

## Cyclic Fair Sequence Problem に対する求解モデル

高木 郁子\*      松浦 隆文      沼田 一道  
東京理科大学  
\*現在は日本電信電話株式会社

(受理 2012 年 3 月 26 日; 再受理 2012 年 8 月 31 日)

**和文概要** 複数種類のジョブと各ジョブ種の必要処理回数が与えられたとき、循環的な単位時間枠の並びに対して各ジョブをできるだけ均等に割当てるスケジューリング問題を cyclic Fair Sequence Problem (cFSP) という。cFSP は均等性を明確に定義付けていないので様々な指標が考えられている。Response Time Variability Problem (RTVP) は cFSP の代表的な問題の一つで、全ジョブ種に関して理想的な出現間隔と実際の出現間隔の差の二乗の和を最小化する問題である。Corominas らは出現間隔を直接的に列挙変数で線形表現した定式化-求解モデルを提案し、それにより単位時間枠並びの長さが 40 程度までの問題例について RTVP の厳密解が求まることを報告している。しかし、この定式化は複雑で求解可能な問題サイズの拡張や他指標への応用が見込めない。本研究では出現間隔を基礎とした様々な指標に柔軟に対応でき、かつ、より大きなサイズの問題例に対して厳密解が求まる定式化-求解モデルを提案する。

**キーワード:** 離散最適化, スケジューリング, 割当てパターン, 均等性,  $m$ 人-TSP

### 1. はじめに

複数の活動 (仕事; ジョブ) を種々の制約条件下で時間軸上に配置する作業をスケジューリングという。スケジューリングにおいて各ジョブ種の均等配置に焦点を当てた問題の一つに cyclic Fair Sequence Problem (cFSP) がある。cFSP は複数種類のジョブと一周期における各ジョブ種の必要処理回数が与えられたとき、一周期に対応する循環的な単位時間枠の並びに対して、各種ジョブをできるだけ均等に割当てる問題である [7]。この問題は、生産計画やリアルタイムシステム、ネットワーク環境におけるユーザ割当など様々な場面で現れる。cFSP は均等性を明確に定義していないので様々な評価指標が考えられている。Response Time Variability Problem (RTVP) は cFSP の代表的な問題の一つで、各ジョブ種の出現間隔について理想的な間隔との差の二乗の和を最小化する問題である [1, 2]。RTVP は NP 困難であるため発見的解法を中心に研究されている。例えば、タブーサーチや遺伝的アルゴリズム等の代表的なメタ戦略や、Hyper Heuristic Method (HH 法) と呼ばれる複数のメタ戦略を統合した発見的解法などを用いて、単位時間枠並びの長さが 200 ~ 800 程度の問題例を解いた結果が報告されている [3]。一方、厳密解法の研究は、出現間隔の扱いが難しいためか、その報告例が少ない。2007 年、Corominas ら [1] は出現間隔を直接的に列挙変数で線形表現した定式化-求解モデルを提案し、それにより単位時間枠並びの長さが 25 以下の RTVP 問題例に対して最適解が求まることを報告している。更に 2010 年には、出現間隔と各ジョブ種の割当て位置 (範囲) を (一般性を失うことなく) 限定した発展モデルを提案し、それにより長さ 40 程度の問題例まで求解可能であることを報告している [2]。しかし、この定式化は複雑で、これ以上の改善による求解可能な問題サイズの拡大や、他の均等性評価指標への応用は難しいと思われる。また、実際的な状況で cFSP を扱う際は、Corominas らのような一つの指標 (RTVP) に限定したものより、様々な評価指標に対応できるモデルの方が実用的である。これをふまえ、本研究では出現間隔を基礎とした様々な評価指標に柔軟に対応でき

る定式化-求解モデルを提案した．比較のための数値実験において，提案モデルに RTVP の指標を適用したものは，Corominas ら [2] の RTVP 専用モデルを上回る性能を示している．

以下，第 2 節では cFSP の問題設定と評価指標について述べる．第 3 節では Corominas らによる求解モデル [1, 2] を簡単に紹介する．第 4 節では cFSP を集合分割問題に帰着させた PPP モデルと，そのアイデアを発展させた TSP-cFSP モデルを提案する．第 5 節では既存モデルと提案モデルを比較した数値実験の結果を示す．第 6 節は結論である．

## 2. Cyclic Fair Sequence Problem

cFSP とは，1 周期内に処理すべきジョブの個数 (種類数)  $n$  と各ジョブ種の必要処理回数  $d_i$  ( $i = 1, 2, \dots, n$ ) が与えられたとき，これら  $D = \sum_{i=1}^n d_i$  個の処理を円環状に並んだ  $D$  個の単位時間枠 (円環セル) にできるだけ均等に割付ける問題である (図 1)．ここで，ジョブ種の集合を  $N = \{1, 2, \dots, n\}$  で表す．円環セルにおける  $D$  個の単位時間枠の並びには，12 時の位置を 1 番として，時計回りに  $1 \sim D$  の位置番号を付ける．この番号の集合を  $P = \{1, \dots, D\}$  とする．循環的な単位時間枠の並びを扱う cFSP では，ジョブ種  $i$  の  $d_i$  回の出現に対応して  $d_i$  個の出現間隔が存在することに注意する．

本研究では，出現間隔を基にした評価指標を包括的に扱う．まず，第  $i$  種ジョブの出現位置を昇順に  $\{p_1^i, p_2^i, \dots, p_{d_i}^i\}$  とし， $k$  ( $1 \leq k \leq d_i$ ) 番目の出現間隔を

$$t_k^i = \begin{cases} p_{k+1}^i - p_k^i & 1 \leq k \leq d_i - 1 \\ p_1^i + D - p_{d_i}^i & k = d_i \end{cases} \quad (2.1)$$

で定義する (図 2)． $d_i = 1$  の場合，このジョブをどの位置に割付けても  $t_1^i = D$  (定数) となり評価指標には影響しないので  $d_i \geq 2$  であるジョブ種についてのみ割付けを考えれば十分である．割付けの対象となるジョブ種の集合を  $N^e = \{i \in N \mid d_i \geq 2\}$  で表す．

ジョブ種  $i$  の理想的な出現間隔は等間隔 ( $D/d_i$ ) であるが，これは整数とは限らないし，ジョブ種同士の競合もあるので，一般に，全ジョブ種を等間隔には配置できない．ここにおいて均等性の評価指標が必要となる．このとき，ジョブ種毎に，理想的な出現間隔  $D/d_i$  と実際の出現間隔  $t_k^i$  との「ズレ (差の絶対値の冪乗)」を考え，それらの和を最小化しようとするのは自然である．すなわち，ジョブ  $i$  の評価値  $f_i$  と全体の評価値  $f$  を以下で与える．

$$f_i = \sum_{k=1}^{d_i} \left| t_k^i - \frac{D}{d_i} \right|^\alpha, \quad f = \sum_{i \in N} w_i f_i = \sum_{i \in N^e} w_i f_i \quad (2.2)$$

ただし  $w_i$  はジョブ種  $i$  の重みである．冪次数  $\alpha$  と重み  $w_i$  の設定によって，評価指標は変化する．RTVP は  $\alpha = 2$  として，各ジョブの重みを均一 ( $w_i = 1$ ) とした問題である． $\alpha = 2$  の場合，ジョブ種ごとのズレの分散 (の重み付き) 和を考えていることになる．また， $\alpha \rightarrow \infty$  とすれば，最大差の重み付き和を最小化する問題となる．以下では，既存モデルとの比較の

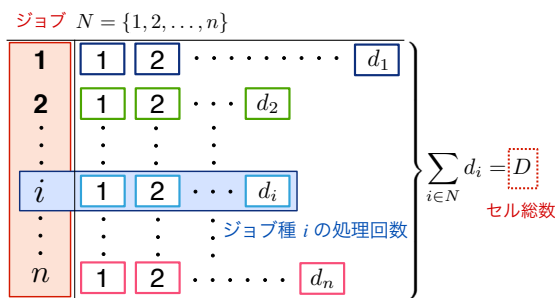


図 1: cFSP の問題設定

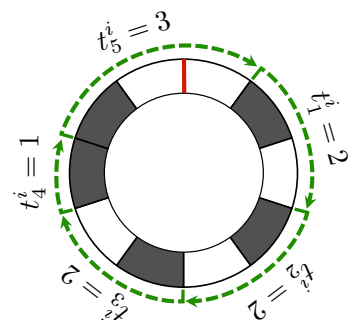


図 2: 出現間隔の与え方

ため, RTVP に沿って議論を進めるが, 4 節で提案する求解モデルは (2.2) 式で与えられる任意の評価指標に適用可能である.

### 3. 既存の厳密解法 (求解モデル)

ジョブ種の出現間隔は相対的な出現位置に依存するので, それを決定変数で記述するためには若干の工夫が必要となる. Corominas ら [1] は 2007 年に各ジョブの出現間隔を直接的に列挙変数で線形表現した求解モデルを提案し, それにより RTVP の厳密解が得られることを報告した. 更に, 2010 年にはそれを強化した求解モデルを提案し求解可能な  $D$  のサイズを拡大した [2]. 前者を旧 Corominas モデル, 後者を新 Corominas モデルと呼ぶ.

#### 3.1. 旧 Corominas モデル

文献 [1] はジョブの出現間隔が整数であることをふまえ, 出現間隔  $t_k^i$  (整数型決定変数) を  $t_k^i = \sum_{j=1}^{D-d_i+1} j \delta_{ik}^j$  と表し, RTVP を整数計画問題として定式化した. ただし,  $\delta_{ik}^j$  は「出現間隔  $t_k^i$  が  $j$  である (1) か否 (0) か」を表す 0-1 決定変数である.  $\sum_{k=1}^{d_i} t_k^i = D$  に注意すると, RTVP の目的関数は  $\sum_{i \in N^e} \sum_{k=1}^{d_i} (t_k^i - \frac{D}{d_i})^2 = \sum_{i \in N^e} \{ \sum_{k=1}^{d_i} (t_k^i)^2 - d_i (\frac{D}{d_i})^2 \}$  である. ここで,  $(t_k^i)^2 = (\sum_{j=1}^{D-d_i+1} j \delta_{ik}^j)^2 = \sum_{j=1}^{D-d_i+1} \sum_{\ell=1}^{D-d_i+1} j \ell \delta_{ik}^j \delta_{ik}^\ell$  であるが,  $j \neq \ell$  のとき  $\delta_{ik}^j \delta_{ik}^\ell = 0$ ,  $j = \ell$  のとき  $\delta_{ik}^j \delta_{ik}^\ell = \delta_{ik}^j$  であることに注意すると,  $(t_k^i)^2 = \sum_{j=1}^{D-d_i+1} j^2 \delta_{ik}^j$  となる. さらに, 第  $k$  番目に配置するジョブ種  $i$  を第  $h$  セルに割当てる (1) か否 (0) かを表す 0-1 決定変数  $y_{ikh} \in \{0, 1\}$  ( $i \in N, 1 \leq k \leq d_i, k \leq h \leq D - d_i + k$ ) を導入し, これを用いて  $t_k^i = \sum_{j=1}^{D-d_i+1} j \delta_{ik}^j$  を表現する. 旧 Corominas モデルの定式化は以下の通りである.

$$\begin{cases}
 \text{minimize} & \sum_{i \in N^e} \sum_{k=1}^{d_i} \sum_{j=1}^{D-d_i+1} j^2 \delta_{ik}^j - \sum_{i \in N^e} \frac{D^2}{d_i} & (3.1) \\
 \text{sub. to} & \sum_{h=k}^{D-d_i+k} y_{ikh} = 1 & (i \in N, 1 \leq k \leq d_i) & (3.2) \\
 & \sum_{i \in N} \sum_{\substack{k=1 \\ (h+d_i-D \leq k \leq h)}}^{d_i} y_{ikh} = 1 & (1 \leq h \leq D) & (3.3) \\
 & \sum_{h=k}^{D-d_i+k} h y_{i(k+1)h} - \sum_{h=k}^{D-d_i+k} h y_{ikh} = \sum_{j=1}^{D-d_i+1} j \delta_{ik}^j & (i \in N^e, 1 \leq k \leq d_i - 1) & (3.4) \\
 & D - \sum_{h=d_i}^D h y_{id_i h} + \sum_{h=1}^{D-d_i+1} h y_{i1h} = \sum_{j=1}^{D-d_i+1} j \delta_{id_i}^j & (i \in N^e) & (3.5) \\
 & \sum_{j=1}^{D-d_i+1} \delta_{ik}^j = 1 & (i \in N^e, 1 \leq k \leq d_i) & (3.6)
 \end{cases}
 \quad (\text{Cor-old})$$

Corominas ら [1] は汎用混合整数計画 (Mixed Integer Programming; MIP) ソルバ CPLEX [5] でこの定式化-求解モデルを解き,  $D \leq 25$  の問題例に対して厳密解を得ている.

#### 3.2. 新 Corominas モデル

新 Corominas モデル [2] は, 上・下界値の情報から (考慮すべき) 出現間隔の範囲を限定し, 0-1 決定変数の数を縮減することで求解可能サイズ ( $D$ ) を拡大したものである. ジョブ種  $i$  の出現間隔の上・下限値を  $UB_i, LB_i$  とするとき, 整数型決定変数  $t_k^i$  を,  $t_k^i = LB_i + \sum_{j=1}^{UB_i-LB_i} \gamma_{ik}^j$  と表現する. この  $UB_i, LB_i$  は, 例えば発見的解法で求めた準最適解を用いて得ることができる. また,  $\gamma_{ik}^j$  は「 $t_k^i$  が  $LB_i + j$  以下である (1) か否 (0) か」を表す 0-1 決定変数である. これを RTVP の指標に代入し, 整理すると (3.7) 式のように表せる. また, 第  $k$  番目に割当てるジョブ種  $i$  の最早割当位置を  $E_{ik}$ , 最遅割当位置を  $L_{ik}$  ( $i \in N^e, 1 \leq k \leq d_i$ ), 第  $h$  セル ( $h \in P$ ) に  $d_i = 1$  のジョブ種を配置する (1) か否 (0) かを 0-1 決定変数  $y_{1h} \in \{0, 1\}$ , 第  $k$

番目に配置するジョブ種  $i$  のセル位置を整数型決定変数  $sl_{ik} (\geq 0)$  (但し,  $E_{ik} \leq sl_{ik} \leq L_{ik}$ ),  $d_i$  が最大のジョブ種を  $i^*$  で表すと, 新 Corominas モデルは以下のように定式化できる.

$$\begin{aligned}
 & \text{minimize} && \sum_{i \in N^e} \sum_{k=1}^{d_i} \sum_{j=1}^{UB_i - LB_i} ((LB_i + j)^2 - (LB_i + j - 1)^2) \gamma_{ik}^j - \sum_{i \in N^e} \left( d_i LB_i + \frac{D^2}{d_i} \right) && (3.7) \\
 & \text{sub. to} && \sum_{h \in E_{ik}}^{L_{ik}} h y_{ikh} = sl_{ik} && (i \in N^e, 1 \leq k \leq d_i) && (3.8) \\
 & && sl_{ik+1} - sl_{ik} = LB_i + \sum_{j=1}^{UB_i - LB_i} \gamma_{ik}^j && (i \in N^e, 1 \leq k \leq d_i - 1) && (3.9) \\
 & && D - sl_{id_i} + sl_{i1} = LB_i + \sum_{j=1}^{UB_i - LB_i} \gamma_{id_i}^j && (i \in N^e) && (3.10) \\
 & && \gamma_{ik}^j \geq \gamma_{ik}^{j+1} && (i \in N^e, 1 \leq k \leq d_i, j = 1, \dots, UB_i - LB_i - 1) && (3.11) \\
 & && \sum_{i \in N^e} \sum_{k=1}^{d_i} y_{ikh} + y1_h = 1 && (h \in P) && (3.12) \\
 & && \sum_{h \in E_{ik}}^{L_{ik}} y_{ikh} = 1 && (i \in N^e, 1 \leq k \leq d_i) && (3.13) \\
 & \text{(Cor-new)} && \sum_{h \in P} y1_h = n - e && && (3.14) \\
 & && \sum_{k=1}^{d_{i^*} - 1} k^2 (sl_{i^*k+1} - sl_{i^*k}) + d_{i^*}^2 (D - sl_{i^*d_{i^*}} + sl_{i^*1}) && && \\
 & && \leq \sum_{k=1}^{d_{i^*} - 1} (d_{i^*} - k + 1)^2 (sl_{i^*k+1} - sl_{i^*k}) + (D - sl_{i^*d_{i^*}} + sl_{i^*1}) && && (3.15) \\
 & && \sum_{k=1}^{d_{i^*} - 1} k^2 (sl_{i^*k+1} - sl_{i^*k}) + d_{i^*}^2 (D - sl_{i^*d_{i^*}} + sl_{i^*1}) && && \\
 & && \leq \sum_{k=1}^{d_{i^*} - 1} (1 + ((k + j) \bmod d_{i^*}))^2 (sl_{i^*k+1} - sl_{i^*k}) + (1 + j)^2 (D - sl_{i^*d_{i^*}} + sl_{i^*1}) && && \\
 & && && && (0 \leq j \leq d_{i^*} - 2) && (3.16) \\
 & && \sum_{j=1}^{UB_i - LB_i} \sum_{k=1}^{d_i} \gamma_{ik}^j = D - d_i \cdot LB_i && (i \in N^e) && (3.17) \\
 & && \sum_{j=1}^{UB_i - LB_i} \sum_{k=1}^{d_i} (1 - \gamma_{ik}^j) = d_i \cdot UB_i - D && (i \in N^e) && (3.18)
 \end{aligned}$$

ここで,  $E_{ik}$ ,  $L_{ik}$  は  $UB_i, LB_i$  を用いて以下の式より求める.

$$\begin{cases} E_{ik} = \max(1 + LB_i \cdot (k - 1); D - UB_i \cdot (d_i - k + 1) + 1) \\ L_{ik} = \min(D - LB_i \cdot (d_i - k); k \cdot UB_i) \end{cases} \quad (i \in N^e, 1 \leq k \leq d_i) \quad (3.19)$$

Corominas ら [2] は汎用 MIP ソルバ CPLEX [5] でこの定式化-求解モデルを解き,  $D \leq 40$  の問題例に対して厳密解を得ている. しかし, 新 Corominas モデルは変数の個数を縮減するために定式化が複雑になり, これ以上の変数縮約や他の評価指標への応用が困難と考えられる. そこで, 以下では新たな観点から cFSP に対する求解モデルを検討する.

#### 4. 提案解法 (求解モデル)

各ジョブ種  $i$  に対して  $d_i$  個の出現位置を  $D$  個の単位時間枠 (円環セル) に割当てるとすべてのパターンを考える. 例えば,  $n = 3$ ,  $D = 10$ ,  $d_i = (5, 3, 2)$  の場合のジョブ種ごとのパター

ン集合 (の一部) は図 3 のようになる．このとき cFSP は，各ジョブ種のパターンから 1 個ずつ選んで円環セルを過不足なく埋め尽くす組合せの中で，評価値の和が最も小さいものを求める集合分割問題とみなせる．このアプローチはパターン生成時にジョブ種  $i$  の評価値  $f_i$  を計算できるので，様々な均等性の評価指標に柔軟に対応できる．以後このパターンを配置パターンと呼ぶ．

4.1. PPP モデル

各ジョブ種の各配置パターンを変数とした上述の cFSP 求解モデルを Pattern Partitioning Problem モデル (以下, PPP モデル) と呼ぶ [9]．配置パターンは長さ  $D$  の 0-1 ベクトルで，その第  $j$  要素は位置番号  $j$  のセル (時間枠) で当該ジョブ種の処理を行なう (1) か否 (0) かを表す．あるベクトル (配置パターン) を  $1 \sim (D-1)$  セル循環的にシフトしたものが元のベクトルと異なれば，それも異なる配置パターンとして考慮する．ジョブ種  $i (i \in N)$  の全配置パターンの総数を  $S_i$  とし，それらを評価指標の昇順 (望ましい方から順) に並べた順序集合の第  $\ell$  番目 ( $1 \leq \ell \leq S_i$ ) のパターンを  $Q_\ell^{(i)}$  と表記する．また， $Q_\ell^{(i)}$  の評価値を  $g_{i\ell}$ ， $Q_\ell^{(i)}$  の第  $j$  要素を  $a_j^{i\ell} \in \{0, 1\}$  とする．以上の定数と，ジョブ種  $i$  の配置パターンとして  $Q_\ell^{(i)}$  を採用する (1) か否 (0) かを表す変数  $x_{i\ell}$  を用いると，PPP モデルは以下のように定式化できる．

$$\begin{cases}
 \text{minimize} & \sum_{i \in N} \sum_{\ell=1}^{S_i} g_{i\ell} x_{i\ell} & (4.1) \\
 \text{sub. to} & \sum_{\ell=1}^{S_i} x_{i\ell} = 1 & (i \in N) & (4.2) \\
 & \sum_{i \in N} \sum_{\ell=1}^{S_i} a_j^{i\ell} x_{i\ell} = 1 & (j \in P) & (4.3) \\
 & x_{i\ell} \in \{0, 1\} & (i \in N, 1 \leq \ell \leq S_i) & (4.4)
 \end{cases}$$

(4.1) 式は選択した配置パターン全体の評価値を表す．(4.2) 式は各ジョブ種から配置パターンを 1 つ選ぶことを表している．(4.3) 式は各時間枠に 1 つのジョブ種が割当てられることを表している．

PPP モデルでは， $D$  が大きくなると保持すべき配置パターン (対応する 0-1 決定変数) の個数が急激に増加するので小規模の問題例しか解くことができない．汎用 MIP ソルバ Gurobi Optimizer 4.5.0 [4] を用いて， $D \leq 24$  の問題例の求解を試みたところ， $D \geq 20$  の問題例で

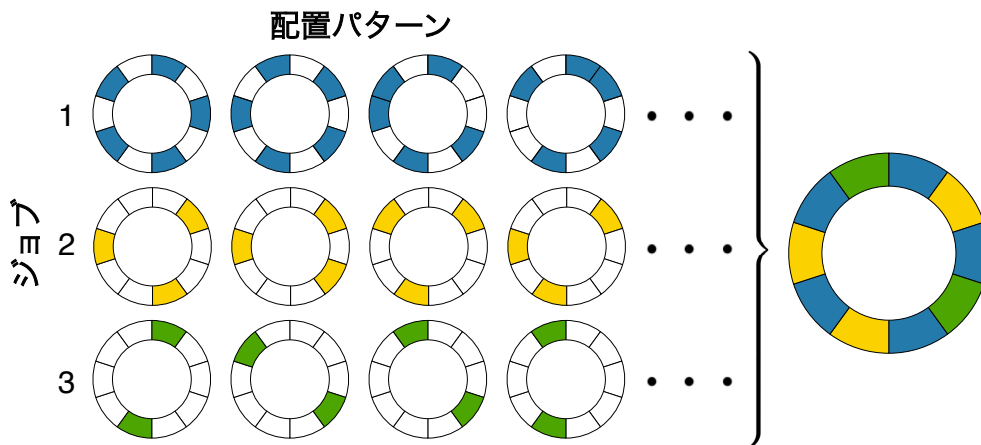


図 3: 集合分割問題としての cFSP

メモリ不足が生じた．そこで，配置パターンを限定的に生成するようにした縮約PPPモデルにより，求解可能サイズの拡大を図る．

#### 4.2. 縮約PPPモデル

各ジョブ種の配置パターンには，最適解に含まれないことを確定できるものや，回転により重複するパターンが含まれているので，これらを排除して能率化を図る．

まず，整数  $D$  の  $d_i$  分割パターン  $\lambda^{(i)} = (\lambda_1^{(i)}, \lambda_2^{(i)}, \dots, \lambda_{d_i}^{(i)})$  を，加え合わせて  $D$  となる  $d_i$  個の整数 (各々1以上  $D - d_i + 1$  以下) の降順の並びと定義する．そして，ジョブ種  $i$  の配置パターンの中で，その出現間隔を降順に並べたものが分割パターン  $\lambda^{(i)}$  に一致するもの全体の集合を  $\Pi(\lambda^{(i)})$  とする．写像  $\Pi$  により，分割パターン全体の集合と配置パターン全体の集合は1対多に対応する． $\Pi(\lambda^{(i)})$  に属する配置パターンの評価値はすべて等しいので，この評価値を分割パターン  $\lambda^{(i)}$  の評価値と定義する．

ジョブ種  $i$  の分割パターンの総数を  $T_i$  とし，それらを評価値の昇順 (望ましい方から順) に並べた順序集合の第  $\ell$  番目 ( $1 \leq \ell \leq T_i$ ) の分割パターンを  $\lambda_\ell^{(i)}$  と表記する．つぎに，各ジョブ種の配置パターン集合を， $d_i$  が最大であるジョブ種  $i^*$  と，それ以外のジョブ種に分けて，それぞれ限定する．前者については，循環的シフトによって一致する配置パターン (ベクトル) の集合を，第1要素が“1”である (任意の) パターン1個で代表させる．ジョブ種  $i^*$  を最初に置くと考えれば，このようにしても一般性を失わない．このようにして  $\Pi(\lambda^{(i^*)})$  を間引いた配置パターン集合を  $\Pi_1(\lambda^{(i^*)})$  で表す．後者については， $\Pi(\lambda^{(i)})$  から第1要素が“1”である配置パターンを除く．除いた後に残る配置パターン集合を  $\Pi_0(\lambda^{(i)})$  で表す．

cFSPの問題例  $(N, d_1, \dots, d_n)$  が与えられたとき，縮約PPPモデルは以下の手順で厳密な最適解を計算する．

##### 縮約PPPモデル計算手順

step1: 適当な発見的解法で問題 (PPP) の上界値  $\bar{z}$  を求める．

step2: (PPP) の連続緩和問題を解き，下界値  $\underline{z}$  を求める．

このとき，ジョブ種  $i^*$  については， $\bigcup_{\ell=1}^{\lfloor \mu T_{i^*} \rfloor} \Pi_1(\lambda_\ell^{(i^*)})$  を，それ以外のジョブ種については， $\bigcup_{\ell=1}^{\lfloor \mu T_i \rfloor} \Pi_0(\lambda_\ell^{(i)})$  を配置パターン (変数) の集合として用いる．

すなわち，各ジョブ種  $i$  について，分割パターン  $\lambda_\ell^{(i)}$  を評価値上位のものから順に  $\mu$  ( $0 < \mu \leq 1$ ) の割合まで考慮して緩和問題を解く．その結果，実行不能あるいは全変数 (配置パターン) の被約費用の中に負のものがあれば， $\mu$  を (徐々に) 増やして最適性の基準が満たされるまで繰り返す．

step3: step2 で得られた双対最適解を用いて，(使用していないものも含めた) 全変数  $x_{il}$  の被約費用  $\tilde{g}_{il}$  を計算し，値0の非基底変数  $x_{il}$  について  $\tilde{g}_{il} \geq \bar{z} - \underline{z}$  であれば  $x_{il} = 0$  に，値1の非基底変数  $x_{il}$  について  $\tilde{g}_{il} \leq \bar{z} - \underline{z}$  であれば  $x_{il} = 1$  に釘付けする．

step4: 釘付けで残った変数 (配置パターン) のみからなる問題を汎用MIPソルバで解く．

#### 4.3. TSP-cFSPモデル

PPPモデルは任意の評価指標を扱える柔軟なモデルであるが，配置パターンを保持しなければならないので小規模の問題例しか解くことができない．また，縮約PPPモデルのように配置パターンの釘付けと段階的生成法を工夫しても，求解可能な問題例のサイズには限界がある．そこで，配置パターンが循環する特徴を生かし，配置パターンを一つの巡回路として捉えた求解モデル [8] を提案する．これは，配置パターンの生成と求解を同時に行なうモデルである．すなわち，円環セル ( $D$  個の単位時間枠の並び) を訪問すべき  $D$  個の都市，ジョブ種  $i$  の  $d_i$  個の出現位置の列 (昇順並べたもの) をセールスマン  $i$  の巡回路とみなす．このとき，配置パターンの割付けは， $n$  人のセールスマンが各自  $d_i$  個の都市を互いに重ならないように担当して訪問する  $n$  人巡回セールスマン問題 ( $n$ -TSP) の実行可能解に対応する．そして，都市間の距離を後述のように定義すると，cFSPは各セールスマンの移動距離 (巡回路長) の和が最小となる円環セルの分割を求める問題に帰着する (図4)．以後，この求解

モデルを TSP-cFSP モデルと呼ぶ．このモデルにおいては， $d_i \geq 2$  であるジョブ種にのみセールスマンを対応させれば十分である． $|N^e|$  人のセールスマンによる円環セルの分割問題の設定をまとめると以下ようになる．

- 第  $i$  セールスマン ( $i \in N^e$ ) は， $d_i$  個のセル (単位時間枠) を時計回りに巡回訪問し開始セルに戻る．
- 各セルはいずれかのセールスマンによって高々1度訪問される．
- 各セールスマンの訪問セルは互いに重複しない．
- 各セールスマンについて，訪問セルが決まれば，それを位置番号の昇順に巡回する．ただし，最大位置番号のセルからは最小位置番号のセルへ移動する．
- 適切なセル間距離の定義の下で，総移動距離の総和が最小となる巡回路の集合は cFSP の最適解を与える．

「セル  $j$  からセル  $k$  への移動」は，(あるジョブ種が) セル  $j, k$  に相次いで出現することを意味するので，その出現間隔「 $k - j$ 」 ( $j < k$  の場合) または「 $k - j + D$ 」 ( $k > j$  の場合) が TSP における距離となる．ただし，評価指標と対応させるために，ジョブ種によって異なる理想間隔  $\frac{D}{d_i}$  との差の冪乗で，ジョブ種に依存するセル  $j-k$  間の距離  $c_{jk}^{(i)}$  を定義する．

$$c_{jk}^{(i)} = \begin{cases} \left| (k - j) - \frac{D}{d_i} \right|^\alpha & j < k \\ \left| (k - j + D) - \frac{D}{d_i} \right|^\alpha & j > k \end{cases} \quad (4.5)$$

ジョブ種  $i$  の出現位置を昇順に  $\{p_1^i, p_2^i, \dots, p_{d_i}^i\}$  とするとき，その配置の評価値 ( $f_i$ ) を (2.1), (2.2) 式で与えたものと， $\{p_1^i, p_2^i, \dots, p_{d_i}^i\}$  をセールスマン  $i$  の巡回路とみて，その巡回路長を (4.5) 式に基づき  $\sum_{k=1}^{d_i-1} c_{p_k^i p_{k+1}^i}^{(i)} + c_{p_{d_i}^i p_1^i}^{(i)}$  と計算したものは明らかに一致する．このようにして，cFSP は円環セルのセル間の距離を (4.5) 式で定義した  $|N^e|$  人巡回セールスマン問題に帰着する．

セールスマン  $i$  がセル  $j$  のつぎにセル  $k$  を訪問する (1) か否 (0) かを 0-1 決定変数  $x_{jk}^{(i)}$  ( $i \in N^e, j, k \in P, j \neq k$ ) で表すと，評価指標  $f_i$  は，

$$f_i = \sum_{j, k \in P: j \neq k} c_{jk}^{(i)} x_{jk}^{(i)} \quad (4.6)$$

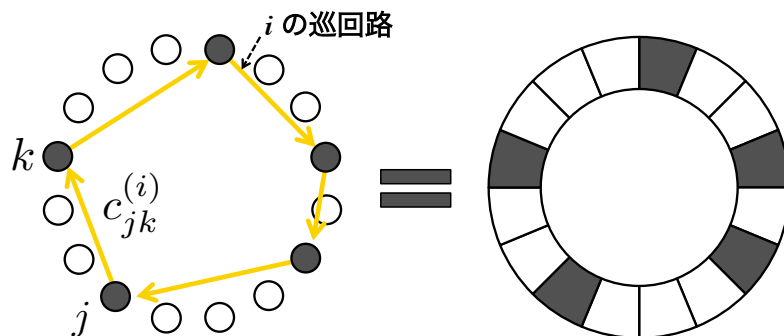


図 4: TSP-cFSP モデル

と書ける．さらに，セールスマン  $i$  が第  $j$  セルを訪問する (1) か否 (0) かを 0-1 変数  $y_j^{(i)}$  ( $i \in N^e, j \in P$ ) で表すと，TSP-cFSP モデルは以下のように定式化される．

$$\begin{cases}
 \text{minimize} & \sum_{i \in N^e} w_i \sum_{j, k \in P (j \neq k)} c_{jk}^{(i)} x_{jk}^{(i)} & (4.7) \\
 \text{sub. to} & \sum_{j \in P} y_j^{(i)} = d_i & (i \in N^e) \quad (4.8) \\
 & \sum_{i \in N^e} y_j^{(i)} \leq 1 & (j \in P) \quad (4.9) \\
 & \sum_{m \in P \setminus \{k\}} x_{km}^{(i)} = y_k^{(i)} & (i \in N^e, k \in P) \quad (4.10) \\
 & \sum_{h \in P \setminus \{k\}} x_{hk}^{(i)} = y_k^{(i)} & (i \in N^e, k \in P) \quad (4.11) \\
 & \sum_{j, k \in P, k < j} x_{jk}^{(i)} = 1 & (i \in N^e) \quad (4.12) \\
 & y_1^{(i^*)} = 1 & (i^* = \arg \max_i d_i) \quad (4.13)
 \end{cases}$$

(4.7) 式は各ジョブの評価指標の総和を最小化する目的関数である．(4.8) 式は，各ジョブは処理回数分 ( $d_i$ ) のセルを訪問することを表す．(4.9) 式は，各セルには高々 1 つのジョブを割当てること ( $d_i = 1$  のジョブの割当てを考えないため) を表す．(4.10) 式 / (4.11) 式は，第  $i$  ジョブが  $j$  番目のセルを訪れるとき，次 / 直前に訪れるジョブは一つのみであることを表す．(4.12) 式は部分巡回路除去制約で， $j > k$  であるような  $j$  番目セルから  $k$  番目セルへの移動はただ 1 回 ( $d_i$  番目に割当てたセルから 1 番目に割当てたセルへの移動のみ) であることを表す．図 5 は， $D = 16$  における  $d_i = 5$  のジョブの割当てパターン (巡回) 例であるが，順次訪問するセルの番号が逆転するのはただ 1 箇所である．(4.13) 式は，回転によって重複する実行可能解を排除するための制約式である．1 番目のセルに処理回数が最大のジョブの処理を必ず割当てて．

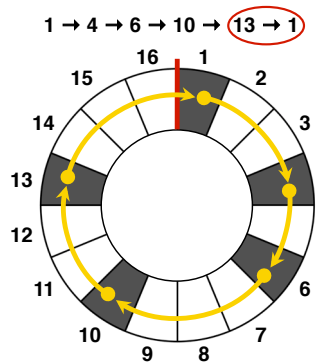


図 5: TSP-cFSP モデルの部分巡回路除去制約

### 5. 数値実験

RTVP を評価指標とした縮約 PPP モデル\*と TSP-cFSP モデルおよび既存の新 Corominas モデル (RTVP 専用) で同じ問題例を解き，提案モデルの有効性を検証する．問題例としては，

\*step1 では，各ジョブ種を  $d_i$  個ずつ計  $D$  個並べたリストの交換近傍探索を 500 回行なう多スタート戦略で上界値  $\bar{z}$  を求めている．



Corominas らが使用した  $D \in [20, 40]$  の 120 例 [6] を用いる．各問題例の  $n, d_i$  は,  $D < 35$  の場合,  $n \in [3, \lceil \frac{D}{2} \rceil]$ ,  $d_i \in [1, \lceil \frac{D-n+1}{2} \rceil]$ ． $D \geq 35$  の場合,  $n \in [3, 12]$ ． $d_i \in [1, \lfloor \frac{D-n+1}{2} \rfloor]$  の範囲で,  $D = \sum_{i \in N} d_i$  を満たすように与えられている．実験は, 汎用 MIP ソルバとして Gurobi Optimizer 4.5.0 [4] を用い, Intel Core2 Duo 2.53GHz, メモリ 2Gbyte (OS: Windows7) のノート PC 上で行った．

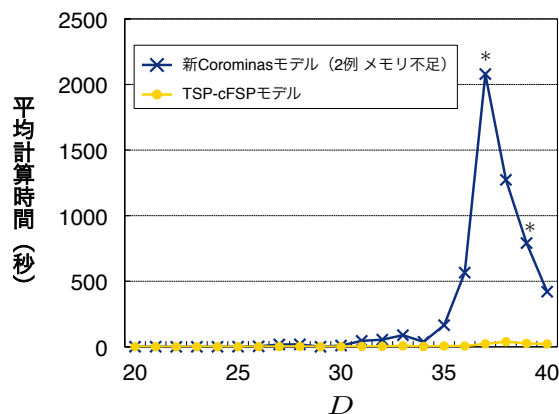
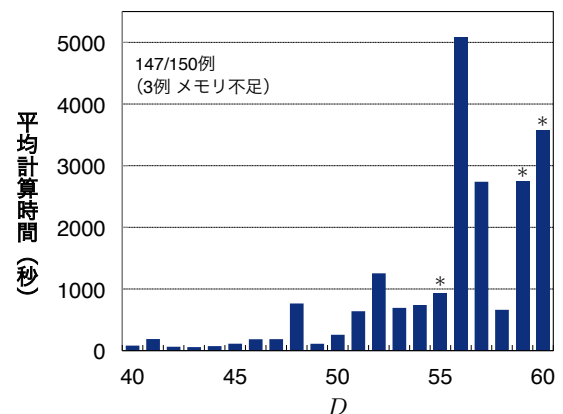
表 1 は, 求解モデルごとに, 最適解が得られた問題例の個数を  $D$  のサイズを 4 区分して示したものである．“# inst” の列は与えられた問題例の数である．新 Corominas モデルは 118 例<sup>†</sup>, TSP-cFSP モデルは 120 例全てに対して最適解を得ているが, 縮約 PPP モデルは 66 例に留まっている<sup>‡</sup>．縮約 PPP モデルを  $D$  のサイズグループごとに見ると,  $D \leq 25$  では使用した全問題例について最適解を得たが,  $D > 25$  では解の得られない場合があり,  $D > 35$  では全ての問題例で最適解を求められなかった．これは,  $\bigcup_{\ell=1}^{\lceil \mu T_i \rceil} \Pi'_1(\lambda_\ell^{(i^*)})$  および  $\bigcup_{\ell=1}^{\lceil \mu T_i \rceil} \Pi_0(\lambda_\ell^{(i)})$  の生成時点でメモリ不足に陥ったためである．縮約 PPP モデルは, 評価指標への柔軟性を有し旧 Corominas モデルより求解能力は高いが, 専用モデルである新 Corominas モデルの求解能力には及ばない．

表 1: 最適解の得られた問題例数

$D$	# inst	Cor-new	red-PPP	TSP-cFSP
[20, 25]	22	22	22	22
[26, 30]	22	22	19	22
[31, 35]	42	42	25	42
[36, 40]	34	32	0	34
Total	120	118	66	120

一方, TSP-cFSP モデルは新 Corominas モデルと (少なくとも) 同程度の求解能力を示しているので, 両モデルが求解に要した時間を比較する．図 6 はサイズ  $D$  ごとの平均計算時間を折れ線グラフで示したものである．新 Corominas モデルがメモリ不足により求解できなかった問題例が存在するサイズの点には「\*」マークを付している．

新 Corominas モデルは  $D = 35$  程度から計算時間が急激に増加し,  $D = 37, 39$  の 2 例でメモリ不足となるが, TSP-cFSP モデルは  $D$  によらず高速に最適解を得ている．新 Corominas

図 6: 新 Corominas / TSP-cFSP モデルの平均計算時間の推移 ( $D \in [20, 40]$ )図 7: TSP-cFSP モデルの平均計算時間の推移 ( $D \in [40, 60]$ )

<sup>†</sup>文献 [2] が報告している最適解は 114 例．

<sup>‡</sup>複数のモデルで最適解が求められた問題例について, 最適値はすべて一致することを確認している．

モデルの変数の個数は TSP-cFSP モデルより少ないが，整数型の変数  $sl_{ik}$  があるので負担が大きく，計算時間の増加をもたらしたと考えられる．一方，TSP-cFSP モデルは集合分割問題と TSP を組み合わせたシンプルな構造を持ち，純粋な 0-1 整数計画問題として記述されるので，計算時間が短く済んでいると考えられる．

つぎに， $D \geq 40$  の問題例 [6] を TSP-cFSP モデルで解き，求解可能サイズ ( $D$ ) を調べた．図 7 は， $D \in [40, 60]$  の問題例 150 例について， $D$  ごとの平均計算時間を棒グラフで示したものである．150 例のうち 147 例の問題例で最適解を得たが，残り 3 例はメモリ不足により解が得られなかった．それでも，TSP-cFSP モデルは求解可能な問題例のサイズ ( $D$  の値) を，新 Corominas モデルの 1.4 倍程度に広げている．

TSP-cFSP モデルにおいて  $D$  が 55 を超えると，新 Corominas モデルにおける  $35 \leq D \leq 40$  の場合と同様，計算時間が急激に増える．これは，どちらのモデルにおいても，問題例のサイズが限界を超えると指数関数的な計算時間の増加が支配的になるためと考えられる．図 6，7 では問題例のサイズを  $D$  で代表させているが，計算時間は  $n$  や  $\{d_1, d_2, \dots, d_n\}$  の構成にも依存する．両図において平均計算時間がピークを過ぎて減少しているのは，比較的解き易い ( $n$  が小さい，あるいは  $n$  が大きく  $d_i$  間の格差が大きい) 問題例を含むためと考えられる．

## 6. まとめ

本論文では，cFSP に対し，出現間隔を基礎とした多様な均等性の評価指標に対応して厳密解を求められる求解モデルを提案した．提案した求解モデルは，円環セルを各ジョブ種の配置パターンで分割する PPP モデルとそれを改善した縮約 PPP モデル，および配置パターンを巡回路とみなしてパターンを生成しながら円環セルを分割する TSP-cFSP モデルである．同一の汎用 MIP ソルバを用いた数値実験において，評価指標を RTVP のものに設定した TSP-cFSP モデルは，既存の RTVP 専用モデルより 4 割程度大きなサイズの問題例を解くことに成功している．

TSP-cFSP モデルを改良して求解可能な問題例サイズをさらに拡大すること，TSP-cFSP モデルの枠組みを利用して新しい発見的解法を開発することは今後の課題である．

## 参考文献

- [1] A. Corominas, W. Kubiak and N. Moreno: Response time variability. *Journal of Scheduling*, **10** (2007), 97–110.
- [2] A. Corominas, W. Kubiak and R. Pastor: Mathematical programming modeling of the Response Time Variability Problem. *European Journal of Operational Research*, **200** (2010), 347–357.
- [3] A. Garcia-Villoria, S. Salhi, A. Corominas and R. Pastor: Hyper-heuristic approaches for the response time variability problem. *European Journal of Operational research*, **211** (2011), 160–169.
- [4] Gurobi Optimizer: <http://www.gurobi.com>
- [5] IBM ILOG CPLEX: <http://www-06.ibm.com/software/jp/websphere/ilog/optimization/core-products-technologies/cplex>
- [6] Institute of Industrial and Control Engineering (UPC)'S HP: The Response Time Variability Problem: <http://www.ioc.upc.edu/EOLI/research/the-response-time-variability-problem>
- [7] W. Kubiak: Fair sequence. In J. Leung (ed.): *Handbook of Scheduling: Algorithms, Models and Performance Analysis* (Chapman and Hall/CRC, Boca Raton, Florida, 2004).
- [8] K. Numata: TSP-like formulation of FSP: *private communication* (2011.8.28).
- [9] 高木郁子, 松浦隆文, 沼田一道: Cyclic Fair Sequence Problem に対する解法について. 日本オペレーションズ・リサーチ学会 2011 年秋季研究発表会アブストラクト集 (2011), 150–151.

沼田一道  
東京理科大学 工学部経営工学科  
〒162-8601 東京都新宿区神楽坂 1-3  
E-mail: numata@ms.kagu.tus.ac.jp

## ABSTRACT

EXACT SOLUTION METHODS  
FOR CYCLIC FAIR SEQUENCE PROBLEM

Ikuko Takagi\* Takafumi Matsuura Kazumiti Numata  
*Tokyo University of Science*  
(\*at present, *Nippon Telegraph and Telephone Corporation*)

We propose new exact solution methods for cyclic Fair Sequence Problem (cFSP). The cFSP is a kind of scheduling problem to allocate fractional executions of given jobs to the cyclic sequence of unit-time slots with the given periodical length, where respective jobs must be processed in specified number of slots per period, and with as equal intervals as possible.

The measure of equal intervals is somewhat ambiguous. Corominas et al. presented exact solution method for cFSP with basic equality measure that minimizes the sum of square of differences between actual and ideal intervals. They reported that instances with less than or equal to 40 periodical time slots or so can be solved through their formulation by a general purpose mixed integer programming (MIP) solver.

We propose three formulations to solve cFSP based on the partition of cyclic slots by pre-generated job allocation patterns: simple partitioning formulation, improved formulation with reduced patterns and Traveling Salesman problem (TSP) like formulation considering patterns as salesman's tours. These formulations can flexibly treat any measure of equality as far as it based on the difference between actual and ideal intervals.

The results of numerical experiments to evaluate proposed methods show that TSP-like formulation outperforms existing ones and that it succeeds to solve 40% or so larger instances than the existing best.