DEVELOPMENT OF A COMPUTATIONAL INTELLIGENCE COURSE FOR UNDERGRADUATE AND GRADUATE STUDENTS

Ganesh K. Venayagamoorthy

Real-Time Power and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
University of Missouri – Rolla, MO 65409, USA
gkumar@ieee.org

Abstract

This paper presents the design, implementation and experiences of a new three hour experimental course taught for a joint undergraduate and graduate class at the University of Missouri-Rolla, USA. This course is unique in the sense that it covers the four main paradigms of Computational Intelligence (CI) and their integration to develop hybrid algorithms. The paradigms covered are artificial neural networks (ANNs), evolutionary computing (EC), swarm intelligence (SI) and fuzzy systems (FS). While individual CI paradigms have been applied successfully to solve real-world problems, the current trend is to develop hybrids of paradigms, since no one paradigm is superior to the others in all situations. In doing so, we are able capitalize on the respective strengths of the components of the hybrid CI system and eliminate weakness of individual components. This course is an introductory level course and will lead students to courses focused in depth in a particular paradigm (ANNs, EC, FS, SI). The idea of an integrated course like this is to expose students to different CI paradigms at an early stage in their degree program. The paper presents the course curriculum, tools used in teaching the course and how the assessments of the students' learning were carried out in this course.

Introduction

A major thrust in the algorithmic development and enhancement is the design of algorithmic models to solve increasingly complex problems and in an efficient manner. Enormous successes have been achieved through modeling of biological and natural intelligence, resulting in "intelligent systems". These intelligent algorithms include neural networks, evolutionary computing, swarm intelligence, and fuzzy systems. Together with logic, deductive reasoning, expert systems, case-based reasoning and symbolic machine learning systems, these intelligent algorithms form part of the field of Artificial Intelligence (AI) [1]. Just looking at this wide variety of AI techniques, AI can be seen as a combination of several research disciplines, for example, engineering, computer science, philosophy, sociology and biology.

There are many definitions to intelligence. The author prefers the definition from [1] - Intelligence can be defined as the ability to comprehend, to understand and profit from experience, to interpret intelligence, having the capacity for thought and reason (especially, to a higher degree). Other keywords that describe aspects of intelligence include creativity, skill, consciousness, emotion and intuition. Computational Intelligence (CI) is the study of adaptive mechanisms to enable or facilitate intelligent behavior in complex, uncertain and changing environments. These adaptive mechanisms include those AI paradigms that exhibit an ability to learn or adapt to new situations, to generalize, abstract, discover and associate.

The four dominant computational intelligence paradigms are neural networks, evolutionary computing, swarm intelligence and fuzzy systems as illustrated in figure 1.

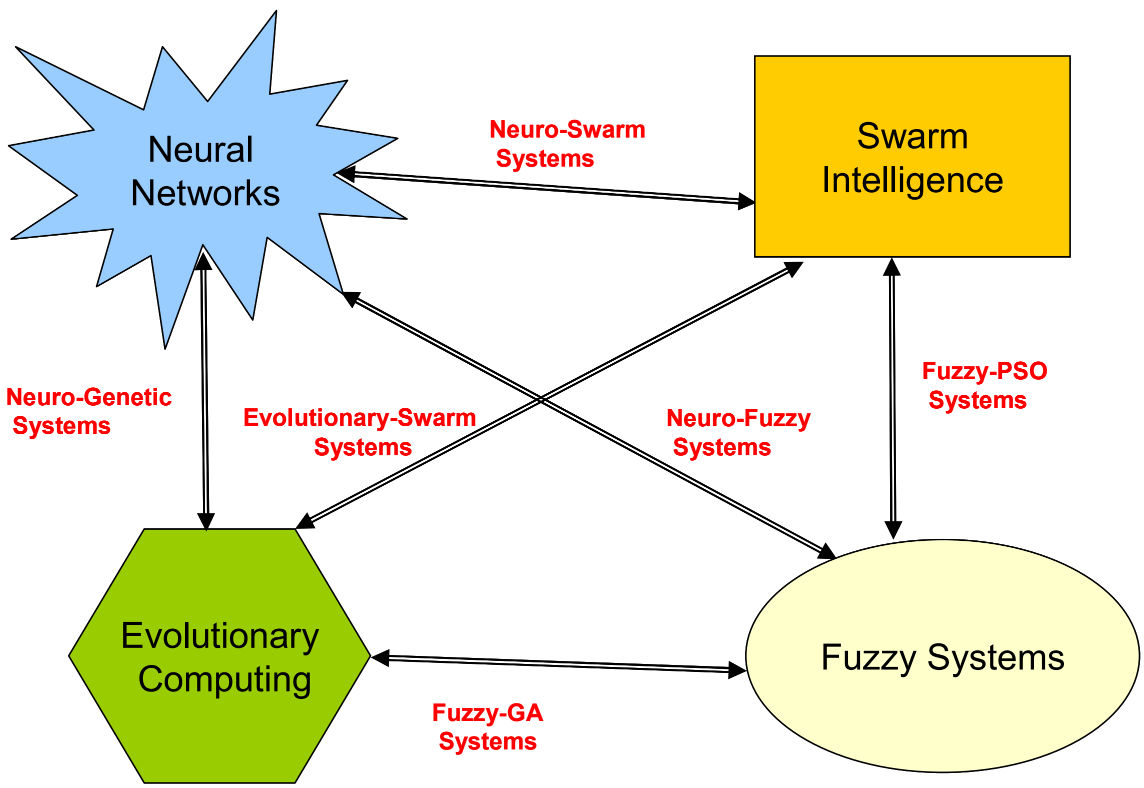

Figure 1: Four main paradigms of CI and possible hybrids

These paradigms can be combined to form hybrids as shown in figure 1 resulting in Neuro-Fuzzy systems, Neuro-Swarm systems, Fuzzy-PSO systems, Fuzzy-GA systems, Neuro-Genetic systems, etc. Designs and developments with hybrid algorithms already exist in literature [2, 3, 4].

The following sections describes the five main parts of the Computational Intelligence course taught at the University of Missouri-Rolla as an experimental course in the Spring semester of 2004. These parts are – neural networks, evolutionary computing, swarm intelligence and fuzzy systems. These are all of introductory nature. Thus, the course is offered at the 300 level allowing both the undergraduate and graduate students to enroll. The advantage of doing is that the undergraduate students are exposed to these emerging computational intelligence field; and for the graduate students, the course introduces them to four main paradigms and their interest in any of paradigms can be broaden by a full semester on either of the courses - neural networks, fuzzy logic, evolutionary computation, so on, which is offered at the University of Missouri-Rolla. The textbook used in teaching this course is entitled “Computational Intelligence” by A P

Engelbrecht [1]. The course was offered in three departments: Department of Electrical and Computer Engineering, Department of Mechanical Engineering and Department of Engineering Management.

Neural Networks

In this part, the topics covered include the artificial neuron model; supervised learning neural networks; unsupervised learning neural networks; Radial Basis Function (RBF) networks. The students are introduced to different neural network architectures and the class is focused on feedforward neural networks. The feedforward neural network and the conventional training algorithm known as the backpropagation [5-8] are described below. The incremental and batch training concepts are taught.

A feedforward neural network can consist of many layers as shown in figure 2, namely: an input layer, a number of hidden layers and an output layer. The input layer and the hidden layer are connected by synaptic links called weights and likewise the hidden layer and output layer also have connection weights. When more than one hidden layer exists, weights exist between the hidden layers.

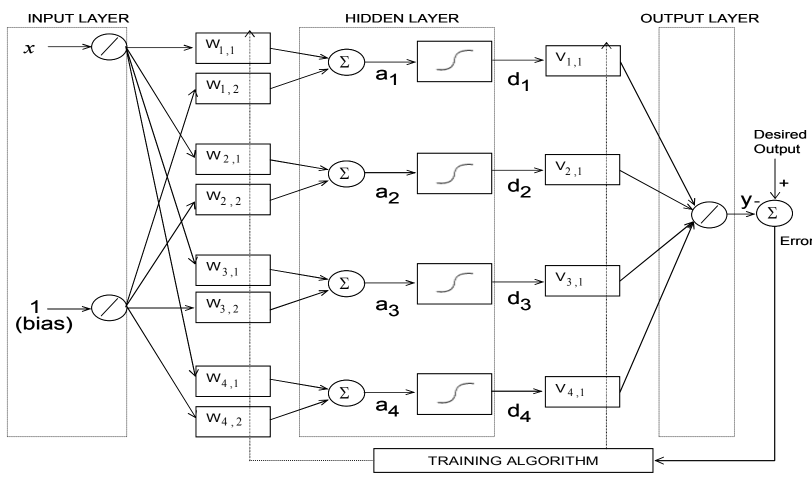

Figure 2: Feedforward neural network with one hidden layer

Neural networks use some sort of "learning" rule by which the connections weights are determined in order to minimize the error between the neural network output and desired output. The learning gain and momentum term have to be carefully selected to maximize accuracy, reduce training time and ensure global minimum. A JAVA based software developed for MLP neural networks developed by the author is used to teach the need to carefully select these parameters and their effects [9]. More details on neural networks can be found in [5].

Evolutionary Computation (EC)

In this part, the topics covered include Genetic Algorithms (GAs), Genetic Programming (GP), Evolutionary Programming (EP), Evolutionary Strategies (ESs). These algorithms are introduced and their numerous applications are demonstrated in class through examples. The differences between these algorithms, when and where these algorithms are applicable is emphasized in class. The main steps in EC algorithms are given below.

- *Initialize the initial generation of individuals.*
- *While not converged*
  - *i) Evaluate the fitness of each individual*
  - *ii) Select parents from the population*
  - *iii) Recombine selected parents using crossover to get offspring*
  - *iv) Mutate offspring*
  - *v) Select new generation of populations*

Swarm Intelligence

In this part, the topics covered include Particle Swarm Optimization (PSO), Ant Colony Optimization, Cultural and Differential Evolution. These algorithms are introduced and their numerous applications are demonstrated in class through examples. The differences between these algorithms, when and where these algorithms are applicable is emphasized in class. Emphasis is given the PSO algorithm and taught in detail. The PSO algorithm is described below.

Particle swarm optimization is an evolutionary computation technique (a search method based on natural systems) developed by Kennedy and Eberhart [10, 11]. PSO like a generic algorithm (GA) is a population (swarm) based optimization tool. However, unlike GA, PSO has no evolution operators such as crossover and mutation and more over PSO has less number of parameters. PSO is the only evolutionary algorithm that does not implement survival of the fittest and unlike other evolutionary algorithms where evolutionary operator is manipulated, the velocity is dynamically adjusted. More details and results are given in [2, 11].

Fuzzy Logic

In this part, the topics covered include Fuzzy Systems, Fuzzy Interference Systems, Fuzzy Controllers, Rough Sets. The design of a fuzzy room temperature controller and fuzzy cruise controller is taught in a step by step fashion through fuzzification process, interference engine, rule set and defuzzication process is carried out in class. More details on fuzz logic can be found in [1, 13].

Hybrid Algorithms

In this part, the students train a feedforward neural network using GA, ES and PSO algorithms, and compare the training results with that of the backpropagation algorithm. In addition, they do a survey of neuro-fuzzy controllers and write a report on how fuzzy memberships can be used within a neural network structure. Some hybrid algorithms are described in [13].

Course Assessment

The course assessment is made of:

- 25% Homework/Assignments

- 25% Midterm Exam
- 40% Group Project Report
- 5% Class presentation
- 5% Quizzes

The course grading is based on six homeworks where they have to write MATLAB/C programs to train feedforward neural networks with the backpropagation algorithms, GA, ES and PSO, train a learning vector quantization (LVQ) network using competitive learning and the last one write a report on neuro-fuzzy structures.

Projects are carried out involving at least two of four CI paradigms and done in groups. Each group is made of four students. The students are encouraged to start early in the semester.

Conclusion

This computational intelligence is unique in the way it covers four major paradigms of CI. The students that took class consisted of two undergraduates and ten graduate students (MS and PhD). The feedback received from students is mainly the great exposure to the different paradigms of computational intelligence in a one semester. One of the undergraduate students who took the class in Spring 2004 is currently doing research in the area of evolvable hardware and had his first publication in the Fall semester of 2004 [12]. The other undergraduate student is interested in doing a MS degree in the area of computational intelligence. All the graduate students who took the course are pursuing a thesis involving one or more paradigm(s) of computational intelligence. The offering of the CI experimental course at the 300 level allowing both undergraduate and graduate students has been a successful effort. The experimental course will be offered once more in the Fall semester of 2005 and thereafter made a permanent course in the Electrical Engineering and Computer Engineering Curriculum at the University of Missouri-Rolla.

References

[1]. A. Engelbrecht, *Computational Intelligence: An Introduction*, John Wiley & Sons, Ltd, England. ISBN 0-470-84870-7.

[2]. V. G. Gudise, G. K. Venayagamoorthy, "Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks", *IEEE Swarm Intelligence Symposium*, Indianapolis, IN, USA, April 24 -26, 2003, pp.110 - 117.

[3]. G. K. Venayagamoorthy, "Adaptive critics for dynamic particle swarm optimization", *IEEE International Symposium on Intelligent Control*, September 2 – 5, 2004, Taipei, Taiwan, pp. 380 -384.

[4]. X. Cai X, N. Zhang, G. K. Venayagamoorthy, D. C. Wunsch, "Time series prediction with recurrent neural networks using a hybrid PSO-EA algorithm", *IEEE-INNS International Joint Conference on Neural Networks*, Budapest, Hungary, July 24 -29, 2004.

[5]. S Haykin, *Neural networks: A comprehensive foundation,* Prentice Hall, 1998, ISBN 0-1327-3350-1.

[6]. P. J. Werbos, "Backpropagation through time: what it does and how to do it". *Proceedings of the IEEE*, Vol: 78:10, Page(s): 1550 –1560, Oct. 1990.

[7]. P. J. Werbos, *The roots of backpropagation*, New York: Wiley, 1994.

[8]. D. E.Rumelhart, G. E. Hinton and R. J. Williams, "Learning representations by back-propagating errors", *Nature*, Vol.323, pp.533-6, 1986.
[9]. G.K. Venayagamoorthy, "Teaching Neural Networks Concepts and their Learning Techniques", *39th ASEE Midwest Section Meeting*, Pittsburg, Kansas, September 29 to October 1, 2004.
[10] J. Kennedy and R. Eberhart, "Particle swarm optimization", *Proceedings of IEEE International Conference on Neural Networks*, Perth, Australia. Vol. IV, pp. 1942–1948.
[11] J. Kennedy and R. C. Eberhart, *Swarm intelligence*, Morgan Kaufmann Publishers, 2001.
[12] P. Moore, G.K. Venayagamoorthy, "Evolving Digital Circuits Using Hybrid Particle Swarm Optimization and Differential Evolution", *Conference on Neuro-Computing and Evolving Intelligence*, Auckland, New Zealand, December 13 – 15, 2004, pp. 71 -73.
[13] S. Rajasekaran, G. A. Pai, *Neural Networks, Fuzzy Logic, and Genetic Algorithms*, Prentice Hall, 2003, ISBN-81-203-2186-3.

Acknowledgment

The support from the National Science Foundation under CAREER Grant: ECS # 0348221 is gratefully acknowledged for this work. The author is grateful for the following departments at the University of Missouri-Rolla for offering the experimental course on computational intelligence to their students: Electrical and Computer Engineering, Mechanical Engineering and Engineering Management.

Biography

Ganesh Kumar Venayagamoorthy received the B.Eng. (Honors) degree with a first class honors in Electrical and Electronics Engineering from the Abubakar Tafawa Balewa University, Bauchi, Nigeria, and the MScEng and PhD degrees in Electrical Engineering from the University of Natal, Durban, South Africa, in March 1994, April 1999 and February 2002, respectively. He was a Senior Lecturer at the Durban Institute of Technology, South Africa prior to joining the University of Missouri-Rolla (UMR), USA as an Assistant Professor in the Department of Electrical and Computer Engineering in May 2002. He is the Director of the Real-Time Power and Intelligent Systems Laboratory at the UMR. His research interests are in computational intelligence, power systems, evolving hardware and signal processing. He has published over 120 papers in refereed journals and international conferences. Dr. Venayagamoorthy is a 2004 NSF CAREER award recipient, the 2004 IEEE St. Louis Section Outstanding Young Engineer, the 2003 International Neural Network Society (INNS) Young Investigator award recipient, a 2001 recipient of the IEEE Computational Intelligence Society (CIS) summer research scholarship and the recipient of five prize papers with the IEEE Industry Application Society (IAS) and IEEE CIS. He is an Associate Editor of the IEEE Transactions on Neural Networks. He is a Senior Member of the IEEE and the South African Institute of Electrical Engineers, a Member of INNS and the American Society for Engineering Education. He is currently the IEEE St. Louis CIS and IAS Chapters Chair, the Chair of the task force on Intelligent Control Systems and the Secretary of the Intelligent Systems subcommittee of IEEE Power Engineering Society. He was Technical Program Co-Chair of the International Joint Conference on Neural Networks (IJCNN), Portland, OR, USA, July 20 – 24, 2003 and the International Conference on Intelligent Sensing and Information Processing (ICISIP), Chennai, India, January 4 – 7, 2004.