Federated Learning Framework for Nurse Scheduling to Lower Fatigue Levels of Nurses

Michal Aibin
School of Computing and Academic Studies
Department of Computing
British Columbia Institute of Technology
Vancouver, Canada
maibin@bcit.ca

F. Chen, M. Menzi, L. Scholte
Z. Jiang, Y. Chen, X. Jin, C. Liu
Khoury College of Computer Sciences
Northeastern University
Vancouver, Canada
vancouver@northeastern.edu

Ania Aibin
School of Health Sciences
Department of Nursing
British Columbia Institute of Technology
Vancouver, Canada
aaibin@bcit.ca

***Abstract*—In this paper, we introduce a federated learning framework to optimize nurse scheduling and reduce fatigue levels among nurses. The proposed approach leverages decentralized machine learning, allowing multiple healthcare institutions to collaboratively improve scheduling models without sharing sensitive data. Our framework enhances scheduling efficiency and nurse well-being by predicting physical and emotional fatigue. We use Gradient Boosting Machines (XGBoost and LightGBM) and Random Forest for feature selection, and Long Short-Term Memory (LSTM) neural network for model training. The performance of these algorithms is evaluated based on their ability to predict fatigue levels and improve scheduling outcomes, compared to Moth Flame, Grey Wolf and Multiverse Optimizers. This research highlights the importance of privacy-preserving data techniques and their application in healthcare to create more effective and humane scheduling systems.**

***Index Terms*—nurse rostering, scheduling, LSTM, big data**

I. Introduction

Employee and volunteer scheduling is crucial for the seamless operation of organizations that operate in shifts, such as those in tech support, retail, and healthcare sectors. These organizations often require a continuous presence, potentially operating 24/7, to maintain the necessary actions [1].

The creation of an appropriate shift schedule demands the consideration of hard constraints—like those imposed by labor laws and the physical impossibility of one employee working multiple shifts simultaneously—as well as soft constraints, which account for employee preferences, such as avoiding work on specific days [2]. Ensuring all hard constraints are met is critical for the validity of a schedule. Moreover, accommodating as many soft constraints as possible is desirable to enhance employee satisfaction. Thus, shift scheduling emerges as a complex mathematical optimization problem.

In addressing the nurse scheduling problem, this study introduces the application of federated learning. This method is particularly pertinent in healthcare, where privacy and data security are paramount [3]. By using federated learning, hospitals and healthcare facilities can collaborate on improving scheduling models while safeguarding individual employee data and adhering to strict privacy regulations.

Expanding on this foundation, our goal is to utilize machine learning and data analysis through federated learning to predict both physical and emotional fatigue levels among nurses. This innovative approach is informed by models and correlations outlined in recent studies [4], [5], enabling the development of a comprehensive framework that addresses scheduling efficiency and prioritizes the well-being of nursing staff. By incorporating these predictive models into the scheduling process, our research aims to create more effective and humane scheduling solutions that reduce nurse fatigue, retain nursing staff and enhance overall patient care.

II. Related Works

Optimization algorithms play a crucial role in scheduling, offering solutions to complex problems by finding the most efficient allocation of resources under given constraints. In healthcare, particularly nurse scheduling, these algorithms address the challenge of assigning shifts that balance hospital needs with nurses' preferences and qualifications. The literature reveals a variety of approaches, including linear programming [6]–[8], integer programming [9], and evolutionary algorithms [2], [10], [11].

On the other hand, we have federated learning that emerges as a pivotal approach in healthcare, addressing the critical need for privacy and security while harnessing the power of decentralized data for medical research and applications. Authors of [3] illustrates that federated learning models can achieve comparable outcomes to traditional centralized models, ensuring robust privacy protections without incurring significant computational overhead. Similarly, Gandhi et al. [12] demonstrate the application of federated learning, emphasizing its capability to preserve the privacy of sensitive medical data across diverse healthcare environments.

This study represents a pioneering effort in applying federated learning to reduce nurse fatigue by creating AI-optimized shift schedules. Unlike previous studies that have primarily focused on optimizing schedules to meet operational efficiency and employee preferences, our work seeks to mitigate physical and emotional fatigue among nursing staff directly. By utilizing federated learning, we enable a collaborative approach to model training across multiple healthcare institutions without compromising data privacy [13]. This method facilitates the development of robust predictive models capable of assessing fatigue levels, thus creating more humane and effective scheduling systems.

III. Nurse Scheduling Problem

The nurse scheduling problem, which can be extended to other industries, includes a mix of hard constraints (must haves) and soft constraints (nice to have). In this context, hard constraints related to legal regulations and organizational requirements must always be met. Soft constraints are fulfilled as best as possible because they relate to personnel wishes and often cannot be completely met [14]. This equation works reasonably well in an environment where resource supply and demand are well balanced. However, the demand for nurses has skyrocketed, and they have become a scarce resource in the market due to dissatisfaction with employment [15], [16].

Let us discuss a sample ward. A total of 25 nurses are assigned shifts according to a repeating 14-day calendar. Each day is split into two shifts, day (Dx) and night (Nx), resulting in 12-hour shifts. A sample schedule is shown in Table I:

TABLE I
An empty schedule without the names of nurses

		Sun	Mon	Tue	Wed	Thur	Fri	Sat
Week 1	**Dx**							
	Nx							
Week 2	**Dx**							
	Nx							

To limit the scope of the scheduling, it is assumed that shifts must be worked in their entirety. We define our variables in Table II:

TABLE II
The set of variables used to define the nurse scheduling problem

Variable	Description
N	The number of nurses that need to be scheduled
W	The number of weeks in the schedule
D	The number of days in the schedule. This is equivalent to $7 \times W$ where 7 is the number of days in a week
S	The number of unique shifts per day
n	A nurse in the set $[0, N)$
w	A week in the set $[0, W)$
d	An day in the set $[0, D)$
s	A shift in the set $[0, S)$
x	A vector of length $N \times D \times S$ representing a schedule. Each component may be 0 or 1
x_i	The i-th component of schedule x
x_{nds}	A component of schedule x representing nurse n, day d, and shift s. This is equivalent to x_i where $i = nDS + dS + s$
$f(x)$	A fitness function that describes the overall nurses' dissatisfaction towards a schedule x. A smaller number is more ideal
$f_n(x)$	A fitness function representing an individual nurse n's dissatisfaction towards a schedule x. A smaller number is more ideal
$p(x)$	A penalty function that is included in $f(x)$ if x represents a schedule that violates hard constraints.

The schedule can be represented mathematically as a vector with $N \times D \times S$ dimensions, where N is the number of nurses (25), S is the number of days in the schedule (14), and S is the number of shifts in a day (2). Alternatively, the number of days D can also be represented as $7 \times W$ where W is the number of weeks in the schedule (2). This yields a schedule x with a dimension of 700 as shown below:

$$x = [x_0 \quad x_1 \quad \dots \quad x_{699}] \text{ where } x_i = \begin{cases} 0, & \text{nurse } n \text{ does not work shift } s \text{ on day } d \\ 1, & \text{nurse } n \text{ works shift } s \text{ on day } d \end{cases} \tag{1}$$

where

i = the index of a component in x
$n = \frac{i}{DS}$
$d = \frac{i \bmod DS}{S}$
$s = i \bmod S$

From this, it becomes clear that there are $2^{700} \approx 5.3 \times 10^{210}$ unique schedules. Due to this massive size, it becomes important to filter infeasible schedules that violate a certain set of hard constraints to reduce the search space for optimization. These hard constraints are defined as follows:

- No nurse may work more than five shifts in a row
- Each nurse must work at least one shift
- An exact number of nurses must be assigned to every shift, depending on the unit (for example, critical care vs medicine)

These constraints can be defined mathematically on the schedule x as follows:

$$\sum_{d=7w}^{7(w+1)} \sum_{s=0}^{S} x_{nds} \leq 5 \qquad \forall n \in [0, N), \forall w \in [0, W)$$

$$\sum_{d=0}^{D} x_{nds} \geq 1 \qquad \forall n \in [0, N), s = 0$$

$$\sum_{n=0}^{N} x_{nds} = K \qquad \forall d \in [0, D), \forall s \in [0, S)$$

In addition, it is understood that each nurse has personal preferences regarding how their shifts are scheduled. These preferences are not mandatory to fulfill and are therefore considered soft constraints. Although not required, an optimal schedule will maximize the collective preferences of all nurses so that high employee morale is maintained. Examples of nurse preferences could include but are not necessarily limited to

- desire to work either night or day shifts
- desire to work particular days of the week
- desire to avoid consecutive shifts
- desire to work with (or not work with) certain nurses

To allow for optimized schedules, each nurse will be assigned a fitness function $f_n(x)$ where n is the identifier of the nurse from $[0, N)$. A fitness function that outputs a large number represents that a nurse is highly dissatisfied with that schedule, while a low number means that the nurse is highly satisfied. Moreover, we will penalize for higher physical and emotional fatigue in the same fitness function. An example of what a nurse fitness function $f_n x$ might look like is provided in Table III. The overall fitness of a schedule is obtained by simply accounting for all nurses' opinions of the schedule:

$$f(x) = \sum_{n=0}^{N} f_n(x) \tag{2}$$

Please note that penalty = 1 or penalty = 3 is applied for each row of violated requirements.

TABLE III
FITNESS FUNCTION CRITERIA

Criteria	Low Fatigue Score, Penalty = 0	Moderate Fatigue Score, Penalty = 1	High Fatigue Score, Penalty = 3	Notes
Total hours worked	<= 40 h	40+ to 48 h	>48 h	
Shift extensions (% days worked)	None	<= 50%	>50%	
Short breaks (<9 hours between shifts)	0	1	>1	
Number of night shifts	0	1–2	>2	
Long breaks (>24 hours between shifts)	>= 2	1	0	
Roster changes	No	Change requested	Change not requested	
Nights for sleep (11pm-7am)	6–7 nights	4–5 nights	0–3 nights	
Days fully rested	6–7 days	4–5 days	0–3 days	
Work schedule flexibility	Highly flexible	Slightly flexible	Rigid	Scale 0 (highly rigid) to 4 (highly flexible)
Workload (Patient:Nurse ratio)	Lower	Moderate	High	Aggregated to unit level
Nurse preferences	Fulfilled	Partially met	Not met	Preferences like shift type, specific days, etc.
Desire to work specific shifts (day/night)	Preference matched	-	Preference not matched	Considering individual nurse preferences
Desire to work specific days of the week	Preference matched	-	Preference not matched	Considering individual nurse preferences
Desire to avoid consecutive shifts	Preferences met	-	Preferences not met	
Desire to work with specific nurses	Preferences met	-	Preferences not met	Can also include avoiding certain nurses

Penalties for certain nurse preference violations can also be represented with matrices. In particular, p_s represents penalties associated with violating shift preferences, p_d represents penalties associated with violating day preferences, and p_c represents penalties associated with violating coworker preferences. Each column of a matrix corresponds to a nurse. Each row of a matrix corresponds to either a shift, day, or another nurse for p_s, p_d, and p_c, respectively. These matrices are exampled in 3, 4, and 5:

$$p_s = \begin{bmatrix} p_{0,0} & \cdots & p_{0,N} \\ p_{1,0} & \cdots & p_{1,N} \end{bmatrix} \tag{3}$$

$$p_d = \begin{bmatrix} p_{0,0} & \cdots & p_{0,N} \\ \vdots & \ddots & \vdots \\ p_{7,0} & \cdots & p_{7,N} \end{bmatrix} \tag{4}$$

$$p_c = \begin{bmatrix} p_{0,0} & \cdots & p_{0,N} \\ \vdots & \ddots & \vdots \\ p_{N,0} & \cdots & p_{N,N} \end{bmatrix} \tag{5}$$

One of the difficulties in generating a fitness function is understanding how to represent both hard and soft constraints. There is existing research on converting constrained optimization problems into unconstrained optimization problems that describe how to do this. An approach worth considering is introducing a penalty function. A penalty function is a common approach that adds or subtracts a penalty to a fitness function based on the amount of constraints that are violated [17]. Using this technique, a massive penalty for hard constraint violations can be introduced. For example, other related works in nurse scheduling have taken the approach of applying a penalty of 999999 for hard constraint violations and smaller numbers for soft constraint violations [18]. By taking this approach, an optimization algorithm should effectively disregard schedules that have hard constraint violations due to a low output of $f_{overall}$, which might look similar to (6)

$$f(x) = p(x) + \sum_{n=0}^{N} f_n(x) \tag{6}$$

where $p(x)$ is the penalty function returning a negative value for inputted schedules that violate hard constraints.

IV. Federated Learning Framework

Federated learning presents a paradigm shift in decentralized information aggregation, enabling multiple entities to contribute towards a collective model without sharing raw data. This research leverages federated learning to tackle the challenge of nurse scheduling across the five regional health authorities in British Columbia, each with its unique operational guidelines and varying numbers of hospitals and wards. British Columbia's healthcare system is partitioned into five health authorities: Vancouver Coastal Health, Fraser Health, Interior Health, Island Health, and the Northern Health Authority. These entities differ in geographic coverage and the scale and scope of operations, as they oversee different numbers of hospitals, each further subdivided into numerous wards with their own scheduling demands.

A. Base Schedule Creation

The foundation of our federated learning application begins with creating base schedules for each ward within the hospitals operated by the health authorities. These schedules are formulated to comply with local regulations, staffing requirements, and initial nurse availability. With 76 hospitals, each with 10+ wards, we created synthetic data for over 30,000 nurses with soft preferences to create a pool of nurses to accommodate each hospital's needs. This decentralized approach enabled us to train local models on each hospital's data, preserving privacy and security. The central server aggregated the local models' parameters to form a global model, which was then redistributed back to the hospitals. This architecture significantly enhanced the robustness of our Nurse Scheduling model, as it used a diverse set of real-world data. The federated learning framework resulted in more accurate and personalized nurse schedules, reducing overall penalty scores and improving nurse satisfaction across hospital settings, as seen in Figure 1.

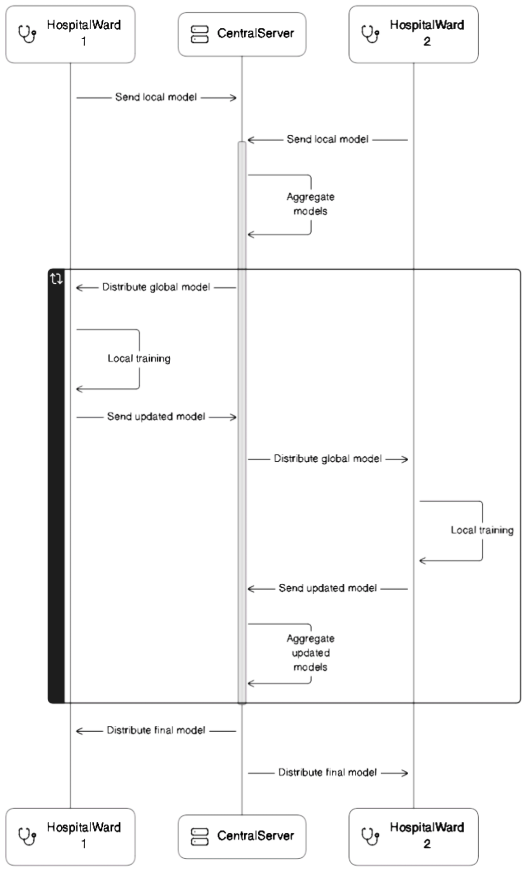

Fig. 1. Federated Architecture Diagram

B. Machine Learning for Feature Selection

The first task was to detect which features from the soft preferences affect the penalty score the most for each specific hospital. In our context, we use the following approaches to model complex relationships between scheduling variables and nurse preferences:

- Gradient Boosting Machines (GBMs), specifically XGBoost and LightGBM
- Random Forest

XGBoost, a popular GBM method, builds a sequence ensemble of decision trees. Each tree corrects the errors of its predecessors, focusing on difficult-to-predict cases. This highly efficient method performs well with large datasets and high-dimensional data. It also provides feature importance scores, helping us understand which preferences influence the penalty score most. Similarly, LightGBM, another GBM variant, constructs trees using a histogram-based approach, making it faster and more memory-efficient. LightGBM also provides insights into feature importance, helping us identify key factors affecting penalty scores.

In addition to GBMs, we used Random Forest, which builds multiple decision trees independently and combines their results. Each tree is trained on a random subset of the data and features, reducing the risk of overfitting and improving generalization. Random Forest also delivers feature importance metrics, allowing us to pinpoint the most significant scheduling variables and nurse preferences that impact penalty scores.

In Table IV, we present the comparison results of the feature importance and feature selection generated by XGBoost, Random Forest, and LightGBM models.

TABLE IV
COMPARISON OF FEATURE SELECTION FOR XGBOOST, RANDOM FOREST, AND LIGHTGBM

Feature Name	Feature Importance (Selected/Not Selected with Importance Score)		
	XGBoost	Random Forest	LightGBM
shift preference	Selected (0.1265)	Selected (0.3368)	Not Selected (0.0000)
consecutive shifts	Selected (0.0519)	Selected (0.0683)	Selected (0.1570)
preferred days	Selected (0.7811)	Selected (0.1997)	Selected (0.1433)
night shift fatigue	Selected (0.0228)	Not Selected (0.0010)	Not Selected (0.0400)
sleep	Selected (0.0089)	Not Selected (0.0053)	Selected (0.1193)
workload	Not Selected (0.0016)	Not Selected (0.0024)	Selected (0.1270)
preferred co-workers	Not Selected (0.0036)	Not Selected (0.0012)	Selected (0.0570)
fully rested days	Not Selected (0.0013)	Not Selected (0.0005)	Selected (0.0210)
long breaks	Not Selected (0.0024)	Not Selected (0.0024)	Selected (0.0970)
roster swaps	Not Selected (0.0000)	Not Selected (0.0000)	Not Selected (0.0000)
extended shifts	Not Selected (0.0000)	Not Selected (0.0000)	Not Selected (0.0000)
weekly hours	Not Selected (0.0000)	Selected (0.0068)	Selected (0.2050)
monthly total working hours	Not Selected (0.0000)	Selected (0.3758)	Not Selected (0.0000)

The comparison results reveal that the 'preferred days penalty' and 'shift preference penalty' are crucial for nurse mental health, as these directly impact their ability to plan and balance work with personal life. XGBoost and Random Forest highlight these as important, while LightGBM focuses on 'workload penalty'. The 'consecutive shifts penalty' consistently affects nurse fatigue, a key feature across all models. The consistent selection of 'preferred days penalty' by all models indicates its significant influence on mental well-being. In contrast, varying selection patterns for other penalties reflect each model's unique evaluation method.

In the next step, we performed the following:

- Local Model Training: Each participating entity (a hospital ward) trains a local model on its data to predict scheduling disruptions.
- Model Aggregation: A central server aggregates these local models into a global model. Aggregation is done by averaging the model parameters and accounting for the data's size or the performance of local models.
- Model Distribution: The aggregated global model is distributed back to the participants for further local training or implementation.

To train the ML model, we used the LSTM architecture [19], repeating the above steps over multiple rounds to improve the model's accuracy and robustness. After the model was trained, we compared the efficiency of the scheduling per hospital against three well-known approaches - Moth Flame [11], Grey Wolf [20] and Multiverse Optimizers [21].

V. Results

The predictive insights obtained from our federated learning models allow us to proactively adjust schedules to minimize disruptions and reduce nurse fatigue. The comparison spans three different training scenarios: Local Training, Federated Learning Training, and Federated Learning Training augmented with Feature Selection, as shown in Table V.

TABLE V
Penalty score for various optimizers, averaged per each ward in each tested hospital (the number of hospitals = 76, with an average of 11 wards per hospital)

	Local Training (Average Penalty Score)	Federated Learning Training (Average Penalty Score)	Federated Learning Training with Feature Selection (Average Penalty Score)
Moth Flame Optimizer	28.28	26.53	24.50
Grey Wolf Optimizer	23.15	21.62	19.89
Multiverse Optimizer	23.70	22.09	20.44
LSTM	23.20	21.67	20.24

For the Moth Flame Optimizer (MFO), local training resulted in an average penalty score of 28.28. Federated learning improved this to 26.53, a 6.2% reduction, and further to 24.50 with feature selection, marking a 7.7% improvement.

The Grey Wolf Optimizer (GWO) saw its average penalty score drop from 23.15 in local training to 21.62 with federated learning, a 6.6% reduction, and to 19.89 with feature selection, indicating an 8.0% improvement.

Similarly, the Multiverse Optimizer (MVO) reduced its average penalty score from 23.70 in local training to 22.09 with federated learning, a 6.8% decrease, and further to 20.44 with feature selection, a 7.5% enhancement.

The Long Short-Term Memory (LSTM) model followed a similar trend, with the average penalty score dropping from 23.20 in local training to 21.67 with federated learning, a 6.6% reduction, and 20.24 with feature selection, another 6.6% improvement.

In summary, applying federated learning consistently improved the performance of all optimizers compared to local training, with feature selection further enhancing these results. The Grey Wolf Optimizer showed the most notable performance improvements, indicating its robustness in handling the nurse scheduling problem within the federated learning framework.

VI. Conclusion

Our study presents a federated learning framework for nurse scheduling that prioritizes reducing nurse fatigue through decentralized machine learning. By enabling hospitals to train models collaboratively while safeguarding data privacy, we have developed a robust system that predicts nurse fatigue and incorporates these predictions into the scheduling process. Specifically, the integration of Grey Wolf Optimization (GWO) has shown superior performance in optimizing scheduling outcomes compared to other methods. This approach addresses the immediate needs of healthcare providers and sets a precedent for applying privacy-preserving techniques in other domains. Future work will explore the use of Stochastic Controlled Averaging to improve convergence rates and handle variability in nurse scheduling data across healthcare institutions.

References

[1] M. Cheng *et al.*, "Nursing care scheduling problem: Analysis of staffing levels," in *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2007, pp. 1715–1720.

[2] J. Schrack *et al.*, "Combining Tabu Search and Genetic Algorithm to Determine Optimal Nurse Schedules," *Canadian Conference on Electrical and Computer Engineering*, vol. 2021-September, 9 2021.

[3] A. Sadilek *et al.*, "Privacy-first health research with federated learning," *NPJ digital medicine*, vol. 4, no. 1, p. 132, 2021.

[4] S. R. Dhaini *et al.*, "Work schedule flexibility is associated with emotional exhaustion among registered nurses in swiss hospitals: A cross-sectional study," *International Journal of Nursing Studies*, vol. 82, pp. 99–105, 2018.

[5] P. Gander *et al.*, "Development and evaluation of a matrix for assessing fatigue-related risk, derived from a national survey of nurses' work patterns," *International Journal of Nursing Studies*, vol. 112, p. 103573, 2020, shift work in nursing: closing the knowledge gaps and advancing innovation in practice.

[6] B. Jaumard *et al.*, "A generalized linear programming model for nurse scheduling," *European journal of operational research*, vol. 107, no. 1, pp. 1–18, 1998.

[7] L. Trilling *et al.*, "Nurse scheduling using integer linear programming and constraint programming," *IFAC Proceedings Volumes*, vol. 39, no. 3, pp. 671–676, 2006.

[8] P. Strandmark *et al.*, "First-order linear programming in a column generation-based heuristic approach to the nurse rostering problem," *Computers & Operations Research*, vol. 120, p. 104945, 2020.

[9] E. B. Bövarsdóttir *et al.*, "A flexible mixed integer programming-based system for real-world nurse rostering," *Journal of Scheduling*, pp. 1–30, 2022.

[10] A. Ayoughi *et al.*, "Dragonfly algorithm application for solving the nurse scheduling problem," in *IEEE ICNC*, 2024.

[11] C. Huang *et al.*, "Emergency Surgical Scheduling Model Based on Moth-flame Optimization Algorithm," in *2023 International Conference on Computing, Networking and Communications (ICNC)*, 2023, pp. 89–94.

[12] N. Gandhi *et al.*, "Leveraging towards privacy-preserving using federated machine learning for healthcare systems," in *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, 2021, pp. 1–6.

[13] C. Zhang *et al.*, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, 2021.

[14] S. Abdennadher and H. Schlenker, "Nurse scheduling using constraint logic programming," *IAAI-99 Proceedings*, p. 6, 1999. [Online]. Available: https://www.aaai.org/Papers/IAAI/1999/IAAI99-118.pdf?q=optimal-scheduling-using-constraint-logic-programming

[15] A. Baumann and M. Crea-Arsenio, "The crisis in the nursing labour market: Canadian policy perspectives," in *Healthcare*, vol. 11, no. 13. MDPI, 2023, p. 1954.

[16] W. Bahlman-van Ooijen *et al.*, "Nurses' motivations to leave the nursing profession: A qualitative meta-aggregation," *Journal of advanced nursing*, vol. 79, no. 12, pp. 4455–4471, 2023.

[17] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11, pp. 1245–1287, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782501003231

[18] W. J. Gutjahr and M. S. Rauner, "An aco algorithm for a dynamic regional nurse-scheduling problem in austria," *Computers Operations Research*, vol. 34, no. 3, pp. 642–666, 2007, logistics of Health Care Management. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0305054805001139

[19] F. A. Gers *et al.*, "Learning to forget: Continual prediction with LSTM," *Neural Computation*, 2000.

[20] S. Mirjalili *et al.*, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997813001853

[21] M. S. . H. A. Mirjalili, S., "Multi-verse optimizer," *Neural Computing and Applications*, vol. 27, pp. 495–513, 2016. [Online]. Available: https://link.springer.com/article/10.1007/s00521-015-1870-7#citeas