Interview Scheduling: An Integer Programming Approach

BY Christian Braathen

DISCUSSION PAPER

NHH

Institutt for foretaksøkonomi
Department of Business and Management Science

FOR 10/2022

ISSN: 2387-3000

April 2022

Interview Scheduling: An Integer Programming Approach

Christian Braathen*

This article provides a general meeting scheduling model to schedule en-masse interviews over multi-day events. Creating such meeting schedules is a complicated, time-consuming task that usually requires considerable manual planning when hundreds of meetings are involved and if the quality of the schedule is a priority. The problem consists of three overlapping and gradually more difficult objectives—that is, at favorable times, to assign meetings, assign individual interviewers consecutive meetings, and assign interviewer pairs consecutive meetings. The model has been applied five times within a student organization, ranging from 17 to 1149 meetings to be scheduled. Compared to former, manually created schedules, both the organization's and the interviewers' perceived interviewing experience is markedly improved. ***Keywords***—meeting scheduling, assignment problems, optimization, integer programming, volunteer management, practice of OR

---

1 Introduction

Volunteerism is a substantial cogwheel in modern-day societies, with an estimated one billion volunteers worldwide (United Nations Volunteers programme, 2018). The United Nations describes it as the thread that binds communities together (United Nations Volunteers programme, 2018), and volunteers achieve direct career benefits from volunteering. Specifically, volunteers have a 27% higher chance to be employed than people who do not volunteer (Corporation for National and Community Service, 2013), which may in part be explained by the fact that 92% of human resource executives consider volunteering to improve leadership skills (Deloitte, 2016). However, since the volunteers do not receive a salary for their efforts (Shin & Kleiner, 2003), it is imperative to delegate tasks and responsibilities they prefer to remain motivated (Gordon & Erkut, 2004; Sampson, 2006).

One organization focusing extensively on providing tasks and responsibilities the volunteers prefer is among Norway's most active student associations. At the Norwegian

*Norwegian School of Economics, Helleveien 30, 5045 Bergen, Norway.
Email: christian.braathen@nhh.no.

School of Economics (NHH), students operate an association consisting of 120 subgroups as well as three larger projects every two years. These projects hire approximately 3000 volunteer students through three hiring rounds each over two years—arranging the second largest cultural festival in Western Norway, Northern Europe's largest student-driven business conference, and a five-day student sporting event for about 2000 participants. Since the projects are time-limited and volunteered by students wanting to volunteer in different positions with different responsibilities throughout their studies, the volunteer turnover rate is practically 100 percent. Hence, the student association must recruit about 3000 volunteer students anew every two years during a few multi-day events recruiting and interviewing the students.

To be inclusive, the student association requires that the project organizations interview every student applying to become a volunteer and that managers for the applicants' preferred positions must conduct the interviews. This concern complicates the scheduling process as only a selected few people can conduct each interview. However, the need to work with preferred tasks and responsibilities extends beyond applicants. Specifically, the interviewers are also volunteers, and their past unsatisfactory interviewing schedules made them less willing to take on such volunteer positions. Hence, the student association has expressed a wish for a decision support tool that can provide better schedules for the interviewers while respecting the preferences of the applicants.

This article aims to simplify the scheduling process by providing an integer programming model that helps schedule en-masse meetings with any number of participants. The model has been applied to NHH's student association multiple times in the last three years. Although the focus is on scheduling many interviews over a multi-day event, the model covers most considerations needed in more general meeting scheduling models. Hence, the model presented in this article can be considered an application of the interview scheduling problem, but the model can also solve the more general meeting scheduling problem. Additionally, as the model provides most considerations already covered by the meeting scheduling literature—in addition to considerations not previously covered but that are also important in this problem—the main contribution of this article is a more general meeting scheduling model that can be used as a basis for other meeting scheduling applications.

The article is organized as follows. Section 2 explores the interview allocation problem together with the meeting scheduling problem. Section 3 presents a case study while the corresponding mathematical model and results are presented in Section 4 and 5, respectively. The article concludes in Section 6 and highlights how the meeting scheduling problem has become more relevant in the post-COVID-19 era and that this increased relevance can be utilized to improve meeting scheduling within organizations.

2 Literature

The interview scheduling problem is a subset of the closely related meeting scheduling problem between two or more parties, except that a secretariat may participate in interviews if necessary. Due to the similarity between the problems, the literature on the meeting scheduling problem should be studied, too. Moreover, both problems can be considered part of a larger class of problems—namely, the assignment problem. Such problems involve assigning parties and potentially assigning parties to a specific time. The parties in assignment problems can be animate objects (agents), inanimate objects (tasks), or both.

In comparison, the meeting scheduling problem involves that n individuals should participate in (up to) a pre-defined, varying number of meetings m_i allocated at certain times during t time slots, being restricted by having up to r simultaneous meetings due to location constraints—such as the number of rooms. While there is usually a substantial number of meetings to be assigned—up to $\sum_{i=1}^{n} m_i$—there are usually far fewer meetings than there are possible meeting starts. This flexibility, coupled with the fact that there are usually only a few parties involved in each meeting, creates a focus on the individuals, allocating their respective meetings at favorable times without wasting their time on unnecessary waiting.

A broad overview of the meeting scheduling literature and key model features are presented in Table 1, and selected key features are described next.

Table 1: Meeting scheduling literature comparison. Model features highlighted in bold are covered only by the model presented in this article.

Model feature	P	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]
More than two parties can meet	C	N	N	C	N	N	N	N	N	N	C	N	N
>1 representatives inside each party	C	N	C	C	N	C	N	N	C	N	N	N	C
Various time slots per meeting	C	N	N	N	N	N	N	N	N	C	N	N	C
Multiple days constraints	C	C	N	N	N	N	N	N	N	C	N	N	N
Infrastructural constraints	C	N	N	C	I	C	C	N	C	I	I	N	N
Consecutive meetings	C	I	I	C	I	I	I	I	I	N	N	N	N
Meeting preferences	C	N	N	N	N	C	C	C	C	N	N	C	C
Multiple venues	C	N	N	N	N	N	N	N	N	N	N	N	N
Schedule all meetings	C	N	C	C	C	C	N	N	N	C	C	C	N
Preserving parties	C	N	N	N	N	N	N	N	N	N	N	N	N
Time preference	C	N	N	N	N	N	C	N	C	N	N	C	N
Unavailability considerations	C	N	N	C	C	C	N	N	C	N	N	C	C
Enforced breaks	C	N	N	N	N	N	C	N	N	N	N	C	C
Seniority	C	N	N	N	N	N	N	N	N	N	N	N	N
Reduce room changes	I	N	N	N	N	C	C	N	N	N	C	N	N
Minimize resource use	I	N	N	N	N	C	C	N	N	N	N	N	N
Moving constraint	N	N	N	N	N	C	C	N	C	N	N	N	N
Fair allocation	N	N	N	N	N	N	N	N	N	N	N	N	N

C: model feature considered; I: model feature implicitly considered; N: model feature not considered.

P: the model presented in this article; [1]: Bartholdi III & McCroan (1990); [2]: Rinaldi & Serafini (2006); [3]: Pesant et al. (2015); [4]: Le Roux et al. (2015); [5]: Gebser et al. (2013); [6]: Gueret et al. (2009); [7]: Huang et al. (2012); [8]: Ernst et al. (2003); [9]: Mausser et al. (1996); [10]: Kiyonari et al. (2006); [11]: Aizam & Sim (2016); [12]: Schrage (2004).

Since meeting scheduling primarily focuses on the individuals, improving their well-being in such a model is essential. Four factors are essential to provide a foundation for the participants' well-being—namely, to consider the preferences of which parties should meet, to respect both time preferences and unavailability considerations, and to prevent the meeting participants from having meetings at different venues over a short time-frame.

Thus far, the literature has inconsistently covered these four foundational well-being considerations. Specifically, considering the preferences of which parties should meet is covered by Gebser et al. (2013), Gueret et al. (2009), Huang et al. (2012), and Ernst et al. (2003), while only Gueret et al. (2009) and Ernst et al. (2003) provides time preferences to the model. Moreover, unavailability considerations are considered by Pesant et al. (2015), Le Roux et al. (2015), Gebser et al. (2013), and Ernst et al. (2003), while the existing literature does not cover multiple venues yet.

Moreover, additional considerations should preferably be made to further improve the participants' well-being. One such consideration is to maximize the consecutive interviews of the interviewers, which has been the most frequent objective in the meeting scheduling literature.

Despite the importance of this consideration, it has usually been modeled with a proxy objective. For instance, Bartholdi III & McCroan (1990) describe the need for consecutive interviews but attempt to model it implicitly by allocating meetings sooner. Rinaldi & Serafini (2006), Gebser et al. (2013), and Le Roux et al. (2015), in comparison, achieved consecutive interviews implicitly by reducing the number of idle time slots.

However, it may be better to reduce the number of breaks instead of time slots. For instance, the meeting participants may consider a long break of three time slots as more favorable than three breaks of one time slot each—especially if the time slot is short. The project organizations in this article favor the approach of reducing the number of breaks, which was also the objective of Pesant et al. (2015).

Reducing the number of breaks brings another key benefit, too. Specifically, when multiple parties attend multiple meetings together, which was also considered by Pesant et al. (2015), then the participants' well-being can be improved further if they can undertake multiple meetings consecutively together. Preserving meeting parties has not been covered by the existing literature yet, but this is primarily because most of the literature schedules meetings of two parties only—rendering the preserving-participants consideration irrelevant.

Finally, while a particular focus on the individuals is important in a meeting scheduling model, focusing on meeting disparity is another important concern. Specifically, such disparity is usually achieved both by allowing a varying number of parties to attend each meeting and by planning with a varying meeting duration. For the former matter, most of the literature studies cases in which only two parties will attend each meeting (Bartholdi III & McCroan, 1990; Rinaldi & Serafini, 2006; Le Roux et al., 2015; Gebser et al., 2013; Gueret et al., 2009; Huang et al., 2012; Ernst et al., 2003), while the latter has largely been ignored by the literature with the exception of Mausser et al. (1996) and Schrage (2004).

Overall, and as shown in Table 1, the literature is collectively exhaustive in covering most of the essential needs in the meeting scheduling problem. However, most of the literature avoids modeling with meeting disparity, which reduces the applicability of the problem when extending it to other cases besides the test instances provided by the literature. Additionally, each article in the literature covers only a smaller subset of the problem's features. Hence, there is a gap in the literature for a more general meeting scheduling model. By collectively allowing for meeting disparity and covering a broader set of the problem's features, we can fill this gap with the model presented in Section 4.

By filling this gap, practitioners may apply the problem to a broader set of use cases than the current literature facilitates, which is the primary aim of this article. Additionally, this article's secondary aim is to contribute to the practice of scheduling in general.

3 Background

At NHH, the student association organizes three projects that hire approximately 3000 volunteer students through three hiring rounds each over the projects' two-year duration. The student association requires that every student applying to become a volunteer in these projects is interviewed, and the interviewers must represent the two sub-groups the student applied to volunteer within. If a student applies to sub-groups led by the same set of managers, then the venue's secretariat will take on the role of the second interviewer.

Interviews should preferably be scheduled consecutively and early in the week. Specifically, the association prefers that interviewers undertake as many of their interviews as possible early on in the interviewing schedule so that the student association has more flexibility in re-allocating interviews if necessary. Within a given day, it is generally favorable to undertake interviews early. The only exception to this intraday preference is that the earliest time slots of 8–9 AM are less favored than 9–10 AM but equally favored as 10 AM–4 PM—see Table 2 for more details. Moreover, the interviewers should preferably conduct interviews consecutively because the interviewers can then utilize their time more efficiently, having more time to study without frequent interruptions to conduct interviews.

Table 2: Time preference ranks, exemplified with a three-day meeting window.

		Day		
From	To	1	2	3
08:00 AM	09:00 AM	2	4	5
09:00 AM	10:00 AM	1	2	3
10:00 AM	04:00 PM	2	4	5
04:00 PM	06:00 PM	3	5	7
06:00 PM	08:00 PM	4	6	8

These interruptions are a common concern among the students. One organizer stated, "In my experience, it is *especially* the recruiting part of student association work most people emphasize when explaining why they do not have the time to engage in [the student association]". These considerations, coupled with the fact that the students are allocated to responsibilities they prefer and the leaders of these responsibilities must conduct these interviews, cause substantial overhead efforts that must be made to plan these interviews.

However, early and consecutive meetings are merely a prerequisite for the student association. As described in Section 2, certain foundational factors are essential to include to ensure a satisfying schedule. For the student association, having two parties that will conduct multiple interviews with a different set of applicants of various duration is a factor that makes the scheduling process particularly complicated. Additionally, the planners should respect interviewers' unavailability[1], ensure that the more senior group member undertakes the interview if (s)he is available because each group may bring both a group leader and deputy leader(s) to undertake interviews, and that no interviewers should need to travel to multiple venues within the same day to undertake interviews. Moreover, all applicants must be interviewed to avoid making any students feel excluded from the otherwise inclusive student association. Since the student association must reserve rooms to hold interviews, there may be a varying amount of rooms available at any one time, which creates an upper limit on the number of simultaneous interviews that can be conducted.

Besides allocating more early and consecutive meetings by applying the mathematical model presented in this article, the student association also wishes for other qualitative improvements. Specifically, they would like to ensure that interviewers get some resting breaks between consecutive interviews so that they do not, for instance, need to interview for eight hours straight on the first day of the scheduling period. If an interview cannot be scheduled consecutively, the association prefers that the interview is still scheduled for a favorable time. However, for interviews that *can* be scheduled consecutively, the student association appreciates if the interviewer pairs are preserved across consecutive interviews because they already share a meeting room. However, preserving pairs have a lower priority to the student association than achieving consecutive interviews.

The project organizations had the following workflow. First, they provided the input data in a spreadsheet, specifying information about the interviewers, applicants, what constitutes meetings, room availability, and time periods to conduct interviews. A meeting in this context is a pre-defined decision about which two interviewers will meet each applicant and for how long. Additionally, the organizers specified when the interviewers were unavailable in the spreadsheet. The information was then converted into sets and parameters and subsequently added to the mathematical programming model described in the next section. Finally, the produced schedules were converted to spreadsheets—both individual per interviewer and collective—for distribution and administration.

Finally, providing better schedules have beneficial implications for the association and

[1]The planners did not consider the applicants' unavailability *ex-ante* because they experienced that only a few applicants needed rescheduling.

its students. Specifically, a mathematical programming model that can help students with this overhead task will give other students better working conditions when undertaking interviews and may also contribute to more students wanting to volunteer and take on managerial roles within the association. Since many of these students will take on future leading positions in the business landscape, and with executives believing leadership skills improve with volunteering (Deloitte, 2016), the benefits of such a decision tool extend beyond the direct effects created within the volunteer organizations.

4 The Mathematical Model

The constraints and objective function presented in this section consist of the following sets, subsets, parameters, and variables:

Table 3: Model components

Sets	
M	Set of meetings.
T	Set of time slots.
P	Set of participants.
D	Set of days.
L	Set of venues.
G	Set of groups.
$\mathbb{I}^{\mathscr{C}}$	Set of $\mathscr{C}$'s indices—$\mathbb{I}^{\mathscr{C}} := \{i+1 \in \mathbb{N}_{\lvert\mathscr{C}\rvert}\}$.
Subsets	
$\mathscr{B}_m \subseteq T$	An indexed set of time slots an interview $m \in M$ may begin.
$\mathscr{I} \subseteq P$	The set of interviewers.
$\mathscr{A} \subseteq P$	The set of applicants.
$\mathscr{S} \subseteq P$	The set of the venue secretariat.
$\mathscr{P}_p \subseteq M$	An indexed set of meetings that participant $p \in P$ may attend.
$\mathscr{T}_d \subseteq T$	An indexed set representing which time slots occur in day $d \in D$.
$\mathscr{L}_l \subseteq M$	An indexed set representing which meetings take place on venue $l \in L$.
$\mathscr{F} \subseteq T$	Set of final time slots each day.
$\mathscr{C}_c \subseteq T$	Indexed set of all intra-day consecutive time slots of length $c+1$.
$\mathcal{P} \subseteq \{\mathscr{I}, \mathscr{I}\}$	The tuple set of interviewer pairs (i, j) who can conduct interviews together.
$\mathcal{G}_g \subseteq \mathscr{I}$	The ordered set of interviewers that belong to group $g \in G$.
$\mathscr{D}_t \subseteq D$	A subset that lists the day that time slot $t \in T$ belongs to.
Parameters	
$d_m \geq 0$	The number of time slots needed for an interview $m \in M$.
$r_{tl} \geq 0$	Maximum number of rooms available at time $t \in T$ on venue $l \in L$.
$c \geq 0$	Maximum number of time slots an interviewer may undertake consecutively.
$s_i \geq 0$	How many simultaneous interviews that interviewer $i \in \mathscr{I}$ can conduct.
$u_{it} \in \{0,1\}$	A binary parameter denoting whether interviewer $i \in \mathscr{I}$ is unavailable at time $t \in \mathscr{T}$.
$p_t \geq 0$	Preference for undertaking an interview at time t.
$\omega^A, \omega^C, \omega^P \geq 0$	Weights for the assignment, consecutiveness, and preservation objectives, respectively.
Variables	
$x_{mt} \in \{0,1\}$	Meeting $m \in M$ is ongoing at time $t \in T$.
$y_{mt} \in \{0,1\}$	Meeting $m \in M$ begins at time $t \in T$.
$z_{ild} \in \{0,1\}$	Interviewer $i \in \mathscr{I}$ undertakes meetings at venue $l \in L$ on day $d \in D$.
$v_{it} \in \{0,1\}$	Interviewer $i \in \mathscr{I}$ attends two consecutive meetings at time $t \in T$ and $t+1 \in T$.
$w_{ijt} \in \{0,1\}$	Interviewer pair $(i,j) \in \mathcal{P}$ is preserved from time $t \in T$ to $t+1 \in T$.

4.1 Time Constraints

We must ensure that the time-related constraints are respected. First, we define that each applicant should have only one interview, equivalent to having one interview start. Additionally, all slots in a multi-slot interview must occur in one day instead of spanning over two days. Hence:

$$\sum_{m \in \mathscr{P}_a} \sum_{t \in \mathscr{B}_m} y_{mt} = 1 \quad \forall a \in \mathscr{A} \tag{1}$$

Second, we must define when the meeting starts and does not start, respectively, to help define that the d_m time slots are assigned consecutively:

$$\sum_{t=\tau}^{\tau+d_m-1} x_{mt} \leq (d_m - 1) + y_{m\tau} \quad \forall m \in M, \tau \in \mathscr{B}_m \tag{2}$$

$$\sum_{t=\tau}^{\tau+d_m-1} x_{mt} \geq d_m y_{m\tau} \quad \forall m \in M, \tau \in \mathscr{B}_m \tag{3}$$

Third, we define the required number of consecutive time slots required for the meeting:

$$\sum_{t \in T} x_{mt} = d_m \quad \forall m \in M \tag{4}$$

4.1.1 Handling infeasibilities

The model may be infeasible for selected test instances primarily for two reasons. First, the number of people applying to meet a selected interviewer may be greater than the time slots available. Second, the interviewers may have declared too many time slots in which they are unavailable that no feasible solution exists. The latter reason becomes more prevalent both the more unavailable a meeting participant is and the more people that attend meetings if they have all declared some unavailability. Additionally, it is generally harder to find a feasible schedule if time unavailability spans whole days—for example, two interviewers who are unavailable in the first and second half of the scheduling period render a schedule infeasible if they are supposed to attend at least one meeting together.

In order to handle infeasibility issues if they occur, constraints (1) and (4) must be modified. Specifically, constraint (1) must be rewritten to a smaller-than-or-equal-to constraint to allow some meetings to not start at all. Additionally, we must rewrite constraint (4) to specify that we either assign all or none of a meeting's time slots, depending on whether a meeting has been defined to start or not, respectively:

$$\sum_{t \in T} x_{mt} = d_m \sum_{\tau \in \mathscr{B}_m} y_{m\tau} \quad \forall m \in M \tag{5}$$

The choice of handling infeasibilities on an as-needed basis instead of by default is deliberate. Specifically, if fewer than all interviews could be scheduled from the onset, an optimal solution could allocate fewer than all interviews to schedule more consecutive interviews or preserve interviewer pairs across consecutive interviews. Since the primary aim of the project organizations is to schedule every applicant for an interview, it is preferred to prioritize scheduling everyone first and then focus on favorable allocations.

It could be possible to either penalize or reward the objective function for missing allocations or for allocating everyone, respectively. However, infeasibility is not a big concern for most test instances presented in this article because, after the first test instance presented in Section 5, the project organizations were told to allow only limited unavailability for each interviewer to ensure that the primary aim—to schedule all interviews—could be achieved. Hence, for the test instances presented in this article, it is wise to handle infeasibilities if they occur instead of handling such concerns to the model outright.

4.2 Infrastructural Constraints

Interviewers may conduct interviews at multiple venues. Due to the geographical distances between the venues, each meeting is pre-defined at a specific location to provide the applicants with short traveling distances. For such cases, the project organizations specify that they want every interviewer to undertake interviews at maximally one venue only per day, while the venue that an applicant will be interviewed at is pre-specified in the application form.

We must hence define if an interviewer is at venue l on day d:

$$\sum_{m \in \mathscr{P}_i \cap \mathscr{L}_l} \sum_{t \in \mathscr{T}_d} x_{mt} \leq |\mathscr{T}_d| z_{ild} \quad \forall d \in D, l \in L, i \in \mathscr{I} \tag{6}$$

If meetings had been undertaken with only two parties, then the allocation problem could have been split along the location dimension into multiple, smaller problems, potentially simplifying the solving process. However, since an arbitrary number of participants can meet and since the participants may meet on multiple occasions and locations, we may usually not be able to split the problem along the location dimension. Therefore, we specify that an interviewer can maximally be at one venue on any given day:

$$\sum_{l \in L} z_{ild} \leq 1 \quad \forall d \in D, i \in \mathscr{I} \tag{7}$$

Finally, as the meetings usually occur physically, there is a constraint on the number of rooms available at time t and venue l:

$$\sum_{m \in \mathscr{L}_l} x_{mt} \leq r_{tl} \quad \forall t \in T, l \in L \tag{8}$$

4.3 Consecutive Interviews Constraints

A key request from the project organizations was to ensure that interviewers could undertake consecutive interviews. Then the constraint specifying when consecutive time slots with allocated interviews are *not* achieved is defined as:

$$\sum_{m \in \mathscr{P}_i} x_{mt} + \sum_{n \in \mathscr{P}_i} x_{n,t+1} - \sum_{o \in \mathscr{P}_i} \sum_{\substack{\tau=\max(1,t-d_o+2): \\ d_o>1}}^{t} y_{o\tau} \geq 2v_{it} \quad \forall i \in \mathscr{I} \backslash \mathscr{S}, t \in T \backslash \mathscr{F} : |\mathscr{P}_i| > 1 \tag{9}$$

The left-hand side of (9) studies whether an interviewer i participates in two meetings at time t and $t+1$. However, a meeting that lasts longer than one time slot may have started at time t or before and still be ongoing at time $t+1$. To avoid defining such a meeting as a consecutive meeting, we need a correction term. The correction is represented by the third term on the left-hand side in (9), which will sum to one if a multi-slot meeting o that interviewer i is participating in is ongoing at time t but will end strictly after time t, and sum to zero otherwise. Additionally, since only consecutive time slots within a day are interesting, we apply this constraint only for intra-day consecutive slots, excluding end-of-day time slots $t \in \mathscr{F}$.

Finally, the constraint applies only to non-secretariat interviewers because the secretariat consists of multiple members that must be present either in a meeting or in the secretariat's office that is located near the meeting rooms. Hence, there is no need for consecutive interviews for the secretariat, and so the constraint applies for $i \in \mathscr{I} \backslash \mathscr{S}$.

4.3.1 Interviewing breaks

Another concern from the project organizations was that if the optimization model performed well enough to assign a multitude of consecutive interviews, the interviewers must also get occasional breaks of at least one time slot. This concern does not apply to the secretariat as they consist of multiple people.

$$\sum_{m \in \mathscr{P}_i} \sum_{t \in \mathscr{C}_c} x_{mt} \leq c \quad \forall i \in \mathscr{I} \backslash \mathscr{S}, c \in \mathbb{I}^{\mathscr{C}} \tag{10}$$

4.4 Preserving Interviewer Pair Constraints

After a dry-run producing a schedule, a project organization specified that it would be beneficial if interview pairs could be preserved from one consecutive interview to another. Preserving the interviewer pairs consequently made it easier for the interviewer pairs to compare candidates as candidates more frequently were scheduled consecutively.

Similarly to (9), a correction term is needed to exclude each multi-slot interview that started at time t or before and continues at time $t+1$. We can then define a constraint specifying that an interviewer pair does *not* remain from one interview m to another interview n:

$$\begin{aligned} \sum_{m \in \mathscr{P}_i \cap \mathscr{P}_j} x_{mt} + \sum_{n \in \mathscr{P}_i \cap \mathscr{P}_j} x_{n,t+1} - \sum_{o \in \mathscr{P}_i \cap \mathscr{P}_j} \sum_{\substack{\tau = \max(1, t-d_o+2): \\ d_o > 1}}^{t} y_{o\tau} \geq 2 w_{ijt} \quad \forall (i,j) \in \mathcal{P}, \\ t \in T \backslash \mathscr{F} : |\mathscr{P}_i \cap \mathscr{P}_j| > 1 \wedge i \notin \mathscr{S} \wedge j \notin \mathscr{S}. \end{aligned} \tag{11}$$

Similar to the consecutive interviews constraint (9), the preserving pairs constraint applies only to non-secretariat interviewers.

4.5 Other Interviewer Constraints

We must define that every interviewer can be in at most one interview at a time. However, some interviews will be undertaken partially by the secretariat, which consists of personnel that can undertake interviews when needed at the venue the secretariat is located in. The secretariat is homogeneous interviewers who can attend any meeting if needed, and each representative within this secretariat may be attending an interview at time t. Thus, a constraint is needed to specify the maximum number of simultaneous interviews that can be attended. The parameter s_i is equal to one for ordinary interviewers and may be greater than one for the secretariat at the venue. Then:

$$\sum_{m \in \mathscr{P}_i} x_{mt} \leq s_i \quad \forall i \in \mathscr{I} \cup \mathscr{S}, t \in T \tag{12}$$

Finally, the project organizations specified that seniority within groups should be respected. Specifically, for every time slot, the more senior group member must be assigned to undertake an interview before the less senior group member when there are multiple interviewers within a group:

$$\sum_{m \in \mathscr{P}_i \cap \mathscr{L}_l} x_{mt} + \sum_{\ell \in L: \ell \neq l} z_{i\ell d} \geq \sum_{n \in \mathscr{P}_j \cap \mathscr{L}_l} x_{nt} \quad \forall t \in T, g \in G \backslash \mathscr{S}, p \in 1, 2, .., (|\mathcal{G}_g| - 1), l \in L,$$
$$i = member(\mathcal{G}_g, p), j = member(\mathcal{G}_g, p+1), d = member(\mathscr{D}_t, 1) : |\mathcal{G}_g| > 1 \wedge u_{it} = 0 \tag{13}$$

The second term on the left-hand side of (13) will be equal to one if the more senior interviewer conducts interviews at a *different* venue than venue l, which will leave the first term on the left-hand side equal to zero. By including both terms on the left-hand side, the seniority concern is respected when the more senior interviewer is present at the venue l and disregarded otherwise. Additionally, the seniority concern is disregarded if the more senior interviewer is unavailable at time t as constraints are defined only if that interviewer is available.

4.6 Objective Function

Finally, the objective function must be defined. With $\omega^A + \omega^C + \omega^P = 1$ representing how much to focus on assigning interviews, achieving consecutive interviews, and achieving preserving interview pairs, respectively, and all at favorable times, the three objectives requested by the student association can be respected at a preferred weighting. While the project organizations primarily favored the two latter objectives, the assignment objective is included in the objective function to ensure that interviews can still be allocated to a favorable time slot, although they cannot necessarily be scheduled consecutively. Then the objective function is defined as:

$$\max_{y,v,w} \quad \sum_{m \in M} \sum_{t \in \mathscr{B}_m} \omega^A p_t y_{mt} + \sum_{\substack{i \in \mathscr{I} \backslash \mathscr{S}: \\ |\mathscr{P}_i| > 1}} \sum_{t \in T \backslash \mathscr{F}} \omega^C p_t v_{it} + \sum_{\substack{(i,j) \in \mathcal{P}: \\ |\mathscr{P}_i \cap \mathscr{P}_j| > 1 \\ \wedge i \notin \mathscr{S} \wedge j \notin \mathscr{S}}} \sum_{t \in T \backslash \mathscr{F}} \omega^P p_t w_{ijt} \tag{14}$$

5 Computational Study

Computations were done on a Linux Ubuntu 18.04 computer with an 18-core, hyper-threaded 2.6 GHz Intel Xeon Platinum 8272CL processor and 144 GB RAM. Problems were solved with the mathematical programming language AMPL and the commercial solver CPLEX version 12.10.0.0. CPLEX ran each instance for one hour unless optimality was found earlier.

5.1 Selecting the Priority Weights

The model was applied to five test instances from equally many hiring rounds within the project organizations. These instances involve scheduling 17–1149 meetings, and the overview of these instances can be seen in Table 4. While the smallest test instance could quickly be scheduled manually, this test instance was nevertheless scheduled by the model to understand how well the model performs on small instances. To select a general set of priority weights $(\omega^A, \omega^C, \omega^P)$, every test instance was run 60 times, differing only by different priority weights $(\omega^A, \omega^C, \omega^P) \in \big\{(i, j, k) : i, j, k \in \{0, 0.1, 0.2, .., 1\} \wedge i+j+k = 1\big\}$. These runs allow us to study how sensitive each objective is to a different tuple of priority weights and in which range we can achieve consistently good results across test instances.

Through experimentation, gradual improvements, and discussions with the project organizations about their preferences, the priority weights $(\omega^A, \omega^C, \omega^P) = (0.3,\ 0.5,\ 0.2)$ were selected. While all three objectives attempt to schedule meetings at favorable times, the assignment objective assigns meetings at favorable times also when consecutive interviews are not achieved. Since this situation frequently occurs for the test instances presented in this article, it is valuable assigning a strictly positive weight to the assignment objective. Additionally, in most interview scheduling instances of the size encountered in the project organizations, interviewers usually interview together with a varying set of different interviewers. This observation makes it challenging to preserve the interviewer pair across consecutive interviews and hence calls for a need to assign the consecutive-interviews objective a large weight. Hence, $(\omega^A, \omega^C, \omega^P) = (0.3,\ 0.5,\ 0.2)$ provides consistently good results for all three objectives—as illustrated with test instance 4 in Figure 1—across all test instances, and these weights also aligned well with the stated ambitions of the project organizations. Consequently, these priority weights will be used when studying the test instances' results next.

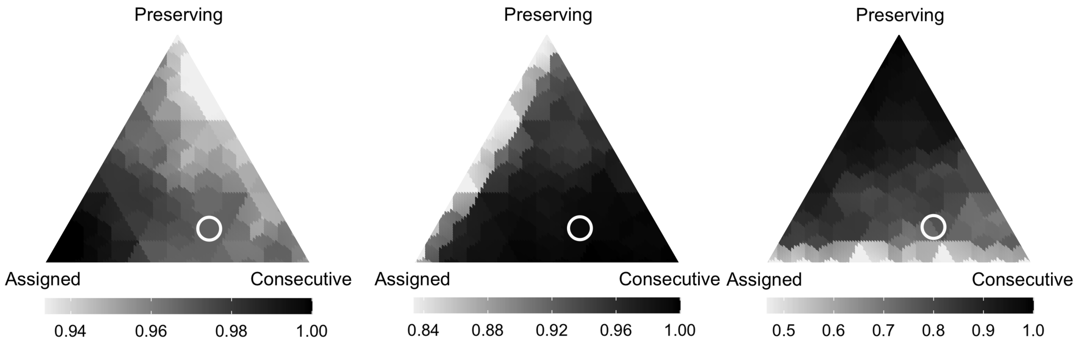

Figure 1: Test instance 4's scaled objective values for the assignment, consecutive, and preserving pairs objectives, respectively. Outlier values have been modified to improve interpretability of each figure. The white circle represents $(\omega^A, \omega^C, \omega^P) = (0.3, 0.5, 0.2)$.

5.2 Comparing Time Preference Allocations

Since the time preference parameter is included in all three terms in the objective function, allocating interviews at favorable time slots is naturally a key concern. Hence, it is valuable to construct a single comparable metric that explores how well time preferences are met across test instances. By employing an area under the curve (AUC) score, we can describe how favorable the assignments are—see Figure 2 for test instance 4's AUC curve.

In a standard AUC curve, a value of one is achieved if all interviews are allocated to the most favored time slots because the area under this cumulative ratio curve equals one. However, assigning all interviews to the most favored time slots may not be feasible because there may not be sufficient time slots, interviewers, or rooms available. Hence, we can adjust the AUC for these limitations by calculating an AUC score that allows comparing how good the solution is across test instances. This adjustment consequently calculates the cumulative *practical* maximum of assignments and is denoted by a solid horizontal line in Figure 2 if it is lower than the total number of assignments. Ultimately, the modified AUC score is calculated as the area below the cumulative, practically maximum number of assignments, and it is this score that is referred to when mentioning the AUC in the remainder of this section.

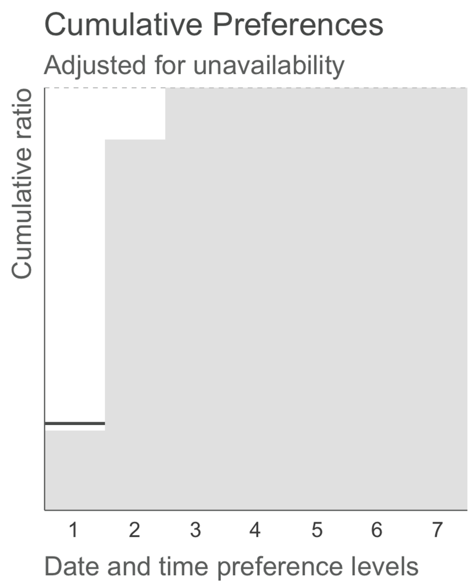

Figure 2: Cumulative ratio of preferences met for test instance 4.

5.3 Test Instances

Table 4: Results for all five test instances employing the priority weights $(w^A, w^C, w^P) = (0.3, 0.5, 0.2)$.

Instance	1	2	3	4	5
Interviewers	56	7	26	21	50
Applicants	812 (807)	17	113	98	1149
Time slots	240	96	96	72	180
Days	5	4	4	3	5
Variables	786 765 (786 765)	6752	60 784	33 978	652 575
Constraints	358 908 (359 001)	5446	28 282	32 357	419 484
Running time (seconds)	Inf. (3600)	1.00	3600	3600	3600
Relative MIP gap (percent)	Inf. (37.8)	0	5.38	1.91	53.71
AUC (percent)	Inf. (85.9)	100.0	97.2	97.7	81.3
Consecutive allocations (percent)	Inf. (90.6)	94.1	98.1	97.2	84.2
Preserving pairs (percent)	Inf. (16.6)	8.3	17.5	27.8	9.3

Each test instance varies in size, performance, or feasibility. Specifically, test instance one attempts to schedule 812 interviews, but due to excessive unavailability among the interviewers—with some being unavailable the first four days and their interviewing partner being unavailable the fifth and final days—the model was rendered infeasible. After modifying the model to handle infeasibilities, as described in Section 4.1.1, 807 of these 812 interviews were successfully scheduled. Due to the size of the problem—given the number of interviewers, applicants, and time slots—the relative MIP gap was as high as 37.8% when the stopping time of 3600 seconds was reached. Nevertheless, 90.6% of the meetings were scheduled consecutively, and an AUC score of 85.9% was achieved.

The results are similar to test instance five, in which 1149 interviews must be scheduled. Despite a high relative MIP gap of 53.71%, the highest among all test instances, 84.2% of interviews are scheduled consecutively, and an AUC score of 81.3% is achieved. Despite few interviewer pairs being preserved in consecutive meetings, the quality of this schedule is a substantial improvement over comparable schedules the project organizations produced manually in the past.

The results improve substantially compared to test instances one and five for smaller test instances. Test instance two schedules only 17 interviews and achieves optimality in one second. Test instance three and four, in comparison, does not achieve optimality within the 1-hour running time but achieve low relative MIP gaps of 5.38% and 1.91%, respectively. While both the AUC scores and the number of consecutive allocations improved markedly compared to instances one and five, the most noteworthy improvement is the increase in the percentage of interviewer pairs being preserved across consecutive interviews.

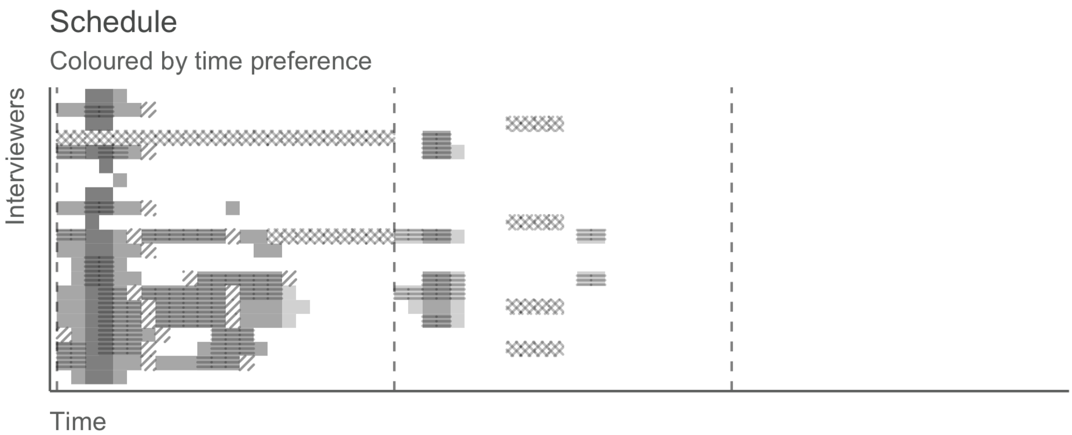

Figure 3: Interview schedule for test instance 4.

Figure 3 shows the schedule in test instance four, when the solver was stopped after 60 minutes. The horizontal axis represents the time slots—with the dashed vertical lines being the day separators—and the vertical axis represents the interviewers. The solid blocks represent the assigned interviews, and the shades represent the time preference—with a darker shade symbolizing a stronger time preference. Moreover, the crosshatches represent the time slots in which the interviewers are unavailable, which means that meetings involving these interviewers should not be scheduled for these times. Additionally, angled lines between blocks of consecutive interviews represent the break we enforce when consecutive interviews would otherwise be too long for the interviewers. Finally, horizontal lines overlapping the solid blocks represent consecutive interviews in which the interviewer pair was preserved.

Additionally, Figure 3 shows that most interviews are allocated at favorable times. However, while all interviews could theoretically have been assigned for the first two time-preference levels, only 87.8% were. The primary reasons for this lower score are the unavailability of two interviewers in particular—which is illustrated with crosshatches in Figure 3—and that the model also attempts both to schedule interviews consecutively and preserve pair allocations. For test instance four, the scheduling quality is satisfying with an AUC of 97.7%, consecutive allocations for 97.2% of the meetings, and preserving pair allocations for 27.8%. These metrics are substantially better than when similar schedules were produced manually, which is testified by the person responsible for the schedule:

> If one compares the plan's quality with how it would have turned out if we had made it manually, we see that it is much better. We had none of the typical "human" errors we experienced when doing it manually in earlier interviewing rounds, such as double-bookings. Instead, the interviewers experienced an excellent flow, with many consecutive interviews early in the week and frequent breaks. That way, we could do interviews for any potential leftover positions at the end of the week. We are incredibly pleased with the produced plan [and the corresponding workflow], as it was efficient, simple to work with, and of good quality.

Moreover, one project organizer noted that the recruiting part of student association work is a key reason students emphasize when explaining why they do not want to engage and volunteer in the student association. These considerations, coupled with the fact that the students are allocated to responsibilities they prefer and the leaders of these responsibilities must conduct these interviews, causes substantial overhead efforts that must be made. Consequently, the student association reports that they have experienced positive effects on their recruiting process due to three main factors.

First, the project organizations have saved substantial time in creating the interview schedule, with the project organizer for test instance one reporting that they reduced their scheduling time from approximately 100 hours to 1 hour. However, while the time savings were large on an individual basis, only few individuals are involved in creating the schedule. Hence, this factor's impact on the student association is only modest.

In comparison, the second factor does not save any meeting time on an individual basis because the number and duration of interviews are predefined, but utilizes the interviewers' time at large more efficiently. Specifically, by scheduling numerous consecutive interviews, the interviewers have more time to study without frequent interruptions to conduct interviews. Both since frequent interruptions were common in the manually produced schedules and since many interviewers are affected by a better schedule, the schedule's impact on the student association has been reported by the project organizations as substantial.

And third, while the project organizations must hire approximately 3000 volunteer students over a two-year duration, many of these students have previously participated in the project organizations but usually in different positions. Therefore, many students have already experienced first-hand how relatively frictionless interviewing rounds were when schedules were produced with the model presented in this article. Consequently, the project organizations have reported that it is generally easier to recruit students because the time spent on recruiting is not as big a concern as it originally was when the schedules were produced manually. Each schedule produced by the model in this article hence creates positive spillover effects, which hopefully will contribute in sustaining a high volunteering volume in these project organizations.

Ultimately, one main aim of these project organizations is to provide future leaders with sought-after experience. As many of the students participating in the project organizations will take on future leading positions in the business landscape and executives believe leadership skills improve with volunteering (Deloitte, 2016), it is the hope that the impact of this decision support tool will extend well beyond its direct effects for the students at NHH.

6 Concluding Remarks

This article provides an applied model to schedule en-masse hiring rounds within volunteer organizations. The problem was solved with a multi-objective integer programming model with three objectives: to assign interviews at favorable times, assign interviewers with consecutive interviews at favorable times, and assign consecutive interviews at favorable times in which the interviewer pairs are preserved.

The weighting of each objective was first assigned with various values, and advantageous weights of each objective were identified. Due to the extensively intersecting properties of the three objectives, the generally most advantageous objective function value was achieved with a high weighting of the preserving pairs objective, a small but strictly positive weighting of the consecutive pairs objective, and an even smaller yet strictly positive weighting of the simple assignment objective.

The model has been implemented in the hiring rounds of the student association at NHH, studying test instances ranging from 17 to 1149 meetings to be scheduled. Compared to former, manually created schedules, both the volunteer organization's and the interviewers' perceived experience is markedly improved when a non-trivial number of meetings are scheduled, with considerable improvements in all three model objectives. Due to the high degree of satisfaction after implementing the model, the student association intends to continue using the model for each future hiring round.

The model presented in this article can be expanded to further contribute to the literature in two primary directions. First, future work should incorporate a fair allocation of meetings to make the individuals' schedules more evenly good. This feature is missing from the literature altogether today. Second, during COVID-19, whole organizations moved to online communication platforms and digitized their work calendars across the organization. Since meetings create attention residue for many office workers, making it particularly difficult to focus on cognitively demanding tasks shortly after such events (Leroy, 2009), it is valuable to schedule these meetings more efficiently—such as more consecutively. Therefore, future work also involves modifying the meeting scheduling problem to produce updated schedules frequently to let workers utilize their working days better.

References

Aizam, N. A. H., & Sim, S. L. P. (2016). Integrating mathematical model and scheduling problem. *AIP Conference Proceedings*, *1775*(1), 030065.

Bartholdi III, J. J., & McCroan, K. L. (1990). Scheduling interviews for a job fair. *Operations Research*, *38*(6), 951–960.

Corporation for National and Community Service. (2013). *Volunteering as a pathway to employment: Does volunteering increase odds of finding a job for the out of work?*

Deloitte. (2016). *Building leadership skills through volunteerism* (2016 Deloitte Impact Survey).

Ernst, A. T., Mills, R., & Welgama, P. (2003). Scheduling appointments at trade events for the Australian Tourist Commission. *Interfaces*, *33*(3), 12–23.

Gebser, M., Glase, T., Sabuncu, O., & Schaub, T. (2013). Matchmaking with answer set programming. *International Conference on Logic Programming and Nonmonotonic Reasoning*, 342–347.

Gordon, L., & Erkut, E. (2004). Improving volunteer scheduling for the Edmonton Folk Festival. *Interfaces*, *34*(5), 367–376.

Gueret, C., Bellenguez-Morineau, O., Pavageau, C., Péton, O., & Poncelet, D. (2009). Optimisation de la planification de bourses d'échanges de technologies. *ROADEF*, *9*, 177–178.

Huang, Y., Zhang, X., & Alexander, P. S. (2012). A heuristic algorithm for optimizing business matchmaking scheduling. *International Journal of Operations Research and Information Systems (IJORIS)*, *3*(4), 59–73.

Kiyonari, Y., Miyano, E., & Miyazaki, S. (2006). Computational complexity issues in university interview timetabling. *Practice and Theory of Automated Timetabling VI*, 448.

Le Roux, A., Bellenguez-Morineau, O., Guéret, C., & Prot, D. (2015). Complexity of a one-to-one meeting scheduling problem with two types of actors. *Journal of the Operational Research Society*, *66*(10), 1730–1741.

Leroy, S. (2009). Why is it so hard to do my work? The challenge of attention residue when switching between work tasks. *Organizational Behavior and Human Decision Processes*, *109*(2), 168–181.

Mausser, H. E., Magazine, M. J., & Moore, J. B. (1996). Application of an annealed neural network to a timetabling problem. *INFORMS Journal on Computing*, *8*(2), 103–117.

Pesant, G., Rix, G., & Rousseau, L.-M. (2015). A comparative study of MIP and CP formulations for the B2B scheduling optimization problem. *International Conference on AI and OR Techniques in Constriant Programming for Combinatorial Optimization Problems*, 306–321.

Rinaldi, F., & Serafini, P. (2006). Scheduling school meetings. *International Conference on the Practice and Theory of Automated Timetabling*, 280–293.

Sampson, S. E. (2006). Optimization of volunteer labor assignments. *Journal of Operations Management*, *24*(4), 363–377.

Schrage, L. (2004). Scheduling job placement interviews at a university.

Shin, S., & Kleiner, B. H. (2003). How to manage unpaid volunteers in organisations. *Management Research News*.

United Nations Volunteers programme. (2018). *The thread that binds–volunteerism and community resilience* (2018 State of the World's Volunteerism Report).

NHH

NORGES HANDELSHØYSKOLE
Norwegian School of Economics

Helleveien 30
NO-5045 Bergen
Norway

T +47 55 95 90 00
E nhh.postmottak@nhh.no
W www.nhh.no