electronics

Article

Adapted Diffusion for Energy-Efficient Routing in Wireless Sensor Networks

Gad Hakim *, Robin Braun and Justin Lipman

School of Electrical and Data Engineering, University of Technology Sydney, Ultimo, NSW 2007, Australia; robin.braun@uts.edu.au (R.B.); justin.lipman@uts.edu.au (J.L.)
* Correspondence: gad.hakim@student.uts.edu.au

Abstract: We present a routing model called adapted diffusion for ad hoc wireless sensor networks that is more energy efficient than directed diffusion. Adapted diffusion is modeled using NetLogo with agent-based modeling. In this agent-based NetLogo model, we set the distance from a random source and the distance from the sink to optimize the routing protocols. By using this routing technique significant energy savings were achieved. We consider a three-dimensional communication network that can be used in a building or a stack of shipping containers. Our model can be extended to a 3D model.

Keywords: wireless sensor network; routing; NetLogo; adaptive diffusion; complex systems

Citation: Hakim, G.; Braun, R.; Lipman, J. Adapted Diffusion for Energy-Efficient Routing in Wireless Sensor Networks. *Electronics* **2024**, *13*, 2072. https://doi.org/10.3390/electronics13112072

Academic Editors: Dongkyun Kim and Stefano Scanzio

Received: 22 March 2024
Revised: 5 May 2024
Accepted: 18 May 2024
Published: 27 May 2024

1. Introduction

1.1. Energy-Efficient WSN Algorithms

Most of the research on WSNs has concentrated on the design of energy- and computationally efficient algorithms and protocols.

This paper introduces a new energy-efficient routing protocol which we model using an agent-based model (ABM) in the NetLogo environment where the nodes have changing energy levels. In our previous paper [1], we described a new ABM of the standard directed diffusion (DD) routing protocol [2]. Additionally, in our previous paper [1], we presented two new routing protocols, lazy diffusion (LD) and gradient diffusion (GD). These new protocols have the potential to be more energy efficient. However, there is no guarantee that the source would be found due to the random search being undertaken, and the energy depletion of the random nodes used to forward packets.

1.2. Contribution of This Paper

In this paper, we present a novel routing protocol we call adapted diffusion (AD). In this protocol, the source and sink are always discovered, and the requested data retrieved. Secure communication is achieved with significant energy saving. The network is made up of nodes. All nodes can forward information to their immediate neighbors and receive information from their immediate neighbors.

The network has a minimum of two special nodes. These are the source (where the information comes from) and the sink (the node that creates an interest, collects the information and forwards data to an external server.) We set up the sink at the edge of the network, and we randomly choose one of the other nodes to be the source.

- The source has sensors that collect information about either their own condition, or their surroundings.
- The sink can request information that is available at the source.
- Each node can select one of its neighbors for transmitting or receiving data based on the conditions we introduce.

1.3. Comparison with Directed Diffusion

A comparison is made between adapted diffusion (AD) and directed diffusion (DD) in energy consumption when transmitting and receiving data. Adaptation occurs when the nodes learn the location of their neighbors in relation to the location of the network sink and source. We demonstrate that significant energy savings are achieved by use of the adapted model.

1.4. The Main Characteristics of NetLogo

- A fully programmable simulator;
- A grid of stationary agents allowing mobile agents to interact;
- Links are created between the agents to allow formation of a network;
- Multiple agents and variables are allowed to create the simulations;
- A wide vocabulary for a programmable primitive language;
- Unlimited number of variables and agents are used in the simulation;
- Complex dynamic systems can be simulated with NetLogo.

With the agent-based approach, the behavior of individual agents can be programmed and NetLogo shows emergent properties arising from their interactions.

The model can be extended to a 3D model using the NetLogo 3D modeling tool.

We intend in future research to further analyze LD and GD compared to AD and DD. In this paper, we concentrated on AD.

The previous GD algorithm only had a gradient from the sink to return the messages to the sink via the sink gradient. The difficulty is that the source position changes whilst the sink is stationary. Hence, we need a changing gradient from the source. We developed the adapt diffusion model, with a new gradient formed from each run as the source location is changed.

2. Wireless Sensor Networks and Routing—Background/Survey

2.1. Literature Survey

Many routing protocols are available depending on the application required. In our situation, we selected DD as our baseline routing protocol. A number of papers have been published with new protocols comparing them to DD and achieving reduced energy usage. For example Lipman [3] used Resource-Aware Information Collection (RAIC), first using Utility-Based Flooding (UBF). The authors compared Resource-Aware Information Collection (RAIC), a distributed two-phased resource-aware approach to information collection in ad hoc networks to DD. Similarly, we are comparing AD to DD.

Wireless sensor networks are networks of sensing nodes that communicate by wireless means with other neighboring nodes and have large areas of use in many industries, the environment, defense, mining, and healthcare. In recent years, nodes/sensors have become cheaper, more efficient, and smarter. They are now able to implement complex routing protocols to forward data to a central location for analysis. The key issues with WSNs relate to their life being limited by the power availability from the batteries, and hence, the restricted life of the network. Energy is used in transmitting and receiving data, and computation. The biggest energy consumer is transmission. When a node uses up its energy, sensor data can no longer be transmitted. Neither can the node forward the sensor data from other nodes. Thus, whole portions of the network can become isolated. The way to extend the life of the network is by optimizing the energy consumption of the routing protocols used to transmit and receive data from the sink (requesting data) to the source (data requested) and back. In this paper, we determine the number of source data retrievals until the data can no longer reach the sink due to insufficient energy for transmission in nodes for particular protocols. Most of the research on WSNs has concentrated on the design of energy- and computationally efficient algorithms and protocols. The purpose of the paper is to describe the new agent-based model of a data-centric protocol, the AD protocol for wireless sensor networks, and a newly developed derivative of the AD protocol. The models are tested for longevity and energy efficiency. In future research, all models will allow for energy harvesting. Our models then test all the protocols to extract time-to-live statistics for various parameters. Changes to several aspects including the routing protocol, the ability of nodes to perform energy harvesting, and the ability to enhance the energy of selected nodes are made to the models. The results of energy consumption and network longevity are plotted.

Substantial research has and is being conducted to optimize the lifetime of WSNs.

Robot path planning bears similarity to WSN networks in optimizing travel path to a source. A number of papers [4,5] have been published on robot path planning. Most path-planning algorithms combine deep reinforcement learning (DRL) and path-planning algorithms to achieve obstacle avoidance and path shortening. These path-planning papers use mathematical modeling. Cai considers a 3D network of autonomous underwater vehicles (AUVs). The simulation results in Cai's paper demonstrate that the integration of the 3D Dubins curve with the MTSP model is successful and effective for solving the 3D target assignment and path-planning problem. Target tracking with multiple AUVs is undertaken. The Dubins path typically refers to the shortest curve that connects two points in the two-dimensional Euclidean plane. Xing also looks at AUV path planning in a complex underwater environment. These are three-dimensional networks where robots such as drones or underwater vehicles travel. The algorithm considers flight paths with the intent of preventing two agents colliding. In our case, the messages travel in a way to prevent revisiting previously visited locations. Xing states that the current means of AUV underwater path planning mainly include the ant colony algorithm [6,7], fuzzy algorithm [8], genetic algorithm [5,9], algorithms based on neural networks [10], and algorithms based on the artificial potential field theory [11,12]. Tang et al. [4] consider optimizing the performance of the artificial electric field algorithm (AEFA) and broadening its application domain in order to aid in providing robot path planning in 3D complex scenes.

Abdullah [13] uses mathematical models of wireless body area networks (WBANs) to monitor patient care and uses a multi-hop routing protocol. The simple energy-aware and reliable (SEAR) routing protocol is proposed to transmit reliable data packets in a WBAN. Energy savings are compared to energy-efficient and reliable routing based on reinforcement learning and fuzzy logic. Our model compares energy savings compared to DD. Methods include allocating resources base levels, a cross-layer design, opportunistic transmission such as sleep–wake scheduling, routing, and clustering, coverage, connectivity and optimal deployment, mobile relays and sinks, data gathering and network coding, data correlation, beamforming, and energy harvesting. Other work includes an ABM of a WSN [14] that does not use NetLogo. Hamzi [15] broadly categorized energy conservation schemes under three main headings: duty-cycling, data-driven, and mobility-driven techniques. Duty cycling is aimed at reducing idle listening when the node's radio waits in vain for frames and overhearing when nodes stay active listening to uninterested frames. Data-driven techniques use some parameters of the data themselves to make decisions to reduce energy consumption during communication, while mobility schemes consider the mobility of the sink or relay nodes as a factor affecting the energy consumed in the network. Energy conservation involves minimizing the communication cost in nodes. As the radio is the biggest consumer of energy, this can be achieved by energy-efficient routing protocols. For example, there is a NetLogo model [16] that investigates a mobile ad hoc network (MANET). In this paper, distributed Dijkstra's shortest path algorithm is used to solve the routing problem. This algorithm is frequently used to solve routing problems in telecommunication networks. This algorithm determines the shortest path between the source and all other nodes in the graph. This paper is different from our paper in that in our paper we track energy usage and use a fixed sink and changing source. We can also modify the model for a mobile sink or multiple sinks.

Gupta et al. [17] focused on various issues of the healthcare system and their solutions. An energy-efficient routing protocol that can provide sensed data to the collection center or data hub for further processing and treatment of the patients was proposed. Here, the authors fixed zones for sending data to the zone head using distance-aware routing, and then, the zone head sent the aggregated data to the data hub wireless sensor network. It was better than the low-energy adaptive clustering hierarchy (LEACH) by 42% and than the distance-based residual energy-efficient protocol (DREEP) by 30% in energy efficiency, and stability was 58% better than LEACH and 39% than DREEP. Mu [18] studies a directional diffusion routing protocol for WBANs, and establishes a gradient with a minimum hop count and the remaining energy of the node, so that it can meet the requirements of the WBANs in terms of reliability and energy consumption. This paper uses the Network Simulator (NS2) software to generate improved directional diffusion protocol (ENDD) and compares it to DD. It compares flood, DD, and ENDD.

There is no similar model to our ABM model in the literature to the best of our knowledge. We also develop two new versions of DD, which we call lazy diffusion and gradient diffusion, which have the potential to reduce the energy usage, and provide us with a vehicle for investigating diffusion protocols.

Silva [19] describes DD as a data-centric protocol focusing on desired data rather than individual nodes and extending the original DD to a DD protocol family, which includes: (1) two-phase pull diffusion, (2) one-phase pull diffusion, and (3) push diffusion. The approach can support multiple sinks and sources. The sink sends interest messages to find the source and the source uses exploratory data messages to find sinks as data are transmitted to all neighbors. This is called the two-phase pull. A key feature of DD is that every sensor node can be task-aware. By this we mean that nodes store and interpret interests, rather than simply forwarding them along. Data is cached in neighboring nodes and moves towards the sink. After this, the exploratory data-reinforced paths are used for transmission. All transmission in DD is hop-by-hop or neighbor-to-neighbor. GEAR protocols [19] (Geographic and Energy-Aware Routing) extend diffusion using geographic locations. This paper further discusses the application performance of different diffusion algorithms.

Many routing protocols are available for this kind of ad hoc wireless sensor network, depending on the application required, or the longevity/energy requirements.

Xia [20] proposed a gradient-based routing protocol considering the minimum hop count and the remaining energy of each node while relaying data from the source node to the sink. A scheme for topology update is provided. The relationship between wireless communication, energy consumption E, and transmission distance $d = E \propto dk$, where k is usually 2∼4, is presented. Hence, short-distance multiple-hop communication is preferable to long-distance direct communication in sensor networks. Hence, we selected communication with neighboring nodes only. The source node will find the minimum hop count path to the sink. In our protocol, we use the hop count and remaining energy of each node as the metrics.

Talib [21] proposes MAC protocols in WSNs, achieving a low duty cycle by employing periodic sleeping and waking. A predictive wakeup MAC (PW-MAC) protocol was made using asynchronous duty cycling, reducing the consumption of the node energy by allowing the senders to predict the receiver´s wakeup times. In our paper, the NetLogo [22] simulation environment is used for sending, receiving, forwarding, and sleeping simulations. Sleeping simulation is achieved by having the nodes only transmitting when data are required or retrieved. At other times, no energy is used.

Guerrero [23] developed a simulation model of a WSN with a complex systems approach to study congestion control operations. Testbeds were used for protocol evaluations. Guerrero [23] considered that this viewpoint provided an alternative way to understand the underlying processes that takes place in wireless sensor deployments. Data packets were regarded as the active entities or agents, which travel across the network to reach an appointed node called the "sink".

Our model is different in that we use NetLogo agent-based modeling and define the distance from the sink and distance from the source to minimize the system's energy consumption. Guerrero [23] did not use energy tracking and used a different simulation model.

In his paper, Wan [24] proposed a gradient mode which is based on one-phase directed diffusion (DD) [19], which considers the residual energy of the node and the number of hops from the sink to an intermediate node synchronously. The model is designed to achieve a balance of energy consumption and real-time traffic. A mathematical algorithm is used, and the gradient is calculated. In our paper, we set up the gradient or hop count in a NetLogo model rather than mathematically. In Wan's study, the sink sends a message to all nodes to identify their distance from the sink and the source is selected to be in the upper right corner, whereas in our model the source can be anywhere in the network. Unlike in Wan's study, where the average energy is higher compared to DD, in our paper average energy is reduced compared to DD.

2.2. Introduction to Complex Adaptive Systems (CASs)

We consider our ad hoc environment as a CAS. A CAS is a system made up of many individual parts or agents. The individual parts, or agents, in a CAS follow simple rules. There is no leader or individual who is coordinating the actions of others. The interactions of the agents leads to generation of emergent patterns.

1. Many simple agents or items compared to the entire system.
2. Nonlinear exchange among components—communication.
3. No central control.
4. Emergent behaviors;
   (a) Hierarchical organization;
   (b) Data processing—computation;
   (c) Dynamic—changing behavior;
   (d) Evolution and learning—adapting frequently;
   (e) Uncertainty;
   (f) Unordered.

Adaptation in complex systems is a basic concept that considers the ability of a system to adjust and evolve in response to changes in its environment or internal conditions. The theory of adaptation is a key component of the broader framework of complex adaptive systems (CASs). Here are some key aspects of the theory of adaptation in complex systems:

- Dynamic environment: Complex systems exist in dynamic environments where the conditions are the continually changing structures and behaviors of systems. Adaptation is essential for a system to cope with these changes effectively. The environment can include both external factors, such as climate or market conditions and stock prices, and internal factors, such as the behavior of other agents within the system. Examples include the study of planetary dynamics.
- Learning and memory: Adaptation often involves learning from experience and the ability to retain information over time. Agents within a complex system may adjust their behavior based on past outcomes, forming a kind of "memory" that allows them to make more informed decisions in the future.
- Feedback mechanisms: Feedback loops play a crucial role in adaptation. Positive feedback loops can reinforce successful strategies or behaviors, leading to adaptation and evolution. On the other hand, negative feedback loops can function as stabilizing forces, helping to regulate and maintain the system within certain bounds.
- Variation and diversity: Adaptation is facilitated by the presence of variation and diversity within the system. In a population of agents, having diverse strategies or traits increases the likelihood that at least some individuals will be well suited to changing conditions. This diversity provides the raw material for natural selection and adaptation.
- Robustness and resilience: Adaptive systems tend to be robust and resilient, meaning functionality can be maintained and recovery achieved from disturbances. The ability to adapt allows a system to absorb shocks, navigate uncertainties, and continue functioning in the face of changing circumstances.

- Evolutionary processes: Adaptation in complex systems often involves evolutionary processes. Over time, successful strategies or traits may become more prevalent in the system, while less successful ones may decline. This process of "survival of the fittest" is a fundamental mechanism of adaptation and evolution.
- Decentralised nature: In many complex systems, adaptation is a decentralised process. Individual agents or components within the system often have the autonomy to adapt based on local information and feedback. The collective behaviour of these adaptive agents then gives rise to emergent system-level patterns.

2.3. Why Our System Is a CAS

We start with a network of nodes with no knowledge of their surroundings. The nodes have starting energy. We create a fixed sink and a random source. The nodes communicate with their neighbors from the sink and identify their location in relation to the sink. Likewise, the nodes communicate with their neighbors from the source and identify their location in relation to the source. So, the nodes have dynamically gathered information on their location. The sink generates an interest and computes the shortest distance to the source using data generated from the source. When the source determines that the data requested by the sink is the data held by the source an event is generated and the shortest distance to the sink is computed using distance from the sink. As the search and retrieval occurs, energy is used by the nodes receiving and transmitting data. Less energy is required to search for the source as a smaller data package is used. The message from the source is larger and uses more energy as data are transmitted to the sink.

When the data are received by the sink a new source is generated with different information. The source now communicates with its neighboring nodes and identifies their location in relation to the source. The system is dynamic in that changes occur such as the source location, and information is communicated by transmission of interest and event packets. The locations of the sink and source are computed. The system adapts to the changing source location.

2.4. Measures of Complexity Include the Algorithm's Complexity

1. Disorganized complexity includes large numbers of variables and uses statistical data.
2. Organized complexity has a moderate number of variables and involves dealing simultaneously with many interrelated factors in a whole.
3. The NetLogo software V6.3.0 is used to study complexity.
4. We may regard the present state of the universe as an effect of its past and the cause of its future.
5. We set up a base station we call a sink. The sink can request information. The node network has source information.

3. What We Have Investigated

3.1. The Simulation

The nodes/sensors are located anywhere in that network. The communications network is a "mesh" where a node forwards information from other nodes.

Data are transferred from node to node. The node can be a receiver or transmitter. Nodes that have information are called sources. Nodes that require information from other nodes are called sinks. The network has at least one sink and can have one or more sources. The sink acts as the forwarder to the external user network. Two or more sinks, or mobile sink models can be investigated in future research with our models. Information collection is a two-stage process initiated by the sink.

1. A sink needs some specified information, called an interest. A diffusion is initiated into the network of requests for that interest.
2. A source that can satisfy a received interest, starts a process of transmitting that interest back to the sink.

Each node can only communicate with its direct neighbors according to the squared energy consumption rule, where the energy required for transmission is proportional to the square of the distance between the nodes. Energy consumption is proportional to the length and repetition of the data packets. Energy consumption is proportional to the square of the number of interest packets propagated. We can enhance our network through nodes replenishing their energy-by-energy harvesting.

In this model, a gradient is developed from the source to determine the distance of each node from the source. Similarly, a gradient is developed from the sink to determine the distance of each node from the sink. To search the data the distance from the source is used to come closer to the source without using a random search. The interest packets follow the distance from the source. When the source is detected an event pack is created and the event packet follows the distance from the sink to reach the sink. A significant energy saving is achieved as no random search is undertaken.

3.2. Proposed Methods

We develop a NetLogo model of adapted diffusion providing energy to the nodes for use in data transfer.

Our system is a CAS as we start with a network of nodes with no knowledge of their surroundings. The nodes have starting energy. We create a fixed sink and a random source. The nodes communicate with their neighbors from the sink and identify their location in relation to the sink. Likewise, the nodes communicate with their neighbors from the source and identify their location in relation to the source. The nodes dynamically gather information on their location. The sink generates an interest and computes the shortest distance to the source using data generated from the source. When the source determines that the data requested by the sink is the data it has, it generates an event and computes the shortest distance to the sink using the distance from the sink. As the search and retrieval occurs energy is used by the nodes receiving and transmitting data. Less energy is required to search for the source as a smaller data package is used. The message from the source is larger and uses more energy as it transmits the data to the sink. When the data are received by the sink a new source is generated with different information. The source now communicates with its neighboring nodes and identifies their location in relation to the source. The system is dynamic in that changes occur such as source location, and information is communicated by transmission of interest and event packets. The locations of the sink and source are computed. The system adapts to the changing source location.

3.3. Simulation/Test Environment

The adaptation that occurs within our network is the changing source gradient.

The idea is to create something quite different to the previous three models. The first model was DD in a NetLogo environment. We created this model for comparison purposes. The second model, lazy diffusion, uses random transmission of interest and random retrieval of data.

Our model description is shown in Figure 1 and Algorithm 1: Setting up the search for the event and return to sink.

In each case, when a new source is generated an interest packet is generated to search for the source using the gradient generated from the source, as explained in Figure 2.

This LD model statistically provides energy savings in the early searches, but the energy usage is dependent on the likelihood of reaching the source. The third model, GD uses a gradient from the sink for returning the event packet. Less energy is used with this model but not all runs reach the source. The challenge in the previous models is modeling a changing source location for optimizing the search for the source. In adapted diffusion, we created a gradient from the source that changes for each run, achieving significant energy savings and increasing the number of messages retrieved. We are working on the idea of creating gradients from both the sink and each of the sources. There is no reason to not have multiple gradients. That is, each node has a value for each gradient, which is the hop count from that source or sink.

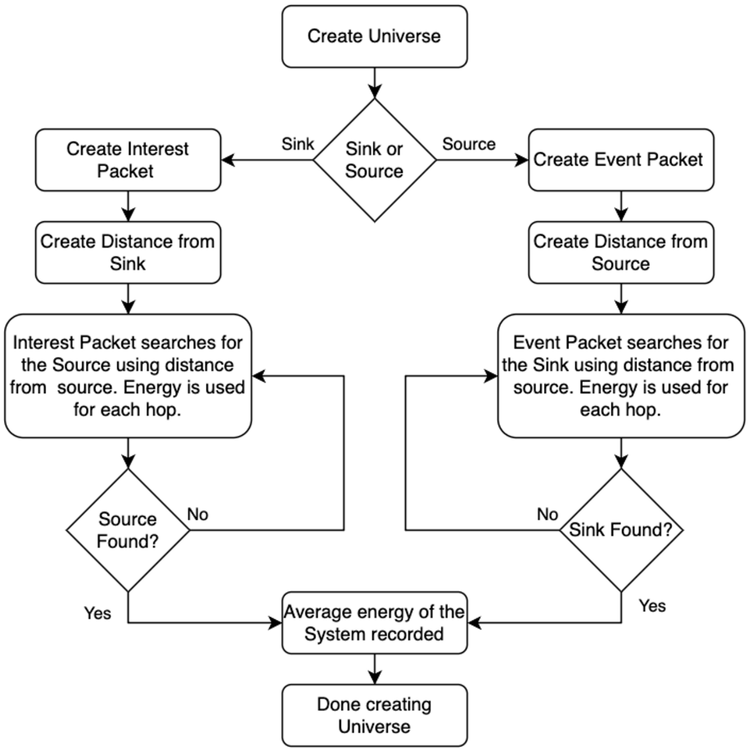


Figure 1. Setting up the search for the event and return to sink.

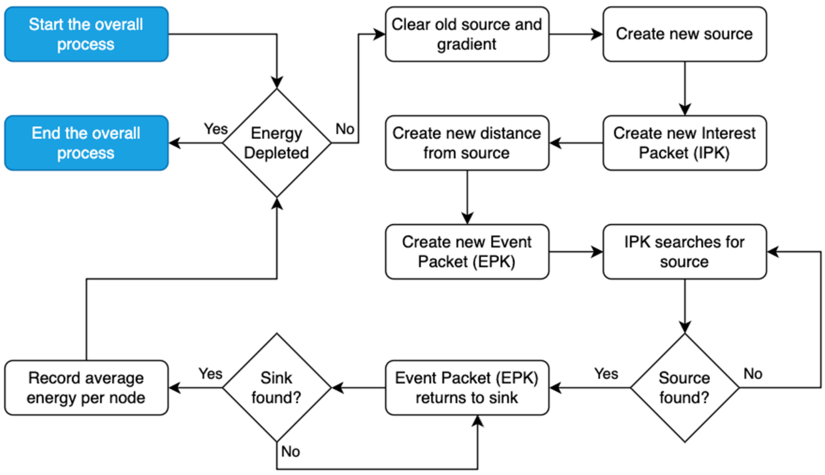


Figure 2. Repeat search for data.

The nodes adapt to learn the distances of their neighbors from the sink and the source. This saves energy while searching but uses energy in creating the gradients (trade-offs).

We know that nodes can discover the energy levels of neighbors. Interest packets traverse by moving up the gradient that is the shortest distance from the source, and then, moving event packets down the gradient that is the shortest distance to the sink from the appropriate source.

When information is requested the generated interest packet approaches the source using the distance from source and selects one of the neighboring nodes closest to the source until the source is reached.

When the interest reaches the source, the event packet with the required information returns the requested data to the sink by moving to a neighbor closer to the sink until the sink is reached.

In Figure 2, we describe the process for repeating the transmission with a new source to search for and retrieve the data. In each case, when a new source is generated an interest packet is generated to search for the source using the new gradient generated from the source, as explained in Figure 2.

When the information reaches the sink, a new source is created with new information, with a new gradient from the new source. When a new request is generated, the process is repeated.

Each time data are transferred as interest packet energy of transmission is depleted from the node used. Higher energy is used to return the data as the event packet has more data.

The gradient we develop is the distance from the source and sink of the nodes in the system.

3.4. Comparison to DD

In our experiments, we considered event packets to be much larger than interest packets, so the energy choices may be quite different. We used twice the energy compared to searching for the event in retrieval of the data from the source. Choices can be made by inspecting energy levels.

In DD, each transmission involves all the nodes, hence all the nodes lose energy when receiving or transmitting data. In our model, only selected nodes are used for data transfer.

We demonstrate that after the first search a new source and gradient from the source is generated and a new search is progressed. Energy is used for the searches and data return. The average energy is recorded for the system.

For clarity, we have divided up the overall process into three "algorithm" blocks (Algorithms 1–3). When running the simulation, the "blocks" are executed sequentially.

Algorithm 1: Setting up the world.

```
Initialization;
while NumNodes=Zero do
    Input: NumNodes
end
/* Set up the size of the World                                    */
Output: size = √NumNodes With X and Y axis
/* Assume a square World                                           */
repeat
    Set initial energy in the nodes
until Until all nodes have initial energy set;
/* Set Global variables                                            */
while Global variables not set do
    /* Set Minimum Energy per Node                                 */
    Input: MinEnergy
    /* Set the energy needed to search for the Source              */
    Input: Energy-cost-to-search-source
    /* Set the energy needed to return the event data to the Sink  */
    Input: Energy-cost-to-return-event
end
```

Algorithm 2: Run the model once.

```
/* Set up the Sink                                              */
while No Sink set do
    Location = return 0,0 ;
    set Sink at Location
end
/* Set up the Source                                            */
while No Source set do
    set X to random(0,size) set Y to random(0,size) set Source at X,Y
end
while No intetrest set do
    load Interest Packet to Sink (ipk)
end
while A node does not have Gradient set do
    set Gradient from Sink = number of Hops from Sink
end
while No event in Source do
    load Event into Node (ipk)
end
while Source not found do
    search ipk uses distance from Source to search for Source
end
while Event not returned to Sink do
    search epk uses distance from Sink to search for the Sink
end
```

Algorithm 3: Start a new search and repeat.

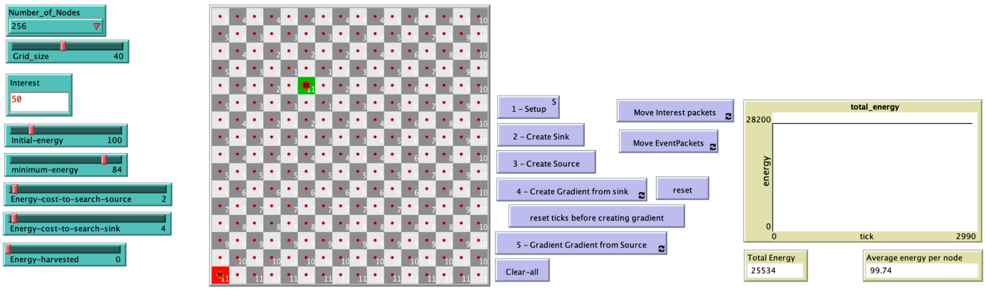
```
begin
    clear Interest /* A new simulation needs a new Interest     */
    clear Source /* A new simulation needs a new Source         */
    clear Distance from Source /* Needed for the new simulation */
    create new Source /*                                        */
    create new Gradient – distance from source /*               */
    create new Interest /*                                      */
    create new ipk and epk /*                                   */
end
while Source not found do
    search ipk uses distance from Source to search Source
end
while Sink not found do
    search epk uses distance from Sink to search Sink
end
record Average energy per node
```

3.5. Directed Diffusion

In this protocol, the interest is propagated to all nodes, using energy in all nodes. As there is no global information available to the Nodes, the one that has the event is unable to signal to the other nodes that the event has been found. The other nodes simply continue diffusing the interest until all nodes have been reached. This results in a general depletion of energy when that is not needed.

3.6. Adapted Diffusion

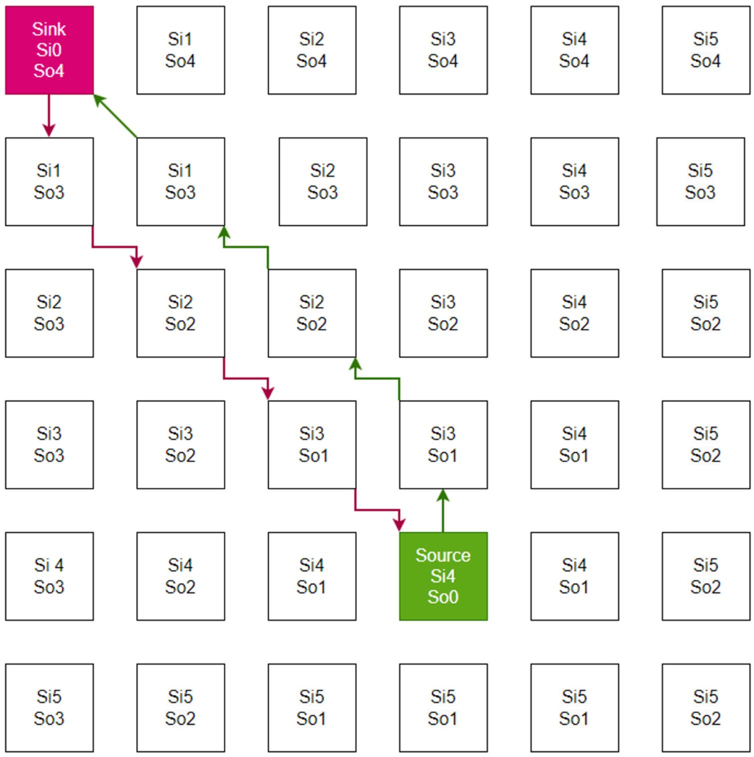
We modified DD to build a gradient from the sink to allow all nodes to identify their distance from the sink. The source node also builds a gradient from its location to allow the nodes to identify their location in relation to the source. The nodes have now adapted to know their location in relation to the sink and the source. Unlike our previous models, now the interest packet generated at the sink searches for the source using the distance from source rather than searching randomly. In the old models, a random search energy is used without necessarily reaching the source; also, the same nodes may be visited multiple times. By using the distance from the source each move comes closer to the source and the source is always found. When the source is found, an event packet is generated, rather than a random search distance from the sink being used to reach the sink. This technique saves energy when compared to our previous models and DD. The NetLogo interface we programmed and used for our AD is shown in Figure 3.



(**a**) NetLogo interface (top part) (**b**) NetLogo interface (bottom part)

Figure 3. NetLogo interface.

The search progresses as shown in Figure 4.



$$S_i \quad = \quad Distance\ from\ Sink$$
$$S_o \quad = \quad Distance\ from\ Source$$

Figure 4. Adapted diffusion network with distances.

4. Results

Our model was run as shown in Figure 5. When the interest reaches the source, the event packet with the required information returns the requested data to the sink by moving to a neighbor closer to the sink until the sink is reached.

1. A gradient is formed from the source, identifying the distance from the source (So).
2. A gradient is formed from the sink, identifying the distance from the sink (Si).
3. The sink searches the source using the distance from the source.
4. When the source is reached the source sends the message to the sink by using the distance from the sink.
5. When the sink is reached, the average energy is recorded as well as the run number.
6. The model is run till the energy level is too low to receive or transmit data.

The energy was compared to our previous model of DD. A graph of the data was generated and is presented in Figure 5.

Extracts from the data are presented in Table 1. The average energy comparison of DD and AD clearly shows significantly reduced energy loss with adapted diffusion compared to DD. The data presented above indicates that more runs are possible with AD compared to DD.

Table 1. Average energy usage comparison of DD and AD.

Model	DD: Average Energy per Node	AD: Average Energy per Node
After 10 runs, search 2 energy units, return 4 energy units	80.1	96.4
After 10 runs, search 6 energy units, return 12 energy units	43.7 @ 9, Fail @ 10	90.9
After 20 runs, search 2 energy units, return 4 energy units	57.9	92.4
After 19 runs, search 6 energy units, return 12 energy units	Fail	81.1

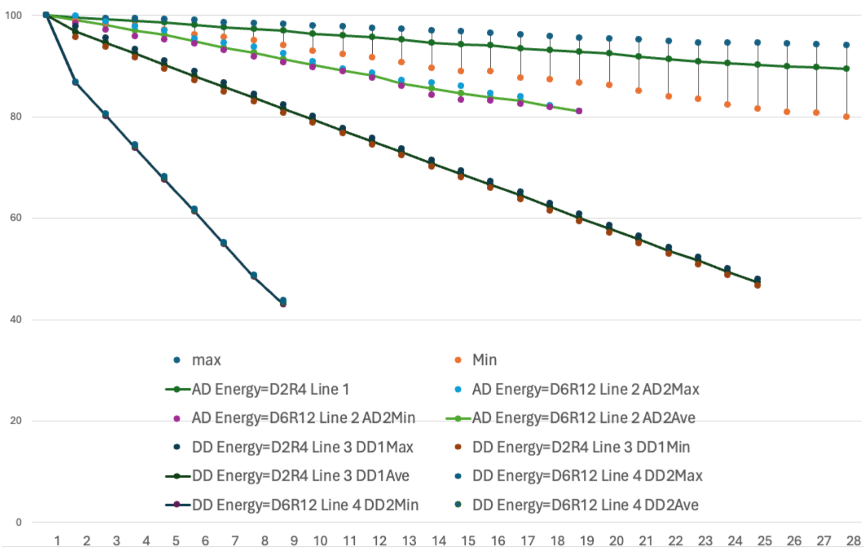


Figure 5. Average energy of directed diffusion vs. adapted diffusion. Energy to search 2 units, and energy to return 4 units. Energy to search 6 units, and energy to return 12 units. The vertical lines are the spread, calculated as standard deviation of the values recorded in each case. The length of each line indicates the standard deviation observed.

5. Discussion

The results are achieved by setting up energy levels for the nodes and the amount of energy required for sending an interest and the energy required for retrieving the event to the sink. After a run, the average energy is recorded for the network. The source is removed and a new source created, then the process is repeated. Running the model led to generating the results presented in Figure 5.

Two energy parameters were selected for running the model and to generate the data in Figure 5 as follows:

1. Energy to search 2 units for interest movement and 4 units for event packet, energy to return source data;
2. Energy to search 6 units for interest movement and 12 units for event packets, energy to return source data.

Higher energy is usually required for transmitting a bigger packet of data from the source, hence there is a higher value for returning data. A smaller package of data is used for searching the source.

An extract of the data from Figure 5 is presented in Table 1 for clarity.

The results clearly indicate that AD leads to significantly less energy usage and extended life of the network. At an energy usage of 2 for searching for the source and 4 for returning data to the sink, for DD after 10 repeat searches the average energy was 81 units, compared to 97 units for AD.

Figure 5 and Table 1 indicate that at an energy usage of 6 for searching for the source and 12 for returning data to the sink, for DD after nine repeat searches the system failed and no more transmission was possible; however, more than double the number of searches was possible with adapted diffusion.

6. Conclusions

We compared a newly developed complex NetLogo model we call adapted diffusion to a NetLogo model we developed with DD. We demonstrated that our new AD model is more energy efficient and can achieve a longer network life than DD. The model is flexible in that different energy levels can be evaluated easily and energy harvesting experiments can be conducted to further studies of diffusion models. Multiple sinks and sources can be introduced and studied.

Author Contributions: Conceptualization, G.H. and R.B.; methodology, G.H. and R.B.; NetLogo software, G.H.; validation, G.H., J.L. and R.B.; formal analysis, G.H. and R.B.; investigation, G.H., J.L. and R.B.; writing—original draft preparation, G.H.; writing—review and editing, G.H., J.L. and R.B.; supervision, R.B. and J.L.; All authors have read and agreed to the published version of the manuscript.

Funding: The first author was in receipt of a Commonwealth of Australia-funded Research Training Scholarship.

Data Availability Statement: Data is contained within the article.

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

References

1. Hakim, G.; Braun, R. Wireless Sensor Network Routing for Energy Efficiency. In Proceedings of the 28th International Conference on Systems Engineering, ICSEng 2021, Wroclaw, Poland, 14–16 December 2021; Springer: Berlin/Heidelberg, Germany, 2022; pp. 329–343.
2. Intanagonwiwat, C.; Govindan, R.; Estrin, D. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, MA, USA, 6–11 August 2000; pp. 56–67.
3. Lipman, J.; Boustead, P.; Chicharo, J.F.; Judge, J. Resource aware information dissemination in ad hoc networks. In Proceedings of the 11th IEEE International Conference on Networks (ICON2003), Sydney, Australia, 28 September–1 October 2003.
4. Tang, J.; Pan, Q.; Chen, Z.; Liu, G.; Yang, G.; Zhu, F.; Lao, S. An Improved Artificial Electric Field Algorithm for Robot Path Planning. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *60*, 2292–2304. [CrossRef]

5. Guo, H.; Shen, C.; Zhang, H.; Chen, H.; Jia, R. Simultaneous trajectory planning and tracking using an MPC method for cyber-physical systems: A case study of obstacle avoidance for an intelligent vehicle. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4273–4283. [CrossRef]
6. Ma, Y.N.; Gong, Y.J.; Xiao, C.F.; Gao, Y.; Zhang, J. Path planning for autonomous underwater vehicles: An ant colony algorithm incorporating alarm pheromone. *IEEE Trans. Veh. Technol.* **2018**, *68*, 141–154. [CrossRef]
7. Cai, W.; Zhang, M.; Zheng, Y.R. Task assignment and path planning for multiple autonomous underwater vehicles using 3D dubins curves. *Sensors* **2017**, *17*, 1607. [CrossRef] [PubMed]
8. Xiong, C.; Chen, D.; Lu, D.; Zeng, Z.; Lian, L. Path planning of multiple autonomous marine vehicles for adaptive sampling using Voronoi-based ant colony optimization. *Robot. Auton. Syst.* **2019**, *115*, 90–103. [CrossRef]
9. Mahmoud Zadeh, S.; Powers, D.M.; Sammut, K.; Yazdani, A.M. A novel versatile architecture for autonomous underwater vehicle's motion planning and task assignment. *Soft Comput.* **2018**, *22*, 1687–1710. [CrossRef]
10. Zhu, D.; Cao, X.; Sun, B.; Luo, C. Biologically inspired self-organizing map applied to task assignment and path planning of an AUV system. *IEEE Trans. Cogn. Dev. Syst.* **2017**, *10*, 304–313. [CrossRef]
11. Zhu, D.; Cheng, C.; Sun, B. An integrated AUV path planning algorithm with ocean current and dynamic obstacles. *Int. J. Robot. Autom.* **2016**, *31*, 382–389. [CrossRef]
12. Chen, Y.B.; Luo, G.C.; Mei, Y.S.; Yu, J.Q.; Su, X.L. UAV path planning using artificial potential field method updated by optimal control theory. *Int. J. Syst. Sci.* **2016**, *47*, 1407–1420. [CrossRef]
13. Abdullah, A.M. An efficient energy-aware and reliable routing protocol to enhance the performance of wireless body area networks. *J. Supercomput.* **2024**, 1–26. [CrossRef]
14. Kaminski, N.J.; Murphy, M.; Marchetti, N. Agent-based modeling of an IoT network. In Proceedings of the 2016 IEEE International Symposium on Systems Engineering (ISSE), Edinburgh, UK, 3–5 October 2016; pp. 1–7. [CrossRef]
15. Hamzi, A.; Koudil, M. Agent based platform for the design and simulation of wireless sensor networks. In Proceedings of the 2012 International Conference on Computer, Information and Telecommunication Systems (CITS), Amman, Jordan, 14–16 May 2012; IEEE: New York, NY, USA, 2012; pp. 1–5.
16. Kout, A.; Labed, S.; Chikhi, S. Netlogo Agent based tool for Modeling and Simulation of Routing Problem in Ad hoc Networks. In Proceedings of the Second International Conference on Advances in Information Processing and Communication Technology-IPCT 2015, Rome, Italy, 11 December 2015; Institute of Research Engineers and Doctors: New York, NY, USA, 2015; pp. 154–160.
17. Gupta, S.K.; Tyagi, S.; Kumar, S. Dual-Hop Direct Diffusion Routing Protocol for Energy-Efficient Wireless Sensor Network. In *Futuristic Trends in Networks and Computing Technologies*; Springer: Singapore, 2022; pp. 349–362.
18. Mu, J.; Yi, X.; Liu, X.; Han, L. An efficient and reliable directed diffusion routing protocol in wireless body area networks. *IEEE Access* **2019**, *7*, 58883–58892. [CrossRef]
19. Silva, F.; Heidemann, J.; Govindan, R.; Estrin, D. Directed diffusion. In *Distributed Sensor Networks*; Tech. Rep. ISI-TR-2004-586; University of Southern California: Los Angeles, CA, USA, 2004.
20. Xia, L.; Chen, X.; Guan, X. A new gradient-based routing protocol in wireless sensor networks. In Proceedings of the Embedded Software and Systems: First International Conference, ICESS 2004, Hangzhou, China, 9–10 December 2004; Revised Selected Papers 1; Springer: Berlin/Heidelberg, Germany, 2005; pp. 318–325.
21. Talib, M.S. Minimizing the energy consumption in wireless sensor networks. *J. Babylon Univ. Appl. Sci.* **2018**, *26*, 17–28. [CrossRef]
22. Wilensky, U.; Rand, W. *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*; MIT Press: Cambridge, MA, USA, 2015.
23. Guerrero, P.E.; Gurov, I.; Santini, S.; Buchmann, A. On the selection of testbeds for the evaluation of sensor network protocols and applications. In Proceedings of the 2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Darmstadt, Germany, 16–19 June 2013; IEEE: New York, NY, USA, 2013; pp. 495–499.
24. Wan, J.; Wu, J.; Xu, X.; Yan, Y. An Efficient Gradient Mechanism of Directed Diffusion in Wireless Sensor Network. In Proceedings of the 2008 International Conference on Computational Intelligence and Security, Suzhou, China, 13–17 December 2008; Volume 1, pp. 427–431. [CrossRef]