

国際会議における発表スケジュールの最適化

2012SE022 五上 裕士 2012SE036 服部 晃大

指導教員：佐々木美裕

1 はじめに

大規模な国際会議では複数のセッションが同時に開催されることが多い。セッションは各時限各部屋で1つ開催される。参加者にとって、参加希望のセッションが同一時限帯に開催されることは参加希望のセッションに参加できない場合や、セッションの途中で別のセッションに移動しなければならない場合があり、望ましくない。また、参加希望のセッションが離れた部屋で開催される場合、部屋の移動に時間がかかることがある。これらの問題点を考慮すると、関連性の高いセッションが別の時間帯で開催され、さらに参加者の部屋の移動の負担が少なくなることが望ましい。

本研究では、伊藤 [2] が提案した部屋割り当てモデルを参考に、スケジュール作成モデルを提案する。

2 国際会議の概要

OR 分野で大規模な会議である INFORMS Annual Meeting(以下, INFORMS) の実際のスケジュールの一部を図 1 に示す。INFORMS では、各研究部会が 1 つまたは複数のクラスタを設定し、研究部会に所属するメンバーがセッションを開催する。

Track	Room	SA 8:00-9:30	Plenary 10:00-10:50	SB 11:00-12:30	SC 1:30-3:00	Plenary 3:10-4:00	SD 4:30-6:00
1	C - Room 201A	Nicholson Student Prize		Nicholson Student Prize	MADM - SC Ops		MADM
2	C - Room 201B	Risk Mgt.		Risk Mgt.	Risk Mgt.		Opt. in Finance
3	C - Room 202A	Computer Science		Opt. - Comp. Opt. & Soft.	Women in OR/MS		Women in OR/MS
4	C - Room 202B	Computing		Computing	Computing		Computing
5	C - Room 203A	Supply Chain		Tutorials	Tutorials		Tutorials

図 1 2011 年 INFORMS の実際のスケジュール

INFORMS の開催は、通常、日曜日から水曜日の 4 日間にわたり、1 日に A, B, C, D の 4 つの時間帯において一般発表が行われる。表 1 に各曜日、各時限ごとの名前を示す。

表 1 各曜日各時限における名前

	1 時限	2 時限	3 時限	4 時限
日曜日	SA	SB	SC	SD
月曜日	MA	MB	MC	MD
火曜日	TA	TB	TC	TD
水曜日	WA	WB	WC	WD

INFORMS における発表件数は、増加傾向にあり、複数の建物(国際会議場やホテル)を用いて開催されることも多い。セッション数の増加、発表会場数の増加に伴い、参加者の移動の負担も年々大きくなっていると考えられる。

3 モデルの説明

3.1 先行研究

伊藤 [2] は、発表スケジュールを所与として、関連性の高いセッション同士を近い部屋に割り当てる方法を提案した。初めに SA の部屋割りを決定し、SA の部屋割りを参照して SB を決定する。次に SB の部屋割りを参照して SC を決定する。これらを繰り返し、最後の WD まで逐次的に部屋割りを行うモデルである。直前の時限を参照し部屋割りを決めるこのモデルでは、SA と SB など近い時限で比べると関連性の高いセッションが近い部屋に割り当てられるが、SA と WA など離れた時限で比べると関連性の高いセッションが離れた部屋で開催されてしまうなどの不都合が生じることがある。

3.2 本研究におけるアプローチの概要

本研究では、はじめに述べた問題と、伊藤のモデルにおける問題を解決するために、関連性の高いクラスタを同一の部屋に割り当てる方法を提案する。これを行うために、各クラスタが開催するセッションの数が所与であることに着目し、あらかじめ関連性の高いクラスタを複数のグループに分け、そのグループを部屋に割り当てる。グループを作成し、部屋に割り当てることで関連性の高いクラスタのセッションが同一時間帯で開催されることを抑えることができるため、参加したいセッションに参加しやすいスケジュールを作成できる。また、関連性の高いセッションをできるだけ同じ部屋で開催できるようになるため、部屋を移動する回数が減り、移動時間の短縮にもつながる。

4 言葉の定義

4.1 クラスタグループ

クラスタ間において関連性の高いクラスタをまとめたグループをクラスタグループと呼ぶことにする。クラスタグループを部屋に割り当てるため、開催日が日曜日から水曜日までの 4 日間であることと 1 日に 4 コマ開催されることを考慮すると、各クラスタグループに含まれるセッションの数の合計を 16 以下にしなければならない。また、クラスタグループをさらにグループ化する際、そのグループはクラスタグループバンドルと呼ぶ。

4.2 クラスタの規模

部屋容量を考慮した部屋割り当てを行うために、クラスタの規模を決定する数値を定義する。規模を示す数値は 1, 2, 3 とし、それぞれ小, 中, 大とする。規模の大きなクラスタが開催するセッションは参加人数が多いと考えられるため、割り当てられる部屋も大きくなければならない。

4.3 クラスタグループの規模

クラスタグループの規模はそのグループ内に含まれているクラスタの規模の最大値とする。クラスタをクラスタグループにまとめる際に規模が2段階離れているクラスタ同士は、たとえ関連性が高くとも同じクラスタグループにまとめないようにする。

4.4 関連度

クラスタグループを作成するにあたり、クラスタ間の関連性を表す関連度という数値を用いる。関連度は1から5の5段階とし、数値が高ければ関連性が高いクラスタとして扱う。表2に関連度の例を示す。

表2 関連度の例

	A	B	C	D
A	5	3	3	4
B	3	5	3	3
C	4	3	5	3
D	4	3	2	5

i 行のクラスタから見た j 列のクラスタへの関連性の数値を表す。クラスタ A から見たクラスタ B の関連度は3であり、クラスタ B から見たクラスタ A の関連度は3である。しかし、クラスタ C から見たクラスタ D の関連度は3であり、クラスタ D から見たクラスタ C の関連度は2となる。このように、関連度の数値はクラスタの視点によって異なる場合があることも想定している。また、クラスタ i からクラスタ $i (i \in A, B, C, D)$ の数値は常に最高値である5をとるように設定する。

5 定式化

このモデルを定式化するにあたり、第1段階(クラスタグループの作成)、第2段階(クラスタグループバンドルの作成)、第3段階(部屋割り当ての最適化)に分割して定式化する。

5.1 クラスタグループの作成

クラスタグループを作成する方法として、本研究では2つの解法を提案する。

ここでクラスタグループをビン、クラスタの容量をビンの容量と考えるとこの問題はビンパッキング問題 [1][3] として定式化できる。

以下の記号を定義する。

I : クラスタの集合

K : クラスタグループの集合

B : クラスタグループの容量

L : クラスタをクラスタグループに入れる際の関連度の下限

N_i : クラスタ $i \in I$ のセッション数

G_{ij} : クラスタ $i \in I$ から見たクラスタ $j \in I$ の関連度

R_i : クラスタグループ $i \in I$ に属するクラスタが開催するセッションの規模

変数は以下の通りである。

$$x_{ijk} = \begin{cases} 1: \text{クラスタグループ } k \in K \text{ にクラスタ} \\ \quad i \in I \text{ とクラスタ } j \in I \text{ が入るとき} \\ 0: \text{上記以外} \end{cases}$$

$$y_{ik} = \begin{cases} 1: \text{クラスタ } i \in I \text{ がクラスタ} \\ \quad \text{グループ } k \in K \text{ に入るとき} \\ 0: \text{上記以外} \end{cases}$$

$$\text{Max. } \sum_{i \in I} \sum_{j \in I} \sum_{k \in K} G_{ij} x_{ijk} \quad (1)$$

$$\text{s.t. } G_{ij} \geq L x_{ijk}, \quad i \in I, j \in I, k \in K \quad (2)$$

$$\sum_{k \in K} y_{ik} = 1, \quad i \in I \quad (3)$$

$$\sum_{i \in I} N_i y_{ik} \leq B, \quad k \in K \quad (4)$$

$$\frac{1}{2}(y_{ik} + y_{jk}) \geq x_{ijk}, \quad i \in I, j \in I, k \in K \quad (5)$$

$$(R_i - R_j) x_{ijk} \leq 1, \quad i \in I, j \in I, k \in K \quad (6)$$

$$(R_i - R_j) x_{ijk} \geq -1, \quad i \in I, j \in I, k \in K \quad (7)$$

$$x_{ijk} \in \{0, 1\}, \quad i \in I, j \in I, k \in K \quad (8)$$

$$y_{ik} \in \{0, 1\}, \quad i \in I, k \in K \quad (9)$$

目的(1)は、同じクラスタグループ $k \in K$ に入るクラスタ $i \in I$ とクラスタ $j \in I$ の関連度 $G_{ij} (i \in I, j \in I)$ を最大化し、使用する部屋数を最小化することである。制約式(2)は、クラスタ $i \in I$ から見たクラスタ $j \in I$ の関連度 $G_{ij} (i \in I, j \in I)$ が L 以下ならば同じクラスタグループ $k \in K$ に入れないための制約である。制約式(3)はクラスタ $i \in I$ は必ずいずれかのクラスタグループ $k \in K$ に入るという制約である。制約式(4)は各クラスタグループ $k \in K$ に入るセッションの合計はクラスタグループの容量 B 以下であるという制約である。制約式(5)はクラスタ $i \in I$ とクラスタ $j \in I$ が同じクラスタグループ $k \in K$ に入らなければ変数 $x_{ijk} (i \in I, j \in I, k \in K)$ は1をとらないという制約である。制約式(6), (7)はクラスタ $i \in I$ とクラスタ $j \in I$ の規模が近ければ同じクラスタグループに入れる制約である。制約式(8), (9)は変数のバイナリ制約である。

クラスタグループの候補を全列挙し、候補から実際に使用するクラスタグループを選択する問題を集合分割問題として解く。クラスタグループの候補は、クラスタ $i (i \in I)$ から見た関連度が高いクラスタ $j (j \in I)$ を候補として使用する。クラスタ i から見た関連度を使用するので、生成されるクラスタグループの候補の中には必ずクラスタ i が含まれる。クラスタグループの候補を作成する際、クラスタグループ内のセッション数が16以下となる組み合わせを全てクラスタグループの候補とする。集合分割問題を用いた解法を以下に示す。

以下の記号を追加する。

K : クラスタグループの候補の集合

c_{jk} : クラスタグループの候補 $k \in K$ にクラスタ $j \in I$ が入っていたら 1 入っていないならば 0 をとる定数
変数は以下の通りである.

$$x_k = \begin{cases} 1: \text{クラスタグループの候補 } k \in K \text{ を} \\ \text{選択した場合} \\ 0: \text{上記以外} \end{cases}$$

$$\text{Min. } \sum_{k \in K} x_k \quad (1)$$

$$\text{s.t. } \sum_{k \in K} c_{jk} x_k = 1, \quad j \in I \quad (2)$$

$$x_k \in \{0, 1\}, \quad k \in K \quad (3)$$

目的 (1) の意味は部屋数を最小化 (使用するクラスタグループの候補の数の最小化) することである. 制約式 (2) はクラスタグループの候補 $k \in K$ を選択する際, クラスタ $j \in I$ が被らないようにするための制約である. 制約式 (3) は変数のバイナリ制約である.

5.2 クラスタグループバンドルの作成

クラスタグループバンドルの作成の問題を考える. ここでクラスタグループバンドルをビン, クラスタグループの容量をビンの容量と考えるとこの問題はビンパッキング問題として定式化できる.

以下の記号を追加する.

I : 1 段階目で作成したクラスタグループの集合

K : クラスタグループバンドルの集合

B : クラスタグループバンドルの容量

N_i : クラスタグループ $i \in I$ がもつセッション数

P_i : クラスタグループ $i \in I$ に属するクラスタの規模の最大値

Q_i : クラスタグループ $i \in I$ に属するクラスタの規模の最小値

変数は以下の通りである.

$$x_{ijk} = \begin{cases} 1: \text{クラスタグループ } i \in I \text{ とクラスタ} \\ \text{グループ } j \in I \text{ をクラスタグループ} \\ \text{バンドル } k \in K \text{ に割り当てるとき} \\ 0: \text{上記以外} \end{cases}$$

$$y_{ik} = \begin{cases} 1: \text{クラスタグループ } i \in I \text{ を} \\ \text{クラスタグループバンドル } k \in K \text{ に} \\ \text{割り当てるとき} \\ 0: \text{上記以外} \end{cases}$$

$$z_k = \begin{cases} 1: \text{クラスタグループバンドル } k \in K \text{ が} \\ \text{使われるとき} \\ 0: \text{上記以外} \end{cases}$$

$$\text{Min. } \sum_{k \in K} z_k \quad (1)$$

$$\text{s.t. } \sum_{k \in K} y_{ik} = 1, \quad i \in I \quad (2)$$

$$\sum_{i \in I} N_i y_{ik} \leq B z_k, \quad k \in K \quad (3)$$

$$(P_i - Q_j) x_{ijk} \leq 1, \quad i \in I, j \in I, k \in K \quad (4)$$

$$(P_i - Q_j) x_{ijk} \geq -1, \quad i \in I, j \in I, k \in K \quad (5)$$

$$x_{ijk} = x_{jik}, \quad i \in I, j \in I, k \in K \quad (6)$$

$$\frac{1}{2}(y_{ik} + y_{jk}) \geq x_{ijk}, \quad i \in I, j \in I, k \in K \quad (7)$$

$$y_{ik} + y_{jk} - 1 \leq x_{ijk}, \quad i \in I, j \in I, k \in K \quad (8)$$

$$x_{ijk} \in \{0, 1\}, \quad i \in I, j \in I, k \in K \quad (9)$$

$$y_{ik} \in \{0, 1\}, \quad i \in I, k \in K \quad (10)$$

$$z_k \in \{0, 1\}, \quad k \in K \quad (11)$$

目的 (1) は, 使用する部屋数の最小化である. 制約式 (2) はクラスタグループ $i \in I$ はいずれかのクラスタグループバンドル $k \in K$ に入るという制約である. 制約式 (3) は各クラスタグループバンドル $k \in K$ に入るセッションの合計はクラスタグループバンドルの容量 B 以下であるという制約である. 制約式 (4), (5), (6) はクラスタグループ $i \in I$ とクラスタグループ $j \in I$ に属するクラスタの規模の差の絶対値が 2 でなければ同じクラスタグループバンドルに入れる制約である. 制約式 (7), (8) はクラスタグループ $i \in I$ とクラスタグループ $j \in I$ が同じクラスタグループバンドル $k \in K$ に入らなければ変数 x_{ijk} ($i \in I, j \in I, k \in K$) は 1 をとらないという制約である. 制約式 (9), (10), (11) は変数のバイナリ制約である.

5.3 部屋割り当ての最適化

第 2 段階で作成したクラスタグループバンドルを部屋に割り当てる.

以下の記号を追加する.

I : クラスタグループバンドルの集合

K : 部屋の集合

G_{ij} : クラスタグループバンドル $i \in I$ と

クラスタグループバンドル $j \in I$ の関連度

D_{kl} : 部屋 $k \in K$ と部屋 $l \in K$ の距離

R_i : クラスタグループバンドル $i \in I$ の規模

C_k : 部屋 $k \in K$ の容量

L : 関連度の高いクラスタグループバンドルを近くの部屋に割り当てるときの関連度の下限

変数は以下の通りである.

$$x_{ijkl} = \begin{cases} 1: \text{関連性の高い 2 つのクラスタグループ} \\ \text{バンドル } i \in I \text{ とクラスタグループ} \\ \text{バンドル } j \in I \text{ を選び, 2 つのクラスタ} \\ \text{グループバンドルを距離が近い部} \\ \text{屋 } k \in K \text{ と部屋 } l \in K \text{ に割り当てるとき} \\ 0: \text{上記以外} \end{cases}$$

$$y_{ik} = \begin{cases} 1: \text{クラスタグループバンドル } i \in I \text{ が部} \\ \text{屋 } k \in K \text{ に割り当てられるとき} \\ 0: \text{上記以外} \end{cases}$$

$$\text{Min. } \sum_{i \in I} \sum_{j \in I} \sum_{k \in K} \sum_{l \in K} D_{kl} x_{ijkl} \quad (1)$$

$$\text{s.t. } \sum_{k \in K} y_{ik} = 1, \quad i \in I \quad (2)$$

$$\sum_{i \in I} y_{ik} \leq 1, \quad i \in I \quad (3)$$

$$(C_k - R_i) y_{ik} \leq 1, \quad i \in I, k \in K \quad (4)$$

$$(C_k - R_i) y_{ik} \geq 0, \quad i \in I, k \in K \quad (5)$$

$$\sum_{k \in K} \sum_{l \in K} x_{ijkl} = 1, \quad i \in I, j \in I, G_{ij} \geq L \quad (6)$$

$$\frac{1}{2}(y_{ik} + y_{jk}) \geq x_{ijk}, \quad i \in I, j \in I, k \in K \quad (7)$$

$$x_{ijkl} \in \{0, 1\}, \quad i \in I, j \in I, k \in K, l \in K \quad (8)$$

$$y_{ik} \in \{0, 1\}, \quad i \in I, k \in K \quad (9)$$

目的 (1) は、関連度 $G_{ij} (i \in I, j \in I)$ の高いクラスタグループバンドル $i \in I$ とクラスタグループバンドル $j \in I$ を選択し、なるべく近い部屋に割り当てることである。制約式 (2) はクラスタグループバンドル $i \in I$ はいずれかの部屋 $k \in K$ に入るという制約である。制約式 (3) は部屋 $k \in K$ に割り当てられるクラスタグループバンドル $i \in I$ はひとつまでという制約である。制約式 (4), (5) はクラスタグループバンドル $i \in I$ の規模と部屋 $k \in K$ の容量が近ければクラスタグループバンドル $i \in I$ を部屋 $k \in K$ に割り当てる制約である。制約式 (6) は、クラスタグループバンドル $i \in I$ からみたクラスタグループバンドル $j \in I$ の関連度 G_{ij} が L 以上ならば $x_{ijkl} (k \in K, l \in K)$ の値は 1 となる制約である。制約式 (7) は $y_{ik} (i \in I, k \in K)$ と $y_{jl} (j \in I, l \in K)$ がともに 1 であるときに x_{ijk} は 1 をとるという制約である。制約式 (8), (9) は変数のバイナリ制約である。

6 計算実験と考察

使用した最適化ソフトウェアは Gurobi 6.5.0 であり、計算環境は (プロセッサ: Intel(R) Core(TM) i7-6700K CPU @ 4.00GHz 4.00GHz 実装メモリ: 32GB) である。

はじめに、実データを用いた計算実験を行う前に、仮データを 2 つ作成し計算実験を行った。仮データの詳細は表 3 のようにした。

表 3 仮データの詳細

クラスタ数	セッションの総数	部屋数
7	47	-
35	171	20

関連度、クラスタの規模、部屋容量はエクセルを使用し、ランダムデータを作成した。クラスタ数が 7 のデータでは第 1 段階のみを行ったため部屋数のデータはない。このデータを用いてビンパッキング問題の解法と集合分割問題の解法を比較した。結果、両方の解法ともに時間はほぼかかることなくクラスタグループを作成でき、クラスタグループは同じものが作成された。

次に、クラスタ数が 35 のデータを使用し、第 1 段階、第 2 段階、第 3 段階を通して計算実験を行った。結果、集合分割問題の解法では第 1 段階、第 2 段階、第 3 段階いずれも時間はほぼかかることなく結果が出たが、ビンパッキング問題の解法では第 1 段階の時点で計算に時間がかかり、結果が出せなかった。

ビンパッキング問題は、組み合わせの数が膨大になってしまい、計算に時間がかかり、結果が出なかったと考えられる。また、第 2 段階にもビンパッキング問題の考え方をを用いているが、集合分割問題の解法では第 1 段階で問題の規模を小さくしているため結果が出たと考えられる。

次に実データを用いた計算実験を行った。ビンパッキング問題の解法を用いた計算実験は仮データの段階で結果が出なかったため実データでの計算実験は行わなかった。実データの詳細は表 4 のようになる。

表 4 実データの詳細

クラスタ数	セッションの総数	部屋数
280	1060	69

関連度、クラスタの規模、部屋容量は手作業で作成した。実データを用いた計算実験では問題の規模が大きすぎたため、計算に時間がかかり、結果が出せなかった。

実データでは仮データの時と異なり、問題が第 2 段階にシフトしてもまだ規模が大きく、計算に時間がかかった。5 時間ほどで第 2 段階の結果は出たが、第 3 段階では 2 次割り当て問題を解くためここでも時間がかかり、結果を出すことができなかった。

今後の課題としてビンパッキング問題の考え方をを用いると、計算時間が膨大になってしまうので、この考え方を使用しない別の案を考える。また、2 次割り当て問題の規模が大きすぎたので、規模を小さくする工夫をすることである。

参考文献

- [1] 藤澤克樹, 梅谷俊治: 「応用に役立つ 50 の最適化問題」, 朝倉書店, 2009.
- [2] 伊藤喜紀: 「国際会議における部屋割り当て問題のモデル化とその解法」, 南山大学大学院 理工学研究科, 2014 年度修士論文, 2015.
- [3] 宮本裕一郎: 「はじめての列生成法」, オペレーションズ・リサーチ誌 2012 年 4 月号 PP.1-3, 2012.