Manipulations of Equivalent Preference in Parallel Allocation Mechanism

Wei Huang¹, Dong Cai²*, Wen Huang², Yuge Lu², Qian Tang² and Chong Huang²

¹ Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, No. 1, Jininglinglu, Guilin City, Guangxi, China
² School of Computer Science and Information Security, Guilin University of Electronic Technology, No. 1, Jininglinglu, Guilin City, Guangxi, China
Email: 1803201001@mails.guet.edu.cn

Abstract. The parallel distribution is a non-centralized mechanism for distributing indivisible items to agents, which can take into account computing efficiency, economic benefits and social equality. However, as most decentralized allocation mechanism, the parallel protocol is not strategy-proof. In this paper, supposed the manipulator has additive preferences with possible indifferences between single objects, we study the most basic manipulation problem under the parallel allocation mechanism. For any given set of items, we proved that the agent¹ can determine whether all objects in the set can be guaranteed in polynomial time. In addition, we given an algorithm for finding pessimism and proved the correctness, completeness and time complexity of polynomial.

Keywords. Parallel protocol; manipulation problem; equivalent preference.

1. Introduction

The problem of resource allocation is a very important issue in computer science and economics. Therefore, artificial intelligence theory and technology research has paid more and more attention to the design of the distribution system for multiple self-interesting agents. Relevant research starts from various realistic environmental constraints (including paid or unpaid allocation [1, 2], whether the price of resources is restricted [3], centralized or non-centralized allocation [4-6], resources to be allocated are divisible [7-11] or indivisible [12-14], etc.). Design and analyze the distribution system for multiple self-interesting agents at the level of practical operation (especially the level of specific process computing) based on consideration of the individual's private interests and the resulting rational behavior (that is, the self-interest Agent pursues the maximization of its private interests).

We mainly study how to design an allocation system for multiple self-interest agents in two aspects: (1) In the case where the self-interested agents are all honest, whether the distribution results that take into account economic benefits and social equality can be calculated in polynomial time; And (2) if one self-interested agents attempt to find dishonest (such as manipulation [15] and conspiracy [16]) action plans that can bring additional benefits, whether the calculation task is difficult. In this paper, we mainly focus on the second aspect, which is to analyze the computational complexity of self-interested agents to manipulate the allocation results.

Bouveret and Lang studied a sequence resource allocation mechanism [2]. Under this sequence mechanism, each self-interested agent does not need to submit any information before the allocation process begins, she only needs to pick her favorite resources from the remaining objects in turn according to a specified order. Then, Kalinowski et al. [17] proved some of Bouveret and Lang’s assumptions about
the optimal agent order in theory, and analyzed some computational problems related to manipulation under the sequence mechanism [18]. In addition, Huang et al. [19] studied a new parallel resource allocation system, and under the assumption that all participating agents have strict preferences, Ref. [20] analyzed the computational complexity of manipulation problems under this system.

In this paper, we consider a restricted model and focus on equivalent preference of manipulator. It is assumed that other agents always act truthfully, i.e., she will ask the favorite remaining items in each round, according to a linear order; manipulator is a manipulator that knows this fact (including other agents’ preference order) and has additive preferences with possible indifferences between single objects (i.e., equivalent preference). For any given set of items, we proved that the agent 1 can determine whether all objects in the set can be guaranteed in polynomial time. In addition, we given an algorithm for finding pessimism and proved the correctness, completeness and time complexity of polynomial.

2. Model and Notations

There is a set of $m \geq 2$ indivisible and distinct items $O = \{o_1, ..., o_m\}$. The items are distributed to a set of agents $N = \{1, ..., n\}$ according to a preference profile $\succ = \{\succ_1, ..., \succ_n\}$ where $\succ_i$ donate the weak preference order of agent $i$ over $O$. Let $\Gamma$ donate the set of all the possible preference profiles for agents in $N$.

Let $u_i : 2^O \to R_+$ be the utility function of agent $i \in N$. For any object $o \in O$, we also write $u_i(o)$ as $u_i(o)$. In this paper, we assume that $u_i(\emptyset) = 0$ and $u_i$ is additive, i.e., $u_i(X) = \sum_{o \in X} u_i(o)$ for any set $X \subseteq O$.

We write $o_1 \succ_i o_2$ to means that agent $i$ values object $o_1$ at least as much as object $o_2$ and use $\succ_i$, i.e., $o_1 \succ_i o_2$ if and only if $o_1 \succ_i o_2$ but not $o_2 \succ_i o_1$. In addition, $\sim_i$ donates agent $i$’s equivalence relation, i.e., $o_1 \sim_i o_2$ if and only if $u_i(o_1) = u_i(o_2)$. So there are some equivalence classes $E^i_1, E^i_2, ..., E^i_k$ for $k_i$ such that $o \succ_i o'$ if and only if $o \in E^i_j$ and $o' \in E^i_{j'}$ for $j \neq j'$. We donate the preferences $o_1 \succ_i o_2 \sim_i o_3$ by the list: $i: \{o_1\} \succ \{o_2, o_3\}$ for a short. In this paper, we assume that only agent $i = 1$ has equivalence classes, i.e., preference profile $\succ = \{\succ_1, \succ_2, ..., \succ_n\}$. For $i \in N \setminus \{1\}$, let $r_i(o) \in \{1, ..., |O|\}$ to donate the rank of object $o$ in agent $i$’s preference, that is, $o \succ_i o'$ if and only if $r_i(o) < r_i(o')$.

Let $E(o)$ donate the set of objects which they have the same value as $o$ for agent 1, i.e., $o \sim_1 o'$ if and only if $\{o\} \cup \{o'\} \subseteq E(o)$. Let $X \subseteq O$, we donate $X' = \sim_1 X$ if and only if (1) $|X'| = |X|$, (2) $|X \cap E(o)| = |X' \cap E(o)|$ for all $o \in O$. We call $X'$ is an equivalent set of $X$. Let $E(X)$ donate the set of all equivalent sets $X'$ of $X$.

Example 1: Consider this preference: agent 1: $\{o_2, o_3\} \succ \{o_5, o_7, o_9\} \succ \{o_4, o_6\}$. Then we know that $E(o_2) = \{o_2, o_3\}$, $E(o_5) = \{o_5, o_7, o_9\}$, $E(o_4) = \{o_4, o_6\}$. Let $X = \{o_2, o_5, o_1\}$, then $E(X) = \{\{o_2, o_5, o_1\}, \{o_2, o_5, o_7\}, \{o_2, o_5, o_9\}, \{o_3, o_5, o_1\}, \{o_3, o_5, o_7\}, \{o_3, o_5, o_9\}\}$.

The description of the parallel allocation mechanism is as follows. In each round, each agent reports a favorite remaining item. If an item is reported by only one agent, then the item is assigned to this agent, otherwise the winner of a simple game (i.e., each agent has the same possibility to win) will get the object when more than one agent asks the same object. Then, repeat the process as long as there are remaining items. We call agent $i$ has pessimism if and only if agent $i$ cannot get the object when more than one agent (including agent $i$) asks that, is, agent $i$ always has been a loser of the simple game.

A strategy $\delta$ over some $O' \subseteq O$ is a finite sequence $\delta(1), ..., \delta(|\delta|)$ such that $\delta(i) \in O'$ and $\delta(i) \neq \delta(j)$ for any $1 \leq i \neq j \leq |\delta|$. Intuitively, assuming that the set of remaining objects is $O'$, $\delta(i)$ specifies the object that agent $i$ reports in the next $i$th round. Some strategies may fail because some objects that agent $i$ intends to report has already been allocated. We say strategy $\delta$ is well defined if for $1 \leq i \leq |\delta|$, object $\delta(i)$ is still available in the next $i$th round, and there is no object available after the next $|\delta|$th round. In the rest of this paper, we only consider well-defined strategies, and a strategy over $O$ is called a strategy for short.

We use $(O, N, \succ, S)$ donates a manipulation problem for the manipulator agent 1 consists of $O, N, \succ$ and a set of target objects $S \subseteq O$. A strategy $\delta$ is successful for $(O, N, \succ, S)$ if assuming
agent’s (agent \( i' \in N \setminus \{1\} \)) behavior is honest, this strategy \( \delta \) can ensure that agent 1 gets all the items in an equivalent set \( S' \in E(S) \).

3. Manipulations for Two-Agents

Firstly, we only consider two agents \( N = \{1,2\} \), \( m \geq 2 \) indivisible objects \( O = \{o_1, \ldots, o_m\} \) and preference profile \( \succ = \{\succ_1, \succ_2\} \).

It is assumed that agent 2 always acts truthfully, i.e., she will ask the favorite remaining items in each round, according to a strict preference order (i.e., there are different benefits for different objects, that is, \( u_2(o_i) \neq u_2(o_j) \) if and only if \( 1 \leq i \neq j \leq |O| \)). Agent 1 is a manipulator that knows this fact and she has a weak preference order over \( O \).

Remark Let \( M = ((>1, >2, \ldots, >n), S) \) be a manipulation problem. Suppose that for all \( t \geq 1, S_t, OA_t \) and \( OD_t \) are constructed according to the target set \( S \) respectively. Then there exists a successful strategy for \( M \) if and only if \( k \geq |1_{\text{st}\{S_t\}}| \) for any \( k \geq 1 \).

We define some notions: Let \( X \subseteq O \) and \( X \neq \emptyset \), \( \text{Worst}(X) = o \in X \text{ s.t. } o' > o \text{ for every } o' \in X \setminus \{o\} \), i.e., \( \text{Worst}(X) \) is a function that will return an object \( o \) such that agent 2 doesn't like it the least according to her preference. Let \( N' = N \setminus \{1\} \), for all \( i \in N', A, B \subseteq O \) such that \( A \cap B = \emptyset, \text{Better}_i(A, B) = \{o \in A(\forall o' \in B) >_i o'\} \), and if \( A \neq \emptyset \) then \( \text{Best}_i(A) = o \in A \text{ s.t. } o >_i o' \text{ for every } o' \in A \setminus \{o\} \). Given a manipulation problem \((O, N, \succ, S), S \subseteq O \), if for any set \( X \) such that \( \sum_{o \in X} r_2(o) \leq \sum_{o \notin BS} r_2(o) \) where \( X, BS \in E(S) \), then we call \( BS \) the boundary set of solving \((O, N, \succ, S)\).

Theorem 1 Given a manipulation problem \((O, N, \succ, S), S \subseteq O \), if \( BS \in E(S) \) is the boundary set and solving \((O, N, \succ, S) \) is judged not to be a successful strategy, then for any set \( X \in E(S) \), there is also not a successful strategy to solve \((O, N, \succ, S) \).

Proof For a manipulation problem \((O, N, \succ, S) \), we can construct the boundary set \( BS \sim S \) according to the preference \( \succ \) and \( S \). Then we know that \( \sum_{o \in BS} r_2(o) \geq \sum_{o \notin BS} r_2(o) \) for any set \( X \sim S \). So there exists the object \( o \in BS \) and \( o \sim o' \) such that \( r_2(o) > r_2(o')(i.e., o' > o) \) for any \( o' \in X \setminus BS \).

According to definition of \( \text{Better}_i(A,B) \), when \( i = 2 \), if the set \( X \sim BS \) and \( X, BS \cap B = \emptyset \), then we can know that \( |\text{Better}_2(X, B)| \geq |\text{Better}_2(B, S)| \). We also know that \( S_k = \text{Better}_k(OA_k \cap X, OA_k \setminus X) \) by remark 1. Then \( BS_k = \text{Better}_2(OA_k \cap BS, OA_k \setminus BS) \) and \( SX_k = \text{Better}_2(OA_k \setminus X, OA_k \setminus X) \) means that the set of equivalent target objects that must be achieved no later than round \( k \) respectively. So \( |1_{\text{st}\{S_k\}}| \geq |1_{\text{st}\{BS_t\}}| \) for any \( k \geq 1 \).

If the object of boundary set \( BS \) cannot all be allocated to agent 1 (i.e., there is no successful strategy to get all the objects in the boundary set \( BS \)). Then we can know that \( |1_{\text{st}\{BS_t\}}| \geq k \) for any \( k \geq 1 \) according to remark 1. Finally we can get:

\[
\bigcup_{1_{\text{st}\{S_k\}}} X_k \geq \bigcup_{1_{\text{st}\{BS_t\}}} BS_t \geq k, \forall k \geq 1
\]

Conclusion, if the object of boundary set \( BS \) cannot all be given by a strategy for \((O, N, \succ, S) \), then the object of any equivalent set \( X, BS \sim S \) cannot all be allocated to agent 1.

We develop Algorithm 1 to find a successful strategy if it exists. This algorithm is adapted from Ref. [20].

**Algorithm 1: Find a Successful Strategy**

**input:** \((O, N, \succ, S)\)

**output:** a successful strategy if it exists

1. \( OA \leftarrow O; E \leftarrow \{E(o) | o \in S\}; j \leftarrow 0; k \leftarrow 1; \)
2. for all \( E(o) \in E; \)
3. \( j \leftarrow |E(o) \cap S|; \)
4. while \( j \neq 0; \)
5. \[ o' \leftarrow \text{Worst}(E(o)); \]
6. \[ E(o) \leftarrow E(o) \setminus \{o'\}, BS \leftarrow BS \cup \{o'\}; \]
7. \[ j \leftarrow j - 1; \]
8. \[ S', S \leftarrow BS; \]
9. \[ \text{while}(OA \neq \emptyset); \]
10. \[ S' \leftarrow \text{Better}_2(S', OA \setminus S'); \]
11. \[ \text{size} \leftarrow \text{size} + |S'|; \]
12. \[ \text{if}(\text{size} \geq k) \text{ then return failure}; \]
13. \[ \text{for all } o \in S'; \]
14. \[ \tau \leftarrow \tau \cdot o; \]
15. \[ OD \leftarrow \{o \in OA\setminus S'|(\exists i \in N')o = \text{Best}_2(OA\setminus S')\}; \]
16. \[ \text{if } k > |S| \text{ and } OD \neq \emptyset; \]
17. \[ \text{Randomly pick an object } o \text{ from } OD; \]
18. \[ \tau' \leftarrow \tau' \cdot o; \]
19. \[ OA \leftarrow OA \setminus (S' \cup OD), S' \leftarrow S' \setminus \]
20. \[ \text{return } \tau \cdot \tau'; \]

**Example 2:** Consider this preference profile and the target set \( S = \{o_2, o_5, o_1\} \), agent 1: \( \{o_2, o_3\} > \{o_5, o_7, o_1\} > \{o_4, o_6\} \). Agent 2: \( o_4 > o_2 > o_1 > o_5 > o_7 > o_6 > o_3 \). The boundary set \( BS = \{o_3, o_5, o_7\} \), and we can get a successful strategy \( o_5 \cdot o_7 \cdot o_3 \cdot o_6 \).

### 4. Manipulations for N-Agents

For further research, we study more agents, but there is only one manipulator, and the behaviors of other agents are honest. In the following study, we have a set of objects \( O = \{o_1, \ldots, o_m\} \), a set of agents \( N = \{1, \ldots, n\} \) and profile \( \succ = \{\succ_1, \succ_2, \ldots, \succ_n\} \). Agent 1 be the manipulator that knows the preference orders of the other agents, and she has a weak preference order.

There is our model of two-layer to solve manipulation problem \( (O, N, \succ, S) \): (1) The outside (outer layer) can judge whether the manipulation problem \( (O, N, \succ, S) \) has a successful strategy; (2) The inside (inner layer) can find the outside’s equivalent target set.

The **outside**: Through the set \( \overline{BS}_k \), we use \( OA_k, OD_k \) and \( S_k \) to describe the representation of the success strategy for the \( k \)th round on **outside** of the parallel allocation process:
- \( OA_k \) donates the set of items remaining after \( k - 1 \) rounds on outside,
- \( BS_k \) donates the equivalent target set of objects that be found in round \( k \) on outside,
- \( S_k \) donates the set of items that must be obtained before \( k \) rounds (including \( k \) rounds) on outside.
- \( OD_k \) defines the set of items acquired by other agents in round \( k \) on outside.

Mathematically, let \( N' = N \setminus \{1\} \), then:

- \( OA_1 = O \)
- \( BS_k = U_{1 \leq j \leq r} \overline{BS}_j^{k} \)
- \( S_k = U_{i \in N'} \text{Better}_i(OA_k \cap U_{1 \leq j \leq k} BS_j, OA_k \setminus U_{1 \leq j \leq k} BS_j) \)
- \( OD_k = \{o \in OA_k \setminus U_{1 \leq j \leq k} BS_j | (\exists i \in N')o = \text{Best}_i(OA_k \setminus U_{1 \leq j \leq k} BS_j)\} \)
- \( OA_{k+1} = OA_k \setminus (S_k \cup OD_k) \)
\textbf{The inside}: When the round of \textit{outside} is \( k \), we use \( OA^k_r, OD^k_r, OS^k_r, ON^k_r, BS^k_r \) to describe the requirements that find the equivalent target set in round \( r \) on \textit{inside}.

- \( OA^k_r \) donates the set of objects that can remain after round \( r \) on \textit{inside},
- \( OS^k_r \) donates the set of objects that may become boundary target set after round \( r \) on \textit{inside},
- \( OD^k_r \) donates the set of objects that may be asked by other agents in round \( r \) on \textit{inside},
- \( ON^k_r \) donates the set of objects that have appeared no matter than round \( r \) on \textit{inside},
- \( BS^k_r \) donates the set of equivalent target objects in round \( r \) on \textit{inside}.

Formally, let \( N' = N \setminus \{1\} \), then:

\begin{itemize}
  \item \( OA^k_r = OA_k \cup U_{i \in I_S(k)} BS_j \)
  \item \( OS^k_r = \bigcup_{o \in S} E(o) \cup \bigcup_{o \in U_{i \in I_S(k)} BS_j} E(o) \)
  \item \( OD^k_r = \{ o \in OA^k_r | (\exists i \in N') o = \text{Best}_i(OA^k_r) \} \)
  \item \( ON^k_r = U_{i \in I_S} OD^k_r \cup \{ OA_1 \setminus OA_k \} \)
  \item \( BS^k_r = \left\{ \begin{array}{l}
\emptyset, \forall o \in OD^k_r, o \notin OS^k_r; \\
\emptyset, \forall o \in OD^k_r, \exists o \in OS^k_r \text{ and } |E(o) \cap S| < |E(o) \setminus ON^k_r|; \\
(E(o) \setminus ON^k_r) \cup OW, \text{ otherwise}
\end{array} \right. \)
  \item \( BS_{r+1}^k = BS^k_r \cup U_{o \in BS^k_r} E(o) \)
  \item \( OA_{r+1}^k = OA^k_r \setminus (OD^k_r \cup BS^k_r) \)
\end{itemize}

The above is some formal definitions of the manipulation problem solving model, \( OW = \text{randomly pick } |E(o) \cap S| - |E(o) \setminus ON^k_r| o \text{ objects from } \{E(o) \setminus OD^k_r\} \).

\textbf{Theorem 2} Let \((O, N, \succ, S)\) be a manipulation problem, Algorithm 2 can find a successful strategy for \((O, N, \succ, S)\) if it exists.

\textit{proof} Given a manipulation problem \((O, N, \succ, S)\). Suppose that for all \( k \geq 1, r \geq 1 \), the \textit{outside} and \textit{inside} are constructed based on the target set \( S \) respectively. Let \( BS = U_{i \in I_S} BS_j \) that means the set of equivalent target items we find behind round \( k \) on \textit{outside}.

(1) For \textit{outside}: we mainly prove whether exists a successful strategy to get all objects of the set \( BS \).

(a) In the case when \( BS \) is a singleton \( \{o\} \), it is easy to find that \( |U_{1 \leq i \leq k} BS_i| = \{ 1 \} \) for every \( k \geq 1 \). So \( BS_1 = \emptyset \) if and only if if \( k > |U_{1 \leq i \leq k} BS_i| \) for any \( k \geq 1 \). If \( BS_1 = \emptyset \) (i.e., no agent \( i \in \{2, \ldots, n\} \) asks \( o \) in 1th round) then the agent 1 can get \( o \) by asking \( o \) at first. If \( BS_1 \neq \emptyset \) (i.e., there must be some agent \( i \in \{2, \ldots, n\} \) asking \( o \) in 1th round) then there is no successful strategy for \((O, N, \succ, S)\).

(b) Suppose this view holds for any \( BS \) such that \( 1 \leq |BS| < p \).

(c) Consider a target set \( BS = \{ o'_1, \ldots, o'_p \} \). Then \( k > |U_{1 \leq i \leq k} BS_i| \) for any \( k \geq 1 \) if and only if \( k > |U_{1 \leq i \leq k} BS'_i| \) for any \( p \geq k \geq 1 \). Let \( BS' = BS \setminus \{ o'_p \} \). Suppose that for all \( t \geq 1 \), \( BS'_t, OA'_t, \text{ and } OD'_t \) are constructed according to \( BS' \) respectively.

\textbf{(Sufficiency)} If \( k > |U_{1 \leq i \leq k} BS_i| \) for any \( p \geq k \geq 1 \). Then \( o'_p \notin U_{1 \leq t \leq k} (BS'_t \cup OD'_t) \subseteq U_{1 \leq t \leq k} (BS'_t \cup OD'_t) \) and \( k > |U_{1 \leq i \leq k} BS_i| \geq |U_{1 \leq i \leq k} BS'_i| \) for any \( p \geq k \geq 1 \). Based on the above statements and assumptions, there is a successful strategy \( \tau' \) for \((O, N, \succ, BS)\) starting by asking the items in \( BS' \). Based on the above, if manipulator asks the item by \( \tau' \) in every round \( k < p \), then \( o'_p \) is not asked by any agent \( i \in \{2, \ldots, |N|\} \) in \( p \)th round. Let \( \tau \) be a strategy asking the item specified by \( \tau' \) in any round \( k < p \), and asking \( o'_p \) in \( p \)th round. So we can say that \( \tau \) is a successful strategy for \((O, N, \succ, BS)\).
(Necessity) If there exists some \( p \geq k \geq 1 \) such that \( k \leq |\bigcup_{i=1}^{k} BS_i| \), then in some round \( k' \leq k \), there must be some agent \( i \in \{2, \ldots, |N|\} \) asking some \( o \in U_{\bigcup_{i=1}^{k} BS_i} \). In this case, there is no successful strategy for \((O, N, \geq, BS)\). Then the view holds for any \( BS \in O \) s.t. \(|S| = p\).

(2) For inside: we mainly proved that there not successful strategy for \((O, N, \geq, X)\) s.t. \( X \sim BS \), if and only if there is not a successful strategy for \((O, N, \geq, BS)\). Firstly, supposed that there is not successful strategy for \((O, N, \geq, BS)\). And \( BS \) is the set of objects which are agent \( i' \) (i.e., \( i' \in N \setminus \{1\} \)) most hated after round \( k \) in outside. Let \( X \sim BS \), there exist an agent \( i' \) and the object \( o \in BS, o \sim o' \) such that \( r_i(o) \geq r_{i'}(o) \) for any object \( o' \in X \setminus BS, j' \in N \setminus \{1\} \). We can construct \( BS_k \) and \( X_k \) according to definition of \( S_k \), so \( |U_{\bigcup_{i=1}^{k} X_i}| \geq |U_{\bigcup_{i=1}^{k} BS_i}| \) for \( k \geq 1 \). Finally, we can conclude that \( |U_{\bigcup_{i=1}^{k} X_i}| \geq |U_{\bigcup_{i=1}^{k} BS_i}| \geq k \) according to assumption and 1). We can proved that agent \( 1 \) cannot get all objects of \( X \) if she can not get all objects of set \( BS \) where \( X \sim BS \).

In conclusion, Algorithm 2 can find a successful strategy for \((O, N, \geq, S)\) if it exists.

We develop Algorithm 2 to find a successful strategy if it exists.

Algorithm 2: Find a Successful Strategy \( \tau \)

**input**: \((O, N, \geq, S)\)

**output**: a successful strategy if it exists

1. \( OA \leftarrow O; BS^* \leftarrow \emptyset; OS \leftarrow \bigcup_{o \in OS} E(o); N' \leftarrow N \setminus \{1\}; \) size, count \( \leftarrow 0; k \leftarrow 1; \)
2. while (\( OA \neq \emptyset \)):
3.  \( \text{if}(\text{count} \neq |S|) : \)
4.  \( OA' \leftarrow OA \setminus BS^* ; \)
5.  \( \text{do:} \)
6.  \( OR \leftarrow \{ o \in OA' | (\exists i \in N') o = \text{Best}_i(OA') \}; \)
7.  \( OS' \leftarrow OS \setminus OR; \)
8.  \( \text{if}(OS' \neq \emptyset) : \)
9.  \( OS^* \leftarrow \{ OA \setminus OA_k \} \cup OS'; \)
10. \( E^* \leftarrow \{ E(o) | o \in OS' \}; \)
11. \( \text{for all } E(o) \in E^*: \)
12. \( \alpha \leftarrow |E(o) \cap S|, \beta \leftarrow |E(o) \setminus OS'|; \)
13. \( \text{if}(\beta \leq \alpha) : \)
14. \( \gamma \leftarrow \alpha - \beta; \)
15. \( OW \leftarrow \text{randomly pick } \gamma \text{ objects} \)
16. \( \text{from } (OR \cap E(o)); \)
17. \( BS \leftarrow \{ E(o) \setminus OS^* \} \cup OW; \)
18. \( OS \leftarrow OS \setminus U_{o \in BS} E(o); \)
19. \( BS^* \leftarrow BS^* \cup BS; \)
20. \( OA' \leftarrow OA' \setminus \{ OS_k \cup BS \}; \)
21. \( \text{while}(BS \neq \emptyset) ; \)
22. \( S^* \leftarrow U_{\bigcup_{i=1}^{N'}} \text{Better}_{x}(BS^*, OA \setminus BS^*); \)
23. \( \text{size} \leftarrow \text{size} + |S^*|; \)
24. \( \text{if}(\text{size} \geq k) \) then return failure;
25. for all \( o \in S^* \):
26. \( \tau \leftarrow \tau \, o \); 
27. \( OD \leftarrow \{ o \in OA \backslash BS^* | (\exists i \in N') o = \text{Best}_i(OA \backslash BS^*) \}; \)
28. if \( k > |S| \) and \( OD \neq \emptyset \):
29. Randomly pick an object \( o \) from \( OD \);
30. \( \tau' \leftarrow \tau' \, o \);
31. \( OA \leftarrow OA \backslash (S^* \cup OD), BS^* \leftarrow BS^* \backslash S^* \), 
\( k \leftarrow k + 1 \);
32. return \( \tau \cdot \tau' \); 

**Example 3:** Consider this preference profile and the target set \( S = \{ a_1, a_4, a_2 \} \), agent 1: \( \{ a_1 \} \succ \{ a_2, a_3 \} \succ \{ a_4, a_6, a_7 \} \succ \{ a_8 \} \), agent 2: \( a_4 \succ a_3 \succ a_7 \succ a_5 \succ a_1 \succ a_6 \succ a_3 \succ a_6 \succ a_3 \), agent 3: \( a_4 \succ a_2 \succ a_1 \succ a_5 \succ a_7 \succ a_6 \succ a_3 \succ a_6 \succ a_3 \), agent 4: \( a_4 \succ a_2 \succ a_1 \succ a_5 \succ a_7 \succ a_6 \succ a_3 \succ a_6 \succ a_3 \).

When \( k = 1 \):
\( OA_1 = \{ a_1, a_2, a_3, a_4, a_5, a_7, a_9, a_9 \}, BS_1 = \emptyset, S_1 = \emptyset, OD_1 = \{ a_4, a_3, a_7 \}; \)
\( k = 2 \):
\( OA_2 = \{ a_1, a_2, a_3, a_4, a_5, a_7, a_9, a_9 \}, BS_2 = \{ a_6, a_2, a_1 \}, S_2 = \emptyset, OD_2 = \{ a_5, a_6, a_9 \}; \)
\( k = 3 \):
\( OA_3 = \{ a_1, a_2, a_6 \}, BS_3 = \{ a_6, a_2, a_2 \}, S_3 = \{ a_6, a_2, a_1 \}, OD_3 = \{ a_6, a_2, a_1 \}; \)
return failure (i.e., there is no successful strategy for this manipulation problem).

5. Conclusion
In this paper, we first studied the manipulation of two agents in the case of equivalence relationship, and giving a polynomial algorithm to determine whether the manipulation problem has a successful strategy. Then, with the help of the above research result, we give a judgment algorithm for n-person manipulation problems.

6. Future Work
At present, there are still few studies on the parallel distribution system, and its various properties also need to be further studied. We have only studied a small part of the field of manipulation under the parallel distribution system. There are still some directions that can be studied:

1. In this paper, we studied the manipulation problem of equivalent preference under pessimism, but the manipulation problem under optimism (that is, multiple agents report the same item at the same time, and the operator is optimistic that the item can be obtained) remains to be studied;

2. In this paper, there is only one manipulator in the manipulation problem. Next, we can discuss the problem of conspiracy by more than one person. In this case, self-interesting agents conspired with each other to find whether the computational complexity of an optimal manipulation strategy is NP-Hard.

Acknowledgment
This work is supported by the National Natural Science Foundation of China under grants No.61105039, No.61463044, No.61662011, and Innovation Project of GUET Graduate Education (Issue Number 2020YCXSS054, 2019YCXSS055).

References
[1] Sandholm T 2002 Algorithm for optimal winner determination in combinatorial auctions Artificial Intelligence 135 1-54.
[2] Bouvier S and Lang J 2011 A general elicitation-free protocol for allocating indivisible goods In Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI’11) pp 73-78.
[3] Talman D and Yang Z 2008 A dynamic auction for differentiated items under price rigidities Economics Letters 99 278-281.
[4] Cramton P, Shoham Y and Steinberg R 2006 Combinatorial Auctions (MIT Press).
[5] Ausubel L M 2006 An efficient dynamic auction for heterogeneous commodities *American Economic Review* **96** (3) 602-629.

[6] Cohler Y J, Lai J K, Parkes D C and Procaccia A D 2011 Optimal envy-free cake cutting *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI’11)* pp 626-631.

[7] Brams S J, Feldman M, Lai J K, Morgenstern J and Procaccia A D 2012 On maxsum fair cake divisions *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI’12)* pp 1285-1291.

[8] Chen Y, Lai J K, Parkes D C and Procaccia A D 2010 Truth, justice, and cake cutting *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)* pp 756-761.

[9] Caragiannis I, Lai J K and Procaccia A D 2011 Towards more expressive cake cutting *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI’11)*.

[10] Procaccia A D 2013 Cake cutting: Not just child’s play *Communications of the ACM* **56** (7) 78-87.

[11] Kalinowski T, Narodytska N, Walsh T and Xia L 2012 Strategic behavior in a decentralized protocol for allocating indivisible goods *Proceedings of Fourth International Workshop on Computational Social Choice*.

[12] Markakis E and Psomas C-A 2011 on worst-case allocations in the presence of indivisible goods *Internet and Network Economics. Lecture Notes in Computer Science (LNCS)* **7090** 278-289.

[13] Brams S J, Kilgour D M and Klamler C 2012 The undercut procedure: an algorithm for the envy-free division of indivisible items *Social Choice and Welfare* **39** 2-3.

[14] Faliszewski P and Procaccia A D 2010 AI’s war on manipulation: Are we winning *AI Magazine* **31** (4) 53-64.

[15] Davies J, Katsirelos G, Narodytka N and Walsh T 2011 Complexity of and Algorithms for Borda Manipulation *Proceedings of the Twenty-fifth AAAI Conference on Artificial Intelligence (AAAI-11)*.

[16] Kalinowski T, Narodytska N and Walsh T 2013 A social welfare optimal sequential allocation procedure *In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI’13)*.

[17] Kalinowski T, Narodytska N, Walsh T and Xia L 2013 Strategic behavior when allocating indivisible goods sequentially *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI’13)*.

[18] Aziz H, Walsh T and Xia L 2015 Possible and necessary allocations via sequential mechanisms *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI’15)*.

[19] Huang W, Lou J and Wen Z 2013 A parallel elicitation-free protocol for allocating indivisible goods *Multidisciplinary Workshop on Advances in Preference Handling (IJCAI’13)*.

[20] Huang W, Zhang L, Huang Y and Lou J 2017 Allocating indivisible objects with a parallel method insensitive to identities *IEEE Access* **99** 1-1.