

# Boundary Node Selection Algorithms in WSNs

Ali Raffei, Mehran Abolhasan and Daniel Franklin

School of Computing and Communications

University of Technology Sydney (UTS)

Sydney, Australia

Ali.Raffei@student.uts.edu.au

{Mehran.Abolhasan,Daniel.Franklin}@uts.edu.au

Farzad Safaei

School of Electrical, Computing and Telecommunications

Engineering (SECTE)

Faculty of Informatics, University of Wollongong (UOW)

Wollongong, Australia

Farzad@uow.edu.au

**Abstract**—Physical damage and/or node power exhaustion may lead to coverage holes in WSNs. Coverage holes can be directly detected by certain proximate nodes known as *boundary nodes* (*B-nodes*). Due to the sensor nodes' redundant deployment and autonomous fault detection, holes are surrounded by a margin of *B-nodes* (*MB-nodes*). If all *B-nodes* in the margin take part in the hole recovery processes, either by increasing their transmission power or by relocating towards region of interest (ROI), the probability of collision, interference, disconnection, and isolation may increase affecting the rest of the network's performance and QoS. Thus, distributed boundary node selection algorithms (*BNS-Algorithms*) are proposed to address these issues. *BNS-algorithms* allow *B-nodes* to self-select based on available 1-hop information extracted from nodes' simple geometrical and statistical features. Our results show that the performance of the proposed distributed *BNS-algorithms* approaches that of their centralized counterparts.

**Keywords**—Boundary selection Algorithms, network damage, coverage hole, emergent cooperation, wireless sensor networks.

## I. INTRODUCTION

Wireless sensor networks (WSNs) are increasingly being used to track objects and monitor various phenomena or areas of interest [1]. Automatic identification and repair of network coverage holes became one of WSN's key research as coverage holes drastically affect the networks' topology and operation [2]. Deliberate or accidental *physical damage* or *attacks* on specific or random parts of the network, uncontrolled node mobility, random and improper node deployment, emergence of natural obstacles or harsh and hostile environments and widespread node power exhaustion are among many factors which can result in the formation or expansion of coverage holes. This paper concentrates on coverage holes centered around specific areas in the network. Such holes can be termed *event-based* or *triggered coverage holes* as they are detected and sensed due to the changes they instigate on their proximate nodes (such as signal or connection loss). Nodes may be classified as *undamaged* or *damaged* (*U-node* and *D-nodes*) depending on the node's proximity to the damaged area (*D-area*) [6]. *D-nodes* can be regarded as *virtual nodes* if their information and pre-damage locations are used by the other nodes to implement boundary-node selection (*BNS*) algorithms. *U-nodes* are also further classified as either *normal nodes* (*N-nodes*) or *boundary nodes*

(*B-nodes*). *B-nodes* are those *U-nodes* which have detected damage events (*D-events*) directly within their respective connectivity ranges.

Since *U-nodes* detect the *D-events* autonomously within their respective ranges, when such an event occurs, the *D-areas* become surrounded by a *margin of B-nodes* (*MB-nodes*) (Fig. 1). In this paper, *B-nodes* selected by the *BNS-algorithms* are denoted *selected boundary nodes* (*SB-nodes*).

*Increase in transmission power of MB-nodes*, or their *movement towards the centre of the hole* are among many different hole recovery strategies. A consequence of these recovery efforts and WSNs' inherent redundancy, is that the probability of *congestion*, *collision*, *interference*, *possible disconnection*, and *isolation* of *B-nodes* increases, potentially affecting the network's overall efficiency. Therefore, it is essential to carefully choose both the *B-node* selection algorithm and the recovery mechanism. To date, little research has been conducted on the use of *D-node* information as the basis for boundary estimation and *B-node* selection.

In this paper, a number of selection algorithms are proposed to enable *B-node* self-selection solely based on the available 1-hop information deduced from *B-nodes*' statistical and geometrical features. By applying the proposed distributed *BNS-algorithms*, despite little or no message exchange overhead and limited available information, *cooperative behavior* among the selected *B-nodes* is shown to emerge [3]. All proposed distributed *BNS-algorithms* are shown to provide network performance close to their centralized counterparts in most cases. Unlike most boundary estimation and selection algorithms, they are only applied on a limited set of nodes adjacent to the damaged zone (*MB-nodes*) instead of all nodes in the network.

Section II briefly describes selected related works. In Section III, methodology and assumption are introduced. Section IV describes the performance metrics used to evaluate the proposed algorithms while Section V presents detailed simulation results. Finally, Section VI summarizes the key conclusions and outlines future research opportunities.

## II. RELATED WORKS

In [4], *statistical*, *topological*, *graph theory* and *geometrical* approaches for boundary detection and estimation are presented. Statistical methods usually work well in dense networks.

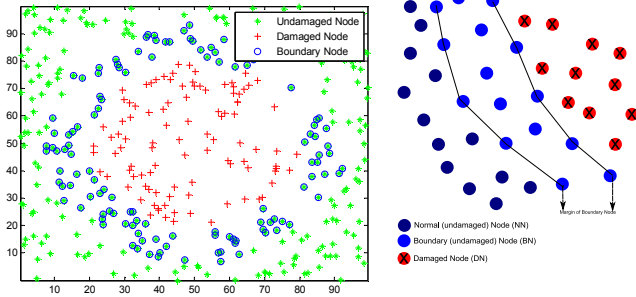


Figure 1 Network Hole and Node Types

Due to algorithmic complexity, centralized implementation and limited network resources, most topological boundary estimation methods are impractical and remain in the theoretical domain. Graph and geometrical boundary estimation methods are used in both centralized and distributed strategies. Authors in [5] defined the concept of an *orifice* as a circular arc with the radius of sensor range that is not covered by any other nodes. With the union of all network orifices, it is possible to define a contour for holes both inside and outside the boundaries of the network. Orifice-based boundary determination schemes rely on determining the regions where node communication ranges intersect and overlap. However, the implicit assumption of connectivity amongst all remaining nodes is not always valid in practice. Authors in [6] propose a boundary discovery algorithm for bidirectional, heterogeneous sensor nodes with global coordination in which nodes are aware of the geometric distance to each of their neighbors. The authors defined the network as being “sufficiently dense” when each node has at least three neighbors. If a given node may be surrounded in a triangle formed by its neighbors, it is classified as an *interior node*; otherwise, it is a *boundary node*. The *network boundary* is then defined as an imaginary line connecting all of the boundary nodes in the network. Since [6] doesn’t consider communication holes inside the network, the boundary in this case is the *perimeter of the entire network*. In [7] a local heuristic-based method for finding boundary nodes is presented. The idea is that every node tries to determine if it is inside its locally computed 1-hop convex hull or not.

Coverage holes are modeled by a variety of geometrical shapes. Regions of nodes’ Voronoi cells outside of nodes’ sensing range are defined as *blind spots* in [8], while an ellipse surrounded with a margin of B-nodes [9], a gap in a rectangular region of deployed nodes [10], a set of small cellular squares lacking elected and active node [11], hexagonal cell shapes [12] are among many proposed model for coverage hole in the literature.

### III. METHODOLOGY AND ASSUMPTIONS

Sensor nodes are modeled as unit disk graphs (UDGs), and are assumed to be homogenous and bi-directionally connected when within each other’s range. Using Matlab, 400 nodes are uniformly randomly distributed in a 100m x 100m region. For the sake of simplicity, all nodes’ communication

and sensing ranges are considered equal ( $R_c=R_s=15m$ ). Sensor nodes may be mobile or static and their location and 1-hop neighbors’ distances are known through GPS or any other localization methods [13]. It is assumed that at the time of observation each node has up-to-date and valid information about its 1-hop neighbors and surroundings. Each B-node maintains a set of undamaged and damaged neighbor nodes (UN-nodes, DN-nodes). Before the D-event, nodes have fully up-to-date information regarding their neighbors; however, after D-event, they cannot determine whether their neighbor has become a B-node or still remained as an N-node. It is also assumed that each B-node autonomously makes its selection decision based on the available 1-hop information from its neighbors before the D-event. Similar to [14], the D-area (hole) is modeled as circle with radius  $r_{hole}=30m$  located at  $(x_{hole}, y_{hole})=(50,50)$ . In the proposed distributed BNS-algorithms, it is assumed that B-nodes are aware of their degree (the number of local one-hop connections) both before and after a D-event, while they are only aware of degrees and distances of their UN-nodes and DN-nodes prior to the D-event. The proposed distributed BNS-algorithms are classified as either *distributed degree-based* or *distributed distance-based*. Centralized algorithms are also proposed to compare with distributed BNS-algorithms.

#### 1) Distributed distance and degree-based BNS-algorithms

In *Min/Max distance (MinD/MaxD)* algorithms, if B-node minimum and/or maximum distances to its DN-nodes are within a given threshold range, the B-node will consider itself as a selected B-node. The number of selected B-nodes depends on the given threshold values. In the *Variance of distances (VarD)* algorithm, the variance of distances from each B-node to its respective DN-nodes is compared with a given threshold. In the *DN-nodes and UN-nodes’ center of mass (DUCM)* algorithm, each B-node’s distances to its DN-nodes ( $D_{b(i)}$ ) and UN-nodes’ ( $U_{b(i)}$ ) centers of mass (CMs) are compared with a given threshold. UN-nodes and DN-nodes’ degrees can be used as masses in DUCM algorithm. B-nodes closer to the D-area are expected to lose greater number of their degree. Thus, the reduction in degree of a given node following the D-event is a logical criterion for B-node selection. In the proposed *absolute degree loss (ADL)* algorithm, each node compares the change in its degree before and after a D-event with a given threshold. The ADL BNS-algorithm does not always result in accurate B-node selection. As an example, a node with degree two may suffer a reduction to degree one or zero after the D-event, but still may not satisfy the threshold. Therefore, ADL is modified to create the *relative degree loss (RDL)* algorithm, in which relative reduction in the degree relative to the initial degree of the node is used as the selection criterion.

#### 2) Centralized BNS-algorithms

In the centralized BNS-algorithm, decisions are based on global knowledge of the network and D-event. To compare our distributed BNS-algorithms with centralised schemes, we propose the *Closer Nodes First (CNF)*, *Voronoi-area (VA)* and *Voronoi-distance (VD)* algorithms. In the CNF algorithm, B-nodes with smaller Euclidean distances to the D-area are

selected first. In Voronoi-area (VA) and Voronoi-distance (VD) algorithms, B-node selection is based on the features of the nodes' Voronoi diagrams [15]. In the Voronoi diagram,  $Cell_{s(i)}$  and  $Cell_{s(j)}$  are neighbors if they share at least one vertex ( $VCell_{s(i)} \cap VCell_{s(j)} \neq \emptyset$ ). B-node cells are classified according to their neighbor cells; if a B-node cell has a D-node neighbor cell, it can be considered as *B-D Voronoi cell*, otherwise; it is classified as a *B-B Voronoi cell*. In the VA algorithm, B-D cells with larger areas are selected first. Similarly, the VD algorithm prefers B-D cells closer to the D-area. If the required number of selected B-nodes is larger than the available number of B-D cells, the selection should also be performed amongst the remaining B-B cells. Thus, in our algorithms, previously selected B-node cells are now considered as D-node cells and their B-B cell neighbors are considered as B-D nodes. To reduce complexity, in VA, instead of Voronoi cells' actual geometric area, their in-circle areas are computed. Since selection is based on the Voronoi cell neighborhood relations, by changing the cell's role of B-B

into B-D cells, we are able to perform stage-wise B-node selection (based on Voronoi proximity) to model failure expansion of D-area. An example of this stage-wise selection is shown in the Table 1. In Table 1, D-nodes, B-nodes and SB-nodes are denoted '+', 'o', and '\*' respectively in the left column (a, c, e, g) and in-circle of cells in each stage of selection on the right column (b, d, f, h).

#### IV. PERFORMANCE METRICS

SB-nodes which are closer to a D-area can obtain more accurate measurements of a D-event. Hence, if B-node  $b_i$  is located at  $(x_i, y_i)$ , coverage hole of radius of  $r_{hole}$  is located at  $(x_{hole}, y_{hole})$  and  $N_{b(i)}$  is number of selected B-nodes, the *Average Distance from damaged area (AVD)* of SB-nodes can be defined as :

$$AVD = \frac{1}{N_{b(i)}} \left\{ \sum_{i \in b} [(x_i - x_{hole})^2 + (y_i - y_{hole})^2]^{1/2} - r_{hole} \right\}, \quad (1)$$

Damage resulting from the D-event only affects the rectangular area of  $[x_{hole} - r_{hole}, x_{hole} + r_{hole}]$ ,  $[y_{hole} - r_{hole}, y_{hole} + r_{hole}]$ , which is termed the *given area*. This area is divided into grid cells and coverage of cells is measured as the number of nodes covering the cells' corner coordinates  $z_i = (x_i, y_i)$ . *Percentage of Coverage (PCov)* is defined as the ratio of 1-covered cells by SB-nodes in the given area to total number of cells in the given area. It should be noted that due to D-nodes, the given area may be not completely covered even if all MB-nodes are selected.

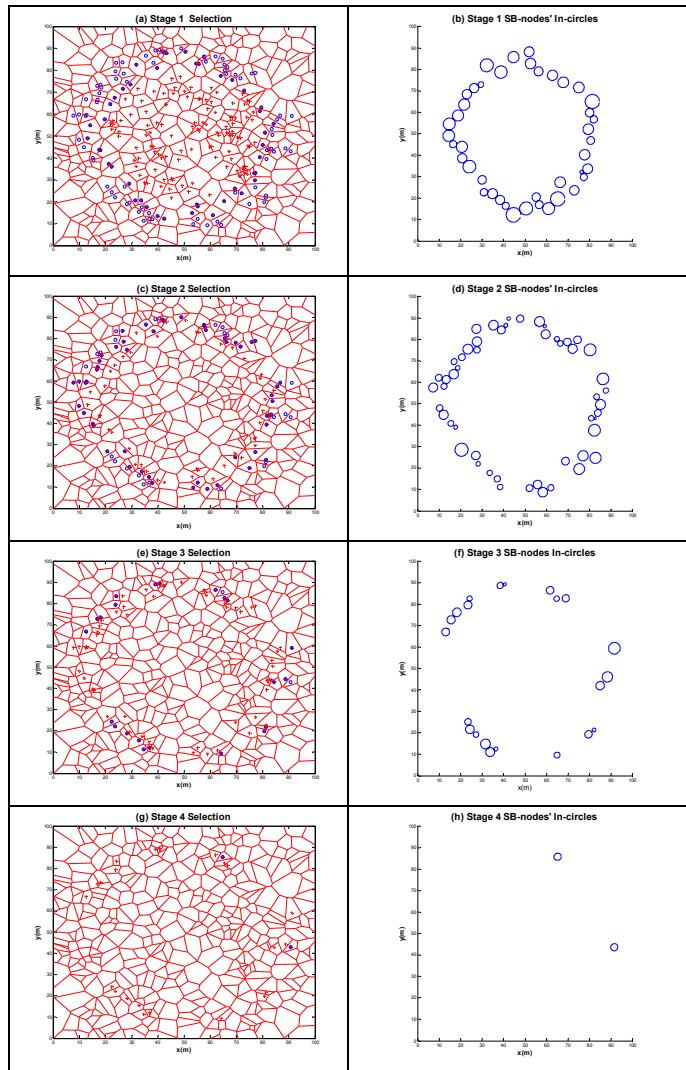


Table 1. Voronoi-based BNS-algorithms

Alg.	Alg. Type	AVD	PCov
RDL	Degree-based	70% -10%	12% -1%
ADL	Degree-based	65% -4%	5% -0%
MaxD	Distance-based	0% -4%	-5% -0%
MinD	Distance-based	71% -10%	11% -0%
DUCM	Distance-based	69% -7%	16% -1%
VarD	Distance-based	62% -5%	9% -0%
VA	Centralized	42% -7%	13% -1%
VD	Centralized	45% -8%	12% -1%
CNF	Centralized	78% -13%	30% -4%

Table 2. BNS-algorithms percentage of AVD, PCov improvement over MB (Number of SB-nodes 20-100)

#### V. SIMULATION RESULTS

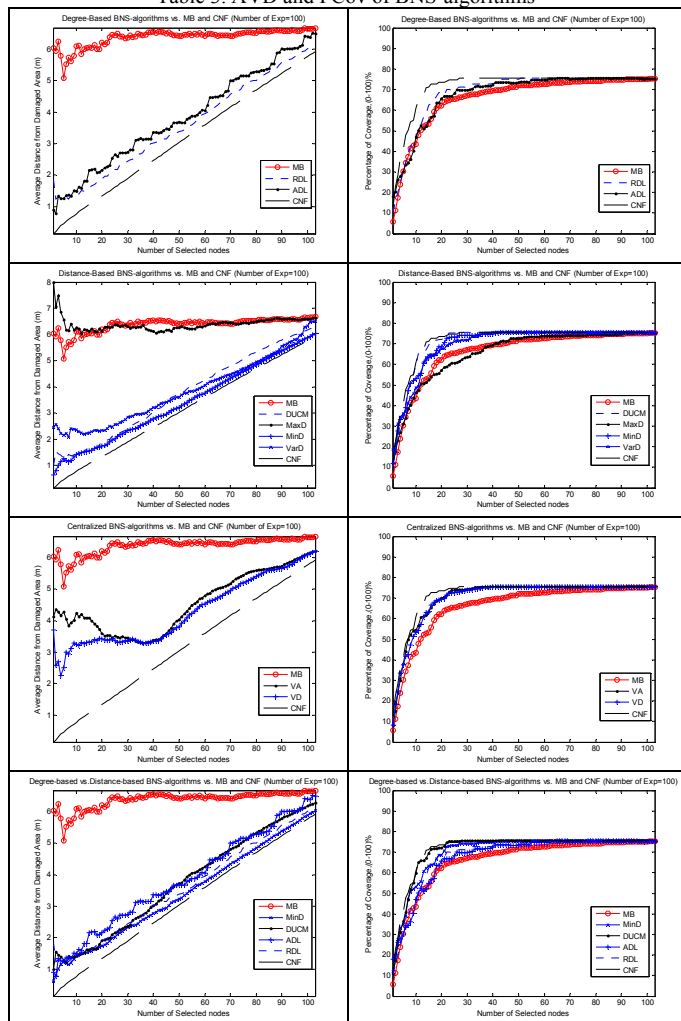
Based on the model and assumption described in section III, the experiment was repeated 100 times with uniform random node distribution for each algorithm. Since in each experiment the number of B-nodes varies, the maximum number of selections in the average is the *minimum* number of B-nodes in 100 samples ( $N_{b(exp1-100)} = \min\{N_{b(exp-1)}, N_{b(exp-2)}, \dots, N_{b(exp-100)}\}$  where  $N_{b(exp-i)}$  is number of B-nodes in the sample experiment  $i$ ). The required number of selected B-nodes (SB-nodes) is varied from 1 to  $N_{b(exp1-100)}$  to observe the AVD and PCov behavior of different algorithms as the number of SB-nodes increases. It can be seen from Table 3 that as the number of selected B-nodes approaches the total number of B-nodes  $N_{b(exp1-100)}$ , AVD and PCov of all BNS-algorithms converge.

Performance of the algorithms are compared as follows: 1) Degree-based BNS-algorithms vs. MB and CNF; 2) Distance-based BNS-algorithms vs. MB and CNF; 3) Centralized BNS-algorithms vs. MB and CNF; 4) Degree-based vs. Distance-

based BNS-algorithms; and 5) AVD and PCov of Selected Degree-based, Distance-based, and Centralized BNS-algorithms. Result with error bars (97.5% confidence intervals) are not included here due to space limit. Bounds for BNS-algorithms' percentage of AVD and PCov improvements over MB selection are shown in Table 2 as the number of selected B-nodes increases from 20 to 100.

From Table 2, it can be seen that except for MaxD, all other BNS-algorithms show improvement over MB. The efficacy of BNS-algorithms is higher when fewer B-nodes are selected, although there is a high level of fluctuation in performance. It also can be seen from Table 2 that Degree-based and Distance-based BNS-algorithms have similar performance both in terms of AVD and PCov. However, each of the BNS-algorithms is useful in different circumstances. In the case of unreliable localization and inaccurate distance measurements, degree-based BNS-algorithm still allow good performance to be achieved at the cost of somewhat poorer coverage. In general, distributed BNS-algorithms have good performance in comparison to centralized algorithms both in terms of AVD and PCov (Table 2, 3).

Table 3. AVD and PCov of BNS-algorithms



## VI. CONCLUSION AND FUTURE WORK

Distributed algorithms for automatic determination of the boundary of a damaged region of a network based on available 1-hop knowledge inferred from a node's statistical and geometrical features are presented. The efficiency of proposed distributed BNS-algorithms is compared with their centralized counterparts. As future work, 1) other 1-hop distributed BNS-algorithm can be considered. 2) Precision of selection can be increased if B-nodes have the ability to predict their neighbors' status with actual message exchange. 3) Tradeoff between frequency and scope of message exchange and level of measurement accuracy remains an interesting and challenging issue.

## VII. ACKNOWLEDGMENT

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

## REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393-422, 2002.
- [2] N. Ahmed, S.S. Kanhere, S. Jha, "The holes problem in wireless sensor networks: a survey," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, pp. 4-18, 2005.
- [3] J. E. Doran, S. Franklin, N.R. Jennings, T. J. Norman, "On Cooperation in Multi-Agent Systems," *The Knowledge Engineering Review*, vol. 12, pp. 309-314, 1997.
- [4] L. Sitanayah, A. Datta, R. Cardell-Oliver, "Heuristic algorithm for finding boundary cycles in location-free low density wireless sensor networks," *Comput. Netw.*, vol. 54, pp. 1630-1645.
- [5] B. Tong and W. Tavanapong, "On Discovering Sensing Coverage Holes in Large-Scale Sensor Networks," *Technical Report TR 06-03 TR 06-03*, 2006.
- [6] J.S. Deogun, S. Das, H.S. Hamza, S. Goddard, "An Algorithm for Boundary Discovery in Wireless Sensor Networks," *HiPC 2005*. vol. 3769, ed: Springer Berlin / Heidelberg, 2005, pp. 343-352.
- [7] M. Fayed and H. T. Mouftah, "Localised convex hulls to identify boundary nodes in sensor networks," *Int. J. Sen. Netw.*, vol. 5, pp. 112-125, 2009.
- [8] L. Jin, J. Jia, G. Chang, X. Wang, "Restoration of coverage blind spots in wireless sensor networks based on ant colony algorithm," in *GEC 2009*, Shanghai, China, 2009, pp. 847-850.
- [9] Y. Fucai, et al., "A modeling for hole problem in wireless sensor networks," in *IWCMC 2007*, Honolulu, Hawaii, USA, 2007, pp. 370-375.
- [10] G. Dini, M. Pelagatti, I.M. Savino, "An Algorithm for Reconnecting Wireless Sensor Network Partitions," in *Wireless Sensor Networks*, ed. 2008, pp. 253-267.
- [11] Z. Jiang, J. Wu, A. Agah, B. Lu, "Topology Control for Secured Coverage in Wireless Sensor Networks," in *MASS 2007,2007*, pp. 1-6.
- [12] C.-Y. Chang, W.-C. Chu, C.-Y. Lin, C.-F. Cheng, "Energy-balanced hole-movement mechanism for temporal full-coverage in mobile WSNs," in *IWCMC 2010*, Caen, France, 2010, pp. 89-93.
- [13] G. Mao, B. Fidan, B.D.O. Anderson, "Wireless sensor network localization techniques," *Computer Networks*, vol. 51, pp. 2529-2553, 2007.
- [14] A. Rafiei, "Boundary Node Selection Algorithms by Simple Geometric properties in WSNs," *AMS 2011*, 2011, pp. 196-201.
- [15] F. Aurenhammer, "Voronoi diagrams-a survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, pp. 345-405, 1991.

© 2011 IEEE. Reprinted, with permission, from Daniel Franklin, Boundary node selection algorithms in WSNs . Local Computer Networks (LCN), 2011 IEEE 36th Conference on, October 2011. This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Technology, Sydney's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to [pubs-permissions@ieee.org](mailto:pubs-permissions@ieee.org). By choosing to view this document, you agree to all provisions of the copyright laws protecting it