Les Cahiers du GERAD ISSN: 0711–2440

Scheduling ISMP 2024

C. Audet, M. Gervais-Dubé, A. Hertz, S. Le Digabel, A. Legrain

G–2025–35

April 2025

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2025
– Bibliothèque et Archives Canada, 2025

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2025
– Library and Archives Canada, 2025

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Scheduling ISMP 2024

Charles Audet [a, b]

Mathieu Gervais-Dubé [a, b]

Alain Hertz [a, b]

Sébastien Le Digabel [a, b]

Antoine Legrain [a, b]

[a] *GERAD, Montréal (Qc), Canada, H3T 1J4*

[b] *Département de mathématiques et de génie industriel, Polytechnique Montréal, Montréal, (Qc), Canada, H3T 1J4*

charles.audet@gerad.ca
mathieu.gervais-dube@polymtl.ca
alain.hertz@gerad.ca
sebastien.le.digabel@gerad.ca
antoine.legrain@polymtl.ca

April 2025
Les Cahiers du GERAD
G–2025–35

Abstract : Researchers around the globe attend the International Symposium on Mathematical Programming (ISMP) to share their latest results in mathematics, algorithms, computation, and modeling. The 2024 edition, held in Montréal over 5 days, gathered close to 1,600 participants and included more than 1,300 talks. Building a reasonable schedule with these figures cannot be done by hand. The objective of this document is to describe how the scheduling, the numerous re-schedulings as well as the room assignment problems were modeled and optimized.

Concerning past ISMP conferences, the schedule of ISMP 2018 was done using a tool called "the Scheduler", and the present work integrates many ideas from the model of the 2012 and 2015 editions described in [6]. More widely, the following studies focus on the optimization of conference schedules [1, 2, 3, 4, 5, 7].

1 Introduction

ISMP[1]has many types of sessions: keynotes, plenaries, panels, meetings, and regular ones. The first four types were scheduled by the organizers, and the present work only deals with the latter type of sessions. Each talk is assigned to one of four clusters (continuous optimization, discrete optimization, optimization under uncertainty, and applications and computation), and each cluster is divided into streams. There are 29 streams, covering most areas of optimization. Each session of a stream is composed of 3 or 4 talks and is directed by a chair.

The scheduling committee imposed a timetable with 13 available time blocks (three per day, except for the opening and closing days – see Figure 1.1) in which the sessions would take place. A total of 32 rooms, of various sizes, are available for the talks. Each talk needs to be assigned to a stream, a room and a time block which forms a triplet. Talks assigned to the same stream, room and time block form a session.

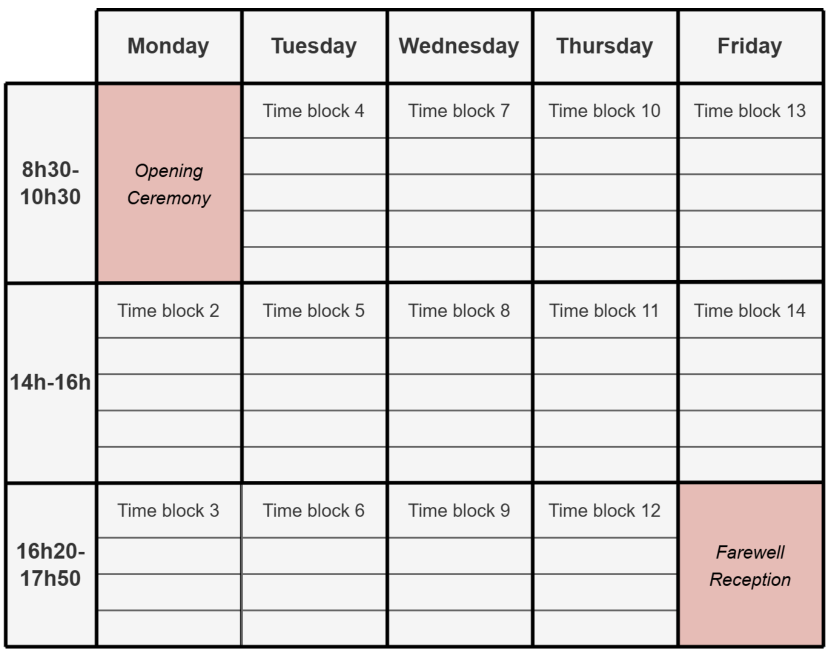


Figure 1.1: Available time blocks for the sessions.

The talk's attributes needed to be taken into account to make a judicious schedule. There were two ways to submit a talk: either by invitation to a stream or a session, or without it. The former is called an *invited talk*, and the latter a *contributed talk*. If a talk is invited to a specific session, then it is grouped with other talks and they are together assigned to their corresponding stream. If a talk is simply invited to a stream, then it is pre-assigned to this stream and will be aggregated with other talks during the optimization process.

In addition, participants had to select two to six keywords from a fixed list, and could select up to two preferred streams for *contributed talks*.

Finally, the general constraints for constructing the schedule are the following.

1. Each talk must be assigned to a single session.
2. A participant cannot speak in or chair two simultaneous sessions.

[1]https://ismp2024.gerad.ca

3. At all time, the number of parallel sessions must not exceed the total number of available rooms.
4. The number of talks in each session is bounded, with values that depend on time blocks.
5. If possible, any pair of sessions belonging to the same stream should not be held during the same time block.

In addition, other objectives need to be considered. Sessions should ideally consist of talks belonging to the same research area. If possible, sessions should be full to make it worthwhile to attend, and each talk should be assigned to a stream that corresponds to its subject. Furthermore, sessions need to be assigned to suitable rooms.

To meet these objectives, the optimization process is pipelined into three steps. First, promising groups of talks with affinities are created using heuristics, by comparing keywords and preferred streams. Combinations of these groups are assigned to streams and time blocks to form sessions that respect the constraints. This talk assignment process is described in Section 2, using a mixed-integer linear program. Second, Section 3 describes how consecutive reoptimizations are performed to satisfy new demands of participants, while modifying the least number of previously assigned talks. Finally, once that all sessions are formed, Section 4 presents a mixed-integer linear program to assign sessions to available rooms.

Numerical results that uses the data of ISMP2024 are shown in Section 5. Finally, Section 6 concludes with possible improvements, and links to MIP formulations for benchmarking purposes.

The formal nomenclature and notations used throughout the formulations of the problems are presented in the appendix.

2 Talk assignment

2.1 Aggregating talks in groups

This section details the method used to aggregate talks into groups of 1 to U_{talk}. All groups formed by this method form the aggregated set $\mathcal{L}$ of groups of talks . For ISMP 2024, the upper bound on the number of talks per session is $U_{\text{talk}} = 4$.

A simple graph $G = (\mathcal{I}, E)$ is created where $\mathcal{I}$ is the set of talks and E is the set of edges. An edge between two talks is present if and only if they have a common keyword.

A group is said to be *promising* if all of its talks have at least one common keyword. Therefore, a necessary condition for a group to be promising is that it needs to be a clique in G. Figure 2.1 illustrates an example.

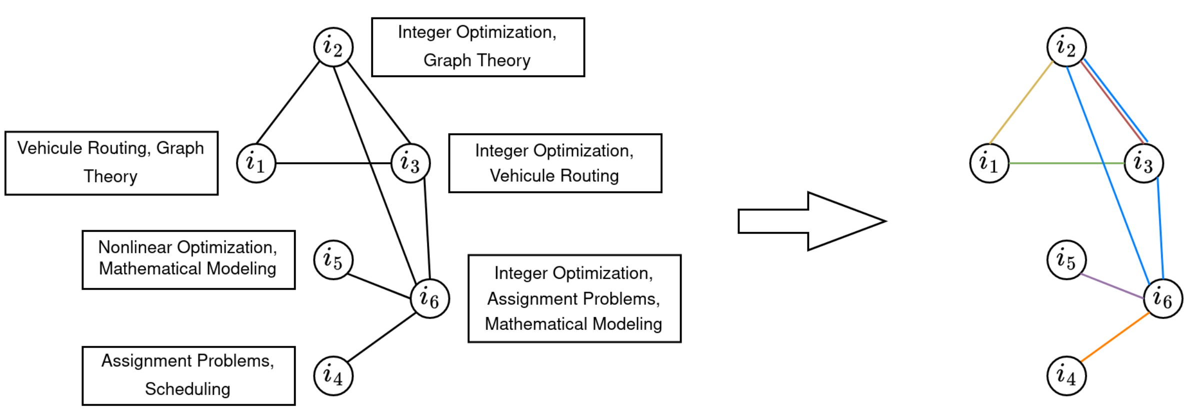


Figure 2.1: Examples of promising groups of talks represented by colored cliques.

Therefore, all cliques of sizes between one and U_{talk} are extracted from the graph. There can be up to $\sum_{i=1}^{U_{\text{talk}}} \binom{|\mathcal{I}|}{i}$ cliques for promising group of talks. However, the number of promising groups is less in practice. First, the diversity of the talks' subjects makes it less probable that there are many cliques with a common keyword. Second, pre-assigned group of talks cannot be separated. The following list summarizes the rules used to select groups of talks that will define $\mathcal{L}$.

1. A clique is not selected if there are no common keywords.
2. If a clique contains a pre-assigned talk, then it must also contain its pre-assigned group, otherwise the clique is not selected.

In the worst-case scenario, there can be up to a quadratic number of cliques of size two. An additional filter is used on them to limit their number, such as requiring the clique to have at least two common keywords.

Note that a talk can be part of multiple groups. The set $\mathcal{L}$ used by the mixed-integer linear program is finalized once that all cliques have been processed.

2.2 The talks' assignment formulation

The model used to assign the talks to streams and time blocks is detailed next. The decision variables are

$$x_{\ell,k,j} := \begin{cases} 1, & \text{if the group of talks } \ell \\ & \text{is assigned to time block } j \\ & \text{of stream } k. \\ 0, & \text{otherwise.} \end{cases}$$

A second variable measures if sessions of capacity four are at full capacity, a third determines if the sessions take place, and a fourth checks if a stream is used.

$$z_{k,j} := \begin{cases} 1, & \text{if the time block } j \notin \mathcal{J}_{\text{short}}, \\ & \text{of stream } k \\ & \text{is filled with talks.} \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{k,j} := \begin{cases} 1, & \text{if the time block } j, \\ & \text{of stream } k \\ & \text{has at least one talk.} \\ 0, & \text{otherwise.} \end{cases}$$

$$s_k := \begin{cases} 1, & \text{if stream } k \text{ contains} \\ & \text{at least one session.} \\ 0, & \text{otherwise.} \end{cases}$$

Finally, symbols used in the general model are introduced.

1. $c(\ell, k) : \mathcal{L} \times \mathcal{K} \to \mathbb{N}$ is a bonus given for assigning the group of talks $\ell \in \mathcal{L}$ to the stream k. It takes into account the internal affinity of the group and the affinity of the stream, as detailed in Section 2.4.
2. $w(k) : \mathcal{K} \to \mathbb{Z}$ is a small negative weight that penalizes the creation of parallel streams so that the stream time blocks are filled when possible, as detailed in Section 2.5.
3. Let $\ell \in \mathcal{L}$, then $n_\ell \in \mathbb{N}^*$ is the number of talks in the group of presentations ℓ.

4. Let $j \in \mathcal{J}$, then let U_j be a bound on the number of presentations for the time block j.

$$U_j = \begin{cases} U_{\text{talk}}, & \text{if } j \in \mathcal{J} \setminus \mathcal{J}_{\text{short}}. \\ U_{\text{talk}} - 1, & \text{if } j \in \mathcal{J}_{\text{short}} \ . \end{cases}$$

and let $L_j = U_{\text{talk}} - 1$ be a lower bound on the number of presentations for the time block j.

2.3 Possible streams of talks

A mining approach is used on the talks' keywords to reduce the number of possible streams for each talk. The idea is to have a set of keywords associated with each stream. If a talk does not have any keyword in common with the stream, then the stream will not be possible for this talk, thereby reducing the number of variables for the problem. The possible keywords for a stream $k \in \mathcal{K}$ are defined as the union of all keywords of all contributed and invited talks that have k as one of its preferred or assigned stream.

Following this idea, the possible streams for a talk are all those with a keyword in common. The possible streams $\mathcal{K}_\ell$ for a group of talks $\ell \in \mathcal{L}$ is the intersection of the possible streams of each of its composing talks.

Equipped with these variables, and using the aggregated set of group of talks $\mathcal{L}$, the talks' assignment model is as follows.

$$\max_{\substack{x \in \{0,1\}^{|I| \times |\mathcal{J}| \times |\mathcal{K}|} \\ y \in \{0,1\}^{|\mathcal{J}| \times |\mathcal{K}|} \\ z \in \{0,1\}^{|\mathcal{J}| \times |\mathcal{K}|} \\ s \in \{0,1\}^{|\mathcal{K}|}}} \quad \sum_{\ell \in \mathcal{L}} \sum_{k \in \mathcal{K}_\ell} \sum_{j \in \mathcal{J}} c(\ell, k) x_{\ell,k,j} + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} z_{k,j} + \sum_{k \in \mathcal{K}} w(k) s_k,$$

subject to

$$\sum_{\ell \in \mathcal{L}_i} \sum_{k \in \mathcal{K}_\ell} \sum_{j \in \mathcal{J}} x_{\ell,k,j} = 1 \ \forall \ i \in \mathcal{I},$$

a talk is assigned to a single session.

$$\sum_{\ell \in \mathcal{L}} n_\ell x_{\ell,k,j} \le U_j y_{k,j}, \ \forall \ k \in \mathcal{K}_\ell \text{ and } \ \forall \ j \in \mathcal{J},$$

is (k, j) an empty session?

$$\sum_{\ell \in \mathcal{L}} n_\ell x_{\ell,k,j} \ge L_j y_{k,j} + z_{k,j}, \ \forall \ k \in \mathcal{K}_\ell \text{ and } \ \forall \ j \in \mathcal{J},$$

is (k, j) a full session (four talks)?

$$\sum_{k \in \mathcal{K}} y_{k,j} \le |\mathcal{R}|, \ \forall \ j \in \mathcal{J},$$

each time block has at most $|\mathcal{R}|$ parallel sessions.

$$\sum_{\ell \in \mathcal{L}_a} \sum_{k \in \mathcal{K}} x_{\ell,j,k} \le 1, \ \forall \ j \in \mathcal{J}, \ \forall \ a \in \mathcal{A},$$

participant a is involved in at most one session of time block j.

$$\sum_{j \in \mathcal{J}} y_{k,j} \le s_k, \ \forall \ k \in \mathcal{K}.$$

is stream $k \in \mathcal{K}$ used?

2.4 Assignation bonuses

The weights of the objective function of the talks' assignment model are detailed in this section.

The bonus scheme is designed to favor certain configurations in the result of the optimization while also taking into account common keywords between talks. Each bonus $c(\ell, k)$ is composed of two parts. The first part takes into account the group affinity through common keywords. The second part takes into account the talks affinities with the stream. This second bonus is greater if the talks' keywords are in common with the stream's keyword and if the stream is the suggested stream of some of the talks. The stream affinity bonus $b_{\text{stream}}(i,k) : \mathcal{I} \times \mathcal{K} \to \mathbb{N}$ is defined below along with the bonus $b_{\text{group}}(\ell) : \mathcal{L} \to \mathbb{N}$ which is used in the bonus scheme of Table 2.1. For ISMP2024, the stream bonuses were manually adjusted by analyzing assignments for different values of $b_{\text{suggested}}$.

$$b_{\text{stream common}}(i,k) := \text{number of common keyword between the stream and talk}, \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}$$

$$b_{\text{suggested}}(i,k) := \begin{cases} 5 \text{ if } k \text{ is the suggested stream 1 of } i, \\ 3 \text{ if } k \text{ is the suggested stream 2 of } i, \\ 0 \text{ otherwise.} \end{cases} \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}$$

$$b_{\text{stream}}(i,k) := b_{\text{stream common}}(i,k) + b_{\text{suggested}}(i,k), \qquad \forall i \in \mathcal{I}, k \in \mathcal{K}$$

$$b_{\text{group}}(\ell) := \text{number of keywords in common between all talks in the group}, \qquad \forall \ell \in \mathcal{L}$$

Table 2.1: Bonus associated to the configurations where $c(\ell,k)$ = internal affinity(ℓ)+stream affinity(ℓ,k), $\forall\, \ell \in \mathcal{L}, k \in \mathcal{K}$.

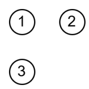
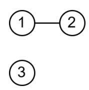
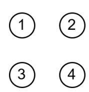
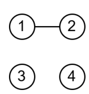
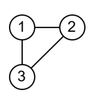
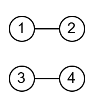
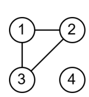
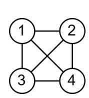
$\mathbf{config(k,j)}$	**Configuration**	**Internal Affinity**(ℓ)	**Stream Affinity**$(\ell,\mathbf{k})$
$\{1,1,1\}$		0	$\sum_{i\in\{1,2,3\}} b_{\text{stream}}(i,k)$
$\{1,2\}$		$2 + b_{\text{group}}(\{1,2\})$	$\sum_{i\in\{1,2,3\}} b_{\text{stream}}(i,k)$
$\{1,1,1,1\}$		0	$\sum_{i\in\{1,2,3,4\}} b_{\text{stream}}(i,k)$
$\{1,1,2\}$		$2 + b_{\text{group}}(\{1,2\})$	$\sum_{i\in\{1,2,3,4\}} b_{\text{stream}}(i,k)$
$\{3\}$		$3 + 3b_{\text{group}}(\{1,2,3\})$	$\sum_{i\in\{1,2,3\}} b_{\text{stream}}(i,k)$
$\{2,2\}$		$4 + b_{\text{group}}(\{1,2\}) + b_{\text{group}}(\{3,4\})$	$\sum_{i\in\{1,2,3,4\}} b_{\text{stream}}(i,k)$
$\{1,3\}$		$3 + 3b_{\text{group}}(\{1,2,3\})$	$\sum_{i\in\{1,2,3,4\}} b_{\text{stream}}(i,k)$
$\{4\}$		$4 - 8b_{\text{group}}(\{1,2,3,4\}) + 3 \sum_{s\in\binom{\{1,2,3,4\}}{3}} b_{\text{group}}(s)$	$\sum_{i\in\{1,2,3,4\}} b_{\text{stream}}(i,k)$

Table 2.1 provides examples of the possible scenarios for the distribution of groups in a session, with their associated bonus calculation. The numbers $1,2,3$ and 4 in the internal affinity column refer to the talks (vertices) shown in the configuration column.

2.5 Stream copies penalties

The stream penalty $w(k),\ k \in \mathcal{K}$ is used to allow talks to be assigned to a given stream even if all of its possible time blocks are full. This results to the creation of parallel sessions for the stream. To get the number of copies of a given stream, the total number of talks that have the stream as a possible stream is divided by the number of talks possible in a stream.

The stream whose possible time blocks are full is called the *original* stream, and the additional ones are called *copies*.

However, it is desired to limit the number of parallel sessions. Therefore, the weight $w(k)$ is negative when $k \in \mathcal{K}$ is a copy of a stream. The weight is designed as follows:

$$w(k) = \begin{cases} 0 & \text{if } k \text{ is an original stream,} \\ -i & \text{if } k \text{ is the } i\text{th copy of a stream.} \end{cases}$$

2.6 Additional constraints

In this section, additional constraints to manage participants demands are provided. Some participants have scheduling restrictions. For example, there can be some visa restrictions or flight booking problems that force participants to be available during a limited set of time blocks. This is easily modeled by setting appropriate decision variables to zero. If participant $a \in \mathcal{A}$ is unavailable for a subset of time blocks $J \subseteq \mathcal{J}$, then

$$x_{\ell,k,j} = 0, \qquad \forall \ell \in \mathcal{L}_a, k \in \mathcal{K}_\ell, j \in J.$$

Another constraint specifies time blocks $J \subset \mathcal{J}$ and/or streams $K \subset \mathcal{K}$ to a talk:

$$\sum_{j \in J} \sum_{k \in (K \cap \mathcal{K}_\ell)} \sum_{\ell \in \mathcal{L}_i} x_{\ell,k,j} \geq 1.$$

Some groups may need to be ordered by the request of stream managers. Therefore, the following constraint may be required. Let $L = (\ell_1, \ell_2, \ldots, \ell_n) \subseteq \mathcal{L}$ be an ordered set of invited groups such that if $\ell_i \in L$ is before $\ell_j \in L$ then ℓ_i must be assigned to a time block that precedes the one of ℓ_j. That is, ℓ_i occurs before ℓ_j during the conference. Let $\mathcal{P}(2, L) : \{(\ell_i, \ell_{i'}) \in L \times L \ :\ i = i' - 1\}$ be the set of subsequent pairs of talks in L. The constraint is as follows.

$$\sum_{\ell \in \mathcal{L}_{\ell_i}} \sum_{k \in \mathcal{K}} x_{\ell,k,j'} \leq \sum_{\ell \in \mathcal{L}_{\ell_{i'}}} \sum_{k \in \mathcal{K}} \sum_{\substack{j \in \mathcal{J}: \\ j \leq_{I_\mathcal{J}} j_{\text{last}}, \\ j \geq_{I_\mathcal{J}} j'+1}} x_{\ell,k,j} \qquad \forall\ (\ell_i, \ell_{i'}) \in \mathcal{P}(2, L), \forall j' \in \mathcal{J} \setminus \{j_{\text{last}}\} \tag{2.1}$$

$$\sum_{\ell \in \mathcal{L}_{\ell_i}} \sum_{k \in \mathcal{K}} x_{\ell,k,j_{\text{last}}} = 0 \qquad \forall (\ell_i, \ell_{i'}) \in \mathcal{P}(2, L) \tag{2.2}$$

2.7 Ordering the talks and chairing

Once all the talks have been assigned to a session $(k, j) \in \mathcal{K} \times \mathcal{J}$, a rule is used to determine the order of the talks and the chair of the session. Before describing the rules for the ordering and chairing, an ordering criterion is defined.

Definition 2.1 (Ordering Criterion)**.** For $(k, j) \in \mathcal{K} \times \mathcal{J}$, let $s_{k,j} := \{i \in \mathcal{I} : i \text{ is assigned to the session } (k, j)\}$, be the set of talks that form a session (k, j).

Then, $s_{k,j}$ respects the talks' ordering criterion if the talks are ordered so that the total number of common keywords between pairs of subsequent talks is a maximum.

The following rules are used to determine the order of the talks for each session, and to assign chairs.

1. In a session, contributed talks are prior to invited talks, and the chosen order is the one that respects the talks' ordering criterion under this constraint.
2. For sessions with only invited talks, the order is determined by the group manager.
3. For sessions with only contributed talks, the order must respect the ordering criterion.
4. If the session is composed of an invited group, the chair of the invited group is the chair of the session.
5. If no chair is assigned to a session, then the chair is the last speaker of the session.

3 Reoptimization

This short section describes the reoptimization process for the reassignment of talks due to unforeseen demands or requests.

After the initial schedule is published online for participants, it is crucial to minimize the number of changes in subsequent schedules to avoid disrupting their event planning. Accordingly, the bonus scheme for the reoptimizations process takes into account the solution of the last published schedule.

For each reoptimization, the bonus $c(\ell, k)$ is increased to take into account the time block dimension $\mathcal{J}$. For this purpose, let $c^i(\ell, k, j), i \in \mathbb{N}$ be the bonuses of the ith optimization, where $c^0(\ell, k, j) = c(\ell, k)\ \forall j \in \mathcal{J}$ are the bonuses associated with the first optimization. Let $\hat{x}^i_{l,k,j}$ be the value of $x_{l,k,j}$ at the solution of the ith optimization. These new bonuses $c^i(\ell, k, j)$ $(i > 0)$ for the ith optimization are modified for the talks so that they ideally stay in the stream k and as close as possible to the time block j and with the group ℓ of talks in which they were assigned. This is done using the following rule.

$$c^i(\ell, k, j) = \begin{cases} c(\ell, k) + 6 & \text{if } \exists \hat{x}^{i-1}_{l,k,j} = 1 :\ \ell \subseteq l, \\ c(\ell, k) + 5 & \text{if } \exists \hat{x}^{i-1}_{l,k,j'} = 1 : j' \neq j \\ & \wedge\ \ell \subseteq l \\ & \wedge (\lfloor \frac{j'-1}{3} \rfloor = \lfloor \frac{j-1}{3} \rfloor), \\ c(\ell, k) & \quad \text{otherwise.} \end{cases}$$

The variables of the reoptimizations models are the same, but the coefficient $c(\ell, k)$ associated with $x_{\ell,k,j}$ in the objective function is replaced by $c^i(\ell, k, j)$.

4 Room assignment

Finally, each session needs to be assigned to a room. Since the talks' assignment model ensured that the number of parallel sessions is bounded by the total number of available rooms ($|\mathcal{R}| = 32$ for ISMP 2024), the room assignment problem is always feasible.

Recall that $\mathcal{K}$ is the set of streams, totally ordered with an arbitrary $I_{\mathcal{K}}$. Let $\mathcal{S}$ be the set of all the sessions, $\mathcal{S}_j$ be the set of sessions that take place in the time block $j \in \mathcal{J}$, and $\mathcal{S}^k$ be the ordered set of sessions part of the stream $k \in \mathcal{K}$, where the sessions in $\mathcal{S}^k$ are ordered by their time of occurrence.

Consider the decision variable $x_{s,r} \in \{0, 1\}$ with $r \in \mathcal{R}$ and $s \in \mathcal{S}$ where

$$x_{s,r} := \begin{cases} 1 & \text{if session } s \text{ takes place in room } r, \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

4.1 Room assignment bonus design

Let $c(s, r) : \mathcal{S} \times \mathcal{R} \to \mathbb{R}$ be the bonus associated with the choice of $x_{s,r}$ and set

$$\mathrm{og}(k) : \mathcal{K} \to \mathcal{K} := \begin{cases} k & \text{if } k \text{ is an original stream.} \\ k' & \text{where } k' \text{ is the original} \\ & \text{stream of } k \text{ if } k \text{ is not an} \\ & \text{original stream.} \end{cases}$$

Let $\mathrm{size}(k) : \mathcal{K} \to \mathbb{N}$ be the total number of sessions that are associated with the stream k.

$$\mathrm{size}(k) = \sum_{\substack{k' \in \mathcal{K} \, : \\ \mathrm{og}(k') = \mathrm{og}(k)}} |\mathcal{S}^{k'}|. \tag{4.2}$$

Let $P_{\mathcal{R}} : \mathcal{R} \to [0, (|R| - 1)]$ be a total ordering on $\mathcal{R}$ that uses the capacity of the rooms.

$$\begin{aligned} &\forall r, r' \in \mathcal{R}, \\ &\ r <_{P_{\mathcal{R}}} r' \ \text{ if } \ \mathrm{Cap}(r) > \mathrm{Cap}(r') \\ &\ r <_{P_{\mathcal{R}}} r' \ \text{ if } \ \mathrm{Cap}(r) = \mathrm{Cap}(r') \text{ and } r <_{I_{\mathcal{R}}} r'. \end{aligned}$$

Let $\mathcal{S}_j$ be the sessions occurring in the time block j. They are ordered with $I_{\mathcal{S}_j} : \mathcal{S}_j \to [0, (|\mathcal{S}_j| - 1)]$ using their stream size.

$$\begin{aligned} &\forall (k, j), (k', j) \in \mathcal{S}_j \\ &(k, j) <_{I_{\mathcal{S}_j}} (k', j) \text{ if } \ \mathrm{size}(k) > \mathrm{size}(k'), \\ &\ \text{or if } \ \mathrm{size}(k) = \mathrm{size}(k') \text{ and } k <_{I_{\mathcal{K}}} k'. \end{aligned}$$

The bonus $c(s, r)$ is designed so that the worst choice of room for the session s gives a bonus equal to one and the best choice gives a bonus equal to 2^7. It follows that

$$2^{7 + b(|\mathcal{R}| - 1)} = 1 \iff b = \frac{-7}{(|\mathcal{R}| - 1).} \tag{4.3}$$

and

$$c(s, r) = 2^{7 - \frac{7}{|\mathcal{R}|} |P_{\mathcal{R}}(r) - I_{\mathcal{S}_j}(s)|} \tag{4.4}$$

In short, since $P_{\mathcal{R}}$ order the rooms non-increasingly by their capacities, and orders the sessions in S_j non-increasingly by their stream sessions' size, then the bonus $c(s, r)$ favors sessions and rooms with identical positions. This is done so that a large room will host sessions of large streams. Figure 4.1 gives an ordering example.

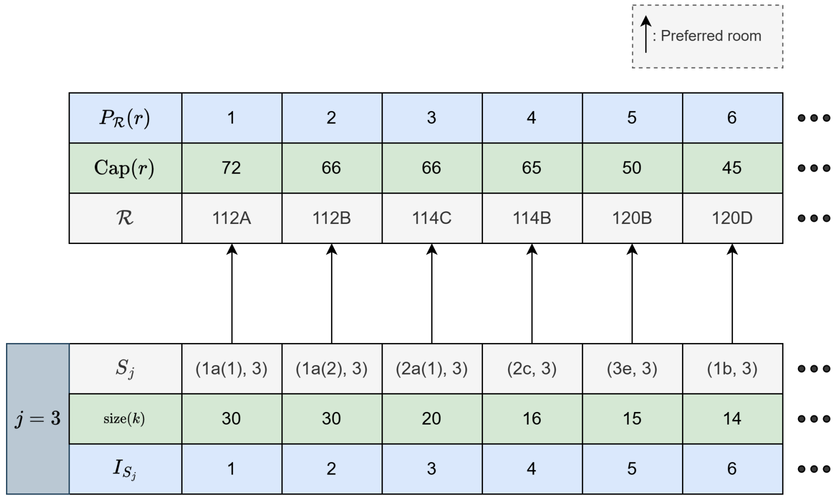


Figure 4.1: Fictive sessions and rooms with their associated ordering and preferred assignments.

4.2 Room assignment model

Definition 4.1 (Ordered Set Without Holes)**.** A subset $A \subseteq \mathcal{S}^k$ is without holes if

$$\forall a, b \in A,\ \nexists\ c \in \mathcal{S}^k \setminus A\ :\ a < c < b$$

Let $\mathcal{P}(3, \mathcal{S}_k)$ be the set of subset of $\mathcal{S}^k$ of size 3 which are ordered by ascending order of $j \in \mathcal{J}$ and without holes.

Given $k \in \mathcal{K}$, $p \in \mathcal{P}(3, \mathcal{S}_k)$ and $r \in R$. The variables $y_{k,p,r}$ are defined as follow.

$$y_{k,p,r} := \begin{cases} 1 & \text{if the three sessions of } k \text{ in } p \\ & \text{are assigned to room } r, \\ 0 & \text{otherwise.} \end{cases} \tag{4.5}$$

The room assignment problem is given in (4.6).

$$\max_{\substack{x \in \{0,1\}^{|\mathcal{S}| \times |\mathcal{R}|} \\ y \in \{0,1\}^{|\mathcal{K}| \times |\mathcal{J}| \times |R|}}} \quad \sum_{r \in \mathcal{R}} \sum_{s \in \mathcal{S}} c(s, r) x_{s,r} + \alpha \sum_{k \in \mathcal{K}} \sum_{p \in \mathcal{P}(3, \mathcal{S}^k)} \sum_{r \in \mathcal{R}} y_{k,p,r} \tag{4.6}$$

subject to

$$\sum_{s \in p} x_{s,r} \geq 3 y_{k,p,r},\ \forall\ k \in \mathcal{K},\ \forall\ p \in \mathcal{P}(3, \mathcal{S}^k),\ \forall\ r \in R, \tag{4.7}$$

consecutive sessions in the same stream should ideally be in the same room.

$$\sum_{r \in R} x_{s,r} = 1\ \forall\ s \in \mathcal{S}, \tag{4.8}$$

each session is assigned to one room.

$$\sum_{s \in \mathcal{S}_j} x_{s,r} \leq 1\ \forall\ r \in R, \tag{4.9}$$

each room has at most one session in time block j,

where $\alpha \in \mathbb{R}_+$.

Therefore, the objective of this model is to have the sessions of a stream stay in the same rooms as much as possible, and to assign, as much as possible, the biggest rooms to the biggest streams.

5 Results and discussion

Some results on the talks' assignment and the room assignment problems for an initial optimization of ISMP 2024 are presented in this section. These results do not reflect the real schedule of ISMP 2024, as around seventeen reoptimizations were conducted. Keynotes and special sessions were also removed from these results.

The room assignment problem 4.6 uses $\alpha = 200$. The $c(s, r)$ bonus scheme used is the one in (4.4) where $b = -\frac{7}{31}$ is rounded to -0.2.

Gurobi version 11.0.0 was used to solve all of the three problems, the talks assignment, the reoptimization and the rooms assignment problems using a 12th Gen Intel(R) Core(TM) i7-12700 CPU with 20 threads.

Figure 5.1 shows the configuration of each session at the solution of the talks' assignment problem given by Gurobi. For example, $(3, 1)$ represents a session of size four where a group of talks of size 3 was assigned with a group of talks of size 1. The "f" postfix indicates that the group is pre-assigned. Figure 5.2 shows the number of keywords in common for the groups which were not pre-assigned as the common keywords of invited sessions are not relevant. Over the 1056 invited talks, 378 talks had 2 keywords, 321 had 3, 212 had 4 and 145 had 5. Over the contributed talks, 50 talks had 2 keywords, 75 had 3, 44 had 4 and 40 had 5.

The Gurobi Optimizer was given a budget of 3,000 seconds, resulting in an optimality gap of 0.2351%. The original size of the problem had 32,111 rows, 1,927,811 columns and 8,366,329 nonzeros. Furthermore, the number of talks to assign is 1,265 with 198 talks that have at least one preferred stream selected. The number of talks which are assigned to one of their preferred stream is 198.

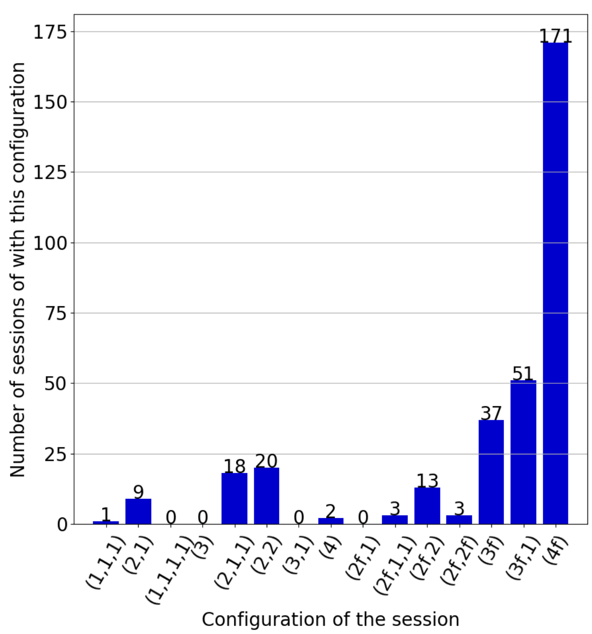


Figure 5.1: Configurations Distribution at the Solution.

The number of parallel sessions taking place for each given time block is given in Figure 5.3.

The solution of the room assignment problem is solved in 20.89 seconds with a gap of 0.0039%. The model has 11,240 rows, 23,456 columns and 58,400 nonzeros. The number of rooms used for each stream is usually one or two, with only two streams using four rooms and two streams using three rooms. Popular streams such as *Nonlinear Optimization* are assigned to the rooms with the greatest capacities.

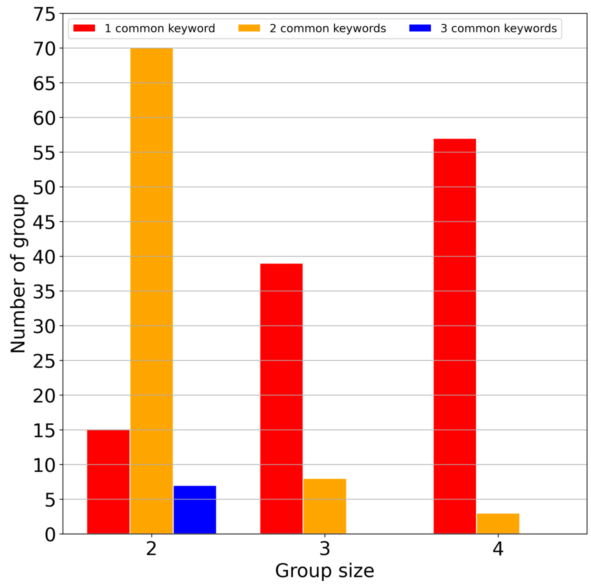


Figure 5.2: Common Keywords at the Solution.

	Monday	Tuesday	Wednesday	Thursday	Friday
8h30-10h30	*Opening Ceremony*	31	32	32	32
14h-16h	32	32	32	32	32
16h20-17h50	10	12	8	11	*Farewell Reception*

Figure 5.3: The number of parallel sessions for each time block.

Seventeen reoptimizations of the schedule were conducted on the real instances of ISMP 2024, and of these, three were published on the website. The solution $\hat{x}$ taken for each reoptimization was the solution of the last published schedule. Overall, around 47 demands were made by participants and organizer that required to move at least one talk. This resulted in 37 talks that were moved to a different day between the first and the second published schedule, while 30 talks were displaced the same day but in a different time block, and 1173 talks did not move. Between the second and the third published schedule, 62 talks were moved to a different day, and 26 to a different time block in the same day. The remaining 1163 talks did not move.

6 Conclusion

This work described the optimization process used to schedule the ISMP 2024 conference. This significantly reduced the amount of manual work required by the scheduling committee. The quality of the solution is difficult to quantify precisely as it relies on the experience of the participants. Due to the hard constraints, it is also possible that some participants were not fully satisfied.

Instances (close to the real models used for ISMP 2024, but anonymized) were submitted to the mixed-integer programming library MIPLIB 2024[2] For those instances, all of the constraints of the participants of ISMP 2024 were included.

Multiple avenues could be explored to improve the quality of the solution and render the process more accurate. First, an automated process could be developed to identify similarities between the abstracts of pairs of talks. This could lead to more precise metrics for the group of talks formation. Furthermore, only cliques were considered to form group of talks. Since this is a strong condition, some information is lost in the process. A more sophisticated process could be designed to investigate groups of talks that are not necessarily cliques.

It is also possible to think about adding soft co-authorship constraints that would try to avoid scheduling pairs of talks in the same time block if they have common author/coauthors.

Finally, a user-friendly interface could be developed, which would allow future ISMP organizing committees to reuse this work.

Appendix - Nomenclature and notations

Given $a < b \in \mathbb{N}$, $[a, b]$ is called a discrete interval and denotes the set $\{a, a+1, \ldots, b\}$. Given $[a, b]$ and a function $f : \mathbb{N} \to \mathbb{N}$, define

$$\big(f(x)\big)_{x\in[a,b]} := \big(f(a), f(a+1), \ldots, f(b)\big). \tag{A.1}$$

Let $k \in \mathbb{N}^*$ and let S be an arbitrary set whose size exceeds k ($|S| > k$). Then

$$\binom{S}{k} := \{s \subseteq S : |s| = k\} \tag{A.2}$$

represents the set of subsets with cardinality k.

A *talk* is a 30-minute presentation submitted by a participant and accepted by the ISMP committee. It excludes keynotes, plenaries and other special talks. The set of all the talks is denoted by $\mathcal{I}$.

A *group of talks*, or for short, a *group* is a set of at most $U_{\text{talk}} \in \mathbb{N}$ talks, where U_{talk} is constant (4 at ISMP). Note that a singleton is also considered as a group. The set of all groups of talks is denoted:

$$\hat{\mathcal{L}} := \{\ell \subseteq \mathcal{I} : |\ell| \le U_{\text{talk}}\}. \tag{A.3}$$

A subset $\mathcal{L} \subseteq \hat{\mathcal{L}}$ respecting additional aggregating constraints (see Section 2.1) is called an *aggregated set of groups of talks*. Section 2.1 describes their use.

A *stream* is a thematic of mathematical optimization that serves as classification tool so that it provides a structure for the schedule and participants can quickly find talks of interest. The set of all the possible streams is $\mathcal{K}$.

A *time block* is a determined interval of time during the week where talks can take place. Time blocks are non-intersecting and the set of all time blocks is denoted $\mathcal{J}$. For practicality, the set of time blocks for ISMP 2024 is $[2, 14]$ (see Figure 1.1). Let $\mathcal{J}_{\text{short}} \subset \mathcal{J}$ be a set of time blocks that has a capacity of talk of $(U_{\text{talk}} - 1)$. At ISMP, these were the last time blocks of Monday through Thursday and correspond to the time blocks in $\mathcal{J}_{\text{short}} = \{3, 6, 9, 12\}$.

[2] https://stage.gurobi.com/resources/miplib-2024-call-for-submissions/

An *assignation of talks* is a set S of ordered triplets $(\ell, k, j) \in \mathcal{L} \times \mathcal{K} \times \mathcal{J}$ respecting the condition

$$\forall\ i \in \mathcal{I}\ \big(\exists!\ \ell \in \mathcal{L} : i \in \ell \wedge (\ell, k, j) \in S\big). \tag{A.4}$$

This condition ensures that for any talk, there exists a single triplet in S that contains the talk in his group.

For a given assignment S of talks, a *session* is a pair $(k, j) \in \mathcal{K} \times \mathcal{J}$ such that there is at least one group $\ell \in \mathcal{L}$ with $(\ell, k, j) \in S$.

A group of talks ℓ is said to be *pre-assigned* if in any assignment S, the condition

$$(\ell', k, j) \in S : i \in \ell \wedge i \in \ell' \implies \ell \subseteq \ell' \tag{A.5}$$

is respected.

A talk is said to be *pre-assigned* to a group if it is in a pre-assigned group of talks.

A talk $i \in \mathcal{I}$ is *pre-assigned to a stream* $k \in \mathcal{K}$ if

$$(\ell, k', j) \in S : i \in \ell \implies (k' = k). \tag{A.6}$$

The set of talks $\mathcal{I}$ is partitioned into *invited* and *contributed* talks. Invited talks are either pre-assigned to a group or pre-assigned to a stream.

A participant is said to be involved in a single talk if -i) he has submitted only one contributed talk and is not part of any pre-assigned group, or (exclusive) -ii) if he chairs and/or speaks in a single pre-assigned group. Otherwise, he is said to be *involved* in multiple talks. The set of all participants involved in multiple talks is denoted $\mathcal{A}$.

The set of groups with a talk involving participant $a \in \mathcal{A}$ is denoted $\mathcal{L}_a$. The set of groups of talks containing $\ell \in \mathcal{L}$ as a subset is denoted $\mathcal{L}_\ell$. The set of groups of talks containing the talk $i \in \mathcal{I}$ is denoted $\mathcal{L}_i$. The set of streams available for the group of talks $\ell \in \mathcal{L}$ is denoted $\mathcal{K}_\ell$.

The set of *rooms* is denoted $\mathcal{R}$, and the maximal capacity of room $r \in \mathcal{R}$ is $\mathrm{Cap}(r) \in \mathbb{N}^*$.

For a given assignation of talks S, and a session (k, j), let $\mathrm{config}(k, j)$ be the multiset

$$\mathrm{config}(k, j) := \big\{|\ell| : (\ell, k, j) \in S\big\}. \tag{A.7}$$

This gives the configuration of the session by the size of its composing cliques. For example, a session (k, j) with two group of presentations $\ell', \ell \in \mathcal{L}$ of sizes 2 and 2 would have $\mathrm{config}(k, j) = \{2, 2\}$.

Finally, a total order I on a set S is an injection $I : S \to \mathbb{N}$ such that any element in S has an image in $\mathbb{N}$. For any $s_1, s_2 \in S$ then $s_1 <_I s_2$ is short for $I(s_1) < I(s_2)$ and $s_1 >_I s_2$ is short for $I(s_1) > I(s_2)$. The total order $I_\mathcal{J}$ is used to order the time blocks by temporal precedence. If the time block j_1 is before j_2 during the week, then $j_1 <_{I_\mathcal{J}} j_2$. Given the total order $I_\mathcal{J}$, then let $j_{\text{last}} \in \mathcal{J}$ be the time block such that $j_{\text{last}} >_{I_\mathcal{J}} j,\ \forall j \in (\mathcal{J} \setminus j_{\text{last}})$. Another total order $I_\mathcal{K}$ is set to be arbitrary and is used in Section 4. Finally, $I_\mathcal{R}$ is an arbitrary total ordering of the rooms $\mathcal{R}$ (see Section 4 for its use).

References

[1] T. Bulhões, R. Correia, and A. Subramanian. Conference scheduling: A clustering-based approach. European Journal of Operational Research, 297(1):15–26, 2022.

[2] R. Correia, T. Bulhões A. Subramanian, P. Huachi, and V. Penna. Scheduling the Brazilian OR conference. Journal of the Operational Research Society, 73(7):1487–1498, 2022.

[3] M.G. Nicholls. A small-to-medium-sized conference scheduling heuristic incorporating presenter and limited attendee preferences. Journal of the Operational Research Society, 58(3):301–308, 2007.

[4] Y. Pylyavskyy, P. Jacko, and A. Kheiri. A generic approach to conference scheduling with integer programming. European Journal of Operational Research, 317(2):487–499, 2024.

[5] N. Rezaeinia, J.C. Góez, and M. Guajardo. Scheduling conferences using data on attendees' preferences. Journal of the Operational Research Society, 75(11):2253–2266, 2024.

[6] A. Tesch. Optimization of Large-Scale Conference Schedules. Bachelor's Thesis, 2013.

[7] B. Vangerven, A.M.C. Ficker, D.R. Goossens, W. Passchyn, F.C.R. Spieksma, and G.J. Woeginger. Conference scheduling – a personalized approach. Omega, 81:38–47, 2017.