



## PROGRAMA DE ASIGNATURA

<b>Nombre del curso</b>	<b>TÓPICOS AVANZADOS EN OPTIMIZACIÓN</b>								
<b>Programa</b>	Magíster en Ciencias de la Ingeniería, mención Ingeniería Eléctrica								
<b>Código</b>	Por definir	<b>Nivel</b>	1 o 2	<b>Tipo</b>	Electiva	<b>T-E-L</b>	4-0-2	<b>Créditos SCT-Chile</b>	5
<b>Descripción del curso</b>	Este curso ofrece una introducción general a la modelación matemática en programación lineal, no lineal y problemas combinatorios. Los estudiantes aprenderán conceptos clave de optimización determinista y estocástica, tanto en problemas continuos como combinatoriales. Además, se explorarán aplicaciones en ingeniería y diversas técnicas de resolución, incluyendo algoritmos exactos y aproximados. El curso también enfatiza la implementación computacional de modelos y algoritmos mediante software especializado, permitiendo evaluar su efectividad. Se fomentará la exploración de nuevos enfoques algorítmicos y su aplicación en temas de tesis.								
<b>Objetivos</b>	<ul style="list-style-type: none"><li>● Comprender conceptos básicos y avanzados sobre técnicas modernas de optimización lineal y no-lineal para problemas de naturaleza continua, convexa y combinatorial en su forma determinista y estocástica en el ámbito de la ingeniería.</li><li>● Modelar matemáticamente nuevas aplicaciones usando técnicas modernas de optimización determinista y estocástica.</li><li>● Explorar nuevas aplicaciones usando los modelos, métodos y algoritmos estudiados relacionados con los temas de tesis de cada estudiante en particular.</li><li>● Realizar implementaciones computacionales que permitan medir la efectividad de los algoritmos propuestos usando software de optimización especializado.</li><li>● Realizar un aporte personal científico y/o de aplicación práctica mediante la implementación de técnicas de optimización.</li></ul>								
<b>Contenidos (Unidades y Lista de Contenidos temáticos)</b>	<p><b>Unidad 1: Introducción a la programación matemática lineal, no lineal, convexa y combinatoria.</b></p> <ul style="list-style-type: none"><li>● Programación Lineal (PL)<ul style="list-style-type: none"><li>○ Definición y formulación de problemas de PL.</li><li>○ Teoremas fundamentales: existencia de soluciones óptimas, dualidad y complementariedad.</li><li>○ Método Simplex.</li><li>○ Resolución con software (CPLEX, GLPK, Matlab, Gurobi).</li></ul></li><li>● Programación No Lineal (PNL)<ul style="list-style-type: none"><li>○ Diferencias entre programación lineal y no lineal.</li><li>○ Funciones objetivo y restricciones no lineales.</li><li>○ Condiciones de optimalidad de Karush-Kuhn-Tucker (KKT).</li><li>○ Métodos de solución:<ul style="list-style-type: none"><li>- Métodos de gradiente.</li><li>- Métodos de Newton y cuasi-Newton.</li></ul></li></ul></li><li>● Optimización Convexa<ul style="list-style-type: none"><li>○ Definición y propiedades de conjuntos y funciones convexas.</li><li>○ Programación convexa vs. no convexa.</li><li>○ Condiciones de optimalidad en problemas convexas.</li></ul></li><li>● Optimización Combinatorial<ul style="list-style-type: none"><li>○ Definición de problemas combinatoriales y su complejidad computacional.</li><li>○ Ejemplos clásicos:<ul style="list-style-type: none"><li>- Problema del Viajante de Comercio (TSP).</li><li>- Problema de Asignación.</li><li>- Problema de la Mochila.</li></ul></li></ul></li></ul>								



## PROGRAMA DE ASIGNATURA

- Métodos de solución:
  - Programación entera y mixta-entera.
  - Métodos exactos: Ramificación y Acotamiento (Branch and Bound), Benders.

### **Unidad 2: Modelos de optimización basados en optimización robusta y estocástica.**

- Introducción a la Optimización Bajo Incertidumbre
- Diferencia entre optimización determinista, robusta y estocástica.
- Fuentes de incertidumbre en problemas de optimización.
- Aplicaciones en ingeniería, logística, redes.
- Optimización Robusta (RO - Robust Optimization)
  - Definición y fundamentos de optimización robusta.
  - Conjuntos de incertidumbre (poliedrales, elipsoidales, intersección de conjuntos).
- Formulación de modelos robustos:
  - Programación Lineal Robusta (RLP).
  - Programación Entera Robusta (RIP).
  - Programación No Lineal Robusta (RNLP).
- Introducción a la modelación probabilística en optimización.
- Programación Estocástica con Restricciones de Probabilidad (Chance-Constrained Programming).
- Aplicaciones en asignación de recursos, diseño de redes y logística.

### **Unidad 3: Algoritmos exactos, heurísticos y de aproximación.**

- Algoritmos Exactos
  - Branch and Bound: Técnica basada en la exploración de un árbol de soluciones con poda de ramas no prometedoras.
  - Branch and Cut: Extensión de Branch and Bound que incorpora cortes para mejorar la eficiencia en problemas de Programación Entera.
- Algoritmos Heurísticos aproximados
  - Los heurísticos buscan soluciones buenas en tiempos razonables, sin garantizar optimalidad. Son útiles en problemas grandes o difíciles.
  - Métodos Greedy (Voraces): Construcción de soluciones paso a paso, eligiendo la mejor opción localmente.
  - Búsqueda Local: Mejoras iterativas a una solución inicial (Ejemplo: Hill Climbing, Simulated Annealing).
  - Tabu Search: Extensión de la búsqueda local que evita ciclos mediante una memoria de soluciones visitadas.
  - Algoritmos Genéticos: Inspirados en la evolución natural, combinan soluciones para mejorar resultados.

### **Unidad 4: Implementación de modelos matemáticos en software especializados.**

- Introducción a la Implementación de Modelos Matemáticos
  - Importancia de la implementación computacional en optimización.
  - Diferencias entre formulación teórica y modelado computacional.
  - Elección del software adecuado según el tipo de problema.
- Herramientas de Modelado y Software de Optimización
  - Lenguajes de modelado: AMPL, GAMS, Pyomo, JuMP.
  - Solvers comerciales: Gurobi, CPLEX, MOSEK, FICO Xpress.



## PROGRAMA DE ASIGNATURA

	<ul style="list-style-type: none"><li>○ Solvers de código abierto: CBC, GLPK, SCIP.</li><li>○ Otros entornos: MATLAB (Optimization Toolbox), R (lpSolve, ompr), Python (PuLP, OR-Tools).</li><li>● Algoritmos Exactos y Aproximados en la Práctica<ul style="list-style-type: none"><li>○ Implementación de algoritmos de Branch &amp; Bound, Branch &amp; Cut</li><li>○ Métodos heurísticos y metaheurísticos (ej. algoritmos genéticos, búsqueda tabú, recocido simulado).</li></ul></li></ul> <p><b>Unidad 5: Aplicación de conceptos teóricos y prácticos a problemáticas específicas relacionados con la tesis de los estudiantes.</b></p> <ul style="list-style-type: none"><li>● Modelación Matemática aplicada a problemas de tesis<ul style="list-style-type: none"><li>○ Formulación de modelos matemáticos específicos (PL, PNL, enteros, estocásticos, etc.).</li><li>○ Elección de restricciones y funciones objetivo-relevantes.</li></ul></li><li>● Métodos exactos y heurísticos en problemas específicos<ul style="list-style-type: none"><li>○ Uso de heurísticas y metaheurísticas para problemas complejos (GRASP, Algoritmos Genéticos, Búsqueda Tabú).</li><li>○ Análisis de enfoques exactos y aproximados en los problemas de tesis.</li></ul></li></ul>
<b>Resultados de aprendizajes esperados</b>	<p><b>Resultado de Aprendizaje General:</b> Implementar modelos y algoritmos de optimización mediante software especializado, evaluando su efectividad en aplicaciones de ingeniería, para explorar enfoques algorítmicos innovadores en el contexto de sus investigaciones y trabajos de tesis.</p> <p><b>Resultados de Aprendizaje Específicos:</b></p> <ol style="list-style-type: none"><li>1. Construir modelos matemáticos de programación matemática lineal, no lineal, convexa y combinatoria formulando adecuadamente los problemas y utilizando las técnicas apropiadas para representar situaciones reales en contextos de optimización.</li><li>2. Formular modelos de optimización basados en optimización robusta y estocástica para la construcción de modelos que garanticen soluciones eficientes y viables en condiciones de incertidumbre.</li><li>3. Aplicar distintos algoritmos exactos, heurísticos y de aproximación utilizando las estrategias más adecuadas según las características del problema de optimización, para obtener soluciones precisas, eficientes o cercanas a la óptima en tiempo razonable.</li><li>4. Implementar modelos matemáticos y algoritmos en software especializados (Matlab, Python, GAMS, etc) desarrollando programas eficientes para la resolución de problemas de optimización, evaluando su efectividad y capacidad de adaptación a diferentes tipos de modelos y condiciones de incertidumbre.</li><li>5. Aplicar conocimientos adquiridos a sus problemas de investigación relacionados con la tesis a desarrollar formulando y resolviendo modelos de optimización adecuados, utilizando algoritmos y software especializado, y proponiendo soluciones innovadoras a desafíos específicos en su área de estudio.</li></ol>
<b>Modalidad de evaluación</b>	<p><b>Estrategias Metodológicas</b> El aprendizaje se lleva a cabo mediante una metodología activa que integra teoría y práctica a través de simulaciones y ejercicios interactivos. Se emplean herramientas digitales para modelar escenarios reales y analizar distintas estrategias de solución. Durante las sesiones, los estudiantes participan en experimentos controlados y análisis de datos con software especializado, lo que facilita la comprensión y aplicación de los conceptos clave del curso.</p>



## PROGRAMA DE ASIGNATURA

	<p><b>Evaluación</b></p> <p>Las evaluaciones se realizarán por medio de:</p> <ul style="list-style-type: none"><li>● 2 trabajos de investigación –documento escrito, código fuente de la implementación y presentación- (corresponde al 50 % de la evaluación cada uno)</li><li>● Trabajo adicional en caso de no cumplir con la expectativa de algún trabajo en modalidad POR.</li></ul>
<b>Bibliografía</b>	<p><b>Básica:</b></p> <ol style="list-style-type: none"><li>1. Hillier, F. &amp; Lieberman, G. (2015). <i>Introduction to Operations Research</i>. Versión en Español: <i>Introducción a la Investigación de Operaciones</i>. Mc Graw Hill. 9th edition</li><li>2. Luenberger, D. G. &amp; Ye, Y. (2015). <i>Introduction to Linear and Nonlinear Programming</i>. Addison Wesley. Versión en Español. <i>“Programación Lineal y No-Lineal”</i>. Addison Wesley. 4th Edition.</li><li>3. Papadimitriou, C. H. (2003). <i>Computational complexity</i>. John Wiley and Sons Ltd.</li><li>4. Glover, F. &amp; Kochenberger, G. (2003). <i>Handbook of Metaheuristics</i>. Kluwers International Series, New York.</li><li>5. Papadimitriou, C. H. &amp; Steiglitz, K. (1998). <i>Combinatorial optimization: algorithms and complexity</i>. Courier Corporation.</li></ol> <p><b>Recomendada:</b></p> <p>Literatura reciente consultada en bases de datos: IEEE Xplore, Springer, Science Direct, Hindawi, MDPI, etc.</p>