数理解析研究所講究録
第1981巻 2016年 98-116



# 実務で現れるスタッフスケジューリングに対する近似解法

廣瀬 貴也 (Takaya Hirose)* 鈴木 翔太 (Shota Suzuki)† 佐藤 悠介 (Yusuke Sato)‡
鈴木 寛人 (Hiroto Suzuki)§ 中田 和秀 (Kazuhide Nakata)*

**概要**

スタッフスケジューリングとは，サービス業などで各スタッフが，いつ，どの時間帯で働き，どのような作業を行うのかを決定することである．スケジュールの質は店舗の円滑な運営やスタッフの満足度に大きく影響するため重要だが，多くの条件を考慮したスケジュールの作成は難しい．本研究ではスタッフスケジューリングを3つの段階に分割し，その各段階に対して実務に基づくモデルを構築し，近似解法を提案する．そして，実データを用いた数値実験により，それらの有効性を示す．

## 1 はじめに

スケジューリングとは，多くの仕事あるいは活動を種々の制約条件のもとで複数の対象物をどのような順番で行うか決定することである．スケジュールの質は仕事の効率，利益に直結するので良いスケジュールを作成することが重要である．スケジューリングは効率的な運用が求められる種々の分野で行われており，人の勤務，スポーツにおける対戦相手，生産，配送などが代表例としてある．その中でも人がいつ，どの時間帯で働き，どのような作業を行うか決定する問題をスタッフスケジューリング問題と呼ぶ．実際にスケジュールを作成するときには様々な条件を考慮しなければならず，作成者の大きな負担となっている．そこで，スケジュール作成者の負担を減らすために自動でスケジュールを作成できる仕組みが待ち望まれている．

スタッフスケジューリング問題はこれまで多く研究されてきた [6], [10], [12]．代表的なものは看護師スケジューリング問題であり古くから盛んに研究されている [3], [8], [10]．看護師スケジューリング問題は多くの複雑な条件を持つことが知られており，大規模な整数計画問題として定式化できるが，最適解を現実的な時間で求めるのは困難である．このような問題に対しては近似解法がよく用いられる．近似解法では最適解が求まる保証はないが，比較的短時間である程度良い解が求まることが多い．また，条件が少し変わっても新たに一からアルゴリズムを設計する必要はなく，計算時間が増大することもあまりない．そこで，タブーサーチ [2], [5]，遺伝的アルゴリズム [11]，アニーリング法 [7] などの近似解法によって看護師スケジューリング問題の近似解を求める手法が研究されている．また，別のアプローチとして問題の構造を使い，いくつかの部分問題を解くことで近似解を求める方法も提案されている [1], [13]．

本研究で扱うスタッフスケジューリングでは各スタッフが，いつ，どの時間帯に働き，どのような作業を行

* 東京工業大学 大学院社会理工学研究科
† 株式会社ディー・エヌ・エー
‡ 日本ユニシス株式会社
§ 株式会社三菱東京 UFJ 銀行

うのかを決定する．これらをすべて同時に決定することは困難であるため，問題を３つの段階に分割して決定することを提案する．しかし，問題を分割しても各段階において実用的な時間で最適解を得ることは難しい．そこで，本研究では各段階に対して近似解法を提案する．

本稿の構成は以下の通りである．２節でスタッフスケジューリングについて説明する．３節，４節，５節ではそれぞれ勤務シフトの生成，勤務シフト表の作成，作業分担表の作成について実務に基づくモデルを構築し，近似解法を提案する．６節では提案した手法の有効性を調べるための数値実験を行う．最後に７節で結論と今後の課題について述べる．

## 2　スタッフスケジューリング問題

スタッフスケジューリング問題はサービス業などで各スタッフが，いつ，どの時間帯に働き，どの作業を行うのかを決定する問題である．実用的なスケジュールを作成するためには考慮すべき条件が非常に多い．条件は店舗の運営に関するものと，スタッフの労働負荷に関するものの２つに分類することができる．店舗の運営に関する条件としては，作業の必要人数，同時に働かせたい，もしくは働かせたくないスタッフのペア，作業の熟練度などである．また，スタッフの労働負荷に関する条件としては，連続勤務日数，飛び休，勤務時間内での休憩の時間や回数などである．このような多くの条件を考慮し，各スタッフがいつ，どの時間帯で働き，どのような作業を行うのかということを全て同時に決定することは難しい．そこで，本研究ではスタッフスケジューリングを以下のように３つの段階に分けて行う．

**第１段階**　勤務シフトの生成
**第２段階**　勤務シフト表の作成
**第３段階**　作業分担表の作成

第１段階は勤務シフトの生成である．非正社員が主力となる店舗では各日における勤務シフトの種類と各勤務シフトに割り当てられるスタッフ数が決まっていない．そこで，各スタッフの希望勤務日，希望勤務時間などを基に勤務シフトの種類と各勤務シフトに割り当てられるスタッフ数を決定する．勤務シフトの生成については３節で詳しく述べる．第２段階は勤務シフト表の作成である．この段階では計画期間の各日に対し各スタッフに勤務シフトまたは休日を割り当てる．勤務シフト表の作成については４節で詳しく述べる．第３段階は作業分担表の作成である．この段階では勤務シフト表を基に１日ごとに勤務シフトが割り当てられたスタッフが各時刻でどの作業を行うのかを決定する．作業分担表の作成については５節で詳しく述べる．このように問題を分割しても，各段階で実用的な時間で最適解を得ることは難しいため，各段階に対して近似解法を提案する．

## 3　勤務シフトの生成

### 3.1　実務に基づいたモデル化

本研究では日々の状況に応じて様々な勤務シフトが割り当てられるアルバイトやパートスタッフなどの非正社員が勤務する小売店や商業施設におけるスタッフスケジューリングを扱う．このような店舗では予め勤務シフトの種類と各勤務シフトに割り当てられるスタッフ数が決まっていない．実務では図１のように各スタッフがスケジュール作成者に希望勤務日，希望勤務時間を提出し，出勤する場合は希望勤務時間内の勤務シフトを

割り当てるという手順が用いられている．なお，問題のモデル化はスケジューリングを専門とするコンサルティング会社である T 社と共同で行った．

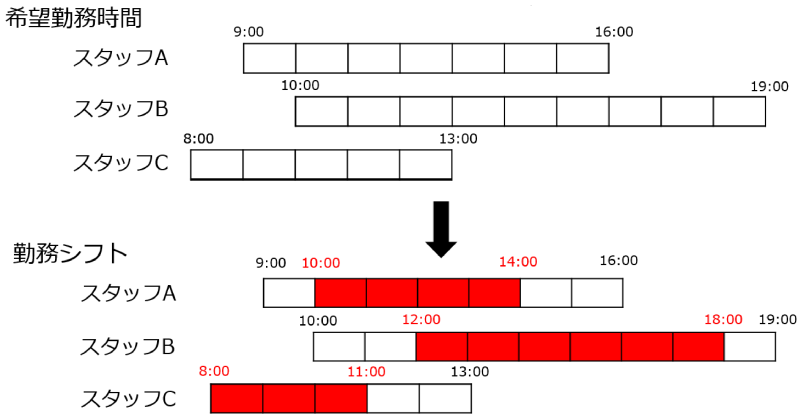


図 1　勤務シフト決定のイメージ

### 3.1.1　考慮する条件

次にモデル化をするにあたりコンサルティング会社との会合を重ねる中で要望があった勤務シフト生成問題において考慮する条件を述べる．

1. 希望勤務日，希望勤務時間
   現在の使われている手順でも考慮されており，希望勤務時間内の勤務シフトを生成する必要がある．
2. 作業スキル
   出勤したスタッフの人数は足りているが作業を行えるスタッフがいないといったことが起こる可能性があるため考慮の必要がある．
3. 休日数
   あるスタッフの希望勤務時間に依る勤務シフトが多く生成されると，勤務シフト表の作成において休日数を違反する可能性があり考慮する必要がある．
4. 固定シフト
   スタッフを特定の勤務シフトで出勤させたい場合にその勤務シフトが生成されなければ割り当てることができないため考慮する必要がある．

これらの条件は次の段階である勤務シフト表作成で必要となる条件の一部を抜き出したものであり，残りの条件を勤務シフト表作成で考慮する．

### 3.1.2　定式化

考慮する条件から勤務シフト生成問題の定式化を行う．各スタッフに対して割り当て可能な勤務シフトを仮割り当てしたときの各時刻の作業を考えることで勤務シフトの種類と各勤務シフトに割り当てられるスタッフ数を決定する．

勤務シフト生成問題は勤務シフトの種類と各勤務シフトに割り当てられるスタッフ数を決定することが目的であるが，その生成された勤務シフトを用いて店舗の業務を円滑に行えることが重要となる．業務を円滑に行うためには作業に対する必要な人数を満たすことが必要であるが，店舗データによってはそもそも必要な人数を満たすことができない可能性もあるため，制約として必要な人数を満たすことを入れることは難しい．そこで，目的関数を必要人数に対する不足人数としてその最小化を考えることで，どのようなデータに対しても最も必要人数を満たした勤務シフトを生成することができる．

集合の定義

$M$：正社員
$M_d$：$d$ 日に出勤出来る正社員
$M_d^F$：$d$ 日に固定勤務シフトのある正社員
$M_{dw}^W$：$d$ 日に作業 $w$ が出来る正社員
$S$：非正社員
$S_d$：$d$ 日に出勤出来る非正社員
$S_{dw}^W$：$d$ 日に作業 $w$ が出来る非正社員
$S_{dj}^P$：$d$ 日に勤務シフト $j$ が可能な非正社員
$F_{id}$：正社員 $i$ の $d$ 日の固定勤務シフト
$P$：全勤務シフト
$P_{id}$：非正社員 $i$ の $d$ 日の可能勤務シフト
$P_{idt}$：非正社員 $i$ の $d$ 日の時刻 $t$ における可能勤務シフト
$R$：登録勤務シフト
$R_t$：時刻 $t$ における可能登録勤務シフト
$W$: 作業
$W_i^M$：正社員 $i$ の出来る作業
$W_i^S$：非正社員 $i$ の出来る作業
$D$：スケジュール期間
$T$：時刻
$T_j$：勤務シフト $j$ の勤務時刻

定数の定義

$w_1, w_2$：重み
$N_{dtw}$：$d$ 日の時刻 $t$ における作業 $w$ の必要人数
$H_i^U$：非正社員 $i$ の休日数の上限
$H_i^L$：非正社員 $i$ の休日数の下限
$PU$：生成する勤務シフト数の上限

変数の定義

$$x_{ijdtw}=\begin{cases}1 & \text{非正社員 } i \text{ が勤務シフト } j \text{ で } d \text{ 日に時刻 } t \text{ で作業 } w \text{ を行う}\\ 0 & \text{それ以外}\end{cases}$$

$$y_{ijdtw}=\begin{cases}1 & \text{正社員iが勤務シフト } j \text{ で } d \text{ 日に時刻 } t \text{ で作業 } w \text{ を行う}\\ 0 & \text{それ以外}\end{cases}$$

$$p_{ijd}=\begin{cases}1 & \text{非正社員 } i \text{ に勤務シフト } j \text{ を } d \text{ 日に割り当て}\\ 0 & \text{それ以外}\end{cases}$$

$$r_{ijd}=\begin{cases}1 & \text{正社員 } i \text{ に勤務シフト } j \text{ を } d \text{ 日に割り当て}\\ 0 & \text{それ以外}\end{cases}$$

$$d_{jd}=\begin{cases}1 & \text{勤務シフト } j \text{ を } d \text{ 日に使う}\\ 0 & \text{それ以外}\end{cases}$$

$s_{dtw}$：$d$ 日の時刻 $t$ における作業 $w$ の不足人数

定式化

$$\text{minimize } w_1 \sum_{d\in D}\sum_{t\in T}\sum_{w\in W} s_{dtw} + w_2 \sum_{d\in D}\sum_{i\in S_d}\sum_{j\in P_{id}} |T_j| p_{ijd} \tag{1}$$

$$\text{subect to } \sum_{i\in S_{dw}^W}\sum_{j\in P_{idt}} x_{ijdtw} + \sum_{i\in M_{dw}^W}\sum_{j\in R_t} y_{ijdtw} + s_{dtw} \geq N_{dtw} \quad \forall d\in D, t\in T, w\in W \tag{2}$$

$$\sum_{w\in W_i^M} y_{ijdtw} = 1 \quad \forall i\in M_d^F, d\in D, j\in F_{id}, t\in T_{F_{id}} \tag{3}$$

$$\sum_{j\in R} r_{ijd} = 1 \quad \forall i\in M, d\in D \tag{4}$$

$$\sum_{j\in P_i} p_{ijd} \leq 1 \quad \forall s\in S, d\in D \tag{5}$$

$$\sum_{w\in W_i^S} x_{ijdtw} = p_{ijd} \quad \forall i\in S, j\in P_{id}, d\in D, t\in T_j \tag{6}$$

$$\sum_{w\in W_i^M} y_{ijdtw} = r_{ijd} \quad \forall i\in M, j\in R, d\in D, t\in T_j \tag{7}$$

$$H_i^L \leq \sum_{d\in D}\sum_{j\in P_{id}} (1-p_{ijd}) \leq H_i^U \quad \forall i\in S \tag{8}$$

$$p_{ijd} \leq d_{dj} \quad \forall i\in S, j\in P_{id}, d\in D \tag{9}$$

$$\sum_{j\in P}\sum_{d\in D} d_{jd} \leq PU \tag{10}$$

$$x_{ijdtw} \in \{0,1\} \quad \forall i\in S, j\in P_{id}, d\in D, t\in T_j, w\in W_i^S \tag{11}$$

$$y_{ijdtw} \in \{0,1\} \quad \forall i\in M, j\in R, d\in D, t\in T_j, w\in W_i^M \tag{12}$$

$$p_{ijd} \in \{0,1\} \quad \forall i\in S, j\in P_{id}, d\in D \tag{13}$$

$$r_{ijd} \in \{0,1\} \quad \forall i\in M, j\in R, d\in D \tag{14}$$

$$d_{jd} \in \{0,1\} \quad \forall j\in P, d\in D \tag{15}$$

$$s_{dtw} \in \mathbb{Z}_+ \quad \forall d\in D, t\in T, w\in W \tag{16}$$

目的関数 (1) は作業に対する作業に対する不足人数総和とスタッフの総労働時間の重み和である．不足人数総和の最小化が本問題の目的であるが，これのみを目的関数とすると余剰スタッフが発生する可能性があり，生成される勤務シフトが実用上使えないものとなってしまう．そこで，目的関数に総労働時間を追加し，重みを掛けて $w_1 >> w_2$ とすることで不足人数総和の最小化を行いかつ余剰スタッフの発生を抑えている．

制約式 (2) は各時刻における各作業の必要人数を満たすようにする制約である．制約式 (3) は各正社員が固定勤務シフトを満たすようにする制約である．制約式 (4) は各正社員が登録勤務シフトで出勤するという制約である．制約式 (5) は各非正社員が出勤する場合は勤務希望時間内の勤務シフトで働くという制約である．制約式 (6)，制約式 (7) は各正社員，非正社員が各時刻に行う作業は 1 つとする制約である．制約式 (8) は各非正社員の休日数の上下限を満たすようにする制約である．制約式 (10) は生成される勤務シフトの上限数を満たすようにする制約である．

### 3.1.3 考慮する条件の選択

3.1.2 では勤務シフト生成問題の定式化を行った．しかし，このままでは規模の小さい店舗の問題でも変数と制約式の個数が数十万以上の規模の大きな問題となってしまい，整数計画ソルバー等を用いて解くことは困難である．したがって，3.1.1 で述べた条件を全て考慮すること難しい．そこで，本研究では問題の規模を小さくするために考慮する条件を限定して元のモデルに対する緩和モデルを用いることで変数と制約式の減少を図ることを試みる．考慮する条件であるが，希望勤務日，希望勤務時間は実務に基づいたモデルが考える上で必要不可欠であるので考慮する条件から外すことはできない．また，固定シフトに関しては固定シフトに関係する変数はなく，制約式も固定シフトの個数しかないため，考慮の有無による影響は少ない．したがって，緩和モデルには以下の 2 つが考えられる．

緩和モデル 1　休日数を考慮しないモデル
緩和モデル 2　作業スキルを考慮しないモデル

緩和モデル 1 では，スケジュール期間に関わる制約がなくなるため問題をスケジュール期間全体で解く必要がなくなる．したがって，問題を各日毎に分けて考えることができるようになり解く問題が小さくなる．緩和モデル 1 に関しては 3.2.1 で述べる．一方，緩和モデル 2 では，必要人数を作業ごとではなく全体として考えることができるようになり，変数と制約式の数が減り緩和モデル 1 と同様に解く問題が小さくなる．緩和モデル 2 に関しては 3.2.2 で述べる．

緩和モデルを用いることで生成される勤務シフトの質は下がる，しかし，休日数に関しては勤務シフト表の作成においても考慮されている条件であり，休日数の条件が緩い場合には考慮する必要は無い．同様に作業スキルにおいても全てのスタッフが作業をできる等の条件下であれば考慮する必要はない．このように店舗の環境に合わせた緩和モデルを用いることで，考慮する条件を限定したことによる影響は少なくなると思われる．

## 3.2 提案手法

### 3.2.1 休日数を考慮しないモデル

提案手法 1 は休日数を考慮しないモデルに対するものである．スタッフ数が少なく営業時間が短い小規模なデータであれば，このまま解いても勤務シフトを生成することも可能であるが，この問題も現実的な状況においては元問題と同様に実用的な時間内に最適解を求めることは困難である．実際に数値実験を行った中で最も規模の小さい小売店 S 社 C 店のデータについてソルバーを用いてが解いたが，1 時間掛けても最適解を求めることはできなかった．表 1 に数値実験で扱うデータを定式化したときの変数の個数と制約式の個数を示した．データやソルバーの詳細に関しては 6 節で説明する．

表 1　変数．制約の個数

| | 変数の個数 | 制約の個数 |
|---|---|---|
| 小売店 S 社 C 店 | 249494 | 247167 |
| 小売店 O 社 I 店 | 928134 | 207058 |
| 小売店 D 社 N 店 | 4907099 | 461477 |

表 1 からも分かるように解く問題を各日に分割しても変数と制約式の個数は数十万以上あり，実用的な時間

で問題を解くことができないため，さらに解き方を工夫する必要がある．今回の問題では特に変数の数が多いため変数の一部を固定して問題を解く．提案するアルゴリズムでは各スタッフに対して，各時刻における作業の割り当てを行う問題を解き，そこで割り当てられた作業を固定し，次に勤務シフトの割り当てを行う問題を解く．

勤務シフトを固定して作業の割り当てを行う問題を作業決定問題，作業を固定して勤務シフトの割り当てを行う問題を勤務シフト決定問題と呼ぶことにする．最初の作業割当問題の段階で希望勤務時間を勤務シフトとして固定しているが，連続勤務時間の上限を超える時間を希望勤務時間として提出しているスタッフがいるような場合には，生成された勤務シフトを作業割当問題の勤務シフトとしてもう一度作業割当問題と勤務シフト割当問題を解くことで余剰となっている時刻のスタッフを削減できる可能性がある．そこで，提案手法では余剰スタッフの削減が終了するまで反復させる．まとめると提案手法1のアルゴリズムは以下のようになる．

提案手法1のアルゴリズム

step1　各スタッフの希望勤務時間を初期の勤務シフトとして定める．
step2　作業決定問題を解く．
step3　勤務シフト決定問題を解く．
step4　余剰スタッフが減少したら勤務シフトを更新して step2 へ．それ以外の場合は終了する．

提案手法1ではそれぞれを固定して問題を解いているために生成される勤務シフトは最適解ではなく近似解である．そのため作業決定問題における最適な作業の割り当てが多数存在するような場合，割り当てによって余剰スタッフの人数が多くなることが考えられる．

**作業決定問題**

作業決定問題では図2のように各スタッフの勤務シフトを固定して各時刻の各作業の不足人数が最小となるように行う各スタッフに作業を割り当てる．固定する勤務シフトは各スタッフが勤務可能なものである必要があるため，最初は各スタッフの希望勤務時間を勤務シフトとして固定し作業を割り当てる．反復するときはその直前で行った勤務シフト決定問題の解を利用する．

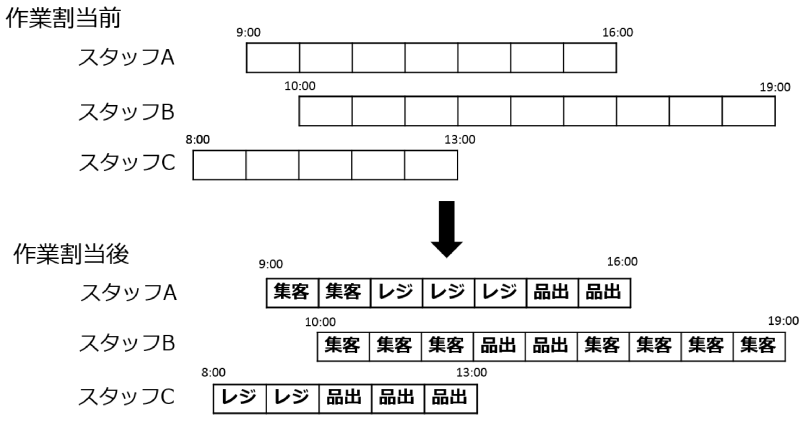


図2　作業決定問題のイメージ

**勤務シフト決定問題**

勤務シフト決定問題では図 3 のように作業割当問題で割り当てられた各スタッフの作業を基に不足人数を最小化するように各スタッフに勤務シフトを割り当てる．

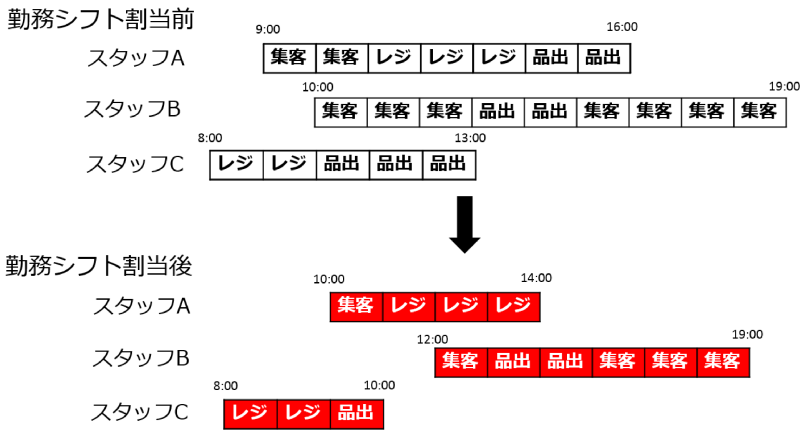


図 3　勤務シフト決定問題のイメージ

### 3.2.2　作業スキルを考慮しないモデル

提案解法 2 は作業スキルを考慮しないモデルに対するものである．この問題を重み付き制約充足問題として定式化すると，変数と制約の数は表 2 のようになる．制約に関しては数千程度になり汎用ソルバー SCOP[15] で解くことが可能になった．

表 2　変数，制約の個数

| | 変数の個数 | 制約の個数 |
|---|---|---|
| 小売店 S 社 C 店 | 225310 | 5046 |
| 小売店 O 社 I 店 | 127332 | 2467 |
| 小売店 D 社 N 店 | 504897 | 4023 |

## 4　勤務シフト表の作成

### 4.1　実務に基づいたモデル化

本節で扱う問題は各スタッフがいつ，どの時間帯で働くかを決定するというものである．このスケジュールを表したものを勤務シフト表を呼び，図 4 は勤務シフト表のイメージである．扱う問題のモデル化はスケジューリング専門とするコンサルティング会社である T 社と共同で行った．

#### 4.1.1　考慮する条件

次に勤務シフト表で考慮する制約条件を述べる．汎用的なモデルを構築するために複数の店舗のヒアリングに基づき考慮する条件を決定した．

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 月 | 火 | 水 | 木 | 金 | 土 | 日 | 月 | 火 | 水 | 木 | 金 | 土 | 日 | 月 | 火 | 水 | 木 | 金 | 土 | 日 | 月 | 火 | 水 | 木 | 金 | 土 | 日 | 月 | 火 | 水 |
| staff1 | 休 | C1 | C1 | 休 | C1 | C1 | 休 | 休 | C1 | C1 | C1 | 休 | 休 | 休 | 休 | 休 | C1 | C1 | C1 | 休 | C1 | C1 | 休 | C1 | C1 | 休 | C1 | C1 | C1 | 休 | 休 |
| staff2 | E1 | C1 | E1 | 休 | E1 | E1 | 休 | C1 | C1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | 休 | C1 | E1 | E1 | 休 | C1 | E1 | 休 | E1 | E1 | E1 | 休 | C1 |
| staff3 | C2 | E1 | C2 | 休 | C2 | C2 | 休 | 休 | C1 | C2 | 休 | C2 | C2 | 休 | 休 | C1 | C2 | 休 | C2 | C2 | 休 | 休 | 休 | 休 | E1 | E1 | 休 | C2 | C2 | 休 | C2 |
| staff4 | 休 | C1 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | C1 | C2 | 休 | C2 | 休 | 休 | C2 | C1 | C2 | 休 | C2 | C2 | 休 | C2 | C1 | C2 |
| staff5 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | C2 | C1 | 休 | C2 | C2 | 休 | E1 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | C1 | C2 | 休 | C2 | C2 | 休 | 休 | C1 | C2 |
| staff6 | 休 | C1 | C2 | 休 | C2 | C2 | 休 | C2 | E1 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | C2 | 休 | C2 | C2 | 休 | C2 | C1 | C2 | C2 | 休 | E1 | E1 | 休 | E1 | C2 |
| staff7 | 休 | C1 | C2 | 休 | C2 | C2 | 休 | C2 | C1 | 休 | 休 | 休 | 休 | 休 | C2 | C1 | 休 | C2 | C2 | 休 | 休 | C2 | E1 | C2 | 休 | C2 | C2 | 休 | C2 | C1 | 休 |
| staff8 | 休 | E1 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | E1 | C2 | C2 | 休 | C2 | C2 | 休 | E1 | C2 | 休 | C2 | C2 | 休 | C2 | E1 | 休 |
| staff9 | C2 | C1 | 休 | C2 | C2 | 休 | C2 | C2 | C1 | 休 | C2 | C2 | 休 | 休 | C2 | C1 | 休 | C2 | C2 | 休 | C2 | C2 | C1 | 休 | C2 | C2 | C2 | 休 | C2 | C1 | 休 |
| staff10 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | C1 | C2 | C2 | 休 | C2 | C2 | C2 | C1 | 休 | C2 | E1 | C2 | 休 | C2 | C1 | 休 | C2 | C2 | 休 | E1 | C2 | 休 | E1 |
| staff11 | C2 | C1 | 休 | E1 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | C2 | C2 | 休 | 休 | E1 | C2 | 休 | C2 |
| staff12 | C2 | 休 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | B4 | C2 | 休 | C2 | C2 | 休 | 休 | C2 | E1 | 休 | C2 | C2 | 休 | 休 | C2 | C2 | 休 | C2 | C2 | C2 | 休 | C2 |
| staff13 | C2 | 休 | E1 | C2 | C2 | 休 | C2 | E1 | 休 | C2 | E1 | C2 | 休 | E1 | C2 | 休 | 休 | E1 | C2 | 休 | 休 | C2 | C1 | 休 | E1 | C2 | 休 | C2 | 休 | C1 | C2 |
| staff14 | C2 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | C2 | E1 | 休 | C2 | E1 | C2 | E1 | 休 | 休 | 休 | E1 | C2 | 休 | E1 | E1 | 休 | 休 | C2 | C2 | 休 | E1 | E1 |
| staff15 | 休 | C1 | E1 | E1 | 休 | 休 | E1 | E1 | C1 | E1 | 休 | C2 | C2 | 休 | E1 | C1 | 休 | E1 | C2 | 休 | C2 | C2 | 休 | E1 | C2 | 休 | E1 | C2 | 休 | C1 | C2 |
| staff16 | E1 | E1 | 休 | E1 | C2 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | C1 | 休 | E1 | E1 | 休 | C2 | E1 | C1 | 休 | E1 | E1 | 休 | E1 | E1 | C1 | 休 |
| staff17 | E1 | E1 | 休 | E1 | C2 | 休 | E1 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | C1 | E1 | 休 | E1 | E1 | 休 | E1 | C1 | 休 | C2 | E1 | 休 | 休 | E1 | C1 | 休 |
| staff18 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | C2 | E1 | 休 | E1 | C1 | E1 | 休 | E1 | 休 | E1 | E1 | 休 | E1 |
| staff19 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | B4 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | C2 | C2 | 休 | C1 | 休 |
| staff20 | E1 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | E1 |
| staff21 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 |
| staff22 | E1 | 休 | 休 | 休 | E1 | 休 | 休 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 |
| staff23 | 休 | C1 | C1 | C1 | 休 | C1 | C1 | C1 | C1 | C1 | 休 | 休 | C1 | 休 | 休 | C1 | C1 | 休 | 休 | C1 | 休 | 休 | C1 | C1 | 休 | 休 | C1 | C1 | 休 | C1 | 休 |
| staff24 | E1 | 休 | E1 | C2 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | C2 | E1 | 休 | E1 | E1 | E1 |
| staff25 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | C2 | 休 | E1 | CD | 休 | E1 | E1 | E1 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 |
| staff26 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | C2 | E1 | 休 | E1 |
| staff27 | E1 | 休 | E1 | CD | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | 休 | E1 | E1 |
| staff28 | E1 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | B4 | 休 | C2 | E1 | E1 | 休 | C2 | E1 | 休 | E1 | E1 | E1 | 休 | E1 | E1 | 休 | E1 | E1 | E1 | 休 |

図 4　勤務シフト表のイメージ

1. 希望勤務日，希望勤務時間を満たす
2. 希望休暇，絶対休暇，絶対出勤を満たす．
3. 休日数の上下限を満たす．
4. 固定シフトを満たす
5. 連続勤務日数の上下限を満たす．
6. 連続休暇日数の上下限を満たす．
7. 各日，各勤務シフトに対する必要人数を満たす．
8. 望ましくない勤務シフトの並びの禁止する．
9. ペアリングを考慮する．
10. 月間勤務時間の上下限を満たす．
11. 各勤務シフトの勤務回数の上限を満たす．
12. 各日，各勤務シフト，各スキルの必要人数を満たす．
13. 飛び休を禁止する．
14. 人件費の上限を満たす．

以上の制約を満たした上で，各スタッフがいつ，どの時間帯で働くかを決定するという問題を扱う．以下では本節で扱う問題を勤務シフト割当て問題と呼ぶことにする．勤務シフト割当て問題は各制約に対する違反度の総和を最小化するという重み付き制約充足問題として定式化することができる．本研究ではこの問題に対し，勤務シフト生成問題によって生成された勤務シフトの種類と各勤務シフトの必要数を用いることを考える．既存手法を直接適用すると，勤務シフトの種類が多いため，解の精度や計算時間が悪化する．そこで，勤務シフト生成問題の解を利用することや，局所探索に探索順序を導入することで解の精度や計算時間の改善を図る．

## 4.2 初期解の構築

本研究では勤務シフト生成問題の解を初期解として利用することで勤務シフト表作成問題での実行可能解を得るための時間を削減することを提案する．勤務シフト割当て問題を解く上で必要な情報として，各日における勤務シフトの種類と数があるが，勤務シフト生成問題での各提案手法では各日における生成された勤務シフトの最適な割り当てを行っているため，そこで割り当てた解を初期解として利用する．この初期解は希望勤務日，希望勤務時間を満たしており，かつ不足人数も考慮されているので質は高い．したがって，既存手法をそのまま適用することに比べ，より良いスケジュールを得られることが期待される．

## 4.3 探索順序の導入

勤務シフト生成問題では生成される勤務シフトの種類は数十から数百になり，既存手法の局所探索にそのまま適用すると，近傍の数が非常に多くなるため探索に多くの時間を要してしまう．そのため，局所探索の効率化が必要になる．研究によって勤務シフト割当て問題に対する近傍は異なるが，多くのものは以下の 2 つの近傍に類するものである．

近傍 1　ある日のある 2 人のスタッフの勤務シフトを交換する．
近傍 2　ある日のあるスタッフの勤務シフトを別の勤務シフトに交換する．

近傍 1 の数はスタッフ数やスケジュール期間に依存するものであり，勤務シフトの種類による影響はない．一方，近傍 2 の数は勤務シフトの種類数によって変化する．そこで今回は近傍 2 に対する探索の効率化を考えていく．また，探索の効率化の方向性は，

(a)　解のペナルティ値の計算を効率化する．
(b)　改善の可能性のない解の探索を省略する．

の 2 つがある．(a) の解のペナルティ値の計算に関しては，近傍 1 と同様に勤務シフトの種類による影響はなく，制約条件の種類や違反箇所の探索の実装によるものである．一方，(b) は勤務シフトの種類により探索領域が増大するため，効率化を考える必要がある．局所探索では探索の前後でペナルティ値があまり変化しないように探索を設計するのがよいと言われている [14]．また，各スタッフは希望勤務時間以内の勤務シフトでしか勤務できないことを考慮すると，暫定の勤務シフトにより近い勤務シフトから交換を行うという，探索に順序を設定することで改善の可能性の高い解を優先的に探索することができ，効率化を図ることができる．そこで，勤務シフトの近さには新たに勤務シフトの類似度という指標を設定する．

集合 $T$ の濃度を $|T|$ としたとき，勤務シフト $A$ の勤務シフト $B$ に対する類似度 $\tau(A,B)$ は勤務シフト $A$ の勤務時刻の集合を $T_A$，勤務シフト $B$ の勤務時刻の集合を $T_B$ として以下のように定義する．

$$\tau(A,B) = |T_A \cup T_B| - |T_A - T_B|$$

例を用いて勤務シフトの類似度の計算を説明する．時刻 8 から時刻 12 まで出勤する勤務シフト $A$，時刻 11 から時刻 13 まで出勤する勤務シフト $B$ とすると，$T_A = \{8,9,10,11,12\}$，$T_B = \{11,12,13\}$，$T_A \cup T_B = \{11,12\}$，$T_A - T_B = \{8,9,10\}$，$T_B - T_A = \{13\}$ より，勤務シフト $A$ の勤務シフト $B$ に対する

類似度 $\tau(A,B)$ 及び勤務シフト $B$ の勤務シフト $A$ に対する類似度 $\tau(B,A)$ は，

$$\tau(A,B) = 3 - 1 = 2$$
$$\tau(B,A) = 3 - 2 = 1$$

となる．この指標を用いて各勤務シフトに対する交換の順序付けを行い，近傍 2 の探索を行う．

# 5　作業分担表の作成

## 5.1　実務に基づいたモデル化

本節で扱う問題は小売店等において，各スタッフがいつ，どの作業を行うか，もしくは休憩をとるかといった，1 日の作業スケジュールを決定するというものである．この作業スケジュールを表したものを作業分担表と呼ぶ．図 5 は作業分担表のイメージを表している．なお, 問題のモデルかはスケジューリングを専門とするコンサルティング会社の T 社と共同で行った．

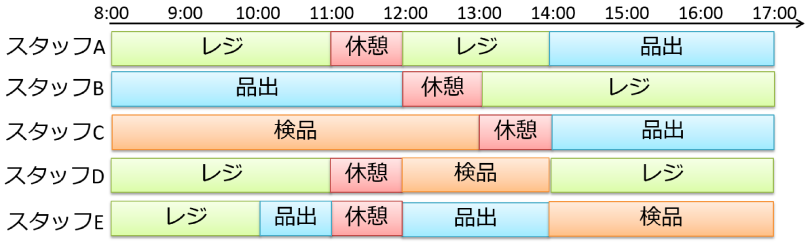


図 5　作業分担表のイメージ

### 5.1.1　考慮する条件

次に作業分担表で考慮する制約条件を述べる．汎用的なモデルの構築を行うために複数の店舗のヒアリングに基づき考慮する条件を決定した．制約条件はハード制約とソフト制約に分類できる．ハード制約とは必ず満たしたい制約条件のことであり，ソフト制約はなるべく満たしたい制約条件のことである．

- ハード制約
  1. 各スタッフは各シフトの時間範囲しか働けない．
  2. 各シフトに対する休憩の時間の長さ，回数を満たす．
- ソフト制約
  1. 各時刻で必要な作業人数を満たす．
  2. 作業ができる人にその作業を割当てる．
  3. 連続勤務時間の上限を満たす．
  4. 連続禁止パターンを考慮する．
  5. 作業の連続性を考慮する．
  6. ペアリングを考慮する．
  7. 各時刻でグループスキル保持者を満たす．

8. 各時刻，各作業でグループスキル保持者を満たす．
9. 各時刻，各作業の必要スコアを満たす．
10. 各作業に対して作業人数の上限を満たす．

以上の制約を満たした上で，誰がどの時刻で作業を行うのか，もしくは休憩をとるのかという 1 日の作業スケジュールを決定する問題を扱う．以下では本節で扱う問題を作業割当て問題と呼ぶことにする．

## 5.2 作業割当て問題の特徴

作業割当て問題には他のスケジューリング問題には無い制約が存在する．それは作業の連続性である．例えば，図 6 と図 7 を見ると，全体では同じ作業量であるが，図 7 では作業が連続していないため実際の業務では移動時間や準備等が余分に必要になるため効率が悪い．したがって，図 6 の方が効率的で望ましい作業分担表となる．この連続性という概念は勤務シフトスケジューリングには無い作業割当て問題固有の特徴と言える．

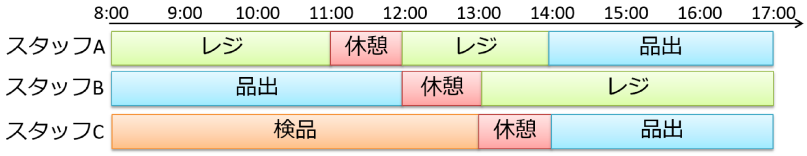


図 6　連続性が考慮されている例


図 7　連続性が考慮されていない例

また，Valouxis ら [13] は 1 ヵ月のシフトを 1 週間に分割するといった問題の構造を利用していたが，作業割当て問題は連続性という特徴が存在するため分割して解を求めるアルゴリズムを適用することは難しい．そこで提案手法では Constantino ら [4] と同様，各スタッフが各作業を行った時のコストを前の作業を考慮して計算し，コストの総和が最小となるように二部グラフのマッチングを用いて解を構成する．この手法は元々看護師スケジューリング問題に対する解法として提案されたものだが，前の作業を考慮するという点が作業割当て問題との相性が非常に良い．Constantino らは看護師スケジューリング問題に対し初期解だけでなく解の改善も二部グラフのマッチングを用いた手法を提案しているが，ペアリングなどの制約条件を考慮していない．そこで提案手法では局所探索によって解の改善を行い，様々な制約条件に対応できるようにしている．

## 5.3 提案手法

作業割当て問題は高速に解を求めることが必要になるため近似解法を提案する．提案手法ではまず二部グラフのマッチングを用いて初期解を構成している．最も早い時刻から順に各スタッフが各作業を行ったときのコ

ストを計算し，コストの総和が最小になるように作業を割当てる．最も早い時刻から順に割当てていくことにより，前の時刻の作業を参照する事が可能になるため連続性を考慮した初期解を構成する事ができる．

しかし，この方法では複数のスタッフに関連する制約条件には対応できない．そこで，他の制約条件を満たすために以下の 4 つの近傍を用いた局所探索により解の改善を行っている．

**近傍 1**　あるスタッフ 2 人の作業を交換する．
**近傍 2**　あるスタッフの作業を他の作業や休憩に変更する．
**近傍 3**　休憩回数が足りない場合に休憩を追加する．
**近傍 4**　休憩開始時刻を他の時刻に変更する．

近傍 1 と近傍 2 は初期解で考慮できなかったペアリングなどの制約に対応するためのものである．また，近傍 3 と近傍 4 は休憩に関する制約を満たすために用いている．本研究では，これらの近傍を用いて局所探索を行い解を求める方法を提案する．

## 5.4　初期解の構成

初期解は最も早い時刻 $t = 1$ から順に各スタッフが各作業を行ったときのコストを計算し，コストの総和が最小になるように作業を割当てていく．この方法により前の時刻の作業を参照する事が可能になるため連続性を考慮した初期解を構成する事ができる．

初期解を構成するために，ある時刻 $t$ においてスタッフ $i$ が作業 $j$ を行った時のコストをそれぞれ計算する．その後，その時刻における必要作業に応じてコスト行列 $C^t$ を作成する．$C^t$ は $n \times n$ の正方行列である．$C^t$ の $(i, j)$ 成分は時刻 $t$ に必要な作業 $j$ をスタッフ $i$ が行った時のコストを表している．もし総必要人数が時刻 $t$ で働けるスタッフの総人数よりも少ない場合は人数の余分が生じる．そこでフリー作業というものを考える．フリー作業とは全ての作業と休憩の中で最もコストが低くなる作業の事を指す．図 8 にコスト行列のイメージを表す．例えば，時刻 $t_1$ において作業できるスタッフの人数を $n = 4$ とし，$t_1$ に必要な作業人数とし

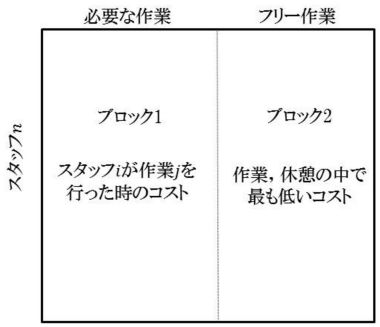


図 8　コスト行列 $C$

て作業 A を 1 人，作業 B を 2 人，作業 C を 1 人とする．その場合，(17) 式のような $4 \times 4$ のコスト行列 $C^{t_1}$

ができる．

$$\begin{array}{c} \\ \text{スタッフ}1 \\ \text{スタッフ}2 \\ \text{スタッフ}3 \\ \text{スタッフ}4 \end{array} \begin{array}{c} \begin{array}{cccc} \text{A} & \text{A} & \text{B} & \text{C} \end{array} \\ \begin{pmatrix} 100 & 100 & 500 & 500 \\ 500 & 500 & 500 & 100 \\ 200 & 200 & 100 & 200 \\ 500 & 500 & 100 & 100 \end{pmatrix} \end{array} \tag{17}$$

また，時刻 $t_2$ において作業できるスタッフの人数を $n=5$ とし，$t_1$ に必要な作業人数は先ほどの例と同じ作業 A を 1 人，作業 B を 2 人，作業 C を 1 人とする．今回はスタッフが 1 人余分にいることになるので，正方行列にするために Free 作業の列を 1 列追加した (18) 式のようなコスト行列 $C^{t_2}$ を作成する．

$$\begin{array}{c} \\ \text{スタッフ}1 \\ \text{スタッフ}2 \\ \text{スタッフ}3 \\ \text{スタッフ}4 \\ \text{スタッフ}5 \end{array} \begin{array}{c} \begin{array}{ccccc} \text{A} & \text{A} & \text{B} & \text{C} & \text{Free} \end{array} \\ \begin{pmatrix} 100 & 100 & 500 & 500 & 100 \\ 500 & 500 & 500 & 100 & 20 \\ 200 & 200 & 100 & 200 & 100 \\ 500 & 500 & 100 & 100 & 20 \\ 500 & 500 & 100 & 100 & 100 \end{pmatrix} \end{array} \tag{18}$$

以上のように作成したコスト行列 $C^t$ をに対してハンガリアン法を用いて割当て問題を解く事により，時刻 $t$ において各スタッフに各作業を割当てた時のコストの総和が最小である解が得られることになる．この方法で初期解を求めると，作業の連続性を考慮した初期解が得られるため，作業割当て問題と非常に相性が良い．また，作業可能な人数と必要人数が一致しているときは必要人数を満たし，作業可能な人数が必要人数よりも多い場合はフリーシフトにより最もコストの低い作業か休憩が割当てられるため，初期解の時点では必要人数を必ず満たすという特徴がある．

### 5.5 局所探索

初期解の構成時には，あるスタッフがある作業を行った場合に関するペナルティしか考慮できなかった．これでは複数のスタッフに関係するペナルティが考慮できない．例えば，ペアリングや作業のスコアの制約などが該当する．そこで，ある 2 人のスタッフを選択して作業を交換する近傍や，あるスタッフの作業を他の作業や休憩に変更する近傍を用いて局所探索を行い，初期解を構成する際には考慮できなかった制約を考慮して解を改善する．

また，休憩の回数が足りない場合に休憩を追加する近傍や，休憩開始時刻を移動する近傍も用いている．これらの近傍は休憩の回数や最低労働時間に関する制約違反を改善するためのものである．

## 6 数値実験

本節では提案手法の評価を行うために数値実験を行い，結果について考察する．混合整数計画問題には整数計画ソルバーの SCIP3.1.0 を，制約充足問題には制約充足ソルバー SCOP[15] を用いた．勤務シフト生成と勤務シフト表作成に関しては CPU: Intel Corei7-3632QM 2.66GHz, RAM: 8GB, OS: Windows8.1 64bit, 作業分担表に関しては CPU: Intel Corei7-3770QM 3.40GHz, RAM: 8GB, OS: Windows7 64bit の計算機を使用した．

## 6.1 使用するデータ

数値実験で使用するデータについて説明する．勤務シフト生成と勤務シフト表作成では実データ3種類，それらを基に作成した人工データ5種類の計8種類のデータを用いた．作業分担表作成では実データ5種類，それらを基に作成した人工データ2種類の計7種類のデータを用いた．

| 店舗 | 正社員数 | 非正社員数 | 作業数 | 営業時間 | 固定シフト | 期間 |
|---|---|---|---|---|---|---|
| 小売店S社C店 | 18 | 0 | 1 | 17 | 0 | 31 |
| 小売店O社I店 | 25 | 0 | 6 | 15 | 0 | 16 |
| 小売店D社N店 | 33 | 0 | 11 | 24 | 0 | 15 |
| データ1 | 20 | 4 | 5 | 10 | 2 | 14 |
| データ2 | 20 | 4 | 10 | 10 | 2 | 14 |
| データ3 | 20 | 4 | 15 | 10 | 2 | 14 |
| データ4 | 20 | 4 | 5 | 20 | 2 | 14 |
| データ5 | 40 | 4 | 5 | 10 | 2 | 14 |

まず勤務シフト生成と勤務シフト表作成で用いたデータについて説明する．実データに関しては小売店O社I店は一般的な店舗，小売店S社C店は作業数が少ない店舗，小売店D社N店は営業時間が長い店舗となっている．人工データに関しては提案手法の傾向を知るために，人工データ1の各項目の値を基に人工データ2と人工データ3は作業数を増加させたデータであり，人工データ4とはスタッフ数を増加させたもの，人工データ5は営業時間数を増加させたデータである．

| | スタッフ数 | 時刻 (15分単位) | シフト | 作業種類 | 休憩種類 | 禁止パターン | 連続勤務時間 | グループ | ペアリング |
|---|---|---|---|---|---|---|---|---|---|
| 小売店D社N店 | 21 | 96 | 92 | 11 | 4 | 0 | 16 | 0 | 0 |
| 小売店D社O店 | 10 | 80 | 80 | 3 | 4 | 0 | 16 | 0 | 0 |
| 百貨店M社A店 | 8 | 44 | 40 | 5 | 4 | 0 | 16 | 0 | 0 |
| 小売店M社N店 | 24 | 44 | 41 | 22 | 4 | 0 | 16 | 0 | 0 |
| 小売店T社 | 18 | 40 | 5 | 7 | 2 | 1 | 16 | 3 | 1 |
| 人工データ1 | 30 | 52 | 7 | 10 | 3 | 3 | 12 | 5 | 2 |
| 人工データ2 | 46 | 44 | 41 | 44 | 4 | 0 | 16 | 0 | 0 |

次に作業分担表作成で用いたデータについて説明する．小売店D社N店, 小売店D社O店, 百貨店M社A店, 小売店M社N店, 人工データ2は禁止パターンやグループ，ペアリングの制約が存在しないデータとなっている．それに対し人工データ1,2は禁止パターンやペアリング，スコア，グループなどの条件があり複雑なデータとなっている．なお，人工データ1は休憩回数を3回にするなど実データにはなかったデータの設定を行った人工データであり，人工データ2は小売店M社N店のデータを基に必要人数を増加させた場合のデータである．

## 6.2 結果・考察

### 6.2.1 勤務シフトの生成

生成した勤務シフトの評価は不足人数と休日数の違反，そして計算時間により行う．不足人数とは生成された勤務シフトを用いて，最適なスタッフの割り当てを行った際に生じる各時刻における各作業に対する必要人数との差である．不足人数の増加は業務の円滑な運営に影響を及ぼすので少ない方が好ましく，最も重要な指標である．ここではある作業に 1 人のスタッフが 1 時間不足している場合を 1 として，その延べ時間を総不足時間として算出した．次に休日数の違反であるが，これはそれぞれの提案手法で生成された勤務シフトを勤務シフト割当問題に適用して求めたスケジュールにおける各スタッフの休日数の上下限制約の違反人数である．勤務シフト割当問題には出勤休暇を決定してから勤務シフトを割り当てるという手法 [13] を用いた．これらの指標を用いることで考慮しなかった条件が生成される勤務シフトに与える影響を見ることができる．

各データに対して，3 節で述べた休日数を考慮しないモデルに対する手法である提案手法 1 と作業スキルを考慮しないモデルに対する手法である提案手法 2 により生成した勤務シフトの総不足時間と休日数の違反及び計算時間を表 3 に示す．提案手法 1 が 1 日ごとに勤務シフトを生成するのに対し，提案手法 2 では期間分の勤務シフトをまとめて生成して求めている．そこで比較のために総不足時間と計算時間に関しては両手法ともに期間分の勤務シフトを生成したものを期間で割り，1 日あたりの各値に関して比較を行った．

表 3　勤務シフト生成問題に対する提案手法の結果

| | 総不足時間 (h) | | 休日数違反 | | 計算時間 (s) | |
|---|---|---|---|---|---|---|
| 店舗 | 提案解法 1 | 提案解法 2 | 提案解法 1 | 提案解法 2 | 提案解法 1 | 提案解法 2 |
| 小売店 S 社 C 店 | 0.00 | 0.00 | 0 | 0 | 17.44 | 5.28 |
| 小売店 O 社 I 店 | 2.84 | 7.25 | 0 | 0 | 6.16 | 6.86 |
| 小売店 D 社 N 店 | 23.53 | 25.93 | 12 | 11 | 19.3 | 7.80 |
| 人工データ 1 | 0.73 | 1.30 | 0 | 0 | 1.63 | 8.09 |
| 人工データ 2 | 0.94 | 3.12 | 0 | 0 | 1.58 | 8.62 |
| 人工データ 3 | 1.50 | 5.41 | 0 | 0 | 1.24 | 7.69 |
| 人工データ 4 | 0.51 | 4.33 | 0 | 0 | 5.38 | 80.56 |
| 人工データ 5 | 6.58 | 8.05 | 15 | 11 | 4.34 | 17.76 |

表 3 を見ると総不足時間に関しては全てのデータに対して提案手法 1 の値が提案手法 2 の値以下となっており，逆に休日数に関しては提案手法 2 の値が提案手法 1 の値以下となっており考慮した条件が反映されていることがわかる．

総不足時間に関しては小売店 S 社 C 店のデータを除いて全てのデータにおいて提案手法 1 が低い値となっており，作業スキルの考慮は非常に有効であることがわかる．小売店 S 社 C 店は作業数が 1 つの特殊な店舗であり，作業が 1 つであると実質的には作業スキルを考慮する必要がないため提案手法 2 と同じ結果になったと考えられる．

休日数に関しては小売店 D 社 N 店と人工データ 5 の 2 つのデータで差がみられ提案手法 2 が低い値となっている．これら 2 つのデータはスタッフ数の大きいデータであり，このような規模の大きなデータに対しては有効である．しかし，ほとんどのデータでは両手法とも休日数の違反は 0 であり，現実の一般的な店舗のデー

タでは休日数の違反が起こることは少ないと考えられる．

最後に計算時間であるが，小売店 S 社 C 店と小売店 D 社 N 店では提案手法 1 が提案手法 2 に比べて時間が掛かっている．先ほど述べたように小売店 S 社 C 店は作業が 1 つであり，作業スキルを考慮する必要がないため提案手法 2 の方が計算時間が短くなっていると考えられる．小売店 D 社 N 店は営業時間が長く作業数の多い店舗である．営業時間の長さは両手法に影響するが，作業数の多さは提案手法 1 のみに影響し，そのため変数と制約式が非常に多くなってしまう．よって提案手法 2 の方が計算時間が短くなっていると考えられる．しかし，作業数の多い人工データ 3 では逆に提案手法 1 が提案手法 2 より高速に解けている．よって計算時間は営業時間や作業数といった値の影響だけでなくデータの構造にも一定の影響を受けることが考えられる．

### 6.2.2 勤務シフト表の作成

4 節で述べた勤務シフト生成問題の解を勤務シフト割当問題の初期解として用いたときの評価を行った．その結果を表 4 に示した．比較手法としては Valouxis[13] を基にした以下のような手法を用いる．

step1 1 週間ごとの出勤休暇の割り当てを整数計画問題を解くことで決定する．
step2 step1 の結果を 1 か月分の出勤休暇の初期点として，近傍探索を行う．
step3 step2 で作成した出勤休暇割り当ての出勤日に対してランダムに勤務シフトに割り当てる．
step4 勤務シフトを入れ替える局所探索を行う．

表 4 勤務シフト生成問題の解を初期解とした結果

| | ペナルティの総和 | | 計算時間 (s) | |
|---|---|---|---|---|
| 店舗 | 提案解法 | 比較解法 | 提案解法 | 比較解法 |
| 小売店 S 社 C 店 | 2000 | 8400 | 54.36 | 93.16 |
| 小売店 O 社 I 店 | 2805 | 18805 | 0.00 | 74.5 |
| 小売店 D 社 N 店 | 35629 | 51229 | 22.89 | 93.86 |
| データ 1 | 6017 | 11217 | 9.21 | 7.07 |
| データ 2 | 5230 | 10030 | 11.7 | 12.12 |
| データ 3 | 6420 | 10226 | 0.93 | 13.67 |
| データ 4 | 6801 | 11801 | 4.15 | 15.96 |
| データ 5 | 31600 | 58000 | 0.02 | 75.23 |

表 4 を見てわかるように，実験を行った全ての店舗において勤務シフト生成問題の解を初期解に利用した方が最適解におけるペナルティ値，計算時間ともに既存手法を直接適用するより良い結果となった．

### 6.2.3 作業分担表の作成

各データに対し提案手法と SCOP を用いた比較手法を適用した時のペナルティの総和と計算時間を表 5 に示す．比較手法では，次のように休憩箇所を決定する問題と，作業を割り当てる問題の 2 段階に分割して重み付き制約充足問題として定式化し，汎用ソルバーである SCOP で解を求めるという手法を用いる．

step1 各勤務シフトに対し休憩の取り方を列挙した休憩パターンを作成した後，整数計画問題として定式化し，各時刻に対して必要人数を満たすようなパターンを求める．

step2　各スタッフに対して作業を行う時刻に作業を割り当てる．

SCOP は解を求める際にオブジェクトと呼ばれる特殊なファイルを生成する時間が必要になるため，SCOP の計算時間はそのオブジェクトの生成時間と最良解が発見されるまでの計算時間の和を表示している．なお，表中のパーセンテージは提案解法で求めた解の総ペナルティを SCOP で求めた解の総ペナルティで割った時の割合を表している．また，各データに対し二つの手法を比較し，優れている方の数値をボールド体で表示している．

表 5　作業割当て問題における提案手法と比較手法の結果

| | | ペナルティの総和 | (%) | 計算時間 | (%) |
|---|---|---|---|---|---|
| 小売店 D 社 N 店 | 提案解法 | 1460 | 144.6 | **6.5** | **33.03** |
| | SCOP | **1010** | | 19.68 | |
| 小売店 D 社 O 店 | 提案解法 | 13820 | 100.1 | **0.188** | **26.11** |
| | SCOP | **13800** | | 0.72 | |
| 小売店 M 社 A 店 | 提案解法 | 22620 | 101.8 | **0.109** | **6.49** |
| | SCOP | **22230** | | 1.68 | |
| 小売店 M 社 N 店 | 提案解法 | **6730** | **99.9** | **7.406** | **28.46** |
| | SCOP | 6740 | | 26.02 | |
| 小売店 T 社 | 提案解法 | **3750** | **98.4** | **1.485** | **7.77** |
| | SCOP | 3810 | | 19.1 | |
| 人工データ 1 | 提案解法 | **293608** | **64.9** | **8.5** | **76.65** |
| | SCOP | 452737 | | 11.09 | |
| 人工データ 2 | 提案解法 | **15760** | **53.5** | **76.86** | **44.99** |
| | SCOP | 29470 | | 170.84 | |

表 5 を見ると，小売店 D 社 N 店，小売店 D 社 O 店，小売店 M 社 A 店に対しては，提案解法よりも SCOP で求めた解の方が若干ペナルティの総和が低く良い結果となっている．しかし，その他のデータに対しては提案解法の方が SCOP で求めた解よりもペナルティの総和が低いことがわかる．特に，人工データ 2 に対しては SCOP の約半分のペナルティの値となっている．このことから人数が多く大規模なデータに対して提案手法は有効な手法であると言える．反対に SCOP は変数が多くなると精度が悪くなる傾向があると推察される．

また，計算時間に関しては全ての実験において提案解法の方が SCOP よりも高速に解を求められていることがわかる．特に差が最も大きいのは人工データ 2 となっている．人工データ 2 は人数が多く大規模なデータとなっているが，大規模なデータに対しても提案解法は有効であると言える．

## 7　おわりに

本稿では，スタッフスケジューリングを勤務シフトの生成，勤務シフト表の作成，作業分担表の作成の 3 段階に分割することを提案した．そして，各段階に対し実務に基づくモデルを構築し，近似解法を提案した．また，実データを用いた数値実験により，それらの有効性を確認した．

現在は問題を分割しているため，各段階で精度が良くても全体として実用的なスケジュールが作成できるとは限らない．そこで，全体として精度の高いスケジュールを作成することが今後の課題として挙げられる．

## 参考文献


[1] P. Brucker, E.K. Burke, T. Curtois, R. Qu and G.V. Berghe, "A shift sequence based approach for nurse scheduling and a new benchmark dataset", *Journal of Heuristics*, Vol.16, No.4, pp.559-573, 2010.

[2] E.K. Burke, P.D. Causmaecker and G.V. Berghe, "A hybrid tabu search algorithm for the nurse rostering problem", *Simulated Evolution and Learning*, Vol.1585, pp.187-194, 1999.

[3] E.K. Burke, P.D. Causmaecker, G.V. Berghe and H.V. Landeghem, "The state of the art of nurse rostering", *Journal of Scheduling*, Vol.7, No.6, pp.441-499, 2004.

[4] A.A. Constantino, D. Landa-Silva, E.L. Melo, C.F.X. Mendonca, D.B. Rizzato and W. Romão, "A heuristic algorithm based on multi-assignment procedure for nurse scheduling", *Annals of Operations Research*, Vol.218, Issue.1, pp.1-19, 2013.

[5] K.A. Dowsland, "Nurse scheduling with tabu search and strategic oscillation", *European Journal of Operational Research*, Vol.106, No.2, pp.393-407, 1998.

[6] A.T. Ernst, H. Jiang, M. Krishnamoorthy and D. Sier, "Staff scheduling and rostering:A review of applications, methods and models", *European Journal of Operational Research*, Vol.153, No.1, pp.3-27, 2004.

[7] M. Hadwan, "A constructive shift patterns approach with simulated annealing for nurse rostering problem", In:*Information Technology(ITSim), 2010 Internatiolal Synposium*, Vol.1, pp.1-6, 2010.

[8] 池上敦子, 繁野麻衣子, "質の高いサービスを提供するためのスタッフスケジューリング", 電子情報通信学会誌, Vol.94, No.9, pp.760-766, 2011.

[9] 池上敦子, 丹羽明, 大倉元宏, "我が国におけるナース・スケジューリング問題", オペレーションズ・リサーチ, Vol.41, No.8, pp.436-442, 1996.

[10] 池上敦子, "ナース・スケジューリング", 統計数理, Vol.53, No.2, pp.231-259, 2005.

[11] J. Majumdar and A.K. Bhunia, "Elitist genetic algorithm for assignment problem with imprecise goal", *European Journal of Operational Research*, Vol.177, No.2, pp.684-692, 2007.

[12] L.H. Tein and R. Ramli, "Recent advancements of nurse scheduling models and a potential path", In:*Proceeding of the 6th IMT-GT Conference on Mathematics, Statistics and its Applications (ICMSA 2010)*, pp.395-409, 2010.

[13] C. Valouxis, C. Gogos, G. Goulas, P. Alefragis, and E. Housos, "A systematic two phase approach for the nurse rostering problem", *European Journal of Operational Research*, Vol.219, pp.426-433, 2012.

[14] 柳浦睦憲, 茨木俊秀, "組合せ最適化—メタ戦略を中心として—", 朝倉書店, 東京, 2001.

[15] LOG OPT, http://www.logopt.com/scop.htm.