An Analysis of the Need for Dedicated Recovery Methods and Their Applicability in Wireless Sensor Networks Running the Routing Protocol for Low-Power and Lossy Networks

Anne-Lena Kampen
Bergen University College, Bergen
ITEM NTNU, Trondheim
Norway
Anne-Lena.Kampen@hib.no
anneleka@stud.ntnu.no

Knut Øvsthus
Bergen University College, Bergen
Norway
knut.ovsthus@hib.no

Øivind Kure
NTNU, Trondheim
Norway
okure@item.ntnu.no

***Abstract*—Wireless Sensor Networks Running (WSN) functionality depends critically on the network connectivity. The connectivity is generally determined by the node density and the nodes' transmission range. However, the applied routing protocol decides the routing path topology. A failing node may disrupt the current path topology such that dedicated recovery methods are needed to ensure a loop-free reconnection of the disconnected nodes. In this article, we estimate the probability that disconnected nodes need dedicated recovery methods in networks where the nodes are randomly located and which use RPL as routing protocol. We further calculate the success rate and overhead cost for different RPL fitted recovery protocols to better judge where the different methods should be used.**

Keywords-WSN; Recovery; Disconnection; Energy; Looping.

I. INTRODUCTION

A common requirement for Wireless Sensor Networks (WSN) is marginal need for human support during operation. Hence, the networks should be able to autonomously handle common error conditions, such as loss of connectivity due to failing nodes [1]. Connectivity loss negatively affects the data throughput and may lead to network partitioning. Thus, nodes should be reconnected without any unnecessary delay. The reconnection process should further expend limited amount of energy to minimize its influence on network longevity.

Reconnection of nodes located such that several neighboring nodes are at the same routing distance from the sink as a failing next-hop node (parent) introduces insignificant delay and energy consumption. However, nodes located such that all neighbors are at a routing distance further away from the sink than a failing parent node cannot make an immediate reconnection. The reason is that the reconnection process may create routing loops if not controlled. Routing loops are created if the disconnected nodes choose their own directly, or indirectly, connected successors as new parent nodes. Dedicated global or local recovery methods are means to ensure against the formation of routing loops during the reconnection process.

Global recovery processes generally postpone reconnection of nodes in loop-prone topologies until the next global network update. Local recovery processes make nodes in the vicinity of a disconnection communicate routing information to enable fast, loop-free reconnections. The most suitable recovery method is decided by the network characteristics and the requirements of the running application.

Our contribution is twofold and relates to recovery in randomly deployed network running Routing Protocol for Low-Power and Lossy Networks (RPL) [2], which is one of the recommended protocols for WSNs. First, we present calculations and simulations to assess the need for dedicated recovery methods to reconnect disconnected nodes. The result can be used as a base to decide whether to introduce dedicated recovery management in applied networks. We further suggest one on-demand recovery method that combines the global and local approaches. The suggested method, along with two additional local recovery methods, is analyzed to better judge where the different recovery methods should be used.

The rest of the article is organized as follows: the related work is introduced in Section 2. The probability that dedicated recovery methods are needed to mend disconnection is presented in Section 3. An analysis of two local recovery methods are presented in Sections 4 and 5. The on-demand method is presented in Section 6. The methods are compared in Section 7. Section 8 comprises the conclusion.

II. RELATED WORK

Network connectivity calculations are presented in several papers, such as [3][4][5]. Zhu et al. [6] conclude that a network with satisfying coverage is connected if the communication range is twice the coverage range. Topology controls methods to maintain k-connected networks are investigated and suggested in [7][8] and [9]. Kleinroch [10] discusses network connectivity based on cited works and presents the node degree needed to achieve network connectivity. However, the analysis performed in these papers focuses on connectivity without considering the

 ISBN: 978-1-61208-374-2

applied routing protocol or reconnection of disconnected nodes. Our analysis is based on the functionality of the applied routing protocol.

Many of the presented recovery protocols suggest movable nodes to reconnect disconnected nodes [11][12]. Nodes are proactive moved to prohibit disconnections, or reactive moved to mend disconnections. However, we assume that the nodes location is static and the goal is to discover all alternative possible recovery paths.

All recovered paths are required to match the routing protocol's path construction method and avoid formation of routing loops. Global methods fulfil this requirement by making disconnected nodes in loop-prone positions postpone reconnection until the next global update. This approach is used in RPL [2].

The disconnected node initiates the recovery process in local recovery methods. The affected node signals its state to adjacent nodes. Depending on the recovery method used, the signaling may be relayed further on to reach nodes eligible to offer new loop-free paths to the disconnected node. Sequence numbers, as used in [13][14], is a common mean applied in such local recovery methods to discover new paths. Other local recovery processes include avoiding any duplicates in the address field during source routing [15] and caching alternative feasible successors paths in case the current route is broken [16]. The feasible successor paths are guaranteed loop free as they report a distance to the destination that is shorter than the current path from the source to the destination.

III. PROBABILITY OF DEDICATED RECOVERY

This section presents an analysis of the need for dedicated recovery methods to mend routing path disconnections. The routing protocol that is used as the basis for our analysis is RPL.

A. Short presentation of RPL

RPL is a soft state routing protocol that creates routes that are directed toward the sink. The overall topology of the routing entries creates Destination Oriented Directed Acyclic Graph (DODAG). A node's logical location in the routing graph is defined by the nodes rank and selected parent, which are two strongly interrelated properties of a node. A node calculates its own rank based on its selected parent rank and the metric-based cost-of-path between itself and its parent.

To prevent against routing loops are nodes running RPL prohibited from increasing their rank in between global DODAG updates. Performing a rank increase means that nodes make a logically move away from the sink in the routing graph, an action which may results in routing loops. Global DODAG updates are initiated by the sink by distributing updated Destination Information Option messages (DIO) that flow like a wave throughout the whole network.

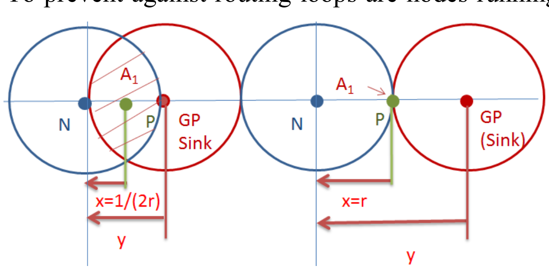

Figure 1. Node arrangement for a node at two-hop distance from the sink.

B. Presentation of extreme popints for probability calculations

To estimate the probability that a dedicated recovery method is needed is complex and depends on the relative location of all possible parents' next-hop node (grandparent). However, the highest and lowest probability limits may be calculated by studying the difference between dedicated recovery need for nodes located at the extreme points. The extreme location for the nodes is at the border of the routing graph. One of the extreme points is represented by the nodes located at a two-hop distance from the sink. The nodes next-to-the-leaf nodes are the highest rank nodes that may require dedicated recovery, and represent the second extreme point. All other nodes that may require dedicated recovery lie between these two borderline cases, so do their average probability.

1) Extreme location one : Two-hop distance node

Figure 1 illustrates a general node arrangement for nodes at two-hop distance from the sink. The blue dot labeled N and the blue circle is the node under consideration and its transmission distance, respectively. The red dot is the grandparent node, and its transmission range is defined by the red circle. The grandparent node is the sink as the node under consideration, N, is a node at two-hop distance from the sink. The parent node is represented by the green dot labeled P. All nodes choose parents that minimize their own rank. Thus, the node and its grandparent cannot communicate directly. Further, if the parent node dies, the node N needs to find a new parent node to maintain the path toward the sink.

According to the loop-avoiding rules of the RPL method, a node is never allowed to increase its rank unless a global update is performed. Hence, node N needs to maintain or improve its rank if the parent node in Figure 1 dies. The only way node N can improve its rank is to achieve a direct connection to the sink, which it cannot. Hence, node N needs to maintain its rank. It follows that it needs to get connected to a node that is directly connected to the sink, i.e., it must keep the sink as its grandparent node after recovery. Thus, the alternative new parent node must reside in the overlapping area of the transmission circle of the grandparent and the transmission circle of the node N (area A in the figure).

2) Extreme location two: Node next-to-the-leaf node

Figure 2 is used as a reference to calculate the permitted-area A for a node next-to-the-leaf node. The multiple circles centered at the sink represent the location for nodes at a specific hop distance from the sink. The area between the red dot representing the sink, and the inner red circle, represents the location for the one-hop nodes. The area

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-374-2

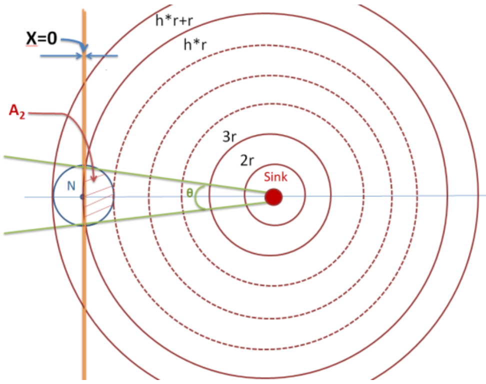

Figure 2. Disconnected node next-to-the-leaf node.

between the inner red circle and the second inner red circle represent the permitted-area for the two-hop nodes and so forth. As we are considering a node next-to-the-leaf node, we assume that the distance between the sink and the node under consideration is so far apart that the curvature of the sink's h-hop circle line cutting through the node N's circle approaches a straight line.

Figure 3 is a segment of Figure 2. The orange vertical lines in Figure 3 illustrate the sink's outer h-hop circle lines as straight lines based on the explanation above. The red shaded area named 'A' illustrate the permitted-area for a parent node of node N.

To find the probability that dedicated recovery is needed we derive the expression for the expected value of the probability that there exists more than one node inside the area A. If there is more than one node in are A, it means that there exists a recovery node after the current parent node dies. To find the wanted expectation we need an expression for the probability that there is another node in A, as well as an expression of the probability density function for location of node N.

C. Probability that there is a recovery node in area A

We assume a uniform node distribution, thus, the number of nodes in an area is given by the Poisson distribution. λ is defined to be the node density, which corresponds to the expected number of nodes in a circular shaped area with radius equal to the transmission range. All nodes have equal transmission range, r.

The probability that there is another node in A in Figures 1 and 2 is given as Prob(more than 1 node in area A | given

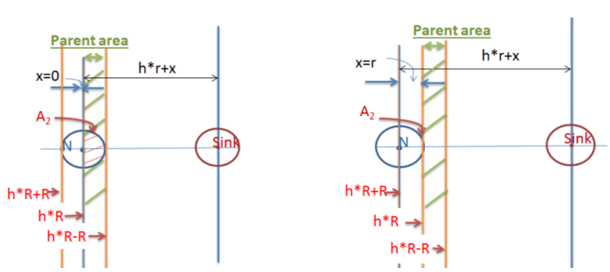

Figure 3. Node arrangement for a node far from the sink.

that there is at least 1 node in area A). Prob(1 or more ∩ at least 1)/ prob(at least 1) = Prob (2 or more)/Prob (at least 1):

$$p(more\ than\ one\ \ in\ A|least\ one\ in\ A) =$$

$$\frac{1 - P(1\ node\ in\ area\ A) - P(0\ nodes\ in\ area\ A)}{1 - P(0\ nodes\ in\ area\ A)} =$$

$$1 - \frac{\lambda A(x)}{e^{\lambda A(x)} - 1} \tag{1}$$

We assume that node N is at a distance y from a grandparent node, as shown in Figure 1 for the two-hop node. y is in the range r, 2r. The area of A_1 is symmetric around y/2=x. Then the area A_1 as a function of x is:

$$A_1(x) = 2r^2\left(cos^{-1}\left(\frac{x}{r}\right) - \sqrt{1 - \left(\frac{x}{r}\right)^2} * \frac{x}{r}\right) \tag{2}$$

According to the node next-to-the-leaf node, the permitted-area is one half of the area A_1 in (2), using y=x.

Notice that the area A_2 in Figure 3 is bigger than are A_1 in Figure 1 when the node N is in its closest position to the sink (left hand side of the figures). Hence, with node N in this position, the permitted-area for the recovering node next-to-the-leaf node is bigger than the permitted-area for the two-hop recovering node.

D. Node next-to-the-leaf-node calculations

We use Figure 2 as reference to calculate the probability density function for the node N location. The sector θ in Figure 2 defines the sector where a recovery parent node N may be located. The cumulative distribution function for the node N's location and the probability density function of node N's location are respectively given by:

$$F_1(y) = \frac{\frac{\pi((h*r)+y)^2}{\theta} - \frac{\pi(h*r)^2}{\theta}}{\frac{\pi(h*r+r)^2}{\theta} - \frac{\pi(h*r)^2}{\theta}} = \frac{y(2hr+y)}{(1+2h)r^2} \tag{3}$$

$$f_1(y) = \frac{2(hr+y)}{(1+2h)r^2} \tag{4}$$

Using the presented equations, we can derive the expression for the expected value of the probability that there exists a recovery node in area A_2 for the node next-to-the-leaf node. The expression is found by combining (1) with (4) and (2). The expected value of P(there exist a recovery node inside area A_2) is:

$$E[P(y)] =$$

$$\int_r^{2r} \{(P(more\ than\ 1\ inA_2(y)|least\ 1\ inA_2(y))f(y)\}dy$$

$$= \int_r^{2r} \left\{\left(1 - \frac{\lambda A_2(y)}{e^{\lambda A_2(y)} - 1}\right) * \frac{2(hr+y)}{(1+2h)r^2}\right\} dy \tag{5}$$

E. Two-hop node calculations

We will now derive the expectation of the probability that there exists a recovery node in area A_1 for the two-hop node, ref. Figure 1. First we need the expression for the probability density function for the location of node N. This is found using Figure 4. According to Figure 4 are the expression for the cumulative distribution and probability density function of the node N given by:

$$F_2(y) = \frac{\frac{\pi(y)^2}{\theta} - \frac{\pi r^2}{\theta}}{\frac{\pi(2r)^2}{\theta} - \frac{\pi r^2}{\theta}} = \frac{y^2 - r^2}{3r^2} \qquad (6)$$

$$f_2(y) = \frac{2y}{3r^2} \qquad (7)$$

Combining (1), (2) and (7), gives the following expected value of the probability for the two-hop node:

$$E[P(y)] = \int_r^{2r} \left\{ \left(1 - \frac{\lambda A_1(y)}{e^{\lambda A_1(y)} - 1}\right) \left(\frac{2y}{3r^2}\right) \right\} dy \qquad (8)$$

To summarize; the expression for the expected value of the probability that there exist recovery nodes for a node next-to-the-leaf node given by (5), and the corresponding expression for a two-hop node given by (8). In other words, this is the probability that dedicated recovery methods are *not* needed to mend routing path disconnections. The calculations performed are based on numerically calculations of the equations.

F. Simulations - dedicated recovery

Simulations are conducted in Java to validate the calculated results for the expected value of the probability that dedicated recovery methods are not needed to mend routing path disconnections.

The simulation for the two-hop node is initialized by placing a node N in a fixed position. The next node is randomly paced in the donut shaped area between r and 2*r form the fixed node. The second node becomes the fixed node's grandparent (the sink). A varying number of nodes are subsequently randomly distributed with average density λ inside the simulation area.

The sought probability for the two-hop node is estimated

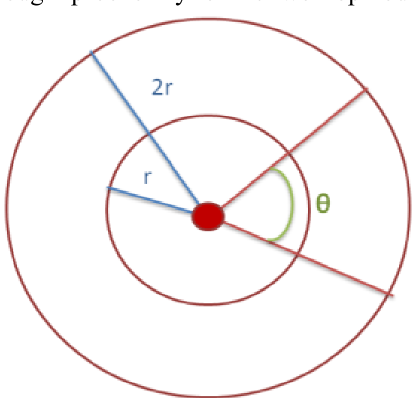

Figure 4. The sector where a possible node N may be located.

based on the percentage of the simulation runs resulting in two or more nodes located inside the overlapping area defined by the node's transmission range and the grandparent's transmission range. Dedicated recovery is needed if the number of nodes in the overlapping area is less than two. The reason is that if there is only one node in the overlapping area, it is definitely the parent node and there are no nodes left in the area when it dies. Simulation runs resulting in zero nodes inside the area is discarded. The simulated result is averaged over 1000 runs for each node density.

According to the node next-to-the-leaf node the simulations is performed by placing two nodes at a distance h*r + x apart, in the same manner as for the two-hop node. 0<x<r. The two nodes represent the node under consideration, N, and the sink. The number of nodes located both inside node N's transmission range and inside a radius of h*r from the sink are counted. The investigated probability is further performed following the same procedure as when calculating the two-hop node probability.

G. Results – dedicated recovery

In this section, we present and discuss the simulated and calculated results of the expected value of the probability that dedicated recovery methods are *not* needed to mend routing paths. The curves in Figure 5 show the expected value of the probability for the extreme points, i.e., the lowest and highest average probability values. The dashed red curve represents the simulated probability of the nodes at two-hop distance from the sink, and the blue curve shows the calculated probability for the two-hop node. The red and the dashed green curve show respectively the calculated and simulated results for the node next-to-the-leaf node. The curvature of the simulation results conform to the curvature of the calculated results validating each other.

The curves in Figure 5 show that the disconnected nodes next-to-the-leaf nodes have lower need for dedicated recovery than the two-hop nodes.

The difference between the curves in Figure 5 is caused by the unequal characteristics of the two extreme points in the routing graph topology. Both the probability density function for the node N's location and the permitted-area are different in the two extreme points. The probability density function for the location of a leaf node N approaches a uniform distribution as the number of hop gets high. This is illustrated in Figure 6. The reason for the uniform distribution is the straightening of the h*r curvature and the related small difference between the circumference of the h*r and h*r+r circle when h is high. This is easily seen looking at Figure 3. On the contrary, the probability density for the location of the two-hop node N is increasing toward the outer circumference. The reason is the increased circumference which increases available deployment area for the node N. This can be observed in Figure 4. Figure 7 shows how the permitted-area varies with distance between the node N and the sink node. The figure shows that the area decreases with increased distance, and also illustrates the slightly bigger permitted-area of the node next-to-the-leaf nodes. Combining the information given in Figures 6 and 7,

 ISBN: 978-1-61208-374-2

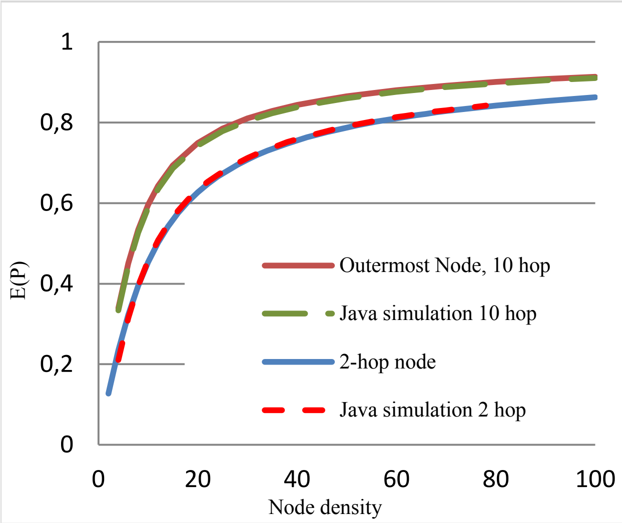

Figure 5. The expected value of the probability that dedicated recovery methods are not needed to mend routing path disconnections.

shows that the location probability of the two-hop node favors the smallest permitted-area size, while the node next-to-the-leaf node gives equal priority to all permitted-area sizes. A smaller area means that the probability that it contain more than one node is lower. Thus, the probability that it contains a candidate recovery parent node is lower.

As expected, and illustrated by the graphs in Figure 5, is the need for a dedicated recovery method decreasing with increased node density. The reason is simply that the probability that more than one node is located inside a defined area increases with node density. However, the probability never reaches 1 although the node density gets high. The reason is due to the explanation given related to Figures 6 and 7: the permitted-area for the recovery nodes is very small for some of the locations of node N. Hence, there is always a probability that some nodes do not have available recovery nodes.

The graphs in Figure 5 show that if all the nodes in a network are required to stay connected some kind of special repair method is needed. According to Takagi and Kleinrock [10], eight is the magic number of neighbors regarding network throughput, and four neighbors are needed to maintain a connected network. Hence, we may assume that an average well design network has a node density between 8 and 20. We define the node density as the number of nodes inside a circular area with radius equal to the transmission range, thus is equals the number of neighbors plus one. The graphs show that the probability that dedicated recovery is not needed is between 40% and 75% at node densities between 8 and 20. Hence, between 25% and 60% of the disconnected nodes needs a dedicated recovery method to get properly reconnected.

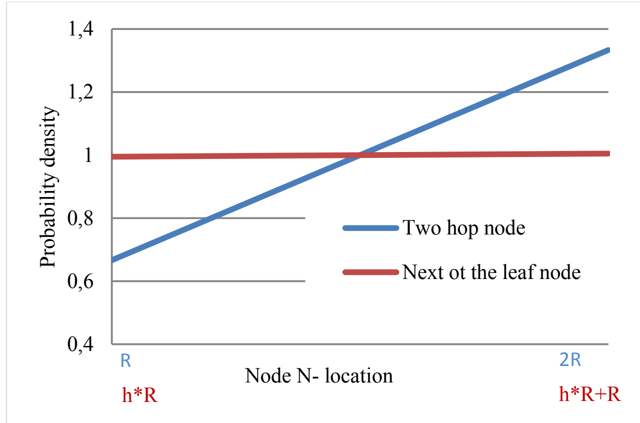

Figure 6. Probability density function of node N's location.

IV. Analysis of Guo et al.'s Method

This section presents an analysis of a local recovery method suggested by Guo et al. [17]. The method forces intermediate nodes on potential recovery paths to adjust their rank to make the path feasible for a disconnected node.

Their method [17] is activated and runs as follows. A poisoned node, which is a node that needs dedicated recovery to reconnect, initiates the recovery process by broadcasting a request. The request is further relayed to the receivers' parent nodes. The process lasts until the requests reach a node with better rank than the requesting node. Receivers with better rank than the requesting node generate a reply and forward it toward the requesting node using the same path as the associated request. The nodes along the path adjust their rank such that a new, valid path for the requesting node is made.

However, the method is not able to find a new valid path for all kind of topologies. The reason is twofold. Requests received form a parent node are silently discarded. Hence, paths pointing through child nodes are never found. In addition, a race condition occurs when siblings of a dying parent node simultaneous enter poisoning state. Nodes with pending requests silently discard received requests. The result may be that paths pointing toward sibling nodes remain undiscovered.

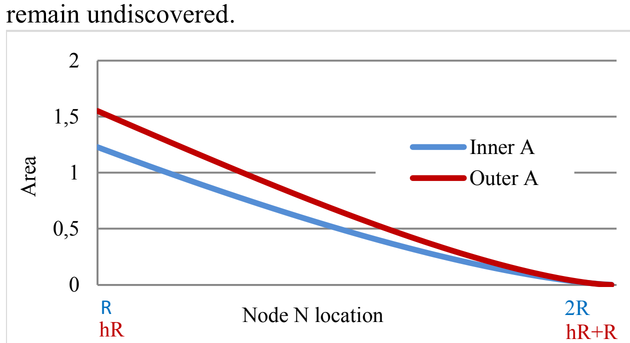

Figure 7. Permitted-area for recovery node.

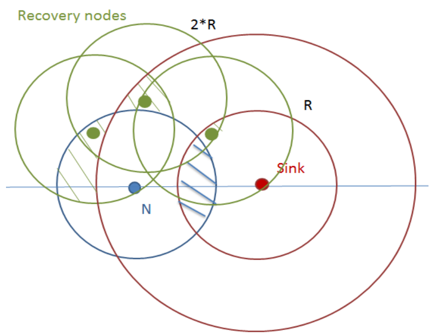

Figure 8. Recovery path goes through higher level node.

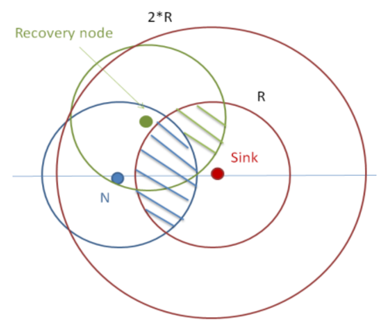

Figure 9. Recovery path goes through sibling node.

A. *Simulation of Guo et al.'s method*

Based on the layout in Figures 8 and 9, we simulate the probability that a poisoned node running Guo et al.'s method discovers a new valid path. The red dot is the sink node. The area between the sink and the inner red circle represents the localization of one-hop (rank one) nodes, the two-hop (rank two) nodes are located between the red circles, and so forth. The blue node N represents a node that is poisoned and need dedicated recovery to get reconnected to the DODAG. The blue circle represent node N's transmission range. The figures illustrate the two scenarios that make the [17] method local recovery succeed.

Figure 9 illustrates the scenario that a reachable node with equal rank as the poisoned node, has a parent node outside the poisoned node's transmission range. Expressed according to the figure, it means that there exists an equal-rank green node in the overlapping area made of the blue circle and the donut shaped area made of the red circles. Further, this equal-rank-node has a parent inside the green shaded area.

The other scenario that makes [17] succeed is if a lower-level node of N (node lying outside the outer red circle) have a path toward the sink that does not include N, or N's parent. According to Figure 8, it means that the node N has a neighbor in the leftmost green shaded area. This neighbor has a parent in the upper green shaded area, which further has a parent in the rightmost green shaded are.

The simulation is implemented in Java. A varying number of nodes are randomly deployed inside a circle shaped area with radius that is three times the transmission range. All the nodes are supposed to have equal transmission range. 5000 runs with different node densities, and node N locations, are performed. The numbers of runs which satisfy one or both of the scenarios discussed above, and indicated in Figures 8 and 9, are counted. This number is normalized by the number of runs where recovery is needed to reconnect N, i.e., the number of runs where only one node reside inside the blue shaded area.

B. *Results on Guo et al.'s method*

Figure 10 shows the simulated probability that a poisoned node gets reconnected after performing Guo et al.'s local recovery procedure. The x-axis shows the node density, λ. As expected is the success probability increasing rapidly with node density. When the node density is 4 the probability is about 40%. When the node density approaches 20, the probability approaches 100%. Thus, the approach of [17] works best in high density networks. The probability that a dedicated recovery method is needed to reconnect disconnected nodes is highest at low node densities; Figure 5. Hence, the lowest probability of solving the problem is in the scenarios where the problem is most likely to occur.

V. ANALYSIS OF THE ACK LOCAL RECOVERY METHOD

This section presents a local recovery method that is based on reliable poisoning of successors (sub-DAG) nodes. We call this method the ACK-method. Reliability is achieved by letting nodes be aware of their children, and make all children acknowledge reception of poisoned information transmitted by the poisoned node. Information about children is achieved by making all nodes inform about their parents in regular transmitted DIO messages.

Receiving ACK form all children enables the poisoned node to increase rank to reconnect to the DODAG. No loop is created because sub-DAG nodes with no alternative recovery parent inform about their poisoned state in the transmitted ACK messages.

The ACK method will mend disconnections as long as the poisoned node receives ACK messages from all its children. Hence, the probability of success using this method depends on probability of successful reception of transmitted packets. We name the probability of successful transmission P_{rec}. A poisoned message is retransmitted once if the

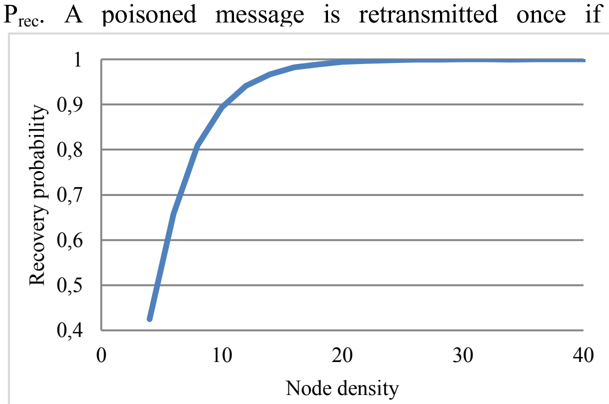

Figure 10. Probability that Guo et al.'s method succeed

poisoned node does not receive ACK from all children. Thus, recovery will not succeed if the two poisoned messages are lost, or if the ACK message is lost.

Thus, assuming one child gives the following success probability of the ACK method: P(ACK succeed one child)=p(First Poisoning messages succeed)*P(ACK succeed)+p(First Poisoning messages do not succeed) *p(Second Poisoning messages succeed)*P(ACK succeed):

$$P(ACK\ succeed\ one\ child) =$$

$$P_{rec} * P_{rec} + [1 - P_{rec}]P_{rec}P_{rec} \quad (9)$$

Assuming that either all or none of the child nodes receive the poisoning message, the expression for the ACK success probability for c child becomes:

P(ACK succeed for c child)= p(First Poisoning messages succeed)*P(ACK succeed)^c+p(First Poisoning messages do not succeed) *p(Second Poisoning messages succeed)*P(ACK succeed)^c:

$$P(ACK\ succeed\ c\ child) =$$

$$P_{rec} * P_{rec}{}^{c} + [1 - P_{rec}]P_{rec}P_{rec}{}^{c} \quad (10)$$

A. *Results ACK method*

The probability that the ACK method succeeds is shown in Figure 11. The x-axis represents the probability that a transmitted packet is received. The blue graph illustrates the success probability for a disconnected node with one child, and the red graph shows the success probability for a disconnected node with five child nodes. As expected is the success probability increasing with increased probability of receiving transmitted messages and with reduced number of child nodes.

VI. ON-DEMAND METHOD

In this section, we present our proposed combination of local and global recovery that may be used to guarantee recovery for all node densities while keeping the network energy consumption as low as possible. We call this method the on-demand method.

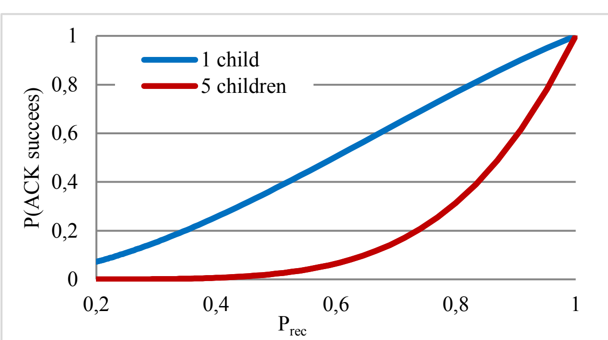

Figure 11. Probability that ACK succeed vs. Prec.

The method functions as follows. A node entering the poisoning state broadcasts an increase-sequence-number request. The request is broadcasted throughout the whole network, which means that it will eventually reach the sink if there exists a path between the poisoned node and the sink. When the sink receives the request it initiates the global recovery algorithm.

The message overhead, hence the energy cost, of running one iteration of the on-demand method is about twice the cost of running one iteration of periodic global update. The reason is that the request is broadcasted throughout the whole network in a manner similar to DIO message during global network update.

Running the update only when nodes are poisoned means that the network wide broadcast is run only when needed, and no periodic global network update is in fact ever needed.

The recovery time using the on-demand method decrease compared to the periodic approach, as the global update is run immediately after the request reach the sink.

VII. ANALYSIS OF RECOVERY OVERHEAD COST

We perform calculations to estimate the overhead cost difference between Guo et al.'s, ACK, on-demand, and periodic recovery methods. The overhead is calculated as the total number of transmitted and received management messages during the recovery process. The overhead cost is proportional to the network energy consumed, which should be as small as possible to limit the recovery process' impact on the network lifetime.

The Guo et al.'s method overhead relates to the transmission of requests and replies. We assume a uniformly distributed network where the average number of neighbors is n. The fraction of neighbors forwarding the request is α, and the fraction of neighbors that replies the request is β. Thus, the number of nodes transmitting the request is (1+ αn). The digit 1 in the expression refers to the poisoning node initiating the request transmission. Each transmitted request is, on the average, received by n neighboring nodes. Hence, the total request cost is (1+ αn)*n. Further, we assume that the request and replies are relayed once. The reply is answered by βn nodes and relayed once by βn nodes. Each transmission is received by n nodes. Hence, the reply cost is (2βn)*n, and the total overhead cost becomes:

$$Guo\ et\ al.'s_{overhead} = 2 * \beta n^2 + (1 + \alpha n) * n \quad (11)$$

The ACK overhead cost relates to the poisoning message and ACK message transmission. The poisoned node transmits a poisoning message, which is received by all neighboring node giving a total cost of 1+n. Further, we assume that the fraction of neighbors that are child of the poisoning node is Δ. Hence, the ACK messages is transmitted by Δn nodes and all messages are received by n nodes, giving a cost of Δn*n. In addition, all nodes in the neighborhood transmit a DIO concluding the recovery process. The DIO is received by all neighbors giving a cost of n*n. Hence, the total cost of the ACK recovery process is:

$$ACK_{overhead} = (1 + \Delta) * n^2 + n + 1 \qquad (12)$$

The on-demand overhead relates to the total number of nodes in the network, N, transmitting requests and DIO messages. All transmitted messages are received by the average number of neighbors.

$$(On - Demand)_{overhead} = 2 * Nn \qquad (13)$$

The overhead according to one run of global recovery relates to all nodes N transmitting DIO messages, which are received by the average number of neighbors.

$$OneRunGlobalRecovery_{overhead} = N * n \qquad (14)$$

A. Results comparing methods

Figure 12 shows the overhead for the different methods using Δ=0.4 (share of neighbors being child of poisoned node), α=0.6 (share of neighbors relaying request) and β=0.2 (share of neighbors replying request). The value of Δ is chosen looking at the right-hand side of Figure 1: We assume that approximately all nodes located inside an area about the same size as the red shaded area are children of a node N. The rest of the neighboring nodes relay the requests received, hence the value of α = 1 - Δ. β is chosen assuming that only a small fraction of nodes receiving the relayed request are able to answer. These values clearly changes according to the network topology. However, the mutual relation between the parameters will generally remain unchanged. Hence, the information given by the figure is valuable. The total number of nodes in the network is 100.

There is a big difference between the local recovery approach methods' overhead and the periodic update, as shown in Figure 12. However, as the local recovery methods cannot guarantee reconnection they require periodic global update to coexist to guarantee full network connectivity.

The figure shows a substantial overhead cost difference between on-demand method and periodic update. However, the great advantage of using the on-demand method is that the method is only trigged by a disconnection. Thus, the overhead cost will be lower than the periodic update method in network with low disconnection probability.

The significance of our findings is the statistical analysis of the need for dedicated recovery presented in Section 3, in addition to the overhead cost for the recovery methods presented in this Section. The statistical analysis showed that dedicated recovery is needed especially in low density networks. In addition, nodes in the vicinity of the sink are most vulnerable and require dedicated recovery. These nodes are critical for sustaining network connectivity. Our overhead cost findings show that cost analysis should be performed as part of real networks' deployment methodology to select an appropriate recovery method. The selected method should either be the on-demand method, or adapting of the periodic global update frequency, as these methods are reliable.

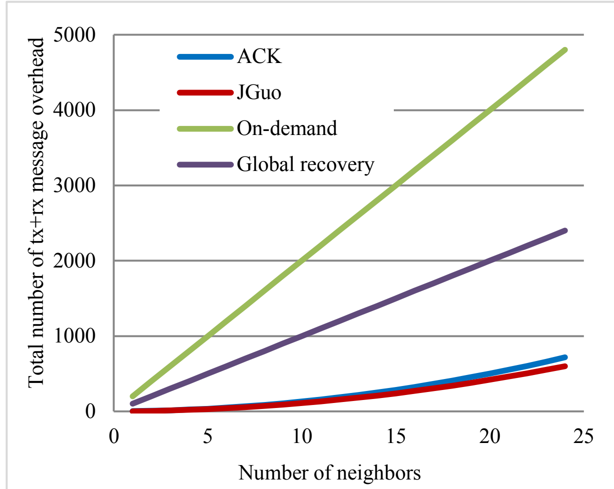

Figure 12. TX+RX message overhead. Number of nodes is 100.

Our results can be used to perform overhead cost calculations for network design. The overhead associated with the on-demand method is calculated combining (13) with the expected number of nodes that need dedicated recovery during a time span. The expected number of nodes is found combining information about the total number of nodes, the nodes failure probabilities, and the probability for disconnected nodes' recovery need, presented in Figure 5.

The overhead cost of adapting the global update frequency is calculated combining (14) with the recovery delay requirement. The delay requirement decides the network update frequency.

VIII. CONCLUSION

Disconnections in WSNs need to be resolved to sustain total network availability and avoid destructive data loss. Whether disconnections needs dedicated methods to regain connectivity depends on the topology in the vicinity of the disconnection.

In this article, we calculated and simulated the probability that dedicated recovery methods are needed to reconnect disconnected nodes in randomly deployed networks. The findings are that dedicated methods are needed in 25% to 60% of the cases when a node is disconnected. These findings demonstrate the significance of including dedicated recovery methods as a part of the network management in real scenarios where the network's availability is crucial. If dedicated recovery is not included, the periodic global update frequency should be adjusted according to the networks' recovery delay requirements. In addition, the findings demonstrate that increased node density may be used as deployment methodology to improve connectivity stability in critical areas of a network.

The findings further show that disconnected nodes close to the sink most often need dedicated recovery. These nodes are critical to sustain network connectivity. Hence, it may be wise to adapt the recovery method according to routing graph location.

The failure frequency increases with network size assuming equal failure probability for the nodes. Hence, the periodic update frequency has to increase with network size in network without any dedicated recovery method. However, increased update frequency increases the nodes' energy consumption causing reduced network lifetime.

Introducing a dedicated recovery method may reduce the load caused by periodically updates. In this article, we calculate the overhead and success rate for two local recovery methods, and one suggested global on-demand recovery method. The two local recovery methods have lowest overhead, but they cannot guarantee reconnection success. The global on-demand method is reliable as reconnections are established if possible. But, it has high overhead. However, using an unreliable recovery method that cannot guarantee connectivity requires a simultaneous periodic global update mechanism to assure the total network connectivity, while a reliable recovery method makes periodic global updates superfluous.

Thus, networks requiring reliable network connectivity should either include an on-demand recovery method, or adjust the global network update frequency. The on-demand method may greatly reduce the long-term network energy consumption. Overhead cost analysis presented in this article may be used for real scenarios to choose between the two methods.

REFERENCES

[1] V. C. Gungor and G. P. Hancke, "Industrial Wireless Sensor Networks: Challenges,," IEEE Transactions On Industrial Electronics, Oct. 2009, pp. 4258-4265.

[2] T. Winter, et. al., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, Mar. 2012.

[3] W. Jia, Y. Fu, and J. Wang, "Analysis of Connectivity for Sensor Networks Using Geometrical Probability," Springer, Lecture Notes in Computer Science, Aug. 2004, pp. 601-611.

[4] Y. Wang, B. M. Kelly, and X. Li, "On the Network Connectivity of Wireless Sensor Networks following a Random and Non-uniform Distribution," IEEE, Sixth International Workshop on Selected Topics in Mobile and Wireless Computing, Oct. 2013, pp. 69-74.

[5] X. Xingt, G. Wangt, J. Wu, and J. Li, "Square Region-Based Coverage and Connectivity Probability Model in Wireless Sensor Networks," IEEE, International Conference on Collaborate Computing, Nov. 2009, pp. 1-8.

[6] C. Zhu, C. Zheng, L. Shu, and G. Han, "A Survey on Coverage and Connectivity Issues in Wireless Sensor Networks," ELSEVIER, Journal of Network and Computer Applications, Mar. 2012, pp. 619-632.

[7] I. Saha, et al., "Distributed Fault-Tolerant Topology Control in Static and Mobile Wireless Sensor Networks", IEEE, 2nd International Conference on Communication System Software and Middelware, Jan. 2007, pp. 1-8.

[8] N. Li and J. Hou, "FLSS: A Fault Tolerant Topology Control Algorithm for Wireless Networks", ACM, Proceedings of the 10th annual international conference on Mobile computing and networking,MobiCom, 2004, pp. 275-286.

[9] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks using Transmit Power Adjustment," IEEE, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, Mar. 2000, pp. 404-413.

[10] H. Takagi and L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals," IEEE Transactions on communications, vol. 32, no. 3, Mar. 1984, pp. 246-257.

[11] K. Vaidya and M. Younis, "Efficient Failure Recovery in Wireless Sensor Networks through Active Spare Designation," IEEE, International Conference on Distributed Computing in Sensor Systems Workshops, June. 2010, pp. 1-6.

[12] M. F. Younis, S. Lee, and A. A. Abbasi, "A Localized Algorithm for Restoring Internode Connectivity in Networks of Moveable Sensors," IEEE, Transactions On Computers, Dec. 2010, pp. 1669-1682.

[13] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July. 2003.

[14] J. Garcia-Luna-Aceves and H. Rangarajan, "A New Framework for Loop-Free On-Demand Routing Using Destination Sequence Numbers," IEEE, International Conference on Mobile Ad-hoc and Sensor Systems, Oct. 2004, pp. 426 - 435.

[15] D. Jhonson, U. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," RFC 4728, Feb. 2007.

[16] J. Raju and J. J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-Free Multipath Routing,", IEEE Eight International Conference on Computer Communication and Networks, Oct. 1999, pp. 522-527.

[17] J. Guo, C. Han, P. Orlik, and J. Zhang, "Loop-Free Routing in Low-Power and Lossy Networks," IARIA, International Conference on Sensor Technologies and Applications, Aug. 2012, pp. 59-66.

 ISBN: 978-1-61208-374-2