2007 1st Annual IEEE Systems Conference
Waikiki Beach, Honolulu, Hawaii, USA April 9-12, 2007

MULTI-CRITERIA DECISION MAKING AND BEHAVIOR ASSIGNMENT IN SENSOR NETWORKS

Prasanna Sirdhar
University of New Mexico
Albuquerque, NM 87131
prasanna@unm.edu

Asad M. Madni
Crocker Capital
Los Angeles, CA 90064
ammadni@yahoo.com

Mo Jamshidi
University of Texas at San Antonio
San Antonio, TX 78249
Mo.jamshidi@utsa.edu

Abstract - The realm of applications for sensor networks is diverse including military, commercial and environmental monitoring. The behavior of these distributed systems of sensors is highly application-specific. The behavior of the sensing system must be tailored in order to meet the demands of the given application. Prioritizing such system behavior for a given application should greatly improve the performance of achieving the desired task. In this paper, we demonstrate the concept of multi-criteria decision making in a system-of-systems application (mobile robotic system and system of sensors) and how the decision making problem can be generalized to adapt the operational behavior of large-scale autonomous system of sensors.

INTRODUCTION

The application of monitoring a large structure such as a bridge, using sensor networks would ideally require sustaining the lifetime of the deployed sensors for a long time, since the redeployment generally can be difficult, both in terms of ease and cost of deployment. In this case, network lifetime is more important than criteria such as accuracy of data, and hence we assign *network lifetime* a higher weighting factor. Consider another instance, application such as habitat monitoring. High network lifetime is desired but not a required behavior. However, more importance or priority needs to be given to efficient communication from the habitat to a command center. Consider yet another application of monitoring chemical or nuclear spill in a region. Such applications have high demands for larger node deployment in order to capture and localize all critical events in the region. Each application thus has varying demands or requirements that need to be satisfied by properly prioritizing the behavior or properties of sensor networks. In order to prioritize the system behavior, we will need to establish *criteria* for prioritizing. For example, if there were three different tasks that needed to be completed, a human might prioritize them based on time, cost or importance. However, if the criteria are of equal importance with a certain kind of interaction between them, prioritizing the system behavior becomes a multi-criteria decision making problem.

Multi-Criteria Decision Making

Multi-criteria decision making (MCDM) [1] is the study of discrete decision problems involving two or more criteria (sometimes conflicting) or objectives. In MCDM problems, the goal is to select an alternative (choice or a system) from a set of relevant alternatives by evaluating a set of criteria. For example, consider the problem of selecting a car from a set of three cars {a, b, c} (which represents our alternatives) based on set of criteria {fuel efficiency, price, luxury}.

Let $\Omega = \{s_1, s_2, \ldots s_m\}$ and $X = \{x_1, x_2, \ldots x_n\}$ be set of alternatives and set of criteria respectively. The decision making process proceeds by formulating a matrix **A** with set of criteria and set of alternatives,

$$\mathbf{A} = \begin{array}{cc} \begin{matrix} s_1 \\ s_2 \\ \vdots \\ s_m \end{matrix} & \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \\ & \begin{matrix} x_1 & x_2 & \cdots & x_n \end{matrix} \end{array}$$

Each entry a_{ij} denotes the degree to which the criterion x_j is satisfied by the alternative s_i. The

1-4244-1041-X/07/$25.00 ©2007 IEEE

idea is to now reduce the multi-criteria problem into a single global criterion problem by aggregating all the elements of matrix **A**, given by $a = H(a_{1j}, a_{2j} \ldots a_{mj})$, where $\mathcal{H}$ is the aggregation operator. Most common aggregation operator is the weighted arithmetic mean; however, this operator does not model interaction among criteria. In this paper, we discuss in detail the application of Choquet integral as an aggregation operator and MCDM problem in sensor networks.

Fuzzy Measure and Choquet Integral

A fuzzy measure [2] on a set of criteria (X) is defined as a mapping function μ: $2^X \rightarrow [0,1]$, where 2^X is the power set of X. Additionally, μ should satisfy the following properties: 1) $\mu(\emptyset) = 0$ and $\mu(X) = 1$, where $\emptyset$ represents the null-set. 2) If A is a subset of B, then $\mu(A) \leq \mu(B)$

For example, consider a set $X = \{x_1, x_2\}$. Power set of X is given by, $P(X) = \{\emptyset, \{x1\}, \{x2\}, \{x1, x2\}\}$. The fuzzy measure on the elements of set P, for example, can be defined as: $\mu(\emptyset) = 0$, $\mu(\{x_1\}) = 0.4$, $\mu(\{x_2\}) = 0.5$ and $\mu(\{x_1, x_2\}) = 1$. If μ is the fuzzy measure on X (set of criteria), then Choquet integral [2][3][4] of a function f:X$\rightarrow$[0,1] with respect to μ is defined as:

$$C_\mu(f(x_1)\ldots f(x_n)) = \sum_{i=1}^{n} (f(x_{(i)}) - f(x_{(i-1)})) \times \mu(Y_{(i)}) \quad (1)$$

where $x_{(i)}$ indicates the indices have been permuted such that $f(x_{(1)}) < f(x_{(2)}) < \ldots < f(x_{(n)})$ and $Y_{(i)} = \{x_{(i)}, \ldots, x_{(n)}\}$. If the fuzzy measure μ is additive, then C_μ represents discrete Lebesgue integral [5].

PROBLEM FORMULATION

Basic Architecture

In this paper, we propose a case study for multi-criteria decision making in mobile robot path planning in an environment deployed with sensor nodes. We can generalize such a decision making process to a more complex system management. We develop an efficient data collection and sensor node replacement scheme for sensor network in a cluttered environment. The autonomous sensor nodes embedded in the environment are generally low powered devices. High events in the environment usually require constant monitoring and dense deployment for precisely localizing the threat events. In order to capture all important events, we would ideally want more nodes deployed in the region of event compared to other regions in the environment. Any dying nodes would also require a replacement (redeployment) in order to sustain the lifetime of entire network. In this paper, we propose a novel methodology of using mobile robot to collect data, replace any dying node and to deploy more nodes in the region of higher events.

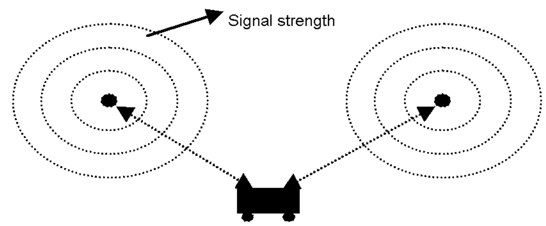

Figure 1. Basic Architecture

Our initial premise to use radio frequency (RF) signal strength alone to determine distance to node was inadequate in providing high data integrity for the following reasons: 1. Consider an analog test signal being transmitted from the robot to a node as show in figure 1. The amplitude (signal strength) of the returned signal detected may be true or may be the result of weak battery power. 2. If multiple analog signals are being transmitted from mobile robot to several nodes, the algebraic addition of these signals may provide an erroneous reading.

Hence signal strength or amplitude detection by itself can only be used as supplementary information in determining the distance to node. An easier implementation is to send out a synchronized pulse from the robot and receive the returned pulse from the sensor node and determine the travel time of pulse. In essence, this is similar to the functionality of a sonar rangefinder. Another way of determining distance is to send out a predetermined beacon signal with node ID. The robot can determine the distance by looking at any two consecutive beacon signals. These signals can be directly generated by the battery. This is an added advantage since the beacon signal in addition to providing distance-to-node information also provides a relative reading of the battery power of the node.

The decision making problem for the robot is to efficiently navigate through the sensor field to reach all the nodes. In the event of multiple paths available to the robot, the robot path planning algorithm would intelligently decide which node to reach first. The robot is challenged with equally “important” paths to navigate in order to fulfill its goal. The goal is to collect data and/or to deploy a node. With advanced technology, robots maybe able to even recharge the battery on the sensor node. However, due to low cost in node construction, we assume it is economical to redeploy a node instead of recharging the battery. The importance of a given path is based on several parameters relating to the sensor nodes in the field. Given a deployed network of sensors, the robot's task is to reach the nodes based on several competing criteria, e.g, a sensor could have critical data that needs to be collected, while, another node may be dying due to low battery power, requiring immediate attention.

Criteria: X = {$x_1, x_2, \ldots x_n$} – set of criteria
X = {distance, battery power, event level, data criticality}
Alternatives: Ω = {$s_1, s_2, \ldots s_m$} – set of systems on which criteria is to be evaluated.Ω = set of nodes
Goal: Evaluate systems/alternatives {$s_1, s_2, \ldots s_m$} based on set of criteria {$x_1, x_2, \ldots x_n$}.
G = Select a sensor node to be reached first

Table 1. Evaluation of Alternatives

criteria / Sensors	Distance	Battery Power	Event Level	Critical Data	Evaluation
sensor *1* (*s_1*)	d-1	b-1	e-1	cr-1	C-1
sensor *2* (*s_2*)	d-2	b-2	e-2	cr-2	C-2
sensor *3* (*s_3*)	d-3	b-3	e-3	cr-3	C-3
⋮	⋮	⋮	⋮	⋮	⋮
sensor *m* (*s_m*)	d-*m*	b-*m*	e-*m*	cr-*m*	C-*m*

C-1…C-*m* in Table 1, are evaluation results based on the current criteria and interaction among the criteria. The methodology used to obtain C-1…C-*m* is by using Choquet integral. A simple pair-wise comparison between two evaluation items can help to determine the *preference* for selecting a particular system (sensor node). For example, if C-1 > C-2, then sensor s_1 is preferred over s_2.

Numerical Simulations

Consider a mobile robot traversing in an environment that is covered with embedded sensors. At each predetermined discrete time interval, the robot evaluates which sensor node to reach first, based on set of criteria X. We identify two different cases for efficient evaluation of three sensor nodes.

Case 1 Criteria: Fuzzy variables/no interaction

Robot needs to reach the node that is **nearest**, has **low** battery power, and has **high** events registered. The alternatives are three nodes to be evaluated. We define the following fuzzy membership function for each criterion.

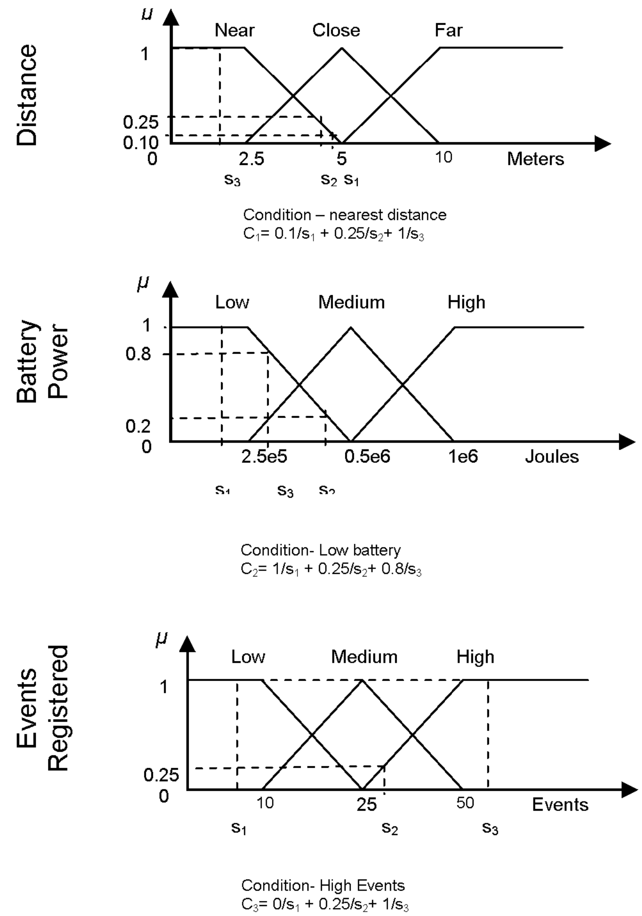

Figure 2. Fuzzy Membership for Criteria

C_1, C_2 and C_3 are the fuzzy sets obtained which expresses goal and conditions in terms of available systems s_1, s_2 and s_3.

$$D = 0.1/s_1 + 0.25/s_2 + 0/s_3 \qquad (2)$$

The decision maker's solution (*D*) is obtained by **max-min** inference [6] on the three sets C_1, C_2 and C_3. *D* is obtained from **min** of each system and represents a fuzzy characterization of the concept of desired system. Using **max**, we can obtain a preference of a given system over another system. In this case, node s_2 is the most desired system to be reached first by the robot.

Case 2 Criteria: Crisp variables/interaction

A detailed description of interaction among criteria is discussed in [7], which gives *correlation*, *complementary* and *preference dependency* as three different forms of interaction among criteria. In our case study on sensor network, criteria such as power level and capturing events are correlated and complementary. For example, in order to capture critical environmental events, a deployed sensor should ideally have a low sleep-time and high sampling frequency. This means that power consumed by the sensor is high, suggesting that power consumption and events are correlated and complementary.

Recall from section 1.3, that Choquet integral is defined over the function f : X→[0,1]. This function f is often called the *utility function* or *score* [8]. The utility function is required to make the criteria comparable, since criteria generally are not measured on a common scale. By using utility function we map the criteria to a common scale, making them *commensurable* as shown below.

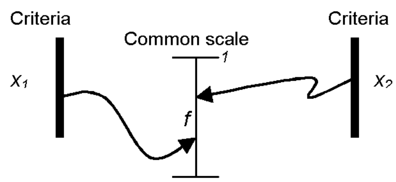

Figure 3. Mapping Criteria

Given the three criteria/attributes related to a node – distance, battery power and events registered, we can generate the following utility function:

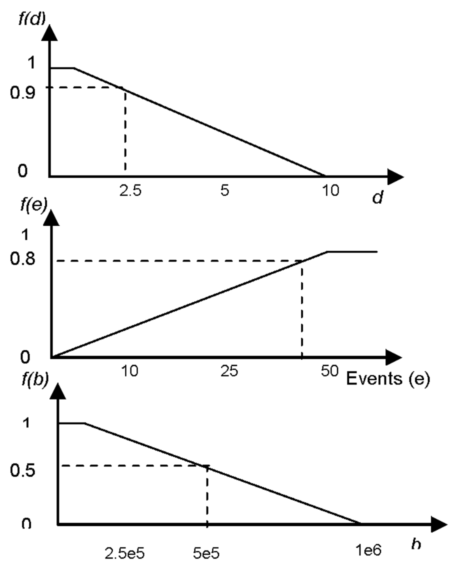

Figure 4. Mapping Criteria

From figure 4, for example, a shorter distance to a given node, generates a high score or utility function, e.g., a distance of 1m, will generate a score *f(d)*=0.9. Similarly, if the number of events generated is high (say 40), then the score is high (*f(e)*=0.8). The overall evaluation of different alternatives (sensor nodes) is obtained by aggregating the utility functions using Choquet integral with appropriate fuzzy measure (which acts like a weighting factor). The weights, fuzzy measure and resultant Choquet integral for three nodes at varying distances, battery level and event (activity) level are tabulated as given below:

Table 2. Choquet Integration

(a) Pair-wise comparison Matrix

	Distance	Battery	Events
Distance	1	3	7
Battery	0.333333	1	5
Events	0.142857	0.2	1

(b) Identified Fuzzy Measures

Sets	Fuzzy Measure
{}	0
{Distance}	0.854756
{Battery}	0.515547
{Distance,Battery}	0.978599
{Events}	0.164453
{Distance,Events}	0.89426
{Battery,Events}	0.604638
{Distance,Battery,Events}	1

(c) Input and Choquet Integrated Values

No.	Distance	Battery	Events	Choquet Integrated Values
1	0.9	0.5	0.1	C-1 = 0.833342
2	0.5	0.9	0.1	C-2 = 0.697658
3	0.1	0.1	0.9	C-3 = 0.231562

From Table 2, we see that C-1 is greater than C-2 and C-3, and hence the robot will decide to go to sensor-1. The decision making is evident from the fact that, for sensor-1 the *distance score f(d)* is set to 0.9, which means that the distance to sensor node from the robot's current position is *very near* (refer figure 3). The pair-wise comparison matrix gives the *importance* or *priority* of a criterion over another criterion in the evaluation process. The interaction degree ξ is set to 0.75 suggesting a *positive interaction*. A *positive interaction* or *positive synergy* [7] between two criteria *i* and *j* represents some degree of opposition between two criteria and the fuzzy measure then becomes *super-additive*, i.e., $\mu(ij) > \mu(i) + \mu(j)$. If ξ =0.5, then fuzzy measure is just additive, $\mu(ij) = \mu(i) + \mu(j)$, and the Choquet integral reduces to weight-average with fuzzy measures acting as weighting factors. Common aggregation operators are compared for three sensor nodes with same criteria – distance, battery and events. It is evident from figure 5, that

Choquet integral and weighted average help to better evaluate the alternatives based on the aggregated values. Weighted average can be considered as a special case of Choquet integral with an interaction degree ξ=0.5.

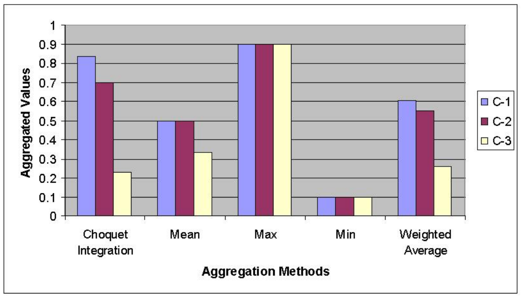

Figure 5. Comparison of Aggregation Operators

GENERALIZATION

Behavior Assignment

The problem of sensor *behavior assignment* has been discussed in [9], which defines behavior assignment as an efficient planning process for determining the sensor functions and usage according to changing situations. Two important processes involved in the behavior assignment are, 1) decision about set of tasks that sensors need to accomplish and 2) scheduling of actions for the sensors. We believe that decision making process is the hardest and important step in behavior assignment. This is because, once the decision is made on what tasks that sensor needs to be doing, scheduling actions for that decision can be implemented simply as a look-up table. The decision making process on what tasks the sensor needs to accomplish depending on the mission plan and situation generally depends on the various criteria involved. Once the behavior pattern for a given application is identified, the state of the sensor network and its performance can be used as feedback for creating training set for learning algorithms such as neural networks.

The above mentioned decision making process for mobile robot can be adapted to cluster of sensor nodes rather than individual sensor node. For example, based on *activity level, number of sensor nodes* and *importance of activity*, preference can be given to a particular cluster for management (power management, node density management, etc.). If the *number of nodes* is critical for the given application, then it gets a high weighting factor. This means that we would require some nodes to be put to sleep. We are thus changing the behavior of the network by tuning one of the parameters (increasing sleep time) based on the needs of the application. We are intelligently analyzing the characteristic of the deployed sensor network for a given application and adaptively changing its operational behavior to suit the changing demands. From decision-theoretic viewpoint, appropriate sensor action needs to be scheduled in order to achieve maximum utility. The utility could be a fuzzy measure, which is used in *Fuzzy Decision Analysis* or an Expected Value, which is used in *Bayesian Decision Analysis* [10] where the expected utility (*Eu*) of selecting an alternative based on the given utility is given by:

$$Eu_i(U(x_i)) = \sum_{i=1}^{n} P(s_i)U(x_i) \tag{3}$$

where $P(s_i)$ is the prior probability of the *i-th* state of nature (in this case, the *i-th* alternative) and $u(x_i)$ is the utility of the given criteria *i.* By using the Principle of Maximum Expected Utility (PMEU) [11], the alternative with maximum expected utility for the set of alternatives is given by:

$$\arg\max_{i=1}^{n} Eu_i \tag{4}$$

This is similar to Choquet integral, where *Eu* represents the evaluation for each sensor based on the criteria. Prior probability acts as a weighting factor in evaluating the alternatives. PMEU relies on the prior probabilities which could be difficult to obtain in certain problem scenarios.

Sensitivity Analysis

We consider sensitivity analysis to study the system behavior over range of parameter variation. A sensitivity measure such as *average response, variance, normalized response or extrema* [12] can be used to evaluate the system behavior for variations in the input parameter.

The applications involving sensor networks often fall into two broad categories – monitoring events or monitoring space. In monitoring events, the deployed sensors should capture the events in the environment that either occurs randomly or at regular time intervals. In monitoring space, the sensor node should be deployed in such a way so as to maximize the area of coverage. One important issue when dealing with sensor network applications is the energy consumption. In this section, we discuss the sensitivity analysis of the given sensor network with respect to variation in the operation power of each sensor node. Specifically, we evaluate the impact of the power variation on the system outputs – total lifetime and area covered. We assume that the sensor network has distributed power, i.e., each node has its own power supply.

We examine the system output behavior in terms of the lifetime of the entire sensor network. There are numerous ways to sustain the lifetime of sensor network [13]: 1) *Global level or system-wide*: Increase the number of redundant sensor nodes. These redundant nodes act as back-up nodes and can take over the task of sensing and signal communication from any dying nodes in order to sustain overall network lifetime. 2) *Local level*: Scheduling and low power operation of each individual sensor node.

By adjusting either network parameter (increase in number of nodes) or node parameter (power scheduling), we can sustain the lifetime of the given sensor network. Whenever such a parameter adjustment is performed, the behavior of the network changes and a sensitivity analysis can be performed to evaluate the behavioral changes. At the node level, in order to conserve battery power, the nodes can be scheduled to sense the environment at or time intervals. Although, some information might be lost, this is an effective way to optimize energy consumption. Figure 6 shows the sleep mode scheduling.

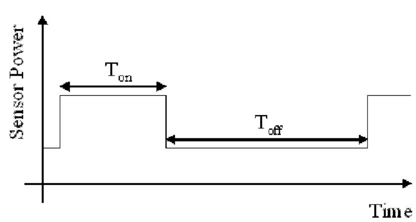

Figure 6. Typical Time Scheduling

The energy E_c consumed by each sensor is:

$$E_c = \sum_{i=1}^{n} T_i P_i \quad (5)$$

where T_i is the time period and P_i is the power at which the sensor operators in the given time period T_i. The lifetime is the reciprocal E_c. Let the sensor work in full power, i.e., $P_i = P_f$ without any scheduling and T_{L1} be the network lifetime based on E_{c1}. If the sensor is scheduled, then the energy consumed after scheduling E_{c2} will be less than E_{c1}, suggesting that lifetime T_{L2} will be greater than T_{L1}. The sensitivity measure is:

$$S_P^{E_c} = \frac{\partial E_c}{\partial P_j} = T_j \quad (6)$$

During the T_{off} period, the sensor could be completely put to sleep (meaning zero power consumption) or it could work at a lower power. The area covered by each of the sensor node considerably reduces when the sensor nodes is working in low power (a portion of full power). Let A_1 be area covered by the sensors when they are working under power P_1 and A_2 be area covered by the sensors when they are working under reduced power P_2. We perform sensitivity analysis as a performance degradation measure with respect to change in power. When the sensors work under reduced power P_2 then area covered A_2 is much less than A_1, indicating a degradation in the performance. The sensitivity measure is:

$$S_P^{Ac} = \frac{A_1 - A_2}{A_1} \cong \frac{P_1 - P_2}{P_1} \quad (7)$$

The total area covered A_T by n sensors is:

$$A_T = \bigcup_{i=1}^{n} A_{ci} \quad (8)$$

If there is negligible/no overlapping between areas associated with individual sensors, then A_T can be approximated to:

$$\bigcup_{i=1}^{n} A_{ci} \cong \sum_{i=1}^{n} A_{ci} = \sum_{i=1}^{n} \pi r_i^2 \quad (9)$$

where A_{ci} is the circular area covered by the *i-th* sensor. The sensitivity measure given in (7) reduces to (10), where P_{1i} is the power at which the *i-th* node works and P_{2i} is the power at which the *i-th* node works after perturbation (reducing the power). This reasoning stems from the fact that when the power of the node is reduced by a certain percentage of full power, the sensing radius of the sensor node reduces.

$$S_P^{Ac} = \frac{\sum_{i=1}^{n}(r_{1i} - r_{2i})^2}{\sum_{i=1}^{n} r_{1i}^2} \cong \frac{\sum_{i=1}^{n}(P_{1i} - P_{2i})^2}{\sum_{i=1}^{n} P_{1i}^2} \quad (10)$$

SoS and Scalability Issues

Consider the system of robots and system of in-situ sensors working synergy to efficiently monitor a region of interest. When two or more scalable systems are working together, the entire system-of-systems platform need not necessarily be scalable, i.e., the *performance*, *availability* and *fault tolerance* might not be in the acceptable thresholds when the load is increased in the system. We consider an idealistic situation, where the sensor nodes are placed in such a way that sensing regions of two or more sensor nodes do not overlap (dead region). In such situations, an increase in sensor node deployment will increase the coverage of the area, thereby increasing the performance. However, this might not be a linear increase due to the failure of some of the already deployed node (dead nodes). The availability and failure of the entire network also increases due to increase in node density (as some nodes can act as redundant back-up nodes). This method of scaling is sometimes referred to as **scale-out**, where more load (nodes) is added to a given system. In **scale-up**, one can add more resources (memory, processor power, low power usage etc.) to a single sensor node so as to increase its performance (coverage) which in turn can affect the scalability of the entire network. However, increase in the number of sensor nodes, might have an adverse effect on the performance of the mobile robot, which has to navigate in the sensor field to collect/disseminate data and/or redeploy more sensor nodes. Therefore, the number of robots operating in the environment needs to be increased thus increasing the demands for efficient decision making. Our proposed approach of evaluating alternatives based on the present scenario/multiple criteria provides an efficient, scalable solution in handling a dynamic situation.

CONCLUSION

In this paper, we demonstrated the importance of multi-criteria decision making in sensor network. To our knowledge, there has been very few directed research relating to decision making in sensor networks. The unpredictable state and dynamic behavior of the environment often requires automated decision making to select the best action in order to improve the performance or increase the network lifetime of the deployed sensor network. Choquet integral and fuzzy measures provide an efficient solution for such decision making process under uncertainty.

REFERENCES

[1] Belton,V., Steward, T.J., "Multiple Criteria Decision Analysis-An Integrated Approach", *Kluwer Academic Publishers*, 2002.

[2] Grabisch, M., "The application of fuzzy integral in multicriteria decision making", *European journal of operational research*, vol. 89:445-456, 1995.

[3] Torra, V., "On integration of numerical information: from arithmetic mean to fuzzy integrals", *Information fusion in data mining*, PhysiinVerlag, 2001.

[4] Denguir-Rekik,A., Mauris,G., Montmain,J., "Propagation of Uncertainty by the Possibility Theory in Choquet Integral-Based Decision Making: Application to an E-Commerce Website Choice Support", *IEEE Trans. on Instrumentation & Measurement*, VOL. 55, NO. 3, June 2006.

[5] Wang Z., Klir, G.J., Wang, W., "Determining fuzzy measures by Choquet integral", *Proc. of the 3rd International Symposium on Uncertainty Modeling and Analysis*, pp 724, 1995.

[6] Klir, G., Yuan, B., "Fuzzy Sets and Fuzzy Logic: Theory and Applications", *Chapter 15: Fuzzy Decision Making*, Prentice Hall NJ, 1995.

[7] Marichal., J-L., "An Axiomatic Approach of the Discrete Choquet Integral as a Tool to Aggregate Interacting Criteria", *IEEE Transactions on Fuzzy Systems,* Vol 8, No. 6, pp. 800-807, Dec 2000.

[8] Huédé, F., Grabisch, M., Labreuche, C., Savéant, P., "Integration and propagation of a multi-criteria decision making model in constraint programming", *Journal of Heuristics*, Vol 12 ,Issue 4-5, pp. 329 – 346, 2006.

[9] Xiong, N., Svensson, P., "Multi-sensor Management for Information Fusion: Issues & Approaches", *Information Fusion,* Elsevier, Vol 3, pp. 163-186, 2002.

[10] Duda, R., Hart, P., Stork, D., "Pattern Classification", *Bayesian Decision Theory,* pp. 20-83, 2nd Ed., Wiley-Interscience Publication, 2001.

[11] Russell, S., Norvig, P., "Artificial Intelligence: A Modern Approach", Prentice Hall, 1995.

[12] G. J. McRae, W. R. Goodin, and J. H. Seinfeld. "Mathematical Modeling of Photochemical Air Pollution", Technical report, Environmental Quality Laboratory, California Institute of Technology, Pasadena, CA, 1982.

[13] Azarnoush, H., Horan, B., Sridhar P., Madni, A M., Jamshidi, M., "Towards Optimization of a Real-World Robotic-Sensor System of Systems", *World Automation Congress*, Hungary, 2006