

ON THE NASH PRODUCT SHARE CRITERION FOR FAIR ALLOCATION PROBLEMS

Trung Thanh Nguyen, Dang Nguyen Le

Hai Phong University, Hai Phong, Vietnam

ttnghuyen.cs@gmail.com, ldnguyenth1976@gmail.com

ABSTRACT: Allocating indivisible items between agents in a fair manner is a fundamental problem that has attracted a lot of interests in the last decades due to the wide range of its applications. There are several common criteria for defining what a fair allocation is, including: max-min share, proportional share, min-max share, envy-freeness, CEEI. In this paper, we introduce a new notion of fairness, called Nash-product share, which can be determined through computing an allocation of maximum Nash-product welfare, assuming that all agents have the same additive utilities. An allocation satisfies this fairness criterion is called a Nash-product fair allocation. We first show that computing the Nash-product share of every agent is NP-hard, even with two agents only. In addition, we prove that the problem of testing the existence of a Nash-product fair allocation is an NP-hard problem when the number of agents is part of the input. Finally, since a Nash-product fair allocation does not always exist, we investigate the problem that, given a problem instance, asks for the largest value c for which there is an allocation such that every agent receives a subset of items of values of at least c times her Nash-product share. For the case where the number of agents is constant, we present a polynomial-time approximation scheme (PTAS).

Keywords: Fair allocation, max-min share, proportional share, Nash-product share, approximation algorithm, complexity.

I. GIỚI THIỆU

Fair allocation is a fundamental problem which is of interest in both computer science and economics due to the wide range of its applications (see a book chapter by Bouveret [8] and Lang and Rothe [15]). This problem concerns with allocating a finite set of goods (that may also be called items or objects) among a group of agents having different preferences over the subsets of goods. We will assume that the preferences of agents are presented by additive (or linear) utility functions. Our goal is to find an allocation that satisfies a certain notion of fairness. Among others, *max-min share*, *proportional share*, and *envy-freeness* are the three notions that have been studied intensively in the literature. The max-min share of an agent is defined as the bundle that the agent can guarantee for herself when partitioning the items into bundles but choosing last. In a proportionally fair allocation between n agents, each agent receives a bundle of value at least $1/n$ of the whole. An allocation is said to be envy-free if no agent wants to exchange her bundle of goods with that of any other agent. These three fairness criteria have attractive theoretical and practical properties (see the references cited above and the references therein). In [7], it was shown that envy-freeness is the strongest among the above fairness notions and implies proportional fairness, which implies max-min fairness—the weakest one. On the other hand, as argued in the literature (see, for example, [14, 16, 17]), an allocation satisfying any of the these fairness notions is not guaranteed to exist in general. Furthermore, checking the existence of an envy-free (proportional fair) allocation is NP-complete (see [7]) even in the simplest setting with two agents having the same additive utility functions. However, it is still not known yet if the same result applied to the case of max-min fair allocation. We would like to emphasize that the idea of applying fairness criteria, which is consider in this paper, is orthogonal to the approach of maximizing egalitarian welfare (i.e., the utility of the worst-off agent), and Nash-product welfare (i.e., the product of individual agents' utilities) which has been intensively studied in the literature.

In this paper we introduce a new notion of fairness called Nash-product share, and study fair allocation problems with respect to this notion. Roughly speaking, a Nash-product share of an agent is the bundle of goods that she receives in an allocation of maximum Nash-product (the Nash-product of an allocation is computed as the n -th root of the product of individual agent utilities), assuming that all agents have the same utility functions. The motivation of the study of the Nash-product share is two folds. Firstly, the fair share typically depends on how to allocate fairly items among identical agents. Budish [9] takes this idea into account and introduce the notion of the max-min share, which relies on the determining of an allocation of maximum egalitarian welfare. Since maximizing the Nash-product welfare is also considered to be a fair way for allocating item (see, e.g., [3, 4, 5, 6, 18, 19]), it is very natural to define the notion of Nash-product share. Secondly, although checking the existence of proportionally fair allocation is NP-complete, we do not know yet if checking the existence of max-min fair allocation is also NP-complete. While this question remains open, it is of great interest to examine if there is a fairness notion that lies between the proportional share and the max-min share in the scale of Bouveret and Lemaitre [7] such that checking the existence of an allocation satisfying this fairness criterion is NP-complete.

Our contribution: We first show that the Nash-product share is stronger than the max-min share but weaker than the proportional share. Furthermore, we prove that computing the Nash-product share is NP-hard even when there are

only two agents with additive utility functions, and give a NP-hardness result for the problem of checking the existence of Nash-product fair allocations. Finally, we consider the problem which determines, given a problem instance, the largest value c for which there exists an allocation in which every agent gets a bundle of goods of utility of at least c times her Nash-product share.

A. Related work

The max-min share was considered for the first time by Budish [9] as a fair criterion for the allocation of courses to students at the Harvard university. Since then, there has been a sequence of papers investigating this criterion [1, 2, 7, 20, 14, 17]. Bouveret and Lemaitre [7] prove that computing max-min shares is NP-hard but leave open the question of whether or not the problem of determining the existence of max-min fair allocations is NP-hard as well. For two agents, it is well-known that there is always a max-min fair allocation which can be found via a simple *cut-and-choose* protocol. However, this is not the case for any problem instance with more than two agents [17]. Hence, many attempts have focused on finding a small constant value c for which there always exists an allocation such that every agent gets a bundle of utility of at least a fraction c of her max-min share. Procaccia and Wang [17] show that $c = 2/3$ and give an (exponential) algorithm for computing an allocation corresponding to this value c . By redesigning parts of the algorithm of Procaccia and Wang [17], Amanatidis et al. [1] show that a $2/3$ -max-min fair allocation can be found in polynomial time. For a small number of agents, the problem is understood better. For three agents, it is shown by Amanatidis et al. [1] that $c = 7/8$. It still remains open whether the lower bound of $7/8$ can be further improved for a higher constant number of agents. Very recently, Aziz et al. [2] have given a polynomial-time approximation scheme for computing an optimal max-min fair allocation, assuming that the number of agents is fixed.

For proportionally fair allocations, the problem of checking the existence of such allocations is NP-complete, even with two identical agents (Bouveret and Lemaitre [7]). A first lower bound for c , which is essentially a function depending both on the number of agents and on the maximum value of any agent for a single good, is provided by Hill [13]. Markakis and Psomas [16] present a polynomial-time algorithm for constructing an allocation with respect to this lower bound. An improvement of this bound is then proposed by Gourves et al. [11, 12].

Since we will study the Nash-product share criterion in this paper, we would like to give a brief overview of results on the problem of computing maximum Nash-product allocations. This problem is shown to be NP-hard by Nguyen et al. [19]. In addition, it has been recently proved that there is a constant factor α for which one cannot have an approximation algorithm of a factor better than α [21]. A first approximation algorithm for the problem with an approximation factor of $O(m)$ is given in [18], where m denotes the number of goods. This result was improved by Cole and Gkatzelis in [6] where the authors obtained an approximation algorithm with a constant factor ≈ 2.89 . Very recently, Cole et al. [4] have achieved a further improved factor of 2 for the problem. For the special case when all agents have the same utility functions, there is a polynomial time approximation scheme (PTAS) for the problem [18].

II. PROBLEM MODELS

In a fair allocation we are given a finite set $A = 1, \dots, n$ of agents and a finite set $O = o_1, \dots, o_m$ of indivisible and nonshareable goods (or items or objects). Each agent i has an individual preference over the subsets of goods, which can be accessed through a *utility function*: $u_i : 2^O \rightarrow \mathbb{Q}^+$ —a mapping from the set of all subsets of O to the set of nonnegative rational numbers. We assume that every utility function u_i is additive (or linear), that is, for any subset $S \subseteq O$, $u_i(S) = \sum_{o \in S} u_i(o)$. Additive functions are perhaps the most simple and compact representation of utility functions and have been widely used in many fair allocation problems. The nonnegative value $u_i(o)$ is called the utility (or the value) of item o for agent i . Let $U = u_1, \dots, u_n$, we call (A, O, U) an *allocation setting*. An *allocation* of goods between agents is defined as a partition $\pi = (\pi_1, \dots, \pi_n)$ of O into n disjoint subsets, where π_i is the subsets of items that is assigned to agent i . We can also call π_i agent i 's share. For any allocation π , it must hold that:

- $\pi_i \cap \pi_j = \emptyset$, for $i \neq j$, (i.e., no item is given to multiple agents),
- $\bigcup_{i=1}^n \pi_i = O$, (i.e., every item is given to some agent).

We denote by Π the set of all such allocations π .

Given an allocation setting (A, O, U) , we want to find an allocation $\pi = (\pi_1, \dots, \pi_n)$ that is *fair*. There are several common criteria for fairness which have been studied intensively in the literature, including: max-min share, proportional share, min-max share, envy-free, and CEEI. For a detailed description of these fairness criteria, we refer to a recent paper by Bouveret and Lemaitre [7]. In this paper we introduce a new criterion, called *Nash-product share*, which are formally defined below. We also define the *max-min share* and *proportional share* since we will make a comparison between these notions of fairness in the next section.

Definition. Given an allocation setting (A, O, U) , define

- agent i 's max-min share as

$$\text{MMS}_i = \max_{\pi \in \Pi} \min_{j=1}^n u_i(\pi_j)$$

- agent i 's proportional share as

$$\text{PS}_i = \frac{u_i(O)}{n} = \frac{\sum_{j=1}^m u_i(o_j)}{n}$$

- agent i 's Nash-product share as

$$\text{NPS}_i = \max_{\pi \in \Pi} \left(\prod_{j=1}^n u_i(\pi_j) \right)^{\frac{1}{n}}$$

An allocation $\pi = (\pi_1, \dots, \pi_n)$ is

- a *max-min fair allocation* if $u_i(\pi_i) \geq \text{MMS}_i$ for all $i \in A$
- a *proportional fair allocation* if $u_i(\pi_i) \geq \text{PS}_i$ for all $i \in A$
- a *Nash-product fair allocation* if $u_i(\pi_i) \geq \text{NPS}_i$ for all $i \in A$

Let us consider the relationship our notions have to one another. Proportional fairness is the strongest among the above fairness notions and implies Nash product fairness, which implies max-min fairness.

Proposition 1. Let (A, O, U) be an allocation setting and $i \in A$. We have $\text{PS}_i \geq \text{NPS}_i \geq \text{MMS}_i$, and, furthermore, all these fairness notions are distinct.

Proof. The chain of inequalities follows from the inequalities between arithmetic and geometric. To see that the max-min fairness does not imply the Nash-product fairness, consider two agents with identical utility functions over four items. The items are valued 1, 2, 3, and 5, respectively. The max-min share of both agents is 5 (with allocations $\{(1,5), (2,3)\}$ and $\{(5), (1, 2,3)\}$), whereas $\text{NPS}_i = \sqrt[3]{30} > 5$. But then no Nash-product fair allocation exists. That the other fairness notions are also distinct is again easy to show using the definitions of arithmetic and geometric.

In this paper we study the following problems.

NPS-Comp	
Given:	<i>An allocation setting (A, O, U), an agent i, and a number $\ell > 0$.</i>
Question:	<i>Do we have $\text{NPS}_i \geq \ell$?</i>
NPS-Exist	
Given:	<i>An allocation setting (A, O, U).</i>
Question:	<i>Is there a Nash-product fair allocation?</i>

III. HARDNESS RESULT AND APPROXIMATION ALGORITHM

A. Hardness result

Theorem 1. **NPS-Comp** is NP-complete.

Proof. One can easily see that **NPS-Comp** belongs to NP. We are now showing the NP-hardness via a reduction from the **PARTITION** problem: Given a set of positive numbers $C = a_1, \dots, a_m$ such that $\sum_{j=1}^m a_j = 2K$ for some positive number K , the question is that whether there is a partition (C_1, C_2) of C such that $\sum_{j \in C_1} a_j = \sum_{j \in C_2} a_j = K$? Let C be an instance of **PARTITION**, we construct an instance I of **NPS-Comp** as follows. There are two agents and m items o_1, \dots, o_m , both agents have the same additive utility functions defined by $u_1(o_j) = u_2(o_j) = a_j$ for all $j = 1, \dots, m$. Now we consider agent 1 and let $\ell = K$. The Nash-product share of agent 1 can be computed as

$$\text{NPS}_1 = \max_{(C_1, C_2)} \sqrt{\sum_{a_j \in C_1} a_j \cdot \sum_{a_j \in C_2} a_j},$$

which is taken over all possible partition (C_1, C_2) of C . One can see easily that $NPS_I = K$ if and only if $\sum_{j \in C_1} a_j = \sum_{j \in C_2} a_j = K$. In other words, I is a yes-instance of **NPS-Comp** if and only if C is a yes-instance of **PARTITION**, and this implies the NP-hardness of the first problem. ■

Theorem 2. **NPS-Exist** is NP-hard.

Proof. To prove NP-hardness part, we rely on a reduction from the NP-complete problem **3-Dimensional Matching (3DM)**, which is defined as follows:

3-Dimensional Matching

Given: Three disjoint sets $X = x_1, \dots, x_q$, $Y = y_1, \dots, y_q$, $Z = z_1, \dots, z_q$ and a collection F of 3-element subsets of $X \times Y \times Z$ (assuming, without loss of generality, that $p > q$).

Question: Does F contain a matching, i.e., a subcollection $F' \subset F$ such that every element of $X \times Y \times Z$ is covered by exactly one member of F' ?

Here we assume that q is even and that $p = 3q/2$. Note that **3DM** is still a NP-complete problem even when restricted to this setting. Let (X, Y, Z, F) be an instance of **3DM**. We will construct a corresponding instance I of **NPS-Exist** with p agents and $p + 2q$ items. Agent i corresponds to the element $F_i \in F$, for $i \in [p]$, where $[p]$ is a shorthand for $1, \dots, p$. There are $3q$ items corresponding to $3q$ elements of $X \times Y \times Z$. Additionally, we create $q/2$ dummy items $D = d_1, \dots, d_{q/2}$.

We now define the utility function of agent i , for $i \in [p]$. For a bundle $F_i = (x, y, z) \in F$, the utilities of items x, y , and z are, respectively, $4q, 2q$ and q . For the remaining $3q - 3$ items, we fix their ordering in an arbitrary way, and then assign a utility of 1 to each of the first $q - 3$ items, and a utility of zero to each of the rest. All the dummy items have the same utility B , where B is a positive integer such that $B > 8q - 3$. The exact value of B will be determined later. One can see that our reduction constructed above can be computed in polynomial time.

Let us compute the Nash product share for every agent $i \in [p]$. This amounts to finding a partition of the set of the items into p disjoint subsets such that the product of the values of the subsets is maximized. Let $\pi = (\pi_1, \dots, \pi_p)$ be such an optimal partition. We first notice that if there is some agent j who received a dummy item d by the allocation π , then d is the only item assigned to her. Indeed, since there are only $q/2$ dummy items, there are at most $q/2$ agents having dummy items in π . We call those agents “big” agents, and the other ones “small” agents. If a big agent received a dummy item d and another item t , then by transferring t to any small agent, the Nash product (i.e., the product of agents’ utilities) will be improved. The same argument can be applied to show that if an agent received an item from the first $q/2 + 3$ largest items in the list, then that is the only one she had in π . Hence, there are $q - 3$ worst-off agents in π , each receiving an item of value 1. The Nash-product share of every agent is

$$NPS = B^{q/2} \cdot 4q \cdot 2q \cdot q^{1/p} = 8q^{2/q} \cdot B^{1/3},$$

since $p = 3q/2$.

We claim that (X, Y, Z, F) is a yes-instance of **3DM** if and only if there is a Nash-product fair allocation for the constructed instance I .

(\Rightarrow) Suppose that (X, Y, Z, F) is a yes-instance of **3DM** and let $F' \subset F$ be a subset that forms an exact cover of $X \times Y \times Z$. Let us consider an allocation π' of I in which agent i is assigned the bundle F_i if $F_i \in F'$, and one dummy item from D , otherwise. It is clear that all the items are allocated and there is no item shared by two or more agents. Therefore, each agent received a bundle whose value is at least $4q + 2q + q = 7q$.

(\Leftarrow) Suppose that (X, Y, Z, F) is a no-instance of **3DM**. Consider an optimal allocation π of I . As argued above, there is a subset T of q agents who do not receive any dummy item from D . Let i be a worst-off agent in T . Note that agent i cannot get the bundle F_i since otherwise, any other agent $j \in T$ also receives her preferred bundle F_j , and this implies that $C_j | j \in T$ forms an exact cover of $X \times Y \times Z$. The best assignment agent i can get is a bundle containing two most preferred items from F_i and one item of value 1. Hence, the utility of her bundle is at most $4q + 2q + 1 = 6q + 1$.

Now, by choosing an positive integer B such that

$$6q+1 < (8q)^{2/q} \cdot B^{1/3} < 7q,$$

or, equivalently,

$$\frac{6q+1^3}{(8q)^{6/q}} < B < \frac{7q^3}{(8q)^{6/q}},$$

it follows that I has a Nash-product fair allocation if and only if (X, Y, Z, F) is a yes-instance of **3DM**. It remains to show that the number B chosen above satisfies $B > 8q - 3$. It suffices to prove that

$$8q - 3 < \frac{6q+1^3}{(8q)^{6/q}},$$

or, equivalently,

$$(8q - 3) \cdot (8q)^{6/q} < (6q + 1)^3.$$

It is not hard to see that the function $f(q) = (6q+1)^3 - (8q-3) \cdot (8q)^{6/q}$ is increasing on the interval $[3, +\infty)$. Hence, it follows that $f(q) > f(3) > 0$ for all $q > 3$. The proof is complete. ■

B. Approximation Algorithm

The result above implies that the problem of computing a Nash-product fair allocation (if exists one) is also NP-hard. This means there is no polynomial-time algorithm for solving this problem, unless $P = NP$. Alternatively, one can investigate the similar question studied by Nguyen et al. [20], but for the case of Nash product fair allocation: Given a problem instance, compute an α -Nash product fair allocation for the maximum possible value of α , that is, an allocation such that every agent i gets a bundle of value of at least $\alpha \cdot \text{NPS}_i$.

Let $I = (A, O, U)$ be an instance problem with additive utility functions. We assume that, w.l.o.g, the Nash product share of every agent is positive and that there is at least one allocation in which the utility of every agent is positive. This ensures that the value of α is nonzero and bounded. Concretely, denote by $c_N(I)$ the largest value such that there exists an allocation in which the utility of every agent is at least a fraction $c_N(I)$ of her Nash-product share. For an allocation $\pi = (\pi_1, \dots, \pi_n)$, we define

$$\alpha(\pi) = \min_{i \in A} \frac{u_i(\pi_i)}{\text{NPS}_i}$$

Now we can define $c_N(I)$ as follows:

$$c_N(I) = \max_{\pi \in \Pi} \alpha(\pi)$$

We will simply write c_N when the instance I is clear from the context. If $c_N \geq 1$ then a Nash-product fair allocation for I exists.

We consider the following optimization problem:

NPS-Opt	
Input:	<i>An allocation setting (A, O, U).</i>
Output:	<i>Compute an allocation π that maximizes $c_N(I)$.</i>

Note that **NPS-Opt** is NP-hard and thus our focus is on designing approximation algorithms which run in polynomial time in the size of the instance and has small constant approximation factor. In what follows, we provide a PTAS which relies on the following recent result by Nguyen et al. [20].

Theorem 3. (Nguyen et al. [20]). If the number of agents is fixed and for any constant $\varepsilon \in (0, 1)$, one can compute a set of feasible allocations P_ε such that for any allocation $\pi \in \Pi$, there is an allocation $\pi' \in P_\varepsilon$ such that $u_i(\pi'_i) \geq (1 - \varepsilon)u_i(\pi_i)$, for all $i \in A$.

Theorem 4. For a fixed number of agents, one can approximate **NPS-Opt** in polynomial time to within a factor of $1 - \varepsilon$, for any fixed constant $\varepsilon \in (0, 1)$.

Proof. Let $I = (A, O, U)$ be an allocation setting and ε be a fixed constant. We first compute a set P_ε using the algorithm of Nguyen et al. [20]. Furthermore, for each agent i we compute the approximate value $(1 - \varepsilon)\text{NPS}_i$ by using the algorithm of Nguyen et al. [19]. Among the allocations in P_ε , we pick the one, says π^* , that maximizes

$$\min_{i \in A} \frac{u_i(\pi_i)}{(1 - \varepsilon)\text{NPS}_i}.$$

Algorithm. PTAS

Input: $I = (A, O, U)$ and $\varepsilon \in (0, 1)$.

Output: An allocation π that maximizes $c_N(I)$.

1: Compute P_ε using Algorithm in [19]

2: Compute $\omega_i \leftarrow \lceil (1 - \varepsilon)\text{NPS}_i \rceil$ all i

3: $\pi^* \leftarrow \arg \max_{\pi \in P_\varepsilon} \min_{i \in A} \frac{u_i(\pi_i)}{\omega_i}$

4: Return π^*

Now let $\pi \in \Pi$ be an optimal allocation for I such that $u_i(\pi_i) \geq (1 - \varepsilon)\text{NPS}_i$, for all $i = 1, \dots, n$. There is an allocation $\pi' \in P_\varepsilon$ such that

$$u_i(\pi'_i) \geq (1 - \varepsilon)u_i(\pi_i) \geq c_N(I) \cdot (1 - \varepsilon) \cdot \text{NPS}_i$$

for all $i = 1, \dots, n$. On the other hand, we have

$$\min_{i \in A} \left\{ \frac{u_i(\pi_i^*)}{(1 - \varepsilon)\text{NPS}_i} \right\} \geq \min_{i \in A} \left\{ \frac{u_i(\pi'_i)}{(1 - \varepsilon)\text{NPS}_i} \right\} \geq c_N(I).$$

It follows that $u_i(\pi_i^*) \geq (c_N(I) - \varepsilon) \cdot \text{NPS}_i$, for all $i = 1, \dots, n$. This completes the proof. ■

IV. CONCLUSION

In this paper we have introduced a new criterion of fairness which we call Nash-product share. This criterion lies between the max-min share and the proportional share in the scale studied by Bouveret and Lemaitre [7]. We have also studied the three problems and obtained several novel results. In fact, we have proved that both the problem of computing the Nash-product share and the problem of checking whether there exists a Nash-product fair allocation for a given allocation setting are NP-hard. In addition, we have presented a polynomial-time approximation scheme for the problem of determining, given an allocation setting i , the largest value of $c_N(I)$ for which every agent receives a bundle of value of at least a fraction $c_N(I)$ of her Nash-product share.

ACKNOWLEDGEMENTS

We would like to thank the anonymous referees for their helpful comments and suggestions. This work was supported by Vietnam National Foundation for Science and Technology Development (NAFOSTED Project No. 102.01-2015.33).

TÀI LIỆU THAM KHẢO

- [1] G. Amanatidis, E. Markakis, A. Nikzad, and A. Saberi. Approximation algorithms for computing maximin share allocations. In Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming, pages 39–51. Springer LNCS 9134, July 2015.
- [2] H. Aziz, G. Rauchecker, G. Schryen, and T. Walsh. Approximation algorithms for max-min share allocations of indivisible chores and goods. In Proceeding of The Thirty-First AAAI Conference on Artificial Intelligence, pages 335–341, 2017.
- [3] I. Caragiannis, D. Kurokawa, H. Moulin, Ariel D. Procaccia, N. Shah, J. Wang: The Unreasonable Fairness of Maximum Nash Welfare. EC 2016: 305–322.

- [4] R. Cole, N. R. Devanur, V. Gkatzelis, K. Jain, T. Mai, V. V. Vazirani, and S. Yazdanbod. Convex Program Duality, Fisher Markets, and Nash Social Welfare. 18th ACM Conference on Economics and Computation (EC 2017).
- [5] S. Branzei, V. Gkatzelis, and R. Mehta. Nash Social Welfare Approximation for Strategic Agents. 18th ACM Conference on Economics and Computation (EC 2017).
- [6] R. Cole and V. Gkatzelis. Approximating the Nash Social Welfare with Indivisible Items. 47th ACM Symposium on Theory of Computing. (STOC 2015).
- [7] S. Bouveret and M. Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Journal of Autonomous Agents and Multi-Agent Systems*, 30(2):259–290, 2016.
- [8] S. Bouveret, Y. Chevaleyre, and N. Maudet. Fair allocation of indivisible goods. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 12, pages 284–310. Cambridge University Press, 2016.
- [9] E. Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.
- [10] R. Cole, V. Gkatzelis, and G. Goel. Mechanism design for fair division: allocating divisible items without payments. In *Proceedings of the 14th ACM Conference on Electronic Commerce*, pages 251–268. IFAAMAS, June 2013.
- [11] L. Gourvès, J. Monnot, and L. Tlilane. A matroid approach to the worst case allocation of indivisible goods. In *Proceedings of the 23th International Joint Conference on Artificial Intelligence*, pages 136–142, August 2013.
- [12] L. Gourvès, J. Monnot, and L. Tlilane. Worst case compromises in matroids with applications to the allocation of indivisible goods. *Theoretical Computer Science*, 589:121–140, 2015.
- [13] T. Hill. Partitioning general probability measures. *The Annals of Probability*, 15:804–813, 1987.
- [14] D. Kurokawa, A. Procaccia, and J. Wang. When can the maximin share guarantee be guaranteed? In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 523–529. AAAI Press, Feb. 2016.
- [15] J. Lang and J. Rothe. Fair division of indivisible goods. In J. Rothe, editor, *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, chapter 8, pages 493–550. Springer-Verlag, 2015.
- [16] E. Markakis and C.-A. Psomas. On worst-case allocations in the presence of indivisible goods. In *Proceedings of the 7th International Workshop on Internet & Network Economics*, pages 278–289, Dezember 2011.
- [17] A. Procaccia and J. Wang. Fair enough: Guaranteeing approximate maximin shares. In *Proceedings of the 15th ACM Conference on Economics and Computation*, pages 675–692. ACM Press, 2014.
- [18] T. Nguyen and J. Rothe. Minimizing envy and maximizing average Nash social welfare in the allocation of indivisible goods. *Discrete Applied Mathematics*, 179: 54–68, 2014.
- [19] N. Nguyen, T. Nguyen, M. Roos, and J. Rothe. Computational complexity and approximability of social welfare Optimization in multiagent resource allocation. *Autonomous agents and multi-agent systems*, 28(2): 256–289, 2014.
- [20] N. Nguyen, T. Nguyen and J. Rothe. Approximate Solutions To Max-Min Fair and Proportionally Fair Allocations of Indivisible Goods. In *Proceeding of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017)*, pages 262–271, Sao Paulo, Brazil, 2017.
- [21] E. Lee. APX-hardness of maximizing Nash social welfare with indivisible items. *Information Processing Letters*, 122: 17–20, 2017.

TIÊU CHUẨN NASH-PRODUCT SHARE TRONG CÁC BÀI TOÁN PHÂN CHIA CÔNG BẰNG

Nguyễn Trung Thành, Lê Đăng Nguyễn

ttnguyen.cs@gmail.com, ldnguyenth1976@gmail.com

TÓM TẮT: Bài toán phân chia tài nguyên một cách công bằng giữa các khách hàng là một trong những bài toán cơ bản thu hút được nhiều quan tâm trong vài thập kỉ qua. Có một vài tiêu chuẩn cho việc đánh giá thể nào là một phép phân chia công bằng, chẳng hạn như: tiêu chuẩn max-min share, tiêu chuẩn proportional share, tiêu chuẩn minmax share, tiêu chuẩn envy-free, và tiêu chuẩn CEEI. Trong bài báo này, chúng tôi giới thiệu một tiêu chuẩn mới cho sự công bằng, đặt tên là tiêu chuẩn Nash-product share, được xác định thông qua việc tính toán một phép phân chia tối đa hóa Nash-product cho trường hợp các hàm tham khảo cộng tính của khách hàng là giống nhau. Một phép phân chia thỏa mãn tiêu chuẩn này được gọi là phép chia công bằng Nash-product. Chúng tôi chỉ ra rằng việc tính toán Nash-product share là NP-khó thậm chí chỉ với 2 khách hàng. Thêm nữa, chúng tôi chứng minh rằng bài toán kiểm tra sự tồn tại của phép chia công bằng Nash-product là bài toán NP-khó khi số các khách hàng không phải là hằng số cho trước. Nếu số các khách hàng là hằng số, chúng tôi thu được một lược đồ xấp xỉ với thời gian đa thức.