8-2013

AN OPTIMIZATION MODEL FOR CLASS SCHEDULING AT A DANCE STUDIO

Chirag Ojha
Clemson University, chiragoojha@gmail.com

AN OPTIMIZATION MODEL FOR CLASS SCHEDULING AT A DANCE STUDIO

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Engineering
Industrial Engineering

---

by
Chirag Ojha
August 2013

Accepted by:
Dr. Scott J. Mason, Committee Chair
Dr. Mary Elizabeth Kurz
Dr. William G. Ferrell

ABSTRACT

Scheduling has been a large area of research for decades. A substantial amount of work has been done to express, classify, and solve scheduling problems. Most of these problems are computationally difficult to solve and require complex algorithms. In this thesis, we develop a mixed-integer linear program for a real world optimization problem at a dance studio. Similar to a university, the students in this studio request a particular class and instructors teach the classes under constrained resources such as a limited number of classrooms. The priorities of instructors as well as dancers are included to further mimic reality. Experimental results confirm the efficacy of the model. Due to the generic nature of the model, it can be used for a wide range of similar timetabling examples with minimum modification.

DEDICATION

This thesis is dedicated to my beloved parents, Omdutt Ojha and Pushpa Ojha and sister, Stutee Ojha for their unconditional support and encouragement. I would also like to dedicate it to my mentor, role model, and inspiration, Dr. Scott J. Mason for his guidance and support throughout the thesis and my stay at Clemson University. His direction, advice and help were essential in shaping this thesis.

ACKNOWLEDGMENTS

I would once again like to thank my advisor, Dr. Scott J. Mason, who passed on to me his passion and zeal for doing research. I also express the deepest appreciation for my committee members, Dr. Mary Kurz and Dr. William Ferrell, whose suggestions and ideas were valuable throughout the course of my thesis. I also offer thanks to my colleagues and friends, Achieng Adongo, Ali Ibrahim Rasheed, Douglas Hansen, Namrata Sengupta, Nargis Hansen, Maria Rodgers, and Sreenath Chalil Madathil for their continued support and help during various stages of my thesis development. Cheryl's Dance Studio in Central, South Carolina deserves a special mention for coming up with the inspiration for this thesis and also providing the necessary data for research and testing.

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

For effective performance in any system, available resources should be utilized in an efficient manner. This can be made possible only if an excellent scheduling system is in place. Scheduling can be defined as, “the allocation of resources over a period of time to perform a collection of tasks” (Noor, 2007). Any scheduling problem involves devising a plan to carry out a number of activities that require constrained resources along with various other constraints to optimize one or more objectives. Developing a schedule can be the solution to many problems including job scheduling in a production facility, student and teacher scheduling at a university, vehicle scheduling in transportation networks, or nurse scheduling at a hospital.

Several techniques can be used to develop a solution for scheduling models. Some of them include mathematical programming, analytical methods, graph coloring approaches, artificial intelligence techniques, heuristics and metaheuristics like simulated annealing, tabu search, genetic algorithms and ant colony optimization (Azimi, 2005). Most of these techniques are meant for a specific type of problem and they need research to adapt them to a different problem. Some techniques are more wide-ranging (e.g., mathematical programming) but if there is a larger dataset, one would have to develop clever heuristics to deal with such a problem.

One of the most popular types of mathematical programming is integer programming. Algorithmic advances in recent years have greatly increased the use of integer programming as a practical technique for solving scheduling problems. The motivation of my research was to develop a weekly class schedule at an area dance studio. This research concept falls within the general area of class scheduling or timetabling. Construction of a schedule for any class whether it is for a university, a dance academy or any other venue, needs to satisfy all the operational rules and needs of the institute, whilst fulfilling the wishes of the students and the teaching staff, if possible. These conditions make the problem a challenging task for staff to solve. Although a manual schedule made in such a setup lasts for years through re-use, changes are frequently needed and patching together historical practices is not always the best policy. Hence, developing a mixed-integer programming model can help adapt to changing constraints and modify the model to fit changing requirements. This has been the focus of my thesis research.

CHAPTER TWO

LITERATURE REVIEW

Timetabling or class scheduling has attracted the interests of many researchers from a variety of disciplines. Since 1960, there has been attention focused towards this problem. Among recent efforts, Daskalaki and Birbas (2005) used a decomposition approach to solve this problem. By doing so, the problem gets divided into two parts and the solution becomes easier to obtain than when solving the entire problem. An issue with this approach is that two models must be formulated and the solution of the first model is often used as the input to the second, which may result in lost quality. Due to recent advancements in computer software and hardware, IP and MIP formulations have started becoming the most favored approach for small to medium-sized instances of such NP hard combinatorial problems. Dasalaki *et al*. (2004) presented a novel 0-1 integer programming formulation for a university timetabling problem. Al-Yakoob *et al*. (2007) presented the mathematical model for assigning faculty members to classes with all the other usual class scheduling problem constraints. The resulting model aims to maximize the individual and collective satisfaction of faculty members in a fair fashion. Dissatisfaction is contained in the objective function.

MirHassani and Habibi (2011) compared different solution approaches for the course timetabling problem and concluded that MIPs and genetic algorithms typically give the best solutions. Heinz and Beck (2012) conducted an experiment with three existing optimization models: MIP, constraint programming and logic-based decomposition. Modern commercial solvers were used to solve instances of the different approaches and it was concluded that MIP models are, at the very least, competitive with any other existing scheduling methods. It was also shown that MIP models are able to provide strong lower and upper bounds and are therefore considered one of the best core technologies to solve scheduling problems. Al-Qaheri *et al*. (2010) developed a sequential three-stage integer programming model for faculty-course-time-classroom assignment. This model, similar to Daskalaki and Birbas (2005), was divided into three parts; however, the new model contained different objectives. The first part dealt with course assignment, the second part dealt with faculty assignment and the third part assigned rooms to courses and faculty. Helber *et al.* (2006) made a university course timetabling model that was then sent to various universities across Germany for its assessment. It was found that professors and assistants preferred using his integer programming model. Helber's model did not take into consideration student preferences, but gave more weight to the availability of professors and the limited number of students in each class.

The emphasis of this thesis research is to design a basic class scheduling model to schedule classes at a local dance studio. However, this same model can also be used to solve generalized class timetabling problems. The model includes a number of different rules and regulations that exist at the dance studio. Most models in the literature are created either from a teacher/faculty or a student point of view to minimize some cost function. While doing this research, I have developed a timetable that is suitable for both the teacher and the student while still minimizing the number of dance classes held per week.

CHAPTER THREE

MATHEMATICAL MODEL

A general class scheduling problem consists of various elements that need to be considered when devising a model for the problem. Of the many elements that define the model, the main ones are student requests for particular classes, instructor preferences for teaching particular dance types, assignment of instructors to the classes, and the ability to fill the classes with students. Students are assigned to classes based on their skill level for the requested class. Skill levels are determined by the instructors based on each student's history of learning and his or her competency with the dance genre. Having taken into consideration these factors, a model is generated that assigns classes to students and instructors to the classes while grouping students of the same skill-level to the same class. The following section explains the model along with the pertinent notation used in the model, the objective function, and constraints.

3.1 Notation

Consider a dance studio where multiple instructors teach multiple classes each week. Each dance class is associated with a genre g, a day of the week d, a time slot t, a class room c and an instructor i. Classes are taught Monday through Thursday from 3:00 PM to 7:00 PM in 45 minute intervals (blocks). Each dancer can request any number of classes based on his or her interests and is represented by IDs. Table 1 presents the pertinent notations for the dance class timetabling model.

Table 1: Notation for Dance Studio Scheduling Model

Sets	
D	Day of the week , indexed by d ($d = 1 \ldots \|D\|$)
T	Time slot of the day , indexed by t ($t = 1 \ldots \|T\|$)
C	Class room number , indexed by c ($c = 1 \ldots \|C\|$
G	Dance genre , indexed by g ($g = 1 \ldots \|G\|$)
I	Dance instructor , indexed by i ($i = 1 \ldots \|I\|$)
N	Individual IDs allotted to each dancer , indexed by n ($n = 1 \ldots \|N\|$)
Parameters	
$Request_{n,g}$	1, if dancer n requests a class of genre g 0, otherwise
$Skill_{n,g}$	Skill level of a dancer n in a particular genre g (1 … $\|Skill\|$)
$Choice_{i,g}$	1 if instructor i is willing to teach dance type g 0, otherwise
Decision Variables	
$a_{d,t,c,g,i}$	1, if genre g is taught in classroom c at time slot t of day d by instructor i; 0, otherwise
$b_{d,t,c,n}$	1, if dancer n is taught in classroom c at time slot t of day d 0, otherwise
$x_{d,t,c,g,i,n}$	1, if dancer n is taught genre g by instructor i in classroom c during time slot t of day d; 0, otherwise
$y_{n,d}$	1, if dancer n is taught on day d 0, otherwise
$z_{i,d}$	1, if instructor i teaches on day d 0, otherwise

We use the sets, parameters and decision variables defined in Table 1 to formulate an integer program for a dance studio class scheduling in the next section.

3.2 Model Formulation

In our model, we try to minimize the total number of classes scheduled each week by using variables $x_{d,t,c,g,i,n}$ and $b_{d,t,c,n}$. At the same time we also try to minimize the total number of days any dancer or instructor has to come each week. This is done using variables $y_{n,d}$ and $z_{i,d}$. Adding weighted coefficients to the variables in the objective function not only allows us to define a multi-objective function but also makes it easier to understand the value obtained for the function. For illustration, if we receive an objective function value of 404.101 at the end of a run, we can easily understand that number of classes scheduled are 4 with 1 instructor and 1 dancer day (4*100 + 4 + 1*0.1 + 1*0.001). The resulting objective function for the model is as shown below along with all pertinent constraints.

$$\text{minimize } (100*\sum_d\sum_t\sum_c\sum_g\sum_i\sum_n x_{d,t,c,g,i,n}) + (\sum_d\sum_t\sum_c\sum_n b_{d,t,c,n}) + (0.1*\sum_n\sum_d y_{n,d}) + (0.001*\sum_i\sum_d z_{i,d}) \quad (1)$$

Subject to

$$\sum_d\sum_t\sum_c\sum_g\sum_i x_{d,t,c,g,i,n} = \sum_g Request_{n,g} \quad \forall n \in N \quad (2)$$

$$\sum_d\sum_t\sum_c\sum_i x_{d,t,c,g,i,n} = Request_{n,g} \quad \forall n \in N,\ g \in G \quad (3)$$

$$\sum_{n} x_{d,t,c,g,i,n} / 1000 \leq a_{d,t,c,g,i} \qquad \forall d \in D,\ t \in T,\ c \in C,\ g \in G,\ i \in I \qquad (4)$$

$$\sum_{g} \sum_{i} a_{d,t,c,g,i} \leq 1 \qquad \forall d \in D,\ t \in T,\ c \in C \qquad (5)$$

$$\sum_{g} \sum_{c} a_{d,t,c,g,i} \leq 1 \qquad \forall d \in D,\ t \in T,\ i \in I \qquad (6)$$

$$a_{d,t,c,g,i} \in \{0,1\} \qquad \forall d \in D,\ t \in T,\ c \in C,\ g \in G,\ i \in I \qquad (7)$$

$$x_{d,t,c,g,i,n} \in \{0,1\} \qquad \forall d \in D,\ t \in T,\ c \in C,\ g \in G,\ i \in I,\ n \in N \qquad (8)$$

Constraint set (2) ensures that the total number of classes assigned to any particular dancer is equal to the sum of classes requested by him or her. Constraint set (3) makes sure that each dancer is assigned the class requested by him or her for a particular dance genre. Therefore, constraint sets (2) and (3) assign the dancers to classes based on the request they have submitted. Constraint set (4) sets the value of decision variables $a_{d,t,c,g,i}$ and $x_{d,t,c,g,i,n}$ appropriately. Additionally no two genres can be taught in the same classroom at the same time of the day and no two instructors can be assigned to the same class which is taken care of by constraint set (5). Constraint set (6) prevents two instructors shall not be assigned to two different classrooms at the same time of the day. Constraint sets (7) and (8) are non-negativity constrains, which imply that the variables should have a value greater than or equal to zero. These seven constraints form the foundation of the model and are referred to as assignment constraints. They are used to assign classes based on requests, dancers to those classes, instructors to each class and each class to a classroom at a particular time slot of the day such that no requirement conflicts with another.

In addition to these assignment constraints, a number of "preference" constraints were developed to further improve the dance studio owner's satisfaction with the produced model results. In other words, while assignment constraints can be considered as "must haves," the preference constraints are meant to improve the solution by including "nice to have" concepts. The following constraints represent the model's preference constraints.

$$\sum_{n} x_{d,t,c,g,i,n} \leq 5 \qquad \forall d \in D, t \in T, c \in C, g \in G, i \in I \tag{9}$$

$$\sum_{t}\sum_{c}\sum_{g} a_{d,t,c,g,i} / 1000 \leq z_{i,d} \qquad \forall d \in D, i \in I \tag{10}$$

$$\sum_{d} z_{i,d} \leq 2 \qquad \forall i \in I \tag{11}$$

$$z_{i,d} \in \{0,1\} \qquad \forall d \in D, i \in I \tag{12}$$

$$\sum_{g}\sum_{i} x_{d,t,c,g,i,n} / 1000 \leq y_{n,d} \qquad \forall d \in D, t \in T, c \in C, n \in N \tag{13}$$

$$\sum_{d} y_{n,d} \leq 2 \qquad \forall n \in N \tag{14}$$

$$y_{n,d} \in \{0,1\} \qquad \forall d \in D, n \in N \tag{15}$$

$$x_{d,t,c,g,i,n} / 1000 \leq Choice_{i,g} \qquad \forall d \in D, t \in T, c \in C, g \in G, i \in I, n \in N \tag{16}$$

$$\sum_{g}\sum_{i} x_{d,t,c,g,i,n} / 1000 \leq b_{d,t,c,n} \qquad \forall d \in D, t \in T, c \in C, n \in N \tag{17}$$

$$x_{d,t,c,g,i,n} + x_{d,t,c,g,i,m} \leq 1 \forall d \in D, t \in T, c \in C, g \in G, i \in I, n \in N, m \in M \ni Skill_{n,g} \neq Skill_{m,g} \tag{18}$$

$$\sum_{g}\sum_{n} (Skill_{n,g} * b_{d,t,c,n}) = \sum_{g}\sum_{m} (Skill_{m,g} * b_{d,t,c,m}) \qquad \forall d \in D, t \in T, c \in C \tag{19}$$

$$b_{d,t,c,n} \in \{0,1\} \qquad \forall d \in D, t \in T, c \in C, n \in N \tag{20}$$

Constraint set (9) ensures that no more than five dancers are scheduled in the same class for the same dance genre. This constraint set is merely for the convenience of the instructors so that they can concentrate equally on all of the students present in the class. Constraint sets (10)–(12) restrict the number of days the instructor (instructor days) comes to the dance studio to teach a particular dance genre. This can change based on the size of the dataset. In this model, we have limited the number of days to two. Constraint sets (13)–(15) restrict the number of days a dancer (dancer days) comes to the studio to learn the dance genre he or she requested. This constraint ensures that all of the requested classes from any particular dancer are scheduled within a maximum of two days. Constraints sets (9), (11) and (14) may change based on the size of the dataset and are therefore a part of preference constraints. Constraint set (16) considers the choice of instructor who teaches any dance genre. Instructors select the dance genres that they are comfortable teaching and the model then assign each instructor to their preferred genre. Constraint sets (17)–(20) consider the skill level of the dancers. Only dancers with the same skill levels are scheduled to any one class. Constraint sets (12), (15) and (20) are non-negativity constraints.

3.3 Model Validation

A sample dataset was created to validate the model formulation. The sample dataset ("Scenario 1") is described in Tables 2 and 3. We used AMPL to write the mathematical program and Gurobi v5.5 to run the instances and obtain results.

Table 2: Detailed Input Values for Scenario 1

Days of the week	4 (Monday through Thursday)
Time slots of the day	5 (45 min time slots from 3:00pm to 6:45 pm)
Classrooms available	2 (Studios A and B)
Dance genres	4 (Hip-hop, Lyrical, Baton and Tap)
Dance instructors	4
Number of dancers	12

Table 3: Request and Skill Matrices of Dancers for Scenario 1

Dancer IDs	Hip-Hop	Lyrical	Baton	Tap
1	1	2	0	0
2	2	1	0	0
3	0	0	1	0
4	0	0	2	2
5	1	0	2	0
6	0	0	1	0
7	1	0	0	0
8	0	0	0	1
9	0	0	0	1
10	1	0	0	0
11	0	1	2	0
12	2	0	0	2

In Table 3, there are two skill levels (1 and 2) for the dancers. A skill level 0 implies that the dancer has not requested this particular dance genre. This matrix is used as the input for our model. The other matrix is the choice matrix that shows the genres each particular instructor wants to teach. The matrix in Table 4 is the choice matrix for this sample Scenario 1 dataset.

Table 4: Choice Matrix for Instructors in Scenario 1

Instructors	Hip-Hop	Lyrical	Baton	Tap
1	1	0	0	1
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

In Table 4, it is clear which instructor wants to teach which genre. For example, Instructor 1 teaches Hip-Hop as well as Tap while the others selected only one dance genre to teach. After solving the model with the data supplied, we get the optimal schedule for this particular scenario. It took 10 seconds for GUROBI to solve the AMPL model and provide the optimal solution (Tables 5 and 6).

Table 5: Optimal Optimization Model Output for Scenario 1

Dance Genre			Day of the week																			
			Monday					Tuesday					Wednesday					Thursday				
			1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Hip-Hop																					
	Classroom	1	1				4															
		2				1																
2	Lyrical																					
	Classroom	1																				
		2	2				1															
3	Baton																					
	Classroom	1		2	2	1																
		2																				
4	Tap																					
	Classroom	1																				
		2		2	2																	

Table 6: Optimal Weekly Schedule for Scenario 1

Day/Slot	3 PM	3:45 PM	4:30 PM	5:15 PM	6:00 PM
Monday	Lyrical Classroom 1 Dancer 1 Instructor 2	Tap Classroom 1 Dancer 8 and 9 Instructor 4	-	Lyrical Classroom 1 Dancer 2 and 11 Instructor 2	Baton Classroom 1 Dancer 3 and 6 Instructor 3
	-	Baton Classroom 2 Dancer 4,5 and 11 Instructor 3	Tap Classroom 2 Dancer 4 and 12 Instructor 4	Hip-Hop Classroom 2 Dancer 1,5,7 and 10 Instructor 1	Hip-Hop Classroom 2 Dancer 2 and 12 Instructor 1
Tuesday	-	-	-	-	-
	-	-	-	-	-
Wednesday	-	-	-	-	-
	-	-	-	-	-
Thursday	-	-	-	-	-
	-	-	-	-	-

A close inspection of Tables 5 and 6 reveal that all of the assignment and preference constraints have been accommodated. Every instructor teaches the class they prefer, all the dancers have been assigned to classes where the requested dance genre was taught, no two classrooms teach the same genre or have the same instructor at any time slot of any day of the week. The number of dancers in a class is limited to five and no dancer or instructor has to come to the studio more than two days. Also from Table 5, we can see that the maximum assignment made to a class in case of Scenario 1 is four, meaning that there are a maximum of four students in a class while the limit is five.

The objective function tries to schedule the minimum number of classes possible. The assignments are done based on the skill level of the dancers. At the same time, using the objective function, we have minimized the total number of classes scheduled. This solution is not only beneficial to the instructors but also for the dancers. The model takes care of both aspects of timetabling. It should be noted that the solution given by the model is only one of the optimal solutions and there can be other schedules having the same value for the objective function.

CHAPTER FOUR

A REAL WORLD CASE STUDY

In order to demonstrate the capabilities of the model with larger, more realistically-sized data sets, we analyzed three different datasets. The model was implemented in AMPL and analyzed by Gurobi on a Dell workstation.

4.1 Scenario 2: 40 Dancers

The first "larger" dataset to be analyzed contained 40 dancers ("Scenario 2"). Table 7 shows the input values for Scenario 2, while Tables 8 and 9 present the optimal solution obtained by the model.

Table 7: Detailed Input Values for Scenario 2

Days of the week	4 (Monday through Thursday)
Time slots of the day	5 (45 min time slots from 3:00pm to 6:45 pm)
Classrooms available	2 (Studios A and B)
Dance genres	9 (Hip-hop, Lyrical, Baton, Jazz, Ballet, Funky Jazz, Preschool, Tap and Point)
Dance instructors	4
Number of dancers	40

As shown in Table 7, we have 40 dancers and four instructors. The process of assignment of the classes considering the constraints (assignment and preference) remains the same. As part of the input, the dancers are asked to request for the dance genre they wish to learn and the instructors are given an option to teach the dance genre they prefer. Since the data set is larger, some of the preference constraints needed to be altered. The limit of students in any particular class becomes 12 but the total number of dancer or instructor days is still restricted to two.

Table 8: Optimization Model Output for Scenario 2

Dance Genre			Day of the week																			
			Monday					Tuesday					Wednesday					Thursday				
			1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Hip-Hop																					
	Classroom	1								12										1		
		2		1	6																	
2	Lyrical																					
	Classroom	1													2	1	1	1				
		2								1	9	5										
3	Baton																					
	Classroom	1																				
		2																				
4	Tap																					
	Classroom	1	4			1					3											
		2					2															
5	Jazz																					
	Classroom	1										1										2
		2						3	3													
6	Ballet																					
	Classroom	1		6			2						1						1			
		2																				
7	Funky jazz																					
	Classroom	1							2												1	
		2																				
8	Preschool																					
	Classroom	1						2						1								
		2																				
9	Point																					
	Classroom	1			2																	
		2	1			1																

Table 9: Optimal Weekly Schedule for Scenario 2

Day/Slot	3 PM	3:45 PM	4:30 PM	5:15 PM	6:00 PM
Monday	Tap Classroom 1 Skill Level 4 Instructor 1	Ballet Classroom 1 Skill Level 1 Instructor 2	Point Classroom 1 Skill Level 1 Instructor 2	Tap Classroom 1 Skill Level 1 Instructor 1	Ballet Classroom 1 Skill Level 2 Instructor 2
	Point Classroom 2 Skill Level 2 Instructor 2	Hip-Hop Classroom 2 Skill Level 3 Instructor 1	Hip-Hop Classroom 2 Skill Level 1 Instructor 1	Point Classroom 2 Skill Level 4 Instructor 2	Tap Classroom 2 Skill Level 2 Instructor 1
Tuesday	Preschool Classroom1 Skill Level 1 Instructor 4	Funky Jazz Classroom 1 Skill Level 1 Instructor 3	Hip-Hop Classroom 1 Skill Level 1 Instructor 4	Tap Classroom 1 Skill Level 1 Instructor 4	Jazz Classroom 1 Skill Level 2 Instructor 4
	Jazz Classroom 2 Skill Level 4 Instructor 3	Jazz Classroom 2 Skill Level 1 Instructor 4	Lyrical Classroom 2 Skill Level 3 Instructor 3	Lyrical Classroom 2 Skill Level 1 Instructor 3	Lyrical Classroom 2 Skill Level 2 Instructor 3
Wednesday	Ballet Classroom 1 Skill Level 4 Instructor 4 -	Preschool Classroom 1 Skill Level 4 Instructor 4 -	Baton Classroom 1 Skill Level 1 Instructor 4 -	Baton Classroom 1 Skill Level 2 Instructor 4 -	Baton Classroom 1 Skill Level 1 Instructor 4 -
Thursday	Lyrical Classroom 1 Skill Level 4 Instructor 4 -	Ballet Classroom 1 Skill Level 3 Instructor 4 -	Hip-Hop Classroom 1 Skill Level 4 Instructor 4 -	Funky Jazz Classroom 1 Skill Level 2 Instructor 4 -	Jazz Classroom 1 Skill Level 3 Instructor 4 -

It can be seen from Table 8 that the maximum number of students in one classroom is 12. The model took approximately 87 minutes to get the optimal solution. From Table 9, we can confirm the result once again. Day 2 Classroom 2 has three classes for the dance genre Lyrical (i.e., 4:30pm, 5:15pm, and 6:00pm) and all of these three classes have a different skill level of dancers. Also, instructors are assigned based on the choice submitted by them. The schedule is as compact as possible while at the same time reducing the objective function which includes the total number of classes, number of dancer days and instructor days.

4.2 Scenario 3: 76 Dancers

Table 10 shows the input values for the next larger case, Scenario 3, while Tables 11 and 12 present a non-optimal solution to the model.

Table 10: Detailed Input Values for Scenario 3

Days of the week	4 (Monday through Thursday)
Time slots of the day	5 (45 min time slots from 3:00 pm to 6:45 pm)
Classrooms available	2 (Studios A and B)
Dance genres	9 (Hip-hop, Lyrical, Baton, Jazz, Ballet, Funky Jazz, Preschool, Tap and Point)
Dance instructors	4
Number of dancers	76

Table 10 shows that in this scenario, we have 76 dancers and four instructors. The process of assignment of the classes considering the constraints (assignment as well as preference) remains the same. As a part of the input, the dancers are asked to request the dance genre they wish to learn and the instructors are also given an option to teach the dance genre they prefer. Since the data set is larger, some of the preference constraints needed to be altered. The limit of students in any particular class becomes 20. The total number of dancer or instructor days is now restricted to three. From Table 11, it can be seen that the maximum number of dancers in a class was restricted to 20, and the model keeps to this with a total of 15.

Table 11: Optimization Model Output for Scenario 3

Dance Genre			Day of the week																			
			Monday					Tuesday					Wednesday					Thursday				
			1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Hip-Hop																					
	Classroom	1	11			3										15						
		2	7												2							2
2	Lyrical																					
	Classroom	1		5						5			13		3		2					
		2																				
3	Baton																					
	Classroom	1												1								
		2								7												
4	Tap																					
	Classroom	1			5																	
		2		4		6										2		2				
5	Jazz																					
	Classroom	1																		5		
		2						3			3			5							2	
6	Ballet																					
	Classroom	1									4								3			
		2			7												2		2	1		
7	Funky jazz																					
	Classroom	1						4														
		2							3				2									
8	Preschool																					
	Classroom	1					1		2													2
		2										1										
9	Point																					
	Classroom	1																2			4	
		2					2															

There are also a few classes that could be combined with the other and hence there is an optimality gap of about 50%. Generally the difference between a best known solution, e.g. the incumbent solution in mixed integer programming, and a value that bounds the best possible solution is called the Optimality Gap. The model takes approximately 180 minutes to give us this result. Also heuristic or meta-heuristic approach solutions could give us better results. After considering the 76 dancer dataset, we finally tried to run the model with a larger dataset of 121 dancers which represents the actual size of the dance studio under study's enrollment.

Table 12: A Suggested Weekly Schedule for Scenario 3

Day/Slot	3 PM	3:45 PM	4:30 PM	5:15 PM	6:00 PM
Monday	Hip-Hop Classroom 1 Skill Level 1 Instructor 1	Lyrical Classroom 1 Skill Level 3 Instructor 2	Tap Classroom 1 Skill Level 3 Instructor 1	Hip-Hop Classroom 1 Skill Level 1 Instructor 2	Preschool Classroom 1 Skill Level 2 Instructor 1
	Hip-Hop Classroom 2 Skill Level 3 Instructor 2	Tap Classroom 2 Skill Level 2 Instructor 1	Ballet Classroom 2 Skill Level 1 Instructor 2	Tap Classroom 2 Skill Level 1 Instructor 1	Point Classroom 2 Skill Level 2 Instructor 2
Tuesday	Funky Jazz Classroom1 Skill Level 1 Instructor 3	Preschool Classroom 1 Skill Level 1 Instructor 4	Lyrical Classroom 1 Skill Level 2 Instructor 3	Ballet Classroom 1 Skill Level 2 Instructor 4	-
	Jazz Classroom 2 Skill Level 3 Instructor 4	Funky Jazz Classroom 2 Skill Level 4 Instructor 3	Baton Classroom 2 Skill Level 1 Instructor 4	Jazz Classroom 2 Skill Level 4 Instructor 3	Preschool Classroom 2 Skill Level 4 Instructor 4
Wednesday	Lyrical Classroom 1 Skill Level 1 Instructor 3	Baton Classroom 1 Skill Level 4 Instructor 4	Lyrical Classroom 1 Skill Level 2 Instructor 3	Hip-Hop Classroom 1 Skill Level 1 Instructor 2	Lyrical Classroom 1 Skill Level 4 Instructor 4
	Funky Jazz Classroom 1 Skill Level 2 Instructor 4	Jazz Classroom 1 Skill Level 1 Instructor 3	Hip-Hop Classroom 1 Skill Level 2 Instructor 2	Tap Classroom 1 Skill Level 2 Instructor 4	Ballet Classroom 1 Skill Level 3 Instructor 2
Thursday	Point Classroom 1 Skill Level 4 Instructor 2	Ballet Classroom 1 Skill Level 4 Instructor 2	Jazz Classroom 1 Skill Level 4 Instructor 4	Point Classroom 1 Skill Level 1 Instructor 2	Lyrical Classroom 1 Skill Level 1 Instructor 2
	Tap Classroom 1 Skill Level 4 Instructor 4	Ballet Classroom 1 Skill Level 1 Instructor 4	Ballet Classroom 1 Skill Level 2 Instructor 2	Jazz Classroom 1 Skill Level 2 Instructor 4	Hip-Hop Classroom 1 Skill Level 4 Instructor 4

4.3 Scenario 4: 121 Dancers

Table 13 shows the input values for Scenario 4, while Tables 14 and 15 present a good, but non-optimal solution to the model.

Table 13: Detailed Input Values for Scenario 4

Days of the week	4 (Monday through Thursday)
Time slots of the day	5 (45 min time slots from 3:00pm to 6:45 pm)
Classrooms available	2 (Studios A and B)
Dance genres	9 (Hip-hop, Lyrical, Baton, Jazz, Ballet, Funky Jazz, Preschool, Tap and Point)
Dance instructors	4
Number of dancers	121

Table 13 shows that in this scenario we have 121 dancers and four instructors. The process of assignment of the classes considering the constraints (assignment as well as preference) remains the same. As a part of the input, the dancers are asked to request for the dance genre they wish to learn and the instructors are also given an option to teach the dance genre they prefer. Due to a larger dataset, we had to change some of our preference constraints. Limit of students in any particular class becomes 30. The total number of dancer or instructor days is restricted to 3.

Table 14: Optimization Model Output for Scenario 4

Dance Genre			Day of the week																			
			Monday					Tuesday					Wednesday					Thursday				
			1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Hip-Hop																					
	Classroom	1	4		7				6													
		2			4						5											
2	Lyrical																					
	Classroom	1									9		7									
		2																5				
3	Baton																					
	Classroom	1				2																8
		2				4						1										
4	Tap																					
	Classroom	1					8					1						10			6	
		2		4			11														13	
5	Jazz																					
	Classroom	1														6				12		
		2							6										6			
6	Ballet																					
	Classroom	1		3										8			7		7			
		2	8					10						6						14		
7	Funky jazz																					
	Classroom	1																				
		2								10												
8	Preschool																					
	Classroom	1						3														
		2															10					
9	Point																					
	Classroom	1								3												
		2											3			2						

It can be seen that the maximum number of dancers in the class was restricted to 30, while the model results has it at 14 (Table 14). Also, there are a few classes that could be combined with others and hence there is an optimality gap of about 26%. The model takes approximately 240 minutes to give us this result. Also, as mentioned earlier, heuristic or meta-heuristic approach solutions could give us a better result. However, this solution provides a feasible starting point for the dance studio's owner to make her own modifications and changes for class scheduling.

Table 15: Optimal Weekly Schedule for Scenario 4

Day/Slot	**3 PM**	**3:45 PM**	**4:30 PM**	**5:15 PM**	**6:00 PM**
Monday	Hip-Hop Classroom 1 Skill Level 1	Ballet Classroom 1 Skill Level 2	Hip-Hop Classroom 1 Skill Level 1	Baton Classroom 1 Skill Level 2	Tap Classroom 1 Skill Level 2
	Ballet Classroom 2 Skill Level 2	Tap Classroom 2 Skill Level 3	Hip-Hop Classroom 2 Skill Level 1	Baton Classroom 2 Skill Level 1	Tap Classroom 2 Skill Level1
Tuesday	Preschool Classroom1 Skill Level 2	Hip-Hop Classroom 1 Skill Level 2	Point Classroom 1 Skill Level 1	Lyrical Classroom 1 Skill Level 4	Tap Classroom 1 Skill Level 3
	Ballet Classroom 2 Skill Level 5	Jazz Classroom 2 Skill Level 2	Funky jazz Classroom 2 Skill Level 1	Hip-Hop Classroom 2 Skill Level 1	Baton Classroom 2 Skill Level 2
Wednesday	Lyrical Classroom 1 Skill Level 2	Ballet Classroom 1 Skill Level 3	Lyrical Classroom 1 Skill Level 3	Jazz Classroom 1 Skill Level 1	Ballet Classroom 1 Skill Level 4
	Point Classroom 1 Skill Level 2	Ballet Classroom 1 Skill Level 3	-	Point Classroom 1 Skill Level 1	Preschool Classroom 1 Skill Level 1
Thursday	Tap Classroom 1 Skill Level 5	Ballet Classroom 1 Skill Level 1	Jazz Classroom 1 Skill Level 3	Tap Classroom 1 Skill Level 6	Baton Classroom 1 Skill Level 3
	Lyrical Classroom 1 Skill Level 1	Jazz Classroom 1 Skill Level 4	Ballet Classroom 1 Skill Level 6	Tap Classroom 1 Skill Level 4	Point Classroom 1 Skill Level 3

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

In this thesis, we developed and tested a mixed-integer, linear programming model for timetabling or class scheduling. The research was motivated by a local dance studio's class scheduling problem. Validation and verification experiments confirmed the accuracy of the model. In addition, the inclusion of preference constraints helps to produce "more satisfactory" class schedules for the dance studio's owner.

From our experimentation, we conclude that although this program gives us a close to optimal solution for a large dataset (121 dancers), it may be necessary to develop and employ some heuristic or meta-heuristic solution approaches to handle such huge problems. While large datasets can be resolved by optimization, this process is quite time intensive. Future work should include developing a heuristic or meta-heuristic solution approach for practical sized problems to hopefully obtain an improved result for such timetabling problems in a shorter amount of time. However, as this problem is analyzed once per year by the studio owner, execution run time is not the most critical concern as compared to the satisfaction of the modeled preference constraints.

REFERENCES

S. Daskalaki, T. Birbas, E. Housos, An Integer programming formulation for a case study in university timetabling, European Journal of Operational Research 153 (2004) 117-135

S. A. MirHassani, F. Habibi, Solution Approaches to the course timetabling problem, Artificial Intelligence Review (2011)

Stefan Heinz, J. C. Beck, Reconsidering Mixed Integer Programming and MIP based Heuristics for Scheduling, Springer-Verlag Berlin Heidelberg (2012) 217-227

Katja Schimmelpfeng, Stefan Helber, Application of a real world university course timetabling model solved by integer programming, Springer-Verlag (2006)

S. A. Mirhassani, A computational Approach to enhancing course timetabling with Integer Programming, Applied Mathematics and Computation 175 (2006) 814-822

Raed Al-Husain, Mohamad K. Hasan, Hameed Al-Qaheri, A sequential three stage integer goal programming model for faculty-course-time-classroom assignment, Informatica 35 (2011) 157-164

Salem M. Al-Yakoob, Hanif D. Sherali, Mathematical programming models and algorithms for a class faculty assignment problem, European Journal of Operational Research 173 (2006) 488-507

APPENDICES

Appendix A

AMPL model file

```
#######SETS#######
set N; # dancer ids
set D; # day of the week indexed by d (1 2 3 4 for Monday, Tuesday,
Wednesday, Thursday)
set T; # time slot of the day indexed by t (45 mins slots considering 5
slots right now)
set I; # instructor of the dance indexed by i (there are 4 teachers 1 2
3 4)
set G; # type of dance indexed by g (there are 9 different types of
dances)
set C; # number of classroom indexed by c (there are 2 classrooms)

######PARAMETERS#####

param Request{N,G} >= 0; # 1 if dancer n requests class g (Binary
parameter)
param Skill{N,G} >=0; # skill level given to a particular dancer
student
param Choice{I,G} >=0; # 1 if instuctor i wants to teach genre g
(Binary parameter)

######DECISION VARIABLE#####

var a{D,T,C,G,I} binary;  # 1 if dance type g is scheduled in classroom
c, taught by instructor i, at time slot t of day d, 0 otherwise
var x{D,T,C,G,I,N} binary;  # 1 if dancer n is taught dance type g in
classroom c by instructor i in time slot t on day d
var y{N,D} binary; # 1 if dancer n is taught on day d
var z{I,D} binary; # 1 if Instructor i teaches on day d
var b{D,T,C,N} binary; # 1 if student n is in classroom c on day d at
time t

######OBJECTIVE FUNCTION#####

minimize Objective:
      100*sum{d in D,t in T,i in I,g in G,c in C,n in N} x[d,t,c,g,i,n]
      + 0.1*sum{i in I,d in D}z[i,d] + 0.001*sum{n in N,d in D}y[n,d]
      + sum{d in D,t in T,c in C,n in N}b[d,t,c,n];

######CONSTRAINTS######

# For every dancer n, the sum of number of classes assigned is equal to
the sum of the number of classes requested by him/her
subject to C1 {n in N}: sum{d in D,t in T,i in I,g in G,c in C}
x[d,t,c,g,i,n] = sum{gg in G} Request[n,gg];
```

```
# For every dancer and dance genres, the sum of the classes matches the
requested ones
subject to C2 {n in N,g in G}: sum{d in D,t in T,i in I,c in C}
x[d,t,c,g,i,n] = Request[n,g];

# To bring a and x to balance each other
subject to C3 {g in G,t in T,d in D,i in I,c in C}: sum{n in N}
x[d,t,c,g,i,n] / 1000 <= a[d,t,c,g,i];
subject to C4 {d in D,i in I}: sum{t in T,g in G,c in C} a[d,t,c,g,i] /
1000 <= z[i,d];
subject to C5 {d in D,n in N,t in T,c in C}: sum{i in I,g in G}
x[d,t,c,g,i,n] / 1000 <= y[n,d];
subject to C6 {t in T,d in D,n in N,c in C}: sum{i in I,g in G}
x[d,t,c,g,i,n] / 1000 <= b[d,t,c,n];

# No two classes in 1 classroom at any time or day and there is a
unique genre and instructor
subject to C7 {c in C,t in T,d in D}: sum{g in G,i in I} a[d,t,c,g,i]
<= 1;

#Only one instrucor at a particular day on a particular time
subject to C8 {i in I,t in T,d in D}: sum{g in G,c in C} a[d,t,c,g,i]
<= 1;

# There is a limit of 12 students in each class
subject to C9 {d in D, t in T, i in I, g in G, c in C} : sum{n in N}
x[d,t,c,g,i,n] <= 5;

# For every insturctor, sum of number of days he/she teaches shall be
less than 2
subject to C10 {i in I} : sum{d in D} z[i,d] <= 2;

# Try to accomodate the students in less than 2 days
subject to C11 {n in N} : sum{d in D} y[n,d] <= 2 ;

# Teachers shall be given choice of selecting their subjects and are
assigned classes that way
#only 1 of the 4 instructors shall teach all dance genres (teachers
select a max of 3 dances to teach)
subject to C12 {d in D,t in T,i in I,g in G,c in C,n in N}:
x[d,t,c,g,i,n] / 1000 <= Choice[i,g];

# Skill level
subject to C13a {d in D, t in T, i in I, g in G, c in C, n in N, m in
N:Skill[n,g]<>Skill[m,g]}: x[d,t,c,g,i,n] + x[d,t,c,g,i,m] <= 1;
subject to C14 {d in D,t in T,c in C}: sum{n in N,g in G}
Skill[n,g]*b[d,t,c,n] = sum{m in N,g in G} Skill[m,g]*b[d,t,c,m];
```

Appendix B

AMPL data file

```
set N:= 1 2 3 4 5 6 7 8 9 10 11 12;
set T:= 1 2 3 4 5;
set I:= 1 2 3 4;
set G:= 1 2 3 4;
set C:= 1 2;
set D:= 1 2 3 4;

param Request :1 2 3 4:=
             1 1 1 0 0
             2 1 1 0 0
             3 0 0 1 0
             4 0 0 1 1
             5 1 0 1 0
             6 0 0 1 0
             7 1 0 0 0
             8 0 0 0 1
             9 0 0 0 1
            10 1 0 0 0
            11 0 1 1 0
            12 1 0 0 1;

param Skill :1 2 3 4:=
           1 1 2 0 0
           2 2 1 0 0
           3 0 0 1 0
           4 0 0 2 2
           5 1 0 2 0
           6 0 0 1 0
           7 1 0 0 0
           8 0 0 0 1
           9 0 0 0 1
          10 1 0 0 0
          11 0 1 2 0
          12 2 0 0 2;

param Choice:  1 2 3 4:=
             1 1 0 0 1
             2 0 1 0 0
             3 0 0 1 0
             4 0 0 0 1;
```

Appendix C

AMPL run file

```
reset;
model ModelName.mod;
data DataFileName.dat;
option show_stats 1;
option omit_zero_rows 1;
option omit_zero_cols 1;
option solver gurobi_ampl;
option gurobi_options
'outlev=1'
'threads=7'
'mipfocus=1'
'timelim=21600'
;
solve;

printf "Day,Time,Classroom,Genre,Instructor,Student ID,x\n">x.out;
printf {d in D,t in T,c in C,g in G,i in I,n in
N:x[d,t,c,g,i,n]>0}:"%i,%i,%i,%i,%i,%i,%i\n",d,t,c,g,i,n,x[d,t,c,g,i,n]
>x.out;
printf "Instructor,day,z\n">z.out;
printf {i in I,d in D:z[i,d]>0}:"%i,%i\n",i,d,z[i,d]>z.out;
printf "Student ID,day,y\n">y.out;
printf {n in N,d in D:y[n,d]>0}:"%i,%i\n",n,d,y[n,d]>y.out;
display x;
display solve_message;
```

Appendix D

AMPL output file

```
Day,Time,Classroom,Genre,Instructor,Student ID,x
1,1,1,1,1,2,1
1,1,2,2,2,2,1
1,1,2,2,2,11,1
1,2,1,3,3,5,1
1,2,1,3,3,11,1
1,2,2,4,1,4,1
1,2,2,4,1,12,1
1,3,1,3,3,3,1
1,3,1,3,3,6,1
1,3,2,4,1,8,1
1,3,2,4,1,9,1
1,4,1,3,3,4,1
1,4,2,1,1,12,1
1,5,1,1,1,1,1
1,5,1,1,1,5,1
1,5,1,1,1,7,1
1,5,1,1,1,10,1
1,5,2,2,2,1,1
```