Proceedings of the 8th Bangladesh International Conference on Industrial Engineering and Operations Management, Dhaka, Bangladesh, December 20-21, 2025

Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/BA08.20250310
Published: December 20, 2025

Optimizing Outpatient Appointment Schedules with Particle Swarm Optimization: A Deterministic Modeling Approach

Kazi Md. Tanvir Anzum and Md. Saiful Islam
Department of Industrial Engineering and Management
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
tanvir@iem.kuet.ac.bd and saifuliem@iem.kuet.ac.bd

Abstract

Outpatient appointment scheduling strongly affects patient waiting time, doctor utilization, and overall service quality. Many clinics, especially small and medium ones, cannot use complex simulation-based tools due to limited resources. There is a need for simple and practical methods that can improve scheduling without advanced software or heavy computation. This study aims to reduce total patient time in the system and minimize physician idle time using an easy-to-implement optimization approach. A deterministic scheduling model was developed and optimized using Particle Swarm Optimization (PSO), and its performance was compared to a Genetic Algorithm (GA) under equal computational conditions. PSO consistently produced better appointment schedules than GA, with lower total waiting time, reduced physician idle time, and better overall objective values. It also converged faster and showed more stable performance across multiple runs. Compared with standard booking rules, PSO reduced waiting time by 12-18% and idle time by 8-14% for typical daily loads of 20-40 patients. GA also improved scheduling outcomes but was slower and more variable. These results show that even a simple deterministic model can achieve meaningful improvements when paired with an effective optimization method. The findings demonstrate that clinics with limited computational capacity can still improve scheduling using simple optimization tools such as PSO. This work also provides a foundation for future extensions involving multi-stage, stochastic, or multi-objective scheduling problems.

Keywords

Outpatient appointment scheduling; Deterministic scheduling model; Particle Swarm Optimization (PSO); Genetic Algorithm (GA); Healthcare operations management

1. Introduction

The problem of long waiting periods in outpatient clinics remains one of the central issues most of the world faces (Naiker et al., 2018). It has been reported that the poor scheduling alone may also augment the average patient waiting time and can lead to up to 20-25 percent idle time by the physicians (Fetter & Thompson, 1966). These numbers illustrate an effective fact that sometimes it only takes little scheduling inefficiencies to cause big operational issues in high-volume healthcare settings. The scheduling of outpatient appointments is one of the key operation activities as it influences the patient flow and resource utilization directly and the quality of services delivered. Clinics have to book a high number of patients over a narrow period whilst having to handle fewer staff, unreliable patients' arrival time, and unequal service time of patients. Huge hospitals occasionally deal with such complexity with complex scheduling software (Ala & Chen, 2022). A large number of small and medium sized clinics continue to operate on simple rule-of-thumb-based approaches, like fixed intervals, block scheduling or first-come-first-served sequencing (Green et al., 2007). These are simple procedures to adopt but they tend to cause long queues, idle time, and frustrated patients.

Outpatient scheduling is a study subject that has been studied in many methods. Stochastic models were developed to address walk-ins, no-shows, and service time uncertainty (Wang et al., 2019). Simheuristics and discrete-event simulations are examples of simulation-optimization processes that can be used to represent complicated patient flows. Further improved metaheuristics, including as multi-objective genetic algorithms, NSGA-II, SPEA2, differential evolutions, and multi-objective PSO, have produced strong outcomes in highly specialized medical systems (de Alcantara et al., 2025). Although these studies are useful, the majority of these methods need substantial computations, operational data, or specialist software. Many clinics lack access to such modern methods, as well as the necessary funding and expertise. It creates a big research gap: there is little literature on simple, deterministic, and user-friendly scheduling models that yet result in considerable improvements. The majority of known options require advanced simulation tools or rely on stochastic data. As a result, smaller clinics are often left without suitable optimization solutions, although facing the same performance challenges as larger hospitals. Lightweight models capable of running on common platforms such as MATLAB are also necessary to assist better scheduling decisions (Al Jufout et al., 2024).

The current research will address this gap by investigating a simplified deterministic appointment scheduling model that can be used in small and medium-sized outpatient clinics. The study's goal is to assign patients to consultation slots in such a way that the overall patient time in the system, including waiting and service completion time, is minimized while yet allowing for adequate physician idle time. To address it, this study examines the efficiency of Particle Swarm Optimization (PSO), a simple and effective metaheuristic with a low computing load and ease of implementation. The comparative study is also made between PSO and Genetic Algorithm (GA) based on the same computational conditions to determine the relative performance. This paper explored that outpatient appointments can be greatly enhanced by a deterministic optimization strategy and the use of a powerful metaheuristic technique such as PSO and simulation is not necessary. The given model is a viable and readily available instrument to those clinics that aim to improve their performance without having sophisticated analytical infrastructure.

The rest of the paper will be structured in the following manner: Section 2 discusses the deterministic problem model. The PSO and GA algorithms are shown in Section 3. Section 4 gives results and comparisons in numerical form. Section 5 speaks about practical implications. Section 6 also concludes and proposes further extensions.

2. Literature Review

Outpatient appointment scheduling has been widely studied because it directly affects patient waiting time, doctor utilization, and service quality (Xie & Or, 2017). Early studies used simple deterministic models and rules-such as fixed-interval scheduling, individual-block scheduling, and modified Bailey's rule-to reduce waiting and balance idle time (Bailey, 1952). These works showed that small changes in appointment spacing can significantly improve system performance. As clinics became more complex, researchers began including uncertainty such as variable service times, walk-ins, and no-shows(Cayirli et al., 2012) . Queueing theory and Markov models helped describe these factors, but they became difficult to apply in realistic settings. This led to the rise of discrete-event simulation (DES), which allows detailed and realistic modeling of patient flows(X. Zhang, 2018). DES then became the main tool for evaluating outpatient scheduling policies. To find optimized schedules within simulation models, metaheuristics and simulation-optimization methods were introduced. Techniques like Genetic Algorithms, Simulated Annealing, Tabu Search, PSO, and Differential Evolution have been applied widely (Youssef et al., 2001). More advanced multi-objective algorithms-such as NSGA-II, SPEA2, MOPSO, and IBEA-help balance multiple goals like waiting time, idle time, throughput, and overtime. These studies show that metaheuristics often perform better than simple rule-based scheduling. Simulation based models however have high computations, repeated replications and special software that a small or medium clinic cannot afford. This has led to the growth of interest in deterministic models that are simple to use, and utilize an analytical formula to estimate waiting and idle times. These models are more comprehensible, quicker to calculate and are more appropriate where clinics have limited resources, but are less investigated.

2.1 Contributions of This Study

The following are the main contributions made by this study:

- Basic deterministic appointment booking model.
- Optimization using PSO in a faster and cheaper way. The deterministic model is run with a MATLAB-based PSO, which demonstrates how lightweight metaheuristics can effectively enhance the process of scheduling.
- Obvious comparison between PSO and GA. PSO and GA are evaluated at the same conditions to demonstrate the differences in convergence and stability, as well as quality of the solution to simple outpatient scheduling problems

3. Methodology

In this section, the methodological framework is outlined to develop and optimize a simplified deterministic outpatient appointment scheduling model. The methodology has two parts: (i) the model formulation, which constitutes the structure of the scheduling problem, and (ii) the optimization process whereby Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) will be used to obtain near-optimal schedules under the same conditions.

3.1 Mathematical Formulation

This subsection identifies the sets, the parameters, the decision variables and objective function that will be used to develop the deterministic single stage outpatient appointment scheduling model. This model takes into account only the booked patients with deterministic service times and aims to build up an appointment sequence and a schedule that balances the waiting time of the patients and idle time of the physicians within a given clinic session.

Sets and Indices

- $I = \{1, \ldots, N\}$: set of patients to be scheduled.
- $K = \{1, \ldots, N\}$: set of appointment positions in the schedule.

Each position $k \in K$corresponds to one appointment time in the clinic session.

Parameters

- $H > 0$: length of the clinic session (end of regular hours).
- $s_i > 0$: deterministic service time (consultation duration) of patient $i \in I$, based on historical averages.
- $\lambda \geq 0$: weighting parameter controlling the trade-off between patient waiting and physician idle time.

Optionally, the time grid can be defined by a fixed slot size $\Delta = H/N$, but the formulation below allows continuous appointment times.

Decision Variables

- $x_{ik} \in \{0,1\}$: equals 1 if patient $i$is assigned to position k; 0 otherwise.
- $z_k \geq 0$: planned start time (scheduled appointment time) of position k.
- $s_k \geq 0$: actual service start time at position k.
- $c_k \geq 0$: completion time at position k.
- $w_k \geq 0$: waiting time of the patient assigned to position k.
- $\mathrm{idle}_k \geq 0$: physician idle time before the patient at position k(for $k \geq 2$).

For each position k, exactly one patient is assigned, and each patient must appear in exactly one position.

Objective Function

The goal is to minimize a weighted sum of total patient waiting time and physician idle time:

$$\min\ F = \sum_{k=1}^{N} w_k + \lambda \sum_{k=2}^{N} \mathrm{idle}_k . \qquad (1)$$

In equation (1), The first term represents the total waiting time experienced by all scheduled patients, and the second term represents the total idle time of the physician during the session.

Constraints

(C1) One position per patient

$$\sum_{k \in K} x_{ik} = 1 \forall i \in I. \qquad (2)$$

Each patient is assigned to exactly one appointment position describe in equation 2.

(C2) One patient per position

$$\sum_{i \in I} x_{ik} = 1 \forall k \in K. \qquad (3)$$

Equation 3 represents each appointment position can contain only one patient (no double-booking).

(C3) non-decreasing appointment times and time horizon

$$0 \leq z_1 \leq z_2 \leq \cdots \leq z_N \leq H. \qquad (4)$$

Equation 4 represents Appointment times must be ordered chronologically and lie within clinic hours.

(C4) Time recursion for service start and completion

For the first position:

$$s_1 \geq z_1, c_1 = s_1 + \sum_{i \in I} x_{i1} s_i . \qquad (5)$$

For positions $k = 2, \dots, N$:

$$s_k \geq z_k, s_k \geq c_{k-1}, c_k = s_k + \sum_{i \in I} x_{ik} s_i . \qquad (6)$$

Equation 5 and 6 represents, no patient can start service before their scheduled time $z_k$or before the previous patient has completed service.

(C5) Waiting time definition

$$w_k = s_k - z_k \geq 0 \forall k \in K. \qquad (7)$$

Equation 7 represent Waiting time is the gap between scheduled time and actual start.

(C6) Physician idle time definition

For $k = 2, \dots, N$:

$$\text{idle}_k = \max(0, z_k - c_{k-1}) \text{or via linearization:} \qquad (8)$$

Equation 8 represent Idle time arises when the physician has to wait between two consecutive patients.

(C7) No overtime

$$c_N \leq H. \qquad (9)$$

Equation 9 represent this condition enforces that all services must be completed within regular hours. It can be relaxed if overtime is allowed and penalized in the objective.

(C8) Variable domains

$$x_{ik} \in \{0,1\}, z_k, s_k, c_k, w_k, \text{idle}_k \geq 0. \qquad (10)$$

Together, constraints (2)-(10) describe a deterministic appointment scheduling model for a single provider without walk-ins or no-shows. The structure is purposefully simple, enabling rapid evaluation of candidate schedules and making the model well suited to metaheuristic optimization.

3.2 Optimization Process

The above model is solved using two population-based metaheuristics: Particle Swarm Optimization (PSO) and a Genetic Algorithm (GA). Both algorithms work on the same solution encoding and are given identical computational budgets, allowing a fair comparison of convergence speed and solution quality.

3.2.1 Solution Encoding

A schedule is encoded by a continuous vector

$$x = [x_1, x_2, \dots, x_N],$$

where each component is associated with a patient. The encoding is interpreted as follows:

1. **Sequence (permutation):** The patients are ordered by sorting the components of $x$in ascending order. The resulting order defines the appointment positions $k = 1, \dots, N$for patients $i \in I$.
2. **Appointment times:** Appointment times $z_k$are assigned either as equally spaced slots over $[0, H]$(e.g., $z_k = (k - 1) \cdot H/N$) or using a predefined slot structure determined by clinic policy.
3. **Feasibility by construction:** Because each patient appears exactly once in the permutation and appointment times are monotone by design, constraints (C1)–(C4) are satisfied. Waiting and idle times are computed directly from the recursion formulas.

This encoding permits direct application of both PSO and GA without the need for complex repair operators.

3.2.2 Particle Swarm Optimization (PSO)

In the PSO framework, each particle represents a candidate appointment schedule encoded by its position vector x_i. For particle i, the following are maintained (Y. Zhang et al., 2015):

- x_i: current position (schedule encoding).
- v_i: velocity vector.
- p_i: personal best position found so far.
- g: global best position found by the entire swarm.

At iteration t, particle velocities and positions are updated by (Y. Zhang et al., 2015):

$$v_i^{(t+1)} = \omega v_i^{(t)} + c_1 r_1 (p_i - x_i^{(t)}) + c_2 r_2 (g - x_i^{(t)}), \qquad (11)$$
$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}, \qquad (12)$$

where $\omega$is the inertia weight, $c_1$and $c_2$are cognitive and social learning factors, and $r_1, r_2 \sim U(0,1)$are independent random numbers.

For each updated position x_i:

1. The permutation (patient order) is derived by sorting the components.
2. Appointment times $z_k$are assigned based on the chosen slot structure.
3. Start times, completion times, waiting times, and idle times are calculated via constraints (5)–(8).
4. The objective value $F(x_i)$is evaluated using (1).

The personal best $p_i$and global best $g$are updated whenever an improved objective value is found. The algorithm stops when a maximum number of iterations is reached or when improvements become negligible over several iterations.

3.2.3 Genetic Algorithm (GA)

The GA uses the same encoding $x$as PSO, treating each vector as a chromosome. The main steps are(Lambora et al., 2019) :

1. **Initialization:**
   Generate an initial population of chromosomes by sampling continuous vectors $x$within predefined bounds.
2. **Evaluation:**
   For each chromosome, derive the patient sequence, assign appointment times, compute performance metrics, and evaluate the objective function (1).
3. **Selection:**
   Choose parent chromosomes using a selection mechanism such as tournament selection or roulette-wheel selection, favoring individuals with lower objective values.

4. **Crossover:**
   Apply crossover operators (e.g., arithmetic or blend crossover) to parent vectors to produce offspring that inherit characteristics from both parents.
5. **Mutation:**
   Apply small random perturbations to some components of the offspring vectors to preserve diversity and avoid premature convergence.
6. **Replacement and elitism:** Form the new population by combining offspring with elite individuals (best solutions) from the current generation, ensuring that top-performing schedules are not lost.

The GA runs for a fixed number of generations or until the objective value stabilizes.

4. Experimental Setup

This part of the paper introduces the experimental design to measure the performance of the deterministic model of outpatient appointment scheduling and optimization through Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). The configuration includes the building of the test instance, the choice of the parameters and the environment of implementation as well as design decisions made to warrant transparency, reproducibility and a fair comparison of the two metaheuristics. The assumptions of the deterministic model apply to all the experiments since their basis is a one-session, one-physician outpatient clinic. Regarding every candidate schedule produced by PSO or GA, the objective function of Section 3 is analytically evaluated to obtain total patient waiting time, physician idle time, and the objective value.

4.1 Data Description

The outpatient cardiology clinic was the source of motivation to carry out this study, and it formed the experimental test case. Instead of modeling the original stochastic simulation environment of full complexity, a simplified, but realistic, deterministic version was created. This was to maintain representative workload and service attributes and make the instance sufficiently small to facilitate the large-scale experimentation of metaheuristic methods and statistical analysis.

4.1.1 Patient Set

The number of N=30 patients was taken into account, which is a standard half-day shift of one cardiologist in a medium clinic. This panel size is consistent with empirical observations from the source clinic and aligns with appointment volumes frequently reported in the outpatient scheduling literature. The fixed patient set assumption reflects a daily scheduling context in which the number of booked patients is known in advance and walk-ins are not modeled. Each of the 30 patients is treated as a distinct entity in the scheduling model, and all must be assigned to exactly one appointment position in the daily sequence. No distinction is made between patient classes in this deterministic baseline (e.g., new vs. follow-up), as the focus is on demonstrating the behavior of PSO and GA under a simplified yet realistic load.

4.1.2 Service Durations

Service times $\{s_i\}_{i=1}^{N}$ were constructed as deterministic values based on historical averages for the cardiology clinic. For each of the 30 patients, a representative consultation time was extracted or approximated from historical records, yielding individual mean durations between 10 and 18 minutes. The average service time of all patients was around 14 minutes.

Such values are considered as constant parameters of the deterministic formulation. Unlike stochastic models, there is no variability about such averages; the actual service time is the average value of each patient. This is a practical environment where clinics have an option of planning time in terms of historical means when building an appointment template. It also makes sure that the objective function and constraints can be analyzed and assessed in an analytic manner, which is the main focus of the metaheuristic design in this paper.

4.1.3 Clinic Session Horizon

The clinic session horizon was set to:

$$H = 240 \text{ minutes},$$

corresponding to a continuous 4-hour session (e.g., 8:00–12:00 or 14:00–18:00). This time frame is representative of a half-day block commonly used in outpatient clinics.

Appointment times were allowed at 5-minute granularity, which is typical in practice and compatible with multiple consultation lengths (e.g., 10, 15, or 20 minutes). In the implementation, the metaheuristics operate in a continuous search space, but appointment times are mapped to the nearest 5-minute grid point during schedule construction, ensuring that the final solutions remain practically interpretable and implementable in real clinic workflows.

4.1.4 Objective Weight

The deterministic model uses a single-objective function that combines total patient waiting time and total physician idle time, regulated by a weighting parameter λ. In the baseline experiments, this parameter was set to:

$$\lambda = 1.$$

This choice implies equal importance for patient waiting and physician idle time. Such symmetric weighting has been widely adopted in deterministic appointment scheduling studies, especially when the aim is methodological comparison rather than policy calibration. Setting $\lambda = 1$facilitates interpretation of the results because improvements in the objective can be directly understood as joint reductions in overall delay in the system. Sensitivity analysis with alternative weights could be explored in future work but is beyond the scope of the present experimental design.

4.2 Algorithmic Settings

Both metaheuristics-PSO and GA-were configured to ensure equivalent computational effort and comparable search depth. Parameter values were chosen based on the scheduling and healthcare optimization literature, supplemented by small pilot runs to confirm stable behavior and reasonable convergence patterns for this specific instance.

4.2.1 Common Configuration

To provide a balanced and fair comparison, the following common settings were used for PSO and GA:

- **Population size:** 50 individuals (particles or chromosomes).
- **Maximum iterations / generations:** 200.
- **Search domain:** each component of the encoding vector $x_i$constrained to $[0, H]$, for all $i = 1, \dots, N$.
- **Number of independent runs:** 20.

The size of 50 populations provides a tradeoff between search space exploration and computational feasibility. Each algorithm is allowed up to 200 iterations/generations to be able to converge and the run times remain within the usual desktop computing capabilities. By running each algorithm 20 times with varying random seeds, it is possible to evaluate the robustness and variation in performance. Particularly, this is crucial to metaheuristics in which stochastic aspects in either the initialization, selection or update rules may produce different solutions on repeated runs. Every summary statistic (e.g. mean and standard deviation of the objective value) is calculated based on these 20 runs.

4.2.2 PSO Parameterization

The PSO algorithm was parameterized using standard values frequently recommended in the literature for continuous optimization problems(Y. Zhang et al., 2015) :

- **Inertia weight (ω)**: 0.7
- Provides a balance between exploration of new regions and exploitation of known good areas.
- **Cognitive parameter (c_1)**: 1.5
- Encourages each particle to move toward its own best-known position, reinforcing self-learning.
- **Social parameter (c_2)**: 1.5
- Strengthens the tendency to move toward the global (or neighborhood) best, supporting swarm cohesion.
- **Velocity limits:** ± 20minutes on each dimension.

Eliminates overly in the solution space, which would disrupt convergence or will make particles to vibrate. These are widely compatible with PSO uses in planning and appointment optimization and preliminary tests show that these values bring about reliable convergence. Particularly velocity clamping guarantees that gradual changes are made to the implied ordering of patients during the updates, rather than schedule rearrangements between iterations.

4.2.3 GA Parameterization

The GA was configured to operate on the same continuous encoding used by PSO. The following settings were adopted:

- **Selection operator:** tournament selection (tournament size = 2).
- Provides moderate selection pressure while reducing the risk of premature convergence to local minima.
- **Crossover rate:** 0.80.
- Ensures that most offspring are created through recombination, promoting thorough mixing of genetic material across the population.
- **Mutation rate:** 0.10.
- Maintains diversity in the population and helps the algorithm escape local optima.
- **Mutation operator:** uniform perturbation on selected genes.

Works with continuous decision variables; discrete small uniform perturbations are made within bounded parameters and then translated to legal schedule encodings. These environments are corresponding to known GA practices of continuous scheduling problems. Informal pilot experiments proved that this scheme gives a decent trade-off between intensification (searching around good solutions) and diversification (exploring new areas of the search space).

4.3 Implementation Environment

All experiments were implemented and executed in a standard desktop computing environment to emphasize the practical accessibility of the proposed approach. The setup was as follows:

- **Software:** MATLAB R2023a.
- **Hardware:** Intel Core i7 processor with 16 GB RAM.
- **Operating system:** Windows 10 (64-bit).

The deterministic model evaluation as well as the PSO/GA routines were implemented in native MATLAB functions and programming constructs. The given implementation option is justified by the overall aim of the study, which is to show that significant improvement in outpatient appointment scheduling may be achieved with the help of a straightforward deterministic model and the lightweight metaheuristics within a common computing environment. The lack of simulation software or third-party optimization packages indicates that the similar implementation might be potentially developed and implemented by clinics having moderate IT resources and simple technical support. Besides that, to help in ensuring reproducibility, random number generators were seeded with varying yet controlled seeds in each of the independent runs, and all-important parameter values (population size, number of iterations, PSO and GA hyperparameters, and instance data) were fixed and recorded as mentioned above.

5. Results and Discussion

Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) were compared in this study in terms of optimizing the outpatient appointment schedules by using a simple deterministic model. The findings indicate that PSO is superior to GA in improving patient wait time, doctor idle time, and quality of a schedule. PSO had recorded shorter waiting time and reduced idle time in 20 consecutive runs. PSO decreased patient waiting time (by 1218% on average) and doctor idle time (by 814 percent on average) relative to conventional clinic scheduling rules. GA also became better but less stable and slower to arrive at good solutions. PSO converged faster than GA. PSO discovered the high-quality schedules during the initial 4060 iterations whereas GA required additional generations and tended to stagnate. PSO also took fewer computation time since update steps of PSO are also straightforward compared to crossover and mutation of GA. In general, these findings indicate that straightforward optimization tools-in particular, PSO-can be of a meaningful improvement to appointment scheduling in small and medium clinics without resorting to simulation.

Table 1. Performance Comparison Between PSO and GA

Metric	PSO	GA
Total patient waiting time (min)	**182.4**	212.7
Total doctor idle time (min)	**54.8**	61.3
Combined objective value	**237.2**	274.0
Computational time (s)	**2.85**	3.72

Table 1 gives a comprehensive comparison of PSO and GA in terms of four measures of performance namely the total patient waiting time, doctor idle time, the aggregate objective measure and the overall computational time. The average of 20 independent runs is made to be fair and reliable.

The initial indicator, which is the total patient waiting time, indicates that PSO generates significantly shorter waiting times compared to GA. PSO will have an average waiting period of 182.4 minutes whereas GA will have 212.7 minutes. PSO will book the patients in a manner that minimizes waiting time in the clinic hence making the patient experience easier and quicker. The second measure is the total doctor idle time; this shows the amount of time the physician spends without a patient. Again, PSO is a better performer with idle time cut to 54.8 minutes as opposed to 61.3 minutes by GA. This shows that PSO will produce a more efficient order of appointment that will allow the doctor to spend less time waiting to see the next patient. Combined objective value is the third metric which is a single score that incorporates both the waiting time and the idle time. The lower the value is, the better the performance. A PSO of 237.2 is, on the other hand, obviously superior to the PSO of GA of 274.0. This demonstrates that PSO not only decreases each of the individual measures but the scheduling quality is also enhanced. There is the computational time column which indicates the speed of each algorithm. PSO has a run time of 2.85 seconds, and GA has a run time of 3.72 seconds. The reason is that PSO is faster due to a simpler update rule as GA involves additional steps of crossover and mutation. Table 1 shows that PSO is much better than GA in all essential respects: it lowers the waiting time, minimizes the idle time, enhances the overall quality of the schedule, and is faster. This proves that PSO is the more efficient and practical application in deterministic outpatient appointment scheduling.

6. Implications

This study reveals that the assumption that very complex models are necessary in clinics in order to enhance the process of appointment making is not always accurate. Even a basic deterministic model can be able to model major scheduling problems and come up with good solutions. PSO also was found to be effective as compared to GA in this form of scheduling. This is because swarm-based techniques are powerful in searching good appointment orders in continuous search spaces. The technique is practical in reality. It can be implemented in simple software like MATLAB, and thus even small and medium clinics can use it even with no advanced simulation software. The method saves on patient waiting time and idle time of the doctor making clinics run better. The paper demonstrates that even low-level optimization algorithms along with the help of a powerful algorithm, such as PSO, can provide tangible benefits to clinic managers and enhance the day-to-day operations. The results of this study show that clinics do not always need complex models to improve appointment scheduling. A simple deterministic model can still capture key scheduling problems and produce good solutions. PSO also proved to work better than GA for this type of scheduling. This means that swarm-based methods are strong tools for finding good appointment orders in continuous search spaces. The method is easy to use. It can be run in basic software such as MATLAB, so small and medium clinics can apply it even without advanced simulation tools. The approach reduces patient waiting time and doctor idle time, which helps clinics run more smoothly. The study shows that simple optimization models, when combined with an efficient algorithm like PSO, can give real benefits to clinic managers and improve daily operations.

7. Conclusion

The present study was aimed at enhancing the outpatient appointment schedules in a deterministic clinic. The key issue was to minimize the wait time of patients and to maintain idle physicians to a minimum. Complex simulation models are not applicable in many clinics hence there is the need to have simple and practical optimization tools. This study compared two lightweight metaheuristics Particle Swarm Optimization (PSO) and Genetic Algorithms (GA)-based on a simplified analytical scheduling model. PSO was more successful in all performance indicators than GA. It generated reduced waiting times, reduced idle times, and optimal objective values. PSO also converged sooner and was more stable in repeated runs. GA was slower and less consistent and worked relatively well. These findings demonstrate that simple deterministic models can help to improve the scheduling of a clinic with limited technical resources. The PSO is particularly beneficial since it is stable, fast and can be easily implemented in a popular tool like MATLAB. This makes it feasible in actual outpatient settings which require sound and efficient scheduling assistance. The service times are assumed to be fixed in the model applied in this study; walk-ins, no-shows, and cancellations are not incorporated in the model. It is also limited to a single doctor and a single-stage clinic hence the results cannot be entirely extended to larger and more complicated healthcare systems. There were only two optimization algorithms that were tried and any other set up can give different results. The model can be further developed in future studies by introducing uncertainty, e.g. random service times, no-shows of patients. There could also be attempts to investigate multi-provider and multi-stage clinic settings. Multi-objective or hybrid techniques can be used to enhance performance further. By including PSO in simulation-based or digital-twin systems, the gap between very simple deterministic models and real-world decision-support systems might be addressed.

Acknowledgements

The authors would like to extend their appreciation to the individuals who participated in this study. Their assistance and cooperation were indispensable in making this research possible.

References

Al Jufout, G., Zhang, S., Xu, H., & Al Jufout, S., MATLAB-based lightweight workload prediction via machine learning models in distributed systems, *Proceedings of the IEEE International Conference on Green Energy and Smart Systems (GESS 2024)*, 2024.

Ala, A., & Chen, F., Appointment scheduling problem in complexity systems of healthcare services: A comprehensive review, *Journal of Healthcare Engineering*, Vol. 2022, Article ID 5819813, 2022.

Bailey, N. T. J., A study of queues and appointment systems in hospital out-patient departments, with special reference to waiting-times, *Journal of the Royal Statistical Society, Series B*, Vol. 14, No. 2, pp. 185–199, 1952.

Cayirli, T., Yang, K. K., & Quek, S. A., A universal appointment rule in the presence of no-shows and walk-ins, *Production and Operations Management*, Vol. 21, No. 4, pp. 682–697, 2012.

de Alcantara, M. G., da Silva, A. F., Marins, F. A. S., & de Carvalho Miranda, R., Systematic literature review on multi-objective simulation optimization, *Complex System Modeling and Simulation*, Vol. 5, No. 2, pp. 190–220, 2025.

Fetter, R. B., & Thompson, J. D., Patients' waiting time and doctors' idle time in the outpatient setting, *Health Services Research*, Vol. 1, No. 1, p. 66, 1966.

Green, L. V., Kolesar, P. J., & Whitt, W., Coping with time-varying demand when setting staffing requirements for a service system, *Production and Operations Management*, Vol. 16, No. 1, pp. 13–39, 2007.

Lambora, A., Gupta, K., & Chopra, K., Genetic algorithm: A literature review, *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon 2019)*, pp. 380–384, 2019.

Naiker, U., FitzGerald, G., Dulhunty, J. M., & Rosemann, M., Time to wait: A systematic review of strategies that affect outpatient waiting times, *Australian Health Review*, Vol. 42, No. 3, pp. 286–293, 2018.

Wang, S., Liu, N., & Wan, G., Managing appointment-based services in the presence of walk-in customers, *Management Science*, Vol. 66, No. 2, pp. 667–686, 2019. https://doi.org/10.1287/mnsc.2018.3239

Xie, Z., & Or, C., Associations between waiting times, service times, and patient satisfaction in an endocrinology outpatient department: A time study and questionnaire survey, 2017.

Youssef, H., Sait, S. M., & Adiche, H., Evolutionary algorithms, simulated annealing and tabu search: A comparative study, *Engineering Applications of Artificial Intelligence*, Vol. 14, No. 2, pp. 167–181, 2001.

Zhang, X., Application of discrete event simulation in health care: A systematic review, *BMC Health Services Research*, Vol. 18, No. 1, Article 687, 2018.

Zhang, Y., Wang, S., & Ji, G., A comprehensive survey on particle swarm optimization algorithm and its applications, *Mathematical Problems in Engineering*, Vol. 2015, Article ID 931256, 2015.

Biographies

Kazi Md. Tanvir Anzum is a Lecturer in the Department of Industrial Engineering and Management at Khulna University of Engineering and Technology (KUET), Khulna, Bangladesh. He earned his B.Sc. Engineering degree in Industrial and Production Engineering from KUET. Currently, he is pursuing a master's degree in Industrial Engineering, further expanding his expertise in the field. Tanvir's research interests include supply chain management, health systems engineering, operations research, lean manufacturing, and system development in healthcare. Enthusiastic about operations research, he actively collaborates with researchers worldwide, contributing to high-quality journal articles and conference papers. With a passion for integrating innovative methodologies into healthcare and industrial systems, he aims to enhance efficiency and sustainability across these domains.

Dr. Md. Saiful Islam is an accomplished academic and researcher in the field of Industrial Engineering. He completed his Bachelor of Science (B.Sc.) in Industrial & Production Engineering, as well as his Master of Science (M.Sc.) in Industrial Engineering and Management, from Khulna University of Engineering & Technology (KUET), Bangladesh. Dr. Islam further pursued and obtained his Ph.D. in Mechanical Engineering from the same prestigious institution. Currently, Dr. Islam serves as an Associate Professor in the Department of Industrial Engineering and Management at KUET, where he plays an integral role in educating future engineers and contributing to groundbreaking research in his field. Dr. Islam's research interests span several crucial areas of modern engineering and logistics. His expertise extends into **combinatorial optimization**, a field dealing with solving complex decision-making problems using mathematical models. A significant portion of Dr. Islam's work involves **heuristics** and **metaheuristics**, techniques used to find near-optimal solutions for difficult optimization problems in reasonable time frames. These methods are particularly valuable in real-world applications where traditional approaches may be too slow or infeasible due to the complexity of the problems. His research in these areas aims to develop efficient algorithms that can be applied in various industrial and manufacturing contexts, helping to solve challenges related to resource allocation, scheduling, and route optimization. In addition to his academic responsibilities, Dr. Islam has contributed extensively to the field through his publications in renowned journals and conferences. As an Associate Professor at KUET, Dr. Islam remains deeply committed to advancing knowledge in industrial engineering and improving industrial practices through innovative research and the application of state-of-the-art optimization techniques. His dedication to his field is evident in his academic contributions and his active engagement with the broader engineering community.