Auction-Based Node Selection of Optimal and Concurrent Responses for a Risk-Aware Robotic Sensor Network

Jamieson McCausland¹, Rami Abielmona¹,², Rafael Falcon², Ana-Maria Cretu³ and Emil Petriu¹

¹ School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada  
² Research & Engineering, Larus Technologies Corporation, Ottawa, Canada  
³Informatique et ingénierie, Université du Québec en Outaouais, Gatineau, Canada  
E-mail: jmcca080@uottawa.ca; rami.abielmona@larus.com; rafael.falcon@larus.com; petriu@eecs.uottawa.ca

Abstract—In this paper, an auction-based node selection technique is considered for a risk-aware Robotic Sensor Network (RSN) applied to Critical Infrastructure Protection (CIP). The goal of this risk-aware RSN is to maintain a secure perimeter around the CIP, which is best maintained by detecting high-risk network events and mitigate them through a response involving the most suitable robotic nodes. These robotic nodes can operate without the use of a centralized system and select amongst themselves the nodes with the best fitness to risk mitigation plan. The robot node that is first aware of a high-risk event becomes an auctioneer. The risk mitigation task is advertised to the entire network. Each robotic node is responsible for calculating their bid metric (i.e. availability metric) for the risk mitigation task. We employ fuzzy logic in the process of the bid calculation, which incorporates the battery level, distance to the event, and redundant coverage to produce an appropriate bid value. The auctioneer only considers the top bidders. The nature of this system is to permit simultaneous mitigation plans to execute on a single RSN by effectively segmenting the network into discrete autonomous groups. Each autonomous group will utilize an evolutionary multi-objective algorithm – the Non-Dominated Sorting Genetic Algorithm (NSGA-II) – to optimize the segment's topology to mitigate the risk. A chromosome length is determined by the number of bids received, but the NSGA-II explored to separate solution spaces to achieve optimal Pareto results. The NSGA-II will seek optimal node positions and determine the optimal set of robotic nodes to utilize of the bids received. The NSGA-II will produce a set of optimized responses for each network segment for a security operator to pick the most suitable response.

Keywords—critical infrastructure protection; genetic algorithm; robotic sensor network; fuzzy logic; self-organization

I. INTRODUCTION

The distributed sensing of an environment by a sensor network for Critical Infrastructure Protection is not a new concept. Typically Wireless Sensor Networks (WSNs) play a large role in this task as they are designed to be spatially distributed to collect data wirelessly over a region of interest. This works well when the WSN deployment topology is optimized, but if the environment changes or any sensor nodes fail, the security of the CIP can be severely compromised. A better solution is required.

Robotic Sensor Networks (RSNs) carry many benefits over the popular WSNs and consist of stationary wireless nodes. The application of this type of sensor network has previously been pursued by many including [1] and [2] for CIP. The task of CIP typically involves appropriate situational assessment, risk detection, mitigation and prevention. The effective safe-guarding of critical infrastructure typically conforms to the following life-cycle:

1. Observe the environment
2. Optimize deployment to optimally prevent/detect risky events
3. Detect risky events
4. Mitigate risky events
5. Repeat

The framework proposed in [1] implements the above life-cycle, but is restricted to a single risky event and a single network response. The risk mitigation processes of multiple high-risk network events are blocked by one-another. That is, a network response must complete before another can begin. This limitation can be detrimental to a deployed RSN that is highly prone to concurrent risky events. The goal of this paper is to derive a technique to create transient independent regions of the network. These independent regions are autonomous bodies, which can execute an independent optimization algorithm to mitigate the present risk.

A fuzzy auction-based node selection approach is considered to allocate robotic sensor nodes to disjointed optimization groups. As a result, several topology optimizations can occur simultaneously. Multiple detected high-risk events can be mitigated through concurrent network responses. Auction sessions are hosted by the node detecting the high-risk event who will accept a bid from all suitable sensor node candidates. Each individual sensor node will calculate its own bid value using a fuzzy rule-base and then decide whether it should bid on the optimization task. The result is a de-centralized system in which sensor nodes can work in a co-operative manner to mitigate the inherent risk. To the best of our knowledge the use of a fuzzy auction-based task allocation system on a risk-aware RSN for CIP has not yet been explored in the literature.
The remainder of this paper has been structured as follows. Section II will discuss any previous related works. Section III will discuss the risk-aware auction protocol. Section IV will cover the self-organization of a RSN segment. Experimental results are disclosed in Section V while the conclusion finalizes the paper in Section VI.

II. RELATED WORK

Mezei et al [3] are interested in robot-robot collaboration to meet a set of mutually benefitting objectives. The objectives are to minimize communication costs, response time, and the cost to performing tasks. A key point in the research of [3] is that the task allocation process through the use of auction aggregation protocols can be conducted in a de-centralized manner. Several auction protocols – k-SAP, SAAP, k-SAAP, k-AAP and RFT – are discussed and their various benefits compared to Simple Auction Protocol (SAP). The work of [3] focusses on a single robot for a single task and only considers a cost metric as a function of distance from the event.

Gerkey et al [4] developed an auction-based task allocation system named MURDOCH. This auction protocol is a publish/subscribe communication model. Here robot-robot collaboration is the primary objective, allowing for an environment of robotic agents to assign work in a cooperative manner. This protocol allows for a heterogeneous robotic population. An auction session messages can be broadcasted on a particular namespace that only robotic agents - equipped for the task - can place a bid. The work in [4] considers only a single winner and monitors the progress of the winner in real-time.

This paper aims to extend the work of [2] where a self-organizing RSN is proposed for CIP, but limited to handling a single high-risk event. In [2] a risk-aware RSN is proposed, using an extended version of the Risk Management Framework of [6]. The NSGA-II is utilized to produce new node locations to mitigate the risk induced by nodes in distress. We hope to extend the novelty [2] to large robotic sensor network and more risky environment. To the best of our knowledge, this has not yet been explored in previous literature.

III. AUCTION-BASED RSN SEGMENTATION

In this section, an auction-based approach to segmenting a risk-aware RSN from [2] into multiple independent optimization groups as risks are discovered in real-time. As already seen, auction protocols have been examined by [3] as an innovative means of robot-robot coordination without the requirement of a centralized management system. As mentioned in Sect. II, the work by [4] introduced an auction protocol named MURDOCH, a publish/subscribe communication model, which will form the basis of our segmentation process. In this paper, robotic nodes are responsible for: detecting risky events, instantiating auction sessions (as an auctioneer), and participating in an optimization session (as a participant). High-risk event detections are fed by the work of [2] and [6]. Raw instrument data is analyzed for risk features to detect high-risk network events. A high-risk network event will be classified as either a robotic node failure or intrusion. Both of these risky event types necessitate a response from the RSN, but each response with suitable objectives fitting for the event type. For instance, a failing node will require the network to fill the coverage gap with other robotic nodes. Alternatively, intrusion detection requires increased sampling of a risky region. By optimizing the topology of the RSN, an increase in the sensor node density in the high-risk region can be achieved. In the context of CIP is highly conceivable that an RSN may be subject to simultaneous high-risk events. Multiple high-events may take the form of:

- multiple node failures;
- node failure(s) and intrusion(s); and
- multiple intrusions

A robotic node may internally detect a problem (e.g., very low battery) and (if applicable) generate a robotic node failure type high-risk event. Nodes can also use external sensor to detect objects in close proximity and generate an intrusion type high-risk event, if necessary. Upon the presence of a high-risk event within the node’s internal risk framework [2], the auction protocol will assert and the segmentation process will begin. In this process the auction protocol is separated into four main processes based on [4].

A. Risk Mitigation Task Announcement

The node discovering the high-risk event assumes the role of auctioneer (from the role of available). Let all possible roles be: available, auctioneer, and participant. As an auctioneer, the primary concern is in forming a network segment and then orchestrating the robotic nodes within. The first step is announcing the auction session to all robotic nodes in the network. This is accomplished by generating a task announcement message to be sent in a broadcast manner to all available nodes. Assuming that the robotic sensor nodes communicate wirelessly over ad hoc networks, neighboring sensor nodes will be able to exchange network messages. This implies $R_{comm} \geq d_{i,j}$, where $R_{comm}$ is the range of the communication radio and $d_{i,j}$ is the distance from the $i^{th}$ node to the $j^{th}$ node. This message incorporates the necessary risk details so that each receiving node can appropriately bid on the task. The task announcement message is of the following format:

```
Network Header  Event Location
P_{NID}(x, y, z)
```

Fig. 1 Task announcement message sent by the auctioneer to notify all robotic nodes of the auction session.

All robotic nodes in role available will receive and accept a task announcement message. Other nodes will forward the message to all neighbors and then discard the task announcement. As an auctioneer a node can be in one of three states: AWAIT_BIDS, CLOSE_AUCTION, and
OPTIMIZE. The initial state of the auctioneer is AWAIT-BIDS where the announcement message has been sent and we need the bids to continue forward.

B. Fuzzy-Based Metric Evaluation

Upon the arrival of the task announcement message, a robotic node becomes aware of the high-risk event. Each robotic node then evaluates their availability metric (bid) to the announced task. This availability metric is based on three primary data sources: battery level, distance to the event, and coverage redundancy. The availability metric is evaluated through the use of fuzzy logic. Each robotic node will utilize a Sugeno fuzzy system [5] to produce a bid response to the auctioneer indicated in the network header of the task announcement message. Figure 2 depicts the fuzzy system available on each sensor node.

![Sugeno fuzzy model](image)

Fig. 2 Sugeno fuzzy model

The battery metric is an incentive to bid based on the fuzzification of the battery’s available energy with membership functions as follows:
- \( \mu_{\text{poor}}(x_{\text{battery}}) \); Triangular membership with parameters \( A=0.0, B=0.0, \text{and } C=0.4 \)
- \( \mu_{\text{average}}(x_{\text{battery}}) \); Trapezoidal membership with parameters \( A=0.1, B=0.35, C=0.65, \text{and } D=0.9 \)
- \( \mu_{\text{good}}(x_{\text{battery}}) \); Triangular membership with parameters \( A=0.6, B=1.0, \text{and } C=1.0 \)

Let \( x_{\text{battery}} \in [0.0,1.0] \) representing the normalized remaining battery energy on the robotic node.

The distance metric is the robotic node’s incentive to bid based on the distance of the node from the event. The closer the robotic node is to the event, the more incentive to bid. The input is \( x_{\text{distance}} \in [0.0,1.0] \), is a mapping of the actual distance, \( y_{\text{distance}} \) to the interval \([0.0,1.0]\). This is calculated by:

\[
x_{\text{distance}} = \begin{cases} \frac{y_{\text{distance}} - d_{\text{NID}}}{y_{\text{distance}}} & \text{if } d_{\text{NID}} \leq y_{\text{distance}} \\ 0.0 & \text{else} \end{cases}
\]

The distance metric uses the following membership functions:
- \( \mu_{\text{far}}(x_{\text{distance}}) \); Trapezoidal membership with parameters \( A=0.0, B=0.0, C=0.65, \text{and } D=1.0 \)
- \( \mu_{\text{near}}(x_{\text{distance}}) \); Triangular membership with parameters \( A=0.5, B=1.0, C=1.0 \)

The redundancy metric is the robotic node’s incentive to bid based on the amount of redundant sensor coverage of the user defined security perimeter. Redundant sensor coverage is the amount of sensor field-of-view overlap resulting in security perimeter points being surveyed by multiple robotic nodes. Redundant coverage promotes the incentive to bid. The input is \( x_{\text{redun}} \in [0.0,1.0] \) and can be determined for the \( i^{th} \) robotic node by:

\[
x_{\text{redun}}^{i} = \frac{|S_{\text{inter}}^{i}|}{|S_{\text{surv}}^{i}|}
\]

where \( S_{\text{inter}}^{i} \), \( S_{\text{surv}}^{i} \), \( S_{\text{surv}}^{j} \) are the set of all redundant security perimeter points by node \( i \), the set of surveyed security points by node \( i \), and the set of surveyed points by node \( j \), respectively. The input value \( x_{\text{redun}}^{i} \) for robotic node \( i \) can be determined by:

\[
x_{\text{redun}} = \begin{cases} \frac{x_{\text{battery}}}{3} + \frac{x_{\text{distance}}}{3} + \frac{x_{\text{redun}}}{3} & \text{ideal} \\ 0.0 & \text{available} \\ \text{reject} & \text{else} \end{cases}
\]

Upon an node’s determination of its availability, it must respond to the auctioneer.

C. Bid Submission

A robotic sensor node must submit a bid to the auctioneer unless its availability is 0.0 (reject). This is accomplished through the sending of a bid message as outlined in Figure 3. The auctioneer is still in state AWAIT-BIDS and must receive a minimum of \( N_{\text{bids}} \) before entering state CLOSE-AUCTION.

![Bid message](image)

Fig. 3 Bid message sent by potential participants to notify the auctioneer.
many active auction sessions until first notified by an auctioneer. All other notifications from other auctioneers are ignored.

D. Close Auction Session

The final stage of the auction-segmentation process is closing the auction. This is performed by the auctioneer by transmitting an auction closed message to all robotic nodes. This message contains a list of node ids indicating the winner bidders in the auction and will now be locked in an optimization session. As a result, the role of the robotic nodes changes from available to participant and cannot be an asset in any other optimization session nor bid in any auction. This is the case until the role of the node returns to available. The auctioneer – also the response leader of the segment – begins an optimization session using the NSGA-II to search for new segment topologies (solutions) to mitigate the risk.

IV. SELF-ORGANIZATION OF A RSN SEGMENT

Disjoint RSN segments of robotic nodes are created in the fuzzy auction-based segmentation process described in the previous chapter. The auctioneer of each segment executes the EMO optimization using the NSGA-II to explore new node positions. The NSGA-II implementation design structure is similar to that of [2]. Major changes relate to: the chromosome structure, the crossover operator, and the mutation operator.

A. Chromosome Design

In [2], the authors used a chromosome design, which represented one possible spatial configuration of a collection of robotic nodes. The encoding structure of the chromosome was a set of integer coordinate indices of all nodes. The new chromosome design allows for the possibility of finding a solution using a subset of the selected nodes for optimization. This was done in the hopes of producing a set of effective yet simple solutions.

The new chromosome consists of two layers: a node selection layer and a target location layer. Figure 4 depicts the chromosome used. As mentioned in the previous section, the auctioneer will acquire the top \(N_{bids}\) which defines the chromosome length of \(N_{bids}\). The first layer sets whether an node is included in the solution, while the second layer defines the target coordinate of the node in the solution. If an node’s gene is disabled, its location will be its initial coordinate, \(P_i^{\text{Initial}}\). Whereas, if an node is included, it will be located at \(P_i^{\text{Target}}\). This is the target coordinate \(j\) for robotic node \(i\).

<table>
<thead>
<tr>
<th>Gene 1</th>
<th>Gene 2</th>
<th>Gene 3</th>
<th>Gene 4</th>
<th>Gene 5</th>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Location Index</td>
<td>Target Location Index</td>
<td>Target Location Index</td>
<td>Target Location Index</td>
<td>Target Location Index</td>
<td>–</td>
</tr>
</tbody>
</table>

Fig. 4 NSGA-II chromosome design consisting of two layers of information. The first layer includes or excludes an node of a solution. The second layer represents the location coordinate index of a solution.

Let the following mathematically represent the chromosome:

\[
\text{Chromosome} = [\alpha^1 \alpha^2 \ldots \alpha^{N_{bids}}]
\]

where \(\alpha^k\), is the \(k\)th gene in the chromosome. This represents the \(k\)th top-bidder in the auction session. The gene can be represented as:

\[
\alpha^k = [\delta^k]_{\varphi^k}
\]

where \(\delta \in \{0,1\}\) is the enabled status. A value of ‘1’ represents enabled and a value of ‘0’ represents disabled. \(\varphi \in \{1,2,\ldots,N_{tgs}\}\) is the target coordinate index, where \(N_{tgs}\) is the total number of available target indices/points.

B. Crossover Operator

In [2] a uniform crossover operator was implemented, where single gene values were randomly exchanged between two parents. Uniform crossover is more exploratory as every gene is swapped, whereas one-point and two-point crossover operators maintain much more of the data structure. This work implements a dual crossover operator which will uniquely crossover both layers of the chromosome. The node selection layer will be subject to one-point crossover and the target location layer will use uniform crossover. Crossover operation will occur with a probability of \(\rho_{\text{crossover}}\).

For the one-point crossover operator, the crossover point \(x_{\text{crossPr}}\) is random value of the set \(\{1,2,\ldots,N_{\text{bids}}\}\). This value represents the splitting point for individuals A and B. If we consider the sets after splitting the individuals:

\[
S_{\text{left}}^A = \{\delta^1_{\text{indA}}, \ldots, \delta^{x_{\text{crossPr}}}_{\text{indA}}\}
\]

\[
S_{\text{right}}^A = \{\delta^{x_{\text{crossPr}}+1}_{\text{indA}}, \ldots, \delta^{N_{\text{bids}}}_{\text{indA}}\}
\]

\[
S_{\text{left}}^B = \{\delta^1_{\text{indB}}, \ldots, \delta^{x_{\text{crossPr}}}_{\text{indB}}\}
\]

\[
S_{\text{right}}^B = \{\delta^{x_{\text{crossPr}}+1}_{\text{indB}}, \ldots, \delta^{N_{\text{bids}}}_{\text{indB}}\}
\]

the resultant sets of node selection values would be:

\[
S_A^A = S_{\text{left}}^A \cup S_{\text{right}}^B
\]

\[
S_B^A = S_{\text{left}}^B \cup S_{\text{right}}^A
\]

For the uniform crossover operator on the target location layer, chromosomes are randomly pulled from either parent chromosome with equal chance, to construct a child chromosome.

C. Mutation Operator

Mutations occur on chromosome genes with a probability of \(\rho_{\text{mutation}}\). For the node selection layer, the enabled state will simply be negated. On the target selection layer, if a gene is to be mutated, a new random target index on the interval of \([0, N_{t}]\) will be selected. Let \(N_{t}\) be the number of target points used.

V. EXPERIMENTAL RESULTS

In our experimentation, Microsoft Robotics Studio [7] was used to simulate an RSN for CIP. The setting is a fenced compound with two access points at paved roads for vehicles. A sensor network of 115 nodes is deployed along the perimeter of the compound’s fence.
The RSN will establish a virtual fence, encapsulating the physical fence around the compound. The main objective of the RSN is to be aware of any movements along the security perimeter, while detecting any risks within the network. Each sensor node uses a simulated Global Positioning System (GPS) module as a primary source of localization and a virtual Laser Range Finder (LRF) as an instrument to detect intruders. The sensing field of view is modeled as a circular region centered on the node’s location with a radius of 3.5 m. Figure 6 show a 2-D graphical representation of the simulation.

A security perimeter (shown in Figure 6 as a finely dotted contour around the CI) is defined manually by the user. Sensor nodes will evaluate their coverage metrics based on surveillance of this contour. The contour is discretized into 750 perimeter points. The sensor nodes are initialized in a distributed manner along the security perimeter with some overlapping sensor coverage. The batteries are initialized using a uniform random distribution between 20% and 100%. In one experimental run of the simulation, two nodes identify themselves as high-risk due to low battery levels. Node 1 is in distress with battery level 19% and node 31 is has battery level 22% and is also in distress. The potential coverage loss of node 1 is 4.61 m and 1.54m from node 31. These nodes require recharging, but until they can be replaced the network must create independent autonomous groups and fill the combined coverage gap of 6.15 m.

To form the autonomous groups, node 1 and node 31 assume roles as auctioneers and announce auction sessions. Both of these nodes seek the most suitable robotic nodes to fill in the coverage gap and so they enter the AWAIT_BIDS state. They remain in this state until $N_{bids} = 10$ bids are received. Table I and Table II tabulate the results of node 1’s and node 31’s auction sessions, respectively.

**TABLE I. AUCTION RESULTS FROM NODE 1**

<table>
<thead>
<tr>
<th>Auctioneer (Node 1)</th>
<th>$x_{batt}$</th>
<th>$x_{cov}$</th>
<th>$x_{dist}$</th>
<th>$y_{await}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.91</td>
<td>0.33</td>
<td>0.79</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>0.93</td>
<td>0.44</td>
<td>0.61</td>
<td>1.00</td>
</tr>
<tr>
<td>114</td>
<td>0.95</td>
<td>0.56</td>
<td>0.61</td>
<td>1.00</td>
</tr>
<tr>
<td>115</td>
<td>0.85</td>
<td>0.22</td>
<td>0.78</td>
<td>0.88</td>
</tr>
<tr>
<td>113</td>
<td>0.93</td>
<td>0.44</td>
<td>0.43</td>
<td>0.86</td>
</tr>
<tr>
<td>4</td>
<td>0.97</td>
<td>0.44</td>
<td>0.41</td>
<td>0.83</td>
</tr>
<tr>
<td>5</td>
<td>0.89</td>
<td>0.56</td>
<td>0.23</td>
<td>0.48</td>
</tr>
<tr>
<td>112</td>
<td>0.80</td>
<td>0.44</td>
<td>0.22</td>
<td>0.38</td>
</tr>
<tr>
<td>6</td>
<td>0.86</td>
<td>0.56</td>
<td>0.05</td>
<td>0.17</td>
</tr>
<tr>
<td>111</td>
<td>0.90</td>
<td>0.56</td>
<td>0.04</td>
<td>0.12</td>
</tr>
</tbody>
</table>

**TABLE II. AUCTION RESULTS FROM NODE 31**

<table>
<thead>
<tr>
<th>Auctioneer (Node 31)</th>
<th>$x_{batt}$</th>
<th>$x_{cov}$</th>
<th>$x_{dist}$</th>
<th>$y_{await}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>0.93</td>
<td>0.67</td>
<td>0.66</td>
<td>1.00</td>
</tr>
<tr>
<td>32</td>
<td>0.91</td>
<td>0.33</td>
<td>0.85</td>
<td>1.00</td>
</tr>
<tr>
<td>33</td>
<td>0.92</td>
<td>0.67</td>
<td>0.69</td>
<td>1.00</td>
</tr>
<tr>
<td>34</td>
<td>0.91</td>
<td>0.56</td>
<td>0.52</td>
<td>1.00</td>
</tr>
<tr>
<td>28</td>
<td>0.84</td>
<td>0.63</td>
<td>0.50</td>
<td>0.91</td>
</tr>
<tr>
<td>30</td>
<td>0.86</td>
<td>0.30</td>
<td>0.83</td>
<td>0.90</td>
</tr>
<tr>
<td>27</td>
<td>0.93</td>
<td>0.44</td>
<td>0.33</td>
<td>0.67</td>
</tr>
<tr>
<td>35</td>
<td>0.93</td>
<td>0.67</td>
<td>0.34</td>
<td>0.61</td>
</tr>
<tr>
<td>36</td>
<td>0.96</td>
<td>0.78</td>
<td>0.19</td>
<td>0.49</td>
</tr>
<tr>
<td>26</td>
<td>0.88</td>
<td>0.67</td>
<td>0.14</td>
<td>0.39</td>
</tr>
</tbody>
</table>
Node 31 received 30 bids and then announces the winners of the best 10. Nodes responding with an availability of 1.0, are considered IDEAL candidates for the optimization task. If $0.0 < y_{\text{availability}} < 1.0$, then the node is considered available. This is determined by the fuzzy system rules. Most nodes did not place a bid, because there was no incentive to bid. Any rule that contains a “Distance is FAR” term will select the REJECT output function.

At this point the nodes have been selected for each independent optimization group. Each segment will begin searching for optimized topologies using generic parameters adapted from [2]: $\rho_{\text{crossover}} = 0.9$, $\rho_{\text{mutation}} = 0.1$, and a population size of 100. The response regions are derived from [2], which are circular response regions with $N_i = 200$. Instead of an exponential function to represent the response region radii, the robotic nodes are able to move $R_{max}$ adapted from [2]:

$$R_{max}^i = \begin{cases} 0 & x_{battery}^i < \lambda_{\text{battery}} \\ y_{\text{movement}} & \text{otherwise} \end{cases}$$

where $\lambda_{\text{battery}} = 0.30$ and is low battery threshold value. $y_{\text{movement}} = 0.05$ and is the battery consumed for each meter travelled. Figure 7 now shows the most one selected solution of each RSN segment.

![Fig. 7] A 2D representation of the optimized solutions. RSN nodes with a red field of view, represents a new optimized position. A gray field of view indicates excluded nodes.

The NSGA-II has been successful in exploring the solution space for high-coverage, but low energy cost solutions. The network response for node 1 is shown at the top of Figure 7. Here a coverage metric of 99.3% was obtained with a cost of 1.19. The optimization has selected new target locations for the node in the group at the top of figure 7 marked in red, which has allows them to shift over to fill in the coverage gap. Six of the robotic nodes were excluded in this solution and remained at their initial position.

In the group managed by node 31 on the right side of figure 7, the optimization algorithm has generated a feasible solution with a coverage metric of 98.9% and energy cost of 0.496. It can be observed that one node close to the auctioneer as shifted over to fill in the coverage gap. Most of the nodes were excluded from the solution NSGA-II converged on a solution that provided optimal coverage with the least energy exerted.

VI. CONCLUSIONS

An RSN deployed for CIP is a challenging task, especially to maintaining a secure perimeter in potentially harsh environment. The importance of being able to detect and mitigate risks in the network in an intelligently manner is paramount. In this paper, we have been successful in simulating an RSN for the protection of remote facility. The network was able to identify multiple sources of risk, organize into small autonomous group, and optimize the topology of each group. Due to the nature of the auction protocol implemented, this process occurred in a decentralized manner, without the need of a base station and central management system. Future work would include more involved modeling of a realistic communication protocol and expanding the types of network events and in turn expanding the network responses.

VII. REFERENCES