Introduction to Artifical Intelligence

Rohit Salgotra

rohits@agh.edu.pl

Faculty of Physics & Appied Computer Science,
30-059 Kraków
Poland

November 19, 2025

Part I

AI Laboratory Research Projects

Course Context

Syllabus Coverage (Labs + Projects)

- Artificial Neural Networks, learning algorithms (supervised/unsupervised)
- Evolutionary algorithms (GA/PSO/DE) and applications
- Expert systems, knowledge representation, reasoning
- Agent-based systems, basic architectures
- Fuzzy logic, uncertainty handling, hybrid AI
- Search / planning (used inside routing and task allocation)

Outcome: Each group completes *one* project with:

- A working AI system (ANN/EA/Agent/Fuzzy Hybrid)
- Experiments, figures, tables, and statistical tests
- A short paper draft suitable for GECCO Student Workshop / CEC / IJCNN

Lab Flow (Common to All Projects)

Lab	Theme and Output
1	Setup, dataset/simulation understanding, metrics ⇒ baseline stats
2	ANN from scratch (MLP) ⇒ working classifier/regressor
3	Optimizers (SGD/Momentum/Adam) ⇒ comparison table
4	Intro to GA (EA baseline) ⇒ simple optimizer integrated
5	Small novelty: e.g. self-adaptive / chaotic mutation
6	Choose project: one of the four routing/dispatch problems
7	Application layer: maps, routes, explanations, membership plots
8	Benchmarking + statistics: Wilcoxon/Friedman tables
9	Paper writing (IEEE LaTeX) ⇒ full draft
10	Final validation + presentation ⇒ submission-ready package

Each project *inherits* this structure and adds domain-specific details.

The Project Themes

Each group chooses **one** project:

- **P1: Emergency Room Overcrowding and Staffing Optimization**
  (Healthcare impact, ML + EA + uncertainty)
- **P2: Smart Street-Light Scheduling**
  (Energy + safety, hybrid ANN + Fuzzy + EA)
- **P3: First-Responder Drone Deployment**
  (Disaster response, ANN + EA, spatial optimization)

All projects:

- Use a common ANN/EA core from Labs 1–5
- Introduce uncertainty (random variation in demand/traffic/conditions)
- Require one **small but clear novelty** (Lab 5)
- End with a reproducible paper (Lab 9–10)

General Implementation Expectations

AGH

Each project must include:

Code structure:

- `data_gen.py` or `data_loader.py`
- `ann_model.py` (forward, backprop, training loop)
- `ea_optimizer.py` (GA/DE/PSO)
- `plots.py` (figures for paper)

Experiments:

- Multiple runs (e.g., 20 runs) with fixed random seeds
- Mean $\pm$ std for metrics

Evaluation:

- At least one statistical test (e.g., Wilcoxon signed-rank)
- Clear comparison between baseline and improved method

Part II

Emergency Room Overcrowding and Staffing Optimization

www.agh.edu.pl

Motivation and Problem

Title Suggestion:
"An Evolutionary Learning Approach for Robust ER Staffing Under Patient-Arrival Uncertainty"

Motivation

- Emergency Rooms (ERs) suffer from overcrowding and long waiting times.
- Patient arrivals vary by time of day, day of week, weather, and epidemics.
- Over-staffing is expensive; under-staffing is dangerous.

Core Problem

- **Predict** near-future patient arrivals (ANN).
- **Optimize** hourly staffing levels (EA) under uncertain arrivals.

Data and Features

Data Sources (options):

Real hospital data (if available), or

Synthetic time-series of arrivals per hour.

Minimum fields to simulate/use:

Time index (hour of day, day of week, maybe month)

Weather category (sunny, rainy, cold, etc.)

Number of arrivals in that hour

Optional: triage severity level fractions (e.g., mild, moderate, severe)

Uncertainty:

Add noise to arrivals: $\tilde{A} = A + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$

ANN for Arrival Prediction

Task: Predict next-hour arrivals from recent history.

Inputs (features):

- Hour of day (encoded as one-hot or $\sin, \cos$)
- Day of week (one-hot)
- Last k hours of arrivals (e.g., $k = 4$)
- Weather category

Output:

- Real-valued prediction: $\hat{A}_{t+1}$ (expected arrivals)

Implementation Tasks (Labs 2–3):

- Implement feed-forward MLP with one or two hidden layers.
- Use MSE loss; compare SGD, Momentum, Adam.
- Plot training and validation loss (Lab 3).

EA for Staffing Schedule

Goal: Allocate doctors/nurses per hour for a planning horizon (e.g., 24h).

Chromosome Encoding:

Vector of length 24: $x = [s_1, s_2, \dots, s_{24}]$

s_h = number of staff scheduled in hour h (integer).

Fitness Function (minimize):

$$f(x) = \alpha \cdot \text{avg waiting time} + \beta \cdot \text{staffing cost} + \gamma \cdot \text{overload penalty},$$

where overload penalty is high when predicted arrivals $\gg$ staff capacity.

Uncertainty Handling:

Evaluate each schedule under N arrival scenarios generated from ANN + noise.

Use **average** or **worst-case** fitness across scenarios.

Novelty Ideas

AGH

Each group must implement at least one small idea beyond a plain GA.

- **Averaging-based robust fitness**:
  - Evaluate each schedule under multiple stochastic arrival scenarios.
  - Use mean and/or variance in fitness.
- **Self-adaptive mutation:**
  - Increase mutation rate when population diversity is low or fitness stagnates.
- **Chaotic mutation:**
  - Use a chaotic map to generate mutation probabilities.
- **Multi-objective EA (optional advanced):**
  - Minimize waiting time vs cost (Pareto front).

Work Flow

Lab	What you implement for P1
1	Create synthetic ER arrival time-series; basic plots (daily/weekly patterns).
2	Implement MLP to predict next-hour arrivals; get training loop working.
3	Compare SGD vs Momentum vs Adam; choose best optimizer; save model.
4	Implement basic GA for staffing schedule (no uncertainty yet).
5	Add uncertainty in arrivals; implement robust fitness (averaging / chaos).
6	Tune GA hyperparameters (pop size, mutation, crossover); run 10–20 trials.
7	Generate figures: prediction curves, staffing heatmaps, GA convergence.
8	Run statistical comparison (baseline vs improved GA); create result tables.
9	Write paper sections: Intro, Method, Experiments, Results.
10	Final reruns, figure polishing, 5-min presentation.

Expected Figures and Tables

Figures:

- Time-series of actual vs predicted arrivals.
- Heatmap: staff levels over hours (baseline vs optimized).
- GA convergence plot: best fitness vs generations.
- Boxplot of waiting time across scenarios.

Tables:

- Prediction error (MAE/MSE) for different optimizers.
- Average waiting time, cost, overloads for baseline vs improved methods.
- p-values from Wilcoxon tests.

Part III

Smart Street-Light Scheduling

Motivation and Problem

Title Suggestion:
"A Hybrid Fuzzy-Evolutionary Approach for Smart Street-Light Scheduling under Uncertain Activity"

Motivation

- Street lighting is essential for safety but consumes energy.
- Human activity and accidents depend on time, weather, and location.
- Static schedules waste power; purely reactive rules can miss risks.

Core Problem

- Predict activity level (ANN).
- Use fuzzy logic to map activity + visibility to lighting priority.
- Optimize schedules using EA to balance safety vs energy cost.

Data and Features

Data sources:

- Weather time-series (temperature, rainfall, fog).
- Synthetic or real pedestrian counts (if available).
- Accident risk (synthetic) as a function of activity and darkness.

Feature examples:

- Hour of day, day of week.
- Weather conditions.
- Historical activity in that street.

ANN for Activity Prediction

Inputs:

- Time-of-day and day-of-week.
- Weather features.

Output:

- Activity index: e.g., expected number of pedestrians.

Labs 2–3 Tasks:

- Implement MLP with regression output (activity index).
- Compare optimizers and choose best.

Fuzzy Logic Module

Fuzzy Inputs:

- Activity: *low*, *medium*, *high*.
- Visibility: *poor*, *ok*, *good*.

Fuzzy Output:

- Lighting priority: *off*, *dim*, *full*.

Example rules:

- IF activity is high AND visibility is poor THEN lighting is full.
- IF activity is low AND visibility is good THEN lighting is off.

EA for Schedule Optimization

Chromosome:

Lighting levels for each hour (or 15-min block) over a day: $x = [L_1, L_2, \ldots, L_T]$.

Fitness:

$f = -\text{safetyScore} + \lambda \cdot \text{energyCost}$.

SafetyScore from fuzzy module and predicted activity.

EnergyCost proportional to time and intensity of lighting.

Novelty:

EA fine-tunes fuzzy membership function parameters.

Compare "handcrafted fuzzy" vs "EA-tuned fuzzy".

Work Flow

Lab	P3 Milestones
1	Create synthetic street segments, weather, and activity data.
2	Build MLP to predict activity given weather + time.
3	Compare optimizers; finalize ANN training.
4	Implement basic EA for lighting schedule (no fuzzy yet).
5	Design fuzzy variables and rule base; integrate into fitness.
6	Add EA-based tuning of membership parameters (novelty).
7	Plot membership functions, schedules vs activity, energy usage charts.
8	Compare baseline vs fuzzy vs fuzzy+EA; run stats.
9	Write paper focusing on hybrid ANN+Fuzzy+EA design.
10	Final reruns, slide prep, group presentations.

Expected Figures and Tables

Figures:

- Fuzzy membership functions (before and after EA tuning).
- Lighting schedule vs time vs activity.
- Energy usage vs safety score trade-off plot.

Tables:

- Comparison of energy cost and safety for:
  - Static schedule
  - Fuzzy schedule (handcrafted)
  - Fuzzy+EA schedule (tuned)
- Statistical test results.

Part IV

First-Responder Drone Deployment

Project VI: Motivation and Problem

Title Suggestion:
"Robust Evolutionary Optimization of Emergency Drone Deployment Under Spatial Demand Uncertainty"

Motivation

- Drones can deliver first-aid kits, AEDs, or small supplies rapidly.
- Emergencies happen in random places with varying frequency.
- Fixed station locations might be sub-optimal under changing patterns.

Core Problem

- Predict spatial emergency risk (ANN).
- Place a limited number of drone bases to minimize response time.

Data and Simulation

Synthetic City Grid:

- Represent city as a 2D area.
- Generate candidate grid points (potential drone base locations).
- Generate emergency call hotspots (e.g., based on Gaussian mixtures).

Features:

- Time-of-day, day-of-week.
- Area type (residential, commercial, etc., synthetic labels).

Uncertainty:

- Add random variation to call frequency in each area.

ANN for Hotspot Prediction

Inputs:
- Time-of-day, day-of-week.
- Area category.

Output:
- Probability or expected number of emergencies in each region.

Hint:
- Implement MLP for hotspot intensity prediction.
- Evaluate prediction quality; choose best optimizer.

EA for Drone Base Placement

Chromosome:

Choose K base locations from candidate set.

Encoding: vector of indices of chosen base locations.

Fitness:

Simulate emergency calls per scenario.

Compute response time as distance/speed from nearest base.

Fitness = average response time (and optional penalties) across scenarios.

Novelty:

Scenario-based robustness: multiple stochastic demand patterns.

Multi-objective EA: minimize avg response time and maximize coverage.

Work Flow

Lab	Milestones
1	Create synthetic city grid and emergency hotspots; visualize map.
2	Build ANN for spatial hotspot prediction.
3	Compare optimizers; finalize hotspot predictor.
4	Implement basic EA for choosing K drone base locations (no uncertainty).
5	Add stochastic demand scenarios; compute robust fitness (averaged).
6	Add novelty: scenario weighting, or multi-objective strategy.
7	Plot base placements, coverage maps, response time histograms.
8	Compare baseline vs improved EA solutions statistically.
9	Write paper draft highlighting spatial, robust optimization.
10	Final reruns, visualization refinement, presentation.

Expected Figures and Tables

AGH

Figures:

- City map with drone base locations and hotspots.
- Response time distribution plots.
- Pareto front (if multi-objective used).

Tables:

- Average response time for baseline vs improved.
- Coverage stats (fraction of area within threshold time).
- Statistical test results.

Common Paper Structure (All Projects)

Each group writes a 6–8 page paper with:

Introduction
Problem, motivation, societal impact, contributions.

Background
Brief on ANN, EA, and any fuzzy/agent methods used.

Problem Formulation
Variables, objective(s), constraints, uncertainty.

Methodology
Data generation, ANN design, EA design, novelty.

Experimental Setup
Data sizes, parameters, metrics, number of runs.

Results and Discussion
Tables, figures, statistical tests, interpretations.

Conclusion and Future Work

Final Deliverables (All Projects)

Each group must submit:

Code:

Scripts for data generation, ANN, EA, plots.
README with instructions to reproduce results.

Results:

Figures (PDF/PNG) used in the paper.
CSV/Excel tables with metrics.

Paper:

LaTeX source + compiled PDF.

Presentation:

5-minute talk in Lab 10.

Questions?