Proceedings of the 2020 Annual General Donald R. Keith Memorial Capstone Conference
West Point, New York, USA
April 30, 2020
A Regional Conference of the Society for Industrial and Systems Engineering

An Optimization Approach to the Make-Up Final Exam Scheduling Problem with Unique Constraints

Lindsey Blanks, Elizabeth Frakes, Koby Hinnant, Swathi Samuel, and John Dulin

Operations Research Program
United States Air Force Academy, Colorado Springs, CO

Corresponding Author: eliz.frakes@gmail.com

Author Note: Blanks, Frakes, Hinnant, and Samuel are senior cadets at the United States Air Force Academy, collectively working on this project as part of a year-long operations research capstone course. The student authors would like to extend thanks to the advisors involved with this project as well as the client organization, United States Air Force Academy Dean of Faculty Registrar. Please forward all questions to Elizabeth Frakes' email above. *This paper was short selected for "Top Paper" of the proceedings.*

Abstract: Final exam scheduling is typically a simple task. Exam scheduling at the United States Air Force Academy, however, is unique in that it must abide by a number of institutional constraints. Some include: ensuring all 4,000 cadets complete their exams within one week, assigning a longer time block for final exams than regular class meetings, and limiting the number of exam periods per day. The Dean of Faculty Registrar's office must also accommodate cadet absences during the scheduled final exam week on short notice. Addressing these issues is currently a reactive process—cadet absences are identified, and then the Registrar creates adjusted schedules by hand. In this study, we create a data-cleaning VBA module for input data and use Xpress software to execute one of three versions of a scheduling optimization model. The model reduces the scheduling task time from several days to less than five minutes.

Keywords: Scheduling, University, Optimization, Unique Constraints, United States Air Force Academy, Final Exam

1. Introduction

Historically, scheduling problems like assigning employees to shifts or students to specific classes have been adequately solved by hand. Today, there are numerous algorithms and models that exist that can immensely decrease the total time spent on the simple, but often tedious task. At the United States Air Force Academy (USAFA), the Dean of Faculty Registrar's office (DFR) must schedule all cadets to take all their finals in a single week, subject to strict USAFA constraints and in addition to the other work they are responsible for throughout a semester.

Previously, when individual students or teams have been gone for at least a portion of the final exam week, DFR has had to reschedule these cadets by hand. This process essentially included writing out all of the absentees and their finals on a white board, and then beginning the laborious task of creating a feasible exam schedule that would work for all students and faculty. For example, in the fall semester, collegiate football bowl games can occur during USAFA's final exam week. The bowl games are announced in early December, potentially giving DFR as little as one week to create a make-up schedule for up to several hundred students. Currently, there is no "optimization" with this process—the first schedule that meets all requirements (i.e. all students' exams are scheduled) is the final product, with no check as to whether there may be a better alternative. With all of the other important jobs assigned to DFR, a better process and tool would allow them to focus their time on more pressing issues.

1.1 Problem Statement

The overall goal of this project was to create an easier way for DFR to schedule make-up exams for the cadets that would miss any portion of the primary final exam week. Instead of DFR creating the schedule by hand with limited time and resources, our team has streamlined this process with a tool (see section 2.3) that completes the task significantly faster than scheduling the exams by hand. This tool allows the user to upload specific cadet information and then automatically schedule selected cadets for make-up exams. It can accommodate several groups of cadets, and includes a user-friendly interface so that multiple users will be able to manage and operate it without the need of a technical background.

By identifying the best process to both 1) select which cadets will be gone during the scheduled final exam time, and 2) reschedule these cadets while following the predetermined final exam rules set by the institution, we make the make-up final exam scheduling process less time consuming and tedious.

1.2 Related Work

Because DFR's scheduling problem must account for different user input each time the model is run (e.g. different teams requiring rescheduling), and the solution generated will need to be one that is not only feasible, but also reasonable, we have focused on an optimization approach that captures the institutional constraints. The importance of generating a solution that is not only numerically possible, but actually executable and sensible, led us to many related works with approaches that involved optimization followed by subsequent steps to ensure applicability. Dimopoulou and Miliotis (2001) designed a flexible solution for college course and exam timetable scheduling that allows the user to add specific assumptions and then uses integer programming (IP) to assign courses to time slots and rooms to create an initial result. Next, a heuristic approach improves the result until a reasonable schedule is reached. The final solution depends heavily on the user's input constraints and decision making rather than just the list of courses and students. Zhaohui and Lim (2000) also found that in order to satisfy all constraints in their scheduling problem for the University of Singapore, they needed to use multiple methods including a knapsack algorithm and an iterative greedy heuristic to set up a graph-coloring solution. This led us to include a series of steps following the initial optimization to ensure our generated schedule adequately addresses "soft" constraints.

Several previous studies also relied on an optimization of their solution followed by subsequent steps that relax certain "soft" constraints in an effort to further improve the initial result. Qu and Burke (2009) provide insight into the process of determining which constraints to loosen or relax depending on the problem. The study conducts analysis on whether or not hyper heuristics can meet the hard constraints, while still optimizing the soft ones. Akbulut and Yolmaz (2013) also approach exam timetabling with sets of constraints to restrict the model, but they adjust their model so that each constraint contains a coefficient used to calculate the efficiency of each algorithm. Rather than labeling a constraint as necessarily "hard" or "soft," these authors allow the model to essentially pick for them based on the value of the particular coefficient. These models have informed our approach in addressing a large range of institutional constraints at USAFA.

2. Methodology

We have created an optimization model using both Xpress and VBA to address our research question. There are three potential outcomes for a scheduling solution: optimal and practical, optimal and not practical, and infeasible. Ideally, the model will return both an optimal and practical solution every time, but due to the variable nature of the problem, that may not always be the case. Therefore, we use a series of heuristic steps to ensure the practicality of the solution.

The user input includes an Excel file pulled from USAFA's student information system with records for all students' finals. Using this input, a user interface developed using VBA allows for selection of the cadets that will be absent during the regular final exam week and subsequently cleans and formats the data into a table of student-final combinations. Xpress then uses this table of student-final combinations to generate a schedule, with the objective function minimizing the number of offerings per course (see Equation 1). To frontload the schedule as much as possible, different weights have been added to the objective function as a "penalty" for each additional period of make-up exams. These penalties are in place to help with the practicality of the model, reducing the total number of scheduled periods and the spread of final offerings. Given the large number of hard constraints and variance in the data, it is difficult to find a practical solution with every possible subset of user inputs. For this reason, subsequent heuristic steps are executed to ensure the resulting schedule is both feasible and reasonable by relaxing some of the "soft" constraints. These steps include testing whether or not the distribution of schedules is balanced. For example, it prevents a schedule where the majority of cadets are scheduled in the first four periods and one or two cadets have finals in the eleventh and twelfth periods if we could instead relax a constraint and move those finals up several periods while spreading out the earlier finals. Furthermore, these heuristics look to see if slight changes or switches in the "soft" constraints can have a significant impact in the overall schedule. Allowing the student with the most finals to take more than two per day could greatly reduce the number of periods if everyone else only has one or two required finals.

2.1 Student Data

To run the optimization model, an input of student data, extracted from CAMIS and loaded into a properly formatted Excel spreadsheet, is needed. CAMIS is the current system being used by USAFA for scheduling, grade inputs, and data housing of cadet information. Although this is the USAFA scheduling system, similar data can be used by other universities if manipulated to mimic the same format as described in this study. Data for this project was collected from the 2019-2020 Academic Year for the Fall Semester. The data set includes over 14,000 observations and seven fields of data including the following: PID (unique student identifier); Class Year (the graduation year of the student); Sq (the current squadron of the student); Majors (the declared undergraduate major the student is pursuing); Course (the course ID of the final the student is taking); Final Period (the pre-determined period assigned to that student for the associated course final); Varsity Sport (student's varsity team name will be here if applicable, otherwise there will be nothing).

2.1.1 Data Cleaning

Once the data is extracted into the format above, a VBA Form is used in order for DFR to select the students to include in the Make-Up Final Exam Schedule. A VBA module is used to load the data into an array and add or delete cadets as needed. The user can include teams as a whole or specify individuals who may be missing the primary finals week for other circumstances. The Form will allow the user to select which final periods the students will miss in order to allow students to take as planned any finals they will not be missing. Once all the cadets are loaded into the array the data will be reformatted into another Excel Spreadsheet (Table 1) to be used as the input for the Xpress optimization model. If a cell is greater than 0, the associated student is taking that many finals administered by the department listed. The "courses" listed in Table 1 are shortened to only show the department; the model is able to schedule multiple courses within the same department in the same final period to reduce the manning needed for finals.

Table 1. Formatted Student Data

PID	AeorEngr	Math	AstroEngr	History	Biology	ECE	Chem	EngrMech
28652	1	1	0	0	0	0	0	0
28757	0	0	1	1	0	0	0	0
29030	0	0	0	0	1	1	0	0
29167	0	1	0	1	0	0	1	0
29428	1	0	0	0	0	0	0	0
29581	0	0	0	1	0	0	0	1
30164	0	0	0	1	0	0	1	0

2.1.2 Output Data Format

A second data file was also provided by DFR, indicating the number of students enrolled in each course's finals periods. The left two columns of this table designate the department and individual course, then their respective rows show which period(s) in which they are scheduled to offer a final and the number of students enrolled in that period. Although this sample schedule itself is a subset of the normal final exam week scheduling, this format for the enrollment table serves as a template for the subset of data observations needed in the make-up exam schedule model. This data serves as a guide for the departments for where and when they will need faculty to proctor exams.

2.2 Assumptions

The following are assumptions made in order to limit the scope of the model.

1. No student preferences: Make-up exams will be scheduled to minimize the number of additional offerings for a specific course; student preferences for make-up final exams are not considered.
2. No make-ups if able: If a student is present during any part of finals week, they will take exams that they can at the originally scheduled time and those finals will be eliminated from the model.
3. Grouping by department: Finals will be grouped and scheduled by department. The departments will assign qualified proctors within each period to make sure questions are able to be answered.
4. Time limit: There will be at most 12 make-up periods (4 days of finals) offered

2.3 Model

To schedule make-up exams for USAFA we created an optimization problem that relies on user input based on the students missing final exam week. After the program is initially run, a series of heuristic steps are executed in VBA to ensure that the solution is not only feasible but also sensible, based on the constraints deemed most important by DFR.

2.3.1 Optimization Model

The optimization model is designed uniquely to minimize both a course's final offerings and the total final periods overall, both aspects of the problem deemed to be most important by DFR. The objective function itself takes into account the first part of the problem, minimizing individual offerings per course. However, the penalties (represented as objective function coefficients) within this function that relate to the total number of final periods are very large, meaning that the total number of periods is simultaneously being minimized. The optimization model includes the following features:

Decision Variables:

$$X_{s,c,p=} \begin{cases} 1 \; if \; Student \; s \; has \; Course \; c \; scheduled \; for \; Period \; p \\ 0 \; othwerwise \end{cases}$$

$$Y_{c,p} = \begin{cases} 1 \; if \; Course \; c \; is \; scheduled \; for \; Period \; p \\ 0 \; otherwise \end{cases}$$

Inputs:

$$Students = array \; of \; student \; names, s = \{1 \ldots S\}$$
$$Courses = array \; of \; Course \; names, c = \{1 \ldots C\}$$
$$Periods = array \; of \; final \; exam \; periods, p = \{1, \ldots, 12\}$$
$$qmatrix = two - dimensional \; array \; of \; (Students, Courses)$$
$$weights \; = \; two - dimensional \; array \; of \; (periods, weights)$$

Objective Function:

$$minimize \; \sum_c \sum_p weights_{p,w} * Y_{c,p} \quad (1)$$

The objective function within our model is used to minimize the final exam offerings per course. This is because of the additional hours and resources that departments must invest for each additional final exam they must write and proctor. There are also weights associated with each final exam period, which steadily increase as the period is later in the week. This helps to "frontload" the final exam week to help accomplish finals in a shorter amount of time.

2.3.2 Constraints

An important part of creating this optimization model was ensuring that all constraints were defined and written correctly to achieve the desired outcome. The first three constraints are the "hard" constraints. These, as requested by DFR, absolutely must be met. The remainder of the constraints were defined as "soft" constraints. Pending feasibility, these constraints would be the first to be relaxed or adjusted if a change needed to be made.

A seemingly obvious but important constraint is one that ensures that every student will take every final assigned to them over the course of the make-up final exam week. The q-matrix input parameter is a particular student and course combination with information on the number of finals a student takes in each course. In order to assure this, each binary variable ($X_{s,c,p}$) representing a particular student (s), course (c), and period (p) combination should add up to the number of finals the student's q-matrix variable has when summed over p.

$$\sum_p X_{s,c,p} = qmatrix_{s,c} \; \forall \; s, c \quad (2)$$

Another important constraint to specify in the model is the fact that a student may not be two places at once. This means that a cadet may only take one final in each period. In the model, this means that for each unique student-period combination, the sum over the courses must be less than or equal to 1.

$$\sum_c X_{s,c,p} \leq 1 \; \forall \; s, p \quad (3)$$

Since our model is trying to minimize the number of final periods offered, we introduce a new binary variable ($Y_{c,p}$) that represents each course-period combination. The variable equals one when the course's final is being offered in that period, and 0 when it is not. In order to connect the two binary variables, the following constraint was introduced. This constraint essentially "turns on" the Y variable once any student is assigned to that course-period combination. The Y binary variable is multiplied by 5,000 because this exceeds the total enrollment of USAFA, so it essentially allows up to 5,000 students to be scheduled for that particular course-period combination.

$$\sum_s X_{s,c,p} \leq Y_{c,p} * 5000 \quad (4)$$

Cadets are also limited in the number of finals they are able to take in one day. There are three final periods each day, so for periods 1 to 3, periods 4 to 6, periods 7 to 9, and periods 10 to 12, a given student's $X_{s,c,p}$ variables must not exceed two when summed over the course and period. After discussing this constraint with DFR, these constraints were labeled as "flexible" due to the fact that they may be relaxed in order to make the solution feasible and reasonable. These relaxations are executed automatically in the code depending on the particular inputs.

$$\sum_c \sum_{p=1}^{3} X_{s,c,p} \leq 2 \ \forall\ s \quad (5)$$
$$\sum_c \sum_{p=4}^{6} X_{s,c,p} \leq 2 \ \forall\ s \quad (6)$$
$$\sum_c \sum_{p=7}^{9} X_{s,c,p} \leq 2 \ \forall\ s \quad (7)$$
$$\sum_c \sum_{p=10}^{12} X_{s,c,p} \leq 2 \ \forall\ s \quad (8)$$

A final set of constraints to consider when coding the model is the requirement that cadets do not take an evening final followed by a morning final the next day. Since there are three final periods per day, this means that a cadet may not be assigned to periods 3 and 4, periods 6 and 7, or periods 9 and 10. This requirement is again categorized as "flexible" and may be relaxed after an initial optimization phase of modeling if necessary.

$$\sum_{p=3}^{4} X_{s,c,p} \leq 1 \ \forall\ s, c \quad (9)$$
$$\sum_{p=6}^{7} X_{s,c,p} \leq 1 \ \forall\ s, c \quad (10)$$
$$\sum_{p=9}^{10} X_{s,c,p} \leq 1 \ \forall\ s, c \quad (11)$$

2.3.3 Alternate Model Constraints

Although the first step of our model involves straight optimization of the schedule, the solution generated may not always be the most reasonable, and in some cases where many students are missing exams, it may take an extremely long time to generate the optimal solution. For this reason, two other versions of the model were created to help eliminate unreasonable solutions quickly and shorten the time the model takes to generate a solution.

Version A is the optimization model described above and coded in Xpress. There are no additional steps or constraints other than those required by DFR described above. This model executes in just a few seconds for up to 150 students. Version B of the model is the Xpress optimization model (Version A) with an added step that schedules the two most "densely populated" departments automatically. The VBA code automatically sorts departments based on the number of students taking make-up finals in their courses following the initial data load of selected cadets requiring make-up exams. Using this information, the Xpress code will pull the two departments that have the most students, or the most density, and will schedule them automatically for periods one and two. For example, if the two departments with the most finals were Math and Management, the following constraints would be added to the Xpress Model code: *Y("Math",1) = 1* and *Y("Mgt",2) = 1*. These constraints are added into the Xpress code automatically using a simple if-statement, so it requires no additional work for the user. Version B of the model schedules up to 180 students in a matter of seconds.

For more than 180 students, we recommend Version C of the model. Version C is the Xpress optimization model with the added step that schedules the two densest departments (Version B) and an additional constraint that automatically schedules all foreign language final exams for the same period, period 5, on the second day. Foreign language exams may all be scheduled at the same time because each student will only ever be taking one of these exams in one department, so there is no reason to check for conflicts with that constraint. Period 5 was chosen because all students may be scheduled this period without concern for the morning-night double final constraint (constraints 8, 9, and 10). Version C will include all previous constraints with the addition eight language constraints using a simple if-statement as described previously, e.g. *Y("Chinese",5) = 1.*

This final version of the model can schedule up to 200 students in less than five minutes and 230 students in less than 25 minutes. Based on client meetings, the maximum number of students that miss all of finals week occurs when the football team, cheerleading team, and Drum and Bugle Corps are all away for a Bowl Game—by conservatively assuming that all members of all teams will be able to travel for all of exam week, the number of students missing comes out to around 200. The additional 30 students were included to account for any students missing for individual reasons. The student cut-off numbers for each version of the model were determined by testing various inputs into the model and testing where additional students had significant impacts on the overall execution time; each version of the model was consistently able to schedule students up to their designed cut-off in less than five minutes.

3. Analysis

Initial tests of the model were run on simple "test" user inputs. In running these tests we could look at microcosms of a schedule and determine whether there was a feasible solution generated and, if so, whether the generated solution was practical. At that point we began testing the model with cadet data from fall 2019. We tested each version with samples of 50, 120, 150, 180, and 200 students, using the same students for each version. Version C of the model was also executed with an input of 230 students, our designed maximum. The model scheduled 200 students in under five minutes. Furthermore, all generated schedules were consistently free of conflicts. Additionally, without relaxing any constraints, the total number of exam periods offered for all tests never exceeded 8 periods, allowing all students to complete their finals in under three days.

Although our testing never required the relaxation of "soft" constraints to reach a feasible solution, we have implemented a simple heuristic that relaxes one constraint (4 through 10, in reverse order) at a time and re-runs the model to allow the user to determine if a better solution can be obtained than produced by the optimization alone. In other words, after the optimization is run the solution is output and recorded. If any questions remain about its reasonability, constraint 10 is eliminated, the model rerun and output reviewed. Constraint 9 is next, and so on, until DFR agrees that the produced schedule is acceptable.

4. Conclusion

The goal of this project was to create an easier way for DFR to schedule make-up final exams for cadets who are absent during all or portions of USAFA's final exam week. Our VBA user interface and Xpress models have significantly reduced the time spent on this task while continuing to create a reasonable schedule for the make-up exams. Instead of creating a schedule by hand on a whiteboard over the course of several days, the DFR staff can now select the cadets that will be absent, initiate the model, and output a schedule in less than an hour. To schedule 120 cadets or fewer, Versions A, B, and C take roughly 20 seconds to output a schedule. Creating a feasible and reasonable schedule for 120 to 200 cadets can be completed in 2 minutes by version C and 5 min by version B. Finally, up to 230 cadets can be scheduled in just 25 minutes by version C. Based on conversations with DFR it seems highly unlikely that so many finals would need to be rescheduled, nonetheless the model can handle it and output a reasonable solution in less than half an hour.

5. References

Akbulut, A., & Yilmaz, G. (2013). University Exam Scheduling System Using Graph Coloring Algorithm and RFID Technology. *International Journal of Innovation, Management and Technology*, *4*(1), 66.

Allen, T. (1974). *Vanishing wildlife of North America.* Washington, D.C.: National Geographic Society.

Boorstin, D. (1992). *The creators: A history of the heroes of the imagination.* New York: Random House.

Devitt, T. (2001, August 2). *Lightning injures four at music festival*. Retrieved January 23, 2002, from The Why? Files: http://whyfiles.org/137lightning/index.html

GVU Center. (n.d.). Retrieved August 8, 2000, from 8th WWW user survy: http://www.cc.gatech.edu/gvu/usersurveys/survey1997-10/

Harlow, H. (1983). Fundamentals for preparing psychology journal articles. *Journal of Comparative and Physiological Psychology, 55*, 893-896.

Henry, W. I. (1990, April 21). Making the grade in today's schools. *Time, 135*, pp. 28-31.

Kalette, D. (1986, July 21). California town counts town to big quake. *USA Today, 9*, p. A1.

Nicol, A., & Pexman, P. M. (1999). *Presenting your findings: A practical guide for creating tables.* Washington, D.C.: American Psychological Association.

Searles, B., & Last, M. (1979). *A reader's guide to science fiction.* New York: Facts on File, Inc.

The safety of genetically modified food crops. (2002, February). Retrieved March 22, 2005, from Health Canada: http://www.hc-sc.gc.ca/english/protection/biologics_genetics/gen_mod_foods/genmodebk.html