

Advanced Metaheuristics

Algorithmic Architectures, Local Search Strategies and Complex Systems

Giorgio Saibene

Level: Master of Science / PhD Track

Course Overview

This advanced graduate curriculum examines the design, implementation, and convergence analysis of modern metaheuristic paradigms for hard combinatorial optimization problems. Students will master the core algorithmic structures and trade-offs between exploration and exploitation required to engineer robust solution methods for complex industrial and logistical systems.

Core Methodological Modules

1. Trajectory-Based Local Search

Advanced mechanics of neighborhood structures and local optimality. Algorithmic deployment of Simulated Annealing, Tabu Search, and Variable Neighborhood Search (VNS). Implementation of short-term and long-term memory structures to escape localized extrema.

2. Population-Based Evolutionary Architectures

Design of trajectory-free metaheuristics. Genetic Algorithms (GA), Memetic Algorithms, and Particle Swarm Optimization (PSO). Coding mechanisms for selection operators, crossover variants, and mutation operators balanced against population diversity metrics.

3. Intensification vs. Diversification Logic

Theoretical and empirical analysis of optimization balance. Parameter tuning frameworks and self-adaptive control mechanisms. Analytical tracking of search space coverage and state-space convergence behavior in highly constrained complex systems.

4. Hybridization Frameworks & Large Neighborhoods

Introduction to Multi-Start methods, GRASP architectures, and Iterated Local Search (ILS). Foundations of Large Neighborhood Search (LNS) and Adaptive Large Neighborhood Search (ALNS) utilizing specialized ruin-and-recreate sub-algorithms.

5. Computational Complexity & Performance Evaluation

Worst-case and average-case empirical analysis of metaheuristic performance. Statistical benchmarking via standardized instance libraries (TSPLIB, VRPLIB). Assessment of solution robustness, scalability constraints, and parallel computing implementation parameters.