Diagnostic Bayesian Networks with Fuzzy Evidence

Nicholas Ryhajlo, Liessman Sturlaugson, and John W. Sheppard
Department of Computer Science
Montana State University
Bozeman, MT 59717
nick.ryhajlo@gmail.com, {liessman.sturlaugson, john.sheppard}@cs.montana.edu

***Abstract*—Diagnostic Bayesian networks, one of the models supported by the Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE) standard, are an important and commonly used tool for modeling systems for fault isolation. When performing the tests specified by the diagnostic Bayesian network, the test program often maps the raw test measurements to discrete *Pass* or *Fail* outcomes. We would like to relax this hard discretization requirement and instead represent degrees of passing and failing. To do this, we propose a method for integrating fuzzy set theory and diagnostic Bayesian networks. Our proposed approach further demonstrates the extensions described in previous work to include gray-scale health information in AI-ESTATE. The previous work demonstrated the use of soft outcomes in AI-ESTATE's Fault Tree Model (FTM); however, no process was given for incorporating the soft outcomes into the other models specified by AI-ESTATE. Here, we describe how to extend the AI-ESTATE Bayesian Network Model (BNM) to incorporate the previously proposed soft outcomes. Because D-matrices and diagnostic logic models can be represented as Bayesian networks, the proposed approach can be adapted to work with AI-ESTATE's D-matrix Inference Model (DIM) and Diagnostic Logic Model (DLM) as well.**

I. Introduction

Standards for prognostics and health management (PHM) provide a framework for managing applications used in diagnostic environments. In the current specifications of the Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE) standard [1] measurements are mapped to discrete TestOutcome values such as *Pass* or *Fail* and diagnoses are mapped to discrete DiagnosisOutcome values such as *Good* or *Bad*. Previous work proposed extensions to the current specifications of AI-ESTATE to allow for real-valued outcomes [2].

These real-valued outcomes allow for a more natural representation of a system. As an example, when tests are performed using continuous-valued data, like a voltage or temperature measurement, it can be difficult to specify specific thresholds when tests pass and when they fail. In addition to this difficulty, a lot of possibly valuable information is lost in this discretization process. The real-valued outcomes can be used to represent partial pass or partial fail conditions. In the context of diagnoses a real-valued outcome can be used to represent gray-scale health of a system or component. We use these real-valued outcomes to represent component or system-level degradation.

Diagnostic Bayesian networks are supported by the AI-ESTATE standard. However they require discrete TestOutcome and DiagnosisOutcome values. We describe a method for integrating real-valued test outcomes as fuzzy evidence into a fuzzy Bayesian network. Additionally we describe the process for using this fuzzy Bayesian network to determine component-level degradation with a real-valued DiagnosisOutcome value. We demonstrate this process by using a fuzzy Bayesian network to model battery capacity degradation from a lithium-ion battery health dataset from the NASA Prognostics Center of Excellence data repository [3]. Specifically, this paper demonstrates the usage of these modifications to the AI-ESTATE standard with a fuzzy Bayesian network to model gray-scale health as component degradation.

II. Background

A. Bayesian Networks

Bayesian networks are probabilistic models corresponding to joint probability distributions that utilize conditional dependencies among random variables. Bayesian networks are used in a wide variety of domains, such as image processing, search, information retrieval, diagnostics, and many others. A Bayesian network uses observations, or evidence, and previously determined conditional probabilities to give the probability of a certain state.

More formally, a Bayesian network $\mathcal{B}$ is a directed, acyclic graph whose vertices correspond to random variables of a distribution, and the edges correspond to conditional dependencies between random variables. Each vertex has an associated conditional probability distribution: $P(X_i|\text{Pa}(X_i))$, where $\text{Pa}(X_i)$ are the parents of vertex X_i. The lack of an edge between two vertices indicates there is no direct interaction between the two nodes. However, these nodes can still interact in certain circumstances. An example of this is a V-structure in which the state of a common child of the two nodes is known.

Bayesian networks are a way of representing joint probability distributions in a more compact way by using conditional dependencies among the random variables. Instead of needing to enumerate the entire joint probability distribution we can just use the product rule from probability to get the following:

$$P(X_1, ..., X_n) = P(X_1) \prod_{i=2}^{n} P(X_i|X_1, ..., X_{i-1})$$

Bayesian networks are able to exploit conditional independence, which is represented in the directed acyclic graph $\mathcal{G}$, to reduce the model's complexity and yield the following:

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i | \text{Pa}(X_i)).$$

Bayesian networks are frequently used because the representation they use is often easier to understand than other graphical models, such as Artificial Neural Networks (ANNs). Additionally, even without the use of evidence, it can be much easier to tell what a particular network is representing and how it will behave in the presence of evidence. Bayesian networks are generally easy for domain experts to construct because of their reliance on conditional probabilities and not on instance-specific weights that are difficult to interpret, such as with ANNs.

B. Fuzzy Sets

Fuzzy set theory acts as an extension to traditional crisp set theory [4]. A set is defined as a collection of distinct objects. Each of these sets is defined by an associated membership function defined in a domain of discourse U. The membership function represents the degree of membership of some object $u \in U$ within a set.

Within the context of traditional crisp set theory, a membership function for set A is defined as

$$\mu_A : U \rightarrow \{0, 1\}.$$

The possible values of 1 or 0 represents an object either being a member of set A or not. On the other hand, for fuzzy sets, we define a membership function as

$$\mu_A : U \rightarrow [0, 1].$$

This extension to the concept of a set membership allows for varying degrees of membership within a set. This extension consequently allows for partial membership within a set, and partial membership in what would otherwise be conflicting sets.

Similar to set operators within crisp set theory, we can also define set operators within fuzzy set theory. For this work we use the fuzzy union (also knowns as the s-norm) and fuzzy intersection (also known as the t-norm). There are various definitions for the fuzzy union and fuzzy intersection operators [5]. We use the ones defined in Equations 1 and 2 for fuzzy union and fuzzy intersection, respectively.

$$\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \quad (1)$$

$$\mu_{A \cap B}(x) = \mu_A(x) \cdot \mu_B(x) \quad (2)$$

C. Fuzzy Random Variables

Fuzzy random variables (FRV) were introduced in 1978 by Kwakernaak [6], [7] and enhanced by Puri and Ralescu [8] to model imprecisely valued functions represented by fuzzy sets that are associated with random experiments [9]. Kwakernaak introduced FRVs as "random variables whose values are not real, but fuzzy numbers" [10]. Central to the concept of the FRV is a concept of "windows of observation." Windows of observation correspond to linguistic interpretations of traditional random variables. An example of this is the task of classifying people by age. The actual age is represented by an ordinary random variable X. However, when we perceive people, we typically assign a linguistic label to their age, such as "old" or "young." The corresponding perceived random variable, ξ, can be conceptualized through the use of linguistic variables, or fuzzy sets.

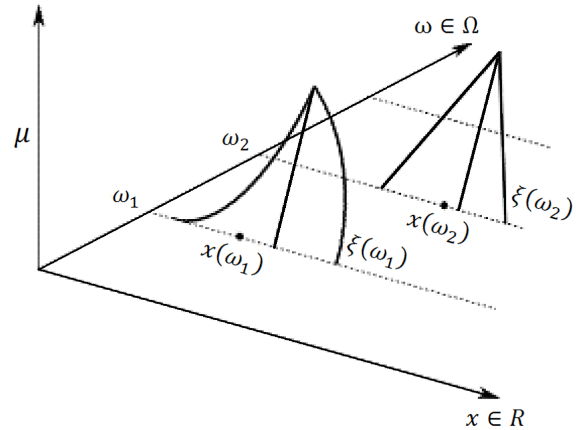

Fig. 1. Visual representation of Fuzzy Random Variables

Thus a fuzzy random variable is a mapping from the sample space, Ω, of the random variable to the class of normal convex fuzzy subsets. Thus, every instance in the sample space is mapped to its own fuzzy membership function. In Figure 1 (from [10]), we can see for each ω_i there is a corresponding membership function. In the context of the age example, ω_i would be an observation of a person, $\xi(\omega_i)$ is the mapping that defines the perception of that persons age, and finally $x(\omega_i)$ is the actual person's age.

Often, these mappings are also defined with particular α-cuts.[1] So, essentially an FRV is a mapping from an event $\omega \in \Omega$ to a fuzzy membership function, which can have a α-cut applied to it. A graphical example of these windows with fuzzy random variables can be seen in 1. In this figure each ω represents an event. Then with each event, there is a window, which is represented by $\xi(\omega)$. Each of these membership functions are specific to each observation of each instance of the random variable X.

D. Fuzzy Bayesian Networks

Bayesian networks are very powerful tools and are used many different situations and domains. They are a useful and compact method for representing joint probability distributions. Similarly fuzzy sets are able to represent data in linguistic terms that help to improve understandability. Additionally, the fuzzy membership function provides a framework for representing degrees of membership in a set.

[1]An α-cut is the crisp set containing the set of elements whose membership values exceed some threshold α.

Combining these two ideas can be conceptually difficult because the meaning of a fuzzy membership value and a probability are very different, yet they are represented similarly (a real number on the range [0,1]). Nevertheless, fuzzy Bayesian networks are not uncommon in the literature. There are many different methods for integrating these two tools presented by various authors. Many of these techniques differ from each other because they are often being used to represent different concepts. In addition to different techniques, nearly every work uses different notation. This can make it difficult to understand the similarities and differences between the various techniques.

1) Fuzzy Bayesian Equation: Possibly the most common method of integrating fuzzy sets and Bayesian networks is presented in [11]. This method manipulates Bayes' rule to incorporate fuzzy membership values into the inference process. Three circumstances are described in which fuzzy membership values can be used. The first uses a fuzzy variable as a query variable and a crisp random variable as evidence.

$$P(\widetilde{A}|B) = \frac{\sum_{i \in I} \mu_{\widetilde{A}}(A_i) P(B|A_i) P(A_i)}{P(B)} \tag{3}$$

The second uses a crisp query variable and fuzzy evidence.

$$P(A|\widetilde{B}) = \frac{\sum_{i \in I} \mu_{\widetilde{B}}(B_i) P(B_i|A) P(A)}{P(\widetilde{B})} \tag{4}$$

The final method uses a fuzzy variable for both the evidence and query.

$$P(\widetilde{A}|\widetilde{B}) = \frac{\sum_{i \in I} \sum_{j \in J} \mu_{\widetilde{A}}(A_i) \mu_{\widetilde{B}}(B_j) P(B_j|A_i) P(A_i)}{P(\widetilde{B})} \tag{5}$$

Each of these methods requires fuzzy marginalization.

$$P(\widetilde{B}) = \sum_{i \in I} \mu_{\widetilde{B}}(B_i) P(B_i) \tag{6}$$

In Equations 3 through 6, I represents the set of all states for the given random variable.

The above method seems to make intuitive sense in the way it combines the probabilities and the fuzzy membership values. A significant drawback of the method, however, is that it requires changes to the underlying inference algorithm used because of the dependence on the fuzzy marginalization process (Equation 6). This means that a custom inference engine must be used.

2) Virtual Evidence: An alternative method of incorporating fuzzy membership values into a Bayesian network is to use virtual evidence [12]. Virtual evidence is a method for incorporating uncertainty of evidence into a Bayesian network. The process of using virtual evidence to incorporate fuzzy values into a Bayesian network is very straightforward. Once the virtual evidence node is added, fuzzy evidence is incorporated directly as another state observation. Virtual evidence is represented in manipulating the conditional probability table of the virtual evidence node.

We illustrate this process with the example network given in Figure 2. We assume we have a simple Bayesian network consisting of two nodes corresponding to *Resistor Short* and *Current Test*. We add a new node to the network representing our virtual evidence on the *Current Test* node. Specifically, we add the node *VE Current Test* as a child of *Current Test*. Note that the convention we are using is nodes that are ovals are query/diagnosis nodes, square nodes represent hidden nodes, and diamonds represent evidence nodes.

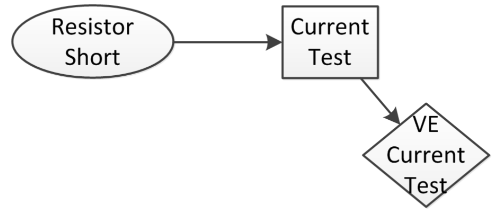

Fig. 2. Simple Example Bayesian Network with a Virtual Evidence Node

This method, similar to the Fuzzy Bayesian Equation method, makes the assumption that the fuzzy membership value can be integrated directly with the probabilities in the network. However, unlike the Fuzzy Bayesian Equation method that uses the membership value as a weight, this method assumes the membership value is a probability. Unfortunately, this creates a semantic disconnect in that virtual evidence is a method for incorporating uncertainty of evidence into a Bayesian network. Fuzzy membership functions, on the other hand, represent a degree of membership of that set. More specifically, fuzzy membership represents a level of state assignment, not the likelihood that the evidence is true.

3) Fuzzy Probability Distribution: Another alternative method for implementing a fuzzy Bayesian network (and the one we chose) is to use fuzzy probability distributions. The fuzzy probability distribution is a composite representation of a probability distribution and a fuzzy state. A probability distribution over a random variable T enumerates probabilities of being in each state of T. For example, assume T has two states *high* and *normal*, the probability distribution for T could be represented as: $P(T = \textit{high}) = 0.7$ and $P(T = \textit{normal}) = 0.3$. We can represent that distribution as:

$$T = \{\textit{high}_{0.7}, \textit{normal}_{0.3}\}$$

If we assume the state ordering to always be consistent, we can simplify the representation to:

$$T = \{0.7, 0.3\}$$

A fuzzy state is made up of fuzzy membership values for each possible state for a random variable. If we use a similar example to above with variable T, a possible fuzzy state could be $\mu_{S=\textit{high}}(x) = 0.6$ and $\mu_{S=\textit{normal}}(x) = 0.4$. Similar to the example above we can represent the fuzzy state as:

$$S = [\textit{high}_{0.6}, \textit{normal}_{0.4}]$$

We can also assume the state ordering to always be consistent and simplify the representation to be:

$$S = [0.6, 0.4]$$

We then combine the representations of the probability distribution and the fuzzy state to create a fuzzy probability distribution X:

$$X = [\{0.7, 0.3\}_{0.6}, \{0.2, 0.8\}_{0.4}]$$

This notation, along with using fuzzy probability distributions with Bayesian networks is presented in [13], [14]. We use this method with some enhancements for our implementation of a fuzzy Bayesian network.

III. Proposed Extension to AI-ESTATE

Since 1990, the IEEE 1232 Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE) standard [1] has been developed and matured by the IEEE Standards Coordinating Committee 20 (SCC 20) on Test and Diagnosis for Electronic Systems as a standard specifically for applying techniques from artificial intelligence to the problems of system test and diagnosis [15]. AI-ESTATE is intended to provide a formal data specification that supports the capture and exchange of types of information commonly used in system test and diagnosis and to define a standard interface between test systems and diagnostic reasoners [16].

The standard uses the EXPRESS information modeling language [17] to define four semantic models, namely fault trees (diagnostic decision trees), D-matrices, logic models, and Bayesian networks. The Extensible Markup Language (XML) schemata have been derived from these semantic models through the Standards for the Exchange of Product model data (STEP) from the International Organization for Standardization (ISO) [18]. Along with these information models, the standard defines a set of services commonly used during test, diagnosis, and repair for supporting standardized interaction between a test application and a diagnostic reasoner. Along with the diagnostic models and services, AI-ESTATE defines an information model for capturing the histories of a diagnostic session as the test system and the reasoner interact by means of the services.

A. Previous Extension to AI-ESTATE

Since the publication of the current version of the AI-ESTATE standard in 2010, we have been seeking ways to promote AI-ESTATE by enhancing its capabilities, furthering a standards-based approach for performing not only diagnostics but also prognostics. Previous work to address requirements for PHM showed how AI-ESTATE can incorporate gray-scale health information instead of only hard outcomes of *Good*, *Candidate*, etc. We demonstrated the use of these gray-scale outcomes on one of the four models, extending the AI-ESTATE Fault Tree Model to encode a fuzzy fault tree [2]. In this paper, we demonstrate the use of this extension on the BNM, which in turn is a generalization of the D-Matrix Inference Model (DIM) and the Diagnostic Logic Model (DLM).

The AI-ESTATE's Bayes Network Model (BNM) is a specialization of Bayesian networks specific to diagnostics. In particular, the BNM defines a Bayesian network in which all of the nodes consist of either BayesDiagnosis entities (with subtypes BayesFault and BayesFailure) or BayesTest entities. BayesDiagnosis entities are not conditionally dependent on anything else, while BayesTest entities can be conditionally dependent on BayesDiagnosis entities or even other BayesTest entities. This common assumption is that random variables corresponding to diagnoses can only depend on test random variables, i.e., the existence of one fault does not cause another fault to occur [19].

This extension to AI-ESTATE for gray-scale health information was previously proposed and demonstrated using a fuzzy fault tree [2]. However, the representative power of the extension is much broader than that single implementation. Here we demonstrate the same extension with another possible use, a fuzzy Bayesian network. We now review this proposed extension to AI-ESTATE.

The AI-ESTATE extension for gray-scale health information relaxes the standard from imposing strictly discrete outcomes on tests, actions, and diagnoses to supporting real-valued outcomes. The extension still allows for discrete outcomes as a special case when discrete outcomes are appropriate and for backwards compatibility with legacy systems and models. The inclusion of real-valued outcomes would greatly expand the representative capability of the standard. Real-valued action and test results could be used by diagnostic reasoners to fine-tune diagnostic conclusions and return diagnoses as current states of degradation rather than strictly *Good* or *Bad* outcomes. Furthermore, support of gray-scale health and degradation information furthers the possibility for more advanced PHM applications, allowing reasoners to represent, incorporate, and reason about degradation of the components of a system through time.

B. Soft Outcomes

Our proposed extension for gray-scale health information is represented in AI-ESTATE using a set of soft outcomes that each use a basis function or a mixture of several basis functions. Degrees of membership within these possibly overlapping sets can represent gray-scale information in the measurements (with test outcomes) and underlying health state (with diagnosis outcomes). This approach is general and enables support by the standard for a variety of model implementations, such as fuzzy logic, artificial neural networks, mixtures of Gaussians, Bayesian networks with continuous random variables, or, as demonstrated in this paper, fuzzy Bayesian networks.

Most of the proposed changes affect only the Common Element Model (CEM), specifically the Outcome entity, as shown in Figure 3. The current standard enumerates a set of discrete outcomes (e.g., *Good*, *Bad*, and *Candidate* for diagnoses; *Pass*, *Fail*, and *Unknown* for tests). The attribute allowedValue of the abstract supertype Outcome represents a single discrete value associated with the specific outcome, based on whether the instantiating subtype is ActionOutcome, TestOutcome, or DiagnosisOutcome.

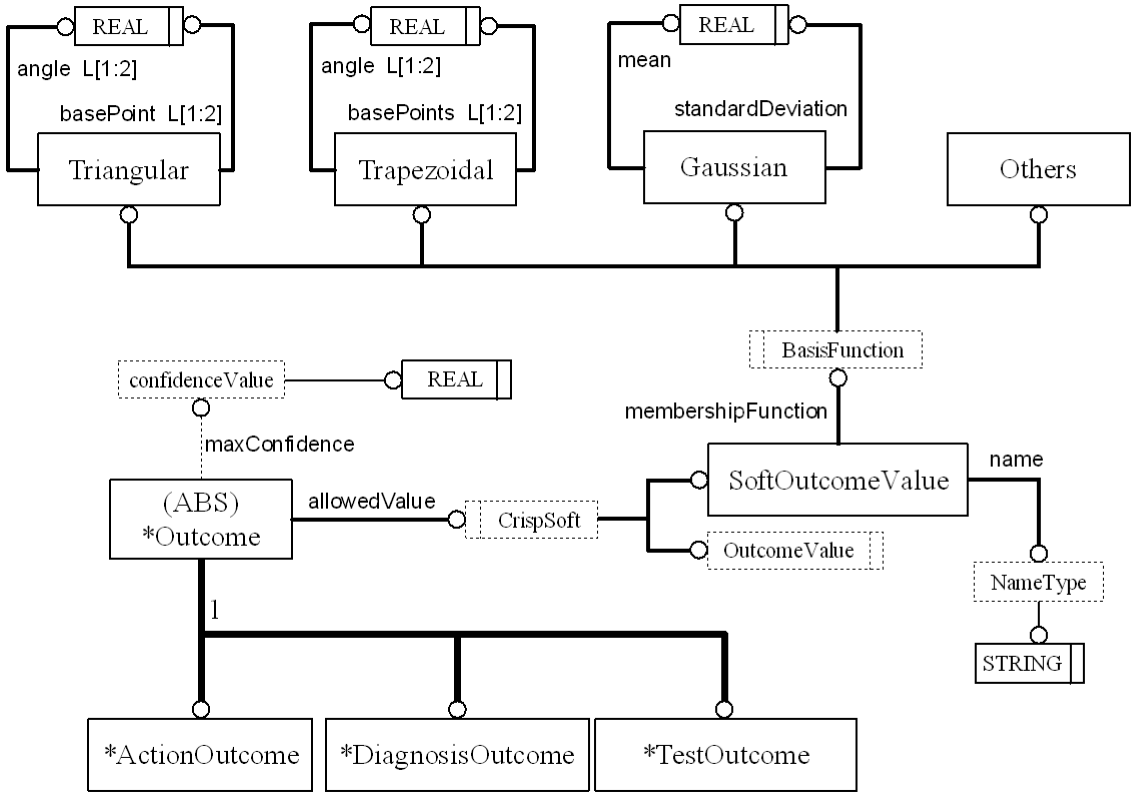

Fig. 3. Suggested modifications to the AI-ESTATE Common Element Model.

The proposed extension expands the allowedValue attribute of the Outcome entity by redefining allowedValue to reference a new SELECT type called CrispSoft, which allows the Outcome to instantiate either the current implementation of a single discrete OutcomeValue attribute or reference new a SoftOutcomeValue that represents individual gray-scale outcomes. The SoftOutcomeValue entity permits an Outcome to specify membership in one or more possible SoftOutcomeValues, each with a linguistic variable name. The associated degree of membership in each of the SoftOutcomeValues would be set at runtime. For ActionOutcomes and TestOutcomes, the degrees of membership would be set at the test application and passed to the reasoner. For DiagnosisOutcomes, the degrees of membership would be computed and returned by the reasoner.

The new entity SoftOutcomeValue contains two attributes, “name” and “membershipFunction.” The attribute “name” is of type NameType and represents the linguistic label represented by the soft outcome, such as *Failed*, *Degraded*, or *Good*. The names of these linguistic variables could be left entirely up to each implementation of the standard, or a set of possible names could be enumerated within the standard. The second attribute, “membershipFunction,” is defined using the BasisFunction SELECT type. This SELECT type associates a particular SoftOutcomeValue with one of the basis function entities. The basis functions model the degree of membership that real-valued outcomes have with the defined linguistic variables.

In the current version of the standard, the Outcome entity also contains an optional attribute maxConfidence of type Confidence. Because the semantics of this attribute were defined with respect to the original crisp outcomes, we proposed adding a rule to the standard such that, if CrispSoft specified soft outcomes, then this attribute would not be instantiated. This avoids the ambiguity that would arise from including a level of confidence with a soft outcome, seeing as the shape of the basis function already gives this property as a degree of membership.

Because instantiations of the Outcome entity occur at the subtype level, the soft outcomes extensions would be inherited by the three implementing subtypes. Soft DiagnosisOutcomes would permit flexible and finer-grained representation of gray-scale health information, while soft TestOutcomes and ActionOutcomes would also be able to handle continuous values for test measurements and other actions. The proposed extension would permit both discrete and continuous outcome types to be specified where needed within a model.

C. Basis Functions

The SoftOutcomeValue entity contains a SELECT type that associates the SoftOutcomeValue with a single basis function. Each type of basis function shown in Figure 3 contains common attributes that define its unique shape. For example, the triangular basis function requires an angle for each side and the length of the base. A Gaussian basis function, on the other hand, requires a mean and standard deviation. The Other entity is a placeholder to demonstrate that any number of other basis functions (e.g., sigmoidal or sinusoidal) could be defined within the standard.

The capability to represent and handle a variety of common basis functions permits flexibility of the standard to support any specific implementation using soft outcomes. For example, the triangular and trapezoidal functions are common membership functions used by fuzzy logic systems, while a hard limiter, linear threshold, and logistic functions are common activation functions in neural networks. A mixture of Gaussian functions could be used in a Bayesian network implementation that supports continuous random variables.

While the standard should include a number of general common basis functions, an additional user-defined basis function should be included to allow users of the standard to implement further methods for incorporating gray-scale health. Although the specification of this user-defined basis function has yet to be determined, including such an entity would permit users of AI-ESTATE to apply the standard extension mechanism to incorporate whatever basis function their implementation needed.

IV. Reasoning with Fuzzy Bayesian Networks

For the fuzzy Bayesian network, we want to be able to use fuzzy evidence instead of crisp, traditional evidence, as well as produce a fuzzy membership value for each diagnosis instead of a probability. We then use this fuzzy membership value as a level of degradation of that component, or gray-scale health. Due to this requirement, we cannot use the modified fuzzy Bayesian equation method. We then use the fuzzy probability distribution representation.

This method for using a fuzzy Bayesian network is presented in [13] and [14]. We use the core methodology presented in those works, with a couple of enhancements. The core process for evaluating these fuzzy Bayesian networks is the propagation of fuzzy states across the underlying Bayesian network from the evidence nodes to the query nodes.

A. Linear Collapse

The propagation of the fuzzy probability distributions across the network requires the enumeration of every possible state of every random variable. This causes the size of the fuzzy probability distribution to grow exponentially in size. To control this explosion, we use an approximation technique from [13] and [14] to limit the state size that these authors call linear collapse.

The linear collapse approximation takes a full fuzzy probability distribution and collapses it down into a single fuzzy state. This is done by taking each probability value represented in the fuzzy probability distribution and weighting it by the associated fuzzy membership value for the calculated state. These are then summed together for each state of the variable being propagated to, similar to the Bayesian update equations described previously.

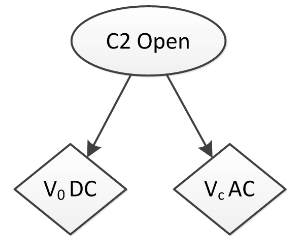

Fig. 4. Bayesian network structure used for example.

TABLE I
Conditional Probability Table for $V_C AC$ node

C2 Open $\rightarrow V_C AC$	P($V_C AC$ = *Pass*)	P($V_C AC$ = *Fail*)
C2 Open = *Good*	1	0
C2 Open = *Candidate*	0.5	0.5

TABLE II
Conditional Probability Table for $V_0 DC$ node

C2 Open $\rightarrow V_0 DC$	P($V_0 DC$ = *Pass*)	P($V_0 DC$ = *Fail*)
C2 Open = *Good*	1	0
C2 Open = *Candidate*	0.5	0.5

TABLE III
Conditional Probability Table for C2 Open node

P(C2 Open = *Good*)	P(C2 Open = *Candidate*)
0.9896	0.0104

We can represent this process mathematically as follows. The first step is to create a set, A, of all the combinations of the previous layers' nodes' states. For example, if we assume we start with the states for variables B and C, assuming that can either be true, t or false, f, then we get

$$A = \{\{B_t, C_t\}, \{B_f, C_t\}, \{B_t, C_f\}, \{B_f, C_f\}\} \quad (7)$$

Next we collapse the fuzzy probability distribution for variable Q with n states.

$$FS = \left[\sum_{i=0}^{|A|} P(Q_1 \mid A_i) \cdot \prod_{j=0}^{|A_i|} \mu\left(A_{i_j}\right), \ldots, \sum_{i=0}^{|A|} P(Q_n \mid A_i) \cdot \prod_{j=0}^{|A_i|} \mu\left(A_{i_j}\right) \right] \quad (8)$$

This process is performed after every step of the propagation process, thus keeping the state space limited to the space of the current variable. This is also the final step used to calculate the fuzzy membership values at the diagnosis node when the propagation process has finished.

B. Example Network

To illustrate the process of propagating fuzzy states, we use a simple network shown in Figure 4. This network has three nodes, the oval, *C2 Open*, is the query or diagnosis node, and the diamonds, V_0 *DC* and V_C *AC* are evidence nodes. The conditional probability tables for the given network are shown in Tables I, II, and III.

Assume we have fuzzy evidence at nodes $V_0\ DC$ and $V_C\ AC$. This is represented as the fuzzy states:

$$V_0\ DC = [Pass_{0.25}, Fail_{0.75}]$$
$$V_C\ AC = [Pass_{0.7}, Fail_{0.3}]$$

We propagate the fuzzy states from $V_0\ DC$ and $V_C\ AC$ to the nodes at the adjacent layer in the network. In this case, the node in the next layer is the diagnosis node, *C2 Open*. Assume there are two states for *C2 Open*, *Good*, and *Candidate*. For simplicity of notation, in our calculations *C2 Open* is represented with *C2*, $V_0\ DC$ is *V0* and $V_C\ AC$ is *VC*. The states *Good*, *Candidate*, *Pass* and *Fail* are represented as: *G*, *C*, *P* and *F* respectively. We propagate the fuzzy values by:

$$\begin{aligned} C2\ Open = \Big[& \{P(C2_G \mid V0_P, VC_P), \\ & \quad P(C2_C \mid V0_P, VC_P)\}_{\mu_{V0_P} \cdot \mu_{VC_P}}, \\ & \{P(C2_G \mid V0_P, VC_F), \\ & \quad P(C2_C \mid V0_P, VC_F)\}_{\mu_{V0_P} \cdot \mu_{VC_F}}, \\ & \{P(C2_G \mid V0_F, VC_P), \\ & \quad P(C2_C \mid V0_F, VC_P)\}_{\mu_{V0_F} \cdot \mu_{VC_P}}, \\ & \{P(C2_G \mid V0_F, VC_F), \\ & \quad P(C2_C \mid V0_F, VC_F)\}_{\mu_{V0_F} \cdot \mu_{VC_F}} \Big] \end{aligned}$$

We then calculate all these values to get the final fuzzy probability distribution:

$$\begin{aligned} C2\ Open &= \Big[\{0.9974, 0.0026\}_{0.25 \cdot 0.7}, \{0, 1\}_{0.25 \cdot 0.3}, \\ & \quad \{0, 1\}_{0.75 \cdot 0.7}, \{0, 1\}_{0.75 \cdot 0.3} \Big] \\ &= \Big[\{0.9974, 0.0026\}_{0.175}, \{0, 1\}_{0.075}, \\ & \quad \{0, 1\}_{0.525}, \{0, 1\}_{0.225} \Big] \end{aligned}$$

Now that we have the fuzzy probability distribution, we collapse it down to a fuzzy state by using the process from Equations 7 and 8 to get:

$$\begin{aligned} C2\ Open = \Big[& 0.9974 \cdot 0.175 + 0 \cdot 0.075 + 0 \cdot 0.525 + 0 \cdot 0.225, \\ & 0.0025 \cdot 0.175 + 1 \cdot 0.075 + 1 \cdot 0.525 + 1 \cdot 0.225 \Big] \\ = \Big[& 0.1746, 0.8254 \Big] \end{aligned}$$

Thus, after the propagation of the fuzzy evidence, the fuzzy membership value for *C2 Open* = $[0.1746, 0.8254]$.

V. Fuzzy Bayesian Network for Li-Ion Battery Degradation

To test our fuzzy Bayesian network implementation, we used a network that was trained on a data set from the NASA Prognostics Center of Excellence data repository [3]. This data set contains data of charging and discharging cycles of lithium-ion batteries at different temperatures. The data set contains values for charge current, charge voltage, battery temperature, battery voltage, battery current, as well as battery capacity. The network that resulted following training is shown in Figure 5. Parameters for the associated conditional probability tables were also learned from this data.

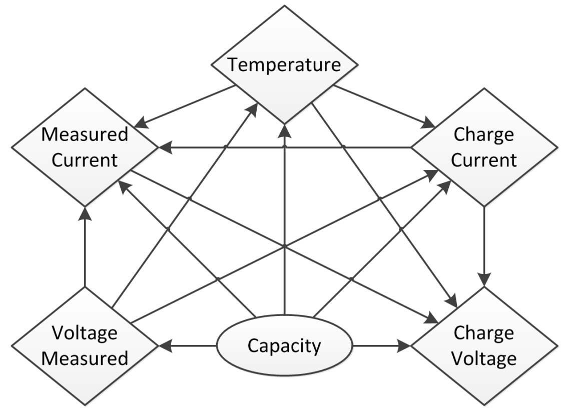

Fig. 5. Constructed Bayesian network for Li-Ion battery capacity degradation.

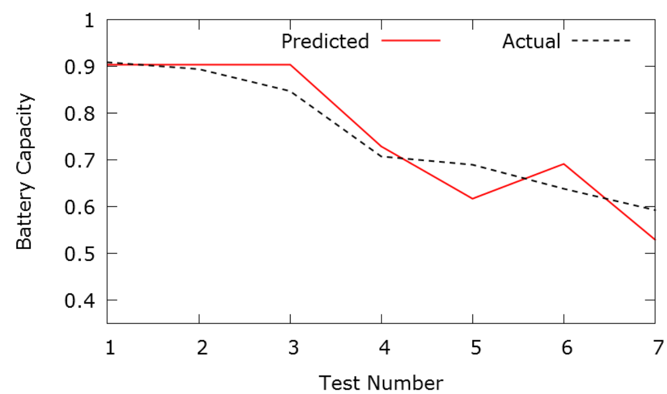

Fig. 6. Normalized Li-Ion battery capacity degradation testing.

The network has 5 evidence nodes and 1 diagnosis node. The diagnosis node, *Capacity*, has two states labeled *Good*, and *Candidate*. When we evaluate the query for the battery capacity diagnosis, a fuzzy state is returned for the node *Candidate*. We use the corresponding fuzzy membership values to represent degradation of battery capacity.

We applied several scenarios that correspond to degradation of battery capacity. For each set of measurements, there is a corresponding battery capacity value derived from actual accelerated life testing (provided in the dataset) that is used as a ground truth in our experiments. This capacity is a continuous value with a 2.0 amp-hour maximum capacity for a new battery, and as the capacity value decreases, the overall health of the battery decreases as well.

Our experiments used seven sample data points to represent battery capacity degradation. To compare the results from the fuzzy Bayesian network to the battery capacity truth we need to match their scales. Since we are interested in quantifying degradation, and we assume that the maximum capacity value is 2.0 amp-hour, we can normalize the given capacity value to be on the scale of $[0, 1]$.

Figure 6 plots this normalized battery capacity as well as the predicted degradation from the fuzzy Bayesian network at each test point. Notice that not only does the predicted value trend in the same direction as the actual battery capacity, the actual values are very similar to each other. This means that, at least in these test cases, a fuzzy Bayesian network can be used to estimate degradation of components within a system.

VI. Conclusion

In this paper, we demonstrated the usage of previously proposed modifications to AI-ESTATE to extend the ability of the standard to represent gray-scale health information. These modifications were demonstrated previously using fuzzy fault trees. We use an implementation of a fuzzy Bayesian network to illustrate the representative power of the previously proposed extensions.

We presented an implementation of a fuzzy Bayesian network designed to represent the gray-scale health information using soft outcomes both on test outcomes and diagnosis outcomes. We demonstrated the fuzzy Bayesian network implementation with a network learned from a data set from the NASA Prognostics Center of Excellence data repository [3] that we used to model battery capacity degradation of lithium-ion batteries. Not only were we able to demonstrate the advantages of the proposed modifications to the AI-ESTATE standard, but we were able to demonstrate the efficacy of the fuzzy Bayesian network in modeling degradation.

As future work we will continue to focus on evaluating the ability of the proposed changes to the AI-ESTATE standard to represent levels of degradation and gray-scale health. Additionally, we will perform more through evaluations and enhancements to the fuzzy Bayesian network implementation.

Acknowledgements

Several people and organizations have been involved in early developments of the soft outcome, fuzzy fault trees, and fuzzy Bayesian networks. We would like to thank the members of the Numerical Intelligent Systems Laboratory at MSU for their discussions and reviews of earlier versions of this work. We also thank Patrick Kalgren and John Gorton at Impact Technologies, as well as Michael Malesich and Jennifer Fetherman at NAVAIR Lakehurst for their insights as the work progressed. Finally, Nick Ryhajlo would like to thank members of his Master's committee, including Hunter Lloyd, Mike Wittie, and advisor John Sheppard, as they guided him in the completion of his MS project on Fuzzy Bayesian Networks for PHM.

References

[1] IEEE Std 1232-2010, *IEEE Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE)*. Piscataway, NJ: IEEE Standards Association Press, 2010.

[2] P. J. Donnelly, L. E. Sturlaugson, and J. W. Sheppard, "A standards-based approach to gray-scale health assessment using fuzzy fault trees," in *AUTOTESTCON, 2012 IEEE*. IEEE, 2012, pp. 174–181.

[3] B. Saha and K. Goebel, "Battery data set," *NASA Ames Prognostics Data Repository*, 2007.

[4] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[5] ——, "Probability measures of fuzzy events," *Journal of Mathematical Analysis and Applications*, vol. 23, no. 2, pp. 421–427, August 1968.

[6] H. Kwakernaak, "Fuzzy random variables I. definitions and theorems," *Information Sciences*, vol. 15, no. 1, pp. 1 – 29, 1978.

[7] ——, "Fuzzy random variables II. algorithms and examples for the discrete case," *Information Sciences*, vol. 17, no. 3, pp. 253 – 278, 1979.

[8] M. L. Puri and D. A. Ralescu, "Fuzzy random variables," *Journal of Mathematical Analysis and Applications*, vol. 114, no. 2, pp. 409 – 422, 1986.

[9] M. A. Gil, M. Lopez-Diaz, and D. A. Ralescu, "Overview on the development of fuzzy random variables," *Fuzzy Sets and Systems*, vol. 157, no. 19, pp. 2546 – 2557, 2006.

[10] A. F. Shapiro, "A fuzzy random variable primer," in *Actuarial Research Clearing House*, 2008.

[11] H. Tang and S. Liu, "Basic theory of fuzzy Bayesian networks and its application in machinery fault diagnosis," in *Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery - Volume 04*, ser. FSKD '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 132–137.

[12] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[13] C. Fogelberg, V. Palade, and P. Assheton, "Belief propagation in fuzzy Bayesian networks," in *1st International Workshop on Combinations of Intelligent Methods and Applications(CIMA) at ECAI'08*, I. Hatzilygeroudis, Ed., University of Patras, Greece, 21–22 July 2008, pp. 19–24.

[14] C. Fogelberg, "Belief propagation in fuzzy Bayesian networks: A worked example," in *Proceedings of the 2008 Oxford University Computing Laboratory Student Conference*, S. Faily and S. Zivny, Eds., October 2008.

[15] J. Luo and Z.-Z. Su, "Design and implementation of intelligent diagnostic system based on AI-ESTATE," in *Information and Computing (ICIC), 2011 Fourth International Conference on*. IEEE, 2011, pp. 237–240.

[16] L. A. Orlidge, "An overview of IEEE P1232 AI-ESTATE. The standard for intelligent reasoning based systems test and diagnosis arrives," in *AUTOTESTCON'96, Test Technology and Commercialization. Conference Record*. IEEE, 1996, pp. 61–67.

[17] ISO 10303-11:1994, *Industrial Automation Systems–Product Data Representation and Exchange–Part 11: The EXPRESS Language Reference Manual*. Geneva, Switzerland: The International Organization for Standardization, 1994.

[18] ISO 10303-28:2007, *Industrial Automation Systems–Product Data Representation and Exchange–Part 28: XML Representation of EXPRESS Schemas and Data Using XML Schemas*. Geneva, Switzerland: The International Organization for Standardization, 2007.

[19] M. A. Kaufman and J. W. Sheppard, "Bayesian modeling: an amendment to the AI-ESTATE standard," in *Autotestcon, 2005. IEEE*. IEEE, 2005, pp. 424–430.