

# 最適化手法によるコンビニエンスストアの 勤務シフトスケジューリング

2016SS065 渋谷博輝

指導教員：小市俊悟

## 1 はじめに

本研究は、あるコンビニエンスストアで働くアルバイト等の勤務シフト作成の効率化を目的とした研究である。現在、勤務シフトの作成は一人の担当者が手作業で行っているが、雇用者側と従業員側の両者の要望を同時に満たすことは容易ではなく、多くの時間を費やしている。さらに手作業であるが故にミスも頻発している。勤務シフトの作成を最適化手法を用いて自動化することで、時間とミスを削減するとともに、両者の要望を満たす最適解を出すことで、両者において満足度の向上を図る。

## 2 勤務シフトについて

以下に、対象とするコンビニエンスストアの勤務体系や勤務シフト作成に際して、考慮すべき事項を説明する。

- 勤務シフトは、早朝、午前、午後、夕方、夜の5つの時間帯でシフトが分かれており、それぞれの就業時間は順に3, 4, 4, 5, 8時間である。本研究では各シフトにおける賃金を順に3100, 3700, 3700, 4600, 8000円とする。給与は各月15日締めで計算される。
- 勤務シフトは各従業員の提出するシフト希望スケジュールとオーナー、マネージャーの予定をもとに作られる。シフト希望スケジュールは勤務可能な日にち、時間帯が提出され、それを満たさなければならない。
- 各従業員には時間帯ごとに経験等に応じて新米とベテランのいずれかの業務レベルが割り当てられる。
- 各シフトで2人の人員が売り場に必要である。その人員にはオーナー、マネージャーの2人が含まれても良い。また、新米2人のみで売り場を回すことはできない。
- 勤務時間が連続7時間越えるとき、1時間の休憩が必要である。すなわち夜勤は各自1時間の休憩が設けられる。この休憩時間のみ、売り場の必要人数は1人となる。
- 夜のシフトは安全面を考慮して、男性1人以上の人員が配置される。
- 睡眠時間の確保のため、夜のシフトに入った場合、その従業員は次の日の夕方のシフトまで勤務できない(夕方のシフトから勤務できる)ものとする。
- 留学生も働いており、法律で留学生の労働時間は週28時間以内と定められている。
- 休憩時間の必要人員の確保、8時間を超えることによ

る割増賃金への人件費増加を避けるため、連続シフトは望ましくない。

- 従業員によっては、入ることは可能であっても、できるだけ避けたい(不慣れな)勤務シフト(時間帯)があるので、それを考慮する。

## 3 整数計画モデル

前節で説明したことを踏まえた勤務シフト作成のための整数計画モデルを以下に説明するが、紙面の都合上、下記はその一部である。給与計算の締め日に合わせ、各月の16日から翌月15日までが勤務シフト作成の対象である。

### 3.1 記号の定義

実行不可能を防ぐために、2人のダミー従業員  $i = 0, 1$  を設ける。また、勤務シフトの作成対象の日にちを連続する5日ごとに区切ったものを五連日と呼ぶ。

**従業員, 日にち, 時間帯に関するパラメータや集合**

$A_1$  : ダミー従業員を除く従業員の集合。

$dlast$  : 勤務シフト作成の対象となる日数。

$D$  : 勤務シフト作成の対象となる日にちの集合。  $D = \{0, 1, 2, \dots, dlast - 1\}$ 。

$D_7$  : 1週間の日の集合。  $D_7 = \{0, 1, 2, 3, 4, 5, 6\}$ 。

$D_5$  : 5日間の日の集合。  $D_5 = \{0, 1, 2, 3, 4\}$ 。

$W$  : 5週分の番号の集合。  $W = \{0, 1, 2, 3, 4\}$

$W_5$  : 7五連日分の番号の集合。  $W_5 = \{0, 1, 2, 3, 4, 5, 6\}$

$T$  : 時間帯の集合。  $T = \{0, 1, 2, 3, 4\}$ 。早朝を0, 午前を1, 午後を2, 夕方を3, 夜を4で表す。

$Ms_t$  : 時間帯  $t$  に働いたときの給料。

$HM_i$  : 従業員  $i$  の希望給料。

### 決定変数

$x_{i,d,t}$  : 勤務シフトの担当を表す変数。

$$x_{i,d,t} = \begin{cases} 1 & \text{従業員 } i \text{ が日にち } d \text{ の時間帯 } t \text{ に働く} \\ 0 & \text{従業員 } i \text{ が日にち } d \text{ の時間帯 } t \text{ に働かない} \end{cases}$$

$lm_i$  : 従業員  $i$  の実際の給料が希望給料に満たない場合の差額。

$em_i$  : 従業員  $i$  の実際の給料が、希望給料に対して過剰な場合の差額。希望給料を実際の給料が15000円以上超えるとき、それ以上を過剰分とする。

$m$  : 全従業員のうち、希望給料に実際の給料が満たない者の差額の最大値。

$v$  : 全従業員のうち、希望給料に対し、実際の給料が過剰な者の差額の最大値。

$money_i$  : 従業員  $i$  の実際の給料。

$LB1_{i,w_1,w_2}$  : 従業員  $i$  が第  $w_1$  五連日に働く回数と第  $w_2$  五連日に働く回数の差。  $LB1_{i,w_1,w_2}$  は初日から、  $LB31_{i,w_1,w_2}$  は最終日から数えた五連日を対象とする。

$ch1_{i,w,t}$  : 第  $w$  週に従業員  $i$  が時間帯  $t$  のシフトに入ることにかかった負荷。  $ch1_{i,w,t}$  は初日から、  $ch31_{i,w,t}$  は最終日から数えた週を対象とする。

### 3.2 目的関数

目的関数は重みパラメータ  $\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta$  を用いて、下記のように定め、その最小化を目的とする。

$$\begin{aligned} & \alpha \sum_{d \in D} \sum_{t \in T} (x_{0,d,t} + x_{1,d,t}) + \beta m + \gamma v \\ & + \delta \sum_{i \in A_1} lm_i + \epsilon \sum_{i \in A_1} em_i \\ & + \zeta \sum_{i \in A_1} \sum_{w_1 \in W_5} \sum_{w_2 \in W_5} (LB1_{i,w_1,w_2} + LB31_{i,w_1,w_2}) \\ & + \eta \sum_{i \in A_1} \sum_{w \in W} \sum_{t \in T} (ch1_{i,w,t} + ch31_{i,w,t}) \end{aligned}$$

各項の説明は次であるが、いずれの項も小さくすることが望ましいものである。

第 1 項 ダミー従業員が勤務シフトに入る回数。

第 2 項 全従業員のうち、希望給料に実際の給料が満たない者の差額の最大値。

第 3 項 全従業員のうち、希望給料に対して実際の給料が過剰である者の差額の最大値。

第 4 項 希望給料に実際の給料が満たない者の差額の和。

第 5 項 希望給料に対して実際の給料が過剰である者の差額の和。

第 6 項 各従業員の 5 日間ごとの勤務日数の偏りを表す値の和。

第 7 項 従業員が望まない時間帯に働くことで生じる負荷の和。

重みパラメータ  $\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta$  の値については、現状で、それぞれ 5000, 1, 1, 0.1, 0.5, 3000, 250 としている。

### 3.3 制約式

2 節で示した条件のほとんどは一般的な制約式で表現できる。この要旨では、給料に関する決定変数を定義通りに規定するために必要な制約式についてのみ説明する。目的が最小化であることから、下記の制約式を設けることで、実際の給料や希望給料との差額が計算できる。

•  $money_i$  が実際の給料を表すための制約式

$$money_i = \sum_{d \in D} \sum_{t \in T} Ms_t x_{i,d,t} \quad (i \in A_1)$$

•  $lm_i$  が従業員  $i$  の実際の給料が希望給料に満たない場合の差額の値を表すための制約式

$$lm_i \geq HM_i - money_i \quad (i \in A_1)$$

•  $m$  が全従業員のうち、希望給料に実際の給料が満たない者の差額の最大値を表すための制約式

$$m \geq lm_i \quad (i \in A_1)$$

•  $em_i$  が従業員  $i$  の実際の給料が、希望給料に対して 15000 円以上過剰な場合の差額の値を表すための制約式

$$em_i \geq money_i - HM_i - 15000 \quad (i \in A_1)$$

•  $v$  が全従業員のうち、希望給料に対し、実際の給料が過剰な者の差額の最大値を表すための制約式

$$v \geq em_i \quad (i \in A_1)$$

## 4 結果と改善方法

整数計画モデルを PuLP[2] を用いて実装し、CBC[1] により解いたところ、シフト希望のデータによっては実行時間が長くなり、解けないものもあった。勤務シフトが得られた場合には、適切な勤務シフトが出力されたことを確認している。複数あるシフト希望のデータに対して、プログラムの実行を繰り返すと、実行時間と得られる勤務シフトは、全従業員の希望給料によって異なることが分かった。そこで、各従業員の希望給料を一律  $p$  倍すること、すなわち、 $HM_i \leftarrow p \times HM_i$  とすることを考え、さらに  $p$  の値として、いくつかの値を試すことにした。実行時間については、基本的に  $p$  の値を大きくすると短くなることがわかった。また、素直に考えると、 $p$  の値として実用的な時間内で勤務シフトが得られる値の中で、1 に近いものを選ぶのが自然であると考えられるが、データによっては、 $p$  の値が大きい方が望ましい勤務シフトを与えることもあった。実行時間や勤務シフトの質を考えると、 $p$  の値を調整するような作業が必要かもしれない。

## 5 おわりに

現状では、プログラム 1 回の実行でも雇用者と労働者の両視点での満足度を一定程度満たすことはできるようだが、完璧に満たすには、希望給料の見直しや、結果に対する簡易的な手直しなど、少々の手作業は必要不可欠なようである。このように依然として、手作業を残す部分はあるが、概ね本研究の目的を達成することができたのではないかと考える。

## 参考文献

- [1] COIN-OR Branch and Cut: <https://github.com/coin-or/Cbc> (アクセス日: 2022/01/13)
- [2] PuLP: <https://coin-or.github.io/pulp/> (アクセス日: 2022/01/13)