Article

Expected Area-Based Real-Time Routing Protocol for Supporting Mobile Sinks in Wireless Sensor Networks

Youngju Nam [1], Hyunseok Choi [1], Yongje Shin [1], Soochang Park [2] and Euisin Lee [1,*]

1   School of Information and Communication Engineering, Chungbuk National University, Cheongju 28644, Korea
2   Department of Computer Engineering, Chungbuk National University, Cheongju 28644, Korea
*   Correspondence: eslee@cbnu.ac.kr

Abstract: To deliver real-time data within a desired time deadline, a spatiotemporal approach has been proposed in wireless sensor networks. In the approach, real-time data can be delivered with a delivery speed calculated by both the distance from a source to a static sink and the desired time deadline. In the case of a mobile sink, because the distance from a source to the sink would be dynamically changed due to its movement, real-time data cannot be delivered due to the inability to calculate a delivery speed. Thus, we propose a novel real-time routing protocol, called EAR2 (expected area-based real-time routing) for mobile sinks. Instead of the immediate distance, EAR2 considers the static distance to the expected area of a mobile sink's location using its movement speed. To satisfy the desired time deadline, EAR2 guarantees that the total summation of the unicasting time to the expected area and the flooding time within the expected area can be smaller than the deadline. To do this, EAR2 calculates a data delivery speed by using the static distance and the unicasting time and exploits the flooding time observed from various network environments. Because EAR2 has a high flooding time and data loss problems due to a single flooding point, we propose a protocol called EAR2M with multiple flooding points, which reduces the flooding time and enhances the reliability of data transmission. We also propose two extensions of EAR2M, called EAR2M_R and EAR2M_E for efficient data transmission from a source to multiple flooding points. They aim to reduce the transmission delay and the energy consumption for efficient data transmission among multiple flooding points, respectively. Simulation results show that EAR2 improves the deadline miss ratio and the energy consumption, with averages improvements of 47.17% and 29.99% over the existing schemes, respectively. Furthermore, EAR2M with four flooding points enhances the deadline miss ratio and the energy consumption by an average of 12.69% and 131.86% over EAR2.

Keywords: wireless sensor networks; mobile sinks; real-time routing

1. Introduction

Recent advances in VLSI, microprocessor and wireless communication technologies, and wireless sensor networks (WSNs) have been deploying for various potential applications in environmental surveillance, military operations, object tracking, and other domains [1–3]. In WSNs, real-time applications such as fire alert and intruder detection require data collection in a desired time deadline [4]. In order to satisfy the desired time deadline, the real-time routing schemes proposed in [4–6] utilize a spatiotemporal approach by which data are delivered with a delivery speed obtained from both the distance from a source to a destination and the desired time deadline. When the schemes disseminate data from the source as far as the destination, every immediate node elects a next-hop node that is one of the one-hop neighbor nodes closer to the destination than itself and has a relay speed larger than the delivery speed. Hence, by relaying data with a faster speed than the delivery speed per every hop, data collection in the desired time deadline with respect to the distance can be achieved [5].

Typically, the existing schemes [4–6] assume the distance between the source and the destination is static, i.e., a constant value. In many practical applications, mobile sinks such as soldiers and firefighters tend to move around within the sensor fields and receive data during their movements [7–9]. Accordingly, when a source disseminates data to a mobile sink in a desired time deadline, the distance between the source and the mobile sink may also be dynamically altered because the mobile sink randomly changes its own locations in a sensor field. This means that the delivery speed should be recalculated by the immediate reflection of the distance alteration during real-time data dissemination to the mobile sink.

In order to immediately reflect the new distances, we may take into account three methods: (1) the source-based, (2) the intermediate-node-based, and (3) the sink-based method. First, in the source-based method, the source obtains the current location of the mobile sink by sink location services [10–12] and then recalculates the new delivery speed to retransmit data to the current location on the desired time deadline. In the second method, an intermediate node currently forwarding data performs the location service to get the current location of the sink. Then, it disseminates the data to the current location with the recalculated new delivery speed. Finally, the sink-based method employs a backward location notification to the intermediate node from the sink when the sink moves to a new location [9]; thus, the intermediate node reflects the new distance for the new delivery speed.

However, these three immediate reflection methods might have considerable problems with respect to real-time data transmission. The current methods of location obtainment used by the sink location services may lead to long term delays, shortening the time interval until the desired time deadline. Because the mobile sink frequently alters its own location, this represents a serious problem. The backward location notification means that the reflection of new distances is away from the destination; thus, it also causes high latency. In addition, in existing schemes, because immediate nodes select the nondeterministic next-hop node, the forwarding path for data transmission from the source to the destination might be altered every instance of transmission. Namely, the sink cannot determine the currently forwarding intermediate node to reflect the new distances. Consequently, in terms of real-time routing to the mobile sink, it is hard to immediately reflect new distances due to high latency and nondeterministic forwarding.

Thus, we proposed a real-time routing protocol, called EAR2 (expected area-based real-time routing) [13] for mobile sinks in wireless sensor networks. To reflect a static distance instead of the immediate distance, EAR2 considers the expected area where a mobile sink will be located by using its movement speed. To deliver real-time data within the desired time deadline, EAR2 guarantees that the total summation of the unicasting time to the expected area and the flooding time within the expected area is smaller than the deadline. To do this, EAR2 calculates a delivery speed using the static distance to the expected area and the unicasting time and observes the flooding time from various network environments. In EAR2, the source transmits data to the expected area with the delivery speed, and the first sensor node in the area then floods the area to deliver the data to the mobile sink that is dynamically located in the area. However, since EAR2 uses a single flooding point, it has a high flooding time and data loss problems [14,15].

Therefore, to solve the problems of EAR2, we propose a protocol called EAR2M that extends EAR2 by exploiting multiple flooding points. By using multiple flooding points, EAR2M reduces the flooding time and enhances the reliability of data transmission. Because EAR2M provides lower flooding times than EAR2, it demands less data delivery speeds and thus achieves lower deadline miss ratios compared with EAR2. The efficiency of the transmission delay and energy consumption are two important challenging issues in wireless sensor networks [16,17] Thus, we also propose two extensions of EAR2M, called EAR2M_R and EAR2M_E for efficient data transmission from a source to multiple flooding points. EAR2M_R and EAR2M_E aim to reduce the transmission delay and the energy consumption for data transmission among multiple flooding points, respectively. Although both EAR2M_R and EAR2M_E have better performance than EAR2M, they have a trade-off

between the transmission delay and the energy consumption. We conducted simulations in various network environments through the network simulator NS2 to evaluate the network performance of the proposed protocols. Simulation results conducted in various environments verified that EAR2 significantly improves the network performance when compared with the existing protocols SPEED [4] and THVR [6] in terms of deadline miss ratio and energy consumption. Simulation results showed that EAR2 enhances the deadline miss ratio and the energy efficiency by 57.94% and 39.80% over SPEED, respectively, and by 36.40% and 20.18% over THVR, respectively. Additionally, the simulation results showed that EAR2M has a better performance than EAR2, and EAR2M with four flooding points improves the deadline miss ratio and the energy consumption by average values of 12.69% and 131.86% over EAR2. We have also demonstrated that EAR2M_R and EAR2M_E have a trade-off between the deadline miss ratio and energy consumption through simulation results.

The remainder of this paper is organized as follows. First, we provide the literature review on real-time routing in Section 2. The network model and the overview of our expected area-based real-time routing (EAR2) protocol are presented in Section 3. Details of EAR2 and EAR2M are described in Sections 4 and 5, respectively. The extensions, EAR2M_R and EAR2M_E of EAR2M are described in Section 6. The simulation results are provided in Section 7 to evaluate the performance of EAR2, EAR2M, EAR2M_R, and EAR2M_E. The paper is concluded in Section 8.

2. Related Work

In wireless sensor networks, several routing protocols [4–6,18–23] have been proposed for providing timeliness in real-time applications. The protocols based on spatiotemporal approaches improve the timeliness. These approaches consider the information on the end-to-end distance (spatial) and transmission delay time (temporal) of relay nodes to make the routing decision. This chapter briefly reviews these real-time routing protocols based on spatiotemporal approaches.

SPEED [4] is the first developed spatiotemporal real-time routing protocol in wireless sensor networks. SPEED is a stateless and localized routing protocol providing a soft real-time communication service by a desired uniform delivery speed across the network under the assumption that the end-to-end delay is proportional to the distance between the source and the destination. It exploits only the local information, i.e., the location and delay information about the neighbor nodes, and not the global topology. Stateless nondeterministic geographic forwarding (SNGF) is a routing module responsible for selecting an appropriate forwarding node in the neighborhood table that meets the desired delivery speed, which in turn balances the load in each flow between multiple concurrent routes. Delay estimation modules are intended to estimate the single-hop delay that helps in making routing decisions. A backpressure re-routing algorithm helps in re-routing packets around large-delay links (i.e., congestion) with minimum control overhead.

To achieve energy efficiency, POWER-SPEED [19] does not use control packets to select the next-hop neighbor according to the destination based on the downstream QoS condition estimation based on the spatiotemporal historic data. Then, POWER-SPEED adjusts the power level of the wireless transmitter to a minimum value to send packets to the next-hop neighbor.

As another protocol to achieve energy efficiency, an energy-aware real-time routing protocol named EA-SPEED [21] uses the residual energy information as well as the single-hop delay and the location information in routing decisions to select the next-hop neighbor to a destination. Then, the weight function of neighbor nodes for the routing decision module is formed based on the information of these three parameters.

For supporting fault tolerance, another extension of SPEED named FT-SPEED [23] is proposed to solve the void problem of SPEED by using a void announce scheme and a void bypass scheme. The void announce scheme prevents packets reaching the void through

other routing paths, while the void bypass scheme forwards packets around two sides of the void.

MMSPEED [5] is an extension of SPEED that provides service differentiation and a probabilistic QoS guarantee. For delivery timeliness, multiple network-wide packet delivery speed options are provided for different traffic types according to their end-to-end deadlines. In supporting service reliability, probabilistic multipath forwarding is used to control the number of forwarding paths based on the required end-to-end arrival probability. In order to provide reliability, MMSPEED exploits the principle of redundancy, i.e., the more paths that a packet is delivered via, the higher the probability of reaching its final destination.

As an extension of MMSPEED for enhancing energy efficiency, Improved MMSPEED (IMMSPEED) [18] uses a multi-path and multi-speed approach, similar to MMSPEED. However, IMMSPEED selects neighbor nodes with higher residual energies as next-hop nodes to a destination. This is different from MMSPEED, which selects neighbor nodes randomly.

THVR [6] and OMLRP [22] attempt to increase the performance of real-time communication with multi-hop network information. Using multi-hop lookahead, sensor nodes gather the information of network topology and delay. In these protocols, a routing decision is made based on the multi-hop (more than two) velocity. The forwarding metric in these protocols is similar to that of SPEED, where the packet deadline is mapped to the velocity requirement. When a node has data to forward, the node searches for the node that has the highest estimator module that calculates the packet delay from the sender to its potential forwarder. The protocols work well in enhancing real-time routing and outperform SPEED in terms of time deadline success ratio.

To deal with path failures and sleep nodes, a real-time routing protocol [20] has been proposed by applying neural networks (NN) to SPEED. Through NNs, the protocol evaluates QoS parameters and selects the optimized path to a destination based on the QoS parameters.

However, the above-mentioned protocols focus only on real-time transmission toward a fixed destination (i.e., the sink in wireless sensor networks) which never moves during the network operation time. However, if the sink moves, the protocols cannot support real-time data dissemination. When a source disseminates data to a mobile sink in a desired time deadline, the mobile sink randomly changes its own location in a sensor field. As a result, the distance between the source and the mobile sink may be also dynamically altered. This means that the delivery speed should be recalculated by the immediate reflection of the distance alteration during real-time data dissemination to the mobile sink.

EAR2 was proposed to overcome the limitation of the previous real-time protocols and support mobile sinks. EAR2 considers the expected location of a mobile sink based on its movement speed. EAR2 guarantees that the total summation of the unicasting time to the expected area and the flooding time within the expected area is smaller than the desired time deadline. However, EAR2 has a high flooding time and data loss problems because it uses a single flooding point. In this paper, EAR2M is proposed to reduce the flooding time and enhance the reliability of data transmission by using multiple flooding points. In another approach, to reduce the flooding time and energy consumption of EAR2, a grid structure-based real-time protocol [24] was proposed with both unicasting, multicasting, and one-hop forwarding. This protocol considers the expected grids of a mobile sink as well as its expected area and uses unicasting from a source node to the expected area, multicasting from the expected area to the expected grids, and one-hop forwarding from the expected grid to where the mobile sink is located in the expected grid. However, this protocol has a high energy consumption overhead for the construction and management of a grid structure for the whole network.

3. Network Model and Protocol Overview

3.1. Network Model

We considered a wireless sensor network that consists of randomly and roughly deployed sensor nodes over a finite, two-dimensional planar region. We assume in this work that each node can obtain its own location information by GPS [25] or localization schemes [26,27]. By periodically exchanging beacon messages [28,29], every node can determine its one-hop neighbor nodes, and obtain their locations and data relay speeds. When a source detects an event, it generates data with the desired time deadline $T_{setdeadline}$ of the event. The desired time deadline of an event depends on the event type. In order to deliver data from a source to a static sink within a desired time deadline $T_{setdeadline}$, the existing protocols [4–6] exploit a data delivery speed S_{speed} calculated by the deadline and the distance *d(source, sink)* between the source and the sink as follows:

$$S_{speed} = \frac{d(source, sink)}{T_{setdeadline}}.$$ (1)

Thus, when the protocols disseminate data from the source to the sink, every immediate node elects a next-hop node that is one of the one-hop neighbor nodes closer to the sink than itself with a relay speed larger than the data delivery speed [4,5]. However, if the sink moves, the distance *d(source, sink)* between the source and the mobile sink changes (i.e., increases or decreases) due to the movement of the mobile sink, so that the data delivery speed S_{speed} is continuously altered due to continuous distance changes. Unfortunately, we cannot also determine the static location of a mobile sink due to its continuous movement. Thus, as the first challenging issue, the static location of a mobile sink must be decided in order to obtain the distance *d(source, sink)* and as a result, to calculate the data delivery speed S_{speed} by using the distance. To do this, we consider the expected area of a mobile sink with its moving direction and speed information and determine its virtual static location from the expected area. We address our method for solving the first challenging issue in Section 4.1. However, since the virtual static location of the mobile sink is just a location within its expected area, it is not its real location. Thus, as the second challenging issue, it needs to deliver data packets to the real location of the mobile sink in real-time. To do this, we consider flooding-based data forwarding and flooding time within the expected area of the mobile sink in order to support real-time data delivery to the mobile sink. We address our method for solving the second challenging issue in Section 4.2.

3.2. Protocol Overview

We propose an expected area-based real-time routing (EAR2) protocol for supporting the real-time routing of data with a desired time deadline $T_{setdeadline}$ from a source node to a mobile sink as shown in Figure 1. EAR2 calculates the expected area (EA) of the mobile sink based on its location, speed, and direction information. Then, it provides real-time data routing from the source node to the mobile sink by guaranteeing that $T_{setdeadline}$ is smaller than the total time of the forwarding time $T_{forwarding}$ from the source node to the single flooding point *P* nearest to the *EA* and the flooding time $T_{flooding}$ within the *EA*. First, we present a method for calculating the expected area (EA) of a mobile sink in Section 4.1. We next present a method for calculating the $T_{flooding}$ and $T_{forwarding}$ in Section 4.2.
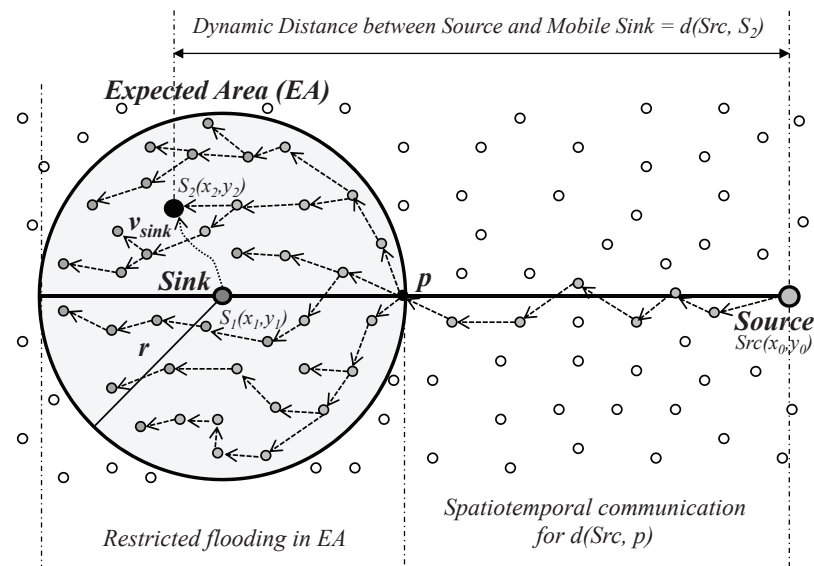
Figure 1. Real-time data routing based on expected area of a mobile sink.

4. Expected Area-Based Real-Time Routing (EAR2)

4.1. Expected Area of a Mobile Sink

For data dissemination to a sink, a source that detects an event needs to know the location of the sink through sink location services. Because we consider a mobile sink that randomly changes its own location in a sensor field, the obtained location cannot be exactly matched with the current location of the sink [10]. To deliver data to the sink within the desired time deadline $T_{setdeadline}$, we expect that the mobile sink will move around during a space of time in a circular area, i.e., the expected area *(EA)*. *EA* is able to be calculated by the obtained location and the velocity of the sink as in previous works [30–32]. The center point of *EA* is the initial obtained location of the sink. Let t_0 be the time when the sink locates the initial location, t_1 be the time when the source obtains the location of the sink, t_2 be the time when the source generates its data to deliver to the mobile sink, and t_3 be the time when the data from the source is received by the sink. The $T_{setdeadline}$ is $t_3 - t_2$. The maximum moving distance r of the mobile sink, i.e., the radius of *EA* with the sink speed V_{sink} is calculated as follows according to the method in [31]:

$$r = (t_3 - t_0) \times V_{sink}. \tag{2}$$

As shown in Figure 1, because r is the maximum distance that the sink can move in a straight line during $t_3 - t_0$, the sink can be located at any position in *EA* at t_3 due to its unrestricted movement, as mentioned in the paper [32].

4.2. Calculation of Data Delivery Speed

In EAR2, data dissemination is divided into two processes: (1) forwarding the data from the source to one of the point of the EA by unicasting and (2) flooding the data into *EA*. The data transmission time, therefore, is the sum of $T_{forwarding}$ as the time for the forwarding process and $T_{flooding}$ as the time for the flooding process. When the $T_{setdeadline}$ is defined as the desired time deadline, to support real-time service, the following condition has to be satisfied according to the same rule mentioned in papers [4,5].

$$T_{setdeadline} \geq T_{forwarding} + T_{flooding} \tag{3}$$

To satisfy Equation (3), the source sets a data delivery speed S_{speed} for the forwarding process, i.e., data transmission from itself to a flooding point *p1* as the nearest point of *EA*. S_{speed} is calculated considering $T_{flooding}$ as follows:

$$S_{speed} = \frac{d(src, p1)}{T_{setdeadline} - T_{flooding}},$$

(4)

where the *d(src, p1)* is the distance from the source to *p1*. $T_{flooding}$ is the expected time that all nodes in *EA* receive the data through flooding. However, the source cannot determine the $T_{flooding}$ within the *EA*. Even though the $T_{flooding}$ can be measured, the measured $T_{flooding}$ varies vastly according to time and environment.

Therefore, for $T_{flooding}$, we exploit the observed flooding time for a circular area with a radius *r* through simulation by using the NS-2.30 network simulator [33,34]. Figure 2 shows the simulation results for the observed flooding time using a single flooding point. We extract the simulation results through numerous fulfillments in different environments (need density, initial flooding node, data packet size, etc.). The graph in Figure 2 shows two lines of a single point for the radius *r*. The above lines and below lines show the worst and best values for flooding time in the circle region with radius *r*, respectively. In other words, the flooding time within the circle region with radius *r* has a value between the worst and best values in a variety of different environments. We use the worst value as the $T_{flooding}$ because the real flooding time may not have any value bigger than the worst value.
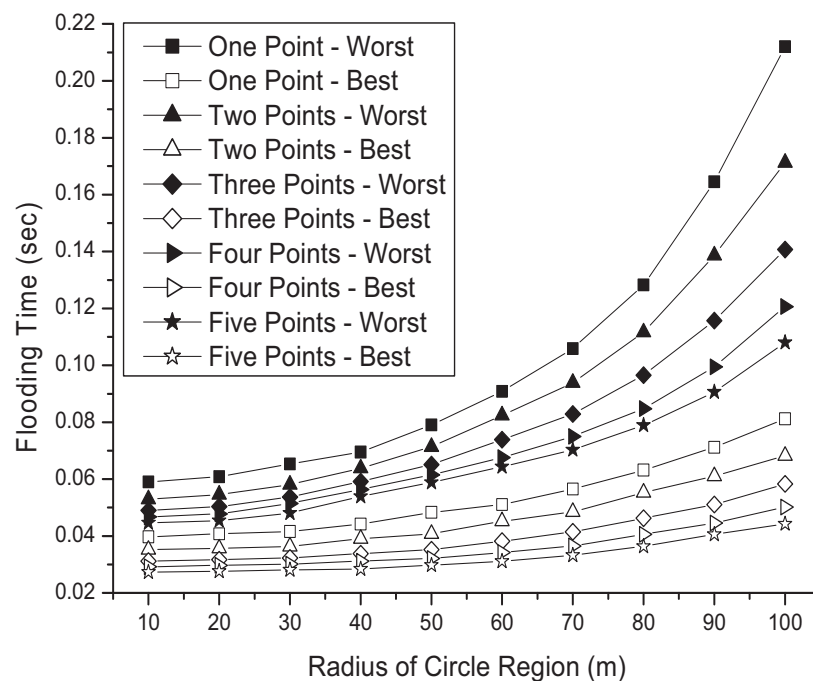
Figure 2. Flooding time for circle region with radius *r*.

Through programming in the module of sensor nodes before their deploying, we enable them to know the flooding time $T_{flooding}$ within the circular region with radius *r*. All sources use the $T_{flooding}$ to deliver their own data to a mobile sink within a desired time deadline $T_{setdeadline}$ in EAR2. After calculating S_{speed} using Equation (4), the source sends the data containing information of the sink ID, S_{speed}, and *EA* location (radius *r* and sink location) to the point *p1* of *EA*. To support real-time service, the data have to be delivered faster than S_{speed}. Every relay node, including the source, elects one of the neighbor nodes that is closer to the point *p1* than itself and has a relay speed larger than S_{speed} as the next-hop node. Among nodes around the point *p1*, the node located in the *EA* first receives the data, stops forwarding the data, and then performs restricted flooding within the *EA*.

Because S_{speed} is calculated considering the worst case, the mobile sink located in the *EA* can receive the data within the desired time deadline.

However, the single-point flooding has several weaknesses. First, as shown in Figure 2, if the radius r is large, the worst value of the $T_{flooding}$ is very large. Accordingly, by Equation (3), the $T_{forwarding}$ is very small and as a result, S_{speed} might be very great. Since such great S_{speed} may not be supported by sensor nodes, real-time transmission to a mobile sink is difficult to provide. Second, if the sink moves toward *p2*, which is the farthest point from the source, the time to deliver the data by flooding increases so the mobile sink may not receive the data within the desired time deadline. Third, because the data forwarding path to a single flooding point is a single path, if data loss on the path happens, then the mobile sink cannot receive data [35,36]. Therefore, in the next section (Section 5), we exploit multiple flooding points in EAR2 to overcome the weaknesses of single-point flooding.

5. EAR2 with Multiple Flooding Points (EAR2M)

To solve the aforementioned three limitations of a single flooding point, we propose a new protocol called EAR2M that exploits multiple flooding points in EAR2 as shown in Figure 3. Flooding consumes much of the energy of the sensor nodes. Within *EA*, multiple-point flooding consumes the same energy as single = point flooding. Multiple-point flooding distributively floods at only around multiple points [37,38], but otherwise, multiple-point flooding allows all nodes in the *EA* to receive the flooding data identically to single-point flooding. However, exploiting multiple-point flooding in EAR2M consumes energy when the source forwards its data to the multiple points by unicast. By consuming the data forwarding energy to multiple points, we reduce the flooding time $T_{flooding}$ and enhance the reliability of data forwarding to the mobile sink in EAR2M.
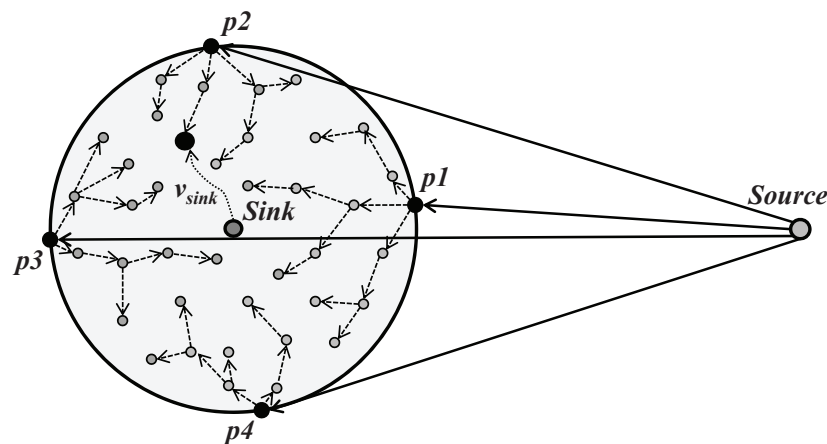

Figure 3. Real-time data routing based on distributed flooding at multiple flooding points.

We also carried out a simulation to measure how the distributed flooding at multiple points influences $T_{flooding}$. As shown in Figure 2, multiple-point flooding can reduce the flooding time $T_{flooding}$ because flooding is distributively performed at multiple points. Figure 2 shows two lines for each flooding point number. The above lines and the below lines show the worst and best values for flooding time in the circle region with radius r, respectively. If the radius of the circular region is large, then when the number of flooding points increases, the worst and best values of $T_{flooding}$ are smaller. On the other hand, if the radius of the circular region is small, the worst and best values of the $T_{flooding}$ are largely unaffected by the number of flooding points.

In EAR2M, to provide real-time data routing using multiple flooding points, we recalculate the data forwarding time $T_{forwarding}$ and the data delivery speed S_{speed} for each flooding point in Equations (3) and (4) through the flooding time $T_{flooding}$ when using multiple flooding points. We use the observed flooding time in Figure 2 as the flooding

time $T_{flooding\_n}$ when exploiting n flooding points. Thus, Equation (3) is recalculated as follows:

$$T_{setdeadline} \geq T_{forwarding} + T_{flooding\_n}. \tag{5}$$

Furthermore, the data delivery speed S_{speed_i} to each flooding point *i* is calculated as follows:

$$S_{speed\_i} = \frac{d(src, p_i)}{T_{setdeadline} - T_{flooding\_i}}. \tag{6}$$

If the source node selects the number and position of flooding points and calculates the data delivery speed S_{speed_i} to each flooding point *i*, it sends its data with the S_{speed_i} information to each flooding point *i* through unicast. If the data arrive at each flooding point, the node the nearest to the point in the *EA* floods the data within the *EA*. All nodes in *EA* receive and flood the data only once regardless of the initial flooding nodes. If any node receives redundant data, it drops the data. Accordingly, the mobile sink in *EA* can receive the data from the source node. Figure 3 shows that the source node forwards its data to four flooding points and the nodes at the flooding points distributively flood the data within *EA*. The number of flooding points greatly influences the performance of EAR2M. Thus, we compare the performance of EAR2M for the number of flooding points in terms of the energy consumption and the deadline success ratio.

6. The Extensions of EAR2M

As described in the previous chapters, EAR2M based on multiple flooding points improves the performance of real-time data dissemination by reducing the flooding time. However, multiple data transmissions from a source to multiple flooding points wastes a significant amount of energy. So, efficient data transmission from the source to the multiple flooding points could reduce the energy consumption. In order to achieve efficient data transmission, the source node transmits its data to one of the flooding points (the closest) and the node on the flooding point relays the data to the other flooding points for data flooding. In this paper, we have two approaches for a flooding point of an *EA* to relay toward the other flooding points as the extension of EAR2M. The first approach, EAR2M_R, as shown in Figure 4a. This approach minimizes the transmission delay from the selected flooding point to the other flooding points for increasing the real-time deadline success ratio. Figure 4b indicates the second approach, EAR2M_E. The second approach aims to minimize the energy consumption of data relay between flooding points. In both EAR2M_R and EAR2M_E, to provide real-time data routing using multiple flooding points, we recalculate the data forwarding time $T_{forwarding}$ and the data delivery speed S_{speed} to each flooding point in Equations (3) and (4). In the next paragraph, the method for recalculation of the data forwarding time $T_{forwarding}$ and the data delivery speed S_{speed} in EAR2M_R and EAR2M_E is described in detail.

To provide real-time data routing using multiple flooding points, EAR2M_R also recalculates the data forwarding time $T_{forwarding}$ and the data delivery speed S_{speed} for each flooding point in Equations (3) and (4) through the flooding time $T_{flooding}$ when using multiple flooding points. Equally, EAR2M_R also uses the observed flooding time in Figure 2 as the flooding time $T_{flooding\_n}$ when exploiting n flooding points. Thus, Equation (3) is equally calculated as Equation (5) of EAR2M as follows:

$$T_{setdeadline} \geq T_{forwarding} + T_{flooding\_n}. \tag{7}$$

However, the data delivery speed S_{speed_i} to each flooding point *i* should be calculated differently than in Equation (6) of EAR2M.

(**a**)

(**b**)
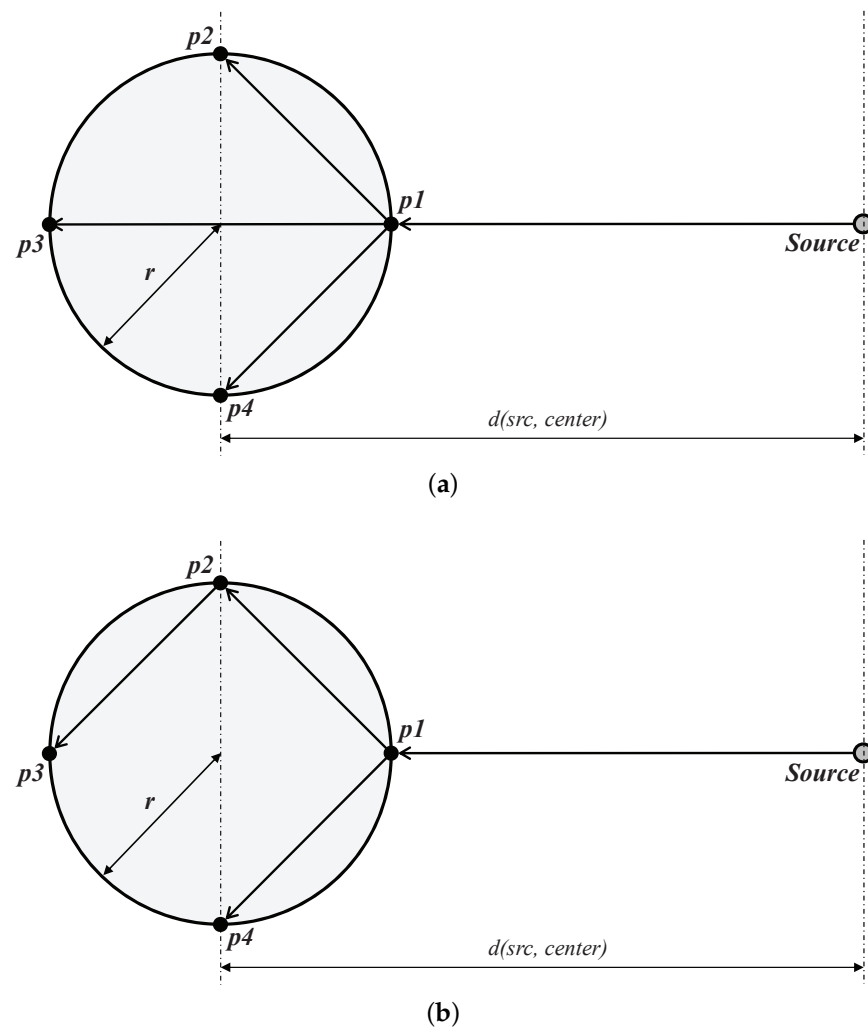
Figure 4. The extended schemes: (**a**) EAR2M_R and (**b**) EAR2M_E.

Because EAR2M delivers the data from a source to each of the multiple flooding points as shown in Figure 3, it individually calculates the data delivery speed S_{speed_i} to each flooding point i. However, in the pursuit of energy efficiency, EAR2M_R forces the data of a source to be delivered from the nearest flooding point (p1) to the other flooding points (p2, p3, and p4), as shown in Figure 4a.

Thus, different from EAR2M, EAR2M_R synthetically calculates the data delivery speed to multiple flooding points. In order to calculate the desired delivery speed, EAR2M_R considers the distance from a source to the farthest flooding point via the closest flooding point. The reason to consider the distance is that the distance to the other flooding points is shorter than the previous one. Namely, the modified delivery speed for the farthest flooding point could be applied to the other flooding points using Equation (7).

Thus, EAR2M_R calculates the data delivery speed S_{speed_fp} to the farthest flooding point fp via the nearest flooding point $p1$ as follows:

$$S_{speed\_fp} = \frac{d(src, p1) + d(p1, fp)}{T_{setdeadline} - T_{flooding\_n}}. \tag{8}$$

In EAR2M_R, if the source node selects the number and position of flooding points, it calculates the data delivery speed S_{speed_fp} for them using Equations (7) and (8). Then, it sends its data with the information of the S_{speed_fp} and the flooding point locations to the closest flooding point (p1 in Figure 4a) to it. Next, the sensor node nearest to the closest flooding point and belongs to the *EA* receives the data from the source. Then, the sensor

nodes send the data of the source with the information of the S_{speed_fp} to each of the other flooding points (p2, p3, and p4 in Figure 4a) by unicasting. Similarly, the sensor node the nearest to each flooding point and belongs to the *EA* receives the data from the sources. Thus, the sensor nodes flood the received data of the source within the *EA* using all flooding points. Furthermore, all sensor nodes in *EA* receive and flood the data only once regardless of the initial flooding nodes. As a result, the mobile sink in *EA* can receive the data from the source node.

Figure 3 shows that the source node forwards its data to four flooding points, and the nodes at the flooding points concurrently flood the data within the *EA*. The number of flooding points greatly influences the performance of EAR2M. Thus, we compare the performance of EAR2M for the number of flooding points in terms of the energy consumption and the deadline success ratio.

In order to reduce the energy consumption of data relaying between flooding points, the second approach, EAR2M_E allows the flooding points to relay the data packet to another flooding point. As described above, EAR2M delivers data packets from a source to flooding points for flooding the packet into the *EA*. The nodes on the flooding point share the data packet and start the data flooding. In the EAR2M_E, in order to minimize the energy cost for sharing the data packet among the flooding point, the adjacent nodes on the flooding points share the data packet directly, as shown in Figure 4b. The direct communication between the adjacent flooding points could reduce the cost for data sharing more greatly than EAR2M_R. Although the farthest distance of the data dissemination is longer, the summation of the data dissemination distance could be shorter and EAR2M_E could reduce the energy consumption. As with EAR2M_R, EAR2M_E synthetically calculates the data delivery speed to multiple flooding points. EAR2M_E calculates the data delivery speed S_{speed_fp} to the farthest flooding point *fp* via multiple flooding points as follows:

$$S_{speed\_fp} = \frac{d(src, p1) + \sqrt{2}d(p1, fp)}{T_{setdeadline} - T_{flooding\_n}}. \tag{9}$$

After EAR2M_E determines the locations and number of flooding points, it calculates the data delivery speed S_{speed_fp} using Equation (9). The source node sends its data toward the closest flooding point. The node on the closest flooding point receives the data packet and relays it toward the other flooding points except the farthest flooding point. For data transmission toward the farthest flooding point, the node on one of the other flooding points relays the data packet. The total distance by the unicasting in EAR2M_R and EAR2M_E is calculated as followings:

$$
\begin{aligned}
TD\_R &= (d(src, center) - r) + 2\sqrt{2}r + 2r \\
&= d(src, center) + (2\sqrt{2} + 1)r, \\
TD\_E &= (d(src, center) - r) + 3\sqrt{2}r \\
&= d(src, center) + (3\sqrt{2} - 1)r,
\end{aligned}
\tag{10}
$$

where $d(src, center)$ is the distance between the source node and the center point of the expected area and *r* is the radius of the expected area. According to the above equations, the farthest unicasting distance of EAR2M_R is less than that of EAR2M_E and the summation of the unicasting distance of EAR2M_R is greater than that of EAR2M_E. In our opinion, we expect that EAR2M_R and EAR2M_E have a trade-off between the transmission delay and the energy consumption, as mentioned in several relevant papers [16,17]. Thus, we prove the trade-off through the experimental results in Section 7.4.

7. Experimental Results

We simulate and analyze the proposed scheme by implementing it in the NS-2.30 network simulator [34]. In this section, we compare the proposed scheme with two other schemes: (1) SPEED [4], which is the most popular real-time protocol based on the relay speed information of one-hop neighbor nodes, and (2) THVR [6], which is another real-time protocol that uses the information of two-hop neighbor nodes. However, since the two schemes do not support sink mobility, a footprint chaining technique [9] is added to the schemes. The simulation network space consists of 1000 sensors randomly deployed in a 500 m × 500 m squared area. The model of sensor nodes is followed by the specification of MICA2 [39,40]. The radio range of each sensor node is 20 m. The source generates a CBR flow with an interval of 0.05 s, and a packet frame size is fixed at 30 bytes. The simulation time is 100 s and IEEE 802.15.4 standard [41] is exploited as the MAC protocol.

7.1. Performance Metrics

We use the following performance metrics:

- Deadline success ratio (DSR): This metric is the ratio of successfully received data packets in the time deadline to all generated data packets from the source node. It is one of the most important metrics in soft real-time systems.
- Energy consumption (EC): This metric is the total energy consumption for the real-time service. Energy consumption is an important metric in wireless sensor networks. The power consumption values of transmitting and receiving are 21 mW and 15 mW, respectively, [40].

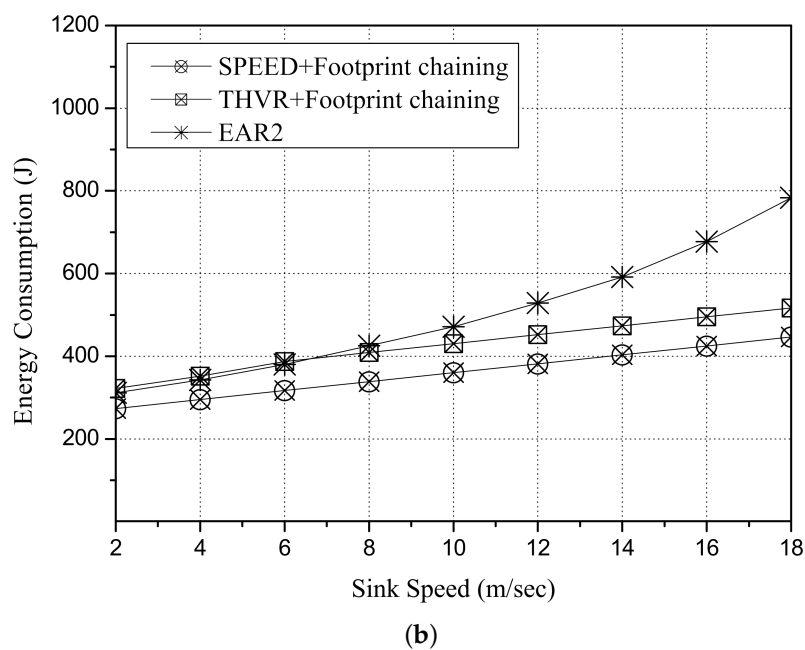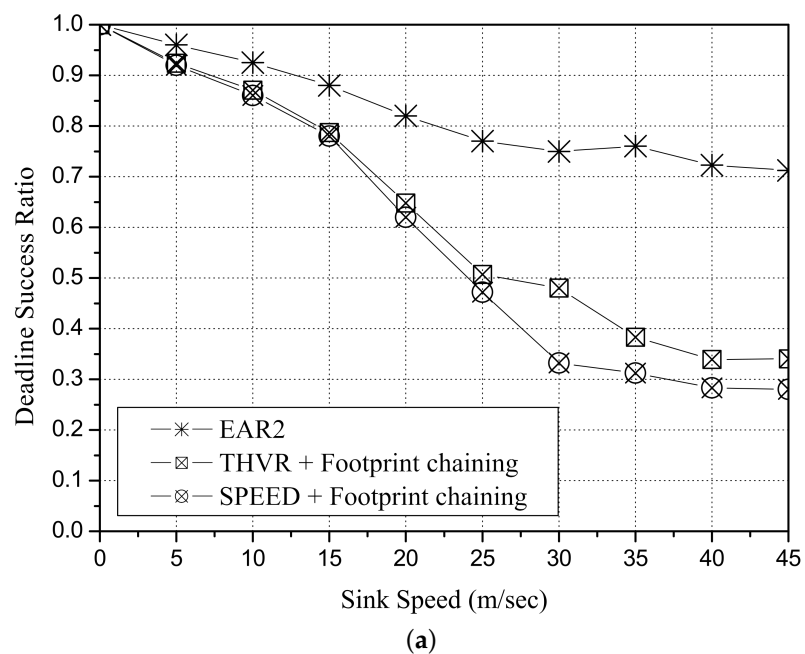### 7.2. Performance Comparison with the Existing Schemes

7.2.1. Impact of Sink Speed

Figure 5a shows a comparison of DSR according to the sink speed. The end-to-end distance between the source and the sink is fixed at 300 m. Since SPEED and THVR employ data delivery speeds calculated from the distance between the source and the initial static location of the sink, the moving sink may not receive the data within the desired time deadline even though the data is delivered to the initial location of the sink on time. EAR2, however, forwards data with the modified delivery speeds that consider sink mobility, so the probability of deadline success is higher than with the other schemes.

Figure 5b indicates the EC of the proposed scheme with those of the other protocols according to the sink speed. We keep the distance between the source and the sink at 300 m. In SPEED and THVR, as the sink speed increases, the data path for footprint chaining from the initial location of the sink to the moved location increases, and the interval between sink location updates decreases. This makes the energy consumption gradually increase. The energy consumption of THVR is higher than that of SPEED due to the initialization cost of discovering two-hop neighbors, but they have a similar increasing pattern. EAR2 consumes the same energy for initialization as SPEED but exploits the restricted flooding to support sink mobility. Because the *EA* is exponentially proportional to the sink speed, EAR2 consumes more energy than the other schemes as the sink moves faster.

7.2.2. Impact of End-to-End Distance

Figure 6a shows a comparison of DSR according to the distance from the source to the sink. The moving speed of the sink is fixed at 20 m/s. As the end-to-end distance increases, the DSRs of SPEED and THVR are decreased rapidly but that of EAR2 is decreased more gently than those of the other schemes. This result is derived from the difference of data delivery speed calculation mentioned in Section 3.1. Because EAR2 forwards data with the modified data delivery speed that considers sink mobility according to end-to-end distance, the DSR is higher than those of the other schemes.

(**a**)

(**b**)

Figure 5. Performance comparison of EAR2 with the existing schemes SPEED and THVR according to sink speed: (**a**) deadline success ratio and (**b**) energy consumption.
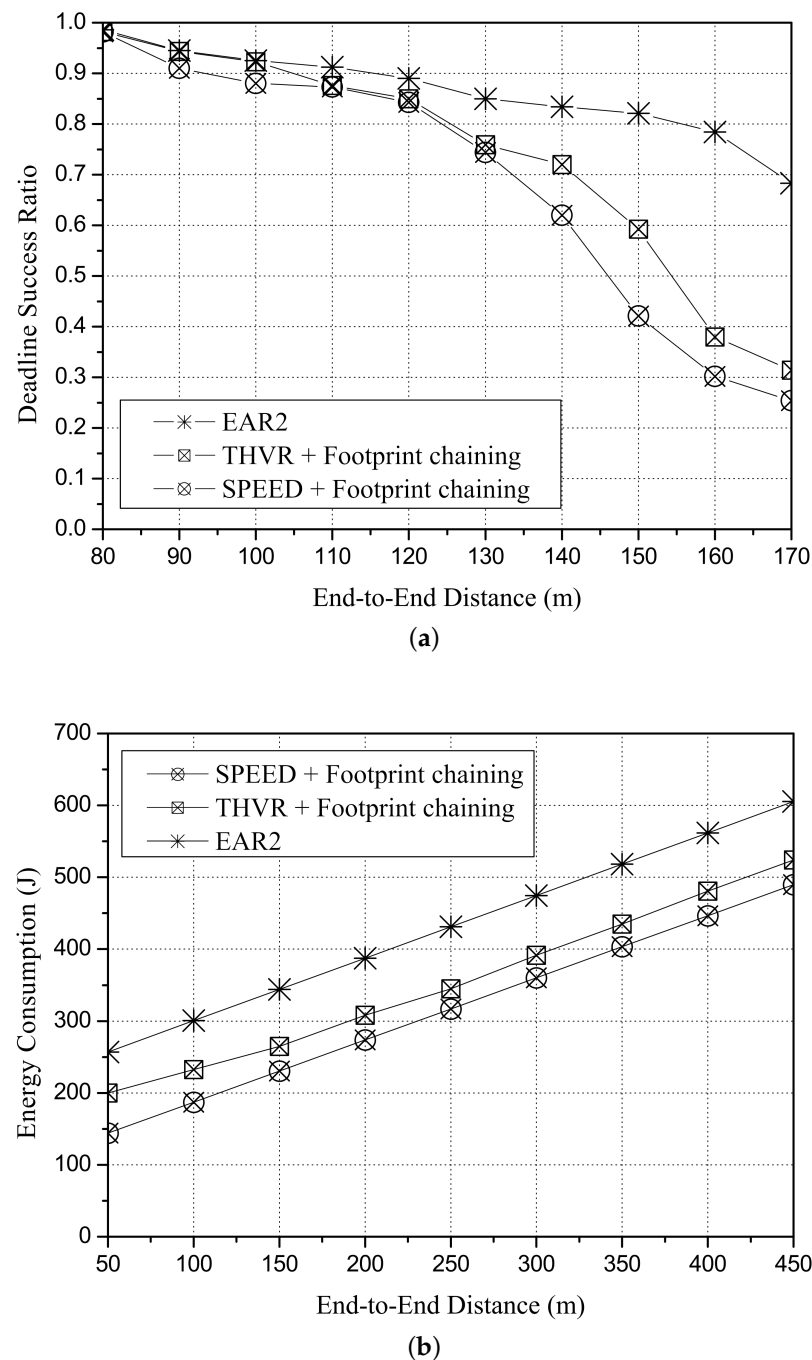
Figure 6. Performance comparison of EAR2 with the existing schemes SPEED and THVR according to end-to-end distance: (**a**) deadline success ratio and (**b**) energy consumption.

We compare the EC of the proposed scheme with that of the other schemes in Figure 6b. The EC of all schemes is proportional to the distance, because more sensor nodes participate in forwarding according to the distance. EAR2 consumes an amount of energy that is slightly higher than the other schemes because it exploits restricted flooding. The difference of the EC between SPEED and THVR indicates the cost of the initial beacon message for neighbor discovery. The EC of the three schemes has a similar increasing pattern because the end-to-end distance does not affect the frequency of the sink location update, the data path for footprint chaining, and the extent of *EA*.

7.3. Performance Comparison of EAR2 and EAR2M

EAR2M exploits multi-path data dissemination through multiple flooding points on the EA to overcome a single flooding point in EAR2. By comparing EAR2 and EAR2M, we can observe the impact of the number of flooding points and therefore the number of data paths according to the sink speed and end-to-end distance.

7.3.1. Impact of Sink Speed

To simulate the impact of sink speed, we fix the end-to-end distance between the source and the sink at 300 m. Figure 7a shows the effect of sink speed on DSR with the different number of flooding points. EA is exponentially proportional to the sink speed. This means the average time that the sink receives data is increased, because the number of the average data forwarding hops for flooding is increased. Because these two values can decrease using data flooding in EA through multiple flooding points, the higher number of flooding points in EAR2M leads to the higher DSR.

Figure 7b shows the EC according to the sink speed with the different number of flooding points. Because the number of nodes that participate in data forwarding is proportional to the number of flooding points, more data flooding points in EAR2M leads to more energy consumption, as shown by the difference in energy consumption at the same sink speed on the graph. As the sink speed increases, energy consumption for restricted flooding increases due to the extension of EA. However, the cost for flooding is not different according to the number of flooding points at the same sink speed. Accordingly, each graph has a similar increasing pattern.

7.3.2. Impact of End-to-End Distance

Figure 8a shows the effect of end-to-end distance on DSR with the different number of flooding points. As the end-to-end distance from the source to the sink increases, the probability of failure to maintain the desired data delivery speeds increases, and thus the data loss ratio also increases. Data dissemination through multiple flooding points can reduce these issues because the data are copied and forwarded in the separated routing path. Therefore, the higher number of flooding points in EAR2M leads to a higher DSR.

Figure 8b shows the EC according to the end-to-end distance with the different number of flooding points. As the end-to-end distance increases, the cost for flooding is fixed but the energy consumption for data transmission from the source to the flooding points increases. Therefore, the energy consumption for data dissemination through more flooding points in EAR2M increases much more rapidly as the end-to-end distance increases.

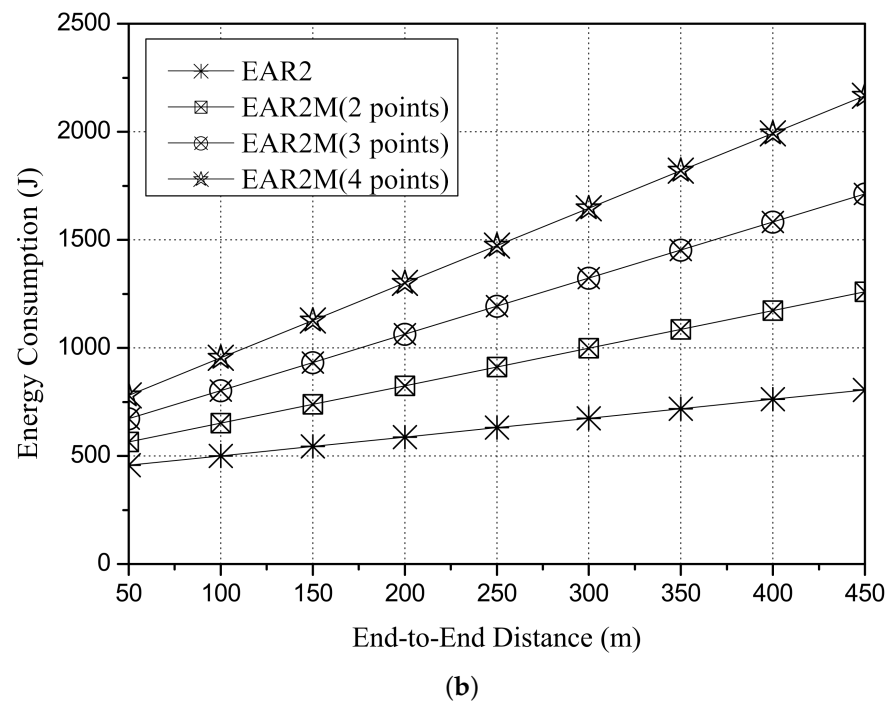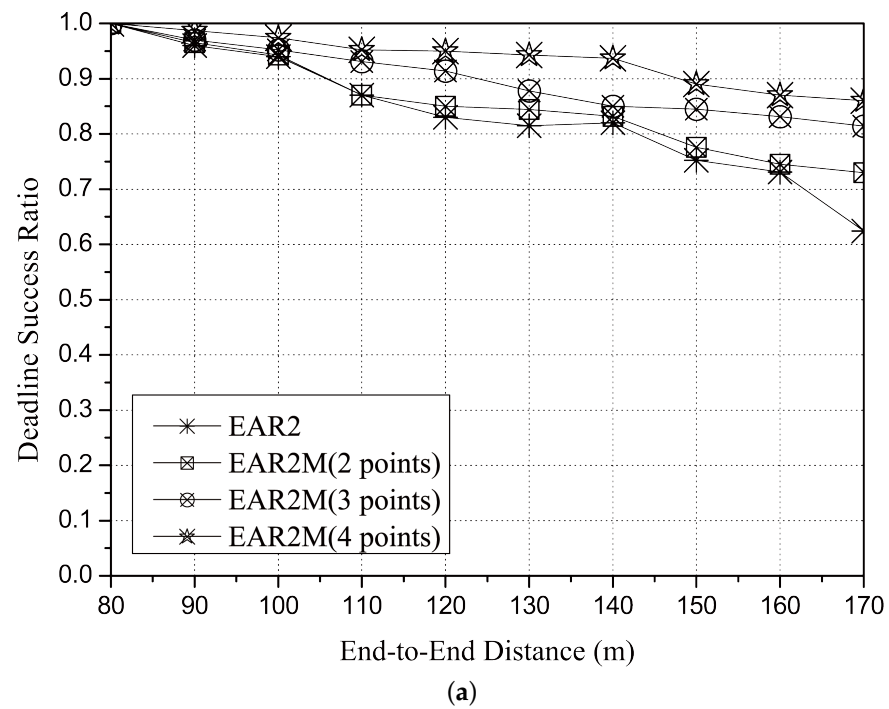### 7.4. Performance Comparison of EAR2M, EAR2M_R, and EAR2M_E

We extend the proposed scheme by allowing the relay between flooding points of the expected area in order to transfer data packets to multiple flooding points in Chapter VI. In this subchapter, we compare the extended schemes of EAR2M with four flooding points.

Figure 9a indicates the DSR according to the end-to-end distance of the proposed schemes (EAR2M, EAR2M_R, and EAR2M_E). In this experiment, we fix the moving speed at 20 m/s. As the end-to-end distance increases, the DSR decreases. In contrast to EAR2M, EAR2M_R and EAR2M_E show a lower DSR because their unicast distance is longer than that of EAR2M and higher data delivery speeds are required. However, the difference is very trivial as the extended distance is extremely small compared to the end-to-end distance.

Figure 9b shows the EC according to the end-to-end distance. As the end-to-end distance becomes larger, more nodes should participate in a path and the energy consumption increases. In the extended schemes, multiple paths for multiple flooding points are merged. Due to the merged path, the cost for unicasting toward flooding points can be reduced. EAR2M_E shows a lower EC than EAR2M_R because EAR2M_E allows the multiple flooding points to relay toward other flooding points.
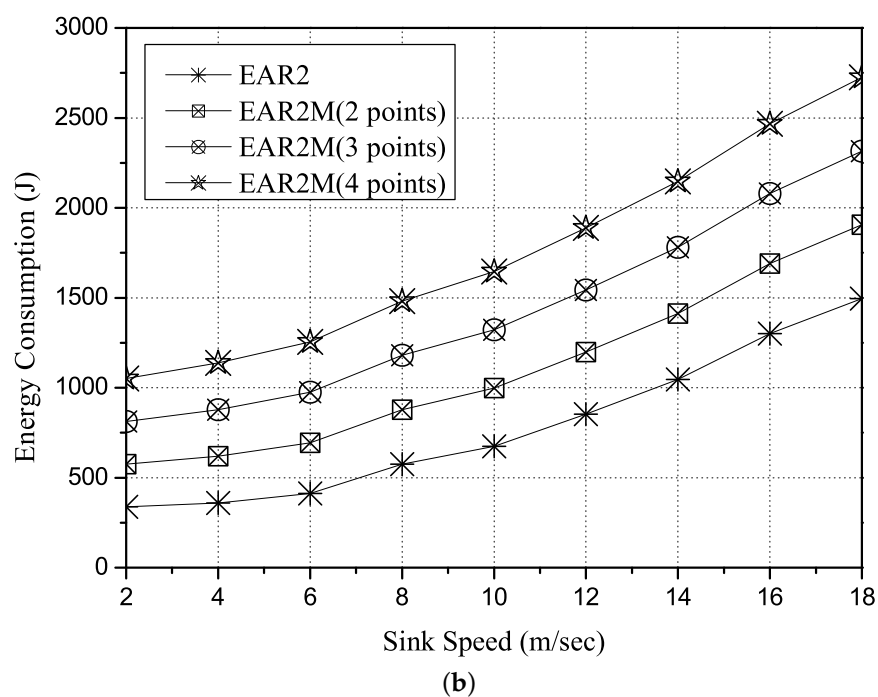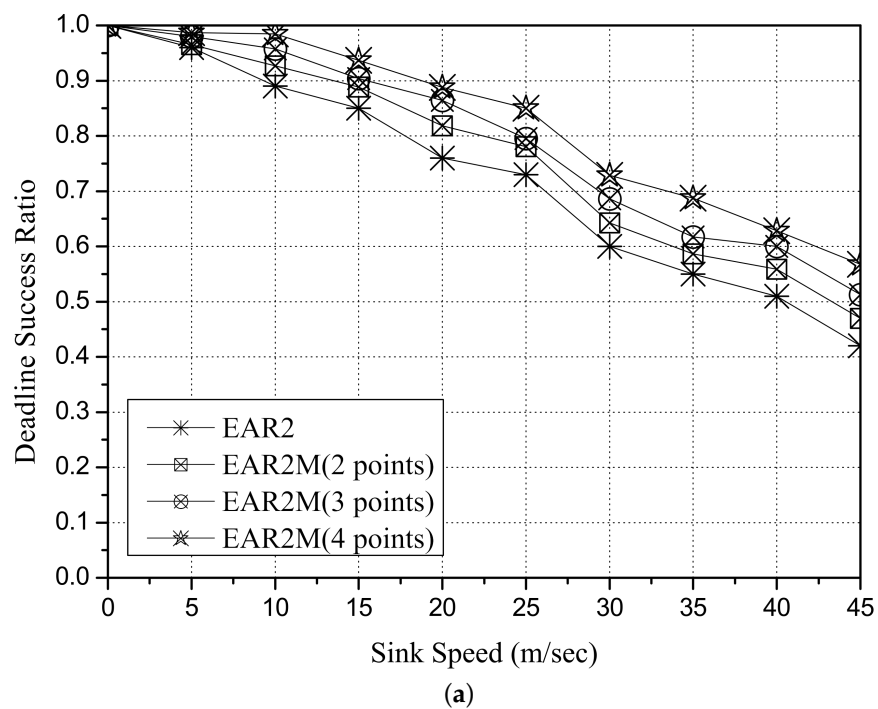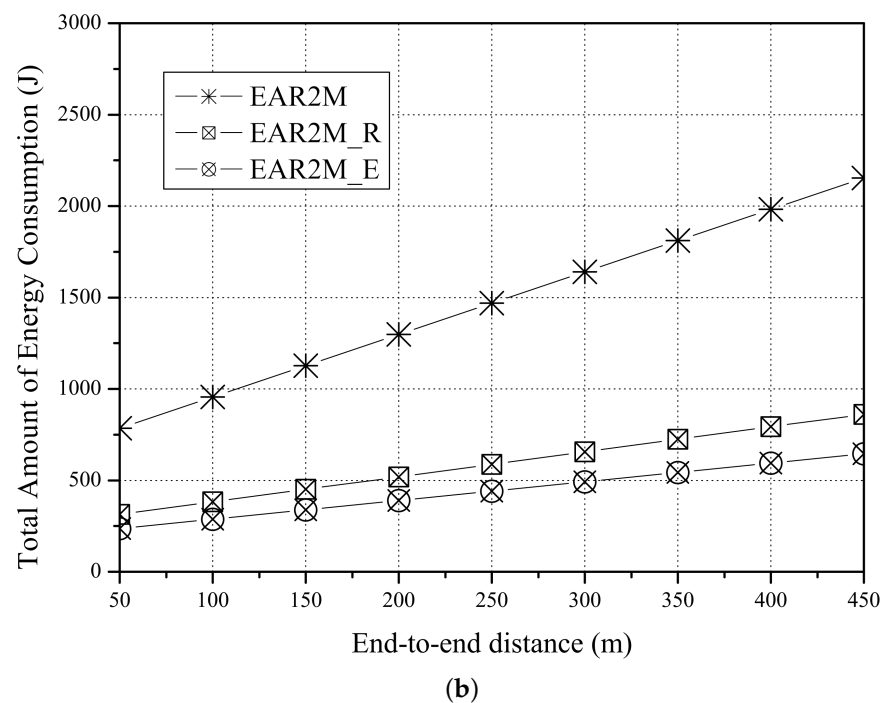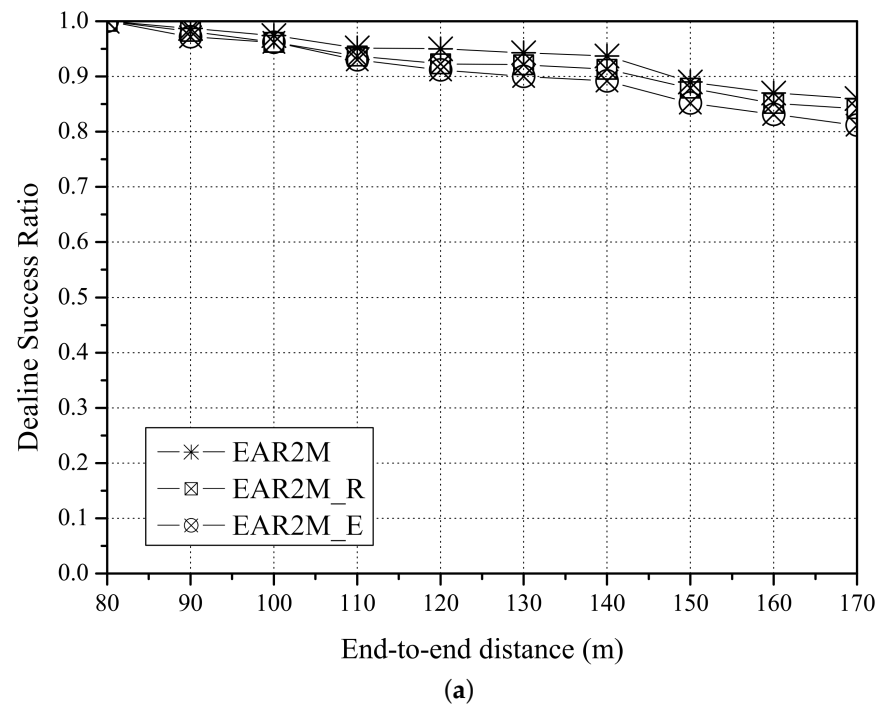
Figure 7. Performance comparison of EAR2 with EAR2M according to sink speed: (**a**) deadline success ratio and (**b**) energy consumption.

(**a**)

(**b**)

Figure 8. Performance comparison of EAR2 with EAR2M according to end-to-end distance: (**a**) deadline success ratio and (**b**) energy consumption.

To simulate the impact of sink speed, we fix the end-to-end distance between the source and the sink at 300 m. Figure 10a shows the DSR according to the moving sink speed. The proposed schemes determine the expected area for moving the sink. The expected area is exponentially proportional to the sink speed. Due to the size of the expected area, all values in all schemes decrease. The results of both EAR2M_R and EAR2M_E are slightly

lower than that of EAR2M because EAR2M_R and EAR2M_E require a higher data delivery speed for a longer distance than EAR2M.
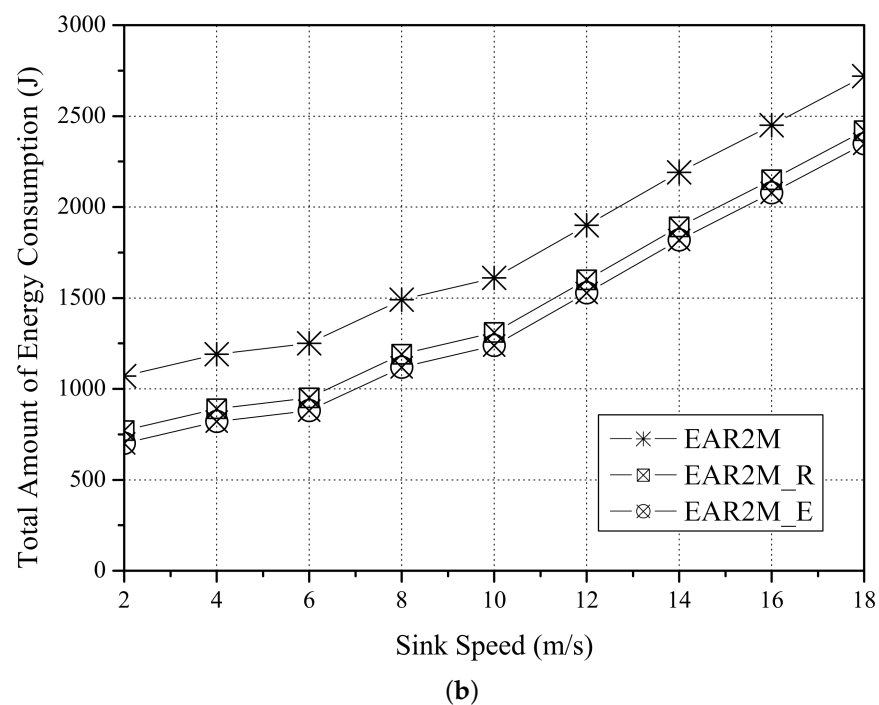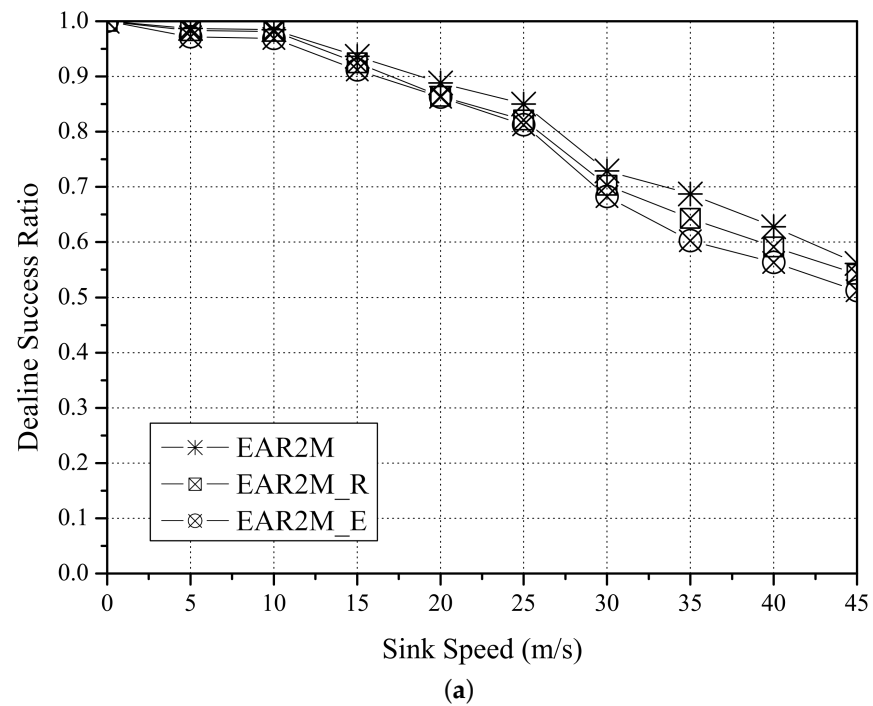

(**a**)

(**b**)

Figure 9. Performance comparison of the extended schemes according to end-to-end distance: (**a**) deadline success ratio and (**b**) energy consumption.

Figure 10b indicates the energy consumption of the proposed schemes according to the sink speed. The extended schemes, EAR2M_R and EAR2M_E, reduce the energy consumption by merging the paths for unicasting and allowing relay via the flooding

points. Although the moving sink speed is faster, the schemes always show a lower energy consumption than EAR2M.

(**a**)

(**b**)

Figure 10. Performance comparison of the extended schemes according to sink speed: (**a**) deadline success ratio and (**b**) energy consumption.

8. Conclusions

In this paper, we have proposed a novel real-time routing protocol called EAR2 for mobile sinks in wireless sensor networks. To calculate a data delivery speed for a mobile sink, we provided a static distance to an expected area where it can be located by its moving

speed and exploited the flooding time within the expected area. Then, EAR2 guarantees that the total summation of the unicasting time to the expected area and the flooding time within the expected area is smaller than the desired time deadline. We also have proposed EAR2M using multiple flooding points to reduce flooding time and enhance the reliability of data transmission. To reduce the transmission delay and the energy consumption using efficient data transmission among multiple flooding points, we also have proposed two extensions of EAR2M, EAR2M_R and EAR2M_E. We have evaluated the performance of our protocols through simulation experiments conducted in various environments by the existing schemes SPEED [4] and THVR [6]. Simulation results verified that EAR2 enhances the deadline miss ratio and the energy efficiency by 57.94% and 39.80% over SPEED, respectively, and by 36.40% and 20.18% over THVR, respectively. Moreover, EAR2M with four flooding points improves the deadline miss ratio and the energy consumption by an average of 12.69% and 131.86% over EAR2. We also have verified that EAR2M_R and EAR2M_E have a trade-off between the deadline miss ratio and energy consumption.

Author Contributions: Conceptualization, H.C.; methodology, Y.N.; software, Y.N.; validation, H.C., Y.S., S.P.; writing—original draft preparation, Y.N.; writing—review and editing, H.C., E.L.; supervision, E.L.; funding acquisition, E.L. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Akyildiz, I.; Su, W.; Cayirci, E. A survey on sensor networks. *IEEE Commun. Mag.* **2002**, *40*, 102–114. [CrossRef]
2. Borges, L.; Velez, F.; Lebres, A. Survey on the characterization and classification of wireless sensor network applications. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1860–1890. [CrossRef]
3. Bensaleh, M.; Saida, R.; Kasem, Y.; Abid, M. Wireless sensor network design methodologies: A survey. *J. Sens.* **2020**, *2020*, 9592836. [CrossRef]
4. He, T.; Stankovic, J.A.; Abdelzaher, T.F.; Lu, C. A spatiotemporal communication protocol for wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2005**, *16*, 995–1006. [CrossRef]
5. Felemban, C.; Lee, C.; Ekici, E. MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks. *IEEE Trans. Mob. Comput.* **2006**, *5*, 738–754. [CrossRef]
6. Li, Y.; Chen, C.S.; Song, Y.Q.; Wang, Z.; Sun, Y. Enhancing real-time delivery in wireless sensor networks with two-hop information. *IEEE Trans. Ind. Inform.* **2009**, *5*, 113–122. [CrossRef]
7. Mehto, A.; Tapaswi, S.; Pattanaik, K. A review on rendezvous based data acquisition methods in wireless sensor networks with mobile sink. *Wirel. Netw.* **2020**, *26*, 2639–2663. [CrossRef]
8. Yu, S.; Zhang, B.; Li, C.; Mouftah, H. Routing protocols for wireless sensor networks with mobile sinks: A survey. *IEEE Commun.* **2014**, *52*, 150–157. [CrossRef]
9. Hamida, E.; Chelius, G. Strategies for data dissemination to mobile sinks in wireless sensor networks. *IEEE Wirel. Commun.* **2008**, *15*, 31–37. [CrossRef]
10. Sharifi, H.; Javadi, H.; Moeini, A.; Hosseinzadeh, M. Unital design based sink location service for wireless sensor networks. *IEEE Access* **2018**, *6*, 28733–28745. [CrossRef]
11. Lee, E.; Yu, F.; Park, S.; Kim, S.; Noh, Y.; Lee, E.-K. Design and analysis of novel quorum-based sink location service scheme in wireless sensor networks. *Wirel. Netw.* **2014**, *20*, 493–509. [CrossRef]
12. Stojmenovic, I.; Liu, D.; Jia, X. A scalable quorum-based location service in ad hoc and sensor networks. *Int. J. Commun. Netw. Distrib. Syst.* **2008**, *1*, 71–94. [CrossRef]
13. Park, S.; Lee, E.; Park, H.; Jung, J.; Kim, S. Strategy for real-time data dissemination to mobile sinks in wireless sensor networks. In Proceedings of the IEEE PIMRC (2010), Istanbul, Turkey, 26–29 September 2010.
14. Cheng, L.; Niu, J.; Gu, Y.; Luo, C.; He, T. Achieving Efficient Reliable Flooding in Low-Duty-Cycle Wireless Sensor Networks. *IEEE/ACM Trans. Netw.* **2016**, *24*, 3676–3689. [CrossRef]

15. Yan, F.; Zhang, X.; Tao, L.; Zhang, Z. Network Coding-Based Flooding with a Mobile Sink in Low-Duty-Cycle Wireless Sensor Networks. *IEEE Trans. Mob. Comput.* **2019**, *18*, 1857–1869. [CrossRef]

16. Dong, M.; Ota, K.; Liu, A.; Guo, M. Joint Optimization of Lifetime and Transport Delay under Reliability Constraint Wireless Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* **2016**, *27*, 225–236. [CrossRef]

17. Cheng, L.; Niu, J.; Luo, C.; Shu, L.; Kong, L.; Zhao, Z.; Gu, Y. Towards minimum-delay and energy-efficient flooding in low-duty-cycle wireless sensor networks. *Comput. Netw.* **2018**, *134*, 66–77. [CrossRef]

18. Saqaeeyan, S.; Roshanzadeh, M. Improved Multi-Path and Multi-Speed Routing Protocol in Wireless Sensor Networks. *Int. J. Comput. Netw. Inf. Secur.* **2012**, *4*, 8–14. [CrossRef]

19. Zhou, Y.; Ngai, E.; Lyu, M.; Liu, J. POWER-SPEED: A power-controlled real-time data transport protocol for wireless sensor-actuator networks. In Proceedings of the IEEE Wireless Communications and Networking Conference 2007, Hong Kong, China, 1–15 March 2007.

20. Kaur, M.; Sharma, A. Routing in WSN network using Neural Network (NN) and SPEED protocol. In Proceedings of the International Conference on Contemporary Computing and Informatics (IC3I) (2016), Greater Noida, India, 14–17 December 2016.

21. Aissani, M.; Bouznad, S.; Fareb, A.; Laidoui, A. EA-SPEED: Energy-aware real-time routing protocol for wireless sensor networks. *Int. J. Inf. Commun. Technol.* **2013**, *5*, 22–44. [CrossRef]

22. Jung, J.; Park, S.; Lee, E.; Noh, S.; Kim, S. OMLRP: Multi-hop information based real-time routing protocol in wireless sensor networks. In Proceedings of the IEEE WCNC (2010), Sydney, Australia, 18–21 April 2010.

23. Zhao, L.; Kan, B.; Xu, Y.; Li, X. FT-SPEED: A fault-tolerant, real-time routing protocol for wireless sensor networks. In Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM) (2007), Shanghai, China, 21–25 September 2017.

24. Lee, E.; Park, S.; Oh, S.; Kim, S.-H.; Nam, K.-D. Real-Time Routing Protocol Based on Expect Grids for Mobile Sinks in Wireless Sensor Networks. In Proceedings of the IEEE Vehicular Technology Conference (VTC-Fall) (2011), San Francisco, CA, USA, 5–8 September 2011.

25. Hofmann-Wellenhof, B.; Lichtenegger, H.; Collins, J. *Global Positioning System: Theory and Practice*; Springer: Berlin/Heidelberg, Germany, 1997; Volume 4.

26. Saeed, N.; Nam, H.; Al-Naffouri, T.; Alouini, M. A state-of-the-art survey on multidimensional scaling-based localization techniques. *IEEE Commun. Surv. Tutorials* **2019**, *21*, 3565–3583. [CrossRef]

27. Buehrer, R.M.; Wymeersch, H.; Vaghefi, R.M. Collaborative sensor network localization: Algorithms and practical issues. *Proc. IEEE* **2018**, *106*, 1089–1114. [CrossRef]

28. Chen, Q.; Gao, H.; Cheng, L.; Li, Y. Label coloring based beaconing schedule in duty-cycled multihop wireless networks. *IEEE Trans. Mob. Comput.* **2019**, *19*, 1123–1137. [CrossRef]

29. Chen, Q.; Kanhere, S.; Hassan, M. Adaptive position update for geographic routing in mobile ad hoc networks. *IEEE Trans. Mob. Comput.* **2013**, *12*, 489–501. [CrossRef]

30. Raw, R.; Lobiyal, D.; Das, S.; Kumar, S. Analytical evaluation of directional-location aided routing protocol for vanets. *Wirel. Pers. Commun.* **2015**, *82*, 1877–1891. [CrossRef]

31. Basagni, S.; Chlamtac, I.; Syrotiuk, V.R.; Woodward, B.A. A distance routing effect algorithm for mobility (DREAM). In Proceedings of the ACM International Conference on Mobile Computing and Networking (Mobicom) (1998), Dallas, TX, USA, 25–30 October 1998.

32. Ko, Y.-B.; Vaidya, N.H. Location-aided routing (LAR) in mobile ad hoc networks. *Wirel. Netw.* **2000**, *6*, 307–321. [CrossRef]

33. Syed, B.; Sajid, S.; Mubashir, R. NS-2 based simulation framework for cognitive radio sensor networks. *Wirel. Netw.* **2018**, *24*, 1543–1559.

34. Zhang, J.; Li, W.; Cui, D.; Zhao, X.; Yin, Z. The NS2-based simulation and research on wireless sensor network route protocol. In Proceedings of the IEEE WICOM (2009), Beijing, China, 24–26 September 2009.

35. Mohammed, H.; Hussain, A.; Fadi, A. A Survey on Multipath Routing Protocols for QoS Assurances in Real-Time Wireless Multimedia Sensor Networks. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1424–1456.

36. Ganesan, D.; Govindan, R.; Shenker, S.; Estrin, D. Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *Mob. Comput. Commun. Rev.* **2001**, *5*, 11–25. [CrossRef]

37. Liu, H.; Jia, X.; Wan, P.; Liu, X. A distributed and efficient flooding scheme using 1-hop information in mobile ad hoc networks. *IEEE Trans. Parallel Distrib. Syst.* **2007**, *18*, 658–671. [CrossRef]

38. Cao, Z.; Zheng, X.; Ma, Q.; Miao, X. COFlood: Concurrent Opportunistic Flooding in Asynchronous Duty Cycle Networks. In Proceedings of the IEEE SECON (2021), Virtual, 6–9 July 2021.

39. Djamel, D.; Miloud, B. Synchronization Protocols and Implementation Issues in Wireless Sensor Networks: A Review. *IEEE Syst. J.* **2016**, *10*, 617–627.

40. Hill, J.; Culler, D. Mica: A wireless platform for deeply embedded networks. *IEEE Micro* **2002**, *22*, 12–24. [CrossRef]

41. Gutierrez, J.; Naeve, M.; Callaway, E.; Bourgeois, M.; Mitter, V.; Heile, B. IEEE 802.15.4: A developing standard for low-power low-cost wireless personal area networks. *IEEE Netw.* **2001**, *15*, 12–19. [CrossRef]