

Nombre y código de la asignatura		Optimización y Métodos Computacionales - DE174					
Área académica		Fundamentación					
Semestre	Créditos	Requisitos	Horas presenciales (HP)			Horas de trabajo independiente	Total de horas
			Teóricas	Prácticas	HP Totales		
1	4	Ninguno	2	1	3	9	12

Año de actualización de la asignatura: 2020

1. Breve descripción

La optimización combinatoria es una de las áreas más jóvenes y activas de las matemáticas discretas. Su considerable crecimiento se debe probablemente a su gran campo de aplicación, dado que muchos problemas de la vida real pueden ser formulados como problemas abstractos de optimización combinatorial. En este curso se describen algunas de las técnicas de optimización combinatorial más reconocidas en la literatura y utilizadas en aplicaciones prácticas. Además, incluye aspectos básicos de la teoría de grafos y se hace énfasis en el estudio y descripción de algunos problemas combinatoriales típicos en diferentes campos de la ingeniería. Dentro del contenido del curso se proponen algoritmos en pseudo-código que pueden ser fácilmente implementados en cualquier lenguaje de programación. Estos algoritmos no están concebidos para alcanzar una alta eficiencia computacional, la idea es que sean de fácil implementación. La optimización computacional de estos pseudo-códigos es un paso que el estudiante puede dar en un nivel posterior, más avanzado. En cada sección se presentan ejemplos aplicados a casos y problemas reales. Además, se plantean ejercicios al final de cada capítulo, diseñados para interiorizar los conceptos más relevantes. En el primer capítulo se presentan algunos conceptos básicos que permiten homogenizar el lenguaje utilizado durante todo el texto guía. El capítulo dos explica algunos problemas combinatoriales que se han clasificado en cuatro grupos: problemas de empaquetamiento, problemas formulados a través de grafos, problemas de asignación y problemas de secuenciación. El capítulo tres presenta las características generales de los algoritmos evolutivos, haciendo énfasis en los algoritmos genéticos. Después se utilizan estos conceptos para introducir algunas bases sobre optimización multiobjetivo. El capítulo cuatro aborda el método de optimización usando colonia de hormigas. El capítulo cinco presenta el método de recocido simulado, y el capítulo seis estudia el método de búsqueda tabú. Todos los métodos son ejemplarizados con problemas típicos.

2. Objetivo general

Analizar, modelar, adaptar e implementar las diferentes técnicas de optimización combinatoria, bio-inspiradas y computacionales, usando problemas típicos y reales de asignación, programación, ruteo y secuenciación, relevantes en el sector industrial.

3. Resultados de aprendizaje de asignatura

El estudiante:

1. Analiza problemas combinatorios reales, estructurando un modelo computacional e identificando parámetros y variables del problema abordado.
2. Modela e interpreta el problema combinatorio, proponiendo estrategias adecuadas de codificación del problema, búsqueda local, exploración y diversidad.
3. Adapta diferentes técnicas existentes y estrategias básicas de exploración del espacio de solución combinatorio a la solución de diferentes problemas abordados, de forma que se asegura la convergencia del método a soluciones óptimas de calidad.
4. Implementa los algoritmos en un entorno de programación estructurada a través de prácticas virtuales.
5. Elabora proyectos relacionados con un problema real de interés industrial o académico, aplicando una técnica de optimización combinatoria.

4. Contenido

1. CONCEPTOS BÁSICOS^[1]. (~ 3 h)

La solución de problemas combinatoriales a través de metaheurísticas se presenta como una alternativa para obtener soluciones aproximadas de buena calidad, en tiempos computacionales razonables. En esta sección se abordan definiciones básicas como: problemas del mundo real, complejidad computacional, solución de problemas NP-difícil, qué es una metaheurística, aspectos básicos de teoría de grafos, óptimo local y global y optimización mono y multi-objetivo.

2. PROBLEMAS Y CODIFICACIÓN^[1]. (~3 h)

Cuando se aborda un problema combinatorial, uno de los aspectos más importantes a tener en cuenta para su solución es cómo representar las variables del problema para que puedan ser interpretadas por un algoritmo computacional. Esta sección aborda temas tales como: problemas de empaquetamiento (empaquetamiento o corte rectangular 2D y 3D, problema de la mochila), problemas representados como grafos, problema del agente viajero, coloreado de grafos, problemas de asignación (asignación generalizada, asignación de horarios de clase), problemas de secuenciamiento (secuenciamiento regular: flow-shop, secuenciamiento no regular: job-shop), algunas formas de codificación, Ejercicios de aplicación e implementación virtual.

3. ALGORITMOS EVOLUTIVOS^[1]. (~12 h)

Los Algoritmos Evolutivos (AEs) imitan los principios evolutivos naturales para constituir procedimientos de búsqueda y optimización. Parten de un hecho, observado en la naturaleza, basado en que los organismos vivos poseen una destreza consumada en la resolución de los problemas que se les presentan y obtienen sus habilidades, casi sin proponérselo, a través del mecanismo de la evolución natural. En esta sección se abordan: algoritmos genéticos (función de adaptación, operadores de selección, operadores de variación, operadores de preservación de elitismo), optimización Multi-objetivo, (optimización, métodos para soluciones no dominadas, distancia de apilamiento, selección por torneo usando apilamiento, método NSGA-II, MOEA para manejo de restricciones, algoritmo genético de Chu and Beasley, Ejercicios de aplicación e implementación virtual.

4. OPTIMIZACIÓN POR COLONIA DE HORMIGAS^[1]. (~6 h)

Las colonias de hormigas, y en general el comportamiento de insectos sociales, son sistemas distribuidos que, a pesar de la simplicidad de sus individuos, presentan una gran organización social estructurada. Como resultado de esta organización, las colonias de hormigas pueden realizar tareas complejas que en muchos casos superan la capacidad individual de una única hormiga. La idea principal consiste en utilizar los principios auto-organizativos que permiten el comportamiento altamente coordinado de las hormigas reales para coordinar poblaciones de agentes artificiales que pueden resolver problemas computacionales. En esta sección se abordan: experimento de los dos caminos, algoritmo simple OCH, problema de la ruta de mínimo costo algoritmo OCH para el PAV, Algoritmo OCH para el MDVRPB, Ejercicios de aplicación e implementación virtual.

5. RECOCIDO SIMULADO^[1]. (~6 h)

El algoritmo de SA puede interpretarse como iteraciones sucesivas del algoritmo de Metrópolis en las que se evalúa y decreta de forma controlada la temperatura o parámetro de control. El algoritmo de Metrópolis simula el proceso físico basado en temperatura en el que los sólidos obtienen estados de baja energía o equilibrio térmico, este proceso es conocido como recocido de sólidos. Durante este proceso, la energía libre del sólido es minimizada. Esta sección aborda la implementación práctica del algoritmo SA.

6. BÚSQUEDA TABÚ^[1]. (~6 h)

La Búsqueda Tabú (Tabu Search - TS) es un procedimiento metaheurístico cuya característica distintiva es el uso de memoria adaptativa y de estrategias especiales de resolución de problemas. Puede verse como una metaheurística que se superpone a una técnica de búsqueda y que se encarga de evitar que dicha técnica caiga en óptimos locales prohibiendo (o, en un sentido más general, penalizando) ciertos movimientos. Esta sección aborda la implementación práctica del algoritmo BT.

7. IMPLEMENTACIÓN COMPUTACIONAL^[1]. (~12 h)

Implementación de todos los pseudocódigos en la plataforma de desarrollo *Matlab (c)*.

5. Recursos y bibliografía

Recursos:

Para el desarrollo de la asignatura se cuenta con textos guía “Optimización combinatoria de la teoría a la práctica”. Como herramientas informáticas, se recomienda el Matlab para el desarrollo de los algoritmos.

Interacción con software comercial (Matlab, Concorde, entre otros). Plataforma y herramientas de Google para clases virtuales.

Bibliografía:

1. Granada M., Santa J. Optimización combinatoria. De la teoría a la práctica Colombia, 2018, ISBN: 9789588859439. Ed. Universidad Libre De Colombia, 2018.
2. Granada M., Moreno, C., Falcón, L., Escobar, J., Zuluaga, A. (2019). Heuristic constructive algorithm for work-shift scheduling in bus rapid transit systems. *Decision Science Letters*, 8(4), 519-530.
3. Granada, M., Toro, E. M., & Gallego, R. (2019). An MIP formulation for the open location- routing problem considering the topological characteristic of the solution- paths. *Networks*, 74(4), 374-388.
4. Granada M., Correa-Tamayo, J. S., Arias-Londoño, A. (2019). Optimal management of vegetation maintenance and the associated costs of its implementation in overhead power distribution systems. *TecnoLógicas*, 22(45), 93-109.
5. Granada M., Cubides, L. C., Arias Londoño, A. (2019). Problema de ruteo de vehículos eléctricos con recogidas considerando la ubicación de estaciones de recarga y la operación del sistema de distribución de energía.
6. Granada M., Montoya, O. D., Hincapie, R. A. (2018, October). Optimal Location of Protective Devices Using Multi-objective Approach. In *Workshop on Engineering Applications* (pp. 3-15). Springer, Cham.
7. Granada M., Bolanos, R., Escobar, J. (2018). A metaheuristic algorithm for the multi-depot vehicle routing problem with heterogeneous fleet. *International Journal of Industrial Engineering Computations*, 9(4), 461-478.
6. Granada M., Chávez, J., Escobar, J., & Meneses, C. (2018). A heuristic algorithm based on tabu search for vehicle routing problems with backhauls. *Decision Science Letters*, 7(2), 171-180.
7. Granada M., Escobar-Falcón, L. M., Álvarez-Martínez, D., Escobar, J. W., & Romero-Lázaro, R. A. (2016). A matheuristic algorithm for the three-dimensional loading capacitated vehicle routing problem (3L-CVRP). *Revista Facultad de Ingeniería Universidad de Antioquia*, (78), 09-20.
8. Granada M., Santa Chávez, J. J., Escobar, J. W., & Meneses, C. A. P. (2015). A metaheuristic ACO to solve the multi-depot vehicle routing problem with backhauls. *International Journal of Industrial Engineering and Management (IJIEM)*, 6(2), 49-58.

6. Metodología

Trabajo en grupos, exposiciones magistrales por parte del profesor, usando técnicas de aprendizaje activo, discusión de casos reales, talleres y presentaciones por parte de los estudiantes. Utilización de herramientas virtuales, tales como: Classroom, JamBoard, Meet, Zoom. Gestión de recepción y entrega de trabajos por Classroom. Se comparte el video de cada clase con los estudiantes. Utilización de material y exposiciones prediseñados.

7. Evaluación

- Primer parcial antes de la semana 6: Conceptos básicos (T1) y problemas y codificación.
- Segundo parcial antes de la semana 12: Algoritmos evolutivos y optimización por colonia de hormigas.
- Examen final - semana 16: recocido simulado, búsqueda tabú y conceptos generales.
- Nota de talleres. Todos los ítems con un porcentaje de 25%.