Truthfulness of a Proportional Sharing Mechanism in Resource Exchange

Yukun Cheng\textsuperscript{1}, Xiaotie Deng\textsuperscript{2}, Qi Qi\textsuperscript{3}, and Xiang Yan\textsuperscript{2}

\textsuperscript{1}Zhejiang University of Finance and Economics, Hangzhou China
ykcheng@amss.ac.cn
\textsuperscript{2}Shanghai Jiao Tong University, Shanghai China
dengxiaotie@sjtu.edu.cn, xyansjtu@163.com
\textsuperscript{3}The Hongkong University of Science and Technology, Hong Kong
kaylaqi@ust.hk

Abstract

In this paper, we consider the popular proportional sharing mechanism and discuss the incentives and opportunities of an agent to lie for personal gains in resource exchange game. The main result is a proof that an agent manipulating the proportional sharing mechanism by misreporting its resource amount will not benefit its own utility eventually. This result establishes a strategic stability property of the resource exchange protocol. We further illustrate and confirm the result via network examples.

1 Introduction

The rapid growth of the wireless and mobile Internet has provided an opportunity for wide applications of exchanging resources or services over networks, which go beyond the peer-to-peer (P2P) bandwidth sharing idea. In a traditional P2P network, each peer is both a supplier and a consumer of resources. Resources (such as processing power, disk storage or network bandwidth) are shared among multiple interconnected peers who use a mechanism to make a portion of their resources directly available to other network participants in a distributed management system [Schollmeier, 2001].

The successes of the resource exchange networks highly depend on the efficient resource utilization. A P2P system can be modeled as a pure exchange economy in which each user brings its own divisible resources to the market and exchanges its resources with its neighbors to derive utility. Formally, the system is modeled as an undirected graph $G$, where each node $i$ represents a peer with $w_i$ units of divisible resources (or weight) to be distributed among its neighbors. The utility $u_i$ is determined by the total amount of resources obtained from its neighbors. An efficient allocation in such an exchange economy can be characterized by the market equilibrium. On the other hand, to encourage users to contribute more, a fair allocation rule is essential. Ideally a user should receive as much as it gives. However, such perfect reciprocation may not be feasible due to the network structure. Then a proportional fairness mechanism is a natural approximation. Under this mechanism, each peer provides each neighbor a portion of its contribution in proportion to what it receives from its neighbors. Recently, Wu and Zhang [Wu and Zhang, 2007] showed that the market equilibrium obtained by a combinatorial method, called bottleneck decomposition is a proportional fair solution. They also showed the proportional dynamics converged to this equilibrium for linear utility functions under mild conditions.

However, agents are strategic. The market equilibrium is determined by agents’ reported information rather than their true information. An agent may have the incentive to misreport its information if it is profitable. Hurwicz [Hurwicz, 1972] showed that it is impossible to design a truthful mechanism that guarantees a market equilibrium outcome in general. Even in a Fisher market game with linear utility, it was shown that an agent may derive a better payoff [Adsul et al., 2010], more specifically, may double its benefit by strategic behaviors [Chen et al., 2014]. The resource exchange game studied in a P2P setting is a special case of Arrow-Debreu market where agents are only interested in their neighbors’ resources and only care about the total amount of resources received. Hurwicz’s impossibility theorem may not hold for restricted settings. A recent work [Cheng et al., 2015] discussed the issue of agents’ manipulations by cheating on their connectivity information. The authors proved that the proportional sharing mechanism is robust to such manipulations. As utilities are determined by allocation that depends on agents’ weights and the network structure, the strategic manipulations of an agent can only be misreporting its weight or misreporting its connection edges in the network. This work leaves a clear open question whether an agent could lie on the amount of resources it can offer. In this paper, we solve this open problem and prove the truthfulness of the proportional sharing mechanism. It shows that the proportional sharing mechanism is truthful, fair and Pareto efficient.

Technical Contributions

Our approach builds on a network bottleneck decomposition structure initially designed for the analysis of the connection of fairness (proportional sharing) and competitiveness (market equilibrium) [Wu and Zhang, 2007] and then the nonmanipulability [Cheng et al., 2015]. For a given sub-set of agents, the total resources of their neighbors has a key importance on their eventual gains with respect to their own
2 Preliminaries

In this section, we model the resource exchange setting by an undirected graph \( G = (V, E; w) \). Each vertex \( u \in V \) represents an agent with an upload resource amount (weight) \( w_u > 0 \) to be exchanged with its neighbors. Let \( \Gamma(u) \) be the neighborhood of \( u \), i.e., the set of vertices adjacent to \( u \) in \( G \). And \( f_{uv} \) is denoted as the fraction of resource \( u \) allocated to its neighbor \( v \). Obviously, \( 0 \leq f_{uv} \leq 1 \) and the resource vertex \( u \) provides to \( v \) is \( w_u f_{uv} \). \( F = (f_{uv})_{u,v \in G} \) is called a feasible allocation if \( \sum_{v \in \Gamma(u)} f_{uv} = 1 \), that means \( u \) allocates all its resource out. The utility of agent \( u \) is defined as \( U_u(F) = \sum_{v \in \Gamma(u)} f_{uv} w_v \), i.e., all received resource from its neighborhood \( \Gamma(u) \).

2.1 Bottleneck Decomposition

Consider an undirected connected graph \( G = (V, E; w) \) with weight \( w : V \to R_+ \). For any set \( S \subseteq V \), we define \( w(S) = \sum_{u \in S} w(u) \) and \( \Gamma(S) = \cup_{u \in S} \Gamma(u) \). It is possible that \( S \cap \Gamma(S) \neq \emptyset \). Define \( \alpha(S) = w(\Gamma(S))/w(S) \), referred to as the inclusive expansion ratio of \( S \), or the \( \alpha \)-ratio of \( S \) for short.

Definition 1 (Maximal Bottleneck) A vertex subset \( B \subseteq V \) is called a bottleneck of \( G \) if \( \alpha(B) = \min_{S \subseteq V} \alpha(S) \). \( B \) is a maximal bottleneck if for any subset \( \tilde{B}, \tilde{B} \subseteq V \) implies \( \alpha(B) \geq \alpha(\tilde{B}) \). \( (B, \Gamma(B)) \) is called the maximal bottleneck pair in \( V \).

Definition 2 (Bottleneck Decomposition) Given \( G = (V, E; w) \). Start with \( V_1 = V, G_1 = G \) and \( i = 1 \). Find the maximal bottleneck \( B_i \) of \( G_i \) and let \( G_{i+1} \) be the induced subgraph on the vertex set \( V_{i+1} = V_i - (B_i \cup C_i) \), where \( C_i = \Gamma(B_i) \cap V_i \), the neighbor set of \( B_i \) in the subgraph \( G_i \). Repeat if \( G_{i+1} \neq \emptyset \) and set \( k = i \) if \( G_{i+1} = \emptyset \). Then we call \( B = \{(B_1, C_1), \cdots, (B_k, C_k)\} \) the bottleneck decomposition of \( G \), \( \alpha_i \) the \( i \)-th \( \alpha \)-ratio and \( \alpha_i = \frac{w(C_i)}{w(B_i)}: i = 1, 2, \cdots, k > \alpha \)-ratio vector.

As stated in [Wu and Zhang, 2007], the problem of computing the bottleneck decomposition can be solved by the parametric maximum flow algorithm in a polynomial time. Further, they also derived some properties of the bottleneck decomposition as follows.

Proposition 3 ([Wu and Zhang, 2007]) Given graph \( G \), the bottleneck decomposition of \( G \) is unique and

1. \( 0 < \alpha_1 < \alpha_2 < \cdots < \alpha_k \leq 1 \);
2. if \( \alpha_i = 1 \), then \( i = k \) and \( B_i = C_i \); otherwise \( B_i \) is an independent set and \( B_i \cap C_i = \emptyset \);
3. if \( B \subseteq V_i \) and \( C = \Gamma(B) \cap V_i \), then \( w(C) \setminus w(B) \geq \alpha_i \), where if the equality holds, then \( B \subseteq B_i \) and \( C \subseteq C_i \).

For the third claim in Proposition 3, we name such a pair of \( (B, C) \) with \( B \subseteq V_i \) and \( C = \Gamma(B) \cap V_i \) as a candidate pair in \( V_i \) for convenience.

2.2 BD Mechanism

Given bottleneck decomposition, an allocation mechanism [Wu and Zhang, 2007] can be determined by distinguishing three cases. For convenience, we call such an allocation mechanism BD Mechanism.

BD Mechanism:

- For \( \alpha_i < 1 \), consider the bipartite graph \( \hat{G}_i = (B_i, C_i; E_i) \) where \( E_i = (B_i \times C_i) \cap E \). By Proposition 3-(2), \( B_i \) is an independent set in \( G \). Let \( \hat{f}_{uv} \) be the amount of bandwidth that vertex \( u \in B_i \) upload to \( v \in C_i \) along edge \( (u, v) \in E_i \). By the max-flow min-cut theorem, there exists flow \( \hat{f}_{uv} \geq 0 \) for \( u \in B_i \) and \( v \in C_i \) such that \( \sum_{v \in \Gamma(u) \cap C_i} \hat{f}_{uv} = w_u \) and \( \sum_{u \in \Gamma(v) \cap B_i} \hat{f}_{uv} = w_v / \alpha_i \). Let \( \hat{f}_{uv} = \alpha_i \hat{f}_{uv} \) which means that \( \sum_{u \in \Gamma(v) \cap B_i} \hat{f}_{uv} = w_v \).
- When \( B_k = C_k \) with \( \alpha_k = 1 \), similarly we construct a bipartite graph \( \hat{G}_k = (B_k, C'_k; E'_k) \) such that
$B_k'$ is a copy of $B_k$. There is an edge $(u, v') \in E'[k]$ if and only if $(u, v) \in E[B_k]$. By Hall’s theorem, for any edge $(u, v) \in E[B_k]$, there exists flow $f_{uw}$ such that $\sum_{v' \in \Gamma(u) \cap B_k'} f_{uv'} = w_u$ and $f_{uw'} = f_{uw}$. Let $f_{uw} = f_{uv}$. 

- For any other edge, $(u, v) \notin B_k \times C_i$, $i = 1, 2, \ldots, k$, define $f_{uv} = 0$.

It is clear that BD Mechanism assigns all resource of each agent to its neighbors from the same pair, that is all available resources exchanged along edges in $B_k \times C_i$, $i = 1, \ldots, k$. In terms of “fairness” and “efficiency”, BD Mechanism is exactly a proportional sharing mechanism.

**Definition 4 (Proportional Sharing Mechanism)** For each vertex $u$, the allocation $(f_{uw} : v \in \Gamma(u))$ of its resource $w_u$ is proportional to what it receives from its neighbors $(w_v \cdot f_{uv} : v \in \Gamma(u))$. That is $f_{uw} = \frac{\sum_{v \in \Gamma(u)} f_{uw}}{\sum_{v \in \Gamma(u)} f_{uw}}$.

**Proposition 5 ([Wu and Zhang, 2007])** BD Mechanism is a proportional sharing mechanism.

As stated before, the P2P system can be modeled an exchange economy where each agent $u$ sells its own divisible resource and use the money earned through trading to buy its neighbors’ resource. An efficient allocation in such an exchange economy can be characterized by the market equilibrium. Given a bottleneck decomposition, if a price vector $P$ is well defined as: $p_u = \alpha_i w_u$, if $u \in B_i$; and $p_u = w_u$ otherwise, then such a price vector combining the allocation $F$ obtained from BD Mechanism satisfying

**Proposition 6 ([Wu and Zhang, 2007])** $(PF)$ is a market equilibrium, where $F = (f_{uw})$ is obtained from BD Mechanism.

### 2.3 Resource Exchange Game

From a system design point of view, though BD Mechanism shall allocate resources among interconnected participants fairly and efficiently it is unknown whether an agent is willing to follow such a distributed network mechanism at the execution level. Can agents make strategic moves for gains in their utilities? Specific to the resource exchange model, the resources all agents have are their private information. They may manipulate BD Mechanism by misreporting the resource they own. We call such a problem with incentive factors the resource exchange game.

In the resource exchange game, let the resource amount reported by agent $u$ be $x_u \in (0, w_u]$. The reason why $x_u$ cannot exceed the true bandwidth $w_u$ is that each agent must upload all its reported resource to its neighbors. The collection $x = (x_1, x_2, \ldots, x_n)$ is referred to as the weight profile and $x_{-u} = (x_1, \ldots, x_{u-1}, x_{u+1}, \ldots, x_n)$ is the weight profile without agent $u$. Thus $x = (x_u, x_{-u})$. Let the bottleneck decomposition be $B(x) = \{ (B_i(x), C_i(x)), \ldots, (B_n(x), C_n(x)) \}$ and BD Mechanism outputs an allocation $F(x)$ based on the given weight profile $x$. The utility of agent $u$ written as $U_u(x)$ is [Wu and Zhang, 2007]:

\[
U_u(x) = \begin{cases} 
  x_u \cdot \alpha_i(x), & u \in B_i(x); \\
  x_u / \alpha_i(x), & u \in C_i(x),
\end{cases}
\]

where $\alpha_i(x)$ is the $i$-th $\alpha$ ratio according to weight profile $x$.

For mechanism design, the notion of truthfulness is perhaps the most important concept.

**Definition 7** A mechanism is truthful if no agent can benefit strictly from misreporting its resource amount irrespective of what is reported by other agents. Formally, given an agent $u$ and profile $x = (x_u, x_{-u})$, it holds that for any $x_{-u}$

\[
U_u(w_u, x_{-u}) \geq U_u(x_u, x_{-u}).
\]

It is obvious that truthfully reporting resource amount $w_u$ is the dominant strategy for each agent $u$ in the resource exchange game, if a mechanism is truthful.

### 3 Truthfulness of BD Mechanism

As we know, the bottleneck decomposition of $G$ depends on the structure of network and the resource all agents have. So for any agent $u$ and any weight profile $x_{-u}$, the bottleneck decomposition $B$ of $G$ shall change with the reported resource $x_u$. Thus $B$ can be viewed as a function of $x_u$ if $x_{-u}$ is fixed. On the other hand, we observe that bottleneck decomposition of $G$ could be the same when $x_u$ falls in an interval. Based on such an observation, we partition interval $(0, w_u]$ into several disjoint subintervals $\{(a_i, b_i)\}$, and construct a series of bottleneck decompositions $\{B_i^x\}$ such that when $x_u \in (a_i, b_i)$, $B(x_u, x_{-u}) = B^i = \{ (B_1^i, C_1^i), \ldots, (B_k^i, C_k^i) \}$. We, in particular, assume that $0 < a_i \leq b_i = a_{i+1} \leq b_{i+1}$. We use symbol "($\cdot$)" to denote the subinterval, because $(a_i, b_i)$ could be one of five forms $[a_i, b_i)$, $(a_i, b_i)$, $(a_i, b_i]$, $(a_i, b_i)$ and $b_i$. If $a_i < b_i$, then $(a_i, b_i)$ contains exactly one point which can be viewed as a special closed interval. Further, in order to keep the one-to-one correspondence between the subintervals and the bottleneck decompositions, we use $a_i$ and $b_i$ to denote the left and right endpoints of the $i$-th subinterval, respectively. Thus for any two adjacent interval $(a_i, b_i)$ and $(a_{i+1}, b_{i+1})$, there are only two cases: 1. $(a_i, b_i)$ and $(a_{i+1}, b_{i+1})$; 2. $(a_i, b_i)$ and $(a_{i+1}, b_{i+1})$; depending on which interval contains the break point $a_{i+1} = b_i$.

In the sequel, we first characterize the pairs agent $u$ belongs to in adjacent $B^i$ and $B^{i+1}$ in Subsection 3.1. Then the monotonicity of utility functions is derived by such useful characterizations in Subsection 3.2.

#### 3.1 Characterization of $\{B^i\}$

In order to show the structure properties of bottleneck decomposition $B$ in-depth, we redefine the bottleneck decomposition in more detail below.

**Definition 8 (Bottleneck Decomposition)** Let $B^i = \{ (B_1^i, C_1^i), \ldots, (B_k^i, C_k^i) \}$ be the bottleneck decomposition of graph $G$ when $x_u \in (a_i, b_i)$ and let the $\alpha$-ratio of $(B_j^i, C_j^i)$ be $\alpha_j^i = w(C_j^i)/w(B_j^i)$, $j = 1, \ldots, k_i$. For pair $(B_j^i, C_j^i)$ with $\alpha_j^i < 1$, each vertex in $B_j^i$ (or $C_j^i$) is called a $B$-class (or $C$-class) vertex. For the special case $B_k^i = C_k^i$, i.e., $\alpha_k^i = 1$, all vertices in $B_k^i$ are categorized as both $B$-class and $C$-class. Define $V_j^i = V$, $V_j^{i+1} = V_j^i - (B_j^i \cup C_j^i)$ for $j = 1, \ldots, k_i - 1$ and $G_j^i$ for the induced subgraph on $V_j^i$. 

189
Note that a vertex in $V^i_{\ell}$ with $\alpha_j \leq 1$ could simultaneously be $B$-class and $C$-class, in the case $B^i_{\ell} = C^i_{\ell}$. Moreover to highlight the $\alpha$-ratio of pairs $(B^i_{\ell}, C^i_{\ell})$ containing agent $u$, we denote it $\alpha_j^i(x_u)$. Further, if $V^i_{h} = V^i_{h+1}$ for some index $h$, then the maximal bottleneck pair in $V^i_{h}$ or $V^i_{h+1}$ just is a candidate pair in the other. So Proposition 3-(3) can be applied on those candidate pairs. In the subsequent discussions, such a technique will be used repeatedly.

In the following, we assume that $u \in B^i_{\ell} \cup C^i_{\ell}$ and $u \in B^i_{\ell+1} \cup C^i_{\ell+1}$ in $B^i$ and $B^{i+1}$ respectively. As we know, the bottleneck decomposition of $G$ changes because of the reported resource of agent $u$. Agent $u$’s strategic move is only able to influence such pairs which are decomposed after pairs that $u$ is in. But those with index less than $j$ and $\ell$ still keep the same. So

**Lemma 9** For any agent $u$, if $u \in B^i_{\ell} \cup C^i_{\ell}$ and $u \in B^i_{\ell+1} \cup C^i_{\ell+1}$ in $B^i$ and $B^{i+1}$ respectively, then $V^i_{h} = V^i_{h+1}$ for each $h = 1, 2, \ldots, \min(j, \ell)$.

Since agent $u$ may be a $B$-class vertex or $C$-class vertex in $B^i$ and $B^{i+1}$ respectively, there are totally 4 cases: $u \in B^i_{\ell} \cap C^i_{\ell+1} \cup C^i_{\ell+1} \cup B^i_{\ell+1}$, $u \in B^i_{\ell+1} \cap B^i_{\ell+1}$ and $u \in C^i_{\ell} \cap C^i_{\ell+1}$ respectively.

Specially for the first two cases, we assume that $\alpha_j^i(x_u) < 1$ if $x_u \in \{a_i, b_i\}$ and $\alpha_j^i(x_u) < 1$ if $x_u \in \{a_{i+1}, b_{i+1}\}$. This assumption is reasonable. For example, if $u \in B^i_{\ell} \cap C^i_{\ell+1}$ and $\alpha_j^i(x_u) = 1$ when $x_u \in \{a_{i+1}, b_{i+1}\}$, then $u$ can be viewed as a $C$-class vertex and $B^i_{\ell}$ can be rewritten as $C^i_{\ell}$, which makes such a case be that of $u \in C^i_{\ell} \cap C^i_{\ell+1}$.

**Lemma 10** For bottleneck decomposition $B^i$ and $B^{i+1}$, it is impossible that $u \in B^i_{\ell} \cap C^i_{\ell+1}$ and $u \in C^i_{\ell} \cap B^i_{\ell+1}$.

**Proof:** (sketch) For simplicity, we only show the impossibility of $u \in B^i_{\ell} \cap C^i_{\ell+1}$ and discuss the case that the corresponding intervals of $B^i$ and $B^{i+1}$ have the forms of $\langle a_i, b_i \rangle$ and $\langle a_{i+1}, b_{i+1} \rangle$. There are two cases that $j < \ell$ and $j \geq \ell$. If $j < \ell$, then $V^i = V^i_{\ell+1}$ by Lemma 9. If $x_u = b_i$, then the bottleneck decomposition is $B^i$ and $(B^i_{\ell+1}, C^i_{\ell+1})$ is a candidate pair of $V^i$, which induces $w(C^i_{\ell+1})/w(B^i_{\ell+1}) = w(C^i_{\ell+1})/w(B^i_{\ell+1})$. The proof for the second one is similar. At this point, the bottleneck decomposition of $G$ is $B^i$ and the candidate role of $(B^i_{\ell+1}, C^i_{\ell+1})$ in $V^i$ keeps that

$$\alpha_{i+1}^i(b_i) = \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1} \setminus \{u\}) + w(B^i_{\ell+1})} \geq \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1})}.$$  

\[(4)\]

If the strict inequality in (4) holds, then there must exist a positive number $\epsilon > 0$ such that $b_i + \epsilon = a_{i+1} + \epsilon \in \langle a_{i+1}, b_{i+1} \rangle$ and

$$\alpha_{i+1}^i(b_i + \epsilon) = \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1} \setminus \{u\}) + (b_i + \epsilon)} \geq \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1})}. \quad (5)$$

As we know, once weight $x_u$ increases up to $b_i + \epsilon = a_{i+1}$ and $\epsilon \in \langle a_{i+1}, b_{i+1} \rangle$, the bottleneck decomposition shall change to be $B^{i+1}$. As a candidate pair in $V^i_{\ell+1}$, the ratio of $(B^i_{\ell+1}, C^i_{\ell+1})$ can not be strictly less than $\alpha_{i+1}^i(b_i + \epsilon)$ which induces the impossibility of (5) and the strict inequality of (4). So

$$\alpha_{i+1}^i(b_i) = \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1} \setminus \{u\}) + w(B^i_{\ell+1})} = \frac{w(C^i_{\ell+1})}{w(B^i_{\ell+1})}. \quad (6)$$

Then $B^{i+1} \subseteq B^i_{\ell+1}$ by Proposition 3-(3) and $u \in B^i_{\ell+1}$, because $u \in B^i_{\ell+1}$. Furthermore, we confirm that $B^i_{\ell+1} \subset B^i_{\ell+1}$. Otherwise the pairs containing $u$ are the same in $B^i$ and $B^{i+1}$ which leads to $B^i = B^{i+1}$. It’s contradicts the partition of $(0, w_u)$.
Next, we shall show that $B^i_j = B^{i+1}_j \cup B^{i+1}_k$. For this purpose, let us define $B^c_i = B_i - B^{i+1}_i \neq \emptyset$ and $C^c_i = C_i^j - C^{i+1}_j \neq \emptyset$. We can see that $(B^c_i, C^c_i)$ is a candidate pair in $V^c_{i+1}$. Moreover, we have

$$\frac{w(C^c_i)}{w(B^c_i)} = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j \setminus \{u\})} + b_i = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j)}$$

where the first equality comes from (6), the second one is from a simple arithmetic calculation and Proposition 3-3 promises the last inequality. Now let us discuss the inequality

$$\frac{w(C^{i+1}_{i+1})}{w(B^{i+1}_j \setminus \{u\})} + b_i \geq \frac{w(C^{i+1}_{i+1})}{w(B^{i+1}_j)} = \alpha^{i+1}_i$$

If the strict inequality of (8) is right, then we can also find a positive number $\epsilon > 0$ such that $b_i + \epsilon = a_{i+1} + \epsilon \in (a_{i+1}, b_{i+1})$ and

$$\alpha^{i+1}_i (b_i + \epsilon) = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j \setminus \{u\})} + (b_i + \epsilon) > \frac{w(C^{i+1}_j)}{w(B^{i+1}_j)}$$

It is easy to see that the bottleneck decomposition of $G$ is $B^{i+1}$ when $x_u = b_i + \epsilon = a_{i+1} + \epsilon \in (a_{i+1}, b_{i+1})$. So by the increasing monotonicity property of $\alpha$-ratios with indexes, we have $\alpha^{i+1}_i (b_i + \epsilon) < \alpha^{i+1}_i$ which illustrates the incorrectness of (9). So the equality of (8) holds and

$$\frac{w(C^c_i)}{w(B^c_i)} = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j \setminus \{u\})} + b_i = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j)}$$

Therefore $B^c_i \subseteq B^{i+1}_j$ by Proposition 3-3. On the other hand, in order to keep the maximality of $B^{i+1}_j$ size, there is only unique possibility that $B^c_i = B^{i+1}_j \cup B^{i+1}_k$.

Note that, with the increasing of $x_u$ from $\langle a_i, b_i \rangle$ to $\langle a_{i+1}, b_{i+1} \rangle$, we can visualize the operations as split and combine to describe the changes of pairs containing $u$. Naturally, we can imagine that there are similar results for the case that $u \in C^i_j \cap C^{i+1}_j$.

**Lemma 13** For bottleneck decomposition $B^i$ and $B^{i+1}$ and $u \in C^i_j \cap C^{i+1}_j$,

1. if corresponding adjacent intervals have the forms as $\langle a_i, b_i \rangle$ and $\langle a_{i+1}, b_{i+1} \rangle$, then $u \in C_{i+1}_j$, $B^i_j = B^{i+1}_j \cup B^{i+1}_j$ and $C^i_j = C^{i+1}_j \cup C^{i+1}_j$.
2. if corresponding adjacent intervals have the forms as $\langle a_i, b_i \rangle$ and $\langle a_{i+1}, b_{i+1} \rangle$, then $u \in C_{i+1}^j$, $B^{i+1}_j = B^i_j \cup B^{i+1}_j$ and $C^{i+1}_j = C^i_j \cup C^{i+1}_j$.

### 3.2 Monotonicity of utility function on $(0, w_u)$

For the sake of convenience, we use $U_u(x_u)$ and $\alpha_u(x_u)$ to denote the utility function and the $\alpha$-ratio function of agent $u$ for any given $x_u$ respectively, where $x_u \in (0, w_u)$. As before shown in (1), $U_u(x_u) = x_u - \alpha_u(x_u)$ if $u \in B$-class; otherwise $U_u(x_u) = x_u/\alpha_u(x_u)$.

First, we discuss the monotone property of $U_u(x_u)$ in any single subinterval $(a_i, b_i)$. Then the bottleneck decomposition of $G$ is $B^i = \{(B^i_1, C^i_1), \cdots, (B^i_k, C^i_k)\}$ when $x_u \in (a_i, b_i)$. So agent $u$ may be in $B^j_j$ or $C^j_j$, $j = 1, 2, \cdots, k^i$. To be specific, its $\alpha$-ratio is

$$\alpha_u(x_u) = \alpha^j_j(x_u) = \{ \begin{cases} \frac{w(C^j_j)}{x_u + w(B^j_j \setminus \{u\})}, & u \in B^j_j; \\ \frac{w(C^j_j)}{x_u + w(C^j_j \setminus \{u\})}, & u \in C^j_j. \end{cases} \}$$

It is easy to observe that the $\alpha$-ratio of $u$ is monotonically increasing if it is a $C$-class vertex; and monotonically decreasing, otherwise. The utility function of agent $u$ is

$$U_u(x_u) = \{ \begin{cases} \frac{x_u w(C^j_j)}{x_u + w(B^j_j \setminus \{u\})}, & u \in B^j_j; \\ \frac{x_u w(C^j_j)}{x_u + w(C^j_j \setminus \{u\})}, & u \in C^j_j, \end{cases} \}$$

which is continuous on $\langle a_i, b_i \rangle$ and derivable on $(a_i, b_i)$. Furthermore, the derivative function of $U_u(x_u)$ can be computed easily, which is nonnegative on $\langle a_i, b_i \rangle$. Combining the continuity of $U_u(x_u)$ on $\langle a_i, b_i \rangle$, we have

**Lemma 14** For any agent $u$, any weight profile $x_u$, and any single interval $\langle a_i, b_i \rangle \subseteq (0, w_u)$, utility function $U_u(x_u)$ is monotonically nondecreasing on $x_u \in (a_i, b_i)$.

Next we shall challenge the monotonicity of $U_u(x_u)$ on the whole interval $(0, w_u)$. One of the biggest difficulties we are facing is that the $\alpha$-ratio function and the utility function of agent $u$ may vary with the bottleneck decompositions. If we can investigate the continuity of $\alpha$-ratio and utility function of agent $u$ at each break point, the monotone property of $U_u(x_u)$ can be achieved by Lemma 14.

**Lemma 15** For bottleneck decomposition $B^i$ and $B^{i+1}$ and $u \in B^i_j \cap B^{i+1}_j$ or $u \in C^i_j \cap C^{i+1}_j$, the utility function $U_u(x_u)$ is continuous at break point.

**Proof:** Here we only show the continuity of $U_u(x_u)$ at break point $x_u = b_i$ for the case that $u \in B^i_j \cap B^{i+1}_j$ and the adjacent intervals have the forms as $\langle a_i, b_i \rangle$ and $\langle a_{i+1}, b_{i+1} \rangle$. Under this case, Lemma 12 promises $u \in B^i_j$. So $\alpha_u(x_u) = \alpha^i_j(x_u)$ if $x_u \in \langle a_i, b_i \rangle$ and $\alpha_u(x_u) = \alpha^{i+1}_j(x_u)$ if $x_u \in \langle a_{i+1}, b_{i+1} \rangle$. Furthermore, equation (6) in the proof of Lemma 12 tells us that if $x_u = b_i$

$$\alpha^i_i(b_i) = \frac{w(C^i_j)}{w(B^i_j \setminus \{u\})} + b_i = \frac{w(C^{i+1}_j)}{w(B^{i+1}_j \setminus \{u\})} + b_i = \alpha^{i+1}_i(b_i)$$

Therefore, the continuity of $\alpha$-ratio function at break point $x_u = b_i$ holds as

$$\lim_{\epsilon \rightarrow 0^+} \alpha_u(b_i + \epsilon) = \lim_{\epsilon \rightarrow 0^+} \alpha^{i+1}_i(b_i + \epsilon) = \alpha^{i+1}_i(b_i) = \alpha^i_i(b_i)$$

And the utility function is also continuous at $x_u = b_i$,

$$\lim_{\epsilon \rightarrow 0^+} U_u(b_i + \epsilon) = \lim_{\epsilon \rightarrow 0^+} (b_i + \epsilon) \cdot \alpha^{i+1}_i(b_i + \epsilon) = b_i \cdot \alpha^i_i(b_i) = U_u(b_i).$$

Combining the monotone property in each subinterval and the continuity at each break point, the monotonicity of $U_u(x_u)$ on whole interval $(0, w_u)$ is derived directly.
Corollary 16 For any agent \( u \) and any given weight profile \( x_u \), utility function \( U_u(x_u) \) is monotonically nondecreasing on \( (0, w_u] \).

Obviously, the monotone property of \( U_u(x_u) \) in Corollary 16 for any given \( x_u \) ensures the correctness of \( U_u(w_u, x_u) = U_u(x_u, x_u) \), where \( x_u \in (0, w_u] \). So the main result of this paper is deduced directly.

Theorem 17 BD Mechanism is truthful for the resource sharing game.

Remarks: The strategy of weight misreporting cannot be replaced by the strategy of edge cutting [Cheng et al., 2015].

Let us consider an example in which three agents have the same weights. Obviously if one edge is cut, the resources can only be allocated along the remaining edges. But if one agent plays the strategy of weight misreporting, the resources are still be allocated on each edge whatever \( x_u \in (0, w_u] \).

4 Numerical Example

In this section, we analyze a representative numerical example to have an intuitive understanding on the above results. Consider the network of Fig. 1. Each cycle represents a vertex and the number in the cycle is depicted the vertex’s weight. As stated before, the bottleneck decomposition of \( G \) shall change with agent \( u \)’s reporting weight \( x_u \) for any given \( x_u \). So the interval \( (0, w_u] \) is partitioned into different disjoint subintervals, each \( \langle a_i, b_i \rangle \) corresponding to a decomposition \( B_i \). For the example shown in Fig. 1, Table 1 lists all subintervals and all pairs containing \( u \) in different subintervals.

![Figure 1: A network with 12 vertices where \( v_5 \) is the agent \( u \) who has a strategic move to misreport its own resource.](image)

From Table 1, there are some important informations worthy of note. First, \( u \) is in the same classes in adjacent \( B_i \) and \( B_i+1 \) that verifies Lemma 10. The relation between adjacent \( B \)-sets or \( C \)-sets is that of containing and contained which has been proved in Lemma 12 and 13. Second in non-adjacent decompositions, \( u \) may be in different classes. For example, when \( x_u \in [1, 3] \), \( u \) is in \( C \)-class with \( \alpha_u(x_u) = (2 + x_u)/15 < 1 \). And when \( x_u \in [16, 32] \), then \( u \) is a \( B \)-class vertex in \( B_2 \) with \( \alpha_u(x_u) = 20/(8 + x_u) < 1 \). So there must be a crucial decomposition \( B_i \). In such a decomposition, \( u \) can be viewed as a \( B \)-class and a \( C \)-class vertex simultaneously which means that \( \alpha_u(x_u) = 1 \) if \( x_u \in \langle a_i, b_i \rangle \). Obviously in Table 1, \( x_u = 13 \) just is the turning point and the corresponding decomposition is the crucial one. Third, if the class containing \( u \) changes with \( x_u \in (0, w_u] \), then the changing process must be from \( C \)-class to \( B \)-class with the increasing of \( x_u \), not vice versa.

<table>
<thead>
<tr>
<th>( x_u )</th>
<th>( B )</th>
<th>( C )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 1)</td>
<td>{v_4}</td>
<td>{u}</td>
</tr>
<tr>
<td>[1, 3)</td>
<td>{v_2, v_4}</td>
<td>{v_1, v_3, u}</td>
</tr>
<tr>
<td>(3, 10.5)</td>
<td>{v_2, v_4, v_10, v_11, v_12}</td>
<td>{v_1, v_3, u, v_9}</td>
</tr>
<tr>
<td>10.5</td>
<td>{v_2, v_4, v_8}</td>
<td>{v_1, v_3, u, v_6, v_7}</td>
</tr>
<tr>
<td>(10.5, 13)</td>
<td>{v_2, v_4}</td>
<td>{v_1, v_3, u}</td>
</tr>
<tr>
<td>13</td>
<td>{v_1, v_2, v_3, v_4, u}</td>
<td>{v_1, v_2, v_3, v_4, u}</td>
</tr>
<tr>
<td>(13, 16)</td>
<td>{v_1, v_3, u}</td>
<td>{v_2, v_4}</td>
</tr>
<tr>
<td>[16, 32)</td>
<td>{v_1, v_3, u, v_8}</td>
<td>{v_2, v_4, v_6, v_7}</td>
</tr>
<tr>
<td>(32, 49)</td>
<td>{v_1, v_3, u}</td>
<td>{v_2, v_4, v_7, v_9}</td>
</tr>
<tr>
<td>49</td>
<td>{v_1, v_3, u, v_10, v_11, v_12}</td>
<td>{v_2, v_4, v_7, v_9}</td>
</tr>
</tbody>
</table>

Table 1: All subintervals, each corresponding one decomposition. The second and third column represent the pair \( (B, C) \) which \( u \) is in, where \( u \) is written in bold.

The left of Fig. 2 well illustrates the property of continuity and monotonically nondecreasing property of \( U_u(x_u) \). But for \( \alpha \)-ratio shown in the right of Fig. 2, \( \alpha_u(x_u) \) is monotonically increasing when \( x_u < 13 \) and it is monotonically decreasing when \( x_u > 13 \). This result just coincides with the facts that \( u \) is in \( C \)-class when \( x_u < 13 \) and in \( B \)-class when \( x_u > 13 \) shown in Table 1. In addition, \( \alpha_u(x_u) \) reaches its peak at \( x_u = 13 \) as \( \alpha_u(x_u) \leq 1 \).

![Figure 2: The figures of utility function \( U_u(x_u) \) and \( \alpha \)-ratio function \( \alpha_u(x_u) \).](image)

5 Conclusion

In this article, we discuss the issue of possible strategic manipulations of agents with respect to BD Mechanism for the application of resource exchange. Our work resolves an open problem on the strategic stability of a resource exchange protocol (i.e., the BD mechanism) from the mechanism design perspective.

We show that, no agent could gain by misreporting its resource amount in a BD Mechanism. Combining the work of Cheng, et al., [Cheng et al., 2015], we establish a strong incentive stability result that an agent could not improve its utility by cutting off any incident edge or by reporting less resource. We note that an agent could not benefit even from the combination of the above two strategies. If an agent does so, it is equivalent to a two-stage strategy. At the first stage, the agent cuts some of the adjacent edges. At the second stage, it decreases its weight in the updated network. The results in [Cheng et al., 2015] and in our paper show that the utility of this agent would be non-increasing. So this completes the research on truthfulness of the popular proportional mechanism on resource exchange.
Acknowledgments
This research was partially supported by the National Nature Science Foundation of China (No. 11301475, 61173011), and by the Research Grant Council of Hong Kong (ECS Project No. 26200314 and GRF Project No. 16213115).

References
[Swap] Swap. www.swap.com