The Egyptian International Journal of Engineering Sciences and Technology

ISSN:1687-1006

Vol. 24 (January 2018) 09–17

http://www.eijest.zu.edu.eg

Unified Modelling Language–Adaptive Neuro-Fuzzy Inference System Models for Understanding the Design Complexity of Oil/Gas Pipeline Intrusion Detection System

Francis A. U. Imouokhome[a*], Emmanuel A. Onibere[b]

[a,b]Department of Computer Science, University of Benin,P. M. B. 1154, Benin City, Nigeria.

ARTICLE INFO

Article history:
Received 17 July 2017
Received in revised form 27 October 2017
Accepted 29 October 2017
Available online 07 June 2018

Keywords:
Vandalism,
Vibration signature,
Intrusion,
Complexity,
Fault Detection,
UML

ABSTRACT

Oil/gas pipeline vandalism is a common and regular occurrence in the oil-producing regions of Nigeria. To ameliorate on the efforts so far made to combat this menace, a proactive system that can detect and identify damage-causing forces on oil/gas pipelines becomes very imperative. Through the use of the Unified Modelling Language (UML), this paper presents models that describe a real-time, intelligent, but complex system that detects, discriminates between varied vibration signatures of oil/gas distribution pipelines, and identifies the signatures due to intrusion on the pipes by vandals. To enhance understanding of the nature and interconnection of its constituent subcomponents, dynamics and design complexities, the system under study is modelled by use of hierarchical activity and state transition diagrams from the Unified Modelling Language (UML) domain. Because of the qualitative nature of its inputs, the system is integrated with Adaptive Neuro-Fuzzy Inference System (ANFIS) to intelligently handle the imprecision in the representation of the input data in human language. Implementation of the system modelled in this study would significantly reduce oil theft/spillage, destruction of properties, and loss of human lives hitherto experienced in the oil-producing regions of Nigeria.

1. Introduction

Vandalism of oil/gas pipeline is a common and regular occurrence in the oil-producing regions of Nigeria. At present, no effective devices or systems are in place to monitor the pipelines and report to appropriate authorities, any cases of intrusion on the pipes by vandals. What is needed to arrest this situation is an intelligent system that can sense, detect and identify vibration signature of the pipe while the act of vandalism is on. Thus intrusion would be detected and breakage of the pipe forestalled because the act would be identified before damages are caused on the pipes. This paper presents models that describe various modules of an intelligent system for detecting and identifying intrusion on oil/gas pipe, and reporting the act to the appropriate authorities in real time. The system classifies the vibration signals into patterns that enable it to discriminate between forces that are due to acts of vandalism and those that are not. Figure 1 shows the architecture of the system adopted for this study, where r(t) is the reference vibration of the pipe under normal condition of fluid flow; d(t) and n(t) are the forces due to external disturbances and noise respectively; and e(t) is the fault symptom due to intrusion on the pipes by vandals. The problem that the system modelled in Figure 1 tries to solve is a

* Corresponding Author. Tel.: +234-706-228-9738
e-mail address: franc.imo@uniben.edu

real world system problem. In addition to imprecise prediction of their behaviours, real-world systems by their nature are composed of many collaborating subsystems, and are fraught with incomplete and vague information about them. These characteristics make real world systems complex.

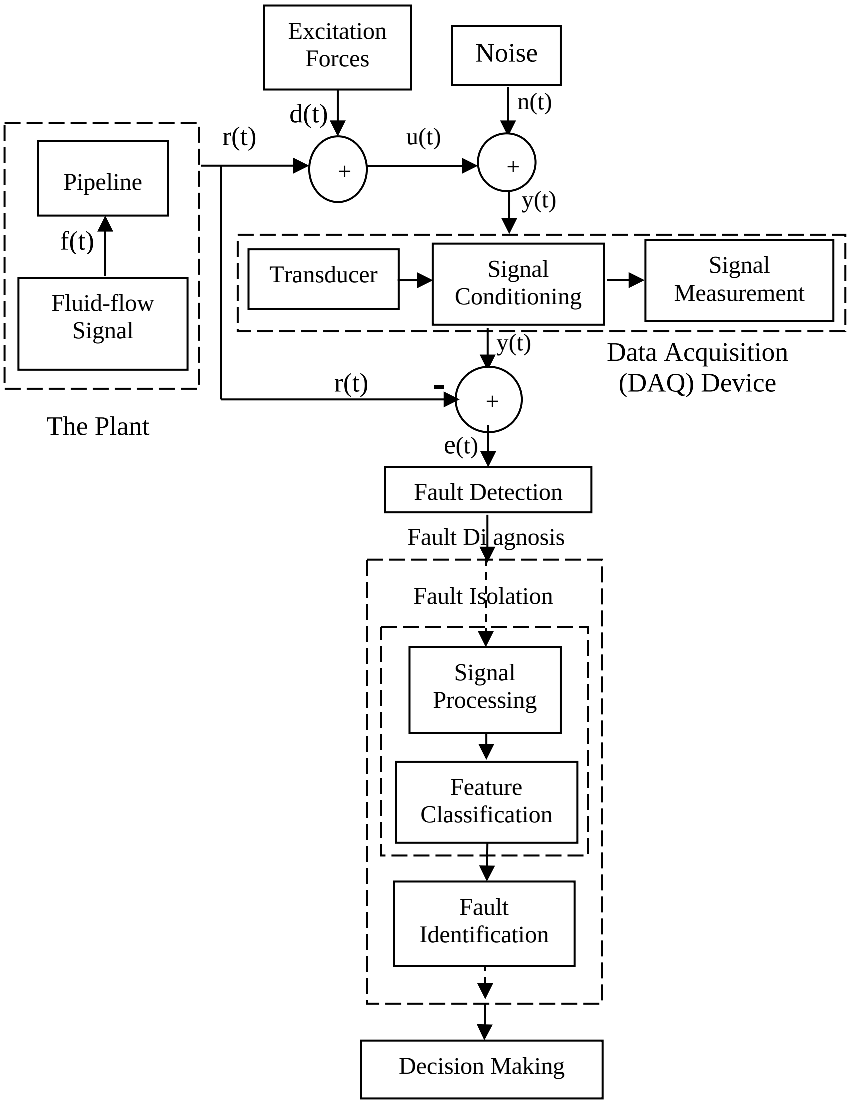

Fig. 1. Pipeline Vibration Diagnostic System Architecture (Onibere & Imouokhome [1])

2. Materials and Methods

A complex system is described as one whose overall activity is not obtained by mere linear summation of the activities of the individual interacting components that the system is made of [2,3,4]. According to [5], the imprecision or vagueness of information regarding the state and behaviour of the system is caused partly by the many interacting components. In order to reduce complexities in the design of a system, it becomes very necessary to have a very good understanding of the nature and interconnection of the constituent subcomponents of the system, their influence on one another, and the possible outcomes of the dynamic processes that the system would undergo. Reduction of the complexity of a system results in hierarchically structured and simplified architecture of the system [6] through control techniques that guarantee that its high-level objectives are implemented by the lower level subsystems [7].

To enhance understanding of its dynamics and design complexities, the system under study is modelled by use of hierarchical activity and state transition diagrams from the Unified Modelling Language (UML) domain. Because of the qualitative nature of its inputs, the system is integrated with Adaptive Neuro-Fuzzy Inference System (ANFIS) to intelligently handle the imprecision or vagueness in the input data as humans would do. A complex system is usually integrated with an intelligent human subsystem that is responsible for formulating the system requirements and decides what the system does, etc.

The Unified Modeling Language (UML) is a multi-purpose object-oriented software-modelling language [8] that matches the growing complexity of real-time systems. Software architecture based on UML models helps in handling complexities of a system and in understanding its domain. UML models enhance communication among analysts, designers, developers, testers and domain experts in their understanding of the process of designing and developing software. UML is an aggregation of different diagram types, such as Functional diagram, Structural diagrams, Interaction diagrams, and Behavioural diagrams (e.g., State diagrams, Activity diagrams, etc.) used for different purposes[9,10,11]. It is a language of communication between programmers, analysts, and designers as they represent their ideas, models, etc., in pictures or diagrams with some textual data. For example various UML diagrams like the use-case diagram, activity diagram, class diagram, etc., were used to portray the prototype intrusion detection system reported in [9].

A methodology called Maude language, which is a synergy of UML Activity Diagrams and Rewriting Logic language, with the aim of drawing from the strengths of both approaches, was presented in [12]. The authors reported that the result of their work is a specification that can be used to transform global dynamic behavioural properties of systems that are expressed in UML models into their corresponding Maude specifications for verifying the system properties.

Obaid et al [13] proposed the use of action language for fUML (ALF) in the design of a state-based ALF model as a tool for the analysis of executable UML models with a view to authenticating the anticipated correctness (predictable behaviour) of a system from its models. The tool describes identification of both the precision of flow of information through hierarchical and flattened UML models, and the influence of system states towards improving the correctness of the system. This is similar to the state-based system under study in this paper, which aims at analysis of large vibration signal data and decides on the features that best fit or match those in the predefined threshold data range. A match is an indication of the occurrence of an intrusion into the pipeline, causing an alarm system to trigger to alert operators of the pipeline to take appropriate action.

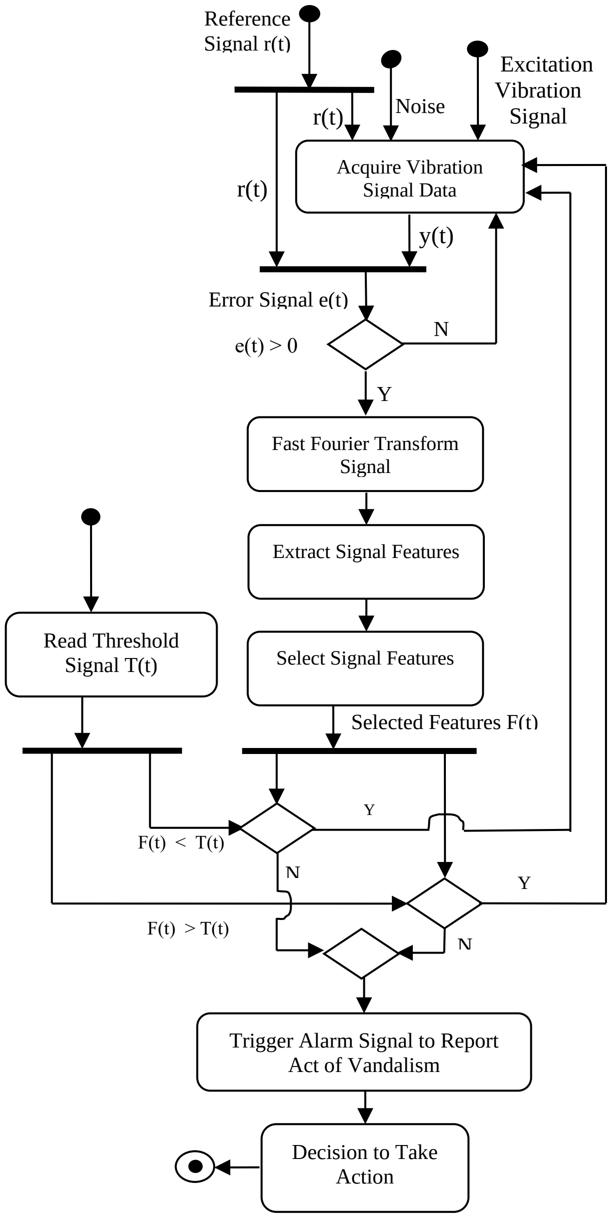

Fig. 2. Activity Diagram of Vibration Signature Detection and Identification System

The UML state diagram on its part was invented as an extension of the traditional state-transition diagrams with the intention to introduce the concept of hierarchy and concurrency into the diagrams [14]. State diagrams are used as a method of describing the behaviour of systems (or sub-systems thereof) as they change their states in reaction to internal or external stimuli (i.e., events).

UML behavioural diagrams are hierarchical representations that precisely but adequately describe the quantitative and qualitative analysis of the functioning and states dynamics of the system under study. This has the advantage of simplicity and ease of understanding the flow of the logic of the system (Figure 1) as modelled in the activity diagrams of Figures 2 and 3.

Figure 2 represents the process of vibration signature analysis which begins with the acquisition of signal data from the vibrating object, followed by processing of the acquired data to extract useful features from them, and selecting and classifying the features to determine the patterns established from the vibration. The patterns are evaluated with the characteristics of normal operations of the system with the aim to determine the existence of a fault (i.e., abnormality in the vibration of the object). Figure 3 is a swimlane diagram showing the flow of activities in Figure 2. It is useful in dividing the system model into activity state partitions. Each swimlane in Figure 3 has a name that reflects the activity within the lane. Figure 3 therefore gives a clear explanation of the activities described in Figure2.

3. Theory/Calculation

From control point of view, systems are analysed by formulating models of the variables that capture the dynamic behaviour of the system [15]. Basically, a model is a conceptual, visual, or analytical representation of a system, which describes a "limited part of reality with related elements" [16]; analytical representation being practical but useful mathematical expressions for representing real-world systems with the aim of predicting the dynamics or behaviour of the systems in a systematic manner [17]. The process of modelling concerns itself with obtaining knowledge of the fundamental components interacting together in a system, and organizing the knowledge in a manner that meets the specific requirements needed to model the system in well-defined and clear mathematical equations [18,19]. This process, according to Cassandras and Lafortune [20], starts with the definition and selection of measurable time-varying input-output variables (t_o, t_f) that are associated with the given system. The variables described by the time function of equation (1), represent the input variables to the system.

$$\{u_1(t),\ldots, u_m(t)\} \qquad t_o \le t \le t_f. \tag{1}$$

where $\{u_1(t),\ldots, u_m(t)\}$= input variables at time (t);
t_o = initial time; and
t_f = final time.
By measuring the input variables over a period of time (t_o, t_n), data are collected for use as input into the system.

While varying the input data in the range $u_1(t)$,,$u_m(t)$ over the same time period (t_o, t_f), another measurable set of variables are selected. These, according to [18], represent the system output variables described by the time function of equation (2).

$$\{u_1(t),\ldots, u_n(t)\} \qquad t_o \le t \le t_f. \tag{2}$$

where $\{u_1(t),\ldots, u_m(t)\}$= output variables at time (t).
The output variables describe the system behaviour (i.e., its response to stimuli) provided by the selected input functions. Knowledge about the behaviour of systems (and/or the control objectives and constraints) may be available in mathematical (quantitative) and/or in qualitative (i.e., descriptive) forms. Consequently, solutions to a complex system problem require the formulation of methods for the qualitative examination of the system's subcomponents dynamics and the construction of suitable techniques to ensure that they are effectively and efficiently controlled [3].

The overall vibration signal (y(t)) measured using the system under study (see Figure 1) comprises of the reference signal (r(t)), externally-induced disturbance signal (d(t)) and noise signal (n(t)) due to electromagnetic flux from the surrounding environment. Equation (3) is a mathematical relationship between these parameters.

$$y(t) = r(t) + d(t) + n(t) \tag{3}$$

where y(t) = Overall vibration signal;
r(t) = Reference signal;
d(t) = Disturbance (i.e., externally-induced) signal; and n(t) = Noise signal.
To determine the existence of a change in the vibration of the pipe, y(t) is compared with the reference signal (r(t)), which represents the normal vibration of the pipe due to fluid flow in it. A deviation from r(t) is an indication of the existence of a fault. This process is expressed mathematically in equation (4).

$$e(t) = y(t) - r(t) \tag{4}$$

where e(t) = Error signal
The error signal is fed into the signal processing module of the system.

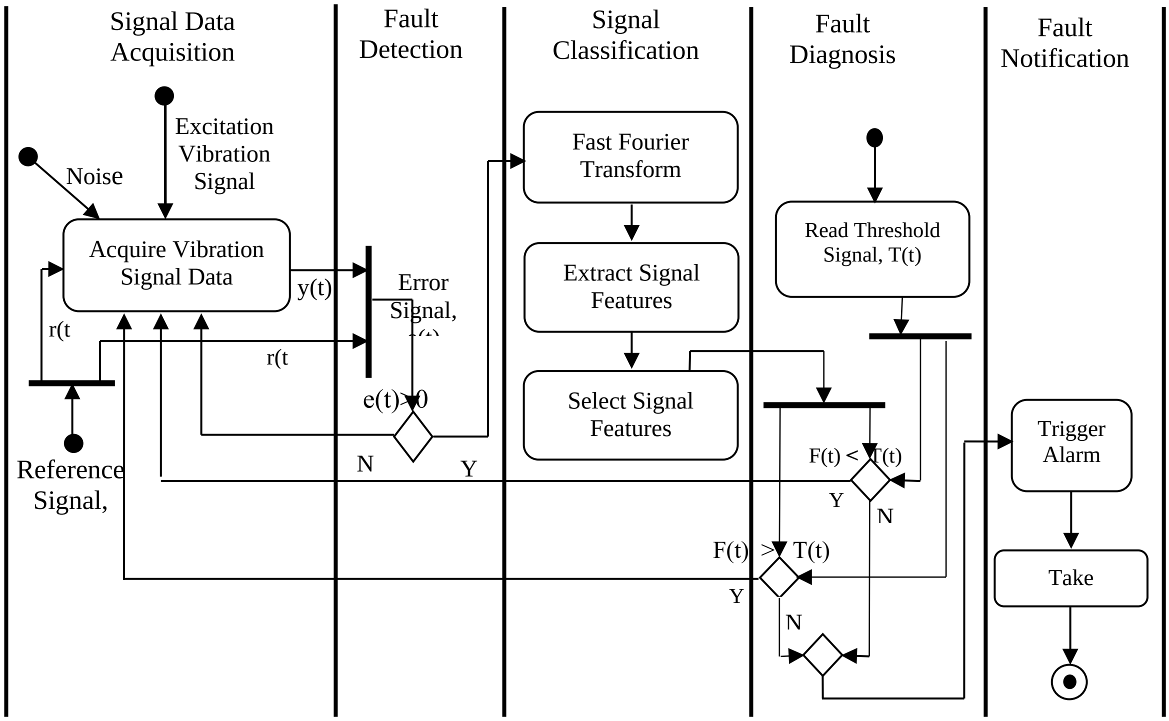

Fig. 3. Activity Diagram with Five Swimlanes Showing Control of Activities in the System

4. Data Acquisition, and Signal Processing and Classification

Pipes vibrate when fluids flow through them causing displacements in their internal structure. These displacements, in the form of vibration, are converted to electrical energy by use of transducers, and the signals are measured and analysed as reflected by the Unified Modelling Language (UML) Activity Diagrams of Figures 2 and 3. These processes are further expatiated upon in the following sections.

4.1 Data Acquisition Module

To acquire vibration signal data from an experimental or actual functioning oil/gas pipeline, an appropriate Data Acquisition (DAQ) device is connected to the pipe through a sensor, and the values measured by the device are read from a computer connected to it. The sensor transforms the vibration signal into electrical energy that is measured by the DAQ as the response of the pipe to excitation due to fluid flow in it or external stimuli. The equipment also senses and measures noise signals (due to electromagnetic flux) from the surrounding environment.

As shown in the UML state diagram of Figure 4, vibration signals from any of the above sources would cause a change in the state of the pipe, causing the DAQ to measure the signal data. The DAQ transitions through various states in the process of reading the vibration of the pipe in response to stimuli. The DAQ is idle when no signals flow into it for measurement.

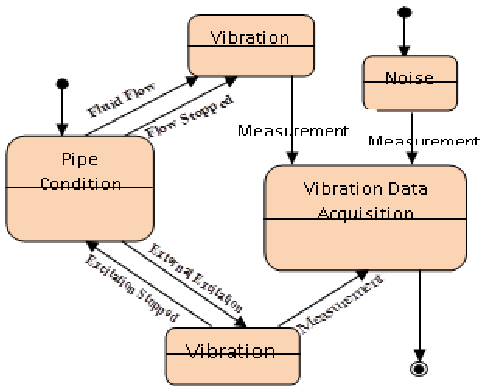

Fig. 4. State Diagram of Vibration Data Acquisition System

A UML state diagram describes the flow of control in each element of a model as they transit from one state to another. It also describes the dynamic behaviour of an element as it responds to stimuli that cause it to change state, after satisfying some specified conditions that cause it to carry out some actions, or delays action for some events to occur. A transition occurs between a source and a target state, when an event in the source state transits to the target state and performs some definite actions when some specified conditions are satisfied [21,22,23].

4.2 Signal Processing and Fault Diagnosis Modules

Figure 5 is the Unified Modelling Language (UML) representation of the various states of the Signature Detection and Identification module of the system of Figure 1.

It portrays the states of the system from the point of retrieving data from the data acquisition system (described in section 4.1), through the state of processing the signal data to extract useful information from it, to the diagnosis state, and finally to the identification of the signature of the vibration caused by an act of vandalism on the pipeline. The final state is that of action taken by the operators of the system.

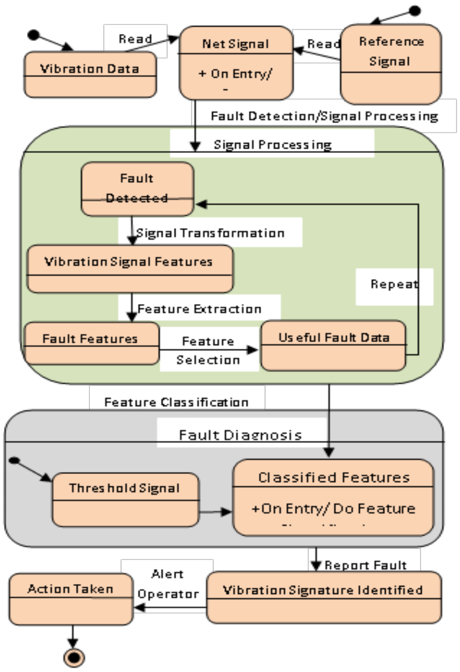

Fig. 5. State Diagram of Signal Processing and Fault Diagnosis Modules

4.2.1 Signal Processing Module

Signal processing is a technique by which computer algorithms are used to analyze and transform acquired signal data with the intention to create meaningful and better representations of the signal, in order to extract detailed and useful information embedded in the signal while blocking out the effects of noise.

For this study, a sample oil/gas pipe was excited with a drilling machine in an attempt to drill a hole into the pipe. Using the Signal Processing Toolbox of version 7.10.0.499 of the MATLAB software, the vibration signal data acquired from this process was transformed into its time domain form as shown in Figure 6.

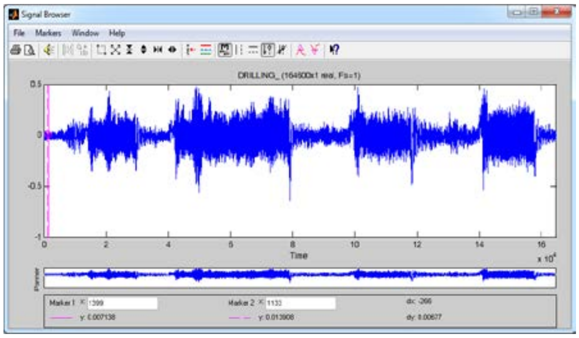

Fig.6. Time Domain Representation of Vibration Signal due to Act of Drilling.

The high frequency and amplitude of the signal reveal the features of the signals in response to the high excitation on the pipe while the drilling process was on. These (frequency and amplitude) vary in proportion to the working force of the drilling machine.

For purposes of diagnosis, the signal was classified into different component patterns.

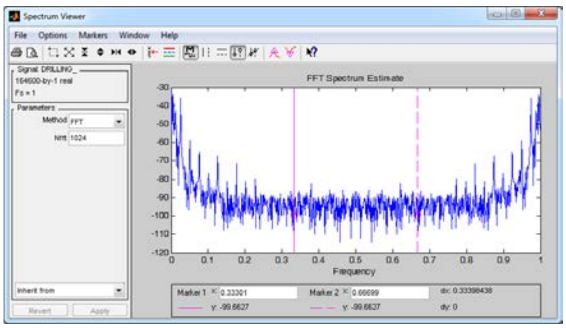

Fig. 7. Vibration Signal Power Spectrum for 1,024 Sample Data Pairs

Towards this, the signal in its time domain data was transformed into its frequency domain or the power spectrum by use of the Fast Fourier Transform (FFT) from the MATLAB Signal Processing Toolbox (as Figure 7 shows).

The objective of feature classification is to distinguish between different classes of patterns representing different conditions of an analysed data on the basis of apriori knowledge or on some statistical information that are extracted from the patterns. Feature classification is performed in the Fault Diagnosis module of Figure 5.

4.2.2 Fault Diagnosis Module

Features of the vibration signal are determined in this module. Since the input values needed by the system under study to solve the problems of classifying, identifying and reporting acts of vandalism must be expressed in qualitative terms, an Adaptive Neuro-Fuzzy Inference System (ANFIS) is embedded into the system as a classifier to achieve the classification of the vibration signal data. ANFIS mimics human intellect to reason and learn in uncertain, ambiguous and vague environments. ANFIS is a synergy of two computing methodologies; namely, artificial neural networks and fuzzy logic. Artificial Neural networks recognise patterns and adapt themselves to cope with changing environmental conditions [24]; while fuzzy logic is a domain of rigorous mathematics [25] which provides an effective formulation for modelling the imprecise and qualitative knowledge of experts, as well as the transmission of the uncertainty in human reasoning. Neuro-fuzzy models represent systems by means of fuzzy *if-then* rules that are embodied in a structure to which learning algorithms from artificial neural networks discipline can be utilised. To classify the acquired experimental vibration signal data into related patterns, an ANFIS model (shown in Figure 8), was designed and used by employing the fuzzy logic toolbox in version 7.10.0.499 of the MATLAB software,. The structure consists of a five-layer neural network that simulates the working principles of the model.

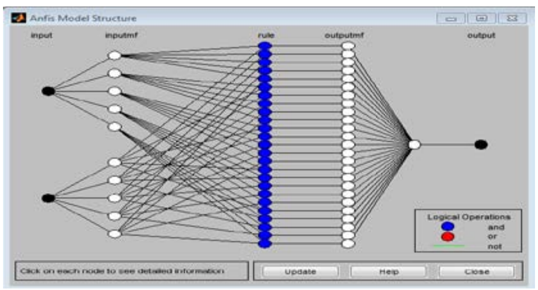

Fig. 8. Fuzzy Inference System Structure

The nodes in the first layer (input) represent the input linguistic variables (Power and Frequency of the vibration signal). The second layer (input mf), called the condition elements layer consists of term-nodes that represent the membership functions for the input variables. The input variables of Figure 8 each consists of five term-nodes namely: VERY LOW, LOW, MEDIUM, HIGH and VERY HIGH. The concept of membership function was introduced into fuzzy set theory as a means to measure the degree to which an element belongs to a particular fuzzy set. The third (rule) layer is a layer of neurons where each neuron represents a fuzzy rule (i.e., *if-then rules)*.

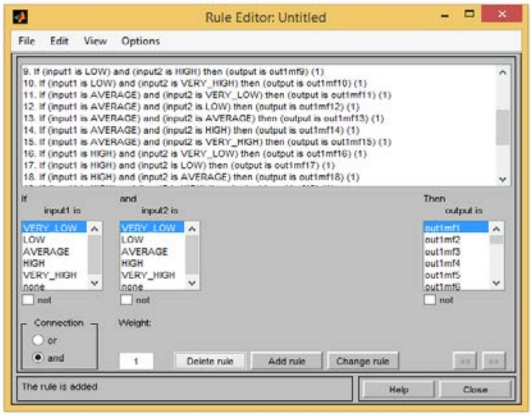

Fig. 9 Fuzzy Rules

Figure 9 is a representation of the Rule Editor showing part of the 25 fuzzy rules generated by the fuzzy system in the fuzzification of input data by the ANFIS.

The fourth layer is the output membership function (i.e., output mf) layer, which consists of the action elements. The output layer is the fifth layer, which aggregates the outputs from the fourth layer to give one single output; in this case, pipe vibration at every instant of time.

From the power spectrum generated, power/frequency data pairs are selected for use as input to the intelligent vibration feature classifier (Figure 8) to determine a match of the input signals with that of the pipe vibration signature due to acts of vandalism. This process represents feature classification or pattern recognition to identify the vibration signature of the pipeline.

Features of the classified signals from the signal processing module are compared with the threshold (T(t)) signal for possible match. The threshold values of the vibration signal from the threshold signal state of Figure 5 are values of the signals due to excitation of the pipe using a drilling machine which is assumed to be the tool used by vandals to bore hole on oil/gas pipes to be able to have access to their contents. When there is a match between the signal pattern and the threshold signal features the system recognises this as an anomaly or a fault, which is transmitted to the fault notification module.

4.3 Fault Notification Module

An alarm system is incorporated into the system to alert operators of the system, by means of appropriate Information and Communication Technology (ICT) tools, to take appropriate action when a fault is identified. Any feature that deviates from T(t) is an indication that the signal is not caused by an act of vandalism, and consequently the alarm system would not be triggered to fire.

5. Conclusion

Models for understanding the complexity of a system for the detection of acts of vandalism on oil/gas pipelines are presented in this paper. Functioning of the system is modelled with UML activity and state diagrams to reveal the relationships that exist between the various modules of the system and their dynamic behaviour. Determination of the vibration signature of pipes excited by external forces such as drilling devices is a real world problem characterized by vagueness in the information acquired in respect of the vibration signals. To achieve the goal of the design of the system under study, the UML model of the system is integrated with Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS is introduced into the system as an intelligent module that mimics the intelligent reasoning capabilities of humans. It is able to cope with the imprecise characteristics of the pipe vibration data, and intelligently classifies the data into patterns that determine the signature of the vibration that is a result of acts of vandalism on the pipe. Once the occurrence of this signature is determined, an alarm module is triggered to prompt relevant authorities to take appropriate actions.

The system presented here is suitable for development of software aimed at detection of intrusion or acts of vandalism on any type/size of oil/gas steel pipes since the models describe the fundamental procedures required for this purpose. The system is therefore not application-specific; because UML activity diagrams and state charts used to model the proposed system are tools generally employed by software developers for specifying, constructing, visualising and documenting objects of a software-based system. They are used also for communication of ideas among developers to achieve the design, development, testing and implementation goals of software.

Implementation of the system modelled in this study would significantly reduce oil theft/spillage, destruction of properties, and loss of lives hitherto experienced in the oil-producing regions of Nigeria.

REFERENCES

[1] E. A. Onibere, F. A. U. Imouokhome, "Framework for Oil/Gas Pipeline Vandalism Detection System," *The Journal of the Nigerian Institution of Production Engineers*, vol. 16, pp. 190 – 200, 2014.

[2] Iordache, O. *Modeling Multi-level Systems, Berlin Heidelberg*: Springer-Verilog, 2011.

[3] C.Nejneru, A. Nicuus, B. Constantin, L. R. Manea, M. Teodorescu, and M. Agop, "Dynamics Control of the Complex Systems via Nondifferentiability", *Journal of Applied Mathematics*, vol. 2013, pp. 1–12, 2013.http://dx.doi.org/10.1155/2013/137056 , Hindawi Publishing Corporation.

[4] R. M. Stuckey, S. Sarkani, and T. A. Mazzuchi, "Complex Acquisition Requirements Analysis Using a Systems Engineering Approach", *Defense ARJ*, vol. 24 no. 2 : 266–301, 2017.

[5] A. Bagdasaryan, "Systems Theoretic Techniques for Modeling, Control and Decision Support in Complex Dynamic Systems," *Artificial Intelligence Resources in Control and Automation Engineering*, pp. 15-72, 2012.

[6] M. Beldjehem, andA. Granular, "Hierarchical Multiview Metrics Suite for Statecharts Quality", *Advances in Software Engineering*, vol. 2013, pp. 1-13, 2013 Available at http://dx.doi.org/10.1155/2013/952178. Hindawi Publishing Corporation.

[7] M. H. Aabidi, A. Jakimi, R. Alaoui,and E.H. El Kinani, "An Object-Oriented Approach To Generate Java Code From Hierarchical-Concurrent and History States", *International Journal of Information & Network Security (IJINS)*, vol.2, no.6, pp. 429-440, 2013, ISSN: 2089-3299

[8] Object Management Group: *Unified Modeling Language* Version 2.5, 2015 http://www.omg.org/cgi-bin/doc?formal/15-03-01.pdf

[9] M. Madiajagan, and P. Garg, "Prototype of Intrusion Detection Model using UML 5.0 and Forward Engineering", *Informatica Economică*, vol. 15, no. 2, pp. 29 – 37, 2011.

[10] H. Osman, and M. R. V. Chaudron, "UML Usage in Open Source Software Development : A Field Study", *Proceedings of ACM/IEEE 16th International Conference on Model Driven Engineering Languages and Systems, Miami*, Florida, USA 29 Sept 2013 through 4 Oct 2013, Experiences and Empirical Studies in Software Modelling (EESSMod 2013), pp. 23-32.

[11] S. Anwar, J. M. Zain, M. F. Zolkipli, Z. Inayat, S. Khan, B. Anthony, and V. Chang, "From Intrusion Detection to an Intrusion Response System: Fundamentals, Requirements, and Future Directions",*Algorithms*, vol. 10, no. 39, pp.1 – 24, 2017; doi:10.3390/a10020039 www.mdpi.com/journal/algorithms

[12] E. Kerkouche, K. Khalfaoui, A. Chaoui and A. Aldahoud, " UML Activity Diagrams and Maude Integrated Modeling and Analysis Approach Using Graph Transformation" *Proceedings of the 7th International Conference on Information Technology (ICIT 2015)*doi:10.15849/icit.2015.0093 © ICIT 2015. Available from http://icit.zuj.edu.jo/ICIT15), last accessed: April 23, 2017.

[13] S. Obaid, S. Asghar, and M. Naeem, "Data Flow Analysis of UML Models by ALF" *International Journal of Trade, Economics and Finance, Vol. 5, No. 1,* pp. 12–18, *2014*.

[14] V. Spinke, "An object-oriented implementation of concurrent and hierarchical state machines",*Information and Software Technology*,vol. 55, pp. 1726–1740,2013. Elsevier B.V.

[15] C. L. Phillips, H. T. Nagle, and A. Chakrabortty, *Digital Control System Analysis and Design Fourth Edition, Global Edition*, Pearson Education Limited, 2015.

[16] A. M. Mughal, *Real Time Modeling, Simulation and Control of Dynamical Systems*, Springer International Publishing, Switzerland, 2016, DOI 10.1007/978-3-319-33906-1.

[17] T. Witelski, and M. Bowen, *Methods of Mathematical Modelling, Springer Undergraduate Mathematics Series*, Springer International Publishing, Switzerland, 2015, DOI 10.1007/978-3-319-23042-9.

[18] F. E. Cellier, and E. Kofman, *Continuous System Simulation*: Springer and Business Media Inc., NY, 2006.

[19] Y. Zeng, C. Rose, W. Taha, A. Duracz, K. Atkinson, R. Philippsen, R. Cartwright, M. O'Malley, "Modeling Electromechanical Aspects of Cyber-Physical Systems", *Journal of Software Engineering for Robotics*, vol. 7, no. 1, pp. 100-119, 2016, ISSN: 2035-3928

[20] C. G. Cassandras, and S. Lafortune, *Introduction to Discrete Event Systems*: Springer Science and Business Media, NY, 2008.

[21] E. Mirzaeian , M. Babazadeh, S. G. Mojaveri, and H. Motameni, "A New Approach to Object Oriented Software Simulation Based on UML Statechart and Colored Petri Net", *International Journal of Modeling and Optimization,* vol. 2, no. 3, pp. 299-303, 2012.

[22] S. Lee, "Unified Modelling Language (UML) for Database Systems and Computer Applications," *International Journal of Database Theory and Application*, vol. 5, no. 1, pp. 157-164, 2012.

[23] S. Dhir, "Impact of UML Techniques In Test Case Generation", *International Journal of Engineering Science and Advanced Technology [IJESAT]*, vol. 2, no.2, pp. 214 – 217,2012

[24] R. Joshi, D. Kumar, "A Neural Network Approach for Recognize Punjabi Font and Character", *International Journal of Computer Science and Technology* (IJCST), vol. 5, no. 3, pp. 102 – 105, 2014.

[25] A. Sharma, and J. H. Kaur, "QOS Comparison of BNP Scheduling Algorithms with Expanded Fuzzy System",*International Journal of Advanced Computer Science and Applications*, vol.4, no. 5, pp. 107–112, 2013.