Semantic Models and Rule-based Reasoning for Fault Detection and Diagnostics: Applications in Heating, Ventilating and Air Conditioning Systems

Parastoo Delgoshaei and Mark A. Austin
Department of Civil and Environmental Engineering,
University of Maryland, College Park, MD 20742, USA
E-mail: parastoo@umd.edu; austin@isr.umd.edu

Daniel Veronica
National Institute of Standards and Technology (NIST),
Gaithersburg, MD 20899, USA
E-mail: daniel.veronica@nist.gov

***Abstract*—This paper discusses an extensible model-based semantic framework for fault detection and diagnostics (FDD) in systems simulation and control. Generally speaking, state-of-the-art fault detection methods are equipment and domain specific. As a result, the applicability of these methods in different domains is very limited. Our proposed approach focuses on developing formal models (ontologies) across categories of domain-specific and domain-independent (time and space) phenomena. It then leverages inference-based reasoning over the ontologies for FDD purposes. Together, these techniques provide a semantic framework for the definition and evaluation of multidisplinary concepts relating to a system. FDD rules associated to those concepts are implemented as inference-based rules and are evaluated by a reasoner. We exercise the proposed method by looking at a FDD problem for heating, ventilating and air-conditioning (HVAC) systems simulation.**

Keywords-Fault Detection and Diagnostics; Ontology; Semantic; Rule-based; Heating Ventilating and Air-conditioning (HVAC) systems.

I. INTRODUCTION

A. Problem Statement

Automated fault detection and diagnostic (FDD) techniques provide mechanisms for condition-based maintenance of engineered systems (e.g., buildings, health monitoring, power plants and aviation systems). FDD is an automated process of detecting unwanted conditions (”faults”) in these systems by recognizing deviations in real-time or recorded data values from expected values, and then diagnosing the causes leading to the faults. Proper implementation of FDD can enable proactive identification and remediation of faults before they become significantly deleterious to the safety, security, or efficiency of the operating system.

During the last decade, considerable research has focused on the development of FDD methods for HVAC&R systems. This work has been driven, in part, by the historically less-than-optimal operation of many state-of-the-art HVAC systems. Today, degraded or poorly-maintained equipment accounts for 15 to 30 % of energy consumption in commercial buildings [1]. Approximately 50 to 67 % of air conditioners (residential and commercial) are either improperly charged or have airflow issues [2] and [3]. Faulty heating, ventilating, air conditioning, and refrigeration (HVAC&R) systems contribute to 1.5 to 2.5 % of total commercial building consumption [4]. Much of this energy usage could be prevented by utilizing automated condition-based maintenance. Yet, in spite of recent advances in building automation and control, automatic methods for FDD of building systems remain at a relatively immature stage of development. Present-day fault diagnostic approaches are domain dependent and semantic-free.

B. Objectives and Scope

In a step toward overcoming these limitations, this paper proposes a semantic framework, composed of ontologies and rules sets, for fault detection and diagnostic analysis of HVAC systems. Our work employs the Web Ontology Language (OWL) [5] and Jena API [6] for the development of semantic models for FDD applications. A semantic model of FDD defines it in terms of inference-based rules expressing conditions within formal, domain-specific ontologies (e.g., mechanical equipment, building, and weather). The remainder of this paper proceeds as follows: Section II contains a brief introduction to the uses of the Semantic Web and its enabling technologies. Section III explains different methods of FDD in building system applications. Section IV describes the proposed methodology and software infrastructure, and a simple example for fault detection in a leaking hot water valve. Sections V and VI provide a discussion of the next steps and conclusions of the work to date.

II. THE SEMANTIC WEB

A. Semantic Web Technology

The World Wide Web is almost thirty years old. Its initial mission was to provide a technical infrastructure for the representation of a “Web of documents and data” and presentation of data/content to humans [7]. In this infrastructure, machines are used primarily to retrieve and render information; humans are expected to interpret and understand the meaning of the content. A second, and much more ambitious, vision for the Web is support for semantic data structures, thereby allowing machines to access and share information, creating paths of machine-to-machine communications carrying semantic meanings instead of mere digital values. Realization of this goal requires mechanisms (i.e., markup languages) for the representation, coordination, and sharing of the formal semantics of data, as well as an ability to reason and draw conclusions (i.e., inference) from semantic data obtained by

 ISBN: 978-1-61208-547-0

following hyperlinks to definitions of problem domains (i.e., ontology models).

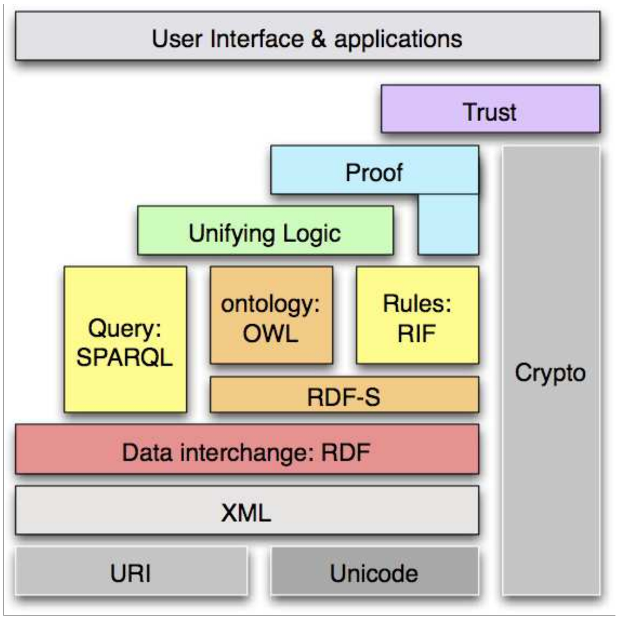

Figure 1. Technologies Used in Semantic Web Layers [8].

Figure 1 illustrates the layers of technologies supporting implementation of the Semantic Web [8]. Each higher layer extends, and provides compatibility with, layers of technology below it. The lower layers provide capability for addressing resources on the Web, linking documents, and integrating diverse forms of information. As a case in point, extended markup language (XML) enables the construction and management of documents composed of structured portable data. The resource description framework (RDF) allows for the modeling of graphs of resources on the Web. An RDF Schema (RDF-S) provides the basic vocabulary for RDF statements, and the machinery to create hierarchies of classes and properties. The Web Ontology Language (OWL) extends RDF-S by adding: (1) Advanced constructs to describe the semantics of RDF statements, (2) Vocabulary support for relationships between classes (e.g., class A is disjoint with class B), and (3) Restrictions on properties (e.g., cardinality). At higher levels of this stack, the ontology-based approach heavily relies on expressive features of the logic formalisms. For example, descriptive logic (DL) is the logical formalism for ontologies defined in OWL. Inference-based rules are rules that infer a new statement from existing statements. Inference-based rules rely on expressive features of the language they are defined in. Together, these features and language capabilities provide the foundations for reasoning - that is, deriving implicit conclusions not explicitly expressed in the ontology - using DL. In the Semantic Web, an inference engine gathers information from ontologies to infer the context that exists. Typically, the ontologies are defined in OWL or RDF-S.

B. Semantic Models

Semantic models consist of ontologies, graphs of individuals (specific instances), and rules derived from engineering models. An ontology represents the concepts of the domain (i.e., mechanical systems, building, weather, or occupant) as object classes, and the relationships between those classes as “Object Properties” (the connection between two objects of two classes). Moreover, the classes may have attributes that are stored as a simple data type “Datatype Properties”. RDF-S and OWL are examples of an ontology DL. They provide ways to define the semantic relationships between concepts in an application domain, as well as the various contexts possible in that domain. The goal is a consistent system of ontological classes, properties, and interrelationships expressing the application domain in a language translatable into machine readable code. Such a language provides a means for the machine to effectively understand and reason about the contextual information. A context may refer to people, building, time, weather and so on. The proposition underlying our work is that Semantic Web technologies could be used for FDD applications in building systems.

III. FDD FOR BUILDING SYSTEMS

Recent advances in building automation technologies provide a means for sensing and collecting the data needed for software applications to automatically detect and diagnose faults in buildings. During the past few decades a variety of FDD techniques have been developed in different domains, including model-based, rule-based, knowledge-based, and simulation-based approaches. Katipamula and Brambley summarizes FDD research for HVAC systems [1]. Their work also describes different fundamental FDD methods under the two main categories of model-based, and empirical (history-based) approaches. The major difference is in the nature of the knowledge used to formulate the diagnostics. Model-based diagnostics evaluate residuals between actual system measurements and *a priori* models (e.g., first principle models). Data-driven empirical strategies, on the other hand, do not require *a priori* models. The models used in model-based methods can be quantitative or qualitative. Quantitative models represent the requisite *a priori* knowledge of the system in terms of mathematical equations, typically as explicit descriptions of the physics underlying system components. Qualitative models, conversely, combine concepts such as descriptive “states” and “rules” into statements that are axiological instead of mathematical, expressing operational correctness or desirability through an axiology, a value system, appropriate to each physical application. As a result, the building system operation can be continuously classified as being either faulty or not faulty. Rule-based strategies are one example of qualitative model-based FDD methods. Rules can be based on first principles or they can be inferred from historical experiments, but in either case they represent expert qualitative knowledge that no purely quantitative representation could model. The first diagnostic expert systems for technical fault diagnosis were developed at MIT by Scherer and White [9]. Since then diagnostic systems have evolved from rule- to model-based approaches.

IV. METHODS

Our methodology entails an ontology-based, inference-based extensible framework to store and reuse data across different applications and domains. This semantic approach has been adapted in the area of healthcare [11], biology [12], [13], and transportation [14]. This section describes how this

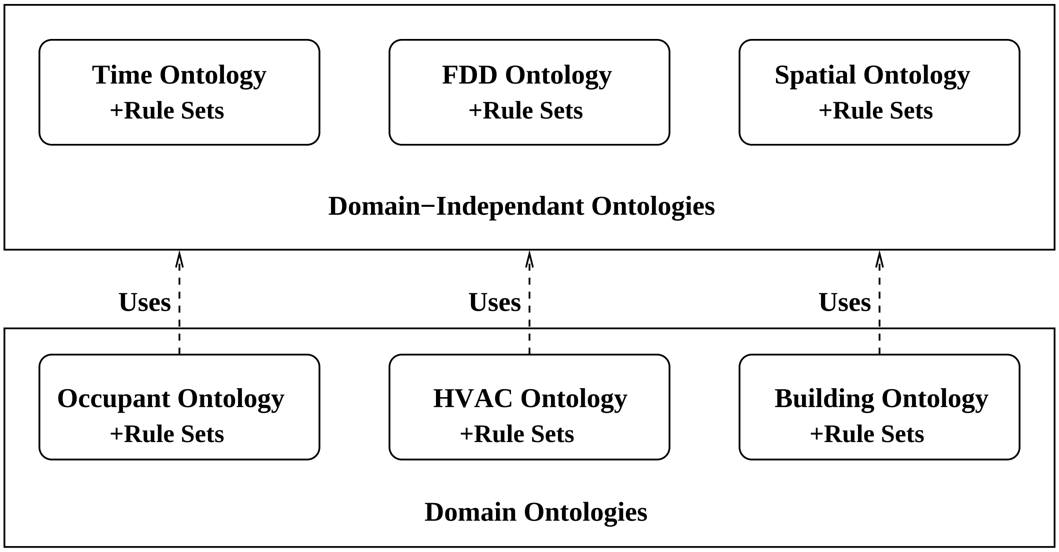

Figure 2. Domain specific and domain independent ontology structure .

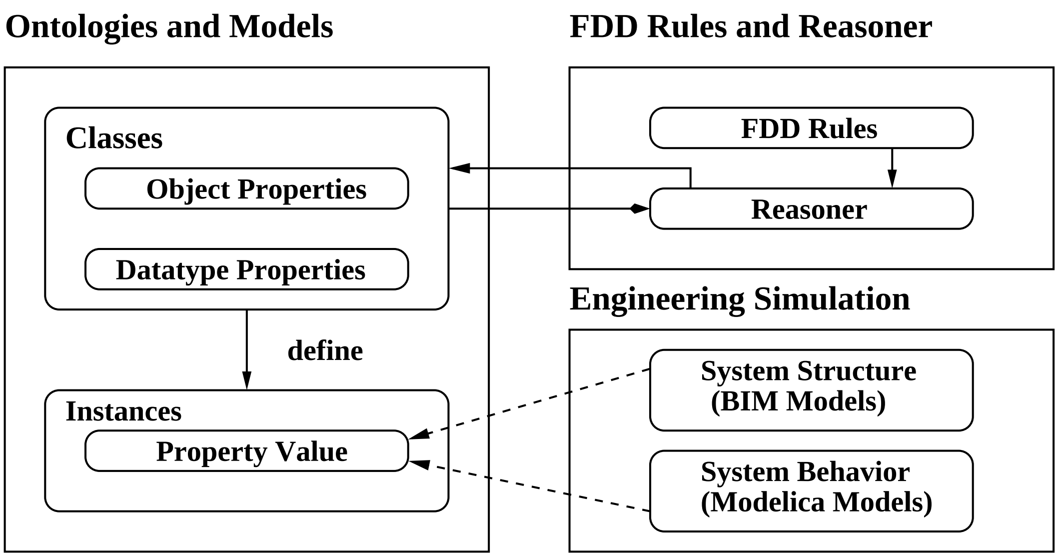

Figure 3. Architecture for coupled integrated semantic physical models in building simulations (Adapted from Delgoshaei, Austin and Pertzborn [10].

framework was utilized to formally model the concepts of the FDD domain.

A. Semantic Information Model

Figure 2 represents a semantic framework tailored for FDD and decision making within, and control of, engineered systems. Domain independent ontologies such as time, space, and FDD are represented in the upper half of the figure, and can be utilized in various engineering applications. For example, HVAC system ontologies along with their rule sets provide mechanisms to reason in time (e.g., if a measurement had occurred in a specific interval), deduce spatial information (e.g., determine if a room is in a specific zone or if a sensor is inside a room), and detect and diagnose faults (e.g., determine if a system fault is the result of leaking or stuck valve).

Figure 3 depicts the connection between the formal representation of a system and engineering models. On the building information modeling (BIM) side of the problem, a structural model of a building can be expressed in an ontology. As a case, Beetz and co-workers [15] have developed a converter to transform any format using EXPRESS schema, e.g., Industry Foundation Classes (IFC) into a Resource Description Framework (RDF) format. IFC is the standard used for BIM [16]. Moreover, to account for the behavior of the system, domain dependent ontologies (i.e., HVAC equipment) are based on models of the physical system and described in languages such as Modelica [17]. In this framework, the properties in the ontology represent the variables of the Modelica models that are updated at each time step. Ultimately, the real building sensors will provide the data to the ontologies.

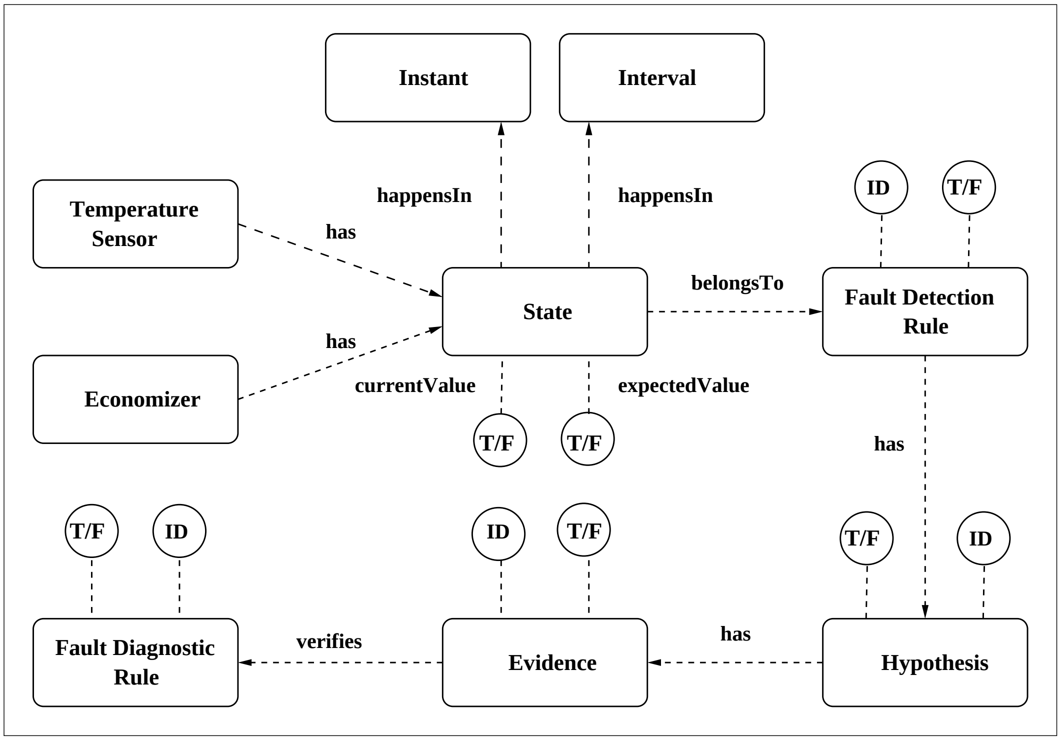

Figure 4. Domain specific and domain independent ontology structure.

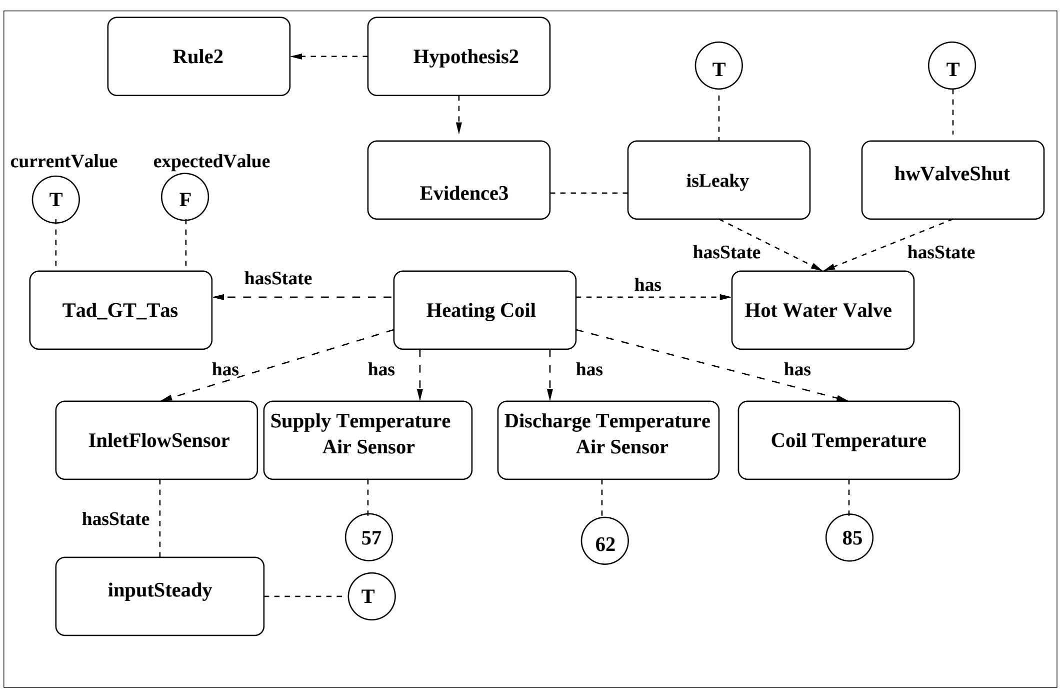

Figure 5. Subset of HVAC ontology at a specific instance of time.

TABLE I. Instances of states, hypotheses, and evidence for heating coil fault detections.

```
State        inputSteady
              hwValveShut
              Tad_GT_Tas
              isLeaky

Hypothesis1  HWVDFail       --> Hot Water Valve Drive Failure.
Hypothesis2  HWLeakValve    --> Hot Water Valve Leaking.
Hypothesis2  TadSensorFail  --> Temperature, air, discharge (Tad) sensor bad.
Hypothesis3  TasSensorFail  --> Temperature, air, supply (Tas) sensor bad.

Evidence3    Coil metal temperature is above Tas
```

Pseudo Jena Rules

```
Rule 1: Assignment rule:

[Rule1: (?c rdf:type eq:Coil) (?c eq:Tad ?v1) (?c eq:Tas ?v2) greaterThan(?v2, ?v1) ->
        (?c eq:Tad_GT_Tas "true" )  ]

Rule 2: Expectation rule:

[Rule2: (?c1 rdf:type eq:sensor) (?c1 FDD:inputSteady "true") (?c2 rdf:type eq:valve)
        (?c2  "hwValveCommandedShut''  "true'') -> (?c3 rdf:type eq:tempSensor) (?s3 rdf:type FDD:State)
        equal(?s3  "Tad_GT_Tas'') (?s3 FDD:expectedValue "false" ) (?s3 FDD:belongsTo "Rule2'' ]

Rule 3: Detection rule:

[Rule3: (?c rdf:type component) (?c FDD:hasState ?s)(?s  FDD:belongsTo ?r) (?s    FDD:expectedState ?es)
        (?s FDD:currentState ?cs) notEqual(?es ?cs) -> (?r FDD:isViolated "true") ]

Rules 4 and 5: Diagnostic rules:

[Rule4: (?r rdf:type rule2) (?r rdf:type evidence3) -> (?h rdf:type hypothesis2)

[Rule5: (?hwv rdf:type valve) (?hwv FDD:shut  "true") (?c rdf:type Coil)
        (?c eq:hasValve ?hwv) (?c FDD:metalTemp ?t1) (?c FDD:Tas ?t2) greaterThan(?t2  ?t1) ->
        (?hwv eq:isLeaky "true") (evidence3  "true" ) ]
```

Figure 6. Fault detection diagnostic rules for operation of a heating coil.

B. Support for Reasoning and Inference

Reasoning and inference are the powerful features of this semantic framework. The inferences are achieved through rules and a reasoner called upon by an engine, which executes the rules and updates the ontology with new inferences that may result.

Diagnostic procedures are required to have hypotheses of the underlying cause-effect relationships, and for our purposes these are represented by the FDD ontology and its rule sets. Figure 4 is a close-up view of the FDD ontology. The main concepts of the FDD ontology are "State", "Rule", ''Hypothesis", and "Evidence." These concepts are related to each other as object properties. Notice that the concept "Detection Rule" is related to the concept "Hypothesis" through the object property "has"; the concepts "Evidence" and "Hypothesis" are linked through the object property "verifies." Also notice that "State" has two boolean datatype properties, "CurrentValue" and "ExpectedValue." A few examples of different individuals in the class "State" include: temperature being within a specific band (T/F) and economizer mode in effect (T/F).

In the FDD ontology there are different categories for the rules. The first category of the inference rules is responsible for setting the current states associated with the system components (assignment). Some of these states will be computed based on function evaluations. As a case in point, a built-in function is called to perform the analysis and determination of the value for the boolean state "The temperature is going back to where it was."

The second category, expectation rules, are responsible for setting expected values for the states if certain conditions are met. In other words, if certain states (antecedent) of the ontology are true, then some other states of the ontology (consequent) are expected to also be true. As a case in point, if the outside temperature is within a specific range (T), then an economizer mode is expected to be in effect (T).

Lastly, the third category of rules is responsible for detecting and diagnosing the faults. The detection process is achieved by comparing the results of current values of a state with the expected values of a state. The diagnostics process is achieved by identifying what evidence holds true and as a result, which hypothesis accounts for the fault.

 ISBN: 978-1-61208-547-0

C. Fault Detection for a Leaking Hot Water Valve

Table I summarizes the list of examples for the class of the FDD concepts shown in Figure 5, a subset of the HVAC ontology with the values of individuals stored in the ontological graph at a specific time. Specifically, it represents a case where a fault has occurred in the system due to a leaking hot water valve.

The execution of rules in the heating coil operation involves four steps. Step 1, the assignment rule: If the discharge air temperature is greater than the supply temperature, then Tad_GT_Tas is set to true, e.g., Rule 1. Step 2, the expectation rule: If the unit has been operating steadily for a specified interval, and the hot water valve is shut, the mean value of discharge air temperature is expected to be less than or equal to the mean of supply air temperature (e.g., Rule 2). Step 3, the detection rule: When the current value of a specific state is not equal to its expected value, then the associated rule is violated (e.g., Rule 3). The final step is a diagnostic rule: If the coil metal temperature is above Tas twenty minutes after the hot water valve is manually driven shut, then the shut valve is still leaking. This presents a significant use of heating energy.

While it is perfectly reasonable to expect that the state variable values will change as a function of time, the expected values in a specific rule will stay constant, and act as the point of reference for detecting faults. As an example, a rule is defined to detect whether a specific piece of HVAC equipment is responding properly to conditions in the rooms it serves. If the data ordinarily sampled do not indicate the equipment is making the proper response (i.e., it “fails” the rule), the engine calls on the reasoner and expert knowledge to use deeper, more extraordinarily obtained data (evidence) to infer a cause (hypothesis) for the improper response.

Figure 6 shows the pseudo Jena rules described over the FDD, Equipment (eq) ontologies. The current value for the state Tad_GT_Tas is determined based on Rule1. Rule 2, sets the expected value for Tad_GT_Tas when the valve is shut off and will set the associated detection rule (rule2). Rule 3 detects the fault. Rule 4, asserts if Evidence3 holds, Hypothesis 3 that the valve is leaking holds true. Rule 5, describes how the values for flow and temperature sensors will determined if Evidence3 holds true.

V. Discussion

The proposed approach is a first step in the development of a model-based semantic framework for FDD. State-of-the-art techniques for FDD in buildings lack real-time rule-checking. Consequently, there is always a lag between the fault detection and potential diagnostics and decision-making. Our position is that developing ontologies for different categories of domain-specific and domain-independent faults facilitates FDD in HVAC system simulation and controls.

VI. Conclusion and Future Work

This paper demonstrates a model-based semantic framework for automated fault detection and diagnostics (FDD) for application to building controls, specifically heating, ventilating and air-conditioning (HVAC). Our preliminary results are promising and indicate that future building control strategies could utilize formal models (ontologies and rules) for the detection of a variety of types of fault. To make building simulations and fault diagnostic procedures more realistic, however, our capability needs to be extended toward real-time simulation and decision making. Our plans are to deploy the proposed method for an actual case study problem, and use the buildings library Modelica models.

VII. Acknowledgment

The first author was supported by a fellowship award from the NIST Graduate Student Measurement Science and Engineering (GMSE) Program.

References

[1] S. Katipamula and M. R. Brambley, “Review Article: Methods for Fault Detection, Diagnostics, and Prognostics for Building SystemsA Review, Part I,” HVAC&R Research, vol. 11, no. 1, 2005, pp. 3–25.

[2] J. A. Siegel and C. P. Wray, “An Evaluation of Superheat-based Refrigerant Charge Diagnostics for Residential Cooling Systems/Discussions,” ASHRAE Transactions 108(1), 2002, p. 965.

[3] W. Kim and J. E. Braun, “Impacts of refrigerant charge on air conditioner and heat pump performance,” in Impacts of Refrigerant Charge on Air Conditioner and Heat Pump Performance, July 10–15 2010, pp. 2433–2441.

[4] M. Wiggins and J. Brodrick, “Emerging Technologies: HVAC Fault Detection,” ASHRAE Journal, April 2012, pp. 78–80.

[5] OWL:, “Web Ontology Language Overview, W3C Recommendation from February, 2004. For details, see http://www.w3.org/TR/owl-features/ (Accessed, April 2017).”

[6] Apache Jena, “An Open Source Java Framework for building Semantic Web and Linked Data Applications, Accessible at https://jena.apache.org (Accessed on 12/12/16),” 2016.

[7] T. Berners-Lee, J. Hendler and O. Lassila, “The Semantic Web,” Scientific American, May 2001, pp. 28–37.

[8] L. Feigenbaum, 2006, Semantic Web Technologies in the Enterprise.

[9] W. T. Scherer and C. C. White, A Survey of Expert Systems for Equipment Maintenance and Diagnostics. Boston, MA: Springer US, 1989, pp. 285–300.

[10] P. Delgoshaei, M. A. Austin and A. Pertzborn, “A Semantic Framework for Modeling and Simulation of Cyber-Physical Systems,” International Journal On Advances in Systems and Measurements, vol. 7, no. 3-4, December 2014, pp. 223–238.

[11] T. Q. Dung and W. Kameyama, Ontology-based Information Extraction and Information Retrieval in Health Care Domain, ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2007, vol. 4654 LNCS, pp. 323–333.

[12] C. Taswell, “DOORS to the Semantic Web and Grid with a PORTAL for Biomedical Computing,” IEEE Trans Inf Technol Biomed, vol. 12, no. 2, 2008, pp. 191–204.

[13] P. Lord, S. Bechhofer, M. D. Wilkinson, G. Schiltz, D. Gessler, D. Hull, C. Goble, and L. Stein, Applying Semantic Web Services to Bioinformatics: Experiences Gained, Lessons Learnt. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 350–364.

[14] D. Corsar, D. Milan, P. Edwards, and J. D. Nelson, The Transport Disruption Ontology. Lecture Notes in Computer Science, vol 9367, Springer, 2015, pp. 329–336.

[15] J. Beetz, J. van Leeuwen and B. de Vries, “IfcOWL: A Case of Transforming EXPRESS Schemas into Ontologies,” Artificial Intelligence for Engineering Design, Analysis and Manufacturing, vol. 23, no. 1, 2009, pp. 89–101.

[16] C. M. Eastman, P. Techolz, R. Sacks and K. Liston, BIM Handbook: A Guide to Building Information Modeling. Hoboken, NJ: John-Wiley and Sons, 2008.

[17] P. Fritzson, Principles of Object-Oriented Modeling and Simulation with Modelica 2.1. Wiley-IEEE Press, 2003.

 ISBN: 978-1-61208-547-0