Evolutionary Optimization of Fuzzy Decision Systems for Automated Insurance Underwriting

Piero P. Bonissone, Raj Subbu, Kareem S. Aggour

Abstract—**A robust method for automating the tuning and maintenance of fuzzy decision-making systems is presented. A configurable multi-stage mutation-based evolutionary algorithm optimally tunes the decision thresholds and internal parameters of fuzzy rule-based and case-based systems that decide the risk categories of insurance applications. The tunable parameters have a critical impact on the coverage and accuracy of decision-making, and a reliable method to optimally tune these parameters is critical to the quality of decision-making and maintainability of these systems.**

I. Introduction

Advances in information technologies are allowing organizations to automate key business processes in an effort to increase throughput and reliability while reducing risk. The success of these ventures is highly dependent on the availability of generalized decision-making systems that are not just able to reliably replicate the human decision-making process, but do so in an explainable, transparent fashion. Insurance underwriting is one such high-volume application domain where intelligent automation can be highly beneficial, and reliability and transparency of decision-making are critical.

A highly trained individual traditionally performs insurance underwriting. A given insurance application is compared against several standards put forward by the insurance company and classified into one of the risk categories (*rate classes*) available for the type of insurance requested. The risk categories then affect the premium paid by the applicant--the higher the risk category, the higher the premium. In our approach, decision-making systems based on fuzzy rule-based reasoning and case-based reasoning are utilized to automate the risk classification of insurance applications of increasing degrees of complexity.

The fuzzy rule-based and case-based decision systems have several decision thresholds and internal parameters that have a critical impact on the coverage and accuracy of decision-making, and a reliable method to optimally tune these parameters is critical to the quality of decision-making and maintainability of these systems.

This work was supported in part by the Digital Underwriting project sponsored by General Electric Corporation.

The authors are with General Electric Global Research Center, One Research Circle, Niskayuna, NY 12309 USA. Email: bonissone@crd.ge.com, subbu@crd.ge.com, aggour@crd.ge.com.

In this paper, we present a robust method for automating the tuning and maintenance of the fuzzy decision-making systems. A configurable multi-stage mutation-based evolutionary algorithm optimally tunes the decision thresholds and internal parameters of fuzzy rule-based and case-based systems that decide the risk categories of insurance applications. Evolutionary algorithms execute utilizing principles of natural evolution and are robust adaptive search schemes suitable for searching non-linear, discontinuous, and high-dimensional spaces. Moreover, this tuning approach does not require an explicit mathematical description of the multi-dimensional search space, and instead relies solely on an objective function that is capable of producing a relative measure of alternative solutions. Such tuning improves the reliability of the automation by minimizing the degree of rate-class assignment mismatch between that of an expert human underwriter and the machine, and supports the maintenance of the accuracy of the decision-making as decision guidelines evolve over time.

In Section II, we present background information on automated insurance underwriting, evolutionary algorithms, and evolutionary optimization of fuzzy decision systems. In Section III, we describe the fuzzy rule-based and case-based decision-making systems. Section IV presents the architecture for evolutionary tuning of the decision systems, and Section V presents simulation results. Section VI concludes this paper.

II. Background

A. Automated Insurance Underwriting

Reported research in the area of automated insurance underwriting is quite sparse. However, there are a few documented approaches. Collins et al. (Collins et al. 1988) describe the application of a neural network to replicate the decision-making of mortgage insurance underwriters by training the system on a database of certified cases. Insurance underwriting based on neural-networks or similar modeling approaches leads to *opaque* decision-making, wherein the learned and encoded interrelationships between decision variables that are used to arrive at decisions is not well defined and explainable. Nikolopoulos et al. (Nikolopoulos et al. 1994) describe the application of evolutionary learning and classification tree techniques to build a knowledge base that determines the termination criteria for an insurance policy.

0-7803-7280-8/02/$10.00 ©2002 IEEE

B. Evolutionary Algorithms

Evolutionary Algorithms (EAs) include genetic algorithms (Goldberg, 1989, Holland, 1994), evolutionary programming (Fogel et al., 1966), evolution strategies (Bäck, 1996), and genetic programming (Koza, 1992). The principles of these related techniques define a general paradigm that is based on a simulation of natural evolution. EAs perform their search by maintaining at any time t a population $P(t) = \{P_1(t), P_2(t), ..., P_P(t)\}$ of individuals. "Genetic" operators that model simplified rules of biological evolution are applied to create the new and more superior population $P(t+1)$. This process continues until a sufficiently good population is achieved, or some other termination condition is satisfied. Each $P_i(t) \in P(t)$, represents via an internal data structure, a potential solution to the original problem. Closely linked to the representation of solutions, is the fitness function $\psi : P(t) \rightarrow R$, that assigns credit to candidate solutions. Individuals in a population are assigned fitness values according to some evaluation criterion. Highly fit individuals are more likely to create offspring by *recombination* or *mutation* operations, whereas weak individuals are less likely to be picked for reproduction and eventually die out. A mutation operator introduces genetic variations in the population by randomly modifying some of the building blocks of individuals. Evolutionary algorithms are essentially parallel by design, and at each evolutionary step a breadth search of increasingly optimal sub-regions of the search space is performed. Evolutionary search is a powerful technique of solving problems, and is applicable to a wide variety of practical problems that are nearly intractable with other conventional optimization techniques. Though practical evolutionary search schemes do not guarantee convergence to the global optimum in a predetermined finite time, they are often capable of finding very good and consistent approximate solutions. Moreover, they are shown to asymptotically converge under mild conditions (Subbu and Sanderson, 2000).

C. Evolutionary Optimization of Fuzzy Decision Systems

Many researchers have explored the integration of fuzzy reasoning and evolutionary algorithms. Cordon et al. (Cordon et al. 1996) present a bibliography of nearly 300 papers that discusses the combination of fuzzy reasoning and evolutionary algorithms for a variety of applications. We limit this discussion to a few key contributions.

Herrera et al. (Herrera et al. 1995) use a genetic algorithm to tune each rule used by a fuzzy logic controller. They utilize a real encoding for a four-parameter representation of a trapezoidal membership function in each term-set. A rule is achieved by the concatenation of membership functions. A member in the genetic search is such a concatenation of the encoding of membership functions. Lee and Takagi (Lee and Takagi 1993) use a genetic algorithm to identify and tune the rules of a fuzzy logic controller wherein they use a binary encoding for each three-parameter representation of a triangular membership function. Bonissone et al. (Bonissone et al. 1996) use a binary encoding in a genetic search that tunes the parameters (membership functions and scale factors) of a fuzzy logic controller that is used to achieve superior control of a freight train. Tsang and Yeung (Tsang and Yeung 1999) describe a method that combines genetic algorithms and neural networks for automated tuning of the parameters of a fuzzy expert system used as an advisor for job placement. Grauel et al. (Grauel et al. 2000) present a methodology for optimizing fuzzy classifiers based on B-splines using an evolutionary algorithm. The tuning algorithm maximizes the performance of breast cancer detection and at the same time minimizes the size of the classifier. Other reports that explore the evolutionary tuning of fuzzy classifiers appear in the references (Ho et al. 2000, Ishibuchi et al. 1996, Wong et al. 2000).

III. Fuzzy Decision Systems

A. Rule-based Decision System

The fuzzy rule-based decision system is based on rule-sets that encode underwriting standards. Each rule-set represents a set of constraints with fuzzy regions at the boundaries of different rate classes. The decision system is derived from knowledge engineering sessions with expert underwriters that helped identify factors such as blood pressure levels and cholesterol levels that are critical in defining the applicant's risk and corresponding premium. The goal is to assign an applicant to the most competitive rate class, providing that the applicant's vital data meet all of the constraints of that particular rate class to a minimum degree of satisfaction. The definition of each constraint and the determination of the minimal degree of satisfaction are design parameters that are initialized via these knowledge engineering sessions. However, there is no guarantee that these parameters are optimal, and there is a need to provide a process that can optimize them according to some predefined metric. Figure 1 illustrates the evaluation of the data from a hypothetical applicant using the rule-based constraints of a given rate class.

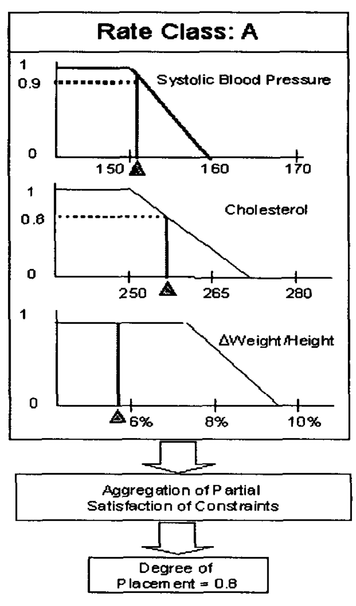

Figure 1: Evaluation of hypothetical applicant data (triangular markers) using fuzzy rule-based constraints.

0-7803-7280-8/02/$10.00 ©2002 IEEE

B. Case-based Decision System

Unlike rule-based decision-making, which provides a generative approach to the representation of knowledge, case-based decision-making represents an analogical approach to the solution of the same problem. For instance, in the judicial system, statutory law is interpreted by a judge and applied to specific instances, which then becomes part of jurisprudence. When the law is not clear or unambiguous, legal precedents are used to help the judge make consistent decisions on new but similar cases. In this analogy, statutory law corresponds to a set of underwriting rules, while the set of precedents corresponds to past applications for which a rate class was determined. In this endeavor, complex cases require the aid of case-based reasoning to determine the correct rate class for an applicant.

Knowledge engineering is used to determine the most efficient indices that represent cases, to define the fuzzy similarity metric that induces the best ranking upon the retrieved cases, and to adapt the solution of the closest cases to the application on hand. The parameters used by the case-based decision system to search for similar cases and to rank them are also design choices that need to be determined and maintained over time for optimal performance. Figure 2 illustrates an example where the applicant data is represented by a point in a multi-dimensional index space, and the points in the surrounding sphere represent potentially similar cases, each with a corresponding color-coded rate class. The corresponding histogram illustrates the distribution of the retrieved neighboring cases, and the information conveyed by this histogram is used by the case-based decision system to make the final decision for a given application.

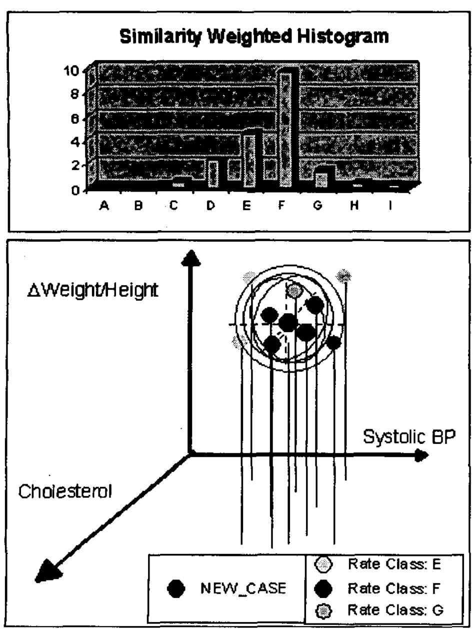

Figure 2: Neighbors closest to a new case in case-based decision-making.

IV. Architecture

In this section, we first discuss the architecture for evolutionary optimization of the decision thresholds and internal parameters of fuzzy rule-based and case-based systems that decide the risk categories of insurance applications. Next, we describe the mechanics of the evolutionary algorithm that performs the optimization.

A. Optimization of Fuzzy Decision-Making System Parameters

Each tunable parameter in either of these decision-making systems is a bounded real. The parameter optimization problem may therefore be mathematically described as a minimization problem

$$\min_{x \in \chi} \psi(x) \quad \text{where } \chi = \chi_1 \times \chi_2 \times \cdots \times \chi_n$$

$$\chi_i \subset \Re \quad \text{and} \quad \psi : \chi \rightarrow \Re_+$$

where χ is an n-dimensional bounded hyper-volume (parametric search space) in the n-dimensional space of reals, x is a parameter vector, and ψ is the objective function that maps the parametric search space to the non-negative real line.

Figure 3 shows a high-level framework of the parameter optimization problem, where χ corresponds to the space of decision engine[1] designs induced by the parameters imbedded in a decision engine, and the objective function ψ measures the corresponding degree of rate-class assignment mismatch between that of the expert human underwriter and the decision-engine for a set of certified test cases. The evolutionary algorithm iteratively generates trial solutions (trial decision engine designs in the space χ), and uses their corresponding consequent degree of rate-class assignment mismatch as the search feedback. Below, we further elucidate on this process.

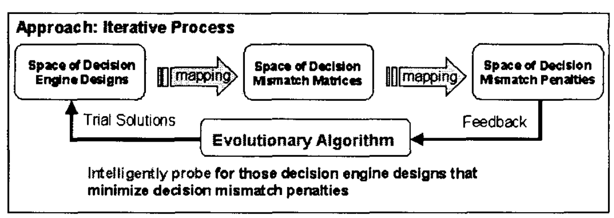

Figure 3: High-level framework for evolutionary optimization of fuzzy decision-making systems.

Figure 4 shows an example of an encoded population maintained by the evolutionary algorithm at a given generation. Each individual in the population is a trial vector of design parameters, and each percentage entry represents a normalized value of a trial parameter that falls within a corresponding bounded real line.

[1] The term decision engine is a term common to both the fuzzy rule-based decision-making system and the fuzzy case-based decision-making system.

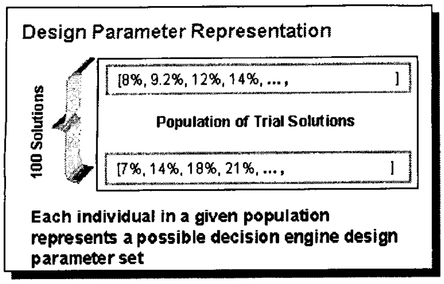

Figure 4: Representation of design parameters in the evolutionary search process.

Each trial solution vector is used to initialize an instance of the decision engine, following which each of the cases in the certified case database is evaluated as shown in Figure 5. In the case of the fuzzy rule-based system, evaluating the cases in the certified case database requires simply iterating through the set and presenting each case to the decision engine for evaluation. However, for the fuzzy case-based system, evaluating each case in the case database requires "leaving-out" that case from the case database prior to formulating the search for its most similar neighbors, and this is done in order to avoid retrieval of a neighbor with perfect similarity to the trial case.

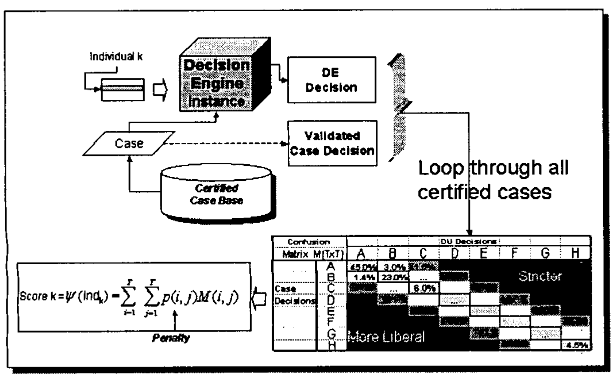

Figure 5: Process for computation of confusion in decision-making.

Once all the cases in the certified case database are evaluated, a square confusion matrix as shown above is generated. In this matrix, the ordered rate classes (left to right, and top to bottom) correspond to increasing risk categories. The rows of this matrix correspond to certified case decisions as determined by an expert human underwriter, and the columns of this matrix correspond to the machine computed decisions for the cases in the certified case database. Let us assume a case with certified case decision C (from the matrix in Figure 5) and machine decision A. This case would count towards an entry in the cell at row 3 and column 1. It is clear that for this case the certified decision is a higher risk category while the machine places this case in a lower risk category. Therefore, for this case the machine has been more liberal in decision-making. If on the other hand both the certified decision and machine decision agree at class C, then this case would have counted towards an entry in the cell at row 3 and column 3. If however the certified case decision is A, but the machine decision is C, then clearly the machine decision is more strict. From this example, we gather that a decision engine that is able to place the maximum number of certified cases along the main diagonal is desirable, and we seek to determine those decision engine parameters that produce such a favorable behavior. Stated alternatively, we desire to minimize the degree of rate-class assignment confusion or mismatch between the certified case decisions and machine assigned decisions. A given confusion matrix **M** is used as the foundation to compute an aggregate mismatch penalty or score. A penalty matrix **P** derived from actuarial studies selectively penalizes various degrees of misclassification and is element-by-element multiplied with the cells of the confusion matrix **M** to generate an aggregate penalty/score for each trial vector of parameters in the evolutionary search.

B. Mechanics of the Evolutionary Search

The evolutionary algorithm utilizes only the selection and stochastic variation (mutation) operations to evolve generations of trial solutions. While the selection operation seeks to exploit known search space regions, the mutation operation seeks to explore new regions of the search space. The theoretical foundation for this algorithm class appears in Subbu and Sanderson (Subbu and Sanderson 2000), and a high-level view of this evolutionary process is shown in Figure 6.

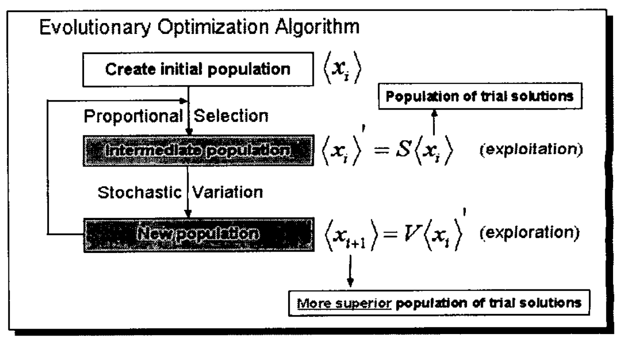

Figure 6: Evolutionary process that performs parameter optimization.

This evolutionary algorithm uses a fixed population size and operates in one or more stages, each stage of which is user configurable. A stage is specified by a tuple consisting of a fixed number of generations and normalized spread of a Gaussian distribution governing randomized sampling. A given solution (also called the parent) in generation i is improved as follows:

- The parent solution is cloned to produce two identical child solutions.
- The first child solution is mutated according to a uniform distribution within the allowable search bounds.
- The second child solution is mutated according to the Gaussian distribution for generation i. If the mutated

solution falls outside of the allowable search bounds, then the sampling is repeated a few times until an acceptable sample is found. If no acceptable sample is found within the allotted number of trials, then the second child solution is mutated according to a uniform distribution.

- The best of the parent and two child solutions is retained and is transferred to the population at generation $i+1$. In addition, it is ensured that the improvement in the best performing individual of each generation of evolution $i+n$ (where n is an increasing whole number) is a monotone function. This is achieved via elitism.

V. SIMULATION RESULTS

In this section, we present select results from the evolutionary tuning of the thresholds and internal parameters of the fuzzy rule-based and case-based decision systems. Each decision system is first evaluated based on a set of sub-optimal handcrafted parameters, and is later evaluated based on a set of evolutionary optimized parameters. While the rule-based decision system is optimized using as reference a set of 2920 certified cases, the case-based decision system is optimized using a subset of 1847 certified cases. We use a reduced set of certified cases for the case-based decision system, since its optimization requires repeated access and retrieval of cases from the case database, which is time consuming. For the purposes of this experimental evaluation, the cases input to the rule-based system and the case-based system belong to the same category of complexity.

We evaluate the performance of the decision systems based on the three metrics below:

- *Coverage*: This corresponds to the number of cases with respect to the total number of input cases whose rate class assignments are decided by a decision system. Each decision system has the option to not make a decision on a case, and to refer an undecided case to the expert human underwriter. A high degree of coverage is desirable from the perspective of automation.
- *Relative Accuracy*: This measures the ability of a decision-making system to make correct decisions on those cases that were not referred to the expert human underwriter. A high degree of relative accuracy is thus desirable.
- *Global Accuracy*: This measures the ability of a decision-making system to make correct overall decisions, which includes making correct rate class decisions, and making a correct decision to refer cases to the expert human underwriter. A high degree of global accuracy is thus desirable.

A. Optimization of the Rule-based Decision System

Table 1 below compares the performance of the un-tuned rule-based decision system to its performance after tuning. Tuning results in an improvement of all three performance metrics. Although the improvement is second-order in nature, the resulting cost savings due to tuning would be significant over a long term of automated decision-making.

Table 1: Performance of the un-tuned and tuned rule-based decision system.

METRIC	SUB-OPTIMAL PARAMETERS	OPTIMIZED PARAMETERS
Coverage	90.38%	91.71%
Relative Accuracy	92.99%	95.52%
Global Accuracy	90.07%	93.63%

We have also performed a five-fold robustness testing of the tuning process, wherein a given set of certified cases is randomly partitioned into five disjoint portions, and the set of cases corresponding to the complement (80% of the cases) of each 20% portion is used for tuning, and each 20% portion is used as a test set. Table 2 below shows the average performance of the rule-based decision system over the five tuning case sets, and the average performance of the rule-based decision system over the five disjoint test case sets. These results show that the evolutionary tuning is robust across several case data sets.

Table 2: Average performance over the five tuning case sets vs the average performance over the five disjoint test case sets.

METRIC	AVERAGE PERF. OVER TUNING SETS	AVERAGE PERF. OVER TEST SETS
Coverage	91.81%	91.80%
Relative Accuracy	94.52%	93.60%
Global Accuracy	92.74%	91.60%

B. Optimization of the Case-based Decision System

Table 3 below compares the performance of the un-tuned case-based decision system to its performance after evolutionary tuning. While tuning has a second-order effect on the key metrics of the rule-based decision system, it has a first-order effect on the coverage and global accuracy of the case-based decision system.

Table 3: Performance of the un-tuned and tuned case-based decision system.

METRIC	SUB-OPTIMAL PARAMETERS	OPTIMIZED PARAMETERS
Coverage	47.97%	98.86%
Relative Accuracy	92.10%	90.80%
Global Accuracy	44.18%	89.77%

C. Discussion

The majority of the *relative inaccuracy* in decision-making of the rule-based and case-based systems is due to misclassification within one rate class of the correct rate class.

Misclassifications beyond one rate class of the correct rate class comprise a small fraction of the error. This is consistent with the fact that misclassifications within one rate class of the correct rate class do not carry a significant penalty.

VI. Conclusions

We have developed two complementary methodologies based on fuzzy rule-based and case-based reasoning to create decision systems that are able to automatically determine risk categories for insurance applications. The decision thresholds and internal parameters of these decision-making systems are tuned using a multi-stage mutation-based evolutionary algorithm. The fitness function used selectively penalizes different degrees of misclassification, and serves as a forcing function that drives correct classifications. The tunable parameters have a critical impact on the coverage and accuracy of decision-making, and a reliable method to optimally tune these parameters is critical to the quality of decision-making and the maintainability of these systems.

Maintenance of the classification accuracy over time is an important requirement considering that decision guidelines may evolve, and so can the set of certified cases. Therefore, it is of paramount importance to maintain the quality of the set of certified test cases that will be used as a benchmark for representing any new behavior desired of the decision-making systems.

References

[1] T. Bäck. *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New York, 1996.

[2] P. P. Bonissone, P. S. Khedkar, and Y. Chen. Genetic Algorithms for Automated Tuning of Fuzzy Controllers: A Transportation Application. In *Proceedings of the IEEE Conference on Fuzzy Systems*, 1996.

[3] E. Collins, S. Ghosh, and C. Scofield. An Application of a Multiple Neural Network Learning System to Emulation of Mortgage Underwriting Judgments. In *Proceedings of the IEEE International Conference on Neural Networks*, 1988.

[4] O. Cordon, F. Herrera, and M. Lozano. A Classified Review on the Combination Fuzzy Logic-Genetic Algorithms Bibliography. *Technical Report DECSAI 95129*, Department of Computer Science and AI, Universidad de Granada, Spain, December 1996.

[5] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence Through Simulated Evolution*. John Wiley, New York, 1966.

[6] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Massachusetts, 1989.

[7] A. Grauel, I. Renners and L. A. Ludwig. Optimizing Fuzzy Classifiers by Evolutionary Algorithms. In *Proceedings of the IEEE 4th International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, 2000.

[8] F. Herrera, M. Lozano, and J. L. Verdegay. Tuning Fuzzy Logic Controllers by Genetic Algorithms. *International Journal of Approximate Reasoning*, 12(3/4), 1995.

[9] S-Y. Ho, T-K. Chen, and S-J. Ho. Designing an Efficient Fuzzy Classifier using an Intelligent Genetic Algorithm. In *Proceedings of the IEEE 24th Annual International Computer Software and Applications Conference*, 2000.

[10] J. H. Holland. *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, Cambridge, Massachusetts, 3rd edition, 1994.

[11] H. Ishibuchi, T. Nakashima and T. Murata. Genetic-algorithm-based Approaches to the Design of Fuzzy Systems for Multi-dimensional Pattern Classification Problems. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, 1996.

[12] J. Koza. *Genetic Programming: On the Programming of Computers by means of Natural Selection*. The MIT Press, Cambridge, Massachusetts, 1992.

[13] M. A. Lee and H. Takagi. Dynamic Control of Genetic Algorithms using Fuzzy Techniques. In *Proceedings of the 5th International Conference on Genetic Algorithms*, 1993.

[14] C. Nikolopoulos and S. Duvendack. A Hybrid Machine Learning System and its Application to Insurance Underwriting. In *Proceedings of the IEEE International Congress on Evolutionary Computation*, 1994.

[15] R. Subbu and A. C. Sanderson. Modeling and Convergence Analysis of Distributed Coevolutionary Algorithms. In *Proceedings of the IEEE International Congress on Evolutionary Computation*, 2000.

[16] E. C. C. Tsang, and D. S. Yeung. Optimizing Fuzzy Knowledge Base by Genetic Algorithms and Neural Networks. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 1999.

[17] C-C. Wong, C-C. Chen, and B-C. Lin. Design of Fuzzy Classification System using Genetic Algorithms. In *Proceedings of the IEEE 9th International Conference on Fuzzy Systems*, 2000.