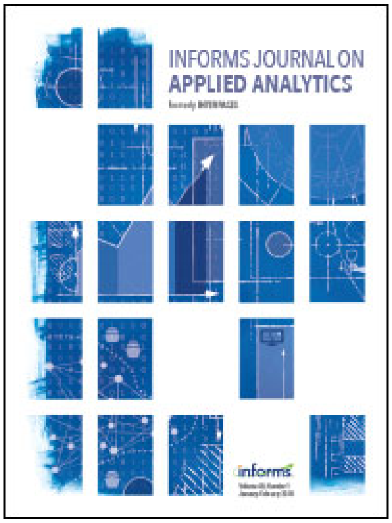


INFORMS Journal on Applied Analytics

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Daily Tutor Scheduling Support at Hopeful Journeys Educational Center

Matthew D. Bailey, Lucas A. Waddell

http://pubsonline.informs.org/journal/inte

INFORMS JOURNAL ON APPLIED ANALYTICS
Vol. 50, No. 5, September–October 2020, pp. 287–297
ISSN 0092-2102 (print), ISSN 1526-551X (online)

Daily Tutor Scheduling Support at Hopeful Journeys Educational Center

Matthew D. Bailey,[a] Lucas A. Waddell[a]

[a] Bucknell University, Lewisburg, Pennsylvania 17837

Contact: matt.bailey@bucknell.edu, https://orcid.org/0000-0002-7513-5941 (MDB); lucas.waddell@bucknell.edu, https://orcid.org/0000-0002-1781-4049 (LAW)

https://doi.org/10.1287/inte.2020.1039

Abstract. Hopeful Journeys Educational Center faces a daily task of assigning tutors to students subject to myriad complex rules and restrictions. The organization's mission, which is to provide individualized education to students with autism spectrum disorders and other developmental disabilities, as well as its limited operating budget and day-to-day resource/demand variability, makes this a uniquely challenging scheduling problem. When we first communicated with Hopeful Journeys, the organization was in critical need of an efficient methodology for producing daily schedules to replace its existing time-consuming and error-prone manual approach. This paper describes the fully open-source, Excel-based optimization tool we developed to support Hopeful Journeys' mission. Our work illustrates the potential to use freely available operations research tools within a "rapid prototyping" approach to provide immediate impact to organizations that lack the resources to utilize commercial software or professional consultants.

Keywords: scheduling • open source • integer programming • spreadsheet modeling

Introduction

Hopeful Journeys Educational Center is a nonprofit school in Beverly, Massachusetts, whose mission is to "provide quality and individualized education to children with autism spectrum disorders and other developmental disabilities" (Hopeful Journeys 2019). This individualized education is implemented through the one-on-one scheduling of approximately 100 tutors to approximately 100 students within 30-minute blocks of time. Students attend the school five days a week from 8:30 a.m. to 2:30 p.m., but the availability of tutors and needs of specific students can vary greatly from day to day. The development of the daily schedule is solely the responsibility of a single school-wide scheduler. In the past, this process involved updating the student need and tutor availability for the following day, cross-referencing 30 worksheets containing information about the students and the tutors' qualifications, and manually building the schedule to satisfy numerous rules and preferences. A previously custom-built Excel macro would check the schedule for errors such as students with unscheduled periods, missing lunch breaks for tutors, and students being assigned to more than one tutor in a period. Each day, the scheduler devoted more than 10 hours to the next day's schedule. As would be expected, this was a challenging and error-prone process requiring a daily midday scramble to correct problems and omissions. This scramble negatively impacted both the work environment for tutors and the learning environment for students, who find transitions challenging.

In the fall of 2018, a new human scheduler was being trained, and he sought help with the organization's scheduling process using an online Excel forum (Reddit's r/Excel). As participants in this forum, we identified that this problem required more sophistication than a simple spreadsheet. We contacted the scheduler directly to inquire further, and, upon learning of the organization's mission and challenges, offered our assistance.

Because of its tightly constrained budget, Hopeful Journeys was seeking a cost-effective solution. Working closely with the new scheduler, we developed an open-source, optimization-based decision support system embedded within its software already in use, Microsoft Excel. This model has been in use since January of 2019, and, as noted within the letter of support (provided at the end of this paper), it has significantly reduced the total time required for the daily scheduling process to under four hours. This process includes gathering all the information on tutor and student availability and obligations for the next day, running the tool, and disseminating the schedule. The scheduler has noted that "the model has really become an integral part of the scheduling; it's truly unbelievable how well it works as we continue to refine how the school functions."

Although it may appear that this is a standard appointment-scheduling model, there are a variety of conditions and preferences, which we show in the bulleted list below, that make this problem a non-standard personnel scheduling of resources (tutors) to jobs (students).

- Students have varying demand for tutoring throughout the day.
- An individual tutor's availability varies throughout the day.
- Each tutor must be scheduled for a 30-minute lunch break within a given window of time each day. The tutors are unavailable to meet with students during this time.
- Tutor qualifications for working with particular students vary.
- Sessions between a student and a tutor can vary from the preferred standard (one hour) to rushed (30 minutes) and even extended (1.5 hours).
- Students have a preferred set of tutors (within assigned teams), but tutors on other teams may assist as necessary (if qualified).
- There are two priority classes of tutors: managers and staff, with preference given to staff, so that managers are free to perform other duties.
- Tutoring capacity varies between one and two students during a period. Each student has a limited set of students for possible pairing, but this is to be avoided if possible.
- Tutors have daily limits on total time with each student and total successive time with the same student.

Van den Bergh et al. (2013) provide a comprehensive review of over 300 papers covering various facets of personnel scheduling, classifying manuscripts by their use of hard and/or soft constraints for addressing issues such as those we list above. Of these manuscripts, only 15% were applied in practice, and education was not identified as a represented application area. Johnes (2015) reviews the published uses of operations research within education administration, including the development of decision-support systems for timetabling (i.e., assigning instructors to courses and time slots) (Foulds and Johnson 2000, Miranda et al. 2012). Although sharing some common features with timetabling, Hopeful Journeys' scheduling task is more complex and is more closely related to problems related to physician scheduling. A complex problem in its own right, physician scheduling is described by Schoenfelder and Pfefferlen (2018, p. 215) as "one of the most complex topics in personnel scheduling due to the high heterogeneity among the individual employees with respect to qualifications and contractual agreements as well as the demand for very specific workforce compositions." Our tutor scheduling shares these same complications, in addition to the challenge of daily, rather than monthly, determination.

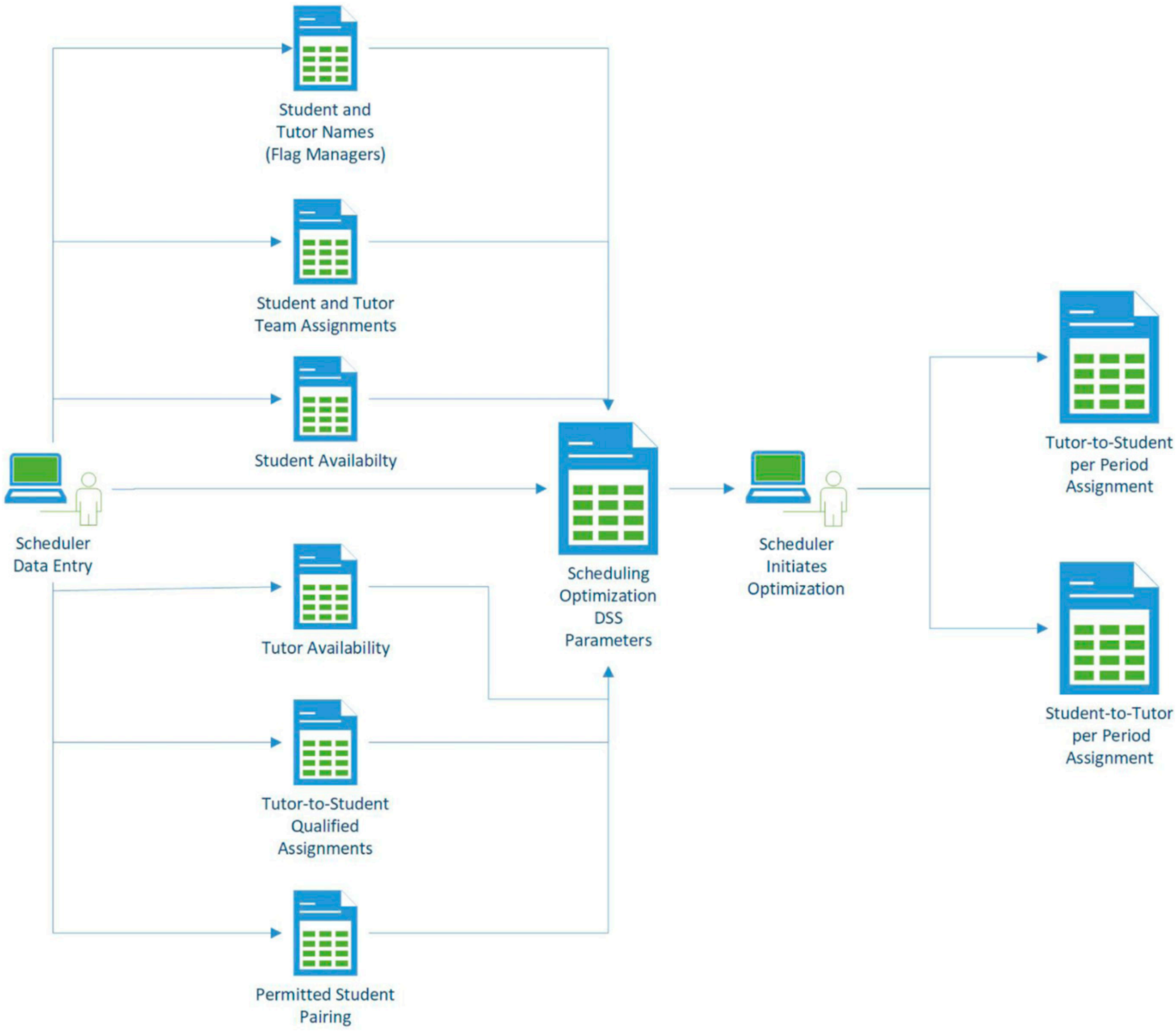
Tool Development and Implementation

Given the size and complexity of the problem, Hopeful Journeys did not have the expertise to model or optimize its daily schedules. Our goal was to develop a tool that the scheduler could use efficiently to create daily student/tutor assignments. We opted for a binary integer programming (BIP) solution strategy over a heuristic approach. This choice was not only for stability and ease of implementation, but also because the problem structure varies greatly from day to day based on student attendance and tutor availability. Retuning a heuristic daily would be burdensome to the scheduler. The standard branch-and-cut solution methodology for BIP has the added advantage that it can be terminated prior to convergence and still provide an incumbent feasible solution.

The final BIP model involves tens of thousands of variables and constraints, which would typically require powerful (and expensive) optimization solvers. Unfortunately, such an expense is impractical for Hopeful Journeys. To this end, we developed the model using a combination of the open-source Python-based optimization modeling package PuLP (Mitchell et al. 2009) and the open-source optimization engine CBC (Forrest et al. 2018). These tools, which are freely available through the open-source initiative Computational Infrastructure for Operations Research (Lougee-Heimer 2003), are included within a free add-in for Microsoft Excel, SolverStudio (Mason 2013). SolverStudio provides open-source support for optimization modeling languages and acts as an interface between the modeling language and the spreadsheet. By using Visual Basic for Applications (VBA), the solution is reported in an easily editable format for direct distribution to managers. We chose an Excel-based framework due to the scheduler's expressed comfort with and preference for this software. There is also precedent from other successful spreadsheet scheduling support systems (Şeref and Ahuja 2008, Heider et al. 2018, Schoenfelder and Pfefferlen 2018) and our past success in the development and implementation of optimization-based decision support systems within this framework (Bailey and Nowak 2018, Bailey and Michaels 2019).

A schematic of the data needs and workflow for our decision support system (DSS) is given in Figure 1. The data workbook is independent of the workbook with the optimization model and report formatting. This permits the scheduler to have a data file for each day of the week and update the files as necessary when student absences and tutor unavailability are announced.

Figure 1. (Color online) The Flowchart Illustrates the Work Flow for the Tutor Scheduling DSS



The data file includes a variety of worksheets where the scheduler inputs the data needed to run the model. Specifically, the scheduler enters all the student and tutor names (Figure 2(a)), the periods and potential lunch periods (Figure 2(b)), and team affiliations for both students and tutors (Figure 3). Additionally, managers are identified, as well as any secondary team to which a tutor belongs. Teams, which we discuss later in this paper, are subgroups of students and tutors within which the scheduler prefers to make assignments. There is also a student-by-tutor matrix, where the scheduler can indicate which tutors are qualified to work with which students (Figure 4), and a student-by-student matrix to indicate which students may be paired together if necessary (Figure 5). Additionally, the scheduler can indicate which periods each tutor is unavailable to work with a student due to other work or personal obligations (Figure 6). The text entered in this grid will appear in the final schedule output. A similar grid is available to indicate in which periods a student requires a tutor (Figure 7). Included in each of these grids are prescheduled student lunches, where groups of students eat together with a preassigned chaperone. If the chaperone is a tutor, this will be viewed as a busy period for the tutor, but does not count as personal lunch on the schedule.

As with many scheduling applications, the Hopeful Journeys problem is overconstrained; schedules that satisfy all the school's wishes simply do not exist. We, therefore, found it to be critical to strike an appropriate balance between formulating hard constraints

Figure 2. (Color online) These Input Tables Contain the Names, Manager Flags, and Tutoring Period Information

(a)

Student Names		Tutor Names	Manager?
JAY		SB	1
JO		MT	0
LA		SAR	0
Lei		AV	0
CA		JOS	0
CU		KS	0
PA		AIN	1
ME		JEN	0
LW		NOE	0
JM		AM	0
CAL		CAY	0
EM		HA	0
JG		KAY	0

(b)

Starting Time	Lunch Period?		Teams
8.5	0		team 1
9	0		team 2
9.5	0		
10	0		
10.5	1		
11	1		
11.5	1		
12	1		
12.5	1		
13	1		
13.5	1		
14	0		

Notes. (a) Student and tutor names. (b) Tutoring and potential lunch periods.

that must be satisfied and soft constraints that may be violated by incurring a penalty in the objective function. By enforcing too many hard constraints, we would run the risk of creating infeasible optimization problems. Too few hard constraints, however, would cause the optimization to produce solutions that are not practical for Hopeful Journeys to implement. We worked closely with the scheduler to partition the scheduling constraints into these two categories.

In our final model, hard constraints include the constraints we show in the bulleted list below; with each, we note the associated mathematical constraint(s) found in the appendix with a fuller description and as part of the full mathematical model formulation.

• Each student can be assigned at most one tutor in each period that the student requires a tutor. A student cannot be assigned a tutor in any period that the student does not require a tutor (A.3(d)).

• An assignment cannot be made in a period if the tutor is unavailable (A.3(e)).

• A tutor cannot be assigned to a student with whom the tutor is not qualified to work (A.3(f)).

Figure 3. (Color online) This Input Table Contains the Team Affiliation for Each Student and Tutor

Students	Team		Tutors	Team	Second Team
JAY	team 1		SB	team 1	0
JO	team 1		MT	team 1	0
LA	team 1		SAR	team 1	0
Lei	team 1		AV	team 1	0
CA	team 1		JOS	team 1	0
CU	team 1		KS	team 1	0
PA	team 1		AIN	team 2	0
ME	team 2		JEN	team 2	team 1
LW	team 2		NOE	team 2	0
JM	team 2		AM	team 2	0
CAL	team 2		CAY	team 2	0
EM	team 2		HA	team 2	0
JG	team 2		KAY	team 2	0

• Students may only be paired together if the pairing is approved (A.3(g)).

• A student cannot be assigned to the same tutor for more than two-and-a-half total hours during the day (A.3(h)).

• A student cannot be assigned to the same tutor for more than one hour and a half in succession (A.3(i)).

• A tutor cannot be assigned lunch if the tutor is also assigned to a student during that period (A.3(j)).

• A tutor cannot be assigned lunch during a period if the tutor is unavailable due to other obligations (A.3(k)).

• A tutor must be assigned exactly one lunch period, unless the tutor is busy with other obligations during all lunch periods (A.3(l)).

Soft constraints, on the other hand, represent "preferred, but not necessary" requirements. A solution is allowed to violate these constraints if it reduces the number of students with unassigned periods. In our final model, there are five soft constraints, each associated with a penalty weight ρ. These weights can be manually adjusted by the scheduler (Figure 8) to indicate that some soft-constraint violations might be more or less preferable than others.

Taken together, the following soft constraints indicate that Hopeful Journeys prefers tutoring assignments that are one-on-one, one-hour appointments with regular staff (not managers) from the same team. We provide the mathematical constraint(s) in the formulation in the appendix.

• The penalty ρ^{three} is incurred if a student is assigned to the same tutor for a one-and-a-half hour block of time (A.3(m)).

• The penalty ρ^{iso} is incurred if a student is assigned to a tutor for an isolated 30-minute block of time (A.3(n)–(p))

• The penalty ρ^{pair} is incurred if two students are paired with the same tutor in the same period (A.3(q)).

• The penalty ρ^{team} is incurred if a student is assigned to a tutor from a different team.

Figure 4. (Color online) This Input Table Contains the Tutor-to-Student Pairing Work-Qualification Flags

		TUTOR												
		SB	MT	SAR	AV	JOS	KS	AIN	JEN	NOE	AM	CAY	HA	KAY
STUDENT		team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2	team 2
JAY	team 1	1	1	1	0	1	1	0	0	0	0	0	0	0
JO	team 1	1	1	0	1	1	1	0	0	1	0	0	1	0
LA	team 1	1	1	1	0	1	0	0	0	1	0	0	1	0
Lei	team 1	1	1	1	0	0	0	0	0	0	0	0	0	0
CA	team 1	1	1	1	1	1	1	0	0	0	0	0	0	0
CU	team 1	1	1	1	1	1	1	0	0	0	0	1	0	0
PA	team 1	1	1	1	1	1	1	0	0	0	0	0	0	0
ME	team 2	1	1	0	0	0	0	0	0	1	1	1	1	1
LW	team 2	1	1	0	1	0	0	1	1	1	1	1	1	1
JM	team 2	0	0	0	0	0	0	1	1	1	0	1	1	1
CAL	team 2	1	1	0	1	0	0	1	0	1	1	1	1	1
EM	team 2	0	0	1	0	0	0	0	0	0	1	0	0	0
JG	team 2	0	0	1	0	0	0	0	0	0	1	0	0	0

• The penalty ρ^{mng} is incurred if a student is assigned to a manager (not regular staff).

The objective of the optimization problem (see A.3(a)–(c) in the appendix) is to maximize the total number of student/tutor assignments that satisfy all the hard constraints, while minimizing the penalties incurred by soft-constraint violations. As indicated by the inequality (A.1) in the appendix, the sum of the possible penalties for making a student–tutor–period assignment should not exceed the reward ρ^{assgn} for making the assignment. This ensures that the optimization will always choose to assign a student to a tutor if possible, even if it means violating several soft constraints.

Reliance on an open-source solver required creativity with our modeling decisions. In the process of building and implementing the optimization model, we discovered that seemingly minor modeling decisions would have a fairly significant impact upon the solution quality and convergence rate. In many initial attempts at solving this problem, the algorithm was not making sufficient progress toward solutions with reasonable optimality gaps. We suspected that the optimization was cycling among different solutions with identical performance—as often occurs in scheduling problems with symmetric resources and demands. However, because of complex interactions between student/tutor availability and qualifications,

Figure 5. (Color online) This Input Table Contains the Feasible Student Pairing Flags

		STUDENT 2												
		JAY	JO	LA	Lei	CA	CU	PA	ME	LW	JM	CAL	EM	JG
STUDENT 1		team 1	team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2
JAY	team 1	0	1	0	0	1	0	0	0	0	0	0	0	0
JO	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
LA	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
Lei	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
CA	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
CU	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
PA	team 1	0	0	0	0	0	0	0	0	0	0	0	0	0
ME	team 2	0	0	0	0	0	0	0	0	0	1	1	1	0
LW	team 2	0	0	0	0	0	0	0	0	0	0	0	0	0
JM	team 2	0	0	0	0	0	0	0	0	0	0	0	0	0
CAL	team 2	0	0	0	0	0	0	0	0	0	0	0	0	0
EM	team 2	0	0	0	0	0	0	0	0	0	0	0	0	1
JG	team 2	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 6. (Color online) This Input Table Shows the Availability of Each Tutor in Each Period

	TUTOR												
	SB	MT	SAR	AV	JOS	KS	AIN	JEN	NOE	AM	CAY	HA	KAY
PERIOD	team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2	team 2
8.5	0	busy	0	busy	0	off-site	0	0	0	0	0	0	0
9	0	busy	0	busy	0	off-site	0	0	0	0	0	0	0
9.5	0	0	0	busy	0	off-site	0	0	0	0	0	0	0
10	busy	0	busy	busy	0	off-site	0	0	0	0	0	0	0
10.5	busy	0	0	0	0	off-site	0	0	0	0	0	0	0
11	0	0	student lunch	0	0	off-site	0	0	0	0	0	0	0
11.5	0	0	0	student lunch	0	off-site	0	0	0	0	0	0	0
12	0	0	0	busy	0	off-site	0	0	0	student lunch	0	0	0
12.5	0	0	0	busy	0	off-site	0	student lunch	0	0	0	0	0
13	0	0	0	0	0	off-site	0	0	0	0	0	0	0
13.5	0	0	0	0	0	off-site	0	0	0	0	0	0	0
14	busy	busy	0	0	0	off-site	0	0	0	0	0	0	0

it was not immediately obvious when resources were symmetric. We also suspected that, although an entire resource may not be symmetric, specific student–teacher–period assignments might be symmetric. To address this, we assigned a small, random additional reward to each student–tutor–period grouping as a way to create a meaningless preference for various assignments and accelerate the optimization convergence. This approach avoided defining an a priori lexicographic ordering, as is done with most symmetry-breaking constraints—for example, Denton et al. (2010) for operating-room scheduling. In the objective function (provided in A.3(a)–(c) in the appendix), γ serves as the random symmetry-breaking parameter, where γ is chosen to be sufficiently small so as not to affect the reproducibility of results. The inequality (A.2) in the appendix provides a general guideline for choosing γ.

We ran trials for various problem instances, both with and without the symmetry-breaking parameter γ included. In some cases, this led to noticeable improvements in both solution quality and convergence time; however, we were unable to determine a pattern for exactly which instances would exhibit this improvement. In practice, the scheduler can toggle this option, depending on his preference or if the optimization is returning an unsatisfactory feasible solution within the preferred solution time (Max Solve Time in Figure 9).

Given the daily nuances of the scheduling decisions, our intent was to support the scheduler's decision-making process by providing an excellent feasible solution that could be implemented with only minor adjustments. Because of the optimization's daily use, computational time is a concern, especially considering the restriction to open-source solvers. In the past, the scheduler created schedules by partitioning the students and tutors into teams of approximately 10 students and 10 tutors and focusing separately on assignments for each team. The subsequent

Figure 7. (Color online) This Input Table Shows the Tutoring Requirements for Each Student in Each Period

	Student												
	JAY	JO	LA	Lei	CA	CU	PA	ME	LW	JM	CAL	EM	JG
PERIOD	team 1	team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2
8.5	0	0	0	Trip	0	Arrive Late	0	Absent	0	Trip	Trip	0	0
9	0	0	0	Trip	0	Arrive Late	0	Absent	0	Trip	Trip	0	0
9.5	0	0	0	Trip	0	Arrive Late	0	Absent	0	Trip	Trip	0	0
10	0	0	0	Trip	0	0	0	Absent	0	Trip	Trip	0	0
10.5	0	0	0	0	0	0	0	Absent	0	0	0	0	0
11	Lunch	Lunch	Lunch	0	0	0	0	Absent	0	0	0	0	0
11.5	0	0	0	Lunch	Lunch	Lunch	Lunch	Absent	0	0	0	0	0
12	0	0	0	0	0	0	0	Absent	Lunch	Lunch	Lunch	0	0
12.5	0	0	0	0	0	0	0	Absent	0	0	0	Lunch	Lunch
13	0	0	0	0	0	0	0	Absent	0	0	0	0	0
13.5	0	0	Out Early	0	0	0	0	Absent	0	0	0	0	0
14	0	0	Out Early	0	0	0	0	Absent	0	0	0	0	0

Figure 8. (Color online) This Table Shows User-Defined Penalties and the Assignment Benefit

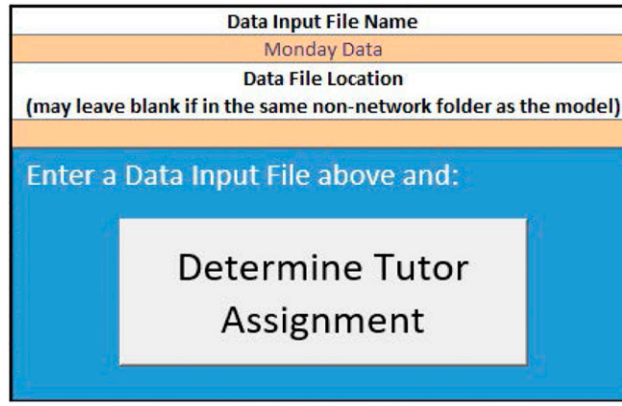
Penalties and Benefit	
1.5 hour Assignments	2
30 minute Assignments	8
Student Pairings	12
Student and Tutor Team Mismatch	4
Manager Assigned Periods	18
Assignment Benefit	200

combination of these team assignments would require significant updates and ignored the ability for tutors to work across teams if needed. Our DSS allows the scheduler to investigate the simultaneous scheduling of all the students. However, for organizational ease and faster computation times, Hopeful Journeys still prefers to have the tutors and students work in teams of roughly 10. Additionally, the scheduler's preferred utilization of the model is to solve in groups of three teams. The data for all 10 teams can be entered together, and schedules can be sequentially determined for each group by changing the "Solving Assignments for Groups of Teams" settings, as shown in Figure 9. The settings, as shown, will solve for a single group of two teams (using team order as defined in Figure 2(b)). This compromise provides a preference for within-team scheduling, but also permits cross-team tutoring support within the group as needed. Based on his experience in creating schedules by hand, the scheduler reports that using the model with this approach provides schedules "about as good as they can be without a human within 20 minutes per group of three. I generally just set the limit [Optimality Gap Limit] to the lowest possible setting and let it run a full 20 minutes each. This seems like a good balance of optimizing

Figure 9. (Color online) The Table Shows Optimization Parameters

Optimization Parameters	
Max Solve Time (minutes)	20
Optimality Gap Limit	0.001
Solving Assignments for Groups of Teams:	
Starting Team	1
Number of Teams in Each Group	2
Number of Groups	1
Other Options:	
Max Tutor Hours per Day with Same Student	2.5
Prioritize Assignments? (Break Symmetry)	1

Figure 10. (Color online) The Figure Shows an Example of a Data File Entry and the Optimization Initiation Interface



and not wasting time." Although commercial solvers such as the Gurobi Optimizer (Gurobi Optimization LLC 2019) can solve three-team groups in a few minutes, the time savings were deemed to not justify the expense for Hopeful Journeys. Similarly, there was the potential to further reduce the computation time by exploiting the problem structure with techniques such as column generation. However, because of Hopeful Journeys' dire need for assistance, we elected to adopt a "rapid prototyping" solution approach, while still providing excellent cost-effective support for the organization's scheduling process.

When the scheduler is satisfied that the input data are correct and the optimization and penalty parameters have been set appropriately, he initiates the solution phase on the model worksheet using the interface shown in Figure 10. Using VBA, the data from the input file (name and location provided) are imported, the optimization model is built by using Python (PuLP), and the model is solved by using CBC. After the optimization converges or the user-entered maximum solution time is met, we provide the user with the soft-constraint violations for the resulting solution (Figure 11).

Figure 11. (Color online) The Output Table Example Reports the Soft-Constraint Violations for a Solution

Penalties and Benefit	Violations in Best Found Solution
1.5 hour Assignments	12
30 minute Assignments	13
Student Pairings	17
Student and Tutor Team Mismatch	34
Manager Assigned Periods	9
Total Periods with Student Need	2
Assignment Benefit	113
Total	21970

Figure 12. (Color online) The Output Table Example Reports the Tutors Who Are Assigned to Each Student

	STUDENT												
	JAY	JO	LA	Lei	CA	CU	PA	ME	LW	JM	CAL	EM	JG
PERIOD	team 1	team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2
8.5	JOS	HA (team 2)	NOE (team 2)	Trip	JOS	Arrive Late	SB	Absent	KAY	Trip	Trip	SAR (team 1)	AM
9	JOS	HA (team 2)	NOE (team 2)	Trip	JOS	Arrive Late	SB	Absent	KAY	Trip	Trip	SAR (team 1)	SAR (team 1)
9.5	MT	SB	NOE (team 2)	Trip	MT	Arrive Late	JOS	Absent	JEN	Trip	Trip	AM	SAR (team 1)
10	MT	NOE (team 2)	HA (team 2)	Trip	MT	CAY (team 2)	JOS	Absent	JEN	Trip	Trip	AM	NEED
10.5	SAR	NOE (team 2)	HA (team 2)	MT	SAR	JOS	AV	Absent	KAY	JEN	CAY	AM	AM
11	Lunch	Lunch	Lunch	MT	SB	JOS	AV	Absent	KAY	JEN	CAY	NEED	AM
11.5	MT	NOE (team 2)	HA (team 2)	Lunch	Lunch	Lunch	Lunch	Absent	KAY	JEN	CAY	SAR (team 1)	SAR (team 1)
12	MT	NOE (team 2)	HA (team 2)	SB	MT	CAY (team 2)	JOS	Absent	Lunch	Lunch	Lunch	SAR (team 1)	SAR (team 1)
12.5	MT	NOE (team 2)	HA (team 2)	SB	MT	CAY (team 2)	JOS	Absent	AIN	KAY	AM	Lunch	Lunch
13	JOS	HA (team 2)	NOE (team 2)	MT	JOS	AV	SB	Absent	JEN	KAY	AM	SAR (team 1)	SAR (team 1)
13.5	JOS	HA (team 2)	Out Early	SAR	JOS	CAY (team 2)	SB	Absent	JEN	NOE	KAY	AM	AM
14	JOS	HA (team 2)	Out Early	SAR	JOS	CAY (team 2)	AV	Absent	JEN	NOE	KAY	AM	AM

Ideally, there would be no violations of any type, but the most critical is ensuring that all students have coverage for all periods required. If coverage gaps exist in the final solution, they appear in the "Total Periods with Student Need," as we show in the example in Figure 11, and the scheduler can quickly allocate any emergency resources needed. The tutor-to-student assignments are formatted by using VBA, as we show in the examples in Figures 12 and 13, to match the layout preferred by the group managers.

Figure 12 displays the tutors assigned to each student during each period of the day. There is an added identifier if the tutor and student are from different teams, as we show for student JO at 8:30 a.m. and 9:00 a.m. (periods 8.5 and 9). Additionally, if a student has an unscheduled period, this is clearly identified, as we show for student EM at 11:00 a.m. (period 11). In the example in Figure 13, we display the schedule for each tutor with identifiers for students from different teams and highlights when a tutor is unscheduled for a period.

Conclusion

As supported by the verification letter at the end of this paper, by eliminating the need for midday adjustments due to scheduling errors, the DSS has directly improved the scheduler's and tutors' daily work environment at Hopeful Journeys Educational Center. Most importantly, the now-stabilized schedule positively impacts the learning environment for 100 students with autism.

Beyond the personal and institutional impacts, this project differs from others in the scheduling literature in several key aspects. First, although each of the complexities enumerated in the introduction's bulleted list has been addressed in other research, it is rare to confront these issues in their entirety within a single problem, especially in applications requiring daily schedules. Second, we were forced to take a nonstandard approach to deal with symmetric resources, because the classical approach of enumerating shifts and reformulating the problem would be ineffective. We are not simply assigning tutors to time

Figure 13. The Output Table Example Reports the Students Who Are Assigned to Each Tutor

	TUTOR												
	SB	MT	SAR	AV	JOS	KS	AIN	JEN	NOE	AM	CAY	HA	KAY
PERIOD	team 1	team 1	team 1	team 1	team 1	team 1	team 2	team 2	team 2	team 2	team 2	team 2	team 2
8.5	PA	busy	EM (team 2)	busy	JAY and CA	off-site			LA (team 1)	JG		JO (team 1)	LW
9	PA	busy	EM (team 2) and JG (team 2)	busy	JAY and CA	off-site			LA (team 1)			JO (team 1)	LW
9.5	JO	JAY and CA	JG (team 2)	busy	PA	off-site		LW	LA (team 1)	EM			
10	busy	JAY and CA	busy	busy	PA	off-site		LW	JO (team 1)	EM	CU (team 1)	LA (team 1)	
10.5	busy	Lei	JAY and CA	PA	CU	off-site		JM	JO (team 1)	EM and JG	CAL	LA (team 1)	LW
11	CA	Lei	student lunch	PA	CU	off-site		JM	LUNCH	JG	CAL	LUNCH	LW
11.5	LUNCH	JAY	EM (team 2) and JG (team 2)	student lunch	LUNCH	off-site		JM	JO (team 1)	LUNCH	CAL	LA (team 1)	LW
12	Lei	JAY and CA	EM (team 2) and JG (team 2)	busy	PA	off-site	LUNCH	LUNCH	JO (team 1)	student lunch	CU (team 1)	LA (team 1)	LUNCH
12.5	Lei	JAY and CA	LUNCH	busy	PA	off-site	LW	student lunch	JO (team 1)	CAL	CU (team 1)	LA (team 1)	JM
13	PA	Lei	EM (team 2) and JG (team 2)	CU	JAY and CA	off-site		LW	LA (team 1)	CAL	LUNCH	JO (team 1)	JM
13.5	PA	LUNCH	Lei	LUNCH	JAY and CA	off-site		LW	JM	EM and JG	CU (team 1)	JO (team 1)	CAL
14	busy	busy	Lei	PA	JAY and CA	off-site		LW	JM	EM and JG	CU (team 1)	JO (team 1)	CAL

slots, but to students as well. Third, the sheer number and complexity of the scheduling rules and restrictions forced us to be intentional about defining hard versus soft constraints. Fourth, despite the rich literature that exists on timetabling in education, little has been published in the area of educational personnel scheduling. Finally, Hopeful Journeys' limited budget, immediate need for a rapidly developed solution, and desire to independently use the finished product on a daily basis compelled us to find a solution that struck the right balance between solution quality, run time, stability, ease of use, and cost-effectiveness.

More generally, although some of the specific issues addressed in this model may be unique to this situation, the approach to addressing these issues with hard constraints and soft constraints for preferences can provide guidance for the development of similar scheduling tools for service organizations. The problem of scheduling specialized resources with limited capacity is not unique to one-on-one student tutoring scheduling. For example, as hospitals evolve from a traditional design with each floor focusing on a fixed specialty (e.g., orthopedics) to flexible multispecialty floors with personalized patient care, the need for equally flexible and efficient physician specialty-to-patient scheduling will increase. Our model presented here could provide an excellent initial framework.

Acknowledgments

The authors thank the scheduler, Joseph Protz, and the Hopeful Journeys Educational Center for their openness and time during this project. The authors also thank the members of the 2019 Daniel H. Wagner Prize for Excellence in Operations Research Practice committee for their helpful feedback. The authors are sincerely honored and appreciative to have been selected as finalists.

Appendix. Mathematical Formulation

Indices and Sets

$s \in \mathcal{S}$ The set of students to be assigned tutors.

$t \in \mathcal{T}$ The set of tutors to be assigned students.

$p \in \mathcal{P}$ The set of half-hour periods during the day when students need tutors. For this model, we assume that $\mathcal{P} = \{1,2,\ldots,P\}$, where P is the number of periods.

$p \in \mathcal{L} \subset \mathcal{P}$ The set of potential lunch periods.

Data

$a_{s1,s2}$ An indicator parameter for $s1, s2 \in S$. If $a_{s1,s2} = 1$, then students $s1$ and $s2$ may be assigned to the same tutor in the same period.

q_{st} An indicator parameter for $s \in \mathcal{S}, t \in \mathcal{T}$. If $q_{st} = 1$, then tutor t is qualified to work with student s.

f_{st} An indicator parameter for $s \in \mathcal{S}, t \in \mathcal{T}$. If $f_{st} = 1$, then student s and tutor t are on the same team.

b_{tp} An indicator parameter for $t \in \mathcal{T}, p \in \mathcal{P}$. If $b_{tp} = 1$, then tutor t is busy (unavailable) in period p.

d_{sp} An indicator parameter for $s \in \mathcal{S}, p \in \mathcal{P}$. If $d_{sp} = 1$, then student s does not require a tutor in period p.

m_t An indicator parameter for $t \in \mathcal{T}$. If $m_t = 1$, then tutor t is a manager.

ρ^{assgn} A user-defined reward parameter for successfully assigning a student to a tutor in a period.

ρ^{three} A user-defined penalty parameter for assigning a student to a tutor for three periods in succession.

ρ^{iso} A user-defined penalty parameter for assigning a student to a tutor for an isolated 30-minute period.

ρ^{pair} A user-defined penalty parameter for assigning two different students to the same tutor in the same period.

ρ^{team} A user-defined penalty parameter for assigning a student to a tutor from a different team.

ρ^{mng} A user-defined penalty parameter for assigning a student to a tutor who is a manager.

γ_{stp} A random symmetry-breaking parameter for $s \in \mathcal{S}, t \in \mathcal{T}, p \in \mathcal{P}$.

As a general guideline, the objective function reward and penalties should be chosen so that

$$\rho^{\text{assgn}} > \max(\rho^{\text{three}}, \rho^{\text{iso}}) + \rho^{\text{pair}} + \rho^{\text{team}} + \rho^{\text{mng}}. \qquad \text{(A.1)}$$

This ensures that the optimization will always choose to assign a student to a tutor if possible, even if it means violating several soft constraints. Because an assignment cannot be part of both a three-period and an isolated-period assignment, we only need to account for the maximum of ρ^{three} and ρ^{iso}.

The symmetry-breaking parameter γ should create arbitrary preferences between different schedules that otherwise have the same objective function value, while not changing the relative ranking of two schedules that have different objective function values. That is, each γ_{stp} should be randomly chosen so that

$$0 \le \gamma_{stp} \ll \Delta \text{ for } s \in \mathcal{S}, t \in \mathcal{T}, p \in \mathcal{P}, \qquad \text{(A.2)}$$

where Δ is the smallest positive amount by which the objective function of two different schedules can differ. If all the reward and penalty parameters are chosen to be integers, then Δ is the greatest common factor of those parameters.

Binary Decision Variables

x_{stp} 1 if student s is assigned to tutor t in period p; 0 otherwise.

ℓ_{tp} 1 if tutor t is assigned lunch in period p (defined for $p \in \mathcal{L}$); 0 otherwise.

z^{three}_{stp} 1 if student s is assigned to tutor t in periods p, $p+1$, and $p+2$; 0 otherwise.

z^{iso}_{stp} 1 if student s is assigned to tutor t in period p, but in neither $p-1$ nor $p+1$; 0 otherwise.

z^{pair}_{tp} 1 if tutor t is assigned to two different students in period p; 0 otherwise.

Constraints

- Each student can be assigned at most one tutor in each period that the student requires a tutor ($d_{sp} = 0$). A student cannot be assigned a tutor in any period that the student does not require a tutor ($d_{sp} = 1$).

$$\sum_{t \in \mathcal{T}} x_{stp} \le 1 - d_{sp} \quad \text{for } s \in \mathcal{S}, p \in \mathcal{P}. \qquad \text{(A.3(d))}$$

- An assignment cannot be made in a period if the tutor is unavailable ($b_{tp} = 1$).

$$b_{tp}\sum_{s\in\mathcal{S}} x_{stp} = 0 \quad \text{for } t\in\mathcal{T}, p\in\mathcal{P}. \tag{A.3(e)}$$

- A tutor cannot be assigned to a student with whom that tutor is not qualified to work ($q_{st} = 0$).

$$(1 - q_{st})\sum_{p\in\mathcal{P}} x_{stp} = 0 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(f)}$$

- Students may only be paired together in the same period p if the pairing is approved ($a_{s1,s2} = 1$).

$$x_{s1,t,p} + x_{s2,t,p} \le 1 + a_{s1,s2} \quad \text{for} \quad s1 \ne s2 \in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P}. \tag{A.3(g)}$$

- A student cannot be assigned to the same tutor for more than two-and-a-half total hours (five 30-minute periods) during the day.

$$\sum_{p\in\mathcal{P}} x_{stp} \le 5 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(h)}$$

- A student cannot be assigned to the same tutor for more than one hour and a half (three 30-minute periods) in succession.

$$\sum_{n=0}^{3} x_{s,t,p+n} \le 3 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P} \text{ where } p \le P-3. \tag{A.3(i)}$$

- A tutor cannot be assigned lunch if that tutor is also assigned to a student during that period.

$$\ell_{tp} \le 1 - x_{stp} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{L}. \tag{A.3(j)}$$

- A tutor cannot be assigned lunch during a period if that tutor is unavailable due to other obligations ($b_{tp} = 1$).

$$b_{tp}\ell_{tp} = 0 \quad \text{for } t\in\mathcal{T}, p\in\mathcal{L}. \tag{A.3(k)}$$

- A tutor must be assigned exactly one lunch period, unless that tutor is busy with other obligations during all lunch periods ($\sum_{p\in\mathcal{L}} b_{tp} = |\mathcal{L}|$).

$$\left(|\mathcal{L}| - \sum_{p\in\mathcal{L}} b_{tp}\right)\sum_{p\in\mathcal{L}} \ell_{tp} = |\mathcal{L}| - \sum_{p\in\mathcal{L}} b_{tp} \quad \text{for } t\in\mathcal{T}. \tag{A.3(l)}$$

- One-hour student-to-tutor assignments are preferred. A feasible solution assigning one hour and a half (three successive 30-minute periods) with the same student–tutor pair would require $z_{stp}^{\text{three}} = 1$ in the following constraint for a solution to be feasible and incur a penalty of ρ^{three} in the objective function:

$$\sum_{n=0}^{2} x_{s,t,p+n} \le 2 + z_{stp}^{\text{three}} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P} \text{ where } p \le P-2. \tag{A.3(m)}$$

- For a feasible solution to have an isolated 30-minute assignment, z_{stp}^{iso} must be equal to one in the associated constraint (A.3(n)) and incur a penalty ρ^{iso} in the objective function. Constraints (A.3(o)) and (A.3(p)) address the cases of the first and last periods of the day.

$$x_{stp} \le x_{s,t,p-1} + x_{s,t,p+1} + z_{stp}^{\text{iso}} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P}\backslash\{1, P\}. \tag{A.3(n)}$$

$$x_{s,t,1} \le x_{s,t,2} + z_{s,t,1}^{\text{iso}} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(o)}$$

$$x_{s,t,P} \le x_{s,t,P-1} + z_{s,t,P}^{\text{iso}} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(p)}$$

- One-on-one assignments are preferred, but (at most) a pair of students can be assigned to the same tutor if necessary. Assigning a pair of students to the same tutor in the same period in a feasible solution requires $z_{tp}^{\text{pair}} = 1$ in the following constraint, incurring a penalty of ρ^{pair} in the objective function.

$$\sum_{s\in\mathcal{S}} x_{stp} \le 1 + z_{tp}^{\text{pair}} \quad \text{for } t\in\mathcal{T}, p\in\mathcal{P}. \tag{A.3(q)}$$

Objective Function. We seek to maximize the reward for assigning a student to a tutor (ρ^{assgn}) reduced by the penalties for each of the soft-constraint violations in the feasible solution. The violations for assigning a tutor to a student from another team ($f_{st} = 0$) and for utilizing a manager ($m_t = 1$) are determined directly within the objective function.

$$\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\left(\rho^{\text{assgn}} + \gamma_{stp}\right)x_{stp}. \tag{A.3(a)}$$

$$-\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\left[\rho^{\text{iso}} z_{stp}^{\text{iso}} + \rho^{\text{team}} x_{stp}(1 - f_{st}) + \rho^{\text{mng}} x_{stp} m_t\right]. \tag{A.3(b)}$$

$$-\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{\substack{p\in\mathcal{P}\\ p\le P-2}} \rho^{\text{three}} z_{stp}^{\text{three}} - \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}} \rho^{\text{pair}} z_{tp}^{\text{pair}}. \tag{A.3(c)}$$

Complete Formulation.

$$\text{Maximize} \quad \sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\left(\rho^{\text{assgn}} + \gamma_{stp}\right)x_{stp}. \tag{A.3(a)}$$

$$-\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\left[\rho^{\text{iso}} z_{stp}^{\text{iso}} + \rho^{\text{team}} x_{stp}(1 - f_{st}) + \rho^{\text{mng}} x_{stp} m_t\right]. \tag{A.3(b)}$$

$$-\sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}}\sum_{\substack{p\in\mathcal{P}\\ p\le P-2}} \rho^{\text{three}} z_{stp}^{\text{three}} - \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}} \rho^{\text{pair}} z_{tp}^{\text{pair}}. \tag{A.3(c)}$$

subject to

$$\sum_{t\in\mathcal{T}} x_{stp} \le 1 - d_{sp} \quad \text{for } s\in\mathcal{S}, p\in\mathcal{P}. \tag{A.3(d)}$$

$$b_{tp}\sum_{s\in\mathcal{S}} x_{stp} = 0 \quad \text{for } t\in\mathcal{T}, p\in\mathcal{P}. \tag{A.3(e)}$$

$$(1 - q_{st})\sum_{p\in\mathcal{P}} x_{stp} = 0 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(f)}$$

$$x_{s1,t,p} + x_{s2,t,p} \le 1 + a_{s1,s2} \quad \text{for } s1 \ne s2 \in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P}. \tag{A.3(g)}$$

$$\sum_{p\in\mathcal{P}} x_{stp} \le 5 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}. \tag{A.3(h)}$$

$$\sum_{n=0}^{3} x_{s,t,p+n} \le 3 \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{P} \text{ where } p \le P-3. \tag{A.3(i)}$$

$$\ell_{tp} \le 1 - x_{stp} \quad \text{for } s\in\mathcal{S}, t\in\mathcal{T}, p\in\mathcal{L}. \tag{A.3(j)}$$

$$b_{tp}\ell_{tp} = 0 \quad \text{for } t\in\mathcal{T}, p\in\mathcal{L}. \tag{A.3(k)}$$

$$\left(|\mathscr{L}| - \sum_{p\in\mathscr{L}} b_{tp}\right)\sum_{p\in\mathscr{L}} \ell_{tp} = |\mathscr{L}| - \sum_{p\in\mathscr{L}} b_{tp} \qquad \text{for } t\in\mathscr{T}. \qquad \text{(A.3(l))}$$

$$\sum_{n=0}^{2} x_{s,t,p+n} \le 2 + z^{\text{three}}_{stp} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}, p\in\mathscr{P} \text{ where } p \le P-2. \qquad \text{(A.3(m))}$$

$$x_{stp} \le x_{s,t,p-1} + x_{s,t,p+1} + z^{\text{iso}}_{stp} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}, p\in\mathscr{P}\backslash\{1,P\}. \qquad \text{(A.3(n))}$$

$$x_{s,t,1} \le x_{s,t,2} + z^{\text{iso}}_{s,t,1} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}. \qquad \text{(A.3(o))}$$

$$x_{s,t,P} \le x_{s,t,P-1} + z^{\text{iso}}_{s,t,P} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}. \qquad \text{(A.3(p))}$$

$$\sum_{s\in\mathscr{S}} x_{stp} \le 1 + z^{\text{pair}}_{tp} \qquad \text{for } t\in\mathscr{T}, p\in\mathscr{P}. \qquad \text{(A.3(q))}$$

$$x_{stp}, z^{\text{iso}}_{stp} \in \{0,1\} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}, p\in\mathscr{P}. \qquad \text{(A.3(r))}$$

$$z^{\text{three}}_{stp} \in \{0,1\} \qquad \text{for } s\in\mathscr{S}, t\in\mathscr{T}, p\in\mathscr{P} \text{ where } p \le P-2. \qquad \text{(A.3(s))}$$

$$\ell_{tp}, z^{\text{pair}}_{tp} \in \{0,1\} \qquad \text{for } t\in\mathscr{T}, p\in\mathscr{P}. \qquad \text{(A.3(t))}$$

References

Bailey MD, Michaels D (2019) An optimization-based DSS for student-to-teacher assignment: Classroom heterogeneity and teacher performance measures. *Decision Support Systems* 119(April): 60–71.

Bailey MD, Nowak M (2018) MeetOpt: A multi-event coaching decision support system. *Decision Support Systems* 112(August): 60–75.

Denton BT, Miller AJ, Balasubramanian HJ, Huschka TR (2010) Optimal allocation of surgery blocks to operating rooms under uncertainty. *Oper. Res.* 58(4-part-1):802–816.

Forrest J, Ralphs T, Vigerske S, Hafer L, Kristjansson B, Fasano J, Straver E, et al. (2018) coin-or/Cbc: Version 2.9.9. Accessed October 15, 2018, http://doi.org/10.5281/zenodo.1317566.

Foulds L, Johnson D (2000) Slotmanager: A microcomputer-based decision support system for university timetabling. *Decision Support Systems* 27(4):367–381.

Gurobi Optimization LLC (2019) Gurobi optimizer reference manual. Accessed January 30, 2019, https://www.gurobi.com/documentation/9.0/refman/index.html.

Heider S, Heins J, Kanet JJ (2018) Applying operations research to scheduling work cells in a manufacturing environment. *Interfaces* 48(6):556–565.

Hopeful Journeys (2019) Overview. Accessed May 31, 2019, http://www.hopefuljourneys.org/overview-our-school.html.

Johnes J (2015) Operational research in education. *Eur. J. Oper. Res.* 243(3):683–696.

Lougee-Heimer R (2003) The Common Optimization INterface for Operations Research. *IBM J. Res. Development* 47(1):57–66.

Mason A (2013) SolverStudio: A new tool for better optimisation and simulation modelling in Excel. *INFORMS Trans. Ed.* 14(1):45–52.

Miranda J, Rey PA, Robles JM (2012) udpSkeduler: A web architecture based decision support system for course and classroom scheduling. *Decision Support Systems* 52(2):505–513.

Mitchell S, Kean A, Mason A, O'Sullivan M, Phillips A, Peschiera F (2009) Optimization with PuLP. Accessed August 15, 2018, https://coin-or.github.io/pulp/index.html.

Schoenfelder J, Pfefferlen C (2018) Decision support for the physician scheduling process at a German hospital. *Service Sci.* 10(3): 215–229.

Şeref MM, Ahuja RK (2008) Spreadsheet-based decision support systems. Burstein F, Holsapple CW, eds. *Handbook on Decision Support Systems 1: Basic Themes*, International Handbooks on Information Systems (Springer, Berlin), 277–298.

Van den Bergh J, Beliën J, De Bruecker P, Demeulemeester E, De Boeck L (2013) Personnel scheduling: A literature review. *Eur. J. Oper. Res.* 226(3):367–385.

Verification Letter

Joseph Protz, School-Wide Scheduler, Hopeful Journeys Educational Center, 28 Tozer Road, Beverly, Massachusetts 01915, writes:

"This daily schedule must ensure that each student is assigned a tutor for each period of the day while accounting for a complex variety of scheduling preferences. The quality of the schedule is vital to assuring each staff and student have a well-rounded and productive day. Due to the changing availability of teachers and needs of students, this was a complex task on which I spent 10–14 hours per day. Even with this significant time commitment, there were often errors, and any changes after the schedule was created would create a significant challenge to adjust the schedule.

"Since January of 2019, I have been using the model for our daily scheduling process, and the difference has been unbelievable. It now takes on average four hours to complete a schedule with much higher quality and accuracy than previously. The errors have been reduced and continue to decrease as I become familiar with the model and refine my data. This has allowed me to spend the remaining time working on requested schedule changes, error checking, and optimization to match our organization's goals. Because of this, it has also reduced the strain on other staff who assist in correcting errors.

"This tool has saved me several hours per day and, more importantly, provided better and more consistent schedules for our students and teachers. The students and the organization have benefited because I can now schedule staff training, manage scheduling requests, and manage the variable needs of the school in a more consistent manner. The impact of this project has been far beyond what I expected when I first reached out for help."

Matthew D. Bailey is department chair and a professor of analytics and operations management within the Freeman College of Management at Bucknell University. He received his PhD in industrial and operations engineering from the University of Michigan. His primary research interests are the theory and application of models for sequential decision making under uncertainty. In particular, he is interested in a broad range of areas within healthcare, scheduling, and adversarial network optimization. This work has resulted in publications in journals such as *Operations Research*, *IIE Transactions*, *EJOR*, *Naval Research Logistics*, *Decision Analysis*, *Decision Support Systems*, and *Networks*.

Lucas A. Waddell is an assistant professor of mathematics at Bucknell University. He received his PhD in mathematical sciences from Clemson University. He also spent several years working as an operations research analyst at Sandia National Laboratories, where he worked on large-scale decision support problems for primarily national security applications. His research interests include nonlinear combinatorial optimization and metaheuristic optimization algorithms.