International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 9 (2016) pp 6637-6642
© Research India Publications. http://www.ripublication.com

Deployment of a fuzzy XNOR based evaluative system for the maintainability features of aspect oriented systems

Mrs. Ananthi Sheshasaayee
Research Supervisor, PG& Research Department of Computer Science,
Quaid-E-Millath (G) College for Women, Chennai, Tamil Nadu, India.

Ms. Roby Jose
Research Scholar, PG& Research Department of Computer Science,
Quaid-E-Millath (G) College for Women, Chennai, Tamil Nadu, India.

Abstract

Software maintenance is considered as one software development activity that makes the most of the cost during software development phase. Maintainability model is a system that can be developed for assessing the maintainability of software. A maintainability model ensures the standards and guidelines that are to be followed to develop a software system that requires less maintenance activities. To develop a maintainability model statistical approach or advanced machine learning approaches could be employed. Aspect Oriented approach of software development is the programming paradigm in which those modules which implement core business logic is isolated from other parts of the program and maintained separately as aspects. Empirical studies conducted previously were able to show that aspect oriented systems are better maintainable ones compared to the predecessor object oriented systems. An exploratory study is essential to find out the factors within itself that affect maintainability of aspect oriented systems. A four factor incorporated maintainability model for aspect oriented system developed using fuzzy logic is proposed in this paper. The model based on fuzzy XNOR approach is able to establish the factors that affect the maintainability of aspect oriented systems efficiently.

Keywords: Aspect Oriented Systems, Aspect oriented metrics, Fuzzy model, Maintenance metrics, Maintainability model, Modularity

INTRODUCTION

Software maintenance is those activities performed with the intention to make alterations to the delivered software system to meet some reasons. The reasons may be any of like resolving error, augment efficacy or acclimatize to a changed environment in terms of hardware or operating system [1]. Even though software maintenance and maintainability are used alternatively offering the implication that both are same, clear distinction exists between both terminologies.

Software maintainability may be considered as a quality attribute that software should possess. The quality maintainability is a collection of sub characteristics [2]. If software maintenance activities are high then the software is considered to be lacking the quality maintainable.

Aspect oriented software development (AOSD) [3]is a software formation method that sanction the compartmentalization of modular unit that have an effect on several classes that implement the logic of the system being developed. This modular unit is termed as concerns and the new programming paradigm implements concerns into reusable module which are termed as aspects. Aspect based programming approach flourished of the fact that object oriented programming based abstractions like objects and classes was not able to capture concern effectively. Aspect based programming is an extension to object oriented programming. Aspect based systems can be developed on the edifice of object oriented programming. Thus it could be correctly stated aspect based development of software complements and not replaces object based paradigm.

The basic schemes and notions in aspect oriented programming [4] are defined based on AspectJ [5], aspect oriented version for Java. The core philosophies in the AspectJ language are joinpoint, joinpoint model, advices and intertype declaration.

Studies in the domain of structured programming and object oriented programming were able to prove that metrics are capable of predicting maintenance efforts of software. Li and Henry [6] with the help of industrial software's were able to institute metrics that are capable of predicting maintainability. So metrics based maintainability model can be used for aspect oriented systems as well.

A range of approaches from statistical to soft computing are available in the domain of knowledge for assessing maintainability. This paper proposes a fuzzy XNOR based system for the metrics based maintainability assessment of aspect oriented systems.

The arrangement of the research paper is as follows. Section II highlights the works related to the research work proposed in the body of knowledge. Section III gives an overview of metrics based model of maintainability. Section IV proposes the methodology with the implementation details. Section V discusses the result and validity of the model. Finally the paper is concluded in Section VI with suggestions for research directions in the future.

PROGRESSION IN THE LITERATURE FOR ASPECT ORIENTED SOFTWARE MAINTENANCE

The preliminary studies in the aspect based systems was focused on the empirical studies to gauge aspect oriented(AO) paradigm with that of object oriented systems (OOS) and to prove AO systems are better maintainable ones compared to OOS. Whilst the AO paradigm established its preeminence over OOS, many researches where focused on the identification of the metrics that influence the maintainability of the AO systems. The subsequent section goes over those studies pertaining to aspect based system maintenance.

A fuzzy based prediction model of maintenance for AO systems was suggested by the researchers [7]. Based on various metrics already established as influencing maintainability of AO systems, the study proposed a prediction framework and the framework was validated using fuzzy logic. The Fuzzy Inference System (FIS) was deficient in providing a benchmark that established the approach as a better one.

Adaptive Neuro Fuzzy Inference System (ANFIS) was used to appraise maintainability of AO system [8]. The investigation carried out on AO metrics using FIS and ANFIS showed that ANFIS approach is capable of predicting maintainability of AO system accurately. The study did not include a comprehensive suite of metrics which shows that the approach cannot be considered accountable for the maintainability assessment of any real time AO systems.

A quantitative analysis was performed in to examine the influence of AO programming by analyzing maintenance acts on a Web Information system [9]. The re-factoring of the system code was done to accommodate aspect orientation. The re-factored AO system had shown better stability and reusability than its OO counterpart because of lesser non commented lines of code, reduced amount of coupling, augmented modularity and cohesion.

A correlation analysis was performed between the metric WOM (Weighted Operations in Module) and changeability of AO systems [10]. The study used changeability as the surrogate measure for maintainability. The modules that were affected by the modification indicated the maintenance efforts. The statistical analysis had shown that WOM is insignificant in predicting maintainability.

The effectiveness of coupling metrics as a criterion for fault proneness was investigated by Burrows et al [11]. The study identified two metrics viz BAC(Base Aspect Coupling) and CDA (Crosscutting Degree of an Aspect) as strong predictors of fault in AO systems.

To calculate software changes a coupling metrics suite was proposed by Zhao et al[12]. The study was able to show that the proposed coupling metrics is strongly correlated with system maintainability.

Eaddy[13] used a correlation analysis between concern metrics and defects of the system. The correlation was performed to examine the consequence crosscutting concern have on the quality non-defective. The study concluded with the need for more empirical studies to draw conclusion about crosscutting concern and defects.

The examination carried by the authors [14] resulted in framework that evaluates maintainability and reusability of AO systems. The framework was validated against two empirical setups and suggested the metrics for maintainability. Based on the metrics a quality model was also proposed by the authors. Even though study suggested that design properties of software such as Separation of Concern (SoC), Coupling and Cohesion can effectively assess software quality; it could not provide ideal values that this metrics should possess.

THE METRICS BASED MODEL OF MAINTAINABILITY

The proposed fuzzy based approach for assessing maintainability is based on maintainability factors [2] and the corresponding internal elements that can be enumerated by metrics. The goal of this work is to form a representation of system to assess the maintainability of AO systems. Since aspect orientation is an extension to object oriented programming, some elements such as coupling, cohesion and size have been identified as both applicable to aspect based systems and object oriented systems[7, 14]. In addition to above mentioned three components, separation of concern and class complexity are the other two internal elements considered for the model. The five internal attributes and the constituent metrics are discussed below.

Internal Attributes

The attributes explained in the subsequent section are linked to the internal trait of the software. These attributes are very much viable in the construction of model of maintainability.

Separation of Concern (SoC)

This is the sort of new element exclusively defining the intricacies introduced by aspect oriented programming. Ceccato first proposed this attribute. For the proposed model of maintainability, the metric associated with the element is Crosscutting Degree of an Aspect (CDA). CDA[15] gives the count of modules in aspect affected because of the presence of pointcuts and introduction

Coupling

The coupling feature represents the scale of association among the modules of the software unit. The attribute is quantifiable by means a number of cohesion metrics such as CFA(Coupling on Field Access), CAE(Coupling on Advice Execution)[15]

Class Complexity (CC)

This new attribute is proposed to represent the complexity arouse as part of certain operations being performed on the modules. The metrics that define this attribute are WOM (Weighted Operations in a Module)[15] and RFM(Response for) [15].

Cohesion

This attribute is responsible for representing the extension to which the elements of a module belong to each other. The aspect oriented metrics considered for the attribute is LCOO (Lack of Cohesion in Operations)[16].

Size

The attribute represents the acceptable amount of software’s source code excluding comments. The attribute is measured by the metrics LOCC [17].

Factors affecting maintainability of AO systems

The factors or sub characteristics of maintainability based on the metrics suite and internal characteristics is sketched in Figure I. The proposed factors that affect aspect oriented system maintainability is based on the methodology followed by [18] and validated using AHP approach[19].

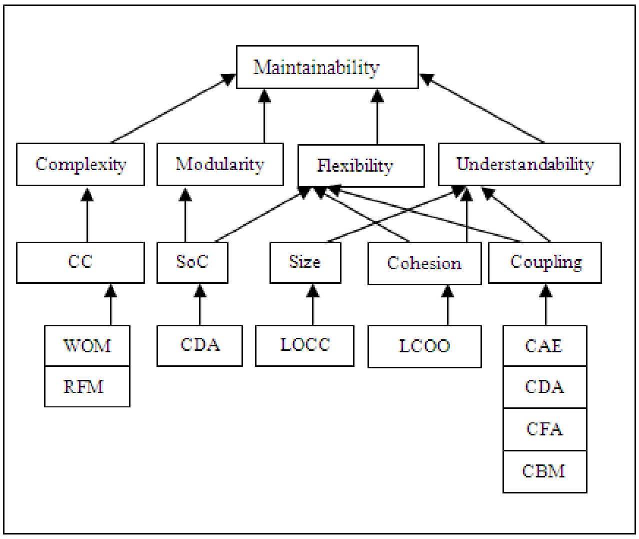

Figure 1: Mapping of metrics to maintainability

THE PROPOSED FUZZY XNOR MODEL OF MAINTAINABILITY

The proposed model

Fuzzy logic was proposed by Zadeh et al[20]. A form of mathematical reasoning where truth value of elements lie between a value 0 and 1 is termed fuzzy logic. Unlike Boolean algebra where elements take either 0 or 1 fuzzy element can have any value between 0 and 1. The capability of the approach to model imprecise data has increased its popularity of being used in multitude of domains. Modeling using the fuzzy approach does not demand unambiguous data, which makes it preferable to those domains were momentous amount of data is not available. This feature of fuzzy logic drives the motivation of using the approach to build a model of maintainability for aspect oriented systems. The four factors of maintainability that is given as input to the system are

(i) Flexibility
(ii) Understandability
(iii) Complexity
(iv) Modularity.

Based on the four input values an optimal value for maintainability is generated. The approach followed for the proposed system is sketched in Figure II.

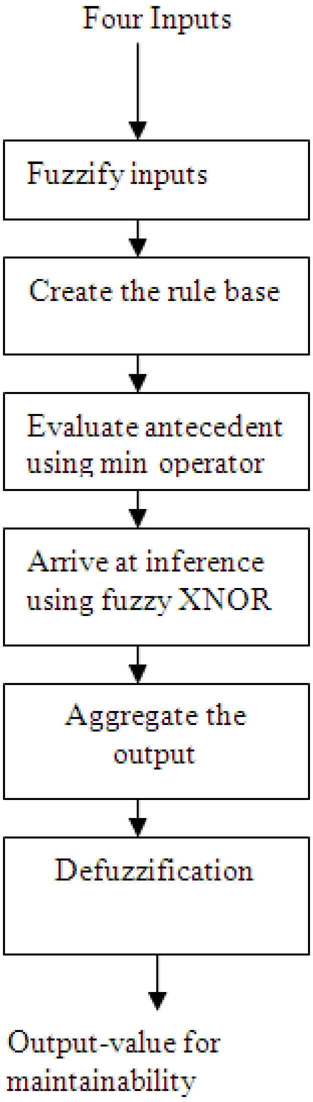
Four Inputs
Fuzzify inputs
Create the rule base
Evaluate antecedent using min operator
Arrive at inference using fuzzy XNOR
Aggregate the output
Defuzzification
Output-value for maintainability

Figure 2: The proposed fuzzy XNOR approach for assessing maintainability of Aspect Oriented systems

Experimental Setup

The proposed fuzzy approach for model of maintainability for Aspect Oriented systems is validated using Fuzzy Logic toolbox in Matlab. The execution was carried out using Mamdani type inference system. The input variables [Complexity, Modularity, Flexibility and Understandability] for the system claims three membership functions(MF). These membership functions with their ranges are low (0 0. 185 0. 37), medium (0. 31 0. 495 0. 68) and high (0. 63 0. 815 1). The output for the Fuzzy Inference System (FIS) is AO software maintenance which is categorized as low (0 0. 212 0. 422), medium (0. 382 0. 4985 0. 6151) and high (0. 5732 0. 7208 1). Both the input and the output is categorized using triangular membership functions which is scaled in the range [0 1].

The rule base was populated with 81 rules. The antecedent part of the rules are evaluated using ‘fuzzy AND’. The implication from antecedent to consequent is implemented using customized operator ‘fuzzy XNOR’ [21]. The aggregate operator is ‘fuzzy OR’. For the defuzzificaton of the output, centroid method is made use of.

RESULTS AND DISCUSSION

Table 1: Input-Output values for fuzzy XNOR-OR combination

Input MF value for				Input	Output
Under standability	Flexibility	Modularity	Complexity	Name	value for maintenance
0. 826	0. 826	0. 846	0. 208	ip1	0. 219
0. 503	0. 503	0. 503	0. 254	ip2	0. 589
0. 128	0. 3864	0. 159	0. 8538	ip3	0. 731

Table 2: Input-Output values for fuzzy AND-OR combination

Input MF value for				Input	Output
Under standability	Flexibility	Modularity	Complexity	Name	value for maintenance
0. 811	0. 869	0. 886	0. 069	ip1	0. 285
0. 451	0. 251	0. 825	0. 355	ip2	0. 491
0. 253	0. 323	0. 253	0. 901	ip3	0. 721

Table 3: Input-Output values for prod-sum combination

Input MF value for				Input	Output
Under standability	Flexibility	Modularity	Complexity	Name	value for maintenance
0. 880	0. 790	0. 881	0. 500	ip1	0. 311
0. 488	0. 505	0. 223	0. 289	ip2	0. 597
0. 462	0. 315	0. 341	0. 838	ip3	0. 810

The paper discusses the maintenance features of software in aspect oriented paradigm by applying fuzzy logic. The result of the proposed work is appraised based on the various combination of T-norm, T-conorm operators. For all the cases the defuzzification employed is centroid approach.

CASE 1: FIS evaluation result of the output value obtained for T norm-T conorm combination fuzzy XNOR-OR

The combination of the fuzzy input values and the interpretation and the corresponding output value for maintenance efforts for the fuzzy XNOR-OR are shown in Table I. In this case the implication operator is the fuzzy XNOR and the aggregation operator is max function which is as well entitled as fuzzy OR.

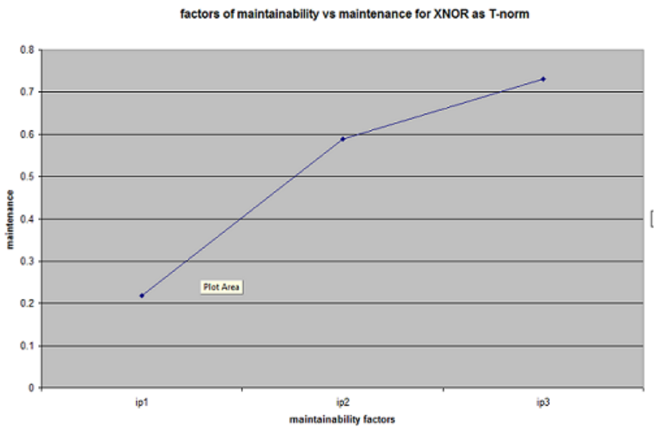

Figure 3: Maintainability factors vs. maintenance (fuzzy XNOR as T-norm)

From Table 1, when understandability, flexibility and modularity are high and complexity is low, least value for maintenance is observed. A least value for maintenance indicates highly maintainable software which is the desirable quality. The plot of the input set of data(fuzzy value for maintainability factors) against the obtained output value for maintenance is shown in figure III.

CASE 2: FIS evaluation result of the output value obtained for T norm-T conorm combination fuzzy AND-OR

On contrary to the previous case, the implication operator employed is fuzzy AND whereas aggregation operator is same as the previous case. The input and output values for the state is shown in Table II. The least value for output obtained is matching with the maintainability factors value of the previous case. In figure IV a plot of the maintainability factors against the value of maintenance generated by the FIS is shown.

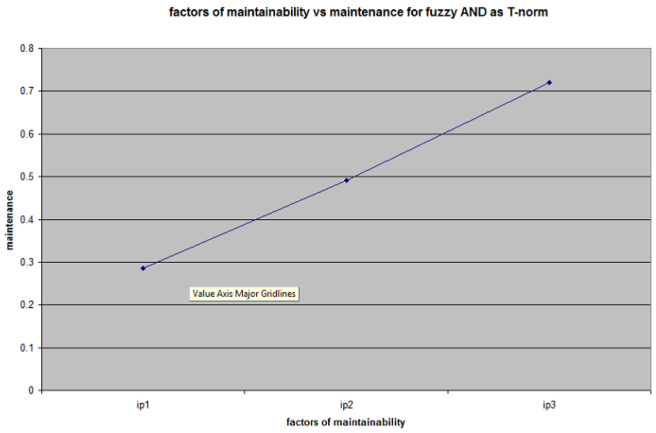

Figure 4: **maintainability factors vs. maintenance (fuzzy AND as T-norm)**

CASE 3: FIS evaluation result of the output value obtained for T norm-T conorm combination prod-sum

The FIS uses prod function as the implication operator and sum as the aggregation operator. Table III shows the input and output values for the prod-sum combination of T norm-T conorm. When maintainability factors are plotted against maintenance for the prod implication operator a plot as shown in Figure V is obtained. The least value for maintenance is obtained for high understandability, flexibility, modularity and medium complexity.

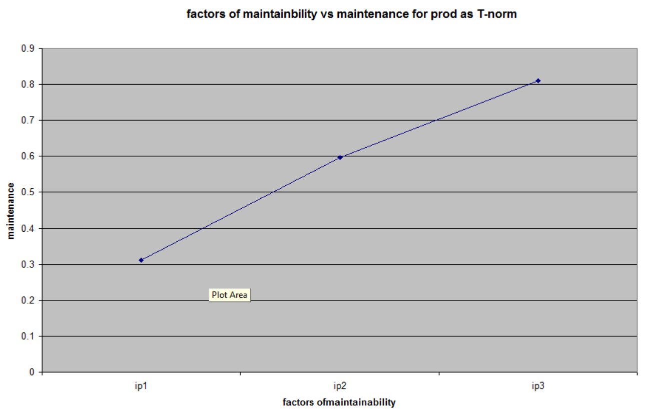

Figure 5: maintainability factors vs. maintenance (prod function as T-norm)

As a comparison of proposed fuzzy XNOR implication function against fuzzy AND or prod implication operator is made it is evident that proposed approach evaluates maintenance of an Aspect based systems with respect to factors of maintainability with the ideal values of factors ie high understandability, modularity, flexibility and low complexity. Even though fuzzy AND evaluates the maintenance with same ideal values, the output value is least for proposed approach.

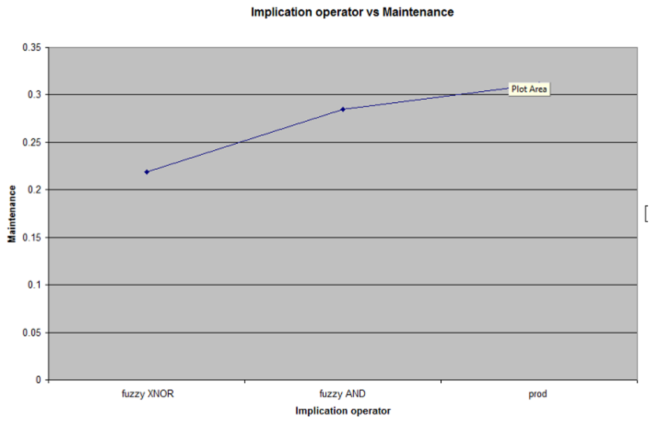

Figure 6: Comparison of T-norm operators for output values

In view of the fact that any system capable of yielding less value of maintenance is the one accepted to be efficient[7], a comparative plot of implication operators against the output yielded(maintenance) by FIS using the implication is depicted in Fig VI. The proposed approach is capable of calculating least value for maintenance for the ideal values for set of maintainability factors.

CONCLUSION

The proposed work is an attempt to build a fuzzy system based model of maintainability for aspect oriented system. For the reason that aspect oriented paradigm is at the burgeoning juncture, there exist not much data about whole suite of metrics capable of evaluating maintainability. This work is an attempt to find such a metrics suite and based on the identified metrics suite build a fuzzy model of maintainability that produces the lowest value for maintenance efforts. The fuzzy XNOR approach is successful in attaining this target as shown in previous section.

In future, the work can be extended to find the real case values of the metrics and identify the ideal values for the aspect oriented metrics suite for least maintenance efforts, thereby matching the object oriented paradigm in terms of measurement and prediction features of maintenance efforts. The proposed work can be enhanced to propose an automated system to be used by the software testers in identifying the source code violations that lead to higher maintenance efforts.

REFERENCES

[1] Edelstein, D. Vera. "Report on the IEEE STD 1219–1993—standard for software maintenance. " *ACM SIGSOFT Software Engineering Notes* 18. 4 (1993): 94-95

[2] ISO, "IEC25010: 2011 Systems and software engineering–Systems and software Quality Requirements and Evaluation (SQuaRE)–System and software quality models. " *International Organization for Standardization* (2011).

[3] Filman, Robert, Tzilla Elrad, and Siobhán Clarke. *Aspect-oriented software development*. Addison-Wesley Professional, 2004.

[4] Kiczales, Gregor, et al. Aspect-oriented programming. *11th European Conference on Object Oriented Programming* in:LNCS, vol. 1241, Springer Verlag, 1997, pp 220-242 1997.

[5] Kiczales, Gregor, et al. "An overview of AspectJ. " *ECOOP 2001—Object-Oriented Programming*. Springer Berlin Heidelberg, 2001. 327-354.

[6] Li, Wei, and Sallie Henry. "Object-oriented metrics that predict maintainability. "*Journal of systems and software* 23. 2 (1993): 111-122.

[7] P K Singh, O Sangwan, ASrivastava " An essence of software maintenance prediction using the fuzzy model for Aspect Oriented Software. " *ARPN Journal of Engineering and Applied Sciences* 9. 9 (2014): 1598-1605

[8] H Momeni, S Zahedian "Aspect Oriented software maintainability assessment using adaptive neuro fuzzy inference system*", Journal of mathematics and Computer Science* 12(2014):243-252

[9] Kulesza U., Sant'Anna C., Garcia A., Coelho R. VonStaa A. and Lucena C. 2006. Quantifying the effects of aspect-oriented programming: A maintenance study. Proc*. 22nd IEEE International Conference on Software Maintenance*. pp. 223-232.

[10] Kumar A., Kumar R. and Grover P. S. 2007. An evaluation of maintainability of aspect-oriented systems: a practical approach. *International Journal of Computer Science and Security.* 1(2): 1-9.

[11] Burrows R., Ferrari F. C., Garcia A. and Taiani F. 2010. An empirical evaluation of coupling metrics on aspect-oriented programs. Proc. *ICSE Workshop on Emerging Trends in Software Metrics,* ACM, Cape Town, South Africa. pp. 53-58.

[12] Shen H., Zhang S. and Zhao J. 2008. An empirical study of maintainability in aspect-oriented system evolution using coupling metrics. Proc. *2nd IFIP/IEEE International Symposium on Theoretical Aspects of Software Engineering*. pp. 233-236.

[13] Eaddy M., Zimmermann T., Sherwood K. D., Garg, V., Murphy G. C., Nagappan N. and Aho A. V. 2008. Do crosscutting concerns cause defects? *IEEE Transactions on Software Engineering.* 34(4): 497-515.

[14] Sant'Anna C., Garcia A., Chavez C., Lucena C. and VonStaa A. 2003. On the reuse and maintenance of Aspect Oriented software: an assessment framework. Proc. Brazilian Symposium on Software Engineering. pp. 19-34.

[15] Ceccato, Mariano, and Paolo Tonella. "Measuring the effects of software aspectization. " *1st Workshop on Aspect Reverse Engineering*. Vol. 12. 2004.

[16] Chidamber, Shyam R., and Chris F. Kemerer. "A metrics suite for object oriented design. " *Software Engineering, IEEE Transactions on* 20. 6 (1994): 476-493.

[17] http://aopmetrics. stage. tigris. org/metrics. html

[18] Fenton, N., Pfleeger, S. "Software Metrics: A Rigorous and Practical Approach". 2. ed. London: PWS, 1997.

[19] T. L. Saaty, *The Analytic Hierarchy Process*, New York: McGraw Hill. International, 1980

[20] Zadeh, L. A., 1965, "Fuzzy Sets, Information and Control", vol. 8 (338-353).

[21] Li, Yingfang, Keyun Qin, and Xingxing He. "Fuzzy XNOR connectives in fuzzy logic. " *Soft Computing* 15. 12 (2011): 2457-2465.