

# Congestion Control in Wireless Sensor Networks based on Bird Flocking Behavior

Pavlos Antoniou<sup>a,\*</sup>, Andreas Pitsillides<sup>a</sup>, Tim Blackwell<sup>b</sup>, Andries Engelbrecht<sup>c</sup>, Loizos Michael<sup>d</sup>

<sup>a</sup>Department of Computer Science, University of Cyprus, P.O. Box 20537, CY-1678 Nicosia, Cyprus

<sup>b</sup>Department of Computing, Goldsmiths, University of London, New Cross, London, SE14 6NW United Kingdom

<sup>c</sup>Department of Computer Science, School of IT, University of Pretoria, South Africa

<sup>d</sup>School of Pure and Applied Sciences, Open University of Cyprus, P.O. Box 24801, CY-1304 Nicosia, Cyprus

---

## Abstract

This paper proposes that the flocking behavior of birds can guide the design of a robust, scalable and self-adaptive congestion control protocol in the context of wireless sensor networks (WSNs). The proposed approach adopts a swarm intelligence paradigm inspired by the collective behavior of bird flocks. The main idea is to 'guide' packets (birds) to form flocks and flow towards the sink (global attractor), whilst trying to avoid congestion regions (obstacles). The direction of motion of a packet flock is influenced by repulsion and attraction forces between packets, as well as the field of view and the artificial magnetic field in the direction of the artificial magnetic pole (sink). The proposed approach is simple to implement at the individual node, involving minimal information exchange. In addition, it displays global self-\* properties and emergent behavior, achieved collectively without explicitly programming these properties into individual packets. Performance evaluations show the effectiveness of the proposed Flock-based Congestion Control (Flock-CC) mechanism in dynamically balancing the offered load by effectively exploiting available network resources and moving packets to the sink. Furthermore, Flock-CC provides graceful performance degradation in terms of packet delivery ratio, packet loss, delay and energy tax under low, high and extreme traffic loads. In addition, the proposed approach achieves robustness against failing nodes, scalability in different network sizes and outperforms typical conventional approaches.

*Keywords:* Wireless Sensor Networks (WSNs), Congestion Control and Avoidance, Bird Flocking Behavior.

---

## 1. Introduction

Over the last two decades, there has been an unprecedented research interest in the area of autonomous networked systems with emphasis on Wireless Sensor Networks (WSNs) [1]. WSNs are being deployed for several mission-critical tasks (e.g. as platforms for health monitoring, process control, environmental observation, battlefield surveillance), and are expected to operate unattended for extended periods of time. Typically, WSNs comprise of small (and often cheap),

---

\*paul.antoniou@cs.ucy.ac.cy

cooperative devices (nodes) which may be (severely) constrained in terms of computation capability, memory space, communication bandwidth and energy supply. In the context of WSNs, autonomous nodes may interact (a) with the environment so as to sense or control physical parameters, and (b) with each other in order to exchange information or forward data towards one or more sink nodes. This mass of interactions, in conjunction with variable wireless network conditions, may result in unpredictable behavior in terms of traffic load variations and link capacity fluctuations. The network condition is worsened due to topology changes driven by node failures, mobility, or intentional misbehavior. These stressful situations are likely to occur in WSN environments, thus increasing their susceptibility to congestion.

The problem of congestion in general is very complex. Furthermore, differing from the general definition of congestion, two new considerations emerged from WSNs, namely the *size of the problem changes*, and the *resources to solve the problem are limited*. The former consideration is mostly encountered due to differing sizes of WSNs expected to range from a few to several million sensor nodes, while the latter issue is an intrinsic characteristic of WSNs. Additionally, the convergent (many-to-one) nature of WSN traffic, especially in single-sink WSNs, necessitates completely different ways of addressing congestion than in traditional end-to-end strategies.

Congestion control (CC) and avoidance includes measures taken for manipulating the traffic within the network in order to combat congestion and avoid congestion collapse. A centralized CC approach cannot be generally applied since it provokes several serious drawbacks [2]. Firstly, such approach leads to excessive communication load in the network which rapidly depletes the batteries [1]. Secondly, centralized decision making for CC results in slow reaction to changes in network and traffic conditions [3]. Thirdly, a centralized approach does not take advantage of the in-network processing capability of WSNs which permits simple processing and decision-making by individual nodes [4]. Finally, in the event of communication failure, the entire system becomes inoperational [5].

The resource limited and unpredictable nature of WSNs necessitates *decentralized, robust, self-adaptive*, and *scalable* mechanisms. Novel CC approaches for WSNs should be *simple* to implement at individual node level with *minimal exchange of information*.

Nature-inspired (natural) designs have inherent powerful characteristics and are often more effective and simpler than man-made designs [6]. Natural systems usually exhibit remarkable survivability and robustness to external stimuli and internal perturbations or loss of units, as well as excellent scaling properties. Adaptation is one of the major strengths of bio-systems as they must respond to the addition or removal of members, as well as to sudden changes in the environment.

WSNs, in many ways, can be likened to social groups found in nature (e.g. bird flocks, ant colonies). Social groups accomplish their tasks collectively (by simple neighbor-to-neighbor interactions), in a decentralized manner, and in the absence of (external) central supervision. This study intends to explore what can be learned from the behavioral tendencies of natural systems for designing robust and scalable network control techniques. Drawing inspiration from the collective behavior of social groups, local behavior between individuals can be dictated easier, leading to emergent global behavior of minimum congestion and direction of information flow to the sink. In this way, self-\* properties, e.g. self-organization and self-adaptation, are not implemented explicitly into individual entities, but emerge as a result of the design of the nature-inspired CC model.

The main idea behind this study is to move packets to the sink, whilst providing congestion control and avoidance by mimicking the *obstacle avoidance behavior of bird flocks* illustrated in Fig. 1. The Flock-based Congestion Control (Flock-CC) approach is proposed, where packets

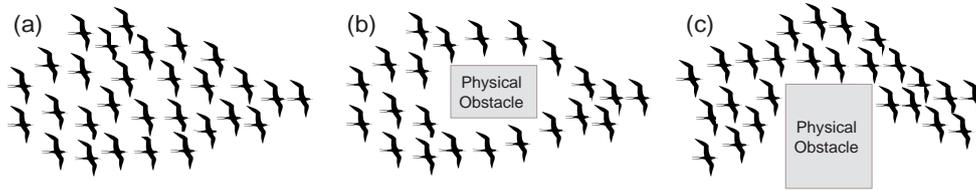


Figure 1: The obstacle avoidance behavior of bird flocks: (a) A flock of (migrating) birds moves through an obstacle-free environment (towards a magnetic pole); (b) an obstacle ‘forces’ the flock to split into 2 subflocks; (c) the obstacle extends further on one side and the flock is reformed along the path bypassing the obstacle on the other side. Flocking behavior was first simulated on a computer in 1986 by Craig Reynolds [7] with his simulation program, Boids.

are modeled as birds flying over a topological space, e.g. a sensor network. The main idea is to ‘guide’ packets to form flocks and flow towards a global attractor (which is the sink in WSNs), whilst trying to avoid obstacles (failing nodes and congested regions). Inspiration for designing Flock-CC is drawn by: (a) the repulsive and attractive interactions among closely located individuals proposed by Couzin et al. [8], (b) the orientational movement of migrating birds towards a global attractor (poles or equator) under the influence of the magnetic field of Earth, and (c) the limited visual perception (field of view) of individuals within the flock.

Initial attempts for the development of the Flock-CC approach are described in [9], [10], and [11]. This paper presents and evaluates an improved Flock-CC model, which mimics more faithfully the bird flocking paradigm, as presented by Couzin et al. [8]. The new model is simpler, involving only two tunable parameters instead of three (hence easier to tune and thereafter deploy), while it maintains comparably good performance characteristics.

Before introducing a detailed description of the Flock-CC approach, it is worth illustrating in a simple WSN some intrinsic properties. The emergent global behavior of the Flock-CC approach can be perceived by the visual representation of flock movements as well as on the basis of the performance evaluation metrics. We experimented with a minimal topology (details of the simulation environment appear in Section 4). Fig. 2(a) shows the movement of packet flocks over a sensor network on the basis of the Flock-CC approach, mimicking the behavior of

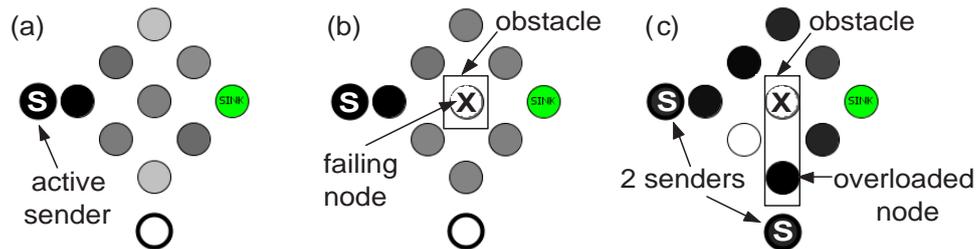


Figure 2: The obstacle avoidance behavior of packet flocks demonstrated through a minimal topology. Color intensity indicates the intensity of packets visiting within a given 1-second time slot. (a) A flock of packets moves through an obstacle-free environment towards the sink; (b) an obstacle (failing node) ‘forces’ the flock to split into 2 subflocks; (c) after the activation of the second sender, the obstacle is supplemented by an overloaded node which causes the packet flock coming from the back of the network to adapt and follow a less congested path to the sink (top part of the network).

natural bird flocks of Fig. 1(a) in a non-congested and failure-free environment. In Fig. 2(b), Flock-CC exhibits the obstacle avoidance behavior of bird flocks in a failure prone environment and in 2(c) in an overloaded environment. In Section 4.3, extensive results reveal the significance of the intrinsic flocking characteristics for improving the performance of WSNs.

The proposed approach primarily targets event-based WSNs used, for example, in process control and disaster recovery missions. In event-based WSNs, packet bursts can be dynamically and randomly initiated at any sensor node within the network since each node is expected to report to the sink once the occurrence of a given event has been detected. It is not uncommon for many neighbouring nodes to initiate transmission at the same time in order to report an event in their vicinity.

The rest of this paper is organized as follows. Section 2 discusses the problem of congestion in WSNs, presents previous work on CC for WSNs, and introduces the paradigm of swarm intelligence, with emphasis on the flocking behavior of birds. Section 3 deals with the proposed flock-based model. Section 4 presents results obtained across a number of scenarios with different topology and traffic conditions. Section 5 draws conclusions and proposes areas of future work.

## 2. Background

This section gives background information on the problem of congestion in WSNs and discusses recent CC approaches. In addition, an overview of swarm intelligence is given with emphasis on the flocking behavior of birds.

### 2.1. The problem of congestion in WSNs

Congestion occurs when the traffic load injected into the network exceeds available capacity at any point of the network. Congestion causes energy waste, throughput reduction, and increase in collisions and retransmissions at the medium access control (MAC) layer. In addition, congestion results in the increase of queueing delays and information loss, leading to the deterioration of the offered quality of service (QoS), decrease of network lifetime and even the decomposition of network topology in multiple components. In traditional Internet wired networks, buffer drops are taken as an indication of congestion while congestion control is usually carried out in an end-to-end manner (i.e. only the source-destination pair is involved) as opposed to the many-to-one nature of information transfer, which is a characteristic of sensor networks. End-to-end CC approaches will not be effective in such error prone environments because the end-to-end nature may result in reduced responsiveness causing increased latency and error rates, especially during long periods of congestion. Furthermore, simulation studies conducted by [12] and [13] revealed that, in WSNs where the wireless medium is shared using Carrier Sense Multiple Access (CSMA)-like protocols, wireless channel contention losses can dominate buffer drops and increase quickly with offered load<sup>1</sup>. The problem of channel losses is worsened around hot spot areas, as for example, in the proximity of an event, or around the sink. In the former case, congestion occurs if many nodes report the same event concurrently, while in the latter case congestion is experienced due to the converging (many-to-one) nature of packets from multiple sending nodes to a single sink node. These phenomena result in the starvation of channel capacity in the vicinity of senders, while the wireless medium capacity can reach its upper limit faster than

---

<sup>1</sup>Buffer losses are likely caused by congestion, whereas contention losses not necessarily.

queue occupancy [14]. Thus, queue occupancy alone cannot accurately serve as an indication of congestion. On the other hand, the study of Hull et al. [13] in a large sensor network testbed revealed that when used alone, wireless channel-based congestion detection performs worse than queue occupancy-based congestion detection. Therefore, congestion is expected to be effectively detected using a combination of buffer occupancy and wireless channel load.

Due to their severely constrained nature, WSNs necessitate simple, autonomous, decentralized CC strategies which promise fast, effective and efficient relief from congestion. Decentralized approaches are expected to be adopted in WSNs. Each node should make decisions based only on locally available information (e.g. buffer load, channel load) since it is impractical for the nodes to have complete information about the system state.

## 2.2. *Conventional congestion control approaches for WSNs*

Early studies in the area of sensor networks had mainly focused on more fundamental networking problems, e.g. medium access control [15], topology [16], routing [17], and energy efficiency [18], largely ignoring network performance assurances. Lately, with the emergence of mission-critical applications (e.g. health monitoring), there has been an increased interest in performance control mechanisms [19], as for example to avoid congestion caused by the uncontrolled use of the scarce network resources.

Various conventional CC approaches can be found in WSNs literature based on traffic manipulation (e.g. rate adaptation [12], [13], [20] multi-path routing [21], [22], [23]), topology control (e.g. clustering formation [24]), and network resource management (e.g. power control, multiple radio interfaces [25]). A comprehensive survey of congestion control approaches proposed for WSNs can be found in [26].

Some conventional CC approaches [12], [13], [21], [20] are based on rate control that alleviates congestion by throttling the injection of traffic in the network. However, rate control attempts to decrease the reporting rate of nodes during (transient or persistent) congestion phenomena and may result in the deterioration of the offered quality of service, perhaps when needed the most. In addition, rate control cannot be efficiently used during transient congestion phenomena caused by aperiodic and short term packet bursts (e.g. in event monitoring applications) due to the slowness of rate-based approaches to react. Clustering is also used for controlling congestion. Clustering formation assumes special roles in the network (e.g. clusterheads), while additional mechanisms are needed for maintaining and re-assigning roles. The problem with this approach is that areas around clusterheads may progressively become collision hot spots. Congestion mitigation based on power control and multiple radio interfaces seems unrealistic in WSNs since the low-cost nodes incentive is violated. On the other hand, multi-path routing has potential to effectively and efficiently alleviate congestion without deteriorating the offered network QoS. A congestion aware routing (CAR) approach [22] dynamically discovers a congestion zone and routes high priority packets inside the zone while low priority packets are routed outside the zone. One of the problems with this approach is how to categorize traffic. Biased Geographical Routing (BGR) [21] alleviates congestion by splitting traffic flows and performing rate control. Both studies in [21], [22] use location information provided by the Global Positioning System (GPS) to discover congestion regions. However, beyond the problems noted earlier for rate-based control, GPS can work only outdoors in the absence of any obstruction, while GPS receivers are expensive and not suitable in the construction of small cheap sensor nodes. In a similar approach, Traffic Aware Dynamic Routing (TADR) [23], a mixed potential field is constructed using depth and normalized queue length to route packets around the congestion areas and scatter the excessive packets along multiple paths consisting of idle and under-loaded nodes. However, the dynamic

conditions of the wireless medium which may cause excessive packet loss in WSNs are not considered, while the authors evaluate their scheme assuming a perfect (but practically infeasible) MAC protocol that provides a stable radio link without causing collisions.

Conventional optimization techniques for CC could also be adopted, but these rely on complex mathematical models and have proven hard to apply in the context of autonomous decentralized environments, while the sensitivity of these models to the dynamic and unpredictable environment is a challenge. Also, centralized multi-objective optimization is not practical especially in large-scale autonomous networks.

### 2.3. *Swarm intelligence and network-oriented approaches*

This study draws inspiration from nature, which has been very successful in effectively solving similar types of complex problems. Recently, nature-inspired computing has been fueled by the emergence of a novel computational paradigm, the so-called Swarm Intelligence (SI) paradigm [6], [27]. SI techniques are motivated by the collective behavior of social insect societies, bird flocks, bacteria and other social organisms living in decentralized, self-organizing, and adapting environments. SI techniques reportedly provide a promising basis for computing environments that need to exhibit these characteristics [6]. Research in SI has provided computer scientists with powerful methods for designing distributed control and optimization algorithms. These methods are applied successfully to a variety of scientific and engineering problems [28]. In addition to achieving good performance on a wide spectrum of ‘static’ problems, swarm-based algorithms tend to exhibit a high degree of flexibility and robustness in dynamic environments [28], and have even been successful in tackling multi-objective problems in ad-hoc networks [29].

Currently, two of the most successful SI classes of algorithms are Ant Colony Optimization (ACO) [30] and Particle Swarm Optimization (PSO) [31]. Both theories were successfully involved in network-oriented studies, especially in the field of ad-hoc and mobile ad-hoc networks (MANETs). A large number of SI-based routing approaches for MANETs can be found in [32], while some notable examples are presented below.

Driven by the collective behavior of ants in finding paths from the colony to food, many researchers [33], [34], and [35], developed ACO-based algorithms to solve the problem of routing in ad-hoc networks. In ACO, artificial ants build solutions by moving on a graph of nodes and, by mimicking real ants, deposit artificial pheromone on the links (that form the route traveled by each ant) in such a way that future artificial ants can build new better solutions. The work of Di Caro et al. [33] incorporates congestion awareness in an end-to-end manner (AntHocNet). According to [36], AntHocNet outperformed the Ad-hoc On-demand Distance Vector (AODV [37]<sup>2</sup>) routing protocol in terms of delivery ratio, end-to-end delay and delay variation (jitter). An important observation was that the advantage of AntHocNet over AODV grew for larger networks, especially in terms of overhead, suggesting that AntHocNet is more scalable than AODV. Rajagopalan et al. [34] presented an Ad-hoc Networking with Swarm Intelligence (ANSI) routing protocol with congestion-aware characteristics. The ANSI protocol was found to outperform AODV in terms of packet delivery, number of packets sent, end-to-end delay, and jitter experiencing fewer route errors as compared to AODV. Xiangquan et al. [35] developed an ACO algorithm to optimize energy, congestion and load balancing along with providing efficient routing in ad-hoc networks.

---

<sup>2</sup>AODV [37] is a conventional single-path ad-hoc routing protocol.

Based on the aforementioned results, it seems that nature-inspired approaches are able to outperform conventional routing algorithms in (mobile) ad-hoc networks. However, the aforementioned approaches are not well suited for the unique features and application requirements of WSNs. According to Akyildiz et al. [1] there are some important differences between ad-hoc and sensor networks. For example, sensor nodes are limited in power, computational capacities, and memory whereas nodes involved in ad-hoc networks may be more powerful and less constrained machines like laptops and PDAs. Also, the number of sensor nodes in a sensor network can be several orders of magnitude higher than the nodes in an ad-hoc network, and can often be densely deployed over large areas. Sensor nodes are prone to failures and are mostly statistically deployed. On the other hand, ad-hoc nodes are usually less densely deployed in small areas.

These differences drive the necessity for novel CC strategies for WSNs, on which this study focuses. It is worth pointing out that there are not many nature-inspired studies that have explicitly focused on avoiding or combating congestion in sensor networks. A notable related approach is the ant-based multi-QoS routing protocol for sensor networks (AntSensNet) [38]. AntSensNet combines a hierarchical structure of the network with the principles of ACO-based routing, whilst trying to satisfy the QoS requirements of different kinds of traffic requested by the applications. By using clustering, AntSensNet aims to avoid congestion after quickly judging the average queue length and solving convergence problems, which are typical in ACO. However, AntSensNet has some important drawbacks that increase protocol's complexity. These are discussed at the end of the performance evaluation section when comparing AntSensNet to Flock-CC.

#### 2.4. *The flocking behavior of birds*

Social groups found in nature (e.g. ant colonies, bird flocks) carry out their tasks collectively in order to contribute to a common goal. Even though individuals cooperate to accomplish a given global mission in a complex world (e.g. foraging, migration, nest building, defence against predators, etc.), an individual has only local perception of the surrounding environment and exhibits specific behavioral tendencies which are governed by a few simple rules.

The proposed approach involves reference to artificial bird flocks consisting of individuals with the same behavioral model (homogeneity) and with finite range of view (perception). These individuals interact with each other as well as with the environment. The design of the interactions in packet groups is influenced by the study of artificial flocks such as the 'boid' simulations of Reynolds [7] and the bio-swarm model of Couzin et al. [8]. The behavior of each individual is influenced by other individuals within its neighborhood.

The approach proposed in this article differs in two aspects from Couzin's model: (1) The bio-swarm model of Couzin was formulated on the metrical (continuous three-dimensional) space, whereas the Flock-CC model is applied on a two dimensional topological (discrete) space defined by the graph of nodes. (2) In Couzin's model (as well as in the Reynolds' model) individuals form flocks and move constantly in a given finite space without any attraction to a global target (final destination). On the other hand, in the Flock-CC packets are expected to form flocks and move towards the sink, mimicking migratory birds behavior. The latter Flock-CC characteristic necessitates the existence of a field of attraction towards the sink, likened to a magnetic pole.

Three inherent characteristics of bird flocks are presented next. These characteristics served as the basis for developing the Flock-CC protocol.

##### 2.4.1. *Repulsion and attraction zones and forces in bird flocks*

The notion of repulsion and attraction zones in bird flocks was modeled by a seminal study of Couzin et al. [8]. Researchers simulated the behavior of individuals as resulting from local repul-

sion, alignment, and attractive tendencies based upon the position and orientation of individuals relative to one another.

In Couzin’s model,  $N$  individuals with position vectors and unit direction vectors are simulated in continuous three-dimensional space. In each time step, individuals assess the position and/or orientation of neighbors within three non overlapping concentric behavioral zones described below and shown in Fig. 3. This information is used to determine a desired direction for each individual for the successive time step using a set of rules described next.

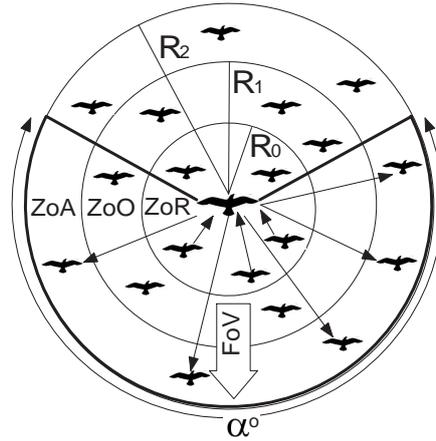


Figure 3: Representation of the area around a bird placed in the center:  $ZoR$  is the zone of repulsion,  $ZoO$  is the zone of orientation, and  $ZoA$  is the zone of attraction. The possible ‘blind area’ behind an individual is also shown. The aperture of the field of view (FoV) is  $\alpha$  degrees.

Each individual attempts to maintain a minimum distance from others within a ‘zone of repulsion’ (ZoR), modeled as a sphere, centered on the individual. This behavioral rule has the highest priority in the model. If at any time,  $t$ , more than one bird resides in the ZoR, the individual bird in the center responds by moving away from the neighbors within this zone. If there is no neighbor within the ZoR, the bird responds to others within the ‘zone of orientation’ (ZoO) and the ‘zone of attraction’ (ZoA). These zones are spherical, except for a blind volume behind the individual within which neighbors are undetectable. This volume is outside of the field of view (FoV) of the individual. The notion of the field of view is explained in the next section. Within the ZoO, the birds align with their neighbors and within the ZoA, the birds move towards each other. Birds cannot see too far, thus there is no interaction with neighbors located outside the ZoA.

Motivated by Couzin’s model, the collective flocking behavior of birds is considered an emergent behavior arising from a few *simple behavioral rules* that are followed by individuals, such as:

1. repel from neighbors (if too close, i.e. within the ZoR) to avoid collisions,
2. attract to neighbors (if apart, i.e. within the ZoA) to maintain coherence among the members of a flock,
3. orient and align with neighbors (within the ZoO),
4. introduce a randomness that allows for exploration of alternative paths.

These behavioral rules govern individual-level interactions which collectively result in the emergence of group-level transitions.

#### 2.4.2. Birds vision and the field of view

As discussed in the previous paragraph, the motion of each bird (individual) is only influenced by its nearest flock mates. Therefore, vision is the most important sense for birds. *The field of view (FoV) determines the extent of the observable world that is seen by each bird at any given moment.* Depending on the placement of the eyes, different animals have different fields of view. Some birds have a  $360^\circ$  degree overall FoV.

In Couzin's model all individuals have a FoV of  $\alpha^\circ$  and a blind volume of  $(360 - \alpha)^\circ$ . Fig. 3 illustrates the blind volume of the individual in the center as a cone with interior angle  $(360 - \alpha)^\circ$  (behind the individual), while the remaining space surrounding the individual is the FoV. An individual with  $\alpha = 360^\circ$  can respond to others in any direction within the behavioral zones.

#### 2.4.3. Magnetic fields and orientation of birds

The aforementioned behavioral rules and the presence of the FoV do not ensure that a bird flock will establish a flight path towards a specific attractor. In the absence of a such a (global) attractor, bird flocks are expected to move around in (coherent) formations without a specific destination. The existence of a global attractor was not a necessity in bird flocks simulated by Couzin's model. On the other hand, WSNs involve a central reference point (sink). The sink is the final destination for all packets, thus global attraction forces are needed to 'guide' packets to move towards the sink.

The orientational movement of migratory birds can serve as an excellent source of inspiration for developing a mechanism for *orientation and attractiveness to a global attractor*. In accordance to [39], migrating birds use the magnetic field of Earth for direction finding, either towards magnetic poles (polewards, northern or southern) or the magnetic equator (equatorwards).

### 3. Flock-based congestion control (Flock-CC)

This section, firstly, presents the concept behind the Flock-CC model, explaining how the movement of packets is modeled as a flock of birds. Secondly, the elements of the Flock-CC model are defined on the basis of the characteristics of natural bird flocks. Thirdly, the elements of the model are composed together to form the Flock-CC protocol.

#### 3.1. The concept

A WSN is viewed as a virtual ecosystem, where multiple packets are generated at source nodes and must be directed towards a dedicated sink node. The main idea of the proposed Flock-CC model is to *'guide' packets to form groups or flocks, and flow towards a global attractor (sink), whilst trying to avoid obstacles such as congestion regions and dead zones (regions with failing nodes)* as illustrated in Fig. 4.

*In the Flock-CC model, each packet is analogous to a bird* with dynamic position and direction updates, which 'flies' over the network undergoing successive hop-by-hop transitions over discrete points in the 2D space, defined by the positions of hosting nodes. The set of sensor nodes comprises the environment where packets move. The sequence of transitions determines the packet's (bird's) trajectory from its source to the sink. Note that this model differs from Couzin model, which is in a continuous 3D space.

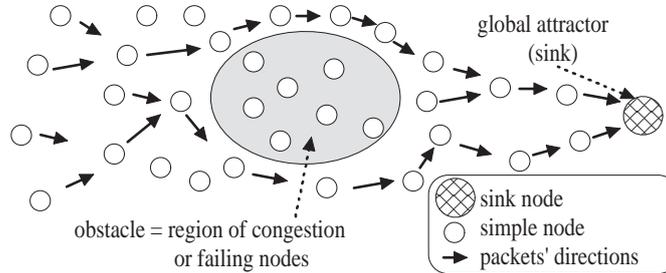


Figure 4: Packet flock moving towards sink whilst avoiding ‘obstacles’.

Conceptually, in order to make moving packets behave like a flock, the basic characteristics of natural flocks discussed in the previous section are incorporated in the Flock-CC model. At each hop, a packet interacts (attraction and repulsion forces) with other packets located on neighboring nodes in the packet’s FoV. Also the packet experiences a magnetic force in the direction of the sink (global attractor). More specifically, the proposed Flock-CC approach is governed by 4 rules. Each packet is:

- *Rule 1*: repelled from neighboring packets located on nodes at close distance (i.e one hop away from packet) within the packet’s FoV,
- *Rule 2*: attracted to neighboring packets located on nodes at medium distance (i.e. two hops away from packet) within the FoV,
- *Rule 3*: oriented and attracted to the global attractor under the influence of the environmental magnetic field, and
- *Rule 4*: experience some perturbation that may help the packets to pick a random route (i.e. trading exploration versus exploitation).

More detailed discussion on the implementation of these features on the Flock-CC approach follows next.

### 3.2. The Flock-CC model elements

The basic elements of the Flock-CC approach are defined in this section: (a) the repulsion and attraction zones and forces, (b) the artificial magnetic field, (c) the field of view (FoV), (d) the desirability function, and (e) randomness. The desirability function synthesizes the first three elements and determines the direction of movement of each packet, whilst randomness allows for exploration.

It is worth noting here that a basic design objective was to keep the number of adjustable parameters as small as practical in order to reduce complexity in optimizing parameter values for different network environments and conditions.

Consider a network of  $N$  autonomous nodes,  $N > 0$ , that are able to generate packets. A finite queue is associated with each node, while the node’s throughput is constrained by the wireless channel capacity. A packet  $i$  and its current hosting node  $n \in \{1, \dots, N\}$  (i.e. packet  $i$  is residing in the queue of node  $n$ ) are taken as points of reference in order to define and discuss repulsion and

attraction zones, the magnetic field and the field of view. The position of node  $n$  determines the position of the hosted packet  $i$ .

Each node  $n$  maintains four one-dimensional tables: (a) an attraction table,  $s_n$ , (b) a repulsion table,  $q_n$ , (c) a transmission table,  $t_n$ , and (d) an adjacency table,  $A_n$ . Each of these tables contains different information for a set of neighboring nodes for which node  $n$  has an active wireless link to. An entry  $s_{nm}$  of the attraction table of node  $n$  contains information about node  $m$ . The same applies for all tables. The contents of each table are discussed later.

All quantities defined herein are regularly sampled at discrete time intervals of  $T$  seconds at each sensor node. Then, the values of these quantities are broadcasted periodically (every  $T$  seconds) to all neighboring nodes (within transmission range), using a dedicated control packet. These periodic control packets are also used to update information about the connectivity of neighboring nodes, and the distance (hop distance) to the sink.

Below, the five basic elements of the Flock-CC approach are defined.

### 3.2.1. Repulsion and attraction zones and forces

The design of the Flock-CC model was primarily motivated by the neighborhood-based model of Couzin et al. [8]. As discussed earlier, the Flock-CC model is applied on a two dimensional topological (discrete) space defined by a graph of nodes. The edges in the graph indicate the existence of wireless connection between two nodes. Each packet is allowed to move in a hop-by-hop manner, over a set of nodes along the path to the sink.

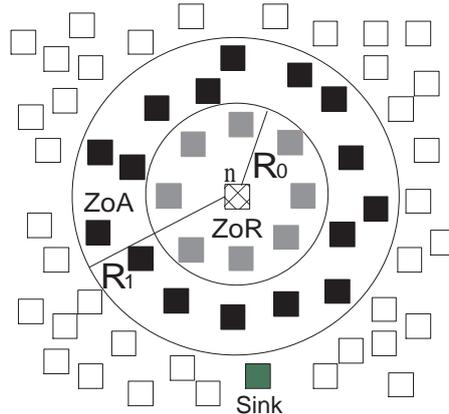


Figure 5: Representation of a sensor network.  $R_0$  is the transmission range of node  $n$  and defines an area (zone of repulsion) that includes all packets located on grey-shaded nodes. The outer area (zone of attraction) includes all packets located on black-shaded nodes.

As shown in Fig. 5, the zone of repulsion is defined as a circle of radius  $R_0$ , the transmission range of node  $n$ , around packet  $i$  that includes all packets at close distance. Practically speaking, these packets reside in the queues of the grey-shaded nodes. On the other hand, the zone of attraction is the outer zone of Fig. 5 and includes all packets at medium distance from packet  $i$ . In practise, these packets reside in the queues of black-shaded nodes (two hops away from node  $n$ ). The zone of orientation proposed in Couzin model is not defined in the context of the Flock-CC approach since the collective motion, orientation and alignment of individuals arises as a result of the artificial magnetic field presented below.

The strength of the repulsive and attractive forces exhibited by packets on nodes one and two hops away respectively is proportional to the number of packets located on these nodes respectively. The number of packets on nodes one hop away (number of packets in the queue) is obtained directly through broadcasted control packets. These control packets are seen as a means of transferring knowledge (propagate information) within the environment (sensor network) that is observable by a packets' eyes.

On the other hand, in practise, the number of packets on nodes two hops away cannot be obtained directly through broadcasted control packets since (black-shaded) nodes two hops away are outside the transmission range of the current hosting node  $n$ . Thus, this number is locally inferred at node  $n$  by measuring the number of successfully transmitted packets sent from nodes one hop away to nodes two hops away. These packets are 'visible' due to the broadcasting nature of the wireless channel. Hence, packet  $i$  is repelled from packets residing in the queues of grey-shaded nodes and attracted to packets moving to black-shaded nodes.

### 3.2.2. Artificial magnetic field

In order to avoid unnecessary routing loops and to minimize packet loss, packets should be 'guided' to establish flight paths towards the sink. The artificial magnetic field is adopted for two reasons: a) to provide orientation and b) attraction of packets to the sink.

The magnetic field of the Earth is responsible for guiding migratory birds to fly polewards or equatorwards as shown in Fig. 6(a). In the same context, the sink node is seen as an artificial magnetic pole within the sensor network and packets are expected to 'fly sinkwards' under the influence of the artificial magnetic field as shown in Fig. 6(b).

*Orientation to the sink is deduced from the direction of the magnetic field (that points to the sink), while attraction to shortest paths to the sink is driven by the magnetic field strength.* The magnetic field strength is higher through the shortest path to the sink and attenuates through longer paths. Orientation allows packets to perceive the direction of the sink, otherwise the absence of attraction to shortest paths towards the sink causes packets to wander around (trapped in loops between neighboring nodes). This looping behavior deteriorates the quality of service offered by the congestion control protocol. In addition, it is impossible for a packet to feel the attraction to shortest paths to the sink without orientating itself toward the sink.

By design, we adopt an approach which does not require geo-location, and thus avoids additional complexity. Orientation of packets to the sink is implemented as follows: The artificial magnetic field within the sensor network must point to the sink node. The direction of the sink is determined on the basis of the hop distance parameter,  $h_j(k)$ ,  $j \in \{1, \dots, N\}$ , indicating the number of hops between each node  $j$  and the sink at the  $k$ th sampling period. Each node evaluates its hop distance parameter by adding one to the smallest hop distance of neighboring nodes. The sink node initiates the process of evaluating the hop distance value at each node by broadcasting (in a control packet) its zero hop distance. The hop distance parameter is gradually propagated (over control packets) from the sink (pole) to each node into the network (environment where packets live). It is worth pointing out that the hop distance may change due to network topology modifications caused by node failures or displacement or other internal causes. If node  $n$  does not hear control messages from a given neighboring node for a maximum of  $T_{lost}$  seconds, then the node  $n$  assumes that the neighbor either has failed or gone out-of-range or is very busy. In this case, the unreachable neighbor is deleted from the adjacency table of node  $n$  containing the hop distances of all nodes one hop away. Similarly, the hop distance parameter of node  $n$  is re-evaluated if the loss of a neighboring node influences the parameter's value. Every change in hop distance values will be propagated backwards within the network and all affected nodes

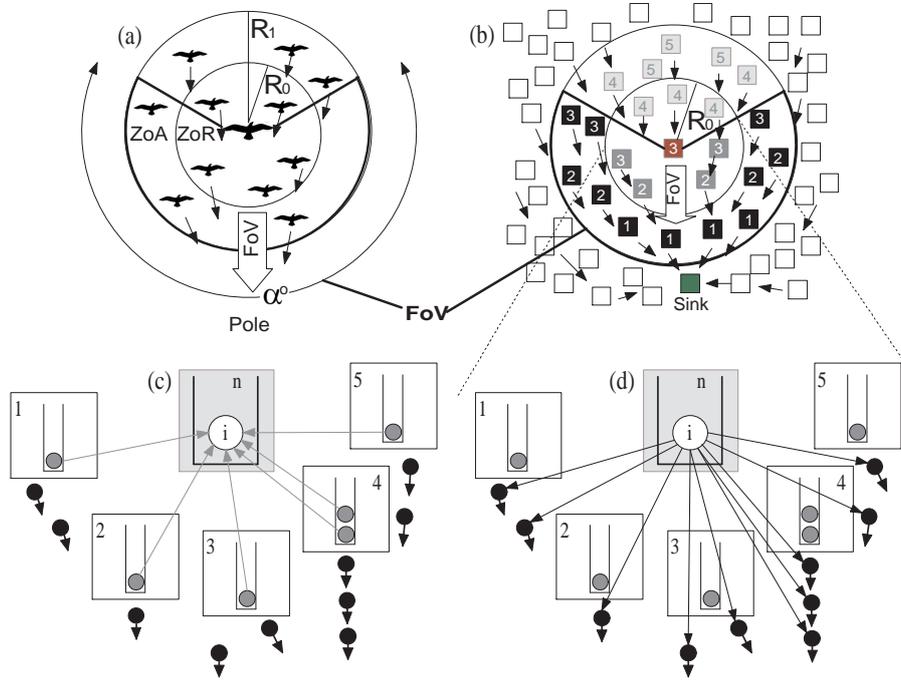


Figure 6: (a) A bird flock moving polewards under the influence of the magnetic field of the Earth (black arrows). The FoV of the bird placed in the center extends forward in the direction of the magnetic pole. (b) Packets generated in a sensor network will move sinkwards under the influence of the artificial magnetic field (black arrows). The number on each node indicates the hop distance from the sink (smaller numbers indicate closeness to the sink). The field of view (FoV) of packet  $i$  (on node  $n$ ) extends forward in the direction of the sink. The ZoR and ZoA around packet  $i$  are redefined as circular zones, except for an area behind the packet  $i$  that is outside the FoV. (c) Repulsion forces exercised on packet  $i$  from packets in the ZoR (heavy-gray-shaded packets). (d) Attraction forces exercised on packet  $i$  from packets in the ZoA (black-shaded packets).

are expected to update their hop related parameters. A simple hop distance protocol is adopted for this study. Note that it is anticipated that missing or incorrect hop count information will not affect Flock-CC adversely. This together with a detailed study of an optimized hop distance protocol, or indeed any other protocol that provides orientation and field strength, are beyond the scope of the current study. For example, contrary to the hop distance metric, the expected transmission count metric ETX [40] can be used as a path metric for multi-hop wireless routing. ETX incorporates link loss ratios, asymmetry in loss ratios between the two directions, and interference among successive links. Yet other path metrics could also be considered, such as product of per-link delivery ratios, end-to-end delay, or weighted cumulative expected transmission time (WCETT) [41]. These techniques however increase the protocol complexity and overhead. Thus the overall benefits may not be that obvious.

The magnetic field is visualized in Fig. 6(b), where the arrows point in the direction of the magnetic pole. A simpler way to visualize the magnetic field is to ‘connect’ the arrows to form ‘magnetic field lines’. The magnetic field lines pass over paths of descending hop distance. Packets retrieve environmental information such as the direction of the artificial magnetic field

(i.e. the hop distances of potential new hosting nodes) from their current hosting node.

Attraction to shortest paths toward the sink are implemented as follows: Besides orientation, the artificial magnetic field creates attraction to shortest paths to the sink. When a packet with orientation to the sink is ready to move, it has to choose between a number of potential new hosting nodes with different (equal or shorter) hop distances, taking into account the attraction and repulsion forces, as synthesized in the desirability function described later. The magnetic field guides packets to move to nodes closer to the sink where the strength of the field is higher. The movement of packets through nodes with shorter hop distances (forward movement) provides faster transitions to the sink and minimization of packet looping. This behavior is realized by enforcing differentiated attraction to potential new hosting nodes. In particular, packets perceive higher attraction to nodes closer to the sink (i.e. having shorter hop distance) than to nodes at equal hop distance. A rule-based strategy for differentiated attraction is proposed for the Flock-CC protocol, as described in Section 3.3.

### 3.2.3. Field of view (FoV)

Motivated by the limited visual perception of birds, packet  $i$  cannot ‘see’ and interact with all packets on nodes in its neighborhood, as shown in Figs. 7(a) and (b). Packet  $i$  can perceive only a fraction of packets, i.e. those located in the FoV, on the observable world of the packet. In this way, the number of interactions (repulsion and attraction forces) of a packet with neighboring packets are reduced resulting in a simpler and less resource demanding Flock-CC approach.

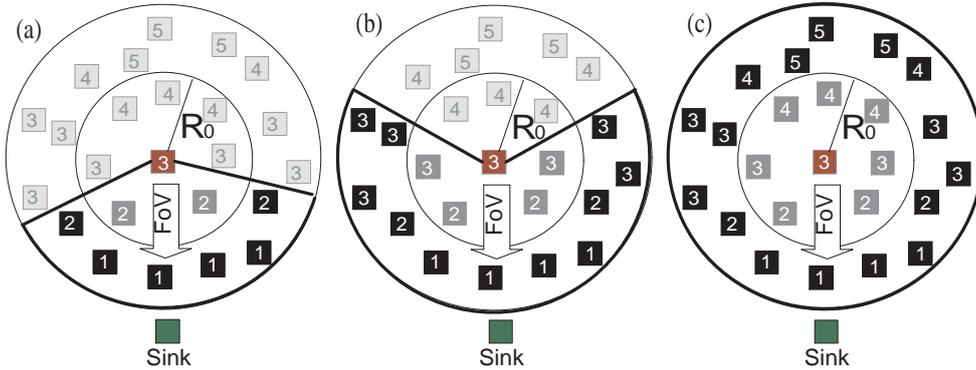


Figure 7: (a) Narrow FoV including nodes with shorter hop distance to the sink. (b) Wide FoV including nodes with shorter or equal hop distance to the sink. (c) Complete 360-degree FoV.

In general, the orientation of a bird’s FoV can be set towards any direction, driving the movement of the bird accordingly. However, the orientation of a bird’s FoV can be affected by the presence of magnetic fields. Migratory birds that need to travel polewards turn the orientation of their head, and thus their FoV, towards the pole. In the same way, the FoV of packets is influenced by the artificial magnetic field and extends forward in the direction of the sink node as shown in Fig. 6(b).

The FoV of a packet may span from a few degrees up to  $360^\circ$ . In the proposed approach, three different FoV apertures were considered, as illustrated in Fig. 7. In the case of *narrow FoV*, packet  $i$  on node  $n$  (node in the middle) can ‘see’ only packets located on nodes with shorter hop distances than node  $n$ . In the case of *wide FoV*, packet  $i$  can ‘see’ packets located on nodes

with shorter or equal hop distances to the sink compared to node  $n$ . In the extreme case, the 360-degree FoV of packet  $i$  includes all nodes in the neighborhood of the packet.

Simulations carried out in [42] showed that the complete 360-degree FoV shown in Fig. 7(c), allowing packets to move in any direction exhibited poor performance. The reason was the *absence of orientation, and thus attraction to the global attractor*. This caused high packet concentration in the network, especially in the vicinity of source nodes, because packets became trapped in loops. Eventually, the overwhelming majority of packets were lost due to collisions in the wireless channel.

A natural way to address the problem of packet loops is to insist on using the narrow FoV where packets are allowed to move forward only, as shown in Fig. 7(a). This natural solution is too rigid, as it artificially excludes consideration of other nodes which are not necessarily on the direct path, but may have higher desirability, and makes paths towards the sink ‘too narrow’, thus increasing contention and node loading as well as minimizing the exploration of new paths. Furthermore, if the same nodes are always considered, then a higher power drain will be observed on these nodes and this may result in a shorter network lifetime. In addition, a problem may arise when front nodes either become unreachable (due to failures) or seen momentarily as unreachable (due to loss of control packets). In this case, packets reaching a dead end will be dropped.

Thus, the wide FoV of Fig. 7(b) that allows packets to be forwarded even to nodes that are placed at equal hop distance from the sink is selected in the proposed Flock-CC protocol. Each node  $n$  keeps the hop distances of its neighbors in its adjacency table,  $A_n$ .

Both the ZoR and the ZoA are redefined to involve the FoV as shown in Fig. 6(b). In this case, the ZoR is redefined as a circular zone of radius  $R_0$ , except for an area behind the packet  $i$  (‘blind’ area) that includes packets on nodes with longer hop distance than the current hosting node. The ZoR involves packets located in the queues of the heavy-gray shaded nodes of Fig. 6(b). Therefore, packet  $i$  is repelled from these packets as illustrated in Fig. 6(c). Each node  $n$  keeps the queue sizes of each node in the ZoR in its repulsion table,  $q_n$ .

Similarly, the ZoA is the outer circular zone of Fig. 6(b) that includes packets on nodes at shorter or equal hop distance compared with the current hosting node (black-shaded nodes) within the FoV. As discussed earlier, since the number of packets on (black-shaded) nodes two hops away cannot be obtained with one broadcast message, each node  $n$  records the number of packets successfully transmitted from each node in the ZoR towards nodes in the ZoA in its attraction table,  $s_n$ . In addition, for each sampling period  $T$ , each node  $n$  records the total number of all packet transmission attempts<sup>3</sup> (including retransmissions) from each node in the ZoR towards nodes in the ZoA in its transmission table,  $s'_n$ . The values of table  $s'_n$  are used for normalization as shown in the next section.

#### 3.2.4. Desirability function

The repulsive and attractive forces (exercised by packets one and two hops away respectively) are synthesized by the decision making process which is invoked by the hosting node of each packet. The synthesis of repulsive and attractive forces is captured through a desirability function. The decision making process results in selecting the most desired next hop node on the basis of avoiding or minimizing congestion phenomena on next hop nodes. The desirability

---

<sup>3</sup>We consider CSMA-like MAC protocols (e.g. IEEE 802.11) which employ a retransmission scheme to increase the reliability of the lossy wireless channel. A lost or erroneous packet is not retransmitted forever, but is dropped after a limited number of unsuccessful retransmission attempts.

function is evaluated once every sampling period and is used for each packet sent within this period.

An  $M$ -dimensional desirability vector,  $\vec{D}(k)$ , is used where  $M \leq N$ , is the number of potential new hosting nodes at the  $k$ th sampling period. The potential new hosting nodes are the nodes in the transmission range of the current hosting node  $n$  within the FoV. Each element,  $D_{nm}(k)$ , of the vector  $\vec{D}_n(k)$  represents the desirability for each node  $m, m \in \{1, \dots, M\}$  measured at node  $n$ . The desirability  $D_{nm}(k)$  of every node  $m$  in the FoV is evaluated once every sampling period  $k$  (at the start of this period) and is used for each packet sent from its hosting node within this period. The desirability of each node  $m$  evaluated at node  $n$  is given by:

$$D_{nm}(k) = s_{nm}^{norm}(k) - q_{nm}^{norm}(k), \quad (1)$$

where

$$s_{nm}^{norm}(k) = \begin{cases} \frac{s_{nm}(k)}{s'_{nm}(k)} & \text{if } s'_{nm}(k) > 0; \\ \xi & \text{otherwise,} \end{cases} \quad (2)$$

where  $\xi \in [0, 1]$  is the *spreading variable* (discussed later),  $s_{nm}(k)$  is the number of successfully transmitted packets from node  $m$  to nodes two hops away from node  $n$ ,  $s'_{nm}(k)$  is the total number of all packet transmission attempts at each node  $m$ , and

$$q_{nm}^{norm}(k) = \frac{q_{nm}(k)}{Q_m}, \quad (3)$$

where the function  $q_{nm}^{norm}(k)$  is the number of packets in the queue of node  $m$ ,  $q_{nm}(k)$ , divided by the queue capacity of each node  $m$ ,  $Q_m$ .

The function  $s_{nm}^{norm}(k)$  can be seen as a measure of quality of the wireless channel loading around node  $m$  as perceived from node  $n$ . The parameter  $s_{nm}^{norm}(k)$  ranges from 0 to 1 and represents the normalized attraction force exercised on packet  $i$  by packets that moved (successfully transmitted) from each node  $m$  to nodes two hops away from packet  $i$ 's current hosting node (i.e. these packets are now within the ZoA). When  $s_{nm}(k) \rightarrow 1$ , the channel around node  $m$  is not congested and a large percentage of packets are successfully transmitted (few packet retransmissions are observed). As  $s_{nm}(k) \rightarrow 0$ , the channel is congested and a small percentage of packets are successfully transmitted, after a large number of retransmissions.

An idle node  $m$ , i.e. with zero total transmission attempts ( $s'_{nm}(k) = 0$ ) does not provide any evidence of the wireless channel quality in the vicinity of the node. Therefore, it cannot be said whether this node is either highly attractive, i.e.  $s_{nm}^{norm}(k) = 1$ , or highly repulsive, i.e.  $s_{nm}^{norm}(k) = 0$ . The spreading variable  $\xi$  is introduced as the normalized attraction force exercised by each node  $l$  that was previously idle ( $s'_{nl}(k) = 0$ ). High values of  $\xi$  result in *packet spreading* since packets are attracted to idle nodes (most probably at the borders of the flock), whereas small  $\xi$  values lead to *coherent flock motion* (low spreading). These observations as well as further explorations of  $\xi$  values are presented in Section 4.

The function  $q_{nm}^{norm}(k)$ , with  $0 \leq q_{nm}^{norm}(k) \leq 1$ , is the percentage of queue occupancy at node  $m$  and represents the normalized repulsion force exercised on packet  $i$  by packets residing in the queue of each node  $m$  (i.e. these packets are now within the ZoR). When  $q_{nm}(k) \rightarrow 0$ , the queue is empty or nearly empty. On the other hand, as  $q_{nm}(k) \rightarrow 1$ , node  $m$  is considered congested due to high queue occupancy.

In the simple case, after evaluating the desirability of each node  $m$ , with  $-1 \leq D_{nm}(k) \leq 1$ , a packet can be forwarded from node  $n$  to node  $m^*$ , where the node  $m^*$  is a node within the FoV

with the highest desirability. Even though the above approach allows for exploitation of existing good paths, it does not allow for exploration. Therefore to this basic approach we add the last flocking behavior characteristic, namely randomness.

### 3.2.5. Randomness

Randomness (or perturbation) is that part of nature which allows for exploration, and perhaps identification of better paths. Randomness also addresses the problem of always transiting to the same (highly desirable) new hosting nodes within a sampling period (recall that desirabilities are evaluated once every sampling period  $k$ ), which can cause high queue occupancy on popular new hosting nodes. The next section includes a detailed description of how randomness is involved in the Flock-CC approach during the process of selecting a new hosting node.

### 3.3. The Flock-CC protocol

All the elements discussed above are composed together to form the Flock-CC protocol. Every time a packet is about to be sent, the decision making process is invoked by the current hosting node to determine the new hosting node. The decision process employs three stages: (a) selection of direction (forward, sideways, backwards<sup>4</sup>) using the notion of the FoV and the magnetic fields, (b) sorting of all nodes in the selected direction in descending order by their desirability, and (c) probabilistic, biased (proportional to desirabilities) selection of the new hosting node.

The position of the sink plays an important role in evaluating the direction of movement of a packet. Under the influence of the magnetic field, a packet turns its FoV towards the sink and perceives the attraction of this artificial pole. The Flock-CC protocol uses a simple rule-based strategy of providing global attraction to the sink through differentiation of attraction to nodes in the FoV. This strategy chooses the set of potential new hosting nodes among nodes placed in the FoV using the following rules, given in decreasing order of priority:

1. Choose all nodes placed *closer* (smaller hop distance) to the sink. If there is at least one node among them with available buffer space then the packet moves forward.
2. If all nodes closer to the sink either have no available buffer space or are seen to be unreachable, choose all nodes placed at *equal hop distance* to the sink. If there is at least one node among them with available buffer space then the packet moves sideways.
3. Otherwise, the packet moves backwards.

The least priority rule allows a packet to move backwards to a node outside of the FoV in the extremely rare situation where all front and side nodes are either unreachable or lack buffer space.

The proposed rule-based strategy is quite similar to the rule-based process followed in Couzin's model [8] when determining the desired direction of each individual. This process was discussed in Section 2.4.1.

After choosing the set of potential new hosting nodes, the selected nodes are sorted in descending order based on their desirabilities. To implement randomness, the new hosting node is selected using a probabilistic selection. In order to provide a fair way of selection, the probability

---

<sup>4</sup>When a wide FoV is used, it is impossible for a packet to move backwards i.e. to nodes that are not observed by a packet's 'eye'. However, this restriction can be relaxed under the extremely rare situation where all front and side nodes are either unreachable or lack buffer space.

of selecting a node can be made proportional to either the desirability of that node, or the rank of the desirability of that node in the descending order of desirabilities.

In the Flock-CC approach, randomness is invoked whenever a packet is ready to choose its new hosting node. Randomness can be implemented on the basis of selection methods used in genetic algorithms [43], for example, roulette wheel selection or rank based selection. Selection methods are used for randomly selecting members from the population of chromosomes with probability proportional to their fitness.

The behavior of the Flock-CC protocol was tested for both roulette wheel selection and rank based selection. Results showed that the rank based selection prevailed over roulette wheel selection. Thus, rank based selection was adopted in the final model of the Flock-CC protocol. Rank based selection ranks each individual  $i$  based on its fitness,  $f_i$ , and a new fitness value,  $f'_i$ , is assigned according to the rank the individual receives. *Individuals in Flock-CC are the potential new hosting nodes,  $M$ , and desirability is seen as the fitness of each potential new hosting node.* The weakest (in terms of desirability) potential new hosting node receives a new fitness value of 1, the second worst receives a value of 2, etc. and the fittest individual receives a value of  $J$ , where  $J = M$  is the number of individuals in the population. The probability  $p_i$  of an individual to be selected is equal to the new fitness of the individual divided by the total new fitness of all the individuals, as follows:

$$p_i = \frac{f'_i}{\sum_{j=1}^J f'_j}. \quad (4)$$

#### 4. Performance evaluation

This section evaluates the performance of the Flock-CC protocol using a number of scenarios with different topologies, traffic loads and instantaneous network conditions (number of idle/source nodes, position of source nodes, failing nodes).

Performance evaluations have focused on tuning the parameters  $T$  and  $\xi$ , on demonstrating the effectiveness of the Flock-CC protocol in mimicking the bird flocking behavior, in combating congestion, and showing the superiority of the protocol against related CC protocols. The Flock-CC protocol was evaluated on the basis of the performance measures presented in the next section. It is worth noting again that by design the behavior of the tunable parameters is well understood, and these were kept at a minimum. More specifically, the evaluations in three directions are:

1. *Parameter selection*: The first step was to experiment with the Flock-CC parameters: the spreading variable  $\xi$  and the sampling period variable  $T$ .
2. *Demonstration of emerging behavior, self-adaptation, robustness against failing nodes and scalability*: The second step highlights the aforementioned properties of the Flock-CC protocol using both visual representations and the performance measures.
3. *Comparative evaluation*: Finally, the Flock-based protocol was compared against related (nature-inspired and conventional) congestion control approaches.

Results were collected from simulation studies conducted using the ns-2 network simulator [44]. The relevant simulation scripts can be found in [42].

#### 4.1. Evaluation setup

##### 4.1.1. Topologies

The evaluation topologies consist of a considerable number of nodes which are deployed either in a 2D lattice, or in a random manner, over an area of  $400 \times 800 m^2$ . A 300-node lattice topology is shown in Figs. 8(a) and (b). Lattice topologies of 200 and 400 were also considered.

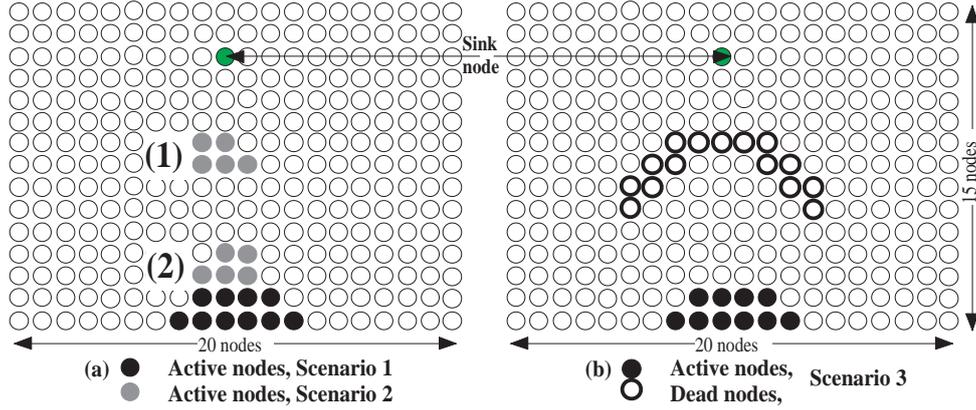


Figure 8: A 300-node evaluation topology with 3 scenarios of operation: (a) Scenarios 1 and 2 with 10 active nodes but different source node placement, (b) Scenario 3 with 10 active nodes and node failures (after 40s). Short description of each scenario is given below.

In lattice topologies, neighboring nodes were placed at the same distance ( $25 \cdot \sqrt{2}m$ ) from each other, in such a way that each node has at most 8 neighboring nodes one hop away. In this way, a dense topology was developed with uniform node placement. The lattice topology was used in the overwhelming majority of scenarios so as to better understand and evaluate the behavior of the flock-based mechanism. In order to experiment with more realistic topologies, nodes were also deployed in a random manner, as described later.

All nodes were homogeneous and identical in hardware and software setup. The radio propagation range of each node was 50m. A two-ray reflection propagation model was used which accounts for the effect of multipath and fading. The radio model assumes that the radio can lock onto a sufficiently strong signal in the presence of interfering signals. The control packet size and the data packet size were set to 10 and 50 bytes respectively. The CSMA-based IEEE 802.11 MAC protocol, provided in the ns-2 simulator, with an exponential backoff policy was adopted, having RTS/CTS packet exchanges disabled for energy saving and low signalling overhead purposes. Two transmission rates of 2 Mbps and 250 Kbps were used. In the former case, the queue size of each node was set to 50 packets, while in the latter case it was set to 15 packets.

##### 4.1.2. Scenarios

The Flock-CC protocol was evaluated under four distinct congestion scenarios to best understand the protocol's behavior and dynamics in responding to the different congestion conditions (persistent and transient hotspots) that could be found in sensor networks. Each scenario corresponded to events occurring in different locations, involving different sets of source nodes. The first three scenarios were related to the 2D lattice topology, while the fourth scenario involved

random topologies. All scenarios were tested on topologies of 300 nodes (scenarios 1 – 3 are illustrated in Fig. 8), while scenario 1 was additionally tested on topologies of 200 and 400 nodes.

In all scenarios, three different traffic loads were considered: low, high and extreme. At high MAC transmission rates (2 Mbps), each sender node was allowed to generate constant bit rate (CBR) traffic of either 25 (low load), or 35 (high load), or 45 (extreme load) pkts/s when triggered by an event. These three cases were considered as slightly congested, congested, and heavily congested network conditions, respectively. In each scenario, all nodes were sending at the same rate. The corresponding traffic rates for the transmission rate of 250 Kbps was 10, 15 and 20 pkts/s.

Each scenario, using different combinations of parameter values, was executed 30 times and the mean values of the metrics over all scenarios are presented below. In selected figures, the mean values are supplemented with 95% confidence intervals. When dealing with non-normal distributions, ‘confidence intervals’ are used as a measure of variability instead of using the standard deviation. A 95% confidence interval contains the middle 95% of the numbers in a list.

The *first scenario* (scenario 1) emulated persistent congestion phenomena and involved a set of 10 closely placed black shaded source nodes shown in the middle bottom part of Fig. 8(a). The same setting was used in the larger topologies. In many real-world WSNs applications, it is quite common to have source nodes closely placed to each other being activated almost at the same time when an external event (e.g. a disaster-related event such as fire, earthquake) is detected. In these cases, persistent hotspots are created in the vicinity of source nodes.

The *second scenario* (scenario 2) emulated transient congestion phenomena. Initially, the front set (1) of 5 grey shaded source nodes of Fig. 8(a) (close to the sink) were sending packets. In this way, a small, transient congestion hotspot (of heavily loaded queues and wireless channel) was created around and in front of source nodes towards the sink. In order to demonstrate the ability of the packets to manoeuvre around the hotspot, the bottom set (2) of 5 grey shaded source nodes started sending packets at  $t = 50$ s. At  $t = 70$ s, the upper set of nodes (1) stopped sending packets.

The *third scenario* (scenario 3) emulated an extreme scenario of node failures. The black shaded source nodes located in the bottom part of the network of Fig. 8 (b) were initially sending packets (as in the first scenario), while bold circled nodes in the middle of the network failed at  $t = 40$ s. The aim was to demonstrate the ability of the packets to manoeuvre around the dead zone. The shape of the dead zone (involving failed nodes) was chosen so as to demonstrate the ability of the algorithm to manoeuvre around areas where packets can get trapped and as such cannot move forward.

Finally, the *fourth scenario* (scenario 4) was used to demonstrate the ability of the Flock-CC protocol to also perform well when the underlying sensor network infrastructure is random. This scenario involved two randomly generated uniform topologies with 10 closely placed source nodes and 1 sink node. The sink was randomly selected from nodes in the network. The two random topologies were: (a) a sparse topology of 175 nodes where the average node degree<sup>5</sup> was 4.72 and (b) a dense topology of 300 nodes where the average node degree was 8.85. An indicative figure of the random topology used can be found in [42].

All scenarios were used for parameter selection and comparative evaluations among all approaches. Additionally, the second scenario was used to evaluate the emerging behavior and

---

<sup>5</sup>The node degree is the number of connections the node has to other nodes in the network.

the self-adaptive behavior of the Flock-CC model. Furthermore, the third scenario was chosen to demonstrate the robustness of the Flock-CC protocol against failing nodes, even under extremely undesirable situations.

#### 4.1.3. Variables

The spreading variable  $\xi$  was chosen to range from 0 to 1 to allow experimenting with zero spreading to full spreading of the flock. The time between successive control packets,  $T$  (sampling period), was assigned the values 0.5, 1.0, 1.5 and 2.0s in all approaches. The selection of  $T$  to be less than or equal to  $2s$  was guided by the desire to maintain responsiveness to changes in the network state. It was also desirable to avoid overwhelming the network with control packets, thus  $T$  should not be very low, hence  $T \geq 0.5$  was also selected.  $T_{lost}$  was set at  $3T$ .

#### 4.1.4. Related approaches for comparison

From the perspective of network layers, the Flock-CC approach provides multi-path routing and congestion control capabilities for WSNs. Flock-CC was compared to five protocols that can be adopted during a congestion crisis period: (a) a conventional congestion-free multi-path routing protocol based on shortest paths (baseline scenario), (b) a typical congestion-aware routing protocol that routes packets over multiple paths, choosing each time the least congested next hop node in terms of queue length (among nodes involved in the shortest paths to the sink), (c) a well known single-path routing protocol for adhoc networks, AODV [37], (d) an ant-based algorithm for multi-path reactive and proactive routing in mobile ad-hoc networks which is based on ideas from Ant Colony Optimization, AntHocNet [33], and (e) an ant-based multi-QoS routing protocol, AntSensNet [38]. These strategies/protocols are summarized in Table 1.

Table 1: Congestion control strategies for comparative analysis.

<i>Strategy</i>	<i>Congestion detection</i>	<i>Routing scheme</i>
Flock-based Congestion Control (Flock-CC)	Queue loading, MAC layer collisions and retransmissions	Multi-path
No Congestion Control (NCC) shortest paths	None	Single-path
Congestion-Aware Routing (CAwR)	Queue occupancy	Multi-path
AODV [37]	None	Single-path
AntHocNet [33]	MAC layer activity	Multi-path
AntSensNet [38]	MAC layer activity	Multi-path

#### 4.1.5. Measures of performance

Three performance measures for congestion control approaches were taken into account: the packet delivery ratio (PDR), the end-to-end delay (EED), and the energy tax. Packet delivery ratio is defined as the ratio of the total number of packets received by the sink to the total number of packets transmitted by source nodes, without taking into account retransmissions. End-to-end delay is defined as the time taken for a packet to be transferred from a source node to the sink.

The energy tax metric calculates the average energy consumption per node per delivered packet, measured in mJoules/delivered packet at the sink.

#### 4.2. Parameter selection

This section evaluates the performance of the Flock-CC protocol and provides parameter selection guidelines. Only topologies of 300 nodes are discussed in this section. Results regarding larger topologies are presented later.

The Flock-CC protocol development was investigated in several stages (detailed results and conclusions can be found in [42]). The notions of the FoV, the magnetic field and randomness were progressively added to the protocol. Each element contributed a significant improvement in the performance of the Flock-CC protocol. In the absence of orientation and attraction to the sink (no FoV and magnetic field), packets were wandering around in the network and got trapped in routing loops. Eventually, the overwhelming majority of packets were lost due to collisions in the wireless channel. The introduction of the FoV and the magnetic field caused a steep rise in PDR of approximately 55% (in scenario 1). Increasingly, the introduction of perturbation provoked further increase in PDR of up to 10% (scenario 1).

More specifically, perturbation was realized on the basis of two random selection techniques: roulette wheel selection and rank based selection. Rank based selection was shown to outperform roulette wheel selection, exhibiting slightly higher PDR values (1 – 3%) compared to the rank based selection across all three scenarios, for almost every value of  $T$  and  $\xi$ . Thus, the rank based selection was incorporated in the Flock-CC protocol.

Below, we present the results and discuss the performance of the final model of the Flock-CC protocol involving all flocking characteristics.

##### 4.2.1. Packet delivery ratio (PDR):

Fig. 9 illustrates the PDR for all three scenarios of Fig. 8 with respect to parameters  $T$  and  $\xi$ . The results presented here consider sending nodes generating traffic at the rate of 35 pkts/s (high traffic load, i.e. the network can be considered congested). Similar observations apply for low (25pkts/s) and extreme (45 pkts/s) traffic loads, thus the figures were omitted.

In scenarios 1, 2 and 3, the highest value of PDR (around 82%, 93% and 78% respectively) was observed for  $\xi = \{0.5, 0.75\}$ . Significant performance deterioration was exhibited for low spreading values,  $\xi = \{0, 0.25\}$ . In the first scenario, poorer performance was also observed for high spreading,  $\xi = 1$ , but not at the same scale. At low  $\xi$  values, weak attraction was exercised by each idle node, even if these nodes were able to accommodate incoming traffic load. Since idle nodes are usually found at the borders of the packet flock, low  $\xi$  values ‘force’ packet flocks to move in a coherent formation. In this case, a number of available paths were left unexploited while other (popular and perhaps, shorter paths to the sink) faced overloading, thus resulting in low PDR. On the other hand, at very high  $\xi$  values (close to 1), each idle node was exercising strong attraction (causing high packet spreading towards the border of the flock). In this case, two problems may occur: (a) There is no evidence of the channel quality in the vicinity of each idle node, thus if the channel is busy a high number of losses may arise. (b) A large number of packets are attracted to the borders of the flock (left and right), where the majority of idle nodes reside. At the same time, packets are also attracted to the sink due to the inherent attraction of the magnetic field. Thus, packets ‘flying’ at the borders of the flock collide with ongoing packets ‘flying’ through the center of the flock, resulting in a low PDR. The slight problem with high  $\xi$  values was more pronounced in the first scenario, where packet spreading was apparent on a larger scale.

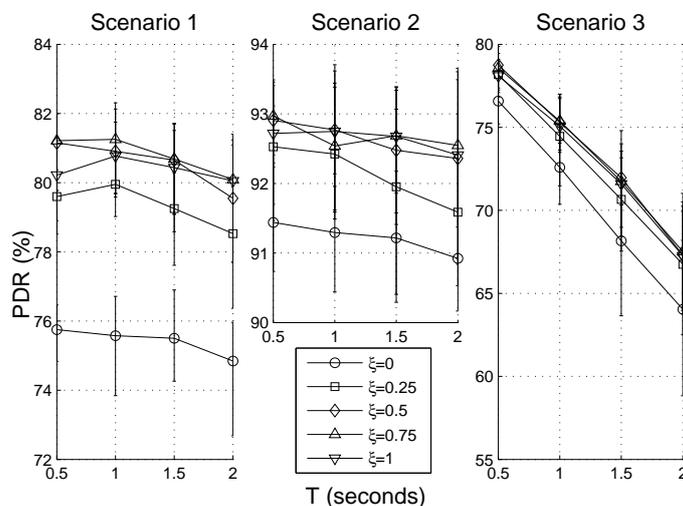


Figure 9: Packet delivery ratio (PDR), Flock-CC with all flocking characteristics, all scenarios (high traffic load).

An interesting observation is that, for scenarios 1 and 2, PDR reached around 82% for  $T = 1$ s, while it exhibited slight decrease for  $T = 1.5$ s and  $T = 2$ s. On the other hand, for the third scenario (failing nodes), the PDR reached 78% for  $T = 0.5$ s (regardless of the value of  $\xi$ ) with steep decreasing trends of 10 – 11% as  $T$  approached 2s. For low  $T$  values where control packets were broadcasted quite frequent, the Flock-CC mechanism was kept updated regarding the network state. Frequent updates are of prime importance in scenarios with failing nodes. This of course happens at the cost of higher energy expenditure-overhead, however with a reduced number of retransmissions expected.

In scenario 2, lighter congestion phenomena occurred compared to scenarios 1 and 3 due to the low number of closely located sending nodes. Traffic load injected into the network in the two small hotspots of scenario 2 did not overload the network resources as much as the injected traffic load in the hotspot of scenarios 1 and 3.

#### 4.2.2. Packet loss:

In scenario 1 (Fig. 10), the number of collisions dominated the number of buffer overflows for all values of  $T$  and  $\xi$ , while the overwhelming majority of packets were lost within the hotspot area. Quite often, due to the shared wireless medium, packets collide before reaching the receiver node. Thus, buffers are less often filling up. The coherent packet movement (through popular nodes in the shortest paths) observed at low  $\xi$  values resulted in high number of overflows. Increasingly, the high number of overflows led to larger number of retransmissions, and thus increased number of collisions. On the other hand, the effect of packet spreading which occurs for high  $\xi$  values resulted in a slight increase of collisions.

For scenario 1, the lowest *total* packet loss was observed at  $\xi = \{0.5, 0.75\}$ . As shown in Fig. 9, these two  $\xi$  values consistently achieved the best performance in terms of PDR. The reason for the efficiency of these two  $\xi$  values is found in the balanced way of packet movement between coherent formation and spreading. The attraction to idle nodes is strong enough to exploit their

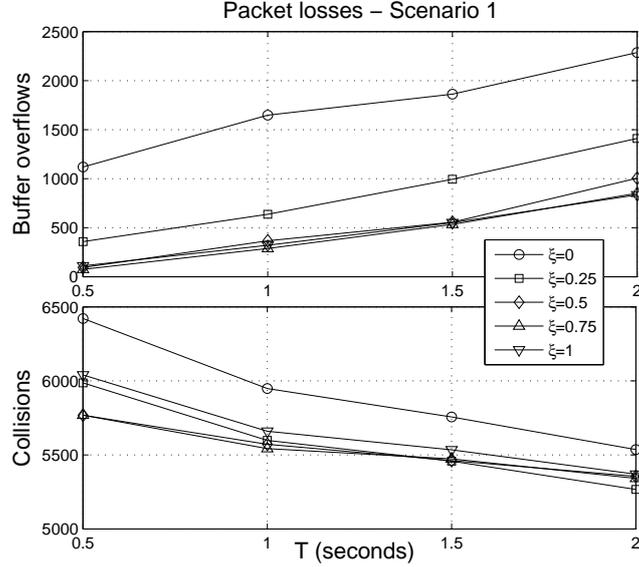


Figure 10: Buffer overflows and collisions, scenario 1, high load.

resources (buffer space, channel capacity) without causing overconsumption.

Another observation in scenario 1 is that, the increase of  $T$  provoked an increase in the number of buffer overflows and a reduction in the number of collisions. Recall that the desirability function is evaluated once every sampling period and packets to be sent within this period chose the most desirable node. At high  $T$  values, where the desirability function was evaluated infrequently, packets tended to choose with high probability the same new hosting node over a longer time period leading to overloading (high queue occupancy) of the chosen node. On the other hand, at low  $T$  values, packets spread more frequently to different new hosting nodes eliminating buffer overloading but causing an increase of channel collisions around the chosen nodes.

Fig. 11 shows buffer overflows and collisions in scenario 3 where major node failures were simulated. Here, the number of collisions was slightly higher than the number of buffer overflows for all values of  $T$  and  $\xi$ , while it is worth noting that the majority of packets were lost around failing nodes. In addition, it can be observed that the influence of  $\xi$  values on both overflows and collisions was the same as in scenario 1. On the other hand, scenario 3 shows a steep rise in the numbers of both overflows and collisions as  $T$  is approaching 2s. In scenarios with failing nodes, as the time between successive control packets grew and the evaluations of desirabilities were performed in a slower pace, the offered quality degraded sharply. The reason is attributed to the phenomenon described in the previous paragraph.

Overall, scenario 3 exhibited higher number of buffer overflows and lower number of collisions than scenario 1 for almost all values of  $T$  and  $\xi$ . This is because in scenario 1 packets were spread in the network throughout the duration of the scenario causing higher collision rates. On the other hand, in scenario 3, the incidence of node failures led packets to find alternative paths to the sink by bypassing the ‘dead’ zone as shown in Fig. 15. In this case, the movement of packets through the new paths to the sink was necessarily cohesive due to the new topology,

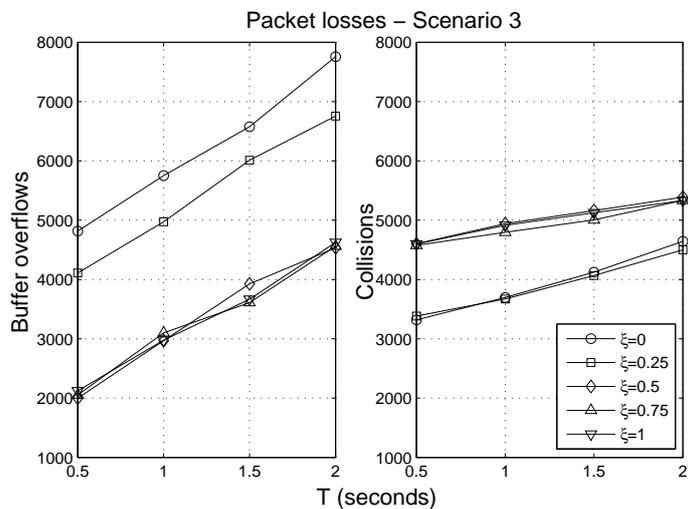


Figure 11: Buffer overflows and collisions, scenario 3, high load (major node failures).

which resulted in this particular example. Thus, packets reaching a diagonal position from the sink had only one (highly attractive) node in their FoV placed closer to the sink. The attraction of packets to nodes along the diagonal led to narrow packet movement and thus, high number of buffer overflows and a lower number of collisions.

#### 4.2.3. End-to-end delay (EED):

The Flock-CC protocol achieved the lowest EED ( $\approx 335$ ms) for  $\xi = 0.75$  and  $T = 0.5$ s. It is worth noting that EED was calculated using packets successfully delivered at the sink, thus in the cases of high packet loss, EED could not be accurately calculated. Hence, results for  $\xi = \{0, 0.25\}$  (coherent packet movement leading to high number of buffer overflows and long delays) were omitted. The significantly low number of losses (especially for low  $T$  values) led to fewer retransmissions, and thus faster transitions of packets to the sink. The rise in  $T$  values was accompanied by an increase in packet loss leading to increased retransmissions and thus larger delays.

#### 4.2.4. Energy tax:

Results concerning the energy tax spent per delivered packet confirmed the intuitive reasoning (made earlier) that frequent updates at low  $T$  values led to higher energy tax. The highest energy tax per delivered packet was spent for  $T = 0.5$ s while slightly less energy was consumed for  $T = 1.0 - 2.0$ s. However, the difference was marginal reaching up to only  $1.5\mu J$  per delivered packet. The energy gain was not significant because at high  $T$  values large number of buffer overflows were observed resulting in higher number of retransmissions which in turn spent much energy. The changes in energy tax values were fairly insensitive to  $\xi$ , but the values  $\xi = \{0.25, 0.5, 0.75\}$  exhibited better behavior.

#### 4.2.5. Concluding remark:

Taking all the results of this section into consideration, a good compromise value for  $\xi$  is 0.75. The value of  $T$  can be set to  $T = 0.5s$  since the gain from minimizing buffer overflows (observed at  $T = 0.5s$ ) is larger than the gain from minimizing energy expenditure (observed at  $T = 2s$ ). Nevertheless, if slight energy gains are of utmost importance, parameter  $T$  can be set to higher values, e.g. equal to 1.0s. Alternatively, an adaptive mechanism can be used to dynamically adjust the value of  $T$  according to changes in the network state. Further experimentation regarding other networking conditions and random topologies are presented next, and set of simple guidelines for tuning parameter values is given at the end of this section.

#### 4.2.6. Low data rate WSNs:

The applicability of the Flock-CC protocol on low data rate WSNs (e.g. 250 Kbps) was also evaluated on the basis of scenario 1 to 3. Low, high and extreme traffic loads were considered. The results were similar to those obtained with high data rates of 2 Mbps (see Fig. 9). The most significant difference was that in low data rate WSNs the overwhelming majority of packet losses were attributed to collisions. Due to the low transmission rates, buffers were occasionally filling up, and rarely leading to overflows.

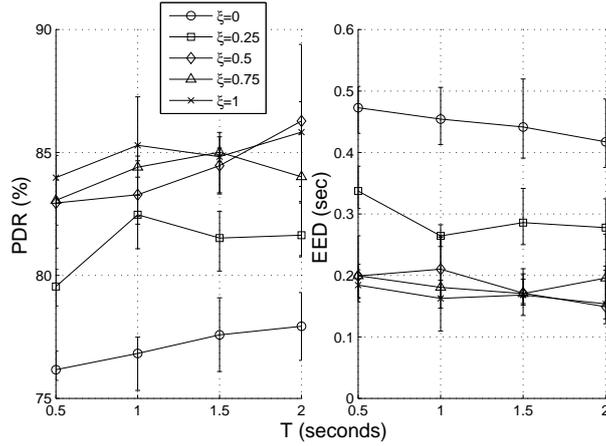


Figure 12: Packet delivery ratio (PDR) and end to end delay (EED), scenario 4 (random dense topology), 25 pkts/s.

#### 4.2.7. Random topologies:

The performance of the Flock-CC protocol was also evaluated on sparsely and densely deployed random topologies (scenario 4). The results for the dense topology in terms of PDR and EED are illustrated in Fig. 12.

The traffic load transmitted from each sender was 25 pkts/s (low load). Due to the high density of nodes (average node degree= 8.85), the PDR was around 10% lower compared to the grid topology of scenario 1 (node degree= 8). As expected, the reduction in PDR was expected due to the extremely high number of collisions. The highest PDR was achieved for  $\xi = 0.5$  and  $T = 2s$ , while acceptable levels of PDR were achieved for  $\xi = \{0.75, 1\}$  and  $T = 1 - 2s$ . The same combination of values led to a low EED. The decrease in the number of collisions as  $T$

approached 2 was reflected in the increasing trends of PDR and the decreasing trends of EED. There was a concurrent, but slight increase in the number of overflows which did not have a major influence on PDR and EED.

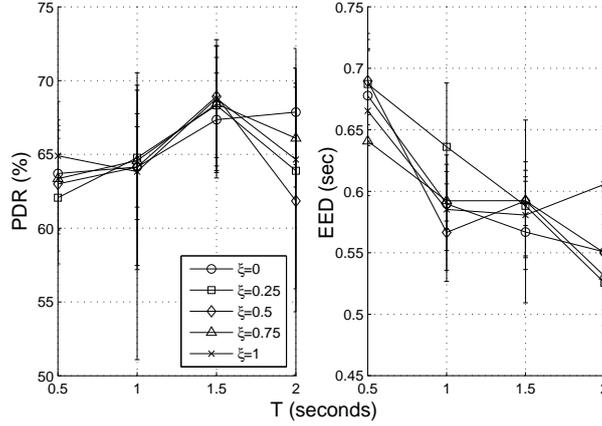


Figure 13: Packet delivery ratio (PDR) and end to end delay (EED), scenario 4 (random sparse topology), 25 pkts/s.

Results for the sparse topology are shown in Fig. 13. As can be seen, the PDR was reduced by up to 20% compared to the dense topology while the EED exhibited increases of up to 0.5s. The poor performance experienced in the sparse topology was attributed to the limited network resources that led to the steep increase of buffer overflows. More specifically, in the sparse topology, buffer overflows increased at almost 10 times regardless of the values of  $T$  and  $\xi$ . This behavior was expected, since the sparse placement of nodes forced packets to move through a small number of paths to the sink. Therefore, the traffic load injected into these paths could not be accommodated by the limited buffer capacity of nodes in these paths. This is of course a design and dimensioning issue, and for example increasing buffer capacity can improve the performance in terms of PDR at the expense of longer delays (higher EED).

#### 4.2.8. Recommended setting of $\xi$ and $T$ :

The results for dense and sparse topologies suggest that a good compromise value for parameter  $\xi$  should be in the range 0.5 to 0.75. The upper value seems more preferable since it was found to be effective and efficient in scenarios with grid topologies with low and high MAC transmission rates. *For the rest of this study, the value of  $\xi$  is set to 0.75.* On the other hand, the value of  $T$  can be set according to the network characteristics. In a controlled environment (e.g. a predefined topology) like a grid topology, for network devices with characteristics as in the simulations, the value of  $T$  can be set to 0.5s (as discussed above), or follow an adjustable approach: Initialize  $T$  to 2s. If a failure (unreachable node) is sensed (see discussion for  $T_{lost}$ ), then  $T$  should be reduced to 0.5s for fast recovery, while  $\xi$  can be left unchanged. After a given period of time with no other node failures sensed, the node can reset  $T$  either directly or progressively to 2s for energy saving purposes. Since the energy gains from this adjustable approach are marginal, the study of this approach is left for future work. Also, it has to be noted that the adjustable approach increases complexity on each node. *For the rest of this study, the value of*

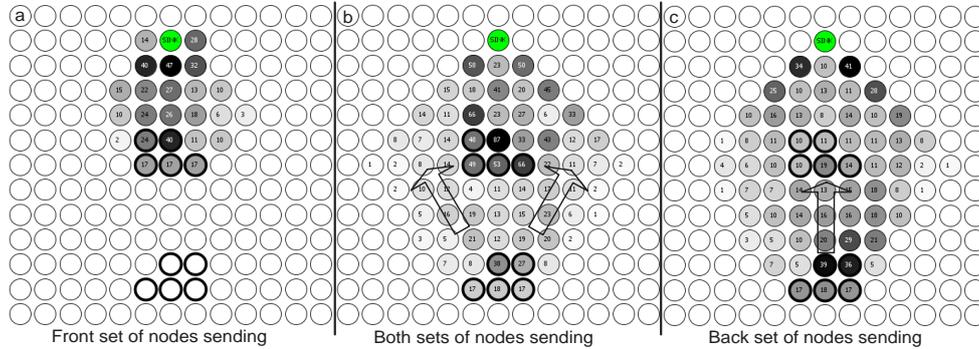


Figure 14: Emergent behavior: Visual representation of moving packets in scenario 2 (high load). The number of packets visiting each node within a 1-second time slot snapshot is indicated inside every node. Darker colors indicate higher number of packets.

$T$  is set to 0.5s. Note that in random topologies, the value of  $T$  can be increased to 2s for maximizing PDR and minimizing energy consumption at the same time. More conclusive results can be extracted by evaluating the Flock-CC approach on a real testbed, which will increase the confidence in the recommended parameters robustness under real implementations.

#### 4.3. Emergent behavior of group-level transitions

This section investigates the emergent behavior of the collective motion of packet flocks through the network using the scenarios 1 – 3. The emergent behavior can be perceived: (a) directly by the visual representation of flock movements and (b) indirectly on the basis of the performance evaluation metrics. The visual representations of Figs. 14 and 15, using the Flock-CC protocol, depict the sink node in the upper side of each figure, while active nodes are highlighted by bold circles.

Evaluation results based on scenario 2 and high traffic load demonstrated the obstacle avoidance behavior of packet flocks. Fig. 14(a) depicts the bird-like motion of packets generated by

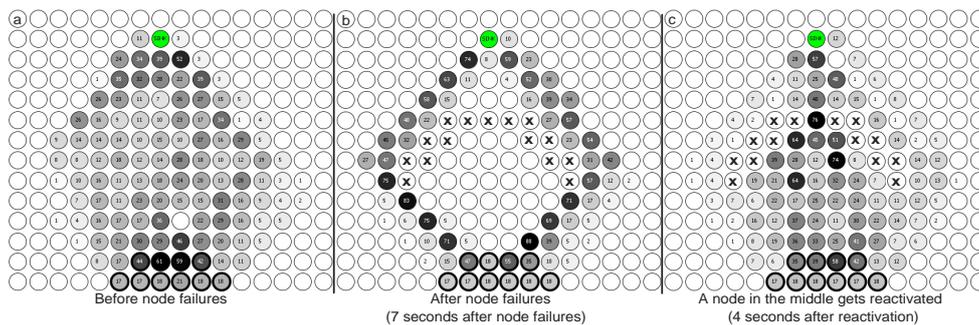


Figure 15: Emergent behavior: Visual representation of moving packets in the presence of failing nodes in scenario 3 (high load) The number of packets visiting each node within a 1-second time slot snapshot is indicated inside every node. Darker colors indicate higher number of packets.

the front set of the five dark grey shaded nodes. In line with the flocking behavior, packets were spread in the network, but not too much, choosing a number of paths to the sink. Fig. 14(b) shows that after the activation of the bottom set of source nodes, the new flock of packets splits apart into two separate groups in order to avoid colliding with packets generated from the front set of source nodes. The direction of motion of the two sub-flocks is highlighted by transparent arrows. Furthermore, when the front set of nodes stopped transmitting, the two sub-flocks re-joined to a single, more coherent flock moving over the area that used to be a congestion region Fig. 14(c).

Fig. 15 shows the emergent behavior of moving packets before and after node failures in scenario 3. The emerging reorganization of moving packets after node failures is shown in Fig. 15(b), where the single flock of packets splits into two smaller compact groups, both of which ‘fly’ around the ‘dead’ zone. For visualization purposes, an extension to the third scenario was made. At some point after failure, the center failed node became alive, thus creating a potential new path down the middle. The adaptive motion of packets through three alternative paths to the sink is shown in Fig. 15(c). After the reactivation of the node in the middle, the magnetic field lines passing through the ‘hole’ helped packets perceive the newly created shortest path leading to the sink. Network resources inside the ‘horse-shoe’ area and close to the ‘hole’ were incapable of accommodating the whole generated traffic load. In the case of a routing protocol forwarding packets on shortest paths, congestive phenomena would be expected to arise inside the ‘horse-shoe’ area towards the ‘hole’. However, under the Flock-CC protocol, packets moving through the ‘horse-shoe’ area exhibited both repulsive as well as attractive forces, readjusting the flows both through the ‘horse-shoe’, as well as around it, in an adaptive dynamic fashion.

This section focused on the ability of moving packets to mimic the obstacle avoidance behavior of bird flocks. Visual representations demonstrated the *self-adaptiveness* of the proposed approach to changing traffic (Fig. 14) and network conditions (Fig. 15). In particular, it was shown that packets were dynamically moving apart to avoid queue and channel loading phenomena or node failures (thus causing traffic spreading among available paths to the sink) and moving back together once the congested or faulty situations elapsed. As a side effect, the network energy expenditure was ‘fairly’ shared among nodes along these paths, resulting in increased network lifetime. If packets were routed via a single path, or were spread among multiple paths in an unbalanced manner, some of these paths could progressively become over-utilized, causing an unbalanced network energy expenditure.

Fig. 16 visualizes the packet movement in scenario 3 (at high traffic load) when one or more of the flocking behavioral rules outlined in Section 3.1 were removed from the Flock-CC model. The objective here is to emphasize the need of these simple rules and demonstrate that the emergent behavior is achieved by the set of all the behavioral rules. The first experiment excludes only randomization (rule 4). The second experiment excludes only local interactions i.e. repulsion and attraction forces (rules 1 and 2). The third experiment excludes both randomization and local interactions. Note that visual representations concerning the third experiment were omitted since they were almost identical to the second experiment. Figs. 16(a) and (d) illustrate the packet movement before node failures. The feature of exploration, which emerges from the *randomized selection* of new hosting nodes, allows for traffic distribution through alternative paths to the sink. This feature was apparent in Fig. 15(a), where the full Flock-CC model involving all flocking characteristics was used. On the other hand, as shown in Fig. 16(a), the exclusion of randomization eliminated packet spreading. The problem was worsened by removing only local interactions as illustrated in Fig. 16(d). Local interactions form the flock and also allow packets ‘exploit’ previously idle nodes (using attraction forces) and avoid congested nodes (using repul-

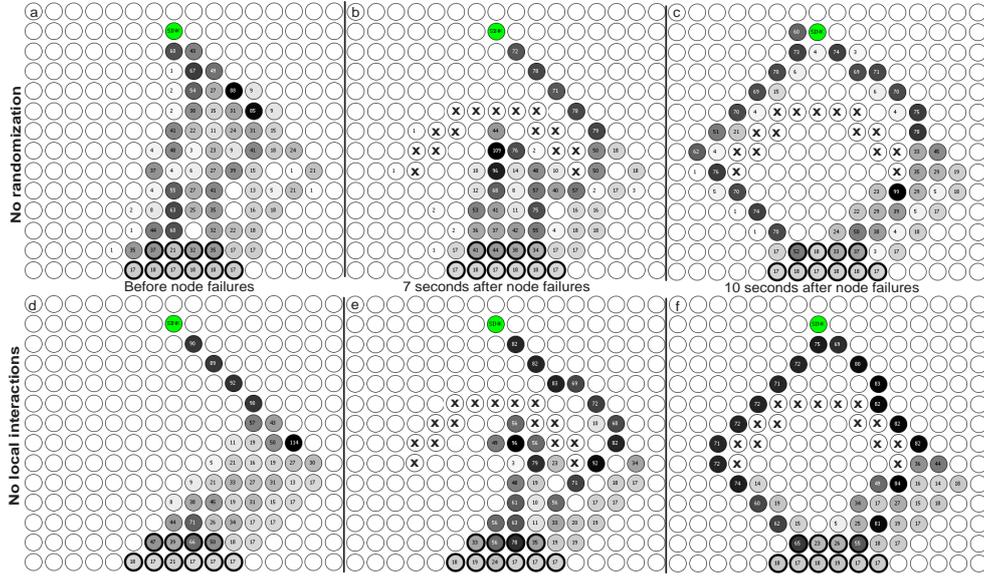


Figure 16: Lack of emergent behavior: The effect of flocking characteristics on packet flocks movement in scenario 3 (high load) when there is: (a)-(c) no randomness (rule 4), (d)-(f) no local interactions (rules 1 and 2).

sive forces). When these forces were omitted, packet paths became too coherent, thus causing deterioration of PDR and EED. The effects on performance measures is discussed below.

Figs. 16(b), (c) and (e), (f) illustrate the packet movement after node failures in scenarios with no randomization, or local interactions respectively. In scenario with no randomization, packets found the alternative path to the sink faster than the scenario with no local interactions. As can be seen in Fig. 16(b), 7 sec. after node failures, a few packets found the way to move at the left hand side of the ‘dead zone’. This happens due to the higher tendency of packets to spread as a result of the attraction and repulsion forces. As shown in 15(b), in the full Flock-CC model, the reaction to failures was more effective since 7 sec. after node failures the packets found both alternative paths to the sink.

The emergent behavior was also perceived through performance evaluation metrics. Fig. 17 compares the full Flock-CC model involving all flocking characteristics against the three subvariants of the Flock-CC model described above.

Fig. 17 shows that the full Flock-CC model exhibited the highest PDR, especially for the scenarios 1 and 3. The exclusion of randomization from the Flock-CC model led to the deterioration of PDR. In scenario 1, the reduction was ranging from 9% for  $T = 0.5s$  to 17% for  $T = 2s$ , whereas in scenario 3 the reduction was from 5% to 11%. In scenario 2, where light congestion phenomena occurred, there was no significant deterioration of quality in the absence of randomization. Traffic distribution through perturbation was highly effective especially in scenarios with large hotspot areas (scenarios 1 and 3). *Thus, the gain (low buffer overflows and collisions) from path exploration (rule 4), emerging from the randomized selection of new hosting nodes, is significant.*

Further PDR deterioration occurred by removing only local interactions. For scenario 1, the

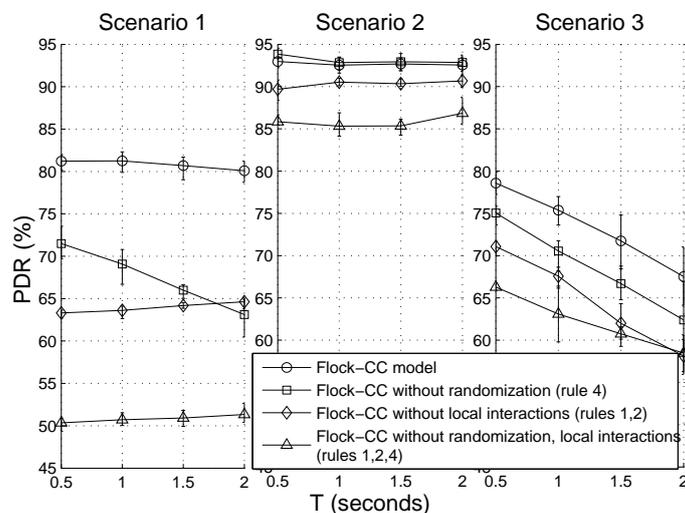


Figure 17: Emergent behavior: The effect of flocking characteristics on packet delivery ratio (PDR).

reduction was around 15%, for scenario 2 the reduction was from 3% to 5% and for scenario 3 the reduction was from 8% to 10%. Clearly, the gain from local interactions among packets is even higher than what is achieved with randomization. In particular, a steep decrease in the numbers of buffer overflows and collisions emerges due to the social activity between neighboring packets (rules 1 and 2).

As expected, the exclusion of both randomization and local interactions led to further deterioration of the PDR. Furthermore, the exclusion of the magnetic field (rule 3) and the FoV had devastating effects on PDR. Results are omitted due to the extremely low values of PDR, which even fell to 1% – 2% (scenario 1, high load).

Results concerning EED evaluations exhibited similar behavior to PDR as shown in Fig. 18.

#### 4.4. Robustness in failure prone environments

Sensor nodes are prone to failures, mainly due to fabrication process problems, environmental factors (disasters), enemy attacks, and battery power depletion. The proposed approach exhibits robustness against node failures due to the inherent tendency of individuals to follow other flockmates that manoeuvre to avoid obstacles such as failing nodes. The third scenario was used to demonstrate the robust nature of the flock-based approach.

Fig. 15 shows network snapshots before and after node failures. It is apparent that the Flock-CC approach displayed outstanding flocking behavior in the presence of numerous node failures (selected in a horse-shoe orientation, which after failures traps packets), and exemplified all of the characteristics of a bird flock in terms of obstacle avoidance and manoeuvring around the zone of dead nodes.

#### 4.5. Scalability

The Flock-CC protocol scalability refers to the ability of the protocol to support WSNs expansion to include more nodes that might be anticipated during the initial network design stage.

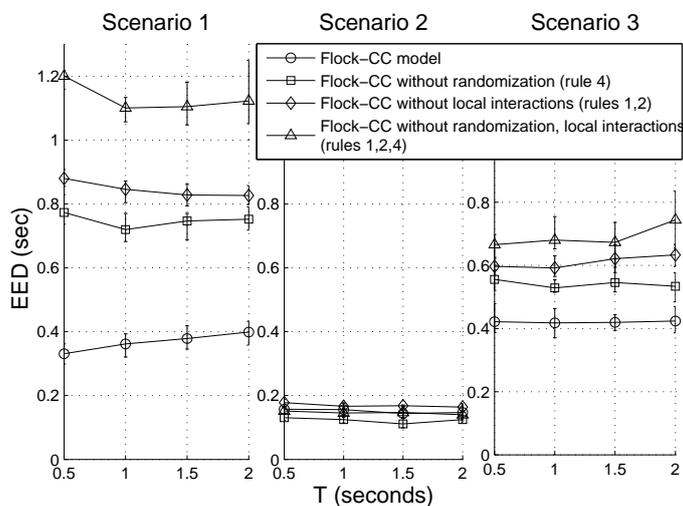


Figure 18: Emergent behavior: The effect of flocking characteristics on end-to-end delay (EED).

This section presents the results obtained when testing Flock-CC on different sizes of networks consisting of 200, 300 and 400 nodes. In each case, a grid topology was used, having a set of 10 source nodes closely placed in the middle bottom part of the network.

As can be seen in Fig. 19, there was a slight rise of PDR values with the increase of network size. The Flock-CC protocol is shown to perform better in large scale networks compared to small scale networks (since diversity increases). As the number of nodes in a network scales up, the amount of available resources increases and packets are able to spread widely through the network, thus minimizing packet losses, primarily due to buffer overflows. On the other hand, in small scale networks packets move in more restrictive coherent formations that increase the likelihood of buffer overflows due to the limited number of paths to the sink. Fig. 19 illustrates the effect of network size on EED. As the network size increases, the time needed for packets to reach the sink shortens (i.e. lower EED). This is because the increase of network resources, and as a result the increase of alternative paths to the sink, reduces the buffer occupancy (and buffer overflows) at each node. Therefore, packets face lower queueing delays and travel from source nodes to the sink at a fast pace. Finally, as expected, EED delay increases with the increase in traffic rates due to the rise in buffer occupancy at each node.

#### 4.6. Comparative studies

The proposed Flock-CC protocol was quantitatively compared against NCC, CAwR and AODV [37] protocols, as well qualitatively compared against AntHocNet [33] and AntSensNet [38]. It is worth pointing out that the comparative evaluation does not include issues such as robustness and scalability. However, due its nature Flock-CC can be expected to outperform (by far) these schemes. The characteristics of each protocol are summarized in Table 1.

Results of Figs. 20 (a)-(f) show that the proposed flock-based approach clearly outperformed NCC (no congestion control) and CAwR (congestion-aware routing) protocols in terms of both PDR and EED, for all traffic loads and scenarios. In addition, as shown in Figs. 20 (g)-(i), Flock-

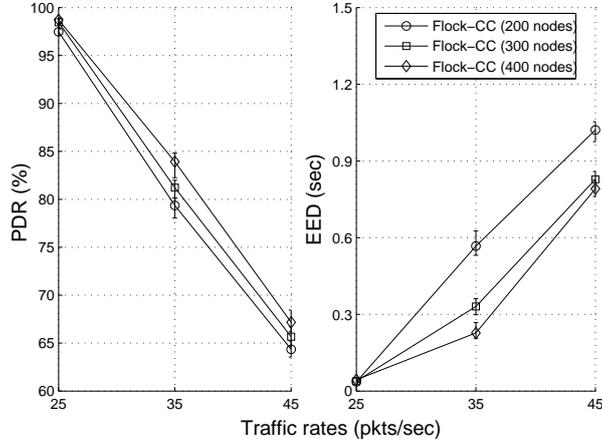


Figure 19: Scalability: The effect of network size in packet delivery ratio (PDR) and end-to-end delay (EED).

CC consumed less energy per delivered packet in low traffic rates (25 pkts/s) for all scenarios and slightly increased energy in higher loads. The results for the AODV protocol were omitted due to the very low performance of the protocol compared to the other protocols shown in Fig. 20. Note that AODV suffers from two limiting factors: (a) it is an end-to-end routing protocol that routes packets from a given source to a given destination through the shortest path connecting each other; (b) it is a reactive routing protocol, meaning that the path needs to be established from source on demand (when source has packets to transmit) using a series of control messages. These control messages are broadcasted in all directions, which is useless when they are sent in the opposite direction of the destination, causing high bandwidth consumption.

From the perspective of PDR, Figs. 20 (a)-(c) show that the Flock-CC approach delivered around 15%, 23% and 19% more packets for scenario 1 than the NCC protocol under low, high and extreme traffic loads respectively. The difference in PDR between scenarios 2 and 3 was smaller. Similarly, in scenario 1, the Flock-CC approach achieved 2% to 8% higher PDR (better performance in low loads with decreasing trends in extreme loads) compared to the CAWR protocol. Differences of 2% to 4% in PDR between scenarios 2 and 3 were observed.

Based on the outcomes of the comparative study, it can be argued that the controlled traffic spreading that emerged from the flocking behavior of packets allowed packets to exploit available resources on nodes involved in multiple paths to the sink, resulting in higher performance. The NCC and CAWR protocols did not allow for packet spreading among all available paths, resulting in over-utilization of some (popular) paths. Under the NCC protocol, packets were solely sent over the shortest path(s), while the CAWR protocol allowed for packet spreading after the appearance of buffer overflows. This behavior led to a high number of packet losses due to buffer overflows. Further results showed that the Flock-CC approach exhibited extremely low buffer overflows compared to both protocols due to the traffic spreading ability of the bird flocking behavior.

In addition, Figs. 20 (d)-(f) show that the Flock-CC approach exhibited the lowest EED among the other protocols for every transmission rate and scenario, because traffic spreading prevented augmented buffer occupancies that contribute to larger queuing delays. As far as

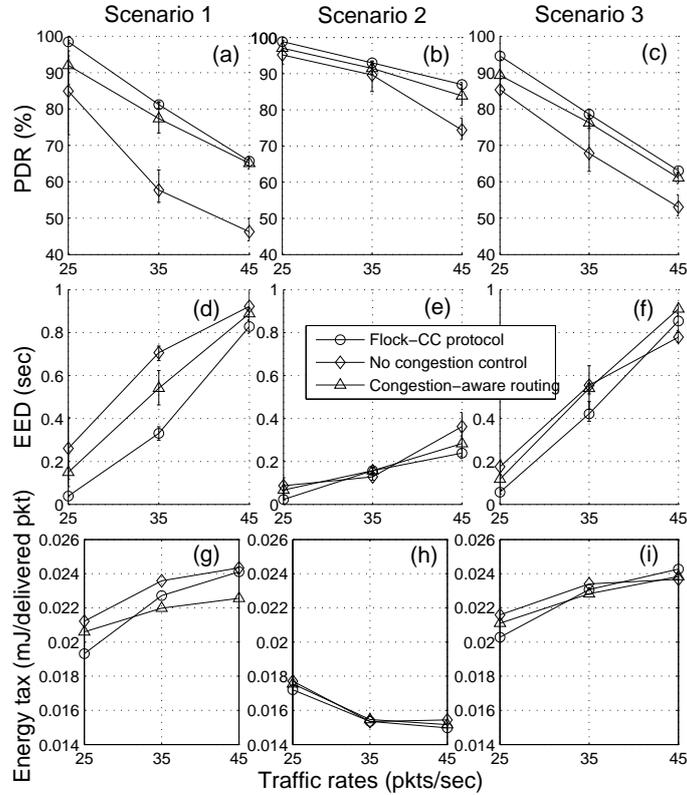


Figure 20: Comparative experiments in scenarios 1 – 3 for  $T = 0.5s$ . The performance of AODV was considerably poorer, thus related results were omitted from this figure.

energy consumption is concerned, Figs. 20 (g)-(i) show that the Flock-CC protocol spent less energy per delivered packet in low loads compared to NCC and CAwR.

Flock-CC was also qualitatively compared against AntHocNet and AntSensNet. Quantitative comparative scenarios are left for future work. AntHocNet showed better performance compared with AODV in terms of data delivery ratio and end-to-end delay [33]. Also, simulation results show that the performance of AntSensNet outperforms AODV in terms of delivery ratio, end-to-end delay and routing overhead [38].

Both AntHocNet and AntSenseNet are quite complicated protocols involving a large number of parameters and equations. These parameters have to be tuned for a variety of network and traffic conditions since they can be sensitive to the environment. On the other hand, the Flock-CC approach is quite simple involving only one parameter and one equation (desirability function).

In addition, the overhead generated by the ants (i.e. control packets injected into the network) is quite high for both AntHocNet and AntSensNet. In both approaches, each sensor node broadcasts periodically a HELLO packet (control packet). Also a number of control packets (that act as forward ants) are sent towards the sink and back for route probes. In Flock-CC protocol, the overhead is lighter since each node broadcasts periodically a control packet only.

In ant-based approaches, the problem is worsened since reactive forward ants (packets used for exploration) store the full array of nodes that they have visited on their way to the destination. In this way, a large amount of information is progressively added to these packets, thus increasing the probability of collision at considerably high levels. On the other hand, in Flock-CC, the size of a control packet remains always small and constant.

The high packet overhead causes scalability problems in AntHocNet and the performance of the protocol degrades as the number of nodes in the MANET increases. AntSensNet is considered to be more scalable than AntHocNet due to the clustered underlying topology. However, clustering formation assumes special roles in the network (e.g. clusterheads), while additional mechanisms are needed for maintaining and re-assigning roles. Also, areas around clusterheads may progressively become collision hot spots and deteriorate congestion. The Flock-CC protocol was shown to be scalable with the increase of sensor nodes without involving any added complexity like clustering.

Furthermore, both AntHocNet and AntSensNet necessitate large memory space to store all information used by the protocol. AntSensNet involves four pheromones for each traffic class. Every node  $x$  has to keep in its pheromone table the four pheromone values for each traffic class that passes through each link connecting node  $x$  and its neighboring nodes. Similarly, in AntHocNet, each node  $i$  maintains one pheromone table  $T_i$ . Pheromone tables contain for every destination a vector with one entry per outgoing link. The entry  $T_{ij}^d$  of node  $i$ 's pheromone table  $T_i$  contains the pheromone value  $\xi_{ij}^d$ , which is a floating point number indicating the relative goodness of taking outgoing link  $j$  on the way to destination  $d$ . Apart from a pheromone table, each node also maintains a neighbor table, in which it keeps track of which nodes it has a wireless link to. Only highly capable sensor nodes with extended memory capacities are able to store this mass of information. On the other hand, in Flock-CC, each node employs four small unidimensional tables. The size of each table is equal to the number of nodes in the FoV.

In addition, AntSensNet requires modifications in the queuing policies of the underlying MAC protocol in order to accept multi-class and multi-priority traffic, whereas Flock-CC cooperates well with a large number of MAC protocols without necessitating any modifications.

## 5. Conclusion and future work

This study focused on the bird flocking behavior to coherently 'move' packets to the sink, whilst combating congestion in wireless sensor networks (WSNs). The synchronized group behavior of birds flocks is mimicked in order to control the motion of packet flocks through a network of constrained sensor nodes, whilst avoiding congestion regions and dead node zones in a robust way. It is worth stressing that despite the simple behavior at individual nodes, the proposed Flock-CC approach contributes to the emergent behavior of high packet delivery ratio, low end-to-end delay and minimal energy tax.

The bio-swarm approach of Couzin [8] was reformulated from a metrical space to a topological space in order to manipulate the direction of motion of packets in WSNs influenced by local interactions among neighboring packets. The Flock-CC protocol is based on four flocking characteristics. Each packet perceived repulsive and attractive forces exhibited by other packets located on neighboring nodes within the FoV. The magnetic field oriented and guided packets towards the sink. Perturbation, which allows for exploration, was also introduced to allow packets to pick random routes, and thus to avoid over-flooding popular (due to their low congestion levels) next hop nodes.

It is worth noting that the Flock-CC model involves only one easily understood tuning parameter, the spreading variable,  $\xi$ , which influences the cohesiveness of the flock, and  $T$ , the sampling time. The spreading variable  $\xi$  was investigated across a large number of scenarios involving grid and random topologies, failure-free and failure-oriented environments, low and high MAC transmission rates. A good compromise value for  $\xi$  was 0.75. The value of parameter  $T$  cannot be hard coded into the model due to its dependence on network conditions and node hardware characteristics. A self-adaptive method of optimally choosing  $T$  based on network conditions (e.g. high  $T$  in failure-free environments, low  $T$  in scenarios with failing nodes) can be usefully employed, but it is left for future work.

The behavioral tendencies involved in the proposed approach were implemented in a simulated environment to study its effectiveness in mimicking the collective behavior of bird flocks. Performance evaluations showed that the proposed mechanism was able to robustly move packets to the sink, whilst alleviating congestion by balancing the offered load. In addition, results showed that the Flock-CC offers high PDR, fast delivery (small EEDs) of packets to the sink, and low energy tax. Furthermore, the Flock-CC approach outperformed typical conventional congestion-aware and congestion-free routing approaches in terms of PDR, EED and energy tax in low, medium and high loads.

Future work will investigate different functions for randomizing the attractiveness to the sink. Also, the performance in terms of power, as well as the inclusion of power in the desirability function remains a matter of further research. In addition, beyond the hop distance, new techniques for sink direction discovery are also left for future work. Also, an interesting idea is to investigate if and how the Flock-CC approach can operate in the presence of multiple sinks. Finally, beyond the simulative evaluation, a real-time implementation would further enhance the confidence in the proposed approach.

## Acknowledgment

This work is supported in part by the GINSENG: Performance Control in Wireless Sensor Networks project funded by the 7<sup>th</sup> Framework Programme under Grant No. ICT-224282 and the MiND2C: Mimicking Nature for Designing Robust Congestion Control Mechanisms in Self-Organized Autonomous Decentralized Networks project funded by the Research Promotion Foundation of Cyprus under Grant No. TPE/EPIKOI/0308(BE)/03.

## References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks (Amsterdam, Netherlands: 1999)*, vol. 38, no. 4, pp. 393–422, March.
- [2] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, 2005.
- [3] R. Jurdak, X. R. Wang, O. Obst, and P. Valencia, "Wireless sensor network anomalies: Diagnosis and detection strategies," in *Intelligence-based Systems Engineering*, A. Tolk and L. Jain, Eds. Springer, 2011.
- [4] T. E. Daniel, R. M. Newman, E. I. Gaura, and S. N. Mount, "Complex query processing in wireless sensor networks," in *Proceedings of the 2nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, ser. PM2HW2N '07, Chania, Crete Island, Greece, 2007, pp. 53–60.
- [5] M. Yu, H. Mokhtar, and M. Merabti, "A survey on fault management in wireless sensor networks," in *Proceedings of the 8th Annual PostGraduate Symp. on the Convergence of Telecommunications, Networking and Broadcasting*, Liverpool, UK, June 2007.
- [6] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. Oxford, 1999.

- [7] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1987, pp. 25–34.
- [8] I. D. Couzin, J. E. N. S. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective memory and spatial sorting in animal groups," *Journal of Theoretical Biology*, vol. 218, no. 1, pp. 1–11, September 2002.
- [9] P. Antoniou, A. Pitsillides, T. Blackwell, and A. Engelbrecht, "Employing the flocking behavior of birds for controlling congestion in autonomous decentralized networks," in *2009 IEEE Congress on Evolutionary Computation*, A. Tyrrell, Ed., IEEE Computational Intelligence Society. Trondheim, Norway: IEEE Press, 18–21 May 2009.
- [10] P. Antoniou, A. Pitsillides, A. P. Engelbrecht, T. Blackwell, and L. Michael, "Congestion control in wireless sensor networks based on the bird flocking behavior," in *4th IFIP TC 6 International Workshop on Self-Organizing Systems IWSOS*, ser. Lecture Notes in Computer Science, T. Spyropoulos and K. A. Hummel, Eds., vol. 5918. Springer, December 2009, pp. 220–225.
- [11] P. Antoniou, A. Pitsillides, A. P. Engelbrecht, and T. Blackwell, "Mimicking the bird flocking behavior for controlling congestion in sensor networks (invited paper)," in *3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies*. Rome, Italy: IEEE conference proceedings, November 2010.
- [12] C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, "CODA: congestion detection and avoidance in sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM Press, 2003, pp. 266–279.
- [13] B. Hull, K. Jamieson, and H. Balakrishnan, "Mitigating congestion in wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 134–147.
- [14] M. Vuran, V. Gungor, and O. Akan, "On the interdependence of congestion and contention in wireless sensor networks," in *ICST SenMetrics, San Diego, CA*, July 2005.
- [15] I. Demirkol, C. Ersoy, and F. Alagoz, "Mac protocols for wireless sensor networks: a survey," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 115–121, 2006.
- [16] S. Jardosh and P. Ranjan, "A survey: Topology control for wireless sensor networks," in *Signal Processing, Communications and Networking, 2008. ICSCN '08. International Conference on*, Jan. 2008, pp. 422–427.
- [17] J. N. Al-karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *IEEE Wireless Communications*, vol. 11, pp. 6–28, 2004.
- [18] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 7, pp. 537–568, May 2009.
- [19] C. Sreenan, J. S. Silva, L. Wolf, R. Eiras, T. Voigt, U. Roedig, V. Vassiliou, and G. Hackenbroich, "Performance control in wireless sensor networks: the ginseng project - [Global communications news letter]," *Communications Magazine*, vol. 47, no. 8, pp. 1–4, Aug 2009.
- [20] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware fair rate control in wireless sensor networks," in *Proceedings of the ACM SIGCOMM 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Pisa, Italy, September 11-15, 2006*, L. Rizzo, T. E. Anderson, and N. McKeown, Eds. ACM, 2006, pp. 63–74.
- [21] L. Popa, C. Raiciu, I. Stoica, and D. S. Rosenblum, "Reducing congestion effects in wireless networks by multipath routing," in *ICNP*. IEEE Computer Society, 2006, pp. 96–105.
- [22] R. Kumar, H. Rowaihy, G. Cao, F. Anjum, A. Yener, and T. L. Porta, "Congestion aware routing in sensor networks," Department of Computer Science and Engineering, Pennsylvania State University, Technical Report 0036, 2006.
- [23] T. He, F. Ren, C. Lin, and S. Das, "Alleviating congestion using traffic-aware dynamic routing in wireless sensor networks," in *SECON*. IEEE, 2008, pp. 233–241.
- [24] K. Karenos, V. Kalogeraki, and S. V. Krishnamurthy, "Cluster-based congestion control for supporting multiple classes of traffic in sensor networks," in *EmNets '05: Proceedings of the 2nd IEEE workshop on Embedded Networked Sensors*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 107–114.
- [25] C.-Y. Wan, S. B. Eisenman, A. T. Campbell, and J. Crowcroft, "Siphon: overload traffic management using multi-radio virtual sinks in sensor networks," in *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2005, pp. 116–129.
- [26] C. Sergiou, P. Antoniou, and V. Vassiliou, "Congestion control protocols in wireless sensor networks: A survey," *Submitted to the IEEE Communications Surveys and Tutorials*.
- [27] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*. NJ: John Wiley & Sons, Jan. 2006.
- [28] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Inspiration for optimization from social insect behaviour," *Nature*, vol. 406, no. 2000, pp. 39–42, 2000.
- [29] D. Constantinou, "Ant colony optimisation algorithms for solving multi-objective power-aware metrics for mobile ad hoc networks," Ph.D. dissertation, University of Pretoria, 2011. [Online]. Available: <http://upetd.up.ac.za/thesis/available/etd-07012011-151336/>

- [30] M. Dorigo and C. Blum, "Ant colony optimization theory: a survey," *Theor. Comput. Sci.*, vol. 344, no. 2-3, pp. 243–278, 2005.
- [31] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, pp. 1942–1948, November 1995.
- [32] R. Singh, D. K. Singh, and L. Kumar, "Swarm intelligence based approach for routing in mobile ad hoc networks," *International Journal of Science and Technology Education Research*, vol. 1, no. 7, pp. 147–153, 2010.
- [33] G. D. Caro, F. Ducatelle, and L. M. Gambardella, "Anthocnet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, 2005.
- [34] S. Rajagopalan and C.-C. Shen, "Ansi: a swarm intelligence-based unicast routing protocol for hybrid ad hoc networks," *J. Syst. Archit.*, vol. 52, pp. 485–504, August 2006.
- [35] Z. Xiangquan, G. Lijia, G. Wei, and L. Renting, "A cross-layer design and ant-colony optimization based load-balancing routing protocol for ad-hoc networks," *Frontiers of Electrical and Electronic Engineering in China*, vol. 2, no. 2, pp. 219–229, April 2007.
- [36] G. D. Caro, F. Ducatelle, and L. M. Gambardella, "Swarm intelligence for routing in mobile ad hoc networks," *Swarm Intelligence Symposium*, pp. 76–83, June 2005.
- [37] C. E. Perkins, "Ad hoc on demand distance vector (aodv) routing," *RFC 3561*, 1997.
- [38] L. Cobo, A. Quintero, and S. Pierre, "Ant-based routing for wireless multimedia sensor networks using multiple qos metrics," *Computer Networks*, vol. 54, pp. 2991–3010, December 2010.
- [39] W. Wiltschko, "Über den einflu statischer magnetfelder auf die zugorientierung der rotkehlchen," *Z. Tierpsychol.*, vol. 25, pp. 537–558, 1968.
- [40] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proceedings of the 9th ACM International Conference on Mobile Computing and Networking (MobiCom '03)*, San Diego, California, September 2003.
- [41] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*, ser. *MobiCom '04*. New York, NY, USA: ACM, 2004, pp. 114–128.
- [42] P. Antoniou, A. Pitsillides, T. Blackwell, A. Engelbrecht, and L. Michael, "From bird flocks to wireless sensor networks: A congestion control approach," Department of Computer Science, University of Cyprus, Tech. Rep. TR-11-5, September 2011. [Online]. Available: [http://www.netrl.cs.ucy.ac.cy/index.php?option=com\\_jombib&task=showbib&catid=60&id=655](http://www.netrl.cs.ucy.ac.cy/index.php?option=com_jombib&task=showbib&catid=60&id=655)
- [43] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1998.
- [44] The Network Simulator NS-2, <http://www.isi.edu/nsnam/ns>. <http://www.isi.edu/nsnam/ns/>.