MIT 6-4 CourseRoad Optimization

Sarah Bentley, Isabella Struckman, Ryan Welch

December 9, 2023

sbentley@mit.edu, istruck@mit.edu, rcwelch@mit.edu

1 Problem Statement & Motivation

In this project, we address the challenge of **designing the ideal class schedule for students in the newly established 6-4 degree program at MIT**. The uniqueness of this major, with its intricate and overlapping requirements, presents a significant hurdle for students in understanding which class combinations fulfill their degree. **Our objective is threefold: to minimize the total hours of class work based on student-reported data, to maximize class ratings, and to find a balanced approach between these two factors**. By adjusting the emphasis placed on each goal, we aim to develop a tailored solution that resonates with individual student preferences. This is not just about creating a schedule; it's about empowering students to make informed decisions that cater to their workload preferences and simultaneously ensure they fulfill all degree requisites. The relevance of this solver is underscored by the major's recent introduction and the inherent complexity of course selection.

2 Data

To solve this optimization problem, we needed access to an extensive amount of data about each course at MIT as well as all of the degree and general institute requirements. For classes, we needed access to class units, ratings, estimated hours per week, terms offered and requirements fulfilled. For general institute and degree requirements, we needed access to the different combinations of required classes needed to graduate.

2.1 Collection

As far as we know, this data is not readily available in any publicly accessible dataset. Therefore, we had to be creative in our approach of gathering this essential data. We wrote algorithms using the Python library BeatifulSoup to scrape the MIT course catalog and accumulate data on the units, requirements fulfilled, graduate level status, prerequisite courses, joint subjects and other variables for each MIT class. Additionally, we wrote algorithms to interact with the MIT Firehose API to scrape student reported data such as ratings, estimated hours per week, approximate enrollment and other factors for each class. We also utilized the Firehose API to scrape the specific GIR and 6-4 major requirements. After pulling the relevant data from each of these sources, we created and preprocessed two different tabular datasets. The first contained all of the courses at MIT with all of their attributes. We processed each of the columns to be in numerical format such that it was easily for the solver to interpret. The second dataset consisted of all GIR and 6-4 major requirements with multi-hot encodings of the classes that satisfy those requirements.

2.2 Estimation

Given our objective function depends on the ratings and estimated hours per week for each course, it is crucial that these values are accurate and complete in our dataset. However, given class ratings and hours are determined by optional end-of-year course evaluations, it was the case that many classes were missing ratings and hours altogether. In particular, we found that 34.1% of all MIT classes had no reported ratings or hours, and 15.7% of classes satisfying GIR and 6-4 requirements had no reported ratings or hours. This was clearly a major issue and we needed to determine a method of estimating the ratings and hours for classes that were missing this data in order for our solutions to be as accurate as possible. Our method was to take all of the classes that did have fully reported data and train a linear multivariate regression model to estimate the ratings and hours from the other attributes that we had. In particular our inputs to the regression model were the class department (one-hot encoded), whether it was HASS, CI-H, CI-HW, lab, or graduate course, along with the average rating and hours of the prerequisite courses, and maximum hours and ratings of the prerequisite courses (0.0 if no prerequisites). We trained this model using the Statsmodels package in Python and used it to predict the ratings and hours for all MIT courses that were missing these attributes and concatenated these results with the data on classes that did have reported ratings to create our fully reported dataset.

2.3 Scaling

The reported ratings and hours for MIT courses had very different distributions, as the mean rating was 5.98 with standard deviation 0.67 and the mean hours was 10.24 with standard deviation 5.35 *(a deeper analysis of these distributions can be viewed in Figure 3 and Figure 4)*. This difference in magnitude of the two metrics would lead to bias in the optimization problem, where we would be giving undue weight towards hours over ratings. Thus, to ensure these two metrics were weighted equally, we performed min-max scaling to scale all of the ratings and hours values to be between 0 and 1 while keeping their original distributions.

3 Methods

Though we only implemented a formulation for 6-4, we designed our problem flexibly so that additional majors could be easily added (and other assumptions reduced) later on. Therefore, many of our design choices were made to maximize replicability. After designing the problem, we implemented our program in Python using a Gurobi solver. We validated our results using a student-made manual approximation of the lowest average hour/week required for each MIT major.

3.1 The Problem

3.1.1 Description

Our decision variable is a 1-hot matrix modeling which classes a student takes for each of their 8 semesters at MIT. Our objective is to minimize the number of total weekly hours a student spends on classes over their career while maximizing the total ratings of courses they take. We introduce a user-defined variable, a, for how relatively important these factors are. If $a = 1$, the objective is to find the set of classes with the lowest time commitment. If $a = 0$, the objective is to find the highest rated set of classes. If $0 < a < 1$, the objective considers hours with weight a and class rating with weight $1 - a$.

Our constraints ensure that the solution is a legal set of classes that, if completed, will allow a student to graduate MIT with a 6-4 degree. In service of clarity and future additions, we separated our constraints into three main categories.

1. General: These constraints are simple and nonspecific to any major or the GIRs.
   (a) Every class is either taken or not taken
   (b) Don't retake classes
   (c) No more than 60 units per semester
   (d) At least 36 units per semester
   (e) Don't take classes outside of season offered
   (f) A class's pre-reqs must be taken before it is taken
2. GIRs: These constraints ensure students complete the GIRs which are unrelated to their major.
   (a) Every required course type (e.g. Bio, Calc 1, HASS-A) is taken. *(These are described in detail in Table 10)*
   (b) Complete at least 180 units beyond GIR's
3. 6-4: These constraints ensure students complete the 6-4 major. *(These are described in detail in the Table 11)*

Between these three components (decision variable, objective, constraints) we are able to model much of the decision making process to complete a 6-4 degree while optimizing for some combination of time requirements and class quality. Nevertheless, we made many assumptions to ease our problem formulation. These are all outlined in the Appendix.

3.1.2 Formulation

Due to the complexity of our formulation, we present a highly abridged version here. A complete and detailed representation can be found in the Appendix.

Variables

C		# of classes at MIT
a		class hour importance
h_i	$i \in C$	reported hours spent on class i
r_i	$i \in C$	student ranking of class i
u_i	$i \in C$	institute assigned units to class i
s_i	$i \in C$	is class i offered in the Spring?
f_i	$i \in C$	is class i offered in the Fall?
em_i	$i \in C$	if i is EECS or Math

Decision Variable

$x_{i,j}$ $\quad i \in \{1, \ldots, C\}, j \in \{1, \ldots, 8\}$
if class i is taken semester j

Objective

$$\min \quad a \sum_{i=1}^{C} h_i \sum_{j=1}^{8} x_{i,j} - (1-a) \sum_{i=1}^{C} r_i \sum_{j=1}^{8} x_{i,j}$$

Constraints

$$x_{i,j} \in \{0,1\} \tag{1}$$

$$\sum_{j=1}^{8} x_{i,j} \leq 1 \quad \forall i \in \{1, \ldots, C\} \tag{2}$$

$$\sum_{i=1}^{|C|} u_i * x_{i,j} \leq 60 \quad \forall j \in \{1, \ldots, 8\} \tag{3}$$

$$\sum_{i=1}^{|C|} u_i * x_{i,j} \geq 36 \quad \forall j \in \{1, \ldots, 8\} \tag{4}$$

$$x_{i,j} - f_i < 1 \quad \forall i \in \{1, \ldots, C\}, \forall j \in \{2,4,6,8\} \tag{5}$$

$$x_{i,j} - s_i < 1 \quad \forall i \in \{1, \ldots, C\}, \forall j \in \{1,3,5,7\} \tag{6}$$

3.2 Implementation

Due to compatibility complications between data pre-processing (which took place using Python) and Julia, we implemented our program in Python using the Gurobi optimizer. This implementation, our original Julia code which is almost synonymous with the final Python code, and all our data, is accessible in our Github.

4 Results

Our results primarily aligned with our expectations, despite a few surprises. We examined three models: M_h with $a = 1.0$ (only hours minimized), M_{hr} with $a = 0.5$ (hours minimized and ratings maximized), and M_r with $a = 0.0$ (ratings maximized). See Figure 1 to see how the results varied over each model. As expected, M_h generally selected the schedule with the fewest hours on average and M_{hr} generally selected the schedule with the highest ratings on average, with M_{hr} in between both. Interestingly, in some semesters we had the opposite result, with M_h having more hours and M_r having lower ratings. These deviations may be due to the constraints on pre-requisite courses and GIR and 6-4 requirements.

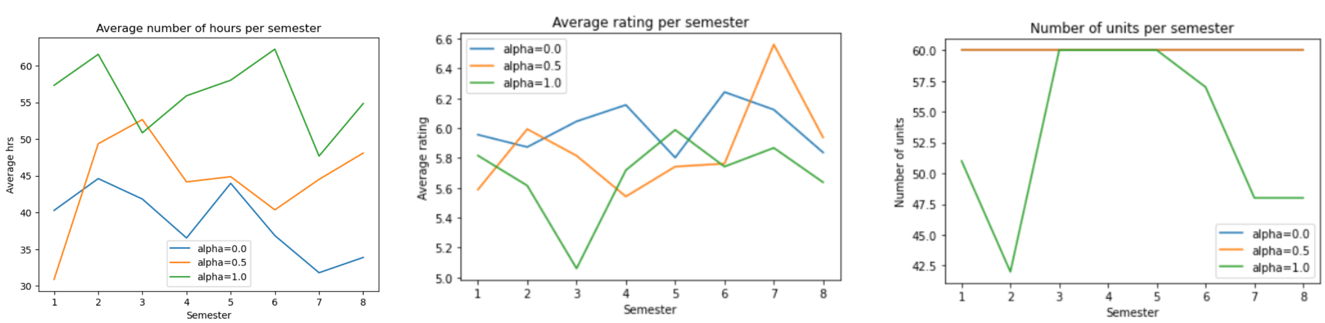

Figure 1: Average hours, average ratings, and total units per semester for each of the three models.

Despite our expectations, we found that all the models always selected over 40 total units per semester, even when minimizing hours. Noticeably, M_{hr} and M_r always selected the maximum number of units per semester. This result is surprising for M_{hr}, as such a large number of units must correspond to a large number of reported hours.

As α increases, it seems that the main effect on the optimal schedule is a decrease in the number of classes taken outside of the requirements. Thus, M_h seems to find the most feasible and realistic schedule for an MIT student, primarily taking necessary classes, in addition to a few low-hour classes.

Interestingly enough, though we don't constrain the optimizer to do so, each of our 3 proposed schedules completes a HASS concentration (which is required for graduation) in the Theater Arts. This likely due to the Theater Arts concentration's relatively lax constraints (students can take any 3 classes in the department), the courses' overall positive reviews, and light time requirements. This result actually aligns very well with our intuition – many of our classmates without any interest in HASS's will concentrate in the Theater Arts due to its ease.

Examining each model's schedule, we can see one key trend: the model doesn't have the knowledge of a typical MIT student. It is missing the schedule knowledge spread through consultations with advisors and word-of-mouth among students. As a result, it doesn't follow the typical order an MIT student would follow, with most GIRs taken freshman and sophomore year and somewhat difficult classes taken under freshman PNR. It would be hard to encode these constraints without creating an actual machine learning model, which would learn this knowledge by examining real student schedules. The following sections examine the outputs of M_h, M_{hr}, and M_r in more detail.

4.1 Minimize Hours ($a = 1.0$)

S1	15.312	18.01	21M.622	6.100A	6.UAT
S2	18.03	20.005	21M.600	6.1200	
S3	18.022	24.234	3.091	6.4590	7.015
S4	18.06	2.980	21W.762	6.1010	6.3000
S5	12.301	21M.830	6.3900	6.8371	6.S977
S6	15.305	21M.862	24.02	6.1210	8.011
S7	21M.704	6.1220	6.4210	8.02	
S8	15.238	18.05	18.204	6.3100	

Table 1: A course road at MIT that earns a 6-4 degree while minimizing hours/week in class. Courses satisfying a GIR are red and courses satisfying a 6-4 requirement are green. 6.4590 satisfies both the HASS GIR requirement and the 6-4 human centric requirement.

Table 1 shows our optimizer's proposed schedule to minimize class hours. We see that M_h selects the schedule with the fewest classes out of all the models. Despite some of the GIRs, such as 8.011 and 8.02, being taken junior and senior year, this schedule seems relatively typical. One perplexing choice is 8.011, which is known to be much harder than 8.01.

4.2 Minimizing Hours and Maximizing Ratings ($a = 0.5$)

S1	15.501	18.03	6.4590	7.015	8.012	
S2	15.3941	15.8141	18.01	20.005	21M.600	21M.862
S3	15.0111	15.874	18.022	21M.704	6.1200	6.UAT
S4	15.238	15.5181	15.6151	18.05	6.100A	8.02
S5	15.312	18.06	21M.830	3.091	6.1010	
S6	21W.762	6.1210	6.3000	6.3100	6.3900	
S7	12.387	21M.622	6.4210	6.8371	6.S977	
S8	15.0251	15.305	15.8471	15.8731	21L.019	6.1220

Table 2: A course road at MIT that earns a 6-4 degree while minimizing hours/week in class and maximizing. Courses satisfying a GIR are red and courses satisfying a 6-4 requirement are green. 6.4590 satisfies both the HASS GIR requirement and the 6-4 human centric requirement.

Table 2 shows the schedule outputted by M_{hr}. This schedule contains far more classes that are not used towards any requirement, likely to maximize ratings. However, it seems to select the same classes to satisfy the 6-4 requirements as M_h, suggesting that the reported hours for each class still affects the decisions. Interestingly, M_{hr} selects a lot of Course 15 classes, perhaps because they have both low hours and high ratings.

4.3 Maximize ratings ($a = 0.0$)

S1	1.057	15.0111	15.305	20.005	3.983	6.100A	6.UAT
S2	11.139	15.0251	15.9001	18.01	6.1010	9.55	
S3	15.874	18.022	21M.605	21M.809	6.9280	8.01L	
S4	1.016	18.03	2.00B	21M.490	6.4200	8.282	
S5	18.C06	6.1200	6.4590	6.8371	7.015		
S6	12.402	2.00A	20.260	21M.480	6.3700	8.022	
S7	12.385	12.387	2.110	3.096	5.112	6.1210	
S8	21W.775	6.1220	6.3000	6.3100	6.3720		

Table 3: A course road at MIT that earns a 6-4 degree while maximizing total ratings. Courses satisfying a GIR are red and courses satisfying a 6-4 requirement are green. 6.4590 satisfies both the HASS GIR requirement and the 6-4 human centric requirement.

Table 3 shows the schedule selected by M_r. Similar to M_{hr}, it selects far more classes than M_h to maximize total ratings. While still selecting many Course 15 classes, M_r also selects a lot of classes that must be popular at MIT, such as 2.00B, 9.55, and more. Also, M_r chooses a different set of classes to satisfy the 6-4 constraints, taking 6.3720 instead of 6.3900 and 6.3700 instead of 18.03.

4.4 Validation

We used a pre-existing hand-calculated spreadsheet storing the minimum hours for every major at MIT as a validation metric. This spreadsheet only took into account class hours and ratings, and did not consider class schedule (e.g. when classes are offered and whether they can all be reasonably taken together). However, its results are a helpful additional metric by which we can ensure our result is reasonable. For a 6-4 degree, it calculates the minimum total weekly hour commitment is 34.2. The average is 38.2. The spreadsheet also estimates that the average 6-4 class rating (not including GIRs) is 5.7. The spreadsheet was provided by Julia Schneider, and is accessible here.

Examining our graphs, they align well with the expected results. When minimizing hours, we average about 33.5 per week which is extremely comparable. It's somewhat surprising our score is lower since we take into account more constraints than the hand-made spreadsheet. However, it might be explained by new class evaluations and our optimal GIR schedule. Their work does not describe how they determined their optimal GIR hours, and it's possible they did not consider courses with surprisingly low reported hours like 8.011 for the first physics GIR. When balancing hours and ratings, our average number of hours per semester is about 39.1 which is also very comparable with their average hours per semester for 6-4. Finally, when balancing hours and ratings, our average class rating is about 5.9 which also aligns well with their average rating for 6-4 courses.

5 Practical Implications

Course 6-4 is very new to MIT, so our analysis may be helpful for lowerclassmen looking for potential class plans. In fact, many other majors already have example class plans available. As 6-4 majors, our team is also interested to see how our schedules may vary from the optimums. Furthermore, if we extended the problem formulation to other majors, our project would be a usable tool for students thinking about optimizing their time at MIT.

6 Appendix: Assumptions & Full Problem Formulation

6.1 Assumptions & Limitations

Due to limitations in time and data, we made a great deal of assumptions to simplify our program. As described in the Data section, we approximated the rating and time commitment of classes for which evaluations weren't available. Some assumptions are associated with a small detail in our constraints, and those can be found in the associated tables later in this Appendix. The others are listed below.

1. Our decision variable inherently encodes the assumption that a student spends 8 semesters on their degree.

2. Our entire problem formulation assumes MIT does not add or remove courses. This assumption is significant since courses are adjusted every semester, however we don't hard-code it into our implementation. Therefore, the actual program can automatically update along with any changes made to the data.

3. Since we're using student evaluations to approximate the time commitment and quality of each class, we're assuming such evaluations are accurate measurements of the objectives. This puts classes with fewer evaluators at a disadvantage for accuracy. Also, since we use the most recent student evaluations, we remove temporal bias of classes whose time commitment and quality may have changed over time but introduce inaccuracy for classes that have different staff in the Fall and Spring or otherwise shift between two semesters.

4. We only consider classes taught in the Fall and the Spring. We both assume students will take no courses over IAP or Summer and that classes once taught in a given season will always be offered that season. The latter though, like assumption 2 above, will automatically update in our program with new data.

5. Since we constrain students to never retake a class, we assume that students never fail courses.

6. Our unit constraints also assume a student never overloads their courses or take a lightload/become a part-time student.

7. The implementation of the constraint ”Don't take classes outside of season offered” in both our problem formulation and implemented program assumes students start in the Fall and never take an odd number of semesters off.

8. Our implementation overall assumes students never test out of any courses

9. The implementation of the constraint ”Complete at least 180 units beyond GIR's” in both our problem formulation and implemented program assumes students have no outside units (from transfer credit or a 5 on AP tests) and that no GIR class, even if not directly satisfying a GIR requirement, is counted as a unit beyond the GIR's.

10. Due to lack of available data, we assume students do not complete a HASS concenration though this is necessary to graduate. Interestingly, many of our results automatically complete one by fulfilling the 8 HASS's totaly GIR

11. Due to lack of available data, we ignore the PE credit and assume students are student-athletes who automatically fulfill it.

12. Though not encoded it our formulation, when producing the program we implemented some practical assumptions to generate reasonable results (since some courses are highly specialized and unhelpful or unreasonable for most students): we removed non-course 6 graduate courses, unique courses for specific programs (like ROTC or medicine), and non-course 6 courses with ¡6 units.

6.2 Variables

Table 4, Table 5, Table 6, Table 7 provide detailed descriptions of every variable we use to model our problem. Each of these variables is stored as described in our program implementation after we pre-process our data.

LP Representation	**Description**	**Assumptions**
semesters = S	# of semesters in a student's career = 8	No early graduation. No extra semesters.
classes = C	# of classes at MIT [nonnegative]	MIT never adds or removes classes.
max pre-reqs = Q	# of pre-reqs for MIT class w/ greatest # of pre-reqs [nonnegative]	No class with more pre-reqs will ever exist.
cost weight = a	How much should hours be weighted (if ratings are weighted 1- a)? Student decided. [$0 \leq a \leq 1$]	

Table 4: Integer variables that are not exclusively related to GIRs or the 6-4 major.

LP Representation	Description	Assumptions
hours = $h_i \quad \forall i \in C$	Average # of student-reported hours per week that class i took to complete last time it was offered [positive]	Course hours don't vary semester-to-semester. All courses have student reported hours.
rating = $r_i \quad \forall i \in C$	Average student rating of class i last time it was offered [positive]	Course quality doesn't change semester-to-semester. All courses have student reported ratings
units = $u_i \quad \forall i \in C$	Units for class i [positive integers]	
fall classes = $f_i \quad \forall i \in C$	Is class i offered in the fall? [binary]	Classes once offered in the fall will always be offered in the fall. No classes are taken outside of the fall/spring.
spring classes = $s_i \quad \forall i \in C$	Is class i offered in the spring? [binary]	Classes once offered in the spring will always be offered in the spring.
eecs/math classes = $em_i \quad \forall i \in C$	Is class i an eecs/math course? [binary]	6-4 classification of eecs/math courses are universal.

Table 5: Vector variables that are not exclusively related to GIRs or the 6-4 major.

LP Representation	Description
hass-a = $\text{HA}_i \quad \forall i \in \{1 \dots C\}$	Is class i a hass-a course? [binary]
hass-s = $\text{HS}_i \quad \forall i \in \{1 \dots C\}$	Is class i a hass-s course? [binary]
hass-h = $\text{HH}_i \quad \forall i \in \{1 \dots C\}$	Is class i a hass-h course? [binary]
$\text{CIH}_i \quad \forall i \in \{1 \dots C\}$	Is class i a cih course? [binary]
$\text{HASS}_i \quad \forall i \in \{1 \dots C\}$	Is class i a hass course? [binary]
$\text{BIO}_i \quad \forall i \in \{1 \dots C\}$	Is class i a bio gir? [binary]
8.01 = $\text{PHYS1}_i \quad \forall i \in C$	Is class i a 8.01 satisfier? [binary]
8.02 = $\text{PHYS2}_i \quad \forall i \in C$	Is class i a 8.02 satisfier? [binary]
18.01 = $\text{CALC1}_i \quad \forall i \in C$	Is class i a 18.01 satisfier? [binary]
18.02 = $\text{CALC2}_i \quad \forall i \in C$	Is class i a 18.02 satisfier? [binary]
$\text{CHEM}_i \quad \forall i \in \{1 \dots C\}$	Is class i a chem gir? [binary]
$\text{REST}_i \quad \forall i \in \{1 \dots C\}$	Units of classes that are rest courses [positive integer]
$\text{LAB}_i \quad \forall i \in \{1 \dots C\}$	Units of classes that are lab courses [positive integer]

Table 6: Variables associated exclusively with GIRs

Variable	Description
$\text{PROGRAMMING}_i \quad \forall i \in \{1 \ldots C\}$	Is class i a 6-4 programming course? 6.0001, 6.100A/6.0001, 6.100L/6.000L
$\text{FOUNDATION}_i \quad \forall i \in \{1 \ldots C\}$	Is class i a 6-4 foundations course? 6.1200/6.042, 6.1010/6.009, 6.1210/6.006
$\text{MATH2}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.S084, 18.C06, 18.06
$\text{MATH3}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.3700/6.041, 6.3800/6.008, 18.05
$\text{DATA}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.3720/6.401, 6.3900/6.036
$\text{MODEL}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.3000/6.003, 6.4110/6.038, 6.4400/6.837
$\text{DECISION}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.3100/6.302, 6.4110/6.038, 6.7201/6.215
$\text{COMPUTE}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.1220/6.046, 6.4400/6.837, 6.7201/6.215
$\text{HUMAN}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: 6.3260/6.207, 6.3950/6.404, 6.4120/6.804, 6.C35, 9.660
$\text{SERC}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: these
$\text{CIM-6\_4}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: these
$\text{CIM2}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: these
$\text{AIDUS}_i \quad \forall i \in \{1 \ldots C\}$	Is class i one of: these

Table 7: Variables associated exclusively with the 6-4 Major

LP Formulation	Description
Pre-Reqs = $PR_{i,q,z}, \forall i \in \{1 \ldots C\}, \forall q \in \{1 \ldots Q\}, \forall z \in \{1 \ldots C\}$	Does class z satisfy pre-req q for class i? $PR_{a,b,c} = 1$ if class a's bth pre-req is satsified by c. $PR_{a,b,i} = 1$ for all i if class a has <b pre-reqs.
GIR Requirements = $G_{i,k}, \forall i \in \{1 \ldots C\}, \forall k \in \{1 \ldots 13\}$	Does class i satisfy all or part of GIR requirement k? Columns of G = [HA, HS, HH, CIH, HASS, BIO, PHYS1, PHYS2, CALC1, CALC2, CHEM, REST, LAB] (len = 13)
GIR Requirement Satisfiers = $G_k^{\textbf{sat}}, \forall k \in \{1 \ldots 13\}$	How many of $G_{:,k}$ are needed to satisfy GIR requirement k? $G_k^{\textbf{sat}}$= [1, 1, 1, 2, 8, 1, 1, 1, 1, 1, 1, 12, 12]
6-4 Requirements = $M_{i,l}, \forall i \in \{1 \ldots C\}, \forall l \in \{1 \ldots 14\}$	Does class i satisfy all or part of GIR requirement l? Columns of $M_{i,l}$ = [PROGRAMMING, FOUNDATION, MATH2, MATH3, DATA, MODEL, DECISION, COMPUTE, HUMAN, SERC, CIM-6_4, CIM2, AIDUS + CIM-6_4, CIM2 + AIDUS] (len = 14)
6-4 Requirement Satisfiers = $M_l^{\textbf{sat}}, \forall l \in \{1 \ldots 15\}$	How many of $M_{:,l}$ are needed to satisfy 6-4 requirement l? $M_l^{\textbf{sat}} = [1, 3, 1, 1, 5, 1, 1, 1, 1, 1, 1, 2, 2, 3]$

Table 8: Variables that are complex or combinations of previously defined variables

6.3 Objective

6.3.1 Decision variables

$x_{i,j}$	$i \in C, j \in \{1, \ldots, 8\}$	if class i was taken in semester j
$K_{i,l}$	$i \in C, l \in \{1, \ldots, 14\}$	if class i counts towards 6-4 requirement l
E_i	$i \in C$	if class i counts as an extra EECS/math class

6.3.2 Objective

$$\min \quad a\sum_{i=1}^{C} h_i \sum_{j=1}^{8} x_{i,j} - (1-a)\sum_{i=1}^{C} r_i \sum_{j=1}^{8} x_{i,j}$$

6.4 Constraints

Table 9, Table 10, and Table 11 show detailed constraints for our problem. Figure **??** provides a visual to visualize a particularly complex 6-4 constraint.

Explanation	LP Representation	Assumptions
Every class is either taken or not taken	$x_{i,j} \in \{0,1\}$	
Don't retake classes	$\sum_{j=1}^{8} x_{i,j} \leq 1 \quad \forall i \in \{1 \ldots C\}$	Classes are never failed nor retaken
Extra math/eecs class must be in em	$E_i \leq em_i \quad \forall i \in \{1, \ldots, C\}$	None
Class only counts for 6-4 requirement if taken	$K_{i,l} \leq \sum_{j=1}^{8} x_{i,j} \quad \forall i \in \{1, \ldots, C\}$ $\forall l \in \{1, \ldots, 14\}$	None
Extra math class can't count for any 6-4 requirement	$E_i + K_{i,l} \leq 1 \quad \forall i \in \{1, \ldots, C\}$ $\forall l \in \{1, \ldots, 14\}$	None
Class only counts for extra math requirement if taken	$E_i \leq \sum_{j=1}^{8} x_{i,j} \quad \forall i \in \{1, \ldots, C\}$	None
Class must satisfy 6-4 requirement to count towards it	$K_{i,l} \leq M_{i,l} \quad \forall i \in \{1, \ldots, C\}$ $\forall l \in \{1, \ldots, 14\}$	None
No more than 60 units per semester	$\sum_{i=1}^{\{1 \ldots C\}} u_i * x_{i,j} \leq 60 \quad \forall j \in \{1, \ldots, 8\}$	Student never overloads
At least 36 units per semester	$\sum_{i=1}^{\{1 \ldots C\}} u_i * x_{i,j} \geq 36 \quad \forall j \in \{1, \ldots, 8\}$	Student never lightloads
Don't take classes outside of season offered	$x_{i,j} - A_{z,i} < 1$ $\forall i \in \{1 \ldots C\}, \forall j \in \{1, \ldots, 8\}$, z = $\begin{cases} \text{Fall} & \text{if } j\%2 = 1 \\ \text{Spring} & \text{if } j\%2 = 0 \end{cases}$	Student start in the Fall & never take odd #s of semesters off
Cannot take a course if you haven't taken its pre-reqs beforehand	$\sum_{a=1}^{\{1 \ldots C\}} x_{i,j} * \sum_{h=1}^{j-1} x_{a,h} * PR_{i,h,a} \geq x_{i,j}$ $\forall i \in \{1 \ldots C\}, \forall j \in \{1, \ldots, 8\}$, $\forall s \in \{1, \ldots, S\}$	Pre-reqs are never flexible and no class can be tested out of

Table 9: Constraints that are not specific to GIRs or the 6-4 Major.

Explanation	LP Representation	Assumptions
Every GIR requirement is satisfied	$G_{i,k} * \sum_{j=1}^{8} x_{i,j} \geq G_k^{\textbf{sat}}$ $\forall k \in \{1, \ldots, 13\}$	Student does not test out of any GIR
To satisfy their respective GIR reqs, 18.01A & 18.02A must be taken one after the other if taken	$x_{p,j} = x_{q,j+1}$ p, q = 18.01A, 18.02A indices, $\forall j \in \{1, \ldots, 7\}$	
Student must take ≥ 180 units beyond GIRs (UBGs)	$\sum_{j=1}^{8} (U_i * x_{i,j}) * \mathcal{H}(\sum_{k=1}^{13} G_{k,i})$ $\geq 180^1$	No GIR class counts for UBGs, even if not fulfilling a GIR req (e.g. take 9 HASSs). AP test 5s don't go to UBGs.

Table 10: Constraints specific to GIRs.

Explanation	LP Representation	Assumptions
Student must take 1 additional EECS/Math course after satisfying below constraints	$\sum_{i=1}^{C} E_i = 1$	
Every other 6-4 Major requirement is satisfied	$\sum_{i=1}^{\{1\ldots C\}} M_{i,l} * K_{i,l} * \sum_{j=1}^{8} x_{i,j} \geq M_l^{\text{sat}}$ $\forall l \in \{1, \ldots, 14\}$	See Figure 2 for CIM assumptions
Different class for each center and CIM req	$\sum_{l=5}^{9} K_{i,l} + K_{i,11} \leq 1$ $\forall i \in \{1, \ldots, C\}$	None

Table 11: Constraints specific to the 6-4 Major.

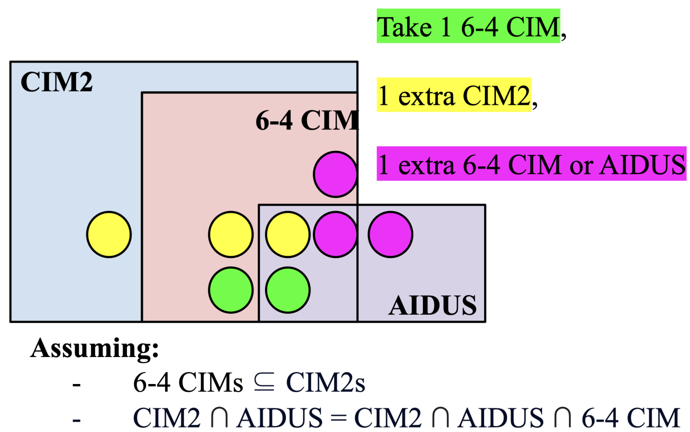

Figure 2: Visual aid to understand the constraints we designed around "Take one 6-4 CIM", "Take one additional EECS CIM", and "Take one additional 6-4 CIM or AIDUS class". The squares show how those class categories share classes, and the the circles represent where a class taken to satisfy the coordinating constraint could fit within the overlapping categories.

6.5 Data

Figure 3 and Figure 4 show the distributions of the reported hours and ratings for MIT classes.

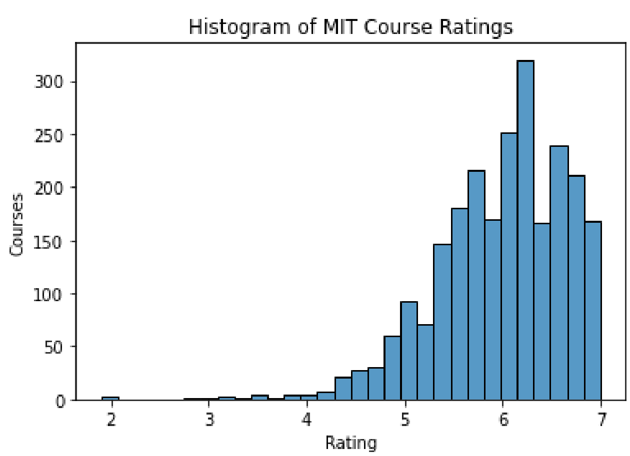

Figure 3: Distribution of reported class ratings

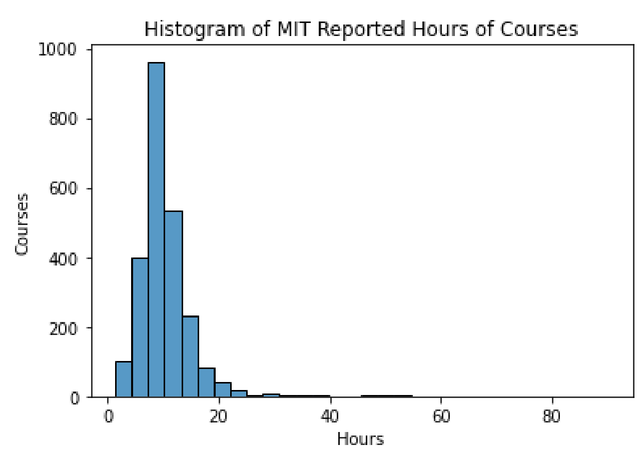

Figure 4: Distribution of reported class hours