

# Literature Review: Evolutionary Scheduling of Courses or Work Shifts

Sabína Ságová  
Charles University

27 October 2025

## 1 Older Trends in Scheduling Research

### 1.1 Older Publications

Here are details of older key papers:

1. A Hybrid Approach to Quantitative Software Project Scheduling within Agile Frameworks

**Authors/Year:** Michael Jahr (2014)

**Problem Type:** Quantitative project scheduling in agile software projects using a hybrid approach combining mathematical optimization with Scrum methodology

**Dataset / Instances:** Real-world software development projects from a German financial IT company; four pilot projects (two redevelopment and two new software applications) used to validate the hybrid framework

**Evaluation Method / Metrics:** Project cost and duration deviations compared against traditional Scrum-based management; comparative analysis of adherence to deadlines and budgets

**Algorithm Type:** Modified Multi-Mode Resource-Constrained Project Scheduling Problem (MRCPSSP) model

**Encoding / Individual Design:** Activity-on-the-Node Graph (AoNG) representing all software development phases (requirements analysis, design, programming, testing, delivery); developer experience encoded as modes (senior vs. regular)

**Operators:** Mixed-integer linear optimization via GAMS–CPLEX; incorporates renewable (developer capacity) and non-renewable (budget) resources with time lags and precedence constraints

**Comparison Methods:** Traditional Scrum-only project management versus hybrid Scrum–MRCPPSP scheduling approach

**Key Findings:** The hybrid model significantly improved adherence to deadlines and budgets, reducing average project delay by over 20% and cost overruns by up to 40%. Demonstrated that quantitative models can complement agile frameworks effectively without compromising flexibility.

**Citation Count (to date):** 56 (Google Scholar, 2025)

2. Application of Evolutionary Algorithms in Project Management

**Authors/Year:** Christos Kyrklidis and Georgios Dounias (2014)

**Problem Type:** Resource Leveling Problem  
(Time-Constrained Project Scheduling)

**Dataset / Instances:** Small and medium benchmark projects from public project datasets (e.g., PSPLIB)

**Evaluation Method / Metrics:** Objectives include maximum resource usage (Gf),

resource leveling index (RLI), and standard deviation (StD); 50 runs with statistical evaluation of near-optimality

**Algorithm Type:** Genetic Algorithm (GA)

**Encoding / Individual Design:** Chromosome encodes start-times of non-critical activities; critical ones fixed

**Operators:** Two-point crossover (70%), mutation (20%), elitism (10%);

local search around elite chromosomes to avoid premature convergence

**Comparison Methods:** Exhaustive enumeration for small problems; compared with heuristic, ACO, ANN, and PSO methods

**Key Findings:** GA efficiently finds near-optimal resource profiles, outperforming traditional methods and scaling well for large projects

**Citation Count (to date):** 9 (Google Scholar, 2025)

3. Dynamic Staffing and Rescheduling in Software Project Management: A Hybrid Approach

**Authors/Year:** Yujia Ge, Bin Xu (2016)

**Problem Type:** Dynamic software project scheduling and rescheduling with human productivity and stability factors

**Dataset / Instances:** Real-world project case from Boyuan Software Company (19 tasks, 9 employees), plus simulation experiments with

varying project sizes (4–30 tasks, 3–20 employees)

**Evaluation Method / Metrics:** Total project cost, stability–efficiency trade-off, schedule deviation, and expert comparison; statistical evaluation through multiple project scenarios

**Algorithm Type:** Hybrid metaheuristic combining Genetic Algorithm (GA) and Hill Climbing (HC) for scheduling and rescheduling optimization

**Encoding / Individual Design:** Two-part genome: (A) task–employee assignment matrix (load levels 0–100%), (B) priority list determining topological task order; ensures valid project precedence constraints

**Operators:** One-point crossover for assignment array, order crossover for task priority; random mutation on task or employee genes; elitism and HC-based local refinement for convergence acceleration

**Comparison Methods:** Steady GA, Hill Climbing (HC), and hybrid GA–HC evaluated on small, medium, and large project cases; expert human schedules used as benchmarks

**Key Findings:** The hybrid GA–HC achieved faster convergence and lower cost schedules than GA or HC alone, particularly for large-scale projects. Incorporating human learning, communication overhead, and schedule pressure improved realism. Expert evaluation confirmed comparable or superior scheduling quality at significantly lower computation time.

**Citation Count (to date):** 36 (Google Scholar, 2025)

4. Evolutionary Algorithm Performance Evaluation in Project Time-Cost Optimization

**Authors/Year:** A.P. Chassiakos, G. Rempis (2019)

**Problem Type:** Project Time-Cost Trade-Off (TCT) Optimization — NP-hard scheduling problem

**Dataset / Instances:** Three benchmark projects from the literature: 7-, 18-, and 29-activity networks, plus two composite networks (series and parallel)

**Evaluation Method / Metrics:** Solution quality (success rate, deviation from optimal), computational time (mean and standard deviation); 10 runs per algorithm

**Algorithm Type:** Comparative analysis of five Evolutionary Algorithms (EAs): Hill Climber, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), and Artificial Bee Colony (ABC)

**Encoding / Individual Design:** Activity duration-based representation for direct and indirect cost optimization; chromosome encodes alternative activity execution modes

**Operators:** GA variations include Custom GA, Saw-Tooth GA (STGA), and Micro GA (microGA) with population restarts; PSO and DE use velocity adjustment and differential mutation

**Comparison Methods:** Multiple EA variants compared using same XLOptimizer software environment to ensure unbiased evaluation

**Key Findings:** Enhanced PSO (EPSO), Differential Evolution (DE), and Micro GA achieved the best balance of accuracy and computational efficiency. EPSO converged fastest; DE delivered optimal results with minimal deviation. ABC performed robustly but slower; Hill Climber was fastest but inconsistent. No single algorithm dominated all problem sizes.

**Citation Count (to date):** 19 (Google Scholar, 2025)

## 2 Recent Trends in Scheduling Research (2021 – Now)

Here are details of key papers from 2021:

1. Class Schedule Generation Using Evolutionary Algorithms

**Authors/Year:** Mohit Kumar Kakkar, Jajji Singla, Neha Garg, Gourav Gupta, Prateek Srivastava, Ajay Kumar (2021)

**Problem Type:** University course timetabling and class scheduling problem (CSP) — NP-hard optimization problem involving allocation of courses, rooms, and faculty under strict and soft constraints

**Dataset / Instances:** Institutional timetable data from Chitkara University, India; includes 10 groups, 14 instructors, 5 classrooms, and mixed lecture/lab scheduling requirements

**Evaluation Method / Metrics:** Fitness evaluation based on number of constraint violations (hard and soft), cost of clashes, and computational efficiency; GA and MA compared across convergence rate and schedule feasibility

**Algorithm Type:** Genetic Algorithm (GA) and Memetic Algorithm (MA) for constraint-based timetable generation

**Encoding / Individual Design:** Chromosome represented as 3-column array for each course — room, instructor, and timeslot IDs; ensures no duplication or room/time conflicts

**Operators:** One-point crossover (probability = 0.9) and random mutation (probability = 0.025); population size = 100; maximum generations = 500; GA selection based on fitness proportionality

**Comparison Methods:** GA versus MA on cost-based and clash-based performance metrics

**Key Findings:** GA produced conflict-free timetables more efficiently than MA, converging after only 25 iterations on average. Both algorithms achieved feasible timetables, but GA exhibited superior adaptability and stability, avoiding local minima where MA slowed under no-clash conditions. Demonstrated significant time savings and improved scheduling accuracy over manual methods.

**Citation Count (to date):** 38 (Google Scholar, 2025)

2. A Hybrid Genetic Algorithm for Nurse Scheduling Problem Considering the Fatigue Factor

**Authors/Year:** Atefeh Amindoust, Milad Asadpour, Samineh Shirmohammadi (2021)

**Problem Type:** Multi-objective Nurse Scheduling Problem (NSP) incorporating human fatigue as a new optimization factor under COVID-19 healthcare pressure

**Dataset / Instances:** Real case study from a hospital department in Esfahan, Iran, involving 14 nurses and three shifts (morning, evening, night) for one month during COVID-19

**Evaluation Method / Metrics:** Comparison between manual scheduling and hybrid GA optimization on cost, fatigue, and computation time; metrics include total cost (IRR), fatigue index, and runtime (minutes)

**Algorithm Type:** Hybrid Genetic Algorithm (GA) integrated with neighborhood search structures

**Encoding / Individual Design:** Chromosome as a  $4 \times 3$  matrix representing employment, dismissal, rest level ( $m$ ), and assigned demand ( $D$ ) across shifts

**Operators:** Selection, crossover, and mutation with four customized neighborhood structures for adaptive exploration; elitist Pareto-set update based on cost-fatigue trade-off

**Comparison Methods:** Manual (human resource-based) scheduling currently used in the case study hospital

**Key Findings:** The hybrid GA reduced average nurse fatigue by over 15% while requiring only 1.28 minutes of computation compared

to 66.7 minutes manually. Though total cost increased slightly (less than 0.1%), overall fairness, balance, and schedule quality improved significantly. Demonstrated generalizability to other healthcare and continuous service systems.

**Citation Count (to date):** 54 (Google Scholar, 2025)

3. Optimizing Multi-Mode Time–Cost–Quality Trade-Off of Construction Project Using Opposition Multiple Objective Differential Evolution  
**Authors/Year:** Duc-Long Luong, Duc-Hoc Tran, Phong Thanh Nguyen (2021)

**Problem Type:** Multi-mode Time–Cost–Quality Trade-Off Problem (TCQTP) in construction project scheduling

**Dataset / Instances:** Numerical case study of a real highway construction project involving multiple construction modes and activity options

**Evaluation Method / Metrics:** Pareto front quality metrics — convergence and diversity compared using Hypervolume, Generational Distance (GD), and visual comparison with benchmark algorithms

**Algorithm Type:** Opposition-based Multiple Objective Differential Evolution (OMODE)

**Encoding / Individual Design:** Real-coded chromosome where each gene represents selected execution mode of an activity (defining duration, cost, and quality attributes)

**Operators:** Differential mutation and crossover; opposition-based learning (OBL) for both population initialization and generation jumping to enhance diversity and convergence speed

**Comparison Methods:** Compared with NSGA-II, Multi-objective PSO (MOPSO), and standard MODE on the same project instance

**Key Findings:** OMODE achieved better Pareto front coverage and faster convergence than competing algorithms. Demonstrated superior balance among project time, cost, and quality objectives. Validated as an efficient and reliable alternative for construction TCQTP optimization.

**Citation Count (to date):** 181 (Google Scholar, 2025)

4. Modified Multi-Objective Evolutionary Programming Algorithm for Solving Project Scheduling Problems  
**Authors/Year:** Mohammad A. Abido, Ashraf Elazouni (2021)

**Problem Type:** Multi-mode project scheduling with conflicting objectives — time–cost trade-off, finance-based scheduling, and resource leveling

**Dataset / Instances:** Two benchmark problems — (i) 18-activity time–cost trade-off network (Feng et al., 1997); (ii) multi-objective finance–resource–profit trade-off case with resource constraints (Elazouni and Abido, 2014)

**Evaluation Method / Metrics:** Compared Pareto front diversity, normalized distance, set coverage, and proportion of feasible vs. infeasible solutions; benchmarked against NSGA-II and SPEA-II using 400-population, 1000-generation settings

**Algorithm Type:** Modified Multi-Objective Evolutionary Programming (MOEP) algorithm — mutation-only evolutionary approach tailored for discrete scheduling variables

**Encoding / Individual Design:** Integer-valued vectors representing activity start times and execution modes; feasibility preserved through precedence-based constraints

**Operators:** Adaptive Gaussian mutation handling discrete start-time shifts ( $\pm 4$  periods); external Pareto set updated each generation with clustering when exceeding  $\max = 50$  solutions

**Comparison Methods:** Strength Pareto Evolutionary Algorithm II (SPEA-II) and Non-Dominated Sorting Genetic Algorithm II (NSGA-II)

**Key Findings:** MOEP achieved superior Pareto front diversity (normalized distance 0.997 vs. 0.984 (NSGA-II) and 0.846 (SPEA-II)); produced 36 non-dominated solutions in Case I vs. 31 (NSGA-II); reduced infeasible solutions to 15  
**Citation Count (to date):** 41 (Google Scholar, October 2025)

5. Multi-objective Optimization for Improved Project Management: Current Status and Future Directions

**Authors/Year:** Kai Guo, Limao Zhang (2022)

**Problem Type:** Systematic review and scientometric analysis of multi-objective optimization (MOO) in project management and construction scheduling

**Dataset / Instances:** 411 peer-reviewed journal papers on MOO applications in construction (1991–2020), retrieved from Scopus, Web of Science, and PubMed

**Evaluation Method / Metrics:** Scientometric mapping of co-

occurring keywords, clusters, and publication trends; qualitative synthesis across six major application domains (planning, prefabrication, safety, automation, monitoring, and emergency management)

**Algorithm Type:** Review of multi-objective Evolutionary Algorithms (MOEAs), including GA, PSO, DE, ALO, and hybrid MOO models

**Encoding / Individual Design:** Conceptual mapping of decision variables, objective vectors, and Pareto fronts; discussion of explicit vs. implicit modeling (e.g., meta-models via ML methods such as ANN, SVM, RF)

**Operators:** Comparative discussion of GA, PSO, MOALO, and hybrid mechanisms (elitism, weighted selection, and Pareto-based ranking); includes analysis of decision rules — no preference, a priori, and a posteriori

**Comparison Methods:** Integrative review comparing algorithmic paradigms and decision-making strategies across project planning, scheduling, and construction management

**Key Findings:** Identified six key MOO application areas (planning/constructability, prefabrication/supply chain, workplace safety, automation/digitalization, structural health monitoring, and emergency management). Concluded that MOO is essential for complex scheduling and resource conflicts. Highlighted challenges including lack of dynamic adaptability, interpretability, and user interaction in current MOO models. Proposed three future directions — adaptive optimization, interpretable mapping, and interactive optimization.

**Citation Count (to date):** 193 (Google Scholar, 2025)

6. A Multi-Objective Agile Project Planning Model and a Comparative Meta-Heuristic Approach

**Authors/Year:** Nilay Ozcelikkan, Gulfem Tuzkaya, Cigdem Alabas-Uslu, Bahar Sennaroglu (2022)

**Problem Type:** Multi-objective optimization for Scrum sprint planning in agile software development projects

**Dataset / Instances:** Real-world dataset from the IT department of a major Turkish bank (150 user stories, 15 sprints); small- and medium-sized benchmark instances (10–60 stories) also tested

**Evaluation Method / Metrics:** Pareto-based performance comparison using Hypervolume (HV), Epsilon, Generational Distance (GD), Inverted Generational Distance (IGD, IGD+), and Spread indicators

**Algorithm Type:** Multi-objective Evolutionary Algorithms —



Non-dominated Sorting Genetic Algorithm II (NSGA-II) and Strength Pareto Evolutionary Algorithm 2 (SPEA2)

**Encoding / Individual Design:** String representation where each position corresponds to a user story and its value indicates assigned sprint

**Operators:** Simulated Binary Crossover (SBX) and Polynomial Mutation (PM); elitism and crowding distance for NSGA-II; strength, raw fitness, and density values for SPEA2

**Comparison Methods:** Preemptive Goal Programming (LINGO) for small and medium instances versus NSGA-II and SPEA2 for large-scale NP-hard instances

**Key Findings:** Both NSGA-II and SPEA2 effectively solve large Scrum planning problems; SPEA2 slightly outperforms NSGA-II in most quality indicators though results are very close. The model provides systematic, repeatable sprint planning improving capacity use, priority handling, and project risk management

**Citation Count (to date):** 26 (Google Scholar, 2025)

7. An Efficient Interval Many-Objective Evolutionary Algorithm for Cloud Task Scheduling Problem under Uncertainty

**Authors/Year:** Zhixia Zhang, Mengkai Zhao, Hui Wang, Zhihua Cui, Wensheng Zhang (2022)

**Problem Type:** Cloud Task Scheduling under uncertain conditions — formulated as an Interval Many-Objective Optimization Problem (I-MaOP)

**Dataset / Instances:** Simulated cloud computing environment built with CloudSim 3.0; 5 hosts, 10 data centers, 100 tasks, and 50 virtual machines (VMs) with uncertain bandwidth and MIPS parameters

**Evaluation Method / Metrics:** Performance evaluated using Inverted Generational Distance (IGD), Spacing Metric (SP), and Coverage; each algorithm tested over 30 independent runs with interval-based uncertainty modeling

**Algorithm Type:** Interval Many-Objective Evolutionary Algorithm (InMaOEA)

**Encoding / Individual Design:** Integer-based task-to-VM assignment representation with interval parameters for MIPS and bandwidth; each individual encodes task–resource mapping

**Operators:** Interval Credibility Strategy for dominance ranking and Interval Congestion Distance based on Hypervolume and Overlap

Degree for diversity preservation; Binary Crossover Operator (BCO,  $p_c = 1$ ) and Polynomial Mutation Operator (PMO,  $p_m = 1/D$ )

**Comparison Methods:** NSGA-III, MaOEA-CSS, ISDE, MaOEA-SIN, and NSGA-III-GBFE under identical experimental conditions

**Key Findings:** InMaOEA achieved superior convergence and diversity under uncertainty. It outperformed five leading many-objective algorithms with the lowest IGD (0.097) and strong coverage (0.391). The approach effectively balanced four competing objectives—task makespan, scheduling cost, load variance, and completion rate—under stochastic network and computational uncertainty. Experimental results validated its robustness and computational efficiency for large-scale, uncertain cloud scheduling.

**Citation Count (to date):** 110 (Google Scholar, October 2025)

8. Self-Adaptive Multi-Objective Evolutionary Algorithm for Flexible Job Shop Scheduling with Fuzzy Processing Time

**Authors/Year:** Rui Li, Wenyin Gong, Chao Lu (2022)

**Problem Type:** Multi-objective flexible job shop scheduling problem with fuzzy processing times (MOFFJSP); simultaneous minimization of makespan ( $C_{max}$ ) and total machine workload (TWL) under uncertainty

**Dataset / Instances:** 23 benchmark FFJSP instances across Lei01–Lei02, Remanu, and FMk datasets; fuzzy processing times modeled as triangular fuzzy numbers; multiple scales with  $n = 5$ –20 jobs and  $m = 4$ –15 machines

**Evaluation Method / Metrics:** Hypervolume (HV), Generational Distance (GD), and Spread metrics; Friedman rank and Wilcoxon signed-rank statistical tests at  $\alpha = 0.05$ ; 30 independent runs per instance

**Algorithm Type:** Hybrid Self-Adaptive Multi-Objective Evolutionary Algorithm based on Decomposition (HPEA) — a hybrid MOEA/D variant with dynamic parameter adaptation, fuzzy modeling, and decomposition-based multi-objective balancing

**Encoding / Individual Design:** Two-vector encoding — one for operation sequence, one for machine assignment; decoding converts to fuzzy schedules using triangular fuzzy numbers (TFN)

**Operators:** Discrete POX crossover and mutation; hybrid initialization (MIX3 rule combining Random, LS, and GW heuristics); Variable Neighborhood Search (VNS) with five local search operators; parameter

adaptation strategy using success–failure memory learning

**Comparison Methods:** MOEA/D, MOEA/D-M2M, NSGA-III, MO-LR, Improved Artificial Immune System (IAIS); all using same fuzzy modeling and encoding schemes for fairness

**Key Findings:** HPEA achieved the best overall performance with highest HV and lowest GD across benchmarks, outperforming all competitors significantly ( $p < 0.05$ ). Parameter self-adaptation improved convergence speed and diversity simultaneously, achieving up to 18% higher HV and 45% lower GD than baseline MOEA/D. Demonstrated superior balance between exploration and exploitation in fuzzy multi-objective scheduling.

**Citation Count (to date):** 127 (Google Scholar, October 2025)

9. Multi-Objective Evolutionary Approach Based on K-Means Clustering for Home Health Care Routing and Scheduling Problem

**Authors/Year:** Mariem Belhor, Adnen El-Amraoui, Abderrazak Jemai, François Delmotte (2023)

**Problem Type:** Home Health Care Routing and Scheduling Problem (HHCSP) — multi-objective optimization integrating nurse routing, patient assignment, and time-window constraints

**Dataset / Instances:** Solomon VRPTW benchmark datasets with 25–100 patients; service times, travel times, and nurse working hours derived from real home healthcare logistics

**Evaluation Method / Metrics:** Pareto-based performance analysis using Hypervolume (HV), Generational Distance (GD), and Spread indicators; 30 independent runs; Wilcoxon signed-rank tests for significance

**Algorithm Type:** Hybrid K-Means clustering with Multi-Objective Evolutionary Algorithms (MOEAs) — K-NSGA-II and K-SPEA2

**Encoding / Individual Design:** Integer permutation encoding representing sequence of patient visits; cluster-based decomposition divides patients into geographic groups using K-Means to improve search efficiency

**Operators:** Simulated Binary Crossover (SBX,  $p_c = 0.9$ ) and Polynomial Mutation (PM,  $p_m = 0.1$ ); elitism and crowding-distance preservation; clustering dynamically updated after each generation to rebalance solution diversity

**Comparison Methods:** Standard NSGA-II, SPEA2, and lexicographic single-objective optimization approaches

**Key Findings:** Hybrid K-MOEA (K-NSGA-II and K-SPEA2) achieved superior convergence and diversity over standalone MOEAs. K-clustering reduced computational time and improved Pareto front coverage, producing solutions up to 15% shorter in total travel time and 20% lower in tardiness. Demonstrated effective scalability and adaptability for dynamic home healthcare routing.

**Citation Count (to date):** 59 (Google Scholar, October 2025)

10. Development of a New Personalized Staff-Scheduling Method with a Work–Life Balance Perspective: Case of a Hospital

**Authors/Year:** Halil İbrahim Koruca, Murat Serdar Emek, Esra Gülmez (2023)

**Problem Type:** Personalized multi-objective staff scheduling problem integrating work–life balance, employee preferences, and hospital demand

**Dataset / Instances:** Real-world dataset from Süleyman Demirel University Hospital (Internal Medicine Department) with 12 physicians over weekly/monthly scheduling horizons

**Evaluation Method / Metrics:** Comparison of three scheduling methods across five real-world working scenarios; metrics include percentage of staff preferences satisfied, fairness, and deviation from hospital demand satisfaction

**Algorithm Type:** Hybrid heuristic framework with three approaches — Seniority Score Priority Assignment Method (SPM), Balanced Fair Assignment Method (BFM), and Genetic Algorithm Method (GAM)

**Encoding / Individual Design:** Staff preferences (working hours, starting times, and departments) stored as database records; GA individuals encode assignments matching personnel requests to hospital constraints

**Operators:** Standard GA selection, crossover, and mutation with variable population size (10 chromosomes), 100% crossover, and 5% mutation rate; BFM balances satisfaction scores; SPM prioritizes seniority

**Comparison Methods:** SPM (seniority-based heuristic) and BFM (fairness-oriented heuristic) compared against GA (optimization-based) within the developed WLB-Tool software

**Key Findings:** The proposed WLB-Tool achieved up to 80% preference satisfaction across scenarios while maintaining hospital staffing needs. GA achieved the most balanced results, BFM offered equitable

preference coverage, and SPM favored senior personnel. Demonstrated that flexible and preference-driven scheduling substantially improves job satisfaction and fairness without increasing cost.

**Citation Count (to date):** 23 (Google Scholar, 2025)

#### 11. Meta-heuristic Approaches for the University Course Timetabling Problem

**Authors/Year:** Sina Abdipoor, Razali Yaakob, Say Leng Goh, Salwani Abdullah (2023)

**Problem Type:** Systematic review and classification of meta-heuristic and hybrid meta-heuristic methods for the University Course Timetabling Problem (UCTP)

**Dataset / Instances:** Comprehensive analysis of benchmark datasets (ITC2002, ITC2007, ITC2019, Socha, Hard) and real-world datasets from multiple universities worldwide

**Evaluation Method / Metrics:** Comparative performance analysis on feasibility, constraint satisfaction, and computational efficiency; assessed solution quality using soft and hard constraint violation counts, hypervolume, and generational distance

**Algorithm Type:** Review of Simulated Annealing (SA), Tabu Search (TS), Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Cuckoo Search (CS), Biogeography-Based Optimization (BBO), and Hybrid Meta-Heuristics (collaborative and integrative)

**Encoding / Individual Design:** Various encodings reviewed — direct timetable representation, matrix-based room-time-course structures, and island model parallel GA architectures

**Operators:** Summary of SA-based neighborhood composition, GA crossover–mutation strategies, PSO velocity–position updates, and hybrid operator integrations for constraint relaxation

**Comparison Methods:** Benchmark comparisons against ITC competition winners and prior hybrid approaches (e.g., TSSP-ILS, SAIRL, and HGALTS)

**Key Findings:** Simulated Annealing and hybrid SA–TS frameworks remain state-of-the-art for benchmark UCTP instances, while GA-based and PSO-based hybrids excel on real-world datasets. Hybrid meta-heuristics (collaborative/integrative) outperform single-solution methods by balancing exploration and exploitation. Identified future challenges include adaptive parameter tuning, multi-objective formula-

tions, and robustness under disruptions.

**Citation Count (to date):** 40 (Google Scholar, 2025)

12. Genetic Algorithm Based Probabilistic Model for Agile Project Success in Global Software Development

**Authors/Year:** Mohammad Shameem, Mohammad Nadeem, Abu Taha Zamani (2023)

**Problem Type:** Agile project success prediction in Global Software Development (GSD)

**Dataset / Instances:** Survey data from 103 agile and GSD practitioners across global organizations

**Evaluation Method / Metrics:** Success probability, cost optimization, efficacy function  $E = \text{Prob} - \text{norm}(C)$ ; statistical validation via t-test ( $p = 0.001$ )

**Algorithm Type:** Genetic Algorithm (GA) combined with Naive Bayes and Logistic Regression classifiers

**Encoding / Individual Design:** Chromosome represents 8 agile project features (e.g., communication, customer involvement, iteration length); scales from 1–5 per feature

**Operators:** Roulette-wheel selection, single-point crossover (0.8), random mutation (0.1); 100 generations, population = 50

**Comparison Methods:** Naive Bayes vs Logistic Regression predictive models integrated with GA

**Key Findings:** GA improved project success prediction to 99.9% (NBC) and 99.1% (LR); identified communication, infrastructure, and customer involvement as top success factors; validated cost-effective feature prioritization

**Citation Count (to date):** 27 (Google Scholar, 2025)

13. An Optimized Case-Based Software Project Effort Estimation Using Genetic Algorithm

**Authors/Year:** Shaima Hameed, Yousef Elsheikh, Mohammad Azzeh (2023)

**Problem Type:** Software Project Effort Estimation (SEE) using Case-Based Reasoning (CBR) optimized by Genetic Algorithm (GA)

**Dataset / Instances:** Seven public benchmark datasets from the PROMISE repository — Albrecht, Kemerer, NASA, Desharnais, China, Maxwell, and Telecom — containing 15–499 project instances each

**Evaluation Method / Metrics:** Mean Absolute Error (MAE), Mean Balanced Relative Error (MBRE), and Mean Inverted Balanced Relative Error (MIBRE); statistical Wilcoxon signed-rank tests and Win–Tie–Lose analysis across all datasets

**Algorithm Type:** Genetic Algorithm (GA)-based parameter optimization of Case-Based Reasoning (CBR)

**Encoding / Individual Design:** Chromosome encodes CBR parameters — number of nearest neighbors ( $k$ ) and feature weights ( $w_j \in [0, 1]$ ) — optimized to minimize prediction error

**Operators:** Selection, crossover, and mutation on real-valued chromosomes; iterative replacement until convergence; fitness function based on effort estimation error

**Comparison Methods:** Classical CBR with fixed  $k$  values (1–5) across all datasets

**Key Findings:** The CBR–GA model outperformed classical CBR across five of seven datasets (Albrecht, Maxwell, Kemerer, China, and Desharnais), reducing MAE by up to 65% and MBRE by up to 80%. The GA effectively tuned feature weights and nearest-neighbor counts, yielding consistent accuracy improvements, particularly for medium-to-large datasets. The approach was less effective for small datasets but demonstrated robustness and statistical superiority via Wilcoxon testing.

**Citation Count (to date):** 71 (Google Scholar, October 2025)

14. Evolutionary Algorithms for Parameter Optimization—Thirty Years Later

**Authors/Year:** Thomas H. W. Bäck, Anna V. Kononova, Bas van Stein, Hao Wang, Kirill A. Antonov, Roman T. Kalkreuth, Jacob de Nobel, Diederick Vermetten, Roy de Winter, Furong Ye (2023)

**Problem Type:** Survey and retrospective review of 30 years of research in parameter optimization using evolutionary algorithms (1993–2023)

**Dataset / Instances:** Not empirical — analytical and literature-based synthesis over continuous, multimodal, multiobjective, and constrained optimization problems

**Evaluation Method / Metrics:** Qualitative and conceptual evaluation of algorithmic progress, benchmarking practices, and methodological evolution

**Algorithm Type:** Evolutionary Algorithms (EAs) and their subfields:

Evolution Strategies (CMA-ES and its derivatives), Genetic Algorithms, Evolutionary Programming, Particle Swarm Optimization, Differential Evolution, Surrogate-Assisted EAs, and Automated Algorithm Design  
**Encoding / Individual Design:** Generalized framework unifying genotypic and phenotypic representations in continuous and discrete search spaces

**Operators:** Mutation, crossover, recombination, and adaptive parameter control (CMA, CSA, 1/5-rule, dynamic step-size adaptation)

**Comparison Methods:** Historical algorithmic baselines (Bäck & Schwefel, 1993) vs. contemporary methods (CMA-ES, MAP-Elites, NSGA-II, GOMEA, AutoML-based EAs)

**Key Findings:** Highlights key breakthroughs such as CMA-ES, parameter self-adaptation, multimodal and quality-diversity search, surrogate-assisted optimization, and automated algorithm design. Emphasizes the need for rigorous benchmarking and cautions against proliferation of metaphor-based algorithms. Suggests “fewer but better-tested” algorithms as a guiding principle for the next generation of EA research.

**Citation Count (to date):** 50, 25 (Google Scholar, 2025)

15. An Effective Collaboration Evolutionary Algorithm for Multi-Robot Task Allocation and Scheduling in a Smart Farm

**Authors/Year:** Hengwei Guo, Zhonghua Miao, J.C. Ji, Quanke Pan (2024)

**Problem Type:** Multi-Robot Task Allocation and Scheduling (MR-TAS) in smart farms; optimization of robot coordination for weeding operations under Agriculture 4.0 context

**Dataset / Instances:** Synthetic smart-farm datasets based on Shouguang greenhouse layouts; small- and large-scale test cases with  $n = 2-7$  robots and  $m = 45-110$  tasks; task times drawn uniformly from  $[10, 100]$

**Evaluation Method / Metrics:** Relative Percentage Increase (RPI), completion time ( $C_{max}$ ), and ANOVA/Friedman statistical tests; Wilcoxon signed-rank tests for pairwise comparison with baselines

**Algorithm Type:** Collaborative Discrete Artificial Bee Colony (CD-ABC) — a hybrid swarm intelligence algorithm integrating dynamic neighborhood search and multi-agent cooperation

**Encoding / Individual Design:** 2D representation of robots (rows) and their allocated tasks (columns); solution structure encodes both



task allocation and sequencing

**Operators:** Novel *Collaborative Following Bee Strategy (CFS)* for escaping local optima; problem-oriented heuristics (Heu\_Dominant, Heu\_Time) for initialization; single- and multi-robot insertion/exchange neighborhoods; dynamic neighborhood list and critical-robot exploitation mechanism

**Comparison Methods:** Simulated Annealing (SA), Quantum ABC (QABC), Ant Colony System (ACS), Improved ABC (IABC), Discrete ABC (DABC), Multi-objective DABC (MDABC), Teaching-Learning-Based Optimization (TLBO)

**Key Findings:** CDABC outperformed all state-of-the-art methods across 270 benchmark instances, achieving the lowest average  $C_{max}$  and RPI; collaborative strategies improved convergence and robustness; yielded up to 18% makespan reduction versus DABC and maintained stability across scaling scenarios. Statistical tests (ANOVA, Wilcoxon, Friedman) confirmed significance at  $\alpha = 0.05$ .

**Citation Count (to date):** 62 (Google Scholar, October 2025)

16. Reinforcement Learning-assisted Evolutionary Algorithm: A Survey and Research Opportunities

**Authors/Year:** Yanjie Song, Yutong Wu, Yangyang Guo, Ran Yan, P.N. Suganthan, Yue Zhang, Witold Pedrycz, Swagatam Das, Rammohan Mallipeddi, Oladayo Solomon Ajani, Qiang Feng (2024, arXiv preprint)

**Problem Type:** Comprehensive survey on integrating Reinforcement Learning (RL) with Evolutionary Algorithms (EAs) for complex optimization, including scheduling, routing, and production planning

**Dataset / Instances:** Aggregated review of over 120 RL-EA studies; includes benchmarks from continuous, combinatorial, and scheduling optimization problems (2012–2023)

**Evaluation Method / Metrics:** Qualitative synthesis and taxonomy-based comparison (integration method, RL-assisted strategy, attribute settings, and application domain)

**Algorithm Type:** Reinforcement Learning-assisted Evolutionary Algorithms (RL-EA)

**Encoding / Individual Design:** Varies by category — chromosomes, policies, or population states represented as agent-environment interactions

**Operators:** RL modules guide crossover, mutation, sub-population,

and parameter adaptation; Q-learning, DQN, PPO, and Actor–Critic frameworks dominate

**Comparison Methods:** Traditional EAs (GA, DE, PSO, MOEA/D) versus RL-EA hybrids; evaluated across single-objective, multi-objective, and multi-modal optimization problems

**Key Findings:** RL-EA methods outperform standalone EAs in convergence, adaptability, and scalability. RL enhances exploration–exploitation balance, operator selection, and parameter tuning; identified promising future directions in deep RL integration, hyper-heuristics, and scheduling applications

**Citation Count (to date):** 78 (Google Scholar, 2025)

17. A Hyper-Heuristic Evolutionary Approach for Multi-Objective Dynamic Project Scheduling under Uncertainty

**Authors/Year:** Peng Yang, Tao Zhang, Zhen Liu, Chao Li (2024)

**Problem Type:** Multi-objective Dynamic Project Scheduling Problem (MODPSP) — addressing uncertain activity durations, changing resource availability, and multi-criteria optimization of project time, cost, and robustness

**Dataset / Instances:** Stochastic RCPSP instances generated from PSPLIB benchmarks; dynamic changes introduced at random time intervals with up to 30% variation in activity duration and 20% in resource capacity

**Evaluation Method / Metrics:** Hypervolume (HV), Generational Distance (GD), and Spacing (SP) metrics; tested on 30 dynamic scenarios with five change events each; statistical validation using Wilcoxon tests ( $\alpha = 0.05$ )

**Algorithm Type:** Hyper-Heuristic Multi-Objective Evolutionary Algorithm (HH-MOEA) combining high-level selection of low-level heuristics with adaptive reinforcement learning control

**Encoding / Individual Design:** Activity list encoding ensuring precedence feasibility; individuals represent dynamic project schedules updated after each change event

**Operators:** Pool of eight low-level heuristics (e.g., swap, insertion, mode change, random restart) governed by reinforcement learning-based selection policy; hyper-heuristic layer dynamically adjusts heuristic probabilities based on past performance

**Comparison Methods:** NSGA-II, MOEA/D, SPEA2, and adaptive GA-based rescheduling algorithms under identical dynamic event

streams

**Key Findings:** HH-MOEA consistently outperformed benchmark algorithms, achieving up to 22% higher HV and 19% lower GD across all scenarios. Demonstrated superior robustness to uncertainty and adaptability to environmental changes. The reinforcement learning-driven heuristic controller improved responsiveness to dynamic variations while maintaining Pareto diversity.

**Citation Count (to date):** 97 (Google Scholar, October 2025)

18. A Knowledge-Guided Bi-Population Evolutionary Algorithm for Energy-Efficient Scheduling of Distributed Flexible Job Shop Problem  
**Authors/Year:** Fei Yu, Chao Lu, Jiajun Zhou, Lvjiang Yin, Kaipu Wang (2024)

**Problem Type:** Energy-Efficient Distributed Flexible Job Shop Scheduling Problem (EEDFJSP) under speed-scaling framework; objectives: minimizing makespan and total energy consumption (TEC)

**Dataset / Instances:** 39 benchmark instances of distributed FJSP extended with energy and speed variables; experiments conducted on diverse distributed manufacturing configurations with multiple factories and speed settings

**Evaluation Method / Metrics:** Pareto-based performance evaluation using convergence and diversity metrics; comparisons over 7 state-of-the-art algorithms; Wilcoxon tests for significance

**Algorithm Type:** Knowledge-Guided Bi-Population Evolutionary Algorithm (KBEA) — a multi-objective evolutionary framework incorporating problem-specific knowledge and energy-saving mechanisms

**Encoding / Individual Design:** Four-layer encoding: factory assignment (FA), operation sequence (OS), machine assignment (MA), and speed assignment (SA); supports variable machine speeds and distributed factories

**Operators:** Five adaptive evolutionary operators per population; knowledge-guided local search for exploitation; adaptive energy-saving strategy based on speed reduction of non-critical operations; adaptive operator selection via success-failure memory

**Comparison Methods:** Compared against NSGA-II, EDA, MA, GWO, TS-KEA, and other DFJSP-focused metaheuristics

**Key Findings:** KBEA significantly outperformed all baseline algorithms in both convergence and energy efficiency; average TEC reduction up to 17% and makespan improvement by 11%. The

bi-population structure with knowledge guidance effectively balanced exploration and exploitation for distributed green manufacturing.

**Citation Count (to date):** 77 (Google Scholar, October 2025)

19. Time–Cost Trade-Off Optimization Model for Retrofitting Planning Projects Using MOGA

**Authors/Year:** Anjali S. Patil, Aditya Kumar Agarwal, Kamal Sharma, Manoj Kumar Trivedi (2024)

**Problem Type:** Multi-objective Time–Cost Trade-Off Optimization (TCTO) model tailored for retrofitting projects in urban Indian contexts

**Dataset / Instances:** Real-world case study from Gwalior, India — comprehensive retrofitting project covering seven aspects (structural, electrical, plumbing, HVAC, fire protection, insulation, finishing), each with three possible execution options and associated time–cost parameters

**Evaluation Method / Metrics:** Performance evaluated using thirteen Pareto-based indicators: Unique Number of Pareto Solutions (UNPS), Spacing Metric (SM), Generational Distance (GD), Spread (Sp), Maximum Spread (MS), Mean Ideal Distance (MID), Spread of Non-Dominant Solutions (SNS), Quality (QM), Diversification (DM), Non-Uniformity (NPF), Hypervolume (HV), Epsilon (E), and Computational Time (CT)

**Algorithm Type:** Multi-Objective Genetic Algorithm (MOGA)

**Encoding / Individual Design:** Real-coded chromosome where each gene represents a selected option for one retrofitting aspect (e.g., structural reinforcement, electrical upgrades, HVAC systems); each option encoded as an integer gene (1–3)

**Operators:** Simulated Binary Crossover (SBX,  $p_c = 1$ ) and Polynomial Mutation (PM,  $p_m = 0.1$ ); population size = 100, 150 generations; elitism and non-dominated sorting with crowding distance for Pareto maintenance

**Comparison Methods:** Multi-Objective Particle Swarm Optimization (MOPSO), Multi-Objective Ant Colony Optimization (MOACO), and Multi-Objective Teaching–Learning-Based Optimization (MOTLBO)

**Key Findings:** MOGA achieved the most diverse and convergent Pareto-optimal solutions among all algorithms, generating 12 high-quality trade-off solutions. Retrofitting time ranged from 106–140 days and cost from INR 8.09–11.82 million. The MOGA outperformed

MOPSO, MOACO, and MOTLBO in 12 of 13 indicators, achieving the highest Hypervolume ( $HV = 0.84$ ) and lowest Generational Distance ( $GD = 0.45$ ). Demonstrated practical viability for complex multi-aspect retrofitting optimization and informed sustainable urban development.  
**Citation Count (to date):** 47 (Google Scholar, October 2025)

20. Development of Time–Cost Trade-Off Optimization Model for Construction Projects with MOPSO Technique  
**Authors/Year:** Aditya Kumar Agarwal, Shyamveer Singh Chauhan, Kamal Sharma, Krushna Chandra Sethi (2024)  
**Problem Type:** Bi-objective optimization of project duration and cost in multi-mode construction project scheduling (Time–Cost Trade-Off, TCT)  
**Dataset / Instances:** Real case study — three-story building construction project in Delhi, India; 19 activities, each with three alternative execution modes; total of 319 feasible scheduling combinations  
**Evaluation Method / Metrics:** Pareto front generation and comparison using Project Time (PT) and Project Cost (PC) objectives; algorithm efficiency evaluated by diversity and convergence of Pareto-optimal fronts versus existing meta-heuristics  
**Algorithm Type:** Multi-Objective Particle Swarm Optimization (MOPSO) with integrated Non-Dominated Sorting (NDS) mechanism  
**Encoding / Individual Design:** Chromosomal representation of activities and their execution modes; each particle encodes alternative mode selections for all project activities  
**Operators:** Velocity–position update with inertia weighting, cognitive/social acceleration coefficients ( $c_1, c_2$ ); Non-Dominated Sorting (NDS) to maintain Pareto fronts; random selection to prevent premature convergence  
**Comparison Methods:** Multi-Objective Genetic Algorithm (MOGA), Multi-Objective Teaching–Learning-Based Optimization (MOTLBO), and Multi-Objective Ant Colony Optimization (MOACO)  
**Key Findings:** The MOPSO model generated ten distinct Pareto-optimal solutions balancing project duration (124–148 days) and cost (12.29–13.65 million INR). Achieved 4–6% improvement in project time and up to 6.5% cost reduction versus competing algorithms. Demonstrated superior convergence and diversity of Pareto fronts, proving effectiveness in complex real-world scheduling environments.  
**Citation Count (to date):** 56 (Google Scholar, October 2025)

21. Gradual Optimization of University Course Scheduling Problem Using Genetic Algorithm and Dynamic Programming

**Authors/Year:** Xu Han, Dian Wang (2025)

**Problem Type:** University Course Scheduling Problem (UCSP) — a large-scale combinatorial optimization challenge involving both independent and joint course scheduling under hard and soft constraints

**Dataset / Instances:** Real-world scheduling data from Beijing Forestry University — 520 course scheduling tasks including combined (multi-class) and independent courses

**Evaluation Method / Metrics:** Compared average and best fitness values, classroom utilization rate, number of classrooms used, and week-to-week scheduling stability against GA, PSO, ACO, and PSM; analyzed similarity of weekly timetables

**Algorithm Type:** Hybrid metaheuristic combining Genetic Algorithm (GA) and Dynamic Programming (DP), termed POGA-DP (Progressively Optimized Genetic Algorithm–Dynamic Programming)

**Encoding / Individual Design:** Chromosome encodes course–teacher–classroom–timeslot tuples across 25 weekly time slots; allows handling of combined courses by multi-class encoding

**Operators:** Judgment-based swap operation for valid crossover, forced mutation with repair mechanism to prevent infeasible offspring, and greedy DP-based classroom assignment for resource utilization

**Comparison Methods:** Benchmarked against traditional GA, Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Producer–Scrounger Method (PSM) under identical instance sets

**Key Findings:** POGA-DP improved schedule quality by 46.99% and reduced classroom use by 29.27% compared to traditional GA; achieved nearly 99% classroom utilization and maintained high stability across weekly timetables. Outperformed all comparators in complex instances with joint scheduling. Demonstrated strong convergence and robustness in large-scale real-world scenarios.

**Citation Count (to date):** 7 (Google Scholar, October 2025)

22. Optimizing Trade-off Between Time, Cost, and Carbon Emissions in Construction Using NSGA-III: An Integrated Approach for Sustainable Development

**Authors/Year:** Amir Prasad Behera, Mayank Chauhan, Gaurav Shrivastava, Prachi Singh, Jyoti Shukla, Krushna Chandra Sethi (2025)

**Problem Type:** Multi-objective optimization for sustainable construction management — simultaneous minimization of project time, cost, and carbon emissions (Time–Cost–Carbon Emission Trade-Off, TCCET)

**Dataset / Instances:** Real-world case study including 13 construction activities (e.g., groundwork, excavation, formwork, finishing) with multiple execution modes; quantitative data on time, cost, and CO<sub>2</sub> emissions for each mode

**Evaluation Method / Metrics:** Quantitative comparison across Pareto fronts using performance metrics — Unique Number of Pareto Solutions (UNPS), Spacing Metric (SM), Generational Distance (GD), Spread (Sp), Mean Ideal Distance (MID), and Hypervolume (HV); sensitivity analysis on  $\pm 10\text{--}20\%$  parameter changes

**Algorithm Type:** Non-dominated Sorting Genetic Algorithm III (NSGA-III)

**Encoding / Individual Design:** Integer chromosome representation of construction activities and their chosen execution modes (1–3); each chromosome encodes mode selection influencing time, cost, and emissions

**Operators:** Simulated Binary Crossover (SBX,  $p_c = 0.9$ ) and Polynomial Mutation (PM,  $p_m = 0.1$ ); reference-point–based non-dominated sorting and crowding-distance diversity preservation

**Comparison Methods:** Compared with NSGA-II, MOPSO, and MOACO on identical case study data; evaluated Pareto front diversity, convergence, and solution quality

**Key Findings:** NSGA-III achieved the highest Hypervolume (HV = 0.800), best convergence (GD = 0.060), and most uniform Pareto distribution (SM = 0.030), outperforming NSGA-II (HV = 0.740). Enabled balanced trade-offs among time, cost, and CO<sub>2</sub> emissions, reducing emissions up to 15% without major cost increase. Validated through sensitivity tests demonstrating robustness and scalability for sustainable construction planning.

**Citation Count (to date):** 31 (Google Scholar, October 2025)

## Summary

In recent years (2021–2025), evolutionary algorithms (EAs) have remained one of the most effective metaheuristic approaches for solving complex

scheduling and resource allocation problems across domains such as education, healthcare, software engineering, construction, and manufacturing. These problems are typically NP-hard and involve the simultaneous optimization of conflicting objectives such as time, cost, energy, workload balance, or environmental impact.

Recent studies have focused on various applications. In education, Genetic Algorithms (GA) and hybrid metaheuristics have been applied to university course timetabling problems using real-world datasets from Chitkara University (India) and Beijing Forestry University (China), achieving significant reductions in scheduling conflicts and improvements in resource utilization. In healthcare, hybrid GA-based nurse and staff scheduling models incorporating fatigue and work-life balance factors have demonstrated enhanced fairness and operational efficiency using hospital data from Esfahan and Süleyman Demirel University. In construction management, multi-objective EAs such as NSGA-II, NSGA-III, Differential Evolution (DE), and Multi-Objective Particle Swarm Optimization (MOPSO) have been employed to optimize the trade-offs between project time, cost, and carbon emissions, often validated on real Indian infrastructure projects. Industrial research has also advanced distributed and energy-efficient job shop scheduling using knowledge-guided and self-adaptive MOEAs on public benchmark datasets such as PSPLIB, Lei, and FMk. In cloud and smart-agriculture contexts, Interval and Hyper-Heuristic EAs have been used to handle uncertainty and multi-robot task scheduling efficiently.

The most frequently used algorithms include traditional GA and its hybrids, Differential Evolution (DE), NSGA-II/III, and Memetic or Reinforcement-Learning-assisted Evolutionary Algorithms (RL-EA). Encoding schemes vary from matrix-based or integer-vector representations to multi-layer or fuzzy encodings that ensure feasibility and interpretability. Typical datasets include PSPLIB benchmarks, institutional timetables, real hospital schedules, cloud-simulated data (CloudSim), and case-specific industrial or construction projects. Evaluation commonly relies on Pareto-based indicators such as Hypervolume (HV), Generational Distance (GD), Spread (SP), and Inverted Generational Distance (IGD).

Although evolutionary algorithms have achieved notable success in various scheduling applications, research in this area continues to evolve. Many studies still focus on static or simplified scenarios, while real-world scheduling environments are often dynamic, uncertain, and multi-dimensional. There is a growing need to improve the adaptability and generalization of these algorithms so that they can perform reliably under changing constraints and incomplete information. Furthermore, the interpretability of obtained Pareto-optimal solutions remains a challenge, as decision-makers often re-



quire transparent explanations of trade-offs among objectives such as time, cost, and resource usage. Automatic parameter control and self-adaptive mechanisms also represent ongoing concerns, as the effectiveness of evolutionary approaches often depends on careful tuning of algorithmic parameters.

In parallel, the increasing use of learning-based adaptation, hybrid optimization frameworks, and sustainability-oriented objectives indicates a gradual shift toward more intelligent and responsible evolutionary scheduling systems. Continued research is therefore essential to integrate these developments into practical, efficient, and interpretable solutions for complex real-world scheduling problems.

Overall, evolutionary computation continues to provide a flexible and powerful framework for real-world scheduling optimization, with ongoing research focusing on adaptability, interpretability, and sustainable optimization under complex constraints.