or more indexes [14, 15].

In recent years, the *slime mold Physarum polycephalum*, a large single-celled amoeboid organism, becomes a new researchers' pet and has shown to be a good technique for solving the shortest-path problem, since it can adapt its organism to forage for patchily distributed food sources, as shown in Figure 1. In this paper, we draw the inspiration from the *Physarum*, introduce the *Physarum* model into WSNs, and improve it through ignoring its dimension and preserving its logical meanings to make it suit for routing selection based on our prior works [16–20]. Our focus is to choose the proper next hops to transmit data to sink in thinking of both routing efficiency and energy equilibrium, which is partially similar to [11].

The rest of this paper is organized as follows. Section 2 gives a brief description of related work. Section 3 formulates the proposed models. Section 4 details the P-iRP. Section 5 discusses the feasibility of P-iSNH. Section 6 evaluates our P-iRP by simulations. Finally, the conclusion is presented in Section 7.

## 2. Related Work

In the aspect of efficient routing, GPSR [2] is a famous greedy routing protocol, which makes greedy forwarding decisions using only information about a router's immediate neighbors in the network topology. Li et al. [21] present a neural network approach to plan the shortest path from the target position to the start position in real time. Kuhn et al. [3] utilize face (or perimeter) routing to go around voids in the topology. Padmanabh et al. [4] consider unbiased random walk on a regular deployment of nodes, forming a hexagonal lattice pattern.

In the aspect of the energy-aware routing, Trajcevski et al. [5] construct a data aggregation tree that minimizes the total

energy cost of data transmission, which is shown as an NP-complete problem, and propose algorithms for addressing it. A battery aware power allocation model was studied in [6] for a single-hop transmission scheme to balance the network energy consumption based on the nonlinear battery parameters proposed in [7]. Chau et al. [8] consider that a portion of the lost charge can be recovered due to the battery's recovery effect and present a battery model. The approaches propose some of the routes that would otherwise need to bypass the hole along the boundary and should start to deviate from their original path further from the hole instead.

Moreover, distribution and clustering problems are also important branches of WSNs' routing. Li et al. [13] consider the problem of nonlinear constraints defined on a graph and give a better solution incorporated with Laplacian eigenmap as heuristic information to solve the problem in distributed scenarios. For maximizing the network lifetime, Rao and Fapojuwo [22] present a battery aware distributed clustering and routing protocol which incorporates the state of the battery's remaining charge and health parameters in computing the charge utility metric at each cluster formation round. Wang and Syue [23] propose a relay selection protocol based on geographical information, in which multihop transmission is realized by concatenation of single cluster-to-cluster hops, where each cluster-to-cluster scheme forms the simplified cooperative network that consists of a single source destination pair and a set of available relays.

However, the trade-off is not comprehensively considered in those papers, which is very necessary for WSNs' routing due to the features of WSNs, for example, nodes' failures, limited bandwidth, and power energy. Bai et al. [10] route the connections in a manner that link failure does not shut down the entire stream but allows a continuing flow for a significant portion of the traffic along multiple paths to address the issues of reliability and energy efficiency. Trajcevski et al. [24] present heuristic approaches to relieve some of the routing load of the boundary nodes of energy holes in location-aware WSNs to balance load and latency. Yu et al. [11] use energy aware neighbor selection to route a packet towards the target region and recursive geographic forwarding or restricted flooding algorithm to disseminate the packet inside the destination region. By allowing the battery to rest for certain duration, without being subjected to heavy loads, Yang and Heinzelman [14] propose sleeping multipath routing, which selects the minimum number of disjoint paths to achieve the trade-off of given reliability requirement and energy efficiency. Sivrikaya et al. [15] propose randomized routing based on Markov chains to balance the load and routing performance.

In recent years, the *Physarum* becomes a new focus of bioinspired method. It is also the original source of our inspiration. Nakagaki et al. [25] validate that the *Physarum* is apparently able to solve shortest path problems as shown in Figure 1. They build a maze, cover it with pieces of *Physarum* (the *Physarum* can be cut into pieces that will reunite if brought into vicinity), and then feed the *Physarum* with oatmeal at two locations. A few hours later, the *Physarum* retracts to a path that follows the shortest path connecting the food sources in the maze. Tero et al. [26] use *Physarum* forms networks comparable efficiency, fault tolerance, and cost to those of real-world infrastructure networks—Tokyo rail system. Tero et al. [27] propose a mathematical model for the behavior of *Physarum* and argue extensively that the model is adequate.

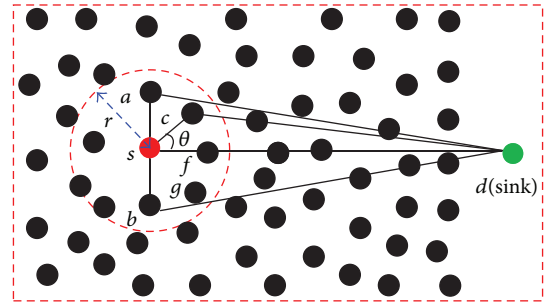



Figure 2: An example of sensor nodes' deployment in WSNs.

## 3. System Models

Our research is built on three assumptions. The first is that all nodes are aware of their locations, which may be achieved through GPS receivers at network deployment time, employing a distributed location discover algorithm shortly after deployment or adopting other positioning methods [1, 28]. The second is that each node is aware of its energy residue [22]. The third is that the link is bidirectional, that is, if a node hears from a neighbor, then its transmission range can reach to the neighbor.

*3.1. Typical WSNs Scenario.* We consider the large multi-hop WSNs which consist of $n$ static sensors. Each node $i$ has a fixed circular transmission range $r$ which determines the set of sensors in which each node can communicate with node $i$ in one hop. We abstract such WSNs using a graph $G = (V, E)$, where each node $v \in V$ represents a sensor, and each edge $e \in E$ represents the existence of one-hop wireless link between two sensors.

We suppose that node $s$ is the source node and node $d$ is the sink, as shown in Figure 2. In most cases, the sink $d$ is placed in the middle of WSNs field to ease traffic burdens of nodes in the right of the WSNs field. In this paper, we only think of the nodes in left of the WSNs field for simplicity and clarity. The transmission range of $s$ is drawn as a dashed circle whose radius is $r$ and center is $s$. We call the angle $\theta$ is the angle of deviation of node $c$, which represents a measurement of node $c$ deviating from the sink $d$. The Euclidean distance of any two nodes, $i$ and $j$, and the angle $\theta_{jid}$ can be calculated following from (1) and (2), respectively

$$L_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \tag{1}$$

$$\theta_{jid} = \arccos \frac{L_{sc}^2 + L_{sd}^2 - L_{cd}^2}{2 \times L_{sc} \times L_{sd}}, \tag{2}$$

where $(x_i, y_i)$ and $(x_j, y_j)$ are the coordinates of nodes $i$ and $j$, respectively.


(a) Traditional data packet format


(b) Our data packet format

SA: Source Address
NA: Next Hop Address
ER: Energy Residue

Figure 3: Formats of data packets.

If the node $s$ needs to transmit data to sink $d$, it will select its next hop in the dashed circle. Since our scenario is location aware, we select the next hop in the right semicircle under normal circumstances. Obviously, the smaller the angle $\theta$ is, the closer the next hop is to the sink for a fixed distance. That is to say, we are apt to choose the node whose $\theta$ is smaller as the next hop. In order to avail discussion, we define the $N_s$ is the set of neighbors of node $s$ (in the dashed circle), $N_s^L$ is the set of left neighbors of node $s$ (in the left semicircle), and $N_s^R$ is the set of right neighbors of node $s$ (in the right semicircle). Then, $N_s$, $N_s^L$, and $N_s^R$ meet the following:

$$\begin{aligned} N_s &= N_s^L \cup N_s^R \\ N_s^L &\cap N_s^R = \emptyset. \end{aligned} \tag{3}$$

In addition, the energy residue of each node is also important for choosing next hop for balancing the energy of WSNs' nodes. When a node chooses its next hop, it would consider the energy residue of the candidates and be apt to pick the node which has much higher energy residue as the next hop. Therefore, it is important to acquire the energy residue of neighbors.

We think of the basic theory of wireless transmission combined with Figure 2 and the data packet format shown in Figure 3(a). If node $s$ transmits a group of data to node $c$, all of the nodes in $N_s$ would receive the wireless radio and check the packet header. The node $c$ matches the field NA and receives the packet. Other nodes mismatch the field NA then ignore the packet and go on sleeping.

In order to acquire the energy residue of neighbors, we add a new field ER to the packet header, which is shown in Figure 3(b). When node $s$ transmits a group of data to node $c$, all of the nodes in $N_s$ extract the fields of SA and ER from the packet header and save ER in local memory according to SA. Then, the node $c$ matches the field NA and receives the packet. Other nodes mismatch the NA then ignore the packet and go on sleeping.

Since each node needs to listen in real time to every packet and try to match its field NA, only adding an operation of saving ER would not add a considerable effect on energy consumption. Therefore, we neglect the cost of acquiring energy residue of neighbors.

*3.2. Physarum-Inspired Path-Finding Model.* Papers in [25–27] exploit the *slime mold Physarum polycephalum* to develop a *Physarum*-inspired path-finding model (PiPf). Suppose that (1) the initial shape of a *Physarum* organism is represented by a graph, (2) the edges represent plasmodial tubes in which protoplasm flows, and nodes are junctions between tubes, (3) the pressures at nodes $i$ and $j$ are $P_i$ and $P_j$, respectively, and the two nodes are connected by a cylinder of length $L_{ij}$ and radius $r_{ij}$, and (4) the flow is laminar and follows the Hagen-Poiseuille equation. Then, the flux through the tube is calculated as in the following

$$Q_{ij} = \frac{\pi r_{ij}^4 \left(P_i - P_j\right)}{8\eta L_{ij}} = \frac{D_{ij}\left(P_i - P_j\right)}{L_{ij}} = \frac{D_{ij}\Delta P_{ij}}{L_{ij}}, \tag{4}$$

where $\Delta P_{ij} = P_i - P_j$ is the difference of pressures, $\eta$ is the viscosity of the fluid, and $D_{ij} = \pi r_{ij}^4/8\eta$ is a measure of the conductivity of the tube. As the length $L_{ij}$ is a constant, the behavior of *Physarum* is described by the conductivities, $D_{ij}$, of the edges.

Equation (4) represents that the flux through the tube $ij$ is determined by $D_{ij}$, $\Delta P_{ij}$, and $L_{ij}$. The better the conductivity of the tube $ij$ is and the larger the pressure difference $\Delta P_{ij}$ is, the more the flux through the tube $ij$ is, while the longer the length of the tube $ij$ is, the less the flux through the tube $ij$ is.

Suppose that the capacity of each node is zero, the conservation law of each node is calculated from the following:

$$\sum_j Q_{ij} = \begin{cases} I, & i = s \ \ j \in N_i, \\ -I, & i = d \ \ j \in N_i, \\ 0, & \text{others } \ j \in N_i, \end{cases} \tag{5}$$

where $I$ is the flux flowing from the source node (or into the sink node). It should be noted that $I$ is a constant in *Physarum* model, which means that the total flux is fixed constant throughout the process.

Equation (5) illuminates the flux relationship in each node. In the source node $s$, $I$ is the flux flowing from it; in the sink node $d$, $I$ is the flux flowing into it; and in intermediate nodes, the sum of flowing from and flowing into is zero.

*Physarum* forages for distributed food sources through the adaptive behavior of the plasmodium. The adaptive behavior is illustrated as follows combined with Figure 4(a), where two food sources are connected by two tubes. Because of $\Delta P_{ij}^1 = \Delta P_{ij}^2$ and $L_{ij}^1 > L_{ij}^2$, the flux $Q_{ij}^2$ will be greater than $Q_{ij}^1$ from (4). Note that $L_{ij}^1$ and $L_{ij}^2$ are kept constant throughout the adaptation process in contrast to $D_{ij}$; therefore, the adaptive behavior can be described by the evolution of $D_{ij}(t)$,

$$\frac{d}{dt} D_{ij} = \varphi\left(\left|Q_{ij}\right|\right) - \delta D_{ij}, \tag{6}$$

where $\delta$ is a decay rate of the tube. Equation (6) implies that the conductivity tends to vanish if there is no flux along the edge, while it is enhanced by the flux. It is natural to assume that $\varphi(\cdot)$ is a monotonically increasing continuous function satisfying $\varphi(0) = 0$.

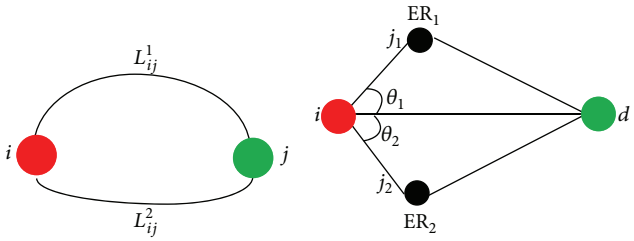



Figure 4: (a) If two food sources are connected by two tubes, the longer tube will vanish with time going by. (b) If there are two candidates for next hop, the node that has the greater $k \times \mathrm{ER} + (1 - k)L\cos\theta$ will be picked as the next hop.

Equation (6) illustrates the variation relationship of the conductivity with time to accommodate the flux distribution of the multipath transmission. In equilibrium ($\varphi(|Q_{ij}|) = \delta D_{ij}$ for all edges), the flow through any edge is steady. In nonequilibrium, the diameter grows or shrinks if $\varphi(|Q_{ij}|)$ is larger or smaller than $\delta D_{ij}$, respectively.

The PiPf consisting of (4), (5), and (6) describes the evolutionary process of *Physarum* to solve the path-finding behavior of self-organized networks.

*3.3. Physarum-Inspired Selecting Next Hop Model.* In this section, we improve the PiPf and make it fit for routing in WSNs based on dimensionless analysis method. That is to say, we improve the PiPf to achieve a *Physarum*-inspired selecting next hop model (P-iSNH). In order to obtain that, there are two problems that need to be solved. The first is which physical quantities are used to replace the conductivity $D_{ij}$, the length $L_{ij}$, and the pressure difference $\Delta P_{ij}$. The second is how to select the proper next hop.

Equation (4) derives from fluid dynamics. $D_{ij}$ is a measure of the conductivity of the tube; $L_{ij}$ is the length of the tube; and $\Delta P_{ij}$ is the differential pressure of tube on both ends. However, the $D_{ij}$, $L_{ij}$, and $\Delta P_{ij}$ cannot be directly used in WSNs where we need to consider the link quality, energy residue, transmission direction, and the distance of one hop.

First, because the $D_{ij}$ is an inherent characteristic of the tube, we should replace the $D_{ij}$ by an inherent physical quantity. Apparently, the link quality $\Phi_{ij}$ is an inherent characteristic relating to wireless link, so we replace the $D_{ij}$ by $\Phi_{ij}$ in our model.

Second, the meaning of $L_{ij}$ is the same as in fluid dynamics. However, in wireless communication, there is a path-loss exponent $\alpha$, which has a great effect on transmission. Therefore, we replace the $L_{ij}$ by $L_{ij}^{\alpha}$ in our model.

Third, we discuss the $\Delta P_{ij}$ combind with Figure 4(b). On one hand, suppose that there is a potential field from node $i$ to sink $d$. The potential difference between $i$ and $j$ can be expressed by $K \times L_{ij}\cos\theta$. Because parameter $K$ is unimportant in the judge process, we use $L_{ij}\cos\theta$ expressing $K \times L_{ij}\cos\theta$. When node $i$ chooses its next hop, it is apt to pick the node whose potential difference is much greater. On the other hand, because the next hop needs to consume energy to deal with data packets, it is apt to pick the node with much higher energy residue as the next hop. Therefore,

| | | | | | |
|---|---|---|---|---|---|
| ER$_{ii}$ | ER$_{ii}$ | | | | |
| ER$_{ij}$ | ER$_{i1}$ | ER$_{i2}$ | ER$_{i\cdots}$ | ER$_{ij}$ | ER$_{i\cdots}$ $j \in N_i$ |
| $\theta_{jid}$ | $\theta_{1id}$ | $\theta_{2id}$ | $\theta_{\cdots id}$ | $\theta_{jid}$ | $\theta_{\cdots id}$ $j \in N_i^R$ |
| $\beta_{ipj}$ | $\beta_{ip1}$ | $\beta_{ip2}$ | $\beta_{ip\cdots}$ | $\beta_{ipj}$ | $\beta_{ip\cdots}$ $j \in N_i^L$ |
| $L_{ij}$ | $L_{i1}$ | $L_{i2}$ | $L_{i\cdots}$ | $L_{ij}$ | $L_{i\cdots}$ $j \in N_i$ |

Figure 5: Conserved data structures.

we replace the $P_j$ by $k \times \mathrm{ER}_j + (1 - k)L_{ij}\cos\theta_{jid}$. Since $P_i$ is the base pressure, we replace $\Delta P_{ij} = P_j - P_i$ by $P_j$ through omitting $P_i$. Using (4), we have

$$Q_{ij} = \frac{\Phi_{ij} \times \left[k \times \mathrm{ER}_j + (1 - k)\, L_{ij}\cos\theta_{jid}\right]}{L_{ij}^{\alpha}}, \tag{7}$$

where $Q_{ij}$ is the virtual communication fluxes through the wireless link $ij$; $\Phi_{ij}$ is the link quality; $\mathrm{ER}_j$ is the energy residue of node $j$; $L_{ij}$ is the Euclidean distance of nodes $i$ and $j$; $\alpha$ is path-loss exponent; $\theta_{jid}$ is the angle of deviation and its range is $[-\pi/2, \pi/2]$; $k$ is a proportionality factor which uses to adjust the weight of $\mathrm{ER}_j$ and $L_{ij}\cos\theta_{jid}$.

Then, we discuss how to choose the proper next hop. As related in Section 3.2, since $\Delta P_{ij}^1 = \Delta P_{ij}^2$ and $L_{ij}^1$ and $L_{ij}^2$ are kept constant throughout the adaptation process in contrast to $D_{ij}$, the PiPf can only achieve the adaptation by the evolution of $D_{ij}(t)$. In our scenario, node $i$ chooses the next hop form candidates as shown in Figure 4(b). Since (1) $\Phi_{ij}^1 = \Phi_{ij}^2$ and $L_{ij}^1$ and $L_{ij}^2$ are kept constant according to the assumptions and (2) $\Delta P_{ij_1}$ and $\Delta P_{ij_2}$ are different and time-varying, we can achieve the adaptation by the evolution of $\Delta P_{ij}(t)$. If letting the monotonically increasing continuous function $\varphi(Q) = Q^{\mu}$, we have

$$\frac{d}{dt}\Delta P_{ij} = \varphi\left(|Q|\right) - \delta\Delta P_{ij} = \left(\frac{\Phi_{ij}\Delta P_{ij}}{L_{ij}^{\alpha}}\right)^{\mu} - \delta\Delta P_{ij}, \tag{8}$$

where $\delta$ is a decay rate of $\Delta P_{ij}$ and $\mu$ is a constant satisfying $\mu > 0$. We use (8) to determine the next hop in our P-iSNH; namely, we choose the node whose $(d/dt)\Delta P_{ij}$ is maximal as the next hop.

## 4. P-iSNH Based Routing Strategy and Algorithm

*4.1. Data Conserved.* In this section, we introduce the data which should be conserved in each node. Because of the same characteristic of each node, we suppose that the $\Phi_{ij}$ of each link is the same and ignore it to simplify discussion.

Each node $i$ needs to conserve the following information: $L_{ij}$ ($j \in N_i$), $\mathrm{ER}_{ij}$ ($j \in N_i$), $\mathrm{ER}_{ii}$, $|\theta_{jid}|$ ($j \in N_i^R$), and $\beta_{ipj}$ ($j \in N_i^L$) which are shown in Figure 5, where node $p$

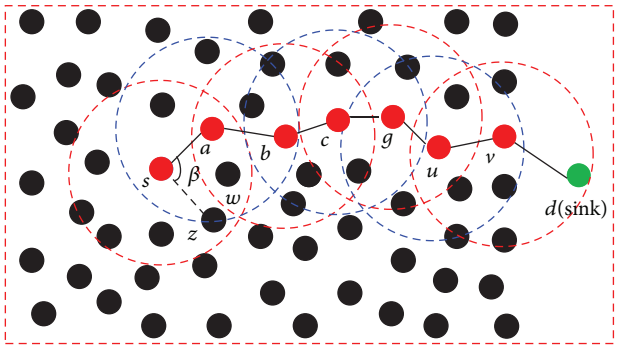



Figure 6: Process of routing selection.

is the previous hop of node $i$, $\mathrm{ER}_{ii}$ represents the $\mathrm{ER}_i$ stored in node $i$, and $\mathrm{ER}_{ij}$ represents the $\mathrm{ER}_j$ stored in node $i$. As our WSNs are location aware, the $L_{ij}$, $\theta_{jid}$, and $\beta_{ipj}$ are easily acquired following from (1) and (2). Note that the nodes in our WSNs are static, and we only need to calculate the $L_{ij}$, $\theta_{jid}$ and $\beta_{ipj}$ once at WSNs deployment time. For the difference of ER and $L_{ij}$, we normalize them to $\widehat{\mathrm{ER}}$ and $\widehat{L}_{ij}$, respectively. Therefore, we obtain

$$Q_{ij} = \frac{k \times \widehat{\mathrm{ER}}_j + (1-k)\,\widehat{L}_{ij}\cos\theta_{jid}}{\widehat{L}_{ij}^{\alpha}} \tag{9}$$

$$\frac{d}{dt}\Delta P_{ij} = \left(\frac{k \times \widehat{\mathrm{ER}}_j + (1-k)\,\widehat{L}_{ij}\cos\theta_{jid}}{\widehat{L}_{ij}^{\alpha}}\right)^{\mu} - \delta\left[k \times \widehat{\mathrm{ER}}_j + (1-k)\,\widehat{L}_{ij}\cos\theta_{jid}\right]. \tag{10}$$

*4.2. Routing Strategy.* If node $s$ needs to send data to the sink $d$, it searches for a routing in the following method. We illustrate the routing strategy combined with Figure 6.

*Step 1.* Each $(d/dt)\Delta P_{sj}$ $(j \in N_s^{\mathrm{R}})$ is calculated following from (10), where $\theta_{jsd}$, $\widehat{\mathrm{ER}}_{sj}$, and $\widehat{L}_{sj}$ are conserved and stored in node $s$ beforehand.

*Step 2.* Each node $j \in N_s^R$ is saved into a temporary array variable *Temp* in descending order by $(d/dt)\Delta P_{sj}$.

*Step 3.* The first node in *Temp* is picked as the next hop of the routing.

*Step 4.* If the next hop $a$ of node $s$ satisfies $N_a^R = \emptyset$, namely, there is an energy hole in the right side of node $a$, the node $a$ will not send *ACK* to $s$. Then, the node $s$ will trigger a specific processing routine.

*Step 5.* If $|Temp[0] \cdot \theta_{jsd} - Temp[1] \cdot \theta_{jsd}| \geq \pi/2$, node $s$ will choose the node $Temp[1]$ as the next hop. Then, the regular processing routine is going on.

*Step 6.* Otherwise, each $(d/dt)\Delta P_{sj}$ $(j \in N_s^L)$ is calculated following from (11) and the nodes are saved into the *Temp* in ascending order by $(d/dt)\Delta P_{sj}$. Then, the first node in *Temp* is chosen as the next hop of the routing and the regular processing routine is going on

$$\frac{d}{dt}\Delta P_{ij} = \left(\frac{k \times \widehat{\mathrm{ER}}_j + (1-k)\,\widehat{L}_{ij}\beta_{asj}}{\widehat{L}_{ij}^{\alpha}}\right)^{\mu} - \delta\left[k \times \widehat{\mathrm{ER}}_j + (1-k)\,\widehat{L}_{ij}\beta_{asj}\right], \tag{11}$$

where $\beta_{asj}$ is the angle of line $sa$ and line $sj$. Equation (11) indicates that it tends to choose a node which sharply deviates from the failing node, for example, $a$, as the next hop to avoid entering the energy hole again.

*Step 7.* The process is repeated, like a rolling wheel, until the sink $d$ is found.

*4.3. Routing Algorithms.* Given the data conserved and routing process in the preceding sections, the P-iRP's algorithms of initialization, regular processing routine, receiving routine, and specific processing routine are described by Algorithm 1, Algorithm 2, Algorithm 3, and Algorithm 4, respectively.

Based on the WSNs scenario in Section 3.1, suppose that the degree of graph $G$ is $D(G)$ which can be regarded as a constant, the complexity of Algorithm 1, Algorithm 2, Algorithm 3, and Algorithm 4 are $O(n \times D(G)^2) = O(n)$, $O(D(G)) = O(1)$, $O(1)$, and $O(D(G)) = O(1)$, respectively. From Figure 2, the number of intermediate nodes from node $s$ to sink $d$ is approximately $\sqrt{5n}/2$ in the worst case. Note that Algorithm 1 is run only once at WSNs deployment time. Therefore, the complexities of P-iRP is $O(\sqrt{5n}/2) = O(\sqrt{n})$ in running time, which greatly reduces the processing delay and saves the energy of sensors.

## 5. P-iSNH Analysis

In this section, we analyze the feasibility of P-iSNH by mathematical theoretical analysis. We study the cases in which two nodes connected to the same node compete to be the next hop, as shown in Figure 4(b).

There are four nodes $i$, $j_1$, $j_2$, and sink $d$. For simplicity, we hereafter replace $L_{ij_1}, L_{ij_2}, Q_{ij_1}, Q_{ij_2}, \Delta P_{ij_1}$, and $\Delta P_{ij_2}$ by $L_1$, $L_2$, $Q_1$, $Q_2$, $\Delta P_1$, and $\Delta P_2$, respectively. In multipath routing, the virtual fluxes along each path are calculated as

$$Q_1 = \frac{\Delta P_1/L_1^{\alpha}}{\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}}, \qquad Q_2 = \frac{\Delta P_2/L_2^{\alpha}}{\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}}. \tag{12}$$

Since $Q_1$ and $Q_2$ are nonnegative, adaptation equation (8) becomes

$$\frac{d}{dt}(\Delta P_1) = \varphi(Q_1) - \delta \cdot \Delta P_1, \qquad \frac{d}{dt}(\Delta P_2) = \varphi(Q_2) - \delta \cdot \Delta P_2. \tag{13}$$

```
(1)  for each node i do
(2)    for each node j(j ∈ N_i^L) do
(3)      R_ij = 1;
(4)      if (L_ij is not initialized)
(5)        initializing and normalizing L_ij and L_ji following (1);
(6)      end if
(7)      for each node k(k ∈ N_j^R) do
(8)        calculating β_ijk following (2);
(9)      end for
(10)   end for
(11)   for each node j(j ∈ N_i^R) do
(12)     R_ij = 1;
(13)     calculating θ_jid following (2);
(14)     if (L_ij is not initialized)
(15)       initializing and normalizing L_ij and L_ji following (1);
(16)     end if
(17)   end for
(18)   R_ii = 1;
(19) end for
```

Algorithm 1: Initialization.

```
(1) for each node j(j ∈ N_i^R) do
(2)   calculating (d/dt)ΔP_ij following from (10);
(3)   save j and (d/dt)ΔP_ij into Temp in descending order by (d/dt)ΔP_ij;
(4) end for
(5) P · SA = i;
(6) P · NA = Temp[0];
(7) P · ER = R_ii;
(8) send P;
```

Algorithm 2: Regular processing.

Setting $\varphi(Q) = Q^{\mu}$, $(d/dt)(\Delta P_1) = 0$, and $(d/dt)(\Delta P_2) = 0$, we have

$$\begin{aligned}\left(\frac{\Delta P_1/L_1^{\alpha}}{\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}}\right)^{\mu} &= \delta \cdot \Delta P_1\\ \left(\frac{\Delta P_2/L_2^{\alpha}}{\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}}\right)^{\mu} &= \delta \cdot \Delta P_2.\end{aligned} \tag{14}$$

After some calculations, we obtain

$$\begin{aligned}\Delta P_1 &= \frac{1}{\delta}\left[\frac{1}{\left(1 + \left(L_1^{\alpha}/L_2^{\alpha}\right)^{1/1-\mu}\right)}\right]^{\mu}\\ \Delta P_2 &= \frac{1}{\delta}\left[\frac{1}{\left(1 + \left(L_2^{\alpha}/L_1^{\alpha}\right)^{1/1-\mu}\right)}\right]^{\mu}.\end{aligned} \tag{15}$$

Namely, there is an equilibrium point given by $(\Delta P_1, \Delta P_2)$. We perform the simulation using MATLAB by setting the parameters $\alpha = 2$, $\mu = 0.8$, $\delta = 0.3$, $L_1 = 10$ and $L_2 = 12$ following from (14), and the solutions are shown in Figure 7, where two curves intersect in a point $E$ which superpose on the equilibrium point $(\Delta P_1, \Delta P_2)$.

We present a linear stability analysis at the equilibrium point in before parameters. The Jacobi matrix $J$ on the right-hand side of (13) is calculated as

$$J = \begin{pmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{pmatrix}, \tag{16}$$

where

$$J_{11} = \frac{\mu {Q_1}^{\mu-1} \Delta P_2/L_2^{\alpha}}{L_1^{\alpha}\left(\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}\right)^2} - \delta$$

$$J_{12} = \frac{-\mu {Q_1}^{\mu-1} \Delta P_1/L_1^{\alpha}}{L_2^{\alpha}\left(\Delta P_1/L_1^{\alpha} + \Delta P_2/L_2^{\alpha}\right)^2}$$

```
(1)  while (no receiving wireless radio)
(2)     node i sleep;
(3)  end while
(4)  wake node i;
(5)  R_iP-SA = P · ER;
(6)  if (i! = P · NA)
(7)     go on sleeping;
(8)  else
(9)     receiving packet P;
(10) end if
```

Algorithm 3: Receiving processing.

```
(1)  if (not receive ACK from next hop before deadline)
(2)     if (|Temp[0] · θ_jsd − Temp[1] · θ_jsd| ≥ π/2)
(3)        P · SA = i;
(4)        P · NA = Temp[1];
(5)        P · ER = R_ii;
(6)        send P;
(7)     else
(8)        for each node j (j ∈ N_i^L) do
(9)           calculating (d/dt)ΔP_ij following (11);
(10)          save j and (d/dt)ΔP_ij into Temp in ascending order;
(11)       end for
(12)       P · SA = i;
(13)       P · NA = Temp[0] · j;
(14)       P · ER = R_ii;
(15)       send P;
(16)    end if
(17) end if
```

Algorithm 4: Specific processing.

$$J_{21} = \frac{-\mu {Q_2}^{\mu-1} \Delta P_2 / L_2^{\alpha}}{L_1^{\alpha} \left(\Delta P_1 / L_1^{\alpha} + \Delta P_2 / L_2^{\alpha}\right)^2}$$
$$J_{22} = \frac{\mu {Q_2}^{\mu-1} \Delta P_1 / L_1^{\alpha}}{L_2^{\alpha} \left(\Delta P_1 / L_1^{\alpha} + \Delta P_2 / L_2^{\alpha}\right)^2} - \delta \tag{17}$$

and the Jacobi matrix at equilibrium point $E$ is denoted $J(E)$. After some calculations, the following formula is obtained

$$J(E) = \begin{pmatrix} \delta\left(\mu Q_2^* - 1\right) & -\delta\mu \dfrac{L_1^{\alpha}}{L_2^{\alpha}} Q_1^* \\ -\delta\mu \dfrac{L_2^{\alpha}}{L_1^{\alpha}} Q_2^* & \delta\left(\mu Q_1^* - 1\right) \end{pmatrix} = \delta \cdot \begin{pmatrix} \mu Q_2^* - 1 & -\mu \dfrac{L_1^{\alpha}}{L_2^{\alpha}} Q_1^* \\ -\mu \dfrac{L_2^{\alpha}}{L_1^{\alpha}} Q_2^* & \mu Q_1^* - 1 \end{pmatrix}, \tag{18}$$

where $Q_1^*$ and $Q_2^*$ are virtual communication fluxes along the first and second wireless links at the equilibrium point $E$. Using the relation $Q_1^* + Q_2^* = I$, we have

$$\det J(E) = \delta\left(1 - \mu I\right), \qquad \operatorname{tr} J(E) = \delta\left(\mu I - 2\right). \tag{19}$$

Note that $\delta$ is the decay rate of $\Delta P_{ij}$ and $\delta > 0$. If we let $I = 1$, thus,

$$\begin{gathered} \det J(E) < 0 \quad \text{for } \mu > 1, \\ \det J(E) > 0 \operatorname{tr} J(E) < 0 \quad \text{for } 0 < \mu < 1. \end{gathered} \tag{20}$$

Since we set $\mu = 0.8$, this means that the equilibrium point $E$ is stable. Therefore, the routing of WSNs will reach to equilibrium with our P-iSNH, which is very important to a routing strategy.

## 6. Simulation Results

We design a simulation platform using C++ to validate P-iRP. In the simulation, 441 sensors are relatively regularly

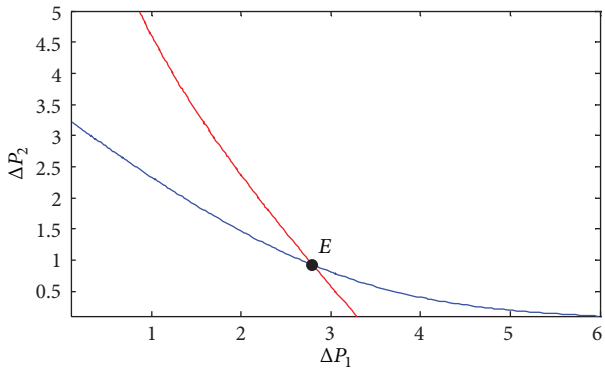



FIGURE 7: Simulation of the solution. In $\alpha = 2$, $\mu = 0.8$, $\delta = 0.3$, $L_1 = 10$, and $L_2 = 12$, two curves which come from (14) intersect in a point $E$, which is the sole equilibrium point.

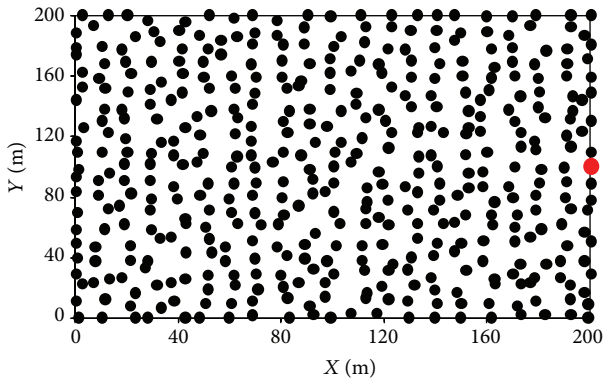



FIGURE 8: Sensor nodes deployment.

deployed in the field of 200 m × 200 m, and the sink node is deployed in the right of the field, shown in Figure 8. The sensing radius of each sensor is 30 m, the original energy of each node is 100, and the energy of sink node is inexhaustible. We suppose that the energy consumption of one transmission is 1, if the transmission distance is 20 m. Therefore, the energy consumption of one transmission of two nodes $i$ and $j$ is $(L_{ij}/20)^{\alpha}$, where $\alpha$ is set to 2.

In order to validate the energy equilibrium, we only choose the nodes in the field of $[(0, 0), (1\text{ m}, 1\text{ m})]$ to transmit data to the sink. If a chosen node transmits a group of data to sink, the P-iRP is used to choose next hops until the sink is found, which is called a *round*. This iterative process will halt after $n$ rounds until WSNs break down. We run GPSR, GEAR ($k = 0.5, 0.9$) and P-iRP ($k = 0.5, 0.9$) 10 times, respectively, to acquire their average value and compare them, where we use $k$ replace $\alpha$ which is used in GEAR to bring into correspondence with P-iRP. If the distance between the nodes and sink is less than 30 m, we let the nodes directly transmit data to sink to quicken convergence of P-iRP, and the energy consumption is set to 1. From Figure 8, there are 18 sensors around the sink. Therefore, the ideal number of rounds of simulation process is $1800/(L_{ij}/20)^{\alpha}$.

*6.1. Energy Equilibrium of P-iRP.* Figure 9 illustrates the energy distribution of GPSR, GEAR ($k = 0.5$), and P-iRP ($k = 0.5$) in different rounds. We can infer that (1) the energy distributions of GPSR are very imbalanced, (2) the energy distributions of GEAR and P-iRP are rather balanced, and (3) the energy distributions of P-iRP are more balanced than those of GEAR.

Figure 10 illustrates the lifetime of WSNs. In GPSR, the first dead node emerges in round of 192, and the WSNs break down in round of 889. In GEAR ($k = 0.5$), the first dead node emerges in round of 910, and the WSNs break down in round of 1223. In P-iRP ($k = 0.5$), the first dead node emerges in round of 1112, and the WSNs break down in round of 1396. In GEAR ($k = 0.9$), the first dead node emerges in round of 1428, and the WSNs break down in round of 1592. In P-iRP ($k = 0.9$), the first dead node emerges in round of 1542, and the WSNs break down in round of 1696. Therefore, the lifetime of GEAR ($k = 0.5$) is 48.8% longer than that of GPSR; the lifetime of P-iRP ($k = 0.5$) is 14.2% longer than that of GEAR ($k = 0.5$); and the lifetime of P-iRP ($k = 0.9$) is 6.5% longer than that of GEAR ($k = 0.9$). From Figure 9 and Figure 10, we can differ that (1) whether considering energy residue of next hops or not will impacts on the lifetime of WSNs greatly, and (2) in energy balanced WSNs, the time period is very short from emerging dead nodes to networks breaking down because all nodes reach to exhausted status of energy in the same time period.

Figure 11 illustrates the dead nodes distributions of GEAR ($k = 0.5$) and P-iRP ($k = 0.5$) in the rounds of 1380. The results show that P-iRP ($k = 0.5$) has much less dead nodes than GEAR ($k = 0.5$). We can also differ that the dead nodes of both algorithms are converged on a specific field but do not spread around the entire range of WSNs, which is useful in deploying such WSNs to prolong the lifetime.

The reasons to gain the results of Figures 9, 10, and 11 are that (1) since GPSR does not take energy into account, it utilizes frequently the "hot" nodes to result in imbalanced energy distributions, (2) since GEAR and P-iRP consider both energy and location of nodes, their energy distributions are rather balanced, and (3) since P-iRP is more elaborate in energy utilization than GEAR, the energy distributions of P-iRP are more balanced than those of GEAR.

*6.2. Efficiency of P-iRP.* Figure 12 illustrates the number of hops that the different algorithms need in different rounds of transmission. By calculating, the average hops of GPSR, GEAR ($k = 0.5$), P-iRP ($k = 0.5$), GEAR ($k = 0.9$), and P-iRP ($k = 0.9$) are 19.4, 24.3, 21.6, 28.9, and 26.8, respectively.

In case of $k = 0.5$, the average hops of P-iRP are 11.3% more than those of GPSR, and the hops of GEAR are 23.3% more than those of GPSR. Combined with Figure 10, the increment of average hops of 11.3% will lead to the increment of lifetime of more than 60% from GPSR to P-iRP, while the increment of average hops of 23.2% will only lead to the increment of lifetime of about 48% from GPSR to GEAR. Therefore, the P-iRP is more efficient in balance of routing efficiency and energy equilibrium than GEAR.

In case of $k = 0.9$, the average hops of P-iRP are 38.1% more than those of GPSR, and the hops of GEAR are 49.0%

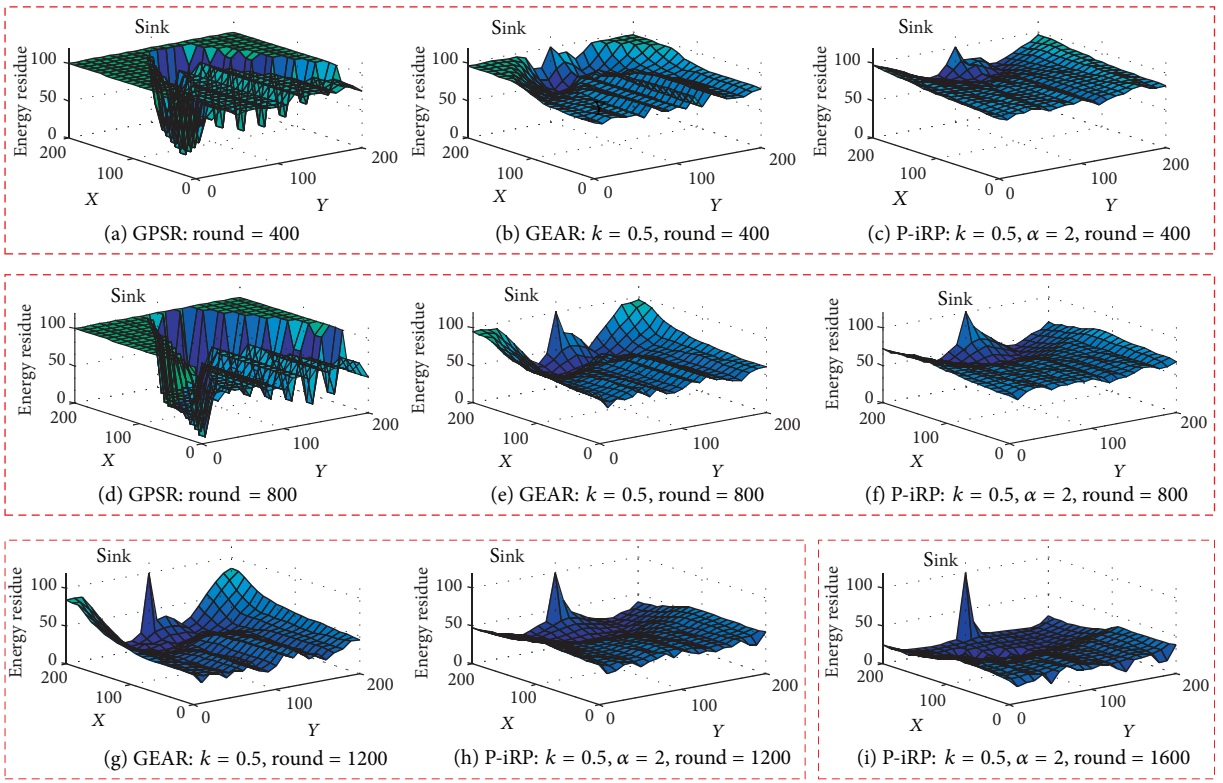



Figure 9: Hops of transmission of each round.

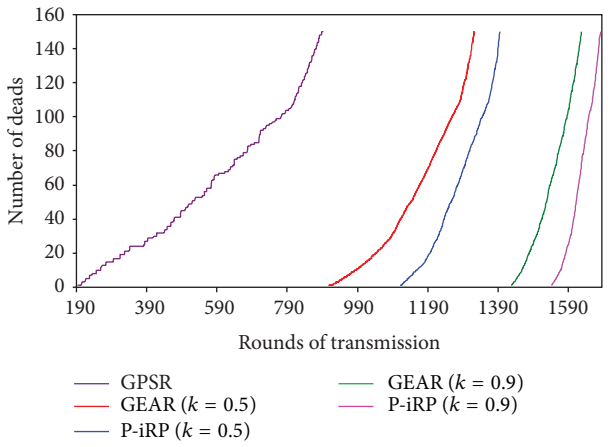


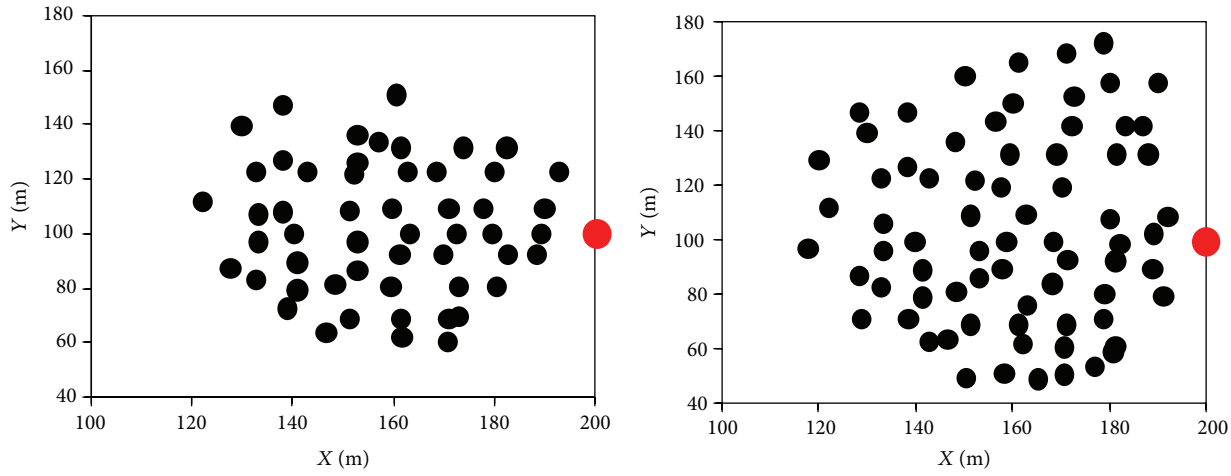

Figure 10: Lifetimes of WSNs.

more than those of GPSR. Combined with Figure 10, the increment of 11.3% of average hops will lead to the increment of about 60% of lifetime from GPSR to P-iRP ($k$ = 0.5), while the increment of 24% of average hops will only cause the increment of 21.5% of lifetime from P-iRP ($k$ = 0.5) to P-iRP ($k$ = 0.9). That is to say, the larger $k$ is, the smaller the increment of $k$ impacts on lifetime of WSNs. Therefore, it is improper to set a larger $k$, so does GEAR.

In addition, Figure 12 implies that (1) GPSR can get much higher routing efficiency than those of GEAR and P-iRP at initial periods of time in WSNs' lifetime, but its efficiency decreases exponentially with the time going, and (2) the routing efficiencies of GEAR and P-iRP are almost the same at initial periods of time in WSNs' lifetime, while the difference becomes gradually large with the time going. The reasons to gain those results are that (1) since GPSR does not take energy into account, the routing efficiency will decrease exponentially after dead nodes emerge and (2) P-iRP can outperform GEAR in routing efficiency, since P-iRP considers comprehensively the distance, energy residue, and location of the next hop, other than only considering energy and location in GEAR.

## 7. Conclusion

The *Physarum* forages for patchily distributed food sources through accommodating its body to form networks with comparable efficiency, fault tolerance, and cost, which is the source of P-iRP's inspiration. For the proposed scenario, the P-iRP ensures the passage of data packets through one by one static sensor nodes to reach the sink. In each intermediate node, the P-iSNH is used to choose the proper next hop. Once an energy hole emerges, a specific processing will be triggered to bypass the hole. The theoretical analysis and simulation results show that the P-iRP possesses many advantages, for example, rather low algorithm complexity for P-iRP, ever-present equilibrium solution for P-iSNH, and effective trade-off between routing efficiency and energy equilibrium compared to other famous algorithms, which greatly reduces the processing delay and saves the sensors' energy and also demonstrates that the P-iRP is applicable to the proposed




(a) P-iRP, $k = 0.5$, round = 1380

(b) GEAR, $k = 0.5$, round = 1380

Figure 11: Distributions of dead sensor nodes.

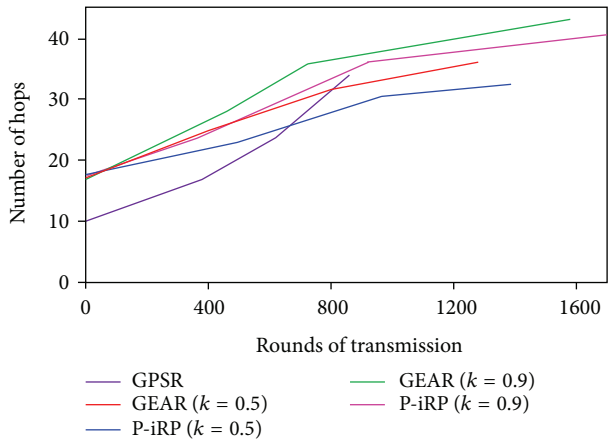



Figure 12: Hops of transmission of each round.

scenario. Furthermore, we consider that the model may also provide a useful help to develop the routing protocol in mobile ad hoc networks, which will be our future focus.

## Acknowledgments

This work was partially supported by the National High-Tech Research and Development Program of China (863 Program) under Grant no. 2011AA010701, in part by the National Basic Research Program of China (973 Program) under Grant no. 2013CB329102, in part by the National Natural Science Foundation of China (NSFC) under Grant nos. 61003283, 61001122, 61232017, U1204614, 61003035, and 61142002, and in part by the Natural Science Foundation of Jiangsu Province under Grant no. BK2011171.

## References

[1] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: a nonlinear neural circuit approach," *Transactions on Embedded Computing Systems*, 2013.

[2] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, New York, NY, USA, August 2000.

[3] F. Kuhn, R. Wattenhofer, and A. Zollinger, "An algorithmic approach to geographic routing in ad hoc and sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 51–62, 2008.

[4] K. Padmanabh, A. M. Reddy V., S. Sen, and P. Gupta, "Random Walk on Random Graph based Outlier detection in wireless sensor networks," in *Proceedings of the 3rd International Conference on Wireless Communication and Sensor Networks (WCSN '07)*, pp. 45–49, Allahabad, India, December 2007.

[5] G. Trajcevski, F. Zhou, R. Tamassia, and B. Avii, "On the construction of data aggregation tree with minimum energy cost in wireless sensor networks: NP-completeness and approximation algorithms," in *Proceedings of the INFOCOM*, pp. 2591–2595, Orlando, Fla, USA, 2012.

[6] M. H. Chaudhary and L. Vandendorpe, "Battery-aware power allocation for lifetime maximization of wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, Cape Town, South Africa, May 2010.

[7] J. Zhang, S. Ci, H. Sharif, and M. Alahmad, "Lifetime optimization for wireless sensor networks using the nonlinear battery current effect," in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, Dresden, Germany, June 2009.

[8] C.-K. Chau, F. Qin, S. Sayed, M. H. Wahab, and Y. Yang, "Harnessing battery recovery effect in wireless sensor networks: experiments and analysis," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1222–1232, 2010.

[9] J. Yang and S. Ulukus, "Optimal packet scheduling in an energy harvesting communication system," *IEEE Transactions on Communications*, vol. 60, no. 1, pp. 220–230, 2012.

[10] S. Bai, W. Zhang, G. Xue, J. Tang, and C. Wang, "DEAR: delay-bounded energy-constrained adaptive routing in wireless sensor networks," in *Proceedings of the INFOCOM*, pp. 1593–1601, Shanghai, China, 2012.

[11] Y. Yu, R. Govindan, and D. Estrin, "Geographical and Energy Aware Routing: a recursive data dissemination protocol for wireless sensor networks," Tech. Rep. UCLACSD TR-01-0023, UCLA Computer Science Department, 2001.

[12] N. Roseveare and B. Natarajan, "Energy-aware distri*f*buted tracking in wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '11)*, pp. 363–368, Cancun, Mexico, March 2011.

[13] S. Li, Z. Wang, and Y. Li, "Using Laplacian Eigenmap as heuristic information to solve nonlinear constraints defined on a graph and its application in distributed range-free localization of wireless sensor networks," *Neural Processing Letters*, vol. 37, no. 3, pp. 411–424, 2012.

[14] O. Yang and W. Heinzelman, "Sleeping multipath routing: a trade-off between reliability and lifetime in wireless sensor networks," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM '11)*, Houston, Tex, USA, December 2011.

[15] F. Sivrikaya, T. Geithner, C. Truong, M. A. Khan, and S. Albayrak, "Stochastic routing in wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '09)*, Berlin, Germany, June 2009.

[16] M. Zhang, C. Xu, J. Guan, R. Zheng, Q. Wu, and H. Zhang, "P-iRP: physarum-inspired routing protocol for wireless sensor networks," in *Proceedings of the VTC*, Las Vegas, Nev, USA, 2013.

[17] C. Xu, T. Liu, J. Guan, H. Zhang, and G.-M. Muntean, "CMT-QA: quality-aware adaptive concurrent multipath data transfer in heterogeneous wireless networks," *IEEE Transactions on Mobile Computing*, 2012.

[18] Y. Cao, C. Xu, J. Guan, F. Song, and H. Zhang, "Environment-aware CMT for efficient video delivery in wireless multimedia sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 381726, 12 pages, 2012.

[19] R. Zheng, M. Zhang, Q. Wu, S. Sun, and J. Pu, "Analysis and application of bio-inspired multi-net security model," *International Journal of Information Security*, vol. 9, no. 1, pp. 1–17, 2010.

[20] C. Xu, F. Zhao, J. Guan, H. Zhang, and G.-M. Muntean, "QoE-driven user-centric VoD services in urban multi-homed P2P-based vehicular network," *IEEE Transactions on Vehicular Technology*, 2012.

[21] S. Li, M. Q. H. Meng, W. Chen et al., "SP-NN: a novel neural network approach for path planning," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '07)*, pp. 1355–1360, Sanya, China, December 2007.

[22] J. Rao and A. Fapojuwo, "A battery aware distributed clustering and routing protocol for wireless sensor networks," in *Proceedings of the WCNC*, pp. 1538–1543, Shanghai, China, 2012.

[23] C.-L. Wang and S.-J. Syue, "An efficient relay selection protocol for cooperative wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '09)*, Budapest, Hungary, April 2009.

[24] G. Trajcevski, F. Zhou, R. Tamassia, B. Avci, P. Scheuermann, and A. Khokhar, "Bypassing holes in sensor networks: load-balance versus latency," in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM '11)*, Houston, Tex, USA, December 2011.

[25] T. Nakagaki, H. Yamada, and Á. Tóth, "Maze-solving by an amoeboid organism," *Nature*, vol. 407, no. 6803, p. 470, 2000.

[26] A. Tero, S. Takagi, T. Saigusa et al., "Rules for biologically inspired adaptive network design," *Science*, vol. 327, no. 5964, pp. 439–442, 2010.

[27] A. Tero, R. Kobayashi, and T. Nakagaki, "A mathematical model for adaptive transport network in path finding by true slime mold," *Journal of Theoretical Biology*, vol. 244, no. 4, pp. 553–564, 2007.

[28] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proceedings of the ACM Mobicom*, pp. 166–179, New York, NY, USA, 2001.