**Solving the Nurse Scheduling Problem**

**using Genetic Algorithms**

**Our Team**

Al-Hussein Mohammed Ahmed 202000144 Level (3) general

Abdelrahman Mohsen Mohammed Ali 202000523 Level (3) general

Moaz Nasser Abdelaziz Al-Azab 202000925 Level (3) general

Ahmed Mohammed Saeed Mohammed 202000071 Level (3) general

Abdullah Samir Mohammed Abbas 202000545 Level (3) general

Ahmed Magdi Shehata Ahmed Level (3) 202000061 Level (3) general

**Project URL:**

<http://shorturl.at/gMS47>

Nurse scheduling problem

**Project idea**

The nurse scheduling problem (NSP) is the operations research problem of finding an optimal way to assign nurses to shifts, typically with a set of hard constraints which all the valid solutions must follow, and a set of soft constraints which define the relative quality of valid solutions.

Solutions to the nurse scheduling problem can be applied to constrained scheduling problems in other fields. The nurse scheduling problem involves the assignment of shifts and holidays to nurses. Each nurse has their own wishes and restrictions, as does the hospital. The problem is described as finding a schedule that both respects the constraints of the nurses and fulfills the objectives of the hospital. Conventionally, a nurse can work

3 shifts because nursing is shift work:

• day shift • night shift • late night shift

In this problem we must search for a solution satisfying as many wishes as possible while not compromising the needs of the hospital.

There are two types of constraints:

1. Hard constraints: if this constraint fails then the entire schedule is invalid.

2. Soft constraints: it is desirable that these constraints are met but not meeting them does not make the schedule invalid.

Some examples of constraints are:

• A nurse does not work the day shift, night shift and late-night shift on the same day (for obversions).

• A nurse may go on a holiday and will not work shifts during this time.

• A nurse does not do a late-night shift followed by a day shift the next day.

The Nurse scheduling problem (NSP) represents a difficult class of multi-objective optimization problems consisting of several interfering objectives between the hospitals and individual nurses. Several constraint-based optimization techniques have been proposed to solve automated nursing scheduling problems in an acceptable computation time but most of these techniques are characterized by premature convergences which inhibit optimal global solution.

The NSP is a staff scheduling problem that intends to assign a set of nurses to work shifts to maximize hospital benefit by considering a set of hard and soft constraints like allotment of duty hours, hospital regulations, and so forth. This nurse scheduling is a delicate task of finding combinatorial solutions by satisfying multiple constraints. The Nurse Scheduling Problem (NSP) is a combination of optimization problem and important management functions performed by nurses who directly affected the hospital services and the patient care. Staff scheduling is the process of constructing work timetables encoding for staff in order to satisfy the demand for services.

**Main functionalities**

1. Calculate the minimum and maximum numbers of shifts assigned to a given nurse in each week.

2. calc the hours worked per week (shift =8).

3. days worked consecutively in month.

4. days off consecutively in month.

**Similar applications in the market**

1.Connecteam (Desktop application/Mobile application)

2.10to8 (Desktop application)

3.Deputy (Mobile application)

4.Nurse grid (Mobile application)

5.eSchedule (Web application)

**Academic publications relevant to the idea**

**Links for academic research papers solving the nurse scheduling problem using the genetic algorithm:**

1.Source: Department of Computing, University of Bradford, Bradford, BD7 1DP, UK

URL:

<https://www.sciencedirect.com/science/article/abs/pii/S0305054803000340>

2.Source : Department of Global Marketing and Logistics, MingDao University, 369 Wen-Hua Road, Peetow, ChangHua 52345, Taiwan, ROC

URL:

[https://www.sciencedirect.com/science/article/abs/pii/S0957417408008348](%20https://www.sciencedirect.com/science/article/abs/pii/S0957417408008348)

3.Source: Department of Electrical and Electronic Engineering, Faculty of Engineering, Mie University, Tsu, Mie, Japan

URL:

[https://ieeexplore.ieee.org/abstract/document/934317/](%20https:/ieeexplore.ieee.org/abstract/document/934317/)

4.Source: Department of Information Systems and Operations Management, Business School, The University of Auckland, Auckland, New Zealand

URL:

<https://www.hindawi.com/journals/jhe/2021/5563651/>

5. Source: Chun-Cheng Lin, Jia-Rong Kang, Ding-Jung Chiang, Chien-Liang Chen

International Journal of Distributed Sensor Networks 11 (7), 595419, 2015

URL:

<https://journals.sagepub.com/doi/full/10.1155/2015/595419>

**Details of the algorithms/approach**

Genetic algorithm (GA) is one of the well-known techniques from the area of evolutionary computation that plays a significant role in obtaining meaningful solutions to complex problems with large search space. GAs involves three fundamental operations after creating an initial population, namely selection, crossover, and mutation. Furthermore, GA is a powerful search and optimization algorithm, which is computational model based on Darwin's biological evolution theory of genetic selection and natural elimination. The GA, however, takes a long computation time in some specific problems because of its iteratively adaptive process for evolution. Therefore, it is indispensable to improve GA for reducing the computation time and preventing local minima efficiently.

**Experiments & Results**

Graphical user interface, application

Description automatically generated

**If We put no of nurses, Population size, evolution grows and evolution loop size all equal 5**

**So our output will be …**

Table

Description automatically generated

**Advantages of our algorithm**

Many real-world optimization problems are hard to solve. They

are characterized by large solution spaces in mixed types of variables, by non-linearity in objective functions and constraints, and

by large computational costs of evaluating candidate solutions.

GAs are well-suited to solving many practical problems, and hence

their popularity in various engineering disciplines.

A deep understanding of the problem at hand is a must; it

guides the user in addressing several aspects in the implementation, such as selecting the appropriate encoding scheme and genetic operators, integrating problem specific information in the implementation, and reducing the cost of evaluations.

Although GAs have been in use for many years, they still provide ample opportunities for further research, particularly in adaptive control of parameter, hybridization with other techniques.

**Disadvantages of our algorithm**

the nurse may work morning and night shifts and when we have to that we write in the reasons for conflicts. Genetic algorithms are often criticized for being too slow. There are several disadvantages of using genetic algorithms. They can be expensive to implement. They can be difficult to debug. They can be difficult to optimize. Genetic algorithms do not scale well with complexity. That is, where the number of elements which are exposed to mutation is large there is often an exponential increase in search space size. This makes it extremely difficult to use the technique on problems such as designing an engine, a house or a plane.

**the algorithm behaves**

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions. Five phases are considered in a genetic algorithm.

* Initial population.
* Fitness function.
* Selection.
* Crossover.
* Mutation.

Fitness function and Crossover techniques are the two main features of the Genetic Algorithm.

**future modifications**

we want to use something like gradient descent in which our algorithm tries to select the best Population size, evolution grows and evolution loop size and the find the fittest schedule automatically without any human entries .