

Introducción a la Computación Evolutiva

Dr. Carlos Artemio Coello Coello
CINVESTAV-IPN
Departamento de Computación
Av. Instituto Politécnico Nacional No. 2508
Col. San Pedro Zacatenco
México, D.F. 07360
e-mail: ccoello@cs.cinvestav.mx
Horario de clases : Miércoles y Viernes de 12:00 a 14:00 hrs

Enfoque del curso

En este curso se estudiarán los conceptos básicos de las técnicas más importantes de computación evolutiva, haciendo especial énfasis en los algoritmos genéticos. Inicialmente, se hará un recorrido histórico en el que se resumirán los logros más importantes en torno a la simulación de los procesos evolutivos como una herramienta para el aprendizaje y la optimización. Posteriormente, se analizarán y compararán de manera general los 3 paradigmas principales que se utilizan hoy en día en la computación evolutiva: las estrategias evolutivas, la programación evolutiva y los algoritmos genéticos. En cada caso se abordará su inspiración biológica, su motivación, su funcionamiento y algunas de sus aplicaciones. Finalmente, se estudiará a mayor detalle el funcionamiento, fundamentos teóricos, implementación y operación de los algoritmos genéticos, que es actualmente el paradigma evolutivo más utilizado por los investigadores que trabajan en esta disciplina.

Forma de evaluación

Para calcular la calificación final del curso, se considerarán los siguientes porcentajes:

Tareas	30%
Primer Examen Parcial	20%
Segundo Examen Parcial	20%
Proyecto Final	30%

Para aprobar el curso, deberá obtenerse un mínimo del 70% del total de puntos en juego durante el cuatrimestre (sin incluir los de bonificación).

Temario

- **Técnicas heurísticas** (1.5 hrs)
 - Problemas P y NP
 - Técnicas clásicas de búsqueda y optimización
 - Lo que el mundo real demanda
 - ¿Qué es una heurística?
 - ¿Realmente necesitamos técnicas heurísticas?
 - Ejemplos de técnicas heurísticas
 - * Búsqueda tabú
 - * Recocido simulado
 - * Escalando la colina

- **Nociones de optimización** (1.5 hrs)
 - Optimización Global
 - Optimización Numérica
 - Optimización Combinatoria
 - Espacios de búsqueda convexos y cóncavos
 - Restricciones explícitas e implícitas
 - Restricciones de igualdad y desigualdad
 - Zona factible y no factible

- **Antecedentes históricos** (1 hr)
 - Inspiración biológica
 - Primeros intentos
 - Cronología de descubrimientos importantes

- **Paradigmas principales** (1 hr)
 - Estrategias evolutivas
 - Programación evolutiva
 - Algoritmo genético
 - Comparaciones

- **La computación evolutiva en el contexto de la Inteligencia Artificial** (1 hr)
 - Críticas (IA Clásica vs. Técnicas Heurísticas)
 - ¿Un nuevo paradigma?

- **Terminología biológica vs. terminología usada en la computación evolutiva**
(1 hr)
 - Glosario Básico
 - Comparaciones
- **Algoritmos Genéticos**
 - Generalidades (1 hr)
 - * Definición
 - * Componentes básicos
 - * Funcionamiento
 - Representación (3 hrs)
 - * Binaria
 - * Códigos de Gray
 - * Real
 - * Programación Genética
 - * Algoritmos genéticos desordenados (*Messy GAs*)
 - * Otras propuestas
 - Selección (2 hrs)
 - * Proporcional
 - * Torneo
 - * Estado Uniforme
 - * Uso de jerarquías
 - Cruza (3 hrs)
 - * Importancia
 - * Un punto
 - * Dos puntos
 - * Uniforme
 - * Casos especiales
 - Mutación (2 hrs)
 - * Importancia
 - * Forma básica
 - * No uniforme
 - * Casos especiales
 - Función de aptitud (3 hrs)
 - * Definición
 - * Uso de “cajas negras”
 - * Manejo de restricciones usando funciones de penalización

- * Ejemplos
- Ajuste de parámetros (1 hr)
 - * Estudios empíricos
 - * Auto-adaptación
- Implementación (3 hrs)
 - * Software propio
 - * Software de dominio público
 - * Sistemas comerciales
- Operadores avanzados (2 hrs)
 - * Diploides y dominancia
 - * Inversión
 - * Micro-operadores
 - Segregación
 - Traslocación
 - Duplicación
 - Borrado
- Teoría (4 hrs)
 - * Teorema de los esquemas
 - * Modelos exactos
 - * Teoría de convergencia
 - * *No Free Lunch Theorems*
 - * Críticas
 - * Decepción
- ¿Cuándo aplicar un algoritmo genético? (1 hr)
 - * Limitaciones
 - * Ventajas
 - * Áreas de aplicación
- Áreas abiertas de investigación (1 hr)
 - * Inspiración biológica
 - * Paralelismo
 - * Teoría
 - * Representación
 - * Operadores
 - * Co-evolución
 - * Algoritmos culturales
 - * Ajuste de parámetros
 - * Auto-adaptación

- * Otras
- Temas avanzados (opcionales) (4 hrs)
 - * Funciones multimodales
 - Nichos
 - Repartición de aptitud
 - Ejemplos
 - * Funciones con objetivos múltiples
 - Técnicas básicas
 - * Manejo de restricciones
 - Función de penalización
 - Pena de muerte
 - Separación de objetivos y restricciones
 - Uso de representaciones y operadores especiales
 - Otras propuestas

Libro de Texto

A.E. Eiben and J.E. Smith, *Introduction to Evolutionary Computing*, Springer, Berlin, Second Edition, 2015, ISBN 978-3-662-44873-1.

Textos Complementarios

- David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
- David Corne, Marco Dorigo & Fred Glover (editores), *New Ideas in Optimization*, McGraw-Hill, London, 1999.
- Sadiq M. Sait & Habib Youssef, *Iterative Computer Algorithms with Applications in Engineering*, IEEE Computer Society, Los Alamitos, California, 1999.
- Melanie Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, Massachusetts, 1996.
- Zbigniew Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Second Edition, 1992.
- John H. Holland, *Adaptation in Natural and Artificial Systems. An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*, MIT Press, Cambridge, Massachusetts, Second Edition, 1992.
- David B. Fogel, *Evolutionary Computation. Toward a New Philosophy of Machine Intelligence*, The Institute of Electrical and Electronic Engineers, New York, 1995.

- Thomas Bäck, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, New York, 1996.
- David B. Fogel, *Evolutionary Computation: The Fossil Record*, The Institute of Electrical and Electronic Engineers, New York, 1998.
- John R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, Massachusetts, 1992.
- Colin B. Reeves (editor), *Modern Heuristic Techniques for Combinatorial Problems*, John Wiley & Sons, Great Britain, 1993.
- Zbigniew Michalewicz & David B. Fogel, *How to Solve It: Modern Heuristics*, Springer, Berlin, 2000.

Material para el curso

Un conjunto de lecturas adicionales para el curso se encuentran disponibles en:
<http://delta.cs.cinvestav.mx/~ccoello/bibgen.html>

Los artículos disponibles en formato electrónico están en postscript, y los demás se encuentran disponibles en forma de fotocopias.

Asimismo, en la página

<http://delta.cs.cinvestav.mx/~ccoello/genetic.html>

hay una serie de direcciones del Web que contienen material de utilidad para el curso. En esta misma página se irán colocando también los acetatos utilizados por el instructor en clase (en formato postscript), y se encuentran disponibles también unas notas de curso elaboradas por el instructor que aunque no corresponden exactamente al temario antes estipulado, pueden ser de ayuda.

Proyecto final

Consistirá en la aplicación de un algoritmo genético a un problema seleccionado por los estudiantes. El reporte de este proyecto deberá incluir información suficiente como para permitir la reproducción de los resultados obtenidos, y se mostrarán gráficas de convergencia, corridas de ejemplo, el código fuente y el análisis estadístico de los datos, así como la bibliografía consultada. Conforme avance el cuatrimestre se irán dando más detalles al respecto.

Avisos importantes

Todas las tareas entregadas después de la fecha acordada, serán PENALIZADAS con un 10% MENOS SOBRE LA CALIFICACIÓN OBTENIDA POR CADA PERIODO DE 24 HORAS POSTERIOR A LA HORA DE ENTREGA, a menos que otras penalizaciones se estipulen específicamente para una cierta tarea (por ejemplo, que no se

admitirá si se entrega después de la fecha indicada). La hora de entrega de todas las tareas será al inicio de la clase, a menos que se estipule lo contrario.

NO SE ADMITIRÁN TAREAS ENTREGADAS POR PARTES. Quien entregue sólo una parte de una tarea, no podrá entregar posteriormente partes complementarias. Sólo se le calificará lo que se haya entregado la primera vez.

Toda tarea entregada después de haber transcurrido una semana o más a partir de la fecha de entrega, se considerará como **NO ENTREGADA**. Quien tenga una o más tareas no entregadas, no tendrá derecho a presentar ninguno de los exámenes ni a la entrega del proyecto final.

Una vez que una tarea haya alcanzado un valor de cero, la penalización se detendrá (es decir, no se considerarán calificaciones negativas), sin embargo, **LA ENTREGA DE TODAS LAS TAREAS Y DEL PROYECTO FINAL ES OBLIGATORIA PARA PODER APROBAR EL CURSO**, aunque el puntaje asignado a cualquiera de ellas (incluyendo el proyecto final) no afectara de manera significativa la calificación final de un alumno.

Las tareas y los exámenes contendrán preguntas opcionales de bonificación que, de ser contestadas correctamente, permitirán recuperar puntos perdidos a lo largo del cuatrimestre. Esto implica que será posible obtener más del 100% de los puntos en una tarea o examen, de contestarse todas las preguntas correctamente, incluyendo las de bonificación. **Las tareas entregadas tardíamente NO recibirán puntos de bonificación.**

El instructor considera como una ofensa muy seria el plagio de tareas de otros compañeros, y en caso de comprobarse éste, se **REPROBARÁ DE FORMA AUTOMÁTICA A LOS INVOLUCRADOS**.

Es necesario que todos los alumnos proporcionen una dirección electrónica en la lista que se circulará en la clase, y que **LEAN FRECUENTEMENTE SU CORREO ELECTRÓNICO**, pues muchas de las dudas y avisos importantes se transmitirán por este medio. De tal forma, es total responsabilidad de los alumnos enterarse oportunamente de lo que se transmita por correo electrónico.