

University of Arkansas, Fayetteville
ScholarWorks@UARK

Industrial Engineering Undergraduate Honors
Theses

Industrial Engineering

5-2013

Decision support tool for Operations Management course and instructor scheduling

Morgan Ulesich

University of Arkansas, Fayetteville

Follow this and additional works at: <http://scholarworks.uark.edu/ineguht>

Recommended Citation

Ulesich, Morgan, "Decision support tool for Operations Management course and instructor scheduling" (2013). *Industrial Engineering Undergraduate Honors Theses*. 4.

<http://scholarworks.uark.edu/ineguht/4>

This Thesis is brought to you for free and open access by the Industrial Engineering at ScholarWorks@UARK. It has been accepted for inclusion in Industrial Engineering Undergraduate Honors Theses by an authorized administrator of ScholarWorks@UARK. For more information, please contact scholar@uark.edu.

Decision Support Tool for Operations Management Course and Instructor
Scheduling

An Undergraduate Honors College Thesis

in the

Department of Industrial Engineering
College of Engineering
University of Arkansas
Fayetteville, AR

by

Morgan Rae Ulesich

This thesis is approved.

Thesis Advisor:

A handwritten signature in blue ink, appearing to be "E. H. R.", is written over a horizontal line.

Thesis Committee:

A handwritten signature in blue ink, appearing to be "E. H. R.", is written over a horizontal line.A handwritten signature in black ink, appearing to be "A. H.", is written over a horizontal line.

1. Problem Definition

Each summer, the Operations Management staff at the University of Arkansas wrestles with the scheduling of the upcoming course and instructor schedules for the next year. The current process of sticky notes and a calendar takes weeks to complete. The staff began wondering if there was a better strategy to complete the process. This was not an easy tasking considering that the scheduling process is more complicated than other programs.

The degree program is a nontraditional curriculum. It is nontraditional in the fact that the majority of the students are working professionals. A nontraditional program requires a nontraditional scheduling of the classes. The program is online based that requires few onsite, night courses to complete a Master's of Science in Operations Management degree. On average, twenty to thirty online courses are offered a term and two to four courses are offered at each of the five live sites over five terms within the academic year. The onsite locations include Fayetteville, AR, Camden, AR, Millington, TN, Little Rock, AR, and Hurlburt Field, FL.

With over sixty instructors, twenty-five courses, five sites, and five terms, the scheduling process gets very complicated and tedious. The scheduling must also account for the many restrictions that the program encompasses. For instance, students, instructors, and the staff have preferences on courses, terms, and sites that must be considered when developing the schedule for the next academic year. The need for a more efficient process was needed now, more than ever.

Because the Operations Management program is tied in with the Industrial Engineering department at the University of Arkansas, a mathematical model to assist with the scheduling was determined as the most viable option. We have a set of discrete decision variables for the problem. Since most of the Operations Management staff doesn't have a background in Operations Research, a decision support tool was also developed that required no previous knowledge to run.

2. Literature Review

Most of the past solutions for schedule optimization have focused on classroom assignment rather than instructor assignment. A study done by Steinbrugge Fant and Nahas focused on classroom assignment by using a genetic algorithm. The algorithm included a nonlinear program with relaxed constraints and a genetic algorithm was used to enhance the search. This technique proved to be successful when tested on Portland State University (2002). Since most of the on-site classes within Operations Management are offered during the night, the majority of classrooms are available; therefore, classroom assignment is not in the scope of this project.

Many different solutions to class-scheduling problems have been researched throughout the years. Most use some form of an optimization algorithm. A heuristic is a problem solving approach based on experienced techniques to solve the problem. Different heuristics have been developed to solve the problem.

Local search heuristics can be used for relatively simple problems. Studies have been done to utilize the simple heuristics. Nandhini and Kanmani designed a mathematical model and a local search heuristic to optimize the problem of course timetabling. The study used a combination of hard and soft constraints to satisfy. The

local search heuristic generated all the possible solutions to the course timetable problem. The algorithm required multiple generations to fully optimize the solution (2010). These types of problems only schedule the classes based off of student preference, and how many students would like to take each class. It doesn't consider faculty, location, or time. Based off these constraints, all of the possible options were enumerated. Optimization software ranked the classes based off of preference, and the faculty assigned the classes based off these rankings (Sampson, Freeland, & Weiss, 1995). Other approaches that have been used are complete enumeration step and a greedy heuristic. These techniques were used for Navy-C Schools. With the combination of the complete enumeration and greedy heuristic, it was found to be just as successful as the optimization software that was previously being used (Apte et al, 1998). The complete enumeration is a very time consuming process, and the greedy heuristic doesn't always produce an optimal solution to the problem.

The most applicable heuristics that have been developed include mathematical modeling as a step. Mathematical models have the ability to include multiple variables and constraints. These models can be programmed to optimize the objective function. There are numerous types of programs, and each type has strengths and weaknesses. A study done by MirHassani solved the scheduling problem with a binary integer program. This study proved to provide comparable results to other studies. However, the search process for the optimal solution was very long. (2006). Multi-stage integer program techniques have proven as suitable options to solve the classroom-scheduling problem. Each previous stage was used to solve the sequential stages (Al-Husain, Hasan, and Al-Qaheri, 2010).

Many scheduling problems have been shown to be best solved with linear programming. This type of optimization has a linear objective function, linear constraints, and all variables are constrained. A study done at the University of Qatar compared developing a linear program (LP) to the typical manual trial and error class scheduling procedure. Even though the LP took longer to develop, it proved to be better in the long run than the trial and error. The LP could solve the problem easily in the future and was used in conjunction with an excel spreadsheet to keep the schedule. Based on the utilization, developing the LP was a superior solution to the typical manual trial and error class scheduling procedure (Nabih El-Bahrawy, 1998). More complicated programs have also been developed to solve the classroom assignment problem. A mixed integer program (MIP) was used by Kuwait University to schedule classes and exams. Because of the complexity of this type of program, preferences of the students and faculty and ratios of the gender of students within the class were considered. The program was solved with CPLEX, an optimization software package (Al-Yakoob & Sherali, 2006, 2007). To solve complicated MIPs, a branch and bound solution method has proven to be the most effective. Purdue University uses the method to schedule almost 2,500 classes (Rudova, Muller, & Murray, 2010). The issue with this solution is that only someone that is fluent in CPLEX can solve the MIP.

Solving a scheduling problem with CPLEX is effective; however, different options of using the program are more applicable. For instance Miranda, Ray and Robles used an integer-programming model to create the optimal schedule, but then implemented the system into a web environment. The web implementation required massive amounts of programming; however, the implementation was very effective. The scheduling

process time decreased from weeks to a couple of hours (2012). Another option for the decision support system as demonstrated by Dinkel, Mote and Venkataramanan would be to combine multiple software packages to ease the use of the program (1989). However, this could run into the same issue of finding a software package that all of the users are fluent in. A decision support system (DSS) that's used in conjunction with the mathematical program must have powerful Operations Research procedures; as well as easy interactive tools that the user can efficiently use (Ferland and Fleurent, 1994). The DSS has to be cohesively used with the program, flexible and dynamic (Al-Qaheri, Hasan, and Al-Husain, 2012). These are things that need to be considered when developing a DSS to work with the program.

This project will develop an integer program (IP) based on the integer variables and linear constraints that will optimize the scheduling of classes within the Operations Management degree program based on location/online, instructors, courses offered, instructor preference, and staff preference. An optimization software package, CPLEX, will be used to produce the optimal solution. These two steps will be combined in a decision support tool within a spreadsheet that will be user friendly and easy to use. The spreadsheet will have the ability to be personalized based on the upcoming academic year. The tool can be used in years to come, and save the Operations Management staff valuable time.

3. Methodology

Creating the model wasn't simple, and therefore took a significant amount of thought. After speaking with the Operations Management staff, the purpose was outlined. The

purpose is to assign the instructors to courses while considering both instructor and staff preference. Because of the complexity of the scheduling of the program scheduling, there are numerous constraints and variables that needed to be considered. These constraints will be discussed in detail when the model is described in depth.

The model developed is general and could be applied to any scheduling instance; however, the constraints are tailored to the needs of the Operations Management program. The model is an IP because all of the variables are integers and the objective function and constraints are linear..

3.1 Sets, Parameters, and Variables

The sets included are as follows: instructors, i , courses, j , terms, k , and sites, l . The instructors are the instructors that are available to teach any course over the entire year. The courses are the courses that will be offered over the year. There are five terms over the year. The terms are each eight weeks in length. There are five sites (not including online) that the Operations Management program is offered at. The sites include Fayetteville, AR, Camden, AR, Millington, TN, Little Rock, AR, and Hurlburt Field, FL.

There are many parameters that the model considers. The following are all positive, integer parameters. The satisfaction of instructor, i , teaching course, j , that the Operations Management determines is defined as s_{ij} . The minimum and maximum number of live courses offered each term at each site is defined as a and b respectively. The minimum and maximum number of online courses offered each term is defined as d and e respectively. The maximum number of courses instructor, i , can teach in term, k ,

is defined as c_{ik} . This parameter is determined by the instructor's preference. The maximum number of online sections of course, j , available per term, k , is defined as f_{jk} . The maximum number of times that site, l , can host a single course is defined as p_l . The maximum number of sections that course, j , can be offered in term, k is defined as o_{jk} . The minimum number of sections that course, j , can be offered in term, k , is defined as g_{jk} .

The following parameters are binary. For each parameter, if the statement is true then the parameter has a value of 1. If instructor, i , is qualified to teach course, j , then parameter $r_{ij} = 1$. If instructor, i , is qualified to teach at site, l , then the parameter $t_{il} = 1$. If instructor, i , is qualified to teach online then the parameter $n_i = 1$. If course, j , is a required course for completion of the Operations Management Program then the parameter $q_j = 1$. If course, j , can be offered online is defined as m_j .

There are two variables in the model, both of which are binary and assignment variables. If instructor, i , teaches course, j , term, k , at site, l , then the assignment variable of x_{ijkl} is assigned a value of one. The second variable, y_{ijk} , is assigned a value of one if instructor, i , teaches course, j , term, k . Because there are two assignment variables, it is easy to add constraints to either live or online courses.

Sets

i = Set of instructors ($i \in V_1$)
 j = Set of courses ($j \in V_2$)
 k = Set of terms ($k \in V_3$)
 l = Set of sites ($l \in V_4$)

Parameters

s_{ij} = Satisfaction when instructor, i , teaches course, j
 a = Minimum number of live courses offered each term at each site
 b = Maximum number of live courses offered each term at each site
 d = Minimum number of online courses offered each term
 e = Maximum number of online courses offered each term

c_{ik} = Maximum number of courses instructor i can teach in term, k

p_l = Maximum number of times site l can host a single course

o_j = Maximum number of sections a course j

g_{jk} = Minimum number of sections a course j that can be offered in term k

$$r_{ij} = \begin{cases} 1 & \text{if instructor } i \text{ is qualified to teach course } j \\ 0 & \text{otherwise} \end{cases}$$

$$t_{il} = \begin{cases} 1 & \text{if instructor } i \text{ is qualified to teach at site, } l \\ 0 & \text{otherwise} \end{cases}$$

$$n_i = \begin{cases} 1 & \text{if instructor } i \text{ is qualified to teach online} \\ 0 & \text{otherwise} \end{cases}$$

$$q_j = \begin{cases} 1 & \text{if course } j \text{ is a required course for completion of the program} \\ 0 & \text{otherwise} \end{cases}$$

$$m_j = \begin{cases} 1 & \text{if course } j \text{ can be offered online} \\ 0 & \text{otherwise} \end{cases}$$

Variables

$$x_{ijkl} = \begin{cases} 1 & \text{if instructor } i \text{ teaches course } j \text{ in term } k \text{ at site } l \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ijk} = \begin{cases} 1 & \text{if instructor } i \text{ teaches course } j \text{ in term } k \text{ online} \\ 0 & \text{otherwise} \end{cases}$$

3.2 Objective Function and Constraints

The objective function maximizes the satisfaction of the staff of instructor, i , teaching course, j . The sum across all instructors, courses, terms, and sites for each variable is multiplied by the satisfaction value to obtain the objective function value. The staff wanted their instructor preference to be considered. So for instance, they might prefer instructor A teaching class X over class Y. This was the most effective way of incorporating the staff preference into the model.

Because of the uniqueness of the Operations Management program, many constraints were necessary to accomplish the assignment. Constraint (1) limits the number of live classes offered at each site, each term. Constraint (2) limits the number

of online classes offered each term. Constraint (3) ensures that only one section of a course is offered live each term at each site. Constraint (4) ensures that the number of times that a live course is offered doesn't exceed the maximum number of times that a single course can be offered at each site. Constraint (5) ensures that each course that has the ability to be taught online is taught at least once during the entire year. Constraint (6) ensures that an instructor teaches at least one course, either live or online, over the entire year. Constraint (7) ensures that each course is offered at least once, either live or online, over the entire year. Constraint (8) ensures that each instructor doesn't exceed the maximum number of courses that he or she is allowed to teach each term. Constraint (9) ensures that the instructor is qualified to teach the course live. Constraint (10) ensures that the instructor is qualified to teach the course online. Constraint (11) ensures that the instructor is qualified to teach the course live at the site. Constraint (12) ensures that the instructor is qualified to teach courses online. Constraint (13) ensures that each required course is offered at least once live over the entire year. Constraint (14) ensures that if a course live, then it's not offered online. Constraint (15) ensures that each term a course does not exceed the maximum number of online sections that can be offered per term. Constraint (16) ensures that the minimum number of online sections is satisfied. In particular, this constraint pertains to the fact that OMGT 5003: Introduction to Operations Management must have at least three sections offered online during terms one and three. These are the terms that new students are brought into the program. Constraints (17) – (20) ensure that the same class isn't offered in back to back terms at each site. Constraint (21) ensures that variables x_{ijkl} and y_{ijk} are binary. The model in its entirety is displayed below.

$$\text{maximize } \sum_{i \in V_1} \sum_{j \in V_2} \sum_{k \in V_3} \sum_{l \in V_4} s_{ij}(x_{ijkl} + y_{ijk})$$

subject to

$$a \leq \sum_{i \in V_1} \sum_{j \in V_2} x_{ijkl} \leq b \quad \forall k, l \quad (1)$$

$$d \leq \sum_{i \in V_1} \sum_{j \in V_2} y_{ijk} \leq e \quad \forall k \quad (2)$$

$$\sum_{i \in V_1} x_{ijkl} \leq 1 \quad \forall j, k, l \quad (3)$$

$$\sum_{i \in V_1} \sum_{k \in V_3} x_{ijkl} \leq p_l \quad \forall j, l \quad (4)$$

$$\sum_{i \in V_1} \sum_{k \in V_3} y_{ijk} \geq m_j \quad \forall j \quad (5)$$

$$\sum_{j \in V_2} \sum_{k \in V_3} \sum_{l \in V_4} x_{ijkl} + \sum_{j \in V_2} \sum_{k \in V_3} y_{ijk} \geq 1 \quad \forall i \quad (6)$$

$$\sum_{i \in V_1} \sum_{k \in V_3} \sum_{l \in V_4} x_{ijkl} + \sum_{i \in V_1} \sum_{k \in V_3} y_{ijk} \geq 1 \quad \forall j \quad (7)$$

$$\sum_{j \in V_2} \sum_{l \in V_4} x_{ijkl} + \sum_{j \in V_2} y_{ijk} \leq c_{ik} \quad \forall i, k \quad (8)$$

$$x_{ijkl} \leq r_{ij} \quad \forall i, j, k, l \quad (9)$$

$$y_{ijk} \leq r_{ij} \quad \forall i, j, k, l \quad (10)$$

$$x_{ijkl} \leq t_{il} \quad \forall i, j, k, l \quad (11)$$

$$y_{ijkl} \leq n_i \quad \forall i, j, k, l \quad (12)$$

$$\sum_{i \in V_1} \sum_{k \in V_3} \sum_{l \in V_4} x_{ijkl} \geq q_j \quad \forall j \quad (13)$$

$$x_{ijkl} + y_{ijk} \leq 1 \quad \forall i, j, k, l \quad (14)$$

$$\sum_{i \in V_1} y_{ijk} \leq o_j \quad \forall j, k \quad (15)$$

$$\sum_{i \in V_1} y_{ijk} \geq g_{jk} \quad \forall j, k \quad (16)$$

$$\sum_{i \in V_1} (x_{ij1l} + x_{ij2l}) \leq 1 \quad \forall j, l \quad (17)$$

$$\sum_{i \in V_1} (x_{ij2l} + x_{ij3l}) \leq 1 \quad \forall j, l \quad (18)$$

$$\sum_{i \in V_1} (x_{ij3l} + x_{ij4l}) \leq 1 \quad \forall j, l \quad (19)$$

$$\sum_{i \in V_1} (x_{ij4l} + x_{ij5l}) \leq 1 \quad \forall j, l \quad (20)$$

$$x_{ijkl}, y_{ijk} \in 0,1 \quad \forall i, j, k, l \quad (21)$$

4. Results and Analysis

4.1 Model Results

The data used for the model came from real data that the Operations Management staff will be using to schedule courses for the 2013-2014 school year. The instructors sent in his or her availability over the next year and stated whether he or she wanted to add a class to his or her repertoire. The staff compiled the data with previous course qualification information. From this data, the staff rated each instructor on course satisfaction on a scale of 1 to 10. The term availability, course qualification, and satisfaction are the parameters that change from year to year. The other parameters are virtually consistent from year to year. The data for these parameters are kept on various spreadsheets throughout the Operations Management department.

Once the data necessary to complete the model was compiled, the model was programmed in AMPL. Through AMPL, the solver CPLEX was used to solve the IP model. The objective function was a total of 7,177. The AMPL model, data, and output files can be seen in the Appendix. Below in Tables 1-10 is a compilation of the output results for each variable. Each table is the schedule for the upcoming 2013-2014 school year.

x_{ijkl} Results

CAMDEN		
Term	Instructor	Course
1	Cash	4613
1	Roberson	5143
2	Cash	4613
2	Perrin	5373
2	Roberson	5733
3	Cash	5013
3	Roberson	5143
4	Cash	4783
4	Perrin	5373
4	Roberson	5113
5	Cash	5013
5	Roberson	5733

Table 1

HFF		
Term	Instructor	Course
1	Casey	5373
1	Donatelli	5113
1	Smith	5013
1	Wilson	4623
2	Donatelli	5733
2	Eveleth	5873
2	Wilson	5133
2	Zilinsky	5423
3	Casey	4623
3	Donatelli	5143
3	WellsM	5463
3	Wilson	5303
4	DelCasatillo	4303
4	Donatelli	5113
4	Smith	5373
4	Wilson	5133
5	Donatelli	5143
5	Eveleth	5873
5	Smith	5013
5	Wilson	5303

Table 2

FAYETTEVILLE		
Term	Instructor	Course
1	Flynn	5123
1	Payne	5873
1	Sandsmark	5113
1	Teague	4623
2	Burgin	5503
2	Sandsmark	5143
2	White	5463
2	Williams	5303
3	Collier	5223
3	JonesP	4623
3	Sandsmark	5113
3	Teague	5733
4	Burgin	5373
4	Flynn	5123
4	Henderson	4783
4	Teague	5873
5	Burgin	5503
5	Teague	5733
5	White	5463
5	Williams	5303

Table 1

LRAFB		
Term	Instructor	Course
1	JonesT	5423
1	Lattanzi	5143
1	WellsP	5733
2	JonesT	5503
2	Lattanzi	5113
2	WellsP	4623
3	JonesT	5423
3	Lattanzi	5143
3	Magri	4303
3	Nethercutt	5463
4	JonesT	5503
4	Lattanzi	5113
4	Nethercutt	5003
4	Robinson	5133
5	Fite	5123
5	JonesT	4623
5	Lattanzi	5733
5	Robinson	5873

Table 4

NSAM		
Term	Instructor	Course
1	Belcher	4303
1	Gagnon	5423
1	Rasmussen	5123
1	Wilke	5143
2	Belcher	4613
2	Rasmussen	5463
2	Ward	5733
2	Wilke	5373
3	Belcher	4303
3	Rasmussen	5433
3	Ward	5113
3	Wilke	5143
4	Gagnon	5423
4	Rasmussen	5463
4	Ward	5733
4	Wilke	4623
5	Belcher	4613
5	Rasmussen	5123
5	Ward	5113
5	Wilke	5373

Table 5

Term 1	
Instructors	Courses
Bean	5133
Brown	4613
Cassady	4873
Costello	5463
Ellixson	5823
Hemphill	5003, 5133
Jackson	5303
Lithgow	5503
Loewer	5633
Mason	5833
Masterson	5823
McGlynn	5873
Mickelson	5423
Nethercutt	5003, 5463
Petrone	4313
PohlE	5013
Richardson	4783, 5503
Roy	5003, 5423
Weiss	5873
WellsM	5123, 5433
Wright	5013
Yeager	5373

Table 6

Term 2	
Instructors	Courses
Bean	5133
Brown	4613
Cassady	4873
Costello	5463
Ellixson	5823
Garner	5823
Hemphill	5133
Jackson	5303
Lithgow	5503
Loewer	5633
Mason	5833
McGlynn	5873
Mickelson	5423
Moores	5303
Morrison	5123
Nethercutt	5003
Petrone	4313
PohlE	5013
Richardson	5503
Rieske	4303
Roy	4783, 5423
Weiss	5873
WellsM	5463
Wright	5013
Yeager	5373

Table 7

Term 3	
Instructors	Courses
Beam	4613, 5373
Bean	5133
Costello	5463
Ellixson	5823
Fite	5123
Hemphill	4623, 5133
Jackson	5303
Lithgow	5503
Loewer	5633
Mason	5833
McGlynn	5873
Mickelson	5423
Moores	5303
Morris	5013
Morrison	5123
Nethercutt	5003
Petrone	4313
PohlE	5003
PohlL	5003
Richardson	5503
Rieske	4303
Roy	4783, 5423
Weiss	5823
WellsM	5433
Wright	5013
Yeager	5373

Table 8

Term 4	
Instructors	Courses
Beam	4613, 5373
Bean	5133
Cassady	4873
Costello	5463
Ellixson	5823
Fite	5123
Hemphill	4623
Jackson	5303
Lithgow	5503
Loewer	5633
Mason	5833
McGlynn	5873
Mickelson	5423
Moores	5303
Morrison	5123
Petrone	4313
PohlE	5013
PohlL	5463
Raynor	4303
Richardson	5503
Rieske	4303
Rosetti	4613
Roy	4783, 5423
Singh	5133
Weiss	5823
WellsM	5433
Wright	5013
Yeager	5373

Table 9

Term 5	
Instructors	Courses
Beam	5003, 5373
Bean	5133
Brown	4613
Cassady	4873
Costello	5463
Ellixson	5823
Halstead	5123
Hemphill	4623
Jackson	5303
Lithgow	5503
Loewer	5633
Mason	5833
McGlynn	5873
Mickelson	5423
Morrison	5123
Nethercutt	5003
Petrone	4313
PohlE	5013
PohlL	5003
Raynor	4303
Richardson	5503
Rieske	4303
Rosetti	4613
Roy	5423
Singh	5133
Weiss	5823
WellsM	5463
Wright	5013
Yeager	5373

Table 10

4.2 Sensitivity Analysis

A sensitivity analysis was performed to determine what would happen to the variables if certain parameters deviated from the norm. One parameter that is likely to change is the maximum number of live classes that can be offered per term. Table 11 below illustrates the changes to the parameter and the resulting objective function value.

<i>b</i> : Max Number of Live Courses		
Iteration	Parameter Value	Obj Function Value
0	4	7177
1	5	7242
2	6	7246
3	7	7246

Table 11

Another parameter that could deviate from the norm is the maximum number of online courses offered per term. If the system could support it, the number could easily increase. Therefore, a sensitivity analysis was done on ten percent increase increments. Below in Table 12 are the results.

<i>e</i> : Max Number of Online Courses		
Iteration	Parameter Value	Obj Function Value
0	30	7177
1	33	7408
2	36	7460
3	39	7460

Table 12

The last parameter that was analyzed was c_{ik} , the maximum number of courses that instructor, i , can teach in term, k . This parameter is very likely to change due to the fact that the availability of the instructors alters from year to year. Understanding the

effects that this parameter can have on the model is essential. Each iteration the parameter for each instructor and each term was changed by the same increment. Below in Table 13 are the results of the analysis.

<i>c[i,k]</i> : Max Number of Live Courses Per Instructor		
Iteration	Parameter Increase	Obj Function Value
0	N/A	7177
1	+1	7860
2	+2	7903

Table 13

4.3 Solution Implementation

Because most of the Operations Management staff doesn't have a background in Operations Research, the majority of the staff does not know how to run AMPL without assistance. Since this model will be used every year to assist in the scheduling process, the likelihood of a staff member with Operations Research knowledge is unknown. To solve this problem, the creation of a decision support tool was necessary.

The tool was developed using Microsoft Excel. The staff will input the set data and through VBA and staff intuition, the parameters for the model are created. Once the parameters are created, these parameters are copied into the text file for the AMPL data file. Some parameters are already in the text file because these parameters do not change from year to year.

Once the data file is complete, the next step is running the model. There are very few commands that the user must input into AMPL to obtain the solution. Once these commands are input, the output and solution can be copied to a text file for easier access. The list of commands to run the model and write the solution to a text file is listed in the instructions for the staff.

Instructions were developed for the Operations Management Staff to use in the future. The instructions include how to work with the Excel tool, how to write the AMPL data text file, a list of commands that must be input in order to run the files within AMPL, and how to convert the AMPL solution to an easier to read text file. These instructions will answer any question that the staff may have. The full list of instructions for the decision support tool can be found within the Appendix.

5. Conclusion and Future Work

For each run of the model, the schedule is just a starting point for the Operations Management staff. Special circumstances might need to be considered year to year and cannot be quantified in the model. Therefore, minor changes to the schedule might be made to satisfy the staff, instructors, and students. For instance, a site might only be able to offer two courses rather than three because there aren't available rooms to teach the courses. This is just one example that might be considered after the initial schedule is developed.

Incorporating some of these extreme circumstances into the model could be something done in the future. As each year passes, some additional trends and constraints might develop that might not have been considered during the development of the model. These constraints can easily be added to the model. As more constraints get added, the tool will also have to be updated. However, these are easy tasks that won't require much additional work.

Something else that may be done is to improve the interface of the decision support tool to make running the model even easier. The excel tool has the capability to

output the AMPL data in the correct format for the data text file. This would save sometime for the staff because many staff members do not have knowledge to write the data file for AMPL. Because of the time constraint, this project was not able to complete this part of the tool. If this were able to be completed, the scheduling process would be even easier for the Operations Management staff.

Even though more work must be done after the initial run of the model and possibly in the future, the decision support tool is still a major step up from post-it notes on a white board. Scheduling used to take three weeks and now the scheduling takes two days. The staff is very thankful for the time that can now be spent elsewhere within the department rather than scheduling the courses.

Appendix

AMPL Model File

```
set instructors;
set courses;
set terms;
set sites;

param satisfaction{instructors, courses}; #satisfaction of instructor teaching the course
live
param minLive; #minimum number of live courses offered
param maxLive; #maximum number of live courses offered
param minOnline;
param maxOnline;
param maxCourses{instructors, terms}; #maximum number of courses an instructor is
allowed to teach
param maxSections{courses}; #maximum number of online sections available a term
param minSections{courses, terms}; #minimum of online sections available a term
param qualifiedCourses{instructors, courses}; #matrix of courses the instructor is
qualified to teach
param qualifiedSites{instructors, sites}; #matrix of sites the instructor is qualified to
teach
param qualifiedOnline{instructors}; #matrix of instructors qualified to teach courses
online
param requiredCourses{courses}; #matrix of required courses
param coursesPerSite{sites}; #number of times a site will host a course
param onlineRequirement{courses}; #binary of whether a course can be offered online

var X{instructors, courses, terms, sites} binary; #if the course is taught
var Y{instructors, courses, terms} binary; #if the course is taught online
var course{instructors};

maximize Satisfaction:
    sum{i in instructors, j in courses, k in terms, l in sites} satisfaction[i,j]*(X[i,j,k,l] +
    Y[i,j,k]);

subject to LimitNumberLiveCourses{k in terms, l in sites}:
    minLive <= sum{i in instructors, j in courses} X[i,j,k,l] <= maxLive;

subject to OneSectionTaught{j in courses, k in terms, l in sites}:
    sum{i in instructors} X[i,j,k,l] <= 1;

subject to LimitNumberOnlineCourses{k in terms}:
    minOnline <= sum{i in instructors, j in courses} Y[i,j,k] <= maxOnline;
```


subject to LiveCourseOnce{j in courses, l in sites}:

$$\sum\{i \text{ in instructors, } k \text{ in terms}\} X[i,j,k,l] \leq \text{coursesPerSite}[l];$$

subject to OnlineCourseAtLeastOnce{j in courses}:

$$\sum\{i \text{ in instructors, } k \text{ in terms}\} Y[i,j,k] \geq \text{onlineRequirement}[j];$$

subject to EnsureInstructorTeaches{i in instructors}:

$$\sum\{j \text{ in courses, } k \text{ in terms, } l \text{ in sites}\} X[i,j,k,l] + \sum\{j \text{ in courses, } k \text{ in terms}\} Y[i,j,k] \geq 1;$$

subject to EnsureCourseOffered{j in courses}:

$$\sum\{i \text{ in instructors, } k \text{ in terms, } l \text{ in sites}\} X[i,j,k,l] + \sum\{i \text{ in instructors, } k \text{ in terms}\} Y[i,j,k] \geq 1;$$

subject to MaximumCoursesTaughtAllowed{i in instructors, k in terms}:

$$\sum\{j \text{ in courses, } l \text{ in sites}\} X[i,j,k,l] + \sum\{j \text{ in courses}\} Y[i,j,k] \leq \text{maxCourses}[i,k];$$

subject to CourseQualification{i in instructors, j in courses, k in terms, l in sites}:

$$X[i,j,k,l] \leq \text{qualifiedCourses}[i,j];$$

subject to OnlineCourseQualification{i in instructors, j in courses, k in terms}:

$$Y[i,j,k] \leq \text{qualifiedCourses}[i,j];$$

subject to SiteQualification{i in instructors, j in courses, k in terms, l in sites}:

$$X[i,j,k,l] \leq \text{qualifiedSites}[i,l];$$

subject to OnlineQualification{i in instructors, j in courses, k in terms}:

$$Y[i,j,k] \leq \text{qualifiedOnline}[i];$$

subject to RequiredCourses{j in courses}:

$$\sum\{i \text{ in instructors, } k \text{ in terms, } l \text{ in sites}\} X[i,j,k,l] \geq \text{requiredCourses}[j];$$

subject to NotBoth{i in instructors, j in courses, k in terms, l in sites}:

$$(X[i,j,k,l] + Y[i,j,k]) \leq 1;$$

subject to NoMoreThanTwoOnlineSections{j in courses, k in terms}:

$$\sum\{i \text{ in instructors}\} Y[i,j,k] \leq \text{maxSections}[j];$$

subject to Courses{i in instructors}:

$$\sum\{j \text{ in courses, } k \text{ in terms, } l \text{ in sites}\} X[i,j,k,l] + \sum\{j \text{ in courses, } k \text{ in terms}\} Y[i,j,k] = \text{course}[i];$$

subject to OfferMoreIntroCourses{j in courses, k in terms}:

$$\sum\{i \text{ in instructors}\} Y[i,j,k] \geq \text{minSections}[j,k];$$

subject to CoursesNotOfferedConsecutiveTerms1{j in courses, l in sites}:
sum{i in instructors} X[i,j,1,l] + sum{i in instructors} X[i,j,2,l] <= 1;

subject to CoursesNotOfferedConsecutiveTerms2{j in courses, l in sites}:
sum{i in instructors} X[i,j,2,l] + sum{i in instructors} X[i,j,3,l] <= 1;

subject to CoursesNotOfferedConsecutiveTerms3{j in courses, l in sites}:
sum{i in instructors} X[i,j,3,l] + sum{i in instructors} X[i,j,4,l] <= 1;

subject to CoursesNotOfferedConsecutiveTerms4{j in courses, l in sites}:
sum{i in instructors} X[i,j,4,l] + sum{i in instructors} X[i,j,5,l] <= 1;

Decision Support Tool Instructions

Excel File Instructions

1. Open file entitled OMGT Scheduling Tool.xlsm
2. If content exists within the file, click the “Clear Tables” button on the “Set Data” worksheet
3. Fill in the instructors, courses, terms, and sites for the upcoming school year under in the respective columns on the “Set Data” worksheet
4. Click the “Populate Tables” and “Populate Set Data” button on the “Set Data” worksheet
5. Go to the “Term Availability” worksheet
6. Fill in the number of courses that each instructor can teach, for each term.
7. Go to the “Site Qualification” worksheet
8. If the instructor is qualified to teach at the site, fill in the cell with a “1”, otherwise fill the cell with a “0”
9. Repeat step 8 for each instructor at each site
10. Go to the “Online Qualification” worksheet
11. If the instructor is qualified to teach online, fill in the cells in column “B” beginning with “B2” with a “1”, otherwise fill the cell with a “0”
12. Repeat step 11 for each instructor
13. Go to the “Course Qualification” worksheet
14. If the instructor is qualified to teach the course, fill in the cell with a “1”, otherwise fill the cell with a “0”
15. Repeat step 14 for each instructor and course
16. Copy the cells, beginning with cell B2, that were just filled in on the “Course Qualification” worksheet
17. Paste these cells, beginning with cell B2, on to the “Course Satisfaction” worksheet
18. Cells that contain 1, will be highlighted with red
19. Fill in the highlighted cells with a number from 1 to 10. 1 being least satisfied and 10 being most satisfied with the instructor teaching that course

Data File Instructions

1. Open file entitled OMGT_Data_File.txt
2. Copy the entire contents of the “Sets” worksheet
3. Paste the contents onto the text file where it states “SET DATA HERE!!”
4. Copy the entire contents of the “Course Satisfaction” worksheet
5. Paste the contents onto the text file where it states “SATISFACTION HERE!!”
6. If any of the parameters defining the minimum and maximum number of courses offered either live or online have changed values, then change the respective parameter
7. Copy the contents of everything below row 1 on the “Term Availability” worksheet
8. Paste the contents onto the text file where it states “TERM AVAILABILITY HERE!!”

9. Copy the entire contents of the “Course Qualification” worksheet
10. Paste the contents onto the text file where it states “QUALIFIED COURSES HERE!!”
11. Copy the contents of everything below row 1 on the “Site Qualification” worksheet
12. Paste the contents onto the text file where it states “QUALIFIED SITES HERE!!”
13. Copy the entire contents of the “Online Qualification” worksheet
14. Paste the contents onto the text file where it states “QUALIFIED ONLINE HERE!!”
15. If any of the other parameters have changed, go through the data file and change the parameters
16. Save the text file as OMGTDataFile.dat within the same file as AMPL (C:/AMPL)
17. Open the text file entitled OMGTModelFile.txt
18. Save the file as OMGTModelFile.mod within the same file as AMPL (C:/AMPL)
19. Create an empty text file entitled Output.out within the same file as AMPL

Running the Model Instructions

1. Open ampl.exe
2. Type in “option solver cplex;” and click enter
3. Type in “model OMGTModelFile.mod;” and click enter
4. Type in “data OMGTDataFile.dat;” and click enter
5. Type in “solve;” and click enter
6. Type in “option omit_zero_rows 1;” and click enter
7. Type in “option omit_zero_cols 1;” and click enter
8. Type in “display X,Y > c:\AMPL\Output.out;” and click enter
9. Open the Output.out file and the solution will be displayed within the file

Bibliography

- [1] Al-Husain, R., Hasan, M. K., & Al-Qaheri, H. (2011). A Sequential Three-Stage Integer Goal Programming (IGP) Model for Faculty-Course-Time-Classroom Assignments. *Informatica*, 35(2), 157-164. Retrieved January 20, 2013
- [2] Al-Qaheri, H., Hasan, M. K., & Al-Husain, R. (2011, December). A Decision Support System for a Three-Stage University Course Scheduler with an Application to College of Business Administration, Kuwait University. *International Journal of Data Analysis and Information Systems*, 3(2), 95-110. Retrieved January 22, 2013
- [3] Al-Yakoob, S. M., & Sherali, H. D. (2006, September). Mathematical Programming Models and Algorithms for a Class-Faculty Assignment Problem. *European Journal of Operational Research*, 173(2), 488-507. Retrieved September 12, 2012, from Science Direct.
- [4] Al-Yakoob, S., Sherali, H. D., & Al-Jazzaf, M. (2010, January). A Mixed-Integer Mathematical Modeling Approach to Exam Timetabling. *Computational Management Science*, 7(1), 19-46. Retrieved January 19, 2013, from SpringerLink.
- [5] Apte, A., Jayasuriya, A., Kennington, J., Krass, I., Mohamed, R., Sorenson, S., & Whitler, J. (1998, September). Class Scheduling Algorithms for Navy Training Schools. *Naval Research Logistics*, 45(6), 533-551. Retrieved September 12, 2012, from Wiley Online Library.
- [6] Dinkel, J. J., Mote, J., & Venkataramanan, M. A. (1989). An Efficient Decision Support System for Academic Course Scheduling. *Operations Research*, 37(6), 853-864. Retrieved January 23, 2013, from JSTOR.
- [7] Ferland, J. A., & Fleurent, C. (1994, April). SAPHIR: A Decision Support System for Course Scheduling. *Interfaces*, 24(2), 105-115. Retrieved October 4, 2012, from Compendex.
- [8] Miranda, J., Rey, P. A., & Robles, J. M. (2012). A Web architecture based decision support system for course and classroom scheduling. *Decision Support Systems*, 52, 505-513. Retrieved January 22, 2013, from ScienceDirect.
- [9] MirHassani, S. A. (2006). A Computational Approach to Enhancing Course Timetabling with Integer Programming. *Applied Mathematics and Computation*, 814-822. Retrieved January 22, 2013, from ScienceDirect.
- [10] Nabih El-Bahrawy, A. (1998). Spreadsheet Class Scheduling for the College of Engineering, University of Qatar. *Engineering Journal of Qatar University*, 11, 193-208. Retrieved September 12, 2012, from Compendex.

- [11] Nandhini, M., & Kanmani, S. (2010). Design of Mathematical Model and Local Search with Heuristics Optimization of Course Timetabling. *Internation Journal of Combinatorial Optimization Problems and Informatics*, 1(2), 20-30. Retrieved January 21, 2013
- [12] Rudova, H., Muller, T., & Murray, K. (2011, April). Complex University Course Timetabling. *Journal of Scheduling*, 14(2), 187-207. Retrieved January 19, 2013, from SpringerLink.
- [13] Sampson, S. E., Freeland, J. R., & Weiss, E. N. (1995, May). Class Scheduling to Maximize Participant Satisfaction. *Interfaces*, 25(3), 30-41. Retrieved September 12, 2012, from JStor.
- [14] Steinbrugge Fant, K., & Nahas, S. (2002, August 7). University Class-Professor Scheduling Model for the Portland State University Computer Science Department. *Management of Engineering and Technology*, 1, 96. Retrieved September 12, 2012, from IEEE Xplore.