Summer 2025

Advanced Optimization

Md Asif Khan, PhD

Course Description

This graduate-level course provides an in-depth understanding of advanced optimization techniques and their application to engineering problems.

The course begins with classical methods (e.g., Newton, quasi-Newton, and simplex) and progresses toward modern optimization methods such as genetic algorithms, swarm intelligence, and differential evolution. Topics also include constrained optimization, convex programming, semi-definite programming, and optimization software tools.

Learning Outcomes

By the end of the course, students will be able to:

- Formulate and classify engineering optimization problems.
- Analyze and apply classical and modern optimization methods.
- Evaluate algorithm performance on benchmark and real-world problems.
- Explore optimization software packages (e.g., MATLAB, SciPy, CVXPY).
- Present and critique optimization techniques or case studies.
- Apply optimization techniques in a final research project or technical application.

Reading Materials

1. ***Metaheuristics: From Design to Implementation by*** *El-Ghazali Talbi**,** John Wiley & Sons, 2009. ISBN: 978-0-470-27858-1*
2. **"Numerical Optimization"** by *Jorge Nocedal and Stephen J. Wright*
3. **"Engineering Optimization: Theory and Practice"** by *S. S. Rao*
4. ***Multi-Objective Optimization Using Evolutionary Algorithms by*** *Kalyanmoy Deb, Volume 16, John Wiley & Sons, 2001.*
5. *Selected IEEE/Elsevier conference/journal articles*

Reading Materials

1. "**Essentials of Metaheuristics**" by Sean Luke

Free PDF: https://cs.gmu.edu/~sean/book/metaheuristics/

Topics: Genetic Algorithms, PSO, Simulated Annealing, Evolution Strategies, Ant Colony.

2. "**Nature-Inspired Algorithms for Optimisation"** by Raymond Chiong (Ed.)

Covers a wide range of algorithms with real-world case studies.

Includes GA, PSO, ACO, Differential Evolution, and Artificial Bee Colony.

Group Creation

Join a group on Canvas

Paper/ Tool Presentation (10%)

- Group of 3
- Discuss from reading material (5)
- Discuss one/ two paper related 5)
- Demo one tool that we can use to address the topic in discussion. Same tool can be used but need to show different problem domain (10)
- 15 min

Course Project

- Please start to work on it ASAP
- First consult with your supervisor(s) if needed
- Find/define a complex optimization problem in your research direction
- Try to make your course project a chapter of your thesis and/or a paper

Project

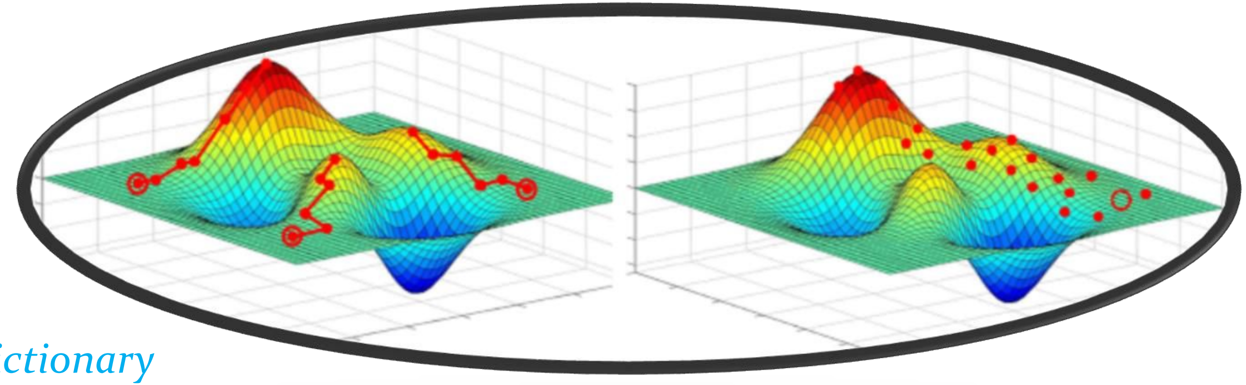
1. Page summary submission (5 pt)
    a. Title
    b. Problem statement
    c. Few reference work
2. Project checkpoint submission (5)
    a. Project paper draft (Introduction, Related work, New papers, challenges)
3. Project final package (20 pt)
    a. Demo presentation
    b. Final paper (IEEE conference format 12 pt as pdf)
    c. Group member evaluation (Individual submission)

Timeline

Finalizing presentation topic	Monday, June 30, 2025
Project summary	Friday, July 4, 2025
Assignment 1 Due	Tuesday, July 15, 2025
Project paper (Introduction, Related work, challenges)	
Assignment 2 Due	Wednesday, July 30, 2025
Final Project Presentations	
Final Project Paper Due	Tuesday, August 5, 2025

What is Optimization?

An act, process, or methodology of making something (as a design, system, or decision) as fully perfect, functional, or effective as possible.



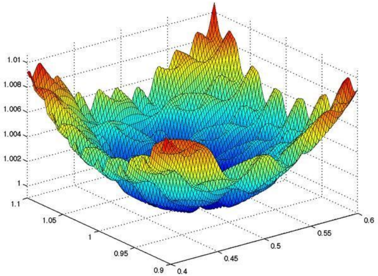
Merriam-Webster Online Dictionary

What Is Optimization?

Examples for Optimization

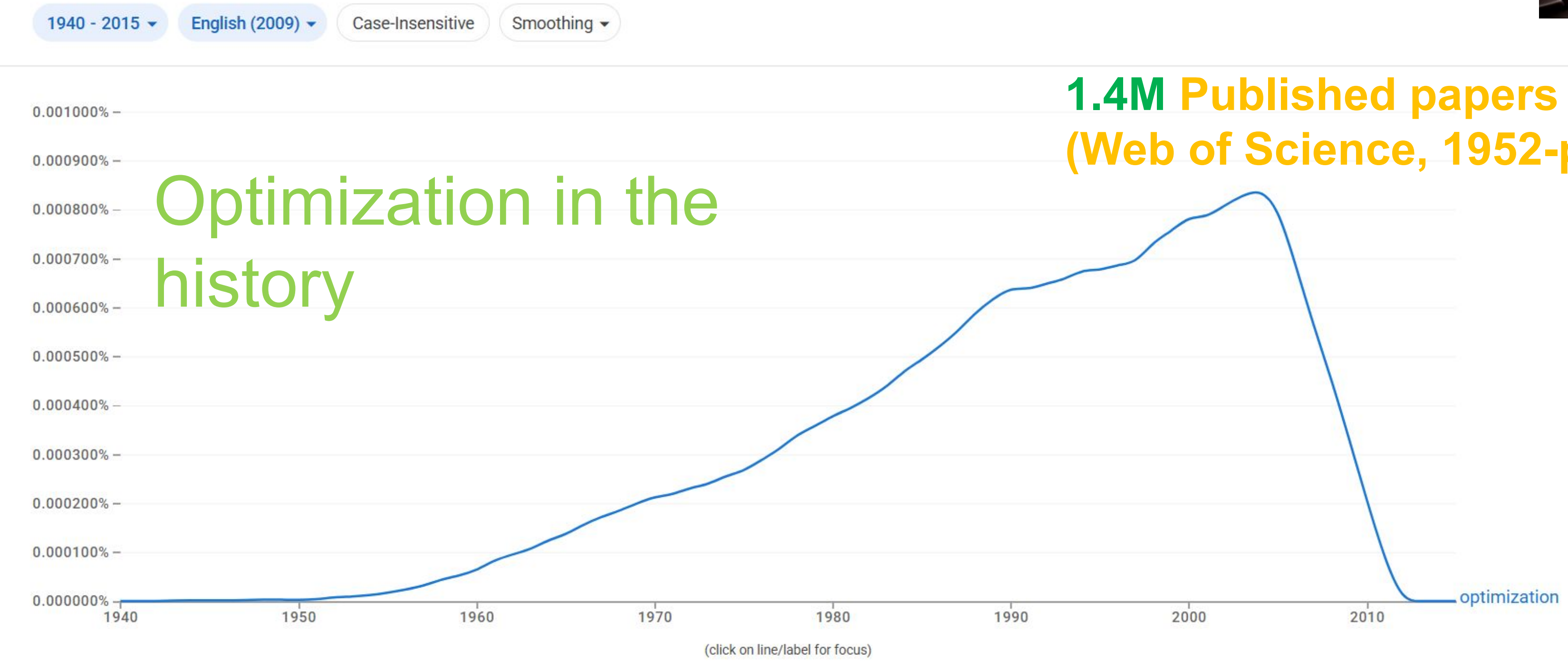
- Maximization of performance, accuracy, efficiency, benefit, stability, durability, life span, …….

- Minimization of development/maintenance cost, manufacturing cost, risk factor, error rate, wasting energy, utilized materials, negative environmental impact, response time, weight, …..



Optimization in the history

1.4M Published papers (Web of Science, 1952-present)



An Optimization Problem

- *Given:*

a function $f : A \to R$ from some set A to the real numbers

- *Looking for:*

  an element x_o in A such that $f(x_o) \leq f(x)$ for all x in A ("minimization") or such that $f(x_o) \geq f(x)$ for all x in A ("maximization").

Horse Racing

The slowest horse is the winner!

Q. How can you run this competition?

Horse Racing

By Switching the horses to convert a minimization problem to the maximization one!

A formal definition of a continuous optimization problem

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p \end{aligned}$$

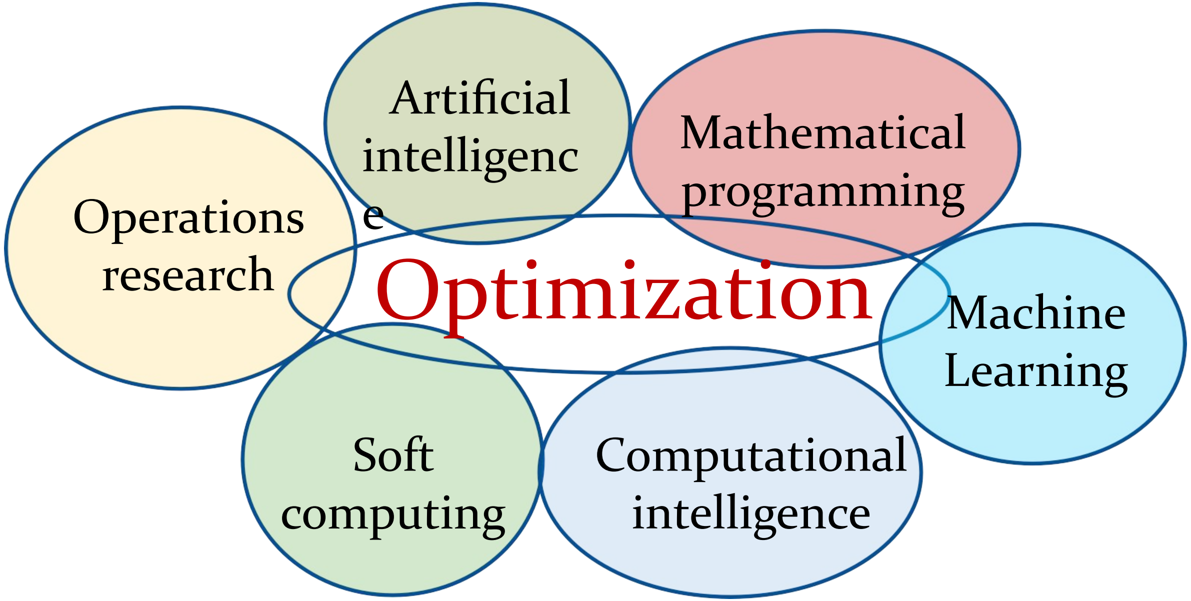
where

- $f(x) : \mathbb{R}^n \to \mathbb{R}$ is the **objective function** to be minimized over the variable x,
- $g_i(x) \leq 0$ are called **inequality constraints**, and
- $h_i(x) = 0$ are called **equality constraints**.

Let us first make our simple intuitive mini-dictionary about type of optimization problems

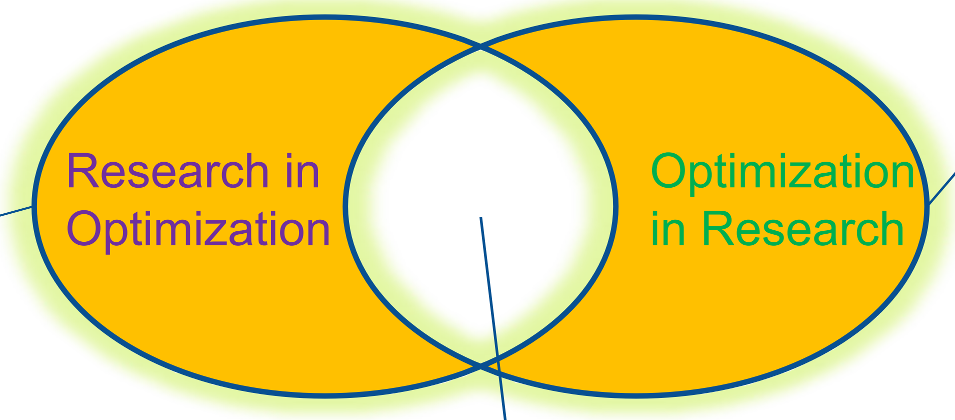
1) Objective function
2) Mathematical/Simulator/Experimental
3) Minimization/maximization
4) Dimension
5) Large-scale
6) Landscape
7) Discrete/continuous/mixed-type
8) Unimodal/multi-modal
9) One solution/many solutions
10) With know/ unknown solution
11) Design/control
12) Exact/approximate solution
13) Single/multi/many objectives
14) Black-box/grey-box problem
15) Constraint, box-constraint
16) Static/Dynamic
17) Noisy
18) Single- or Multi-level
19) Deceptive
20) Combinatorial
21) Expensive
22) Variable dimension size
23) Linear/nonlinear
24) Convex/concave
25) Interactive (single- or multi – objective)
26) Separable/non separable

Where is the home of optimization?



Research and Optimization

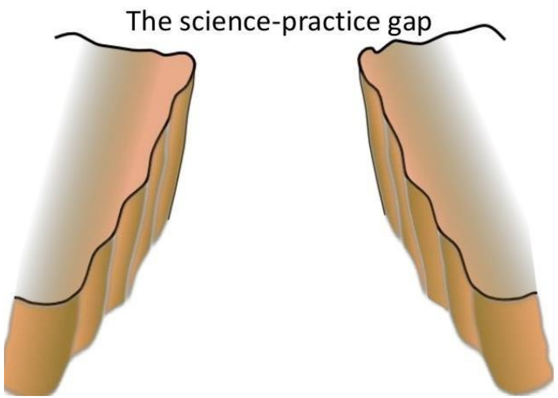
Solving real-world optimization problems using existence algorithms



Improving optimization algorithms

Tailoring optimization algorithm to solve specific real-world problems

Theory and Practice!

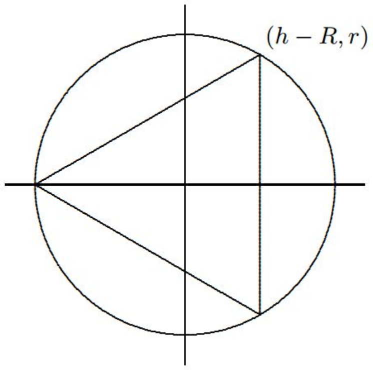


Theory is when one knows everything but nothing works. Practice is when everything works but nobody knows why.

If theory and practice go hand in hand: nothing works and nobody knows why!

A simple optimization problem (Application of the Derivative)

If you fit the largest possible cone inside a sphere, what fraction of the volume of the sphere is occupied by the cone? (Here by "cone" we mean a right circular cone, i.e., a cone for which the base is perpendicular to the axis of symmetry, and for which the cross-section cut perpendicular to the axis of symmetry at any point is a circle.)



Cone in a sphere.

https://www.whitman.edu/mathematics/calculus/calculus_06_Applications_of_the_Derivative.pdf

A simple optimization problem (Application of the Derivative)[Cont.]

Notice that the function we want to maximize, $\pi r^2 h/3$, depends on *two* variables. This is frequently the case, but often the two variables are related in some way so that "really" there is only one variable. So our next step is to find the relationship and use it to solve for one of the variables in terms of the other, so as to have a function of only one variable to maximize. In this problem, the condition is apparent in the figure: the upper corner of the triangle, whose coordinates are $(h-R, r)$, must be on the circle of radius R. That is,

$$(h-R)^2 + r^2 = R^2.$$

We can solve for h in terms of r or for r in terms of h. Either involves taking a square root, but we notice that the volume function contains r^2, not r by itself, so it is easiest to solve for r^2 directly: $r^2 = R^2 - (h-R)^2$. Then we substitute the result into $\pi r^2 h/3$:

$$\begin{aligned} V(h) &= \pi(R^2 - (h-R)^2)h/3 \\ &= -\frac{\pi}{3}h^3 + \frac{2}{3}\pi h^2 R \end{aligned}$$

https://www.whitman.edu/mathematics/calculus/calculus_06_Applications_of_the_Derivative.pdf

A simple optimization problem (Application of the Derivative)[Cont.]

We want to maximize $V(h)$ when h is between 0 and $2R$. Now we solve $0 = f'(h) = -\pi h^2 + (4/3)\pi hR$, getting $h = 0$ or $h = 4R/3$. We compute $V(0) = V(2R) = 0$ and $V(4R/3) = (32/81)\pi R^3$. The maximum is the latter; since the volume of the sphere is $(4/3)\pi R^3$, the fraction of the sphere occupied by the cone is

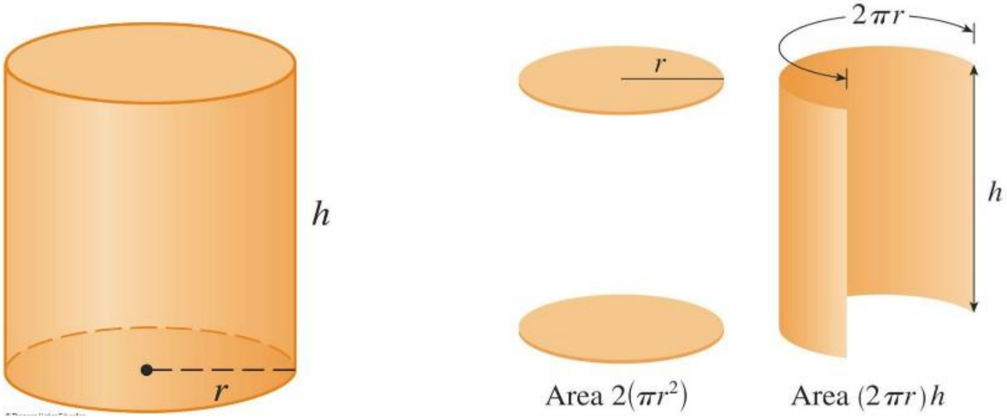
$$\frac{(32/81)\pi R^3}{(4/3)\pi R^3} = \frac{8}{27} \approx 30\%.$$

$\square$

https://www.whitman.edu/mathematics/calculus/calculus_06_Applications_of_the_Derivative.pdf

EXAMPLE : A manufacturer needs to make a cylindrical can that will hold 1.5 liters of liquid. Determine the dimensions of the can that will minimize the amount of material used in its construction.

Solution: We first draw a picture:



The next step is to create a corresponding mathematical model:

$$\text{Minimize: } A = 2\pi r^2 + 2\pi rh$$

$$\text{Constraint: } V = \pi r^2 h = 1500$$

Analytical Vs. Algorithmic Optimization

Analytical optimization involves deriving explicit formulas using mathematics. Think solving equations directly—like using derivatives, Lagrange multipliers, or KKT conditions to find exact “optima.”

Algorithmic optimization refers to computational methods (e.g., gradient descent, interior-point methods). These are iterative techniques that numerically approximate solutions, especially when closed-form expressions are impossible or impractical.

Analytical Vs. Algorithmic Optimization

Five types of problems to solve

- Type 1: Problems with known global solution(s)
- Type 2: Problems with best known solution(s)
- Type 3: Problems with unknown solution but with known optimal value for objective function
- Type 4: Problems with unknown solution and unknown optimal value for objective function

Name a science or engineering field with no fingerprint of optimization there?

Applications of Optimization

- Countless
- Many challenging applications in science and industry can be formulated as optimization problems.
- Minimization of time, cost, and risk

Power & Energy Systems

Optimal Placement of Renewable Energy Sources in Smart Grids
Use Genetic Algorithms or Particle Swarm Optimization to minimize loss and cost.

Economic Load Dispatch using Quadratic Programming
Solve constrained power dispatch with equality and inequality constraints.

Maintenance Scheduling of Thermal Power Units using GA
Inspired by Rao's examples; optimize scheduling under reserve constraints.

Multi-objective Optimization of a Microgrid Design
Balance cost, reliability, and emissions using NSGA-II.

Exergetic Optimization of Tri-generation Systems
Model using real data, solve using Differential Evolution.

Power & Energy Systems

Optimal Placement of Renewable Energy Sources

- **SajjadAsefi/RenewableEnergyManagement**: PSO-based microgrid management with renewables and demand response https://github.com/SajjadAsefi/RenewableEnergyManagement

- **ahmadi26/DG-Placement**: Optimal sizing and placement of distributed generation (PV/WT) using PSO, TLBO, BFO https://github.com/ahmadi26/DG-Placement

Economic Load Dispatch (Quadratic Programming)

- **nkpanda97/EconomicDispatch**: MILP-based multi-period economic load dispatch with Pyomo https://github.com/nkpanda97/EconomicDispatch
- **abhishekmalali/Economic-Dispatch-Optimization-methods**: MATLAB solvers for economic dispatch https://github.com/abhishekmalali/Economic-Dispatch-Optimization-methods

Power & Energy Systems

Maintenance Scheduling of Thermal Units

- Consider adapting maintenance scheduling using GA from **tohid-yousefi/Meta-Heuristics** repo
  https://github.com/topics/economic-dispatch-problem

Multi-objective Microgrid Design

- **vnu056/Project-details**: GA+PSO hybrid to optimize microgrid loss and cost
  https://github.com/vnu056/Project-details/blob/main/README.md

Exergetic Optimization of Tri-generation Systems

- You could adapt optimization examples from **oemof** or **oHySEM** for tri-generation planning
  https://github.com/IIT-EnergySystemModels/oHySEM

Robotics & Control Systems

Fuzzy Logic Controller Optimization for Cruise Control
Tune fuzzy parameters using PSO or DE.

PID Controller Tuning using Constrained Optimization
Apply line search and trust-region methods for parameter tuning.

Trajectory Planning for 6-DOF Robotic Arm
Minimize energy or time using convex/nonlinear optimization.

Swarm Robotics Coordination via PSO
Design local rules and global fitness function for movement coordination.

Inverse Kinematics using Constrained Optimization
Use Lagrangian multipliers or KKT conditions.

Robotics & Control Systems

Fuzzy Logic Cruise Control Optimization

- **IsuruSankhajith/fuzzy-cruise-controller**: Fuzzy cruise control implementation https://github.com/IsuruSankhajith/fuzzy-cruise-controller-

Swarm Robotics Coordination with PSO

  - **vinaychetnani/Particle-Swarm-Optimization**: PSO in smart grid scheduling, adapt to swarm coordination https://github.com/vinaychetnani/Particle-Swarm-Optimization

Inverse Kinematics via Constrained Optimization

- Use Python solvers like **Gekko** to formulate kinematics problems https://github.com/qpsolvers/qpsolvers

Communication & Signal Processing

Adaptive Beamforming via Convex Optimization
Design optimal weight vectors to maximize signal-to-interference ratio.

Antenna Array Pattern Synthesis
Shape antenna radiation patterns using GAs or QP.

Compressive Sensing Signal Reconstruction
Sparse signal recovery using L1-norm minimization.

Channel Allocation in Wireless Sensor Networks
Optimize channel use under interference constraints.

Modulation Scheme Selection using Multi-Objective GA
Tradeoff between BER, power, and bandwidth.

Adaptive Beamforming (Convex Optimization)

- Use **qpsolvers** or **HiGHS** QP solvers for beamforming weight calculation https://github.com/qpsolvers/qpsolvers

Antenna Array Pattern Synthesis

- Leverage **qpsolvers** to adjust amplitude/phase weights for beam steering

Compressive Sensing Reconstruction

- Implement basis pursuit using **HiGHS** solver

Channel Allocation in Wireless Networks

- Use **HiGHS** or **qpsolvers** to formulate interference minimization

Machine Learning & AI

Hyperparameter Optimization for Neural Networks using Optuna
Automated tuning of learning rate, batch size, etc.

Feature Selection using GA for Classification Tasks
Improve model performance by selecting optimal feature subsets.

Training Energy-Efficient Neural Networks
Minimize energy usage under accuracy constraint.

Optimizing Weights in Neuro-evolutionary Algorithms
Use PSO to evolve ANN weights instead of backpropagation.

Constraint-Based Adversarial Example Generation
Use optimization to generate inputs that fool ML models under bounds.

Machine Learning & AI

Neuro-evolution with PSO

- Use the PSO fuzzy control code as template for evolving network weights
- https://pmc.ncbi.nlm.nih.gov/articles/PMC10676367/

Embedded Systems & VLSI

Task Scheduling in Multiprocessor Systems using PSO
Minimize total execution time with communication overheads.

Floorplanning of VLSI Circuits via Simulated Annealing
Optimize area and wire length with hard constraints.

Energy-aware Scheduling in Real-Time Embedded Systems
Use LP/QP for minimizing energy under deadline constraints.

Optimized Sensor Placement in IoT Networks
Maximize coverage and minimize cost using GA.

Logic Gate Minimization using Metaheuristics
Reduce gate count in combinational logic using GA or SA.

Embedded Systems & VLSI

Multiprocessor Task Scheduling with PSO

- Adapt **vinaychetnani/Particle-Swarm-Optimization** for task scheduling
  https://github.com/vinaychetnani/Particle-Swarm-Optimization

- **Energy-aware Scheduling in Real-Time Systems**
  - Use QP solver like **HiGHS** for energy/deadline constraints

- **Sensor Placement in IoT via GA:** Use **tohid-yousefi/Meta-Heuristics** GA framework for placement optimization