Dynamic ring-based forwarder selection to improve packet delivery in ultra-dense nanonetworks

Farah Hoteit
FEMTO-ST Institute, Univ. Bourgogne Franche-Comté, CNRS
Montbéliard, France
farah.hoteit@univ-fcomte.fr

Dominique Dhoutaut
FEMTO-ST Institute, Univ. Bourgogne Franche-Comté, CNRS
Montbéliard, France
dominique.dhoutaut@univ-fcomte.fr

Winston K.G. Seah
School of Engineering and Computer Science, Victoria University of Wellington
Wellington, New Zealand
winston.seah@ecs.vuw.ac.nz

Eugen Dedu
FEMTO-ST Institute, Univ. Bourgogne Franche-Comté, CNRS
Montbéliard, France
eugen.dedu@univ-fcomte.fr

ABSTRACT

Nanotechnology is a revolutionary field of science and the design of nanometer-sized devices opens the door to a wide range of novel applications. Electromagnetic nanonetworks are networks of nanodevices communicating in the terahertz band. Nanonetworks can be ultra-dense, which makes it a very challenging environment for traditional routing protocols. In face of extreme density, they tend to either select too many forwarders or devote too many resources to find a small and optimal subset.

Selecting too many forwarders means that the channel will be encumbered by many copies of the same packet and a lot of power will be drained. On the other hand, trying to select a small and optimal subset of forwarders incurs an initially prohibitive computational and communication overhead. Therefore, we previously proposed a ring mechanism that can be applied under existing protocols and optimize their behavior.

However, the proposed ring had a fixed width, which was manually set and did not adapt to the local density, an important parameter in heterogeneous networks. In the current article we solve this problem by automatically selecting the ring width based on the local node density. Extensive simulations of our scheme applied to four routing protocols, using a dense nanonetwork simulator, show a dynamic ring that drastically reduces the number of forwarders used for transmission in the network, without sacrificing the packet delivery ratio and thus optimizing the network usage.

KEYWORDS

Routing, Congestion, Nanonetwork, Dense network, Scalability

ACM Reference Format:

Farah Hoteit, Dominique Dhoutaut, Winston K.G. Seah, and Eugen Dedu. 2022. Dynamic ring-based forwarder selection to improve packet delivery in ultra-dense nanonetworks. In *The Ninth Annual ACM International Conference on Nanoscale Computing and Communication (NANOCOM '22), October 5–7, 2022, Barcelona, Spain.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3558583.3558849

NANOCOM '22, October 5–7, 2022, Barcelona, Spain

ACM ISBN 978-1-4503-9867-1/22/10…$15.00
https://doi.org/10.1145/3558583.3558849

1 INTRODUCTION

Nanotechnology permits the design of integrated devices at the nanoscale and inspires novel applications that can connect to the Internet of Nano-Things (IoNT). The communications among nanodevices can be electromagnetic, molecular, acoustic or mechanical nanocommunications. This article focuses on electromagnetic nanonetworks, where nanonodes can radiate signals in the terahertz band (0.1–10 THz) using graphene antennas [8].

Electromagnetic nanonetworks are used in various fields. Applications in nanomedicine include health monitoring systems and Drug Delivery Systems (DDSs). Health systems use nanosensors for monitoring different concentration levels of molecules in the blood and for detection of infectious intra-body agents. DDS uses nanoactuators for delivering nanoparticles and drugs into the body [7]. In Software-Defined Metamaterials (SDMs), nanonetworks and metamaterials (artificial structures with un-natural properties) are combined, allowing the user to send commands to nanodevices to perform geometrically-altering actions on the metamaterial and tuning of its electromagnetic behavior [12]. Other applications include wireless robotic materials, industry, military and agriculture.

Nanonetworks are significantly differ from traditional networks, albeit sharing some characteristics with wireless sensor networks (WSNs). First of all, nanodevices face extreme hardware limitations - transceiver, memory, processor, power unit, etc. Therefore, nanodevices have a very small communication range and are only able to execute very simple protocols (no matrix multiplication or maintaining long lists of neighbours for example). Due to energy scarcity and hardware simplicity, nanonodes cannot use carrier signals like traditional networks; a lightweight modulation was proposed, Time Spread On-Off Keying (TS-OOK), where bit 1 is sent as a 100 femtosecond-long (= T_p) pulse with energy, and bit 0 is defined as a silence without energy consumption [9]. On top of those hardware limitations, nanonetworks can be much denser than WSNs, e.g., thousands of neighboring nodes. This makes traditional routing schemes for WSNs not directly applicable in nanonetworks, for example Ad-hoc On-Demand Distance Vector Routing (AODV) has been modified into a hierarchical version to make it efficient in nanonetworks [18]. To conclude, it is necessary to design new lightweight and highly scalable forwarding and routing protocols, to guarantee successful packet delivery with a lower overhead.

In this context, we consider the following problem. In a nanonetwork, a source node either sends a message to all other network nodes (flooding) or to one destination node in a multi-hop fashion. In a dense network this generates an avalanche of forwarders, leading to collisions, congestion and useless energy consumption. We look for methods to reduce the number of forwarding nodes, while still achieving a successful message delivery to the intended destination(s).

To solve this problem, we previously proposed a ring-based forwarder selection in ultra-dense networks, which is an algorithm implemented under routing schemes [6]. This algorithm improves the routing by selecting the forwarders among ring neighbours only, and not among all neighbours. This is done by limiting the forwarding area to a ring at the border of the communication range, using two control packets. The controls are sent to different communication ranges only once, right before the very first transmitted data packet. Note that there is no Global Positioning System (GPS)-like module in nanonodes allowing them to compute distances between two nodes.

However, the limitation of the proposed ring is that it is static and manually set. A static ring width contains more forwarders in a dense environment compared to a non-dense environment. A large number of forwarders lead to congestion, while a small number of forwarders cause packet loss.

To overcome this limitation, we hereby propose an efficient extension of the ring-based forwarder selection. This scheme, *dynamic ring*, dynamically adapts the ring width to the local density. We evaluate it and show that it improves the routing by better selecting the forwarding nodes and by reducing their number while keeping a successful packet delivery.

The article is organized as follows: Section 2 presents the related work. Section 3 discusses the dynamic ring scheme. Section 4 evaluates the dynamic ring using extensive simulations. Section 5 draws some conclusions.

2 RELATED WORK

The routing schemes applicable to nanonetworks need to be simple because constrained nanodevices cannot store nor process large routing tables, neighbouring information or complex network knowledge in general. In this section, we present some schemes aiming to reduce the number of forwarders, for a detailed view the reader is directed to [22].

2.1 Flooding schemes

In pure flooding, every node in the network forwards the data packet that it receives for the first time. This flooding is not scalable and results in redundant transmissions and broadcast storms in dense environments. In probabilistic flooding [16], nodes forward packets with a static probability *proba* and discard it otherwise. The probability should be carefully chosen depending on the scenario, in order to guarantee the message delivery with a minimal number of forwarders.

Backoff flooding [2] is a highly efficient flooding scheme, where the number of forwarders is notably reduced. Only nodes receiving few copies of data packets (less than redundancy r) forward the packet. The count of data copies is done in a time window proportional to the number of neighbours, estimated using the Density Estimator for Dense Networks (DEDeN) [1].

In the Lightweight Self-tuning Data Dissemination for dense nanonetworks (LSDD) [21], nodes classify themselves as forwarders or non-forwarders (passive auditors) using packet-receive statistics (including success or failure on packet reception and integrity checks).

2.2 Unicast (or merely zone-cast) schemes

In RADAR routing [14], the nanonetwork is a circular area and a central entity emits radiation at an angle. Nanonodes found inside the angle of radiation are in the ON-state, and all the other nodes are in the OFF-state. RADAR only consists in blind flooding inside the angle of radiation. Nevertheless, a large angle can still cause a broadcast storm, and the destination node must be in ON state to receive the packet.

The Deployable Routing System for Nanonetworks (DEROUS) establishes a point-to-point communication in circular and radial paths. In the setup phase, a beacon node set at the center of a 2D circular area sends a packet, and nanonodes self classify as infrastructure (always retransmit) or user (never retransmit) using packet reception quality. During the routing, nanonodes that are between the source and destination become forwarders to finally form a circular or radial routing path. Thus, DEROUS limits the retransmissions to a circle area [11].

In Coordinate and Routing system for Nanonetworks (CORONA), the network is a 2D rectangular area with *uniformly* distributed nanonodes and four anchor nodes placed at each corner. During the setup phase, anchors transmit their packets in sequence, allowing nanonodes to set their coordinates as hop counts from the anchors. During the routing phase, only nanonodes that are in a *rhombus area* between source and destination nodes forward the packet [20]. The Stateless Linear-path Routing (SLR) [19] extends CORONA to a 3D cubic space, and selects as forwarders only the nodes that are on a *line* between the source-destination pair.

All these protocols may benefit from further enhancements by limiting the number of forwarders at the Medium Access Control (MAC)-level while still maintaining packet delivery to the destination(s). For this, we previously proposed the ring that is presented in the following.

2.3 Basic ring as a forwarder selection scheme

The basic ring-based forwarder selection algorithm [6] focuses on limiting the area of forwarding to the ring near the border of the communication range, as shown in Fig. 1. A node requires three conditions to be a forwarder: it has received the data packet for the first time, it is on the ring, and it is selected as forwarder by the routing protocol implemented on top of the ring. Therefore, it is not the whole ring that transmits, but only the forwarders that confirm the previous conditions.

The ring is bound by two communication ranges corresponding to two control packets sent at different transmit powers. These two

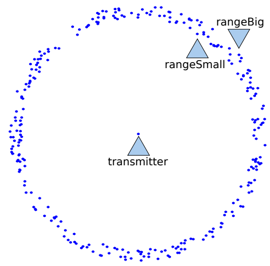

Figure 1: Basic ring algorithm: the transmitter is at the center of the communication range and only nodes in the ring area between rangeBig and rangeSmall are potential forwarders.

control packets are sent *only once*, before the first data packet to be forwarded by a node. Control packets may be retransmitted if the network topology changes. Nodes that receive the high-power control packet and do not receive the low-power control packet are considered on the ring and become candidate forwarders. The only assumption made in this scheme is that a nanonode can send packets at different transmit powers. This scheme is similar to greedy forwarding schemes [4, 10] using distance-from-neighbours metric, which is unavailable in our (resource-constrained) dense nanonetwork context.

3 DYNAMIC RING SCHEME

This paper proposes an enhancement of the ring algorithm, called the *dynamic ring*, which makes nodes configure the ring width automatically, based on a desired number of local ring neighbours and the local density. It is presented in the following.

The basic ring algorithm uses a static ring width for all forwarding nodes in the network, with the region of the ring lying between two ranges: *rangeBig* and *rangeSmall*. The dynamic ring scheme aims to *automatically* set the ring width for every forwarder. In the new scheme, *rangeBig* is kept fixed and equal to the communication range to make the forwarding progress faster, while *rangeSmall* varies and makes the ring thinner or thicker. The challenge is to find the appropriate *rangeSmall* value. This value is inferred from the desired number of ring neighbours per hop, denoted by N.

Finding the *optimal* (minimal) set of forwarders among the 1-hop neighbours is an NP complete problem. A recent research [13] finds the distributed minimum of multipoint relays (MPRs) selection in dense mesh networks. However, their definition of a dense network (up to 150 nodes) is still low compared to our own scenario (10 000 nodes), and as the method requires powerful hardware for mesh nodes and a sufficiently stable network, it cannot be applied in our context. The optimal value also depends on the routing protocol used and the desired redundancy. Finding the optimal value for various applications is out of the scope of this article. For now, we consider that the network user tests and finds a low, but sufficiently high value of ring neighbours N to guarantee delivery.

An efficient dynamic ring scheme executed by nodes takes into account the network's high density and the nanodevice's hardware constraints. However, one as to be careful when choosing a method to adjust the ring width. In the following, two acknowledgment methods are presented briefly and are shown to not be efficient in nanonetworks. They are followed by a dynamic ring method that is demonstrated to work in nanonetworks in Section 4.

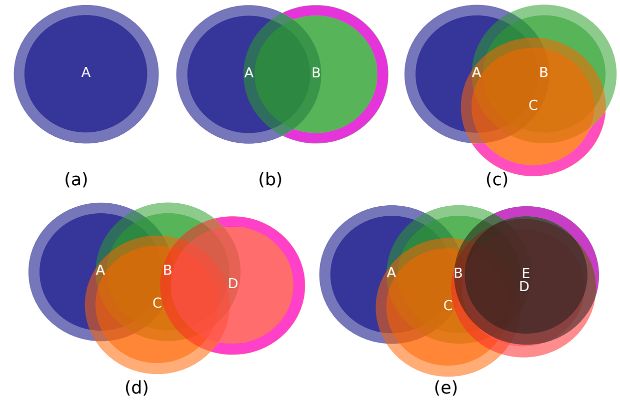

Figure 2: Drawbacks of dynamic ring with implicit ack: only the pink areas include new forwarders that will send the data packets counted as acknowledgements.

First method: dynamic ring with implicit acknowledgment. In this method, a forwarding node starts with any value for *rangeSmall*, and updates it based on the number of acknowledgments received compared to the required number N of forwarders. The considered (implicit) acknowledgments are the copies received by the node from forwarders belonging to its own forwarding ring. If the number of forwarders is equal to N, then the ring width is fine; whereas a smaller number of forwarders requires a thicker ring and vice-versa.

This evident method has two drawbacks. The first one is that N, a unique value for all nodes, only sometimes applies to *partial* rings. These partial rings do not contain "old forwarders", i.e. nodes that have already seen the same data packet before and that will not retransmit it to avoid forwarding loops, as detailed here 4.2. For instance, in Fig. 2 (a), A is the first transmitter and its whole ring, including B, forwards. In Fig. 2 (b), showing the second hop, only the yet uncovered region of the ring of B, i.e. the pink region at right that has not yet received the data packet, will forward and will be counted as acknowledgments. Thus, the ring of B is much smaller than the ring of A, yet the value N erroneously applies for both of them. Subsequent images in the same Fig. 2 illustrate that, as the number of hops increases, the ring (the pink areas), that represent new forwarders at each hop and that are counted as acknowledgments, have smaller and smaller surfaces and number of nodes.

The second drawback is that a nanonode may not able to count the acknowledgments, as they are in the order of hundreds. This is due to the high local density and the hardware constraints. The resource constraints of a nanonode in energy, memory and data processing require the node to use large backoffs before transmission window at the MAC level to avoid further collisions.

Second method: dynamic ring with explicit acknowledgment. To solve the misleading number of acknowledgments given by the previous method, the explicit acknowledgment method makes a ring

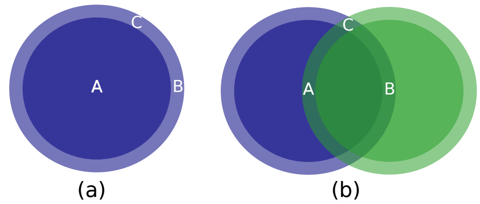

Figure 3: High traffic problem in dynamic ring width with explicit ack.

neighbour (either forwarder or not), after receiving its data packet, generate a new control packet to be received by the transmitter for the acknowledgment counts. Still, this method suffers from another misleading ring neighbours count (instead of forwarders count), where ring neighbours with already seen data packets are ignored. Indeed, as shown in Fig. 3 (a), A is the first transmitter, B and C are ring neighbours of A and so they send their corresponding acknowledgments to it; again in Fig. 3 (b), B is the second transmitter, A and C need to send their corresponding acknowledgments to it, although A has already seen the data. Another problem with this method is the high traffic caused by the additional control packet, wasting energy and generating collisions.

Third and selected method: dynamic ring using node density estimation. The previous two methods have issues, hence they cannot be used to dynamically adjust the ring width. Here, we present the third method, the dynamic ring with DEDeN, which is the method we select for the dynamic ring.

This method takes as input the number N denoting the required ring neighbours per hop. During the network initialization, nodes perform the *local density* estimation using a density estimator. One such estimator is Density Estimator for Dense Networks (DEDeN) [1]. In DEDen, nodes distributively estimate the local density without having to rely on a full (and very costly) exchange of hello packets. Instead, over the course of multiple rounds, they have an increasing probability to send a small probe packet. At each round, they also count the number of probes received. Knowing the sending probability, they can compute with an increasing confidence their number of neighbors.

We recall that *rangeBig* is the communication range. Instead, *rangeSmall* corresponds to the desired number of ring neighbours. Hence, the ratio of ring neighbours N to the neighbours L (= local density above) should be equal to the ratio of the ring to the communication circle:

$$N/L = RingArea/circleArea \tag{1}$$

$$N/L = \pi(rangeBig^2 - rangeSmall^2)/(\pi rangeBig^2) \tag{2}$$

hence

$$rangeSmall = \sqrt{rangeBig^2 - N * rangeBig^2/L} \tag{3}$$

The ring computation is presented in Algo. 1. The rangeSmall value is set directly using a formula. It works also for heterogeneous networks (node densities are different in different parts of the network) and for different propagation models (e.g., unit disc and shadowing models). However, if the topology changes with time, the local density changes too, and thus *rangeSmall* needs to be recomputed during runtime.

4 EVALUATION OF THE SCHEME

This section presents how the dynamic ring with DEDeN is applied to four routing protocols: pure flooding, probabilistic flooding, backoff flooding and SLR, and how they compare to traditional routing protocols.

4.1 Available simulation software

Nano-Sim [15] and TeraSim [5] are *ns3* plug-ins with heavy footprint, lacking scalability (usable up to around one thousand nodes) and thus cannot be used in our context of dense networks.

On the contrary, BitSimulator [3] can simulate hundreds of thousands of nodes that, additionally, can be visualized. Its scalability is confirmed in a study comparing these three electromagnetic nanonetwork simulators [17]. VisualTracer is the visualization tool of a 2D nanonetwork, and will be used in the following evaluation figures. All results in BitSimulator are fully reproducible, due to the use of random seeds RNG for different runs. Therefore, we use BitSimulator in this article.

4.2 Scenarios

The simulation parameters are shown in Table 1. The scenario is a heterogeneous nanonetwork of 10 000 nodes distributed over three horizontal bands, each with a homogeneous density (5500, 3000, and 1500 respectively), in a 2D square area of 36 mm^2. This highly dense scenario corresponds to applications in software-defined metamaterials and in in-body communication, for example.

Nodes have omnidirectional nanoantennas with a default communication range $CR = 1000\,\mu$m and can change the range using a different transmission power (for control packets). The network dimensions along with the communication range result in x/CR = 6 mm / 1 mm = 6 hops in each dimension, which are enough for the routing protocol evaluations. The computed average number of neighbours per node is 906. For more realistic results, the propagation model used is the shadowing, with a 100% packet reception rate at distance $[0, d]$ from the transmitting node, a decreasing packet reception rate from 100% to 0% in the interval $[d, CR]$, and a zero packet reception rate at distance $> CR$, where d is configurable.

RangeBig is set to the default communication range (to increase the forwarding progress), and rangeSmall value is dynamically chosen by nodes depending on the local density, according to Eq. 2.

The packet payloads are random sequences of "1"s and "0"s. The data packet size is 1003 bits and the two control packets sizes are 101 and 102 bits. These values are distinctive so that they can be spotted easily in the output log files.

A source node in the top of the network generates a Constant BitRate (CBR) flow of 50 packets. The source either floods the whole network, or transmits to one destination node (found in the bottom). Since a node sends controls only once before the very first forwarded data packet, the cost of the control packets fades out over 50 data packets.

The dynamic ring proposed algorithm is implemented in three flooding schemes: pure flooding, probabilistic flooding and backoff flooding, and one destination-oriented scheme: SLR. For backoff

Algorithm 1: Dynamic ring width with DEDeN.

```
Data:
N = desired number of ring neighbours per node
L = number of neighbours per node = local density
rangeBig = default communication range
sourceID = identifier of source node (first transmitter node)
transmitterID = identifier of current transmitter node
seqNo = sequence number
ctrlBigSeqNo = seqNo of high-power control packet = 0
ctrlSmallSeqNo = seqNo of low-power control packet = 0
dataSeqNoMap= [sourceID, seqNo]
amIonRingMap = [transmitterID, ctrlBigSeqNo, ctrlSmallSeqNo]
countAckMap = [sourceID, seqNo, rangeSmall]
needToSendControl = true
Result: Find appropriate ring width for forwarders
Upon packet reception (type, sourceID, seqNo,
  transmitterID)
if type is DATA then
    if dataSeqNoMap[sourceID] does not exist OR
     dataSeqNoMap[sourceID] < seqNo then
        // no packet has been seen from this source
           before OR seqNo of this packet is higher
           than the highest already seen seqNo from
           this source
        dataSeqNoMap[sourceID] = seqNo // insert new
           sourceID OR update the highest already seen
           seqNo from this source
        if call amIOnRing AND routing protocol selects me as
         forwarder then
            call forwardDataPackets
        end
    end
else if type is CONTROL-BIG then
    ctrlBigSeqNo[transmitterID] = seqNo
else if type is CONTROL-SMALL then
    ctrlSmallSeqNo[transmitterID] = seqNo
bool function amIOnRing
// I am on the ring if: (I received controlBig from
   this transmitter and did NOT receive controlSmall
   from the same transmitter) OR (I received
   controlBig from this transmitter with higher seqNo
   than controlSmall)
return (ctrlBigSeqNo[transmitterID] exists AND
  ctrlSmallSeqNo[transmitterID] does not exist) OR
  (ctrlBigSeqNo[transmitterID] > ctrlSmallSeqNo[transmitterID]);
function forwardDataPacket
// N/L = RingArea/circleArea
rangeSmall = sqrt(rangeBig^2 − N * rangeBig^2/L)
if needToSendControl then
    send ControlBig to rangeBig
    send ControlSmall to rangeSmall
    needToSendControl = false
end
send Data
if countAckMap[sourceID,seqNo] not found in countAckMap then
    insert [sourceID, seqNo, rangeSmall] in countAckMap // use
       same rangeSmall for all packets from the same
       source or for only one packet seqNo
end
```

Table 1: Simulation parameters.

Parameter	Value
Size of simulated area	6 mm * 6 mm
Number of nodes	10 000
Communication range	1 000 μm
RangeBig	1 000 μm
RangeSmall	variable
Data packet size	1 003 bit
Control packet sizes	101, 102 bit

flooding, the maximum number of data copies received in a time window must not exceed 2 packets in order for the node to forward ($redundancy = 2$). For probabilistic flooding, the probability value is set to the minimum probability *proba* that gives 100% delivery in each scenario that is 6% for without the ring and 10% with dynamic ring (found through testing).

The dynamic ring scheme starts with the DEDeN initialization phase in order for nodes to know their local density and compute their rangeSmall values. We recall that DEDeN initialization can be repeated when the network changes its topology. The CBR flow starts after the DEDeN and SLR initialization phases, to not interfere with them.

To desynchronize node forwarding in ultra-dense networks and to reduce collisions, nodes choose a random backoff before forwarding, from a fixed window in pure flooding, probabilistic flooding and SLR, and from a dynamic window in backoff flooding.

To avoid forwarding loops, nodes forward packets they receive for the first time only. Node temporarily record the source ID and the data packet sequence number, so that they do not re-forward copies of the same data packet, as detailed in Algo. 1 .

The evaluation uses the 4 routing protocols above with 2 variants each (without the ring and with dynamic ring), and for 10 different random number generator seeds for the backoff time before transmission. This results in 80 simulations with 50 data packets each.

Our ring scheme aims to reduce the number of forwarders by placing them at the border of the communication, while keeping a 100% successful packet delivery to all the nodes (in flooding schemes) or the destination node (in unicast schemes). Thus, the evaluation metrics are the number of forwarders and the delivery ratio. A good network performance means a successful packet delivery to the destination with minimum resources (forwarders). The cumulative number of forwarders per packet along with the cumulative number of receivers per packet are averaged over the 10 runs and the 50 packets.

4.3 Dynamic ring with DEDeN

We recall that in the dynamic ring, the ring is set at the start (at the first data packet) as in the original ring. Afterwards, the dynamic ring uses density information from DEDeN density estimator to automatically adapt the ring width (rangeSmall) in order to include N ring neighbours in the ring per hop. In the particular case where the local density is smaller than N, the rangeSmall value is set to zero and all neighbours become ring neighbours. For the following simulations, we use $N = 60$ for all the routing protocols. This value

Table 2: Evaluation results in a 10 000 node network averaged for 10 runs and 50 packets each.

	Without ring	With dynamic ring
Pure flooding:		
forwarders per packet	10 000	1 949.2
receivers per packet	10 000	10 000
Probabilistic flooding:	*proba* = 6%	*proba* = 10%
forwarders per packet	601.59	273.512
receivers per packet	9 999.9	9 999.47
Backoff flooding:		
forwarders per packet	79.934	52.242
receivers per packet	9 999.97	9 999.55
SLR:		
forwarders per packet	901.688	129.116
Destination reached	100%	100%

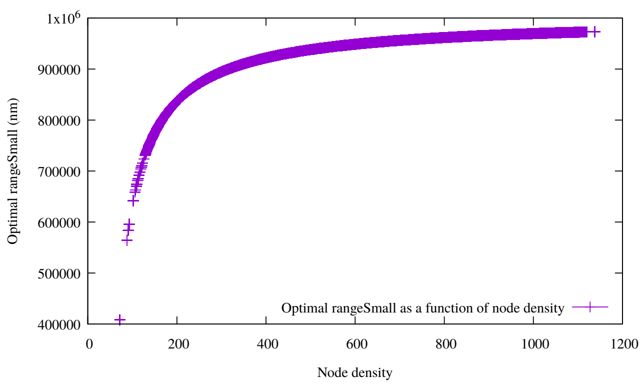

Figure 4: The dynamic ring with DEDeN in pure flooding assigning different rangeSmall values for nodes depending on their density.

includes non-forwarders and forwarders; non-forwarders are not only nodes that previously forwarded a copy of the data packet, but also nodes that are not chosen by the routing scheme to forward.

Table 2 shows the final comparison of the average values of the 10 runs with 50 packets each for all the different combinations of the routing schemes. We provide a separate web site[1] to reproduce results along with the simulation's description.

Effect of the dynamic ring on pure flooding. Fig. 4 confirms that the dynamic ring assigns indeed ring widths (rangeSmall) to nodes, depending on their local density. The higher the density, the higher the rangeSmall value is, and the thinner the ring width is. When the local density is very high, the rangeSmall value approaches the communication range (10^6 *nm*). However, when the density ≈ 120 (that is double the number of ring neighbours), the rangeSmall value is approximately half the communication range.

Table 2 also confirms the expectations: the number of senders per packet is reduced by 80%, from 10 000 to 1949.2, with 100% delivery rate. Fig. 5 shows the placement of the forwarders on rings.

[1]http://eugen.dedu.free.fr/bitsimulator/nanocom22

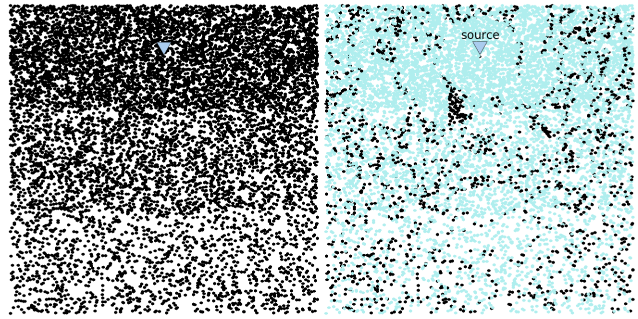

Figure 5: Pure flooding without (left) and with the dynamic ring with DEDeN (right); forwarders in black, receivers in blue, first packet of the first run only.

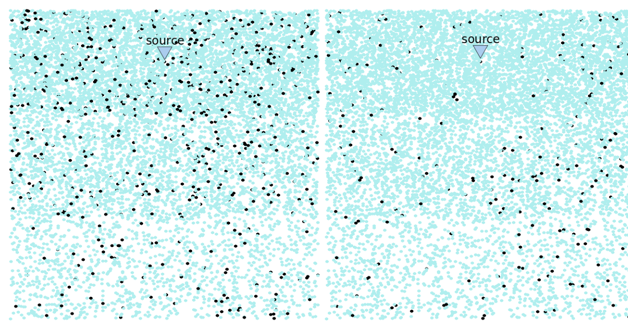

Figure 6: Probabilistic flooding without (left) and with the dynamic ring with DEDeN (right); forwarders in black, receivers in blue, first packet of the first run only.

Effect of the dynamic ring on probabilistic flooding. Table 2 shows that the probabilistic dynamic ring is efficient in reducing the number of forwarders per packet by 54% from 601 to 273, while the delivery rate is of 99.9%. Fig. 6 shows the difference in the placement of forwarders from random (left) to on rings (right).

Effect of the dynamic ring on backoff flooding. The dynamic ring improves backoff flooding as seen in Table 2, where the number of relay nodes per packet decreases by 34% (from 79.9 to 52.2), with almost all nodes receiving the packet (99.99%). Fig. 7 shows fewer and better placed forwarders with the dynamic ring (right) compared to no ring (left). This is an exceptional result, given that backoff flooding is already a highly efficient flooding.

Effect of the dynamic ring on SLR. Table 2 shows that the number of forwarders is reduced by 85%, from 901 (without the ring) to 129 (with the dynamic ring), while keeping 100% successful packet delivery. Fig. 8 visually shows this reduction and the optimized placement of forwarders on border of ranges.

To conclude, the dynamic ring allows to optimize all the presented routing protocols by choosing a ring width value. The dynamic ring selects forwarders on border of communication ranges in rings and significantly reduces the number of forwarders per packet while keeping 100% delivery rate to the destination(s).

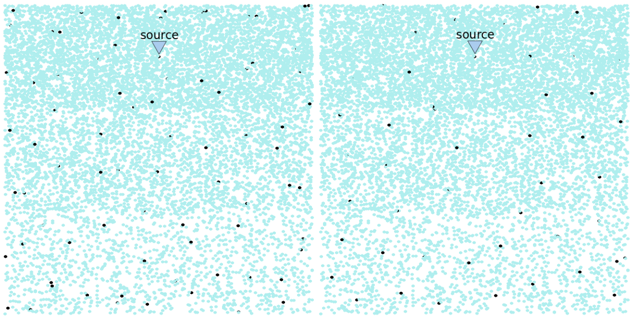

Figure 7: Backoff flooding without (left) and with the dynamic ring with DEDeN (right); forwarders in black, receivers in blue, first packet of the first run only.

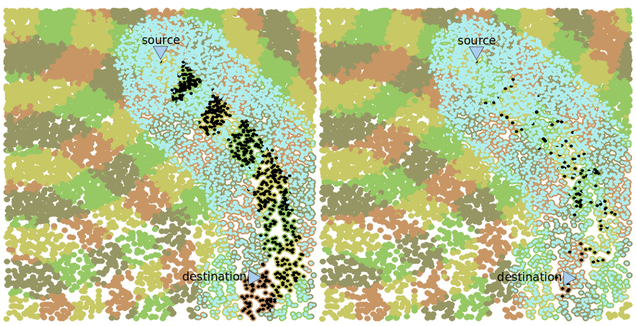

Figure 8: SLR without (left) and with the dynamic ring with DEDeN (right); forwarders in black, receivers in blue, first packet of the first run only.

5 CONCLUSION AND FUTURE WORK

This article presents an optimization for the routing schemes in electromagnetic nanonetworks, using an improved method of the ring: the dynamic ring. The original ring uses two control packets that delimit a fixed ring area, and forwarders are chosen only from this area. Classical acknowledgment methods that count the number of acknowledgments from the ring to update its width, are proved not to work in the context of electromagnetic nanonetworks.

The dynamic ring scheme with a density estimator automatically selects a ring width value for each node. The scheme is implemented in a nanonetwork simulator below four routing protocols: three flooding schemes and one destination-oriented scheme. The results are compared using two metrics: the number and placement of forwarders and the packet delivery rate. They show that the scheme generally selects forwarding nodes found at the border of communication ranges and reduces the number of forwarders per packet, while maintaining a successful packet delivery.

Future work includes studying the optimal number of ring neighbours in the dynamic ring and understanding the effect of local number of forwarders on the delivery rate.

ACKNOWLEDGMENTS

This work has been funded by Pays de Montbéliard Agglomération (France).

REFERENCES

[1] Thierry Arrabal, Dominique Dhoutaut, and Eugen Dedu. 2018. Efficient density estimation algorithm for ultra dense wireless networks. In *International Conference on Computer Communications and Networks (ICCCN)*. IEEE, Hangzhou, China, 1–9.

[2] Thierry Arrabal, Dominique Dhoutaut, and Eugen Dedu. 2018. Efficient multi-hop broadcasting in dense nanonetworks. In *17th IEEE International Symposium on Network Computing and Applications (NCA)*. IEEE, Cambridge, MA, USA, 385–393.

[3] Dominique Dhoutaut, Thierry Arrabal, and Eugen Dedu. 2018. BitSimulator, an electromagnetic nanonetworks simulator. In *5th ACM Int. Conf. on Nanoscale Computing and Comm. (NanoCom)*. ACM, Reykjavik, Iceland, 1–6.

[4] Holger Füßler, Jörg Widmer, Michael Käsemann, Martin Mauve, and Hannes Hartenstein. 2003. Contention-based forwarding for mobile ad hoc networks. *Ad Hoc Networks* 1, 4 (2003), 351–369.

[5] Zahed Hossain, Qing Xia, and Josep Miquel Jornet. 2018. TeraSim: An *ns-3* extension to simulate Terahertz-band communication networks. *Nano Communication Networks* 17 (Sept. 2018), 36–44.

[6] Farah Hoteit, Eugen Dedu, Winston K.G. Seah, and Dominique Dhoutaut. 2022. Ring-based forwarder selection to improve packet delivery in ultra-dense networks. In *IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, Austin, TX, USA, 2709–2714.

[7] Renato Iovine, Valeria Loscrì, Sara Pizzi, Richard Tarparelli, and Anna Maria Vegni. 2017. Electromagnetic nanonetworks for sensing and drug delivery. In *Modeling, Methodologies and Tools for Molecular and Nano-scale Communications*. Springer, 473–501.

[8] Josep Miquel Jornet and Ian F. Akyildiz. 2010. Graphene-based nano-antennas for electromagnetic nanocommunications in the terahertz band. In *4th European Conference on Antennas and Propagation*. IEEE, Barcelona, Spain, 1–5.

[9] Josep Miquel Jornet and Ian F. Akyildiz. 2014. Femtosecond-Long Pulse-Based Modulation for Terahertz Band Communication in Nanonetworks. *IEEE Transactions on Communications* 62, 5 (May 2014), 1742–1753.

[10] Chunchao Liang, Sunho Lim, Manki Min, and Wei Wang. 2014. Geometric broadcast without GPS support in dense wireless sensor networks. In *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*. IEEE, 405–410.

[11] Christos Liaskos, Angeliki Tsioliaridou, Sotiris Ioannidis, Nikolaos Kantartzis, and Andreas Pitsillides. 2016. A deployable routing system for nanonetworks. In *2016 IEEE International Conference on Communications (ICC)*. IEEE, Kuala Lumpur, Malaysia, 1–6.

[12] Christos Liaskos, Ageliki Tsioliaridou, Andreas Pitsillides, Ian F. Akyildiz, Nikolaos V. Kantartzis, Antonios X. Lalas, Xenofontas Dimitropoulos, Sotiris Ioannidis, Maria Kafesaki, and C.M. Soukoulis. 2015. Design and Development of Software Defined Metamaterials for Nanonetworks. *IEEE Circuits and Systems Magazine* 15, 4 (2015), 12–25.

[13] Leonardo Maccari, Mirko Maischberger, and Renato Lo Cigno. 2018. Where have all the MPRs gone? On the optimal selection of Multi-Point Relays. *Ad Hoc Networks* 77 (2018), 69–83.

[14] Semanta Raj Neupane. 2014. *Routing in resource constrained sensor nanonetworks*. Master's thesis. Tampere University of Technology, Tampere, Finland.

[15] Giuseppe Piro, Luigi Alfredo Grieco, Gennaro Boggia, and Pietro Camarda. 2013. Nano-Sim: simulating electromagnetic-based nanonetworks in the network simulator 3. In *6th International ICST Conference on Simulation Tools and Techniques (SimuTools)*. ACM, Cannes, France, 203–210.

[16] Daniel Gutiérrez Reina, Sergio Toral, Princy Johnson, and Federico Barrero. 2015. A survey on probabilistic broadcast schemes for wireless ad hoc networks. *Ad Hoc Networks* 25 (Feb. 2015), 263–292.

[17] Emre Sahin, Orhan Dagdeviren, and Mustafa Alper Akkas. 2021. An Evaluation of Internet of Nano-Things Simulators. In *6th International Conference on Computer Science and Engineering (UBMK)*. IEEE, Ankara, Turkey, 670–675.

[18] Suraiya Tairin, Novia Nurain, and ABM Alim Al Islam. 2017. Network-level performance enhancement in wireless nanosensor networks through multi-layer modifications. In *International Conference on Networking, Systems and Security (NSysS)*. IEEE, Dhaka, Bangladesh, 75–83.

[19] Ageliki Tsioliaridou, Christos Liaskos, Eugen Dedu, and Sotiris Ioannidis. 2017. Packet routing in 3D nanonetworks: A lightweight, linear-path scheme. *Nano Communication Networks* 12 (June 2017), 63–71.

[20] Ageliki Tsioliaridou, Christos Liaskos, Sotiris Ioannidis, and Andreas Pitsillides. 2015. CORONA: A Coordinate and Routing system for Nanonetworks. In *2nd ACM International Conference on Nanoscale Computing and Communication*. ACM, Boston, MA, USA, 1–6.

[21] Ageliki Tsioliaridou, Christos Liaskos, Sotiris Ioannidis, and Andreas Pitsillides. 2016. Lightweight, self-tuning data dissemination for dense nanonetworks. *Nano Communication Networks* 8 (2016), 2–15.

[22] Xin-Wei Yao, Ye-Chen-Ge Wu, and Wei Huang. 2019. Routing techniques in wireless nanonetworks: A survey. *Nano Communication Networks* 21 (2019), 100250.