7S-07

公平なインターバルスケジューリング問題に関する研究

酒井 洸星 *　　岩崎 敦 †
Kousei Sakai　　Atsushi Iwasaki

1 はじめに

本研究では，公平なスケジュールを求めるためのアルゴリズムを考察する．とくに区間スケジューリング問題に着目し，ゲーム理論分野でよく使われる公平性を表す解概念の１つであるナッシュ積 [5, 2] を最大化するようなスケジュールを求める問題を提案する．スケジューリング問題とは資源（マシンや人）を様々な活動（ジョブやタスク）へ時間軸を考慮して割り当てる問題である．その応用は様々であり，工場のラインやアルバイトの勤務シフトや計算機内のジョブの制御など多岐にわたっており，オペレーションズ・リサーチや人工知能の分野で古くから研究が進められている．

従来のスケジューリング問題では，工場の生産性や割り当てるジョブの数などを最大化してきた．しかし，少子高齢化にともなう労働人口の減少や働き方改革の影響で，従業員や働き手の満足度（効用）を考慮してスケジュールを作成することが求められつつある．また，従業員がお互いに不満を持たないようになるべく負担を分散することが望ましいが，単に功利主義的に従業員の効用の合計を最大化するだけでは，特定の従業員に負担が集中するなど，お互いの効用に大きな差が出てしまう．

公平分割 (fair division) [6, 1] は，資源割当問題における公平性を扱うゲーム理論と計算機科学の学際領域で，発展著しい研究分野である．アルゴリズム理論の観点から，無羨望性やマックスミニ性といった従来の公平性概念を拡張し，多様な資源割当問題での計算量を分析し，解概念同士の関係性を体系化している．例えば，文献 [7] では，区間スケジューリング問題においてマックスミニ性を目的関数としたときの近似アルゴリズムを提案評価している．これに対して，本研究ではナッシュ積を最大化する区間スケジューリング問題を整数計画問題として定式化し，分析する．

2 モデル

インターバルスケジューリング問題ではジョブとそれを処理する機械からなる系を考える．n 個のジョブの集合を $J = \{j_1, j_2, \dots, j_n\}$ とし，$m \geq 2$ 台のマシンの集合を $A = \{a_1, a_2, \dots, a_m\}$ とする．それぞれのマシンは独立したエージェントが制御するとする．離散時間を仮定し，任意の $t \in \mathbb{N}_+$ に対して，$[t, t+1)$ を t 番目の時間スロットとする．各ジョブ j_i はリリース時刻 $r_i \in \mathbb{N}_+$，〆切時刻 $d_i \in \mathbb{N}_+$，処理時間 $p_i \in \mathbb{N}_+$ をもち，$p_i \leq d_i - r_i + 1$ を満たす．処理時間 p_i はどのマシンで処理するかによらずジョブに対して一定であるとする．$[r_i, d_i]$ をジョブ間隔と呼び，連続する時間スロットの集合として表す：$\{r_i, r_i + 1, \dots, d_i\}$. 各ジョブはジョブ間隔 $[r_i, d_i]$ に含まれる p_i 個の連続する時間スロットを用いて処理される．各マシンは多くても１つのジョブを各時間スロットで請け負う．あるジョブの集合 $J' \subseteq J$ が実行可能であるとは，J' に含まれる全てのジョブがひとつも重複することなく処理できることを意味する．あるジョブ $j_i \in J$ を請け負ったエージェント a_j は非負の効用 v_{ij} を受け取る．エージェント a_j の効用はジョブの集合に対して加法的であると仮定する．全てのエージェントの効用のベクトルを $\mathbf{v}_A$ とする．任意の実行可能なジョブの集合 J' に対する a_j の効用を $\sum_{i \in J'} v_{ij}$ となる．

あるスケジュールもしくは割当を J の順序部分パーティションとして $X = (X_1, \dots, X_m)$ と表し，X_j をエージェント a_j に割り当てられたジョブの集合となる．ここで任意の $j \neq j'$ に対して，$X_j \cap X_{j'} = \emptyset$ および $X_1 \cup \cdots \cup X_m \subseteq J$ となる．あるスケジュール $\mathbf{X}$ が実行可能とは，全ての $a_j \in A$ について X_j が実行可能，つまり X_j に含まれる全てのジョブが a_j によって正しく処理されることをいう．ある問題のインスタンス $(J, A, \mathbf{v}_A)$ がリジッドとなるのは $p_i = d_i - r_i + 1$ となるときで，実行可能性制約を区間グラフで記述でき，任意のジョブ集合に対する効用を多項式時間で計算できる．

任意のスケジューリング問題は一般には制約付き資源配分問題と等価である．その目的は処理するジョブの（重み付き）総量を最大化することが多い．重みをエージェント a_j は非負の効用 v_{ij} だと解釈すれば，$\sum_{j \in A} \sum_{i \in X_j \in \mathbf{X}} v_{ij}$ を最大化するようなスケジュール $\mathbf{X}$ を求める問題となる．これはエージェントの効用の合計である社会的余剰を最大化しており，（パレート）効率的なスケジュールを求める問題になっている [3]. ジョブやエージェントの特性によってはエージェント間の効用に大きな差ができることもある．

これに対して，ジョブをエージェントに公平に割り当てるようにすることを考える．効率性とは異なり，決定版となる指標は存在しない．例えば，他者への割当が自分に割り当てられたときの効用が自分への割当が与える効用より高いとき，その他者へ羨望をもつという．このような羨望が発生しない無羨望割当 (Envy-free assignment) を公平と呼ぶことができるだろう [2, 3]. 一方で，もっとも効用が小さいエージェントの

* 電気通信大学情報理工学域
† 電気通信大学大学院情報理工学研究科

効用を最大化するようなマックスミニ基準を公平と呼ぶこともある．これらの概念は，社会的余剰も含めて，互いに包含関係にはなく，複雑なトレードオフの関係にある．一方で比較的効率性と公平性をバランスする概念として，ナッシュ積 (Nash Social Welfare) がある [5].

ナッシュ積はエージェントの効用の幾何平均として定義するので，

$$\prod_{i\in A}\sum_{i\in X_j\in\mathbf{X}} v_{ij} \tag{1}$$

を最大化するような $\mathbf{X}$ を見つける問題となる．本研究ではこれをアイゼンバーグ–ゲールの凸計画問題 (Eisenberg-Gale Convex Program) として定義する [4].

3 凸計画問題

本節では，ナッシュ積を最大化するインターバルスケジューリング問題を凸計画問題として定式化する．ナッシュ積の定義より，式 1 の対数 $\sum_{j\in A}\sum_{i\in X_j}\log v_{ij}$ を目的関数にしても一般性を失わない．スケジュール $\mathbf{X}$ を記述するために，ジョブ j_i をエージェント a_j の時間スロット t に割り当てるか否かを表す決定変数 $x_{ijt}\in\{0,1\}$ を導入する．つまり，ジョブ j_i をエージェント a_j の時間スロット t で請け負っていれば $x_{ijt}=1$，さもなければ 0 とする．ただし，各ジョブ j_i には，r_i, d_i, p_i が紐付いているため，時間スロット t が $[r_i, d_i]$ の範囲外であれば，x_{ijt} は必ず 0 になるとする．さらに補助変数としてジョブ j_i をエージェント a_j の時間スロット t で開始したか否かを表す決定変数 $y_{ijt}\in\{0,1\}$ を導入する．ジョブ j_i をエージェント a_j の時間スロット t で開始していれば $y_{ijt}=1$，さもなければ 0 とする．また，x_{ijt} と同様に，時間スロット t が $[r_i, d_i]$ の範囲外であれば，y_{ijt} は必ず 0 になるとする．これらの変数を使って各種制約を記述する．

式 4 は任意のエージェント a_j に時間スロット t で割り当てられるジョブは高々一つであることを表す．同様に式 4 は任意のジョブ j_i を時間スロット t で割り当てられるエージェントは高々一人であることを表す．式 6 は任意のジョブ j_i が開始されるのは高々 1 回だけであることを表す．

式 7-9 はジョブの継続時間をはかるための制約である．式 7 はもし時間スロット t でジョブが開始される ($y_{ijt}=1$) ならば，$x_{ijt}-x_{ijt-1}$ は 1 でならなければならない．そうでなければ，$x_{ijt}-x_{ijt-1}$ は 0 でならなければならない．もしすでにジョブが開始されていれば，x_{ijt} と x_{ijt-1} の両方が 1 になり，その差は 0 にならなければならない．

式 8 は，もし時間スロット t でジョブが開始される ($y_{ijt}=1$) ならば，$x_{ijt}=0$ になることは絶対にない．つまり，時間スロット t で開始したジョブは必ず誰かに割り当てられていることを意味する．式 9 は，もし時間スロット t でジョブを開始するには，時間スロット $t-1$ でジョブが継続していない ($x_{ijt}=0$) でなければならないことを意味する．最後に式 10 は，一度ジョブを開始したら最低 p_j スロット以上，処理を継続することを意味する．

$$\max \quad \sum_{j=1}^{n}\log u_j \tag{2}$$

$$\text{s.t.} \quad u_j \le \sum_{i=1}^{n}\sum_{t=1}^{l} v_{ij}y_{ijt} \qquad \forall j\in A \tag{3}$$

$$\sum_{i=1}^{n} x_{ijt}\le 1 \qquad \forall j\in A, t\in T \tag{4}$$

$$\sum_{j=1}^{n} x_{ijt}\le 1 \qquad \forall i\in J, t\in T \tag{5}$$

$$\sum_{j=1}^{n}\sum_{t=1}^{l} y_{ijt}\le 1 \qquad \forall i\in J \tag{6}$$

$$x_{ijt}-x_{ijt-1}\le y_{ijt} \qquad \forall i\in J, j\in A, t\in T \tag{7}$$

$$x_{ijt}\ge y_{ijt} \qquad \forall i\in J, j\in A, t\in T \tag{8}$$

$$1-x_{ijt}\ge y_{ijt} \qquad \forall i\in J, j\in A, t\in T \tag{9}$$

$$x_{ijt}\ge y_{ijt} \qquad \forall i\in J, j\in A, t\in\{1,\ldots,p_i\} \tag{10}$$

$$x_{ijt}\in\{0,1\} \qquad \forall i\in J, j\in A, t\in T \ \text{ s.t. } r_i\le t\le d_i$$

$$y_{ijt}\in\{0,1\} \qquad \forall i\in J, j\in A, t\in T \ \text{ s.t. } r_i\le t\le d_i-p_i+1$$

4 おわりに

本研究ではナッシュ積を最大化する区間スケジューリング問題を整数計画問題として定式化した．今後，文献 [3] の双対平均法にもとづくアルゴリズムを構成し，計算実験での評価を進める予定である．

参考文献

[1] G. Amanatidis, H. Aziz, G. Birmpas, A. Filos-Ratsikas, B. Li, H. Moulin, A. A. Voudouris, and X. Wu. Fair division of indivisible goods: A survey, 2022.

[2] I. Caragiannis, D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang. The unreasonable fairness of maximum nash welfare. *ACM Transactions on Economics and Computation (TEAC)*, 7(3):1–32, 2019.

[3] Y. Gao and C. Kroer. Infinite-dimensional fisher markets and tractable fair division. *arXiv preprint arXiv:2010.03025*, 2020.

[4] K. Jain and V. V. Vazirani. Eisenberg–gale markets: Algorithms and game-theoretic properties. *Games and Economic Behavior*, 70:84–106, 2010.

[5] M. Kaneko and K. Nakamura. The Nash social welfare function. *Econometrica: Journal of the Econometric Society*, pp. 423–435, 1979.

[6] J. Lang and J. Rothe. *Fair Division of Indivisible Goods*, pp. 493–550. Springer Berlin Heidelberg, 2016.

[7] B. Li, M. Li, and R. Zhang. Fair scheduling for time-dependent resources. In *Advances in Neural Information Processing Systems*, Vol. 34, pp. 21744–21756, 2021.