

Artificial Intelligence in Computer Science Teaching and Research

Nelishia Pillay

School of Computer Science
University of KwaZulu-Natal
KwaZulu-Natal, South Africa
+ 27 033 2605644

pillayn32@ukzn.ac.za

ABSTRACT

This paper examines the current and future roles of Artificial Intelligence (AI) in Computer Science (CS) teaching and research. The paper firstly looks at how AI fits into the CS undergraduate and postgraduate teaching curriculum in South African universities. In addition to this AI can also be used in the instruction of CS courses. At the intersection of AI teaching and research, is research into the instruction of AI courses. An overview of this is provided. Masters and Phd degrees in Artificial Intelligence in South African universities are usually research based and do not include coursework. The paper describes the current and future trends in Artificial Intelligence research.

Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]:
Computer Science Education

General Terms

Documentation.

Keywords

Artificial intelligence teaching, artificial intelligence research.

1. INTRODUCTION

Artificial intelligence is a growing subfield of Computer Science. The term “Artificial Intelligence” was coined in 1956 at a conference aimed at using computers to simulate human intelligence. Game playing and theorem proving are two of the earlier attempts at getting computers to think intelligently. Search and logic formed the basis of the first AI systems. The limitations of these methods were identified and led to further development of the field and areas such as expert systems, neural networks, evolutionary algorithms, swarm intelligence and agent-based

technologies, just to name a few. The contribution of the paper is an overview of the current status of the field of AI and new directions in this domain. As such the paper serves as a resource for new and current educationalists and researchers in the field. This paper takes a step back and looks at the role that AI plays in Computer Science teaching and research since its inception, as well as areas for future development. The paper describes how AI is incorporated into the Computer Science curriculum. The paper also discusses how AI is used in the instruction of Computer Science courses. Research in artificial intelligence has grown rapidly since its inception. The three main directions of research are the application of AI techniques, the improvement of existing techniques and the development of new techniques. The following sections provide an overview of the current and future role of AI in Computer Science teaching and research.

2. ARTIFICIAL INTELLIGENCE IN COMPUTER SCIENCE TEACHING

2.1 AI in the Computer Science Curriculum

The revised Computing Curricula report for Computer Science composed by the ACM and IEEE [1] describes which AI (Intelligent Systems) topics are core and which are elective. These are listed in Table 1. The main aim of incorporating AI into the Computer Science undergraduate curriculum is to provide students who are majoring in CS with an introduction to each of the subfields of AI. A course in artificial intelligence is usually in the final year of study, i.e. the third year of a three year degree or the fourth year of a four year degree. This serves as a foundation for students who will continue with further studies in CS and AI, while at the same time arming students, who will move on to industry, with alternative methodologies that can be used to solve problems in various application domains. Earlier AI courses at this level covered the core topics, however currently some advanced topics are also included in the curriculum. Topics traditionally taught in an undergraduate course are uniformed search, informed search, game playing, logic and resolution, expert systems, inference, uncertainty, fuzzy logic, planning, neural networks, genetic algorithms, agents, and natural language processing [2]. Some universities include modules on a specific topic, such as expert systems, in addition to a general AI module, at the undergraduate level. The outcomes of an AI course at this level include the understanding of the concepts, knowledge of the advantages and limitations of the different methodologies and an

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Conference'04, Month 1–2, 2004, City, State, Country.

Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

ability to implement the methodologies for specific application domains.

Table 1. Intelligent Systems Topics[1]

Topic	
Fundamental Issues, e.g. history of artificial intelligence, ethical issues in AI.	Core
Basic Search Strategies, e.g. problem spaces, brute force search, two player games.	Core
Knowledge Based Reasoning, e.g. propositional and predicate calculus.	Core
Advanced Search, e.g. hill climbing, genetic algorithms, simulated annealing.	Elective
Advanced reasoning, e.g. reasoning on action and change, uncertainty.	Elective
Agents, e.g. agent architectures, agent theory, believable agents.	Elective
Natural Language Processing, e.g. parsing algorithms, language translation.	Elective
Machine Learning, e.g. inductive learning, learning neural networks.	Elective
Planning Systems, e.g. planning as search, planning and robotics.	Elective
Robotics, e.g. sensing, robot programming, world models.	Elective
Perception, e.g. image formation, image and object detection.	Elective

The advanced topics are usually taught at a postgraduate level, such as an Honours degree. These courses usually focus on a particular AI area such as, neural networks, expert systems, or genetic programming [3]. An in-depth study of the particular topic is provided. According to Congdon [3], it is essential that students are given exposure to developing systems for the particular domain. The course should also introduce students to the research methodology for the particular area [4]. Students must be able to pose a research question, conduct the required experiments and present their findings in a research paper [3]. The methodology of comparing the performance of the particular method with other AI techniques and testing the statistical significance of the result should also be included in the syllabus [3]. This sets the stage for students wanting to pursue research at a Masters level in this domain.

A survey of Artificial Intelligence courses at ten South African universities was conducted via the Internet. Two of the universities do not offer any Artificial Intelligence at the undergraduate or Honours level. An additional three universities do not teach AI in the undergraduate degree but include AI modules in the Honours curriculum. Two universities offer a final year undergraduate course including AI as one of the topics. The remaining three universities offer a final year AI module covering the core AI topics and some of the advanced topics.

Eight of the universities offer AI courses at Honours level. These topics provide an in-depth look at advanced AI topics, e.g. neural networks, evolutionary algorithms, particle swarm optimization, intelligent systems design.

2.2 AI in the Instruction of Computer Science Courses

Artificial intelligence plays an important role in the instruction of Computer Science courses. According to Kumar [5], AI courses are often used as a means of enabling students to develop advanced programming skills. Programming AI systems require students to draw on their knowledge of programming and data structures from the previous two years of their undergraduate degree, and optimize the amount of memory used while at the same time minimizing processing time when programming.

Artificial intelligence has also been used in teaching other Computer Science topics to motivate students. Vaughn [6] has suggested using AI techniques such as expert systems or neural networks to detect when a system is under attack in a course on computer security. According to Soh [7], a multi-agent system (MAS) can be used to analyze network congestion in a course on computer networking. Wick et al. [8] use a genetic algorithm to illustrate the difference between the template and strategy design patterns in a course on design patterns. The authors have also used genetic algorithms to show the difference between inheritance and delegation in an introductory programming course.

Intelligent teaching agents [9] or intelligent tutoring systems (ITSs) [10] are online systems that provide individualized tuition to students and as such play the role of an online tutor or teacher. These systems maintain a model of the student's knowledge in the particular domain, e.g. errors made, misconceptions. Based on the student model the intelligent tutoring system decides on what to teach the student next and what instructional methodology to employ. An ITS usually has an agent-based architecture, with the function of each module being performed by an intelligent agent. Artificial intelligence techniques form the basis of each module. An agent communication language is used for communication between the modules. An example of such an architecture, namely a generic architecture for intelligent programming tutors [11], is depicted in Figure 1.

There has been a fair amount of research into the development of intelligent tutoring systems for Computer Science courses. Most of the earlier work has focused on tutoring systems for teaching programming. A number of ITSs were developed to teach programming, e.g. PROUST [12] and INTELLITUTOR [13]. Anderson et al. [14] created a tutor to assist students in learning how to program in Lisp. Alpert et al. [15] and Xu et al. [16] have developed intelligent tutoring systems, namely, MoleHill and SIPLeSII to assist students in learning to program in Smalltalk. Based on these studies Pillay ([11], [17]) proposes a generic architecture for intelligent programming tutors. Devezdic et al. [18] have created an ITS, FLUTE, to tutor students in the domain of the theory of formal languages. Topics covered include finite automata, pushdown automata, regular grammars, context-free grammars and the relationship between grammars and automata.

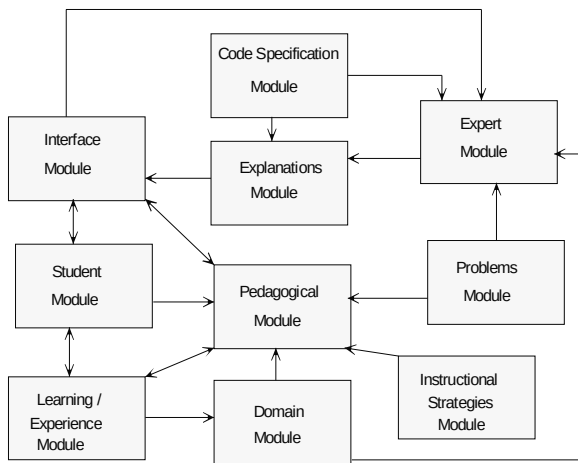


Figure 1. A generic architecture for intelligent programming tutors [11]

Most of these ITSs have been developed and/or tested in a research laboratory environment and have not as yet made their move into the classroom. One of the areas for further development would be bridging the gap between research and classroom use of ITSs.

3. RESEARCH INTO TEACHING ARTIFICIAL INTELLIGENCE

At the intersection of AI teaching and research, is research into improving the teaching of AI. According to Russell et al. [19], AI courses are challenging to teach. There has been a fair amount of research into the teaching of AI courses.

Hill et al. [20] take an agent-based approach. Agents are required to solve tasks in a particular environment, in this case Wumpus world, using introductory AI techniques including state space representation, search, game playing and first order logic. Students are so caught up in creating the agent environment that they do not realize that they are learning the different AI concepts at the same time.

Imberman [21] uses LEGO robots to teach neural networks and the backpropagation learning algorithm. The use of LEGO robots to visualize the process has enabled students to clearly distinguish between training and testing a neural network, processes which are usually confused. Kumar [22] also proposes using LEGO robots to teach AI topics such as search, expert systems, game playing and neural networks.

Pillay [23] found that in teaching abstract concepts, as in the case of a first course on genetic programming, the use of animation in lectures to illustrate such concepts assists students to grasp them more easily. For example, illustrating step by step the application of a genetic operator such as crossover.

A lot of research on the teaching of AI has also focused on choosing programming projects that students would find motivating. For example, Popyack [24], uses a Blackjack project to teach Markov processes. Similarly, Eskin et al. [25] and Kolas et al. [26] use the game Othello to teach genetic programming and introductory AI concepts respectively.

There is still a lot of scope in this research area to identify learning difficulties experienced by students and develop more effective teaching strategies for the different AI topics.

4. DIRECTIONS IN ARTIFICIAL INTELLIGENCE RESEARCH

Research in the field of artificial intelligence began with developing methods for certain applications, e.g. game playing, theorem proving. As these techniques were applied limitations of the techniques were identified which led to improvements in the method, e.g. informed search was an improvement of uninformed search. In some cases the identification of such limitations has resulted in the development of new AI techniques, e.g. the limitations of expert systems have led to the discovery of the field of machine learning. Thus, the three main trends in artificial intelligence research are applications of AI to specific problem domains, the further development of existing techniques and the derivation of new techniques.

Artificial intelligence has been successfully used to solve problems in numerous application domains, e.g. simulated pilots, doctor advisory system to assist students with identifying drug therapies for AIDS patients, search engines, computer games, adaptive user interfaces, personalized assistants, natural-language comprehension ([3], [27], [28]). As new fields in Computer Science, medicine, etc are discovered the application of AI grows. This is evident from the latest proceedings of the World Congress on Nature and Biologically Inspired computing [29], in which a majority of the papers reported on the application of artificial intelligence rather than theoretical aspects.

Further development of existing techniques basically involves studying why an AI method works and also determining its limitations and finding methods to overcome them. For example, how the genetic operators in a genetic programming system work and how they can be improved [30]. One of the limitations of genetic programming is its susceptibility to converge prematurely. Thus, there has been a lot of research into methods for preventing such premature convergence ([31], [32]).

Most AI techniques model human intelligence or biological/natural processes to solve problems. The evolution of such techniques is constantly growing as a result of existing methods not being able to solve problems in a new application or the remediation of the limitations of existing techniques. Examples of such newly developed AI methods include bee colony optimization [33] and the developmental approach which mimics the processes of cell biology [34].

Two additional areas of artificial intelligence research which need attention is the use of parallel processing to reduce the runtime of AI systems and bridging the gap between AI research labs and the "real-world" application environment. Most artificial intelligence techniques, such as genetic algorithms, are computationally expensive. With the introduction of multi-core processors, parallel processing can be used to reduce the runtime of such techniques. While a lot of AI research is being conducted in various application domains, in some cases the systems developed are prototypes which are adequate for research but cannot be utilized as a fully-fledged system. This gap between research based and user-friendly AI systems needs to be bridged.

5. CONCLUSION

It is evident from this paper that Artificial Intelligence plays a significant role in both the teaching of Computer Science and Computer Science research, and will continue to do so in the future. Artificial intelligence forms a crucial component of both the Computer Science undergraduate and postgraduate curriculum and as such serves as a foundation for students wishing to pursue research in this domain at the Masters level. In the area of teaching AI, there is a lot of scope for research into the identification of learning difficulties experienced by students and the identification of instructional strategies to improve the teaching of AI. Three continuing trends in AI research are the application of AI, further development of existing AI methods and the derivation of new AI techniques. The role that multi-core technologies can play in reducing the processing overhead of certain AI techniques needs to be investigated. In addition to this ways of bridging the gap between AI research prototypes and usable systems must be identified.

6. REFERENCES

- [1] Computer Science Curriculum 2008: An Interim revision of CS 2001. ACM, IEEE Computer Society, <http://www.acm.org/education/curricula/ComputerScience2008.pdf>.
- [2] Merzbacher, M. 2001. Open Artificial Intelligence – One Course for All. *SIGCSE Bulletin-inroads*, SIGCSE '01 Proceedings, 33, 1 (March 2001), 110-113, ACM.
- [3] Congdon, C., B. 2001. Machine Learning for the Masses. *Intelligence: New Visions of AI in Practice*, 12, 2 (Summer 2001), 15-16, ACM.
- [4] Lankewiz, L., B. 2001. Undergraduate Research in Genetic Algorithms. *SIGCSE Bulletin-inroads*, SIGCSE '01 Proceedings, 33, 1 (March 2001), 114-118, ACM.
- [5] Kumar, D. 2000. How Much Programming? What Kind? *Intelligence: New Visions of AI in Practice*, 11, 4 (Winter 2000), 15-16, ACM.
- [6] Vaughn, R. 2000. Application of Security to the Computing Science Classroom. *SIGCSE Bulletin-inroads*, SIGCSE '00 Conference Proceedings, 23, 1(March 2000), 90-94, ACM.
- [7] Soh, L. 2004. Using Game Days to Teach a Multiagent System Class. *Inroads- SIGCSE Bulletin*, SIGCSE '04 Proceedings, 36, 1(March 2004), 219-223, ACM.
- [8] Wick, M. R., and Phillips, A., T. 2002. Comparing the Template Method and Strategy Design Patterns in a Genetic Algorithm. *Inroads-SIGCSE Bulletin*, 34, 4 (December 2002), 76-80, ACM.
- [9] Rosbottom, J. and Moulin, C. 1998. Using Intelligent Agents to Change the Delivery of Education. *SIGCSE Bulletin-inroads*, ITiCSE '98 Proceedings, 30, 3(September 1998), 303, ACM.
- [10] Beck, J., and Stern, M.K. 1999. Bringing Back the AI to AI&ED. In *Artificial Intelligence in Education: Open Learning Environments: New Computational Technologies to Support Learning, Exploration and Collaboration*, S. P. Lajoie, M. Vivet, Ed. IOS Press, 233-240.
- [11] Pillay, N. 2003. Developing Intelligent Programming Tutors for Novice Programmers. *SIGCSE Bulletin-inroads*, 35, 2, 78-82, ACM.
- [12] Johnson, W. L. 1990. Understanding and Debugging Novice Programs. In *Artificial Intelligence and Learning Environments*, W.J. Clancy, E. Soloway, Ed. MIT Press, 51-97.
- [13] Ueno H. 2000. A Generalized Knowledge-Based Approach to Comprehend Pascal and C Programs, *IEICE Transactions on Information Systems*, E81-D, 12, 1323-1329, IOS Press.
- [14] Anderson J.R., Boyle C.B., Corbettm A.T., Lewis M.W. 1990. Cognitive Modelling and Intelligent Tutoring, in *Artificial Intelligence*, Clancy, Ed. Elsevier, 7- 49.
- [15] Alpert , S. R., and Singley M. K., Carroll, J., M. 1999. Multiple Instructional Agents in an Intelligent Tutoring System. In *Proceedings of the AIED '99 - Workshop on Animated and Personified Pedagogical Agents* (Le Mans, France, July 1999).
- [16] Xu, S., and Chee, Y., S. 1999. SIPLES-II: An Automatic Program Diagnoses System for Programming Learning Environments. In *Artificial Intelligence in Education- Open Learning Environments: New Technologies to Support Learning, Exploration, and Collaboration*, S.P. Lajoie and M. Vivet, Ohmsha, Ed. IOS Press, 397 - 404.
- [17] Pillay, N. 2000. A Generic Architecture for the Development of Intelligent Programming Tutors. *International Journal of Continuing Engineering Education and Life Long Learning*, 10, 1-4, 275-285, Inderscience.
- [18] Devedzic, V., Debenham, J., and Popvic D. 2000. Teaching Formal Languages by an Intelligent Tutoring System. *Educational Technology and Society*, 3, 2, ISSN 1436-4522.
- [19] Russell, I., Markov, Z., and Neller, T. 2006. Teaching AI Through Machine Learning Projects. *Inroads-SIGCSE Bulletin*, ITiCSE '06 Proceedings, 38, 3(September 2006, 323), ACM.
- [20] Hill, J., M., D. and Alford, K., L. 2004. A Distributed Task Environment for Teaching Artificial Intelligence with Agents. *SIGCSE Bulletin*, SIGCSE '04 Conference Proceedings, 36, 1(March 2004), 224-228, ACM.
- [21] Imberman, S., P. 2003. Teaching Neural Networks Using LEGO Handy Board Robots in an Artificial Intelligence Course. *SIGCSE Bulletin – inroads*, SIGCSE '03 Proceedings, 35, 1 (March 2003), 312-316, ACM.
- [22] Kumar, A., R. 2005. LEGO Robots for AI. *Inroads-SIGCSE Bulletin*, ITiCSE 2005 Proceedings, 37, 3(September 2005), 418, ACM.
- [23] Pillay, N. 2004. A First Course in Genetic Programming. *Inroads- SIGCSE Bulletin*, 36, 4 (December 2004), 93-95, ACM.
- [24] Popyack, J., L. 2009. Blackjack Playing Agents in an Advanced AI Course. *Inroads-SIGCSE Bulletin*, ITiCSE 2009, 41, 3(September 2009), 208-212, ACM.

- [25] Eskin, E. and Siegel, E. 1999. Genetic Programming Applied to Othello: Introducing Students to Machine Learning Research. *SIGCSE Bulletin-inroad*, SIGCSE '99 Proceedings, 31, 1(March 1999), 242-246, ACM.
- [26] Kolas, O. and Farup, I. 2003. Increasing Assignment Motivation Using a Game AI Tournament. *Inroads – SIGCSE Bulletin*, ITiCSE 2003 Proceedings, 35, 3 (September 2003), 269, ACM.
- [27] Meeden, L. 1999. IAAI Applications Point the Way. *Intelligence: New Visions of AI in Practice*, 10, 1 (Spring 1999), 6, ACM.
- [28] St. Amant, R. and Young, R., M. 2001. Artificial Intelligence and Interactive Entertainment, *Intelligence: New Visions of AI in Practice*, 12, 2 (Summer 2001), 17-19, ACM.
- [29] Abraham, A., Carvalho, A., Herrera, F., and Pai, V. 2009. *Proceedings of the 2009 World Congress on Nature and Biologically Inspired Computing*, IEEE, ISBN: 978-1-4244-5612-3.
- [30] White, D., R., and Poulding, S. 2009. A Rigorous Evaluation of Crossover and Mutation in Genetic Programming. Genetic Programming. *Lecture Notes in Computer Science*, 5481/2009, 220-231, Springer-Berlin.
- [31] Andre, J., Siarry P., and Dognon T. 2001. An Improvement of the Standard Genetic Algorithm Fighting Premature Convergence in Continuous Optimization. *Advances in Engineering Science*, 32, 1, 49-60, Elsevier.
- [32] Mawhinney, D., and Ciesielski, V. 2002. Preventing Early Convergence in Genetic Programming by Replacing Similar Programs. In *Proceedings of the 2002 Congress on Evolutionary Computation*, 67-72, IEEE.
- [33] Narasimhan, H. 2009. Parallel Artificial Bee Colony (PABC) Algorithm. In *Proceedings of the 2009 World Congress on Nature and Biologically Inspired Computing (NABIC 2009)*, 306-311, IEEE.
- [34] Pillay, N., and Banzhaf, W. 2008. A Developmental Approach to the Uncapacitated Examination Timetabling Problem. *Lecture Notes in Computer Science*, 5199, 276-285, Springer-Berlin.