WSNs Coverage Hole Partial Recovery by Nodes' Constrained and Autonomous Movements Using Virtual α-chords

Ali Rafiei, Mehran Abolhasan, Daniel R. Franklin
University of Technology Sydney, Australia
ali.rafiei@student.uts.edu.au
{mehran.abolhasan, daniel.franklin}@uts.edu.au
Farzad Safaei
University of Wollongong, Australia
Farzad@uow.edu.au

***Abstract*—In WSNs, in order to recover from coverage holes and to mitigate their indirect/direct effects on networks' performance, different recovery strategies such as increasing proximate nodes' transmission range and/or relocation of nodes towards coverage holes seem to be appropriate solutions. Since the majority of a mobile node's energy is consumed by movement and since nodes' residual energy may be affected by damage events, node movements should be performed sparingly. Conventional nodes' information exchange in real-time applications with security and interference concerns are neither practical nor secure. Therefore, for the aforementioned scenarios, at the price of possible node collisions, disconnections, and reasonable compromises, promising distributed and autonomous node movement algorithms based on limited 1-hop neighbour knowledge are proposed. Our proposed autonomous and constrained node movement model based on a node's 1-hop perception provides a feasible and rapid recovery mechanism for large scale coverage holes in real-time and harsh environments. Our model not only maintains moving nodes' connectivity to the rest of network to some extent, but also offers emergent cooperative recovery behaviour among autonomous moving nodes. Our movement model based on virtual chords formed by nodes and their real and virtual 1-hop neighbours, not only confines node movement range, but also takes the issue of moving nodes' connectivity into account. Suitable performance metrics for partial recovery via constrained movement are introduced to compare the performance and efficiency of our model with conventional Voronoi-based movement algorithms. Results show that our proposed model's performance is comparable with Voronoi-based movement algorithms.**

***Index Terms*—Coverage holes; autonomous and constrained movements; Wireless sensor networks; virtual chord.**

I. INTRODUCTION

Due to the vast applications of wireless sensor networks (WSNs) [1][2], they are a key focus of attention for academic and industrial research. Deployed sensor nodes [3] can be used to detect fire [4], tsunamis [5], to monitor wildfire [6], earthquakes [7], habitats [8], environment [9], and active volcanoes [10]. New generations of sensors deployed and embedded in a variety of environments such as structures [11], underground [12], air (as unmanned aerial vehicles) [13], underwater [14], or on the sea surface [15] can be used to detect many events and phenomena, notify other nodes, and respond to the events. In addition to emerging WSN applications, diverse nodes' deployment [3], mobility, and movement patterns [16] offer new remedies to WSNs' challenges [17][18]. Despite continuous reduction in cost/size and increase in nodes' battery/processing power, an economically justifiable degree of redundancy in deployed nodes should be considered in order to have flexibility and robustness in node failure-prone environments with harsh conditions. Depending on application and environment, a trade-off between nodes' density [3] and mobility [19] (uncontrolled and controlled movements [20]) should taken into account if a proper level of quality of service is to be achieved.

Having severe direct/indirect effects on the networks' integrity and performance, large scale *coverage holes* caused by en masse node failures in a given area(s), should be avoided [21] and/or mitigated as much as possible with different recovery strategies. In WSNs, it is not always possible to deploy new nodes in unsupervised and harsh environments and dropping random nodes cannot guarantee desirable node formations and distributions. Although it may not be so economical, by benefiting from the redundant nature of deployed nodes, coverage holes to some extent can be repaired either by transmission power adjustment or the relocation of a selected set of currently deployed nodes (e.g., damaged area proximate nodes). Since movements consume the majority of nodes' energies, they should be moved carefully. Therefore, the amount of movements for proximate nodes known as *boundary node* (*B-nodes*) which participate in the recovery of damaged areas should be done sparingly; otherwise, nodes' energy exhaustion results in further cascaded failures.

Though for precise nodes movements a reasonable amount of message exchanges are required, in real-time scenarios with security and interference considerations, they are neither desirable nor secure. Therefore, by putting the burden of autonomous decision and more processing on individual B-nodes who directly detected the damage events, the number of exchanged messages can be kept as small as possible. Autonomous movement decision-making has the drawback

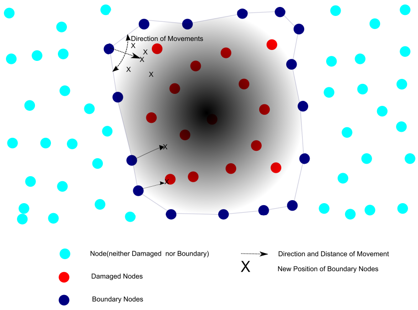


Fig. 1: Coverage Hole and Node Types

of increasing the possibility of collision and disconnection among nodes. Moreover, improvised, unconstrained, and careless movements towards damaged area(s) may cause multiple newly formed coverage holes. It should be noted, however, that for the sake of temporal coverage, a coverage hole may be virtually displaced via controlled group node movements as in [22]. Our model of autonomous and constrained node movements towards the coverage hole tries to maintain connectivity of the moving nodes with their 1-hop immediate neighbours. These autonomous movements in each of the moving nodes are inferred solely from the limited knowledge of node's 1-hop neighbours before the *damage event* as well as the perceived 1-hop neighbors' status change after damage event without *any* additional message exchanges. In our model each boundary node forms a α-*virtual chord* through its selected real and virtual neighbours (the other endpoint of the given chord) with the length of $\alpha \cdot 2 \cdot R_c \leq 2 \cdot R_c$ $(0 \leq \alpha \leq 1)$ (Fig. 2). Each virtual chord's endpoints (i.e., node's real and virtual neighbours) lie on the circumference of the two distinct circles with equal radius of R_c. One of these circles is considered a *valid circle* if it is closer to the damaged area. With α-virtual chord node movement, relocated B-node from s to s' maintains the connectivity with its real and virtual neighbours n and n', provided its real neighbour n is not a moving B-node. In our proposed model, not only an ensemble nodes' emergent cooperative movement behaviour [23] is manifested, but also group mobility and cooperative behaviour of moving nodes can be changed by using different values of α. So by changing α, the direction of moving nodes towards the coverage hole changes to the direction of nodes circulating around it. The former is suitable for the case of hole recovery while the latter is geared to prevent cascaded failure and failure expansion around damaged area as lower residual energy B-nodes can be replaced with other moving nodes with higher energy as a result of nodes constrained circular movements (Fig. 3). These types of different group mobility behaviours can be implemented by collection of nodes' autonomous movements via local decisions made based on simple nodes' geometrical and statistical features.

To our best knowledge, very few works considered partial recovery of large scale coverage hole via autonomous constrained node movements for time-sensitive scenarios with security consideration. Moreover, few works used damaged nodes' statistical and geometrical features as the landmarks in nodes' local and autonomous decision making processes. We have also defined proper performance metrics in order to compare the performance and efficiency of our proposed model with conventional Voronoi-based movement models (VOR and MinMax) [24][18]. In Section II, we present current work on nodes movement. In Section III, our model and assumptions are introduced. In Section IV, our proposed performance metric are briefly discussed, and finally, in Sections V and IV, result, conclusion, and future work are respectively presented.

II. RELATED WORK

Mobility in wireless sensor networks is a double edge sword; on one hand, undesirable and uncontrolled mobility causes coverage holes and topological instability, while on the other hand, coverage hole(s) can be repaired by controlled mobility and movement of nodes [20][17]. Thus node relocations [18] are important in enhancing networks coverage and connectivity [25], by offering *temporal coverage* in addition to *spatial coverage* [26] for an area in which the number of nodes is not sufficient to cover it all time. Thus via controlled mobility, a trade-off between the number of required deployed nodes and the required coverage of the given area can be reached [19]. Controlled mobility not only is able to repair the coverage hole but also it can correct irregularities of uncontrolled mobility [20]. After deployment, especially in hostile and hazardous environments, it is almost impossible to have centralised control over sensors. Thus, in such case in order to repair coverage holes, nodes not only should be able to decide autonomously on their movements but also they should not exhaust their energy as majority of nodes' residual energy would be consumed by their movements. There are a variety of relocation, movement and deployment model in the literature [18][24][27][28][29] [30][31][32][33] which mainly aim to keep network coverage, balance node deployments, and repair small coverage holes due to improper node deployments, single or random node failure.

Movement algorithms can be divided into (virtual) radial [33] and angular [32] *force-based*, *flip-based* [28] and *Voronoi-based* [24] movement algorithms. Movement based on virtual potential repulsion and attraction [33] between pairs of nodes and the movement of nodes as the result of aggregation of these forces are inspired by physical laws of nature. Virtual angular force [32] tries to connect the partitions and parts of network by using collaborative movement of mobile nodes applying on the angle of moving nodes.

In order to exert proper levels of virtual repulsion and attraction, nodes should be globally aware of the their targeted density. Since the movement algorithm is applied to all nodes, movement contains oscillation due to mutual interaction of nodes; consequently, an unnecessary amount of nodes' energy is consumed. In flip-based movement algorithm [28], the given area is divided into regions and a head node is elected for each region. In the case of head failure and unbalance number of nodes, nodes from the neighbour regions would flip into the given region. In flip-based movement, the head node for each region should be selected, which requires message exchange among nodes. Since movement is confined to neighbouring regions, the recovery of large scale coverage hole may consists of many iterations of nodes flipping into their neighbouring regions with an agreed-upon granularity. So flip-based movement algorithms are expected to be inefficient for real-time scenarios with large scale holes. In Voronoi-based movement algorithms, [24], the area is decomposed into Voronoi diagrams [34] depending on the deployment and distribution of nodes. If a node fails or part of area is not covered by the sensor network, nodes move with regard to their Voronoi vertices to compensate for *void area(s)*. Voronoi-based movement most often is required to have global knowledge to form Voronoi diagrams. Voronoi-based movement algorithms are not geared for large scale coverage holes as they result in newly formed small coverage holes. They also suffer from oscillation and consequently energy exhaustion if recovery is performed in an iterative style. Complex and centralised node movements and even distributed algorithms (with pre-computed movements) have a good energy management, however, they are not efficient for real-time scenarios as they suffer from unacceptable delay, particularly under very fast-changing conditions. So diffused information and nodes' notifications are not valid and already obsolete for the decision making process.

III. METHODS AND ASSUMPTIONS

A. Sensor Model and Node Types

Homogeneous sensor nodes are modelled based on the unit disk graph (UDG) [35] and are bidirectionally connected if they are within each other's ranges. Nodes are randomly deployed with uniform distribution in a rectangular area of $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$. To avoid unnecessary complexity, it is assumed that transmission range (R_c) and sensing range (R_s) are equal. Although no central coordination is required and a local coordination system is applicable in our model, sensors' locations may be known by GPS or any other localisation methods [36]. Sensor nodes are classified into *damaged* nodes (*D-nodes*) if they reside inside the *damaged area* (*D-area*); otherwise, they are considered as *undamaged* nodes (*U-nodes*). Those proximate U-nodes to D-area which directly detect the *damage event* (*D-event*) within their ranges are further classified into *boundary nodes* (*B-nodes*). B-nodes detect the D-event as they sense any significant changes within their ranges such as signal loss or disconnection due to the failure of their neighbours. It should be noted that noise, false alarms, or transient, periodic, and frequent failure of nodes, and link instability are excluded in our model.

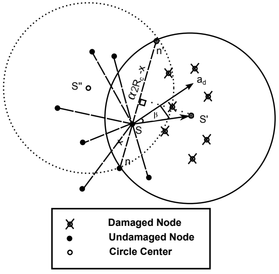


Fig. 2: B-Node, its real, virtual neighbors and virtual chord

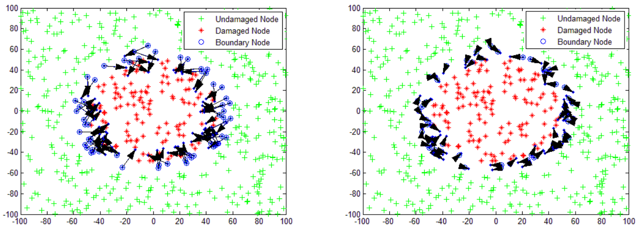
B. Coverage hole

Coverage hole are modelled in different forms in the literature [22][37][38]. Similar to [39][40], coverage holes are modelled as a circle of radius r_{Hole} with the centre at x_{Hole}, y_{Hole}. Since each B-node autonomously perceives the D-event and its damaged neighbours, a *margin* of B-nodes (*MB-nodes*) are formed around the D-area. Thus, to benefit from WSNs redundancy and to reduce possibility of interference and collision, a set of the B-nodes defined as *selected B-nodes* (*SB-nodes*) [39][40] are selected which may partake in a possible recovery process by moving towards the *region of interest* (*ROI*). SB-nodes may be selected by a distributed algorithm or centrally selected based on the agreed criteria. Similar to [39][40], B-nodes are selected in a distributed fashion based on B-node's 1-hop geometrical and statistical features.

C. Selected Neighbor Nodes

B-node's neighbours can be classified into D-nodes or U-node depending on their location relative to the coverage hole. Based on the type of B-node's neighbours, they can be defined as the *undamaged neighbour nodes* (*UN-nodes*) or *damaged neighbour nodes* (*DN-nodes*. At the time of the D-event, each B-node's distances to both sets of its UN-nodes and DN-nodes as well as their degrees of connectivity are used as *landmarks* in decision making processes. Therefore, if a B-node selects a set of its UN-node(s) via some selection algorithms, those UN-nodes are considered as *selected undamaged neighbour nodes*(real neighbours). *Virtual selected undamaged neighbour nodes* are the fictitious B-nodes' neighbours (virtual neighbours) which are connected to B-node's UN-nodes via *virtual chords* defined as α-chord with the length of $\alpha \cdot 2 \cdot R_c \leq 2 \cdot R_c$ ($0 \leq \alpha \leq 1$) (Fig. 2). Three neighbour node selection algorithms, namely *closest neighbour*, *random neighbour*, and *β-angle* are presented in Algorithm 1.

In the closest neighbour algorithm, a B-node's closest 1-hop neighbour is selected, while in the random select algorithm one of the B-node's 1-hop neighbours is randomly selected. In the β-angle algorithm, for each B-node and the undamaged node in its neighbour set, set of angles can be formed between normal direction of the virtual chords (Fig. 2) and distance vector from the B-node to its D-nodes centre of mass. B-node's neighbour whose aforementioned angle is closer to β than any other of B-nodes's 1-hop undamaged neighbours are considered as the selected undamaged neighbour and should be unique. If more than one undamaged neighbour can be selected based on the mentioned conditions, only one of them should be randomly chosen as the selected undamaged neighbour. In finding the centre of mass of B-nodes' undamaged and damaged neighbours, if the neighbours' degrees of connectivity are taken into account (Algorithm 1) they can be considered as weighted, β-angle algorithms with $w = 1$; otherwise the are called β-angle with $w = 0$.



(a) $\alpha = 0$ (b) $\alpha = 1$

Fig. 3: Chord Movement Algorithm (R_c=15, N=600, β=0)

D. Movement Model

Movement algorithms can be divided into following states: 1) undamaged neighbour nodes of moving B-nodes are selected based on criteria presented in Algorithm 1; 2) with regard to the suitable virtual chord parameters α and R_c, the location of B-nodes' virtual neighbours are obtained for each B-node; 3) new locations of moving B-nodes computed by selecting one of two circles which pass through the endpoints of the virtual chord of each B-node (Algorithm 2). The selected circle is defined as the *valid circle* through which the chord is obtained. The Valid circle is the circle with its centre closer to damage area; 4) the B-node then moves to the centre of the valid circle with probability p (uniform distribution) and q otherwise, such that $p + q = 1$. Here we assumed $p = 1$ in which all B-nodes move towards the coverage hole. It should be noted that connectivity of B-nodes to its neighbours can not be fully guaranteed. This is because after the damage event, B-nodes are not able to distinguish if their undamaged neighbours are moving B-nodes or or not. As an example, Fig. 3 shows how changing parameter α affects B-nodes' collective movement behaviour in our coverage hole recovery model. β-angle with $\alpha = 0/1$ in Fig. 3 show the direction of moving B-nodes towards/around coverage hole.

IV. PERFORMANCE METRICS

We have compared our proposed movement algorithm with the two Voronoi-based movement algorithms (VOR-MinMax) [24] via three types of proposed performance metrics. In Voronoi-based algorithms, B-nodes were selected similarly to our previous work [40]. In modelling Voronoi movement algorithms, we have also considered the problem of nodes with out-of-area and infinite Voronoi vertices. The proposed performance metrics are classified below:

Coverage-based metrics: We define *percentage of recovery* as the percentage of recovered networks' 1-coverage after the recovery process. In other words, the metric shows by using the given movement algorithm what percentage of lost 1-coverage is recovered in the network.

Connectivity-based metrics: We define *percentage of connectivity* as the percentage of moving B-nodes which are directly connected to rest of network (those nodes which did not participate in the recovery process) with at least one link over the total number of moving B-nodes. This performance metric shows the effect of movement algorithms on the connectivity of moving nodes and how many of the moving B-nodes are still directly connected to the rest of the network after their movements.

Algorithm 1: Nodes' neighbors selection Algorithms

```
Input:
$s_i^b$ : B-node $i$ ($i = 1, \cdots, m$),  $N_{s_i}^h$ : $s_i^b$'s h-hop neighbours
$N_{s_i}^{h_u}$ : h-hop U-node neighbours of $s_i^b$
$N_{s_i}^{h_d}$ : h-hop D-node neighbours of $s_i^b$
$\vec{X}_{s_i}^{s_j^{h_u}}$ distance vector from $s_i$ to $s_j$ ($j$ in $N_{s_i}^{h_u}$)
$\vec{X}_{s_i}^{s_j^{h_d}}$ distance vector from $s_i$ to $s_j$ ($j$ in $N_{s_i}^{h_d}$)
$\beta - angle$ : angle parameter
$d_{S_j}^{h_u}$ degree of $s_j$ ($j$ from $N_{s_i}^{h_d}$)
$d_{S_j}^{h_d}$ degree of $s_j$ ($j$ from $N_{s_i}^{h_u}$)
Output: Set of selected h-hop neighbour $s_j^{h_u s_i^b}$

case Closest if closest neighbour selected
    foreach B-Node $s_i^b$ do
        Find $N_{s_i}^{h_u}$
        foreach h-hop UN-nodes $s_j^{h_u}$ do
            Calculate $\vec{X}_{s_i}^{s_j^{h_u}}$
        Calculate $arg_j Min \left( \left| \vec{X}_{s_i}^{s_j^{h_u}} \right| \right)$
case Random if Random neighbor Selected
    foreach B-Node $s_i^b$ do
        Find $N_{s_i}^{h_u}$
        Calculate $arg_j Random \left( N_{s_i}^{h_u} \right)$
case $\beta$-angle if $\beta$-angle is Selected
    foreach B-Nodes$_i^b$ do
        Find $N_{s_i}^{h_u}$ and $N_{s_i}^{h_d}$
        foreach h-hop(DN-node $s_j^{h_d}$, UN-node $s_j^{h_u}$) do
            Find $d_{S_j}^{h_d}$, $d_{S_j}^{h_u}$
            Calculate $\vec{X}_{s_i}^{s_j^{h_d}}$ ,$\vec{X}_{s_i}^{s_j^{h_u}}$
        Calculate $\vec{X}_{CM\,si}^{\;h_d} = \frac{\sum (\vec{X}_{s_i}^{s_j^{h_d}}) \cdot d_{S_j}^{h_d}}{\sum d_{S_j}^{h_d}}$
        Calculate $\vec{X}_{CM\,s_i}^{\;h_u} = \frac{\sum (\vec{X}_{s_i}^{s_j^{h_u}}) \cdot d_{S_j}^{h_u}}{\sum d_{S_j}^{h_u}}$
        foreach h-hop UN-node $s_j^{h_u}$ do
            Calculate $\angle\gamma_{s_i}^{h_u S_j} = \angle \left( \vec{X}_{CM\,si}^{\;h_d}, \vec{X}_{s_i}^{s_j^{h_u}} \right) - \angle\beta$
        Calculate $arg_j Min \left( \left| cos(\angle\gamma_{s_i}^{h_u S_j}) \right| \right)$
```

```
Algorithm 2: Formation of Chord Algorithm
Input:
$s_i^b$ : B-node $i$ ($i = 1, \cdots, m$),    $\tau$ : threshold
$s_j^{h_{u_{s_i^b}}}$ :  Selected h-hop U-node neighbour $s_j$
$\alpha$-chord parameter,    $R_c$ transmission Range
$N_{s_i}^{h_u}$ : h-hop U-node neighbours of $s_i^b$
$N_{s_i}^{h_d}$ : h-hop D-node neighbours of $s_i^b$
Output:
B-nodes $s_i^b$'s new location (coordinates), ${s'}_i^b(x,y)$
foreach B-node $s_i^b$ do
    Find $s_i^b$'s current location (coordinates) of $s_i^b(x,y)$
    Find $CM_i^{h_u(x,y)}$ : $s_i^b$'s h-hop UN-nodes' center of mass
    Find $CM_i^{h_d(x,y)}$ : $s_i^b$'s h-hop DN-nodes' center of mass
    Calculate $chord-\alpha_i$, virtual node ${s'}_j^{h_{u_{s_i^b}}}$ from $\alpha_i$ and $R_c$
    Find $C_{\alpha_i}^{(x,y)_{k,k'}}$ (circle center(s)) of $chord_{\alpha_i}$
    foreach $chord_{\alpha_i}$ and $C_{\alpha_i}^{(x,y)_{k,k'}}$ do
        if $\left\|C_{\alpha_i}^{(x,y)_k} - CM_i^{h_d(x,y)}\right\| < \left\|C_{\alpha_i}^{(x,y)_{k'}} - CM_i^{h_d(x,y)}\right\|$ then
            $C_{Valid_{\alpha_i}}^{(x,y)} = C_{\alpha_i}^{(x,y)_k}$
        else if $\left\|C_{\alpha_i}^{(x,y)_k} - CM_i^{h_d(x,y)}\right\| > \left\|C_{\alpha_i}^{(x,y)_{k'}} - CM_i^{h_d(x,y)}\right\|$ then
            $C_{Valid_{\alpha_i}}^{(x,y)} = C_{\alpha_i}^{(x,y)_{k'}}$
        else if $\left\|C_{\alpha_i}^{(x,y)_k} - CM_i^{h_d(x,y)}\right\| = \left\|C_{\alpha_i}^{(x,y)_{k'}} - CM_i^{h_d(x,y)}\right\|$ then
            Calculate rand $p \sim U[0,1]$
            if $p > \tau$ then
                $C_{Valid_{\alpha_i}}^{(x,y)} = C_{\alpha_i}^{(x,y)_k}$
        else
            $p < \tau$
        $C_{Valid_{\alpha_i}}^{(x,y)} = C_{\alpha_i}^{(x,y)_{k'}}$
    ${s'}_i^b(x,y) = C_{Valid_{\alpha_i}}^{(x,y)}$
```

Distance-based metrics: We define *average movement* as the ratio of total amount of movement to the number of participating nodes in recovery process. Average movement can be used with other metrics to better understand the behaviour of movement algorithms in coverage hole recovery process.

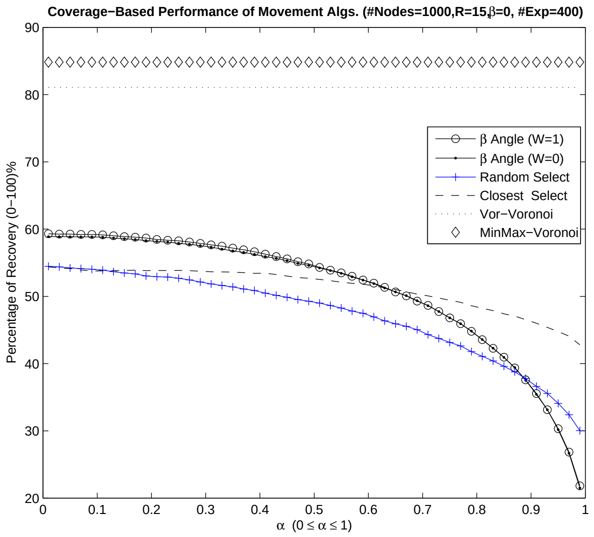


Fig. 4: Percentage of Recovery

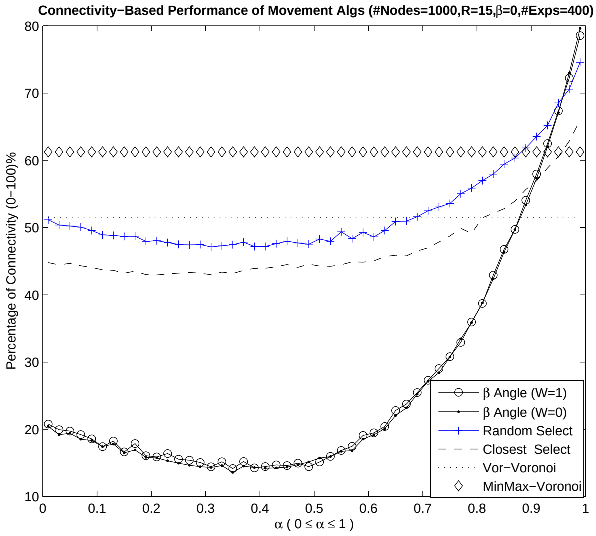


Fig. 5: Percentage of Connectivity

V. RESULTS

Using Matlab, N=1000 nodes with communication and sensing range of 15 ($R_c = R_s = 15$ m) are uniformly deployed with random distribution in a rectangular area of $[-100, 100] \times [-100, 100]$. Similarly to [39][40], coverage holes are modelled as circles with radius $r_{Hole} = 50$ m located at $(x_{Hole}, y_{Hole}) = (0, 0)$. The experiment was repeated $\#Exp = 400$ times for all movement algorithms. Chord parameter (α) is continuously changed from 0 to 1 to examine its effect in the performance and node collective behaviour of the proposed movement algorithms. Results with error bars (97.5% confidence intervals) are not included here due to space limit (Figs. 4-6).

Performance metrics of movement algorithms are also shown in Table I. With regard to Figs. 4-6, as α continuously

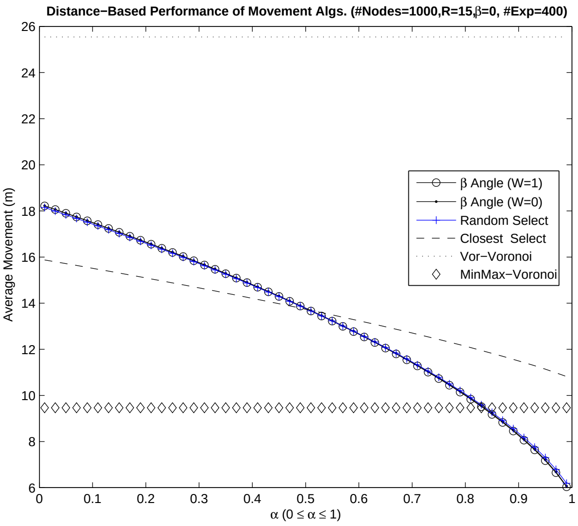


Fig. 6: Average Movement

changes from 0 to 1, the percentage of recovery and nodes' average movements of virtual chord movement algorithms decreased but at the same time their percentages of connectivity increased, which shows that B-nodes' collective behaviour and direction of movements shift gradually from moving towards to circulating around coverage hole. Each of these collective mobility behaviours can be used for different purposes. In our model, α can be chosen in such a way as to achieve proper percentage of connectivity, percentage of recovery with given amount of nodes' movement.

Results from Figs. 4-6 and Table I show that although proposed chord movement algorithm is autonomous and require few or no message exchange, its performance is comparable to Voronoi-based movements.

In the real-time scenarios with security and interference concerns, using Voronoi-based algorithms requires global knowledge of the network. So even if higher coverage and connectivity is offered, in these scenarios, they not practical. It should be noted that performance of our proposed model would change with regard to other network parameters such as network node density, node range, and coverage hole radius, node deployment distribution, etc. Therefore, their effects should be examined in more detail.

VI. CONCLUSION AND FUTURE WORK

A new autonomous and constrained node movement model is proposed to partially/wholly recover large scale coverage holes in real-time scenarios with interference and security consideration. Our proposed model of autonomous decision making is based on the available 1-hop knowledge at the time of the damage event. By introducing the concept of α-chords, our proposed model not only taken the connectivity of moving nodes into account, but it also shows an emergent cooperative recovery behaviour. To compare our proposed model with conventional Voronoi-based algorithms, suitable performance metrics were introduced.

Algs.	α	**Recovery(%)**	**Connectivity(%)**	**Avg. Mov.(m)**
β-angle (w=1)	0	59.3000	20.5821	18.2413
	0.25	58.1561	14.9848	16.1297
	0.50	54.3100	15.7554	13.6721
	0.75	46.3983	32.1165	10.5888
	1.0	21.8333	84.2525	5.6323
β-angle(w=0)	0	58.8752	20.4436	18.2227
	0.25	57.8314	14.4888	16.1230
	0.50	54.2239	15.7458	13.6797
	0.75	46.4474	31.7605	10.6149
	1.0	21.5037	84.0388	5.6850
Closest	0	54.3857	43.9619	15.7489
	0.25	53.8395	42.9475	14.7811
	0.50	52.5149	43.9536	13.7172
	0.75	49.2130	48.7582	12.5073
	1.0	42.8042	67.2597	11.0456
Random	0	54.4647	52.0741	18.1173
	0.25	52.5196	49.3625	16.0505
	0.50	49.0044	49.2323	13.6488
	0.75	42.9059	55.0499	10.6438
	1.0	30.0398	77.9009	5.8566
Vor	0	81.0696	51.3128	25.5432
	0.25	81.0696	51.3128	25.5432
	0.50	81.0696	51.3128	25.5432
	0.75	81.0696	51.3128	25.5432
	1.0	81.0696	51.3128	25.5432
MinMax	0	84.8375	61.5663	9.4616
	0.25	84.8375	61.5663	9.4616
	0.50	84.8375	61.5663	9.4616
	0.75	84.8375	61.5663	9.4616
	1.0	84.8375	61.5663	9.4616

TABLE I: Performances of Movement Algorithms

As future work, new autonomous constrained node movements models can be defined. The issue of trade-off between nodes' amount of exchanged information and degree of node autonomy can be investigated. The problem of nodes' connectivity and collisions should also be addressed in more details in future autonomous models.

In order to show the effects of a coverage hole on its proximate nodes, node residual energy models should be included in recovery models. Undesirable secondary effects of imprudent node movements such as formation of new coverage holes should be examined more comprehensively. Probabilistic autonomous prediction of nodes' neighbours status without exchanging any additional messages to achieve emergent cooperative behaviour via autonomous nodes is also expected to be an interesting future work. New models of one-time autonomous node movements instead of iterative nodes' movements can be considered to reduce the problem newly formed coverage holes, oscillation, and energy in the network.

VII. ACKNOWLEDGMENT

This research was supported by the Australian Research Council (ARC) discovery research grant No. DP0879507.

REFERENCES

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] C. Xiao, Y. Peng, and M. Yu, “The deployment method and movement control strategy in mobile wireless sensor networks,” in *International Symposium on Computer Science and Computational Technology, ISCSCT 2008*, vol. 2, Dec. 2008, pp. 520–523.

[4] A. Khadivi and M. Hasler, “Fire detection and localization using wireless sensor networks,” in *Sensor Applications, Experimentation, and Logistics*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, N. Komninos, O. Akan, P. Bellavista, J. Cao, F. Dressler, D. Ferrari, M. Gerla, H. Kobayashi, S. Palazzo, S. Sahni, X. S. Shen, M. Stan, J. Xiaohua, A. Zomaya, and G. Coulson, Eds. Springer Berlin Heidelberg, 2010, vol. 29, pp. 16–26.

[5] S. Bhima, A. Gogada, and R. Garimella, “A tsunami warning system employing level controlled gossiping in wireless sensor networks,” in *Proceedings of the 4th international conference on Distributed computing and internet technology*, ser. ICDCIT 2007. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 306–313.

[6] Y. Li, Z. Wang, and Y. Song, “Wireless sensor network design for wildfire monitoring,” in *The Sixth World Congress on Intelligent Control and Automation, WCICA 2006*, vol. 1, 2006, pp. 109 –113.

[7] M. Suzuki, S. Saruwatari, N. Kurata, and H. Morikawa, “A high-density earthquake monitoring system using wireless sensor networks,” in *Proceedings of the 5th international conference on Embedded networked sensor systems*, ser. SenSys 2007. New York, NY, USA: ACM, 2007, pp. 373–374.

[8] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless sensor networks for habitat monitoring,” in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ser. WSNA 2002. New York, NY, USA: ACM, 2002, pp. 88–97.

[9] T. Arici and Y. Altunbasak, “Adaptive sensing for environment monitoring using wireless sensor networks,” in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 4, March 2004, pp. 2347–2352 Vol.4.

[10] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, “Deploying a wireless sensor network on an active volcano,” *Internet Computing, IEEE*, vol. 10, no. 2, pp. 18–25, March-April 2006.

[11] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, “Health monitoring of civil infrastructures using wireless sensor networks,” in *Proceedings of the 6th international conference on Information processing in sensor networks*, ser. IPSN 2007. New York, NY, USA: ACM, 2007, pp. 254–263.

[12] M. Li and Y. Liu, “Underground structure monitoring with wireless sensor networks,” in *Proceedings of the 6th international conference on Information processing in sensor networks*, ser. IPSN 2007. New York, NY, USA: ACM, 2007, pp. 69–78.

[13] K. Daniel, S. Rohde, and C. Wietfeld, “Leveraging public wireless communication infrastructures for uav-based sensor networks,” in *2010 IEEE International Conference on Technologies for Homeland Security (HST)*, Nov. 2010, pp. 179–184.

[14] I. F. Akyildiz, D. Pompili, and T. Melodia, “Underwater acoustic sensor networks: research challenges,” *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, 2005.

[15] M. Jiang, Z. Guo, F. Hong, Y. Ma, and H. Luo, “Oceansense: A practical wireless sensor network on the surface of the sea,” in *IEEE International Conference on Pervasive Computing and Communications, 2009. PerCom 2009*, March 2009, pp. 1–5.

[16] R. R. Roy, *Handbook of mobile ad hoc networks for mobility models.* New York, NY: Springer, 2011.

[17] B. Liu, P. Brass, O. Dousse, P. Nain, and D. Towsley, “Mobility improves coverage of sensor networks,” in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc 2005. New York, NY, USA: ACM, 2005, pp. 300–308.

[18] B. Wang, H. B. Lim, and D. Ma, “A survey of movement strategies for improving network coverage in wireless sensor networks,” *Computer Communications*, vol. 32, no. 1314, pp. 1427–1436, 2009.

[19] W. W. V. Srinivasan and K.-C. Chua, “Trade-offs between mobility and density for coverage in wireless sensor networks,” in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, ser. MobiCom 2007. New York, NY, USA: ACM, 2007, pp. 39–50.

[20] J. Luo, D. Wang, and Q. Zhang, “Double mobility: coverage of the sea surface with mobile sensor networks,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 13, pp. 52–55, June 2009.

[21] F. Yu, E. lee, Y. Choi, S. Park, D. Lee, Y. tian, and S.-H. Kim, “A modeling for hole problem in wireless sensor networks,” in *Proceedings of the 2007 international conference on Wireless communications and mobile computing*, ser. IWCMC 2007. New York, NY, USA: ACM, 2007, pp. 370–375.

[22] C.-Y. Chang, W.-C. Chu, C.-Y. Lin, and C.-F. Cheng, “Energy-balanced hole-movement mechanism for temporal full-coverage in mobile wsns,” in *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, ser. IWCMC 2010. New York, NY, USA: ACM, 2010, pp. 89–93.

[23] J. E. Doran, S. Franklin, N. R. Jennings, and T. J. Norman, “On cooperation in multi-agent systems,” *Knowl. Eng. Rev.*, vol. 12, pp. 309–314, September 1997.

[24] M. Argany, M. Mostafavi, and F. Karimipour, “Voronoi-based approaches for geosensor networks coverage determination and optimisation: A survey,” in *2010 International Symposium on Voronoi Diagrams in Science and Engineering (ISVD)*, June 2010, pp. 115–123.

[25] A. Ghosh and S. K. Das, “Coverage and connectivity issues in wireless sensor networks: A survey,” *Pervasive and Mobile Computing*, vol. 4, no. 3, pp. 303–334, 2008.

[26] C. Liu and G. Cao, “Spatial-temporal coverage optimization in wireless sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 4, pp. 465–478, April 2011.

[27] M. Younis and K. Akkaya, “Strategies and techniques for node placement in wireless sensor networks: A survey,” *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.

[28] S. Chellappan, X. Bai, B. Ma, D. Xuan, and C. Xu, “Mobility limited flip-based sensor networks deployment,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 2, pp. 199–211, Feb. 2007.

[29] A. Sekhar, B. Manoj, and C. Murthy, “Dynamic coverage maintenance algorithms for sensor networks with limited mobility,” in *Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005*, March 2005, pp. 51–60.

[30] X. Bai, S. Li, and J. Xu, “Mobile sensor deployment optimization for k-coverage in wireless sensor networks with a limited mobility model,” *IETE Technical Review*, vol. 27, no. 2, p. 124, 2010.

[31] S. Poduri and G. Sukhatme, “Constrained coverage for mobile sensor networks,” in *2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA 2004*, vol. 1, April-1 May 2004, pp. 165 – 171 Vol.1.

[32] A. Casteigts, J. Albert, S. Chaumette, A. Nayak, and I. Stojmenovic, “Biconnecting a network of mobile robots using virtual angular forces,” in *2010 IEEE 72nd Vehicular Technology Conference Fall (VTC 2010-Fall)*, Sept. 2010, pp. 1–5.

[33] Y. Zou and K. Chakrabarty, “Sensor deployment and target localization based on virtual forces,” in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2, March-3 April 2003, pp. 1293–1303 vol.2.

[34] F. Aurenhammer, “Voronoi diagramsa survey of a fundamental geometric data structure,” *ACM Comput. Surv.*, vol. 23, pp. 345–405, September 1991.

[35] S. Schmid and R. Wattenhofer, “Algorithmic models for sensor networks,” in *20th International Parallel and Distributed Processing Symposium, IPDPS 2006*, April 2006.

[36] G. Mao, B. Fidan, and B. D. Anderson, “Wireless sensor network localization techniques,” *Computer Networks*, vol. 51, no. 10, pp. 2529 – 2553, 2007.

[37] Z. Jiang, J. Wu, A. Agah, and B. Lu, “Topology control for secured coverage in wireless sensor networks,” in *IEEE International Conference on Mobile Adhoc and Sensor Systems, (MASS) 2007*, Oct. 2007, pp. 1–6.

[38] G. Dini, M. Pelagatti, and I. M. Savino, “An algorithm for reconnecting wireless sensor network partitions,” in *Proceedings of the 5th European conference on Wireless sensor networks*, ser. EWSN 2008. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 253–267.

[39] A. Rafiei, “Boundary node selection algorithms by simple geometrical properties in wsns,” in *Fifth Asia Modelling Symposium (AMS), 2011*, May 2011, pp. 221–226.

[40] A. Rafiei, M. Abolhasan, D. Franklin, and F. Safaei, “Boundary node selection algorithms in wsns,” in *The 36th IEEE Conference on Local Computer Networks (LCN), 2011*, October 2011, pp. 255–258.