Partitioning Students into Cohorts during COVID-19

Richard Hoshino[1] and Irene Fabris[2]

[1] Northeastern University, Vancouver, British Columbia, Canada
r.hoshino@northeastern.edu
[2] Quest University Canada, Squamish, British Columbia, Canada
irene.fabris@questu.ca

Abstract. The COVID-19 pandemic has forced educational institutions to make significant changes to safeguard the health and safety of their students and teachers. One of the most effective measures to reduce virus transmission is partitioning students into discrete cohorts.

In primary and middle schools, it is easy to create these cohorts (also known as "learning groups"), since students in each grade take the same set of required courses. However, in high schools, where there is much diversity in course preferences among individual students, it is extremely challenging to optimally partition students into cohorts to ensure that every section of a course only contains students from a single cohort.

In this paper, we define the Student Cohort Partitioning Problem, where our goal is to optimally assign cohorts to students and course sections, to maximize students being enrolled in their desired courses. We solve this problem by modeling it as an integer linear program, and apply our model to generate the Master Timetable for a Canadian all-boys high school, successfully enrolling students in 87% of their desired courses, including 100% of their required courses. We conclude the paper by explaining how our model can benefit all educational institutions that need to create optimal student cohorts when designing their annual timetable.

Keywords: School Timetabling · Integer Programming · Optimization.

1 Introduction

The COVID-19 virus has led to the worst global pandemic in over a hundred years. Since the first case was identified in December 2019, the disease has spread worldwide, leading to 131 million cases and 2.8 million deaths as of April 1, 2021 [13]. In addition to destabilizing world economies, the pandemic has also had a profound impact on education, with nearly 87% of the world's students, i.e., 1.5 billion learners in over 170 countries, affected by school closures [23]. The switch to Remote Learning has been overwhelming for many students who live in conditions that are not suitable for home study, and has further exacerbated social inequalities such as access to technology [14] [20] .

In many countries, governments and school boards have invested considerable resources to safely bring students back to school. There are numerous measures to mitigate COVID-19 transmission in schools and the Center for Disease Control [8] divides these mitigation measures into three categories: *personal* controls (e.g. hand hygiene, masks, and physical distancing), *engineering* controls (e.g. air ventilation systems, plexiglass barriers), and *administrative* controls (e.g. scheduling breaks and meals at different times).

According to the CDC, the most important administrative control is to ensure that the same group of students learn together each day, avoiding interactions with other student groups. These learning groups, also known as *cohorts*, are already in place at many primary schools, since the same group of students stay together for the entire year, learning from a single teacher. Creating cohorts is straightforward for students in many middle schools and junior high schools, since every student in each grade takes the same set of required courses.

However, creating cohorts is far more complex for students in senior high school (usually 16- and 17-year-olds), since these students have a much larger set of course options available. While some schools allow students to self-select into a particular stream (e.g. Arts, Sciences) where they only take courses with students from that stream, many schools encourage or require students to take courses from all disciplines, with each discipline having numerous options.

To illustrate the challenge of partitioning students into cohorts, consider a scenario where each of 9 students (S_1 to S_9) wishes to enroll in four courses chosen among 12 course offerings (C_1 to C_{12}).

Suppose we have the following set of desired courses.

Student	Desired Courses
S_1	C_1, C_4, C_7, C_{10}
S_2	C_1, C_5, C_8, C_{11}
S_3	C_1, C_6, C_9, C_{12}
S_4	C_2, C_4, C_9, C_{11}
S_5	C_2, C_5, C_7, C_{12}
S_6	C_2, C_6, C_8, C_{10}
S_7	C_3, C_4, C_8, C_{12}
S_8	C_3, C_5, C_9, C_{10}
S_9	C_3, C_6, C_7, C_{11}

We can construct a timetable that grants each of the 9 students all four of their requests, with each course taking place in one time slot.

Course	Enrolled Students	Time Slot		Course	Enrolled Students	Time Slot
C_1	S_1, S_2, S_3	1		C_7	S_1, S_5, S_9	3
C_2	S_4, S_5, S_6	1		C_8	S_2, S_6, S_7	3
C_3	S_7, S_8, S_9	1		C_9	S_3, S_4, S_8	3
C_4	S_1, S_4, S_7	2		C_{10}	S_1, S_6, S_8	4
C_5	S_2, S_5, S_8	2		C_{11}	S_2, S_4, S_9	4
C_6	S_3, S_6, S_9	2		C_{12}	S_3, S_5, S_7	4

For example, student S_1's timetable is C_1, C_4, C_7, C_{10}. If we give a score of 1 point whenever a student is enrolled in a desired course, we see that the above timetable achieves the best possible score of $9 \times 4 = 36$.

Now suppose we need to split the 9 students and 12 courses into 3 cohorts. Suppose we partition our students and courses into these cohorts of equal size:

Cohort	Students	Courses
1	S_1, S_2, S_6	C_1, C_4, C_8, C_{10}
2	S_4, S_5, S_9	C_2, C_5, C_7, C_{11}
3	S_3, S_7, S_8	C_3, C_6, C_9, C_{12}

Since each course is only available to students from that cohort, we can no longer enroll students in all of their desired courses. For example, student S_1 is enrolled in three desired courses (C_1, C_4, C_{10}) but not in C_7 because this course is only offered to students in Cohort 2, and S_1 is assigned to Cohort 1. We can show that the above cohort partition yields a timetable scoring 21 total points.

Assuming that each of the three cohorts must contain 3 students and 4 courses, we can show that 21 points is optimal. Thus, the best possible cohort partition produces a 41.7% reduction from the maximum score of 36.

This simple example illustrates the challenge of cohort partitioning. At large high schools, where there are thousands of students with heterogeneous course preferences, school administrators are pressured to ensure that their students can enroll in their desired courses. Due to COVID-19, the additional requirement of student cohorts makes timetabling even harder.

In the Canadian province of British Columbia, where both authors reside, the government mandated cohorts of size at most 120, for every high school in the province. Thus, a small high school with 400 students (100 students for each of Grades 9, 10, 11, 12) could treat each grade as a single cohort, and would only need to forbid students from taking courses outside of their cohort (e.g. a Grade 11 student taking Grade 12 math). However, for a large high school with 2000 or more students, creating these cohorts is incredibly challenging.

The provincial government announced the 120-student cohort limit on July 29, 2020, as the centerpiece of their Back to School plan [1]. Thus, schools had just over a month to implement this policy, to ensure their students and staff could return in September under the new guidelines. Numerous solutions were reported: restricting student choice by eliminating courses, scheduling certain courses outside of school hours by making them virtual, and hiring more teachers to teach additional "sections" of a course (e.g. one section of Calculus per cohort). Canadian High Schools usually define a course *section* as an offering distinguished from other course sections by time slot, classroom, and teacher (e.g. offering multiple sections of AP Calculus on different days and times).

Some schools simply ignored the cohort policy, as they would have been forced to re-design their Master Timetable. For example, at one high school, several sections of a course have students from *six* different cohorts, with the students in each cohort required to sit together in the same part of the classroom [21].

One school hired the authors of this paper to apply Linear Programming techniques to maximize the students' ability to take their desired courses while ensuring that each section of every course only consists of students from a single cohort. We recently developed an algorithm for optimizing student course preferences in school timetabling [12], and in this paper, we expand upon this work by introducing and solving the Student Cohort Partitioning Problem (SCPP).

This paper proceeds as follows. In Sections 2 and 3, we define the SCPP and provide a brief literature review on related work that involves partitioning students into cohorts. In Section 4, we describe our solution to the SCPP by formulating it as an integer linear program. In Section 5, we apply our model (with over 1.5 million binary decision variables) to generate the Master Timetable for a Canadian high school, partitioning 328 students and 196 course sections into three cohorts. In Section 6, we discuss the limitations of cohort-based timetabling on student choice, and in Section 7, we conclude the paper with questions and directions for future research.

2 Problem Definition

To avoid confusion in how we label our sets, we now rename timeslots as *blocks* and cohorts as *learning groups*. Let I be the set of individual students, T be the set of teachers, C be the set of courses, S be the set of sections, B be the set of blocks, and L be the set of learning groups.

Each course has one or more sections, and each course section is represented by the pair (c, s), where $c \in C$ and $s \in S$.

In the School Timetabling Problem (STP), our goal is to find a feasible assignment of course sections to teachers and blocks. The more general version of the STP is a combinatorial optimization problem, which asks for the best assignment satisfying all of the hard constraints while maximizing the satisfaction of the teachers being assigned their desired courses in specific blocks.

The Post-Enrollment Course Timetabling Problem (PECTP) was introduced just over a decade ago [17], as part of the second International Timetabling Competition. In the PECTP, points are awarded for enrolling students in any section of a desired course. In addition to all of the constraints in the STP (e.g. no teacher can be assigned to two courses in the same block), the PECTP involves additional student-related hard constraints, such as ensuring that no student is enrolled in multiple sections of the same course.

In this paper, we define the Student Cohort Partitioning Problem (SCPP) to be identical to the PECTP, with these three additional requirements:

(i) Each student $i \in I$ is assigned a learning group $l \in L$.
(ii) Each course section $(c, s) \in C \times S$ is assigned a learning group $l \in L$.
(iii) Student i can be enrolled in section s of course c only if both i and (c, s) are assigned to the same learning group l.

The SCPP can be modeled as a 0-1 integer linear program, where the binary variable $X_{t,s,c,b,l}$ ($Y_{i,s,c,b,l}$) equals 1 if teacher t (student i) is assigned to section s of course c in block b and learning group l, and is equal to 0 otherwise.

In our earlier example with 9 students and 12 cohorts, our optimal solution placed student 8 in learning group 3, assigning this student to courses $3, 6, 9, 12$, which are offered in blocks $1, 2, 3, 4$, respectively. Since each of these courses has a single section ($s = 1$), we have $Y_{8,1,3,1,3} = Y_{8,1,6,2,3} = Y_{8,1,9,3,3} = Y_{8,1,12,4,3} = 1$.

Let $D_{t,c,b}$ be the *desirability* of teacher t being assigned to course c in block b. This coefficient will be a function of teacher t's ability and willingness to teach course c, combined with their availability in block b. Let $P_{i,c,b}$ be the *preference* of student i being enrolled in course c in block b. (We assume that the coefficients $D_{t,c,b}$ and $P_{i,c,b}$ are independent of the section s and learning group l.)

Then, subject to all of the hard constraints, our integer linear program aims to maximize the following objective function:

$$\sum_{t\in T}\sum_{s\in S}\sum_{c\in C}\sum_{b\in B}\sum_{l\in L} D_{t,c,b} \cdot X_{t,s,c,b,l} + \sum_{i\in I}\sum_{s\in S}\sum_{c\in C}\sum_{b\in B}\sum_{l\in L} P_{i,c,b} \cdot Y_{i,s,c,b,l}.$$

In Section 4, we present the hard constraints for this timetabling optimization problem. But first, we present a brief summary of related work.

3 Related Work

Partitioning students into learning groups is a complex challenge faced by school administrators. This explains why scholars in Operations Research have devised innovative techniques to create these student partitions, and apply them to real-life timetabling instances to serve educational institutions all over the world.

Computer Science students at Boston University are optimally matched with peers to form learning groups that increase collective student learning [2] [3]. The practice of creating these cohorts, known as *team formation*, is an NP-Hard combinatorial optimization problem that has been tackled in the last two decades using techniques such as simulated annealing [5], branch and cut [9], and genetic algorithms [24].

Researchers have applied resolvable complete block designs to pre-assign students to groups, ensuring that no pair of students works together on more than one group assignment. This is an application of the Social Golfer Problem [22] to education. For example, Baker et al. [4] deploy both linearized IP and Constraint Logic Programming (CLP) models to maximize exposure of MBA students to each other at Dartmouth College's Tuck School of Business.

As we do in our paper, many educational institutions want to *section students*, i.e. they want to partition students into non-overlapping groups while maximizing the percentage of students being enrolled in their requested courses. A comprehensive introduction to *student sectioning* is found in Kristiansen et al. [15], which presents the High School Student Sectioning (HSSS) problem, formulating it as an integer program, and solving fifteen real-life instances at Danish Schools using Gurobi, a state-of-the-art MIP solver [16].

Over the last decade, there have been two main approaches to solve instances of the Student Sectioning problem: sectioning during course timetabling and batch sectioning after a complete timetable is developed [18].

In the first approach, students who request similar combinations of courses are grouped into the same course sections to minimize potential student conflicts. Thus, the initial sectioning phase precedes the assignment of course sections to time slots. In this sense, sectioning becomes a pre-processing stage prior to timetabling. Two examples of this practice are Carter's homogeneous sectioning [7] and Schindl's regular sub-division of the students using a conflict graph [19].

In the second approach, the task of assigning course sections to time slots is performed first, and only after are students assigned to sections of their desired courses. For example, Müller et al. [18] solve a batch sectioning problem for Purdue University using a heuristic based on an iterative forward search, which progressively searches different neighborhoods of the solution space finding feasible yet incomplete solutions which are optimized in subsequent iterations.

Our Student Cohort Partitioning Problem (SCPP) most closely resembles the batch sectioning formulation, specifically the work by Goebbels et al. [10] which models a batch sectioning problem at Niederrhein University as an IP and finds an optimal solution using IBM's ILOG CPLEX 12.80 solver. To construct the solution, they first optimally partition students into groups of homogeneous sizes and then match these groups to courses previously assigned to time slots.

Despite some similarities, the SCPP differs from the literature in two important ways. First, we partition both students *and* courses into learning groups. Second, each student's timetable comprises of a personalized set of requested courses, rather than the same set of courses as everyone else in their cohort.

In the next two sections, we present the key results of this paper. First, we model the SCPP, an extension of the Post-Enrollment Course Timetabling Problem that was made necessary by the COVID-19 pandemic. Second, we introduce a multi-stage heuristic that finds a close-to-optimal solution of the SCPP, and apply it to generate the Master Timetable for a Canadian high school, partitioning 328 students and 196 course sections into 3 learning groups.

4 Mathematical Model

Let I be the set of individual students, T be the set of teachers, C be the set of courses, S be the set of sections, B be the set of blocks, and L be the set of learning groups.

Our integer linear program (ILP) aims to maximize

$$\sum_{t\in T}\sum_{s\in S}\sum_{c\in C}\sum_{b\in B}\sum_{l\in L} D_{t,c,b}\cdot X_{t,s,c,b,l} + \sum_{i\in I}\sum_{s\in S}\sum_{c\in C}\sum_{b\in B}\sum_{l\in L} P_{i,c,b}\cdot Y_{i,s,c,b,l},$$

where $D_{t,c,b}$ is the desirability coefficient of teacher t being assigned to course c in block b, and $P_{i,c,b}$ is the preference coefficient of student i being assigned to course c in block b.

We define the following four binary variables, two of which appear in our objective function.

(i) For each $t \in T, s \in S, c \in C, b \in B$, and $l \in L$, let $X_{t,s,c,b,l}$ equal 1 if teacher t is assigned to section s of course c in block b and learning group l, and is equal to 0 otherwise.

(ii) For each $i \in I, s \in S, c \in C, b \in B$, and $l \in L$, let $Y_{i,s,c,b,l}$ equal 1 if student i is assigned to section s of course c in block b and learning group l, and is equal to 0 otherwise.

(iii) For each $s \in S$, $c \in C$, and $l \in L$, let $\widehat{X}_{s,c,l}$ equal 1 if section s of course c is assigned to learning group l, and is equal to 0 otherwise.

(iv) For each $i \in I$ and $l \in L$, let $\widehat{Y}_{i,l}$ equal 1 if student i is assigned to learning group l, and is equal to 0 otherwise.

We now present the hard constraints.

Each section of a course can belong to at most one learning group.

$$\sum_{l \in L} \widehat{X}_{s,c,l} \leq 1 \qquad \forall\, s \in S,\ c \in C \tag{1}$$

For every learning group, each section of a course is assigned to exactly one teacher and is scheduled in exactly one block.

$$\sum_{t \in T} \sum_{b \in B} X_{t,s,c,b,l} = \widehat{X}_{s,c,l} \qquad \forall\, s \in S,\ c \in C,\ l \in L \tag{2}$$

No teacher can be assigned to two different courses in the same block.

$$\sum_{s \in S} \sum_{c \in C} \sum_{l \in L} X_{t,s,c,b,l} \leq 1 \qquad \forall\, t \in T,\ b \in B \tag{3}$$

Course c must be timetabled exactly O_c times, where O_c is the number of sections of course c that will be offered.

$$\sum_{t \in T} \sum_{s \in S} \sum_{b \in B} \sum_{l \in L} X_{t,s,c,b,l} = O_c \qquad \forall\, c \in C \tag{4}$$

Each student must belong to exactly one learning group. (Note that teachers may belong to multiple learning groups.)

$$\sum_{l \in L} \widehat{Y}_{i,l} = 1 \qquad \forall\, i \in I \tag{5}$$

No learning group can contain more than M_l students, where M_l is the maximum size of a learning group.

$$\sum_{i \in I} \widehat{Y}_{i,l} \leq M_l \qquad \forall\, l \in L \tag{6}$$

If a student is enrolled in a course section in learning group l, then *both* the student *and* the course section must belong to learning group l.

$$Y_{i,s,c,b,l} \leq \widehat{Y}_{i,l} \qquad \forall\, i \in I,\ s \in S,\ c \in C,\ b \in B,\ l \in L \tag{7}$$

$$Y_{i,s,c,b,l} \leq \widehat{X}_{s,c,l} \qquad \forall\, i \in I,\ s \in S,\ c \in C,\ b \in B,\ l \in L \tag{8}$$

No student can be enrolled in more than one course in the same block.

$$\sum_{s \in S} \sum_{c \in C} \sum_{l \in L} Y_{i,s,c,b,l} \leq 1 \qquad \forall\, i \in I,\ b \in B \tag{9}$$

No student can be enrolled in multiple sections of the same course.

$$\sum_{s \in S} \sum_{b \in B} \sum_{l \in L} Y_{i,s,c,b,l} \leq 1 \qquad \forall\, i \in I,\ c \in C \tag{10}$$

Section s of course c can have at most $M_{s,c}$ students, where $M_{s,c}$ is the maximum enrollment for this course section.

$$\sum_{i \in I} \sum_{l \in L} Y_{i,s,c,b,l} \leq M_{s,c} \qquad \forall\, s \in S,\ c \in C,\ b \in B \tag{11}$$

This is our model for the Student Cohort Partitioning Problem (SCPP). Our solution is found by maximizing the objective function of this integer linear program subject to these eleven constraints.

In practice, the large majority of these variables $X_{t,s,c,b,l}$ and $Y_{i,s,c,b,l}$ will be pre-set to 0, since teachers are qualified to only teach a small subset of the offered courses, and students will only want to enroll in a small subset of these courses. By fixing these variables to be 0, we can solve the SCPP whenever our sets $|T|, |I|, |S|, |C|, |B|, |L|$ are of reasonable size.

While our ILP model is guaranteed to output an optimal solution, the computing time grows exponentially as the problem size increases. Thus, for a large school with hundreds of students and course offerings, we might not be able to solve the ILP. This motivates the need for approximation algorithms.

We conclude this section of the paper by proposing two heuristics: a multi-step approach called "Progressive Assignment" that orders the courses by the preference coefficient $P_{i,c,b}$, and a Large Neighbourhood Search (LNS). Both heuristics are inspired by the principle that when tackling an NP-complete problem, it is good practice to reduce the initial intractable problem to a series of simpler tractable subproblems [6].

In our first heuristic, we divide the course sections into k different groups, sorted by course priority. In other words, courses with the highest preference coefficients $P_{i,c,b}$ are partitioned first. We solve the SCPP on this smaller subset of courses in C, and lock in the assignments $\widehat{X}_{s,c,l}$ found in the optimal solution. We then include these assignments as hard constraints in the following step, where we solve the SCPP on the next subset of courses in C.

Thus, we build the timetable progressively, assigning only a subset of the course sections at a time. While the $\widehat{X}_{s,c,l}$ variables stay fixed throughout our Progressive Assignment algorithm, the values of $\widehat{Y}_{i,l}$ and $Y_{i,s,c,b,l}$ change in each of the k steps.

We allow this because once the $\widehat{X}_{s,c,l}$ assignments are set, our ILP quickly finds the best possible assignment of students to learning groups, and students to course sections, to generate the optimal solution at each step. This ensures that the final solution produced by the Progressive Assignment is close (although most certainly not equal) to the optimal solution for the original SCPP.

In our second heuristic, we employ a Large Neighbourhood Search to iteratively improve our solution. Given any solution to the SCPP (e.g. the solution found in our Progressive Assignment), we lock in all but h of the variables $\widehat{X}_{s,c,l}$, set them as hard constraints, and then re-calculate the SCPP to generate a new solution where some of these h course sections may be re-assigned to different learning groups. Since our initial solution was produced by one of the $|L|^h$ assignments of learning groups to these h course sections, our new SCPP solution cannot be worse than our input solution. Thus, our LNS is analogous to "hill climbing" because it performs iterative and incremental changes on an arbitrary initial solution to find a better solution.

We may stop the search at any time, either after a fixed time limit or when the algorithm appears to have converged. This heuristic, like all local search algorithms, may get stuck in a local minimum, especially if the value of h is small. However, when h is sufficiently large, the results of the LNS get better at each step, until a close-to-optimal solution is found.

When these two algorithms are combined, we can rapidly generate a nearly-optimal (or possibly optimal) solution to complex SCPP problems. We now apply these algorithms on a real-world instance, an all-boys high school in Canada.

5 Application

St. George's School (SGS) is located in Vancouver, the most populous city in the Canadian province of British Columbia. SGS is one of Canada's leading independent schools, with an enrollment of 1200 students. Founded in 1930, the school's mission is to "inspire their students to become fine young men who will shape positive futures for their families and the global community".

For the SGS administration, the biggest challenge is creating the timetable for the students in Grades 11 and 12. In 2020-2021 this represented 328 students, with 165 juniors and 163 seniors at this all-boys high school. Unlike students in the lower grades who take mostly required (core) courses, there are numerous elective courses in the final two years, and each student wants to enroll in a different combination of courses from the over *one hundred* options available.

The majority of these course options are offered to students in both Grades 11 and 12, and traditionally the school has viewed students from these two grades as a single cohort. Due to the government's mandate of a 120-person cohort limit, these 328 students needed to be partitioned into $|L| = 3$ learning groups.

The $|I| = 328$ individual students requested a total of 2303 courses, which is fewer than the maximum total of $|I| \times |B| = 328 \times 8 = 2624$. This occurred because the students could take a "self-study period", i.e., a spare block.

Certain courses were canceled due to low enrollment. Based on the student requests, the school decided to offer $|C| = 89$ different courses, of which 41 were single-section courses. Of the 48 multi-section courses, many had two or three sections, though one course (Social Studies 11) had $|S| = 8$ sections, since this course was mandatory for all of the Grade 11 students. In all, there were 196 total course sections.

The SGS timetable has $|B| = 8$ blocks, and each of the 196 course sections needed to be scheduled in one of these 8 blocks.

Each school day consists of four 70-minute class periods and one lunch break, with a four-day "tumbling timetable". Students alternate between their four blocks (A, B, C, D) on Days 1 and 3 and their four blocks (E, F, G, H) on Days 2 and 4, repeating this pattern for the entire academic year.

Class	Day 1	Day 2	Day 3	Day 4
Period 1	A	E	C	G
Period 2	B	F	D	H
Lunch	Lunch	Lunch	Lunch	Lunch
Period 3	C	G	A	E
Period 4	D	H	B	F

Since the assignment of blocks to days and time slots is fixed by the school administration, creating the optimal timetable is equivalent to optimally assigning course sections to blocks.

The school leadership team also *pre-assigned* each of the 196 course sections to one of the $|T| = 50$ teachers at the senior school. Pre-assigning teachers to course sections reduces our ILP's objective function to maximizing student preferences.

Mathematically this is equivalent to setting the desirability coefficient $D_{t,c,b}$ to 0 if teacher t is assigned to at least one section of course c in some block b, and ensuring a hard constraint of $X_{t,s,c,b,l} = 0$ whenever teacher t is not assigned to section s of course c. The pre-assignment of course sections to teachers reduced the problem size from $(|T| + |I|) \cdot |S| \cdot |C| \cdot |B| \cdot |L|$ to a smaller problem with $|I| \cdot |S| \cdot |C| \cdot |B| \cdot |L| = 328 \times 196 \times 8 \times 3 = 1542912$ total binary variables.

The school set the following weights for the preference coefficients $P_{i,c,b}$:

5 points if c is a high-priority elective course
3 points if c is a medium-priority elective course
1 point if c is a low-priority elective course

Of the 196 course sections, 149 were elective courses while the remaining 47 represented *required* courses – i.e., a student desiring a required course had to be registered in at least one section of that course.

We modeled high-priority courses as soft constraints assigning them a large weight in our student course preference matrix, whereas *required* courses were treated as hard constraints which we hard coded in our ILP.

For their entire history, St. George's School has created their timetable manually. Given the challenges of solving a combinatorial optimization by hand, the school has always pre-assigned each course section to one of the eight blocks *before* creating each student's timetable.

While this pre-assignment ensures that the educators know their exact teaching schedule before they go on their summer holidays, this of course has a significant impact on the Objective Function, since each course section is locked into a block rather than optimized to maximize students getting into their desired courses. (The school has hired us to create their 2021-2022 timetable, which will not include teacher pre-assignments or block pre-assignments.)

Our optimization program, written in Python, inputs an Excel sheet consisting of all the course data and the individual student requests. For the actual optimization, we use COIN-OR Branch and Cut (CBC), an open-source MIP solver, with the Google OR-Tools linear solver wrapper [11].

We first generated the optimal timetable for just $|L| = 1$ learning group, which was trivial since course sections were pre-assigned to blocks. Using the model described in Section 4, our ILP computed the solution in just 8.5 seconds on an 8GB Lenovo laptop running Windows 10 with a 2.1 GHz processor. The Python code and input files deployed to design the optimal timetable are found in a repository at https://github.com/ifabrisarabellapark/CPAIOR2021.

The results are presented below, grouped by course priority.

Priority Type	Total Requested	Total Enrolled
Required	807	807
High	541	533
Medium	166	155
Low	789	727
TOTAL	2303	2222

When there is only one learning group (i.e., Pre-COVID), the objective function of our optimal timetable has value $533 \times 5 + 155 \times 3 + 727 \times 1 = 3857$, with students being enrolled in $2222/2303 = 96\%$ of their desired courses.

In the $2303 - 2222 = 81$ instances where a student was not assigned a section of a desired course, the majority of them were due to over-capacity, including all $8 + 11 = 19$ of the unassigned High and Medium Priority requests.

Fortunately, all 19 of these requests had a reasonable alternative – for example, the five students not getting into AP Chemistry 12 could instead take Chemistry 12, which was offered in the same block and had plenty of available seats.

We then applied our ILP model to partition the students and course sections, to find a solution to our Student Cohort Partitioning Problem (SCPP). By definition, we knew that we could not exceed a success rate of 96%.

Since SGS requested learning groups to be evenly balanced, we set $M_l = 110$ as the maximum cohort size since we had $|I| = 328$ students and $|L| = 3$ learning groups. As expected, our Python program could not solve the ILP within our pre-set limit of 12 hours, and so we applied our approximation algorithms.

We first solved the ILP for just the Required courses, and locked in the learning groups for these 47 course sections. We then performed our Progressive Assignment for the remaining course sections in order of priority: High, then Medium, then Low. This entire process took less than five minutes, giving us an initial assignment of course sections to learning groups ($\widehat{X}_{s,c,l}$), students to learning groups ($\widehat{Y}_{i,l}$) and students to course sections ($Y_{i,s,c,b,l}$).

We then applied the Large Neighbourhood Search (LNS), which locked in the learning groups for $196 - h$ course sections, while allowing the remaining h course sections to be re-assigned by our ILP. We found that reshuffling $h = 30$ was ideal for our problem size, so that each step of the search computed in an average time of 60 seconds. Our local search algorithm converged to the same solution within two hours for every single trial we ran.

This solution scores $518 \times 5 + 147 \times 3 + 525 \times 1 = 3556$ points, with students getting into $1997/2303 = 87\%$ of their desired courses, including 100% of their required courses. The results are presented below.

Priority Type	Total Requested	Progressive Assignment	LNS
Required	807	807	807
High	541	514	518
Medium	166	144	147
Low	789	524	525
TOTAL	2303	1989	1997

Our Master Timetable, with students being enrolled in 87% of their desired courses, was delivered to SGS on August 20, 2020, less than two weeks after the authors were introduced to this project. This gave plenty of time for the school administrators to announce the learning groups of each student and each course section, and set up each student's on-campus activities (co-curriculars, lunch, entrance times, and exit times) to occur exclusively within their learning group.

Each student was given a coloured ID card (Orange, Pink, Grey) corresponding to their learning group, in addition to their 8-block timetable. In the 13% of cases where the students were not given a desired course, the Registrar's Office made simple adjustments, such as enrolling students in a similar course (e.g. AP Chem 12 to Chem 12), and inviting students to register for a different elective course. Within a few days, each student had a satisfactory timetable.

St. George's School has hired the authors to build the 2021-2022 Master Timetable and has decided to no longer pre-assign course sections to teachers and blocks. By doing this, the students will be able to get into a higher percentage of their desired courses, regardless of whether the number of learning groups stays at 3, or returns to 1 in a post-COVID world.

6 Discussion

Our final model for the Student Cohort Partitioning Problem (SCPP) has a total of $|I||S||C||B||L| = 328 \times 196 \times 8 \times 3 = 1542912$ total binary variables. The size of this model prevented us from confirming that our solution was optimal. Naturally, we wondered how close our implemented solution, with an 87% success rate and an objective value of $518 \times 5 + 147 \times 3 + 525 \times 1 = 3556$ points, was to the optimal solution for $|L| = 3$ learning groups.

To answer this question, we considered the easier problem of $|L| = 2$ learning groups with at most $M_l = 220$ students in a cohort. If we restrict the set of elective courses to just one priority class (High, Medium, Low) at a time, then we can solve each of these three separate ILPs, each in just a few minutes.

Any solution to the SCPP with $|L| = 3$ and $M_l = 110$ is automatically a solution to the easier SCPP with $|L| = 2$ and $M_l = 110 + 110 = 220$. Therefore, our results for $|L| = 2$ and $M_l = 220$, marked in bold, provide a theoretical upper bound to our optimization problem for St. George's School with $|L| = 3$.

Priority Type	Total Requested	$\|L\| = 2, M_l = 220$	$\|L\| = 3, M_l = 110$
Required	807	**807**	807
High	541	**518**	518
Medium	166	**155**	147
Low	789	**682**	525
Total Assignments	2303	**2162**	1997
Objective Value	3992	**3737**	3556

The implemented solution scoring 3556 points is less than 5% below the theoretical upper bound which scored $518\times5+155\times3+682\times1 = 3737$ points. Our solution is provably optimal for the set of Required and High priority courses.

We presume that the actual optimal solution is much closer to 3556, since our model mandates we enroll students in all of the elective courses *simultaneously*, rather than treating each priority class *separately*, as we did when computing the upper bound. Thus, our construction is an overestimate of the actual optimal solution for our problem with $|L| = 3$ and $M_l = 110$.

We were fortunate to find a solution satisfying 87% of student course requests, since partitioning students into groups could have led to a terrible outcome for the school, had the students selected a more heterogeneous set of courses.

To illustrate this point, consider a small school with $|I| = 32$ students, where each student is required to select five courses, one from each of the five sets: $\{A_1, A_2\}$, $\{B_1, B_2\}$, $\{C_1, C_2\}$, $\{D_1, D_2\}$, and $\{E_1, E_2\}$. Furthermore, suppose that each of the $2^5 = 32$ students selects a different set of 5 courses.

Assume each classroom can hold at most 16 students. Since there are 16 students who request each course, there is only the need to offer one section of each course. If learning groups do not exist (i.e., $|L| = 1$), then it is trivial to enroll all 32 students into all 5 of their courses. We simply assign each pair of courses (e.g. A_1 and A_2) to be taught in the same time slot, which guarantees a timetable where all $32 \times 5 = 160$ course requests are satisfied.

Now suppose $|L| = 2$, and we need to partition the $|I| = 32$ students and $|C| = 10$ courses into two learning groups. It is straightforward to see that each course partition is identical, by symmetry. For example, if we assign courses $\{A_1, B_1, C_1, D_1, E_1\}$ to the first learning group and courses $\{A_2, B_2, C_2, D_2, E_2\}$ to the second learning group, then each student is simply assigned to the learning group that gives them the most number of desired courses. For example, the student who wants $\{A_2, B_1, C_2, D_1, E_2\}$ is assigned to the second learning group since this student would prefer a timetable with 3 desired courses instead of 2.

We can show that 2 students get into all five courses, 10 students get into four courses, and 20 students get into three courses. Assuming each desired course assignment has a score of 1 point, the objective value is $2\times5+10\times4+20\times3 = 110$, resulting in a success rate of $\frac{110}{160} = 0.6875$. By making the small switch from $|L| = 1$ learning groups to $|L| = 2$, our success rate drops from 100% to 68.75%.

This construction generalizes, from 2^5 students and 10 courses to 2^n students and $2n$ courses, showing that when there are $|L| = 2$ learning groups, it is possible for the students to select their courses so that the optimal solution to the SCPP results in a success rate that converges to $\frac{1}{2} = 50\%$ as $n \to \infty$.

For all $|L| > 2$, we conjecture that a similar construction with $|L|^n$ students and $|L|$ learning groups results in the optimal SCPP solution having a success rate converging to $\frac{1}{|L|}$. We have verified this result for $|L| = 3$ and $|L| = 4$ using Python. In other words, creating learning groups has a significant impact on students being assigned their desired courses, especially when there are many single-section courses and the students have heterogeneous course preferences.

We were fortunate that the $|I| = 328$ students at St. George's School selected their courses in such a way that enabled us to enroll them in 87% of their desired courses (1997 out of 2303). This success percentage is notable given that our theoretical upper bound shows that at most 2162 of these 2303 course requests can be fulfilled, even with $|L| = 2$ learning groups.

7 Conclusion

In this paper, we defined the Student Cohort Partitioning Problem (SCPP) and modeled it using an integer linear program. We then applied our model on a real-world problem instance at a Canadian high school, partitioning $|I| = 328$ students into $|L| = 3$ learning groups while enrolling these students in 87% of their desired courses, including 100% of their required courses.

Our collaboration with St. George's School was a success, and the implementation of the learning groups was done smoothly and effectively. This is part of the reason why the school has reported *zero* COVID-19 cases among the students and staff of the Senior School thus far in the 2020-2021 academic year.

We recognize that our research on the SCPP has only begun, and further research will need to be conducted to assess the scalability of our work to larger data sets. A natural first step is to take the set of benchmark instances for the Post-Enrollment Course Timetabling Problem (PECTP) and amend them with two or more learning groups, to create benchmark instances for the SCPP.

The authors have signed contracts with five different high schools in British Columbia, to build each school's 2021-2022 Master Timetable. Here are the questions we will be asking ourselves as we proceed with our work in the coming months:

(a) Could SCPP solutions be improved by adding a few more sections of certain courses? And if so, what would be the cost to the school? If the school had enough money to add X extra sections, which sections should be added to maximize the number of students being assigned their desired courses?

(b) How will our SCPP model handle additional requirements, such as requiring *teachers* to belong to a single learning group? Should our model assign a penalty whenever a teacher is assigned to multiple learning groups, and include this penalty in the objective function?

(c) Given rising COVID-19 numbers in Canada (especially with new variants), some schools may decide to restrict the number of days that each learning group attends the school. How would this policy change affect our timetabling solutions?

(d) What are the best techniques to find fast nearly-optimal solutions to larger SCPP instances? For example, would Constraint Programming techniques and/or Benders decomposition outperform the methods presented in this paper?

In British Columbia, schools will not "return to normal" until they reach the final phase of the provincial Back to School plan, which is conditional on wide vaccination and immunity among the population. As a result, schools in our province, as well as in other areas of Canada and the world, will likely need to design their 2021-2022 timetables to include multiple Learning Groups.

While the global pandemic continues, our SCPP formulation is extremely applicable for educational institutions, as our model enables schools to create cohorts to maximize students being able to take their desired courses, while simultaneously reducing the spread of COVID-19 through optimal partitioning.

8 Acknowledgments

The authors thank the reviewers for their insightful comments that significantly improved the presentation of this paper. The authors also thank the administrators at St. George's School for making this collaboration possible. Specifically, we acknowledge Sarah Coates (Associate Principal of Academics), Andrew Shirkoff (Director of Risk Management), Jan Chavarie (Head of Applications Support), and Jessie Bahia (Registrar).

References

1. B.C.'s Back to School Plan, https://www2.gov.bc.ca/gov/content/education-training/k-12/covid-19-return-to-school#learning-group. Last accessed 12 Apr 2021
2. Bahargam, S., Erdos, D., Bestavros, A., Terzi, E.: Personalized Education; Solving a Group Formation and Scheduling Problem for Educational Content. In: Proceedings of the 8th International Conference on Educational Data Mining, pp. 488-492. International Educational Data Mining Society, Madrid (2015)
3. Bahargam S, Erdos D, Bestavros A, Terzi E. Team Formation for Scheduling Educational Material in Massive Online Classes. arXiv preprint arXiv:1703.08762 (2017)
4. Baker, K. R., Magazine, M. J., Polak, G. G.: Optimal block design models for course timetabling. Operations Research Letters **30**(1), 1–8 (2002)
5. Baykasoglu, A., Dereli, T., Das, S.: Project team selection using fuzzy optimization approach. Cybernetics and Systems: An International Journal **38**(2), 155–185 (2007)
6. Bessiere, C., Carbonnel, C., Hebrard, E., Katsirelos, G., Walsh, T.: Detecting and exploiting subproblem tractability. In: IJCAI International Joint Conference on Artificial Intelligence, pp. 468–474. AAAI Press, California (2013)
7. Carter, M. W.: A comprehensive course timetabling and student scheduling system at the University of Waterloo. In: Burke E., Erben W. (eds) PATAT 2000, LNCS, vol. 2079, pp. 64–82. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44629-X_5
8. CDC Operational Considerations for Schools, https://www.cdc.gov/coronavirus/2019-ncov/global-covid-19/schools.html. Last accessed 12 Apr 2021
9. Chen, S. J., Lin, L.: Modeling team member characteristics for the formation of a multifunctional team in concurrent engineering. IEEE Transactions on Engineering Management **51**(2), 111–124 (2004)
10. Goebbels, S., Pfeiffer, T.: Optimal Student Sectioning at Niederrhein University of Applied Sciences. In: Neufeld, J. S., Buscher, U., Lasch, R., Möst, D., Schönberger, J. (eds.) Operations Research Proceedings 2019, pp. 167—173. Springer, Cham, (2020). https://doi.org/10.1007/978-3-030-48439-2_20
11. Google OR-Tools: Fast and portable software for combinatorial optimization, https://developers.google.com/optimization. Last accessed 12 Apr 2021
12. Hoshino R., Fabris I.: Optimizing Student Course Preferences in School Timetabling. In: Hebrard E., Musliu N. (eds.) CPAIOR 2020, LNCS, vol. 12296, pp. 283–299. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58942-4_19
13. Johns Hopkins University & Medicine, https://coronavirus.jhu.edu/map.html. Last accessed 12 Apr 2021
14. Khlaif, Z. N., Salha, S.: The unanticipated educational challenges of developing countries in Covid-19 crisis: A brief report. Interdisciplinary Journal of Virtual Learning in Medical Sciences **11**(2), 130–134 (2020).
15. Kristiansen, S., Sørensen, M., Stidsen, T. R.: Student Sectioning at High Schools in Denmark. In: 6th Multidisciplinary International Conference on Scheduling: Theory and Applications, pp. 628-632. Springer, Belgium (2013)
16. Kristiansen, S., Sørensen, M., Stidsen, T. R.: Integer programming for the generalized high school timetabling problem. Journal of Scheduling **18**(4), 377–392 (2015)
17. Lewis, R., Paechter, B., McCollum, B.: Post enrolment based course timetabling: A description of the problem model used for track two of the second international timetabling competition (2007)

18. Müller, T., Murray, K.: Comprehensive approach to student sectioning. Annals of Operations Research **181**(1), 249–269 (2010)
19. Schindl, D.: Student sectioning for minimizing potential conflicts on multi-section courses. In: Proceedings of the 11th International Conference of the Practice and Theory of Automated Timetabling (PATAT 2016) pp. 327-–337. Springer, Udine, Italy (2016)
20. The Hill: Coronavirus shining light on internet disparities in rural America, `https://thehill.com/blogs/congress-blog/technology/488848-coronavirus-outbreak-shining-an-even-brighter-light-on`. Last accessed 12 Apr 2021
21. The Squamish Chief: concern about mixed-cohort classrooms, `https://www.squamishchief.com/news/local-news/amid-covid-19-worries-concern-emerges-about-mixed-cohort-classrooms-1.24199269`. Last accessed 12 Apr 2021
22. Triska, M.,; Musliu, N.: Solving the Social Golfer Problem with a GRASP. In: Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling, (PATAT 2008). Springer, Montréal (2008)
23. UNESCO: COVID-19 impact on education, `https://en.unesco.org/covid19/educationresponse`. Last accessed 12 Apr 2021
24. Wi, H., Oh, S., Mun, J., Jung, M.: A team formation model based on knowledge and collaboration. Expert Systems with Applications **36**(5), 9121–9134 (2009)