Simple Envy-Free and Truthful Mechanisms for Cake Cutting with a Small Number of Cuts

Takao Asano ✉
Chuo University, Tokyo, Japan

Abstract

For the cake-cutting problem, Alijani, et al. [2, 25] and Asano and Umeda [4, 5] gave envy-free and truthful mechanisms with a small number of cuts, where the desired part of each player's valuation function is a single interval on a given cake. In this paper, we give envy-free and truthful mechanisms with a small number of cuts, which are much simpler than those proposed by Alijani, et al. [2, 25] and Asano and Umeda [4, 5]. Furthermore, we show that this approach can be applied to the envy-free and truthful mechanism proposed by Chen, et al. [13], where the valuation function of each player is more general and piecewise uniform. Thus, we can obtain an envy-free and truthful mechanism with a small number of cuts even if the valuation function of each player is piecewise uniform, which solves the future problem posed by Alijani, et al. [2, 25].

2012 ACM Subject Classification Theory of computation → Algorithmic game theory and mechanism design

Keywords and phrases cake-cutting problem, envy-freeness, fairness, truthfulness, mechanism design

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2021.68

Acknowledgements The author would like to thank Professor Shigeo Tsujii of Chuo University.

1 Introduction

The problem of dividing a cake among players in a fair manner has attracted the attention of mathematicians, economists, political scientists and computer scientists [6, 7, 11, 13, 14, 15, 16, 22, 23, 24] since it was first considered by Banach and Knaster [11] and Steinhaus [27, 28]. The cake-cutting problem is often used as a metaphor for prominent real-world problems that involve the division of a heterogeneous divisible good [10, 14, 26, 32].

Formally, the cake-cutting problem is stated as follows [13]: Given a divisible heterogeneous cake C represented by an interval $[0, 1)$ and n players $N = \{1, 2, \ldots, n\}$ where each player $i \in N$ has a valuation function v_i over the cake C, divide the cake C and find an allocation of the cake C to the players that satisfies one or several fairness criteria. In the cake cutting literature, one of the most important criteria is *envy-freeness* [7]. In an envy-free allocation, each player considers her/his own allocation at least as good as any other player's allocation.

A *piece* A of cake C is a finite union of disjoint subintervals X of C. A piece A can also be viewed as a set of disjoint subintervals X of C. For a general valuation function v_i of player $i \in N$ which is integrable or piecewise continuous, the value $V_i(A)$ of a piece A of cake C for player i can be written by $\int_{x \in A} v_i(x)dx$. Thus, the value $V_i(A)$ of the piece A of disjoint subintervals X of C for player i is $V_i(A) = \sum_{X \in A} V_i(X)$.

Since general valuation functions may not have a finite discrete representation as an input to the cake-cutting problem, most algorithms and computational complexity analyses are based on oracle computation models. Among them a most popular computation model for general integrable valuation functions is the Robertson-Webb model based on two types of queries: evaluation and cut [24]. For envy-freeness, Stromquist [23, 30] showed that there is no finite envy-free cake cutting algorithm that outputs a contiguous allocation to each player for any $n \geq 3$, although an envy-free allocation with a contiguous interval allocation to each player is guaranteed to exist [29, 31]. Note that any cake cutting algorithm that outputs a

32nd International Symposium on Algorithms and Computation (ISAAC 2021).
Editors: Hee-Kap Ahn and Kunihiko Sadakane; Article No. 68; pp. 68:1–68:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

contiguous allocation to each player uses $n-1$ cuts on the cake C. If a contiguous allocation to each player is not required, Aziz and Mackenzie [6] showed that there is an envy-free cake cutting algorithm with $O(n^{n^{n^{n^{n^n}}}})$ queries. Procaccia showed that any envy-free cake cutting algorithm requires $\Omega(n^2)$ queries [21]. Furthermore, Deng, Qi and Saberi [14] showed that finding an envy-free allocation using $n-1$ cuts on cake C is PPAD-complete when valuation functions are given explicitly by polynomial-time algorithms, although their result requires very general (e.g., non-additive, non monotone) valuation functions [18].

In recent papers, some restricted classes of valuation functions have been studied [7, 8, 10, 12, 13, 20]. Piecewise uniform and piecewise constant valuation functions are two special classes of valuation functions [2, 7, 13, 25]. For a nonnegative valuation function v on cake C, let $D(v) = \{x \in C \mid v(x) > 0\}$. Thus, we can consider that $D(v)$ consists of several disjoint maximal contiguous intervals. Then v is called *piecewise uniform* if $v(x) = v(y)$ holds for all $x, y \in D(v)$. Similarly, v is called *piecewise constant* if, for each contiguous interval I in $D(v)$, $v(x') = v(x'')$ holds for all $x', x'' \in I$. Note that $v(x) \neq v(y)$ may hold for $x \in I$ and $y \in J$ when I, J are two distinct maximal contiguous intervals in $D(v)$ of piecewise constant valuation v. Thus, a piecewise uniform valuation is always piecewise constant. One of the most important properties of these valuation functions is that they can be described concisely. Kurokawa, Lai, and Procaccia [19] proved that finding an envy-free allocation in the Robertson-Webb model when the valuation functions are piecewise uniform is as hard as solving the problem without any restriction on the valuation functions.

The cake-cutting problem has been studied not only from the viewpoint of computational complexity but also from the game theoretical point of view [2, 7, 8, 9, 13, 20, 25]. Chen, Lai, Parkes, and Procaccia [13] considered a strong notion of truthfulness (denoted by strategy-proofness), in which the players' dominant strategies are to reveal their true valuations over the cake. They presented an envy-free and truthful mechanism for the cake-cutting problem based on maximum flow and minimum cut techniques [34] when the valuation functions are piecewise uniform. Aziz and Ye [7] considered the problem when valuation functions are piecewise constant and piecewise uniform. They designed three algorithms CCEA, MEA, and CSD for piecewise constant valuations, which partially solve an open problem for piecewise constant valuations posed by Chen et al. in [13]. They showed that CCEA runs in $O(n^5 M^2 \log(\frac{n^2}{M}))$ time, where M is the number of subintervals defined by the union of discontinuity points of the players' piecewise constant valuations ($M \leq 2\sum_{i \in N} m_i$ where m_i is the number of maximal contiguous intervals in $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of piecewise constant valuation v_i). They also showed that, when CCEA and MEA are restricted to piecewise uniform valuations, CCEA and MEA become essentially the same as the mechanism in [13]. However, note that CCEA, MEA and the mechanism in [13] for dividing the cake use $\Omega(nM)$ cuts [2, 25].

Alijani, Farhadi, Ghodsi, Seddighin, and Tajik [2, 25] considered that the number of cuts is important and considered the following cake-cutting problem by requiring $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of piecewise uniform valuation v_i of each player $i \in N$ to be a single contiguous interval C_i in cake C: Given a divisible heterogeneous cake C, n strategic players $N = \{1, 2, \ldots, n\}$ with valuation interval $C_i \subseteq C$ of each player $i \in N$, find a mechanism for dividing C into pieces and allocating pieces of C to n players N to meet the following conditions: (i) the mechanism is envy-free; (ii) the mechanism is truthful; and (iii) the number of cuts made on cake C is small. And they gave an envy-free and truthful mechanism with at most $2n-2$ cuts [2, 25], although their original mechanism is not actually envy free [5] and corrected later by themselves. Asano and Umeda [4, 5] also gave an alternative envy-free and truthful mechanism with at most $2n-2$ cuts.

In this paper, we give envy-free and truthful mechanisms with a small number of cuts, which lead to a much simpler mechanism than those proposed by Alijani, et al. [2, 25] and Asano and Umeda [4, 5]. Thus, we can obtain a much simpler envy-free and truthful mechanism with at most $2n-2$ cuts which runs in $O(n^3)$ time for the above cake-cutting problem. Furthermore, we show that this approach can be applied to the envy-free and truthful mechanism proposed by Chen, et al. [13] for the more general cake-cutting problem where the valuation function v_i of each player $i \in N$ is piecewise uniform. Thus, this approach can make their envy-free and truthful mechanism use at most $2M-2$ cuts and we solve the open problem posed by Alijani, et al. [2, 25], where $M \leq 2\sum_{i\in N} m_i$ and m_i is the number of maximal contiguous intervals in $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of v_i as mentioned above.

2 Preliminaries

We are given a divisible heterogeneous cake $C = [\,0,1) = \{x \mid 0 \leq x < 1\}$[1], n strategic players $N = \{1, 2, \ldots, n\}$ with valuation interval $C_i = [\alpha_i, \beta_i) = \{x \mid 0 \leq \alpha_i \leq x < \beta_i \leq 1\} \subseteq C$ of each player $i \in N$. We denote by $\mathcal{C}_N$ the (multi-) set of valuation intervals of all the players N, i.e., $\mathcal{C}_N = (C_1, C_2, \ldots, C_n)$. We also write $\mathcal{C}_N = (C_i : i \in N)$. Valuation intervals $\mathcal{C}_N$ is called *solid*, if, for every $x \in C$, there is a player $i \in N$ whose valuation interval $C_i \in \mathcal{C}_N$ contains x. As in [2, 4, 7, 25], we will assume that $\mathcal{C}_N$ is solid, i.e., $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$, throughout this paper.

A union X of mutual disjoint sets $X_1, X_2, \ldots, X_k$ is denoted by $X = X_1 + X_2 + \cdots + X_k = \sum_{\ell=1}^{k} X_\ell$. A *piece* A_i of cake C is a union of mutually disjoint subintervals $A_{i_1}, A_{i_2}, \ldots, A_{i_{k_i}}$ of C. Thus, $A_i = A_{i_1} + A_{i_2} + \cdots + A_{i_{k_i}} = \sum_{\ell=1}^{k_i} A_{i_\ell}$. A partition $A_N = (A_1, A_2, \ldots, A_n)$ of cake C into n disjoint pieces $A_1, A_2, \ldots, A_n$ is called an *allocation* of C to n players N if each piece $A_i = \sum_{\ell=1}^{k_i} A_{i_\ell}$ is allocated to player $i \in N$. We also write $A_N = (A_i : i \in N)$. Thus, in allocation $A_N = (A_i : i \in N)$ of C to n players N, $\sum_{i\in N} A_i = C$ holds and $A_i = \sum_{\ell=1}^{k_i} A_{i_\ell}$ is called an *allocated piece* of C to player $i \in N$.

For an interval $X = [x', x'')$, we denote by $cl(X)$ the *closure* of X and thus $cl(X) = [x', x''] = \{x \mid x' \leq x \leq x''\}$. For an interval $X = [x', x'')$ of C, the *length* of X, denoted by $len(X)$, is defined by $x'' - x'$. For a piece $A = \sum_{\ell=1}^{k} X_\ell$ of cake C, the *length* of A, denoted by $len(A)$, is defined by the total sum of $len(X_\ell)$, i.e., $len(A) = \sum_{\ell=1}^{k} len(X_\ell)$. For each $i \in N$ and valuation interval C_i of player i, the *value* of piece $A = \sum_{\ell=1}^{k} X_\ell$ for player i, denoted by $V_i(A)$, is the total sum of $len(X_\ell \cap C_i)$, i.e., $V_i(A) = \sum_{\ell=1}^{k} len(X_\ell \cap C_i)$. For an allocation $A_N = (A_i : i \in N)$ of cake C to n players N, if $V_i(A_i) \geq V_i(A_j)$ for all $j \in N$, then the allocated piece A_i to player i is called *envy-free* for player i. If, for every player $i \in N$, the allocated piece A_i to player i is envy-free for player i, then the allocation $A_N = (A_i : i \in N)$ to n players N is called *envy-free*.

Let $\mathcal{M}$ be a mechanism (i.e., a polynomial-time algorithm in this paper) for the cake-cutting problem. Let $\mathcal{C}_N = (C_i : i \in N)$ be an arbitrary input to $\mathcal{M}$ and $A_N = (A_i : i \in N)$ be an allocation of cake C to n players N obtained by $\mathcal{M}$. If $A_N = (A_i : i \in N)$ for every input $\mathcal{C}_N = (C_i : i \in N)$ to $\mathcal{M}$ is envy-free then $\mathcal{M}$ is called *envy-free*.

Now, assume that only player $i \in N$ gives a false valuation interval C'_i and let $\mathcal{C}'_N(i) = (C'_j : j \in N)$ (all the other players $j \neq i$ give true valuation intervals C_j and thus $C'_j = C_j$ for each $j \neq i$) be an input to $\mathcal{M}$ and let an allocation of cake C to n players N obtained

[1] We assume $C = [\,0,1) = \{x \mid 0 \leq x < 1\}$. We also assume, if an interval $X = [x', x'') = \{x \mid x' \leq x < x''\}$ of $C = [\,0,1)$ is cut at $y \in X$ with $x' < y < x''$ then X is divided into two subintervals $X' = [x', y)$ and $X'' = [y, x'')$.

by $\mathcal{M}$ be $A'_N(i) = (A'_j : j \in N)$. The values of $A_i = \sum_{\ell=1}^{k_i} A_{i_\ell}$ and $A'_i = \sum_{\ell=1}^{k'_i} A'_{i_\ell}$ for player i are $V_i(A_i) = \sum_{\ell=1}^{k_i} len(A_{i_\ell} \cap C_i)$ and $V_i(A'_i) = \sum_{\ell=1}^{k'_i} len(A'_{i_\ell} \cap C_i)$ (note that $V_i(A'_i) \neq \sum_{\ell=1}^{k'_i} len(A'_{i_\ell} \cap C'_i)$). If $V_i(A_i) \geq V_i(A'_i)$, then there is no merit for player i to give false C'_i and player i will report true valuation interval C_i to $\mathcal{M}$. For each player $i \in N$, if this holds for all such C'_is, then $\mathcal{M}$ is called *truthful* (allocation $A_N = (A_i : i \in N)$ obtained by $\mathcal{M}$ is also called *truthful*).

For valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ and an interval $X = [x', x'')$ of cake C, let $N(X)$ be the set of players i in N whose valuation interval C_i is contained in X and let $\mathcal{C}_{N(X)}$ be the (multi-) set of valuation intervals in $\mathcal{C}_N$ which are contained in X. Thus, $N(X) = \{i \in N \mid C_i \subseteq X, C_i \in \mathcal{C}_N\}$ and $\mathcal{C}_{N(X)} = (C_i \in \mathcal{C}_N : i \in N(X))$. The *density* of interval $X = [x', x'')$ of C, denoted by $\rho(X)$, is defined by $\rho(X) = \frac{len(X)}{|N(X)|} = \frac{x''-x'}{|N(X)|}$. The density $\rho(X)$ is the average length of pieces of the players in $N(X)$ when the part X of cake C is divided among the players in $N(X)$. Let $\mathcal{X}$ be the set of all nonempty intervals in C. Let $\rho_{\min}$ be the minimum density among the densities of all nonempty intervals in C, i.e., $\rho_{\min} = \min_{X \in \mathcal{X}} \ \rho(X)$. Let $\mathcal{X}_{\min} = \{X \in \mathcal{X} \mid \rho(X) = \rho_{\min}\}$. Thus, $\mathcal{X}_{\min}$ is the set of all intervals of minimum density in C. Note that, for each interval $X = [x', x'') \in \mathcal{X}_{\min}$, there are valuation intervals $C_i = [\alpha_i, \beta_i), C_j = [\alpha_j, \beta_j) \in \mathcal{C}_N$ with $x' = \alpha_i$ and $x'' = \beta_j$. Thus, the set of all intervals of minimum density in C can be computed in $O(n^2)$ time. An interval $X \in \mathcal{X}_{\min}$ is called a *maximal interval of minimum density* if no other interval of $\mathcal{X}_{\min}$ contains X properly. A *minimal interval of minimum density* is similarly defined.

3 Core Mechanism $\mathcal{M}_1$

For cake $C = [0, 1)$, n strategic players $N = \{1, 2, \ldots, n\}$, and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with $C_i = [\alpha_i, \beta_i) \subseteq C$ of each player $i \in N$, each mechanism $\mathcal{M}$ in [2, 25] and [4, 5] uses a small number of cuts and finds an allocation $A_N = (A_i : i \in N)$ to players N satisfying the following: (a) $\mathcal{M}$ is envy-free; (b) $\mathcal{M}$ is truthful; (c) $A_i \subseteq C_i$ for each $i \in N$; and (d) $\sum_{i \in N} A_i = C$. However, their mechanisms were quite complicated.

In this paper, we give a much simpler envy-free and truthful mechanism with a small number of cuts. For this purpose, we first give a core mechanism $\mathcal{M}_1$ which assumes that cake $C = [0, 1)$ is an interval of minimum density $\rho_{\min}$ in $C = [0, 1)$ (thus, $\rho_{\min} = \frac{1}{n}$).

Algorithm 1 Core Mechanism $\mathcal{M}_1$.

Input: Cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$ and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i) \subseteq C$ of each player $i \in N$ and $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$, where $C = [0, 1)$ is an interval of minimum density $\rho_{\min} = \frac{1}{n}$ in cake $C = [0, 1)$.

Output: Allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$, $len(A_i) = \rho_{\min}$ for each $i \in N$ and $\sum_{i \in N} A_i = C$.

sort $\mathcal{C}_N = (C_i : i \in N)$ in a lexicographic order with respect to (β_i, α_i) and assume $C_1 \leq C_2 \leq \cdots \leq C_n$ in this lexicographic order;

set $A_1 = [a_1, b_1) \subseteq C_1$ with length $\rho_{\min}$ such that $a_1 = \alpha_1$ and $b_1 = \alpha_1 + \rho_{\min}$;

for $i = 2$ **to** n **do**

  set $A_i = [a_i, b_i) \setminus \sum_{\ell=1}^{i-1} A_\ell$ with length $\rho_{\min}$ such that $[a_i, b_i) \subseteq C_i$ and a_i is the leftmost endpoint in $C_i \setminus \sum_{\ell=1}^{i-1} A_\ell$;

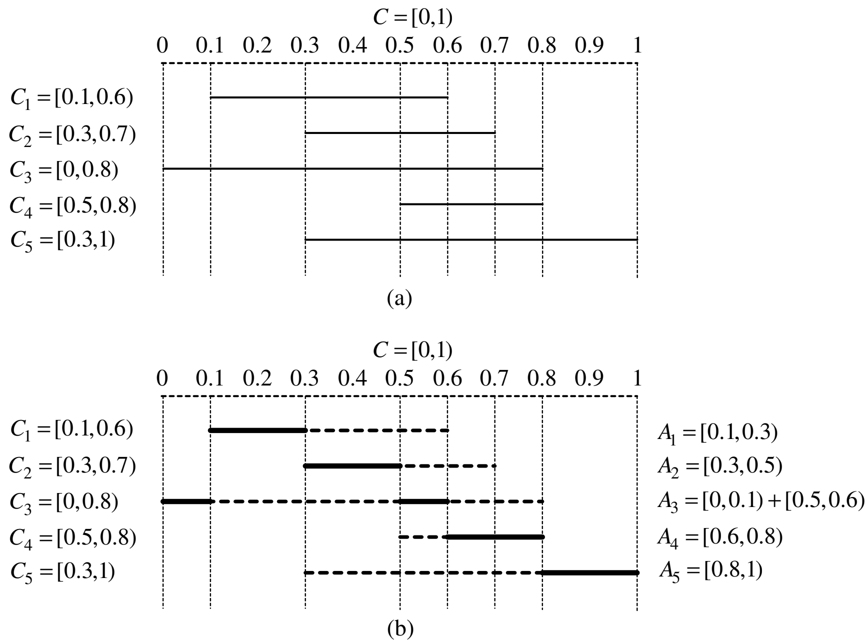

Figure 1 (a) Solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with $\rho(C) = \rho_{\min} = 0.2$. (b) Allocation $A_N = (A_i : i \in N)$ obtained by $\mathcal{M}_1$.

Figure 1 shows an example of solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with $\rho(C) = \rho_{\min} = 0.2$ and an allocation $A_N = (A_i : i \in N)$ obtained by $\mathcal{M}_1$.

▶ **Theorem 1.** *For cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$, and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i)$ of each player $i \in N$ and $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$, let $C = [0, 1)$ be an interval of minimum density $\rho_{\min} = \frac{1}{n}$ in cake $C = [0, 1)$. Then, $\mathcal{M}_1$ finds an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = \rho_{\min}$ for each $i \in N$ and $\sum_{i \in N} A_i = C$ in $O(n \log n)$ time. Furthermore, the number of cuts made by $\mathcal{M}_1$ on cake C is at most $2n - 2$.*

Proof. The number of cuts made on cake C is clearly at most $2n - 2$, since $\mathcal{M}_1$ uses at most two cuts at a_i and b_i for each $i \in N$ to obtain A_i and no cut is required at $0, 1$ of cake $C = [0, 1)$. Similarly, it can be easily shown that $\mathcal{M}_1$ runs in $O(n \log n)$ time, since lexicographical sorting of $\mathcal{C}_N = (C_i : i \in N)$ requires $O(n \log n)$ time and a_i, b_i for each $i \in N$ can be found in $O(\log n)$ time based on appropriate data structures.

We next prove the proposition that $\mathcal{M}_1$ correctly finds an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = \rho_{\min}$ for each $i \in N$ and $\sum_{i \in N} A_i = C$ by induction on n.

If $n = 1$ then the proposition is clearly true.

Now we assume that the proposition is true for $n - 1$ players and consider $n \geq 2$ players. Of course, $A_1 = [a_1, b_1) \subseteq C_1 = [\alpha_1, \beta_1)$ with $a_1 = \alpha_1$ and $b_1 = \alpha_1 + \rho_{\min}$ (thus of length $\rho_{\min}$) is allocated to player 1, since $\rho(C_1) \geq \rho_{\min}$ and thus the length of C_1 is $\beta_1 - \alpha_1 \geq \rho_{\min}$. Then we delete $A_1 = [a_1, b_1)$ and virtually consider $a_1 = b_1$ (we call this as *virtual shrinking* of hollow interval A_1 after deletion of A_1). Note that, since we performed virtual shrinking of hollow interval $A_1 = [a_1, b_1)$ and virtually considered $a_1 = b_1$, the remaining cake $C' = C \setminus A_1$ may be considered as a single interval and each $C'_k = C_k \setminus A_1$ $(k \in N \setminus \{1\})$ may also be considered as a single interval. Thus, the resulting cake-cutting problem may be considered to be the same as the original cake-cutting problem, that is, it consists of cake $C' = C \setminus A_1$

which is a single interval, players $N' = N \setminus \{1\}$, and valuation intervals $\mathcal{C}'_{N'}$ with a single interval $C'_k = C_k \setminus A_1 \subseteq C'$ of each remaining player $k \in N'$. The solidness of $\mathcal{C}'_{N'}$ (i.e., $\bigcup_{C'_i \in \mathcal{C}'_{N'}} C'_i = C'$) also holds, which can be obtained as follows.

If $\alpha_1 > 0$ then there is a valuation interval $C_j = [\alpha_j, \beta_j) \in \mathcal{C}_N$ with $\alpha_j = 0$ and $\beta_j \geq \beta_1$ by the solidness of $\mathcal{C}_N$ (i.e., $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$) and the fact that $\mathcal{C}_N$ was sorted in the lexicographic order, and thus, $\bigcup_{C'_i \in \mathcal{C}_{N'}} C'_i = C'$.

If $\alpha_1 = 0$ then let $\alpha = \min\{\alpha_i \mid C_i = [\alpha_i, \beta_i) \in \mathcal{C}_N \setminus \{C_1\}\} < 1$. Then we have $\alpha \leq \rho_{\min}$, since if $\alpha > \rho_{\min}$ then, for interval $X = [\alpha, 1)$, we would have $N(X) = N' = N \setminus \{1\}$ and $\rho(X) = \frac{len(X)}{|N(X)|} = \frac{1-\alpha}{n-1} < \rho_{\min}$ by $\rho_{\min} = \frac{1}{n}$, a contradiction since $\rho_{\min}$ is the minimum density of all intervals of C and thus $\rho(X) \geq \rho_{\min}$. This implies that $\bigcup_{C'_i \in \mathcal{C}_{N'}} C'_i = C'$ even if $\alpha_1 = 0$.

The density ρ' of intervals in this resulting cake-cutting problem can be easily shown to satisfy $\rho'(X') \geq \rho_{\min}$ for each nonempty interval X' of C' and $\rho'(C') = \rho_{\min}$. Actually, $\rho'(C') = \frac{len(C')}{|N'|} = \frac{1-\rho_{\min}}{n-1} = \frac{1-\frac{1}{n}}{n-1} = \frac{1}{n} = \rho_{\min}$. Each nonempty interval $X' \subseteq C'$ can be written by $X' = X \setminus A_1$ for some interval $X = [x', x'') \subseteq C$. Let $Y = X' \cup A_1 = X \cup A_1$ (it is possible that there are many X, but Y is uniquely determined). Of course, $X' = Y \setminus A_1$.

If Y is not a single interval of C, then $cl(X) \cap cl(A_1) = [x', x''] \cap [\alpha_1, \alpha_1 + \rho_{\min}] = \emptyset$ (i.e., $x' < x'' < \alpha_1 < \alpha_1 + \rho_{\min}$ or $\alpha_1 < \alpha_1 + \rho_{\min} < x' < x''$), which implies that there is unique $X \subseteq C$ with $X' = X \setminus A_1 = Y \setminus A_1 = X$ and $\rho'(X') = \rho(X) \geq \rho_{\min}$.

Thus, we can assume that Y is a single interval $Y = [y', y'')$ of C with $y' \leq \alpha_1 < \alpha_1 + \rho_{\min} \leq y''$ by $A_1 = [\alpha_1, \alpha_1 + \rho_{\min}) \subseteq Y = X \cup A_1$. For each $k \in N' = N \setminus \{1\}$ and $C_k = [\alpha_k, \beta_k) \in \mathcal{C}_N$, if $C_k \subseteq Y$, then we have $\emptyset \neq C'_k = C_k \setminus A_1 \in \mathcal{C}'_{N'}$ and $C'_k \subseteq X' = Y \setminus A_1$ (since if $C_k \subseteq A_1$ then $\beta_k \leq \alpha_1 + \rho_{\min} \leq \beta_1$ by $A_1 = [\alpha_1, \alpha_1 + \rho_{\min}) \subseteq C_1 = [\alpha_1, \beta_1)$ and we would have $\beta_k = \alpha_1 + \rho_{\min} = \beta_1$ and $A_1 = C_1$ by the fact that $\mathcal{C}_N = (C_i : i \in N)$ was sorted in a lexicographic order with respect to (β_i, α_i), and thus $\rho(A_1) \leq \frac{1}{2}\rho_{\min} < \rho_{\min}$, a contradiction). Similarly, if $C_k \not\subseteq Y$, then it is clear that $\emptyset \neq C'_k = C_k \setminus A_1 \in \mathcal{C}'_{N'}$ and $C'_k \not\subseteq X' = Y \setminus A_1$. Thus, $N'(X') = \{k \in N' \mid C'_k \in \mathcal{C}'_{N'}, C'_k \subseteq X'\} = \{k \in N \setminus \{1\} \mid C_k \in \mathcal{C}_N, C_k \subseteq Y\} = N(Y) \setminus \{1\}$. This implies $|N'(X')| = |N(Y)| - 1$ or $|N'(X')| = |N(Y)|$. If $1 \notin N(Y)$ (i.e., if $C_1 = [\alpha_1, \beta_1) \not\subseteq Y = [y', y''))$ then $y'' < \beta_1$ by $y' \leq \alpha_1$ and we have $N(Y) = \emptyset$ since $\mathcal{C}_N = (C_i : i \in N)$ was sorted in a lexicographic order with respect to (β_i, α_i). Thus, if $1 \notin N(Y)$ then $|N'(X')| = |N(Y)| = 0$ and $\rho'(X') = \frac{len(X')}{|N'(X')|} = \infty \geq \rho_{\min}$. Now, assume $1 \in N(Y)$. Then, $|N'(X')| = |N(Y)| - 1$ and, by $len(Y) = |N(Y)|\rho(Y)$ and $\rho(Y) \geq \rho_{\min}$, we have $\rho'(X') = \frac{len(X')}{|N'(X')|} = \frac{len(Y)-len(A_1)}{|N(Y)|-1} = \frac{|N(Y)|\rho(Y)-\rho_{\min}}{|N(Y)|-1} \geq \frac{|N(Y)|\rho_{\min}-\rho_{\min}}{|N(Y)|-1} = \rho_{\min}$.

Thus, since we performed virtual shrinking of hollow interval $A_1 = [a_1, b_1)$ and virtually considered $a_1 = b_1$, the resulting cake-cutting problem with density ρ' may be considered to be the same as the original cake-cutting problem, that is, it consists of cake $C' = C \setminus A_1$ which is a single interval of length $1 - \frac{1}{n}$ and $\rho'(C') = \rho'_{\min} = \rho_{\min} = \frac{1}{n}$, players $N' = N \setminus \{1\}$, and solid valuation intervals $\mathcal{C}'_{N'}$ with a single interval $C'_k = C_k \setminus A_1 \subseteq C'$ of each remaining player $k \in N'$ and $\bigcup_{C'_i \in \mathcal{C}'_{N'}} C'_i = C'$. Note that the lexicographic order of $\mathcal{C}'_{N'}$ is the same as that of $\mathcal{C}_{N'}$ and $C'_2 \leq \cdots \leq C'_n$ holds. Note also that, if we consider the cake C' of length $1 - \frac{1}{n}$ as being of length 1 by virtually multiplying $\frac{n}{n-1}$ then the minimum density $\rho'_{\min} = \rho_{\min} = \frac{1}{n}$ will become $\rho'_{\min} = \frac{1}{n-1}$ and we can use induction hypothesis as usual.

By induction hypothesis, the proposition is true in the resulting cake-cutting problem and we can obtain an allocation $A'_{N'} = (A'_i : i \in N')$ with $A'_i \subseteq C'_i$ and $len(A'_i) = \rho_{\min}$ for each $i \in N'$ and $\sum_{i \in N'} A'_i = C'$. From $A'_{N'} = (A'_i : i \in N')$, we can obtain an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = \rho_{\min}$ for each $i \in N'$ as follows: if $A'_i = [a'_i, b'_i)$ contains the virtually shrunken interval $A_1 = [a_1, b_1)$ then let $A_i = [a_i, a_1) + [b_1, b_i)$ by considering $a_1 \neq b_1$; otherwise, let $A_i = A'_i$. This is called *inverse virtual shrinking* of A_1. Thus, the proposition is true for n players. ◀

4 Application to Mechanism of Asano and Umeda [4]

For a given input of cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$, and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i)$ of each player $i \in N$, the mechanism of Asano and Umeda [4] first finds all the maximal intervals of minimum density $\rho_{\min}$. Let $H_1 = [h'_1, h''_1), \ldots, H_L = [h'_L, h''_L)$ be all the maximal intervals of minimum density $\rho_{\min}$ in cake $C = [0, 1)$. Their mechanism then cuts cake $C = [0, 1)$ at both endpoints of each H_ℓ ($\ell = 1, \ldots, L$). As shown in [4], the closures of two distinct maximal intervals of minimum density are disjoint and these cuts at both endpoints of each maximal interval of minimum density can be done independently. By these cuts, the original cake-cutting problem is reduced into two types of cake-cutting subproblems of type (i) and type (ii) as follows:

- **(i)** the cake-cutting problem within each maximal interval $H_\ell = [h'_\ell, h''_\ell)$ ($\ell = 1, \ldots, L$) of minimum density (which consists of cake H_ℓ, players $N(H_\ell)$ whose valuation intervals are in H_ℓ and valuations $\mathcal{C}_{N(H_\ell)}$ with density ρ); and
- **(ii)** the cake-cutting problem obtained by deleting all $H_\ell = [h'_\ell, h''_\ell)$ ($\ell = 1, \ldots, L$), i.e., the cake-cutting problem for cake $C' = C \setminus \sum_{\ell=1}^{L} H_\ell$, players $N' = N \setminus \sum_{\ell=1}^{L} N(H_\ell)$ and valuations $\mathcal{C}'_{N'}$ (which consists of valuations $C'_k = C_k \setminus \sum_{\ell=1}^{L} H_\ell \neq \emptyset$ for all $k \in N'$) with density ρ' and $\bigcup_{C'_k \in \mathcal{C}'_{N'}} C'_k = C'$.

Note that the cake-cutting problem of type (i) is almost the same as the original cake-cutting problem, since cake H_ℓ is a single interval, each valuation $C_k \in \mathcal{C}_{N(H_\ell)}$ is also a single interval, and the valuation intervals $\mathcal{C}_{N(H_\ell)}$ is solid as shown in [4]. Thus, based on the core mechanism $\mathcal{M}_1$ (Algorithm 1), for each $\ell = 1, \ldots, L$, we can find an allocation $A_{N(H_\ell)} = (A_i : i \in N(H_\ell))$ with $A_i \subseteq C_i$ and $len(A_i) = \rho_{\min}$ for each $i \in N(H_\ell)$ and $\sum_{i \in N(H_\ell)} A_i = H_\ell$.

On the other hand, the cake-cutting problem of type (ii) is different from the original cake-cutting problem, because the resulting cake $C' = C \setminus \sum_{\ell=1}^{L} H_\ell$ may become a set of two or more disjoint intervals and each remaining valuation $C'_k = C_k \setminus \sum_{\ell=1}^{L} H_\ell \neq \emptyset$ may also become a set of two or more disjoint intervals. However, the cake-cutting problem of type (ii) can be solved in almost the same way by *virtually shrinking* all H_ℓ. That is, we virtually shrink each *hollow interval* $H_\ell = [h'_\ell, h''_\ell)$ (since H_ℓ was already deleted) and virtually consider $h'_\ell = h''_\ell$. By virtually shrinking of all $H_\ell = [h'_\ell, h''_\ell)$, cake $C' = C \setminus \sum_{\ell=1}^{L} H_\ell$ becomes a single interval $C'^{(S)}$, players $N' = N \setminus \sum_{\ell=1}^{L} N(H_\ell)$ remains the same, each valuation $C'_k \in \mathcal{C}'_{N'}$ becomes a single interval $C'^{(S)}_k$ of $C'^{(S)}$, and the valuation intervals $\mathcal{C}'^{(S)}_{N'} = (C'^{(S)}_k : k \in N')$ becomes solid (i.e., $\bigcup_{k \in N'} C'^{(S)}_k = C'^{(S)}$). Thus, by virtually shrinking of all H_ℓ, the cake-cutting problem of type (ii) above can be reduced to the cake-cutting problem of type (i) for cake $C'^{(S)}$, players $N' = N \setminus \sum_{\ell=1}^{L} N(H_\ell)$, solid valuation intervals $\mathcal{C}'^{(S)}_{N'} = (C'^{(S)}_k : C'_k \in \mathcal{C}'_{N'})$ with $\bigcup_{k \in N'} C'^{(S)}_k = C'^{(S)}$ and the same density $\rho'^{(S)} = \rho'$, which can be solved recursively. Note that, if $\rho(C) > \rho_{\min}$ then the minimum density $\rho'_{\min}$ of intervals in the cake-cutting problem of type (ii) satisfies $\rho'_{\min} > \rho_{\min}$ as shown in [4, 5].

From an allocation $A'^{(S)}_{N'} = (A'^{(S)}_k : k \in N')$ to players N' where $A'^{(S)}_k$ is the allocated piece of cake $C'^{(S)}$ to player $k \in N'$ with $A'^{(S)}_k \subseteq C'^{(S)}_k$ and $\sum_{i \in N'} A'^{(S)}_k = C'^{(S)}$, we obtain an allocation $A'_{N'} = (A'_k : k \in N')$ to players N' where A'_k is the allocated piece of cake C' to player k with $A'_k \subseteq C'_k$ and $\sum_{i \in N'} A'_k = C'$ as follows: if $A'^{(S)}_k$ contains a shrunken interval $H^{(S)}_\ell$ of hollow interval H_ℓ, then let A'_k be the set of disjoint intervals obtained from $A'^{(S)}_k$ by restoring each shrunken interval $H^{(S)}_\ell$ in $A'^{(S)}_k$ to the original hollow interval $H_\ell = [h'_\ell, h''_\ell)$; otherwise, let $A'_k = A'^{(S)}_k$. They called this *inverse shrinking* of all H_ℓ ($\ell = 1, \ldots, L$) in [4].

We call the method based on the core mechanism $\mathcal{M}_1$ (Algorithm 1) described above the *modified mechanism of Asano and Umeda*. The details are in Section 5. Note that all the maximal intervals of minimum density $\rho_{\min}$ can be obtained in $O(n^2)$ time, since there are

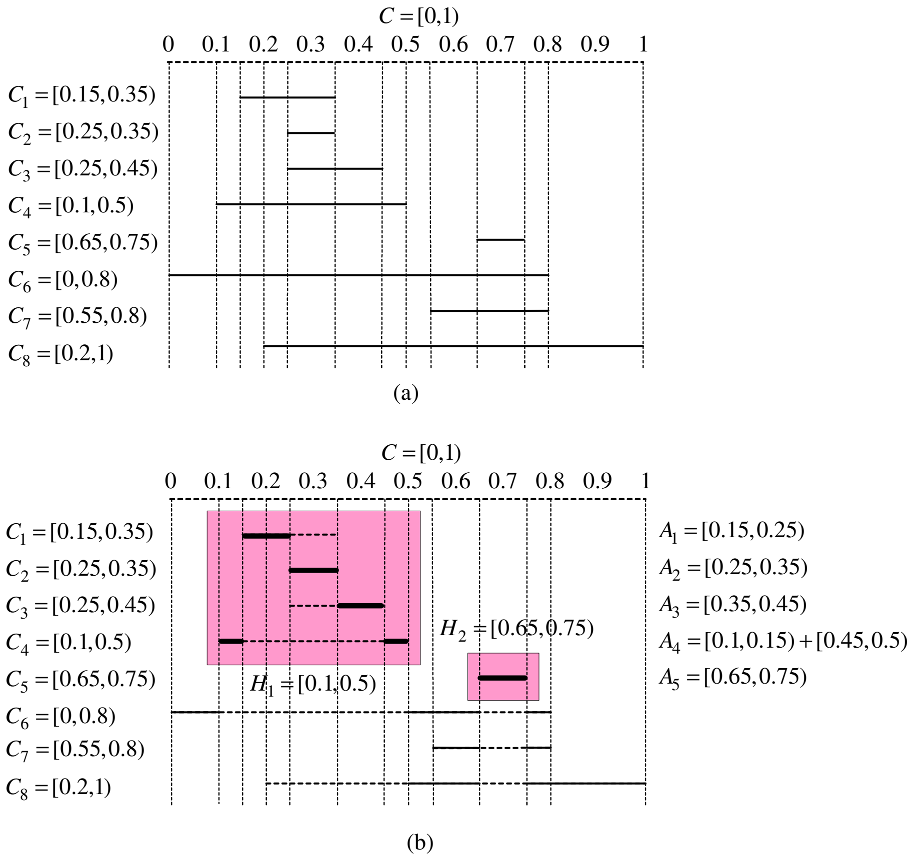

Figure 2 (a) Example of $\mathcal{C}_N = (C_i : i \in N)$. (b) The maximal intervals H_1, H_2 of minimum density $\rho_{\min} = 0.1$, with $N(H_1) = \{1, 2, 3, 4\}$, $A_{N(H_1)} = (A_i : i \in N(H_1))$, $N(H_2) = \{5\}$, $A_{N(H_2)} = (A_5)$ in the 1st iteration.

at most $2n$ endpoints of the valuation intervals in $\mathcal{C}_N$ and the endpoints of each interval of minimum density are endpoints of some valuation intervals. The number of cuts made over C is also at most $2n - 2$. Envy-freeness and truthfulness will be given in Section 5, although they were given in [5] (also given in [2, 13, 25]). Thus, the following theorem holds as in [4].

▶ **Theorem 2.** *The modified mechanism of Asano and Umeda correctly finds, in $O(n^3)$ time, an envy-free and truthful allocation $A_N = (A_i : i \in N)$ of cake C to n players N with $A_i \subseteq C_i$ for each player $i \in N$ and $\sum_{i \in N} A_i = C$. Furthermore, the number of cuts made over C by the mechanism is at most $2n - 2$.*

For an input example in Figure 2(a), the modified mechanism of Asano and Umeda works as shown in Figure 2(b) and Figure 3.

5 Details of Modified Mechanism of Asano and Umeda

As we described in Section 4, Mechanism of Asano and Umeda [4] can be significantly simplified based on $\mathcal{M}_1$ (Algorithm 1). Actually, $\mathcal{M}_1$ can be slightly modified and used as Procedure CutMaxInterval$(\cdot, \cdot, \cdot)$ in Mechanism of Asano and Umeda [4] as follows.

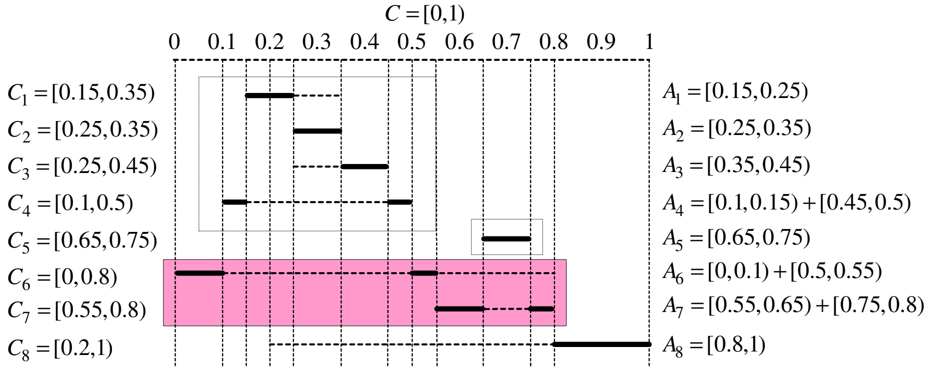

Figure 3 The second and third iterations for the example in Figure 2. In the second iteration, the minimum density is $\rho_{\min} = 0.15$ and $N(H_1) = \{6, 7\}$, $A_6 = [0, 0.1) + [0.5, 0.55)$ and $A_7 = [0.55, 0.65) + [0.75, 0.8)$. In the third (last) iteration, the minimum density is $\rho_{\min} = 0.2$ and $N(H_1) = \{8\}$ and $A_8 = [0.8, 1)$.

Procedure CutMaxInterval$(R, H, \mathcal{D}_R)$.

Input: Cake $H = [h', h'')$, players R and solid valuation intervals $\mathcal{D}_R = (D_i : i \in R)$ with valuation interval $D_i = [\alpha'_i, \beta'_i) \subseteq H$ of each player $i \in R$ and $\bigcup_{D_i \in \mathcal{D}_R} D_i = H$, where H is an interval of minimum density $\rho_{\min}$ in H.
Output: Allocation $A_R = (A_i : i \in R)$ with $A_i \subseteq D_i$ and $len(A_i) = \rho_{\min}$ for each $i \in R$ and $\sum_{i \in R} A_i = H$.

sort $\mathcal{D}_R = (D_i = [\alpha'_i, \beta'_i) : i \in R)$ in a lexicographic order with respect to (β'_i, α'_i) and assume $R = \{r_1, r_2, \ldots, r_{|R|}\}$ and $D_{r_1} \le D_{r_2} \le \cdots \le D_{r_{|R|}}$ in this order;
set $A_{r_1} = [a_{r_1}, b_{r_1}) \subseteq D_{r_1}$ with length $\rho_{\min}$ such that $a_{r_1} = \alpha'_{r_1}$ and $b_{r_1} = \alpha'_{r_1} + \rho_{\min}$;
for $i = 2$ **to** $|R|$ **do**
  set $A_{r_i} = [a_{r_i}, b_{r_i}) \setminus \sum_{\ell=1}^{i-1} A_{r_\ell}$ with length $\rho_{\min}$ such that $[a_{r_i}, b_{r_i}) \subseteq D_{r_i}$ and a_{r_i} is the leftmost endpoint in $D_{r_i} \setminus \sum_{\ell=1}^{i-1} A_{r_\ell}$;

Thus, our modified mechanism of Asano and Umeda can be written as follows (we omit inverse virtual shrinking).

Algorithm 2 Modified Mechanism of Asano and Umeda.

Input: Cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$ and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i) \subseteq C$ of each player $i \in N$ and $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$.
Output: Allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ for $i \in N$ and $\sum_{i \in N} A_i = C$.
CutCake$(N, C, \mathcal{C}_N)$;

Now we will give a proof of envy-freeness and truthfulness of Modified Mechanism of Asano and Umeda (Alogorithm 2) described in Theorem 2, which is almost the same as given in [13, 25].

Let T be the number of recursive calls CutCake$(\cdot, \cdot, \cdot)$ in Modified Mechanism of Asano and Umeda (Algorithm 2). Note that, although we use $D^{(S)}$, $D_k^{(S)} \in \mathcal{D}_{P'}^{(S)}$, $\mathcal{D}_{P'}^{(S)}$ in CutCake$(P', D^{(S)}, \mathcal{D}_{P'}^{(S)})$ which are obtained from D', $D'_k \in \mathcal{D}'_{P'}$, $\mathcal{D}'_{P'}$ by virtual shrinking of all $H_1, \ldots, H_L$, we will not distinguish them from now on, since we just performed *virtual*

Procedure CutCake($P, D, \mathcal{D}_P$).

Input: Cake D which can be considered to be a single interval, players P, and solid valuation intervals $\mathcal{D}_P = (D_i : i \in P)$ with valuation interval $D_i = [\alpha'_i, \beta'_i) \subseteq D$ of each player $i \in P$ and $\bigcup_{D_i \in \mathcal{D}_P} D_i = D$ (the density of each interval X of D is denoted by $\rho(X)$).

Output: Allocation $A_P = (A_i : i \in P)$ with $A_i \subseteq D_i$ for $i \in P$ and $\sum_{i \in P} A_i = D$.

Find all the maximal intervals of minimum density $\rho_{\min}$ in the cake-cutting problem with cake D, players P and solid valuation intervals $\mathcal{D}_P$;
Let $H_1 = [h'_1, h''_1), \ldots, H_L = [h'_L, h''_L)$ be all the maximal intervals of minimum density $\rho_{\min}$;
for $\ell = 1$ **to** L **do**
  cut cake D at both endpoints h'_ℓ, h''_ℓ of H_ℓ;
  $R_\ell = \{k \in P \mid D_k \subseteq H_\ell, D_k \in \mathcal{D}_P\}$; $\mathcal{D}_{R_\ell} = (D_k \in \mathcal{D}_P : k \in R_\ell)$;
  CutMaxInterval($R_\ell, H_\ell, \mathcal{D}_{R_\ell}$);
$P' = P$; $D' = D$;
for $\ell = 1$ **to** L **do** $P' = P' \setminus R_\ell$; $D' = D' \setminus H_\ell$;
// $P' = P \setminus \sum_{\ell=1}^{L} R_\ell$ and $D' = D \setminus \sum_{\ell=1}^{L} H_\ell$
if $P' \neq \emptyset$ **then**
  $\mathcal{D}'_{P'} = \emptyset$;
  for each $D_k \in \mathcal{D}_P$ with $k \in P'$ **do** $D'_k = D_k \setminus \sum_{\ell=1}^{L} H_\ell$; $\mathcal{D}'_{P'} = \mathcal{D}'_{P'} + \{D'_k\}$;
  Let $D^{(S)}$, $D_k^{(S)} \in \mathcal{D}_{P'}^{(S)}$, $\mathcal{D}_{P'}^{(S)}$ be obtained from D', $D'_k \in \mathcal{D}'_{P'}$, $\mathcal{D}'_{P'}$ by virtual shrinking of all $H_1, \ldots, H_L$;
  CutCake($P', D^{(S)}, \mathcal{D}_{P'}^{(S)}$);

shrinking. Thus, we consider $D^{(S)} = D'$, $(D_k^{(S)} \in \mathcal{D}_{P'}^{(S)}) = (D'_k \in \mathcal{D}'_{P'})$ and $\mathcal{D}_{P'}^{(S)} = \mathcal{D}'_{P'}$. We denote by CutCake($P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}}$) the t-th recursive call of CutCake($\cdot,\cdot,\cdot$). Note that the first call of CutCake($\cdot,\cdot,\cdot$) is CutCake($N, C, \mathcal{C}_N$). Let $\rho_{\min}^{(t)}$ be the minimum density of the cake cutting problem with cake $D^{(t)}$, players $P^{(t)}$ and the solid valuation intervals $\mathcal{D}_{P^{(t)}}$. Clearly, $C = D^{(1)} \supset D^{(2)} \supset \cdots \supset D^{(T)}$ and $N = P^{(1)} \supset P^{(2)} \supset \cdots \supset P^{(T)}$. Furthermore, as shown in [4, 5], the inequality

$$\rho_{\min}^{(1)} < \rho_{\min}^{(2)} < \cdots < \rho_{\min}^{(T)}$$

holds. We denote by CutMaxInterval($R_\ell^{(t)}, H_\ell^{(t)}, \mathcal{D}_{R_\ell^{(t)}}$) the CutMaxInterval($\cdot,\cdot,\cdot$) called in CutCake($P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}}$), where $H_\ell^{(t)}$ is a maximal interval of minimum density $\rho_{\min}^{(t)}$ in cake $D^{(t)}$, players $P^{(t)}$ and solid valuation intervals $\mathcal{D}_{P^{(t)}}$ by virtualy shrinking of hollow pieces in $C \setminus D^{(t)}$. For each player $i \in N$, there is $t \in \{1, \ldots, T\}$ such that allocation $A_i \subseteq C_i$ to player i is determined in CutMaxInterval($R_\ell^{(t)}, H_\ell^{(t)}, \mathcal{D}_{R_\ell^{(t)}}$). Thus, it is clear that Modified Mechanism of Asano and Umeda (Algorithm 2) finds an allocation $A_N = (A_i : i \in N)$ of cake C to n players N with $A_i \subseteq C_i$ for each player $i \in N$ and $\sum_{i \in N} A_i = C$.

Envy-freeness can be proved as follows (which is almost the same as given in [13, 25]). Let $A_i \subseteq C_i$ to player i be determined in CutMaxInterval($R_\ell^{(t)}, H_\ell^{(t)}, \mathcal{D}_{R_\ell^{(t)}}$) called in the t-th recursive call CutCake($P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}}$). Thus, $V_i(A_i) = len(A_i) = \rho_{\min}^{(t)}$. Similarly, let $A_j \subseteq C_j$ to player j be determined in CutMaxInterval($R_{\ell'}^{(t')}, H_{\ell'}^{(t')}, \mathcal{D}_{R_{\ell'}^{(t')}}$) called in the t'-th recursive call CutCake($P^{(t')}, D^{(t')}, \mathcal{D}_{P^{(t')}}$). If $t' \le t$ then $V_i(A_j) = len(A_j \cap C_i) \le len(A_j) = \rho_{\min}^{(t')} \le \rho_{\min}^{(t)} = len(A_i) = V_i(A_i)$. Otherwise (i.e., if $t' > t$), although $V_j(A_j) = len(A_j) = \rho_{\min}^{(t')} > \rho_{\min}^{(t)} = len(A_i) = V_i(A_i)$, we have $A_j \cap C_i = \emptyset$ and $V_i(A_j) = len(A_j \cap C_i) = 0 \le \rho_{\min}^{(t)} = len(A_i) = V_i(A_i)$. Thus, envy-freeness is clear.

Truthfulness can be proved as follows. Assume that only player $i \in N$ gives a false valuation interval C'_i and let $\mathcal{C}'_N(i) = (C'_j : j \in N)$ be an input to Modified Mechanism of Asano and Umeda (Algorithm 2) and $A'_N(i) = (A'_j : j \in N)$ be the obtained allocation of cake C to n players N (in the argument below, we assume that $\mathcal{C}'_N(i) = (C'_j : j \in N)$ is solid, although this restriction can be removed by a little more complicated argument). Let $A_i \subseteq C_i$ to player i be determined in CutMaxInterval$(R^{(t)}_\ell, H^{(t)}_\ell, \mathcal{D}_{R^{(t)}_\ell})$ called in the t-th recursive call CutCake$(P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}})$. Similarly, let $A'_i \subseteq C'_i$ with $A'_i = \sum_{\ell'=1}^{k'_i} A'_{i_{\ell'}}$ to player i be determined in CutMaxInterval$(R'^{(t')}_{\ell'}, H'^{(t')}_{\ell'}, \mathcal{D}'_{R'^{(t')}_{\ell'}})$ called in the t'-th recursive call CutCake$(P'^{(t')}, D'^{(t')}, \mathcal{D}'_{P'^{(t')}})$. Thus, $V_i(A_i) = \mathit{len}(A_i) = \rho^{(t)}_{\min}$ and $V_i(A'_i) = \sum_{\ell'=1}^{k'_i} \mathit{len}(A'_{i_{\ell'}} \cap C_i) \le \sum_{\ell'=1}^{k'_i} \mathit{len}(A'_{i_{\ell'}}) = \rho'^{(t')}_{\min}$, where $\rho'^{(t')}_{\min}$ is the minimum density of the intervals in cake $D'^{(t')}$ with $P'^{(t')}$ and $\mathcal{D}'_{P'^{(t')}}$. We divide the case into two cases: (i) $\rho'^{(t')}_{\min} \le \rho^{(t)}_{\min}$; and (ii) $\rho'^{(t')}_{\min} > \rho^{(t)}_{\min}$.

(i) $\rho'^{(t')}_{\min} \le \rho^{(t)}_{\min}$. In this case, it is clear that $V_i(A_i) = \mathit{len}(A_i) = \rho^{(t)}_{\min} \ge \rho'^{(t')}_{\min} = \sum_{\ell'=1}^{k'_i} \mathit{len}(A'_{i_{\ell'}}) \ge \sum_{\ell'=1}^{k'_i} \mathit{len}(A'_{i_{\ell'}} \cap C_i) = V_i(A'_i)$.

(ii) $\rho'^{(t')}_{\min} > \rho^{(t)}_{\min}$. In this case, $t' \ge t$ holds, which can be shown as follows.

Suppose contrarily that $t' < t$. Then for two inputs $\mathcal{C}_N = (C_j : j \in N)$ and $\mathcal{C}'_N(i) = (C'_j : j \in N)$, Modified Mechanism of Asano and Umeda (Algorithm 2) makes the same behavior before the t'-th recursive calls CutCake$(P^{(t')}, D^{(t')}, \mathcal{D}_{P^{(t')}})$ and CutCake$(P'^{(t')}, D'^{(t')}, \mathcal{D}'_{P'^{(t')}})$. Thus, $P'^{(t'')} = P^{(t'')}$, $D'^{(t'')} = D^{(t'')}$ and $\mathcal{D}'_{P'^{(t'')}} \setminus \{D'^{(t'')}_i\} = \mathcal{D}_{P^{(t'')}} \setminus \{D^{(t'')}_i\}$ for each $t'' = 1, \ldots, t'$, where $D^{(t'')}_i = C_i \cap D^{(t'')}$ and $D'^{(t'')}_i = C'_i \cap D'^{(t'')}$. Let $X = [x', x'')$ be a maximal interval of minimum density $\rho^{(t')}_{\min}$ in $D^{(t')} = D'^{(t')}$. Thus, $\rho^{(t')}(X) = \rho^{(t')}_{\min}$. Furthermore, $D^{(t')}_i = C_i \cap D^{(t')} \not\subseteq X$, since otherwise (i.e., if $D^{(t')}_i \subseteq X$) A_i to player i would be determined in the t'-th recursive call CutCake$(P^{(t')}, D^{(t')}, \mathcal{D}_{P^{(t')}})$, which is a contadiction (since $t > t'$ and A_i is determined in the t-th call CutCake$(P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}})$). Thus, if $D'^{(t')}_i = C'_i \cap D'^{(t')} \not\subseteq X$ then $\rho'^{(t')}(X) = \rho^{(t')}(X)$ holds by $D^{(t')}_i \not\subseteq X$, and otherwise (i.e., if $D'^{(t')}_i \subseteq X$) $\rho'^{(t')}(X) < \rho^{(t')}(X)$ holds by $D^{(t')}_i \not\subseteq X$. This implies that $\rho'^{(t')}(X) \le \rho^{(t')}(X)$ and $\rho'^{(t')}_{\min} \le \rho'^{(t')}(X) \le \rho^{(t')}(X) = \rho^{(t')}_{\min} < \rho^{(t)}_{\min}$ by $t' < t$. However, this is a contradiction, since $\rho'^{(t')}_{\min} > \rho^{(t)}_{\min}$ in this case.

Thus, we have $t' \ge t$ in this case of $\rho'^{(t')}_{\min} > \rho^{(t)}_{\min}$. As we mentioned above, for two inputs $\mathcal{C}_N = (C_j : j \in N)$ and $\mathcal{C}'_N(i) = (C'_j : j \in N)$, Modified Mechanism of Asano and Umeda (Algorithm 2) makes the same behavior before the t-th recursive calls CutCake$(P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}})$ and CutCake$(P'^{(t)}, D'^{(t)}, \mathcal{D}'_{P'^{(t)}})$. Thus, $P'^{(t)} = P^{(t)}$, $D'^{(t)} = D^{(t)}$ and $\mathcal{D}'_{P'^{(t)}} \setminus \{D'^{(t)}_i\} = \mathcal{D}_{P^{(t)}} \setminus \{D^{(t)}_i\}$, where $D^{(t)}_i = C_i \cap D^{(t)}$ and $D'^{(t)}_i = C'_i \cap D'^{(t)}$. For each player $j \in N$ with A_j determined in CutMaxInterval$(R^{(t)}_\ell, H^{(t)}_\ell, \mathcal{D}_{R^{(t)}_\ell})$ in the t-th call CutCake$(P^{(t)}, D^{(t)}, \mathcal{D}_{P^{(t)}})$, let A'_j be determined in the t'_j-th call CutCake$(P'^{(t'_j)}, D'^{(t'_j)}, \mathcal{D}'_{P'^{(t'_j)}})$. Thus $t'_j \ge t$.

We will show that $\rho'^{(t)}_{\min} \ge \rho^{(t)}_{\min}$. If $t' = t$ then this is true since $\rho'^{(t)}_{\min} = \rho'^{(t')}_{\min} > \rho^{(t)}_{\min}$. Now we assume $t' > t$. Let $X = [x', x'')$ be a maximal interval of minimum density $\rho'^{(t)}_{\min}$ in $D'^{(t)} = D^{(t)}$. Thus, $\rho'^{(t)}(X) = \rho'^{(t)}_{\min}$. Furthermore, $D'^{(t)}_i = C'_i \cap D'^{(t)} \not\subseteq X$ holds, since otherwise (i.e., if $D'^{(t)}_i \subseteq X$) A'_i to player i would be determined in the t-th recursive call CutCake$(P'^{(t)}, D'^{(t)}, \mathcal{D}'_{P'^{(t)}})$ and $t' = t$ (which contradicts $t' > t$). Thus, if $D^{(t)}_i = C_i \cap D^{(t)} \not\subseteq X$, then $\rho'^{(t)}(X) = \rho^{(t)}(X)$ by $D'^{(t)}_i \not\subseteq X$ and $\rho'^{(t)}_{\min} = \rho'^{(t)}(X) = \rho^{(t)}(X) \ge \rho^{(t)}_{\min}$. If $D^{(t)}_i \subseteq X$, then $\rho'^{(t)}(X) > \rho^{(t)}(X)$ by $D'^{(t)}_i \not\subseteq X$ and $\rho'^{(t)}_{\min} = \rho'^{(t)}(X) > \rho^{(t)}(X) \ge \rho^{(t)}_{\min}$. By the argument above, we have $\rho'^{(t)}_{\min} \ge \rho^{(t)}_{\min}$.

Thus, for each $j \in R_\ell^{(t)} \setminus \{i\}$, we have $len(A'_j) = \rho'^{(t'_j)}_{\min} \geq \rho'^{(t)}_{\min} \geq \rho^{(t)}_{\min}$ and $A'_j \subseteq C_j \cap D'^{(t)} = C_j \cap D^{(t)} \subseteq H_\ell^{(t)}$. By $\sum_{j \in N} A'_j = C$ and $\sum_{j \in R_\ell^{(t)}} A_j = (\cup_{j \in R_\ell^{(t)}} C_j) \cap D^{(t)} = H_\ell^{(t)}$, we have

$$\begin{aligned} V_i(A'_i) &= len(A'_i \cap C_i) = len(A'_i \cap C_i \cap D^{(t)}) \\ &\leq len(H_\ell^{(t)}) - \sum_{j \in R_\ell^{(t)} \setminus \{i\}} len(A'_j \cap C_j \cap D^{(t)}) = len(H_\ell^{(t)}) - \sum_{j \in R_\ell^{(t)} \setminus \{i\}} len(A'_j) \\ &\leq len(H_\ell^{(t)}) - \rho^{(t)}_{\min}(|R_\ell^{(t)}| - 1) = |R_\ell^{(t)}|\rho^{(t)}_{\min} - \rho^{(t)}_{\min}(|R_\ell^{(t)}| - 1) \\ &= \rho^{(t)}_{\min} = V_i(A_i). \end{aligned}$$

Thus, truthfulness of Modified Mechanism of Asano and Umeda (Algorithm 2) is proved.

6 Second Mechanism $\mathcal{M}_2$

In this section, we give the second version $\mathcal{M}_2$ which can be applied to the envy-free and truthful mechanism proposed by Chen, et al. [13] where the valuation function of each player is more general and piecewise uniform. We are given a cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$, and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i) \subseteq C$ of each player $i \in N$ as before. We are also given $(s_i : i \in N)$ such that there is an allocation $A'_N = (A'_i : i \in N)$ to players N with $A'_i \subseteq C_i$ and $s_i = len(A'_i) > 0$ for each $i \in N$ and $\sum_{i \in N} A'_i = C$ (thus $\sum_{i \in N} s_i = 1$). Note that there is no need to have such an allocation $A'_N = (A'_i : i \in N)$ in hand.

Then $\mathcal{M}_2$ is almost the same as $\mathcal{M}_1$ and can be written as follows.

Algorithm 3 Second Mechanism $\mathcal{M}_2$.

Input: Cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$ and solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ with valuation interval $C_i = [\alpha_i, \beta_i)$ of each player $i \in N$ and $\bigcup_{C_i \in \mathcal{C}_N} C_i = C$ and $(s_i : i \in N)$ for players N such that there is an allocation $A'_N = (A'_i : i \in N)$ to players N with $A'_i \subseteq C_i$ and $len(A'_i) = s_i$ for each $i \in N$ and $\sum_{i \in N} A'_i = C$ (thus $\sum_{i \in N} s_i = 1$).

Output: Allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = s_i$ for each $i \in N$ and $\sum_{i \in N} A_i = C$.

sort $\mathcal{C}_N = (C_i : i \in N)$ in a lexicographic order with respect to (β_i, α_i) and assume $C_1 \leq C_2 \leq \cdots \leq C_n$ in this lexicographic order;

set $A_1 = [a_1, b_1) \subseteq C_1$ with length s_1 such that $a_1 = \alpha_1$ and $b_1 = \alpha_1 + s_1$;

for $i = 2$ **to** n **do**

  set $A_i = [a_i, b_i) \setminus \sum_{\ell=1}^{i-1} A_\ell$ with length s_i such that $[a_i, b_i) \subseteq C_i$ and a_i is the leftmost endpoint in $C_i \setminus \sum_{\ell=1}^{i-1} A_\ell$;

Figure 4 shows an example of solid valuation intervals $\mathcal{C}_N = (C_i : i \in N)$ and $(s_i : i \in N)$ with $\sum_{i \in N} s_i = 1$ and an allocation $A_N = (A_i : i \in N)$ obtained by $\mathcal{M}_2$. By an argument similar to one in Proof of Theorem 1 we have the following theorem.

▶ **Theorem 3.** *$\mathcal{M}_2$ correctly finds an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = s_i$ for each $i \in N$ and $\sum_{i \in N} A_i = C$ in $O(n \log n)$ time. Furthermore, the number of cuts made by $\mathcal{M}_2$ on cake C is at most $2n - 2$.*

Proof. Since the time complexity $O(n \log n)$ and the number of cuts at most $2n - 2$ can be obtained by the same argument as in Proof of Theorem 1, we only prove that $\mathcal{M}_2$ correctly finds an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$, $len(A_i) = s_i$ and $\sum_{i \in N} A_i = C$.

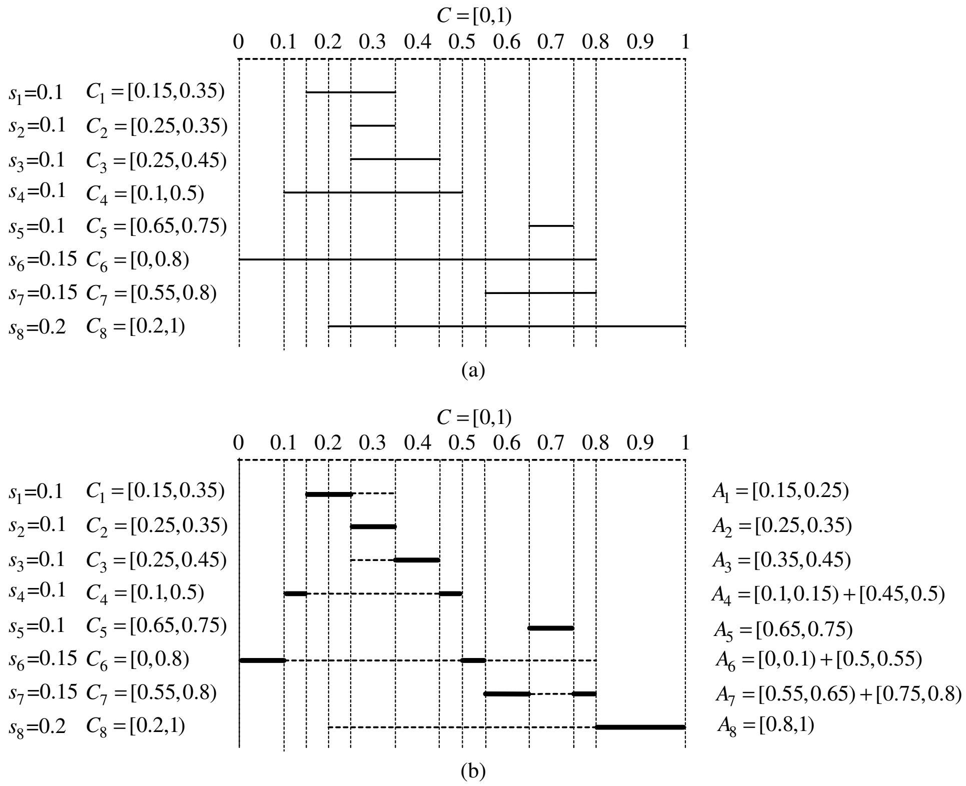

Figure 4 (a) Example of $\mathcal{C}_N = (C_i : i \in N)$ and $(s_i : i \in N)$ with $\sum_{i\in N} s_i = 1$. (b) Allocation $A_N = (A_i : i \in N)$ obtained by $\mathcal{M}_2$.

Suppose contrarily that $\mathcal{M}_2$ could not set $A_i \subseteq C_i$ with length s_i for some $i \in N$. Let j be the minimum among such i s and let $J = \{1, 2, \ldots, j\}$. Of course, $j > 1$, since we assumed that there is an allocation $A'_N = (A'_i : i \in N)$ to players N with $A'_i \subseteq C_i$ and $len(A'_i) = s_i$ for each $i \in N$ and $\sum_{i\in N} A'_i = C$ (thus $C_1 = [\alpha_1, \beta_1)$ is of length at least s_1 and $A_1 = [a_1, b_1) = [\alpha_1, \alpha_1 + s_1) \subseteq C_1$). Now we consider valuation intervals $\mathcal{C}_J = (C_i : i \in J)$. Note that each $C_i = [\alpha_i, \beta_i) \in \mathcal{C}_J$ satisfies $\beta_i \le \beta_j$, since $\mathcal{C}_N = (C_i : i \in N)$ was sorted in the lexicographic order with respect to (β_i, α_i). Thus, $\mathcal{M}_2$ could set $A_i = [a_i, b_i) \setminus \sum_{\ell=1}^{i-1} A_\ell \subseteq C_i = [\alpha_i, \beta_i)$ with length s_i for each $i \in J \setminus \{j\}$ but could not set $A_j = [a_j, b_j) \setminus \sum_{\ell=1}^{j-1} A_\ell \subseteq C_j = [\alpha_j, \beta_j)$ with length s_j. This implies that $C_j \setminus \sum_{\ell=1}^{j-1} A_\ell$ is of length $s'_j < s_j$. Let

$$A''_i = A_i \quad (i \in J \setminus \{j\}), \qquad A''_j = C_j \setminus \sum_{\ell=1}^{j-1} A''_\ell = [a_j, \beta_j) \setminus \sum_{\ell=1}^{j-1} A_\ell.$$

Thus, $\sum_{i\in J} A''_i$ of allocation $(A''_i : i \in J)$ consists of several maximal contiguous intervals. Let $I = [a, b)$ be the rightmost maximal contiguous interval among the maximal contiguous intervals in $\sum_{i\in J} A''_i$ (Figure 5). Thus, $b = \beta_j$. Define $K \subseteq J$ by

$$K = \{j\} \cup \{i \in J \mid A''_i \cap I \neq \emptyset\}.$$

Now we consider valuation intervals $\mathcal{C}_K = (C_i : i \in K)$. Then each $C_i \in \mathcal{C}_K$ is contained in I, which can be obtained as follows.

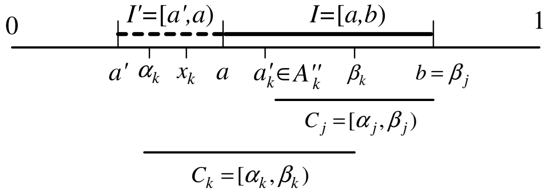

Figure 5 Illustration of $I = [a, b)$ and $I' = [a', a)$.

Of course, $C_j = [\alpha_j, \beta_j)$ is contained in I. Actually, since $C_j \setminus \sum_{\ell=1}^{j-1} A''_\ell = [a_j, \beta_j) \setminus \sum_{\ell=1}^{j-1} A_\ell$ is of length $s'_j < s_j$ and $A''_j = C_j \setminus \sum_{\ell=1}^{j-1} A''_\ell = [a_j, \beta_j) \setminus \sum_{\ell=1}^{j-1} A_\ell$, we have: if $A''_j = \emptyset$ then $C_j \subseteq \sum_{\ell=1}^{j-1} A''_\ell$ and a single contiguous interval C_j is contained in the rightmost maximal contiguous interval I in $\sum_{\ell=1}^{j} A''_\ell = \sum_{\ell=1}^{j-1} A''_\ell$ (i.e., $C_j \subseteq I$); and otherwise (i.e., if $A''_j \neq \emptyset$), $C_j \subseteq A''_j \cup \sum_{\ell=1}^{j-1} A''_\ell = \sum_{\ell=1}^{j} A''_\ell$ and a single contiguous interval C_j is contained in the rightmost maximal contiguous interval I in $\sum_{\ell=1}^{j} A''_\ell$.

Now suppose that there were $i \in K \setminus \{j\}$ such that $C_i \in \mathcal{C}_K$ is not contained in I. Thus, $I = [a, b)$ is a proper subinterval of $[0, b) = [0, \beta_j)$ (i.e., $a > 0$) and $C_i = [\alpha_i, \beta_i) \in \mathcal{C}_K \setminus \{C_j\}$ contains a point x in $[0, b) \setminus I = [0, a)$. Let $k \in K \setminus \{j\}$ be the minimum among such is and let x_k be a point of $C_k = [\alpha_k, \beta_k) \in \mathcal{C}_K$ contained in $[0, a) = [0, b) \setminus I$. Note that $C_k \cap I \supseteq A''_k \cap I \neq \emptyset$ since $k \in K \setminus \{j\} \subseteq J \setminus \{j\}$. Thus, $\beta_k \leq \beta_j$ and $\alpha_k \leq x_k < a \leq a'_k < \beta_k$ for some $a'_k \in A''_k \cap I \neq \emptyset$. Furthermore, since we chose $I = [a, b) \neq [0, b)$ as the rightmost maximal contiguous interval among the maximal contiguous intervals in $\sum_{i \in J} A''_i$, we have $\sum_{i \in J} A''_i \neq [0, b) = [0, \beta_j)$. Let $I' = [a', a)$ be the rightmost maximal contiguous interval in $[0, b) \setminus \sum_{i \in J} A''_i$ (Figure 5). Since $C_k = [\alpha_k, \beta_k)$ is a contiguous interval and satisfies $\alpha_k \leq x_k < a \leq a'_k < \beta_k$, we can choose x_k with $x_k \in I' \cap C_k \neq \emptyset$. Thus, $x_k \notin A''_k$ by $I' \cap A''_k \subseteq I' \cap \sum_{i \in J} A''_i = \emptyset$. Then, however, $\mathcal{M}_2$ would have tried to include x_k into A''_k rather than $a'_k \in A''_k \cap I \neq \emptyset$, because $\mathcal{M}_2$ sets $A''_k = A_k = [a_k, b_k) \setminus \sum_{\ell=1}^{k-1} A''_\ell \subseteq C_k$ with length s_k such that a_k is the leftmost endpoint in $C_k \setminus \sum_{\ell=1}^{k-1} A''_\ell$. This is a contradiction.

Thus, each $C_i \in \mathcal{C}_K$ is contained in I and $\bigcup_{i \in K} C_i \subseteq I$. By the argument above, we have

$$\bigcup_{i \in K} C_i = I = \sum_{i \in K} A''_i,$$

since $A''_h \cap I = \emptyset$ for $h \in J \setminus K$ and $I = \sum_{i \in J} A''_i \cap I = \sum_{i \in K} A''_i \cap I \subseteq \sum_{i \in K} A''_i \subseteq \sum_{i \in K} C_i$ by the definitions of I and K and $A''_i \subseteq C_i$ for each $i \in K$. Thus,

$$\sum_{i \in K} len(A''_i) = s'_j + \sum_{i \in K \setminus \{j\}} s_i = len(I) = b - a < s_j + \sum_{i \in K \setminus \{j\}} s_i$$

since $s'_j < s_j$. However, this is a contradiction, since we assumed that there is an allocation $A'_N = (A'_i : i \in N)$ to players N with $A'_i \subseteq C_i$ and $len(A'_i) = s_i$ for each $i \in N$ and $\sum_{i \in N} A'_i = C$, and thus

$$s_j + \sum_{i \in K \setminus \{j\}} s_i = \sum_{i \in K} len(A'_i) \leq len(\bigcup_{i \in K} C_i) = len(I) = b - a.$$

Thus, $\mathcal{M}_2$ correctly finds an allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$, $len(A_i) = s_i$ and $\sum_{i \in N} A_i = C$. ◀

7 Application to Mechanism of Chen et al. [13]

By Theorem 3, in order to obtain an envy-free and truthful allocation $A_N = (A_i : i \in N)$ with $A_i \subseteq C_i$ and $len(A_i) = s_i$ for each $i \in N$ and $\sum_{i\in N} A_i = C$, we only need $(s_i : i \in N)$ such that there is an envy-free and truthful allocation $A'_N = (A'_i : i \in N)$ to players N with $A'_i \subseteq C_i$ and $len(A'_i) = s_i$ for each $i \in N$ and $\sum_{i\in N} A'_i = C$. Thus, Theorem 3 can be applied to the mechanism of Chen, et al. [13] where the valuation function v_i of each player $i \in N$ is more general and piecewise uniform: Given a cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$ and solid piecewise uniform valuation functions $(v_i : i \in N)$ such that $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of each valuation function v_i consists of $m_i \geq 1$ maximal contiguous intervals $C_{i_1}, \ldots, C_{i_{m_i}}$ in C and $\bigcup_{i\in N} D(v_i) = C$.

The mechanism of Chen, et al. [13] finds an envy-free and truthful allocation $A'_N = (A'_i : i \in N)$ such that $\sum_{i\in N} A'_i = C$ and $A'_i = \sum_{j=1}^{m_i} A'_{i_j}$ with $A'_{i_j} \subseteq C_{i_j}$ for each $i \in N$ and for each $j = 1, 2, \ldots, m_i$. Thus, we can set $s_{i_j} = len(A'_{i_j})$ and apply Theorem 3 to obtain an envy-free and truthful allocation $A_N = (A_i : i \in N)$ such that $A_i = \sum_{j=1}^{m_i} A_{i_j}$ for each $i \in N$ with $A_{i_j} \subseteq C_{i_j}$ and $len(A_{i_j}) = s_{i_j}$ for each $j = 1, 2, \ldots, m_i$ with at most $2(\sum_{i\in N} m_i) - 2$ cuts. Note that, we can delete all C_{i_j} if $s_{i_j} = len(A'_{i_j}) = 0$, and thus, we can assume $s_{i_j} = len(A'_{i_j}) > 0$ for each $i \in N$ and for each $j = 1, 2, \ldots, m_i$.

In summary, we have the following corollary.

▶ **Corollary 4.** *Suppose that we are given $(s_{i_j} : i \in N, j = 1, 2, \ldots, m_i)$ such that there is an envy-free and truthful allocation $A'_N = (A'_i : i \in N)$ to players N satisfying $\sum_{i\in N} A'_i = C$ and $A'_i = \sum_{j=1}^{m_i} A'_{i_j}$ with $A'_{i_j} \subseteq C_{i_j}$ and $len(A'_{i_j}) = s_{i_j} > 0$ for each $i \in N$ and for each $j = 1, 2, \ldots, m_i$ for the cake-cutting problem with cake $C = [0, 1)$, n players $N = \{1, 2, \ldots, n\}$ and solid piecewise uniform valuation functions $(v_i : i \in N)$ such that $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of each piecewise uniform valuation function v_i consists of $m_i \geq 1$ maximal contiguous intervals $C_{i_1}, \ldots, C_{i_{m_i}}$ in C and $\bigcup_{i\in N} D(v_i) = C$ (such $A'_N = (A'_i : i \in N)$ to players N can be obtained, for example, by Mechanism of Chen, et al. [13]). Then, Second Mechanism $\mathcal{M}_2$ (Algorithm 3) correctly finds an envy-free and truthful allocation $A_N = (A_i : i \in N)$ with $\sum_{i\in N} A_i = C$ and $A_i = \sum_{j=1}^{m_i} A_{i_j}$ with $A_{i_j} \subseteq C_{i_j}$ and $len(A_{i_j}) = s_{i_j}$ for each $i \in N$ and for each $j = 1, 2, \ldots, m_i$ in $O(\sum_{i\in N} m_i \log \sum_{i\in N} m_i)$ time. Furthermore, the number of cuts made by $\mathcal{M}_2$ on cake C is at most $2(\sum_{i\in N} m_i) - 2$. Thus, Mechanism of Chen, et al. [13] can be implemented to make at most $2(\sum_{i\in N} m_i) - 2$ cuts on cake C.*

8 Concluding Remarks

We gave a much simpler envy-free and truthful mechanism with a small number of cuts for the cake-cutting problem posed in [2, 25]. Furthermore, we showed that this approach can be applied to the envy-free and truthful mechanism proposed by Chen, et al. [13] for the more general cake-cutting problem where the valuation function of each player is piecewise uniform. Thus, we can make their envy-free and truthful mechanism use $2\sum_{i\in N} m_i - 2$ cuts and settle the problem posed by [2, 25], where m_i is the number of maximal contiguous intervals in $D(v_i) = \{x \in C \mid v_i(x) > 0\}$ of each player i's piecewise uniform valuation v_i.

If we require the piecewise uniform valuation v_i of each player i to be a single contiguous interval C_i in cake C, then Modified Mechanism of Asano and Umeda can be implemented to run in $O(n^2 \log n)$ time based on parametric flows on the network arising from valuation intervals C_i [3] (Parametric flows and parametric searching have been studied by many researchers [1, 17, 33]). We expect this would lead to a faster envy-free and truthful mechanism for the general piecewise uniform valuations.

References

1 Hassene Aissi, S. Thomas McCormick, and Maurice Queyranne. Faster algorithms for next breakpoint and max value for parametric global minimum cuts. In *21st International Conference on Integer Programming and Combinatorial Optimization*, pages 27–39, 2020.

2 Reza Alijani, Majid Farhadi, Mohammad Ghodsi, Masoud Seddighin, and Ahmad S. Tajik. Envy-free mechanisms with minimum number of cuts. In *31st AAAI Conference on Artificial Intelligence, 2017*, pages 312–318, 2017.

3 Takao Asano. Envy-free and truthful cake-cuttings based on parametric flows. In *RIMS Kôkyûroku 2182, Kyoto University, April, 2021*, pages 1–39, 2021.

4 Takao Asano and Hiroyuki Umeda. Cake cutting: An envy-free and truthful mechanism with a small number of cuts. In *31st International Symposium on Algorithms and Computation*, pages 15.1–15.16, 2020.

5 Takao Asano and Hiroyuki Umeda. An envy-free and truthful mechanism for the cake-cutting problem. In *RIMS Kôkyûroku 2154, Kyoto University, April, 2020*, pages 54–91, 2020.

6 Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for any number of agents. In *57th Annual Symposium on Foundations of Computer Science, 2016*, pages 416–427, 2016.

7 Haris Aziz and Chun Ye. Cake cutting algorithms for piecewise constant and piecewise uniform valuations. In *10th International Conference on Web and Internet Economics, 2014*, pages 1–14, 2014.

8 Xiaohui Bei, Ning Chen, Xia Hua, Biaoshuai Tao, and Endong Yang. Optimal proportional cake cutting with connected pieces. In *26th AAAI Conference on Artificial Intelligence, 2012*, pages 1263–1269, 2012.

9 Xiaohui Bei, Ning Chen, Guangda Huzhang, Biaoshuai Tao, and Jiajun Wu. Cake cutting: envy and truth. In *26th International Joint Conference on Artificial Intelligence*, pages 3625–3631, 2017.

10 Steven J. Brams, Michal Feldman, John K. Lai, Jamie Morgenstern, and Ariel D. Procaccia. On maxsum fair cake divisions. In *26th AAAI Conference on Artificial Intelligence, 2012*, pages 1285–1291, 2012.

11 Steven J. Brams and Alan D. Taylor. *Fair Division: From Cake Cutting to Dispute Resolution.* Cambridge University Press, 1996.

12 Ioannis Caragiannis, John K. Lai, and Ariel D. Procaccia. Towards more expressive cake cutting. In *22nd International Joint Conference on Artificial Intelligence, 2011*, pages 127–132, 2011.

13 Yiling Chen, John K. Lai, David C. Parkes, and Ariel D. Procaccia. Truth, justice, and cake cutting. *Games and Economic Behavior*, 77:284–297, 2013.

14 Xiaotie Deng, Qi Qi, and Amin Saberi. Algorithmic solutions for envy-free cake cutting. *Operations Research*, 60:1461–1476, 2012.

15 Jeff Edmonds and Kirk Pruhs. Balanced allocations of cake. In *47th Annual Symposium on Foundations of Computer Science, 2006*, pages 623–634, 2006.

16 Jeff Edmonds and Kirk Pruhs. Cake cutting really is not a piece of cake. In *17th ACM-SIAM Symposium on Discrete Algorithms*, pages 271–278, 2006.

17 Giorgio Gallo, Michael D. Grigoriadis, and Robert E. Tarjan. A fast parametric maximum flow algorithm and applications. *SIAM Journal on Computing*, 18:30–55, 1989.

18 Paul W. Goldberg, Alexandros Hollender, and Warut Suksompong. Contiguous cake cutting: hardness results and approximation algorithms. In *34th AAAI Conference on Artificial Intelligence, 2020*, pages 1990–1997, 2020.

19 David Kurokawa, John K. Lai, and Ariel D. Procaccia. How to cut a cake before the party ends. In *27th AAAI Conference on Artificial Intelligence, 2013*, pages 555–561, 2013.

20 Avishay Maya and Noam Nisan. Incentive compatible two player cake cutting. In *8th International Conference on Web and Internet Economics, 2012*, pages 170–183, 2012.

21 Ariel D. Procaccia. Thou shalt covet thy neighbor's cake. In *21st International Joint Conference on Artificial Intelligence*, pages 239–244, 2009.
22 Ariel D. Procaccia. Cake cutting: not just child's play. *Commun. ACM*, 56:78–87, 2013.
23 Ariel D. Procaccia. Cake cutting algorithms. *Handbook of Computational Social Choice (F. Brandt , V. Conitzer, U. Endriss, J. Lang, and A.D. Procaccia, Eds., Cambridge University Press)*, Chapter 13:313–329, 2016.
24 Jack M. Robertson and William A. Webb. *Cake Cutting Algorithms: Be Fair If you Can.* A.K. Peters, 1998.
25 Masoud Seddighin, Majid Farhadi, Mohammad Ghodsi, Reza Alijani, and Ahmad S. Tajik. Expand the shares together: envy-free mechanisms with a small number of cuts. *Algorithmica*, 81:1728–1755, 2019.
26 Katerina Sherstyuk. How to gerrymander: A formal analysis. *Public Choice*, 95:27–49, 1998.
27 Hugo Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.
28 Hugo Steinhaus. Sur la division pragmatique. *Econometrica (suppliment)*, 17:315–319, 1949.
29 Walter Stromquist. How to cut a cake fairly. *American Mathematical Monthly*, 87:640–644, 1980.
30 Walter Stromquist. Envy-free cake divisions cannot be found by finite protocols. *The Electronic Journal of Combinatorics*, 15:#R11 (pp.1–10), 2008.
31 Francis E. Su. Rental harmony: Sperner's lemma in fair division. *American Mathematical Monthly*, 106:930–942, 1999.
32 William Thomson. Children crying at birthday parties. why? *Economic Theory*, 31:501–521, 2007.
33 Takeshi Tokuyama. Minimax parametric optimization problems and multi-dimensional parametric searching. In *33rd ACM Symposium on Theory of Computing*, pages 75–83, 2001.
34 David P. Williamson. *Network Flow Algorithms.* Cambridge University Press, 2019.