Genetic Participatory Algorithm and System Modeling *

Yi Ling Liu
School of Electrical and Computer Engineering
University of Campinas
Campinas SP, Brazil
yilingli@dca.fee.unicamp.br

Fernando Gomide
School of Electrical and Computer Engineering
University of Campinas
Campinas SP, Brazil
gomide@dca.fee.unicamp.br

ABSTRACT

This paper suggests a genetic participatory learning algorithm and illustates its use in fuzzy systems modeling. The algorithm emerges from the concepts of participatory learning, selective transfer, and differential evolution. In genetic participatory learning the current population plays an important role in shaping evolution of the population individuals themselves. Selection uses compatibility between best and ramdonly chosen individuals. Exchange of information between individuals employes a recombination operator built from a selective transfer mechanism, whereas mutation proceeds analogously as in differential evolution. Recombination and mutation operations are affected by compatibility between individuals. An application example regarding fuzzy modeling of an electric maintenance problem using actual data serves to illustrate the effectveness of the algorithm, and to compare with alternative participatory and genetic fuzzy systems approaches. Computational results suggest that genetic participatory learning produces accurate and competitive models when compared with current state of the art approaches.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Algorithms

Keywords

Evolutionary Participatory Learning, Selective Transfer, Differential Evolution, Fuzzy Modeling

*(Produces the permission block, and copyright information). For use with SIG-ALTERNATE.CLS. Supported by ACM.

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.

1. INTRODUCTION

The development of algorithms for problem solving has been a major subject in applied mathematics, computer science and engineering. Evolutionary algorithms provide a path to solve complex problems in many areas, especially in optimization and system modeling. They are robust, do not need too much specialization to specific classes of problems, and deliver good solutions within reasonable time [1]. Indeed, gentic algorithms are amongst the most widely used approaches for automated decision making and problem solving [2]. Genetic algorithms rely on the idea that environment causes natural selection in a population of individuals, and survival of the fittest induced by natural selection improves population fitnesss.

However, in the real world survival of the fittest saga, there appears to be additional processes going on. First, besides being determined by some external requirement, fitness is always strongly affected by the population itself [3]. The population influences the reproductive suitability of individuals through interaction, compatibility, and immitation. There is also a sort of fitness function learning because the combination of external requirements with interaction, compatibility, and immitation produces the fitness function itself. One method to capture the role that the population plays in evolution is participatory learning (PL). Interaction and immitation can be grasped by recombinaton through selective transfer. Selective transfer is a one way transfer of substrings mechanism.

Genetic algorithms (GA) and differential evolution (DE) are important evolutionary approaches. Generally speaking, while DE maintains a population of individuals using mutation, recombination and selection steps working in sequence during generations, GA reverses these steps. Both keep whichever individual whose fitness is best.

Recently [4] introduced an evolutionary participatory learning algorithm (DPLAT) which uses a mutation scheme developed from the arousal mechanism of participatory learning, in addition to selective transfer and compatitility indexes between individuals. The processing steps proceed similarly as in differential evolution algorithms. This paper suggests a new class of genetic participatory learning algorithm (GPLAT) based uppon selective transfer, compatibility indexes and mutation as in DPLAT, but the processing sequence is as in a GA. The performance of the algorithm is evaluated using actual data to develop a fuzzy model for an electric maintenance instance (ELE) reported in [5]. The GPLAT is compared against a differential participatory learning algorithm (DPLAX), a genetic participatory learning algorithm (GPLAX), both with arithmetical-like crossover instead of selective transfer, the DPLAT of [6], and with a state of the art genetic fuzzy system approach [5].

This paper is organized as follows. The next section briefly overviews genetic participatory learning and its participatory operators, namely, participatory selection, participatory selective transfer, and mutation. Section 3 sumarizes the use of GPLAT to develop a fuzzy rule-based model for the electrical maintenance data ELE. The results provided by GPLAT are compared against a state of the art approach [5], the evolutionary participatory algorithms DPLAT [4], GPLAX and DPLAX [6]. Section 4 concludes the paper and summarizes issues for further investigation.

2. GENETIC PARTICIPATORY LEARNING

In this section, we briefly review the participatory learning paradigm and explain the participatory selection, selective transfer, and mutation operators.

2.1 Participatory Learning

The concept of participatory learning explored in this paper was itroduced in [7]. The basic idea of PL is that the process of learning depends on what is already known or believed. A key point of the idea of PL is that an observation has the greatest impact in causing learning or knowledge revision when it is compatible with the current knowledge. Participatory learning emphasizes that the current knowledge participates in the process of learning about itself. Thus, a fundamental part of this learning scheme is the compatibility between observation and knowledge. The basic scheme of participatory learning is depicted in Figure 1. What is essential is to note that the current knowledge, in addition to give via the lower loop, a standard against which the observations are compared, directly affects the process used for learning via the upper loop. The upper loop indicates that the current knowledge affects how the system accepts and processes input information. In PL an arousal mechanism monitors the performance of the learning process by observing the compatibility of the current knowledge with the observations. This information is then feedback via the upper arousal loop in terms of an arousal index that subsequently affects the learning process. A formal mechanism to

Figure 1: Participatory learning approach.

update knowledge is a smoothing like algorithm:

$$v(t+1) = v(t) + \alpha\rho_t(d(t) - v(t)) \tag{1}$$

where $v(t+1)$, $v(t)$ and $d(t)$ are n-dimensional vectors corresponding to the current knowledge, previous knowledge, and current observation, respectively. The parameter $\alpha \in [0,1]$ is the basic learning rate and $\rho_t \in [0,1]$ is a compatibility index which measures the compatibility degree between knowledge and observation at step t.

A device to monitor compatibility index values is an arousal index. Expression (1) can be rewritten to incorporate the arousal index as follows:

$$v(t+1) = v(t) + \alpha\rho_t^{1-a_t}(d(t) - v(t)) \tag{2}$$

where $a \in [0,1]$ is the arousal index.

One way to compute the compatibility index ρ_t is as follows:

$$\rho_t = 1 - \frac{1}{n}\sum_{k=1}^{n}|d_k(t) - v_k(t)|. \tag{3}$$

It is interesting to note that in participatory learning if $\rho_t = 0$, then $v(t+1) = v(t)$ which means that the current observation $d(t)$ is completely incompatible with the current knowledge $v(t)$. This condition implies that the system is not open to any learning from the current observation. On the other hand if $\rho_t = 1$, then $v(t+1) = d(t)$ for $\alpha = 1$. In this case the observation is in complete agreement with the current knowledge and thus the system is fully open for learning. Also, notice that the basic learning rate α is modulated by the compatibility degree. This helps to attenuate swings due to values of d which are far from v, which gives a way to smooth the effect of conflicting observations.

One procedure to obtain the arousal index is the following:

$$a_{t+1} = a_t + \beta((1 - \rho_{t+1}) - a_t) \tag{4}$$

where $\beta \in [0,1]$ controls the rate of change of arousal. The higher a_t, the less confident is the learning system in current knowledge. The arousal index can be understood as the complement of the confidence in the knowledge structure currently held.

The genetic participatory learning algorithm (GPLAT) exploits the formulation above to develop its operators and algorithm.

2.2 Selection

Assume a space $S = S^t \cup S^{t+1}$ of strings of fixed length n. Let $S^t \subset S$ denote the old population, and $S^{t+1} \subset S$ denote the new population. Let $s^* \in S$ be the best individual from the objective function point of view, and $s \in S^{t+1}$ and $s^{'} \in S^t$ be two individuals chosen randomly. In GPLAT the selection process first computes the compatibility degrees between s and $s^{'}$ with s^* and chooses the one most compatible. More precisely, the compatibility degrees $\rho^s(s, s^*)$ and $\rho^{s^{'}}(s^{'}, s^*)$ as follows:

$$\rho^s = 1 - \frac{1}{n}\sum_{k=1}^{n}|s_k - s_k^*|, \tag{5}$$

and

$$\rho^{s^{'}} = 1 - \frac{1}{n}\sum_{k=1}^{n}|s_k^{'} - s_k^*|, \tag{6}$$

and the individual whose compatibility degree is the largest is selected. In this step it is important to notice that selection depends on both, the objective function which identifies s^*, and on ρ, which measures the compatibility degree between s^* and the candidate individuals. The purpose of combining the objective function with compatibility degrees is to shape the reproductive fitness of individuals. Jointly, s^* and ρ, decide which individual is selected. Thus the individuals themselves are part of the selection procedure. This a manifestation of the participatory nature of the GPLAT.

2.3 Selective Transfer

In [8] Birchenhall et al. suggested to replace selection and crossover operators by an operator involving selective transfer. Essentially, selective transfer is a filtered replacement of substrings from one string to another, without excluding the possibility that the entire sequence is copied. Noticeably, selective transfer is derived from Holland crossover, but it is based on one-way transfer of strings, not on exchange of strings, and its behavior is likely to be very different from the combination of selection and crossover.

GPLAT employes the selective transfer concept as follows. Assume that an individual $p_{selected}$ is found using the compatilibity as described in the Subsection 2.2. Next, randomly choose two positions $h \leq k$ in the $p_{selected}$ string and toss a fair coin. If the coin turns head, then the substring from $p_{selected}(h)$ to $p_{selected}(k)$ of $p_{selected}$ is replaced by the corresponding substring from $s^*(h)$ to $s^*(k)$ of s^*. If the coin turns tail, then the substring from $p_{selected}(1)$ to $p_{selected}(h-1)$ and from $p_{selected}(k+1)$ to $p_{selected}(n)$ are replaced by the corresponding substrings of s^*. The new string formed by the selective transfer process is denoted by c.

Selective transfer is similar to crossover in standard genetic algorithms, but there are some differences. The most important one is that selective transfer uses one-way relocation of substrings from the best individual to the individual selected. This is important because the selective transfer is much more schemata destructive than the standard crossover. The exploration or exploitation character of the selective transfer operator depends on both, the values of h and k, and the nature of the encoding mechanism of the chromosomes.

2.4 Mutation

There are many ways to perform mutation in evolutionary algorithms. For instance, in DE arithmetic combinations of selected individuals produces mutation [9]. More specifically, mutation in DE produces new individuals p_m as follows:

$$p_m = x_{m_1} + F \cdot (x_{m_2} - x_{m_3}) \qquad (7)$$

where random indexes $m_1, m_2, m_3 \in \{1, 2, ..., M\}$ and $F > 0$ is a real and constant factor during the entire evolution process, whose variation is proportional to $(x_{m_2} - x_{m_3})$.

In genetic participatory learning, participatory mutation generates a new individual p_m using the compatilibity between individuals $p_{selected}$ and c, and the arousal index as follows:

$$p_m = s^* + \rho^{1-a}(p_{selected} - c) \qquad (8)$$

Notice that the compatibility and arousal indexes control the extension of the variation of s^*. In this sense, participatory mutation is similar to differential evolution mutation. The exploration or exploitation character of mutation is modulated by compatibility.

The participatory genetic learning algorithm can be summarized as follows.

1. Let $objf \rightarrow f$
2. Generate S^t and S^{t+1} randomly
3. Set $a(0) \rightarrow 0$
4. While $t \leq t_{max}$ do
   Evaluation:
   choose $s \in S^{t+1}$ and $s^{'} \in S^t$ randomly.
   evaluate individuals of S^t and S^{t+1} using $objf$.
   choose the best individual s^*.
   Selection:
   compute $\rho^s(s, s^*)$ and $\rho^{s^{'}}(s^{'}, s_*)$.
   if $\rho^s \geq \rho^{s^{'}}$ then $p_{selected} = s$ else $p_{selected} = s^{'}$.
   Selective Transfer:
   choose h, k, $h \leq k$, and $r \in [0, 1]$ randomly.
   if $r \leq 1/2$ then $c = [s^*(1, 1:h), p_{selected}(1, h+1:k), s^*(1, k+1:n)]$; else $c = [p_{selected}(1, 1:h), s^*(1, h+1:k), p_{selected}(1, k+1:n)]$.
   Mutation:
   compute $\rho_m = \rho(p_{selected}, c)$
   compute $p_m = s^* + \rho_m^{1-a_{t+1}}(p_{selected} - c)$.
   if $objf(p_m)$ better than $objf(s^*)$ then $p_m \rightarrow S^{t+1}$.
5. Return the best individual.

3. COMPUTATIONAL RESULTS

This paper explores GPLAT, evolutionary participatory learning algorithms, and a genetic fuzzy system to develop a fuzzy rule-based model using actual data concerning an electrical maintenance problem. The application considered in this section uses the embedded genetic fuzzy rule-based system approach to learn the data base and a simple method to derive fuzzy rule bases of fuzzy rule-based models. We compare the performance of the genetic participatory algorithm (GPLAT) with the participatory algorithms DPLAX, GPLAX [6], and DPLAT [4], and with a state of the art genetic fuzzy system (GFS) approach of [5]. Essentially, a GFS is a fuzzy rule-based system together with a learning procedure based on genetic algorithms. For comparison purposes, we use the same enconding framework and data set of [5]. In the electric maintenance modeling addressed here, we assume four (alternatively, two) input variables and one output variable. The dataset (ELE) contains 1056 samples.

Table 1 shows the average of 12 runs of the DPLAT, GPLAT, DPLAX and GPLAX. The population size was 61 and evolution stops after 1000 generations. The selection of 2 inputs in the Wang-Mendel rule generation method (WM) was done as in [5, 10]. The results obtained by the GFS of [5] are shown in Table 2.

Looking at the results of Tables 1 and 2 we may conclude the following. The average mean-squared error (MSE) achieved by the different models evolved by the DPLAT, GPLAT, DPLAX and GPLAX were lower than the average values of the GFS of [5]. The 4 input model envolved by the DPLAT, GPLAT, DPLAX and GPLAX using WM with different number linguistic labels is more accurate than when the number of linguistic label are kept fixed. The lowest value of the MSE is achieved by GPLAX, except for WM(5) with 5 linguistic labels. Also, the standard deviation of the MSE of WM with variable number of labels is smaller, except for DPLAT and GPLAX.

In the 2 input case, GPLAT using WM performs best among the DPLAT, DPLAX, GPLAX, FSMOGFS and FSMOGFS+TUN from the point of view of the MSE and standard deviation. Further, the processing time spent by DPLAX to produce this result was 16, 59 seconds, whereas FSMOGFS+TUN spent 105, 3 seconds.

4. CONCLUSION

This paper has suggested GPLAT, a class of genetic learning algorithm based on participatory learning. The main

Method	WM(3)		WM(5)		WM(7)		WM		WM	
Inputs	4		4		4		4		2	
Rules	27		65		103		80		8	
DPLAT	MSE	Time	MSE	Time	MSE	Time	MSE	Time	MSE	Time
Mean	45648,42	45,21	23363,50	54,83	14006,13	84,03	12267,40	65,16	6438,34	37,45
SD	5106,75	1,11	2460,22	0,54	3054,52	1,35	2533,93	0,63	54,10	1,07
GPLAT	MSE	Time	MSE	Time	MSE	Time	MSE	Time	MSE	Time
Mean	47414,33	29,90	24521,58	55,19	15503,70	85,88	13581,03	72,37	6420,73	39,15
SD	6117,38	0,52	3758,45	0,92	4171,06	1,69	3768,39	3,11	52,76	0,64
DPLAX	MSE	Time	MSE	Time	MSE	Time	MSE	Time	MSE	Time
Mean	46828,17	31,81	24816,17	56,97	14870,96	84,51	12794,97	67,02	6460,96	16,59
SD	3890,26	0,72	4287,94	0,25	4791,31	1,47	3502,38	1,44	140,97	0,06
GPLAX	MSE	Time	MSE	Time	MSE	Time	MSE	Time	MSE	Time
Mean	45014,17	31,57	23769,67	55,58	13251,73	84,01	12058,16	67,84	6434,54	19,09
SD	5821,81	0,67	3421,24	0,51	3140,42	1,40	3153,25	0,90	136,18	0,20

Table 1: Average MSE of Participatory Evolutionary Algorithms for ELE dataset

Method	WM(3)	WM(5)	WM(7)	FSMOGFS	FSMOGFS^e	FSMOGFS+TUN	FSMOGFS^e+TUN^e
Input	4	4	4	2	2	2	2
Rules	27	65	103	10	9	9	8
Mean	192241	56135	53092	16018	16153	8803	9665
SD	9658	1498	1955	314	450	739	823

Table 2: Average MSE Values of Different Algorithms for ELE dataset[5]

characterisitc of participatory learning is the role that the current population plays in shaping evolution. The genetic participatory learning algorithm uses participatory selective transfer and selection operators to exchange information between individuals. Selection, selective transfer and compatibility help to improve the performance of learning and allow trade-offs between the exploration and exploiatation duirng generations. Mutation uses compatibility and arousal indexes to handle diversity. The approach was evaluated using actual data to model an electric maintenance problem and its performance was compared against alternative approaches. Computational results show that genetic participatory learning algorithms perform better than current state of the art genetic fuzzy systems approach. Further work shall test participatory evolutionary approaches with distinct datasets and applications, report statistical analysis, and perform theoretical analysis of the algorithms.

5. ACKNOWLEDGMENTS

The last author is grateful to the Brazilian National Research Council (CNPq) for grant 304596/2009-4.

6. REFERENCES

[1] A. Eiben and J. Smith, Introduction to Evolutionary Computing, Springer, Berlin Heidelberg, 2010.
[2] F. Rothlauf, Design of Modern Heuristics: Principles and Applications, Springer, Berlin Heidelberg, 2011.
[3] R. Yager, “Participatory Genetic Algorithms,” BISC Group List, message posted on August 29, 2000.
[4] Y. L. Liu and F. Gomide, “Evolutionary Participatory Learning in Fuzzy System Modeling,” Proc. IEEE Fuzzy Systems, India, submitted for publication, 2013.
[5] R. Alcalá, M. Gacto, and F. Herrera, “A Fast and Scalable Multiobjective Genetic Fuzzy System for Linguistic Fuzzy Modeling in High-Dimensional Regression Problems,” IEEE Transactions on Fuzzy Systems, vol. 19, no. 4, pp. 666-681, 2011.
[6] Y. L. Liu and F. Gomide, “Fuzzy System Modeling with Participatory Evolution,” Proc. Joint IFSA/NAFIPS Conference, Canada, submitted for publication, 2013.
[7] R. Yager, “A model of participatory learning,” IEEE Trans. Syst., Man, Cybern., vol. 20, no. 5, pp. 1229-1234, 1990.
[8] C. Birchenhall, N. Kastrinos, and S. Metcalfe, “Genetic algorithms in evolutionary modelling,” J. Evol. Econ., vol. 7, pp. 375-393, 1997.
[9] R. Storn and K. Price, “Differential Evolution - A simple and efficient heuristic for global optimization over continuous spaces,” J. Global Optimization, vol. 11, pp. 341-359, 1997.
[10] L. Wang and J. Mendel, “Generation fuzzy rules by learning from examples,” IEEE Trans. Syst., Man, Cybern., vol. 22, no. 6, pp. 1414-1427, 1992.