Summer 2011

CS 766: Evolutionary Computation

John C. Gallagher
Wright State University - Main Campus, john.gallagher@wright.edu

Follow this and additional works at: http://corescholar.libraries.wright.edu/cecs_syllabi
Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation
http://corescholar.libraries.wright.edu/cecs_syllabi/624

This Syllabus is brought to you for free and open access by the College of Engineering and Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact corescholar@www.libraries.wright.edu.
Objectives and Goals
This course explores evolutionary computation from a historical, theoretical, and application viewpoint. An overview of the most common evolutionary search techniques is presented. Specific topics include in the overview are: genetic algorithms, evolutionary programming, evolutionary strategies, and genetic programming. The fundamental issues driving the choice of problem representation and specific genetic operators are discussed. Various applications of evolutionary computation to problems in control, optimization, and pattern recognition are examined.

At the end of this course you should be able to read and critically evaluate most neural network papers published in major journals in the area. In addition, you should be able to implement evolutionary computation techniques for a variety of applications.

Meeting Time and Place
Monday and Wednesday, 4:10 PM – 5:25 PM
155 Russ Hall

Instructor
John C. Gallagher
352 Russ Engineering
john.gallagher@wright.edu (email preferred to phone)

Office Hours
3:00 – 4:00 PM and 5:30 – 6:30 PM Mondays and Wednesdays

Textbook
Introduction to Evolutionary Computing
A. E. Eiben and J. E. Smith
Springer-Verlag, New York, 2007
ISBN: 978-3-540-40184-1

Programming Languages and Tools
The course will be programming language agnostic. You may complete projects and assignments using any programming environment you like, so long as the instructor has a way to compile and run your assignments. Specific choices for programming tools will be discussed during course lectures.
Grading

Student grades will be determined by assessment of each of the following:

i. ability to discuss the relative merits of contemporary methods in Evolutionary Computation (EC)
ii. ability to implement and debug EC systems
iii. ability to identify and articulate open issues yet to be addressed

Each student will be provided with ample opportunity to demonstrate these abilities through written examinations, programming assignments, oral presentations, and a term project. We will use a standard 90/80/70/60 scale for letter grades.

Grades will be computed as follows:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-3 Homework Exercises</td>
<td>25%</td>
</tr>
<tr>
<td>1 Midterm Examination</td>
<td>25%</td>
</tr>
<tr>
<td>1 Final Examination</td>
<td>25%</td>
</tr>
<tr>
<td>1 Course Project</td>
<td>25%</td>
</tr>
</tbody>
</table>

Term Project / Presentation
Students will be required to complete individual term projects in which EC methods are applied to a practical problem of interest to the student. Students will be expected not only to create and test solutions, but also explain how their solutions relate to ongoing research in the area. The instructor will provide project topics, but it is preferred that students develop projects directly related to their own personal research interests. Deliverables include an oral presentation to the class and a written document formatted as a conference paper. Additional details, requirements, and advice will be provided in lecture.

Midterm Exam
The midterm exam will be given approximately half way through the quarter and will cover all material discussed in class up to the day of the exam. The exam will be designed to test breadth of knowledge on techniques and concepts. The exam will be closed book, however, students will be permitted a single one-sided page of handwritten notes. The notes must be original copies (no photocopies of a common set of notes) and unique to each student in the exam.

Final Exam
The final exam will be similar to the midterm, except that it will cover all material covered in the course and students will be permitted TWO one-sided sheets of handwritten notes.
Academic Integrity
It is the policy of Wright State University to uphold and support standards of personal honesty and integrity for all students. The formal university code of student academic conduct can be viewed at:

http://www.wright.edu/students/judicial/integrity.html

PLEASE BE FAMILIAR WITH THE ACADEMIC CODE OF CONDUCT. If the instructor detects infractions, he will follow the procedures outlined in the formal university policy. These policies are easily available for review. Therefore, ignorance of the law is no defense.

Additional Information

Absences
Class attendance will not be a direct factor in student grades, but will strongly affect the quality of one's class experience. Students are expected to attend every class, as things may make less sense to students that do not attend classes or who arrive late. Students registering after the term begins are responsible for all missed material and should not expect that due dates will be altered.

Office Hours
Office hours are not private lectures. It is expected that students attend and participate in lectures and use office hours for additional discussion of issues related to class topics. Related topics include clarification of lecture points, remediation advice, or expansion beyond textbook materials. You will get the most out of office hour visits by preparing specific questions and/or examples ahead of your visit. Make an attempt to solve problems on your own before coming to office hours even if you don't expect to solve the problem correctly. The instructor may be able to help diagnose problems in understanding or execution – but only if there are concrete examples of your work to examine.

Class Lecture Materials
Copies of the slides used in lecture, supplementary textbooks, and additional course-related information will be available dropbox links that will be emailed to students as needed. Students can retrieve documents using any standard web browser.

Additional Needs
Students with disabilities or any additional needs are encouraged to make an appointment with the instructor to discuss any accommodations that may be necessary. We enthusiastically work with the WSU Office of Disability Services (http://www.wright.edu/students/dis_services/) and strongly encourage you to consult with them as well.
<table>
<thead>
<tr>
<th>Week</th>
<th>Course Topics</th>
<th>Readings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>What is evolutionary computation, its applications, and its strengths?</td>
<td>Chapters 1 &amp; 2</td>
</tr>
<tr>
<td>2</td>
<td>Genetic Algorithms</td>
<td>Chapter 3</td>
</tr>
<tr>
<td>3</td>
<td>Evolutionary Strategies</td>
<td>Chapter 4</td>
</tr>
<tr>
<td></td>
<td>Evolutionary Programming</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Tuning Evolutionary Algorithms</td>
<td>Chapter 5</td>
</tr>
<tr>
<td>5</td>
<td>Genetic Programming and Variants</td>
<td>Chapter 8</td>
</tr>
<tr>
<td>6</td>
<td>Learning Classifier Systems</td>
<td>Chapter 6</td>
</tr>
<tr>
<td>7</td>
<td>Particle Swarm Algorithms</td>
<td>Chapter 7</td>
</tr>
<tr>
<td></td>
<td>Ant Colony Algorithms</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Multimodal Problems</td>
<td>Chapter 9 &amp; 10</td>
</tr>
<tr>
<td></td>
<td>Memetic Algorithms</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Theoretical Aspect of Evolutionary Algorithms</td>
<td>Chapter 11</td>
</tr>
<tr>
<td>10</td>
<td>Constraint Handling in Evolutionary Algorithms</td>
<td>Chapter 12</td>
</tr>
</tbody>
</table>