Jewel Journal of Scientific Research (JJSR) 6(1&2): 129–136, 2021
©Federal University of Kashere-Nigeria (Print ISSN: 2384 – 6267, Online ISSN 2756-651X)
jjsr.fukashere.edu.ng

On Invigilator - Exam Assignment Using PHP

[*1]Ahmed, H., [2]Muhammad, M. B. and [1]Yakubu, N.

[1]Department of Statistics and Operations Research, Modibbo Adama University, Yola, Adamawa State.

[2]Department of Mathematics and Computer Science, Federal University of Kashere, Gombe, Gombe State, Nigeria.

[*]**Corresponding Author**: hassanahmed.official@gmail.com; +2348032417537

Abstract

The development of weekly course timetables and/or test schedules in schools, colleges, and universities, as well as the timetabling of transportation services such as buses, trains, and airplanes, are all examples of timetabling problems. However, the most emphasis has been paid to educational timetabling, which has piqued interest. Accordingly, this paper is concerned with the invigilator-exam assignment problem. A web-based Invigilator Assignment System using PHP and mySql, consists of a mathematical model; a database storing the information and web-based user interface is constructed to solve the problem by providing an environment for a practical usage. Interestingly, the interaction between the environment and database is made possible by PHP and HTML. Consequently, MySql is used for database to put and retrieve records. The core of the system is the mathematical model developed for obtaining the exact solution. The examination time table for second semester for the Department of Statistics and Operations Research, Modibbo Adama University, Yola, for 2019/2020 academic session was used to assign invigilators. 18 Invigilators were available for 32 time slots. All invigilators were assigned successfully using the proposed system. The program showed that the time taken to generate the assignment dropped to 5 seconds. This proves efficient for invigilator-exam assignment using the proposed approach.

Keywords: PHP, mySql, invigilator-exam assignment, multi-objective programming, web development

Received: 16th Aug, 2021 *Accepted:* 11th Oct., 2021 *Published Online:* 25th Dec., 2021

Introduction

Timetabling problems range from construction of weekly course timetables and/or exam schedules in schools, colleges, and universities through to the timetabling of transport facilities such as buses, trains, or aircrafts. However, it is in the area of educational timetabling that most attention has been focused with continued interest (Johnson, 1993). The examination timetabling problem, as one of the educational timetabling problems, represents a major administrative activity for academic institutions is often a difficult and demanding process and it affects a significant number of people.

An invigilator is the person who supervises students during an examination. Invigilator-exam assignment is a problem of assigning invigilators to exams in such a way that there are no conflicts or clashes. Usually, an invigilator is assigned more than one exam, and more than one invigilator is needed for an exam. An invigilator should not be scheduled to invigilate more than once in the same time slot.

There may be preferences, inconvenient assignments, and pre-assignments related to invigilator-exam and/or invigilator-time slot pairs. Invigilator-exam assignment problem actually has a multi-objective structure like real-life decision problems. In recent years, multi-objective decision making has become a promising field, and attracted more and more researchers (Hu and Li, 2006; Huan *et al*., 2007; Kennedy and Tavana, 2006; Theetranont and Haddawy, 2007).

The examination scheduling problem is a highly complex combinatorial problem consisting of NP-complete subproblems (Bullneihmer, 2017). Invigilator-exam assignment is often done separately8–10 from the scheduling of examinations to time slots and rooms. In the literature there are so many researches related to other educational timetabling problems (Burke *et al*., 2002) though not so many for invigilator-exam assignments. A review about automated timetabling problems is given by Burke and Petrovic.

A model is developed for Carleton University in Ottawa to assign invigilators to exams. The system combines some problem-specific heuristics, a genetic algorithm framework and a simple user interface based on readily available software tools. Marti *et al.* (2000) modelled the problem as an integer program with a weighted objective function that combines a preference function and a workload-fairness function. It uses the concept of combining good solutions in order to yield a better solution. A scatter search solution procedure is used (Al-Yakoob, *et al.*, 2007) *to* consider two subproblems: the exam timetabling and the invigilator assignment problems.

A mixed-integer programming model is developed for invigilator assignment problem related to Kuwait University and it incorporates the invigilators' preferences for specific days and time periods. There are other papers related to exam scheduling that deal with some considerations of invigilators not by assigning them to the exams but by integrating their related parameters to the problem solution process. For instance, (Cowling *et al.*, 2002) gathered views from students and invigilators on their current exam or invigilation timetable and what they hope will be considered in the next generation of exam timetabling software. From the survey, they saw that invigilators are not satisfied with their current timetable and they would like to see more work being done on invigilation duties. Invigilator assignment problem relatively has not been well described and intensively worked on yet as much as the other exam scheduling problems. There may be a few reasons of that: First, it is not common to have invigilators at each institution during the exams though each educational institution has exam terms. Using grade curves, and thanks to honorary codes, institutions sometimes eliminate this duty.

There are also universities where students are allowed to have their exams over web services. And lastly, it is the possibility of being dominated by other dimensions of the exam scheduling process that probably makes invigilator assignment less important, comparing to the complexity of other dimensions of the problem. On the other hand, the manual solution of the timetabling problem usually requires many person-days of work. In addition, the solution obtained may be unsatisfactory in some respect. For these reasons, a considerable attention has been devoted to automated timetabling (Schaerf, 1999). The manual process of assigning invigilators may also involve many man hours. As the difficulty of the problem increases due to a large number of invigilators, exam and constraints, and invigilator preferences, an automated assignment system is often required. Besides, the problem becomes more difficult to obtain an optimal solution. With these reasons, while traditionally being considered in the operational research field, these problems later have been tackled with techniques belonging also to Artificial Intelligence. In recent applications, the analytic models to solve them are often embedded as a decision engine within the respective processes. In the past few years, the World Wide Web has facilitated, nurtured, and promoted a broad resurgence in the use of decision technologies to support decision-making tasks (Bhargava, 2007). Web-based decision support systems have attracted increasing attention from researchers and organizations (Dong et al, 2004). The integration of optimization engines and the Internet allows any user on intranet and extranet to execute an optimization model anytime and anywhere via web-based interfaces. It is worth mentioning that several web-based decision support systems have been developed and computer support is inevitable not only to solve the problem but also to provide a flexible and automated environment. Besides these features, these web-based systems would not require special installation in users' personal computers and would work across all computing platforms. Anyone with a computer connected to a Web browser would be able to access and use it. Because of its centralized nature, application improvements and new developments would be instantly available to users without the need of software redistribution and reinstallations. These mentioned advantages of Web environment are combined with multi-objective (criteria) decision-making process in several studies (Al-Aomar, 2008). However, in his review paper, Schaerf states that many authors believed that the timetabling problem cannot be completely automated with two reasons: There may be properties that make one timetable better than another that cannot easily be expressed in an automatic system. On the other hand, since the search space is usually huge, a human intervention may bias the search toward promising directions that the system by itself might be not able to find. Besides, automating the process of timetabling can avoid clashes but it may not consider other desirable factors, like personal preferences, unavailability, and so on. For the above reasons, most of the systems allow the user at least to adjust manually the final output.

For this study, a web-based automated system was developed with its user interfaces and database to solve invigilator-exam assignment problem is introduced using PHP, mySql, HTML and jQuery. The system optimizes objectives related to individual loads and tendencies and provides a flexible environment for data entry. The issue of taking invigilators' time preferences could be helpful to eliminate the drawbacks about personnel preferences and unavailability. This paper suggests a number of approaches and emphasizes three points: recent heuristics and evolutionary timetabling algorithms, multicriteria decision making, and cased-based reasoning approach.

Constructing the Invigilator Assignment System.

The context diagram of the Automated Invigilator Assignment System developed is given in Fig. 1.

Fig. 1. The context diagram of automated invigilator assignment system.

The main users of the system are invigilators and the administrator. Each user has access to the related information. The administrator controls activities related to security (who have access to which data), since he/she has access to the source code. The outcome of the system is also published to the students and the other users of the system.

Appropriate data files have been created related to exams, time slots, and invigilators. Availabilities of invigilators are incorporated in the system. Data are displayed in tables. The data handling sub-system enables the user with pop-up menus to enter, modify, and display all the appropriate data in which he/she has access. Interactions among the database, model, and users are enabled by web-based interfaces constructed with Macromedia Flash software. Data flow between interfaces and database is realized by HTML technology. Besides, PHP and mySql are used for the database transactions.

Mathematical model

A multi-objective mixed-integer programming model is developed for invigilator exam assignment problem. The model parameters, decision variables, constraints, and the objectives are defined as follows:

Parameters and sets:

$I = \{1, \ldots, m\}$ Invigilators

$J = \{1, \ldots, n\}$ Examinations

$T = \{1, \ldots, k\}$ Time slots

$U = \{(i,t) | \exists i \in I, \exists t \in T\}$ The set of unavailable time slots for invigilators

$V = \{(i,j) | \exists i \in I, \exists j \in J\}$ The set of pre-assignments for invigilators

g_j : The number of invigilators required for the j-th exam

a_j : Weight of the j-th exam

s_{jt} : 1 if exam j is scheduled on time slot t, 0 otherwise

m_{it}: 1 if invigilator i is not available on time t, $(i,t) \in U$

p_{ij} : 1 if invigilator i is preassigned to course j; 0 otherwise, $(i,j) \in V$

c_{it}: Assignment cost of i-th invigilator to t-th time slot

w_i: Load ratio of the i-th invigilator

Decision Variables:

$$y_{ij} : \begin{cases} 1, & \text{if the } i\text{-th invigilator is assigned to } j\text{-th exam} \\ 0, & \text{otherwise} \end{cases}$$

The positive decision variables r_1 and r_2 are used for linearization of min–max structure in the model. Then the general variant of the exam timetabling problem can be formulated as follows:

$$\min \sum_i \sum_t \sum_j c_{it}\ y_{ij}\ s_{jt} + r_1 + r_2$$

$$\text{(C1)} \quad \textstyle\sum_j y_{ij} s_{jt} \le 1, \qquad \forall (i,t),\ i \in I,\ t \in T$$

$$\text{(C2)} \quad \textstyle\sum_i y_{ij} = g_j, \qquad \forall j \in J$$

$$\text{(C3)} \quad y_{ij} = p_{ij}, \qquad (i,j) \in V$$

$$\text{(C4)} \quad \textstyle\sum_j y_{ij} s_{jt} = m_{it} - 1, \qquad (i,t) \in U$$

$$\text{(C5)} \quad \textstyle\sum_j y_{ij} a_j \le w_i r_1, \qquad \forall (i,t),\ i \in I,\ t \in T$$

$$\text{(C6)} \quad \textstyle\sum_j y_{ij} s_{jt} \le r_2, \qquad \forall i \in I,\ \exists t \in T$$

c_1 presents the clash-free requirement (no invigilator can be assigned to two exams at the same time). c_2 guarantees assigning required number of invigilators to each exam. Pre-assignments may be needed on assigning specific invigilators to the exams of specific courses and are performed via c_3. An invigilator is allowed to have restrictions on specific time slots which are considered as disabled for him. For instance, some invigilators are graduate students and they may have their own lessons or exams during the exam period. For this purpose, in c_4, m_{it} is defined as the parameter that takes value 1 when invigilator i is not available on time t, in other words when (i, t) ∈ U. To prevent the assignment procedure from doing this, 1 is subtracted from m_{it} to guarantee that there will not be such an assignment.

Reader may think that it would be enough to simply construct (C4) as:

$$\sum_j y_{ij}\ s_{jt} = 0, \quad (i,t) \in U.$$

m_{it} is undefined if invigilator i is available on time t. It has a value only for (i, t) ∈ U. Since the time period preferences of invigilators are gathered via web-based user interfaces, it is needed to store the related preference data first in a parameter called m_{it} for this case. Then the system uses the entered data to construct the mathematical model. It is assumed thatU $\neq \emptyset$. The other preference parameters, c_{it}, are directly used as they are entered into the web-based system. As shown in Table 2, line 3 refers unavailable time slot t for an invigilator i, these data correspond to m_{it} = 1 while 1 and 2 represent c_{it}.

On the other hand, the number of duties should be fair considering the number of times being appointed as invigilator c_5 and the number of duties on unavailable time slots c_6. c_5 and c_6 are the linearization of the:

$$r_1 = \min\{\max_i(\textstyle\sum_j y_{ij}a_j)\}$$

which minimizes the maximum individual load over the feasible solution set; and

$$r_2 = \min\{\max_i(\textstyle\sum_j y_{ij}s_{jt})\}$$

∃(i,t) which minimizes the maximum individual total assignment on undesired time slots. A feasible solution is one which completely satisfies all the hard constraints. The model has a multi-objective structure. The objectives are to minimize the assignment cost, the total maximum load of invigilators, and maximum total assignment on undesired time slots, simultaneously. Weighted sum method is used to scalarize them though with this kind of sum, the approach does not guarantee to find all pareto-optimal solutions due to the fact that the convexity conditions do not hold. And this is because of the 0–1 decision variable that the problem has. However, there are studies in the literature that solve the problem with this kind of scalarization.

mySql

The parameters of the mathematical model that are stored in a database which consists of the tables labeled as USERS, EXAMSCHEDULE, INVIGILATORS, COURSES, ASSIGNMENTS, and PREASSIGNMENTS. Some parameters are directly extracted from these tables by simple selection queries (e.g. number of invigilators, m, from INVIGILATORS table; number of required invigilators, c_{it} and exam weights, aj, from EXAMSCHEDULE table). But the parameters related to more than one tables such as pre-assignments, p_{ij} , and assignment cost, c_{it}, are required more complex selection queries to extract. The system enables invigilators to enter and modify their preferences with an interface. The generated solution displayed has an interaction with the database by PHP and mySql codes.

User interfaces

User interfaces allow users to define the system parameters and get related reports about assignment program. So, data for invigilator assignment problem are collected by the user interfaces and sent to database by using PHP and mySql queries, and then administrator starts to construct the mathematical model. Most of the parameters collected by the user interfaces are used to form the mathematical model to be solved.

There are four types of users in the system: administrator, invigilators, instructors, and students. When a user-id is entered, the related page will appear on the screen followed by checking of the status of being authorized. The system enables the administrator to enter, modify, and display the required data. Administrator can do the following tasks: Invigilator List construction, Exam Schedule and Pre-assignments entrances, Feasibility check, Run the system run to produce the assignment, and publish the outcome.

The invigilators can enter and modify their preferences and restrictions with the interfaces. They are asked to identify their intensity levels for undesirable timeslots as not preferred, strongly not preferred, and impossible as shown in Table 2. The entered values represent c_{it} parameter for not preferred, strongly not preferred wishes, and m_{it} for impossible wishes. The impossible wishes, are handled as hard constraints as seen in Sec. 2.1, (C4). Instructors and students can use the system only to see the published data. At any stage, the information processed to date can be displayed and/or printed to relevant users in a variety of formats. The updated database is processed by the administrators to produce invigilator-exam assignments.

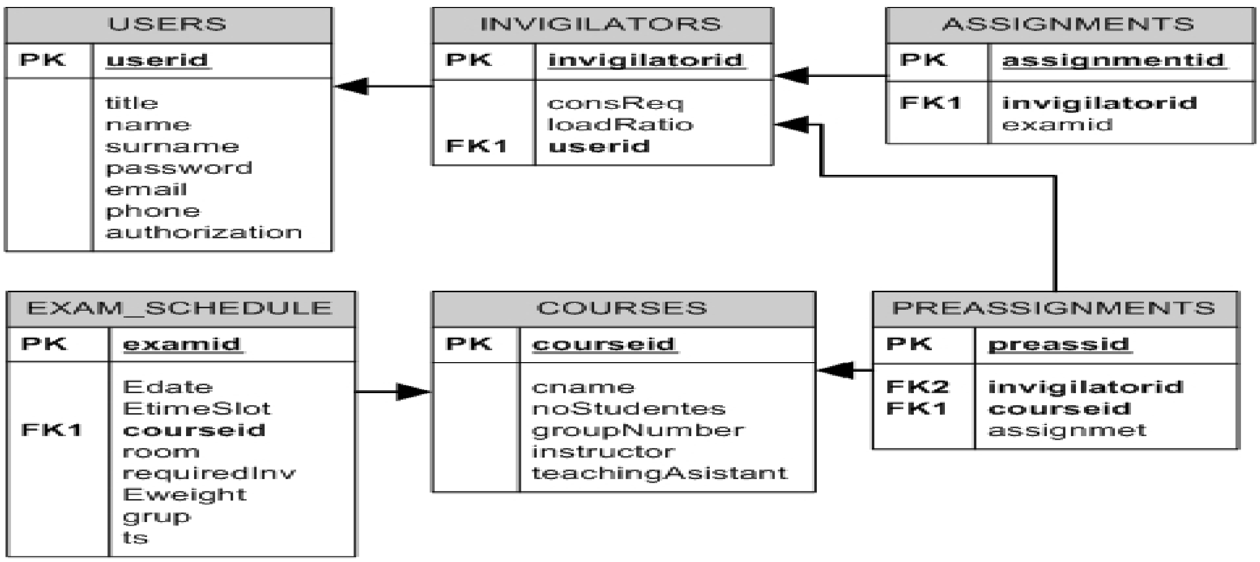

Figure 2. Tables and how their relationships run in the database.

Implementation

The proposed system is tested with data provided by the Modibbo Adama University, Yola, Statistics and Operations research Department. The academic year in the University consists of two Semesters as First and Second. At the end of each semester there is an examination period. The exam schedule is predetermined by the University Time Table Committee by assigning the common courses of the department to some specific time slots. Once the final exam schedule is determined by the University Time Table Committee, the department assign their invigilators to exams. The Examination office of the Department extracts from the general Time Table and make its own by allocating the uncommon courses to time slots. Then Invigilators are assigned to all courses.

The academic year for the tested data is Second semester 2019/2020. The proposed system under consideration has 18 invigilators. The Examinations took 18 days with 32 time slots. The task is to allocate optimally these invigilators to these 32 time slots using the optimal solution of the model. Table 1 below show the results of the invigilator assignment.

Table 1. Summary of number of times an invigilator is assigned from the proposed system.

Invigilator	Number of examinations
1	7
2	10
3	11
4	10
5	11
6	10
7	11
8	13
9	11
10	14
11	10
12	11
13	10
14	11
15	13
16	13
17	13
18	13

Conclusion

The existing manual invigilator-exam assignment system in the Department under consideration has some problems like time and man-power needed for constructing the assignment, difficulty in considering the invigilator preferences, and unavailability of tracking the changes in the system. Besides, it is always hard and almost impossible to reach the optimum assignment by solving the problem manually. A user-friendly decision support system based on a multi-objective mixed-integer programming model is introduced for invigilator-exam assignment problem with practical use. The system has the appropriate facilities for providing help to the users to implement an assignment schedule. Interactive tools are available to enter and or modify data. Using the new system, the time it takes for invigilator assignment dropped off from hours to few seconds. It would be most intriguing to see in future how other timetabling problems like course allocation and lecture time tables could be formulated mathematically and optimal solutions found. Consequently, programs using technologies like PHP, JAVA, or even Python could be used to design systems based on the optimal solutions to the mathematical problems to solve the problems.

References

Al- Aomar R. and Dweiri F. (2008), A customer-oriented decision agent for product selection in web-based services, *Int. J. Inf. Technol. Decis. Mak.* 7(1): 35–52.

Al-Yakoob S. M., Sherali H. D. and Al-Jazzaf M. (2007), A mixed-integer mathematical modeling approach to exam timetabling, *Comput. Manage. Sci.*, 7(1):19-46 doi:10.1007/s10287-007-0066-8.

Awad R. M. and Chinneck J. W. (1998). Proctor assignment at Carleton University, *Interfaces* 28(2): 58–71.

Bardadym, V. A. (1996). Computer-aided school and university timetabling: A new wave. *Lect. Notes Comput. Sci.* 1153: 22–45.

Bhargava, H., Power, D. J. and Sun, D. (2007). Progress in Web-based decision support technologies, *Decis. Support Syst.* 43(4): 1083–1095, doi:10.1016/j.dss.2005.07.002.

Bullnheimer, B. (1997). An examination scheduling model to maximize students' study time, *Lect. Notes Comput. Sci.* 1408**:** 78–91.

Burke, E. K., Jackson, K., Kingston, J. H. and Weare R. F. (1997). Automated University timetabling: The state of the art, *Comput. J.* 40(9): 565–571.

Burke, E. K. and Petrovic, S. (2002). Recent research directions in automated time-tabling, *Eur. J. Oper. Res.* 140(2): 266–280.

Chang, T., Meade, N., Beasley J., and Sharaiha Y. (2000). Heuristics for cardinality constrained portfolio optimisation, *Comput. Oper. Res.* 27(13):1271–1302, doi:10.1016/S0305-0548(99)00074-X.

Cowling, P., Kendall G. and Hussin, N. M. (2002). A survey and case study of practical examination timetabling problems, in *Proc. 4th Int. Conf.*

Practice and Theory of Automated Timetabling (PATAT2002), 258–261.

Dong, J., Du, H. S., Lai K. K. and Wang, S. (2004). XML-based decsion support systems: Case study for portfolio selection, *Int. J. Inf. Technol. Decis. Mak.* 3(4): 651–662.

Eom, S. and Kim, E. (2006). A survey of decision support system applications (1995–2001), *J. Oper. Res. Soc.* 57(11): 1264–1278.

Fernandez, C. J. and Trolinger, T. N. (2007). Development of a web-based decision support system for crop managers: Structural considerations and implemantation case, *Agron. J.* 99(3): 730–737.

Hu, C. and Li, S. (2006). Enhanced interactive satisfying optimization approach to multiple objective optimization with preemptive priorities, *Int. J. Inf. Technol. Decis. Mak.* 5(1): 47–63.

Huang. C.-H., Kao H.-Y. and Li H.-L. (2007). Decision on enterprise computing solutions for an international tourism, *Int. J. Inf. Technol. Decis. Mak.* 6(4): 687–700.

Johnson, D. (1993). A database approach to course timetabling. *J. Oper. Res. Soc.* 44(5): 425–433.

Mart R., Loureno H. and Laguna M., Assigning proctors to exams with scatter search, in *Computing Tools for Modeling Optimization and Simulation: Interphases in Computer Science and Operations Research*, eds. M. Laguna and J. L. Gonzlez-Velarde, (Kluwer, Boston, USA, 2000), pp. 215–227.

Reis L. P. and Oliveira E. (2000). A language for specifying complete timetabling problems, *Lect. Notes Comput. Sci.* 2079**:** 322–341.

Schaerf, A. (1999) A survey of automated timetabling, *Artif. Intell. Rev.* 13(2): 87–127, doi:10.1023/A:1006576209967.

Shiminzu Y., Tanaka Y. and Kawada A. (2004). Multi-objective optimization system, MOON2 on the Internet, *Comput. Chem. Eng.* 28(5): 821–828.

Tavana M. and Kennedy D. T. (2006). N-Site A distributed consensus building and negotiation support system, *Int. J. Inf. Technol. Decis. Mak.* 5(1): 123–154.

Theetranont C. and Haddawy P. (2007). Integrating visualization and multi-attribute utility theory for online product selection, *Int. J. Inf. Technol. Decis. Mak.* 6(4): 723–750.

White G. M., Xie B. S. and Zonjic S., Using tabu search with longer-term memory and relaxation to create examination timetables, *Eur. J. Oper. Res.* 153**:** 80–91, doi:10.1016/S0377-2217(03)00100-0.