

Galapagos

Studying the Educational Value of Genetic Algorithms

Andy Meneely
Department of Computer Science
Senior Project
Calvin College
2005-2006

<http://GalapagosGA.sourceforge.net>

Introduction

The focus of this project is to study the value of genetic algorithms from a pedagogical standpoint. One of my motivations for this project is that genetic algorithms are not considered to be part of the standard curriculum, yet they contain some elements in them which are both interesting and teachable. Students are not exposed to many stochastic algorithms, nor do they usually see algorithms which are highly adaptable to a wide range of problems. I find them to be teachable because they are built on a paradigm which students are familiar with: evolution through natural selection.

In addition to being teachable, using genetic algorithms typically involves heavy use of a framework. In most of the curriculum, students develop programming projects which are composed of mostly their code. They may use libraries for specialized purposes, but their code still does most of the work. With genetic algorithms, students must use a framework which means that they must (1) understand the overall structure of the framework, and (2) know where to insert their code. After working in industry on projects which relied heavily on frameworks, I believe this aspect of genetic algorithms complements our current curriculum.

Background

As far as my research shows, I don't know if anyone has approached genetic algorithms this way. For the most part, evolutionary computation in general is not very widely studied. According to the leading AI textbook "Artificial Intelligence: A Modern Approach", genetic algorithms are not exactly mainstream:

“At present, it is not clear whether the appeal of genetic algorithms arises from their performance or from their aesthetically pleasing origins in the theory of evolution. Much work remains to be done to identify the conditions under which genetic algorithms perform well.” (Russell 119)

Encountering this general sentiment of genetic algorithms was what motivated my approach.

Design

Since genetic algorithms remain independent of the problems they are solving, it makes little sense to have students build these from scratch. Most research done on genetic algorithms use highly extensible frameworks for their work, anyway. As I set out to study genetic algorithms on my own, I

found that most of these frameworks were highly flexible and highly optimized, but not built for usability. If students were to use genetic algorithms, they would be turned off by the numerous programming details involved. Building my own genetic algorithm framework with usability as its highest priority became my project – and I called it Galapagos.

Results

The exact design of Galapagos was not set in stone when I first began (ala “Waterfall”). Instead, I used Test-Driven Development as my development model to build this framework. In order to motivate the flexibility of the framework, I had multiple example problems that I unit-tested throughout the process. By the end of the first semester, I had my framework in stable condition.

The second semester was spent building the pedagogically motivated add-ons to Galapagos (e.g. XML compatibility) and writing documentation and tutorials for the entire project. The entire focus of the project for second semester was being able to use Galapagos to teach students about genetic algorithms and frameworks in an accessible way.

Discussion

Once I had my framework, add-ons, documentation, and tutorials, I had to determine if my efforts were fruitful. I began to do a series of mock laboratories with students and professors to test out my framework. For the first lab, I chose professors and seniors to try it out since I figured they would not get caught up in some of the details while also providing helpful feedback.

The first mock laboratory was only a half-success: the framework held up, but the tutorials were lacking. I decided to spend a considerable amount of time revising the tutorials, while also making a few changes to the framework in terms of set-up. The second mock lab I did was also with a senior – since I had rewritten the tutorials, I wanted to see if I was on the right track – and it was a huge success. My student asked great questions and I found him very interested in the topic. At the time of this writing, I have done one more lab with a sophomore – which was a success – and I plan on doing another lab with a freshman next week.

Conclusion

I believe that genetic algorithms *are*, in fact, very useful for teaching students about frameworks. They do require a small amount of set-up time and explaining time, but I believe this is

required for all laboratory exercises in Computer Science. The framework that I produced is extensible enough for zealous students to use it for their own studies. If future CS majors would like to study Genetic Algorithms, I would highly recommend them using Galapagos.

Galapagos as an Open Source project will continue beyond this year. I hope to refactor some of the design patterns used in this framework to increase usability and to simply make the code more flexible. I will most likely write a series of tutorials in addition to the two I have written and revised. Lastly, I hope that other schools pick up Galapagos and try to use it, in which case I would collaborate with them on it.

References

Eiben, A E., and J E. Smith. Introduction to Evolutionary Computing. New York City: Springer, 1998.

Russell, Stuart, and Peter Norvig. Artificial Intelligence: A Modern Approach. 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2003.

Appendix

Galapagos is an ongoing project, so the website and the code will evolve after the year has finished. A snapshot of the website will be handed in along with a JAR file of the code, but the best way to see the current state of the project is to visit its website at <http://GalapagosGA.sourceforge.net>. The SourceForge website supports browsing of the CVS repository so that anyone can view the code.