

Optimization of Large-Scale Conference Schedules

Bachelorarbeit von
Alexander Tesch

Erstgutachter: Prof. Dr. R. Borndörfer
Zweitgutachter: Prof. Dr. Dr. h.c. mult. M. Grötschel

vorgelegt am
Fachbereich Mathematik der
Technischen Universität Berlin



Juli 2013

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Die selbständige und eigenständige Anfertigung versichert an Eides statt:

Berlin, den 16. Juli 2013

Unterschrift:

Contents

1	Introduction	4
2	The Conference Scheduling Problem	4
2.1	Previous Work	8
2.2	Notation	9
3	Problem Decomposition	9
4	Combining Talks to Sessions	11
4.1	Clique Partitioning approach	11
4.2	Capacitated Facility Location approach	14
5	Time Assignment	17
5.1	Mathematical Optimization Model	18
5.1.1	Core model - Modeling an evenly distributed schedule	18
5.1.2	Room capacity restrictions	23
5.1.3	Capacity labeling	26
5.1.4	Lower bound constraints	30
5.1.5	Chairman constraints	31
5.1.6	Individual scheduling requests	32
5.1.7	Even attractivity distribution	32
5.2	Robustness of capacity compliance	33
6	Stream Assignment	40
6.1	Mathematical Optimization Model	41
6.2	Computational complexity of the stream assignment	43
6.3	Cluster swap heuristic	43
7	Room assignment	45
7.1	Room distances	46
7.2	Similarity values	46
7.2.1	Cluster similarity	46
7.2.2	Stream similarity	48
7.3	The room assignment problem	48
7.3.1	Room assignment problem and MIP formulation	48
7.3.2	Previous work and related problems	51
7.3.3	Improving the computation time of the RAP	52
8	Conclusion	54

9	Computational results	55
9.1	Combination of Talks to Sessions	55
9.2	Capacity labels	56
9.3	ISMP problem instance	58
9.3.1	Instance 1: $n = 500$	59
9.3.2	Instance 2: $n = 750$	62
9.3.3	Instance 3: $n = 1000$	65
9.3.4	Instance 4: $n = 1250$	68
9.3.5	Instance 5: $n = 1500$	71
9.3.6	Instance 6: $n = 1750$	74
9.3.7	Instance 7: $n = 2000$	77
9.3.8	Instance 8: $n = 2250$	80
9.3.9	Instance 9: $n = 2500$	83

Zusammenfassung

Heutzutage gibt es Konferenzen zu fast jeden möglichen Themenbereich. Dabei unterliegt jeder Konferenz ein Veranstaltungsplan, der alle Aktivitäten einer Konferenz räumlich und zeitlich einordnet. Bei wissenschaftlichen Konferenzen bestehen diese Aktivitäten zumeist aus einzelnen Vorträgen, die in sogenannten *Sessions* zusammengefasst sind. Das Organisationskomitee von diesen Konferenzen ist damit beauftragt, jeder Session eine feste Zeit und einen festen Raum innerhalb der Konferenz zuzuordnen. Dabei soll eine Reihe von komplexen Nebenbedingungen eingehalten werden. Insbesondere bei größeren Konferenzen wird das Zuordnungsproblem dann allgemein sehr schwer überschau- und lösbar.

In dieser Arbeit werden wir die Planungsanforderungen erläutern, die bei größeren Konferenzen entstehen. Weiterhin stellen wir einen Lösungsansatz vor, der Veranstaltungspläne mit Hilfe von mathematischer Optimierung berechnet. Dabei wird das allgemeine Planungsproblem in kleinere Teilprobleme zerlegt, die nacheinander mittels gemischt-ganzzahliger Programmierung modelliert und gelöst werden.

Der Ausgangspunkt unserer Ergebnisse war die Erstellung eines Veranstaltungsplans für das *International Symposium on Mathematical Programming 2012 (ISMP)* an der *TU-Berlin*. Die ISMP ist einer der größten Konferenzen im Bereich der angewandten Mathematik und Optimierung. Es wurden 1740 Vorträge zu 595 Sessions zusammengefasst, die über eine Woche in 40 parallelen Räumen stattfanden. Das Veranstaltungsprogramm der ISMP 2012 wurde mit Hilfe der hier vorgestellten Modelle erstellt¹.

¹siehe SPIEGEL Online-Artikel: 'Das optimieren wir jetzt mal' [25] vom 23.08.2012

1 Introduction

Scientific conferences play an important role in almost all areas of today's science and research. They offer a wide platform for international researchers and developers for communication and scientific exchange of current problems, experiences and knowledge in nearly all fields of research. Such conferences usually consist of talks about different topics that are given by authorized speakers. The planning process of a conference includes the creation of a scientific program schedule, that means a local and temporal allocation of all available talks.

As the number of talks can grow into thousands for large-scale conferences, the conference scheduling becomes a complex challenge. Furthermore there exist a great bandwidth on various and individual requirements on the allocation of the talks. In this thesis we discuss the major problems that occur during the development of a conference schedule and provide a framework how proper conference schedules can be generated computationally. For that, we use *Mixed-Integer-Programming* techniques to find schedules that respect the difficult range of additional constraints.

The basis of this thesis was the development of an optimized conference schedule of the *International Symposium on Mathematical Programming 2012* (ISMP 2012) in Berlin, Germany. The ISMP is one of the largest conferences in the field of applied mathematics and mathematical optimization and constitutes a great attraction for international mathematicians and researchers. We will present our results based on the problem instance that we used for the ISMP 2012.

2 The Conference Scheduling Problem

In the following we will describe the major problems and requirements that arise when scheduling large conferences.

Combination of talks to sessions

In the beginning, the organizers are mostly confronted with a huge number of submitted talks. At most conferences, each talk must be given within a *session*. A session includes a couple of talks that are given successively within the session, without any longer breaks in general. At the ISMP 2012, each session consisted of at most three talks, where each talk was limited to 30 minutes speaking time, therefore each session had a total duration of 90 minutes.

The first problem the organizers have to face is to find a suitable aggregation of all submitted talks to sessions. Some sessions may already be formed by the speakers themselves, but a large amount of talks must still be combined in a proper way. It is intended to have talks of similar content in the same session. In addition, sessions that are only partially filled with talks should be avoided, since the attendees of a session should be motivated to attend for the entire session length.

Session assignment

Once all talks have been assigned to sessions, the next planning step is to create a sci-

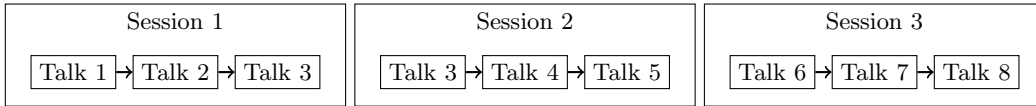


Figure 1: Creating sessions

entific program schedule. We assume there are fixed time slots when sessions can take place. Moreover, each time slot is supposed to have a constant duration that is equal to the length of one session. At larger conferences, the number of sessions normally exceeds the number of available time slots, such that sessions have to take place in parallel. Therefore, a minimum number of $\lceil \frac{n}{T} \rceil$ rooms is needed to allocate all sessions, where n denotes the number of sessions and T the number of time slots. In our considerations the creation of a scientific program schedule solely deals with the allocation of sessions to time slots and rooms. But in addition a huge variety of constraints must be satisfied.

In the following we will look at a typical instance of a conference schedule and describe the related problems by looking at the schedule from different views.

Assume we have given a fixed set of time slots $\{T1, T2, \dots\}$, as well as a set of rooms $\{A, B, \dots\}$. Furthermore, we suppose that we have given sessions $\{1, \dots, n\}$. Each session must be assigned to exactly one time and room slot and each time-/room slot can be occupied by at most one session.

R/T	T1	T2	T3	T4	T5	T6
room A	4	20	9		21	16
room B		6	1	25	23	15
room C	5	17	2	8	20	3
room D	7	18	24	19		
room E	12	10	13	11	22	

Figure 2: Session view

The figure shows an exemplary schedule in the *session view*, where the numbers correspond to a particular session. Such a schedule can be computed quite easily, since it refers to a usual assignment problem. In practice, the amount of additional constraints make this problem more complex.

Cluster scheduling

Even before the paper submitting phase, the organizers normally offer a set of *clusters* for the conference. A cluster can be seen as generalized topic that each talk belongs to. We expect that each talk is part of exactly one cluster. The classification of talks to clusters is usually done by the speakers themselves during the abstract submission. Therefore, we assume that this information is given in advance. It is further required that all talks in the same session share the same cluster.

To extend the previous example, we additionally regard a set of clusters $\{1, 2, \dots\}$.

Each session belongs to exactly one of these clusters. The next figure shows the *cluster view* of the schedule, where the numbers correspond to the cluster of each scheduled session.

R/T	T1	T2	T3	T4	T5	T6
room A	1	1	1		1	1
room B		4	4	3	3	3
room C	2	2	6	6	6	6
room D	5	5	5	5		
room E	6	6	6	6	7	

Figure 3: Cluster view

Obviously, the clusters could have different numbers of sessions. Another requirement on the schedule is that each cluster is supposed to be equally available at each time slot, as shown in the figure. Then, participants will have a higher chance to attend sessions of any cluster of the conference. In addition, it is recommended that the sessions of one cluster take place in the same room. This provides a better orientation and recognition value in an unknown environment for international visitors.

Room capacity restrictions

Sessions of different clusters will generally deal with different topics, so the interest in these sessions may deviate. There might also be talks that are given by more famous speakers, or talks that present new achievements in some field of research such that there exists a high general interest in single sessions or talks respectively. Consequently it is obvious that there will occur deviations in the number of attendees of each session. Since every room has a limited number of seats, it is in the responsibility of the organizers to ensure that more attractive sessions will be planned to larger rooms. Overcrowded lecture rooms are one of the major problems, especially at large conferences. For that, we suppose there are constant capacity demands for every session (more on this later). The final schedule has to comply with the room capacities of every assignment. If we extend the previous example, the figure below shows the schedule in the *capacity view*.

R/T	capacity	T1	T2	T3	T4	T5	T6
room A	200	92	22	75		53	34
room B	55		55	48	12	25	42
room C	120	70	118	23	90	48	35
room D	30	29	30	28	25		
room E	45	43	26	11	44	33	

Figure 4: Capacity view

The numbers correspond to the capacity demands of every session, or to the number of attendees respectively. Note that the total number of attendees can vary for each time

slot, as not all participants attend for the entire length of the conference. In particular, there are time slots that are attended by more people. From our observations, the peak is usually in the early mid of the conference.

In our approach we focus on the mere compliance with room capacities. That means we do not depreciate, if a small session is scheduled into a proportional larger room.

Even attractiveness distribution

As there are more or less attractive sessions at the conference, it is also not desirable to have too many attractive sessions scheduled in parallel. That is because each time slot should be equally attractive, as well as there must not appear too many attractive sessions at the same time. If we would have given a value of attractiveness of each session, then the schedule should respect a roughly even attractiveness distribution over all time slots, see the figure below.

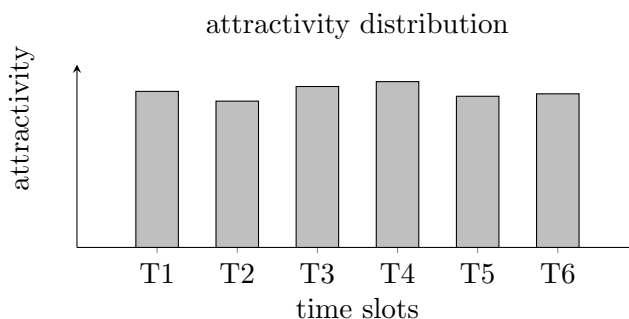


Figure 5: Attractivity view

The *attractivity view* shows the total session attractiveness at each time slot. In the best case, the values are equal at each time slot.

Room distances

Apart from the room capacity, the assignment to rooms includes another difficulty. Whenever a talk has been given within a session, there is a short break of approximately five minutes to answer questions about the content of the talk. Many participants use these breaks for room changes to other talks. Leaving attendees will have to overcome a certain distance to the room of the next talk. As it is hard to estimate which talk each attendee is changing to, we assume that it will be a talk of similar content. For this reason, the distance between similar parallel sessions is supposed to be as small as possible. Of course, this property should hold for all time slots.

The figure below shows an exemplary room map with room changing relations (blue) between similar sessions at some time slot. On the left figure there is shown a proper room assignment, while the right figure illustrates an improper room assignment.

In particular, large-scale conferences include room at several floors and in different buildings, where distances have to be considered. There it is very important that distances are kept short.

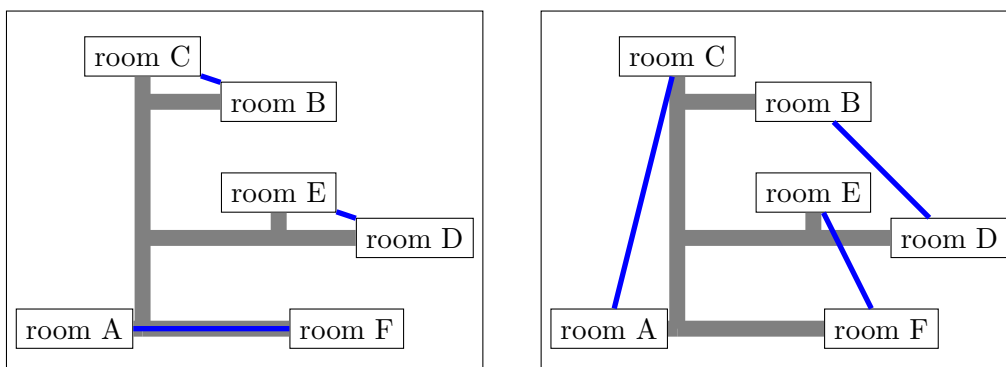


Figure 6: Short (left) vs. long (right) room changing distances at some fixed time slot

All in all, these are the main points that must be respected by the final schedule. There are even more constraints, like individual scheduling requests of single sessions. For instance, one speaker is able to attend only the first day of the conference, then it has to be constrained in the schedule. Moreover, some persons have to attend to several sessions at the conference, then it must be ensured that these sessions will not overlap at the same time. More constraints of this type are presented in the upcoming sections.

We conclude that conference scheduling is a very complex issue. A scheduling 'by hand' would be a very time consuming and nearly impossible challenge for large conferences where thousands of different talks and persons are involved. Therefore, mathematical optimization methods find a great application, whenever conference organizers have to deal with the complexity of such problems. In the following, we will present an approach to compute conference schedules that respect the mentioned conditions. As mentioned, the presented optimization models were developed to optimize the ISMP 2012 conference schedule. The problem instance of the ISMP 2012 included 1740 talks which were combined to 595 sessions. These had to be scheduled to five days, each having three time slots for the allocation of sessions. All sessions have been planned, using the minimum number of 40 rooms for parallel sessions. Until now, this was the largest ISMP that has ever taken place.

To provide a better understanding of the underlying combinatorial structures, the optimization models were extended and improved after the ISMP 2012. We will refer to the extended results in this thesis.

2.1 Previous Work

The problem of optimizing conference schedules was barely studied in the past. For a closer insight into that problem, we used an existing master thesis [21] that dealt with an equal problem of conference scheduling. For further references, we consulted papers of timetabling models, that mostly dealt with the optimization of university and course

schedules, see [1], [2], [3] and [8]. As our view on the problem is quite specific for itself, we mostly searched for the underlying combinatorial problems, for which we then used further literature.

2.2 Notation

We denote the set of talks at the conference by K . Each session s can be seen as sequence of successive talks $s = (k_1, \dots, k_m)$ ($m \leq 3$ at ISMP). The set of all sessions is denoted as S . Furthermore each session takes place at a certain time slot and in a certain room. The set of all time slots at the conference is given by T and R is the set of available rooms. In addition, each talk and each session belongs to a specific cluster. Then C denotes the set of clusters and we denote the cluster of some talk k by c_k and the cluster of some session s by c_s . Similarly, the sessions that belong to cluster c is defined are given by the set S_c .

In the upcoming sections, we will present combinatorial problems that have a representation via graphs. We usually define an undirected graph as $G = (V, E)$, where V denotes the set of vertices and E the set of edges between the vertices. We normally denote the set of adjacent vertices of a given vertex v by $\delta(v) = \{u \mid (u, v) \in E\}$. This notation is used for different problems in this thesis. The same holds for the Mixed-Integer-Programs, where the main variables are usually described by x, y and z . We emphasize that all models are regarded separately in terms of notation, if not mentioned otherwise. But it should be clear from the context which model is currently used.

3 Problem Decomposition

Our first testings revealed quickly that the simultaneous assignment of sessions to time- and room slots is very hard to solve and too time consuming for practical purposes. As the requirements on a large conference schedule are subject to constant changes, we decided that it would be more helpful to use an optimization model that computes solutions in a shorter time span. Therefore, we use a problem decomposition approach, where the entire problem is subdivided into smaller subproblems that are easier and faster to solve. Our decomposition includes the following subproblems:

- (1) *Combination of Talks to Sessions*
- (2) *Time Assignment*
- (3) *Stream Assignment*
- (4) *Room Assignment*

The subproblems are solved successively, where each step uses the solutions of its predecessors. The first optimization step *combines all talks to sessions* (1) such that similar talks will share the same session. The obtained set of sessions is used for the

further procedure. In the *time assignment* (2), each session gets assigned to a particular time slot of the conference. It is optimized to evenly distribute the sessions of each cluster over the entire length of the conference. Afterwards, all sessions are arranged to *streams* in the *stream assignment* (3). A stream is a sequence of sessions (s_1, s_2, \dots, s_T) that successively take place in the same room. It is desired that rooms are occupied by sessions of mainly one cluster for the whole conference. In the last step, each stream is allocated to a particular room in the *room assignment* (4). The distances between streams, that contain similar sessions, is to be minimized. All subproblems are modeled as *Mixed-Integer-Programs*. An illustration on the entire optimization process is given in the figure below.

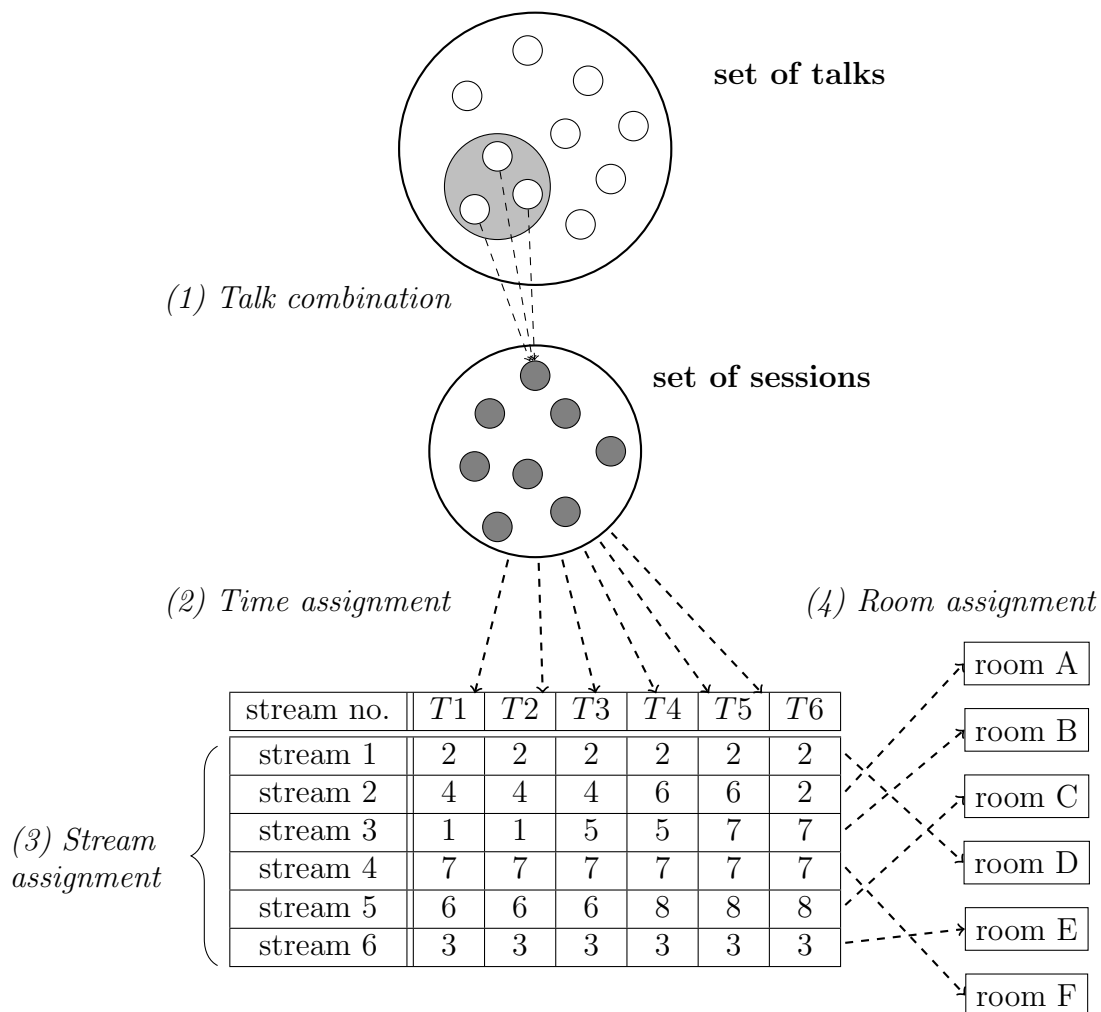


Figure 7: Sequence of optimization models for the conference scheduling problem

4 Combining Talks to Sessions

After the paper-submission phase, the conference organizers have to combine the submitted talks to sessions. The problem is to find a suitable combination of talks to sessions, where talks with overlapping content should be given within the same session. In particular, only talks of the same cluster will be combined. Furthermore sessions are supposed to be completely filled with talks, since partially filled sessions are rather unattractive to attend.

In the following we present two Mixed-Integer-Programming approaches that make use of different combinatorial problems. Both approaches yield an aggregation of all talks to sessions. The total similarity between talks in one session is maximized, as well as the total idle time in the created sessions is minimized. From our point of view, the first model yields a better representation of the actual problem, but is more difficult to solve. On the contrary, the second model is much easier to solve but just partially reflects the underlying problem. In addition, the needed problem data is more difficult to obtain and to characterize.

We have to mention, that the models of this section have not been used for the ISMP 2012. But in retrospect, they find a great application, when the conference planning is considered as a whole. For this reason, our computational experience was made with randomized data.

In the following we will consider the generalized case, where talks may have arbitrary lengths. Therefore, we denote the constant duration of all sessions by D and the length of talk k by d_k . In particular, for the ISMP instance it holds $d_k = 30$ for all $k \in K$ and $D = 90$ (in minutes), what we could also simplify to $D = 3$ and $d_k = 1$ for all $k \in K$.

4.1 Clique Partitioning approach

For this problem we have given an undirected graph where each talk $k \in K$ represents a vertex. Every talk is supposed to have a similarity relation to all other talks of the same cluster $c \in C$. These relations are modeled as undirected edges from an edge set $E = \{(k, l) \mid k, l \in K, k < l, c_k = c_l\}$. Every edge (k, l) has an edge weight $\sigma_{kl} \in \mathbb{R}$ that defines the similarity between talks k and l . Its value indicates the overlap in the content of the talks. The higher the value of σ_{kl} the higher is the overlap. The induced graph is given by $G = (K, E)$.

All talks within the same session have a similarity relation to each other. Hence, we will regard each session as complete subgraph of G , a clique. The problem is to find a partition of K into disjoint subsets K_1, \dots, K_m , such that for all $i \in \{1, \dots, m\}$ each K_i induces a session or a clique respectively. We define the size of a clique, as the number of vertices in the induced subgraph. The total similarity of the sessions, that is the total edge weight of all cliques, is to be maximized. This combinatorial problem is also known as the *Clique Partitioning Problem (CPP)* that was shown to be \mathcal{NP} -hard. It has been studied by Grötschel and Wakabayashi [4][22] who introduced different cutting plane techniques for this problem. Ji and Mitchell [23] considered the problem with minimum clique size requirement and propose a branch-and-cut algorithm.

Our considerations include further restrictions concerning the length of the sessions and talks. For every session K_i the sum of the lengths of all talks in K_i must not exceed the duration of the session, that is $\sum_{k \in K_i} d_k \leq D$. Thus, d_k can be seen as additional vertex weight, where each clique bounds the weight of all contained vertices by D . These knapsack conditions within a clique partitioning problem has been studied by Mehrotra and Trick [24].

Furthermore, our model penalizes the unused capacity of each clique, as this corresponds to the idle time in each session which is to be minimized. In addition, we have another requirement on the maximum number of cliques in the clique partitioning, since the number of sessions is bounded by the number of available time- and room slots.

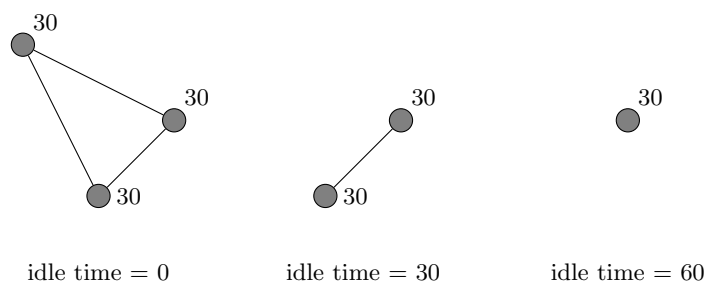


Figure 8: Three possible sessions (cliques) with duration of $D = 90$ that contain talks of length $d_k = 30$ and the induced idle time

In the following we will present an integrated Mixed-Integer-Program that solves the problem with respect to the mentioned constraints.

Mixed Integer Program

For every edge $(k, l) \in E$ we introduce binary variables x_{kl} , that take on the value 1, if talks k and l appear in the same session. The number of possible sessions is bounded by the number of available assignment slots, that is $|T| \cdot |R|$. Consequently, the number of cliques in the graph must be restricted to this amount. From the modeling view, restricting the number of cliques is a non-trivial matter, so we introduce further variables. Let $W = \lceil \frac{D}{\min_{k \in K} d_k} \rceil$, then $B = \{1, \dots, W\}$ denotes the set of possible clique sizes. Furthermore, let z_{kb} denote the binary variable which takes on the value 1, if talk k is contained in a session of size $b \in B$, and 0 otherwise. Moreover, let $y_{kb} \geq 0$ denote the amount of idle time that is induced by talk k , if it is contained in a session of size b . A more detailed explanation on this variable is given later. At large conferences, it is usual that sessions can also be created by the speakers themselves. Therefore let K_+ denote the set of talk pairs of the same cluster, which are supposed to appear in the same session. The Mixed Integer Program reads as follows:

$$\begin{aligned}
\max \quad & \lambda \sum_{(k,l) \in E} \sigma_{kl} x_{kl} - (1 - \lambda) \sum_{k \in K} \sum_{b \in B} y_{kb} \\
\text{s.t.} \quad & x_{jk} + x_{kl} - x_{jl} \leq 1 \\
& x_{jk} - x_{kl} + x_{jl} \leq 1 \quad \forall (j, k), (k, l), (j, l) \in E \quad (1) \\
& -x_{jk} + x_{kl} + x_{jl} \leq 1 \\
& \sum_{b \in B} z_{kb} = 1 \quad \forall k \in K \quad (2) \\
& y_{kb} \leq D \cdot z_{kb} \quad \forall k \in K, b \in B \quad (3) \\
& \sum_{l \in \delta(k)} x_{kl} = \sum_{b \in B} (b - 1) \cdot z_{kb} \quad \forall k \in K \quad (4) \\
& d_k + \sum_{l \in \delta(k)} d_l x_{kl} + \sum_{b \in B} b \cdot y_{kb} = D \quad \forall k \in K \quad (5) \\
& \sum_{k \in K} \sum_{b \in B} \frac{1}{b} z_{kb} \leq |T| \cdot |R| \quad (6) \\
& x_{kl} = 1 \quad \forall (k, l) \in K_+ \quad (7) \\
& x_{kl} \in \{0, 1\} \quad \forall (k, l) \in E \\
& y_{kb} \geq 0 \quad \forall k \in K, b \in B \\
& z_{kb} \in \{0, 1\} \quad \forall k \in K, b \in B
\end{aligned}$$

The similarities between talks in the same sessions are maximized in the objective function, as well as the total idle time in all sessions is minimized. The coefficient $\lambda \in [0, 1]$ weights the importance between the two objectives. Constraints (1) are the *triangle inequalities*, which indicate that whenever two edges in a triangle (j, k, l) are selected, then also the third edge must be selected. Together with the integrality condition $x_{kl} \in \{0, 1\}$ for all $(k, l) \in E$, the integral points that satisfy these inequalities yield a complete description of all clique partitionings in G , see [4]. Inequalities (2) say that each talk must be assigned to a session of a particular size. If some talk k is not contained in a session of size b , then it induces no idle time y_{kb} there (3). Constraints (4) state that the number of adjacent edges of some talk k is exactly the size of the clique of k minus one. Further, the sum of the lengths of all talks in a session must not exceed the maximal duration D of a session, what is modeled by (5). The term $\sum_{b \in B} b \cdot y_{kb}$ is used to count the correct amount of idle time in the objective function, because every clique member shares the idle time. Inequality (6) says that the number of sessions must not exceed the maximum number of sessions, what is $|T| \cdot |R|$. In addition x_{kl} must be fixed to one for all $(k, l) \in K_+$ (7).

Reduction of the problem size

In practical implementation, the triangle inequalities (1) have a huge impact on the complexity of this problem, as the number of constraints grows rapidly, even for smaller instances. Therefore, it was hard to find proper solutions for our problem instances. For this reason, we did a reduction approach to reduce the number of constraints to a manageable amount. Our approach was to eliminate 'redundant' edges. In our model we regard an edge as redundant, if it does not satisfy a certain value σ of similarity. More precise, all edges $(k, l) \in E$ with $\sigma_{kl} < \sigma$ are eliminated. This seems useful, as edges with a lower similarity value are rather unattractive to choose. Therefore, we define the reduced edge set as $E' = \{(k, l) \in E \mid \sigma_{kl} \geq \sigma\} \subseteq E$ and apply the model to the sparse graph $G' = (K, E')$. With this reduction approach, the completeness property of the subgraphs is lost in general due to the elimination, so the triangle inequalities do not give a full representation of the clique partitionings in G' anymore. Thus, we have to replace the triangle inequalities (1) by the following set of constraints to regain a full description.

$$\begin{aligned} x_{jk} + x_{kl} - x_{jl} &\leq 1 && \forall (j, k), (k, l), (j, l) \in E' \\ x_{jk} - x_{kl} + x_{jl} &\leq 1 && \forall (j, k), (k, l), (j, l) \in E' \\ -x_{jk} + x_{kl} + x_{jl} &\leq 1 && \forall (j, k), (k, l), (j, l) \in E' \end{aligned} \quad (1')$$

$$\begin{aligned} x_{jk} + x_{kl} &\leq 1 && \forall (j, k), (k, l) \in E' : (j, l) \notin E', j < l \\ x_{jk} + x_{jl} &\leq 1 && \forall (j, k), (j, l) \in E' : (k, l) \notin E', k < l \\ x_{kl} + x_{jl} &\leq 1 && \forall (k, l), (j, l) \in E' : (j, k) \notin E', j < k \end{aligned} \quad (1'')$$

Inequalities (1') are the triangle inequalities, as used before for each triangle in the sparse graph. If one edge of an arbitrary triangle in G is lost due to the elimination, we have to ensure that at most one of the two remaining edges belongs to a clique of the clique partitioning of G' (1''). An equivalent interpretation of these inequalities is to set $x_{kl} = 0$ in (1) for all eliminated edges $(k, l) \notin E'$, where (1') and (1'') are the usual triangle inequalities. The benefit arises from the fact that triangles with two deleted edges induce a redundant triangle inequality.

Computational experience

Since this problem is \mathcal{NP} -hard, it was not surprising that the problem was not solvable with the general formulation. Therefore, we introduced the elimination approach, in order to reduce the huge number of triangle inequalities in the model. This made it possible for us to compute solutions for problem sizes of the ISMP, or even larger. But in general, we can be far from the optimum, so we considered another formulation that is presented in the next section.

4.2 Capacitated Facility Location approach

The main idea of this model approach is to assign each talk $k \in K$ to a session $s \in \bar{S}$ of a preformed session set \bar{S} . Whenever a talk is assigned to a session, then this session

is 'opened'. Here it is important that each session of \bar{S} already belongs to a certain cluster $c \in C$ such that talks are only assigned to sessions of equal cluster. It also ensures that talks of different clusters are not assigned to the same session. Therefore let $\bar{S}_k = \{s \in S \mid c_s = c_k\}$ for all $k \in K$ and $K_s = \{k \in K \mid c_s = c_k\}$ for all $s \in \bar{S}$. Furthermore, each assignment (k, s) of talk k to session s has a similarity value σ_{ks} that indicates the similarity of talk k to session s . The first objective is to maximize the total similarity of the talks to the sessions, while each talk must be assigned to a particular session. Secondly, the occurring idle time of each session should be minimized. In order to respect the maximum possible number of sessions, we have to bound the total number of opened sessions by $|T| \cdot |R|$. As in the previous model, it must be ensured that the total length of all assigned talks does not exceed the duration limit D of each session. In the end, each opened session $s \in \bar{S}$ forms a 'real' session at the conference that contains all assigned talks.

In combinatorial optimization, this problem variant is better known as *n-Single Source Capacity Location Problem (n-SSCFLP)*. There we have given a set of customers and a set of potential facility locations, where each customer must be assigned to exactly one opened facility. The total assignment cost and the total opening cost of all facilities should be minimized. Moreover, the capacity of each facility must be sufficient to serve the demand of all assigned customers. Also, the number of opened facilities is bounded by a constant n . The n-SSCFLP belongs to the class of \mathcal{NP} -hard problems.

If we transform our problem to the n-SSCFLP, each talk k corresponds to a customer that has demand d_k , as well as each session s is associated with a potential facility that has capacity D and opening cost equal to zero. The assignment cost is given by $-\sigma_{ks}$, thus σ_{ks} is the profit to assign customer k to facility s . The bound on the maximum number of opened facilities is $n = |T| \cdot |R|$ in our case.

In contrast to the general formulation of the n-SSCFLP, we also want to minimize the idle time in each session, what can be seen as additional cost for 'unused capacity' of each facility. Additionally, there are talk pairs which have to be scheduled within the same session. We will see that these conditions can be implemented with a common n-SSCFLP formulation. The figure below shows an illustration to this modeling variant.

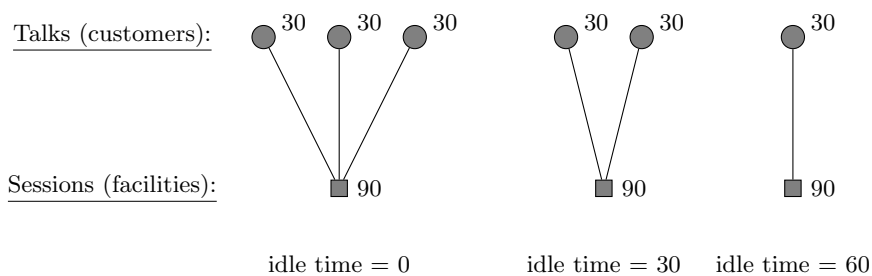


Figure 9: Different assignments of talks with length $d_k = 30$ to sessions with duration of $D = 90$ and the induced idle time (unused capacity)

Mixed Integer Program

For every assignment (k, s) of talk k to session s , there is the binary variable x_{ks} which takes on the value 1, if talk k is assigned to session s , and 0 otherwise. The variable $y_s \geq 0$ denotes the amount of idle time in each session s . Furthermore, the decision variable z_s is 1, if session s is opened, and 0 otherwise. Equally to the previous model, K_+ denotes the set of talk pairs, that should be allocated to the same session. Especially it must hold $c_k = c_l$ for all $(k, l) \in K_+$. The Mixed-Integer-Program states as follows:

$$\max \quad \lambda \sum_{(k,s) \in E} \sigma_{ks} x_{ks} - (1 - \lambda) \sum_{s \in \bar{S}} y_s$$

$$\text{s.t.} \quad \sum_{s \in \bar{S}_k} x_{ks} = 1 \quad \forall k \in K \quad (1)$$

$$\sum_{k \in K_s} d_k \cdot x_{ks} + y_s = D \cdot z_s \quad \forall s \in \bar{S} \quad (2)$$

$$x_{ks} \leq z_s \quad \forall k \in K, s \in \bar{S}_k \quad (3)$$

$$\sum_{s \in \bar{S}} z_s \leq |T| \cdot |R| \quad (4)$$

$$x_{ks} = x_{ls} \quad \forall (k, l) \in K_+, s \in \bar{S}_k \quad (5)$$

$$\begin{aligned} x_{ks} &\in \{0, 1\} & \forall k \in K, s \in \bar{S}_k \\ y_s &\geq 0 & \forall s \in \bar{S} \\ z_s &\in \{0, 1\} & \forall s \in \bar{S} \end{aligned}$$

The objective function maximizes the total similarity of the assignments of talks to sessions, as well as the total idle time in the created sessions is minimized. The coefficient $\lambda \in [0, 1]$ weights the importance between the two objectives. Constraints (1) ensure that every talk is assigned to exactly one session. Inequalities (2) express that the talk lengths may not exceed the duration of each session and its gap is exactly the idle time of the session. Inequalities (3) are not necessary for the formulation, but they yield stronger bounds for linear relaxation of the problem. In (4), the total number of opened sessions must not exceed the number of available session slots, that is $|T| \cdot |R|$. Constraint (5) guarantees, that all talks in K_+ are assigned to the same session.

If we take a closer look to the model, we see that it can be transformed to a common n-SSCFLP. For that, we rearrange inequality (2) and see that we can get rid of the y_s variables:

$$0 \leq y_s = D \cdot z_s - \sum_{k \in K} d_k \cdot x_{ks} \quad (2)$$

$$\sum_{k \in K} d_k \cdot x_{ks} \leq D \cdot z_s \quad \forall s \in \bar{S} \quad (2')$$

Constraints (2) are replaced by (2') which now correspond to the usual capacity requirement of an n-SSCFLP. The definition of y_s in (2) is then inserted into the objective function:

$$\begin{aligned} & \max \sum_{(k,s) \in E} (\lambda \sigma_{ks} + (1 - \lambda) d_k) x_{ks} - \sum_{s \in \bar{S}} (1 - \lambda) D \cdot z_s \\ = & \max \underbrace{\sum_{k \in K} (1 - \lambda) d_k}_{const.} + \sum_{(k,s) \in E} \lambda \sigma_{ks} x_{ks} - \sum_{s \in \bar{S}} (1 - \lambda) D \cdot z_s \\ \Leftrightarrow & \min \sum_{(k,s) \in E} \sigma'_{ks} x_{ks} + \sum_{s \in \bar{S}} f_s z_s \end{aligned}$$

Now, inequalities (1) - (4) correspond directly to the n-SSCFLP formulation with new assignment cost $\sigma'_{ks} = -\lambda \sigma_{ks}$ and facility opening cost $f_s = (1 - \lambda) D$. This has the natural interpretation that the idle time just depends on the number of opened sessions, while the total duration of the talks is constant anyway.

Constraints (5), which expressed that talk pairs $(k, l) \in K_+$ must be assigned to the same session, can also be eliminated by replacing all talks, that must be scheduled into one session, by a new greater talk that has the total duration of all replaced talks.

Computational experience

For our computational results we choose $\bar{S} = K$, that means it is possible to open a session for each talk. As the similarities occur between talks, we used the same similarities, as in the clique partitioning model. In contrast to the previous approach, this problem is properly solvable without reducing the problem size. But in contrast to the CPP, it does not yield a proper formulation of the underlying problem, as some similarity relations (between the customers) are not considered. Therefore it gives just a partial representation of the actual problem.

5 Time Assignment

In this section we will present the next optimization step, the *time assignment*. It yields an assignment of all sessions to time slots. The entire time assignment model includes a huge variety of different constraints. For this reason, we will present a general Mixed-Integer-Programming formulation first, until further classes of constraints are introduced. Based on that, we will give an approach to the *robust optimization* of the schedule, that relates to the uncertain number of attendees of every session and the compliance with room capacities.

5.1 Mathematical Optimization Model

In the optimization model, each session $s \in S$ is assigned to a time slot $t \in T$. This is modeled by decision variables x_{st} , with:

$$x_{st} = \begin{cases} 1, & \text{if session } s \text{ takes place at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

A solution of the time assignment model yields a partition of S into disjoint sets S_1, \dots, S_T with $S_t \cap S_{t'} = \emptyset$ for all pairwise distinct $t, t' \in T$. In that sense, S_t contains all sessions that are scheduled at time slot t . These sets are used as input for the upcoming optimization models. In the next sections we present a comprehensive description of the core-model, which is followed by further classes of constraints.

5.1.1 Core model - Modeling an evenly distributed schedule

Each conference participant has individual preferences concerning the choice of his session. But it is natural to assume that every participant focuses only on a selection of topics that he is interested in. These topics are generalized given by the set of clusters C . As each session belongs to exactly one cluster c , the sessions of a cluster are supposed to be equally available at every time slot of the conference.

That means for every cluster $c \in C$ the set $S_c \subseteq S$, what is the set of sessions that belongs to c , must be evenly distributed over all $t \in T$. For a given cluster c the following illustration shows how S_c should be distributed over the time slots in the best case.

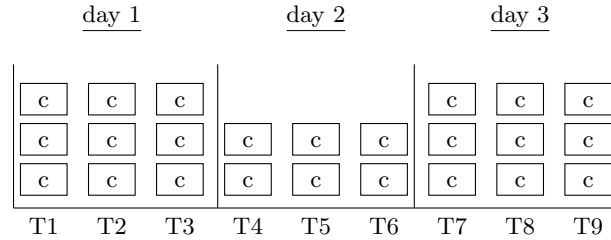


Figure 10: Exemplary time assignment for S_c

The figure shows the distribution of S_c over an exemplary conference with time slots $T = \{T1, \dots, T9\}$. This example shows the best distribution for S_c , in a sense, that will be explained during the further procedure. We see that S_c is evenly distributed over all time slots, so participants, who are interested in cluster c , are able to attend the sessions of c at every time slot. Each day of the conference has exactly three time slots. At day 1 and day 3 there occur three parallel sessions of S_c , while at day 2 there are scheduled two parallel sessions. Time slots with a maximum number of sessions should appear together at entire days (day 1 and day 3). This is desired, because 'cluster gaps' should be avoided during a single day, that means one room is supposed to be occupied

by sessions of one cluster for the whole day. We also want that S_c can be scheduled in the minimum possible number of rooms. With this time distribution, it is possible that S_c occupies the minimum number of three rooms.

In the following we present a Mixed-Integer-Program to obtain such an evenly distributed schedule for arbitrary subsets of sessions $S' \subseteq S$. For that, we need to introduce further definitions and properties of the schedule.

Let b_t denote the number of sessions of a fixed cluster at time slot t . Assume that a participant is interested in this specific cluster only, that means he will attend sessions of this cluster only. Additionally, we have to assume that he is able to attend only one session at each time slot. Let ω denote the number of different session programs s_1, \dots, s_T that our participant can choose for the whole conference.

Claim 1. ω is maximal if and only if $|b_t - b_{t'}| \leq 1$ for all pairs of time slots $t, t' \in T$.

Proof. As it is possible to choose only one session per time slot, the number of different session programs is the product over the number of available sessions at each time slot, except those with $b_t = 0$. Let T' be the set of time slots t with $b_t > 0$, then he can choose $\omega = \prod_{t \in T'} b_t$ different session programs for the conference.

Let ω be maximal and assume that $b_t - b_{t'} > 1$ for two different time slots $t, t' \in T$. Swapping one session from t to t' yields:

$$(b_t - 1)(b_{t'} + 1) \prod_{t \in T' \setminus \{t, t'\}} b_t = (b_t b_{t'} + b_t - b_{t'} - 1) \prod_{t \in T' \setminus \{t, t'\}} b_t > b_t b_{t'} \prod_{t \in T' \setminus \{t, t'\}} b_t = \omega$$

This forms a contradiction on ω being maximal.

To show the other direction, let $|b_t - b_{t'}| \leq 1$ for all $t, t' \in T'$ and assume that ω is not maximal. Any deviation of the distribution that preserves $|b_t - b_{t'}| \leq 1$ yields the same ω . Consequently in a distribution, where ω is maximal, it must hold $b_t - b_{t'} > 1$ for two time slots t and t' . But then we could always find a greater ω , equally to the case before. This contradicts to the fact that ω is maximal in such a distribution. \square

Since we expect many participants to concentrate on a few clusters only, the claim yields a nice property for the schedule. There, the number of different session programs for one cluster is maxed out.

The next definitions are needed to define an evenly distributed schedule, as shown in Figure 10.

Definition 2. Let $S' \subseteq S$ be an arbitrary subset of sessions. We call a time slot t maximal for S' , if $b_{t'} \leq b_t$ for all $t' \in T$.

Definition 3. Let D denote the set of days for the conference. We define T_d as the set of time slots, which belong to day $d \in D$. We say a set of sessions $S' \subseteq S$ is bundled at day d , if all time slots $t \in T_d$ are maximal for S' .

Definition 4. We say a set of sessions $S' \subseteq S$ has an evenly distributed schedule (EDS), if the following properties hold:

- (i) $|b_t - b_{t'}| \leq 1$ for all $t, t' \in T$
- (ii) the number of bundled days is maximal, with respect to (i)

The time assignment of S_c in Figure 10 has an EDS, because two time slots differ by at most one session (i), and the number of bundled days (day 1 and 3) is maximal (ii). In our considerations it is not important which days are bundled, as long as the number of bundled days is maximal.

Next, we introduce a Mixed-Integer-Programming approach, that models an EDS for arbitrary session sets $S' \subseteq S$. For that, we define \mathcal{S} as the set of session-subsets, for which an EDS should be applied. Since every cluster is supposed to be evenly distributed we define $\mathcal{S} = \bigcup_{c \in C} \{S_c\}$, that means the elements of \mathcal{S} are the session sets of each cluster. For simplicity in the further notation we use the index set J to denote all EDS sets with $\mathcal{S} = \bigcup_{j \in J} \{S_j\}$. This will be helpful to describe further EDS sets that are introduced later.

Mixed Integer Program

For each $j \in J$ we define the variable $U_j \geq 0$ that is an upper bound on the number of sessions of $S_j \in \mathcal{S}$ at all time slots. Furthermore let y_{jt} be a binary variable which takes on the value 1, if time slot t is maximal for S_j , and 0 otherwise. The binary variable z_{jd} is equal to 1, if S_j is bundled at day d , and 0 otherwise. The following Mixed-Integer-Program produces an assignment of each session to a time slot, such that each S_j receives an EDS. Therefore $\lambda_1, \lambda_2, \lambda_3 > 0$ are weight coefficients, that are specified later.

$$\begin{aligned}
\min \quad & \lambda_1 \sum_{j \in J} U_j + \lambda_2 \sum_{j \in J} \sum_{t \in T} y_{jt} - \lambda_3 \sum_{j \in J} \sum_{d \in D} z_{jd} \\
\text{s.t.} \quad & \sum_{t \in T} x_{st} = 1 \quad \forall s \in S \quad (1) \\
& U_j + y_{jt} - \sum_{s \in S_j} x_{st} \geq 1 \quad \forall j \in J, t \in T \quad (2) \\
& \sum_{t \in T} y_{jt} \geq 1 \quad \forall j \in J \quad (3) \\
& z_{jd} - y_{jt} \leq 0 \quad \forall j \in J, d \in D, t \in T_d \quad (4) \\
& x_{st} \in \{0, 1\} \quad s \in S, t \in T \\
& U_j \geq 0 \quad \forall j \in J \\
& y_{jt} \in \{0, 1\} \quad \forall j \in J, t \in T \\
& z_{jd} \in \{0, 1\} \quad \forall j \in J, d \in D
\end{aligned}$$

The objective function minimizes U_j , what is the maximal number of sessions that are scheduled at some time slot. In addition, the number of maximal time slots is minimized and the number of bundled days is maximized for every $S_j \in \mathcal{S}$. Inequalities (1) mean, that every session must be assigned to exactly one time slot. Constraints (2) indicate, that U_j is bounded from below by the number of sessions at time slot t , if t is maximal for S_j ($y_{jt} = 1$). Otherwise, if t is not maximal for S_j ($y_{jt} = 0$), then U_j must differ by at least one from the number of sessions at t . This is because at least one maximal time slot must exist for every S_j (3). Inequalities (4) say, that if S_j is bundled at day d , then all $t \in T_d$ must be maximal.

Note that U_j is always integral in an optimal solution of the model. Moreover, the integrality condition of z_{jd} can be relaxed to $0 \leq z_{jd} \leq 1$.

The next figure illustrates the relation between an arbitrary time assignment and the induced variables values.

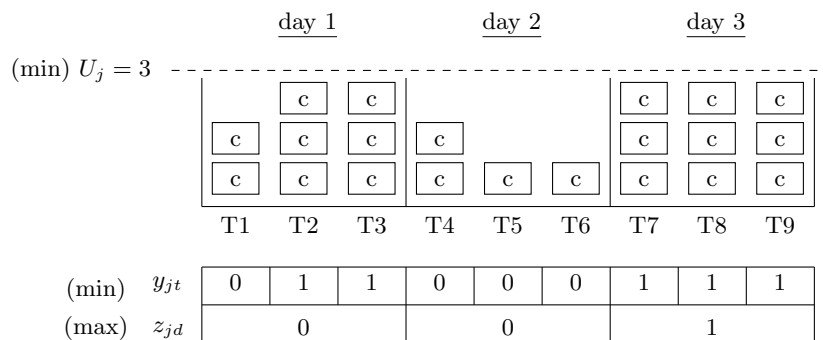


Figure 11: Time assignment and the induced variable values and for session set $S_j = S_c$

Remark. In particular, an optimal solution of the optimization model yields an EDS for every S_j , only if the weight coefficients $\lambda_1, \lambda_2, \lambda_3$ are chosen correctly. Therefore we have to make a distinction between the variables in the objective function to obtain an EDS for every S_j with $j \in J$.

Proposition 5. *Let $\lambda_1 > (|T| - 1) \cdot \lambda_2$ and $\lambda_2 > \lambda_3$. A solution of the above time assignment model is optimal with respect to weight coefficients $\lambda_1, \lambda_2, \lambda_3 > 0$, if and only if all $S_j \in \mathcal{S}$ with $j \in J$ have an evenly distributed schedule (EDS).*

Proof. Let $x = (x_{st})_{s \in \mathcal{S}, t \in T}$ be the assignment vector of an optimal solution of the time-assignment with $\lambda_1 > (|T| - 1) \cdot \lambda_2$ and $\lambda_2 > \lambda_3$. For the proof we look at an arbitrary session set S_j and assume that it has no evenly distributed schedule. Therefore, there must either exist time slots t, t' with $\sum_{s \in S_j} x_{st} - \sum_{s \in S_j} x_{st'} > 1$ or the number of time slots is not maximal, when $\sum_{s \in S_j} x_{st} - \sum_{s \in S_j} x_{st'} \leq 1$.

At first, we will consider the case of $\sum_{s \in S_j} x_{st} - \sum_{s \in S_j} x_{st'} > 1$ for time slots $t, t' \in T$ and make a case-by-case-analysis. W.l.o.g we can assume that t is a maximal time slot.

Case 1: t is the only maximal time slot.

Swapping a session from time slot t to t' decreases U_j by one, because U_j was bounded from below by the number of sessions at a maximal time slot. As t was the only maximal time slot, U_j can be decreased by one due to this swap. The number of maximal time slots could increase to at most $|T| - 1$, since there exists at least one maximal time slot in any time assignment. However, the number of bundled days can only be increased due to the swap, so we obtain a cost reduction of $-\lambda_1 + (|T| - 1) \cdot \lambda_2 < 0$ in the worst case. But that is a contradiction on x being optimal.

Case 2: t is not the only maximal time slot.

Swapping one session from t to t' decreases the number of maximal time slots by one. Since there is at least one further maximal time slot, U_j could not be decreased and remains the same. However, at most one bundled day could be destroyed due to the swap. Then we would obtain a cost reduction of $-\lambda_2 - (-\lambda_3) < 0$, what is a contradiction on x being optimal.

Consequently, if x is optimal, then it holds $|\sum_{s \in S_j} x_{st} - \sum_{s \in S_j} x_{st'}| \leq 1$ for all $t, t' \in T$. If this property holds, then also the number of bundled days is maximal, because it is maximized in the objective function.

For the other direction we require all S_j with $j \in J$ to have an EDS, that is given by the assignment vector $x = (x_{st})_{s \in S, t \in T}$ and the variables U, y, z . We assume, that x is not optimal for the objective function $\min \sum_{j \in J} (\lambda_1 U_j + \lambda_2 \sum_{t \in T} y_{jt} - \lambda_3 \sum_{d \in D} z_{jd})$. Thus, there must exist at least one $j \in J$, for which the sub-objective function $\min \lambda_1 U_j + \lambda_2 \sum_{t \in T} y_{jt} - \lambda_3 \sum_{d \in D} z_{jd}$ has strictly higher cost than in the optimal solution. Let x^*, U^*, y^*, z^* be the variables values of an optimal time assignment. Again, we do a proof by cases. As already mentioned, one of the following cases must occur for some $j \in J$:

Case 1: $U_j^* < U_j$

In an EDS, U_j is already minimal, so U_j^* cannot be smaller, as it is bounded from below by the same amount of sessions.

Case 2: $\sum_{t \in T} y_{jt}^* < \sum_{t \in T} y_{jt}$

Any EDS-preserving deviation of the schedule yields the same amount of maximal time slots. Therefore, the only possibility to decrease $\sum_{t \in T} y_{jt}$, is to swap a session to an existing maximal time slot t' . Then, t' would form the only maximal time slot and the amount of maximal time slots would particularly be minimal ($= 1$). As U_j is bounded from below by the number of sessions at any time slot, U_j is increased by one due to this swap. For such a deviation, the largest possible difference in the number of maximal time slots ranges from $|T|$ to 1 in the best case. Then the cost changes to $\lambda_1 - (|T| - 1)\lambda_2 > 0$.

Case 3: $\sum_{d \in D} z_{jd} < \sum_{d \in D} z_{jd}^*$

If we increase the number of bundled days, then also the number of maximal time slots must be increased by at least one. This yields the cost change of $\lambda_2 - \lambda_3 > 0$.

Finally we can exclude case 1. For case 2 and 3 it is not possible to decrease the number of maximal time slots or to increase the number of bundled days respectively, without increasing the cost. Finally, there exists no $j \in J$, for which the optimal solution

has strictly smaller cost, that means x must be optimal. This contradicts the assumption. \square

Proposition 5 holds for both directions, consequently no evenly distributed schedule is excluded with the Mixed-Integer-Program formulation. With respect to that, we apply the following weight coefficients to the objective function:

$$\min \quad 2|T| \sum_{j \in J} U_j + 2 \sum_{j \in J} \sum_{t \in T} y_{jt} - \sum_{j \in J} \sum_{d \in D} z_{jd}$$

5.1.2 Room capacity restrictions

Each session differs in its cluster, content, speakers and the contained talks. Naturally, each attendee has own interests and preferences concerning the choice of his session. Hence, all sessions will obviously have different numbers of attendees. As each session takes place in a certain room, it is necessary that the available number of seats is not exceeded. Especially for large conferences, where the room sizes may be significantly different, the compliance with room capacities is a very important requirement on the schedule.

In this section we will refer to the work of Lübbecke and Lach [2][3]. They present inequalities that satisfy room capacity restrictions within a time assignment. This is a non-trivial matter, as we are just considering assignments of sessions to time slots in this model. We further want give a proof that these inequalities are sufficient to model the compliance with room capacities in a time assignment.

In the following let u_r denote the capacity of a room $r \in R$ and let a_s indicate the number of attendees in a session $s \in S$. A more detailed study on how a_s is determined, is given in Section 5.2, but we naturally assume that $a_s > 0$.

Definition 6. *An assignment (s, r) of a session s to room r is called feasible, if the number of attendees does not exceed the capacity, that means $a_s \leq u_r$.*

Let S_t be the set of all sessions, that are scheduled at time slot t . This set is actually not defined yet, because it is only given by a solution of the time assignment. But in the following we will give sufficient conditions for S_t to comply with the room capacities. These conditions can be formulated by means of linear inequalities, as proposed in [2] and [3]. For that, we model each session $s \in S_t$ and every room $r \in R$ as vertices in the vertex set $S_t \cup R$. Furthermore let E be a set of the undirected edges, that model all feasible assignments (s, r) between sessions $s \in S_t$ and rooms $r \in R$. We define $G_t = (S_t \cup R, E)$ as the feasible assignment graph for time slot t . In particular, G_t is a bipartite graph with the bipartition $\{S_t, R\}$.

Definition 7. *Let $G = (V, E)$ be an undirected graph. $\mathcal{M} \subseteq E$ is called a matching of G , if the edges of \mathcal{M} have no vertex in common. \mathcal{M} is a perfect matching, if every vertex $v \in V$ is incident to an edge in \mathcal{M} .*

Lemma 8. *The sessions of S_t comply with the capacities of R if and only if the feasible assignment graph G_t contains a matching \mathcal{M} such that each vertex of S_t is incident to an edge in \mathcal{M} .*

Proof. Let $S_t = \{s_1, \dots, s_m\}$, then S_t complies with room capacities if there are feasible assignments $(s_1, r_1), \dots, (s_m, r_m)$ for all sessions of S_t to distinct rooms $r_i \neq r_j$ for all $i, j \in \{1, \dots, m\}$ with $i \neq j$. These feasible assignments directly correspond to the edges of the matching \mathcal{M} . The other direction is analogous. \square

Consequently, our optimization model must ensure, that there exists such a matching for all G_t with $t \in T$. As proposed in [2] and [3], we refer to the theorem of Hall, that yields a sufficient condition on the existence of such a matching.

Hall's Matching Theorem. *Let $G = (V, E)$ be an undirected bipartite graph with the bipartition $V = A \cup B$ and the edge set E . Let $\Gamma(A) \subseteq B$ be the set of vertices that are adjacent to vertices in A . G contains a perfect matching if and only if $|A'| \leq |\Gamma(A')|$ for all $A' \subseteq A$.*

We will apply this theorem to our feasible assignment graph G_t with $A = S_t$ and $B = R$. Note that the theorem also works for $|S_t| \leq |R|$, as the number of sessions at some time slot may be smaller than the number of rooms. Let $S' \subseteq S$ be an arbitrary subset of sessions and let $\Gamma(S') \subseteq R$ be the set of rooms that are adjacent to sessions of S' in G_t . The cardinality of $S' \cap S_t$ is given by $\sum_{s \in S'} x_{st}$. The following linear inequalities model the Hall condition for all feasible assignment graphs G_t :

$$\sum_{s \in S'} x_{st} \leq |\Gamma(S')| \quad \forall S' \subseteq S, t \in T$$

In general the number of these constraints is exponential and thereby hardly applicable in practice. We will give a proof that the constraints can be reduced to a polynomial number. We show that, for each subset $S' \subseteq S_t$, there exist a polynomial number of dominating inequalities.

Definition 9. *Let $d_1^T x \leq b_1$ and $d_2^T x \leq b_2$ be valid linear inequalities for the time assignment. We say inequality $d_2^T x \leq b_2$ dominates $d_1^T x \leq b_1$, if there exists a $\lambda > 0$ such that $d_1 \leq \lambda d_2$ and $\lambda b_2 \leq b_1$.*

Remark: In particular, each dominated inequality in a linear programming formulation becomes redundant.

Let r_0 be a 'dummy' room with capacity $u_{r_0} = 0$ and let R_{max} denote the set of rooms that have the highest capacity, as there might exist more than one. Further let $R_0 = R \cup \{r_0\} - R_{max}$ and $S_{>r} = \{s \in S \mid a_s > u_r\}$ be the set of sessions that have no feasible assignment to room r .

Lemma 10. *Let $S' \subseteq S$ be an arbitrary subset of sessions. Then there exists a room $r' \in R_0$ such that the following holds:*

$$(i) \ S' \subseteq S_{>r'}$$

$$(ii) \ \Gamma(S') = \Gamma(S_{>r'})$$

Proof. Let $s' = \arg \min_{s \in S'}(a_s)$ be the smallest session of S' .

Further let $r' = \arg \max_{r \in R: u_r < a_{s'}}(u_r)$ be the largest room, that can not include session s' . We show that (i) and (ii) holds for r' .

(i) Let s be an arbitrary session in S' . It holds $a_s \geq a_{s'} > u_{r'}$ and therefore $s \in S_{>r'}$ by definition of $S_{>r'}$.

(ii) Let $r \in \Gamma(S')$, then there exists a feasible assignment (s, r) with $s \in S'$. It holds $s \in S_{>r'}$ because of (i). That means (s, r) is a feasible assignment for a session $s \in S_{>r'}$ and therefore $r \in \Gamma(S_{>r'})$.

For the other inclusion let $r \in \Gamma(S_{>r'})$. It holds $u_{r'} < a_{s'} \leq u_r$, because r' is the largest room that can not include s' and $u_r > u_{r'}$ by definition of r . That means (s', r) is a feasible assignment. Since $s' \in S'$ it follows $r \in \Gamma(S')$. \square

Finally, Lemma 10 yields the following dominating inequalities for every $S' \subseteq S$:

$$\sum_{s \in S'} x_{st} \leq \sum_{s \in S_{>r'}} x_{st} \leq |\Gamma(S_{>r'})| = |\Gamma(S')| \quad \forall t \in T$$

We do the reformulation of $\Gamma(S_{>r'}) = \{r \in R \mid u_r > u_{r'}\} =: B_{r'}$ and add the reformulated linear inequalities to the existing time assignment model formulation:

$$\sum_{s \in S: a_s > u_r} x_{st} \leq |B_r| \quad \forall r \in R_0, t \in T \quad (5)$$

Remark: Rooms with equal capacity describe the same inequality. Therefore, we could also refer to a subset of rooms of R_0 whose capacities are pairwise different.

Corollary 11. *There exist a polynomial number of inequalities that dominate the Hall inequalities.*

Corollary 12. *The inequalities (5) are sufficient for the existence of a matching in the feasible assignment graph G_t . Thus, a feasible solution of the time assignment model yields the compliance with room capacities at every time slot $t \in T$.*

Summarized the main argument for the reducible number of constraints is, that we can exploit the sorting of the attendance values a_s and capacities u_r . If an assignment of a session s to room r is feasible, then an assignment of a smaller session to r is also feasible. This implies a redundancy of inequalities in our model. An illustration to this

L	S	L	S	L	S	L	S
L	S	L	S	L	S	L	S
T1	T2	T3	T4	T5	T6	T7	T8

Figure 13: Unsuitable time assignment for S_c

the small sessions must be scheduled to small rooms, because larger rooms are occupied. If we apply a provisional room assignment on this time distribution, the schedule would look similar to the following figure:

large rooms:	L		L		L		L	
	L		L		L		L	
small rooms:		S		S		S		S
		S		S		S		S
	T1	T2	T3	T4	T5	T6	T7	T8

Figure 14: Expected room assignment: large sessions to large rooms, small sessions to small rooms

This worst case example shows that big and small sessions are alternating between large and small rooms over the entire length of the conference. That means people, who want to attend this cluster, have to change the room at every consecutive time slot. Furthermore this cluster would probably occupy four different rooms, even if two rooms might be possible. Another disadvantage is the time overlap of the large sessions, because they are more popular in general. Since one participant can visit only one talk at the same time, he would miss the other popular talk. These situations have to be excluded for real-world instances. A more suitable time assignment would be the following:

large rooms:	L	L	L	L	L	L	L	L
small rooms:	S	S	S	S	S	S	S	S
	T1	T2	T3	T4	T5	T6	T7	T8

Figure 15: Even distribution of large and small sessions

In this time assignment, large and small sessions are evenly distributed over the time slots. We generalize this to the fact that an even distribution must additionally hold for sessions with different capacity demand. To achieve such a distribution in our schedule, our approach is to apply the EDS constraints of the previous section to further sets of sessions. In contrast to the last example we need a more precise categorization of the needed capacities of the sessions.

Since we consider feasible assignments only, it holds $a_s \leq u_r$ for each assignment of a session s to room r . Hard capacity constraints will restrict the assignments, such that

u_r is generally not much greater than a_s , because larger rooms are occupied by larger sessions. Let $\mathcal{L} = \{u_r \mid r \in R\}$ be the set of room capacities. We define a *capacity labeling* as a function

$$\pi : S \rightarrow \mathcal{L}, s \mapsto \min\{u_r \mid a_s \leq u_r\}.$$

It describes the minimum possible capacity of an available room that a session can be allocated to. Thus it can be seen as the capacity, where each session 'fits best' in the sense of hard capacity constraints. In particular, all sessions with the same label can be regarded as equal in terms of needed capacities.

Let $S_{cl} = \{s \in S \mid c_s = c, \pi(s) = l\}$ be the session sets that contain all sessions that belong to cluster c and capacity label l . Let $\mathcal{L}_c = \{l \in \mathcal{L} \mid S_{cl} \neq \emptyset\}$ be the set of labels that are covered by sessions of cluster c . Then, similar to the example before, we apply the EDS constraints to each S_{cl} by should be evenly distributed over the time slots. Therefor updating the EDS set \mathcal{S} to:

$$\mathcal{S} = \bigcup_{c \in C} \{S_c\} \cup \bigcup_{c \in C} \bigcup_{l \in \mathcal{L}_c} \{S_{cl}\}$$

Since we introduce additional variables and constraints for every EDS set, the problem size of the model is enlarged with this step.

Extended example

We want to present a further example for a fixed cluster c , where the sets S_c and S_{cl} are evenly distributed, but a better time assignment would be possible. For instance, let us assume that the sessions of c belong to the capacity labels $\mathcal{L}_c = \{l_1, l_2, l_3, l_4\}$ with $l_1 < l_2 < l_3 < l_4$. From the definition of \mathcal{S} an EDS is applied to the following sets: $S_c, S_{cl_1}, S_{cl_2}, S_{cl_3}, S_{cl_4}$. In particular $S_{cl_i} \subseteq S_c$ for all $i = 1, \dots, 4$. An exemplary time assignment for S_c is shown in the figure below.

l_4	l_4	l_4	l_4	l_4	l_4	l_1	l_1	l_1
l_3	l_3	l_3	l_2	l_2	l_2	l_2	l_2	l_2
T1	T2	T3	T4	T5	T6	T7	T8	T9

Figure 16: Time assignment for labeled sessions of S_c

The entire set S_c , as well as each labeling set $S_{cl_1}, S_{cl_2}, S_{cl_3}, S_{cl_4}$, is evenly distributed. Once again, we refer to the case of hard capacity constraints. From our previous observations, sessions with lower capacity labels will be scheduled into smaller rooms, while sessions of a higher label will appear in larger rooms. This is because sessions of a higher capacity label will also require a higher minimal capacity. Again, we apply a provisional room assignment to the time distribution of the previous figure, now with respect to capacity labels.

large room:	l_4	l_4	l_4	l_4	l_4	l_4			
medium room:	l_3	l_3	l_3	l_2	l_2	l_2	l_2	l_2	l_2
small room:							l_1	l_1	l_1
	T1	T2	T3	T4	T5	T6	T7	T8	T9

Figure 17: Hard capacities: high labels to large rooms, low labels to small rooms

The labels l_1, l_4 are characteristically assigned to large and small rooms, while the labels l_2, l_3 could appear in the same room of medium size. This might be possible, if the difference of l_2 and l_3 is not too large. In a worse case we can assume, that they cannot share the same room, because shifts due to hard capacity constraints are possible. The number of rooms that one cluster occupies, is to be minimized in the next optimization step. With this time distribution we would probably occupy three rooms for this cluster, but we do not know as the room assignment is done in the end of the entire optimization process. That leads to the question, if we could find a better time assignment for this case. The idea is to integrate sessions of a lower capacity label into the stream of the next higher capacity label. To continue the example, for every $i \in \{1, 2, 3\}$ the sessions of the labels l_i and l_{i+1} should be evenly distributed. Consequently the EDS constraints are applied for these additional sets of sessions. The figure below shows an improved time assignment with respect to additional EDS constraints for successive labels.

stream l_4 :	l_4	l_4	l_4	l_4	l_4	l_4			
stream l_3 :							l_3	l_3	l_3
stream l_2 :	l_2	l_2	l_2	l_2	l_2	l_2			
stream l_1 :							l_1	l_1	l_1
	T1	T2	T3	T4	T5	T6	T7	T8	T9

Figure 18: EDS for consecutive capacity labels

In contrast to the example before the successive labels $(l_1, l_2), (l_2, l_3), (l_3, l_4)$ are evenly distributed. Note that it is still possible for the labels l_2, l_3 to appear in the same room, because they received an EDS constraint. Unlike the previous time assignment, the labels l_1, l_2 and l_3, l_4 are not assigned in parallel anymore, because they also received an EDS constraint. Now it is possible for them to appear in the same room, if the capacity restrictions would be sufficient. If the integration of l_1, l_2 and l_3, l_4 into one room would be possible, then only two rooms are used for cluster c instead of three.

Inspired from that idea, we enumerate the labels (for which there exists at least one session) of each cluster c by $\mathcal{L}_c = \{l_1, l_2, \dots\}$ and update the EDS set \mathcal{S} .

$$\mathcal{S} = \bigcup_{c \in C} \{S_c\} \cup \bigcup_{c \in C} \bigcup_{l \in \mathcal{L}_c} \{S_{cl}\} \cup \bigcup_{c \in C} \bigcup_{i=1}^{|\mathcal{L}_c|-1} \{S_{cl_i} \cup S_{cl_{i+1}}\}$$

Apart from \mathcal{S} the Mixed Integer Program remains the same, as it was introduced in Section 5.1.1 where \mathcal{S} was defined with help of an index set J .

Summarized, the capacity labeling contributes to a time-dependent presorting by the needed room capacities of each session. This is supposed to provide a better room assignment in the next optimization step. It is an intuitional approach that we believe to improve the later solutions. For our contribution in this thesis we have not proven, if a time assignment with labeling constraints would always yield a better schedule. But from our observations, they contribute significantly to a better scheduling result. An computed example with and without labeling constraints is given in the section 'computational results'.

5.1.4 Lower bound constraints

The time assignment model focuses on the even distribution of all session sets $S_j \in \mathcal{S}$ with $j \in J$. For each of these sets we applied EDS constraints. A further advantage of the EDS constraints is that there exist strong lower bounds.

Let $S_j \in \mathcal{S}$ be a set of sessions that belongs to an EDS and let U_j, y_{jt} and z_{jd} be the corresponding variables, as they were introduced in Section 5.1.1. Further let $N = \min_{d \in D} |T_d|$. The objective function of the time assignment model reads as:

$$\min \sum_{j \in J} (\lambda_1 U_j + \lambda_2 \sum_{t \in T} y_{jt} - \lambda_3 \sum_{d \in D} z_{jd})$$

Definition 13. Consider the following parameters:

- (i) $U_j^* = \lceil \frac{|S_j|}{|T|} \rceil$
- (ii) $y_j^* = |S_j| \bmod |T|$
- (iii) $z_j^* = \lfloor \frac{|S_j| \bmod |T|}{N} \rfloor$

Claim 14. Every feasible solution of the time assignment and every $j \in J$ satisfies the following inequalities:

- (1) $U_j^* \leq U_j$
- (2) $\lambda_1 U_j^* + \lambda_2 y_j^* - \lambda_3 z_j^* \leq \lambda_1 U_j + \lambda_2 \sum_{t \in T} y_{jt} - \lambda_3 \sum_{d \in D} z_{jd}$

Proof. (1) U_j is the upper bound on the number of sessions of S_j at each time slot. Each session of S_j has to be scheduled, so the minimum number of sessions at each time slot is $\frac{|S_j|}{|T|}$. Since the number of scheduled sessions at each time slot is integral, it holds $\lceil \frac{|S_j|}{|T|} \rceil = U_j^* \leq U_j$.

(2) Proposition 5 says that an EDS yields the optimal objective value for each S_j . Therefore, we show that the lower bound in (2) is the objective value of an EDS of S_j . The minimal possible value of U_j is U_j^* , what was already shown in (1). Especially in an EDS it holds $U_j = U_j^*$. There are exactly $|S_j| \bmod |T|$ leftover time slots, where S_j is maximal. Each bundled day of S_j consists of at least N maximal time slots. That means from the number of maximal time slots in the EDS, we can create at most $\lfloor \frac{|S_j| \bmod |T|}{N} \rfloor$ bundled days. As these are the best possible values for each of the variables in an EDS, the inequality (2) yields a lower bound for S_j . \square

For each $j \in J$ we can add the following constraints to the time assignment model:

$$U_j^* \leq U_j \quad \forall j \in J \quad (6)$$

$$\lambda_1 U_j^* + \lambda_2 y_j^* - \lambda_3 z_j^* \leq \lambda_1 U_j + \lambda_2 \sum_{t \in T} y_{jt} - \lambda_3 \sum_{d \in D} z_{jd} \quad \forall j \in J \quad (7)$$

From our observations, the lower bound constraints contribute significantly to the computation process. In our test instances we obtained smaller integrality gaps by adding these lower bounds to the model formulation such that the time assignment model was almost optimally solvable.

5.1.5 Chairman constraints

Every session at the ISMP conference has a chairman who gives a short introduction about the talks and the speakers in the session. A chairman is also able to give a talk at the conference, what does not necessarily have to be contained in the session he attends as chairman. This special case, where a chairman must be present in several sessions at the same time, has to be excluded. Therefore let S_p^{talk} be the set of sessions, where a person $p \in P$ gives a talk and S_p^{chair} the set of sessions p attends as chairman. For our case at ISMP the number of talks a person may give was restricted to one, i.e. $|S_p^{talk}| = 1$. The avoidance of time overlaps of the corresponding sessions is modeled by the linear inequality:

$$\sum_{s \in S_p^{talk} \cup S_p^{chair}} x_{st} \leq 1 \quad \forall t \in T, p \in P : |S_p^{talk} \cup S_p^{chair}| > 1 \quad (8)$$

5.1.6 Individual scheduling requests

During the planning process of large conferences, like the ISMP 2012, there occur a series of individual scheduling requests from the conference participants. We figured out three main types of requests: *interval*, *no overlap* and *precedence* requests. Each type of request is modeled as additional linear inequality in the time assignment.

Interval requests

An interval requests asks to schedule a session within a certain interval of time slots. Therefore let S_I denote the set of sessions that belong to an interval request and let $[t_s^-, t_s^+]$ be the desired time slot interval. The following inequality ensures that these sessions are scheduled within the requested interval:

$$\sum_{t \in [t_s^-, t_s^+]} x_{st} = 1 \quad \forall s \in S_I \quad (9)$$

No overlap requests

The *no overlap* requests ask that a pair of sessions (s_1, s_2) is not scheduled not in parallel. Here let $S_N \subseteq S^2$ be the set of session pairs that must not overlap. The requests are modeled by the linear inequalities:

$$x_{s_1 t} + x_{s_2 t} \leq 1 \quad \forall (s_1, s_2) \in S_N, t \in T \quad (10)$$

Precedence requests

Precedence requests constrain a session to be scheduled before another session. In general, such requests occur, when there are dependencies between two sessions according to their title and content. For example, there might be precedence constraints between sessions with the titles: *Scheduling I*, *Scheduling II*, *Scheduling III* that have to be scheduled in direct succession.

We differentiate between *hard* and *soft* precedence constraints. Hard constraints state that two sessions have to appear at two successive time slots, while soft constraints allow further time slots in between. Let $S_P^{hard}, S_P^{soft} \subseteq S^2$ denote the set of session pairs with hard and soft precedence constraints. Each element (s_1, s_2) indicates that s_1 must be scheduled before s_2 . Let $T' = T - \max(T)$, then hard and soft precedence constraints are modeled as:

$$x_{s_1 t} = x_{s_2 t+1} \quad \forall (s_1, s_2) \in S_P^{hard}, t \in T' \quad (11) \quad (\text{hard precedence})$$

$$x_{s_1 t} \leq \sum_{t' \in T: t' \geq t+1} x_{s_2 t'} \quad \forall (s_1, s_2) \in S_P^{soft}, t \in T' \quad (12) \quad (\text{soft precedence})$$

5.1.7 Even attractivity distribution

At each conference there are some talks that are more popular than others, because of different interests in specific talks, famous speakers or new results in research. Obviously,

more people are attracted to attend those sessions. Therefore it is important to have an equal amount of interesting talks available at each time slot to establish an attractive scientific program for the whole conference.

To determine a degree of popularity for each session, we have to distinguish between 'attractive' and 'less attractive' sessions. In that sense, we introduce an attractivity coefficient $\alpha_s > 0$ that denotes the attractivity of each session s . The higher the value, the more people are attracted to attend this session.

There are many possibilities to define this value. For the ISMP, we used a weighted sum of potential attractivity indicators to obtain α_s , like: *number of submitted talks for c_s , stated preferences at the webpage registration and empirical attendee numbers of a previous conference*. Since the attractivity of a session is in the eye of the beholder, we define α_s to be the attractivity of an average participant. The values of α_s are now used to implement an even attractivity distribution over the length of the conference.

For that, we define the variable $A_{min} \geq 0$, which is a lower bound on the sum of attractivities at every time slot $t \in T$. Similarly, the variable $A_{max} \geq 0$ yields an upper bound for every time slot $t \in T$. The even attractivity distribution is modeled by linear inequalities and an additional term in the objective function:

$$A_{min} \leq \sum_{s \in S} \alpha_s x_{st} \quad \forall t \in T \quad (13)$$

$$A_{max} \geq \sum_{s \in S} \alpha_s x_{st} \quad \forall t \in T \quad (14)$$

Thus, for each pair of time slots the absolute difference in the sum of attractivities is at most $A_{max} - A_{min}$. Since the attractivities should be evenly distributed, $A_{max} - A_{min}$ is minimized. Finally we add the following term to the objective function with weight coefficient $\lambda_4 \geq 0$:

$$\min \quad \lambda_4 (A_{max} - A_{min})$$

The coefficient λ_4 can be adjusted when looking at the produced schedules.

In Section 5.2 we will use the property of an even attractivity distribution to approximate the uncertain number of attendees a_s by use of a stochastic model. We will see that a balanced attractivity distribution even has good side effects.

5.2 Robustness of capacity compliance

In the previous subsection we introduced attractivity values α_s to determine the average interest in every session $s \in S$ at the conference. With respect to these values we derive a stochastic model to determine the number of attendees a_s for every session. In the real world, as well as in our model concept, the number of attendees is an uncertain quantity. In our stochastic modeling approach, the needed capacity of every session is calculated

with use of a nonlinear model. We will give an approach to deal with the nonlinearity of that model, as well as the integration to the existing time assignment formulation.

First, let n denote the total number of attendees at the conference or, respectively, the maximum number of attendees at a specific time slot. The total number of attendees may vary between the time slots. But from the perspective of an organizer, we have to assume that each participant takes part in the whole conference. Therefore, we suppose that n attendees are present at each time slot. We further assume that every person has average preferences regarding the choice of his session, that means his choice between the sessions depends on the attractivity coefficients α_s and we also suppose that each participant makes his choice independently from the other participants. It is expected that every participant chooses a session at each time slot t , so the choice depends on the attractivities α_s of the sessions, that are scheduled at t . Consequently, we define the probability to choose some session s at time slot t , as the ratio between α_s and the total attractivity at t . An illustration to the choice modeling is given in Figure 18.

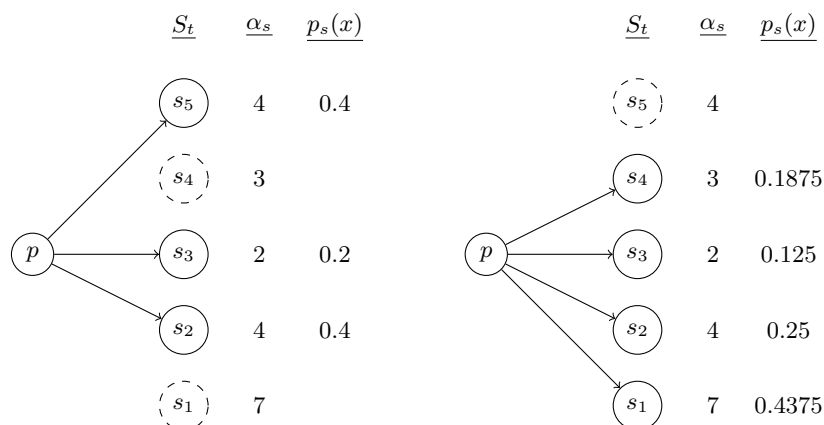


Figure 19: Selection probabilities $p_s(x)$ of a participant p for two different time assignments

The figure shows two different time assignments for time slot t . Since the probability to choose one particular session depends on the selection of sessions that are available at t , the choice probabilities may change for two different time assignments, as for s_2, s_3 in the example. From these assumptions we define the following stochastic concepts:

Definition 15. Let $\Omega = \{\omega_1, \omega_2\}$ be the decision set to choose (ω_1) or not to choose a session (ω_2). For all sessions $s \in S$ and all participants $j \in P$ let $X_{sj} : \Omega \rightarrow \{0, 1\}$ be discrete and stochastically independent random variables that are defined as

$$X_{sj}(\omega) = \begin{cases} 1, & \omega = \omega_1 \\ 0, & \omega = \omega_2 \end{cases}$$

$X_{sj} = 1$ states that participant j chooses session s and $X_{sj} = 0$ otherwise. Further let S_t denote the set of sessions at t and $t(s)$ be the time slot, when session s is scheduled.

It is assumed that there exists at least one session at each time slot, i.e. $\sum_{s \in S} x_{st} \geq 1$ for all $t \in T$. We define the discrete probability function $\mathbb{P} : \Omega \rightarrow [0, 1]$ by:

$$\mathbb{P}(X_{sj} = 1) = \frac{\alpha_s}{\sum_{s' \in S_{t(s)}} \alpha_{s'} x_{s't(s)}} = p_s(x)$$

$$\mathbb{P}(X_{sj} = 0) = 1 - p_s(x).$$

Then $p_s(x)$ denotes the probability that participant j will attend session s . Moreover, each participant chooses session s with the same probability. Since α_s refers to average preferences.

Note that α_s referred to average preferences and $\alpha_s > 0$ for all $s \in S$. It is easy to see that \mathbb{P} defines a probability function. We also notice that $p_s(x)$ is a rational function in x and especially non-linear. This can be interpreted by the fact that the choice on session s depends on the selection of sessions that are available at time slot $t(s)$. This selection is determined by the decision variables x_{st} of the time assignment.

Definition 16. Each random variable X_{sj} has a value of either one or zero (success or no success). Therefore, X_{sj} defines a Bernoulli trial. The sequence X_{s1}, \dots, X_{sn} over all participants implies a Bernoulli process for every session s . We define the random variable X_s as the number of successes in this Bernoulli process. In the model, this is analogous to the number of participants who choose session s .

$$X_s = \sum_{j \in P} X_{sj}$$

In particular, we obtain the binomial distribution function. That yields the probability that exactly k persons will choose session s :

$$\mathbb{P}(X_s = k) = \binom{n}{k} p_s(x)^k (1 - p_s(x))^{n-k} =: B(k, n, p_s(x))$$

Similarly, the cumulative distribution function yields the probability that at most k persons attend session s :

$$\mathbb{P}(X_s \leq k) = \sum_{j=0}^k \binom{n}{j} p_s(x)^j (1 - p_s(x))^{n-j} =: F(k, n, p_s(x))$$

Note that the probabilities still depend on the decision variables x_{st} .

From these definitions we want to determine the number of attendees a_s . One possibility would be to choose the expected value of X_s , that is:

$$a_s(x) = \mathbb{E}[X_s] = n \cdot p_s(x)$$

But in general, the use of expected values will not work well in practice. This is because the probability that some session exceeds the expected value is quite large. For our considerations we want to have a degree of certainty that the calculated number of attendees will not be exceeded. Therefore, we define a certainty value $\gamma_s \in [0, 1]$ which defines the probability, that session s will not be exceeded. Thus, the number of attendees is defined as the minimum number k , such that the probability that at most k participants choose session s is greater than γ_s :

$$a_s(\gamma_s, x) = \min\{k \in \mathbb{N} \mid \mathbb{P}(X_s \leq k) \geq \gamma_s\}$$

$$\Leftrightarrow a_s(\gamma_s, x) = \min\{k \in \mathbb{N} \mid F(k, n, p_s(x)) \geq \gamma_s\}$$

Two problems remain with this formulation. First, the total needed capacity increases for higher values of γ_s . If we choose them arbitrarily large, then the capacity constraints might get infeasible. We further assume that γ_s is chosen, such that all capacity constraints are satisfied. The next problem is the nonlinearity of the number of attendees $a_s(\gamma_s, x)$, because the inequality (5), see Section 5.1.2, uses constant values for a_s .

One important observation is, if we use a constant value $\tilde{a}_s \geq a_s(\gamma_s, x)$, then the room capacity will also be sufficient for the stochastic values $a_s(\gamma_s, x)$, because we are considering feasible assignments only, that means:

$$a_s(\gamma_s, x) \leq \tilde{a}_s \leq u_r \Rightarrow a_s(\gamma_s, x) \leq u_r \Rightarrow (s, r) \text{ is a feasible assignment.}$$

If we would use values smaller than $a_s(\gamma_s, x)$, then it is not ensured that capacities are still sufficient.

In our approach we create upper bound values \tilde{a}_s for every session s , what implies the compliance with room capacities for the stochastic values. The disadvantage of upper bounds is the increase in the needed room capacities. It is necessary that the gap between \tilde{a}_s and $a_s(\gamma_s, x)$ is kept tight. We present an approximation of $\approx (1 + 2\epsilon + \epsilon^2)$ ($\epsilon \geq 0$) on the stochastic values. This approach makes use of the approximability of an even attractivity distribution which was introduced in Section 5.1.6.

Let $\epsilon \geq 0$ and let $C = \frac{1}{|T|} \sum_{s \in S} \alpha_s$ be the mean attractivity at each time slot. The constant probability to choose session s is defined as $\tilde{p}_s = \frac{\alpha_s}{C}(1 + \epsilon)$. We apply the following inequalities to the existing time assignment model:

$$C \leq A_{min}(1 + \epsilon) \quad (15) \quad A_{max} \leq C(1 + \epsilon) \quad (16)$$

Proposition 17. *Let x be a feasible solution of the time assignment with respect to inequalities (15) and (16). Then \tilde{p}_s yields an $(1+2\epsilon+\epsilon^2)$ -approximation on the stochastic probabilities $p_s(x)$ for all $s \in S$, that means $p_s(x) \leq \tilde{p}_s \leq p_s(x)(1 + 2\epsilon + \epsilon^2) \quad \forall s \in S$.*

Proof. First, we show $p_s(x) \leq \tilde{p}_s$:

$$p_s(x) = \frac{\alpha_s}{\sum_{s' \in S} \alpha_{s'} x_{s't(s)}} \leq \frac{\alpha_s}{A_{min}} \leq \frac{\alpha_s}{C} (1 + \epsilon) = \tilde{p}_s.$$

The first inequality yields constraint (13) of the time assignment model, and the second inequality follows from constraint (15).

The second inequality $\tilde{p}_s \leq p_s(x)(1 + 2\epsilon + \epsilon^2)$ follows from :

$$\tilde{p}_s = \frac{\alpha_s}{C} (1 + \epsilon) \leq \frac{\alpha_s}{A_{max}} (1 + \epsilon)^2 \leq \frac{\alpha_s}{\sum_{s' \in S} \alpha_{s'} x_{s't(s)}} (1 + \epsilon)^2 = p_s(x)(1 + \epsilon)^2 = p_s(x)(1 + 2\epsilon + \epsilon^2)$$

The first inequality is a direct consequence of constraint (16) and the second follows from inequality (14). \square

Remark 18. *Note that we compute a solution of the time assignment model with respect to fixed parameters $\epsilon \geq 0$ and $\gamma_s \in [0, 1]$. For arbitrary small ϵ the optimization model may get infeasible, because a completely even attractivity distribution is not possible. Also for values of $\gamma_s \rightarrow 1$, the model may also get infeasible, as the needed capacities will increase.*

If we add inequalities (15) and (16) to our model, then we obtain an approximation factor of $(1 + 2\epsilon + \epsilon^2)$ on the probability to choose session s . In the following we use the approximated constant probabilities \tilde{p}_s to define a constant upper bound on the number of attendees \tilde{a}_s with respect to the certainty value γ_s :

$$\tilde{a}_s(\gamma_s) = \min\{k \in \mathbb{N} \mid F(k, n, \tilde{p}_s) \geq \gamma_s\}$$

Proposition 19. *For arbitrary $\gamma_s \in [0, 1]$, the constant $\tilde{a}_s(\gamma_s)$ defines an upper bound for the stochastic values $a_s(\gamma_s, x)$, that means:*

$$a_s(\gamma_s, x) \leq \tilde{a}_s(\gamma_s) \quad \forall s \in S$$

Proof. By proposition 17 it holds $p_s(x) \leq \tilde{p}_s$. For fixed k and n the functional value of the cumulative binomial distribution function F decreases with increasing probability, that means $F(k, n, \tilde{p}_s) \leq F(k, n, p_s(x))$. Now, let $k = a_s(\gamma_s, x)$ and $\tilde{k} = \tilde{a}_s(\gamma_s)$. We assume $\tilde{a}_s(\gamma_s) < a_s(\gamma_s, x)$ or $\tilde{k} < k$ respectively. By definition of $a_s(\gamma_s, x)$ and $\tilde{a}_s(\gamma_s)$, it holds $\gamma_s \leq F(\tilde{k}, n, \tilde{p}_s)$ and $\gamma_s \leq F(k, n, p_s(x))$. From that, it follows:

$$\gamma_s \leq F(\tilde{k}, n, \tilde{p}_s) \leq F(\tilde{k}, n, p_s(x)) < F(k, n, p_s(x))$$

The last inequality uses the strict monotony of the cumulative binomial distribution function. That forms a contradiction on k being minimal for $F(k, n, p_s(x)) \geq \gamma_s$, since it also holds for $\tilde{k} < k$. Consequently it follows: $k \leq \tilde{k} \Leftrightarrow a_s(\gamma_s, x) \leq \tilde{a}_s(\gamma_s)$. \square

We found constant upper bound values on the stochastic number of attendees. That means if we choose the values $\tilde{a}_s(\gamma_s)$ instead of the stochastic values $a_s(\gamma_s, x)$, then every feasible assignment of sessions to rooms will also be feasible for the stochastic values.

Quality of the upper bound

Now we want to make a statement concerning the quality of the upper bound. As higher values of $\tilde{a}_s(\gamma_s)$ will increase the needed capacities, it would be helpful to have this bound as tight as possible. To show the approximation quality, we transform problem of the discrete binomial distribution to continuous intervals, using the normal distribution. For large n , it is known that the normal distribution approximates the binomial distribution for given expected value and variance of the binomial distribution. We will show that the transformation to the continuous case yields an approximation of $\tilde{a}_s(\gamma_s) \leq a_s(\gamma_s, x)(1 + 2\epsilon + \epsilon^2)$, what then nearly corresponds to the discrete case.

At first, we define the cumulative distribution function of the standard normal distribution:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-y^2/2} dy.$$

To evaluate the cumulative normal distribution function in a certain point x we use the known substitution $\Phi(\frac{x-\mu}{\sigma})$ with given expected value μ and variance σ^2 . In particular, the binomial distribution obtains $\mu = np$ and $\sigma^2 = np(1-p)$. For large n there is the approximation.

$$F(k, n, p) \approx \Phi\left(\frac{k-\mu}{\sigma}\right).$$

Let $\epsilon' = 2\epsilon + \epsilon^2$. As already shown in proposition 17, it holds $p_s(x) \leq \tilde{p}_s \leq p_s(x)(1 + \epsilon')$. We abuse notation and denote $p = p_s(x)$ and $\tilde{p} = \tilde{p}_s$. The expected values and variances of the binomial distribution for p and \tilde{p} are known to be the following:

$$\begin{aligned} \mu &= np, & \sigma^2 &= np(1-p) \\ \tilde{\mu} &= n\tilde{p}, & \tilde{\sigma}^2 &= n\tilde{p}(1-\tilde{p}) \end{aligned}$$

Let $\gamma_s \in [0, 1]$ be the certainty value. There are values $a, \tilde{a} \in \mathbb{R}$ for which $\Phi(\frac{a-\mu}{\sigma}) = \Phi(\frac{\tilde{a}-\tilde{\mu}}{\tilde{\sigma}}) = \gamma_s$. These values correspond to the number of attendees in the continuous case. The next proposition refers to the case of $0.5 \leq \gamma_s$, what is usual for practical problems, so we actually do not restrict the problem here.

Proposition 20. *For a given certainty value $\gamma_s \in [0.5, 1]$, let $a, \tilde{a} \in \mathbb{R}$ with $\Phi(\frac{a-\mu}{\sigma}) = \Phi(\frac{\tilde{a}-\tilde{\mu}}{\tilde{\sigma}}) = \gamma_s$. Then \tilde{a} approximates a within a factor of $(1 + \epsilon')$.*

Proof. Equally to the observations of proposition 17, Φ yields, at some fixed, point smaller functional values for higher probabilities. Therefore it holds $a \leq \tilde{a}$, similar to the discrete case.

For the other inequality, it holds $a - np = a - \mu \geq 0$ because of $\gamma_s \geq 0.5$. We further know that $\tilde{p} \leq p(1 + \epsilon')$. We do the following assessment:

$$\begin{aligned}
a - np &\leq \sqrt{\frac{1-p}{1-\tilde{p}}}(a - np) \leq \sqrt{\frac{(1+\epsilon')np(1-p)}{n\tilde{p}(1-\tilde{p})}}(a - np) \\
&\leq \sqrt{\frac{np(1-p)}{n\tilde{p}(1-\tilde{p})}}(1 + \epsilon')(a - np) \leq \sqrt{\frac{np(1-p)}{n\tilde{p}(1-\tilde{p})}}(a(1 + \epsilon') - n\tilde{p}) \\
&\Leftrightarrow \frac{a-np}{\sqrt{np(1-p)}} \leq \frac{a(1+\epsilon')-n\tilde{p}}{\sqrt{n\tilde{p}(1-\tilde{p})}}
\end{aligned}$$

With use of the last inequality we obtain:

$$\gamma_s = \Phi\left(\frac{\tilde{a}-n\tilde{p}}{\sqrt{n\tilde{p}(1-\tilde{p})}}\right) = \Phi\left(\frac{a-np}{\sqrt{np(1-p)}}\right) \leq \Phi\left(\frac{a(1+\epsilon')-n\tilde{p}}{\sqrt{n\tilde{p}(1-\tilde{p})}}\right)$$

Looking at the left- and right-hand side, the monotony of Φ yields $\tilde{a} \leq a(1 + \epsilon')$. \square

Summarized, the transformation of the problem to continuous intervals yields an approximation factor of $\epsilon' = 1 + 2\epsilon + \epsilon^2$ for \tilde{a}_s . We do not achieve this approximation in general for the binomial distribution. But since for large n , the approximation quality of $\Phi\left(\frac{\tilde{a}-\tilde{\mu}}{\tilde{\sigma}}\right) \approx F(\tilde{a}, n, \tilde{p})$ increases, we can conclude that it also yields a sufficient good approximation for our considerations. We tested instances of $n = 500-2000$ participants, where values of $\epsilon = 0.008$ (approximation factor of ≈ 1.016) were achievable for the ISMP instance and for general instances $\epsilon \approx 0.03$ is a realistic guideline for larger conferences.

The following table illustrates the previous assessments for the number of attendees concerning the two probabilities $p_s(x)$ and \tilde{p}_s .

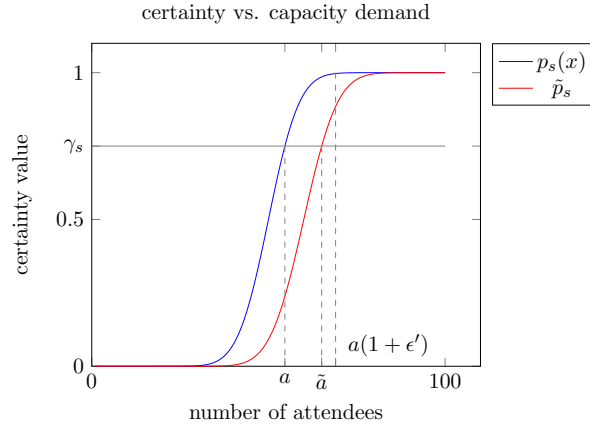
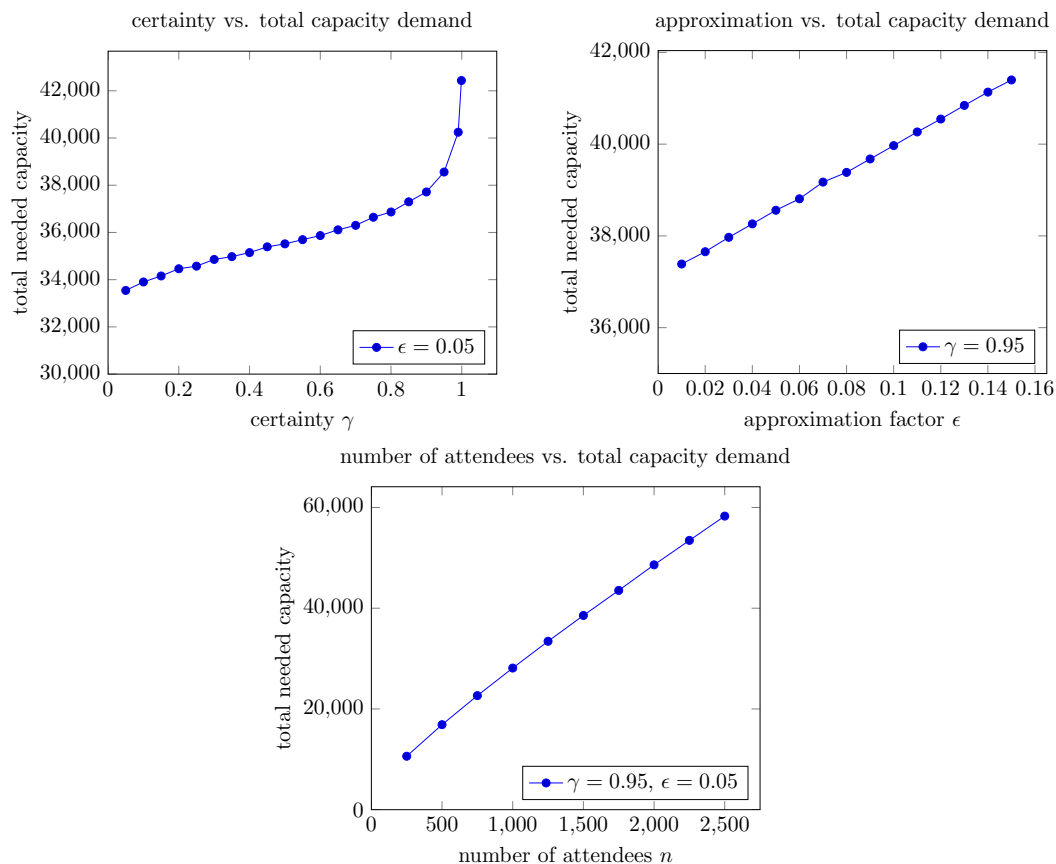


Figure 20: cumulative normal distribution for session s with probabilities $p_s(x) \leq \tilde{p}_s$

Relation to the total capacity demand

If we want to introduce a general certainty value γ , which states that no session of the conference will exceed the capacity, then we have to choose values with $\prod_{s \in S} \gamma_s \geq \gamma$. One simple possibility is to define $\gamma_s := \gamma^{\frac{1}{|S|}}$. The following tables show the increase in total capacity with respect to the certainty value γ and approximation factor ϵ . The

data refers to the ISMP problem instance with $n = 1500$ participants.



For fixed $\epsilon \geq 0$, the total capacity demand grows slowly for smaller values of γ . The absolute certainty that no session exceeds the capacity ($\gamma = 1$) occurs, when $a_s = n$ for all $s \in S$, what would be $|S| \cdot n = 892,500$ in the left figure. Therefore, the total capacity demand grows rapidly for γ close to one. In general, $\gamma = 1$ is not achievable for real-world instances. In contrast to that, for fixed certainty value γ the total capacity demand grows almost linear with approximation factor ϵ . Since we want to achieve values of ϵ close to zero, the problem is to find a small value of ϵ such that the model remains feasible. But also for small ϵ we restrict the schedule in the property that the attractiveness at each time slot is constrained to deviate from the mean attractiveness by only ϵ . The goal is to find a good balance between the values of γ and ϵ by testing different configurations of the underlying problem instance.

6 Stream Assignment

The second optimization step is called the *stream assignment* that uses the solution of the time assignment to connect sessions to *streams*. Formally, a *stream* is a sequence

(s_1, \dots, s_T) of sessions, that take place in the same room. In particular, a stream has no assignment to a certain room yet, as it only contains sessions in temporal order. A stream can also be empty, that means it contains no sessions.

This model focuses on the minimization of used streams of each cluster, what implies to find a distribution over the minimum possible number of rooms. Another objective is, that the sessions of one cluster should occur successive in the same stream. We will call this successive occurrence of a certain cluster in a stream, a *time connection*. More detailed, a cluster c has a time connection at time slot t , if there are sessions $s, s' \in S_c$ that appear successive in the same stream at time slots t and $t+1$. In the objective, the number of time connections is to be maximized.

A first observation shows, that both objectives are not equivalent to each other. Let us have a look at the following examples of a stream assignment. In Figure 21 we see that cluster c occupies a minimum number rooms, what is obviously not sufficient for a maximum number of time connections, even if c has an EDS. Considering the other case, if we have a maximal number of time connections, it is also not ensured that the number of rooms will be minimal, see Figure 22.

stream 1:	c	*	c	*	c	*	c	*	c
stream 2:	c	c	*	c	*	c	*	c	*
	T1	T2	T3	T4	T5	T6	T7	T8	T9

Figure 21: Minimal number of used rooms for S_c

stream 1:	1	1	1	2	2	2	4	4	4
stream 2:	4	4	4	3	3	3	1	1	1
	T1	T2	T3	T4	T5	T6	T7	T8	T9

Figure 22: Maximal number of time connections for each cluster $c = 1, 2, 3, 4$

In the next sections we present a Mixed-Integer-Program that assigns every session to a stream with respect to the mentioned objectives. Furthermore, we will introduce a small heuristic that modifies the solution of the stream assignment, such that a further set of constraints can be validated.

6.1 Mathematical Optimization Model

In this next part we introduce the MIP formulation to that problem. We will abuse notation and choose similar variables, like in the time assignment. Both models are regarded as completely independent from each other in terms of notation. Even though we mentioned that, by definition, a stream is independent from any room, we will use an assignment of sessions to rooms in this model. That is because we have to ensure that the stream assignment also satisfies the room capacities. Therefore, this model approach

is regarded as an allocation to rooms, but the final room of each stream may change during the last optimization step, the 'room assignment'.

Mixed Integer Program

From the solution of the time assignment, we obtained disjoint sets S_1, \dots, S_T , where S_t contains all sessions that are scheduled at time slot $t \in T$.

Further let $A = \{(s, r) \mid s \in S, r \in R, a_s \leq u_r\}$ be the set of feasible assignments from sessions to rooms. We define the binary variable x_{sr} for all feasible assignments $(s, r) \in A$:

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

For each $c \in C$, the sessions of S_c are supposed to occupy a minimum number of rooms. Therefore, we introduce the decision variable z_{cr} that takes on the value 1, if any session of cluster c takes place in room r , and 0 otherwise. To model the time connections, we define the binary variable y_{crt} that has the value 1, if there exist two sessions of cluster c in room r at time slots t and $t + 1$. Further let $R_s = \{r \in R \mid a_s \leq u_r\}$ denote the set of feasible rooms for session s and $S_r = \{s \in S \mid a_s \leq u_r\}$ the set of feasible sessions for room r . For simplification, let $S_{rt} = S_r \cap S_t$ and $S_{crt} = S_c \cap S_r \cap S_t$. We further define $T' = T - \max(T)$ and $\lambda \in [0, 1]$ is a weight coefficient between the two objectives.

$$\min \quad \lambda \sum_{c \in C} \sum_{r \in R} z_{cr} - (1 - \lambda) \sum_{c \in C} \sum_{r \in R} \sum_{t \in T'} y_{crt}$$

$$\text{s.t.} \quad \sum_{r \in R_s} x_{sr} = 1 \quad \forall s \in S \quad (1)$$

$$\sum_{s \in S_{rt}} x_{sr} \leq 1 \quad \forall r \in R, t \in T \quad (2)$$

$$\sum_{s \in S_{crt}} x_{sr} - z_{cr} \leq 0 \quad \forall c \in C, r \in R, t \in T \quad (3)$$

$$y_{crt} - \sum_{s \in S_{crt}} x_{sr} \leq 0 \quad \forall c \in C, r \in R, t \in T' \quad (4)$$

$$y_{crt} - \sum_{s \in S_{crt+1}} x_{sr} \leq 0 \quad \forall c \in C, r \in R, t \in T' \quad (5)$$

$$\begin{aligned} x_{sr} &\in \{0, 1\} & \forall s \in S, r \in R_s \\ y_{crt} &\in \{0, 1\} & \forall c \in C, r \in R, t \in T' \\ z_{cr} &\in \{0, 1\} & \forall c \in C, r \in R \end{aligned}$$

The objective function minimizes the number of rooms that are occupied by every cluster, as well as it maximizes the number of time connections of each cluster. The value of λ weights the importance between the objectives. Inequalities (1) state, that every session is assigned to a feasible room. At every time slot, each room can be occupied by at most one feasible assignment (2). Constraint (3) says, if a session of cluster c is scheduled in room r at any time slot, then cluster c occupies room r . The time connections are modeled with constraints (4) and (5): if there is a time connection of cluster c in room r at time slot t , then also sessions of cluster c must be scheduled in room r at time slots t and $t + 1$. Note that the variables y_{crt}, z_{cr} can be relaxed to: $0 \leq z_{cr} \leq 1$ and $0 \leq y_{crt} \leq 1$.

6.2 Computational complexity of the stream assignment

The solving time of the entire stream assignment model was too high for our understanding of practical implementation. We believe that the high computation time results from occurring symmetries in the model. More precise, there are assignments that do not affect the objective function. There it makes no difference whether to choose the one or the other room for a particular cluster. This makes the solving process more complex.

We tested several modifications of the stream assignment MIP. It turned out, that if we delete constraints (3) and the variables z_{cr} from the model, then the running time improves greatly. That means, when only the time connections (y_{crt} variables) are maximized, then the stream assignment is nearly optimally solvable in short time. In most cases this also leads to a surprisingly good schedule. But as already mentioned, a maximal number of time connections does not exclude that the sessions of one cluster can be widely distributed over the set of rooms. This may result in long walking distances between parallel sessions of equal clusters. Our solutions were taken from the relaxed model.

6.3 Cluster swap heuristic

A solution of the stream assignment MIP yields an assignment of all sessions to streams. In the model, every stream was associated with a room to ensure the compliance with room capacities, that means it holds $a_s \leq u_r$ for each assignment of a session s to a room r . Let us consider a time slot t , where two sessions $s_1, s_2 \in S_c$ of cluster c are scheduled. Assume that the number of attendees of session s_1 is smaller than the number of s_2 , that means $a_{s_1} < a_{s_2}$. Even though all capacity constraints might be satisfied, it could be that s_1 is scheduled to a room that has greater capacity than the room of s_2 . This case is not restricted by the model formulation, as long as the capacity constraints remain feasible. From the point of view of a participant, it might be confusing why an apparently smaller session is scheduled in a large room, while another session of the same cluster is at nearly maxed out capacity. This case is supposed to be excluded.

A simple heuristic restores this property for the schedule by swapping the relevant sessions in the correct order according to their capacities or capacity demand respectively, see below.

Example

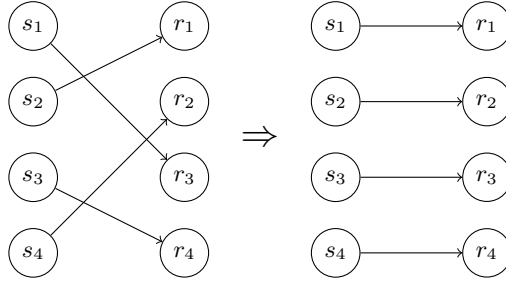


Figure 23: Application of the swap heuristic

In the figure above, let s_1, \dots, s_4 be sessions of cluster c that are scheduled into rooms r_1, \dots, r_4 at some time slot t . We assume that $a_{s_1} \leq \dots \leq a_{s_4}$ and $u_{r_1} \leq \dots \leq u_{r_4}$. The left figure shows the assignment that is obtained from the stream assignment, while the right side shows the allocation after the swap heuristic, where the sessions of S_c are sorted in correct order with respect to their capacity demand.

We also tried a formulation of this restriction by means of additional linear inequalities in the stream assignment model, but the computation time increased drastically due to this implementation. Then we decided to use this simple heuristic. As the procedure swaps only between sessions of the same cluster, the objective value of the previous MIP model would also remain the same, since for each cluster no time connection or room is lost or added due to the swaps. Also no further constraints are violated.

For the input of the heuristic, we need a matrix M , where each row corresponds to a room $r \in R$ and each column to a time slot $t \in T$. The entries are the sessions $s_{rt} \in S$ that are scheduled in room r and time slot t . The matrix is given from a solution of the stream assignment, where each stream was associated with a room. There are also given vectors u and a , that contain the room capacities of every room or the number of attendees of every session respectively. The heuristic swaps the sessions within the matrix. In particular, it swaps only sessions of the same cluster at the same time slot. Possible swaps are performed for every cluster $c \in C$.

The first loop iterates over all time slots $t \in T$. For time slot t , the sets S_c and R_c are defined for every cluster c . Every session s that is scheduled at t , is added to S_c , while the room r , where s is scheduled, is added to R_c (6-10). Next, we iterate over all clusters $c \in C$ and apply the *sort* function to S_c and R_c . The sets are sorted by the number of attendees or the room capacities respectively. The method *greedyAssign* performs a greedy assignment of the sorted sessions $s \in S'_c$ to the sorted rooms $r \in R'_c$ in the order of the sorting. The assignment matrix is iteratively updated, until swaps are executed for all time slots.

The *sort* and *greedyAssign* methods, obviously, run in polynomial time, so the entire algorithm is polynomial for the given input sizes. From our observations, the algorithm

Algorithm 1 Cluster Swap Heuristic

Require: assignment matrix M with entries $s_{rt} \in S, r \in R, t \in T$

Ensure: swapped matrix M'

```
1: procedure CLUSTERSWAP( $M$ )
2:    $M' \leftarrow M$ 
3:   for all  $t \in T$  do
4:      $R_c \leftarrow \emptyset \quad \forall c \in C$ 
5:      $S_c \leftarrow \emptyset \quad \forall c \in C$ 
6:     for all  $r \in R$  do
7:        $s \leftarrow s_{rt}$ 
8:        $c \leftarrow c_s$ 
9:        $R_c \leftarrow \{r\}$ 
10:       $S_c \leftarrow \{s\}$ 
11:    end for
12:    for all  $c \in C$  do
13:       $S'_c \leftarrow \text{sort}(S_c, a_s)$ 
14:       $R'_c \leftarrow \text{sort}(R_c, u_r)$ 
15:       $M' \leftarrow \text{greedyAssign}(M', t, R'_c, S'_c)$ 
16:    end for
17:  end for
18: return  $M'$ 
19: end procedure
```

has almost no effect on the computation time.

7 Room assignment

This section deals with the last optimization step, the *room assignment*. From the solutions of the previous optimization models, we obtained the time slot and the stream of every session. That means we know how the sequences of sessions that appear in the same room, look like. Finally, each stream must be assigned to room that is available for the conference. Walking distances between streams with similar content should be minimized.

Regarding this as a separate subproblem, is definitely important for large scale conferences. The sessions of the ISMP 2012 occupied 40 different rooms that were distributed over two different buildings and 10 floors. Since many participants change to other talks within a session, they have only five minutes to change the room until the next talk begins. Conference attendees will feel uncomfortable when they have to rush between the talks, what is caused by too long walking distances. Thus, our approach is to allocate 'similar' streams to 'nearby' rooms.

In the following we will explain the concepts of the regarded 'distance' between the rooms and the 'similarity' between two streams. Furthermore, we will discuss the associ-

ated combinatorial problem and also present a Mixed-Integer-Programming formulation to solve it.

7.1 Room distances

For every pair (i, j) of rooms, we define a value δ_{ij} that represents the distance between the rooms i and j . One possibility is to make use of door-to-door distances, using coordinates of every room. A distance measure, for example *Euclidean*- or *Manhattan*-distances, can be used for this. In particular, the term 'distances' does not only have to refer to pure distances. It could also indicate a degree of displeasure to change between two rooms. For instance, penalty values for going up a stairway, taking the elevator, the crossing of a street, changing the building or taking crowded corridors can be mentioned here. A good choice of the distance values depends on the specific room map for the conference.

At ISMP 2012 we used 3-dimensional room coordinates (x_r, y_r, z_r) , where x_r, y_r denote the coordinates in the plane and z_r the floor of a room r . We also introduced penalty values λ_1, λ_2 for changing the floor or the building respectively. The corridor ways and the building structure of TU-Berlin is similar to *Manhattan*-distances, so the distances δ_{ij} between two rooms (i, j) were defined as:

$$\delta_{ij} = \begin{cases} |x_i - x_j| + |y_i - y_j| + \lambda_1 |z_i - z_j|, & \text{if } (i, j) \text{ in the same building} \\ |x_i - x_j| + |y_i - y_j| + \lambda_1 |z_i + z_j| + \lambda_2, & \text{otherwise} \end{cases}$$

Note that a participant has to overcome the z -coordinate twice when he changes the building. This is an idea on how those values can be determined. Naturally, we will restrict to $\delta_{ij} \geq 0$.

7.2 Similarity values

Regarding room changes, streams with similar content should be assigned to nearby rooms. Therefore, we define a similarity value for each pair of streams which depends on the included sessions. We assume that participants either choose sessions with the same or a similar cluster for a room change. Thus, we will define a similarity value between clusters first, until we construct a similarity value between streams on the basis of that definition.

7.2.1 Cluster similarity

We consider two clusters as 'similar', if there are overlaps in the topic or the content. The higher the overlap the higher is the similarity. For the definition it is useful to create superordinate attributes that give an idea of the possible contents. From the number of matching attributes between two clusters, we calculate a similarity coefficient. Usually, similarity coefficients in a range from $[0, 1]$ are used.

For example, let A and B denote the set of attributes that belong to two clusters. Often used similarity coefficients are the following, see [15]:

Jaccard index:

$$J(A, B) := \frac{|A \cap B|}{|A \cup B|}$$

Dice-Sørensen index:

$$DS(A, B) := \frac{2|A \cap B|}{|A| + |B|}$$

Tversky index: coefficients $\alpha, \beta \geq 0$:

$$T(A, B, \alpha, \beta) := \frac{|A \cap B|}{|A \cap B| + \alpha|A - B| + \beta|B - A|}$$

Note that $T(A, B, 0.5, 0.5) = DS(A, B)$. At ISMP we referred to the following tags or attributes respectively: *Theory, Computation, Application, Discrete, Nonlinear, Stochastic*. For each cluster it was decided whether it matches one of these attributes or not.

Example

attribute/set	A	B	$A \cap B$	$A \cup B$	$A - B$	$B - A$
Theory	1	1	1	1	0	0
Computation	1	1	1	1	0	0
Application	0	1	0	1	0	1
Discrete	1	0	0	1	1	0
Nonlinear	0	1	0	1	0	1
Stochastic	0	0	0	0	0	0
Σ	3	4	2	5	1	2

The table shows the set of attributes A or B that correspond to two exemplary clusters. It show whether it matches one of these attributes ($= 1$) or not ($= 0$). From the previous definitions, the similarity coefficients for this example would be:

$$\begin{aligned} J(A, B) &= 0.4 \\ DS(A, B) &= 0.571 \\ T(A, B, 1, 0.5) &= 0.5 \end{aligned}$$

Modifications with fractional values in the table could be used as well. Again, this is just an approach, how similarities between clusters could be defined. This is a fast and easy method to implement in practice. Dependent on the choice of the attributes, it also yields a good representation of the similarities between two clusters. For the upcoming procedure we denote the similarity value for a cluster pair (c, c') by $\beta_{cc'}$. We further restrict to $\beta_{cc'} \geq 0$.

7.2.2 Stream similarity

In this section, we define the similarity values between each pair of streams. From the definition, each stream is a sequence of sessions (s_1, \dots, s_T) that occur successively in the same room. Let i and j denote two streams with sessions s_{it} and s_{jt} at time slot t . Furthermore, let c_{it} denote the cluster of session s_{it} . The main goal is to minimize the walking distances between parallel sessions. Secondly, we also want sessions at successive time slots to take place in nearby rooms. Therefore, we have to consider all relations between sessions at parallel and successive time slots. Below there is shown a figure with the regarded relations.

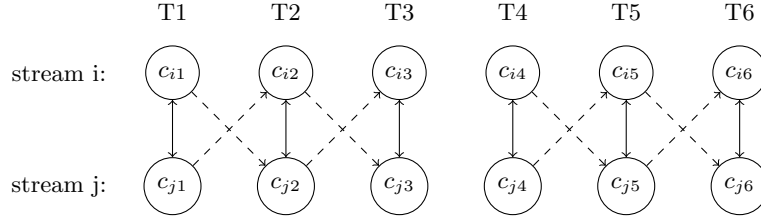


Figure 24: Room changing relations between two streams

Note that we do not need to consider the relations between different days (where $d_t \neq d_{t+1}$). Let $T' = T - \max(T)$, then the similarity value σ_{ij} between streams i and j , is defined as weighted sum over the cluster similarities $\beta_{cc'}$ of the regarded relations:

$$\sigma_{ij} = \sum_{t \in T'} w_1 \underbrace{(\beta_{c_{it}c_{jt}} + \beta_{c_{jt}c_{it}})}_{\text{parallel relations}} + \sum_{t \in T': d_t = d_{t+1}} w_2 \underbrace{(\beta_{c_{it}c_{j,t+1}} + \beta_{c_{i,t+1}c_{jt}})}_{\text{relations at successive time slots}}$$

The additional weight coefficients $w_1, w_2 \geq 0$ are applied to distinguish between the importance of parallel and successive relations. Parallel room changes appear more important, because the breaks in parallel sessions are much shorter. Our scenarios use weights of $w_1 = 1$ and $w_2 = 0.5$.

7.3 The room assignment problem

Finally, the streams are assigned to rooms such that the distance between similar streams is minimized. We denote this problem as the *Room Assignment Problem* (RAP). In the next sections we introduce the associated combinatorial problem of the RAP. Furthermore, we give a Mixed-Integer-Programming formulation of the RAP and discuss the complexity of that problem. In the last subsection we present an approach to reduce the problem size to compute instances that are comparable to the ISMP.

7.3.1 Room assignment problem and MIP formulation

Let I denote the set of streams that is obtained by a solution of the stream assignment. The set of sessions that are contained in stream $i \in I$, is defined by S_i . We denote

$a_i^{max} = \max_{s \in S_i} (a_s)$ as the largest session in stream i . An assignment of stream i to room r is feasible, if and only if $a_i^{max} \leq u_r$ because each session of the stream must comply with the room capacity. To deduce a combinatorial model, every feasible assignment (i, r) of a stream i to room r is modeled as separate vertex in a vertex set V , which is given by:

$$V = \{(i, r) \mid i \in I, r \in R, a_i^{max} \leq u_r\}$$

Between each pair $(i_1, r_1), (i_2, r_2) \in V$ of assignments, we have to consider the distance $\delta_{r_1 r_2}$ and the similarity $\sigma_{i_1 i_2}$ of the rooms and streams respectively. Therefore, the relations between two assignments $u, v \in V$ are modeled as undirected edges. Since each stream is allocated exactly once, as well as each room is occupied by one stream, the edges between assignments of equal streams or rooms are excluded. The edge set E reads as follows:

$$E = \{(i_1, r_1, i_2, r_2) \mid (i_1, r_1), (i_2, r_2) \in V, i_1 < i_2, r_1 \neq r_2\}$$

The induced undirected graph is defined as $G = (V, E)$.

Definition 21. Let $G = (V, E)$ be an undirected graph with vertex set V and edge set E . We say G is complete, if each pair of vertices in V is adjacent. A complete subgraph of G is called a clique. Furthermore, a clique with the largest possible number of vertices is called a maximum clique.

Definition 22. Given a graph $G = (V, E)$, a stable set $S \subseteq V$ is a subset of vertices, where each pair of vertices in S is non-adjacent.

The graph G of the room assignment problem has a special structure. Let $V_i = \{(i, r) \mid (i, r) \in V\}$ and $V_r = \{(i, r) \mid (i, r) \in V\}$ be the set of assignments that correspond to a stream i or room r respectively. By the definition of the edge set, each V_i and V_r is a stable set, because no edge occurs between equal streams and rooms. Consequently, for I and R , there exists a decomposition of G into disjoint stable sets with $V = \bigcup_{i \in I} \{V_i\}$ and $V = \bigcup_{r \in R} \{V_r\}$, since each vertex is covered by one of these sets.

The problem of the room assignment consists in finding a selection of assignments (vertices) such that each stream and each room belongs to exactly one assignment. Whenever two assignments are selected, the edge between them must be selected as well. Assume that each edge has a certain cost, depending on the distance and the similarity values between the incident assignments, and let $b = |I| = |R|$ denote the number of streams or rooms respectively. We can describe the problem as follows:

Find a clique of size b in G such that the total edge cost of the clique is minimal.

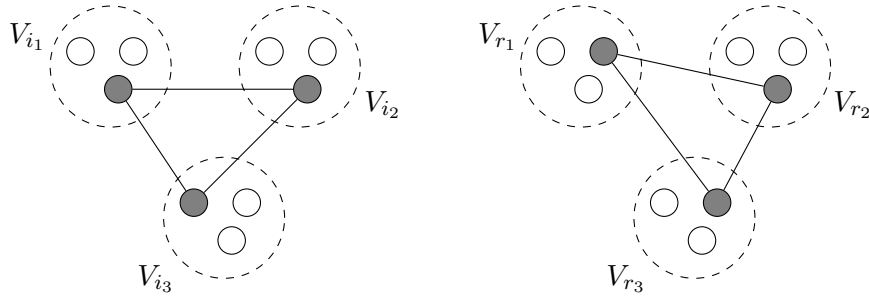


Figure 25: Two distinct decompositions of G with $|I| = |R| = 3$ into stable sets and the selected clique

Note that a feasible solution of the stream assignment implies the existence of a clique of size b because room capacities were considered explicitly. In addition, the problem description already excludes multiple assignments of streams and rooms. From the definition of E , there is a partition into b stable sets V_i or V_r respectively. Since there is one stable set that covers all vertices of a stream or room respectively, and a clique must be complete, multiple assignments of rooms or streams are excluded. Moreover, each of the partitions contains exactly b stable sets, therefore, one vertex from each stable set must be selected. In particular, b is also the size of the largest clique in G because an additional vertex would necessarily contain a stream and room that is already among the b selected vertices. Thus, each clique of size b is also a maximum clique.

Mixed Integer Program

We define the binary variables x_v to be 1, if an assignment $v = (i, r) \in V$ of a stream i to room r is selected, and 0 otherwise. Furthermore, the binary variable y_{uv} is 1, if the edge (u, v) between two assignments $u, v \in V$ is selected. We denote the cost of each edge $(u, v) = (i_1, r_1, i_2, r_2)$ by $\sigma_{i_1 i_2} \delta_{r_1 r_2} = \lambda_{uv}$. Here, σ_{ij} is the similarity value between streams i_1, i_2 and $\delta_{r_1 r_2}$ the distance value between rooms r_1, r_2 , as introduced in the previous sections. The total cost of the selected edges is minimized. Therefore, it is more valuable to choose greater distances for lower similarities and shorter distances for higher similarity values. The Mixed-Integer-Program reads as follows:

$$\begin{aligned}
\min \quad & \sum_{(u,v) \in E} \lambda_{uv} y_{uv} \\
\text{s.t.} \quad & \sum_{v \in V} x_v = b \tag{1} \\
& \sum_{v \in V_i} x_v \leq 1 \quad \forall i \in I \tag{2} \\
& \sum_{v \in V_r} x_v \leq 1 \quad \forall r \in R \tag{3} \\
& y_{uv} - x_u \leq 0 \quad \forall (u,v) \in E \tag{4} \\
& y_{uv} - x_v \leq 0 \quad \forall (u,v) \in E \tag{5} \\
& x_u + x_v - y_{uv} \leq 1 \quad \forall (u,v) \in E \tag{6} \\
& (b-1)x_v = \sum_{u \in \delta(v)} y_{uv} \quad \forall v \in V \tag{7} \\
& x_v \in \{0, 1\} \quad \forall v \in V \\
& y_{uv} \in \{0, 1\} \quad \forall (u,v) \in E
\end{aligned}$$

Note that the integrality condition for y_{uv} can be relaxed to $0 \leq y_{uv} \leq 1$ for all $(u, v) \in E$. The objective function minimizes the total cost, which is the sum over the cost of all selected edges. Inequality (1) says, that exactly b vertices must be chosen. It must also be ensured that at most one stream and room is chosen from the selected vertices that are given by constraints (2) and (3). The next inequalities (4) and (5) indicate, that whenever an edge between two assignments u and v is selected, then also u and v must be selected. Similarly, constraint (6) yields that whenever two assignments u and v are chosen, then also the edge (u, v) must be selected. The inequalities of (7) state that whenever an assignment v is chosen, then there must exist $b-1$ incident edges to v because the selected clique has size b .

As a special remark we have to mention that inequalities (2) and (3) are redundant in this MIP formulation. Summing up (7), together with (4) yields

$$b - 1 = (b - 1) \sum_{v \in V_i} x_v = \sum_{v \in V_i} \sum_{u \in \delta(v)} y_{uv} \leq \sum_{v \in V_i} \sum_{u \in \delta(v)} x_u$$

If we subtract this inequality from (1), we get $\sum_{u \in V_i} x_u \leq 1$, that is (2). Constraint (3) is obtained analogous. These inequalities will be needed later, so they are already introduced with this model.

7.3.2 Previous work and related problems

Since the room assignment problem is quite special for itself, we generalize this problem. The considered problem is to find a clique of size b with minimal edge weight. We could also relax this formulation to find a clique of size $\geq b$ that has minimal edge weight

because, as mentioned, the largest clique in the room assignment graph already has size b . We will also refer to complete graphs for the generalization. Therefore, we reformulate the problem to: *Given an undirected complete graph $G = (V, E)$ with edge cost λ_e , find a clique of size $\geq b$ that has minimal edge cost.*

The problem of finding minimal edge weighted cliques with minimum clique size requirement was rather studied for one cliques only, but for clique partitionings, see Mitchell and Ji [23]. We can also mention the closely related *Maximum Edge Weighted Clique Problem (MEWCP)* that was extensively studied in the recent years, see [5],[6],[7] and [19]. The problem can be stated as follows: given an undirected complete graph, find a clique of size $\leq b$ that has maximal edge weight.

It is possible to reduce the RAP to MEWCP by changing the edge weights to $\lambda'_e = \Lambda - \lambda_e > 0$, where Λ is a large constant. If we solve the induced MEWCP, we would also obtain the maximal clique of size b , as the considered edge weights are strictly positive. If \mathcal{C} would be such an optimal clique, then

$$\begin{aligned} \sum_{e \in E(\mathcal{C})} \lambda'_e = \sum_{e \in E(\mathcal{C})} (\Lambda - \lambda_e) = b(b-1)/2 \cdot \Lambda - \sum_{e \in E(\mathcal{C})} \lambda_e \text{ is maximal} \\ \Leftrightarrow \sum_{e \in E(\mathcal{C})} \lambda_e \text{ is minimal} \end{aligned}$$

In particular, every valid inequality of the reduced MEWCP problem is also valid for the original RAP, as no possible clique is excluded.

However, the MEWCP is \mathcal{NP} -hard, as the *Maximum Clique Problem (MCP)* emerges as a special case, that was also shown to be \mathcal{NP} -hard. The MCP is to find a clique in a given graph that has maximum cardinality.

In contrast to the general MCP, the problem of finding a maximum clique in the graph of the RAP is easy to obtain, since a maximum clique represents a feasible assignment of streams to rooms. That means, finding a maximum clique in the RAP graph corresponds to an assignment problem what can be computed in polynomial time. But the additional consideration of edge weights and the lower bound on the clique size makes this problem more difficult to solve.

A more generalized option to regard the MEWCP is to consider the so-called *Boolean Quadric Polytope*, that describes the induced integer polytope of the general MEWCP formulation without clique size requirement. Studies on this polytopal structure have been made in [18] and [20].

7.3.3 Improving the computation time of the RAP

From our computational experience, the room assignment problem is very hard to solve. For our problem instances the computation time exceeds the limit that we consider to be useful for practical implementation. If all edges between all feasible assignments would be considered, then the problem becomes too large to solve. Our approach is to reduce the problem size to make progress, also with hard instances.

Again, the main idea is to delete edges of the underlying graph G to a moderate number. Especially, we delete edges that we do not consider as 'important' for our problem. Every selected edge $(u, v) = (i, r, j, r')$ between assignments $u = (i, r)$ and

$v = (j, r')$ contributes to the objective function with cost $\lambda_{uv} = \lambda_{irjr'} = \sigma_{ij}\delta_{rr'}$. The first observation that we make is that we can eliminate edges with $\sigma_{uv} = 0$, because they will not even affect the objective value, if we would consider them or not. In general this is the case, if the similarity value σ_{ij} between streams i and j is equal to zero, that means both streams have nothing in common with respect to their content. Consequently our approach is to eliminate edges that do not satisfy a certain value of similarity. As we are focusing on the assignment of similar streams to nearby rooms, this seems to be an useful approach. Therefore, let the parameter σ define a lower bound on the similarity values up to which edges are considered in the model. We redefine the edge set as follows:

$$E' = \{(i, r, j, r') \mid (i, r), (j, r') \in V, i < j, r \neq r', \sigma_{ij} \geq \sigma\} \subseteq E$$

The new graph $G' = (V, E')$ consists of the same assignments $v \in V$, but with reduced edge set E' . All variables $y_{uv} = y_{irjr'}$ with $\sigma_{ij} < \sigma$ are eliminated, what reduces the problem size. The relaxed optimization model of RAP is applied to the sparse graph G' .

Remark 23. *The elimination of edges due to σ will violate the clique structure in general. That means we do not know whether each vertex v has exactly $b - 1$ incident edges in E' . As we still want to select b vertices, constraint (5) is not valid for G' and must be deleted from the model formulation. Every other inequality is still valid for G' .*

In particular, inequalities (2) and (3) are not redundant anymore due to this relaxation. In contrast, the deleted constraints (5) were 'good' inequalities and contributed to a good performance with respect to the computation time. We introduce valid inequalities that are similar to the deleted ones. For every assignment $v \in V$ we define $b_v = b_{ir} = |\{j \mid j \in I \setminus \{i\}, \sigma_{ij} \geq \sigma\}|$. In this formulation, b_v is the number of adjacent assignments of vertex v in E' . Then the following inequalities are valid for the reduced model and replace inequalities (5) of the previous model:

$$b_v x_v \leq \sum_{(u,v) \in E'} y_{uv} \quad \forall v \in V \quad (5')$$

8 Conclusion

In this thesis we provide a framework to compute conference schedules that respect a wide sets of constraints. We also introduced an approach to the robust optimization of the conference schedule, which relates to the uncertain number of attendees and the compliance with room capacities. We gained new insights into the combinatorial structures of the problems that occur during the planning of a conference.

In practice, the optimization models contributed significantly to the creation of a conference schedule that was used at ISMP 2012. Especially, the compliance with hundreds of individual scheduling requests was a great challenge. The computed conference schedules, see 'computational results', look appropriate to be used at real conferences. They provide us with the information, if the room capacities will be sufficient to accommodate all conference participants. In this context, Mixed-Integer-Programming yields a perfect application to solve the difficulties of conference scheduling.

But the model has also some weaknesses concerning the regarded problem data. The approach of computing a robust schedule highly depends on the used attractivity coefficients α_s for each session. These coefficients were determined due to attractivity indicators. A remaining problem is to decide whether the used attractivity indicators yield a good representation of an average attendee behavior. That means we actually do not know which sessions are selected by a particular attendee. From our experience at ISMP 2012, the main attractivity indicator for a session is the popularity of the speaker, since famous speakers will generally have a great audience. It will be important for future conferences to obtain this particular attractivity information for each session. It would also be helpful to integrate insider knowledge for a more realistic estimation, because not every speaker must be known by the organizers.

For further research on this topic, it would also be interesting if there are efficient methods that produce a schedule without the distinction of a time- and room assignment. As we considered mainly MIP models for the different subproblems, it would be interesting to develop an algorithmic approach for that problem. Moreover, it would be fascinating to deal with the question, if there is a schedule that can optimally satisfy the personal session preferences of every participant, when this information would be known in advance. This could also be regarded from a game theoretic perspective. The leads to the individualization of the schedule that respects individual preferences in a best possible way.

Special thanks to Prof. Dr. Ralf Borndörfer for his helpfulness and patience for this thesis.

9 Computational results

All presented results were computed on a Intel Core 2 Extreme X9650 3.00GHz with Gurobi Optimizer version 5.5.0.

9.1 Combination of Talks to Sessions

- random similarities $\sigma_{kl} \in [0, 1]$
- weight coefficient $\lambda = 0.75$ for both models
- $D = 3, d_k = 1 \quad \forall k \in K$
- $|K|$ denotes the number of talks
- $|S|$ denotes the number of computed sessions
- σ is the lower bound on the considered similarity relations, that means only edges with $\sigma_{kl} \geq \sigma$ are in the problem graph (clique partitioning only)
- $|E|$ is the number of used edges in the underlying problem graph (inclusive reduction)
- all instances were computed with a constant time limit of 600 seconds
- both models were tested for the same instances, see Section 4.2 'computational experience'
- instance with $|K| = 1740$ and max. $|S| = 630$ refers to the ISMP proportion

Clique partitioning

instance					computation			solution		
$ K $	min. $ S $	max. $ S $	σ	$ E $	# vars	# constr	% gap	$ S $	idle time	avg similarity
500	167	200	0	5,106	8,109	105,091	6.83	176	28	0.843
1000	334	400	0.6	8,435	14,438	149,592	3.81	343	29	0.901
1500	500	600	0.75	11,830	20,833	195,429	2.53	509	27	0.923
1740	580	630	0.8	12,618	23,061	193,757	2.12	592	36	0.930
2000	667	750	0.85	12,230	24,233	161,547	1.93	687	61	0.934

Capacitated Facility Location

instance				computation			solution		
$ K $	min. $ S $	max. $ S $	$ E $	# vars	# constr	% gap	$ S $	idle time	avg similarity
500	167	200	5,606	6,610	7,111	1.76	176	28	0.901
1000	334	400	21,937	23,941	24,942	2.29	343	29	0.951
1500	500	600	48,353	51,357	52,858	1.65	509	27	0.966
1740	580	630	65,024	68,508	70,249	1.28	588	24	0.969
2000	667	750	85,211	89,215	91,216	1.13	675	25	0.974

9.2 Capacity labels

- random generated clusters and attractivity values
- instance was computed with and without cluster labeling constraints
- $n = 1000$ attendees

Schedule without capacity labels

- cluster time connections: 398

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	1	1	1	1	1	1	1	1	1	11	11	11	11	11	6
H 0107	144	11	11	11	11	11	11	11	11	11	11	11	11	22	11	8
H 0110	198	5	5	5	5	5	5	5	5	5	5	5	6	12	12	12
H 0111	99	3	3	3	3	3	3	3	3	3	15	2	17	17	17	17
H 0112	99	18	9	9	9	3	3	3	3	3	3	3	3	3	3	3
H 1012	262	13	22	22	22	22	15	15	15	4	8	8	8	20	8	8
H 1028	231	17	17	8	8	8	8	8	8	8	8	1	13	13	13	13
H 1029	41	6	6	11	22	8	8	8	8	8	8	8	8	8	8	5
H 1058	263	4	4	18	18	18	18	18	18	18	7	21	4	4	4	4
H 2013	261	9	9	7	14	14	14	14	14	14	14	14	14	14	14	14
H 2032	236	12	12	13	13	7	7	7	7	7	7	7	7	7	7	7
H 2033	67	23	22	6	6	6	2	2	14	14	2	2	2	2	2	2
H 2035	30	2	2	2	2	2	19	19	19		18	18		16	16	16
H 2036	45	10	10	2	2	2	4	4	4	17	17	17	17	17	20	20
H 2038	50	16	21	16	15	15	13	13	17	12	19	19	19	19	19	19
H 2051	30	9	13	15	1	1	1	17	2	2	2	8	8	8		
H 2053	261	15	15	20	20	20	20	11	11	19	19	19	19	19	19	19
H 3002	40	19	19	19	19		6	6	6	6	4	4	23	23	23	14
H 3003A	20	8	8	17	12	12	12		15			18				
H 3004	50	4	6	6	6	5	6	6	6	6	6	6	19	1	1	1
H 3005	80	19	19	19	19	19	19	19	19	19	10	10	10	10	10	10
H 3008	30	14	14	14	5	16	16		17	17	17	17				
H 3010	388	20	20	20	11	22	22	22	22	22	22	22	22	22	22	22
H 3012	40	7	7	4	4		20	20	20	11		7	7		17	17
H 3013	40	11	13	13	13	13	22	22	22	22	18	15	15	15	15	15
H 3021	30	19	19	19	7		4	4	12	9	19	19	21	21	24	9
H 3027	80	1	1	1	17	17	17	17	17	16	16	16	16	16	16	16
H 3503	140	2	24	24	4	4	15	15	15	15	15	15	15	15	15	15
MA 004	235	15	15	15	10	12	12	12	4	4	6	6	6	20	4	4
MA 005	235	16	16	16	16	6	10	10	10	1	1	1	1	1	1	1
MA 041	152	23	23	23	23	23	23	13	13	13	9	9	9	9	9	9
MA 042	140	13	8	8	8	4	9	9	9	20	20	20	18	12	12	23
MA 043	152	21	21	21	21	21	21	12	23	23	23	23	18	18	21	21
MA 141	70	24	24	24	24	24	24	24	24	24	24	24	24	24	14	24
MA 144	68	3	3	3	19	19	2	16	16	16	5	5	5	5	5	24
MA 313	40	17	17	17	17	17	17	16	21	21	21	21	21	14	2	2
MA 376	32	6			3	13	13	23	23	23	23	23	23	23	23	23
MA 415	50	14	14	14	14	14	23	24	24	24	24	14	2	2	18	18
MA 549	32	20	20		20	20	14	14	12	12	12	12	12	4		
MA 550	32	24	16	9	9	9	5				13	13	5	6	6	
MA 649	32	8	11	10	18		21	21	16	10	21	24	24	24	24	11
MA 650	32	22	18	12	12	11	11	23	20			20	20		20	20

Same instance with capacity labels

- cluster time connections: 508 \approx 27.6% increase

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	22	22	22	22	22	22	12	12	12	12	12	12	12	12	12
H 0107	144	22	22	22	22	11	14	14	15	15	22	22	22	22	22	22
H 0110	198	18	18	18	18	18	18	18	18	18	18	18	18	18	18	23
H 0111	99	7	7	7	7	7	7	7	7	7	7	7	7	7	7	20
H 0112	99	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
H 1012	262	12	12	12	15	4	15	16	16	16	16	16	16	16	16	16
H 1028	231	8	8	8	8	8	8	8	8	8	8	16	16	16	16	16
H 1029	41	8	8	8	8	8	8	8	8	8	8	8	8	21	21	21
H 1058	263	24	24	24	6	6	6	6	6	6	6	6	6			1
H 2013	261	17	17	17	17	17	17	17	17	17	9	9	9	9	9	9
H 2032	236	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
H 2033	67	9	9	9	9	9	9	3	3	3	8	8	8	8	8	8
H 2035	30	13	13	13		22					12	12	12			7
H 2036	45	1	1	1	1	1	1	4	4	4	4	4	4	4	4	4
H 2038	50	11	11	11	17	17	13	6	6	6	6	6	6	6	6	6
H 2051	30	6	6	6	23	23	23	23	23	23	23	23	23	23	23	23
H 2053	261	14	14	14	23	23	23	23	23	23	11	11	23	23	23	23
H 3002	40	19	19	19	19	19	17	17	17	17	17	17	17	17	17	17
H 3003A	20	15	15	15											14	14
H 3004	50	23	23	23	23	19	19	19	19	19	3	3	3	17	17	17
H 3005	80	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
H 3008	30	16	16	16	16	16	16	19	19	19	19	19	19	19	19	19
H 3010	388	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
H 3012	40	6	6	6	24	24	24	11	11	11	15	15	15	15	15	15
H 3013	40	20	20	20	20	20	20	16	16	16	16		11	11	11	11
H 3021	30				18	18	18	13	13	13	17	17	17	6	6	6
H 3027	80	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 3503	140	20	20	20	20	20	20	20	20	20	20	20	20	20	2	2
MA 004	235	10	10	10	10	10	10	10	10	10	10	10	10	1	18	18
MA 005	235	21	21	21	21	21	21	21	21	21	14	14	14	14	14	14
MA 041	152	19	19	19	2	2	2	2	2	2	2	2	2	2	2	2
MA 042	140	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
MA 043	152	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
MA 141	70	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
MA 144	68	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MA 313	40	2	2	2	2	2	2	1	1	1	7	7	7	22	22	22
MA 376	32	23		9	9	9	9	9			24	24	24	24	24	24
MA 415	50	5	5	5	14	14	14	14	14	14	14	14	14	14	10	10
MA 549	32	17	17	17	5	5	5				21	8	8	8	8	8
MA 550	32	14	14	14	14	14		2	2	2	2	2	2	3	3	3
MA 649	32	12	12	12	12	12	12	22	22	22	20	20	20	20	20	20
MA 650	32	21	21	21	21			9	9	9	21	21	21		12	12

9.3 ISMP problem instance

The ISMP 2012 problem instance was computed with the following parameters:

- 595 sessions from 24 different clusters
- 5 days, each with 3 time slots \Rightarrow 15 time slots
- 42 rooms
- 116 interval requests
- 45 no overlap requests
- 85 hard precedence requests
- $\epsilon = 0.05$ (robustness approximation factor)
- room assignment: consider similarities $\sigma_{ij} \geq \sigma = 0.5 \cdot \max_{(i,j) \in E}(\sigma_{ij})$

Cluster list	# sessions
Approximation and Online Algorithms	16
Combinatorial Optimization	85
Complementarity and Variational Inequalities	20
Conic Programming	30
Constraint Programming	10
Derivative-free and Simulation-based Optimization	11
Finance and Economics	19
Game Theory	24
Global Optimization	18
Implementations and Software	18
Integer and Mixed-Integer-Programming	46
Life Sciences and Healthcare	13
Logistics, Traffic and Transportation	29
Mixed-Integer-Nonlinear-Programming	18
Multi-objective Optimization	15
Nonlinear Programming	41
Non-smooth Optimization	14
Optimization in Energy Systems	30
PDE-constrained Optimization	21
Robust Optimization	16
Sparse Optimization and compressed Sensing	14
Stochastic Optimization	37
Telecommunications and Networks	20
Variational Analysis	30

Multiple attendances of single persons (sessions of each person must not overlap!)

# sessions to be present	2	3	4	5	6	7	8
# persons	152	28	8	4	2	1	1

The problem instance was computed for different numbers of attendees n to simulate the effect of hard capacity restrictions on the final schedule. We used a constant time limit of 600 seconds for each model. The *robust view* shows the probabilities that the room capacities will be sufficient (not be exceeded) for each assignment. This is calculated on the basis of the probabilities that we have introduced in Definition 15.

9.3.1 Instance 1: $n = 500$

- $\gamma_s = 0.95$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Time assignment	10,779	5,815	600	0.01
Stream assignment	69,974	91,945	315.76	optimal
Room assignment	47,665	138,926	395.4	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 1012	262	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
H 1028	231			20		2		2	2	2	2	2	2	2	2	
H 1029	41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
H 1058	263	15	15	15	15	15	15	15	15	15	15	15	15	15	15	
H 2013	261	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
H 2032	236	21	21		21	21	21	21	21	21	21	21	21	21	21	
H 2033	67	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2035	30	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2036	45	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2038	50	12	12	12	12	12	12	12								
H 2051	30	9	9	9	9	9	9	9	9	9	9	9	9	9	9	
H 2053	261	17	17	17	17	17	17	17	1	17	17	17	17	17	17	
H 3002	40	5	5	5	10	10	10	10	10	10	10	10	10	10	10	
H 3003A	20	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3004	50	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3005	80	16	16	16	16	16	16	16	16	16	11	16		16	23	
H 3008	30			22	22	22	22	22	22	22	22	22	22	22	22	
H 3010	388	14	14	14	14	14	14	14	14	14	14	14	14	7	7	
H 3012	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 3013	40	22	22	22	22	22	22	22	22	22		6		6	6	
H 3021	30	23	23	23	23	23	23	23	23	23	23	23	23	23	23	
H 3027	80	9	9	9				5	5	5	5	5	5		5	
H 3503	140	10	10	10	10	10	10	7	23	3	23	23	23	3		
MA 004	235	13	13		13	13	13	13	13	13	13	13	13	13	13	
MA 005	235	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
MA 041	152	8	8	8							14	14	14	14	14	
MA 042	140	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
MA 043	152	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
MA 141	70	19	19	19	19	19	19	8	8	8	8	8	8	8	8	
MA 144	68	3	3	3	19	19	19	19	19	19	19	19	19	19	19	
MA 313	40	6	6	6	6	6	6		6		12	12	12	12	12	
MA 376	32	8	8	8	8	8	8	8	8	8	8	8	8	19	19	
MA 415	50	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
MA 549	32	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
MA 550	32	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	23	24	23	23	23	23	23	23	23	32	23	34	19	19	19
H 1058	263	13	18	18	13	18	18	18	18	18	18	18	18	18	18	18
H 1012	262	28	28	28	28	28	28	20	28	28	28	28	28	28	28	28
H 2053	261	24	24	24	24	17	24	45	42	17	24	24	24	24	24	24
H 2013	261	18	18	18	18	18	18	18	18	18	18	18	18	18	18	13
H 2032	236	17	24		24	24	24	24	24	24	24	24	24	24	24	24
MA 005	235	23	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 004	235	18	18		18	18	18	18	18	18	13	18	18	18	18	18
H 1028	231			20		17		13	17	17	17	17	17	17	17	17
H 0110	198	13	17	17	17	17	17	13	13	17	17	13	17	17	17	17
MA 041	152	22	22	22							16	16	16	23	23	23
MA 043	152	17	12	17	17	17	17	12	17	17	17	17	17	17	17	17
H 0107	144	17	17	17	17	17	17	17	17	17	17	17	19	17	17	17
MA 042	140	17	17	17	17	17	17	23	21	22	17	17	17	17	17	17
H 3503	140	21	21	21	21	21	21	19	16	24	16	16	16	26		
H 0111	99	19	20	17	17	17	17	17	17	19	17	17	20	19	17	17
H 0106	99	17	19	17	17	17	17	17	17	17	17	17	19	17	17	17
H 0112	99	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17
H 3027	80	31	23	23				13	13	13	13	13	13		13	
H 3005	80	20	20	20	20	20	20	20	20	20	26	20			20	16
MA 141	70	17	17	17	19	17	17	32	22	22	22	22	22	22	22	22
MA 144	68	18	18	18	17	17	17	17	17	17	17	12	17	17	17	17
H 2033	67	21	21	24	21	21	21	21	21	21	21	21	21	21	21	21
MA 415	50	19	19	19	19	19	19	14	14	19	19	19	19	19	19	19
H 2038	50	14	14	14	14	14	14	10								
H 3004	50	20	20	20	25	24	26	26	25	25	20	20	20	20	20	20
H 2036	45	26	24	26	21	21	21	21	21	21	21	21	24	24	21	26
H 1029	41	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
MA 313	40	16	16	16	16	16	16		16		14	14	14	14	14	14
H 3002	40	10	13	13	27	21	21	21	21	27	21	21	21	21	21	21
H 3013	40	18	18	18	25	22	27	18	18	18		16		23	23	27
H 3012	40	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 549	32	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
MA 649	32	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
MA 550	32	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
MA 376	32	22	22	22	22	22	22	21	16	16	22	22	22	23	23	23
MA 650	32	16	16	16	16	16	16	16	11	16	16	16	16	16	16	16
H 3021	30	16	16	16	16	16	16	11	16	16	16	16	16	16	16	16
H 2051	30	23	23	16	23	23	23	23	23	23	23	23	23	23	23	23
H 3008	30			18	18	18	18	18	18	18	18	18	18	18	18	18
H 2035	30	21	15	21	21	21	21	21	21	21	15	21	21	21	21	24
H 3003A	20	20	20	20	20	20	20	14	14	20	20	20	20	14	14	20

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231			1.0		1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0							1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0			
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	1.0	1.0	1.0				1.0	1.0	1.0	1.0	1.0	1.0		1.0	
H 3005	80	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0			1.0	1.0	1.0
MA 141	70	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 144	68	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2033	67	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 415	50	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2038	50	1.0	1.0	1.0	1.0	1.0	1.0	1.0								
H 3004	50	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2036	45	0.99	0.99	0.99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	1.0	1.0	0.99	0.99
H 1029	41	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 313	40	1.0	1.0	1.0	1.0	1.0	1.0		1.0		1.0	1.0	1.0	1.0	1.0	1.0
H 3002	40	1.0	1.0	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3013	40	1.0	1.0	1.0	0.99	0.99	0.99	1.0	1.0	1.0		1.0		0.99	0.99	0.99
H 3012	40	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 549	32	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 649	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 550	32	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 376	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 650	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3021	30	0.99	0.99	0.99	0.99	0.99	0.99	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2051	30	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3008	30			0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2035	30	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3003A	20	0.97	0.97	0.97	0.97	0.97	0.97	0.99	0.99	0.97	0.97	0.97	0.97	0.99	0.99	0.97

9.3.2 Instance 2: $n = 750$

- $\gamma_s = 0.95$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Time assignment	11,283	6,691	600	0.05
Stream assignment	68,657	91,945	220.86	optimal
Room assignment	41,609	121,229	175,59	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 1012	262	23		23	23	23	23	23	23	23	23	23	23	23	23	
H 1028	231		2		2	2	2	2	2	2	2	2	5	3	3	
H 1029	41	21	21	21	21	21	21	21	21	21	21	21	21	21	21	
H 1058	263	6		6	6	6	6	6	6	6	6	6	6	6	6	
H 2013	261	3	3	3	3	3	3	3	3	3	9	9	9	9	9	
H 2032	236	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
H 2033	67	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2035	30	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2036	45	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2038	50	10	10	10	10	10	10	10	10	10	10	10	10	10	10	
H 2051	30	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 2053	261	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 3002	40	22	22	22			16				22	22	22			
H 3003A	20	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3004	50	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3005	80	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
H 3008	30	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3010	388	15	15	15	15	15	15	15	15	15	15	15	15	15	15	
H 3012	40	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3013	40	16	16	16	20	11		22	7	16	16	16	16	16	16	
H 3021	30	12	12	12	12	12	12	12	12	12	12	12	12	12	12	
H 3027	80	17	17	17	17	17	17	17	17	17	17	17	17	17	17	
H 3503	140	5	5	5	5	5	5	5	5	5		1	10	10	10	
MA 004	235	8	8	8	8	8	8	8	8	8	8	8	8	8	8	
MA 005	235	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
MA 041	152	9	9	9	9	9	9	9	9	9	9	9	14	14	14	
MA 042	140	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
MA 043	152	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
MA 141	70	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
MA 144	68	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
MA 313	40	7	7	7												
MA 376	32	23	23	23	23	23	23	3	3	3	3	3	3	3	3	
MA 415	50	14	14	14	14	14	14	14	14	14	14	14	14	14	14	
MA 549	32		13	13	13	13	13	13	13	13	13	13	13	13	13	
MA 550	32	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	26	26	26	26	26	26	18	18	26	26	26	26	26	26	26
H 1058	263	22		22	22	22	22	33	33	39	22	22	22			
H 1012	262	16		22	22	22	22	22	22	22	22	22	22	22	22	22
H 2013	261	25	25	25	25	25	25	34	37	25	32	23	32	32	32	32
H 2053	261	26	26	26	26	26	26	26	26	26	38	35	31	26	26	26
H 2032	236	51	51	51	51	51	51	51	51	61	51	51	51	51	51	51
MA 004	235	32	32	32	32	32	32	22	22	30	32	32	32	32	32	32
MA 005	235	23	23	17	23	23	23	32	32	32	32	32	45	32	32	32
H 1028	231		24		24	24	24	24	24	24	24	24	24	19	25	18
H 0110	198	24	24	24	24	24	26	24	24	24	24	24	24	24	24	24
MA 041	152	32	32	32	32	32	32	32	32	32	44	32	32	46	34	49
MA 043	152	23	23	23	23	23	23	23	17	17	23	23	23	23	23	23
H 0107	144	24	17	24	17	17	17	24	24	24	24	24	17	24	24	24
H 3503	140	13	19	19	19	19	19	19	19	19			51	29	39	38
MA 042	140	23	23	23	33	23	31	23	23	23	23	23	23	30	23	23
H 0106	99	24	24	24	24	24	27	24	24	27	24	24	24	24	24	24
H 0111	99	27	24	24	24	28	27	28	24	27	24	24	24	24	24	24
H 0112	99	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3005	80	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
H 3027	80	35	35	35	35	35	24	35	35	35	24	64		35	35	35
MA 141	70	40	40	40	28	40	40	40	40	28	40	40	40	40	40	40
MA 144	68	27	23	23	33	33	33	23	23	23	23	23	23	23	23	23
H 2033	67	30	37	30	34	30	35	30	35	37	30	30	30	30	30	30
H 2038	50	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29
H 3004	50	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
MA 415	50	32	32	32	32	32	32	23	32	23	32	32	32	32	23	32
H 2036	45	37	30	35	30	30	30	37	30	30	30	30	21	30	30	30
H 1029	41	34	34	34	34	34	34	34	34	34	24		34	34	34	34
H 3002	40	26	26	26			28				26	26	26			
H 3013	40	28	28	28	40	35		26	27	28	28	28	28	28	28	28
MA 313	40	27	19	19												
H 3012	40	37	35	37	28	28	28	28	28	28	28	28	28	35	34	35
MA 376	32	22	22	22	22	22	22	25	25	25	25	25	25	25	25	25
MA 650	32	22	22	22	22	22	22	22	22	22	22	22	16	22	22	22
MA 649	32	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
MA 549	32		32	26	26	26	26	26	26	26	26	26	26	26	26	26
MA 550	32	26	26	18	26	26	26	26	26	26	26	26	26	26	26	26
H 3008	30	28	28	28	28	28	28	28	28	20	20	20	20	28	28	28
H 3021	30	19	19	19	19	19	19	19	19	19	19	19	14		19	
H 2051	30	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26
H 2035	30	30	30	30	30	30	30	30	30	30	30	30	21	30	30	30
H 3003A	20	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0				
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231		1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0			1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3005	80	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99		1.0	1.0	1.0	1.0
MA 141	70	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 144	68	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2033	67	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2038	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3004	50	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 415	50	0.99	0.99	0.99	0.99	0.99	0.99	1.0	0.99	1.0	0.99	0.99	0.99	0.99	1.0	0.99
H 2036	45	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.0	0.99	0.99	0.99
H 1029	41	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99		0.99	0.99	0.99	0.99	0.99
H 3002	40	0.99	0.99	0.99			0.99				0.99	0.99	0.99			
H 3013	40	0.99	0.99	0.99	0.97	0.99		0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 313	40	0.99	1.0	1.0												
H 3012	40	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 376	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 650	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 649	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 549	32		0.97	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 550	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3008	30	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3021	30	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99		0.99		
H 2051	30	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2035	30	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.97	0.99	0.97	0.98	0.98
H 3003A	20	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98

9.3.3 Instance 3: $n = 1000$

Solution process

- $\gamma_s = 0.95$ (certainty value)

Model	# variables	# constraints	CPU time(s)	%gap
Time assignment	11,619	7,310	600	0.07
Stream assignment	65,657	91,934	207.93	optimal
Room assignment	31,670	91,928	600	40.05

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 1012	262	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
H 1028	231	2	2	2	2	2	2	10	10	10	2	2	2	2	2	
H 1029	41	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
H 1058	263	14	14	14	14	14	14	14	14	14	14	14	14	14	14	
H 2013	261	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
H 2032	236	17	17	17	17	17	3	17	17	17	17	17	17	17	17	
H 2033	67	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2035	30	23	23	23	23	23	23	23	23	23	23	23	23	23	23	
H 2036	45	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2038	50	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
H 2051	30	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 2053	261	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 3002	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 3003A	20															
H 3004	50	16						22	22	22	22	22	22	22	22	
H 3005	80	22	22	22	22	22	22	22	22	22	20	6	6	6	6	
H 3008	30	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
H 3010	388	21	21	21	21	21	21	11	21	21	21	21	21	21	21	
H 3012	40	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
H 3013	40	3	3	3	3	3	3	3	3	3	3	3	22	22	22	
H 3021	30	6	6	6	6	6	6		6	5	5	5	5	5	5	
H 3027	80	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
H 3503	140	9	9	9	9	9	9	9	9	9	9	9	9	9	9	
MA 004	235	8	8	8	8	8	8	8	8	8	8	8	9	9	9	
MA 005	235		3	1	8	8	8	8	8	8	8	8	8	8	8	
MA 041	152	23	22		16	16	16	16	16	16	16	16	16	16	16	
MA 042	140	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
MA 043	152	16	16	16	16	16	16	16	16	16	16	7	16	16	16	
MA 141	70	5	5	5	5	5	5	14	14	14	3	3	3	3	3	
MA 144	68	10	10	10	10	10	10	10	10	10	10	10	10	10	10	
MA 313	40	15	15	15	15	15	15	15	15	15	15	15	15	15	15	
MA 376	32	12	12	12	12	12	12		12	12	12	12	12	12	12	
MA 415	50	7	7	7	19	19	19	19	19	19	23	23	23	23	23	
MA 549	32	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
MA 550	32	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	44	44	44	44	44	44	38	31	44	44	44	44	44	44	44
H 1058	263	42	42	42	42	42	42	59	42	63	42	42	44	42	42	42
H 1012	262	33	33	33	33	33	33	33	42	33	33	33	33	33	33	33
H 2013	261	51	51	51	51	51	36	51	51	51	51	51	51	51	51	51
H 2053	261	42	38	40	30	30	30	30	30	30	30	30	30	30	30	30
H 2032	236	45	45	45	31	31	47	45	45	45	84	45	45	45	45	45
MA 005	235		32	79	41	41	41	29	38	29	41	41	41	41	41	41
MA 004	235	41	41	41	41	41	41	41	41	41	59	41	41	29	41	41
H 1028	231	31	31	31	31	31	31	38	38	38	31	31	31	31		
H 0110	198	31	31	31	31	31	31	31	31	31	22	31	31	31	31	31
MA 041	152	28	49		36	36	36	36	36	36	36	36	36	36	36	36
MA 043	152	44	36	36	45	45	36	36	36	36		36	35	47	48	46
H 0107	144	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
MA 042	140	36	36	36	36	36	36	25	25	25	36	25	36	36	36	36
H 3503	140	41	41	41	41	41	41	41	41	41	41	41	41	41	57	41
H 0111	99	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
H 0106	99	31	35	31	34	36	35	31	31	31	35	35	35	35	31	31
H 0112	99	22	31	22	31	31	31	31	31	31	31	22	22	31	31	31
H 3027	80	66	66	66	66	66	66	66	66	66	66	66	66	66	66	66
H 3005	80	33	33	40	33	33	33	33	33	33		36	28	43	43	50
MA 141	70	17	24	24	24	24	24	29	29	29	32	44	32	32	32	32
MA 144	68	38	38	38	38	38	38	50	49	38	38	38	38	38	38	38
H 2033	67	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38
H 3004	50	36						33	33	33	33	33	33	33	33	45
H 2038	50	44	44	44	38	38	38	48	43	44	38	38	38	48	48	48
MA 415	50	35	35	35	42	42	42	35	30	30	28	28	28	28		
H 2036	45	38	38	38	27	27	38	38	38	38	38	38	38	38	38	38
H 1029	41	33	33	33	33	33	33	24	33		33	33	33	33	33	33
H 3012	40	35	24	24	35	35	35	35	35	35	35	35	35	35	35	35
MA 313	40	33	33	33	33	33	33	33	33	33	23	23	33	33	33	33
H 3013	40	32	22	32	32	32	32	32	32	32	32	32	32	33	33	33
H 3002	40	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33
MA 549	32	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
MA 649	32	28	28	28	28	28	28	28	28	28	28	20	28	28	28	28
MA 376	32	24	24	24	24	24	24			17	24	24	24	24	24	24
MA 550	32	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
MA 650	32	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28
H 3021	30	28	28	28	28	28	28		28	24	24	24	24			
H 3008	30	30	30	30	30	30	30	30	30	30	30	30	21	30	30	30
H 2035	30	28	28	28	28	28	28	28	28	28	28	28	28	20	28	28
H 2051	30	21	30	30	30	30	30	30	30	30	30	30	30	21	30	30
H 3003A	20															

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0			
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3005	80	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	1.0	1.0	1.0	1.0	1.0
MA 141	70	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	1.0	1.0	1.0	1.0	1.0
MA 144	68	1.0	1.0	1.0	1.0	1.0	1.0	0.99	0.99	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2033	67	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3004	50	0.99						0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2038	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 415	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99		
H 2036	45	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 1029	41	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99		0.99	0.99	0.99	0.99	0.99	0.99
H 3012	40	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 313	40	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3013	40	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3002	40	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 549	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 649	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 376	32	0.99	0.99	0.99	0.99	0.99	0.99		0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 550	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 650	32	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3021	30	0.98	0.99	0.99	0.98	0.98	0.99		0.99	0.99	0.99	0.99	0.99			
H 3008	30	0.97	0.97	0.98	0.97	0.97	0.97	0.97	0.98	0.97	0.97	0.97	0.99	0.98	0.97	0.97
H 2035	30	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.99	0.98	0.98	0.98	0.98	0.99	0.98	0.98
H 2051	30	0.99	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.99	0.97	0.97
H 3003A	20															

9.3.4 Instance 4: $n = 1250$

- $\gamma_s = 0.85$, model infeasible for $\gamma_s = 0.9$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,703	7,446	600	0.06
Stream assignment	64,078	91,932	600	0.1
Room assignment	19,864	56,580	23.72	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday			
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
H 1012	262	3	3	3	3	9	9	10	10	10	2	2	2	2	2	2	
H 1028	231	13	13	13	13	13	3	13	13	13	13	13	13	13	13	13	
H 1029	41	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
H 1058	263	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	
H 2013	261	17	17	17	17	17		17	17	17	17	17	17	17	17	17	
H 2032	236	16	16	16	16	16	16	6	6	6	6	6	6	6	19	19	19
H 2033	67	11	11	11	11	11	11	20	11	7	11	11	11	11	11	11	11
H 2035	30	12	12	12	12	12	12	12									
H 2036	45	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 2038	50	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 2051	30	5	5	5	2	2	2	11		11	12	12	12	12	12	12	12
H 2053	261	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
H 3002	40	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
H 3003A	20																
H 3004	50	16	16	16	16	16	16	16	16	16	16	16	22	16	16	16	16
H 3005	80	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3010	388		2	7	22	22	22	22	22	22	16	16	16	16	16	16	16
H 3012	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3013	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3027	80	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
H 3503	140	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
MA 004	235	8	8	8	8	8	8	8	8	8	8	8	8	1			11
MA 005	235	19	19	19	8	8	8	8	8	8	8	8	8	8	8	8	8
MA 041	152	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
MA 042	140	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
MA 043	152	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
MA 141	70	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
MA 144	68	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
MA 313	40	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
MA 376	32	6	6	6	9		5	5	5	5	5	5	5			23	
MA 415	50	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
MA 549	32	23			23	23	23	7	7		14	14	14			6	6
MA 550	32	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388		41	38	51	56	45	37	37	37	40	28	40	40	40	40
H 1058	263	50	50	50	50	50	50	50	50	50		50	34	50	50	50
H 1012	262	35	35	35	53	46	46	42	42	42	38	35	35	35	35	39
H 2053	261	75	75	75	75	75	75	75	75	75	75	75	75	92	75	75
H 2013	261	50	50	50	35	35		50	50	50	50	50	50	50	97	50
H 2032	236	40	40	40	28	28	28	48	57	48	31	31	31	33	33	33
MA 005	235	48	48	48	46	46	46	31	31	42	46	46	46	46	46	46
MA 004	235	46	46	46	46	46	46	46	46	46	46	67	46	75		43
H 1028	231	37	37	37	37	47	50	37	37	37	37	37	37	37	37	37
H 0110	198	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
MA 043	152	33	33	33	23	23	33	33	33	33	33	33	33	33	33	33
MA 041	152	33	33	33	33	33	33	33	33	33	39	33	33	33	23	33
H 0107	144	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
H 3503	140	47	47	47	47	47	47	47	47	47	67	47	72	49	47	47
MA 042	140	33	33	33	33	33	33	33	33	33	48	43	45	33	33	33
H 0106	99	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
H 0111	99	38	35	35	35	39	39	35	35	35	38	35	35	35	38	35
H 0112	99	35	35	35	35	35	35	35	35	35	35	35	35	24	24	35
H 3027	80	58	58	58	58	58	40	58	58	58	58	58	58	58	58	58
H 3005	80	51	51	53	40	40	40	40	40	40	40	40	40	51	50	54
MA 141	70	42	42	42	42	42	42	42	57	56	42	42	42	42	42	42
MA 144	68	46	46	46	46	46	65	46	46	46	46	46	46	46	46	46
H 2033	67	50	50	50	54	54	54	40	43	38	43	43	43	49	50	54
MA 415	50	35	24	35	35	35	35	35	35	35	35	35	35	35	35	35
H 2038	50	43	43	43	43	43	43	43	43	43	43	43	43	43	43	43
H 3004	50	40	40	40	40	40	40	40	40	40	40	40	37	40	40	40
H 2036	45	43	43	43	43	43	43	43	43	43	43	43	43	43	43	43
H 1029	41	37	37	37	25	37	37	37	37	37	37	37	37	37	37	37
H 3012	40	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37
H 3013	40	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37
H 3002	40	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38
MA 313	40	36	36	36	25	25	36	36	36	36	36	36	36	36	36	36
MA 649	32	31	31	22	31	31	31	31	31	31	31	31	31	31	31	31
MA 550	32	31	31	31	31	31	31	31	31	31	31	31	31	31	22	31
MA 549	32	31			31	31	31	26	26		32	32	32		31	31
MA 650	32	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
MA 376	32	31	31	31	32		26	26	26	26	26	26	26		31	
H 3008	30	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
H 2035	30	26	26	26	26	26	26	18								
H 3021	30	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
H 2051	30	18	26	26	24	24	24	29		29	26	26	26	26	26	26
H 3003A	20															

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	0.99	0.99	0.99	0.99	0.99	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3005	80	0.99	0.99	0.99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	0.99	0.99
MA 141	70	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 144	68	0.99	0.99	0.99	0.99	0.99	0.97	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2033	67	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 415	50	0.99	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2038	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3004	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2036	45	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
H 1029	41	0.98	0.98	0.98	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
H 3012	40	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
H 3013	40	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
H 3002	40	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
MA 313	40	0.98	0.98	0.98	0.99	0.99	0.98	0.98	0.97	0.97	0.97	0.98	0.98	0.97	0.97	0.97
MA 649	32	0.93	0.94	0.99	0.93	0.93	0.93	0.94	0.93	0.93	0.93	0.94	0.94	0.93	0.93	0.93
MA 550	32	0.93	0.94	0.94	0.94	0.93	0.93	0.94	0.93	0.93	0.93	0.94	0.94	0.93	0.99	0.93
MA 549	32	0.93	0.94	0.94	0.94	0.93	0.93	0.99	0.99	0.99	0.9	0.91	0.91	0.94	0.94	0.94
MA 650	32	0.93	0.94	0.94	0.93	0.93	0.93	0.94	0.93	0.93	0.93	0.94	0.94	0.93	0.93	0.93
MA 376	32	0.94	0.94	0.94	0.92	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.93	0.93	0.93
H 3008	30	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
H 2035	30	0.98	0.98	0.98	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3021	30	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
H 2051	30	0.99	0.98	0.98	0.99	0.99	0.99	0.93	0.92	0.98	0.98	0.98	0.98	0.98	0.98	0.98
H 3003A	20															

9.3.5 Instance 5: $n = 1500$

- $\gamma_s = 0.6$, model infeasible for $\gamma_s = 0.65$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,661	7,423	600	0.06
Stream assignment	63,974	91,932	274.37	optimal
Room assignment	22,074	63,347	202.86	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	22	3	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 1012	262	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
H 1028	231	2	2	2	2	2	2	2	2	2	6	6	6	3	3	3
H 1029	41	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
H 1058	263	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
H 2013	261	17	17	17	17	17	3	17	17	17	17	17	17	17	17	17
H 2032	236	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
H 2033	67	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 2035	30	12	12	12	12	12	12	12	12	12	12	12	12		12	
H 2036	45	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 2038	50	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
H 2051	30	5	5	5		19		5	5	5	2	2	2			
H 2053	261	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
H 3002	40	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
H 3003A	20															
H 3004	50	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3005	80	22	22	22	22	22	22	14	14	14	22	22	22	22	22	22
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3010	388	16	16	16	22	22	22	22	22	22	22	22	22	10	10	10
H 3012	40	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
H 3013	40	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3027	80	21	21	21	21	21	21	21	21	21	21	11	21	21	21	21
H 3503	140	13	13	13	13	13	13	20	13	13	13	13	13	13	13	13
MA 004	235	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
MA 005	235	19	19	19	19	1	19	8	8	8	8	8	8	8	8	8
MA 041	152	7	7	7	9	9	9	16	16	7	16	16	16	16	16	16
MA 042	140	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
MA 043	152	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
MA 141	70	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
MA 144	68	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
MA 313	40	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
MA 376	32	23	23	23	23	23	23	23	23	23	23	23	23	6		6
MA 415	50	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
MA 549	32	23		23	6	6	6	6	6	6	23	23	23	23	23	23
MA 550	32												5	5	5	5
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	58	54	54	39	39	39	39	39	39	60	55	48	62	60	45
H 1058	263	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45
H 1012	262	34	34	34	51	45	48	34	34	34	34	34	34	34	34	34
H 2013	261	54	54	54	108	36	58	54	54	54	54	54	36	54	54	54
H 2053	261	83	83	83	83	83	83	83	83	83	83	83	83	83	83	83
H 2032	236	51	51	51	34	34	41	34	34	34	34	34	34	34	34	34
MA 005	235	34	34	34	34	101	34	49	49	49	49	49	73	49	49	49
MA 004	235	49	49	49	49	49	49	33	33	45	49	49	49	49	49	49
H 1028	231	36	36	36	36	36	36	41	36	36	52	52	61	37	37	37
H 0110	198	36	40	43	41	41	41	36	41	41	36	36	36	36	36	36
MA 041	152	40	40	40	49	49	70	43	43	40	43	43	43	55	53	57
MA 043	152	43	43	43	43	43	43	43	43	43	43	43	43	43	43	43
H 0107	144	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3503	140	39	39	39	39	39	39	42	50	39	39	39	39	39	39	39
MA 042	140	43	43	43	43	43	43	43	29	43	29	29	29	43	43	43
H 0111	99	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 0112	99	36	36	36	36	24	24	36	36	36	36	36	36	36	36	36
H 0106	99	39	53	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3005	80	39	39	39	39	39	39	53	79	73	39	39	39	39	39	39
H 3027	80	53	53	53	53	53	53	53	53	53	53	45	36	53	53	53
MA 141	70	63	63	42	63	63	63	63	63	63	63	63	63	63	63	63
MA 144	68	49	49	49	49	33	49	49	49	49	49	49	49	49	49	49
H 2033	67	58	58	54	45	45	45	45	45	45	54	58	54	52	58	54
H 3004	50	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39
H 2038	50	45	45	45	45	45	45	45	45	45	45	30	45	45	45	45
MA 415	50	50	50	50	50	50	50	33	50	50	33	33	50	50	50	50
H 2036	45	45	45	45	30	45	45	45	45	45	45	45	45	45	45	45
H 1029	41	34	34	34	34	34	34	34	34	34	23	23	34	34	34	34
H 3012	40	38	38	38	38	38	38	38	38	38	26	38	26	38	38	38
MA 313	40	40	27	27	40	40	40	40	40	40	40	40	40	40	40	40
H 3002	40	37	37	37	37	37	37	37	37	37	37	37	37	37	37	25
H 3013	40	39	39	39	39	39	39	39	39	26	39	39	39	39	39	39
MA 376	32	32	32	22	32	32	32	32	32	32	32	32	32	32		32
MA 649	32	22	32	32	32	32	32	32	32	32	32	32	32	32	32	32
MA 550	32												27	27	27	27
MA 549	32	32		32	32	32	32	32	32	32	32	32	32	32	32	32
MA 650	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
H 3021	30	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28
H 2035	30	27	27	27	27	27	27	27	27	27	27	27	27		18	
H 3008	30	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28
H 2051	30	18	27	27		23		27	27	27	24	24	24			
H 3003A	20															

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3005	80	1.0	1.0	1.0	1.0	1.0	1.0	0.99	0.83	0.94	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.0	0.99	0.99	0.99
MA 141	70	0.94	0.94	0.99	0.95	0.95	0.94	0.93	0.95	0.94	0.93	0.92	0.95	0.94	0.94	0.93
MA 144	68	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 2033	67	0.97	0.96	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.95	0.99	0.99	0.96	0.99
H 3004	50	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.99	0.99	0.99
H 2038	50	0.91	0.91	0.89	0.92	0.92	0.91	0.9	0.92	0.92	0.9	0.99	0.92	0.92	0.91	0.9
MA 415	50	0.76	0.75	0.72	0.78	0.78	0.75	0.99	0.78	0.77	0.99	0.99	0.78	0.76	0.75	0.73
H 2036	45	0.74	0.72	0.7	0.99	0.75	0.72	0.7	0.75	0.74	0.7	0.68	0.76	0.74	0.72	0.7
H 1029	41	0.96	0.96	0.95	0.96	0.96	0.96	0.95	0.96	0.96	0.99	0.99	0.96	0.96	0.96	0.95
H 3012	40	0.84	0.82	0.81	0.85	0.84	0.82	0.81	0.84	0.84	0.99	0.8	0.99	0.84	0.82	0.81
MA 313	40	0.72	0.99	0.99	0.74	0.74	0.71	0.69	0.74	0.73	0.69	0.67	0.74	0.73	0.71	0.69
H 3002	40	0.88	0.87	0.86	0.89	0.89	0.87	0.86	0.89	0.88	0.86	0.85	0.89	0.88	0.87	0.99
H 3013	40	0.8	0.79	0.77	0.82	0.81	0.79	0.77	0.81	0.99	0.77	0.76	0.82	0.8	0.79	0.78
MA 376	32	0.71	0.7	0.99	0.73	0.72	0.7	0.68	0.72	0.72	0.68	0.67	0.73	0.74		0.71
MA 649	32	0.99	0.7	0.67	0.72	0.72	0.7	0.68	0.72	0.72	0.68	0.66	0.73	0.71	0.7	0.68
MA 550	32												0.95	0.95	0.94	0.94
MA 549	32	0.71		0.68	0.75	0.75	0.72	0.71	0.75	0.74	0.68	0.67	0.73	0.71	0.7	0.68
MA 650	32	0.71	0.7	0.67	0.72	0.72	0.7	0.68	0.72	0.72	0.68	0.66	0.73	0.71	0.7	0.68
H 3021	30	0.86	0.85	0.84	0.87	0.87	0.85	0.84	0.87	0.86	0.84	0.83	0.87	0.86	0.85	0.84
H 2035	30	0.88	0.87	0.86	0.89	0.88	0.87	0.86	0.88	0.88	0.86	0.85	0.89		0.99	
H 3008	30	0.86	0.85	0.84	0.87	0.87	0.85	0.84	0.87	0.86	0.84	0.83	0.87	0.86	0.85	0.84
H 2051	30	0.99	0.89	0.88		0.98		0.88	0.9	0.9	0.95	0.94	0.96			
H 3003A	20															

9.3.6 Instance 6: $n = 1750$

- $\gamma_s = 0.25$, model infeasible for $\gamma_s = 0.3$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,577	7,287	600	0.03
Stream assignment	63,490	91,931	277.51	optimal
Room assignment	17,906	50,783	29.00	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	22	22	22	22
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 1012	262	17	17	17	17	17	17	17	17	17	17	17	17	17	3	17
H 1028	231	8	8	8	8	8	8	8	8	8	8	8	8	7	7	7
H 1029	41	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
H 1058	263	16	16	16	16	16	16	20	1	2	2	2	2	2	2	2
H 2013	261	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
H 2032	236	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
H 2033	67	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
H 2035	30	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
H 2036	45	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
H 2038	50	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
H 2051	30		3		12	12	12	12	12	12	12	12	12	12	12	12
H 2053	261	8	8	8	22	22	22	6	6	6	6	6	6	14	14	14
H 3002	40	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
H 3003A	20	12														
H 3004	50	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
H 3005	80	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3010	388	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
H 3012	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3013	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3027	80	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
H 3503	140	21	21	21	21	21	21	21	21	21	21	16	21	21	21	21
MA 004	235	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
MA 005	235	13	13	13	13	13	7	13	13	13	13	13	13	13	13	13
MA 041	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 042	140	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 043	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 141	70	19	19	19	3	3	3	19	19	19	10	10	10	8	8	8
MA 144	68	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
MA 313	40	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
MA 376	32	23	23	23	23	23	23					2		23		
MA 415	50	2	2	2	2	2	2	8	8	8	8	8	8	8	8	8
MA 549	32	6	6	6		11		16	16	16	16	16	16		6	6
MA 550	32			23	23	23	23	23	23	23	23	23	23	23	23	23
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	51	51	51	51	51	51	51	51	51	33	51	51	51	51	51
H 1058	263	57	55	56	60	56	59	66	88	36	36	36	41	36	36	36
H 1012	262	56	56	56	36	56	56	56	56	56	56	56	56	116	55	36
H 2013	261	88	88	88	88	88	88	88	109	88	88	88	88	88	88	88
H 2053	261	50	46	50	39	39	39	32	32	32	64	53	53	77	51	84
H 2032	236	51	51	51	51	55	51	51	51	51	51	51	51	33	33	33
MA 004	235	39	39	39	39	25	39	39	39	39	39	39	39	39	39	39
MA 005	235	39	39	39	52	39	41	39	39	39	39	39	39	39	39	39
H 1028	231	50	50	50	50	50	50	50	50	77	51	74	51	41	41	41
H 0110	198	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
MA 043	152	46	46	46	46	46	46	46	46	46	46	46	46	46	46	46
MA 041	152	60	60	56	56	60	60	46	46	46	46	46	46	56	54	56
H 0107	144	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3503	140	55	55	55	55	55	55	55	55	55	36	43	55	55	55	55
MA 042	140	46	46	46	46	46	46	46	46	30	46	46	46	46	46	46
H 0106	99	42	44	42	40	36	36	36	36	42	41	41	39	49	57	63
H 0111	99	36	36	36	24	24	24	36	36	36	36	24	36	36	36	36
H 0112	99	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3027	80	66	66	66	66	66	66	43	43	66	66	66	66	66	66	66
H 3005	80	34	34	34	34	34	34	34	34	34	52	46	49	34	34	34
MA 141	70	34	34	22	37	37	60	34	34	34	46	46	46	50	50	50
MA 144	68	46	46	46	46	46	46	46	46	46	64	62	46	46	46	46
H 2033	67	41	34	34	34	34	34	52	52	52	34	34	34	34	34	34
MA 415	50	36	36	36	36	36	36	32	32	50	50	50	50	50	50	50
H 2038	50	43	43	43	43	43	43	43	43	43	43	43	43	43	43	43
H 3004	50	34	34	34	34	22	22	34	34	34	34	34	34	34	34	34
H 2036	45	43	43	43	43	43	43	43	43	43	43	43	43	43	43	43
H 1029	41	41	41	41	41	41	41	41	41	41	41	41	41	27	27	41
MA 313	40	25	38	25	38	38	38	38	38	38	38	38	38	38	38	38
H 3012	40	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39
H 3013	40	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39
H 3002	40	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37
MA 650	32	32	32	32	32	32	32	32	21	32	32	32	32	32	32	32
MA 549	32	32	32	32		30		28	28	28	28				32	32
MA 649	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
MA 376	32	32	32	21	32	32	32						24		32	
MA 550	32			32	32	32	32	32	32	32	32	32	32	32	32	32
H 2035	30	17	26	26	26	26	26	26	26	26		26				
H 3008	30	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
H 2051	30		24		27	27	27	27	27	27	27	27	27	27	27	27
H 3021	30	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
H 3003A	20	17														

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	0.99	0.99
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3027	80	0.92	0.94	0.92	0.92	0.93	0.93	0.99	0.99	0.93	0.92	0.93	0.92	0.94	0.93	0.93
H 3005	80	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	0.99	0.99	1.0	1.0	1.0
MA 141	70	0.99	0.99	1.0	0.99	0.99	0.87	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 144	68	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.58	0.7	0.99	0.99	0.99	0.99
H 2033	67	0.99	0.99	0.99	0.99	0.99	0.99	0.94	0.95	0.95	0.99	0.99	0.99	0.99	0.99	0.99
MA 415	50	0.96	0.97	0.96	0.96	0.96	0.97	0.99	0.99	0.41	0.39	0.41	0.4	0.43	0.41	0.41
H 2038	50	0.76	0.78	0.74	0.75	0.77	0.77	0.74	0.78	0.77	0.76	0.77	0.76	0.78	0.77	0.77
H 3004	50	0.98	0.98	0.98	0.98	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
H 2036	45	0.49	0.52	0.47	0.48	0.5	0.51	0.46	0.51	0.5	0.48	0.5	0.49	0.52	0.5	0.5
H 1029	41	0.39	0.41	0.37	0.38	0.4	0.41	0.36	0.41	0.4	0.38	0.4	0.39	0.98	0.98	0.4
MA 313	40	0.99	0.51	0.99	0.48	0.5	0.5	0.46	0.51	0.5	0.48	0.5	0.48	0.51	0.5	0.5
H 3012	40	0.45	0.47	0.43	0.44	0.46	0.47	0.42	0.47	0.46	0.44	0.46	0.45	0.47	0.46	0.46
H 3013	40	0.45	0.47	0.43	0.44	0.46	0.47	0.42	0.47	0.46	0.44	0.46	0.45	0.47	0.46	0.46
H 3002	40	0.57	0.6	0.56	0.56	0.58	0.59	0.55	0.59	0.59	0.57	0.59	0.57	0.6	0.58	0.58
MA 650	32	0.35	0.38	0.34	0.35	0.36	0.37	0.33	0.97	0.37	0.35	0.37	0.35	0.38	0.36	0.36
MA 549	32	0.39	0.41	0.37		0.52		0.63	0.66	0.66	0.64				0.4	0.4
MA 649	32	0.35	0.38	0.34	0.35	0.36	0.37	0.33	0.37	0.37	0.35	0.37	0.35	0.38	0.36	0.36
MA 376	32	0.36	0.38	0.96	0.35	0.37	0.37					0.9			0.37	
MA 550	32			0.34	0.35	0.37	0.37	0.34	0.37	0.37	0.35	0.37	0.36	0.38	0.37	0.37
H 2035	30	0.99	0.69	0.65	0.66	0.67	0.68	0.65	0.68	0.68		0.68				
H 3008	30	0.58	0.61	0.57	0.58	0.59	0.6	0.57	0.6	0.6	0.58	0.6	0.59	0.61	0.59	0.6
H 2051	30		0.81		0.62	0.63	0.64	0.6	0.64	0.63	0.62	0.64	0.62	0.64	0.63	0.63
H 3021	30	0.58	0.61	0.57	0.58	0.59	0.6	0.57	0.6	0.6	0.58	0.6	0.59	0.61	0.59	0.6
H 3003A	20	0.62														

9.3.7 Instance 7: $n = 2000$

- $\gamma_s = 0.05$, model infeasible for $\gamma_s = 0.1$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,577	7,287	600	0.04
Stream assignment	63,500	91,934	209.48	optimal
Room assignment	23,757	68,369	95.12	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	2	2	2	2	2	2	2	2	3	3	3
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
H 1012	262	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
H 1028	231	7	7	7	2	2	2	2	2	2	2	2	2	2	2	2
H 1029	41	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
H 1058	263	17	17	17	1	17	17	17	17	17	17	17	17	17	17	17
H 2013	261	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
H 2032	236	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21
H 2033	67	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
H 2035	30	12	12	12	12	12	12	12	12	12	12	12	12	15		15
H 2036	45	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
H 2038	50	16	16	16		16		16	16	16	16	16	16	8	8	8
H 2051	30	5	5	5	5										16	
H 2053	261	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
H 3002	40	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
H 3003A	20															12
H 3004	50	11	22		10	10	10	10	10	10	10	10	10	10	10	10
H 3005	80	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3010	388	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
H 3012	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3013	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
H 3027	80	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
H 3503	140	8	8	8	8	3	8	9	9	20	19	19	19	19	19	19
MA 004	235	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
MA 005	235	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
MA 041	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 042	140	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 043	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
MA 141	70	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
MA 144	68	10	10	10	10	10	10	6	6	6	22	22	22	22	22	22
MA 313	40	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
MA 376	32	6	6	6	6	6		23		9		23		23	23	23
MA 415	50	15	15	15	15	15	15	15	15	15	15	15	15	7	15	2
MA 549	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 550	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
MA 649	32	14	14	14		8		5	5	5	5	5	5	6	6	6
MA 650	32	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	80	52	87	52	52	52	52	52	55	52	52	52	52	52	52
H 1058	263	56	56	56	114	36	122	56	56	56	56	56	56	56	56	36
H 1012	262	91	91	91	91	91	91	91	91	91	91	91	91	91	91	91
H 2053	261	51	51	51	51	51	51	51	76	51	51	51	51	51	51	51
H 2013	261	68	68	68	68	68	43	68	68	43	68	68	68	68	68	68
H 2032	236	56	56	56	56	56	56	56	56	56	35		56	56	56	56
MA 005	235	33	33	33	33	33	33	33	33	33	53	46	50	33	33	33
MA 004	235	21	33	21	33	33	33	33	33	33	33	33	33	33	33	33
H 1028	231	41	41	41	36	36	40	36	36	36	36	36	36	36	36	36
H 0110	198	36	36	36	36	36	22	22	22	22	36	36	36	36	36	36
MA 043	152	29	46	46	46	46	46	29	46	46	46	46	46	46	46	46
MA 041	152	46	46	46	46	46	46	46	46	46	46	46	46	46	46	46
H 0107	144	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3503	140	50	50	50	32	56	50	51	51	68	53	53	53	33	41	33
MA 042	140	56	56	56	61	56	61	46	46	46	54	62	61	46	46	46
H 0112	99	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 0106	99	36	36	36	22	36	36	36	36	36	36	36	36	36	23	36
H 0111	99	36	36	36	41	42	41	36	36	36	41	41	41	36	36	43
H 3005	80	39	39	39	39	39	39	39	39	39	39	52	39	39	39	39
H 3027	80	50	50	50	50	50	79	50	50	50	50	50	50	50	50	50
MA 141	70	36	36	36	36	36	36	36	36	36	61	36	36	36	36	36
MA 144	68	46	46	46	64	46	66	65	54	54	38	38	38	57	64	49
H 2033	67	43	43	43	43	43	43	62	58	56	57	60	57	43	43	43
H 2038	50	43	43	43		43		43	43	43	43	43	43	50	46	50
MA 415	50	38	38	38	38	38	38	38	38	38	38	38	38	41	38	36
H 3004	50	46	38		46	46	46	46	46	46	46	46	46	46	46	46
H 2036	45	27	27	27	43	43	43	43	43	43	43	43	43	43	43	43
H 1029	41	25	25	41	41	41	41	41	41	41	41	41	41	41	41	41
H 3002	40	39	39	39	39	39	39	39	39	39	36	39	24	39	39	39
H 3012	40	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38
H 3013	40	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38
MA 313	40	33	33	33	33	33	33	33	33	33	33	33	33	21	33	33
MA 649	32	32	32	32		32		25	25	25	25	25	25	31	31	31
MA 376	32	31	31	31	31	31		31		32		31		31	31	31
MA 650	32	31	31	31	31	31	31	31	31	31	31	19	31	31	31	31
MA 549	32	31	19	31	31	31	31	31	31	31	31	31	31	31	31	31
MA 550	32	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
H 3008	30	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26
H 3021	30	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26
H 2051	30	15	25	25	25										27	
H 2035	30	25	25	25	25	25	25	25	25	25	25	25	25	24		24
H 3003A	20															16

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99
H 3005	80	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.99	0.99	0.99
H 3027	80	0.99	0.99	0.99	0.99	0.99	0.12	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MA 141	70	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.49	0.99	0.99	0.99	0.99	0.99
MA 144	68	0.96	0.96	0.96	0.28	0.96	0.16	0.21	0.74	0.75	0.99	0.99	0.99	0.55	0.26	0.9
H 2033	67	0.98	0.98	0.98	0.98	0.98	0.97	0.32	0.47	0.6	0.54	0.33	0.52	0.98	0.98	0.98
H 2038	50	0.42	0.41	0.42		0.42		0.42	0.4	0.41	0.42	0.39	0.4	0.1	0.26	0.11
MA 415	50	0.73	0.72	0.73	0.74	0.73	0.7	0.73	0.72	0.73	0.73	0.7	0.71	0.55	0.73	0.83
H 3004	50	0.25	0.69		0.28	0.28	0.24	0.28	0.26	0.27	0.27	0.25	0.26	0.26	0.27	0.27
H 2036	45	0.96	0.96	0.96	0.18	0.18	0.15	0.18	0.17	0.17	0.17	0.15	0.16	0.17	0.17	0.17
H 1029	41	0.93	0.92	0.12	0.13	0.12	0.1	0.12	0.11	0.12	0.12	0.11	0.11	0.11	0.12	0.12
H 3002	40	0.15	0.15	0.15	0.16	0.15	0.13	0.15	0.14	0.15	0.26	0.13	0.93	0.14	0.15	0.15
H 3012	40	0.16	0.16	0.16	0.17	0.16	0.14	0.16	0.15	0.16	0.16	0.14	0.15	0.15	0.16	0.16
H 3013	40	0.16	0.16	0.16	0.17	0.16	0.14	0.16	0.15	0.16	0.16	0.14	0.15	0.15	0.16	0.16
MA 313	40	0.44	0.43	0.43	0.44	0.44	0.4	0.43	0.42	0.43	0.43	0.4	0.42	0.98	0.43	0.43
MA 649	32	0.09	0.09	0.09		0.12		0.53	0.52	0.53	0.53	0.5	0.52	0.14	0.14	0.14
MA 376	32	0.14	0.14	0.14	0.15	0.14		0.13		0.09		0.11		0.12	0.13	0.12
MA 650	32	0.13	0.12	0.12	0.13	0.13	0.11	0.13	0.12	0.12	0.12	0.87	0.12	0.12	0.13	0.12
MA 549	32	0.13	0.88	0.12	0.13	0.12	0.11	0.12	0.12	0.12	0.12	0.11	0.12	0.12	0.12	0.12
MA 550	32	0.13	0.12	0.12	0.13	0.12	0.11	0.12	0.12	0.12	0.12	0.11	0.12	0.12	0.12	0.12
H 3008	30	0.31	0.3	0.31	0.31	0.31	0.28	0.31	0.3	0.3	0.31	0.28	0.29	0.3	0.31	0.3
H 3021	30	0.31	0.3	0.31	0.31	0.31	0.28	0.31	0.3	0.3	0.31	0.28	0.29	0.3	0.31	0.3
H 2051	30	0.97	0.38	0.39	0.4									0.24		
H 2035	30	0.35	0.34	0.34	0.35	0.34	0.32	0.34	0.33	0.34	0.34	0.32	0.33	0.46		0.47
H 3003A	20															0.39

9.3.8 Instance 8: $n = 2250$

- $\gamma_s = 0.01$, model infeasible for $\gamma_s = 0.05$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,619	7,355	600	0.05
Stream assignment	63,355	91,937	600	0.02
Room assignment	22,495	64,542	61.03	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99		2		2	2	2	2	2	2	2	2	2	2	2	
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 1012	262	13	13	13	13	13	13	13	13	13	13	13	13		13	
H 1028	231	2	2	2	2	2	2	2	2	2	2	3	2	3	3	
H 1029	41	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
H 1058	263	8	8	8	1	22	20	16	16	16	16	16	16	16	17	
H 2013	261	10	10	10	10	10	10	10	10	10	10	10	10	10	10	
H 2032	236	17	17	17	17	17	17	17	17	17	17	17	17	11	11	
H 2033	67	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 2035	30	12	12	12		12	16	12	12	12	12	12	12	12	12	
H 2036	45	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 2038	50	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
H 2051	30	16	16	16					11		3	2	11			
H 2053	261	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
H 3002	40	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
H 3003A	20															
H 3004	50	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3005	80	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3010	388	7	7	7	9	9	9	9	9	9	9	9	9	9	9	
H 3012	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 3013	40	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3027	80	14	14	14	14	14	14	14	14	14	22	22	22	22	22	
H 3503	140	8	8	8	8	8	8	8	8	8	8	8	8	8	8	
MA 004	235	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
MA 005	235	19	19	19	19	19	19	14	14	14	14	14	14	14	14	
MA 041	152	11	11	11	11	11	11	11	11	11	11	11	8	8	8	
MA 042	140	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
MA 043	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
MA 141	70	21	21	21	21	21	21	21	21		21	21	21	21	21	
MA 144	68	9	9	9	9	9	9	10	10	10	6	6	6	6	7	
MA 313	40	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
MA 376	32	5	5	5	5	5	5	23	23	23	5			5	5	
MA 415	50	15	15	15	15	15	15	15	15	15	15	15	15	15	15	
MA 549	32			23	6	6	6	6	6	6	8	8			23	
MA 550	32	23	23	23	23	23	23	23	23	23	23	23	23	23	23	
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	42	42	42	53	80	53	53	53	53	53	53	53	53	53	
H 1058	263	84	52	52	122	39	71	44	44	44	44	44	44	130	59	
H 1012	262	40	40	40	40	40	40	40	40	40	40	40	40	40	40	
H 2053	261	71	71	71	71	71	44	44	71	71	71	71	71	71	71	
H 2013	261	47	47	47	47	47	47	67	47	69	47	47	47	47	47	
H 2032	236	59	59	59	59	59	59	36	36	59	59	59	59	64	64	
MA 004	235	97	97	97	97	97	97	97	97	97	97	97	97	97	97	
MA 005	235	55	55	55	34	42	34	84	54	92	54	54	54	54	54	
H 1028	231	36	41	36	42	45	43	42	42	42	36	37	42	64	37	
H 0110	198	36	36	36	36	36	36	36	36	36	36	36	36	36	36	
MA 043	152	48	48	48	48	48	48	48	48	48	48	48	48	48	48	
MA 041	152	64	56	59	48	48	48	59	59	59	48	48	52	52	52	
H 0107	144	36	36	22	36	36	36	36	36	36	36	36	36	36	36	
H 3503	140	52	52	52	52	52	52	52	52	52	52	52	47	52	52	
MA 042	140	48	48	48	48	48	48	48	48	48	48	48	48	48	48	
H 0106	99		36		36	36	36	36	36	36	36	36	36	36	36	
H 0111	99	36	36	36	36	36	36	36	36	36	36	36	36	36	36	
H 0112	99	22	22	36	36	36	36	36	36	36	22	36	36	36	36	
H 3005	80	44	44	44	44	44	44	61	58	63	59	59	64	44	44	
H 3027	80	58	54	54	54	54	54	33	33	33	39	39	39	67	51	
MA 141	70	58	58	58	58	58	58	36	58		58	58	58	58	58	
MA 144	68	53	53	53	33	53	53	47	47	47	56	56	68	31	31	
H 2033	67	55	48	52	34	34	34	34	34	34	34	34	34	34	34	
H 3004	50	44	44	44	44	44	44	44	44	44	44	44	44	44	44	
H 2038	50	42	25	25	42	42	42	42	42	42	42	42	42	42	42	
MA 415	50	39	39	39	39	39	39	39	39	39	23	39	23	39	39	
H 2036	45	34	34	34	34	34	34	34	20	20	34	34	34	34	34	
H 1029	41	40	40	40	40	40	40	40	40	40	40	40	40	40	24	
MA 313	40	37	37	37	37	37	37	37	37	37	37	37	37	37	37	
H 3002	40	34	34	34	20	34	34	34	34	34	34	34	34	34	34	
H 3012	40	39	39	39	39	39	39	39	39	39	39	39	39	39	39	
H 3013	40	39	39	39	39	39	39	39	39	39	39	39	39	39	39	
MA 549	32			32	31	31	31	31	31	31	32	32			19	
MA 649	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	
MA 650	32	32	19	32	32	32	32	32	32	32	32	32	32	32	32	
MA 376	32	15	25	25	25	25	25	32	32	32	25			25	25	
MA 550	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	
H 2035	30	25	25	25		15	27	25	25	25	25	25	25	25	25	
H 2051	30	27	27	27					29		22	22	29			
H 3021	30	26	26	26	26	26	26	26	26	26	26	26	26	26	26	
H 3008	30	26	26	26	26	26	26	26	26	26	26	26	26	26	26	
H 3003A	20															

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0106	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0111	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0112	99	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3005	80	0.99	0.99	0.99	0.99	0.99	0.99	0.67	0.8	0.55	0.75	0.74	0.46	0.99	0.99	0.99
H 3027	80	0.8	0.9	0.91	0.91	0.92	0.91	0.99	0.99	0.99	0.99	0.99	0.99	0.39	0.95	0.73
MA 141	70	0.34	0.31	0.33	0.34	0.36	0.33	0.99	0.37	0.36	0.35	0.31	0.36	0.32	0.37	0.37
MA 144	68	0.51	0.48	0.49	0.99	0.53	0.5	0.8	0.82	0.8	0.39	0.38	0.02	0.99	0.99	0.96
H 2033	67	0.36	0.7	0.53	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 3004	50	0.13	0.12	0.13	0.13	0.14	0.13	0.13	0.15	0.13	0.14	0.14	0.12	0.14	0.12	0.15
H 2038	50	0.24	0.98	0.98	0.24	0.25	0.23	0.24	0.26	0.24	0.25	0.25	0.22	0.25	0.22	0.26
MA 415	50	0.4	0.37	0.39	0.4	0.42	0.39	0.4	0.43	0.4	0.99	0.41	0.99	0.41	0.38	0.43
H 2036	45	0.41	0.39	0.4	0.41	0.43	0.4	0.41	0.99	0.99	0.42	0.42	0.39	0.42	0.39	0.43
H 1029	41	0.04	0.03	0.04	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.04	0.03	0.04	0.04	0.85
MA 313	40	0.07	0.06	0.06	0.07	0.07	0.06	0.07	0.08	0.07	0.07	0.07	0.06	0.07	0.06	0.08
H 3002	40	0.17	0.15	0.16	0.94	0.18	0.16	0.16	0.18	0.16	0.17	0.17	0.15	0.17	0.15	0.18
H 3012	40	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.04	0.03	0.03	0.03	0.03	0.03	0.03	0.04
H 3013	40	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.04	0.03	0.03	0.03	0.03	0.03	0.03	0.04
MA 549	32	0.02	0.02	0.02	0.03	0.03	0.03	0.03	0.04	0.03	0.02	0.02	0.02	0.03	0.02	0.72
MA 649	32	0.02	0.02	0.02	0.02	0.03	0.02	0.02	0.03	0.02	0.03	0.03	0.02	0.03	0.02	0.03
MA 650	32	0.02	0.68	0.02	0.02	0.03	0.02	0.02	0.03	0.02	0.03	0.03	0.02	0.03	0.02	0.03
MA 376	32	0.95	0.25	0.25	0.26	0.28	0.26	0.02	0.03	0.02	0.27	0.27	0.25	0.27	0.25	0.28
MA 550	32	0.02	0.02	0.02	0.02	0.03	0.02	0.02	0.03	0.02	0.03	0.03	0.02	0.03	0.02	0.03
H 2035	30	0.13	0.12	0.13	0.88	0.07	0.13	0.14	0.13	0.14	0.14	0.12	0.14	0.12	0.14	0.14
H 2051	30	0.07	0.06	0.07	0.07	0.07	0.07	0.03	0.03	0.3	0.33	0.03	0.03	0.03	0.03	0.03
H 3021	30	0.11	0.1	0.1	0.11	0.12	0.11	0.11	0.12	0.11	0.12	0.11	0.1	0.11	0.1	0.12
H 3008	30	0.11	0.1	0.1	0.11	0.12	0.11	0.11	0.12	0.11	0.12	0.11	0.1	0.11	0.1	0.12
H 3003A	20															

9.3.9 Instance 9: $n = 2500$

- $\gamma_s = 0.001$, model infeasible for $\gamma_s = 0.01$ (certainty value)

Solution process

Model	# variables	# constraints	CPU time(s)	%gap
Room assignment	11,619	7,355	600	0.07
Stream assignment	63,308	91,935	600	0.1
Room assignment	19,056	54,350	28.43	optimal

Cluster view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 0106	99	2	2	2	2	7	2	2	2	2	2	2	2	2	2	
H 0107	144	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0110	198	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0111	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 0112	99	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
H 1012	262	16	16	16	16	16	16	22	22	22	22	22	22	22	22	
H 1028	231	22	22	22	22	22	22	22	22	22	22	22	22	14	14	
H 1029	41	22	22	22	22	22	22	22	22	22	22	22	22	22	22	
H 1058	263	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
H 2013	261	9	9	9	9	9	9	9	9	9	9	9	9	9	9	
H 2032	236	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
H 2033	67	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
H 2035	30	12	12	12	12	12	12	12	12	12	12	12	12			
H 2036	45	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 2038	50	16	16	16	16	16	16	16	16	16	16	16	16	16	16	
H 2051	30	5	5	5		2	2		11							
H 2053	261	17	17	17	17	17	17	17	17	17	17	3	17	17	17	
H 3002	40	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 3003A	20											23				
H 3004	50		20	3	3	3	3	7	7	7	2	2	2	2	2	
H 3005	80	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
H 3008	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3010	388	2	2	2	19	19	19	19	19	19	1	6	6	6	6	
H 3012	40	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
H 3013	40	13	13	13	22	13	13	13	13	13	13	13	13	13	13	
H 3021	30	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
H 3027	80	10	10	10	10	10	10	10	10	10	10	10	10	9	16	
H 3503	140	13	13	13	13	13	13	13	13	13	13	13	13	13	13	
MA 004	235	8	8	8	8	8	8	8	8	8	8	8	8	8	8	
MA 005	235	8	8	8	8	8	8	8	8	8	10	10	10	10	10	
MA 041	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
MA 042	140	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
MA 043	152	11	11	11	11	11	11	11	11	11	11	11	11	11	11	
MA 141	70	14	14	14	14	14	14	14	14	14	14	14	14	14	14	
MA 144	68	21	21	21	21	21	21	21	21		21	21	21	21	21	
MA 313	40	15	15	15	15	15	15	15	15	15	15	15	15	15	15	
MA 376	32	23			23	23	23	23	23	23	23	23	23	23	23	
MA 415	50	7	7	7	7	7	7	7	7	7	7	7	7	7	7	
MA 549	32	6		6	6	6	6	16	16	16		16			9	
MA 550	32	23	23	23	23	23	23	5	5	5	6	5		5	5	
MA 649	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
MA 650	32	18	18	18	18	18	18	18	18	18	18	18	18	18	18	

Capacity view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	36	36	36	42	34	34	56	56	56	101	31	31	57	70	57
H 1058	263	73	73	44	73	73	73	73	73	73	73	73	73	73	73	73
H 1012	262	62	59	66	61	61	65	39	39	39	52	61	39	39	39	39
H 2013	261	54	54	54	54	54	54	54	54	54	54	54	54	54	54	83
H 2053	261	60	60	60	60	36	60	60	60	60	36	59	137	60	60	60
H 2032	236	101	101	101	101	101	101	101	101	101	101	101	101	101	128	101
MA 005	235	53	53	53	53	53	53	32	32	48	48	48	48	48	48	48
MA 004	235	53	53	53	53	53	53	53	53	87	53	53	53	53	53	53
H 1028	231	39	39	39	69	39	39	39	39	39	39	39	39	88	55	96
H 0110	198	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
MA 041	152	48	48	48	48	48	48	48	48	48	29	48	48	48	48	48
MA 043	152	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
H 0107	144	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 3503	140	40	40	40	40	55	40	40	40	40	40	40	40	40	40	40
MA 042	140	66	66	60	48	48	48	66	66	58	48	48	48	60	60	60
H 0106	99	43	43	42	36	42	41	36	36	36	42	43	42	45	36	36
H 0111	99	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
H 0112	99	36	36	36	21	36	36	36	36	36	36	36	36	36	21	21
H 3005	80	34	34	34	34	34	34	56	49	52	34	34	34	34	34	34
H 3027	80	48	48	48	48	48	48	48	48	48	71	68	48	54	45	54
MA 141	70	55	55	55	59	55	55	55	55	55	55	55	55	33	33	33
MA 144	68	59	59	59	59	59	59	59	36		59	59	59	59	59	59
H 2033	67	37	37	65	37	37	37	37	37	37	37	37	37	37	37	37
H 3004	50		44	37	22	37	37	25	25	42	36	36	36	36	36	36
MA 415	50	42	42	42	42	42	42	42	42	42	42	42	42	42	42	42
H 2038	50	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45
H 2036	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45
H 1029	41	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39
H 3002	40	34	34	34	20	20	34	34	34	34	34	34	34	34	34	34
MA 313	40	39	39	39	23	39	23	39	39	39	39	39	39	39	39	39
H 3012	40	34	34	34	34	19	34	34	34	34	34	34	34	34	34	34
H 3013	40	40	40	40	39	40	23	40	40	40	40	40	40	40	40	40
MA 649	32	18	31	31	31	31	31	31	31	31	31	31	31	31	31	31
MA 376	32	31			31	31	31	31	31	31	31	31	31	31	31	31
MA 549	32	31		31	31	31	31	26	26	26		26			32	
MA 650	32	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
MA 550	32	31	31	31	31	31	31	24	24	24	31	24		24	24	24
H 3008	30	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
H 2051	30	13	24	24		21	21		29							
H 3021	30	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
H 2035	30	25	25	25	25	25	25	25	25	25	25	25	25	14		
H 3003A	20												18			

Robust view

room	capacity	Monday			Tuesday			Wednesday			Thursday			Friday		
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15
H 3010	388	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1058	263	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1012	262	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2013	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2053	261	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 2032	236	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.99	1.0
MA 005	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 004	235	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 1028	231	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0110	198	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 041	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 043	152	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 0107	144	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
H 3503	140	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
MA 042	140	0.99	0.99	0.99	1.0	1.0	1.0	0.99	0.99	1.0	1.0	1.0	1.0	0.99	0.99	0.99
H 0106	99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 0111	99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
H 0112	99	0.99	0.99	0.99	1.0	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.0	1.0
H 3005	80	0.99	0.99	0.99	0.99	0.99	0.99	0.6	0.91	0.76	0.99	0.99	0.99	0.99	0.99	0.99
H 3027	80	0.91	0.91	0.93	0.91	0.92	0.92	0.92	0.92	0.91	0.05	0.11	0.92	0.69	0.96	0.67
MA 141	70	0.24	0.23	0.27	0.09	0.25	0.26	0.24	0.25	0.22	0.24	0.26	0.24	0.99	0.99	0.99
MA 144	68	0.05	0.05	0.07	0.05	0.06	0.06	0.06	0.96		0.06	0.06	0.05	0.05	0.05	0.05
H 2033	67	0.93	0.92	0.01	0.92	0.93	0.94	0.93	0.93	0.92	0.93	0.93	0.93	0.93	0.92	0.92
H 3004	50		0.02	0.25	0.97	0.23	0.23	0.92	0.92	0.06	0.26	0.27	0.26	0.25	0.24	0.24
MA 415	50	0.06	0.06	0.08	0.06	0.07	0.07	0.06	0.07	0.06	0.06	0.07	0.06	0.06	0.06	0.05
H 2038	50	0.02	0.02	0.03	0.02	0.03	0.03	0.02	0.03	0.02	0.02	0.03	0.02	0.02	0.02	0.02
H 2036	45	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
H 1029	41	0.0	0.0	0.01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
H 3002	40	0.04	0.04	0.05	0.82	0.83	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.03
MA 313	40	0.0	0.0	0.0	0.6	0.0	0.63	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
H 3012	40	0.04	0.04	0.05	0.04	0.84	0.05	0.04	0.05	0.04	0.04	0.05	0.04	0.04	0.04	0.04
H 3013	40	0.0	0.0	0.0	0.0	0.0	0.58	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MA 649	32	0.47	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MA 376	32	0.0			0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MA 549	32	0.0		0.0	0.0	0.0	0.0	0.04	0.04	0.03		0.04			0.0	
MA 650	32	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MA 550	32	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.11	0.09	0.0	0.11		0.1	0.09	0.09
H 3008	30	0.03	0.03	0.03	0.02	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02
H 2051	30	0.76	0.05	0.06		0.15	0.15		0.0							
H 3021	30	0.03	0.03	0.03	0.02	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02
H 2035	30	0.04	0.03	0.04	0.03	0.04	0.04	0.04	0.04	0.03	0.04	0.04	0.04	0.72		
H 3003A	20												0.01			

References

- [1] S. Daskalaki, T. Birbas, E. Housos: *An integer programming formulation for a case study in university timetabling*, European Journal of Operational Research 153 (2004) 117-135.
- [2] G. Lach, M.E. Luebbecke: *Optimal University Course Timetables and the Partial Transversal Polytope*, C.C. McGeoch (Ed.): WEA 2008, LNCS 5038, pp. 235-248, 2008.
- [3] G. Lach, M.E. Luebbecke: *Curriculum based course timetabling: new solutions to Udine benchmark instances*, Ann. Oper. Res. 194(1): 255-272, 2012.
- [4] M. Groetschel, Y. Wakabayashi: *Facets of the clique partitioning polytope*, Mathematical Programming 47 (1990) 367-387.
- [5] E. M. Macambira, C.C. de Souza: *The Edge-Weighted Clique Problem: valid inequalities facets and polyhedral computations*, Relatório Técnico IC-97-14, 1997.
- [6] K. Park, K. Lee, S. Park: *An extended formulation approach to the edge-weighted maximal clique problem*, European Journal of Operational Research 95 (1996) 671-682.
- [7] B. Alidaee, F. Glover, G. Kochenberger, H. Wang: *Solving the maximum edge weight clique problem via unconstrained quadratic programming*, European Journal of Operational Research 181 (2007) 592-597.
- [8] D. Werra: *An introduction to timetabling*, European Journal of Operational Research 19 (1985) 151-162 .
- [9] C.P. Rangan, M.S. Chang, G.J. Chang, C.K. Wong: *The Vertex-Disjoint Triangles Problem*, J. Hromkovic, O. Sykora (Eds.): WG'98, LNCS 1517, pp. 26-37, 1998.
- [10] R. Hassin, S. Rubinstein: *An approximation algorithm for maximum triangle packing*, Journal Discrete Applied Mathematics Volume 154 Issue 6 (2006) 971-979.
- [11] G. Manic, Y. Wakabayashi: *Packing triangles in low degree graphs and indifference graphs*, EuroComb 2005 DMTCS proc. AE, 2005, 251-256.
- [12] Z.Z. Chen, R. Tanahashi, L. Wang: *An Improved Randomized Approximation Algorithm for Maximum Triangle Packing*, R. Fleischer and J. Xu (Eds.): AAIM 2008, LNCS 5034, pp. 97-108, 2008.
- [13] S.Tayur, M.Dawande, P. Keskinocak, J. M. Swaminathan: *On Bipartite and Multipartite Clique Problems*, Journal of Algorithms 41, 388-403 (2001).
- [14] M.R. Garey, D.S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*.

- [15] D.A. Jackson, K.M. Somers, H.H. Harvey: *Similarity Coefficients: Measures of co-occurrence and association or simply measures of occurrence?* Am. Nat. 1989, Vol. 133, pp. 436-453.
- [16] J.R. Correa, N. Megow, R. Raman, K. Suchan: *Cardinality Constrained Graph Partitioning into Cliques with Submodular Costs*, Preprint submitted to CTW 2009.
- [17] Z. Svitkina, L. Fleischer: *Submodular Approximation: Sampling-based Algorithms and Lower Bounds*, May 31, 2010
- [18] A. N. Letchford, M. M. Soerensen: *A New Separation Algorithm for the Boolean Quadric and Cut Polytopes*, July 2011, Revised November 2012
- [19] M. Hunting, U. Faigle, W. Kern: *A Lagrangian relaxation approach to the edge-weighted clique problem*, 23 August 1999,
- [20] E. Boros, I. Lari: *Some properties of the Boolean Quadric Polytope*
- [21] B. Winter *Mathematische Optimierungsverfahren für die Konferenzplanung*, August 2011
- [22] M. Grötschel, Y. Wakabayashi: *Composition of Facets of the Clique Partitioning Polytope*, R. Bodendiek R. Henn (Hrsg.), Topics in Combinatorics and Graph Theory, Essays in Honour of Gerhard Ringel, Physica-Verlag, Heidelberg, 1990, 271-284.B.
- [23] Xiaoyun Ji, John E. Mitchell: *The Clique Partition Problem with Minimum Clique Size Requirement*, Journal Discrete Optimization archive Volume 4 Issue 1 Pages 87-102 , March 2007
- [24] A. Mehrotra and M. A. Trick: *Cliques and Clustering: A Combinatorial Approach*, Operations Research Letters, 22:1-12, 1998.
- [25] SPIEGEL Online: *Das optimieren wir jetzt mal*, 23.08.2012.
<http://www.spiegel.de/wissenschaft/mensch/mathematiker-optimieren-eigenen-kongress-a-851157.html>