Hospital Admission and Scheduling Using Stochastic Optimization

Course: Numerical Optimization for Large scale problems and stochastic Optimization

Abedal Salam Al Ashi Abou Shoushe
Politecnico di Torino
s336648

Metaheuristic Used: Tabu Search

Code Repository

Date: January 20, 2025

Abstract. This project addresses the challenge of hospital scheduling—assigning patients to rooms (with capacity and gender restrictions), operating theaters, and nurses (with varying skill levels). A Tabu Search metaheuristic is used to iteratively refine a randomly generated initial solution, minimizing constraint violations (e.g., overcapacity, uncovered shifts, underqualified nurses). The final schedules are then checked against an external validator, which reports both feasibility and overall cost, ensuring that the proposed solution aligns with key hospital constraints.

1 Introduction

Context. In a hospital environment, scheduling patients, rooms, nurses, and operating theaters is inherently complex. Each patient requires a room (with limited capacity and gender constraints), may need surgical intervention in an operating theater, and must be cared for by nurses with adequate skill levels. Additionally, mandatory patients must always be scheduled, while elective patients may be postponed if resources are insufficient.

Scope. This implementation focuses on assigning both mandatory and elective patients to multiple rooms over a fixed planning horizon (for example, 14 days). It also assigns nurses—who each have specific skill levels and scheduled working shifts—to rooms so that patient care is covered. While the approach can be extended to additional complexities, the current work handles a single hospital context with multiple rooms, a predefined set of patients, operating theaters, and nurse schedules.

2 Solution Approach: Tabu Search

2.1 Why Tabu Search?

Tabu Search is a metaheuristic algorithm well-suited for solving combinatorial optimization problems like hospital scheduling. It effectively explores the solution space while avoiding local minima by using a tabu list to prevent revisiting recently explored solutions. Additionally, it dynamically adapts to complex constraints such as room capacity, nurse skill levels, and operating theater availability, making it an excellent choice for this problem.

2.2 Initial Solution Generation

The algorithm begins by generating a random initial solution. Patients are assigned to rooms, ensuring that the admission day is on or after their surgery release day while avoiding immediate room capacity violations. Elective patients may be skipped during this initial generation if resources are insufficient, helping to prioritize mandatory patients. Operating theaters and surgeons are randomly assigned for patients requiring surgeries. Nurses are also randomly assigned to rooms and shifts, without explicitly verifying feasibility for skill levels or shift coverage at this stage. This initial solution provides a baseline for the Tabu Search to iteratively improve upon.

2.3 Neighborhood Generation

To explore the solution space, a neighborhood is generated by applying small modifications to the current solution. These modifications include changing a patient's admission day within the feasible range (on or after the surgery release day), reassigning a patient to a different room while ensuring capacity and compatibility constraints are met, and reassigning the surgeon or operating theater to balance workloads and improve feasibility.

2.4 Cost Function

The cost function evaluates the quality of a solution by summing penalties for constraint violations:

- **Hard Constraint Penalties:** Extremely high penalties (e.g., thousands) are assigned to ensure violations such as room overcapacity, unscheduled mandatory patients, and gender mixing are avoided.
- **Soft Constraint Penalties:** Lower penalties are assigned for issues such as patient delays, nurse workload imbalance, and unscheduled elective patients.
- The total cost is used to rank neighbors and prioritize solutions that minimize both hard and soft violations.

This approach allows the Tabu Search to balance feasibility with solution quality, gradually improving the schedule while adhering to critical constraints.

3 Experiments & Results

3.1 Validator Output

- **Room Violations:**
  - Room r1:
    - Gender-mixed on days 4 and 5.
    - Overloaded by 2 on days 1, 2, and 3.
    - Overloaded by 1 on days 4 and 5.
  - Room r2:
    - Gender-mixed on days 1, 2, and 3.
    - Incompatible with patient p20.
    - Overloaded by 1 on multiple days.
  - Room r3:
    - Gender-mixed on days 11, 12, and 13.
    - Incompatible with patients p10 and p27.
  - Room r4:
    - Gender-mixed on days 2, 4, and 5.
    - Overloaded by 1 on day 4.

- **Patient Violations:** Patient p07 admitted at 11, after the last possible date (4).
- **Uncovered Shifts:**
  - Room r0: Uncovered in shifts 26 (day 8, night) and 35 (day 11, night).
  - Room r1: Uncovered in shifts 5 (day 1, night) and 11 (day 3, night).
  - Room r2: Uncovered in shifts 5, 13, 17, and 24 across various days.

Nurse Violations:

- Underqualified nurses for patients in various rooms and shifts. Examples include:
  - Nurse n10 underqualified for patient p05 in room r0, shift 7 (day 2, late).
  - Nurse n05 underqualified for patient p30 in room r0, shift 22 (day 7, late).

Excessive Workload:

- Nurse n00:
  - Excessive workload of 11 in shift 4 (day 1, late).
  - Excessive workload of 15 in shift 7 (day 2, late).
- Nurse n01: Excessive workload of 20 in shift 22 (day 7, late).

Patient Delays:

- Examples include:
  - Patient p06 delayed by 9 days.
  - Patient p07 delayed by 10 days.

Unscheduled Elective Patients: Elective patients p04 and p32 remain unscheduled.

3.2 Violations Table

Violation Type	**Count**
RoomGenderMix	10
PatientRoomCompatibility	3
SurgeonOvertime	0
OperatingTheaterOvertime	0
MandatoryUnscheduledPatients	0
AdmissionDay	1
RoomCapacity	9
NursePresence	0
UncoveredRoom	16
Total Violations	**39**

Table 1: Summary of Violations.

3.3 Cost Table

Cost Type	**Weight**	**Cost**	**Total (Weight × Cost)**
RoomAgeMix	5	10	50
RoomSkillLevel	5	137	685
ContinuityOfCare	1	282	282
ExcessiveNurseWorkload	10	538	5380
OpenOperatingTheater	30	15	450
SurgeonTransfer	10	7	70
PatientDelay	5	115	575
ElectiveUnscheduledPatients	150	2	300
Total Cost	-	-	**7792**

Table 2: Detailed Cost Breakdown of the Scheduling Solution.

4 Discussion and Conclusion

4.1 Discussion

The results of the Tabu Search algorithm on the `data/instances/i02.json` instance highlight significant improvements across multiple iterations. Starting from an initial cost of 4,243,165, the algorithm achieved a final cost of 7,258 after 20 iterations. This demonstrates the algorithm's ability to iteratively refine the solution by exploring the neighborhood space effectively.

The reduction in violations, particularly in *SurgeonOvertime* and *OperatingTheaterOvertime*, validates the neighborhood generation strategy. By incorporating checks for maximum surgery times and operating theater availability, these constraints were entirely resolved in the final solution. However, certain violations such as *RoomGenderMix*, *RoomCapacity*, and *UncoveredRoom* persisted. For instance, gender-mixed violations occurred across multiple rooms and days, with notable occurrences in rooms r1, r2, and r4. Additionally, uncovered rooms in shifts further impacted the quality of the solution.

The excessive workload assigned to nurses also remained a challenge, as evidenced by the numerous violations recorded across shifts and days. This indicates a need to refine the neighborhood generation to consider workload balance and improve nurse assignments. The delays in patient admissions, particularly for elective patients, contributed significantly to the cost, despite the effective scheduling of all mandatory patients. Elective patients p04 and p32 remained unscheduled, emphasizing the impact of resource constraints.

Conclusion. The Tabu Search algorithm successfully reduced the overall cost and addressed critical constraints, particularly in areas such as *SurgeonOvertime* and *OperatingTheaterOvertime*. The steady decrease in cost through iterations highlights the robustness of the neighborhood generation and selection strategies. However, the persistence of specific violations, such as *RoomGenderMix*, *RoomCapacity*, and *UncoveredRoom*, suggests opportunities for further improvement.

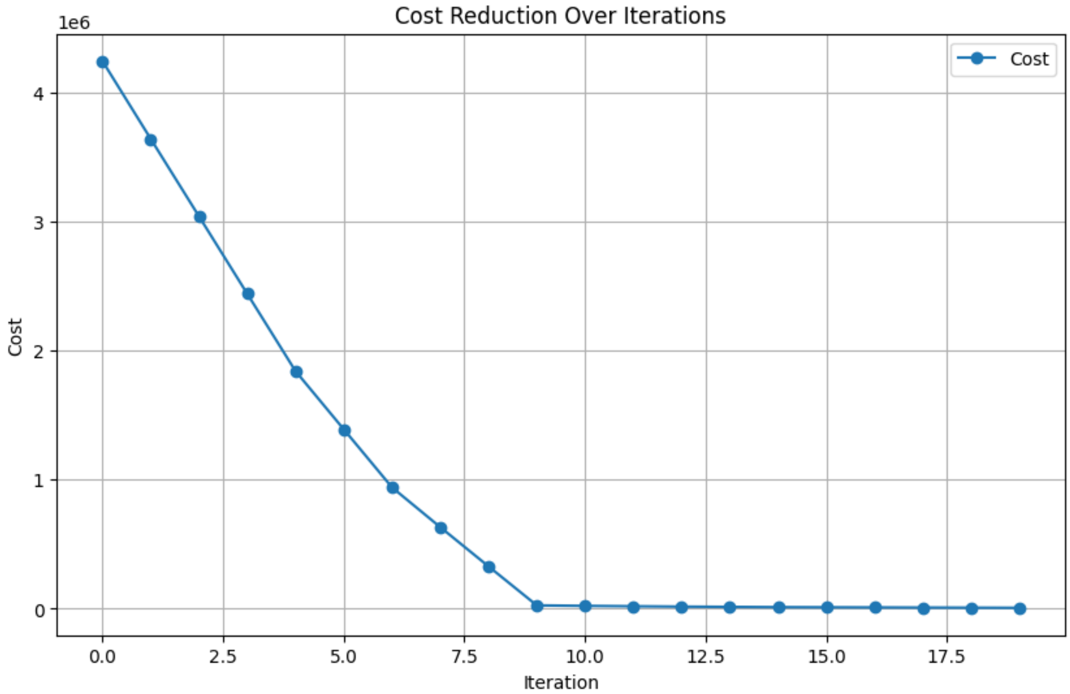

Figure 1: Cost reduction over 30 iterations of Tabu Search.

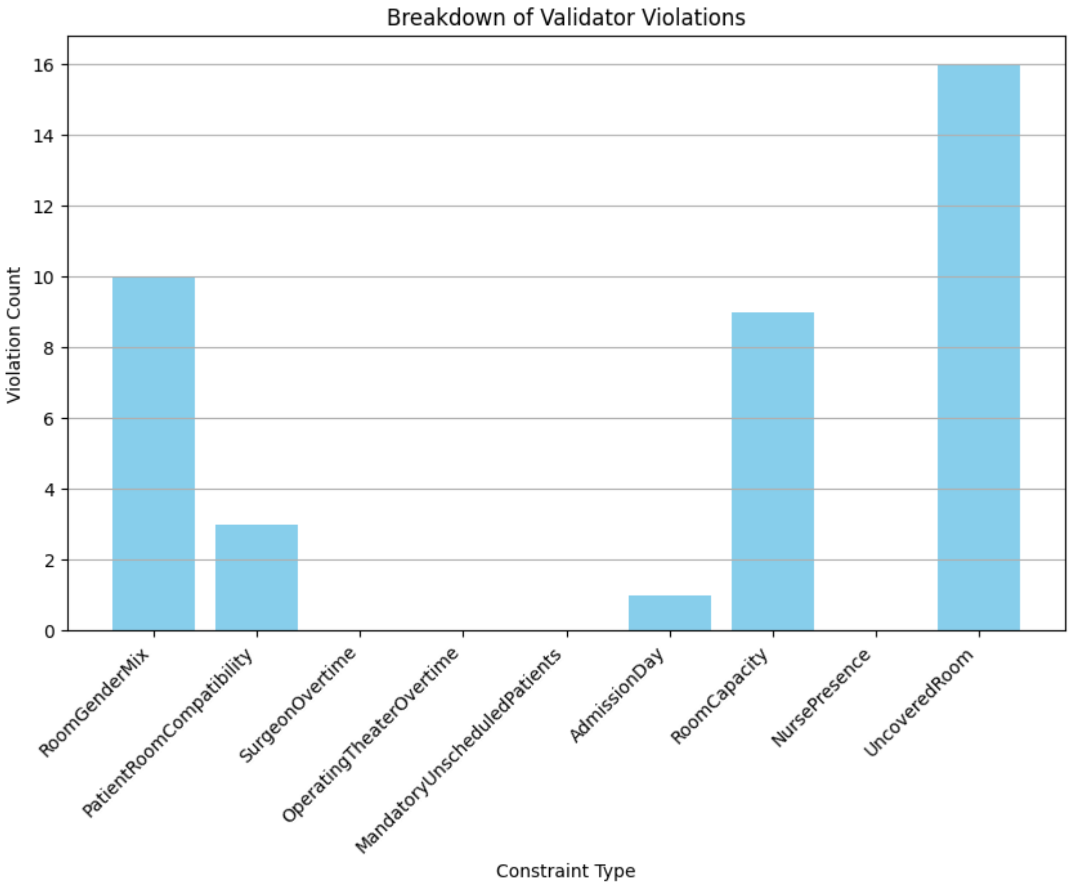

Figure 2: Summary of Violations.