Employing the Flocking Behavior of Birds for Controlling Congestion in Autonomous Decentralized Networks

Pavlos Antoniou, *Student Member, IEEE*, Andreas Pitsillides, *Senior Member, IEEE*, Tim Blackwell, *Member, IEEE*, and Andries Engelbrecht, *Senior Member, IEEE*

***Abstract*— Recently a great emphasis has been given on autonomous decentralized networks (ADNs) wherein constituent nodes carry out specific tasks collectively. Their dynamic and constrained nature along with the emerging need for offering quality of service (QoS) assurances drive the necessity for effective network control mechanisms. This study focuses on designing a robust and self-adaptable congestion control mechanism which aims to be simple to implement at the individual node, and involve minimal information exchange, while maximizing network lifetime and providing QoS assurances. Our approach combats congestion by mimicking the collective behavior of bird flocks having global self-* properties achieved collectively without explicitly programming them into individual nodes. 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). Unlike the bio-swarm approach of Couzin, which is formulated on a metrical space, our approach is reformulated on to a topological space (graph of nodes), while repulsion/attraction forces manipulate the direction of motion of packets. Our approach provides sink direction discovery, congestion detection and traffic management in ADNs with emphasis on Wireless Sensor Networks (WSNs). Performance evaluations show the effectiveness of our self-adaptable mechanism in balancing the offered load and in providing graceful performance degradation under high load scenarios compared to typical conventional approaches.**

I. Introduction

Over the last few years, research work in the area of autonomous networked systems has gained momentum. Autonomous decentralized networks (ADNs), such as ad-hoc and sensor networks [1], can be used as platforms for health monitoring, battlefield surveillance, environmental observation, etc. However, these new environments are extremely *dynamic, unreliable* and often *large scale*. Typically ADNs comprise of small (and sometimes cheap), cooperative devices (nodes) which may be constrained by computation capability, memory space, communication bandwidth and energy supply. These nodes can be either deployed all at once, or one by one. Autonomous nodes may interact (a) with each other in order to exchange information or forward data towards one or more sink nodes, and (b) with the environment so as to sense or control physical parameters. As shown in Fig. 1, all nodes that at a particular moment sense an event (grey-shaded nodes), are activated, and send their packets, in a multi-hop manner, to a dedicated sink node. The mass of network interactions, in conjunction with variable wireless network conditions, may result in unpredictable behavior in terms of traffic load variations and link capacity fluctuations. The problem is worsened if topology modifications driven by node failures, or mobility are considered. These stressful situations are likely to occur in ADN environments due to their complex nature and the possible scenarios of operation, thus increasing their susceptibility to congestion. Congestion causes energy waste, throughput reduction, increase in collisions and retransmissions at the medium access control (MAC) layer, increase of queueing delays and even information loss leading to the deterioration of the offered QoS, decrease of network lifetime and even the decomposition of network topology in multiple components.

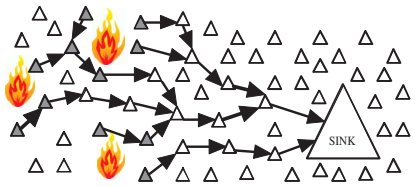

Fig. 1. An autonomous decentralized network for fire monitoring.

ADNs are expected to be deployed for several mission-critical tasks and operate unattended for extended durations. The unpredictable nature of operation drives the necessity for **robust, self-adaptable** and **scalable** mechanisms which are vital to the mission of dependable ADNs. Novel approaches should be **simple** to implement at individual node level with **minimal exchange of information**. This study focuses on a new congestion control (CC) mechanism that is expected to exhibit all the aforementioned characteristics, which is designed on the basis of the flocking behavior of birds. In particular, packets are modeled as birds flying over a topological space, e.g. a sensor network, and the main idea is to 'force' or 'guide' packets to form flocks and flow towards a global attractor (sink), whilst trying to avoid obstacles (congestion regions). Motivated by the limited visual perception of birds, each packet moves within a finite field of perception. The direction of motion is influenced by (a) repulsion and attraction forces exercised by neighboring packets, as well as (b) the gravitational force in the direction of the sink.

P. Antoniou is with the Department of Computer Science, University of Cyprus, 75 Kallipoleos Street, Nicosia, Cyprus (phone: +357-22-892687; fax: +357-22-892701; email: paul.antoniou@cs.ucy.ac.cy).
A. Pitsillides is with the Department of Computer Science, University of Cyprus, 75 Kallipoleos Street, Nicosia, Cyprus (phone: +357-22-892695; fax: +357-22-892701; email: andreas.pitsillides@cs.ucy.ac.cy).
T. Blackwell is with the Department of Computing, Goldsmiths College, University of London, New Cross, London, SE14 6NW, UK (phone: +44-(0)20-7919-7859; email: tim.blackwell@gold.ac.uk).
A. Engelbrecht is with the Department of Computer Science, University of Pretoria, South Africa (phone: +27-12-420-3578; email: engel@cs.up.ac.za).

Our approach provides **sink direction discovery**, **congestion detection** and **traffic management** for fast overload relief in ADNs with emphasis on Wireless Sensor Networks (WSNs). The main challenges are to achieve (a) low number of collisions and retransmissions, (b) low packet loss resulting in high reliability and low energy tax, (c) low latency, (d) acceptable throughput, and (e) fault tolerance.

This study draws inspiration from nature which has been very successful in 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 [2], [3]. SI techniques, motivated by the collective behavior of social insect societies and animal groups living in decentralized, self-organizing and adapting environments, reportedly provide an excellent basis for computing environments that need to exhibit these characteristics. 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 [4]. 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 [4]. Social groups found in nature (e.g. bird flocks) carry out their tasks collectively in order to contribute to a common goal. Even though individuals coordinate to accomplish a given global mission in a complex world (e.g. foraging, migration, defence against predators), each one of them has only local perception of the surrounding environment and exhibits specific behavioral tendencies which are governed by a few simple rules. ADNs in many ways, have common characteristics to social groups (e.g. nodes can be likened to constituents of these social groups) and they are in great need to accomplish their tasks collectively (by simple neighbor-to-neighbor interactions), in a decentralized manner, in the absence of (external) central supervision. These bio-systems usually exhibit robustness to internal perturbations (e.g. topological changes) or loss of units, and remarkable scaling properties. Adaptation is one of their major strengths as they must respond to addition or removal of members, and sudden changes in the environment. Drawing inspiration from the collective behavior of social groups, the local behavior can be better dictated and an emergent global behavior of minimum congestion and direction of information flow to the sink can be provoked. In this way, self-* properties e.g. self-organization, self-adaptation, will not be implemented explicitly into individual devices or nodes, but may arise as a result of evolutionary design.

The remainder of this paper is organized as follows. Section II deals with previous work. Section III presents the concept of our approach and the proposed model. Section IV presents performance evaluation results based on simulation studies. Section V draws the conclusion and future work.

II. Related Work

Congestion occurs when the traffic load injected into the network exceeds available capacity at any point of the network. In traditional wired networks, buffer drops are taken as indication of congestion while CC is usually carried out in an end-to-end manner (i.e. only the source-destination pair is involved). On the other hand, simulation studies conducted by [7] and [8] revealed that in WSNs where the wireless medium is shared using Carrier Sense Multiple Access (CSMA)-like protocols, wireless channel losses (link-level congestion symptom) dominate buffer drops (node-level congestion symptom) and increase quickly with offered load. The problem of channel losses is worsened around hot spot areas, as for example, in the area of an event, or around the sink. In the former case, wireless medium 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 reaches its upper limit faster than queue occupancy [20]. Thus, queue occupancy alone cannot accurately serve as an indication of congestion. 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 high error rates, especially during long periods of congestion.

Various CC schemes can be found in WSN literature based on traffic manipulation (e.g. rate control [7], [8], multi-path routing [9], [10]), topology control (e.g. clustering formation [11]), and network resource management (e.g. power control, multiple radio interfaces [12]). The majority of CC approaches are based on rate control, usually using some form of back pressure, that alleviates congestion by throttling the injection of traffic in the network. However, rate control attempts to decrease nodes' reporting rate during congestion and may result in the deterioration of the offered network QoS when applications' requirements are not met. In addition, rate control can not be efficiently used during transient congestion phenomena caused by aperiodic and short term packet bursts (e.g. event monitoring applications). Furthermore, 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. Congestion mitigation based on power control and multiple radio interfaces seems unrealistic in WSNs since the low-cost nodes incentive is violated. Multi-path routing can effectively alleviate congestion without deteriorating the offered network QoS. However, recent studies on multi-path routing [10] require knowledge of nodes' locations which is practically difficult to obtain (high implementation cost). Due to the many-to-one nature and the sheer size of some WSNs, routing techniques for ad-hoc networks cannot be effectively applied. On-demand routing protocols [13] involve high path-finding latency and result in network clogging due to excessive packet flooding. Table-driven protocols [14] consume a significant amount of bandwidth for path maintenance and display slow reaction on restructuring and failures.

Conventional optimization techniques for CC may 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.

From the viewpoint of Swarm Intelligence (SI) [2], [3], a considerable number of models based on self-propelled particles have been developed to solve a variety of problems, some in ad-hoc networks, by means of collective motion. Currently, two of the most successful SI classes of algorithms are Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) [3]. The research framework behind the majority of network-oriented studies involving self-propelled particles was fueled by the ACO theory proposed by Dorigo et. al [15]. Driven by the collective behavior of ants in finding paths from the colony to food, many researchers [16], [17] tried to solve the problem of routing. In ACO, artificial ants build solutions by moving on the problem graph and, by mimicking real ants, deposit artificial pheromone on the graph in such a way that future artificial ants can build better solutions. The optimal solution can be derived from previously (over suboptimal paths) transmitted messages. The work of Di Caro [16] incorporates congestion awareness in an end-to-end manner (AntHocNet), whereas [17] tries to optimize energy, congestion and load balancing parameters along with providing efficient routing in ad-hoc networks. PSO is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space. This approach can be used to tackle specific network-oriented problems like deployment and energy efficiency. Shih [18] implemented a PSO-based clustering algorithm to reduce the traffic load using data aggregation or compression and improve the system's lifetime. Results show that nature-inspired techniques outperform conventional routing algorithms. According to [19], AntHocNet outperforms AODV in terms of delivery ratio, end-to-end delay and jitter. An important observation is that the advantage of AntHocNet over AODV grew for larger networks, especially in terms of overhead, suggesting that AntHocNet is more scalable than AODV. The problem of congestion in ADNs is partly addressed using SI techniques, but solely in the context of ad-hoc networks. It has to be noted that ad-hoc and sensor networks have some fundamental differences (see [1] for a detailed explanation) something which drives the necessity for novel CC strategies in the context of sensor networks, on which this study focuses.

III. Flock-based Congestion Control Approach

WSNs necessitate decentralized CC strategies which promise fast, effective and efficient relief from congestion. This section presents a decentralized approach where CC is carried out in a hop-by-hop manner, having all nodes along a network path involved in the procedure.

A. The Concept

The main idea behind our concept is to *'force' or 'guide' packets to form groups or flocks and flow towards a global attractor (sink), whilst trying to avoid obstacles (congestion regions)* as shown in Fig. 2.

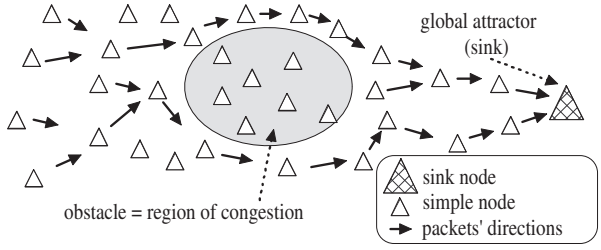

Fig. 2. Packet flock moving towards sink whilst avoiding 'obstacles.

The design of the spatial dynamics in packet groups is influenced by the study of artificial flocks like the 'boid' animations of Reynolds et al. [5] and the bio-swarm model of Couzin et al. [6]. These studies involve reference to artificial flocks consisting of individuals with finite range of perception which interact with each other. The behavior of each individual is influenced by other neighbors located in its vicinity (neighborhood). Typically, individuals **repel each other at close range to avoid collisions, attract each other at intermediate range to maintain coherence among the members of a flock**, whereas no forces are exhibited between them at long range. These behavioral rules govern individual-level interactions which collectively result in the emergence of group-level transitions.

In this study, an ADN can be viewed as a virtual ecosystem, where multiple packets are generated and must be directed to a dedicated sink (Fig. 1). The bio-swarm approach of Couzin [6] is reformulated from the metrical space on to a topological space (graph of nodes) in order to manipulate the direction of motion of packets in ADNs. Each packet is modeled as 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 sequence of transitions determines a packet's trajectory from its source to the sink. On each hosting node, a packet has to choose its next hop hosting node from the set of neighboring nodes. However, each packet can only 'see' the fraction of its neighboring nodes located within its field of perception. In order to avoid collisions, or to prevent the emergence of heavily congested areas, packets should be attracted to low packet concentration areas, and be repelled from high packet concentration areas within the field of perception. In this context, the levels of congestion on each neighboring node (node-level congestion, e.g. queue loading) and around it (link-level congestion, e.g. wireless channel load) should be taken into account. However, packets following such directives may not be directed to the sink, if, for example, they are coupled in loop transitions between uncongested nodes.

These undesirable effects can be avoided by employing a measure of attractiveness to the global target (sink). This is realized on the basis of nodes' proximity to the sink, i.e. a neighboring node located closer to the sink is expected to exhibit stronger attractiveness.

In order to model the relationships among individual particles, a spatial neighborhood is defined around each individual that is divided into two zones, namely the zone of repulsion (separation) and the zone of attraction (cohesion). A group of moving packets can behave like a flock, when each packet *attracts to the sink under the influence of a global force field, (within zone of attraction) attracts to all neighboring packets (cohesion), and (within zone of repulsion) repels from neighboring packets (separation)*. The zone of repulsion can be interpreted as individuals maintaining personal space, or avoiding collisions. Attraction represents the tendency of organisms to join groups and to avoid being on the periphery, thus providing coherence and maintaining a shared neighborhood (which may be a sub-flock or the entire flock).

B. The Model

In this section, the model of our FCC approach is described. All quantities are considered at the discrete instant kT, with T the sampling period. Consider a network of N autonomous nodes that are able to generate data packets and forward them to the sink. Each node n, $n \in [1..N]$, is associated with a queue of size Q packets, and a wireless channel of capacity L bits per second. Each node n in WSNs, is able to measure periodically some parameters that may serve as congestion indicators. For example, R_n^{in} is the incoming packet rate, R_n^{out} is the successful outgoing packet rate, and q_n is the instantaneous queue size. As discussed in Section II, queue occupancy alone cannot accurately serve as an indication of congestion. Thus, a more effective congestion indicator (e.g. a combination of the aforementioned parameters) may be needed to capture short-term congestion conditions. In a harsh WSN environment, packets may be dropped due to queue overflows, link layer failures (e.g. collisions), and routing failures (e.g. when a next hop node cannot be reached) on each node. In this study, we consider that congestion symptoms such as packet drops (e.g. due to buffer overflows, collisions, unreachable nodes) and high queue loading (e.g. when incoming>outgoing traffic) may appear on node n as a loading indicator, given by the ratio:

$$p_n = \frac{R_n^{in} + q_n/T - R_n^{out}}{R_n^{in} + q_n/T} \tag{1}$$

It has to be noted that possible packet retransmissions are not taken into account in the evaluation of Eq. 1. All parameters involved in this equation are measured at the end of every sampling period T, except for q_n which is obtained at the beginning, in order to be able to determine the total number of packets traversing node n within the sampling period. The parameter p_n reflects the ratio of packets per T that either remained in the queue (queue loading) or dropped, over all packets traversing node n between the $k-1$-th and the k-th period. When $p_n \to 0$, the rate of packet drops at node n is close to zero and the queue is empty or nearly empty. On the other hand, as $p_n \to 1$, node n is considered congested due to a high number of packet drops or high queue occupancy. p_n can serve as an early indication of incoming load on node n, providing fast response to congestion phenomena. Each node n evaluates p_n in a periodic manner (every T seconds) and broadcasts it using a small control packet to all neighboring nodes. This packet includes information related to the quality of the wireless channel in the vicinity of the node. Each node estimates the quality of the shared wireless channel (useful link service rate, r_n), using information taken from the medium access control (MAC) protocol. We consider Carrier Sense Multiple Access (CSMA)-like protocols (e.g. IEEE 802.11, IEEE 802.15.4) 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. The total number of all packet transmission attempts at node n within a period T is denoted by R_n^{out*}, where $R_n^{out*} = R_n^{out}$+retransmits within that period. The useful link service rate at node n is denoted by:

$$r_n = \frac{R_n^{out}}{R_n^{out*}} = \frac{R_n^{out}}{R_n^{out} + retransmits} \tag{2}$$

When $r_n \to 1$, the channel is not congested and a large percentage of packets are successfully transmitted (few packet retransmissions are observed). As $r_n \to 0$, the channel is congested and a small number of packets are transmitted, often after a large number of retransmissions.

From WSNs perspective, an effective CC scheme should prevent the occurrence of packet drops and high queue loading, as well as the emergence of collision hot spots. Our approach focuses on both directions on the basis of repulsion and attraction forces. In particular, repulsion and attraction forces are synthesized in the decision making process of a packet, when a new hosting node is about to be chosen. The current position of each packet i is determined by the position of the hosting node n and its direction is evaluated before leaving its current hosting node with respect to (a) repulsion/attraction forces exhibited by neighboring packets, and (b) the gravitational force in the direction of the sink.

The zones of attraction and repulsion are defined within the neighborhood of each packet (or alternatively its hosting node). However, motivated by the limited visual perception of birds, each hosting node can only 'see' a small fraction of its neighboring nodes. The visible area is defined by the field of perception of each node. For example, the conical area defined by the two zones of Fig. 3 (the repulsion and the attraction zones which are explained later), represents the field of perception of the black-shaded node. In order to achieve attractiveness to the global target, the field of perception should cover the area in the direction of the sink node. In this case, packet flocks that are guided to move only within the field of perception of their hosting nodes will reach the sink within a finite number of hops. In our model, the direction of the sink can be deduced by the hop-distance

variable, $h_n(kT)$, indicating the number of hops between node n and the sink at the k-th period. The number of hops to the sink for each node may change due to network topology modifications caused by node failures and an appropriate update scheme should be implemented (this is beyond the scope of this paper). Nodes located closer to the sink are expected to have smaller hop-distance values (see Fig. 3) and should be chosen with higher probability as next hop hosting nodes. The value of this parameter is also included in the periodic control packet.

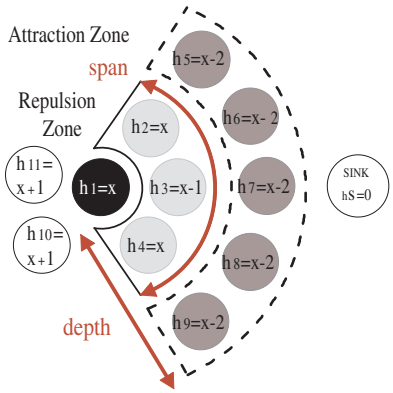

Fig. 3. Representation of spatial neighborhood around each individual: The field of perception involving the zones of repulsion and attraction. Note hop distance of current node h1=x.

After sink direction discovery, the field of perception is defined in terms of a conical area, which will be further decomposed into the attraction and the repulsion zones (see next paragraph). The conical area representing the field of perception depends on two parameters: the depth and the span of the area, as shown in Fig. 3. The depth of the conical area indicates how many hops away from a hosting node n a node should be in order to be considered inside the cone. It is worth pointing out that all nodes around a hosting node n can be categorized into three sets: (a) nodes located closer to the sink than node n, (b) nodes located further from the sink than node n, and (c) nodes having equal hop-distance with node n. The span of the area determines which of these set(s) will be included within the perception field. For example, in Fig. 3, the perception field of the black-shaded node involves nodes located closer to the sink as well as those having equal hop-distance (see the hop-distances on each node), but are located at most 2 hops away from the hosting node. This kind of perception field is used in the rest of this study.

The field of perception is decomposed into the attraction and the repulsion zones. Our FCC approach combats congestion by using **attraction to nodes experiencing low packet drop ratio (e.g. due to buffer overflows, collisions, unreachable nodes) and low queue loading per period T, while repulsion is employed to prevent the emergence of collision hot spots by choosing nodes with higher useful link service rate**. Within a bird flock, individuals repel each other at close range to avoid collisions, attract each other at intermediate range to maintain coherence among the members of a flock, whereas no forces are exhibited between them at long range. In accordance with this behavior of bird flocks, Fig. 3 illustrates that in our model, the zone of attraction comprises packets located on neighboring (light gray-shaded) nodes one hop away from the hosting node, while the zone of attraction consists of packets located on (heavy gray-shaded) nodes two hops away. Each packet i, that is ready to choose its next hop hosting node, can only move to one of the nodes one hop away from its current hosting node, since the rest of the network nodes are not directly reachable. On the other hand, each packet can interact with packets either located on nodes one hop away, or just moved to nodes two hops away. In both cases, the intensity of the exercised (by neighboring packets) force will be carried out by the involved node one hop away. This statement will be made clear below. Consider that the black-shaded node of Fig. 3 generates some packets when triggered by an external event, which are directed to the sink 'flying' within the perception field of their hosting node. Our FCC approach can prevent the emergence of collision hot spots in the vicinity of (light gray-shaded) nodes one hop away, when each packet i (generated at the black-shaded node) is repelled from the packets located on nodes one hop away, that either are dropped after a transmission, or a retransmission, or remain in the queue of their hosting nodes. The intensity of the repulsion force exercised by each node one hop away will be proportional to the number of these packets per T, e.g. proportional to $1 - r_k$ for each node k one hop away. In addition, low percentage of packet drops and queue loading on each node one hop away will be achieved when each packet i is attracted to packets located on nodes one hop away involving low packet concentration. Each node is expected to maintain low packet concentration when the incoming packet rate is smaller than the outgoing rate. Therefore, low packet concentration on each node one hop away can be achieved when each packet i is attracted to packets transmitted from nodes one hop away to nodes two hops away while subtracting the contribution of incoming packets. The intensity of the attraction force will be proportional to the difference between the outgoing and the incoming packet rates per T, e.g. proportional to $1 - p_k$ for each node k one hop away. The attraction force will be exercised by the involved node one hop away on behalf of packets that moved further. Even though the attraction force is exercised by a node one hop away, the zone of attraction comprises packets located on nodes two hops away.

According to the span of the perception field, even though a packet cannot move backwards (i.e. in the opposite direction of the sink), it may not be directed to the sink, if, for example, it is coupled in loop transitions between uncongested nodes (within the field of perception) having equal hop-distance to the sink with its previous hosting node. This undesirable effect can be avoided by employing a measure of local attractiveness to the set of nodes located closer to the sink. This can be realized on the basis of response thresholds [21]. We consider that the field of

perception of the hosting node n consists of M nodes one hop away (potential next hops), where $M \subseteq N$. Each packet i located on node n, infers the congestion levels of each node m, $m \in [1..M]$, (prior to choosing its next hop node) on the basis of attraction/repulsion forces. In order for the packet to avoid loop transitions, preferential transition towards nodes closer to the sink should be experienced. This behavior is captured by response thresholds, a nature-inspired mechanism which refers to the likelihood of reacting to a stimulus associated with a given task. Each task can be characterized by a different threshold θ, which determines the tendency of an individual to respond to the task-associated stimulus. In our model, a response threshold refers to the likelihood of packet i to be engaged in task m, i.e. to choose node m as its new hosting node, on the basis of node-m-associated stimulus s_{im}. This probability is expressed by:

$$Rth_{im} = \frac{s_{im}^2}{s_{im}^2 + \theta_m^2}, \tag{3}$$

where θ_m is the node-m-associated threshold. The intensity of attraction to each node m (i.e. the probability of choosing node m) depends on its proximity to the sink in terms of hop-distance. Therefore, the tendency of individual packet i to respond to the node-m-associated stimulus will be determined on the basis of different threshold for each node m, θ_m. Nodes associated with lower thresholds will be chosen as new hosting nodes with higher probability. Thus, nodes within the field of perception located closer to the sink will be associated with lower thresholds than nodes located at equal hop-distance with the current hosting node. In our approach, each packet i picks a random stimulus from 0 to 1 for each node m and evaluates Eq. 3 using node-m-associated threshold ($0 \leq \theta_m \leq 1$). The randomly chosen stimulus help packets to pick a random route, while response thresholds preserve the preference to each node. In this study, the value of $\theta_m = \theta_0$ corresponds to the threshold associated with nodes closer to the sink, while $\theta_m = 1 - \theta_0$ is associated with equal hop-distance nodes in the field of perception. When $\theta_0 = 0$, nodes located closer to the sink are chosen with probability equal to 1. This narrow perception field forces packets to form very coherent flocks whilst moving towards the sink. When θ_0 increases to 0.1, the perception field spans across nodes with equal hop-distance. However, these nodes are chosen with small probability. As θ_0 grows, equal hop-distance nodes (associated with the the decreasing threshold $1 - \theta_0$) are chosen with higher probability and the packet flock becomes less coherent.

Traffic management is performed on per packet basis in a hop-by-hop manner. Whenever a packet is about to be sent, the decision making process is invoked to determine the next hop node. Each hosting node evaluates the next hop node for its packets based on an M-dimensional desirability vector, $\overrightarrow{D}_i$. The node exhibiting the strongest desirability is chosen to be the new hosting node. Each element, D_{im}, of the vector $\overrightarrow{D}_i$ represents the desirability of packet i for each node m. The desirability $D_{im}(t)$ of a packet i (located on node n) for every node m is evaluated at time t, $kT < t < (k+1)T$, when the packet is about to be sent. $D_{im}(t)$ depends on: (a) the current congestion conditions on each node m, $p_m(kT)$, with respect to incoming and outgoing packet rates between $kT - T$ and kT, (b) the percentage of successfully transmitted packets, $r_m(kT)$, between $kT - T$ and kT, and (c) the likelihood, Rth_{im}, of choosing node m as new hosting node. A non linear form of the desirability function $D_{im}(t)$ is given by:

$$D_{im}(t) = Rth_{im} * [\alpha * r_m(kT) + (1 - \alpha) * (1 - p_m(kT))], \tag{4}$$

where the parameter α regulates the influence of parameters $r_m(kT)$ and $p_m(kT)$. $D_{im}(t)$ measures the tendency of each packet i to move towards node m, which is proportional to (a) low values of p (attraction to nodes with low ratio of packets dropped and queue overloading per T), and to (b) high values of r (repulsion from nodes with highly erroneous wireless channel). The next hop node for packet i, is chosen to be the node m^* that corresponds to the maximum element in the vector $\overrightarrow{D}_{im}(t)$, i.e. $m^* = max\{\overrightarrow{D}_{im}(t)\}$.

IV. Performance Evaluation

Packet-level simulation was used to investigate the performance of our approach. The effectiveness of our model in combating congestion by mimicking the collective behavior of bird flocks was studied. Our model was implemented in the ns-2 network simulator [22] and extensive simulations were performed by applying our model in a WSN environment. The evaluation topology comprised 200 nodes which were deployed in a 2D lattice as shown in Fig. 4. Neighboring nodes are placed at the same distance ($50\sqrt{2}$m) from each other while their transmission range is 50m. The parameter

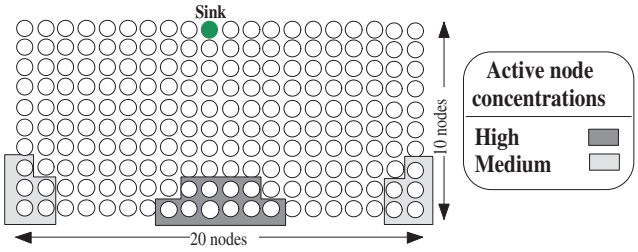

Fig. 4. Evaluation topology.

α and the response threshold θ_0 are the most important control knobs of our system and are expected to influence its forwarding behavior. α was chosen between 0 and 1, while θ_0 was selected to range from 0 to 0.4. The time between successive control packets T (sampling period) was set to 1 second. The selection of 1 second is guided by the desire to maintain responsiveness to changes in the network state and to avoid overwhelming the network with control packets. Queue size Q was set to 50. Active nodes were generating constant bit rate (CBR) traffic at the rates of $25, 35$ and 45pkts/sec. These cases can be considered as slightly congested, congested and heavily congested, respectively.

The IEEE 802.11 MAC protocol was used with transmission rate of 2Mbps, avoiding the use of RTS/CTS packets.

A comparative study was made among 2 different scenarios of concentrations of active nodes. As shown in Fig. 4, heavy grey shaded area involves high active node concentration, while light grey shaded area involves medium active node concentration. Also, system performance was compared to other typical conventional techniques. Two common measures of congestion were taken into account: the packet delivery ratio (PDR) and end-to-end delay (EED). PDR 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. EED is defined as the time taken for a packet to be transferred from a source node to the sink.

Each scenario was executed 10 times and the average values of metrics over all scenarios are presented below. Initially, simulation scenarios involving high active node concentration were considered, which correspond to the majority of real operational scenarios in densely deployed WSNs. In these networks, a considerably high number of nodes generate traffic when triggered by a special event. In the first scenario, active nodes generated traffic at the rate of 35pkts/sec (i.e. the network can be considered congested).

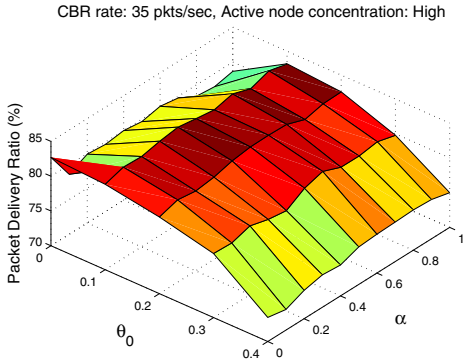

Fig. 5. Packet Delivery Ratio for CBR traffic of 35pkts/sec under high active node concentration.

Figs. 5 and 6 illustrate the impact of α and θ_0 on packet delivery ratio (PDR) and end-to-end delay (EED) respectively. As can be seen, the value of θ_0 influenced heavily the percentage of packets delivered at the sink. As discussed before, when $\theta_0 = 0$, the packet flock is more coherent. This was found to be inefficient in scenarios with high active node concentration because the perception field of packets was too narrow (only nodes located closer to the sink were involved in). Therefore, packets could not 'see' possible non-congested nodes in the vicinity of their hosting nodes, but were forced to move through a limited number of nodes resulting in starvation of resources on these nodes. However, the strategy of $\theta_0 = 0$, provided fast transitions (EED) of packets to the sink ($1-1.4$ secs), as shown in Fig. 6, because the packets trajectories employed the shortest paths to the sink. Under high packet flock coherence ($\theta_0 = 0$), the percentage of packets delivered to the sink (PDR) was

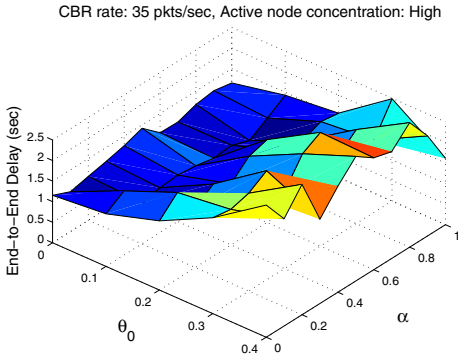

Fig. 6. End-to-End Delay for CBR traffic of 35pkts/sec under high active node concentration.

maximum (around 83%) for $\alpha = 0$ (i.e. only the ratio of packets dropped, or queue loading per T were taken into consideration), and minimum (around 78%) for $\alpha = 0.9$ (i.e. high influence of the wireless channel service rate on the desirability function). The role of α and its relationship with θ_0 is discussed below. The increase of θ_0 to 0.1, enhanced the field of perception of each hosting node and the packet flock became less coherent. Fig. 7, depicts the change in the packet flock coherence for $\theta_0 = 0$ and $\theta_0 = 0.1$, when $\alpha = 0.5$. The visual network representation reveals the number of packets arrived at each node. Clearly, for $\theta_0 = 0.1$ the flock spanned across larger fraction of nodes compared with $\theta_0 = 0$.

When $\theta_0 = 0.1$, each packet was allowed to explore larger fraction of nodes in the area around its hosting node, and the traffic was spread among them. This strategy achieved

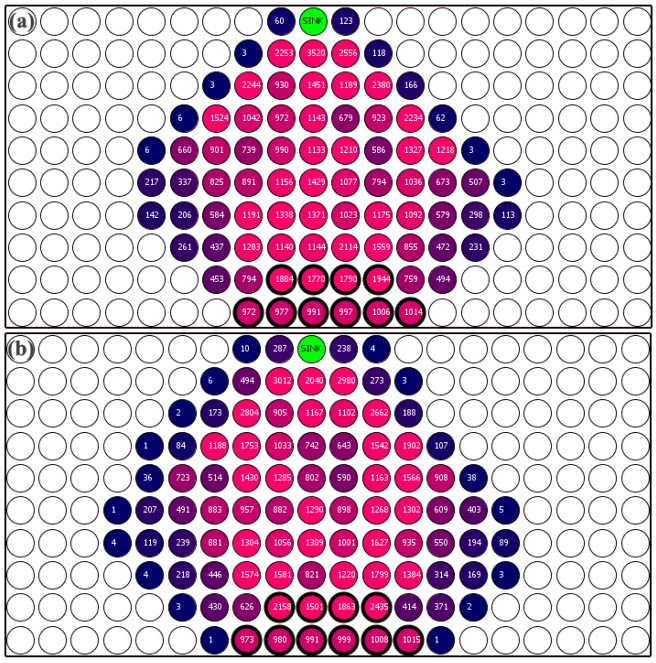

Fig. 7. Number of packets arrived at each node (a) $\theta_0 = 0$, (b) $\theta_0 = 0.1$.

the best results in terms of PDR (reaching 84%) and EED (around 1 second), regardless of the values of α, as shown in Figs. 5 and 6. In the case of $\theta_0 = 0.1$, PDR reached its peak (84%) for $\alpha = 0.4$. This combination of θ_0 and α values comprised the best case scenario also for rates 25 and 45 pkts/sec, as depicted in Fig. 8. The choice of $\theta_0 = 0.1$ was dictated by the desire to exploit a larger fraction of nodes towards the sink whilst avoiding loop transitions. As shown in Fig. 5, further increase of θ_0 above 0.1, amplified the tendency of packets to move towards nodes with equal hop-distance from sink. This behavior gave rise to loop transitions, thus increasing the possibility of collisions, not only within the sending area but in every part of network involved in packet forwarding. Also Fig. 5 shows that a decrease of 8% in PDR was perceived for $\theta_0 = 0.4$ compared with corresponding scenarios (of equal α) for $\theta_0 = 0.1$. In addition, when the packet flock was less coherent, increased EED values were observed due to the tendency of packet to follow longer paths to the sink (Fig. 6).

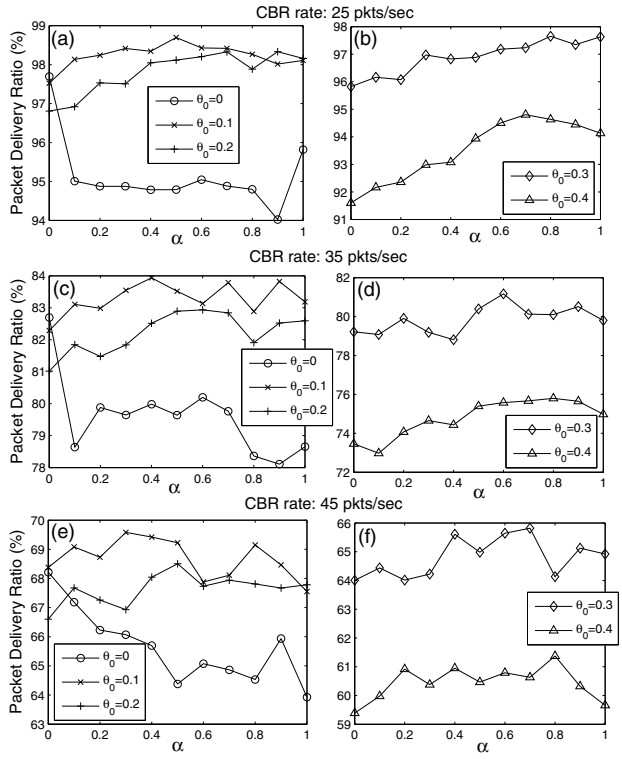

Fig. 8. Relationship between θ and α at high concentration.

According to Eq. 4, α regulates the influence of parameters r and p on the desirability for next hop nodes. When $\alpha = 0$, each packet can choose its new hosting node on the basis of parameter p (Eq. 1), which represents the packets per T that either were dropped, or remained in the queues of potential next hop nodes. Based on simulation studies, we found that the values of p were often close to 0 because the MAC protocol was managing to transmit packets (even after a large number of retransmissions, which were captured in parameter r). Thus, all potential next hop nodes were considered uncongested and the traffic was spread among them. When $\alpha = 0.1$, parameter r (i.e. the wireless link service rate) is engaged in the decision making process about next hop nodes (Eq. 4). Since r values were significantly greater than p, the parameter r was basically influencing next hop decisions. Through simulation studies, it was found that parameter r amplified the tendency of packets to move towards previously inactive nodes within the perception field (i.e. to move at the borders of the flock) in order to avoid collision hot spots occurring inside the network. As shown in Figs. 8(a), (c), and (e), this behavior was not effective whenever the perception field was narrow ($\theta_0 = 0$). More specifically, for $\theta_0 = 0$ and $\alpha \neq 0$, the number of delivered packets deteriorated in all transmission rates because the packets tended to exploit previously inactive nodes at the borders of the flock, while within the next few steps were forced (by the narrowness of the perception field) towards the center of the flock causing collisions with ongoing packets. With the increase of θ_0, $0.1 \leq \theta_0 \leq 0.4$, the packet flock can better exploit the high availability of nodes within perception field at given α values. For example, when $\theta_0 = 0.1$, PDR reached its peak value for $\alpha = 0.4$ (for all transmission rates).

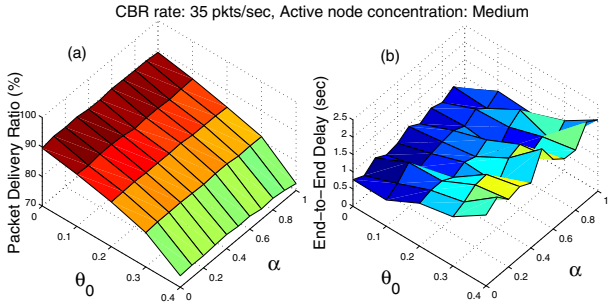

Fig. 9. Packet Delivery Ratio and End-to-End Delay for CBR traffic of 35pkts/sec under medium active node concentration.

Fig. 9 shows the influence of α and θ_0 on PDR and EED values, for medium active nodes concentration. It can be observed that PDR values increased gradually from 70% to 90% with the decrease of θ_0 from 0.4 (low flock coherence) down to 0 (high flock coherence). The event location was considered at 45^o angle to the sink compared with the high concentration scenario. Thus, the effective path in the direction of the sink was narrow, i.e. the line connecting the center of the area of the event and the sink. When $\theta_0 = 0$, all packets moved over this path forming coherent flocks. With the increase of θ_0, packets were allowed to move at the borders of the flock, but within the next few steps they were re-attracted to the center of the flock (moving in the direction of the sink) causing collisions with ongoing packets.

Our FCC approach (that incorporates both routing and congestion control capabilities) was compared with (a) a conventional congestion-free routing protocol based on shortest paths like [13], [14], and (b) a typical congestion-aware routing protocol that routes packets over multiple paths, choosing

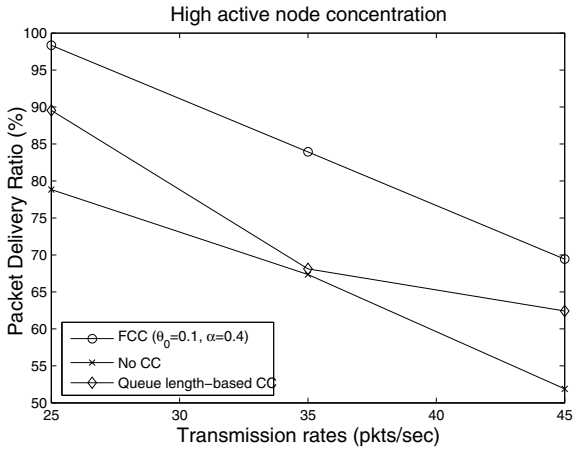

Fig. 10. Comparative experiments.

each time the least congested node in terms of queue length. The values of α and θ_0 were set to 0.4 and 0.1, which obtained the better performance in the simulation studies for the high active nodes concentration scenario. Results (Fig. 10) showed that our approach clearly outperformed all other approaches under high concentration, where congestion phenomena are more intensive, for all transmission rates. In particular, the FCC approach delivered 20% more packets than the congestion-free protocol and $7-20\%$ more packets compared with the congestion-aware routing protocol. In the case of medium active nodes concentration, our approach outperformed slightly over the other two approaches.

V. Conclusions and Future Work

In this study, we investigated how we can employ nature inspired models to combat congestion in ADNs. We mimic the synchronized group behavior of birds flocks and their ability to avoid obstacles so as to control the motion of packet flocks through a sensor network. Our aim is to guide packets towards the sink whilst avoiding congestion regions.

We reformulate the bio-swarm approach of Couzin from a metrical space to a topological one (graph of nodes) in order to manipulate the direction of motion of packets in WSNs influenced by local interactions among neighboring packets. In our model, each packet perceives repulsive and attractive forces exhibited by other packets located on neighboring nodes within the field of perception of its hosting node, and decides a new hosting node on the basis of a desirability function. We simulated the behavioral tendencies involved in our model to study its effectiveness in mimicking the collective behavior of bird flocks. Performance evaluations show that our mechanism is able to alleviate congestion by balancing the offered load, while offering acceptable PDR especially in high load scenarios and fast delivery (small EEDs) of packets to the sink. Also, it outperforms typical conventional congestion-aware and congestion-free routing approaches in terms of PDR in low, medium and high loads. For future work we plan to investigate the behavior of our mechanism in the presence of failing nodes as well as to study its influence on energy expenditure. Also, the influence of the design parameters (α, θ_0) will be investigated further for different scenarios of operation and topologies.

Acknowledgment

The authors would like to thank Dr. Loizos Michael of UCY for his valuable comments. This work is supported in part by the GINSENG: Performance Control in Wireless Sensor Networks project funded by the 7^{th} 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] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Computer Networks **38** (2002) 393-422.
[2] Bonabeau, at al.: Swarm Intelligence: From Natural to Artificial Systems. Proc. of Sciences of Complexity, Santa Fe Institute, (1999).
[3] Engelbrecht A. P.: Fundamentals of Computational Swarm Intelligence. Wiley (2005).
[4] Bonabeau, E., Dorigo, M., Theraulaz, G.: Inspiration for optimization from social insect behavior. Nature, **406** (2000) 39–42.
[5] Reynolds, C.: Flock, Herds, and Schools: A Distributed Behavioral Model. Proc. of ACM SIGGRAPH '87, **21** (1987) 25–34.
[6] Couzin, I. D., et al.: Collective Memory and Spatial Sorting in Animal Groups. Journal of Theoretical Biology, **218** (2002) 1–11.
[7] Wan, C., Eisenman, S., Campbell, A.: CODA: Congestion Detection and Avoidance in Sensor Networks, Proc. of the 1st Int. Conf. on Embedded Net. Sensor Systems, (2003) 266–279.
[8] Hull, B., Jamieson, K., Balakrishnan, H.: Mitigating Congestion in Wireless Sensor Networks, Proc. of the 2nd Int. Conf. on Embedded Net. Sensor Systems, (2004) 134–147.
[9] Popa, L., Raiciu, C., Stoica, I., Rosenblum, D.: Reducing Congestion Effects in Wireless Networks by Multipath Routing, Proc. of 14th IEEE Int. Conf. on Network Protocols, (2004) 96–105.
[10] Kumar, R., Rowaihy, H., Cao, G., Anjum, F., Yener, A., Porta, T.: Congestion Aware Routing in Sensor Networks, Tech. Report **36**, The Pennsylvania State University, (2006).
[11] Karenos, K., Kalogeraki, V., Krishnamurthy, S.V.: Cluster-based Congestion Control for Sensor Networks, ACM Trans. on Sensor Nets, **4**.
[12] Wan, C., Eisenman, S., Campbell, A., Crowcroft, J.: Siphon: Overload traffic management using multi-radio virtual sinks in sensor networks, Proc. ACM SenSys, (2005).
[13] Perkins, C., Royer, E., Das, S.: Ad hoc On-demand Distance Vector (AODV) Routing, RFC 3561.
[14] Perkins, C., Bhagwat, P.: Highly Dynamic Destination-Sequenced Distance Vector (DSDV) for Mobile Computers. Proc. of the SIGCOMM (1994), 234–244.
[15] Dorigo, M., et al.: Ant Colony Optimization Theory: A Survey. Proc. of Sciences of Complexity, Santa Fe Institute, **344** (2005) 243–278.
[16] Di Caro, G., at al.: AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. European Trans. on Telecommunications, Self-organization in Mobile Networking, **16** (2005) 443–455.
[17] Xiangquan, Z., Lijia, G., Wei, G., Renting, L.: A cross-layer design and ant-colony optimization based load-balancing routing protocol for ad-hoc networks. Higher Education Press, **2** (2007) 219–229.
[18] Shih, F.: Particle Swarm Optimization Algorithm for Energy-Efficient Cluster-Based Sensor Networks. IEICE **E89-A** (2006) 1950–1958.
[19] Di Caro, et al.: Swarm intelligence for routing in mobile ad hoc networks. Proc. of the IEEE Swarm Intelligence Symposium, (2005).
[20] Vuran, M. C., Gungor, V. C., Akan, O. B.: On the Interdependence of Congestion and Contention in WSNs. Proc. of ICST SenMetrics (2005).
[21] Theraulaz, G., Bonabeau, E., Daneubourg, J-.L., Response Threshold Reinforcement and Division of Labour in Insect Societies. Proc. of Royal Society of London, **265** (1998) 327–332.
[22] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/