Self-adaptable Routing Scheme for In-network Processing

Xiwei Zhao[1], **S. Kami Makki**[2] and **Niki Pissinou**[1]
[1]Institute of Telecommunications and Information Technology,
Florida International University
e-mail : {xzhao001, pissinou}@fiu.edu
[2]Department of Electrical Engineering & Computer Science,
University of Toledo
e-mail : kmakki@eng.utoledo.edu

Abstract The wireless sensor network (WSN), is a type of self-organized distribution system without complex infrastructure requirement. However, WSN's independence and un-attended usage limit its power supply and life expectancy. These become a critical issue for any WSN implementation. The in-network processing which, intends to minify data volume locally can greatly reduce the energy consumption of data delivery over a long distance to the sink. However, open problems are still remain with in-network processing, such as how to carry out in-network processing, and how to combine routing scheme to the sink (e.g. the long distance delivery). For any WSN application, the following pre-assumption is vital. There exists a physical signal field which bridges physical events to the sensors of WSN, otherwise WSN does not work. In fact, the physical signal field can be depicted by sensory data of all the nodes, and can even be used for local data convergence (e.g. in-network processing). An immediate solution which we propose is that, sensory data of the nodes can be converged to local extremes of the physical signal field along its gradient direction. Furthermore, the physical signal field is integrated with the cost field routing scheme. The cost field routing is in charge of delivering data to the sink. During the in-network processing, the delivery direction is derived by combining linearly the gradient directions of the cost field and physical signal field. Also simulation results show that the proposed schemes are robust, adaptable, and reliable to the variation of physical events.

Keyword: Wireless Sensor Network, In-network process, Routing algorithm

1. Introduction

In recent years, improvements of micro-electronics promote applications of WSN. However, for each sensor node, its independence and un-attended usage confine its limited power supply no matter how powerful its embedded battery is. Thus, energy efficiency is always a key issue for the WSN implementation.

For any particular application, each node senses signal and generates sensory data independently. However, all sensory data are context-related, because they are in the same application. The relation of context-related of sensory data can be exploited for data processing, such as compressing or aggregating. This can reduce total data volume. Consequently, the energy consumption due to the long distance delivery to the sink can also be reduced. In general, sensor nodes are clustered by their location, and then all sensory data inside a cluster converge together for aggregation. Only high-level (aggregated) data (with smaller volume) are delivered to the sink. The process of local data convergence for aggregation is called in-network processing [1]. In the case when the nodes that generate data are far away from the sink, the in-network processing (local data convergence and aggregation) can save energy greatly.

One thing that we should be aware is that, context-related doesn't mean location-related and vice versa. For example, two wireless sensor nodes are far away from each other, and generate context-related data. In this case, in-network processing wouldn't be necessary because the delivery cost for aggregation is expensive. In other words, only those location-related nodes with context-related data can benefit from in-network processing.

In terms of the physical events that WSN is monitoring, some signals must be emitted from the event and can be sensed by WSN. Otherwise, WSN can not sense the physical events. In practice, the signal spreads among the regions, and its intensity may vary from point to point. The signal intensity of the region forms a physical signal field, and this field is the pre-requirement for WSN's application.

Before exploiting the physical signal field for in-network process, let's investigate how to express the physical signal field over WSN. In general, any event report should include three elements: when, where, and what. The sensory data value is the content of "what" and it corresponds to the signal intensity of the physical signal field at the node. The information of "when" and "where" can also be marked by the node that generates data. Thus, each node holds all information of the physical signal field at its point. The whole picture of the physical signal field can be constructed by collecting information from all the nodes of WSN.

In fact, all data packets that are generated from sensor nodes can also be sorted using their time-related, location-related and context-related information. The context-related and time-related, properties of the packet can be exploited for data aggregation. However all those three properties can be employed for local data convergence of in-network processing. In this paper, time-related is inferred by sampling frequency, and only context-related and location-related are considered for local data convergence.

In section 2 & 3, we review issues regarding cost field routing and examine the related work on the in-network processing. In section 4, the main idea is presented, and the physical signal field of the physical event is analyzed in detail. The algorithm and details of implementation are presented in section 5. In section 6, we discuss the performance of the algorithm and conduct simulation. In section 7, future studies are suggested.

2. Background

2.1 Gradient routing and “CoS”

In 2000, Robert D. Poor proposed his Gradient Routing (GRAd) over Ad hoc networks [2]. The distinguished feature of Ad hoc network is that none of nodes know others without hearing something from others. On-demand routing is an immediate choice for ad hoc network routing, because it follows the rule that “if you wish to be spoken to, you must speak first”. GRAd belongs to on-demand routing.

In [3], gradient routing is applied to WSN and enhanced to be GRAdient Broadcast (GRAB). Similar to GRAd, the scheme GRAB constructs and maintains a cost value field around the sink. At each node, its gradient direction of the field can be measured by probing its neighbor nodes. In GRAB, a band of the routing paths is selected for the data delivery, and a mechanism is proposed to control the width of the band.

The sink initiates construction of the cost field. The sink propagates advertisement (ADV) packets, in which the “cost value” is included. When an ADV packets starts from the sink, its cost value is zero. When an ADV arrives at a node, the node adds the cost of last hop of the ADV (the cost of last hop can be measured by the node) to the cost value that is carried by ADV. The new cost value is also copied to the memory of the node. In this way, the cost value is distributed and changed hop by hop. The cost value and the location of all nodes construct a cost field. If the height denotes the cost value at each node, then the field looks like a funnel in 3-D space.

When a data packet is sent out from a source node, it will carry the cost value of the source node. On the other hand, when a packet arrives at a node, the node will check the cost value of the packet first. If the cost value of the packet is smaller than (and equal to) that of the node, the packet is simply discarded. Otherwise, the node replaces the cost value of the packet with that of its own, and forwards the packet.

Be aware that more than one delivery path may exist, and all of those paths construct a band path to the sink. The width of the band can be controlled by the difference of cost values. When the packet arrives at a node, the difference between cost value of the packet and that of the node is calculated. For a given threshold, if the difference is larger than the threshold, the packet will be forwarded; otherwise, the packet will be discarded. The threshold may vary from its maximum (maximum can be estimated based on the transmission radius of a node and node density of WSN) to zero, and correspondently, the band of delivery paths would be adjusted from narrow to wide.

“CoS” (Center of Stimulus), an interesting concept, is also proposed in GRAB. According to the descriptions of GRAB in [3], any stimulus would create a field of sensing signal strength (e.g. the physical signal field) and more than one node (maybe tens of hundreds) may detect the stimulus. If each of those nodes sends a report, then resources will be wasted heavily. Thus, an election is necessary among those nodes that detect the same stimulus, and only one node would be elected to send the report to the sink. The elected node is called “CoS”. The election of CoS is based on the field of sensing signal strength. Only the node with the strongest signal would be CoS and generates a report to the sink. However, since the paper focuses on the scheme of “GRAB” which considers the construction of routing path to the sink, the discussion of in-network processing is simplified to CoS election.

2.2 In-network processing

To gain energy efficiency, sensory data should be processed locally. This is called in-network processing. In a particular application, sensory data are context-related. However, Source nodes may scatter among WSN: some of them may be close to each other; others may be far away from each other. It’s obvious that only those sources that are close to each other need to be clustered together for in-network processing.

Most schemes simply cluster nodes by their location, no matter whether they are sources or not. Some schemes divide the region into square patches, and each patch corresponds to one cluster. One node of each cluster acts as the cluster header. The cluster header can be elected by nodes of the cluster, or be predefined. Within a cluster, sensory data converge to the cluster header for in-network aggregation.

Other schemes cluster nodes by their routing path to the sink. In the case of the shortest path tree of the sink, branches are used for clustering, and nodes at the same branch are clustered together. In this way, the data of a source that is far from the sink may pass by source nodes that are near the sink, because those source nodes are at the same routing branch. Also in-network aggregation can be executed hop by hop, along the shortest path to the sink.

In some schemes, the sensing signal strength (of the physical signal field) is discussed. To identify boundary of the physical event, a scheme sets the minimum threshold of sensing signal strength. The header inside an event boundary is elected by the strength of sensing signal and therefore the strongest will win [14]. Just as in GRAB, CoS election is also based on sensing signal strength, and only the data of CoS are delivered to the sink.

In this paper, the physical signal field (e.g. the field of sensing signal) would be used for both node clustering and data converging. Following the “Cost-field routing” [2][3], we will combine the gradient directions of two fields (the cost field and the physical signal field) for data convergence.

3. Related work

Similar to database, WSN is also data-central. In [18], many techniques of databases, such as query and data retrieval, are applied to WSN. The property of data-central is investigated for in-network processing in [4]. Query is diffused (from the sink) among each cluster firstly. This is called directed diffusion. Nodes are then activated by the query, and sensory data (as a reply of the query) are delivered (from nodes) along the reverse path of the query diffusion. The sensory data can be aggregated at intermediate nodes when they passed by. In [15], different tags are assigned to sensory data with different semantic information. In this way, monitoring and in-network processing are easily conducted even in heterogeneous sensor network.

Routing issues of in-network processing are discussed in some papers [5, 8, 9, 10, 11, 17, 19]. In [5], In-network processing is investigated in term of energy costs and time delay, and three sub-optimal routing schemes are proposed. The first one is “Center at Nearest Source (CNS)”. The idea of CNS is that: within a cluster, the node nearest to the sink is selected to be the header; the in-network aggregation is conducted at the header; the aggregated data are sent to the sink directly. The second one is “Shortest Paths Tree (SPT)”. This means that the shortest routing path tree of the sink is used, and the data can be aggregated along the path. The third one, “Greedy Incremental Tree (GIT)”, is an iterative scheme. GIT grows up step by step. At each step, the nearest source to the current tree is added through the shortest path to the tree. The growth would stop when all source nodes are connected to the tree. For all three schemes above, the routing scheme toward the sink must be given in advance. In [9], in-network processing is analyzed regarding to energy efficiency. The nodes are clustered by the branch of the routing path tree toward the sink. The data sampling is triggered by time and by the variation of sensing signal.. Similar ideas can also be found in [10] and [11]. In [17], virtual infrastructure “Rail” is defined for data delivery. “Rail” acts as a rendezvous area of data. Thus, all data can be aggregated within the “Rail”. In [8], sensor nodes are clustered by location, and the cluster header is predefined. In [19], sensor network is split into sectors to avoid transmitting collision.

The routing path of WSN can be adaptable to the environment. In [6], each node routes the data based on energy of its own and its neighbors. In [7], sensory data packets are agent-based, and agent-based packets determine the routing path hop by hop toward the sink. Whenever a packet arrives at a node, its agent section will require the node to select the next hop of the packet delivery, based on remaining energy of the node and transmission cost of the next hop. Local target protocol (LTP) [12] is an extension of cost field routing. Moreover, LTP could make the data packets detour the concave holes of WSN. Similar to the cost field routing, during the “search phase” of LTP, the node locates its neighbors that are close to the sink, and the utmost neighbor is selected. During the “direct transmission phase” of LTP, the data packet is sent to that utmost neighbor. On the other hand, if this kind of neighbor cannot be found during the “search phase”, and the data has not arrived at the sink, “backtrack phase” is invoked. Then the data packet is sent back to the node where it came from.

Also, node clustering of in-network processing can also adapt to environment conditions. In [14], node clustering is associated with the location of physical events. Clusters are bounded by the signal amplitude that is sensed by nodes (e.g. physical signal intensity). A threshold is given in advance. The node is identified to be outside the cluster if its sensed signal amplitude is smaller than the threshold. Within the cluster, the header is also elected according to the physical signal amplitude, and the node with the highest signal amplitude would win. In [16], a localization system is proposed. Spotlight device (localization-related hardware) resides at an actuator. The spotlight device steers a laser light to measure the location of each node, and then informs the node of its measurement result. In fact, the process can also be described as: the spotlight device initiates an event (a laser light emitting). After that, the node senses the event (the laser light hits a node), and generates sensory data. At last, the sensory data is sent back (reflection) to the spotlight device for data processing. Moreover, node clustering can be conducted based on node location. Training protocol is an innovative scheme for nodes clustering [13]. In WSN, two nodes are selected to initiate a training process. These two nodes would construct two cost-fields respectively, and each node of WSN has two cost values correspondently. Those two cost values can be used as location coordinates. As a result, nodes with “similar” coordinates are grouped together to form a cluster.

Cluster is only a logical concept of a network which corresponds to locations. Therefore data convergence scheme within a cluster is obviously location related. On the other hand, physical events (or their physical signal field) are always location related. Thus, the data convergence can be location-related if data convergence associated with physical events (or their physical signal field) directly. In this way, nodes would be clustered naturally, and thus, there is no need to take care of node clustering any more.

4. Physical signal field

For any WSN application, an assumption is required that, there must be a signal emitted from the physical event, and we call this signal, a “sensing signal”. For instance, a group of people can spread in an area and act as sensors. Now, an object can move toward an area just as a physical event can emerge. Somebody may see the object is coming, somebody may see a black dot is moving, and somebody may see nothing. When the object goes into an area, more people can see the object clearly. But there are still some who are standing far away from the object and can not see it.

Now, we describe the example above in another way: assume there is a sight field around the object; that is the field intensity of any point which is related to the distance from the point to the object. If a person (sensor) stands at a point that is much closer to the object, the field intensity is relatively high and the person will see the object clearly. If another person stands at a point that is a little far from the object, the field intensity is relatively low and the person will see the object vaguely. With the increase of distance (between the person and the object), the field intensity decreases, and the observed image would become smaller and vaguer and finally disappears.

In general, any physical event has a field, and sensors would generate sensory data based on the field intensity. In the following discussion, we will call the field “physical signal field”. A physical event is always accompanied by its physical signal field and the physical signal field bridges physical event to the sensor nodes.

However, it’s not necessary that the physical event should always be surrounded by its physical signal field. Furthermore, the location where the event occurs can be different from where its physical signal field is located. Second, the physical signal fields initiated by different events can be overlapped. If we let the height of a 3-D space represent the intensity of a sensing signal, the shape of the physical signal field will look like mountains spanning among the region.

The physical signal field of physical events can also be used for routing, just as operation of the cost-field routing. From the above picture, we can see that the area can be divided into several small pitches by valleys and saddles, and this can also be used for the node clustering in WSN. Also, this clustering scheme is adaptable to the variation of the field, and to the case that some nodes may die.

Along the gradient direction of the physical signal field, data can be converged to these “mountain peaks”, and then aggregated at there. By exploiting the physical signal field of the physical event, overhead is avoided for the node clustering. In addition, this natural clustering scheme is more flexible and robust than any other clustering schemes.

5. Algorithm

In general, the routing scheme of WSN can be divided into two parts. One is in charge of delivering data packets to the sink, and the other is for in-network processing. Corresponding to the two parts, two different tags are assigned to all packets (or messages). One tag is “sink routing” (SR); the other is “in-network processing” (INP). We assume that the scheme of sink routing is given (the sink routing can be Reverse Path Forwarding, Cost Field Forwarding, Geographical Forwarding, etc.). In this paper, only the routing scheme for in-network processing is discussed.

Gradient direction of the physical signal field can be employed for in-network processing by many different ways. For instance, the gradient direction can be used directly, and drive messages (or packets) converge to local peak of the physical signal field. The gradient direction can also be combined with other routing schemes. Here, two routing schemes are proposed for in-network processing. The first one is called Center Convergence Scheme (CCS), which routes data to local peaks of the physical signal field. The second one is called Linear Combination Scheme (LCS), in which the routing direction is derived from the linear superposition of two directions: gradient direction of the physical signal field and the gradient direction of the cost field.

5.1 Center Convergence Scheme

In CCS, only gradient direction of the physical signal field is used for in-network processing. Whenever a message is generated at a node, it will be tagged by INP first. After that, the node probes its neighbors to find out who has the highest signal intensity (probing process). If signal intensity of the node is higher than that of all its neighbors, then the tag of the message should change to SR; otherwise, the message should be sent to the neighbor with the highest signal intensity. If the message is received from other nodes, its tag should be checked first: for the message with SR, the node will deliver it following the sink routing scheme; for the message with INP, a probing process is invoked just as the description above. In addition, buffer is defined at each node for the message aggregation.

CCS includes two programs that need to be executed separately and simultaneously at each node. Those two are written in pseudo codes by adapting the structure of C language.

The first program is to identify the tag of the message. When a message arrives, its tag will be checked first. If the tag is INP, then the message will be put into the buffer; if not, its tag must be SR, and the message will be delivered to the sink based on the sink routing scheme (We assume that the scheme of sink routing is given). If the message is generated by the node itself, the message will always be tagged INP, and put into the buffer. The pseudo codes of the first program are shown below.

```
if (message is generated at the node)
{       add the title INP to the message
        put the message into the buffer
}else if (message is received from other nodes)
{       check the title of the message
        if (the title is INP)
        {       put the message into the buffer
        }else if (the title is SR)
        {       deliver the message to sink
        }
}
```

The second program is used to probe its neighbors periodically, and then forwards INP or SR messages. When a predefined timer runs out, all messages in the buffer are aggregated. Next, we should assign tags to the aggregated messages. If the node is at a local peak of the physical signal field, the aggregated messages should be tagged SR, and are delivered by the given sink routing scheme. Otherwise, the tag should be INP, and the aggregated messages are sent toward the local peak along the gradient direction of the physical signal field.

```
while (the pre-defined timer runs out)
{       aggregate messages in the buffer
        let Val= data sensed by the sensor
        probe Val of neighbor nodes
        if (exist neighbor with Val larger than itself)
        {       title the aggregated message with INP
                select the neighbor with largest Val
                send the message to this neighbor
        }else
        {       title the aggregated message with SR
                deliver the message to sink
        }empty the buffer
        reset the timer
}
```

5.2 Linear Combination Scheme

LCS is a scheme for in-network processing. In addition, LCS requires the scheme of sink routing to be “Cost Field Based Forwarding” (“GRAB”).

So far, there are two fields over WSN: the cost field and the physical signal field. At each node, two gradient directions exist corresponding to those two fields. In LCS, those two gradient directions are linearly combined. The result of combination is used to direct local data convergence (of in-network processing).

One more thing we need to be aware of is that, for the cost field, the minimum cost value of 0 is at sink and the whole field looks like a funnel. On the other hand, the physical signal field looks like a mountain (which is a flip of funnel). Here, we simply let the cost value times -1 (flip the funnel over). Let the cost value be V_c and the signal intensity value be Vi, then its linear superposition value (LSV) is LSV = $-\alpha V_c + \beta V_i$, in which α and β are constant factors larger than zero.

Similarly, when a message is generated at a node, it should be tagged by INP first (all messages should be tagged either by INP or SR). For any message with an SR tag, the node will deliver it to the sink by GRAB (cost field routing). On the other hand, for the message with an INP tag, a probing process will be initiated by the node. The probing process is almost the same as in the central convergence scheme (CCS) except the probing value. This time, the probing value (which is signal intensity in CCS) is substituted by the linear super-position value (LSV). If the node is with its LSV highest, then the tag of the message will change to be SR; otherwise, the INP message will be sent to the neighbor with the highest LSV.

Two programs are designed to execute at each node separately and exchange data through the buffer. If the message is with INP tag, then it will be put into buffer; if its tag is SR, then the message will be delivered to the sink. The message generated by the node, will always be tagged by INP first, and then put into the buffer. The first program is the same as that of center convergence scheme (CCS).

```
if (message is generated at the node)
{       attach INP tag to the message
        put the message into the buffer
}else if (message is received from other nodes)
{       check the tag of the message
        if (the tag is INP)
        {       put the message into the buffer
        }else if (the tag is SR)
        {       send to the sink by cost field routing
        }
}
```

The second program is in charge of probing control and aggregation control. In addition, a timer is defined to synchronize the entire process. When the timer runs out, all messages in the buffer are aggregated, and the outputs are temporarily tagged with INP. Next, the probing process is initiated. If the node is with the LSV highest among its neighbors, those aggregated messages (which are temporarily tagged INP) change their tag to SR, and are delivered, based on the cost field forwarding (GRAB). Otherwise, those aggregated messages are sent to the neighbor with the highest LSV.

```
LSV = -∞
if (Vi exists) calculate LSV = -αVc + βVi
while (the timer is run out )
{       aggregate all messages in the buffer
        probe LSV of neighbor nodes
        if (exist a neighbor with larger LSV)
        {       tag the aggregated messages with INP
                select the neighbor with largest LSV
                send the aggregated message to
        }else
        {       tag the aggregated messages with SR
                send the aggregated messages
                toward sink by cost field routing
        }empty the buffer
        reset the timer
}
```

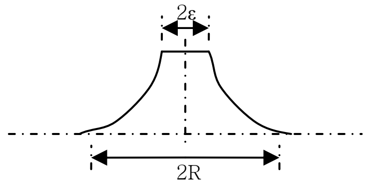
5.3 Analysis of CCS and LCS

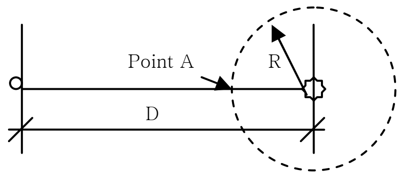
If the given scheme of sink routing is Cost Field Based Forwarding, the center convergence scheme (CCS) is just a special case of linear combination scheme (LCS) with "α" being equal to zero. In this sub-section, we will discuss the performance of LCS by adjusting α, β, and compare the performance of LCS and CCS.

To simplify the analysis, we assume the physical signal field covers a circle area with radius R, where the highest intensity (peak) of the field is at the centre of the circle. The distribution of intensity is the function below:

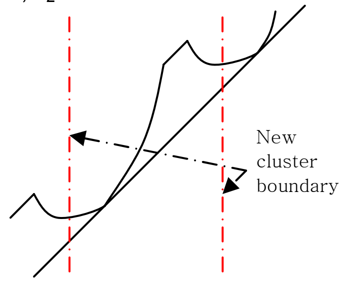
$V_i = 0$, for $\rho > R$;
$V_i = c_1/\rho^2$, for $R \geq \rho \geq \varepsilon$;
$V_i = c_1/\varepsilon^2$, for $\varepsilon > \rho > 0$.

in which, ρ is the distance to the field center; c_1 and ε are constant larger than zero.

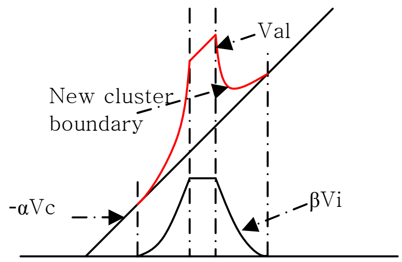




The distance from the sink to the centre of the physical signal field is D, D > R. For each node, the cost value Vc= c_2*d, in which d is the distance to the sink, and c_2 is a constant, $c_2>0$.



Even through the two fields are linearly combined, the non-linear property of the physical signal field causes the nodes (inside the physical signal field) to separate into two parts, as shown in the figure below.



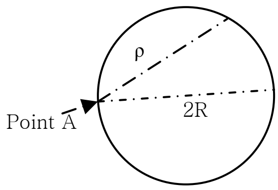
It's obvious that the maximum value of LSV (inside the physical signal field) should be on the straight line from sink to the field center. Along the line, let ρ still be the distance to the center, and then we have,

$$LSV = -\alpha\, c_2(D-\rho) + \beta\, c_1/\rho^2$$

The derivative of LSV is
$LSV' = \alpha c_2 - 2\beta c_1/\rho^3 = 0 \Rightarrow \rho^3 = 2\beta c_1/(\alpha c_2)$ …(A)

So, the new cluster boundary is at $\rho=(2\beta c_1/\alpha c_2)^{1/3}$. In fact, the new cluster boundary doesn't simply divide the physical signal field into two small parts. In the case that several adjacent physical fields exist, the new cluster boundary introduces a new clustering scheme which is shown in the figure above. (In CCS, the boundary of the physical signal field is also the boundary of node clustering.) In addition, the relative position of the convergent point changes according to the changes of the new boundary. From the figure above, the new boundary moves toward the center (or convergent point). This can also be described as the convergent point moves toward the new cluster boundary.

Next, let's investigate the cost of in-network processing. Two cases are considered: the convergent point is at the center of the physical signal field; and the convergent point is at point A (the point inside the physical signal field which is nearest the sink). To make it simple, we calculate the cost in the continuous situation (in which nodes are distributed among the region continuously; each node generates a message; and the energy cost of each message delivery equals the distance that the message travels). In addition, messages would not be aggregated along its way until they reach to convergent point.



When convergent point is at the field center

Cost at center= $\int_0^R \rho \cdot 2\pi\rho \, d\rho$ = $\pi R^3 \cdot 2/3$ …(B)

When the convergent point is at point A

Cost at point A = $\int_0^{2R} \rho \cdot 2\arccos[\rho/(2R)]\rho \, d\rho$

$=80R^3/9$ …(C)

which is about 4 times larger than the cost at the center. However, the distance from point A to the sink is smaller than the distance from the field center, and thus there is a trade-off on the convergent point selection.

By the equation $\rho^3 = 2\beta c_1/\alpha c_2$, and $\varepsilon \le \rho \le R$, we have the range of β/α, that is $c_2\varepsilon^3/(2c_1) \le \beta/\alpha \le c_2R^3/(2c_1)$. If $\beta/\alpha > c_2R^3/(2c_1)$, messages converge to the peak of the physical signal field, and the cluster boundary is the same as the field boundary (the same as CCS). If the physical signal field is isolated from the others. That is, when $\beta/\alpha < c_2\varepsilon^3/(2c_1)$, all messages of the physical signal field converge to "point A". When several physical signal fields are adjacent to each other and $\beta/\alpha < c_2\varepsilon^3/(2c_1)$), all those physical fields are welded into a larger field, and messages converge to "point A" of this new field. If several physical signal fields are adjacent to each other, and $c_2\varepsilon^3/(2c_1) \le \beta/\alpha \le c_2R^3/(2c_1)$, nodes will be clustered by the new boundary rather than by the natural boundary of the physical signal fields.

Based on the analysis above, the domain of β/α is given. However, we still need to be aware that there are many assumptions we have proposed for the analysis. Thus, in practice, the domain of β/α should only be regarded as a reference rather than a rule.

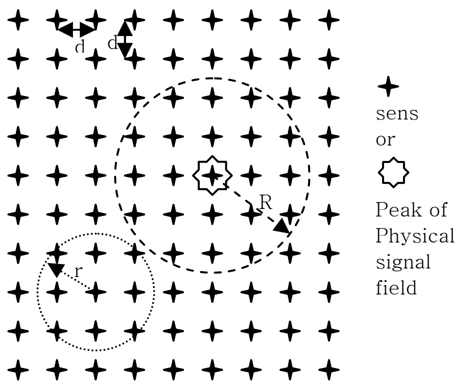


Fig. 1. Sensors on uniform square

6. Simulation

In comparison with other schemes, our scheme can cluster nodes with zero overhead. In addition, the routing paths within a cluster are constructed naturally. Furthermore, the scheme is robust even if some nodes may die, it can also adapt to the variations of the physical signal field.

Within the region, nodes are settled on a uniform square grid, with the grid spacing "d". The transmission radius of each node is "r". The physical signal field covers a round area with radius "R". The peak of the field is at the center. (Assume only one peak exists in the field.)

Of course, "r" should be larger than "d", otherwise no link exists except a bunch of isolated nodes. Let "E" be the energy cost of each hop transmission per message, and the data are only aggregated at the field peak. The entire energy consumption of in-network processing should be

$$= \sum_{i=1}^{\lfloor R/r \rfloor} \pi \, [(i \cdot r)^2 - (i-1)^2 \cdot r^2] \cdot 1/d^2 \cdot i \cdot E$$

$$+ \pi (R^2 - \lfloor R/r \rfloor^2 \cdot r^2) \cdot 1/d^2 \cdot \lceil R/r \rceil \cdot E$$

$$= \pi \cdot \frac{1}{d^2} E \cdot r^2 \lceil R/r \rceil \cdot \left\{ R^2/r^2 - \frac{1}{6}\lfloor R/r \rfloor - \frac{1}{3}\lfloor R/r \rfloor^2 \right\} \text{ …(D)}$$

in which $\lceil R/r \rceil$ denotes the ceiling of R/r; $\lfloor R/r \rfloor$ denotes the floor of R/r, and the node density is $1/d^2$.

From (D), the energy cost is in direct proportion to the node density. If R<r, the energy cost of in-network routing equals to $\pi ER^2/d^2$. When some nodes die and the node density decreases, d would increase (Anyway, the upper bound of d is r). If d is equal to r, then the energy cost is $\pi E\lceil R/r \rceil\{R^2/r^2-(1/6)\lfloor R/r \rfloor-(1/3) \lfloor R/r \rfloor^2\}$. In fact, the same results can be drawn when sensor nodes are scattered among the area randomly with a uniform distribution.

The simulation runs over a square area (400m X 400m). Sensor nodes are arranged on a uniform square grid. d can be 12m, 16m or 20m separately. Let transmission radius r of each node be 25m. The radius of the physical field R varies from 10m to 100m (the peak of the field is at the center). As shown in figure 2, three curves (corresponding to d=12; d=16; d=20) overlap together. Immediately, the overlap means that the entire cost divided by node density (cost*d2) is independent from d. In fact, the overlap also infers that the cost is in direct proportion to the node density (note: the same result we have drawn from our analysis).

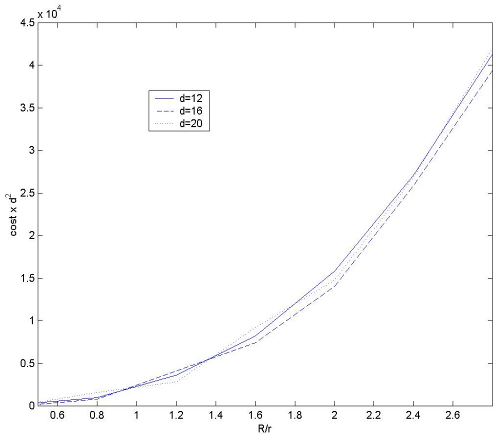


Fig. 2. In-network processing cost I

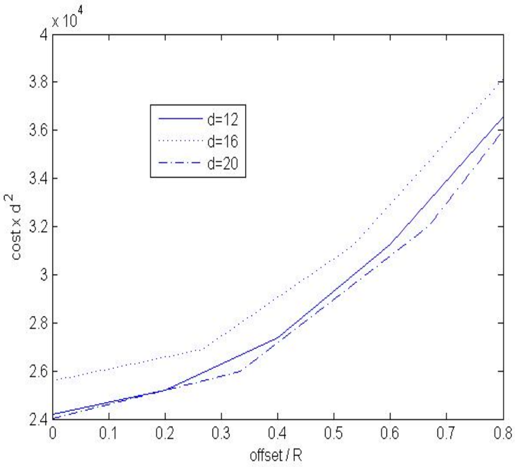


Fig. 4. Cost vs. offset

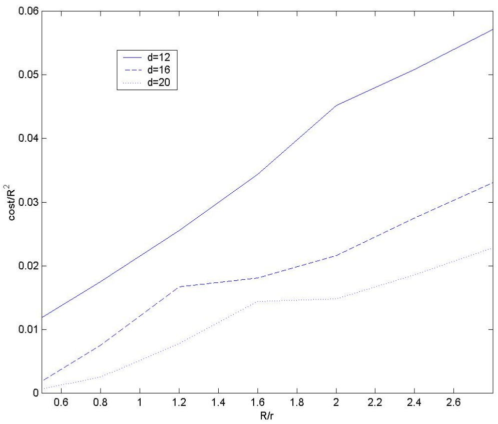


Fig. 3. In-network processing cost II

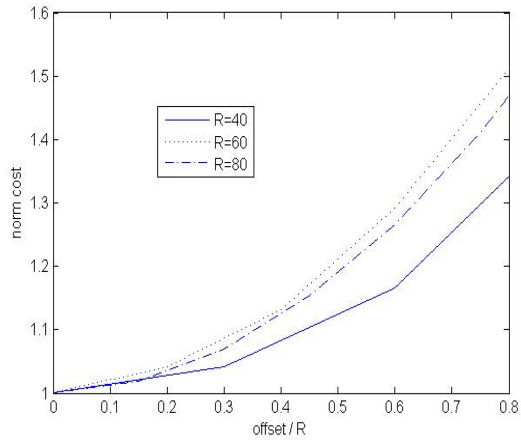


Fig. 5. Normalized cost vs. offset

The entire cost increases sharply when R/r is large. In general, R is determined by the physical event, thus, we can only change r to adjust energy cost. However, the variation of r also causes the variation of E (cost of each transmission per message), so there exist an optimal r to minimize the entire cost of in-network processing, corresponding to a particular R.

From figure 3, the average cost (cost/R^2) rises with the increment of R/r and the increment of node density ($1/d^2$). In fact, we can also draw same conclusions from the expression (D). As r is fixed, the total cost is a value with the order $O(R^3)$. Thus, the average cost (cost/R^2) should be with the order O(R), which is roughly in direct proportion to R/r. From expression (D), it is obvious that the average cost (cost/R^2) is in direct proportion to the node density ($1/d^2$).

To investigate the performance of linear combination scheme (LCS), let r = 25m, and R = 50m. In addition, let the convergent point move from the field center to the field boundary (offset increasing) to imitate the variation of β/α. As shown in figure 4, all 3 curves are close to each other (just as the case that they overlap in figure 2). Roughly, we can say: with the convergent point moving toward the field boundary (or offset increasing), the average cost of in-network processing (cost·d^2) increases.

In figure 5, d is 12m. The cost of in-network processing is normalized by the cost when offset is 0. From figure 5, the normalized cost is in direct proportion to the offset.

Anyway, based on the calculation result of the cost (expression B and C), normalized cost would always be less than 40/(3π) no matter how much R or offset is.

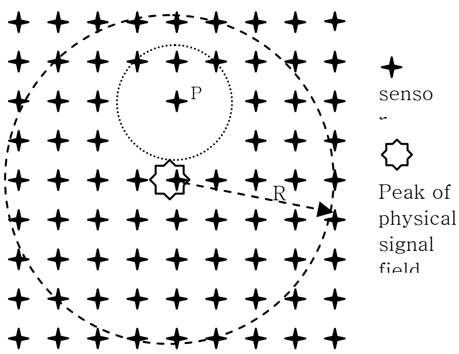


Fig. 6. Concave hole in WSN

When some nodes die, a hole may emerge in WSN, as shown in figure 6. For the hole with a convex boundary, our schemes could make packets detour the hole during in-network processing. Even for some holes with a concave boundary that can not be detoured, the cluster will break up into 2 or more new small clusters, and the data will converge to their local peaks. In figure 6, besides the peak of the physical signal field, sensor P will act as a new local peak. All nodes inside the field are split into two clusters.

In fact, concave holes are very common in WSN, and can prevent the cluster from being too big, even through the physical signal field (R) is big.

7. Future work

In conclusion, our schemes are robust and reliable. However, there are still two cases which, we need to discuss them further. The first is, when several neighbor nodes have the same intensity value of the physical signal field. In other words, the physical signal field is flat over the area of those nodes. In such case, we should introduce an idling scheme to let some of those nodes to sleep. The second case is that one cluster covers extra large area and too many sensor nodes are included, just as we analyzed before, the performance of the scheme may get worse (concave holes are not a definite solution).

In LCS, we adopt a cost-field routing scheme for each sensor forwarding data to the sink. Each sensor node has two scale values. One is the cost value to the sink, and the other is the intensity value of the physical signal field. Those two values can also be used as coordinates of each sensor. In this way, sensor nodes can be divided into several clusters according to their coordinates.

References

[1] John Heidemann, Fabio Silva, Chalermek Intanagonwiwat "Building Efficient Wireless Sensor Networks with Low-Level Naming" Proceedings of the 18th ACM symposium on Operating systems principles 2001

[2] R. Poor. "Gradient Routing in Ad Hoc Networks". http://www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf 2000

[3] F. Ye, G. Zhong, S. Lu, and L. Zhang. "GRAdient Broadcast: A Robust Data Delivery Protocol for Large Scale Sensor Networks". *ACM Wireless Networks (WINET),* 11(2), March 2005

[4] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, and Fabio Silva, "Directed Diffusion for Wireless Sensor Networking" IEEE/ACM TRANS. ON NETWORKING, Vol. 11, No. 1, Feb. 2003

[5] Bhaskar Krishnamachari, Deborah Estrin, Stephen Wicker "The Impact of Data Aggregation in Wireless Sensor Networks" Proceedings of the 22nd International Conference on Distributed Computing Systems, P575~578 , 2002 ISBN:0-7695-1588-6

[6] Bo Hong and Viktor K. Prasanna "Optimizing a Class of In-network Processing Applications in Networked Sensor Systems" The 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems

[7] Long Gan, Jiming Liu, and Xiaolong Jin "Agent-Based, Energy Efficient Routing in Sensor Networks" AAMAS'04, July 19-23, 2004, New York, New York, USA

[8] Fan Ye, Haiyun Luo, Jerry Cheng, Songwu Lu, Lixia Zhang "A Two-Tier Data Dissemination Model for Large-scale Wireless Sensor Networks" MOBICOM'02, September 23–28, 2002, Atlanta, Georgia, USA

[9] Mohamed A. Sharaf, Jonathan Beaver, Alexandros Labrinidis, Panos K. Chrysanthis "Balancing energy efficiency and quality of aggregate data in sensor networks" The VLDB Journal (2004) 13: 384–403 / Digital Object Identifier (DOI) 10.1007/s00778-004-0138-0

[10] Jonathan Beaver, Mohamed A. Sharaf,Alexandros Labrinidis, Panos K. Chrysanthis "Location-Aware Routing for Data Aggregation in Sensor Networks" Proceedings of Geo Sensor NetworksWorkshop

[11] Jonathan Beaver, Mohamed A. Sharaf, Alexandros Labrinidis, Panos K. Chrysanthis "Power-Aware In-Network Query Processing for Sensor Data" Proceedings of the 3rd ACM MobiDE Workshop

[12] IOANNIS CHATZIGIANNAKIS, SOTIRIS NIKOLETSEAS and PAUL SPIRAKIS "Efficient and Robust Protocols for Local Detection and Propagation in Smart Dust Networks" Mobile Networks and Applications 10, 133–149, 2005

[13] A. WADAA, S. OLARIU and L. WILSON "Training a Wireless Sensor Network" Mobile Networks and Applications, Feb.2005 Volume 10 Issue 1-2

[14] Qing Fang, Feng Zhao, Leonidas Guibas "Lightweight Sensing and Communication Protocols for Target Enumeration and Aggregation" MobiHoc '03, June 1-3, 2003, Annapolis, Maryland, USA. 2003

[15] Lionel M. Ni, Yanmin Zhu, Jian Ma "Semantic Sensor Net: An Extensible Framework" ICCNMC 2005: 1144-1153

[16] Radu Stoleru, Tian He, John A. Stankovic, David Luebke "A High-Accuracy, Low-Cost Localization System for Wireless Sensor Networks" SenSys'05, November 2-4, 2005, San Diego, CA, USA

[17] Jeong-Hun Shin, Jaesub Kim, Keuntae Park, Daeyeon Park "Railroad: Virtual Infrastructure for Data Dissemination in Wireless Sensor Networks" PEWASUN'05, October 10-13, 2005, Montreal, Quebec, Canada

[18] Ratnabali Biswas, Kaushik Chowdhury, Dharma P. Agrawal "Attribute Allocation in Large Scale Sensor Networks" DMSN'05, August 29, 2005, Trondheim, Norway

[19] Divyasheel Sharma, Vladimir I. Zadorozhny, Panos K. Chrysanthis "Timely Data Delivery in Sensor Networks using Whirlpool" DMSN'05, August 29, 2005, Trondheim, Norway

Authors

Niki Pissinou

Dr. Niki Pissinou received her Ph.D. in Computer Science from the University of Southern California, her M.Sc. in Computer Science from the University of California at Riverside, and her B.S.I.S.E. in Industrial and Systems Engineering from The Ohio State University. She is currently a tenured professor and the director of the Telecommunication & Information Technology Institute at the Florida International University. Dr. Pissinou has published over hundred refereed publications and has received best paper awards. She has also co-edited eight volumes. Dr. Pissinou has also served as a steering committee, general chair and program committee member of over hundred program and organizational committees of IEEE and ACM sponsored technical conferences.

S. Kami Makki

Dr. S. Kami Makki has received a M.S. degree in Computer Science and Engineering from the University of New South Wales, and Ph D. degree in Computer Science from The University of Queensland. Before joining the University of Toledo, he had a number of academic positions and research appointments, and also worked in public and private industries for a number of years. Currently, he is an assistant professor in Department of Electrical Engineering and Computer Science of University of Toledo. His research area includes Distributed Systems, Web and Multimedia Databases, Intelligent Web Applications, E-commerce and Electronic Services, Mobile and Wireless Network & Security.

Xiwei Zhao

Received a B.S. degree in Mechanical Engineering from Zhejiang University, China, 1991, and M.S. degree in the Department of Automation from Tsinghua University, China, 2001. Currently, he is a Ph D. candidate in Department of Electrical & computer Engineering, Florida International University. His research direction is Wireless Networking and Forensics.