

A Novel Ontology-Based Method to Represent and Classify Failure Modes of Sensors

Guoqi LI, Juan ZHANG

Science and Technology on Reliability and Environmental Engineering Laboratory,
School of Reliability and System Engineering, Beihang University (BUAA), Beijing 100191, China

Tel.: +86-13691465549

E-mail: gqli@buaa.edu.cn

Received: 16 September 2013 / Accepted: 15 October 2013 / Published: 23 December 2013

Abstract: It is generally recognized that sensors play important roles in modern control systems and failure modes of sensors are key points for safety and reliability analysis of such systems. Failure modes rely only on specific components rather than the system architecture, and therefore could be reused in different analysis. In this paper, we present a novel ontology-based method to represent and classify failure modes of sensors, so that they could be reused effectively and sufficiently. The principle and essence of the method are described in details appending a demonstration illustrating how to construct a concrete ontology with the method. Furthermore, the retrieve method based on semantic similarity is introduced. Additionally, a free software tool is developed for the method. The method and the tool could both be used in other fields with similar scenarios. *Copyright © 2013 IFSA.*

Keywords: Failure mode of sensors, Ontology-based method, Failure mode and effect analysis, Semantic similarity, Software tool.

1. Introduction

1.1. Failure Mode of Sensors for FMEA

As early as in 1949, procedures for conducting FMEA (Failure Mode and Effects Analysis) were described in US armed forces military procedures document MIL-P-1629 [1]. In 1979, Reifer firstly introduced FMEA into software analysis [2]. Nowadays, FMEA is generally appreciated in reliability engineering in industry fields.

There is a steady trend toward the use of digital systems of increasing complexity in safety-critical applications. As we all known, sensors play important roles [3] and are basic components in digital systems, especially control systems [4].

Snooke states that there are three causes of failure in digital systems [5]:

1. Abnormal value input to the software from its environment;
2. Failure in the hardware upon which the software is executed;
3. Logical/algorithmic/semantic error in the implementation code (a bug).

For the third one, software testing is the answer and the second one is easy to figure out, however, the first one, generally recognized, is a big challenge. It is mainly related to failure of sensors, because the systems depend on sensors to perceive the changes of the environment.

Consequently, failure modes of sensors are key point for FMEA of modern digital systems.

1.2. The Possibility and Necessity of Reuse

Procedures in doing FMEA, either hardware or software, are the same: [6]:

1. Identification of systems and functions;
2. Identification of failure modes;
3. Determination of effects of failure modes;
4. Identification of possible causes;
5. Documentation and risk reduction.

The second step of the procedure is very important. Failure modes, however, rely only on specific components rather than the system architecture, and therefore could be reused in different FMEAs. Other steps are specific with various of system designs and rarely reused.

In this paper, we present a novel ontology-based method to represent and classify failure mode of sensors, so that they could be reused effectively and sufficiently. In the following section, the principle and essence of the method are described. To demonstrate how to construct a concrete ontology with the method, a case study is given in the last subsection of section 2. Section 3 introduces the retrieve method based on semantic similarity and section 4 gives a software tool to aid the method. Finally, we draw conclusions and address our future works.

2. Ontology for Failure Mode of Sensors

2.1. Principle

Ontology describes a shared and common understanding of a domain that can be communicated between people and heterogeneous software tools. From the point of view of information modeling, ontologies make a commitment to an unambiguous representation of the concepts of a specific domain of discourse rather than to the structure of a data container. The objectives of developing ontology are in the following [7]:

1. To facilitate sharing/exchange of information and knowledge;
2. To support integration of tools;
3. To provide the same perspectives with collaborating teams and tools;
4. To create a common vocabulary;
5. To describe unambiguous definitions that both computers and teams can understand.

For safety analysis of critical systems, ontology-based methods are hot topics of recent years. In 2009, V. Ebrahimpour and etc. provided an ontology approach to support FMEA studies [7]. Main idea of the paper is applying ontology to formal the vocabularies used in the result tables of FMEA. In 2010, Martin Molhanec and etc. reported the Ontology based FMEA of lead free soldering process [8]. In the paper, a proposal of a user's guide for the FMEA procedure based on an ontology paradigm is presented.

Ptolemy II [9], which is an open-source software framework supporting experimentation with actor-oriented design, also resorted to ontology-based method. The Ptolemy II Ontologies package extends the Ptolemy II type inference engine to provide an efficient and scalable framework for general static analysis of arbitrary properties on a Ptolemy model [10].

What's we are most concerned now is, for failure mode databases, how to use ontology-based methods to reuse of failure mode? Failure modes are generally stored in databases and nearly every big company majored in safety critical systems have their own failure mode databases [11]. To redefine the failure modes in the existing databases is costly and impractical. Given the above facts, we provide a novel ontology-based method to annotate, rather than to formalize directly, the failure modes in existing databases, so that the knowledge of them could be reused.

2.2. Essence

In fact, we select the strategies applied in Gene Ontology [12], which are popular used in the research filed of bioinformatics.

Fig. 1 shows Gene Ontology and databases annotated by Gene Ontology. The terms in Gene Ontology were organized as a directed acyclic graph. Every node represents a concept. The solid line with arrow points to more concrete concept than its origin. Every term in Gene Ontology has an identification number, called GO ID (Gene Ontology IDentification), of the form GO:XXXXXXX, where X is a digit. When add a Gene Ontology term to an item in a table of a database, the item is assigned an ontological knowledge according to Gene Ontology. The technique of ontology makes databases annotated by Gene Ontology to an open, specified and computer-understandable knowledge base, which minimizes the gap between computational and biological researchers.

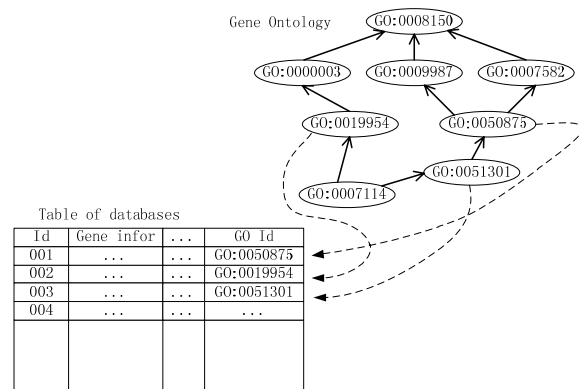


Fig. 1. Gene Ontology and databases annotated by Gene Ontology. The relationship between Gene Ontology and databases annotated databases are represented by dashed line with arrow.

The essence of the idea is to annotate, rather than to formalize, records in the database, so that it is easy to add semantic information, namely ontological knowledge, to existing databases. The scenario is similar with our requirements mentioned above, accordingly, we introduce the idea to management of failure modes [13].

2.3. Case Study

We define yoyo-ontology [14] as a directed acyclic graph. Each node represents a category and has three data fields: ID, Category and Properties. Nodes are connected with directed lines. Different heads of the lines, such as arrow, circle or others, mean different relationships between the nodes. The lines point to more concrete concepts than its origins. In fact, yoyo-ontology is created by imitating the mechanism of Gene Ontology [12]. Tbox is the directed acyclic graph and Abox is the annotated databases. Although yoyo-ontology is a light weight and informal method for building user defined ontology, it is effective and easy to be integrated with existing databases.

Fig. 2 shows the ontology built for description and classification of failure mode of sensors. The ontology is constructed with 3 layers. Layer I is aspect concept layer. The top node, whose "ID" is 1000 and "category" is "failure mode of sensor", has two aspect concepts: one is sensor and the other is failure mode, represented by the two child nodes respectively. Layer II is abstract concept layer, in which each top-level category is decomposed or refined as many abstract categories. For instance, in terms of input value, sensors could be divided into displacement sensors, pressure sensors, speed sensors, temperature sensors, gas sensors etc.; while in terms of mechanism, sensors could be divided into resistive, inductive, capacitive and potential type [14]. Similarly, failure modes could be decomposed as complete failure, partial failure, intermittent failure, failure over time and over-performance of function. Layer III is concrete concept layer, in which many abstract categories are combined or refined as categories of things exist in reality. In this layer, we define many sensors popularly used in contemporary automobile, e.g., variable gap inductive pressure sensor is a concrete concept and is defined by four nodes in abstract concept layer.

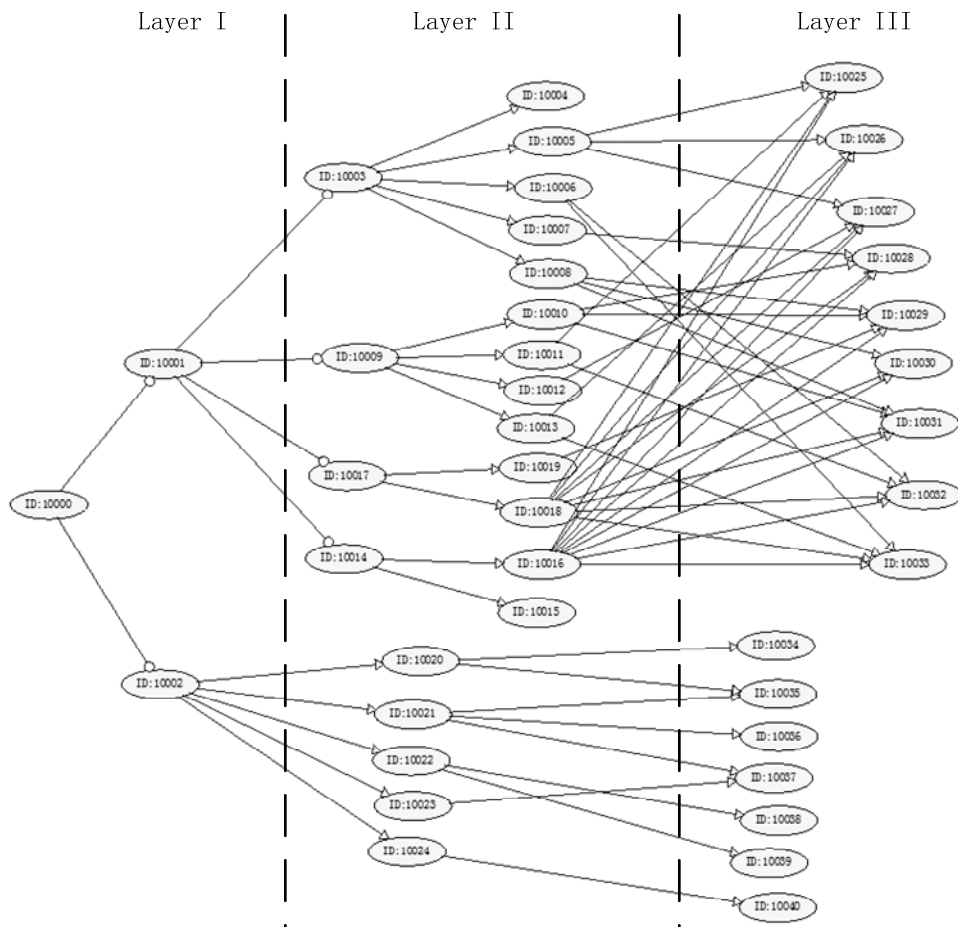


Fig. 2. An ontology for description and classification of failure mode of sensor. The graph is divided into 3 layers. In the figure, each node represents a category. Nodes are connected with directed lines. Different heads of the lines refer to different relationships between the nodes. Arrow ended line means relationship of "is-a" and circle ended line means "part-of". The lines point to more concrete concepts than its origins.

Here, we present a demonstration for construct a yoyo-ontology. For practical engineering, yoyo-ontology for failure mode of sensor is much more complicated and should be constructed by domain experts. The complete description of the ontology in Fig. 2 with detailed information of the nodes and their relationships is available by email.

After the ontology is constructed, the next step is to annotate the failure mode database. Select failure modes and corresponding identifications from the table of the target database and export the result-set as a flat file. The exported flat file, which is raw data set, has two columns of data separated with tabs. With yoyo-ontology, a free tool introduced in next section, the flat file could be loaded for annotation. For each line of the raw data set, if the failure mode belonging to a category, represented by a node, then annotate the ID of the node to the line. One line may be annotated with more than one node. The annotated nodes must comply with the two policies:

1. If a node is annotated to a line, then its ancestor nodes annotated to the line, if exist, should be deleted. The policy could be satisfied because we defined the ontology as a directed acyclic graph;
2. Annotated nodes for a line should cover all the aspects of the ontology and should come from the concrete concept layer, despite the fact that coming from the abstract concept layer is also allowed.

3. Retrieve Based on Semantic Similarity

For a set of annotated nodes A and B, we define the semantic similarity as:

$$\text{Similarity}(A, B) = 1 - \frac{\text{Card}(X - Y) + \text{Card}(Y - X)}{\text{Card}(X) + \text{Card}(Y)},$$

where X is the union of A and all the ancestor nodes of every member of A. The relationship of Y and B is

similar with X and A. Card(X) means cardinality of the set X, namely, the number of members of the set.

For the two sets, X and Y, $X - Y$ means a new set, each of whose member belongs to X but not belongs to Y, namely:

$$X - Y = \{x | x \in X, x \notin Y\}$$

Of course, other kinds of definition of the semantic similarity are possible. We just give an illumination and it has been implemented.

In order to facilitate the implementation of the proposed method and to prove the availability in practical engineering, a software tool, yoyo-ontology [13], is developed, which will be introduced in the following section.

4. A Tool

The no limited edition of yoyo-ontology 0.5 and further editions could be got from website: <http://yoyo-ontology.buaa.edu.cn>. We intended to open the source of the project, however haven't found suitable license. Anyway, we promise the software is free forever. In the following, the function and operation of the software will be described respectively.

Fig. 3 is the main interface of yoyo-ontology 0.5. The five buttons on the toolbar, labeled with "1", are operations on the files stored ontology; labeled with "2" are operations on the files of embedded database, labeled with "3" are used to edit the graphical model of the ontology and labeled with "4" is entry to the module of calculating semantic similarity of a specified item with others. There are two tab widgets in the main interface. The tab widget on the left side is graphic and XML editors of ontology and the right side are for embedded databases and output.

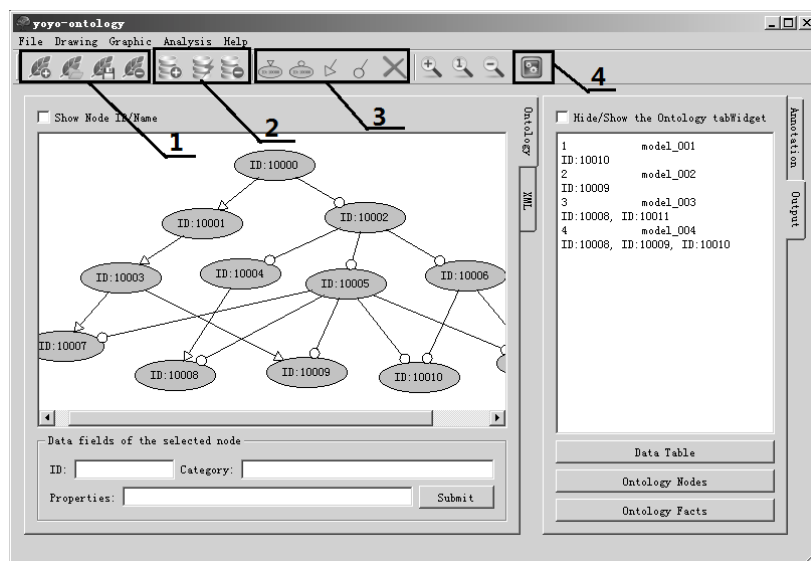


Fig. 3. The main interface of yoyo-ontology 0.5. The tab widget on the left side is graphic and XML editors of ontology and the right side are for embedded databases and output.

Three steps are needed to finish the annotation:

1. Load or create a database file using the buttons labeled with “2”. In the group box of “The raw data set”, upper right of the Fig. 3, click the “Edit” button and in the pop-up dialog box, the raw data set can be added or appended to the list in the “The raw data set”. Click a record in the list and its annotation will be listed in the list of “Annotation” group box, located below.

2. Load or create an ontology file using the buttons labeled with “1”. Edit the ontology with graphic tools labeled with “3”. Click a node of the graphic model of the ontology, in the group box of “Data fields of selected node”, lower left of the Fig. 3, the three data fields of the selected node will be shown and could be edited and submitted.

3. Select a record listed in the “The raw data set” and double click a node on the ontology, the node ID will be added to the list in the “Annotation” group box, lower right of the Fig. 3.

After finished all the annotation, the result could be output to flat files. Select the “Output” tab of the right tab widget in the main interface and set the check box of “Hide/Show the Ontology tabWidget”, you will see Fig. 4. Click one of the three buttons in the bottom of the figure and you will get one of the three outputs:

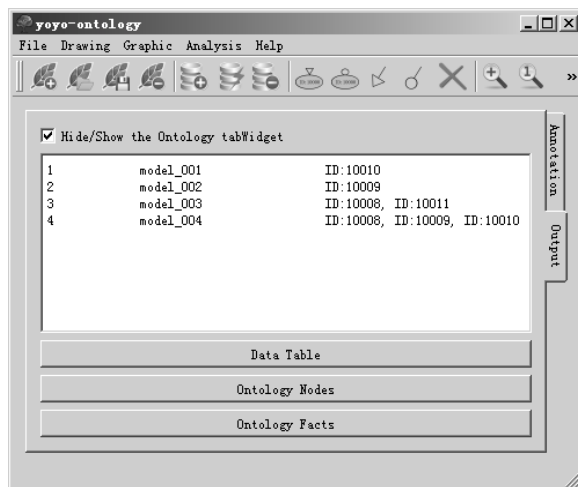


Fig. 4. Screenshot of the “Output” panel of the yoyo-ontology 0.5.

1. Annotated data set. The first two columns are similar with the raw data set and the third column is IDs of nodes of the ontology.

2. Ontology nodes. List all the data fields of all the nodes on the ontology. There are also three columns in the output flat file: first is node IDs of the ontology; second is the category; third are the properties.

3. Ontology facts. List all the relationships between the nodes of the ontology and they could be used as input to infer engines for semantically search or knowledge discovery.

To calculate the semantic similarity between a specific annotation and others defined in the former section, click the button labeled “4” in Fig. 2 and you will get the windows in Fig. 5. Select the specific annotation in the left side list and click the button “Search for Similarity”, then the result will be shown in the right side.

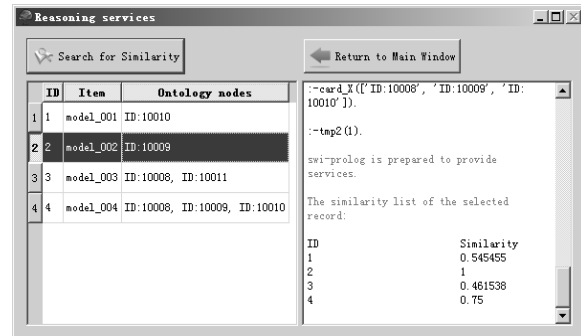


Fig. 5. Screenshot of the “Search for Similarity” dialog of the yoyo-ontology 0.5.

5. Conclusions and Future Works

In this paper, we present a novel ontology-based method to represent and classify failure modes of sensors, so that they could be reused effectively and sufficiently. The principle and essence of the method are described in detail with a demonstration to construct a concrete ontology with the method. Furthermore, the retrieve method based on semantic similarity is introduced. Additionally, a free software tool is developed for the method.

In the future we plan to develop and provide better designed ontologies of failure modes of sensors. In addition, we will continually improve yoyo-ontology on the precondition of unholding its lightweight features.

Acknowledgements

The no limited edition of yoyo-ontology 0.5 and further editions and the complete description of the ontology in Fig. 2 could be got from website: <http://yoyo-balance.buaa.edu.cn> or <http://fmeca.net> or by email: gqli@buaa.edu.cn or keepthinking@hotmail.com.

This work is supported by the Fundamental Research Funds for the Central Universities (Program No.YWF-13-B05-001).

References

- [1]. U. S. Department of Defense, Procedures for performing a failure mode, effects and criticality analysis, *US armed forces military procedures document MIL-P-1629*, Nov. 1949.

- [2]. D. J. Reifer, Software failure modes and effects analysis, *IEEE Transactions on Reliability*, Vol. R-28, Issue 3, 1979, pp. 247 – 249.
- [3]. I. Ullah, F. Ullah, Q. Ullah, and S. Shin, Sensor-Based Robotic Model for Vehicle Accident Avoidance, *Journal of Computational Intelligence and Electronic Systems*. Vol. 1, Issue 1, 2012, pp. 57-62.
- [4]. Y. She, Output Feedback Control of Electronic Throttle Valve, *Journal of Mechatronics*, Vol. 1, Issue 1, 2012, pp. 3-11.
- [5]. N. Snooke, Model-based failure modes and effects analysis of software, in *Proceedings of the DX04*, Carcassonne, France, June 2004, pp. 221 – 226.
- [6]. H. Pentti and H. Atte, Failure model and effects analysis of software-based automation systems, tech. rep., *STUK*, 2002.
- [7]. V. Ebrahimipour, K. Rezaie and S. Shokravi, An Ontology Approach to Support FMEA Studies, in *Proceeding of the Annual Reliability and Maintainability Symposium*, 2009.
- [8]. Molhanec, P. Mach and D. A. B. Mensah, The Ontology based FMEA of Lead Free Soldering Process, in *Proceeding of the 33rd International Spring Seminar on Electronics Technology*, 2010.
- [9]. D. H. Chang and E. A. Lee, A model-based design methodology for cyber-physical systems, in *Proceeding of the 7th International Conference on Wireless Communications and Mobile Computing*, 2011.
- [10]. C. Shelton, E. Latronico and B. Lickly, Static Analysis using the Ptolemy II Ontologies Package, in *Proceeding of the 9th Biennial Ptolemy Miniconference*, Berkeley, CA, 2011.
- [11]. W. M. Goble and J. V. Bukowski, Development of a mechanical component failure database, in *Proceeding of the Annual Reliability and Maintainability Symposium*, Orlando, FL, Jan, 2007, pp. 451 – 455.
- [12]. M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, G. Sherlock, and G. O. Consortium. Gene Ontology: tool for the unification of biology, *Nature Genetics*, Vol. 25, 2000. pp. 25-29.
- [13]. G. Li, Ontology-Based Reuse of Failure Modes in Existing Databases for FMEA: Methodology and Tool, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E96-A, No. 7, 2013, pp. 1645-1648.
- [14]. A. Ray and D. De, Intelligent Body Sensor Network for Pervasive Health Monitoring: A Survey, *Journal of Computational Intelligence and Electronic Systems*, Vol. 1, Issue 1, 2012, pp. 67-81.

2013 Copyright ©, International Frequency Sensor Association (IFSA). All rights reserved.
(<http://www.sensorsportal.com>)



**Sensors Industry
News**

**FREE Monthly
IFSA Newsletter**

ISSN 1726-6017

SUBSCRIBE NOW
subscribe@sensorsportal.com