Tight Incentive Analysis on Sybil Attacks to Market Equilibrium of Resource Exchange over General Networks

SUBMISSION 576

The Internet-scale peer-to-peer (P2P) systems usually build their success on distributed protocols. For example, the well-known BitTorrent network for resource exchange is based on the *proportional response protocol*, where each participant exchanges its resources with its neighbors in proportion to what it has received in the previous round. The dynamics of such a protocol has been proved to converge to a market equilibrium. On the other hand, it requires thorough incentive analysis to show the robustness of such protocol, as the distributed agents may strategically manipulate the system once they are able to benefit. Recent studies have developed strategyproofness results of the proportional response protocol against agent deviations in the forms of weight cheating and edge deleting. However, the protocol is not truthful against Sybil attacks, under which an agent may create several fictitious identities and control these fictitious identities to exchange resource with others. In this paper, we apply the concept of *incentive ratio* to measure how much the utility of a strategic agent in a market equilibrium can be improved by playing Sybil attacks. Similar to approximation ratio, price of anarchy and competitive ratio, a smaller incentive ratio means how close our solution is for an agent not to manipulate the system. We prove a tight incentive ratio of two for any agent launching Sybil attacks over general networks. The tight incentive ratio of two closes an open problem modeling the successful tit-for-tat protocol for Internet resource exchanging, and also presents a complete picture in this line of theoretical studies with real applications.

Manuscript submitted for review to the 23nd ACM Conference on Economics & Computation (EC'22).

1 INTRODUCTION

The Internet revolution has created opportunities for commercial innovations to rise rapidly. Most visibly, economies of sharing are designed, implemented and made successful crossing industries and country boundaries [Hamari et al., 2016, Ma et al., 2017]. Among them, BitTorrent jump-started its P2P system with a remarkable performance [Pouwelse et al., 2005], using only local information for its tit-for-tat strategy design to incentivize users of the service. The strategy is modeled as a *proportional response protocol* on an undirected network $G = (V, E)$. Each vertex $v \in V$ represents an agent and each edge $(u, v) \in E$ represents the exchanging link between agents u and v. The neighborhood of v is denoted by $\Gamma(v) = \{u \in V | (u, v) \in E\}$. The agent v owns $w_v > 0$ units of (idle) resource to upload in resource exchange system.

Definition 1.1 (Proportional Response Dynamics). Let $x_{vu}(t)$ denote the resource allocation from node v to u on the link (v, u) at round t. Initiate $x_{vu}(0) = w_v/d_v$, where d_v is the degree of v in G; for $t \geq 0$,

$$x_{vu}(t+1) = \frac{x_{uv}(t)}{\sum_{k \in \Gamma(v)} x_{kv}(t)} \cdot w_v. \tag{1}$$

Wu and Zhang [Wu and Zhang, 2007] proved that proportional response dynamics converges, and surprisingly, to a market equilibrium allocation. This connection between market equilibrium and the proportional response dynamics has had a series of followup works, attracting scholars from a diversity of research fields, including its relationship with distributed gradient decent algorithm for Fisher market [Birnbaum et al., 2011], tâtonnement in market equilibrium dynamics [Cheung et al., 2020, 2012, 2018], competitive, cooperative, and fair network services [Georgiadis et al., 2015], large-scale equilibrium computation [Gao and Kroer, 2020], bidding game market equilibrium [Brânzei et al., 2021] and market learning [Cheung et al., 2021].

Recently, studying the incentives of strategic behavior has become a trend in a number of game-theoretic scenarios, including adversarial learning [Ahmadi et al., 2021, Barreno et al., 2010, Lowd and Meek, 2005], Bayesian auction [Deng et al., 2020, Tang and Zeng, 2018], leader-follower equilibrium in normal-form games [Birmpas et al., 2021], Fisher market [Chen et al., 2016, 2012, 2011], and security games [Gan et al., 2019]. In particular, incentive analysis study of agent behaviors in distributed economic models will come to play an important role, such as proposed for P2P networks [Babaioff et al., 2007, Feldman and Chuang, 2005]. It was shown [Cheng et al., 2015, 2016] that no agent can gain more utility in the market equilibrium of the proportional response protocol by different kinds of manipulations, such as cutting exchanging edges or misreporting the amount of its resources.

However, under another kind of attack, Sybil attacks [Douceur, 2002], the market equilibrium of proportional response protocol has been shown to be not incentive-compatible [Chen et al., 2017]. When launching a Sybil attack, an agent creates multiple fictitious identities and redistributes its resource among them, such that it can control these identities to exchange resource with others for an increased utility value (formally defined in Definition 2.9).

To measure the benefit of unilateral strategic behavior from a single agent, the solution concept of *incentive ratio* [Chen et al., 2012, 2011] is applied, which is defined as the ratio of one's maximum utility by a Sybil attack to the utility when it plays truthfully. Studies have shown that the incentive ratios of the proportional response protocol against Sybil attacks are all no more than two on several special networks, such as trees, cliques and cycles [Chen et al., 2017, 2019, Cheng et al., 2020]. However, the question on general networks remains a challenge:

What is the matching bound of incentive ratio of proportional response protocol against Sybil attacks over general networks?

In this paper we settle this open problem by a tight bound of two for the incentive ratio on general networks.

THEOREM 1.2. *The incentive ratio of proportional response protocol against Sybil attacks over general networks is exactly two.*

1.1 Additional Related Work

The market equilibrium of proportional response dynamics has been characterized [Wu and Zhang, 2007] through a combinatorial structure over networks, called *Bottleneck Decomposition* (the formal definition is Definition 2.2). Bottleneck decomposition is similar to the insight of the combinatorial algorithm for computing market equilibrium in a linear Fisher market [Devanur et al., 2008], which is also the first polynomial time algorithm for these computation problems. There has been a series of works studying polynomial-time algorithms to compute a market equilibrium in linear Arrow-Debreu market and Fisher market [Devanur et al., 2008, Duan et al., 2016, Duan and Mehlhorn, 2015, Jain, 2007, Orlin, 2010, Ye, 2008], which implies the polynomial-time tractability of the market equilibrium of proportional response protocol.

In general, the market equilibrium price vector and associated allocations are determined by the information of participating agents, e.g., the endowments and the preference parameters of agents' utility function in both Fisher market and Arrow-Debreu market [Arrow and Debreu, 1954, Fisher, 2006], and the neighborhood and the amount of resources in the P2P network resource exchanging system. However, it turns out that there is a possibility for agents to manipulate market equilibria by strategically misreporting their private information. Several works have shown the non-truthfulness in Fisher markets with different types of utility functions [Adsul et al., 2010, Chen et al., 2016, 2012, 2011, Mehta et al., 2014]. Even worse, the Nash equilibrium in the market by agent manipulative behavior may result in bad price of anarchy [Brânzei et al., 2014], a measure [Koutsoupias and Papadimitriou, 2009] evaluating a competitive outcome in comparison with a social optimum outcome.

Although it has been shown that behaving truthfully is not a dominate strategy in market equilibrium, several positive results have been established [Chen et al., 2016, 2012, 2011] based on the solution concept of *incentive ratio*, proposed in [Chen et al., 2011]. In a spirit similar to approximate truthfulness [Archer et al., 2004, Kothari et al., 2005, Schummer, 2004], incentive ratio characterizes the quantity of utility improvement of any single strategic agent. Informally, it is defined as the ratio of the strategic agent's maximum utility by a strategic behavior to the utility of its truthful behavior. The incentive ratios of 2, 2, 2, $e^{1/e} \approx 1.445$ (respectively) have been established for Fisher markets with Linear, Leontief, Weak Gross Substitute, Cobb-Douglas utility functions (respectively) [Chen et al., 2016, 2012, 2011]. Similar to approximation ratio [Williamson and Shmoys, 2011], price of anarchy [Koutsoupias and Papadimitriou, 1999, Roughgarden and Tardos, 2002] and competitive ratio [Sleator and Tarjan, 1985], a smaller incentive ratio means how much close our solution is for an agent not to manipulate the system. We adopt the argument in [Chen et al., 2011] that a small incentive ratio implies "*no one can benefit much from (complicated) strategic considerations, even if one has complete information of the others.*"

Sybil attacks (also called False-Name attack) have been extensively studies in many scenarios, such as voting [Aziz et al., 2011, Wagman and Conitzer, 2014], auctions [Gafni et al., 2020, 2021, Iwasaki et al., 2010, Yokoo et al., 2004], social networks [Brill et al., 2016, Conitzer et al., 2010], matching [Todo and Conitzer, 2013], mechanism design [Todo et al., 2011]. It is commonly regarded as among the most damaging attack in P2P networks [Trifa and Khemakhem, 2014], since it can subvert the security of P2P networks "by creating a large number of pseudonymous identities, using them to gain a disproportionately large influence" [Wikipedia contributors, 2021]. Thus, a bounded incentive ratio for the proportional response protocol against Sybil attacks over general networks would be surprising. Once a strategic agent tries to launch a Sybil attack, it needs to make an incredible amount of communication effort to gather the full information of the game and design the complicated strategy. So the bounded incentive ratio, along with the limited knowledge of the strategic agent in this distributed environment, would discourage it from launching a Sybil attack.

1.2 Organization

In Section 2, we formally introduce the resource exchanging model, Sybil attacks on the resource exchanging system and the solution concept of incentive ratio. Section 3 provides the proof of our main theorem. To be specific, in Section 3.1 we present two key observations (Lemma 3.2 and Corollary 3.3), and then state the upper bounds for two cases, according to the two types of cheating agent respectively. In Section 3.3 and Section 3.4, we prove two central lemmas to obtain the tight upper bound. To demonstrate a good understanding of our proof, a numerical example in Section 3.5 is illustrated. In Section 4, we summarize this paper and discuss possible future work.

2 THE RESOURCE SHARING MODEL

This work studies a resource exchanging model on a P2P network, modeled on an undirected graph $G = (V, E)$. Recall that $v \in V$ represents an agent and $(u, v) \in E$ represents the exchanging edge between u and v. The neighborhood of v is denoted by $\Gamma(v) = \{u \in V | (u, v) \in E\}$. Each agent v owns $w_v > 0$ units of (idle) resource. Let $\boldsymbol{w} = (w_v)_{v \in V}$ be the weight profile of V and $\boldsymbol{w}_{-v}$ be that of $V \setminus \{v\}$.

Let x_{vu} be the amount of resource that agent v allocates to its neighbor u. The collection $X = (x_{vu})_{v \in V, u \in \Gamma(v)}$ is called an *allocation* on $(G; \boldsymbol{w})$. The utility of v from allocation X is $U_v(X) = \sum_{u \in \Gamma(v)} x_{uv}$, i.e., the total resource received from all neighbors.

Resource exchanging on P2P network can be represented as a *pure exchange market*: each agent sells its resource to neighbors for a revenue, which is used to buy resource from its neighbors. *Market equilibrium* is a well-known standard solution concept in exchange economies, which can characterize efficient allocations .

Definition 2.1 (Market Equilibrium). Let p_v be the total price for the resource w_v of agent v. Price vector $\boldsymbol{p} = (p_v)_{v \in V}$ along with allocation X is a market equilibrium in the exchange market, if and only if for any $v \in V$, the following conditions hold: (1) market clearance: $(w_v - \sum_{u \in \Gamma(v)} x_{vu}) \frac{p_v}{w_v} = 0$; (2) budget constraint: $\sum_{u \in \Gamma(v), w_u \neq 0} \frac{x_{uv}}{w_u} p_u \leq p_v$; (3) individual optimality: the allocation X maximizes $U_v(X) = \sum_{u \in \Gamma(v)} x_{uv}$ at the current price vector $\boldsymbol{p}$. Thus X is called a market equilibrium allocation.

By the result in [Wu and Zhang, 2007], a market equilibrium, that is the convergence of the proportional response protocol, could be characterized by a combinatorial structure called *bottleneck decomposition*. For a subset $S \subseteq V$, the weight of S is $w(S) = \sum_{v \in S} w_v$, and the neighborhood of S is defined as $\Gamma(S) = \cup_{v \in S} \Gamma(v)$. Note that $\Gamma(S) \cap S = \emptyset$ if and only if S is an independent set. Define $\alpha(S) = w(\Gamma(S))/w(S)$, named as the α-ratio of S. A set B is called a *bottleneck*, if $\alpha(B) = \min_{S \subseteq V} \alpha(S)$. In addition, the bottleneck with the maximal size, is called the *maximal bottleneck*. For a network $(G; \boldsymbol{w})$, there may be several bottlenecks but the maximal bottleneck is unique.

Definition 2.2 (Bottleneck Decomposition). For a network $(G; \boldsymbol{w})$, start with $V_1 = V$, $G_1 = G$, and $i = 1$. Find the maximal bottleneck B_i of G_i and define the neighbor set of B_i in G_i is C_i, i.e., $C_i := \Gamma(B_i) \cap V_i$. Let G_{i+1} be the induced subgraph $G[V_{i+1}]$, where vertex set $V_{i+1} = V_i - (B_i \cup C_i)$. Repeat if $G_{i+1} \neq \emptyset$, and set $k = i$ and stop the procession if $G_{i+1} = \emptyset$. $\mathcal{B} = \{(B_1, C_1), \cdots, (B_k, C_k)\}$

is named as the bottleneck decomposition of G, in which (B_i, C_i) is the i-th bottleneck pair and $\alpha_i = w(C_i)/w(B_i)$ is the α-ratio of (B_i, C_i).

Proposition 2.3. *([Wu and Zhang, 2007]) In a network $(G; \boldsymbol{w})$,*
(1) the bottleneck decomposition $\mathcal{B}$ of G is unique;
(2) $0 < \alpha_1 < \alpha_2 < \cdots < \alpha_k \leq 1$;
(3) if $\alpha_i = 1$, then $i = k$ and $B_i = C_i$, otherwise B_i is an independent set and $B_i \cap C_i = \emptyset$.

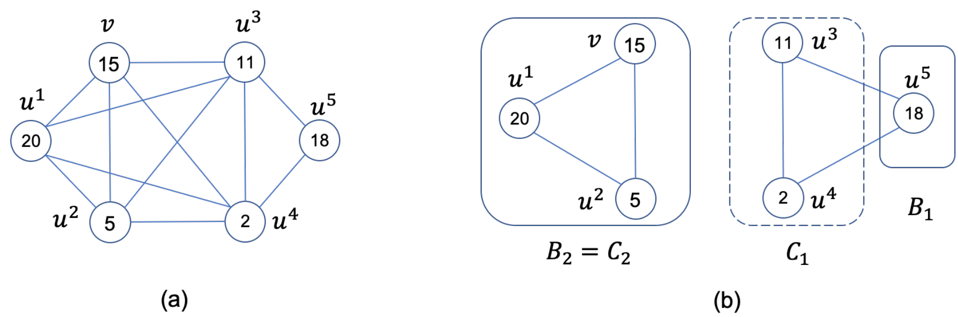

Fig. 1. (a) Network $(G; \boldsymbol{w})$; (b) the bottleneck decomposition of G.

To present a good understanding of bottleneck decomposition, we provide an example shown in Figure 1. The first bottleneck pair is $(B_1, C_1) = (\{u^5\}, \{u^3, u^4\})$ with $\alpha_1 = 13/18$. The second bottleneck pair is $(B_2, C_2) = (\{v, u^1, u^2\}, \{v, u^1, u^2\})$ with $\alpha_2 = 1$. Intuitively, $\alpha_1 < \alpha_2$, B_1 is an independent set, $B_1 \cap C_1 = \emptyset$, and $B_2 = C_2$ with $\alpha_2 = 1$, verifying Proposition 2.3-(2) and (3).

Based on the the bottleneck decomposition, all the vertices can be categorized into two classes.

Definition 2.4 (B-class and C-class). Let $\mathcal{B}$ be the bottleneck decomposition of the network $(G; \boldsymbol{w})$. For pair (B_i, C_i) with $\alpha_i < 1$, vertices in B_i (or C_i) are called B-class (or C-class) vertices. If the last pair is in the form $B_k = C_k$ with $\alpha_k = 1$, then all vertices in B_k are categorized both as B-class and C-class.

Given the bottleneck decomposition $\mathcal{B}$, an allocation can be explored with the help of the maximum flow on a constructed network, which is named as the *BD allocation mechanism.*

Definition 2.5 (BD Allocation Mechanism). Given the bottleneck decomposition $\mathcal{B}$, an allocation X (named as *BD allocation*) can be determined by distinguishing three cases.

• For (B_i, C_i) with $\alpha_i < 1$, consider the bipartite graph $\widehat{G} = (B_i, C_i; E_i)$ with $E_i = (B_i \times C_i) \cap E$. Construct a network $N = (V_N, E_N)$ with $V_N = \{s, t\} \cup B_i \cup C_i$ and directed edges (s, u) with capacity w_u for $u \in B_i$, (v, t) with capacity w_v/α_i for $v \in C_i$ and (u, v) with capacity $+\infty$ for $(u, v) \in E_i$. The max-flow min-cut theorem ensures a maximal flow $\{f_{uv}\}$, $u \in B_i$ and $v \in C_i$, such that $\sum_{v \in \Gamma(u) \cap C_i} f_{uv} = w_u$ and $\sum_{u \in \Gamma(v) \cap B_i} f_{uv} = w_v/\alpha_i$. Let $x_{uv} = f_{uv}$ and $x_{vu} = \alpha_i f_{uv}$.

• For (B_k, C_k) with $\alpha_k = 1$ $(B_k = C_k)$, construct a bipartite graph $\widehat{G} = (B_k, B'_k; E'_k)$ where B'_k is a copy of B_k. There is an edge $(u, v') \in E'_k$ if and only if $(u, v) \in E[B_k]$. Construct a network by the above method, for any edge $(u, v') \in E'_k$, there exists flow $f_{uv'}$ such that $\sum_{v' \in \Gamma(u) \cap B'_k} f_{uv'} = w_u$. Let $x_{uv} = f_{uv'}$.

• For any other edge $(u, v) \notin B_i \times C_i$ or $(u, v) \notin C_i \times B_i$, let $x_{uv} = 0$.

Proposition 2.6 ([Wu and Zhang, 2007]). *Proportional response dynamics converges to BD allocation X. In addition, if we set $p_v = \alpha_i w_v$, when $v \in B_i$; and $p_v = w_v$ when $v \in C_i$, then the price* *vector $p = (p_v)$ together with the BD allocation X is a market equilibrium in the resource exchanging problem. Each agent v's utility from X is $U_v = w_v \cdot \alpha_i$ if $v \in B_i$, $U_v = w_v/\alpha_i$ if $v \in C_i$.*

Definition 2.7 (α-ratio). For a network $(G;\boldsymbol{w})$ and $v \in V$, the α-ratio of v is denoted by α_v, where $\alpha_v = \alpha_i$ if $v \in B_i \cup C_i$.

Definition 2.8 (η-ratio). For a network $(G;\boldsymbol{w})$ and $v \in V$, the η-ratio of v is denoted by η_v, where $\eta_v = \alpha_v$ if $v \in B_i$ and $\eta_v = 1/\alpha_v$ if $v \in C_i$.

Notice that $U_v = w_v \cdot \eta_v$ holds according to Proposition 2.6 and Definition 2.8. So we also call η-ratio as "exchange ratio".

In the rest of paper, we would conduct the incentive analysis on Sybil attacks (whose formal definition is in subsequent subsection) to the market equilibrium allocation (i.e., the BD allocation).

2.1 Sybil Attack and Incentive Ratio

Although the allocation in a market equilibrium is fair and efficient, recently, the agent's incentives have been further studied. In particular, a strategic agent may launch a Sybil attack to manipulate the market equilibrium allocation for increased utilities. Several works [Chen et al., 2017, 2019, Cheng et al., 2020] show that an agent indeed benefits strictly from a Sybil attack on different special networks, including trees, cliques, and cycles, implying non-truthfulness of the proportional response protocol.

In this subsection, we present the definitions of Sybil attacks and incentive ratio formally. We provide a concrete example to illustrate there is a network on which one agent's utility by a Sybil attack is arbitrarily close to two times compared with its utility by truthful behavior. After this example, we also give some intuitions, which help the readers to understand why a strategic agent can benefit from a Sybil attack. This concrete example also gives a self-contained lower bound proof of our main theorem.

Definition 2.9 (Sybil Attack). Consider a network $(G;\boldsymbol{w})$ and a strategic agent v. When v plays a Sybil attack, it would split itself into m fictitious nodes $\{v^1, \cdots, v^m\}$, and assigns amount w_{v^i} of resource to each node v^i, satisfying $0 < w_{v^i} \leq w_v$ and $\sum_{i=1}^m w_{v^i} = w_v$. Each fictitious node may be adjacent to several neighbors of v's, but there are no connection between fictitious nodes.

The resulting network, called a Sybil network, is denoted by $\tilde{G} = (\tilde{V}, \tilde{E})$, in which the fictitious nodes set is $\Lambda = \{v^1, \cdots, v^m\}$ [1]. After playing Sybil attack, v obtains new utility, which is the sum of utilities from all fictitious nodes in $\tilde{G}$. We denote the utility of v by $U_v(\tilde{G}; w_{v^1}, \cdots, w_{v^m}, \boldsymbol{w}_{-v})$, i.e.,

$$U_v(\tilde{G}; w_{v^1}, \cdots, w_{v^m}, \boldsymbol{w}_{-v}) := \sum_{i=1}^{m} U_{v^i}(\tilde{G}; w_{v^1}, \cdots, w_{v^m}, \boldsymbol{w}_{-v}).$$

When launching a Sybil attack by a strategic agent v, we assume that the fictitious nodes connect to its original neighbors in $\Gamma(v)$. The main reason for this assumption is in the distributed system, each agent has limited knowledge about other participants, so that it only connects to its neighbors, and thus has its fictitious nodes connect these neighbors. In addition, we also assume that these fictitious nodes have no inner connections among themselves, since exchanging idle resource between any fictitious nodes neither brings utility to strategic agent v, nor increases the utility from neighbors.

Example 2.10. Consider a network G containing 4 vertices as shown in Figure 2-(a). The weights of all vertices are $w_v = w_{u^1} = w_{u^2} = w_{u^3} = 1$. In a BD allocation X from BD allocation mechanism, $x_{vu^1} = x_{u^1v} = 1$, $x_{u^2u^3} = x_{u^3u^2} = 1$, $x_{vu^2} = x_{vu^3} = x_{u^2v} = x_{u^3v} = 0$. Thus, $U_v(x) = 1$.

[1] In this paper, we use the notation $\tilde{\cdot}$ to distinguish elements in Sybil networks.

Now agent v strategically splits itself into two fictitious nodes v^1 and v^2, and assigns ϵ and $1-\epsilon$ to v^1 and v^2 respectively. The resulting Sybil network $\tilde{G}$ is shown in Figure 2-(c). In the corresponding BD allocation, $x_{v^1u^1} = \epsilon, x_{u^1v^1} = 1, x_{u^2u^3} = x_{u^3u^2} = (1+\epsilon)/2, x_{v^2u^2} = x_{v^2u^3} = x_{u^2v^2} = x_{u^3v^2} = (1-\epsilon)/2$. So, $U_{v^1} = 1, U_{v^2} = 1-\epsilon$, meaning $U_v = U_{v^1} + U_{v^2} = 2-\epsilon$. The total utility of v by the Sybil attack approaches 2 if ϵ is arbitrarily small, which is almost two times its original utility.

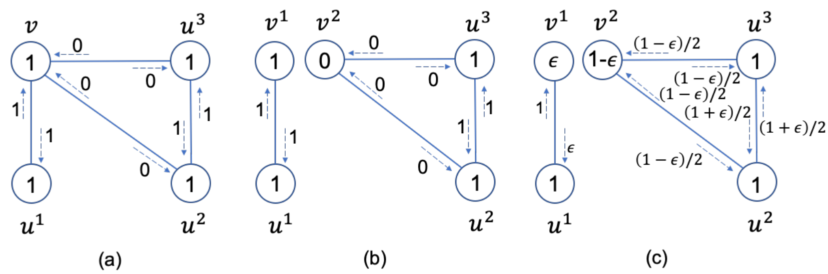

Fig. 2. Tight bound example.

Based on the example in Figure 2, we try to give both mathematical and economical intuitions about why splitting a node can be beneficial to the attacker. In the original network (Figure 2-(a)), v allocates all its resource to u^1 in return for all u^1's resource under the equilibrium. By splitting itself, v only needs to allocate arbitrary small amount of resource to u^1 while still obtains all u^1's resource. Thus, v is able to allocate the rest resource in exchange for other neighbors' resource. In this specific example, v may consider such a Sybil attack after knowing itself being the only neighbor of u^1 by incident. Economically speaking, each agent in a market has an unique price for its resource under the equilibrium, but by splitting itself into several fictitious nodes, an agent is able to set different prices when trading with different neighbors.

To efficiently characterize the extent to which utilities can be increased under BD allocation mechanism, the solution concept of *incentive ratio* is introduced, whose formal definition is as follows.

Definition 2.11 (Incentive Ratio). In a resource sharing game, given a network $(G; \mathbf{w})$, the incentive ratio of agent v under BD allocation mechanism against a Sybil attack is

$$\zeta_v = \max_{\substack{1 \le m \le d_v \\ w_{v^i} \in [0, w_v], i=1,2,\ldots,m, \sum_{i=1}^m w_{v^i} = w_v; \tilde{G}}} \frac{U_v\left(\tilde{G}; w_{v^1}, \ldots, w_{v^m}, \mathbf{w}_{-v}\right)}{U_v\left(G; \mathbf{w}\right)},$$

where $\tilde{G}$ and $(w_{v^1}, \ldots, w_{v^m}, \mathbf{w}_{-v})$ are the Sybil network and weight profile after playing a Sybil attack. The incentive ratio against Sybil attacks is $\zeta = \max_{G, \mathbf{w}, v \in V} \zeta_v$.

Example 2.10 well demonstrates that the incentive ratio of BD allocation mechanism against Sybil attack is at least two, showing the lower bound of incentive ratio is two. Previous works on the tight bound of incentive ratio have been done, however only limited to some special networks, such as trees, cliques, and rings. Thus what is the matching bound of incentive ratio on general networks is a big challenge. In this paper, our main theorem tightens up the incentive ratio against a Sybil attack over general networks.

3 THE PROOF OF MAIN THEOREM

In this section, we present all key observations and a sketch of the proof of our main theorem.

The lower bound directly follows from Example 2.10, while the proof for the upper bound of two is much more sophisticated and is shown in the following subsections. In Section 3.1 we present two key observations (Lemma 3.2 and Corollary 3.3), and then provide the upper bounds for two cases, according to the two types of cheating agent respectively. In Section 3.3 and Section 3.4, we prove two central lemmas to obtain the tight upper bound.

3.1 Resource Reserved Binary (RRB) Split.

We would like to introduce a special kind of Sybil attack, called *resource reserved binary (RRB) split* in this subsection, which leads a neat corollary (Corollary 3.3) simplifying our proof.

Definition 3.1 (Resource Reserved Binary (RRB) Split). Consider a BD allocation $X = (x_{uv})$ on network $(G; \boldsymbol{w})$. When playing resource reserved binary (RRB) split, the strategic agent first partitions its neighborhood $\Gamma(v)$ into two disjoint subsets: N_1 and N_2, and then splits itself into two nodes v^1 and v^2 along with weights w_{v^1} and w_{v^2}, such that v^ℓ is connected to each neighbor $u \in N_\ell$, and $w_{v^\ell} = \sum_{u \in N_\ell} x_{vu}$, $\ell = 1, 2$.

After playing RRB-split, the resulting network changes to $\tilde{G} = (\tilde{V}, \tilde{E})$, where $\tilde{V} = V \setminus \{v\} \cup \{v^1, v^2\}$ and $\tilde{E} = E \setminus (\cup_{u \in \Gamma(v)} (u, v)) \cup (\cup_{\ell=1}^{2} \cup_{u \in N_\ell} (u, v^\ell))$. Note that $\Gamma(v^\ell) = N_\ell$, $\ell = 1, 2$. Figure 2-(b) illustrates the resulting network after v playing RRB-split where $w_{v^1} = x_{vu^1} = 1$ and $w_{v^2} = x_{vu^3} = 0$. On this intermediate network, the neighbors u^1 and u^3 both receive the same amount of resource as before, and also return the same to v^1 and v^2 respectively, bringing the same utility to v after RRB-split. Clearly, there is a strong relationship between $\tilde{G}$ and G, inspiring us to derive a market equilibrium of $\tilde{G}$ directly from the one of G.

Lemma 3.2. *Consider a market equilibrium (p, X) on network $(G; \boldsymbol{w})$, where X is a BD allocation, and p is the corresponding price vector (Proposition 2.6). On the resulting network $(\tilde{G}; w_{v^1}, w_{v^2}, \boldsymbol{w}_{-v})$ after RRB-split by agent v, if we construct price vector $\tilde{p}$ and allocation $\tilde{X}$ as: $\tilde{p}_u = p_u$, if $u \neq v^1, v^2$, $\tilde{p}_{v^\ell} = \frac{w_{v^\ell}}{w_v} p_v$, $\ell = 1, 2$; and $\tilde{x}_{uw} = x_{uw}$, if $u, w \neq v^1, v^2$, and $\tilde{x}_{uv^\ell} = x_{uv}$, $\tilde{x}_{v^\ell u} = x_{vu}$ for each $u \in \Gamma(v^\ell)$, $\ell = 1, 2$; then we have*

(1) $(\tilde{p}, \tilde{X})$ is a market equilibrium of $\tilde{G}$;
(2) $U_u(\tilde{X}) = U_u(X)$ and $\eta_u(\tilde{X}) = \eta_u(X)$ if $u \neq v^1, v^2$;
(3) $U_{v^1}(\tilde{X}) + U_{v^2}(\tilde{X}) = U_v(X)$;
(4) if $w_{v^\ell} > 0$, $\eta_{v^\ell}(\tilde{X}) = \eta_v(X)$, $\ell = 1, 2$;
(5) if $w_{v^\ell} = 0$, $\eta_{v^\ell}(\tilde{X}) = \max\{1/\eta_u(X) | u \in N_\ell\} \leq \eta_v(X)$, $\ell = 1, 2$.

The key observation of Lemma 3.2 is that $(\tilde{p}, \tilde{X})$ is a market equilibrium of $\tilde{G}$.

Suppose that agent v obtains the maximal utility by splitting itself into m fictitious nodes and each node is connected to some of v's neighbors. If there is a fictitious node whose degree is more than one, then we let it play once RRB-split. Conduct such an operation successively until each node is a leaf. Lemma 3.2 guarantees the total utility of agent v remains the same maximal utility during the whole splitting process. Meanwhile, after this process, if there are several leaf fictitious nodes connecting to one same v's neighbor, then these nodes must be in the same bottleneck pair, the same class in precise, with the same α-ratio by the definition of bottleneck decomposition. This means that these nodes can be regarded as one single node with its weight being the summation of all these leaf nodes' weights.

So we have the following corollary.

Corollary 3.3. *Given the network $(G;\boldsymbol{w})$ and a strategic agent $v \in V$, the maximal incentive ratio of agent v by Sybil attack can be achieved by splitting into d fictitious nodes, each neighbor of v is connected to one node, where d is the degree of v on G.*

Corollary 3.3 simplifies the analysis for the incentive ratio and makes us only focus on how to assign the resource among d fictitious nodes. In the rest of this paper, we only need to consider this special Sybil attack by splitting into d fictitious nodes, each of them adjacent to one neighbor of v.

3.2 Two Main Lemmas

By Corollary 3.3, it is enough for us to discuss the specified strategy, in which the strategic agent v splits into d nodes, $\{v^1, \cdots, v^d\}$, and connects each v^i to one neighbor u^i.

Denote the specified Sybil network by $G^* = (V^*, E^*)$, where $\Lambda^* = \{v^1, \cdots, v^d\}$ and $V^* = V \setminus \{v\} \cap \Lambda^*$. Then our objective in the rest of this paper is to prove

$$U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) \leq 2 \cdot U_v(G;\boldsymbol{w}), \tag{2}$$

where $(w^*_{v^1}, \cdots, w^*_{v^d})$ is any feasible weight assignment satisfying $\sum_{i=1}^d w^*_{v^i} = w_v$. For the simplicity of notation, we use $\alpha^*_{v^i}$ and $\eta^*_{v^i}$ (respectively) to denote the α-ratio and η-ratio (respectively) of v^i on $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$.

The rest proof for the upper bound of two is from the following two lemmas, depending on whether v is in B-class or v is in C-class on the original network $(G;\boldsymbol{w})$.

Lemma 3.4. *If v is in B-class with $\alpha_v < 1$ on $(G;\boldsymbol{w})$, then*

$$U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) \leq 2 \cdot U_v(G;\boldsymbol{w}).$$

Lemma 3.5. *If v is in C-class with $\alpha_v \leq 1$ on $(G;\boldsymbol{w})$, then*

$$U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) \leq 2 \cdot U_v(G;\boldsymbol{w}).$$

In the proof of each lemma, we will design a multi-stage process to transform the initial network $(G;\boldsymbol{w})$ to the ultimate network $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$. Intuitively, the utility of strategic agent v after Sybil attack only depends on the ultimate weight assignment $(w^*_{v^1}, \cdots, w^*_{v^d})$. Then we will upper bound the changes of agent v's utility in each stage, and finally derive the overall upper bound. Next we introduce the processes for two cases of v is in B-class and v is in C-class separately. For better understanding, we also provide a resource exchange network example when an agent adopting Sybil attack in Appendix A, illustrating how these transformation processes look like.

3.3 Proof sketch of Lemma 3.4

For the case that v is a B-class vertex with $\alpha_v < 1$, Figure 3 illustrates the four-stage process to transform initial network $(G;\boldsymbol{w})$ with BD allocation X to the Sybil network $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$. The details are in Appendix A.3.

Stage 1: Pre-Processing (Figure 3 (a) to (b)). Partition v's neighborhood $\Gamma(v)$ on the initial network $(G;\boldsymbol{w})$ into two disjoint subsets: $\check{N} = \{u^i | x_{vu^i} > w^*_{v^i}\}$ and $\hat{N} = \{u^i | x_{vu^i} \leq w^*_{v^i}\}$, and play RRB-split (Definition 3.1) to split v into $\check{v}$ and $\hat{v}$, where $\Gamma(\check{v}) = \check{N}$ and $\Gamma(\hat{v}) = \hat{N}$. Let $(\tilde{G}^1; w_{\check{v}^1}, w_{\hat{v}^1}, \boldsymbol{w}_{-v})$ be the resulting network at the end of Stage 1, on which the fictitious node set is $\Lambda^1 = \{\check{v}^1, \hat{v}^1\}$, and $\boldsymbol{w}_{\Lambda^1} = (w_{\check{v}^1}, w_{\hat{v}^1})$. By Lemma 3.2, we split v into two fictitious nodes promising v's utility remains unchanged (Claim 1). Then we will decrease $w_{\check{v}^1}$ and increase $w_{\hat{v}^1}$.

Claim 1. $U_v(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}, \boldsymbol{w}_{-v}) = U_v(G;\boldsymbol{w})$.

Stage 2: Adjusting Technique (Figure 3 (b) to (c)). Informally, the Adjusting Technique is to decrease $w_{\check{v}^1}$ and increase $w_{\hat{v}^1}$ by a same value (denoted by z) as long as the exchange ratio of $\check{v}^1$

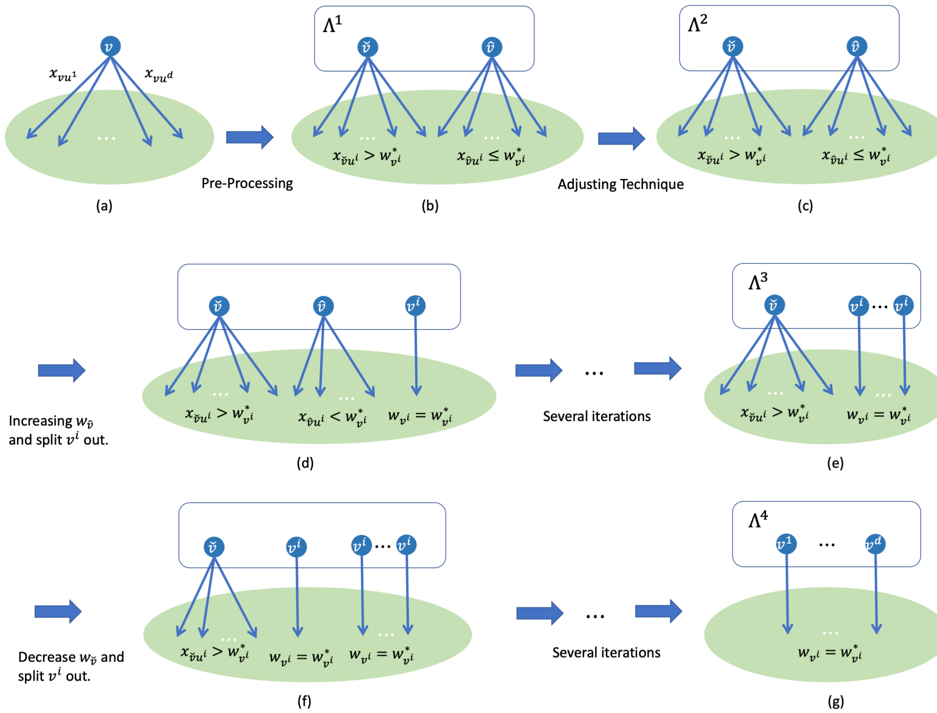

Fig. 3. Proof Sketch of Lemma 3.4

and $\hat{v}^1$ are the same, i.e., $\eta_{\check{v}^1} = \eta_{\hat{v}^1}$. Then we know that $\Delta U_v = -\eta_{\check{v}^1} \cdot z + \eta_{\hat{v}^1} \cdot z = 0$, so during this stage agent v's utility still remains unchanged (Claim 2). The resulting network at the end of Stage 2 is denoted by $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v})$, where Λ^2 contains $\check{v}^2$, $\hat{v}^2$, and some fictitious nodes with degree of one. Adjusting Technique is formally defined in Algorithm 3 in Appendix B.

Claim 2. $U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) = U_v(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}, \boldsymbol{w}_{-v}) = U_v(G; \boldsymbol{w})$.

Stage 3: Increasing Process (Figure 3 (c) to (e)). In this stage, we increase $w_{\hat{v}^2}$ and give an intuitive upper bound of v's utility. Note that our partition guarantees $x_{\hat{v}^2 u^i} \leq w^*_{v^i}$ for each u^i connected to $\hat{v}^2$. Let us increase $w_{\hat{v}^2}$ continuously. During this process, the condition of $\sum_{u^i \in \Gamma(\hat{v}^2)} x_{\hat{v}^2 u^i} = w_{\hat{v}^2}$ always holds, so there is at least one allocation $x_{\hat{v}^2 u^i}$ strictly increases. Once $w_{\hat{v}^2}$ increases to some threshold, at which there is one neighbor u^i satisfying $x_{\hat{v}^2 u^i} = w^*_{v^i}$, then $\hat{v}^2$ executes RRB-split to split v^i out. By the property of RRB-split (Lemma 3.2), $w_{v^i} = w^*_{v^i}$. Then we leave w_{v^i} unchanged after then (see Figure 3 (c) to (d)). Continue to increase $w_{\hat{v}^2}$ until there is another u^i satisfying $x_{\hat{v}^2 u^i} = w^*_{v^i}$, split it out, and then repeat. This iteration ends up with $\hat{v}^2$ being replaced by several leaves and the weight of each leaf being $w^*_{v^i}$ (see Figure 3 (d) to (e)). Let $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v})$ be the network at the end of Stage 3, where Λ^3 contains some fictitious nodes with degree of one and $\check{v}^3$.

Here is one key property of the case v is a B-class vertex with $\alpha_v < 1$ in the initial network $(G; \boldsymbol{w})$ (which does not hold for the case of v is a C-class vertex with $\alpha_v(G; \boldsymbol{w}) \geq 1$). Because v is a B-class vertex with $\alpha_v < 1$ in the initial network $(G; \boldsymbol{w})$, all fictitious nodes split in this stage would be in B-class. Furthermore, the α-ratios (which are also η-ratios) of all these fictitious nodes would $\leq \alpha_v(G; \boldsymbol{w})$ (which $= \eta_v(G; \boldsymbol{w})$). Such a property is illustrated intuitively by Table 1 in Appendix A.

Since v's utility does not change when RRB-split is executed, we only need to consider the changes of v's utility when $w_{\check{v}^2}$ increases. Indeed, v's utility increases in this stage (which is reasonable since the amount of resources of v increases), but we can prove an upper bound for the amount of increased utilities by the property above. The change of v's utility is formally given in the following claim.

Claim 3.

$$\begin{aligned} U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v}) \leq & \quad U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) + \eta_v(G; \boldsymbol{w})(w(\Lambda^3) - w(\Lambda^2)) \\ \leq & \quad U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) + \eta_v(G; \boldsymbol{w}) w_v \\ = & \quad U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) + U_v(G; \boldsymbol{w}). \\ \leq & \quad 2 \cdot U_v(G; \boldsymbol{w}). \end{aligned}$$

Stage 4: Decreasing Process (Figure 3 (e) to (g)). In this stage, we decrease $w_{\check{v}^3}$ and give an upper bound of v's utility. Note that during the Stage 3, $x_{\check{v}^2 u^i}$ remains unchanged and $\check{v}^2$ is exactly $\check{v}^3$. Thus, at the beginning of Stage 4, $x_{\check{v}^3 u^i} > w^*_{v^i}$. Similar to Stage 3, decrease $w_{\check{v}^3}$ until there is one u^i satisfying $x_{\check{v}^3 u^i} = w^*_{v^i}$, execute RRB-split to split v^i out from $\check{v}^3$, and repeat (see Figure 3 (e) to (f)). The iteration ends up with $\check{v}^3$ being replaced by several leaves and the weight of each of them being $w^*_{v^i}$ (see Figure 3 (f) to (g)). At the end of Stage 4, the network $(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v})$ is exactly the ultimate one $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$.

The key observation in this stage is the monotonicity of v'utility when the weight of $\check{v}$ is decreasing. To be specific, we can prove for any vertices set S contains $\check{v}$, i.e., $\check{v} \in S$, the total utility of S is non-increasing. Such a property could also be illustrated intuitively by Table 1 in Appendix A.

Similarly, v's utility remains unchanged when we execute RRB-split. So we have the following claim.

Claim 4. $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) = U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v}) \leq U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v}) \leq 2 \cdot U_v(G; \boldsymbol{w})$.

3.4 Proof sketch of Lemma 3.5

Comparing with the proof when v is a B-class vertex in $(G; \boldsymbol{w})$ with $\alpha_v < 1$, it is more complicated to obtain the upper bound of two for the case of v is a C-class vertex in $(G; \boldsymbol{w})$ with $\alpha_v \leq 1$. Intuitively, the main reason is the nice property in the stage 3 in Section 3.3 does not hold any more (but the property in the stage 4, the monotonicity of v' utility holds in general). The whole process in this subsection consists of six stages as follows.

Stage 1: Pre-Processing (Figure 4 (a) to (b)). Same as Stage 1 in Section 3.3, $\Gamma(v)$ on initial network $(G; \boldsymbol{w})$ is partitioned into two subsets: $\check{N} = \{u^i | x_{vu^i} > w^*_{v^i}\}$ and $\hat{N} = \{u^i | x_{vu^i} \leq w^*_{v^i}\}$, and v plays RRB-split to split into $\check{v}$ and $\hat{v}$. The resulting network after Stage 1 is denoted by $(\tilde{G}^1; w_{\check{v}^1}, w_{\hat{v}^1}, \boldsymbol{w}_{-v})$, on which the fictitious node set is $\Lambda^1 = \{\check{v}^1, \hat{v}^1\}$, and $\boldsymbol{w}_{\Lambda^1} = (w_{\check{v}^1}, w_{\hat{v}^1})$. Directly from the property of RRB-split (Lemma 3.2), we have

Claim 5. $U_v(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}, \boldsymbol{w}_{-v}) = U_v(G; \boldsymbol{w})$.

Stage 2: Adjusting Technique (Figure 4 (b) to (c)). This stage is also the same as Stage 2 in Section 3.3. The resulting network after Stage 2 is denoted by $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v})$, where Λ^2 contains $\check{v}^2$, $\hat{v}^2$, and some fictitious nodes with degree of one. During this stage, v's utility remains unchanged.

Claim 6. $U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) = U_v(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}, \boldsymbol{w}_{-v}) = U_v(G; \boldsymbol{w})$.

Stage 3: Decreasing Process (Figure 4 (c) to (d)). In this stage, we decrease $w_{\check{v}^2}$ and utilize the monotonicity of v's utility. During the decreasing process at least one allocation $x_{\check{v}^2 u}$ decreases,

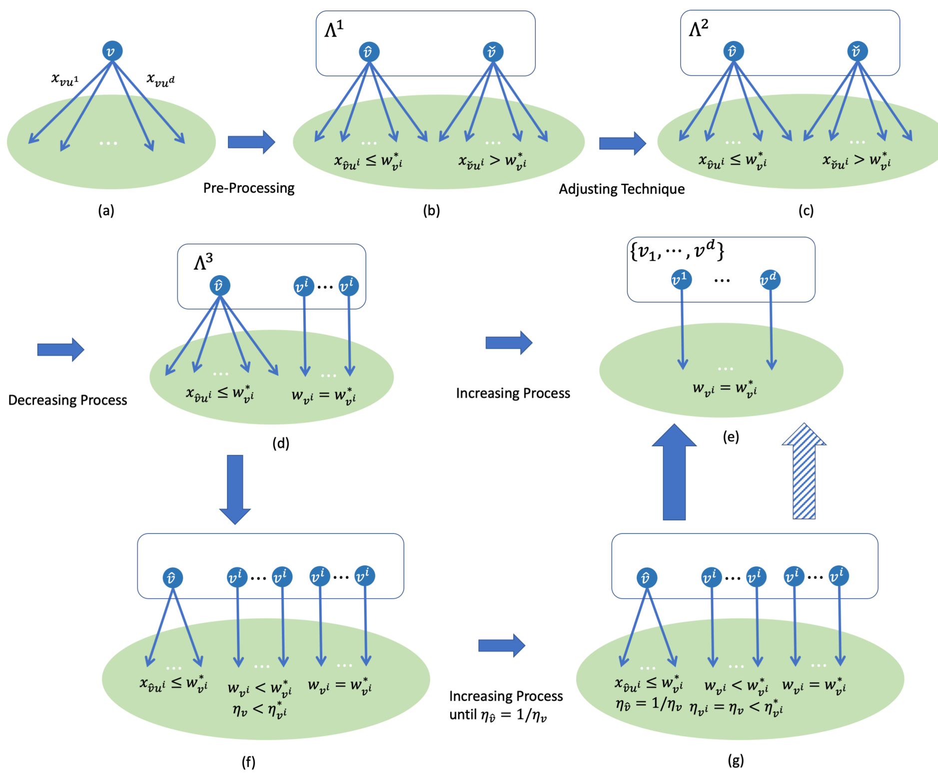

Fig. 4. Proof Sketch of Lemma 3.5

we decrease $w_{\check{v}^2}$ until there is one neighbor u^i satisfying $x_{\check{v}^2 u^i} = w^*_{v^i}$. At this time, let us execute RRB-split to split v^i out from $\check{v}^2$. Continue such decreasing and RRB-splitting operations, until $\check{v}^2$ is totally replaced by leaves and the weight of each of them being $w^*_{v^i}$. The network at the end of Stage 3 is denoted by $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v})$, and then Λ^3 only contains $\hat{v}^3$ and several fictitious nodes with degree of one with $w_{v^i} = w^*_{v^i}$.

Note that v's utility remains unchanged when we execute RRB-split. Combining with the monotonicity of U_S for any S containing $\check{v}^2$ (the monotonicity mentioned in Section 3.3), we have the following claim.

Claim 7. $U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v}) \le U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v})$.

Note that the ultimate network can be achieved by simply increasing $w_{\hat{v}^3}$ gradually and execute RRB-split until $\hat{v}^3$ is totally replaced by leaves with $w_{v^i} = w^*_{v^i}$ (we call it *simple transform process* and see Figure 4 (d) to (e)).

Recall that the η-ratio satisfies that $\forall u \in \tilde{G}, U_u = \eta_u \cdot w_u$ (informally). The key challenge here is how to upper bound the η-ratio. This task has been achieved in Section 3.3 because of the specific property of B-class vertices. Although the case in this subsection is a little bit more complicated, we can also achieve it.

For network $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v})$, let us consider a neighbor subset of $\hat{v}^3$, defined by $N' = \{u^i \in \Gamma(\hat{v}^3) | \eta^*_{v^i} \le \eta_v(G; \boldsymbol{w})\}$. (Recall that $\eta^*_{v^i} := \eta_{v^i}(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$.) We can prove that N' is not empty. If $\eta^*_{v^i} \geq 1/\eta_v(G;\boldsymbol{w})$ for any $u^i \in N'$, then the desired upper bound of two can be proved (Lemma 3.6) by the simple transform process.

Lemma 3.6. *If $\eta^*_{v^i} \geq 1/\eta_v(G;\boldsymbol{w})$ for any $u^i \in N'$, then*

$$U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) \leq U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v}) + U_v(G;\boldsymbol{w}) \leq 2 \cdot U_v(G;\boldsymbol{w}).$$

Under the condition in Lemma 3.6, we shall show that $\Gamma(\hat{v}^3) = N'$, which means for each v^i split from $\hat{v}^3$, its exchange ratio on the ultimate network is no more than v's exchange ratio $\eta_v(G;\boldsymbol{w})$. Intuitively, the growth of v' utility will not exceed $\eta_v(G;\boldsymbol{w})w_v = U_v(G;\boldsymbol{w})$, so the upper bound of incentive ratio of two holds.

On the other hand, if the condition in Lemma 3.6 does not hold, there will be some fictitious nodes whose exchange ratio increases as $w_{\hat{v}}$ increases and exceeds $\eta_v(G;\boldsymbol{w})$ at last. Fortunately, we can prove that if there is one node v^i with $\eta^*_{v^i} > \eta_v(G;\boldsymbol{w})$, then there must be some other fictitious nodes having $\eta^*_{v^i} < \eta_v(G;\boldsymbol{w})$. The trade-off between them eventually leads to the proof of the tight upper bound of two for the incentive ratio. We formally prove this by the following stages.

Stage 4: RRB-split Process (Figure 4 (d) to (f)). In this stage, we play RRB-split to split each node v^i from $\hat{v}^3$ satisfying $\eta^*_{v^i} > \eta_v(G;\boldsymbol{w})$, and assign its weight as $x_{\hat{v}^3u^i}$, the allocation from $\hat{v}^3$ to u^i on $\tilde{G}^3$. Note that it is possible that there are some leaves having their weights $w_{v^i} < w^*_{v^i}$ at this stage. As we only conduct the RRB-split operations in Stage 4, the utility of v remains unchanged. The network at the end of Stage 4 is denoted by $(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v})$.

Claim 8. $U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v}) = U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v})$.

Stage 5: Increasing Process (Figure 4 (f) to (g)). In this stage, we increase $w_{\hat{v}^4}$ and execute RRB-split once there is one u^i satisfying $x_{\hat{v}^4u^i} = w^*_{v^i}$. It is worth noting that when $w_{\hat{v}^4}$ increases, $\eta_{\hat{v}^4}$ is non-increasing (this property holds in general). To deal with the situation that some fictitious node's η-ratio may exceed $\eta_v(G;\boldsymbol{w})$, we add one another stop condition, that is when $\eta_{\hat{v}^4}$ decreases to $1/\eta_v(G;\boldsymbol{w})$ (which ≤ 1), Stage 5 ends. The condition of $1/\eta_v(G;\boldsymbol{w}) \leq 1$ ensures that there will be no fictitious node whose exchange ratio changes from less than or equal to $\eta_v(G;\boldsymbol{w})$ to greater than $\eta_v(G;\mathbf{w})$. Furthermore, all fictitious nodes v^i with $\eta_{v^i}(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v}) > \eta_v(G;\boldsymbol{w})$ are not impacted during this stage. The network at the end of Stage 4 is denoted by $(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v})$.

Claim 9. $U_v(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v}) \leq U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v}) + \eta_v(G;\boldsymbol{w}) \cdot (w(\Lambda^5) - w(\Lambda^4))$.

Stage 6: Updated Version of Increasing Process (Figure 4 (g) to (e)). Before the last stage starts, we first prove $\eta_{\hat{v}^5} = 1/\eta_v(G;\boldsymbol{w})$ and for each v^i with $w_{v^i} < w^*_{v^i}$, $\eta_{v^i} = \eta_v(G;\boldsymbol{w})$. In this stage the exchange ratio of leaves with $w_{v^i} < w^*_{v^i}$ will exceed $\eta_v(G;\boldsymbol{w})$ (Recall the Stage 4, each of these leaves satisfies $\eta^*_{v^i} > \eta_v(G;\boldsymbol{w})$).

First let us increase $\hat{v}^5$'s weight $w_{\hat{v}^5}$ alone. If there is a neighbor u^i satisfying $x_{\hat{v}^5u^i} = w^*_{v^i}$, then conduct RRB-split to split v^i out from $\hat{v}^5$, and continue to increase $w_{\hat{v}^5}$. By doing so, we shall show that $\eta_{\hat{v}^5}$ does not increase (might decrease, and the exchange ratio of all leaves with $w_{v^i} < w^*_{v^i}$ increases. Once there is a leaf v^i with $w_{v^i} < w^*_{v^i}$ and $\eta_{v^i} = \eta^*_{v^i}$, then increase w_{v^i} and $w_{\hat{v}^5}$ simultaneously while keeping $\eta_{v^i} = \eta^*_{v^i}$, until $w_{v^i} = w^*_{v^i}$. Finally, continue to increase $w_{\hat{v}^5}$ and repeat the same process to play RRB-split once there is one neighbor u^i with $x_{\hat{v}^5u^i} = w^*_{v^i}$, and increase w_{v^i} and $w_{\hat{v}^5}$ simultaneously for any leaf v^i with $w_{v^i} < w^*_{v^i}$ and $\eta_{v^i} = \eta^*_{v^i}$. This iteration ends when the ultimate network $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$ is achieved.

Although we can transform $(G^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v})$ into ultimate network by the process above (see Figure 4 the left arrow from (g) to (e)), but the upper bound of the utility (Lemma 3.7) needs more sophisticated analysis, which needs to combine several induction steps (see Figure 4 the right arrow from (g) to (e)).

Lemma 3.7. $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) \le U_v(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v}) + (w_v - w(\Lambda^5)) + w(\Lambda^4)$.

From the all claims and lemmas above, we are ready to prove Lemma 3.5.

Proof of Lemma 3.5.

$$\begin{aligned} U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v}) &\le U_v(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v}) + (w_v - w(\Lambda^5)) + w(\Lambda^4) \\ &\le U_v(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}, \boldsymbol{w}_{-v}) + \eta_v(G; \boldsymbol{w}) \cdot (w_v - w(\Lambda^5) + w(\Lambda^4)) \\ &\le U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}, \boldsymbol{w}_{-v}) + \eta_v(G; \boldsymbol{w}) \cdot w_v \\ &= U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}, \boldsymbol{w}_{-v}) + U_v(G; \boldsymbol{w}) \\ &\le U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}, \boldsymbol{w}_{-v}) + U_v(G; \boldsymbol{w}) = 2 \cdot U_v(G; \boldsymbol{w}). \end{aligned}$$

□

Through Lemma 3.4 and Lemma 3.5, we obtain the upper bound of two for the proportional response protocol against Sybil attack over general networks. Together with the lower bound from Example 2.10, we finish the proof of Theorem 1.2.

3.5 Numerical Example

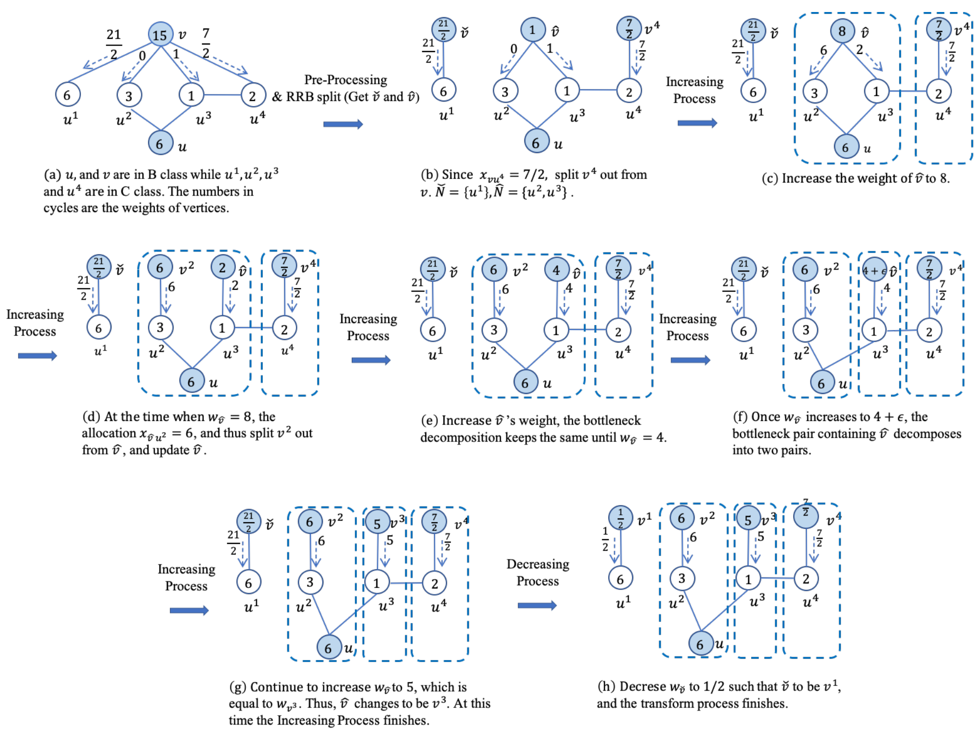

Fig. 5. The transform process from the initial network to the ultimate one.

To provide an intuitive understanding of our results, we introduce a resource exchange network example in Fig. 5 to illustrate the transform process depicted in the proof sketches. The original network is in Fig. 5-(a), in which the strategic agent v is a B-class vertex on the initial network with a weight of $w_v = 15$. The weights of other agents are shown by the numbers in the circle, and the amounts of resource v allocates to its neighbors are shown by the numbers along the dashed arrows. Consider that v plays a Sybil attack, splitting itself into four fictitious nodes, each connected to one of its neighbors $\{u^1, \cdots, u^4\}$, and let the weight assignments be $w_{v^1} = 1/2$, $w_{v^2} = 6$, $w_{v^3} = 5$, and $w_{v^4} = 7/2$ on the ultimate network. Fig. 5-(a) to (g) shows the transform process, as well as the changes in the bottleneck decomposition. Specifically, at the beginning, $x_{vu^4} = 7/2 = w_{v^2}$, v splits v^4 out in the pre-processing, and then through an RRB split operation, v is split into $\hat{v}$ with $w_{\check{v}} = 21/2$ and $\hat{v}$ with $w_{\check{v}} = 1$, in Fig. 5-(b). Then during the increasing process, when $w_{\hat{v}}$ increases to 8, shown in Fig. 5-(c), the allocation along the edge $x_{\hat{v}u^2} = w_{v^2} = 6$. Thus, $\hat{v}$ splits v^2 out in Fig. 5-(d) and $w_{\hat{v}}$ is updated to be 2. Continue to increase the weight of $\hat{v}$ and the bottleneck decomposition keeps the same until $w_{\hat{v}} = 4$ in Fig. 5-(e). Once $w_{\hat{v}} > 4$, bottleneck pair containing $\hat{v}$ is decomposed, as shown in Fig. 5-(f). The increasing process finishes when the weight of $w_{\hat{v}}$ is increased to 5, which is equal to w_{v^3}, and then $\hat{v}$ changes to v^3 in Fig. 5-(g). Finally, the decreasing process begins as the weight of $\check{v}$ decreases to $1/2$ and the ultimate network is obtained in Fig. 5-(h).

Let S be the set of fictitious nodes controlled by agent v. In Table 1, we show the changes of S, the total weight of all fictitious nodes $w(S)$, the total utility of agent v U_S, and the $\hat{v}$'s exchange ratio $\eta_{\hat{v}}$. Specifically, for each network, we present the weight change and utility change comparing with the previous one network, denoted by $\Delta w(S)$ and ΔU_S respectively. Combining with the value of $\eta_{\hat{v}}$, we can see that ΔU_S is bounded by $\eta_{\hat{v}} \Delta w(S)$ during the increasing process, and bounded by 0 during the decreasing process. This coincides with Claim 3 and Claim 4 in Section 3.3, respectively.

Table 1. The changes of S, $w(S)$, $\Delta w(S)$, U_S, ΔU_S, and $\eta_{\hat{v}}$

network	S	$w(S)$	$\Delta w(S)$	U_S	ΔU_S	$\eta_{\hat{v}}$
(a)	v	15	/	60/7	/	4/7
(b)	$\check{v}, \hat{v}, v^4$	15	0	72/7	0	4/7
(c)	$\check{v}, \hat{v}, v^4$	22	7	72/7	12/7	2/7
(d)	$\check{v}, \hat{v}, v^2, v^4$	22	0	72/7	0	2/7
(e)	$\check{v}, \hat{v}, v^2, v^4$	24	2	21/2	3/14	1/4
(f)	$\check{v}, \hat{v}, v^2, v^4$	$24 + \epsilon$	ϵ	21/2	0	$1/(4 + \epsilon)$
(g)	$\check{v}, v^2, v^3, v^4$	25	$1 - \epsilon$	21/2	0	1/5
(h)	v^1, v^2, v^3, v^4	15	-10	21/2	0	/

4 CONCLUSION

In this paper, we study the proportional response protocol for agents' resource exchange over general networks, conducting incentive analysis on Sybil attacks to market equilibrium. The solution concept of incentive ratio is applied to measure the effect of Sybil attack, in the sense of any agent's maximal benefit obtained by adopting the attack strategy. Our work develops a tight incentive ratio bound of two on the proportional response protocol against Sybil attacks over general networks. Technically, this result of the tight bound closes the gap in the past literature on Sybil attack over specific classes of networks. The detailed proofs also provide more delicate characterizations about how market equilibrium allocation changes with respect to an agent's strategic behaviors.

Practically, such a small incentive ratio bound implies any agent has limited incentive to adopt such a strategy. As in the resource exchange environment, a decentralized agent has to make an incredible effort to obtain enough information and do complicated computation to pursue such a small improvement in its utility. This not only theoretically justifies the underlying success of the resource exchange applications such as BitTorrent, but also opens up the possibility may the result be further extended to other market scenarios. For example, the proportional response protocol, or other similar protocols, has also been proved to guarantee market equilibrium allocations for agents in exchange market with more general utility functions [Brânzei et al., 2021], and further in Fisher markets [Birnbaum et al., 2011, Cheung et al., 2018, Zhang, 2011] and large-scale markets [Gao and Kroer, 2020]. The equilibrium solutions are always facing the threats from participating agents' possible strategic behaviors, but the corresponding incentive analysis is still an open problem. Since the allocations obtained by these proportional-like approaches probably share similar combinatorial structures, it would be of great interest to derive similar bounded of incentive ratios, accomplishing the robustness of these systems against adversary attacks.

REFERENCES

Bharat Adsul, Ch Sobhan Babu, Jugal Garg, Ruta Mehta, and Milind Sohoni. 2010. Nash equilibria in Fisher market. In *International Symposium on Algorithmic Game Theory*. Springer, 30–41.

Saba Ahmadi, Hedyeh Beyhaghi, Avrim Blum, and Keziah Naggita. 2021. The strategic perceptron. In *Proceedings of the 22nd ACM Conference on Economics and Computation*. 6–25.

Aaron Archer, Christos Papadimitriou, Kunal Talwar, and Éva Tardos. 2004. An approximate truthful mechanism for combinatorial auctions with single parameter agents. *Internet Mathematics* 1, 2 (2004), 129–150.

Kenneth J Arrow and Gerard Debreu. 1954. Existence of an equilibrium for a competitive economy. *Econometrica: Journal of the Econometric Society* (1954), 265–290.

Haris Aziz, Yoram Bachrach, Edith Elkind, and Mike Paterson. 2011. False-name manipulations in weighted voting games. *Journal of Artificial Intelligence Research* 40 (2011), 57–93.

Moshe Babaioff, John Chuang, and Michal Feldman. 2007. Incentives in peer-to-peer systems. *Algorithmic Game Theory* (2007), 593–611.

Marco Barreno, Blaine Nelson, Anthony D Joseph, and J Doug Tygar. 2010. The security of machine learning. *Machine Learning* 81, 2 (2010), 121–148.

Georgios Birmpas, Jiarui Gan, Alexandros Hollender, Francisco J Marmolejo-Cossío, Ninad Rajgopal, and Alexandros A Voudouris. 2021. Optimally Deceiving a Learning Leader in Stackelberg Games. *Journal of Artificial Intelligence Research* 72 (2021), 507–531.

Benjamin Birnbaum, Nikhil R Devanur, and Lin Xiao. 2011. Distributed algorithms via gradient descent for Fisher markets. In *Proceedings of the 12th ACM conference on Electronic commerce*. 127–136.

Simina Brânzei, Yiling Chen, Xiaotie Deng, Aris Filos-Ratsikas, Søren Kristoffer Stiil Frederiksen, and Jie Zhang. 2014. The Fisher Market Game: Equilibrium and Welfare. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, Carla E. Brodley and Peter Stone (Eds.). AAAI Press, 587–593. http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8373

Simina Brânzei, Nikhil Devanur, and Yuval Rabani. 2021. Proportional dynamics in exchange economies. In *Proceedings of the 22nd ACM Conference on Economics and Computation*. 180–201.

Markus Brill, Vincent Conitzer, Rupert Freeman, and Nisarg Shah. 2016. False-Name-Proof Recommendations in Social Networks.. In *AAMAS*. 332–340.

Ning Chen, Xiaotie Deng, Bo Tang, and Hongyang Zhang. 2016. Incentives for strategic behavior in fisher market games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

Ning Chen, Xiaotie Deng, Hongyang Zhang, and Jie Zhang. 2012. Incentive ratios of fisher markets. In *International Colloquium on Automata, Languages, and Programming*. Springer, 464–475.

Ning Chen, Xiaotie Deng, and Jie Zhang. 2011. How profitable are strategic behaviors in a market?. In *European Symposium on Algorithms*. Springer, 106–118.

Zhou Chen, Yukun Cheng, Xiaotie Deng, Qi Qi, and Xiang Yan. 2017. Limiting user's Sybil attack in resource sharing. In *International Conference on Web and Internet Economics*. Springer, 103–119.

Zhou Chen, Yukun Cheng, Xiaotie Deng, Qi Qi, and Xiang Yan. 2019. Agent incentives of strategic behavior in resource exchange. *Discrete Applied Mathematics* 264 (2019), 15–25.

Yukun Cheng, Xiaotie Deng, and Yuhao Li. 2020. Tightening Up the Incentive Ratio for Resource Sharing Over the Rings. In *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 127–136.

Yukun Cheng, Xiaotie Deng, Yifan Pi, and Xiang Yan. 2015. Can Bandwidth Sharing Be Truthful?. In *International Symposium on Algorithmic Game Theory*. Springer, 190–202.

Yukun Cheng, Xiaotie Deng, Qi Qi, and Xiang Yan. 2016. Truthfulness of a Proportional Sharing Mechanism in Resource Exchange.. In *IJCAI*. 187–193.

Yun Kuen Cheung, Richard Cole, and Nikhil R Devanur. 2020. Tatonnement beyond gross substitutes? Gradient descent to the rescue. *Games and Economic Behavior* 123 (2020), 295–326.

Yun Kuen Cheung, Richard Cole, and Ashish Rastogi. 2012. Tatonnement in ongoing markets of complementary goods. In *Proceedings of the 13th ACM Conference on Electronic Commerce*. 337–354.

Yun Kuen Cheung, Richard Cole, and Yixin Tao. 2018. Dynamics of distributed updating in Fisher markets. In *Proceedings of the 2018 ACM Conference on Economics and Computation*. 351–368.

Yun Kuen Cheung, Stefanos Leonardos, and Georgios Piliouras. 2021. Learning in Markets: Greed Leads to Chaos but Following the Price is Right. *arXiv preprint arXiv:2103.08529* (2021).

Vincent Conitzer, Nicole Immorlica, Joshua Letchford, Kamesh Munagala, and Liad Wagman. 2010. False-name-proofness in social networks. In *International Workshop on Internet and Network Economics*. Springer, 209–221.

Xiaotie Deng, Tao Lin, and Tao Xiao. 2020. Private data manipulation in optimal sponsored search auction. In *Proceedings of The Web Conference 2020*. 2676–2682.

Nikhil R Devanur, Christos H Papadimitriou, Amin Saberi, and Vijay V Vazirani. 2008. Market equilibrium via a primal–dual algorithm for a convex program. *Journal of the ACM (JACM)* 55, 5 (2008), 1–18.

John R Douceur. 2002. The sybil attack. In *International workshop on peer-to-peer systems*. Springer, 251–260.

Ran Duan, Jugal Garg, and Kurt Mehlhorn. 2016. An improved combinatorial polynomial algorithm for the linear Arrow-Debreu market. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on discrete algorithms*. SIAM, 90–106.

Ran Duan and Kurt Mehlhorn. 2015. A combinatorial polynomial algorithm for the linear Arrow–Debreu market. *Information and Computation* 243 (2015), 112–132.

Michal Feldman and John Chuang. 2005. Overcoming free-riding behavior in peer-to-peer systems. *ACM sigecom exchanges* 5, 4 (2005), 41–50.

Irving Fisher. 2006. *Mathematical investigations in the theory of value and prices, and appreciation and interest*. Cosimo, Inc.

Yotam Gafni, Ron Lavi, and Moshe Tennenholtz. 2020. VCG under Sybil (False-Name) Attacks-A Bayesian Analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 1966–1973.

Yotam Gafni, Ron Lavi, and Moshe Tennenholtz. 2021. Worst-case Bounds on Power vs. Proportion in Weighted Voting Games with an Application to False-name Manipulation. *Journal of Artificial Intelligence Research* 72 (2021), 99–135.

Jiarui Gan, Qingyu Guo, Long Tran-Thanh, Bo An, and Michael Wooldridge. 2019. Manipulating a learning defender and ways to counteract. *Advances in Neural Information Processing Systems* 32 (2019).

Yuan Gao and Christian Kroer. 2020. First-order methods for large-scale market equilibrium computation. *Advances in Neural Information Processing Systems* 33 (2020).

Leonidas Georgiadis, George Iosifidis, and Leandros Tassiulas. 2015. Exchange of services in networks: competition, cooperation, and fairness. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*. 43–56.

Juho Hamari, Mimmi Sjöklint, and Antti Ukkonen. 2016. The sharing economy: Why people participate in collaborative consumption. *Journal of the association for information science and technology* 67, 9 (2016), 2047–2059.

Atsushi Iwasaki, Vincent Conitzer, Yoshifusa Omori, Yuko Sakurai, Taiki Todo, Mingyu Guo, and Makoto Yokoo. 2010. Worst-case efficiency ratio in false-name-proof combinatorial auction mechanisms. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*. 633–640.

Kamal Jain. 2007. A polynomial time algorithm for computing an Arrow–Debreu market equilibrium for linear utilities. *SIAM J. Comput.* 37, 1 (2007), 303–318.

Anshul Kothari, David C Parkes, and Subhash Suri. 2005. Approximately-strategyproof and tractable multiunit auctions. *Decision Support Systems* 39, 1 (2005), 105–121.

Elias Koutsoupias and Christos Papadimitriou. 1999. Worst-case equilibria. In *Annual symposium on theoretical aspects of computer science*. Springer, 404–413.

Elias Koutsoupias and Christos Papadimitriou. 2009. Worst-case equilibria. *Computer science review* 3, 2 (2009), 65–69.

Daniel Lowd and Christopher Meek. 2005. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. 641–647.

Liang Ma, Xin Zhang, and Gao Shan Wang. 2017. Identifying the reasons why users in China recommend bike apps. *International Journal of Market Research* 59, 6 (2017), 767–786.

Ruta Mehta, Nithum Thain, László A Végh, and Adrian Vetta. 2014. To save or not to save: The fisher game. In *International Conference on Web and Internet Economics*. Springer, 294–307.

James B Orlin. 2010. Improved algorithms for computing Fisher's market clearing prices: Computing Fisher's market clearing prices. In *Proceedings of the forty-second ACM symposium on Theory of computing*. 291–300.

Johan Pouwelse, Paweł Garbacki, Dick Epema, and Henk Sips. 2005. The bittorrent p2p file-sharing system: Measurements and analysis. In *International Workshop on Peer-to-Peer Systems*. Springer, 205–216.

Tim Roughgarden and Éva Tardos. 2002. How bad is selfish routing? *Journal of the ACM (JACM)* 49, 2 (2002), 236–259.

James Schummer. 2004. Almost-dominant strategy implementation: exchange economies. *Games and Economic Behavior* 48, 1 (2004), 154–170.
Daniel D Sleator and Robert E Tarjan. 1985. Amortized efficiency of list update and paging rules. *Commun. ACM* 28, 2 (1985), 202–208.
Pingzhong Tang and Yulong Zeng. 2018. The price of prior dependence in auctions. In *Proceedings of the 2018 ACM Conference on Economics and Computation*. 485–502.
Taiki Todo and Vincent Conitzer. 2013. False-name-proof matching. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*. 311–318.
Taiki Todo, Atsushi Iwasaki, and Makoto Yokoo. 2011. False-name-proof mechanism design without money. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*. Citeseer, 651–658.
Zied Trifa and Maher Khemakhem. 2014. Sybil nodes as a mitigation strategy against sybil attack. *Procedia Computer Science* 32 (2014), 1135–1140.
Liad Wagman and Vincent Conitzer. 2014. False-name-proof voting with costs over two alternatives. *International Journal of Game Theory* 43, 3 (2014), 599–618.
Wikipedia contributors. 2021. Sybil attack — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Sybil_attack&oldid=1050473139 [Online; accessed 22-October-2021].
David P Williamson and David B Shmoys. 2011. *The design of approximation algorithms*. Cambridge university press.
Fang Wu and Li Zhang. 2007. Proportional response dynamics leads to market equilibrium. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*. ACM, 354–363.
Yinyu Ye. 2008. A path to the Arrow–Debreu competitive market equilibrium. *Mathematical Programming* 111, 1 (2008), 315–348.
Makoto Yokoo, Yuko Sakurai, and Shigeo Matsubara. 2004. The effect of false-name bids in combinatorial auctions: New fraud in Internet auctions. *Games and Economic Behavior* 46, 1 (2004), 174–188.
Li Zhang. 2011. Proportional response dynamics in the Fisher market. *Theoretical Computer Science* 412, 24 (2011), 2691–2698.

A PRELIMINARY PROPOSITIONS AND TECHNIQUES

In this section, some preliminary propositions are proposed to help us to prove the main result.

A.1 The Changes of α-Ratio, η-Ratio and Bottleneck Decomposition w.r.t. Single Parameter x

In this subsection, we would characterize the changes of α-ratio, η-ratio and bottleneck decomposition when the structure of G and other agents' weights remains unchanged, except only one agent v's weight x changes from a to b. Under this setting, that is G and $\boldsymbol{w}_{-v}$ are given, we can view $\mathcal{B}(G;\boldsymbol{w})$, $\alpha_u(G;\boldsymbol{w})$ and $\eta_u(G;\boldsymbol{w})$, for each $u \in V$, as the functions on one variable x, which are simplified as $\mathcal{B}(x)$, $\alpha_u(x)$ and $\eta_u(x)$, $x \in [a,b]$, in this subsection. Furthermore, we also denote the B-class vertices and C-class vertices by $B(G;x,\boldsymbol{w}_{-v})$ and $C(G;x,\boldsymbol{w}_{-v})$ respectively.

Proposition A.1 ([Cheng et al., 2016]). *Given a network $(G;x,\boldsymbol{w}_{-v})$, in which v's weight $x \in [a,b]$. Then $\alpha_v(x)$ is continuous on $[a,b]$ and is less than or equal to 1. In addition there exist three cases:*

(1) $\alpha_v(x)$ is non-decreasing and v is always in C-class when $x \in [a,b]$;
(2) $\alpha_v(x)$ is non-increasing and v is always in B-class when $x \in [a,b]$;
(3) there is one interval $[c,d]$ with $a \le c \le d \le b$, satisfying $\alpha_v(x) = 1$ and v is both in B and C-class, when $x \in [c,d]$, and
  (a) $\alpha_v(x)$ is non-decreasing and v is in C-class when $x \in [a,c]$;
  (b) $\alpha_v(x)$ is non-increasing and v is in B-class when $x \in [d,b]$.

Here we use an example network to illustrate the third case of Proposition A.1. In Figure 6-(a), we show the network, where $w_{u^1} = 20, w_{u^2} = 11, w_{u^3} = 5, w_{u^4} = 2, w_{u^5} = 18$, and vertex v represents the agent whose weight x changes from 5 to 45. In Figure 6-(b) and (c), we show how $\alpha_v(x)$ and $\eta_v(x)$ changes accordingly. Precisely, when $x \in [5,15]$, $\alpha_v(x)$ is non-decreasing and v is in C-class; when $x \in [5,15]$, $\alpha_v(x) = 1$ and v is both in B-class and C-class; and when $x \in [25,45]$, $\alpha_v(x)$

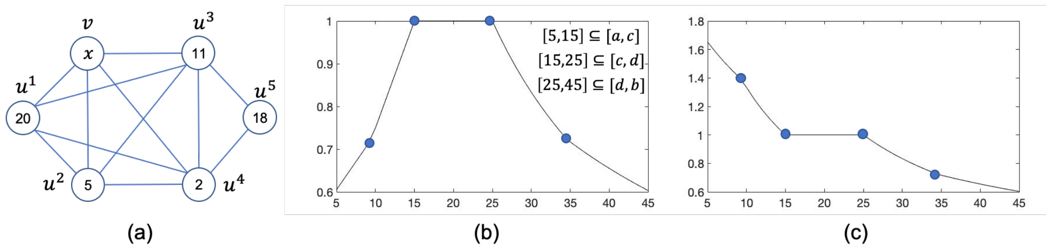

Fig. 6. The changes of $\alpha_v(x)$ and $\eta_v(x)$.

is non-increasing and v is a B-class vertex. The corresponding bottleneck decomposition of the network along with the change of x is shown in Table 2.

x	$\mathcal{B}$
$(5, \frac{85}{9}]$	$B_1 = \{u^1, u^5\}, C_1\{\mathbf{v}, u^2, u^3, u^4\}$
$(\frac{85}{9}, 15)$	$B_1 = \{u^5\}, C_1 = \{u^3, u^4\}; B_2 = \{u^1\}, C_2 = \{\mathbf{v}, u^2\}$
$[15, 25]$	$B_1 = \{u^5\}, C_1 = \{u^3, u^4\}$; $B_2 = C_2 = \{\mathbf{v}, u^1, u^2\}$
$(25, \frac{450}{13})$	$B_1 = \{u^5\}, C_1 = \{u^3, u^4\}$; $B_2 = \{\mathbf{v}\}, C_2 = \{u^1, u^2\}$
$[\frac{450}{13}, 45)$	$B_1 = \{\mathbf{v}, u^5\}, C_1 = \{u^1, u^2, u^3, u^4\}$

Table 2. All subintervals, each corresponding one decomposition, where v is written in bold.

Motivated by this example, it is easy to observe that when $\mathcal{B}(x)$ changes with variable $x \in [a, b]$, it is possible that $\mathcal{B}(x)$ remains the same in some subintervals of $[a, b]$. Based on this observation, $[a, b]$ is partitioned into a number of disjoint subintervals $\{\langle a_i, b_i\rangle\}_i$. W.l.o.g, assume $a \le a_i \le b_i = a_{i+1} \le b_{i+1} \le b$. The symbol "$\langle\rangle$" is used to denote the subinterval, since the subinterval could be one of five forms $[a_i, b_i]$, $[a_i, b_i)$, $(a_i, b_i]$, (a_i, b_i) and $a_i = b_i$. When $x \in \langle a_i, b_i\rangle$, the bottleneck decomposition is represented as $\mathcal{B}(x) = \mathcal{B}^i = \{(B_1^i, C_1^i), \cdots, (B_{k^i}^i, C_{k^i}^i)\}$, and α_t^i denotes the α-ratio of pair (B_t^i, C_t^i), $t = 1, \cdots, k^i$.

From Table 2, we can find that v is always in the same class in two adjacent $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$. Hence, if the class that v is in changes when $x \in [a, b]$, then there must exist an interim interval $[c, d] \subseteq [a, b]$, as stated in the third case of Proposition A.1, such that v could be viewed as a B-class or C-class vertex simultaneously. Furthermore, (B_j^i, C_j^i) and $(B_\ell^{i+1}, C_\ell^{i+1})$ only have the relationship of dividing and merging in the example. These properties about the bottleneck pairs that v belongs to in two adjacent $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$ indeed hold for any networks, which is characterized by Cheng, Deng, Qi and Yan [Cheng et al., 2016] as follows:

Proposition A.2 ([Cheng et al., 2016]). *Suppose v is in bottleneck pairs (B_j^i, C_j^i) and $(B_\ell^{i+1}, C_\ell^{i+1})$ in two adjacent bottleneck decompositions $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$. Then,*

(1) $v \in B_j^i \cap B_\ell^{i+1}$ or $v \in C_j^i \cap C_\ell^{i+1}$;

(2) if $v \in C_j^i \cap C_\ell^{i+1}$, then

(a) $B_j^i = B_j^{i+1} \cup B_{j+1}^{i+1}, C_j^i = C_j^{i+1} \cup C_{j+1}^{i+1}, v \in C_j^i \cap C_{j+1}^{i+1}$ ($\ell = j+1$), and $\alpha_j^i(b_i) = \alpha_j^{i+1}(b_i) = \alpha_{j+1}^{i+1}(b_i)$;

(b) $B_j^i \cup B_{j+1}^i = B_j^{i+1}, C_j^i \cup C_{j+1}^i = C_j^{i+1}, v \in C_j^i \cap C_j^{i+1}$ ($\ell = j$), and $\alpha_j^{i+1}(b_i) = \alpha_j^i(b_i) = \alpha_{j+1}^i(b_i)$;

(3) if $v \in B_j^i \cap B_\ell^{i+1}$, then

(a) $B_\ell^i = B_\ell^{i+1} \cup B_{\ell+1}^{i+1}, C_\ell^i = C_\ell^{i+1} \cup C_{\ell+1}^{i+1}, v \in B_\ell^i \cap B_\ell^{i+1}$ ($j = \ell$), and $\alpha_\ell^i(b_i) = \alpha_\ell^{i+1}(b_i) = \alpha_{\ell+1}^{i+1}(b_i)$;

(b) $B_\ell^i \cup B_{\ell+1}^i = B_\ell^{i+1}, C_\ell^i \cup C_{\ell+1}^i = C_\ell^{i+1}, v \in B_{\ell+1}^i \cap B_\ell^{i+1}$ ($j = \ell+1$), and $\alpha_\ell^{i+1}(b_i) = \alpha_\ell^i(b_i) = \alpha_{\ell+1}^i(b_i)$.

Based on Proposition A.2 and Definition 2.8 of η-ratio, we continue to derive following propositions. This proposition essentially shows that when other agents' weight profile $\boldsymbol{w}_{-v}$ are given, the η-ratio of v is a monotonically non-increasing function of w_v.

Proposition A.3. *Given a network $(G; x, \boldsymbol{w}_{-v})$, in which v's weight $x \in [a, b]$. Then $\eta_u(x)$ is continuous on $[a, b]$, $\forall u \in V$. Furthermore, $\eta_v(x)$ is monotonically non-increasing on $[a, b]$.*

Proof. Suppose that $[a, b]$ is partitioned into h subintervals, i.e., $[a, b] = \cup_{i=1}^{h} \langle a_i, b_i \rangle$, so it is sufficient to prove that $\eta_u(x)$ is continuous on $\langle a_i, b_i \rangle$ and at the break point b_i, $i = 1, \cdots, h-1$. Suppose v is in bottleneck pair (B_j^i, C_j^i). Then each pair (B_t^i, C_t^i), $t \neq j$, does not contain v, and then its α-ratio α_t^i is a constant. Therefore, $\eta_u(x)$ is a constant and continuous on $\langle a_i, b_i \rangle$. If u is in (B_j^i, C_j^i), then $\eta_u(x) = \alpha_j^i(x) = \alpha_v(x)$, if $u \in B_j^i$, and $\eta_u(x) = 1/\alpha_j^i(x) = 1/\alpha_v(x)$, otherwise. Since $\alpha_v(x)$ is continuous on $\langle a_i, b_i \rangle$ by Proposition A.1, $\eta_u(x)$ is also continuous on $\langle a_i, b_i \rangle$.

To illustrate the continuity of $\eta_u(x)$ at break point b_i, let us consider the situation that v is a C-class vertex in two adjacent $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$. By Proposition A.2-2(a), $B_j^i = B_j^{i+1} \cup B_{j+1}^{i+1}$ and $C_j^i = C_j^{i+1} \cup C_{j+1}^{i+1}$, when x changes between $\langle a_i, b_i \rangle$ and $\langle a_{i+1}, b_{i+1} \rangle$. If $u \in B_j^i$, it may be in B_j^{i+1} or B_{j+1}^{i+1}. So $\eta_u(x) = \alpha_j^i(x)$, when $x \in \langle a_i, b_i \rangle$, and $\eta_u(x) = \alpha_j^{i+1}(x)$ or $\eta_u(x) = \alpha_{j+1}^{i+1}(x)$, when $x \in \langle a_{i+1}, b_{i+1} \rangle$. Since $\alpha_j^i(b_i) = \alpha_j^{i+1}(b_i) = \alpha_{j+1}^{i+1}(b_i)$ by Proposition A.2-2(a), we have $\lim_{x \to b_i} \eta_u(x) = \eta_u(b_i)$, showing $\eta_u(x)$ is continuous at b_i. The discussion for other cases is similar.

The monotonic property of $\eta_v(x)$ can be derived directly from Proposition A.1 and Definition 2.8, as shown in Figure 6-(c). $\square$

The subsequent two propositions characterize the influence of the changing of v's weight x, on other bottleneck pairs except for the ones containing v. In advance, let us define $(B_u(x), C_u(x))$ to be the bottleneck pair containing u when v's weight is $x \in [a, b]$, with its α-ratio of $\alpha_u(x) = \frac{w(C_u(x))}{w(B_u(x))}$.

Proposition A.4. *Given a network $(G; x, \boldsymbol{w}_{-v})$. Suppose v decreases its weight x from b to a ($a < b$). If v is in C-class when $x = b$, then*

(1) for any vertex $u \in V$, the class that u is in remains unchanged when $x \in [a, b]$. Formally, if $u \in B(G; a, \boldsymbol{w}_{-v})$, then $u \in B(G; x, \boldsymbol{w}_{-v})$, and if $u \in C(G; a, \boldsymbol{w}_{-v})$, then $u \in C(G; x, \boldsymbol{w}_{-v})$, $\forall x \in [a, b]$;

(2) for any vertex u with $\alpha_u(b) > \alpha_v(b)$ or $\alpha_u(b) < \alpha_v(a)$, we have $(B_u(x), C_u(x)) = (B_u(b), C_u(b))$ and $\alpha_u(x) = \alpha_u(b)$, $\forall x \in [a, b]$;

(3) for any vertex u with $\alpha_u(b) \in [\alpha_v(a), \alpha_v(b)]$, we have $\alpha_u(x) \in [\alpha_v(a), \alpha_v(b)]$, $\forall x \in [a, b]$;

(4) for any vertex $u \in V$, if $x \leq y$, then $\alpha_u(x) \leq \alpha_u(y)$.

Proof. From Proposition A.1, we can deduce that, if v is a C-class vertex when $x = b$, then v is always in C-class and $\alpha_v(x)$ is non-decreasing when $x \in [a, b]$. So $\alpha_v(x) \leq \alpha_v(y)$ for $a \leq x \leq y \leq b$. Now we first prove the last property, i.e., for any vertex $u \in V$, if $x \leq y$, then $\alpha_u(x) \leq \alpha_u(y)$.

Suppose that $[a, b]$ is partitioned into h subintervals, i.e., $[a, b] = \cup_{i=1}^{h} \langle a_i, b_i \rangle$. Since $\alpha_u(x)$ is continuous on $[a, b]$, it is sufficient to prove that $\alpha_u(x)$ is monotonous on $\langle a_i, b_i \rangle$. Suppose v is in bottleneck pair (B_j^i, C_j^i). Then each pair (B_t^i, C_t^i), $t \neq j$, does not contain v. Therefore for each vertex $u \in B_t^i \cup C_t^i$, $\alpha_u(x) = \alpha_t^i$ is a constant, monotonously on $\langle a_i, b_i \rangle$. If u is in (B_j^i, C_j^i), then $\alpha_u(x) = \alpha_v(x)$. As $\alpha_v(x)$ is monotonous on $\langle a_i, b_i \rangle$, $\alpha_u(x)$ is also monotonous on $\langle a_i, b_i \rangle$.

Combined with the continuity of $\eta_u(x)$ (Proposition A.3), for any $u \in V$, if u is in B-class, then u's α-ratio is non-increasing, which means u is always in B-class when $x \in [a, b]$. The proof for C-class vertices is similar, which completes the first property.

Next we use the iterative method to deduce the second property. When $x = b = b_h$, for any $u \in V$ with $\alpha_u(b) > \alpha_v(b)$, we know that u and v are not in the same bottleneck pair. So when $x \in \langle a_h, b_h \rangle$, we have $(B_u(x), C_u(x)) = (B_u(b), C_u(b))$ and $\alpha_u(x) = \alpha_u(b)$ is a constant. Apply the the continuity of $\alpha_u(x)$, we have $\alpha_u(a_h) = \alpha_u(a_h + \epsilon)$ for any sufficiently small $\epsilon > 0$, so $\alpha_u(a_h) = \alpha_u(b)$. Furthermore, by the monotonicity of $\alpha_v(x)$, we have $\alpha_v(a_h) \leq \alpha_v(b) < \alpha_u(b) = \alpha_u(a_h)$. When w_v decreases from a_h to $a_h - \epsilon$ for any sufficiently small $\epsilon > 0$, by the continuity, we know $\alpha_v(a_h - \epsilon) < \alpha_u(a_h - \epsilon)$, which means v and u are still not in the same bottleneck pair. This implies when $x = a_h - \epsilon$, for each $u \in V$ with $\alpha_u(b) > \alpha_v(b)$, we have $(B_u(a_h - \epsilon), C_u(a_h - \epsilon)) = (B_u(b), C_u(b))$ and $\alpha_u(a_h - \epsilon) = \alpha_u(b)$. Then we only need to prove $(B_u(x), C_u(x)) = (B_u(a_h - \epsilon), C_u(a_h - \epsilon))$ and $\alpha_u(x) = \alpha_u(a_h - \epsilon)$ when $x \in [a, a_h - \epsilon]$, which can be deduced by the iterative method. The case $\alpha_u(b) < \alpha_v(a)$ is similar.

For any u with $\alpha_u(b) \in [\alpha_v(a), \alpha_v(b)]$, by the monotonicity of $\alpha_u(x)$, we have $\alpha_u(x) \in (0, \alpha_v(b)]$ for any $x \in [a, b]$. Note that when $\alpha_u(x)$ is decreasing, it must be u and v are in the same bottleneck pair, which means when $\alpha_u(x)$ is decreasing, $\alpha_u(x) = \alpha_v(x)$ always holds. Since $\alpha_v(x) \geq \alpha_v(a)$ all the time, we have if $\alpha_u(b) \geq \alpha_v(a)$, then $\alpha_u(x) \geq \alpha_v(a)$.

The desired result has been proved. □

Proposition A.4 has an intuitive interpretation that when x decreases from b to a and v is a C-class vertex when $x = b$, then v is always in C-class. In addition, any bottleneck pair, having its α-ratio be larger than $\alpha_v(b)$ and smaller than $\alpha_v(a)$, is not impacted by the changing of x. Some similar results can be obtained under the setting that v's weight x increases from a to b and v is in B-class when $x = a$. Clearly, by Proposition A.1, if v is a B-class vertex when $x = a$, then v is always in B-class and $\alpha_v(x)$ is non-increasing, when $x \in [a, b]$.

Proposition A.5. *Given a network G and other agents' weight profile $\mathbf{w}_{-v}$. Suppose v increases its weight x from a to b ($a < b$). If v is in B-class when $x = a$, then*

(1) for any vertex $u \in V$, the class that u is in remains unchanged when $x \in [a, b]$. Formally, if $u \in B(G; b, \mathbf{w}_{-v})$, then $u \in B(G; x, \mathbf{w}_{-v})$, and if $u \in C(G; b, \mathbf{w}_{-v})$, then $u \in C(G; x, \mathbf{w}_{-v})$, $\forall x \in [a, b]$;
(2) for any vertex $u \in V$ with $\alpha_u(a) > \alpha_v(a)$ or $\alpha_u(a) < \alpha_v(b)$, we have $(B_u(x), C_u(x)) = (B_u(b), C_u(b))$ and $\alpha_u(x) = \alpha_u(a)$, $\forall x \in [a, b]$;
(3) for any vertex u with $\alpha_u(b) \in [\alpha_v(b), \alpha_v(a)]$, we have $\alpha_u(x) \in [\alpha_v(b), \alpha_v(a)]$, $\forall x \in [a, b]$;
(4) for any vertex $u \in V$, if $x \leq y$, then $\alpha_u(x) \geq \alpha_u(y)$.

Proposition A.4 and Proposition A.5 only discuss the cases that v is always in C-class or B-class when $x \in [a, b]$. The situation would be complicated when the class of v changes from C-class to B-class, when x increases from a to b. The following proposition is related with this complicated situation, obtained by Proposition A.4 and Proposition A.5.

Proposition A.6. *Given a network G with weight profile $(x, \mathbf{w}_{-v})$, in which x is the weight of v. Suppose x increases from a to b. Then*

(1) for any $u \in V$ with $\alpha_u(a) < \min\{\alpha_v(a), \alpha_v(b)\}$, $(B_u(x), C_u(x)) = (B_u(a), C_u(a))$, $\forall x \in [a, b]$.
(2) for any $u \in V$ with $\alpha_u(b) \geq \min\{\alpha_v(b), \alpha_v(a)\}$, $\alpha_u(x) \geq \min\{\alpha_v(b), \alpha_v(a)\}$, $\forall x \in [a, b]$;

Proof. If v is always in C-class or v is always in B-class during w_v increases from a to b, then this proposition is obvious by Proposition A.4 and Proposition A.5. So in the following, we only consider the case that the class of v changes from C-class to B-class.

We first partition $[a, b]$ into two subintervals as $[a, c]$ and $[c, b]$, $a \leq c \leq b$, such that v is in C-class and B-class on $[a, c]$ and $[c, b]$, respectively. Here we need to state that if there is $x' \in [a, c]$ and $x'' \in [c, b]$ such that $\alpha_v(x') < 1$ and $\alpha_v(x'') < 1$, then $\alpha_v(c) = 1$ as v can be viewed as a C-class vertex and a B-class vertex simultaneously.

By Proposition A.4, if $\alpha_u(c) < \alpha_v(a)$ then $(B_u(x), C_u(x)) = (B_u(c), C_u(c))$ for any $x \in [a, c]$. On the other hand, by Proposition A.5, we have, if $\alpha_u(c) < \alpha_v(b)$, then $(B_u(x), C_u(x)) = (B_u(c), C_u(c))$ for any $x \in [c, b]$. Combining these two conclusions, it is not hard to see, if $\alpha_u(c) < \min\{\alpha_v(a), \alpha_v(b)\}$, then $(B_u(x), C_u(x)) = (B_u(c), C_u(c))$ for any $x \in [a, b]$. Such a result can be further derived that, if $\alpha_u(b) \geq \min\{\alpha_v(b), \alpha_v(a)\}$, then $\alpha_u(x) \geq \min\{\alpha_v(b), \alpha_v(a)\}$, for any $x \in [a, b]$; □

A.2 Bound of the Total Utility of a Specified Set w.r.t. Single Parameter x

In this subsection, we turn to characterize the bound of the total utility of a specified vertex set, under the setting that the network G and other agents' weight profile $\mathbf{w}_{-v}$ are given, except for v's weight $x \in [a, b]$. Particularly, the total utility of S is defined as $U_S(x) := \sum_{u \in S} U_u(x)$, a function on variable x. As stated in previous subsection, $[a, b]$ can be partitioned into several disjoint subinterval $\{\langle a_i, b_i \rangle\}_i$, such that the bottleneck decomposition $\mathcal{B}(x) = \mathcal{B}^i$ remains unchanged, when $x \in \langle a_i, b_i \rangle$. The uncertainty whether each $\langle a_i, b_i \rangle$ is closed or open makes our discussion complicated. The following lemma is a tool to deal with this uncertainty.

Lemma A.7. *Let $f(x)$ be a continuous function on $[a, b] = \cup_i \langle a_i, b_i \rangle$, and k be a constant. For any subinterval $\langle a_i, b_i \rangle$, and any $[y, z] \subseteq \langle a_i, b_i \rangle$, if $0 \leq f(z) - f(y) \leq k(z - y)$, then $0 \leq f(b) - f(a) \leq k(b - a)$.*

Proof. By the condition of $f(z) - f(y) \leq k(z - y)$, $\forall [y, z] \subseteq \langle a_i, b_i \rangle$, we have $\lim_{y \to a_i}(f(z) - kz) - (f(y) - ky) \leq 0$. The continuity of $f(x)$ on $[a, b]$ guarantees $\lim_{x \to a_i}(f(x) - kx) = f(a_i) - ka_i$. Combining above two limits, $f(z) - kz \leq \lim_{y \to a_i}(f(y) - ky) = f(a_i) - ka_i$. At the same time, $\lim_{z \to b_i} f(z) - kz = f(b_i) - kb_i$. Thus $f(b_i) - kb_i \leq f(a_i) - ka_i$, implying $f(b_i) - f(a_i) \leq k(b_i - a_i)$. Since $[a, b] = \cup_i \langle a_i, b_i \rangle$, then

$$f(b) - f(a) = \sum_{\langle a_i, b_i \rangle \subseteq [a,b]} f(b_i) - f(a_i) \leq \sum_{\langle a_i, b_i \rangle \subseteq [a,b]} k(b_i - a_i) = k(b - a).$$

The proof for the other side of $0 \leq f(b) - f(a)$ is similar. □

Firstly, directly from the continuity of $\eta_u(x)$ (Proposition A.3), we have

Lemma A.8. *Given a network G with weight profile $(x, \mathbf{w}_{-v})$, in which $x \in [a, b]$ is the weight of v. Then $U_S(x)$ is continuous on $[a, b]$, for any vertex subset $S \subseteq V$.*

Next, we would propose the bounds of $U_S(x)$ by applying Lemma A.7.

Lemma A.9. *Given a network G with weight profile $(x, \mathbf{w}_{-v})$, in which $x \in [a, b]$ is the weight of v. For any $S \subseteq V$ containing v, we have $U_S(b) - U_S(a) \geq 0$.*

Proof. As $[a, b] = \cup_i \langle a_i, b_i \rangle$ and $U_S(x)$ is continuous on $[a, b]$, to obtain $U_S(b) - U_S(a) \geq 0$, it is enough for us to prove $U_S(z) - U_S(y) \geq 0$ for any $[y, z] \subseteq \langle a_i, b_i \rangle$, by Lemma A.7. Let us assume $v \in B^i_j \cup C^i_j$ in bottleneck decomposition $\mathcal{B}^i$ when $x \in \langle a_i, b_i \rangle$. Then the other pair (B^i_t, C^i_t), $t \neq j$, does not contain v and its α-ratio of α^i_t is fixed, implying the utilities of all vertices in (B^i_t, C^i_t) are fixed too, if $x \in \langle a_i, b_i \rangle$. It follows that $U_u(z) - U_u(y) = 0$ for any $u \in (B^i_t, C^i_t)$ and any $[y, z] \subseteq \langle a_i, b_i \rangle$.

Next we discuss the utility of the vertex in $B^i_j \cup C^i_j$. Specially, if $\alpha^i_j = 1$, showing $B^i_j = C^i_j$, then $U_u(x) = w_u$, for any $u \neq v \in B^i_j \cup C^i_j$, and $U_v(x) = x$. Obviously, for any $[y, z] \subseteq \langle a_i, b_i \rangle$, $U_u(z) - U_u(y) = 0$ if $u \neq v$, and $U_v(z) - U_v(y) = z - y \geq 0$. If follows $U_S(z) - U_S(y) = z - y \geq 0$.

If $\alpha_j^i < 1$, then

$$\alpha_j^i(x) = \alpha_v(x) = \begin{cases} \frac{w(C_j^i)}{w(B_j^i \setminus (S\cap B_j^i)) + w(S\cap B_j^i \setminus \{v\}) + x}, & v \in B_j^i; \\ \frac{w(C_j^i \setminus (S\cap C_j^i)) + w(S\cap C_j^i \setminus \{v\}) + x}{w(B_j^i)}, & v \in C_j^i. \end{cases}$$

The utilities of $U_{S\cap B_j^i}(x)$ and $U_{S\cap C_j^i}(x)$ are

$$U_{S\cap B_j^i}(x) = \begin{cases} \frac{w(C_j^i)\cdot(w(S\cap B_j^i \setminus \{v\}) + x)}{w(B_j^i \setminus (S\cap B_j^i)) + w(S\cap B_j^i \setminus \{v\}) + x}, & v \in B_j^i; \\ \frac{w(S\cap B_j^i)}{w(B_j^i)} \cdot (w(C_j^i \setminus \{v\}) + x), & v \in C_j^i. \end{cases}$$

$$U_{S\cap C_j^i}(x) = \begin{cases} \frac{w(S\cap C_j^i)}{w(C_j^i)} \cdot (w(B_j^i \setminus \{v\}) + x), & v \in B_j^i; \\ \frac{w(B_j^i)\cdot(w(S\cap C_j^i \setminus \{v\}) + x)}{w(C_j^i \setminus (S\cap C_j^i)) + w(S\cap C_j^i \setminus \{v\}) + x}, & v \in C_j^i. \end{cases}$$

It is not hard to verify that $\partial U_{S\cap B_j^i}(x)/\partial x \geq 0$ and $\partial U_{S\cap C_j^i}(x)/\partial x \geq 0$, meaning $U_S(x)$ is non-decreasing on $[y, z]$. Thus, we have $U_S(z) - U_S(y) \geq 0$. This completes the proof. □

Lemma A.10. *Given a network G with weight profile $(x, \mathbf{w}_{-v})$, in which $x \in [a, b]$ is the weight of v. Suppose v is a B-class vertex when $x = a$ and $\alpha_v(a) < 1$. Let S be the vertex set satisfying: (1) $v \in S$; and (2) for each vertex $u \in S$, if u is in C-class when $x = a$, then $\alpha_u(a) > \alpha_v(a)$. Then $U_S(b) - U_S(a) \leq \eta_v(a)(b - a)$.*

Proof. By the same analysis for Lemma A.7, we only need to prove $U_S(z) - U_S(y) \leq \eta_v(a)(z-y)$ for any $[y, z] \subseteq \langle a_i, b_i \rangle$. Of course, when $x \in \langle a_i, b_i \rangle$, we have $U_u(z) - U_u(y) = 0$ for each $u \in B_t^i \cup C_t^i$, $t \neq j$, since $v \in B_j^i \cup C_j^i$. Due to the condition that v is in B-class when $x = a$, and the property that $\eta_v(x)$ is non-increasing on $[a, b]$, it is easy to deduce that $\eta_v(x) \leq \eta_v(a) = \alpha_v(a) < 1$, implying v is always a B-class vertex when $x \in [a, b]$. In addition, the condition that for each vertex $u \in S$, if u is in C-class when $x = a$, then $\alpha_u(a) > \alpha_v(a)$, makes sure that $(B_u(x), C_u(x)) = (B_u(a), C_u(a))$ and $\alpha_u(x) = \alpha_u(a)$ for any $x \in [a, b]$ by Proposition A.5-(2). It means that the bottleneck pair containing u remains unchanged when $x \in [a, b]$. Therefore, such a vertex u is always in C-class and $\alpha_u(x) = \alpha_u(a) > \alpha_v(a) \geq \alpha_v(x)$, when $x \in \langle a_i, b_i \rangle$. This implies $S \cap C_j^i = \emptyset$.

Overall, we conclude that $v \in B_j^i$ and $S \cap C_j^i = \emptyset$. So when $x \in [y, z] \subseteq \langle a_i, b_i \rangle$,

$$\alpha_j^i(x) = \alpha_v(x) = \frac{w(C_j^i)}{w(B_j^i \setminus (S \cap B_j^i)) + w(S \cap B_j^i \setminus \{v\}) + x},$$

and the utility function $U_{S\cap(B_j^i \cup C_j^i)}(x)$ is

$$U_{S\cap(B_j^i \cup C_j^i)}(x) = \frac{w(C_j^i) \cdot (w(S \cap B_j^i \setminus \{v\}) + x)}{w(B_j^i \setminus (S \cap B_j^i)) + w(S \cap B_j^i \setminus \{v\}) + x}.$$

It is easy to verify that $U_S(z) - U_S(y) = U_{S\cap(B_j^i \cup C_j^i)}(z) - U_{S\cap(B_j^i \cup C_j^i)}(y) \leq \eta_v(y)\cdot(z-y) \leq \eta_v(a)\cdot(z-y)$. This completes the proof. □

Lemma A.11. *Given a network G with weight profile $(x, \mathbf{w}_{-v})$, in which $x \in [a, b]$ is the weight of v. Suppose v is a B-class vertex when $x = a$ and $\alpha_v(a) < 1$. Let S be the vertex set satisfying for each vertex $u \in S$, if u is in B-class when $x = a$, then $\alpha_u(a) > \alpha_v(a)$. Then $U_S(b) - U_S(a) \leq (b - a)$.*

Lemma A.11 is analogous to Lemma A.10. The proof of Lemma A.11 is also highly similar to the proof of Lemma A.10.

Lemma A.12. *Let $\mathcal{B} = \{(B_1, C_1), \cdots, (B_k, C_k)\}$ be the bottleneck decomposition on the network G with weight profile $\boldsymbol{w} = (w_v, \boldsymbol{w}_{-v})$. Suppose $v \in B_i$, and S is the set satisfying $S \subseteq B_i \cup C_i$ and $v \in S$. If v increases its weight from w_v to $w_v + \Delta x$ ($\Delta x > 0$) such that the bottleneck decomposition remains unchanged, i.e., $\mathcal{B}(G; \boldsymbol{w}) = \mathcal{B}(G; w_v + \Delta x, \boldsymbol{w}_{-v})$ and $U_S(w_v + \Delta x) - U_S(w_v) > \Delta x$, then $U_{S \cap B_i}(w_v + \Delta x) < w(S \cap C_i)$.*

Proof. First, we shall claim that $\alpha_v(w_v) = \alpha_i(w_v) < 1$. If not, then $\alpha_i(w_v) = 1$, implying $B_i = C_i$, and thus $\alpha_i(w_v + \Delta x) = \alpha_i(w_v) = 1$ because of the condition of $\mathcal{B}(G; \boldsymbol{w}) = \mathcal{B}(G; w_v + \Delta x, \boldsymbol{w}_{-v})$. It follows $U_S(w_v + \Delta x) = \sum_{u \neq v \in S} w_u + (w_v + \Delta x)$ and $U_S(w_v) = \sum_{u \neq v \in S} w_u + w_v$, resulting in $U_S(w_v + \Delta x) - U_S(w_v) = \Delta x$. This contradicts the condition that $U_S(w_v + \Delta x) - U_S(w_v) > \Delta x$, and hence $\alpha_v(w_v) = \alpha_i(w_v) < 1$. In addition, we have

$$
\begin{aligned}
U_S(w_v) &= \sum_{u \in S \cap B_i} w_u \alpha_i(w_v) + \sum_{u \in S \cap C_i} \frac{w_u}{\alpha_i(w_v)}; \\
U_S(w_v + \Delta x) &= \sum_{u \in S \cap B_i} w_u \alpha_i(w_v + \Delta x) + \Delta x \alpha_i(w_v + \Delta x) + \sum_{u \in S \cap C_i} \frac{w_u}{\alpha_i(w_v + \Delta x)}.
\end{aligned}
$$

So

$$
\begin{aligned}
U_S(w_v + \Delta x) - U_S(w_v) &= \left(\sum_{u \in S \cap B_i} w_u + \Delta x \right) \alpha_i(w_v + \Delta x) - \sum_{u \in S \cap B_i} w_u \alpha_i(w_v) \\
&\quad + \sum_{u \in S \cap C_i} w_u \left(\frac{1}{\alpha_i(w_v + \Delta x)} - \frac{1}{\alpha_i(w_v)} \right).
\end{aligned}
$$

Note that

$$
\begin{aligned}
& \left(\sum_{u \in S \cap B_i} w_u + \Delta x \right) \alpha_i(w_v + \Delta x) - \sum_{u \in S \cap B_i} w_u \alpha_i(w_v) \\
= \;& \frac{w(B_i \cap S) + \Delta x}{w(B_i) + \Delta x} \cdot w(C_i) - \frac{w(B_i \cap S)}{w(B_i)} \cdot w(C_i) \\
= \;& \frac{w(B \setminus S) \Delta x}{w(B_i)(w(B_i) + \Delta x)} \cdot w(C_i) \\
= \;& \alpha_i(w_v + \Delta x) \frac{w(B_i \setminus S)}{w(B_i)},
\end{aligned}
$$

and

$$
\left(\frac{1}{\alpha_i(w_v + \Delta x)} - \frac{1}{\alpha_i(w_v)} \right) = \left(\frac{w(B_i) + \Delta x}{w(C_i)} - \frac{w(B_i)}{w(C_i)} \right) = \frac{\Delta x}{w(C_i)}.
$$

So we have

$$
U_S(w_v + \Delta x) - U_S(w_v) = \frac{w(S \cap C_i)}{w(C_i)} \Delta x + \alpha_i(w_v + \Delta x) \frac{w(B_i \setminus S)}{w(B_i)} \Delta x.
$$

Therefore,

$$\begin{aligned}
& U_S(w_v + \Delta x) - U_S(w_v) > \Delta x \\
\iff\quad & \frac{w(S \cap C_i)}{w(C_i)} \cdot \Delta x + \alpha_i(w_v + \Delta x)\frac{w(B_i \setminus S)}{w(B_i)} \cdot \Delta x > \Delta x \\
\iff\quad & \frac{w(S \cap C_i)}{w(C_i)} + \alpha_i(w_v + \Delta x)\frac{w(B_i \setminus S)}{w(B_i)} > 1 \\
\iff\quad & \alpha_i(w_v + \Delta x)\frac{w(B_i \setminus S)}{w(B_i)} > \frac{w(C_i \setminus S)}{w(C_i)} \\
\iff\quad & \alpha_i(w_v + \Delta x)w(B_i \setminus S) > \frac{w(C_i \setminus S)}{\alpha_i(w_v)} \\
\implies\quad & \alpha_i(w_v + \Delta x)w(B_i \setminus S) > w(C_i \setminus S).
\end{aligned}$$

By the definition of $\alpha_i(w_v + \Delta x) = w(C_i)/(w(B_i) + \Delta x)$, then

$$\begin{aligned}
& \alpha_i(w_v + \Delta x)[(w(B_i) + \Delta x) - w(B_i \setminus S)] < w(C_i) - w(C_i \setminus S) \\
\implies\quad & \alpha_i(w_v + \Delta x) \cdot (w(S \cap B_i) + \Delta x) < w(S \cap C_i).
\end{aligned}$$

From the last inequality, we can deduce that $U_{S\cap B_i}(w_v+\Delta x) < w(S\cap C_i)$. It completes this proof. □

A.3 Main Techniques

Recall that our process of transforming the initial network $(G;\boldsymbol{w})$ into the ultimate network $(G^*; w^*_{v^1}, \cdots, w^*_{v^d}, \boldsymbol{w}_{-v})$ is first to RRB-split v into $\check{v}$ and $\hat{v}$, and then do $d-2$ times of RRB-split during the process of decreasing $w_{\check{v}}$ and increasing $w_{\hat{v}}$. In this subsection, we strictly define the following three processes: Decreasing Process (Algorithm 1), Increasing Process (Algorithm 2), and Adjusting Technique (Algorithm 3) in the form of algorithms. In the rest of this paper, we assume that other agents' weight profile $\boldsymbol{w}_{-v}$ is given. For the simplicity of notations, we shall use $(\tilde{G};\boldsymbol{w}_\Lambda)$ to characterize the network $(\tilde{G};\boldsymbol{w}_\Lambda,\boldsymbol{w}_{-v})$ and G^* to denote the specified Sybil network (V^*, E^*), where v splits itself into $\Lambda^* = \{v^1, \cdots, v^d\}$.

Decreasing Process. In the Decreasing Process, we decrease $w_{\check{v}}$ to make $x_{\check{v}u^i}$ equal to $w^*_{v^i}$ and then RRB-split v^i out. The detailed process is presented in Algorithm 1. We can understand Decreasing Process as a generalization of the process of decreasing $w_{\check{v}}$. Recall some properties of decreasing the weight of a vertex in previous subsections, such as the monotonicity of U_S for any $\check{v} \in S$ (Lemma A.9) and the changes of bottleneck decomposition (Proposition A.4). Intuitively, doing RRB-split impacts nothing, so in the next subsection, we shall show that some similar properties of Decreasing Process also hold.

Increasing Process. Symmetrically, in the Increasing Process, we increase $w_{\hat{v}}$ to make $x_{\hat{v}u^i}$ equal to $w^*_{v^i}$ and then RRB-split v^i out. The detailed process is presented in Algorithm 2. We can also understand Increasing Process as a generalization of the process of increasing $w_{\hat{v}}$. Since RRB-split impacts nothing, the properties of Increasing Process are also similar to the properties of the process of increasing the weight of a vertex, such as the upper bound of utility (Lemma A.10) and the changes of bottleneck decomposition (Proposition A.5), which will also be shown in the next subsection. Furthermore, compare with Decreasing Process, we need to let Increasing Process end when $\eta_{\hat{v}} = 1/\eta_v(G;\boldsymbol{w})$ at Stage 5 in Section 3.4, so we add a parameter η' for the additional stop condition.

Adjusting Technique. Now we consider a special case. Recall after the Stage 1 in both Section 3.3 and Section 3.4, the strategic agent v is RRB-split into two fictitious node $\check{v}$ and $\hat{v}$. It is possible that $\eta_{\hat{v}} = \eta_{\check{v}}$ on $\tilde{G}$, implying $\hat{v}$ and $\check{v}$ have the same η-ratio. Such a situation only happens when $\hat{v}$ and $\check{v}$ are in the same bottleneck pair, and in addition $\hat{v}$ and $\check{v}$ are both in B-class or C-class.

ALGORITHM 1: Decrease($\tilde{G}'$, $\boldsymbol{w}_{\Lambda'}$, $\check{v}'$)

Input: A Sybil network $\tilde{G}' = (\tilde{V}', \tilde{E}')$, weight profile $\boldsymbol{w}_{\Lambda'}$, $\check{v}' \in \Lambda'$.
Output: A Sybil network $\tilde{G} = (\tilde{V}, \tilde{E})$ of G, weight profile $\boldsymbol{w}_{\Lambda}$.

Let $\tilde{G} \leftarrow \tilde{G}', \boldsymbol{w}_{\Lambda} \leftarrow \boldsymbol{w}_{\Lambda'}$, $\check{v} \leftarrow \check{v}'$.
do
    while $d_{\check{v}} > 1$ **and** $\exists u^i \in \Gamma(\check{v}), x_{\check{v}u^i} = w^*_{v^i}$ **do**
        Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\check{v}) \setminus \{u^i\}$.
        Let $(\tilde{G}, (w_{v^i}, w_{\check{v}}), v^i, \check{v}) \leftarrow$ RRB-split$(\tilde{G}, \boldsymbol{w}_{\Lambda}, \check{v}, N_1, N_2)$.
    end
    Find the largest $z \geq 0$ such that $\forall u^i \in \Gamma(\check{v}), x_{\check{v}u^i} \geq w^*_{v^i}$ on $(\tilde{G}; w_{\check{v}} - z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}\}})$.
    Let $w_{\check{v}} \leftarrow w_{\check{v}} - z$.
while $z > 0$;
return $\tilde{G} = (\tilde{V}, \tilde{E}), \boldsymbol{w}_{\Lambda}$.

ALGORITHM 2: Increase($\tilde{G}'$, $\boldsymbol{w}_{\Lambda'}$, $\hat{v}'$, η')

Input: A Sybil network $\tilde{G}' = (\tilde{V}', \tilde{E}')$, weight profile $\boldsymbol{w}_{\Lambda'}$, $\hat{v}' \in \Lambda'$, threshold η'.
Output: A Sybil network $\tilde{G} = (\tilde{V}, \tilde{E}), \boldsymbol{w}_{\Lambda}, \hat{v}$.

Let $\tilde{G} \leftarrow \tilde{G}', \boldsymbol{w}_{\Lambda} \leftarrow \boldsymbol{w}_{\Lambda'}$, $\hat{v} \leftarrow \hat{v}'$.
do
    while $d_{\hat{v}} > 1$ **and** $\exists u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} = w^*_{v^i}$ **do**
        Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\hat{v}) \setminus \{u^i\}$.
        Let $(\tilde{G}, (w_{v^i}, w_{\hat{v}}), v^i, \hat{v}) \leftarrow$ RRB-split$(\tilde{G}, \boldsymbol{w}_{\Lambda}, \hat{v}, N_1, N_2)$.
    end
    Find the largest $z \geq 0$ such that:
      (1) $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} \leq w^*_{v^i}$ on $(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\hat{v}\}})$.
      (2) $\eta_{\hat{v}}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\hat{v}\}}) \geq \eta'$.
    Let $w_{\hat{v}} \leftarrow w_{\hat{v}} + z$.
while $z > 0$;
return $\tilde{G} = (\tilde{V}, \tilde{E}), \boldsymbol{w}_{\Lambda}, \hat{v}$.

Take the case $\eta_v(G;\boldsymbol{w}) < 1$ for an example, if we directly increase $w_{\hat{v}}$, since $w_{\hat{v}}$ and $w_{\check{v}}$ are in the same bottleneck pair, the allocation $x_{\check{v}u^i}$ might change. Thus, we apply Adjusting Technique (Algorithm 3) to deal with this situation before the Increasing Process or the Decreasing process. Specifically, when $\eta_{\check{v}} = \eta_{\hat{v}}$ happens, we continuously increase $w_{\hat{v}}$ to $w_{\hat{v}} + z$ and decrease $w_{\hat{v}}$ to $w_{\hat{v}} - z$ at the same time, such that the bottleneck decomposition remains unchanged during the weight changing process. The detailed process is presented in Algorithm 3.

We shall show that during the Adjusting Technique, v's utility remains unchanged and each other vertex's α-ratio remains unchanged (Proposition A.13). Furthermore, at the end of Adjusting Technique, when we increase $w_{\hat{v}}$ (or decrease $w_{\check{v}}$), the allocation $x_{\check{v}u^i}$ ($x_{\hat{v}u^i}$) will not be impacted (Proposition A.14).

The stop conditions of Adjusting Technique includes three cases: (1) $\hat{v}$ and $\check{v}$ are both replaced by several fictitious nodes with degree of one and then the ultimate network is achieved; (2) $\eta_{\hat{v}} \neq \eta_{\check{v}}$, which means $\hat{v}$ and $\check{v}$ are in the different bottleneck pairs or different classes; (3) a critical state at

ALGORITHM 3: Adjusting Technique($\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}', \hat{v}'$)

Input: A Sybil network $\tilde{G}' = (\tilde{V}', \tilde{E}')$, weight profile $\boldsymbol{w}_{\Lambda'}$, $\check{v}', \hat{v}' \in \Lambda'$.
Output: A Sybil network $\tilde{G} = (\tilde{V}, \tilde{E})$, weight profile $\boldsymbol{w}_{\Lambda}$, $\check{v}, \hat{v} \in \Lambda$.

1 Let $\tilde{G} \leftarrow \tilde{G}', \boldsymbol{w}_{\Lambda} \leftarrow \boldsymbol{w}_{\Lambda'}, \check{v} \leftarrow \check{v}', \hat{v} \leftarrow \hat{v}'$.
2 **do**
3   **while** $d_{\check{v}} > 1$ **and** $\exists u^i \in \Gamma(\check{v}), x_{\check{v}u^i} = w^*_{v^i}$ **do**
4   Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\check{v}) \setminus \{u^i\}$.
5   Let $(\tilde{G}, (w_{v^i}, w_{\check{v}}), v^i, \check{v}) \leftarrow$ RRB-split$(\tilde{G}, \boldsymbol{w}_{\Lambda}, \check{v}, N_1, N_2)$.
6   **end**
7   **while** $d_{\hat{v}} > 1$ **and** $\exists u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} = w^*_{v^i}$ **do**
8   Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\hat{v}) \setminus \{u^i\}$.
9   Let $(\tilde{G}\ (w_{v^i}, w_{\hat{v}}), v^i, \hat{v},) \leftarrow$ RRB-split$(\tilde{G}, \boldsymbol{w}_{\Lambda}, \hat{v}, N_1, N_2)$.
10   **end**
11   **if** $\eta_{\check{v}} \neq \eta_{\hat{v}}$ **then**
12   **return** $\tilde{G} = (\tilde{V}, \tilde{E}), \boldsymbol{w}_{\Lambda}, \check{v}, \hat{v}$.
13   **end**
14   Find the largest $z \geq 0$ such that:
        (1) $\mathcal{B}(\tilde{G}; \boldsymbol{w}_{\Lambda}) = \mathcal{B}(\tilde{G}; w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}})$.
        (2) $\forall u^i \in \Gamma(\check{v}), x_{\check{v}u^i} \geq w^*_{v^i}$ on $(\tilde{G}; w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}})$.
        (3) $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} \leq w^*_{v^i}$ on $(\tilde{G}; w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}})$.
    Let $w_{\check{v}} \leftarrow w_{\check{v}} - z$, $w_{\hat{v}} \leftarrow w_{\hat{v}} + z$.
15 **while** $z > 0$;
16 **return** $\tilde{G} = (\tilde{V}, \tilde{E}), \boldsymbol{w}_{\Lambda}, \check{v}, \hat{v}$.

which if further increasing $w_{\hat{v}}$ or decreasing $w_{\check{v}}$ by a sufficiently small $\epsilon > 0$, the bottleneck pair containing $\hat{v}$ and $\check{v}$ will be decomposed into two pairs.

Note that when executing Adjusting Technique, once the current allocation $x_{\hat{v}u^i} = w^*_{v^i}$ or $x_{\check{v}u^i} = w^*_{v^i}$, we shall split the corresponding fictitious node out and redefine $\hat{v}$ or $\check{v}$, as shown in line 3 to 8, before increasing $w_{\hat{v}}$ and decreasing $w_{\check{v}}$. If there is a value $z > 0$ satisfying the three conditions in line 11, then let $w_{\check{v}} \leftarrow w_{\check{v}} - z$, $w_{\hat{v}} \leftarrow w_{\hat{v}} + z$, obtain a resulting network $\tilde{G}$ with weight profile $(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}})$ and go back to line 2. The condition to adjust the weight of $\hat{v}$ and $\check{v}$ simultaneously in Adjusting Technique is that $\check{v}$ and $\hat{v}$ are in the same bottleneck pair, and they are both in B-class or C-class. So if we decrease $w_{\check{v}}$ to $w_{\check{v}} - z$ and increase $w_{\hat{v}}$ to $w_{\hat{v}} + z$, and keep the bottleneck decomposition unchanged, then $\check{v}$ and $\hat{v}$ are still in the same bottleneck pair, whose α-ratio is not impacted, and therefore the sum of utilities from $\check{v}$ and $\hat{v}$ is also not impacted. In addition, if the current allocation on edge $(\check{v}, u^i)$ (or edge $(\hat{v}, u^i)$) is equal to the ultimate weight $w^*_{v^i}$, then Adjusting Technique will executes RRB-split to split v^i out and let its weight of $w^*_{v^i}$ be fixed. Therefore, after playing Adjusting Technique, the strategic agent v's utility is unaffected.

A.4 Properties of Main Techniques in Appendix A.3

Firstly, we show that the Adjusting Technique preserve all agents' weight, the strategic agent's utility, and some properties of the BD allocation, but may change the η-ratio.

Let X be the BD allocation on the network $(G; \boldsymbol{w})$. Recall in the Stage 1 of both Section 3.3 and Section 3.4, we partition the strategic agent v' neighbors $\Gamma(v)$ into $\check{N} = \{u^i | x_{vu^i} > w^*_{v^i}\}$ and

$\hat{N} = \{u^i | x_{vu^i} \leq w^*_{v^i}\}$, and play RRB-split$(G, \boldsymbol{w}, v, \check{N}, \hat{N})$. Let $(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}', \hat{v}')$ be the obtained network, in which fictitious node set $\Lambda' := \{\check{v}', \hat{v}'\}$ with $\boldsymbol{w}_{\Lambda'} = (w_{\check{v}'}, w_{\hat{v}'})$.

Proposition A.13. *Let $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v}, \hat{v})$ be the output of Adjusting Technique$(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}', \hat{v}')$. Then*
(1) $w(\Lambda) = w(\Lambda')$.
(2) $U_v(\tilde{G}; \boldsymbol{w}_\Lambda) = U_v(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$.
(3) $\forall u^i \in \Gamma(\check{v})$, $x_{\check{v}u^i} \geq w^*_{v^i}$ and $\forall u^i \in \Gamma(\hat{v})$, $x_{\hat{v}u^i} \leq w^*_{v^i}$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$.
(4) $\forall v' \in \Lambda \setminus \{\check{v}, \hat{v}\}$, $\eta_{v'} = \eta_v(G; \boldsymbol{w})$, $d_{v'} = 1$, $w_{v'} = w^*_{v^i}$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$.
(5) $\eta_{\check{v}}(\tilde{G}; \boldsymbol{w}_\Lambda) = \eta_v(G; \boldsymbol{w})$, $\eta_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda) \leq \eta_v(G; \boldsymbol{w})$.

Proof. Firstly, $w_{\check{v}'} > 0$ and it is possible $w_{\hat{v}'} = 0$. This implies $\eta_{\hat{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'}) \leq \eta_{\check{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'}) = \eta_v(G; \boldsymbol{w})$ by Lemma 3.2. Furthermore, if $\eta_{\hat{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'}) < \eta_{\check{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$, then $w_{\hat{v}'} = 0$. In the following, we show Proposition A.13-(1)(2)(3)(4)(5) hold at each step in the Adjusting Technique (Algorithm 3).

In the two While loops (line 3-8), several times of RRB-split are executed, so Proposition A.13-(1)(2)(3)(5) hold by Lemma 3.2. In addition, there is only one fictitious node v^i is split with $d_{v^i} = 1$ and $w_{v^i} = w^*_{v^i}$, so Proposition A.13-(4) holds.

If Algorithm 3 does not return the output in line 10, it must be $\eta_{\check{v}} = \eta_{\hat{v}}$, implying $\check{v}$ and $\hat{v}$ are in the same bottleneck pair. Suppose $\mathcal{B}(\tilde{G}; \boldsymbol{w}_\Lambda) = \{(B_1, C_1), \cdots, (B_k, C_k)\}$ and $\check{v}, \hat{v} \in (B_j \cup C_j)$. For the sake of convenience, we shall omit $\tilde{G}$ without ambiguity when analyzing line 11.

Firstly, Proposition A.13-(3) holds because of the second and third conditions in line 11. By the first condition of $\mathcal{B}(\boldsymbol{w}_\Lambda) = \mathcal{B}(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}})$, each bottleneck pair (B_ℓ, C_ℓ), $\forall \ell \in \{1, \cdots, k\}$, remains unchanged. Note that with each weight update, only $w_{\check{v}}$ decreases by z and $w_{\hat{v}}$ increases by z, so $w(\Lambda)$ remains unchanged and Proposition A.13-(1)(4) hold.

Furthermore, the weight of each vertex in $\tilde{V} \setminus \{\check{v}, \hat{v}\}$ remains unchanged, which means for each $u \in \tilde{V} \setminus (B_j \cup C_j)$, $\eta_u(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) = \eta_u(\boldsymbol{w}_\Lambda)$ and $U_u(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) = U_u(\boldsymbol{w}_\Lambda)$. From this observation, we shall focus on the bottleneck pair (B_j, C_j). The proof of Proposition A.13-(2)(5) is mainly based on the fact that $\alpha_j(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) = \alpha_j(\boldsymbol{w}_\Lambda)$. Thus, $\forall u \in B_j \cup C_j$, $\eta_u(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) = \eta_u(\boldsymbol{w}_\Lambda)$, so Proposition A.13-(5) holds.

Finally,

$$\begin{aligned} U_v(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) - U_v(\boldsymbol{w}_\Lambda) &= U_{\check{v}}(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) - U_{\check{v}}(\boldsymbol{w}_\Lambda) \\ &\quad + U_{\hat{v}}(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) - U_{\hat{v}}(\boldsymbol{w}_\Lambda) \\ &= \eta_{\check{v}}(w_{\check{v}} - z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\check{v}, \hat{v}\}}) \cdot (z - z) \\ &= 0, \end{aligned}$$

which means Proposition A.13-(2) holds. □

According to the stop condition of the Adjusting Technique, on the output network $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v}, \hat{v})$, $\hat{v}$ and $\check{v}$ either belong to two different bottleneck pairs, or will belong to two different bottleneck pairs if their weights are further increased and decreased together respectively by any small $\epsilon > 0$. Intuitively, for the former case, we shall prove that in the next stage, simply increasing $\hat{v}$'s weight (or decreasing $\check{v}$'s weight) does not influence the bottleneck pair containing $\check{v}$ ($\hat{v}$ respectively). Now we need to prove that this property also holds for the later case, as on $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v}, \hat{v})$ simply increasing $\hat{v}$'s weight (or decreasing $\check{v}$'s weight) by any small $\epsilon > 0$, $\hat{v}$ and $\check{v}$ will also belong to two different bottleneck pairs.

Proposition A.14. *Let $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v}, \hat{v})$ be the output of Adjusting Technique$(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}', \hat{v}')$. If $\hat{v}$ and $\check{v}$ are in the same bottleneck pair and are both in B-class or in C-class on $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v}, \hat{v})$, then for any sufficiently small $\epsilon > 0$*

(1) if $\check{v}$ is a C-class vertex on $(\tilde{G};\boldsymbol{w}_\Lambda)$ with $\alpha_{\check{v}}(\tilde{G};\boldsymbol{w}_\Lambda) < 1$, then

$$\alpha_{\check{v}}(\tilde{G};w_{\check{v}}-\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) < \alpha_{\hat{v}}(\tilde{G};w_{\check{v}}-\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) = \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda).$$

(2) if $\hat{v}$ is a B-class vertex on $(\tilde{G};\boldsymbol{w}_\Lambda)$ with $\alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda) < 1$,

$$\alpha_{\hat{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) < \alpha_{\check{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) = \alpha_{\check{v}}(\tilde{G};\boldsymbol{w}_\Lambda).$$

Proof. Since $\check{v}$ and $\hat{v}$ are in the same bottleneck pair on $(\tilde{G},\boldsymbol{w}_\Lambda,\check{v},\hat{v})$, $\eta_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda) = \eta_{\check{v}}(\tilde{G};\boldsymbol{w}_\Lambda)$. Let $\mathcal{B}(\tilde{G};\boldsymbol{w}_\Lambda) = \{(B_1,C_1),\cdots,(B_k,C_k)\}$ be the bottleneck decomposition of network $(\tilde{G},\boldsymbol{w}_\Lambda,\check{v},\hat{v})$. Suppose that $\check{v},\hat{v} \in (B_j \cup C_j)$. According to the stop condition, we know for arbitrary small $\epsilon > 0$, $\mathcal{B}(\tilde{G};\boldsymbol{w}_\Lambda) \neq \mathcal{B}(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$. Here we only prove Proposition A.14-(1) as the proof of Proposition A.14-(2) is similar. To compare the α-ratio of $\hat{v}$ and $\check{v}$ on the network $(\tilde{G};w_{\check{v}}-\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v}\}})$, we first analyze how α-ratio changes from $(\tilde{G};\boldsymbol{w}_\Lambda)$ to $(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$, then analyze it from $(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$ to $(\tilde{G};w_{\check{v}}-\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v}\}})$.

We first prove that $\forall u \in \tilde{V}\setminus(B_j\cup C_j)$, (B_u,C_u) on $(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$ is equal to (B_u,C_u) on $(\tilde{G};\boldsymbol{w}_\Lambda)$. Combined with the fact that $\forall u \in \tilde{V}\setminus\{\check{v},\hat{v}\}$, w_u remains unchanged, it is enough for us to focus on the changes of (B_j,C_j). Suppose that $\epsilon > 0$ is sufficiently small. Since $\eta_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda) > 1$ and $\eta_{\hat{v}}$ is continuous by Proposition A.3, we have $\eta_{\hat{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) > 1$, which means $\hat{v}$ is a C-class vertex on $(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$. Thus, by the continuity of $\eta_u(x)$, for each $u \in \tilde{V}$ with $\alpha_u(\tilde{G};\boldsymbol{w}_\Lambda) > \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda)$ (or $\alpha_u(\tilde{G};\boldsymbol{w}_\Lambda) < \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda)$), then $\alpha_u(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) > \alpha_{\hat{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$ (or $\alpha_u(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) < \alpha_{\hat{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$). Combined with Proposition A.4-(2), we have (B_u,C_u) on $(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$ is equal to (B_u,C_u) on $(\tilde{G};\boldsymbol{w}_\Lambda)$. Similarly, we have (B_u,C_u) on $(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$ is equal to (B_u,C_u) on $(\tilde{G};\boldsymbol{w}_\Lambda)$. Furthermore, by the proof above and the Proposition A.4-(1), we have $C(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}}) = C(\tilde{G};\boldsymbol{w}_\Lambda)$ and $B(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}}) = B(\tilde{G};\boldsymbol{w}_\Lambda)$.

The subsequent analysis focuses on the changes of (B_j,C_j). In this proof, assume that the function $w(S)$ is define on the weight profile $(\boldsymbol{w}_\Lambda)$, while not the weight profile $(w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$. Let $\{(B'_j,C'_j),(B'_{j+1},C'_{j+1}),\cdots,(B'_{j+\ell},C'_{j+\ell})\}$ be the disjoint pairs that the pair (B_j,C_j) splits into on $(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$. Since $\mathcal{B}(\tilde{G};\boldsymbol{w}_\Lambda) \neq \mathcal{B}(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$, we have $\ell \geq 1$. By $C(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}}) = C(\tilde{G};\boldsymbol{w}_\Lambda)$ and $B(\tilde{G};w_{\check{v}}-z,w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}}) = B(\tilde{G};\boldsymbol{w}_\Lambda)$, we have $\forall t \in \{0,\cdots,\ell\}, B'_{j+t} \subsetneq B_j, C'_{j+t} \subsetneq C_j$. We claim that $\check{v} \in C'_j$ and $\hat{v} \notin C'_j$.

Note that $\alpha_j = w(C_j)/w(B_j)$. Let α'_{j+t} be the α-ratio of pair (B'_{j+t},C'_{j+t}) on $(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$. On one hand, $w(C'_j)/w(B'_j) \geq \alpha_j$. Recall the definition of Bottleneck Decomposition, (B_j,C_j) is the j-th maximal bottleneck pair on $(\tilde{G};\boldsymbol{w}_\Lambda)$, so there is no vertices set $S \subseteq \tilde{V}_j$ satisfying $w(\Gamma(S))/w(S) < w(C_j)/w(B_j)$. On the other hand, $\alpha'_j < \alpha_j$. Note that $B_j \subseteq \tilde{V}_j$ is a candidate bottleneck and $(w(C_j)+\epsilon-\epsilon)/w(B_j) = w(C_j)/w(B_j) = \alpha_j$ on $(\tilde{G};w_{\check{v}}-\epsilon,w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$. Combined with $B'_j \subsetneq B_j$ and the concept of *maximal bottleneck*, we can prove that $\alpha'_j < \alpha_j$.

If $\check{v},\hat{v} \in C'_j$ or $\check{v},\hat{v} \notin C'_j$, it is easy to verify $\alpha'_j = w(C'_j)/w(B'_j)$, since there are two trivial cases $\alpha'_j = w(C'_j)/w(B'_j)$ and $\alpha'_j = (w(C'_j)+\epsilon-\epsilon)/w(B'_j)$. Then we have $\alpha'_j = w(C'_j)/w(B'_j) \geq \alpha_j$, which leads to a contradiction. Thus, there is only one of $\check{v}$ and $\hat{v}$ in C'_j. The case that only $\hat{v}$ is in C'_j also leads to a contradiction as above. The claim that $\check{v} \in C'_j$ and $\hat{v} \notin C'_j$ is true.

Thus, $\alpha'_j = (w(C'_j)-\epsilon)/(w(B'_j)) < \alpha_j$ for any sufficiently small ϵ. Take the limit on this formula,

$$\lim_{\epsilon\to 0}\frac{w(C'_j)-\epsilon}{w(B'_j)} = \frac{w(C'_j)}{w(B'_j)} \leq \alpha_j.$$

Combined with $w(C'_j)/w(B'_j) \geq \alpha_j$, we have $w(C'_j)/w(B'_j) = \alpha_j$. Since $w(C_j)/w(B_j) = \alpha_j$, we have $w(C_j \setminus C'_j)/w(B_j \setminus B'_j) = \alpha_j$.

Similarly as above, we can prove $\forall t \in \{1, \cdots, \ell\}, w(C'_{j+t})/w(B'_{j+t}) = \alpha_j$. Furthermore, $\hat{v} \in C'_{j+\ell}$ and $\alpha'_{j+\ell} > \alpha_j$ on $(\tilde{G}; w_{\check{v}} - \epsilon, w_{\hat{v}} + \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$.

Now we consider the bottleneck decomposition of the network $(\tilde{G}; w_{\check{v}} - \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v}\}})$. It is obtained by decreasing the weight of $\hat{v}$ on the network $(\tilde{G}; w_{\check{v}} - \epsilon, w_{\hat{v}} + \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$ from $w_{\hat{v}} + \epsilon$ to $w_{\hat{v}}$. Then the α-ratio of pair $(B'_{j+\ell}, C'_{j+\ell})$ decreases continuously from $(w(C'_{j+\ell}) + \epsilon)/w(B_{j+\ell})$ to $w(C'_{j+\ell})/w(B_{j+\ell})$, where $w(C'_{j+\ell})/w(B'_{j+\ell})$ is exactly equal to α_j. Thus, we have $\alpha_{\check{v}}(\tilde{G}; w_{\check{v}} - \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) < \alpha_{\hat{v}}(\tilde{G}; w_{\check{v}} - \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) = \alpha_{\check{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$. Furthermore, recall the beginning of this proof, $\hat{v}$ will be an increasing fictitious node on $(\tilde{G}; w_{\check{v}} - \epsilon, w_{\hat{v}} + \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\check{v},\hat{v}\}})$ for any sufficiently small $\epsilon > 0$. Thus, after weight of $\hat{v}$ decreases from $w_{\hat{v}} + \epsilon$ to $w_{\hat{v}}$, $\hat{v}$ will also be an increasing fictitious node on $(\tilde{G}; w_{\hat{v}} + \epsilon, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$. The desired result has been proved.

□

Finally, we show how agent v's utility changes during the Decreasing Process (Algorithm 1) and the Increasing Process (Algorithm 2). For simplicity, let $(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}')$ be a Sybil network with a decreasing fictitious node $\check{v}' \in \Lambda'$ and $(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \hat{v}')$ a Sybil network with an increasing fictitious node $\hat{v}' \in \Lambda'$.

The following proposition is a generalization of Lemma A.9 and Proposition A.4. More specifically, we show the monotonicity of utility of v also holds during Decreasing Process (Proposition A.15-(2)) and the vertices whose α-ratio is greater than the decreasing fictitious node are not impacted if the increasing fictitious node is a C-class vertex (Proposition A.15-(3)).

Proposition A.15. *Let $(\tilde{G}, \boldsymbol{w}_\Lambda, \check{v})$ be the output of Decrease$(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \check{v}')$. We have*

(1) $d_{\check{v}} = 1$ and $w_{\check{v}} = w^*_{v^i}$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$, where u^i is the neighbor of $\check{v}$.
(2) $U_v(\tilde{G}; \boldsymbol{w}_\Lambda) \leq U_v(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$.
(3) if $\check{v}'$ is a C-class vertex with $\alpha_{\check{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'}) < 1$, then for any vertex $u \in \tilde{V}'$ with $\alpha_u(\tilde{G}'; \boldsymbol{w}_{\Lambda'}) > \alpha_{\check{v}'}(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$, we have $(B_u(\tilde{G}; \boldsymbol{w}_\Lambda), C_u(\tilde{G}; \boldsymbol{w}_\Lambda)) = (B_u(\tilde{G}'; \boldsymbol{w}_{\Lambda'}), C_u(\tilde{G}'; \boldsymbol{w}_{\Lambda'}))$ and $\alpha_u(\tilde{G}; \boldsymbol{w}_\Lambda) = \alpha_u(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$.

Proof. Proposition A.15-(1) is directly from the stop condition of Decreasing Process.

Proposition A.15-(2) holds at each step of Algorithm 1. Precisely, in the While loop (line 3-5), U_v remains unchanged from the property of RRB-split (Lemma 3.2). In the line 6-7, the weight of $\check{v}$ decreases and $U_\Lambda(\tilde{G}; w_{\check{v}} - z, \boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) \leq U_\Lambda(\tilde{G}; \boldsymbol{w}_\Lambda)$ by the monotonicity of U_Λ since $\check{v} \in \Lambda$ (Lemma A.9).

For Proposition A.15-(3), we first show $\alpha_{\check{v}}(G; \boldsymbol{w}_\Lambda)$ is non-increasing. Specifically, in line 3-5 $\alpha_{\check{v}}$ remains unchanged and in line 6-7, $\alpha_{\check{v}}(\tilde{G}; w_{\check{v}} - z, \boldsymbol{w}_{\Lambda\setminus\{\check{v}\}}) \leq \alpha_{\check{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$ by Proposition A.1 since $\check{v}$ is a C-class vertex all the time. Then we can apply Proposition A.4-(2) to deduce Proposition A.15-(3). □

The following proposition is a generalization of Lemma A.10 and Proposition A.5-(2). More specifically, we show a similar upper bound of v's utility holds during Increasing Process (Proposition A.16-(2)) and the vertices whose α-ratio is greater than the increasing fictitious node are not impacted if the increasing fictitious node is a B-class vertex (Proposition A.16-(3)).

Proposition A.16. *Let $(\tilde{G}, \Lambda, \boldsymbol{w}_\Lambda, \hat{v})$ to be the output of Increase$(\tilde{G}', \boldsymbol{w}_{\Lambda'}, \hat{v}', \eta')$. We have*

(1) if $\eta_{\hat{v}} > \eta'$, then $d_{\hat{v}} = 1$ and $w_{\hat{v}} = w^*_{v^i}$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$, where u^i is the neighbor of $\hat{v}$.

(2) *if the following two conditions hold (1) $\hat{v}'$ is a B-class vertex with $\alpha_{\hat{v}'} < 1$ on $(\tilde{G}';\boldsymbol{w}_{\Lambda'})$; (2) each fictitious node $v' \in \Lambda' \setminus \{\hat{v}'\}$ is a B-class vertex on $(\tilde{G}';\boldsymbol{w}_{\Lambda'})$, then $U_v(\tilde{G};\boldsymbol{w}_{\Lambda}) \leq U_v(\tilde{G}';\boldsymbol{w}_{\Lambda'}) + \eta_{\hat{v}'}(\tilde{G}';\boldsymbol{w}_{\Lambda'}) \cdot (w(\Lambda) - w(\Lambda'))$.*

(3) *if $\hat{v}'$ is a B-class vertex with $\alpha_{\hat{v}'}(\tilde{G}';\boldsymbol{w}_{\Lambda'}) < 1$, then for any vertex $u \in \tilde{V}'$ with $\alpha_u(\tilde{G}';\boldsymbol{w}_{\Lambda'}) > \alpha_{\hat{v}'}(\tilde{G}';\boldsymbol{w}_{\Lambda'})$, we have $(B_u(\tilde{G};\boldsymbol{w}_{\Lambda}), C_u(\tilde{G};\boldsymbol{w}_{\Lambda})) = (B_u(\tilde{G}';\boldsymbol{w}_{\Lambda'}), C_u(\tilde{G}';\boldsymbol{w}_{\Lambda'}))$ and $\alpha_u(\tilde{G};\boldsymbol{w}_{\Lambda}) = \alpha_u(\tilde{G}';\boldsymbol{w}_{\Lambda'})$.*

Proof. For the proof of Proposition A.16-(1), note that the Algorithm 2 ends in the line 7. In addition, the condition $\eta_{\hat{v}} > \eta'$ means that it is not $\eta_{\hat{v}}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) \geq \eta'$ that limits $z = 0$. So Proposition A.16-(1) holds directly.

Proposition A.16-(2) is an extension of Lemma A.10. We first claim that $\eta_{\check{v}} \leq \eta_{\check{v}'}(\tilde{G}';\boldsymbol{w}_{\Lambda'})$ at each step in the algorithm. In the While loop (line 3-5), $\eta_{\hat{v}}$ is non-increasing by Lemma 3.2. In the line 6, the weight of $\hat{v}$ is increasing and $\eta_{\hat{v}}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$ is non-increasing by Proposition A.3. In addition, since $\hat{v}$ is always in B-class and $w_{\hat{v}}$ increases, by Proposition A.5-(1), each fictitious node $v' \in \Lambda\{\hat{v}\}$ remains in B-class.

Now we analyze the utility function. In the While loop (line 3-5), U_v remains unchanged by Lemma 3.2-(3). In the line 7, combined with the condition $\forall v' \in \Lambda$, v' is a B-class vertex, we have $U_v(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) - U_v(\tilde{G};\boldsymbol{w}_{\Lambda}) \leq \eta_{\hat{v}}(\tilde{G};\boldsymbol{w}_{\Lambda}) \cdot z \leq \eta_{\hat{v}'}(\tilde{G}';\boldsymbol{w}_{\Lambda'}) \cdot z$ by Lemma A.10. Add up these utilities, then the desired result has been proved.

Proposition A.16-(3) is an extension of Proposition A.5-(2) and its proof is similar to the proof of Proposition A.15-(3). □

B PROOF OF LEMMA 3.2

Proof of Lemma 3.2. To obtain the first claim, we shall prove the three conditions of market equilibrium in Definition 2.1 are satisfied under $(\tilde{\boldsymbol{p}}, \tilde{X})$. Based on RRB-split and the condition that $(\boldsymbol{p}, X)$ is a market equilibrium on $(G;\boldsymbol{w})$, $\sum_{w\in\Gamma(u)} \tilde{x}_{uw} = \sum_{w\in\Gamma(u)} x_{uw} = w_u$ for each $u \neq v^1, v^2$, and $\sum_{u\in\Gamma(v^\ell)} \tilde{x}_{v^\ell u} = \sum_{u\in\Gamma(v^\ell)} x_{vu} = w_{v^\ell}$, $\ell = 1, 2$. Market clearance condition holds. Next we focus on the case that v is a B-class vertex on $(G;\boldsymbol{w})$, without loss of generality, $v \in B_i$, to prove the conditions of budget constraint and individual optimality. Under this case, all neighbors uploading resource to v must be in C_i by BD allocation mechanism. Recall the prices defined in Proposition 2.6 are $p_u = \alpha_i w_u$ if $u \in B_i$ and $p_u = w_u$ if $u \in C_i$. Based on the construction of $\tilde{\boldsymbol{p}}$ and $\tilde{X}$, it is not hard to see the total payment that v^ℓ, $\ell = 1, 2$, pays out is

$$\text{payment}(v^\ell) = \sum_{u\in\Gamma(v^\ell)} \frac{\tilde{x}_{uv^\ell}}{w_u}\tilde{p}_u = \sum_{u\in\Gamma(v^\ell)\cap C_i} x_{uv} = \sum_{u\in\Gamma(v^\ell)\cap C_i} \alpha_i x_{vu} = \alpha_i w_{v^\ell}, \tag{3}$$

as $\tilde{p}_u = p_u = w_u$ when $u \in \Gamma(v^\ell) \cap C_i$ and $x_{uv} = \alpha_i x_{vu}$ by BD allocation mechanism. On the other hand, the revenue v^ℓ receives from $(\tilde{\boldsymbol{p}}, \tilde{X})$ is $\tilde{p}_{v^\ell} = \frac{w_{v^\ell}}{w_v} p_v = \alpha_i w_{v^\ell}$, showing the budget constraint holds for v^ℓ, and in addition v^ℓ's payment is just equals to its revenue, reaching the payment's upper bound. At the same time,

$$U_{v^\ell}(\tilde{X}) = \sum_{u\in\Gamma(v^\ell)} \tilde{x}_{uv^\ell} = \sum_{u\in\Gamma(v^\ell)\cap C_i} x_{uv} = \alpha_i w_{v^\ell}, \tag{4}$$

equal to v^ℓ's payment by Equation (3). The fact that this payment has reached the upper bound, guarantees v^ℓ receives its maximal utility $U_{v^\ell}(\tilde{X})$ and thus the individual optimality condition holds for v^ℓ.

For the case that $v \in C_i$ on G, the payment of v^ℓ, $\ell = 1, 2$, is

$$\text{payment}(v^\ell) = \sum_{u\in\Gamma(v^\ell)} \frac{\tilde{x}_{uv^\ell}}{w_u}\tilde{p}_u = \sum_{u\in\Gamma(v^\ell)\cap B_i} \alpha_i x_{uv} = \sum_{u\in\Gamma(v^\ell)\cap B_i} x_{vu} = w_{v^\ell}. \quad (5)$$

Equation (5) holds since $\tilde{p}_u = p_u = \alpha_i w_u$ and $x_{vu} = \alpha_i x_{uv}$ when $v \in C_i$ and $u \in B_i$. The revenue of v^ℓ is $\tilde{p}_{v^\ell} = \frac{w_{v^\ell}}{w_v} p_v = w_{v^\ell}$, as $p_v = w_v$ when v is a C-class vertex. Therefore, v^ℓ's payment is equal to its revenue, meaning the budget holds, and the payment reaches its upper bound. We can compute the utility of v^ℓ

$$U_{v^\ell}(\tilde{X}) = \sum_{u\in\Gamma(v^\ell)} \tilde{x}_{uv^\ell} = \sum_{u\in\Gamma(v^\ell)\cap B_i} x_{uv} = \sum_{u\in\Gamma(v^\ell)\cap B_i} \frac{1}{\alpha_i} x_{vu} = \frac{1}{\alpha_i} w_{v^\ell} = \frac{1}{\alpha_i}\text{payment}(v^\ell). \quad (6)$$

Clearly, as payment(v^ℓ) reaches its upper bound, v^ℓ obtains the maximal utility $U_{v^\ell}(\tilde{X})$. The analysis for other agent $u \neq v^1, v^2$ is similar.

Based on the construction of $\tilde{X}$, we have $U_u(\tilde{X}) = \sum_{w\in\Gamma(u)} \tilde{x}_{wu} = \sum_{w\in\Gamma(u)} x_{wu} = U_u(X)$ and as $w_u > 0$, $\eta_u(\tilde{X}) = U_u(\tilde{X})/w_u = U_u(X)/w_u = \eta_u(X)$ by Definition 2.8. In addition, $U_{v^1}(\tilde{X}) + U_{v^2}(\tilde{X}) = \sum_{u\in N_1} x_{uv} + \sum_{u\in N_2} x_{uv} = \sum_{u\in\Gamma(v)} x_{uv} = U_v(X)$. Thus the second and the third claims hold.

If $v \in B_i$ on G, then $\eta_v(X) = \alpha_i$ and otherwise $\eta_v(X) = 1/\alpha_i$ by Definition 2.8. Based on Equation (4) and Equation (6), we know $U_{v^\ell}(\tilde{X}) = \alpha_i w_{v^\ell}$ if $v \in B_i$ on G and $U_{v^\ell}(\tilde{X}) = w_{v^\ell}/\alpha_i$ if $v \in C_i$ on G. When $w_{v^\ell} > 0$, $\eta_{v^\ell}(\tilde{X}) = U_{v^\ell}(\tilde{X})/w_{v^\ell}$ by Definition 2.8. So $\eta_{v^\ell}(\tilde{X}) = \eta_v(X)$. The fourth claim holds.

For the last claim, without loss of generality assume $w_{v^1} = 0$, and thus $w_{v^2} = w_v$. So $x_{vu} = 0$ for each neighbor $u \in N_1$ and all neighbors satisfying $x_{vu} > 0$ must be in N_2. Now let us consider the case that $v \in B_i$ on G. By Proposition 2.3, we know $u \in C_j$ with $j \leq i$. Consider the neighbor $u' \in N_1$, such that $u' \in C_{j'}$ and $j' (\leq i)$ is the largest index, and therefore $\alpha_{j'} = \max\{\alpha_u(X) | u \in N_1\}$. Then the bottleneck decomposition of $\tilde{G}$ is $\tilde{\mathcal{B}} = \{(\tilde{B}_1, \tilde{C}_1), \cdots, (\tilde{B}_{j'}, \tilde{C}_{j'}), \cdots, (\tilde{B}_i, \tilde{C}_i), \cdots (\tilde{B}_k, \tilde{C}_k)\}$, in which $(\tilde{B}_\ell, \tilde{C}_\ell) = (B_\ell, C_\ell)$, for each $\ell \neq j', i$, $\tilde{B}_{j'} = B_{j'} \cup \{v^1\}$, $\tilde{C}_{j'} = C_{j'}$, and $\tilde{B}_i = B_i \setminus \{v\} \cup \{v^2\}$, $\tilde{C}_i = C_i$. Obviously, the bottleneck pair's α-ratio $\tilde{\alpha}_\ell = \alpha_\ell$ for each $\ell = 1, \cdots, k$. By Definition 2.8, since $v^1 \in \tilde{B}_{j'}$ and $v \in B_i$, we know $\eta_{v^1}(\tilde{X}) = \alpha_{j'} \leq \alpha_i = \eta_v(X)$. In addition, as all $u \in N_1$ are C-class vertices, and then $\alpha_u(X) = 1/\eta_u(X)$, $\eta_{v^1}(\tilde{X}) = \max\{1/\eta_u(X) | u \in N_1\}$. The analysis for the case that $v \in C_i$ on G is similar. □

C PROOF OF LEMMA 3.4

Claim 1 and Claim 2 are directly from the properties of RRB-split in Lemma 3.2 and the properties of Adjusting Technique in Proposition A.13, respectively. In addition, there are more properties of $(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1})$ and $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2})$ as follows.

Claim 10. *Let $(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1})$ be the network after Stage 1, then*

(1) $w(\Lambda^1) = w_v$.

(2) $\eta_{\hat{v}^1}(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}) \leq \eta_{\check{v}^1}(\tilde{G}^1; \boldsymbol{w}_{\Lambda^1}) = \alpha_v(G; \boldsymbol{w}) < 1$.

Claim 11. *Let $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2})$ be the network after Stage 2, then*

(1) $w(\Lambda^2) = w(\Lambda^1)$.

(2) $\forall v' \in \Lambda^2 \setminus \{\check{v}^2, \hat{v}^2\}$, v' is a B-class vertex with $\alpha_{v'} = \alpha_v(G; \boldsymbol{w})$, $d_{v'} = 1$, $w_{v'} = w^*_{v^i}$ on $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2})$;

(3) $\forall u^i \in \Gamma(\check{v}^2)$, $x_{\check{v}^2 u^i} \geq w^*_{v^i}$ and $\check{v}^2$ is a B-class vertex with $\alpha_{\check{v}^2} = \alpha_v(G; \boldsymbol{w}) < 1$;

(4) $\forall u^i \in \Gamma(\hat{v}^2)$, $x_{\hat{v}^2 u^i} \leq w^*_{v^i}$ and $\hat{v}^2$ is a B-class vertex with $\alpha_{\hat{v}^2} \leq \alpha_{\check{v}^2} = \alpha_v(G; \boldsymbol{w}) < 1$.

Recall that there are three stop conditions of the Adjusting Technique (Algorithm 3), in which the first one, $\hat{v}^1$ and $\check{v}^1$ are both replaced by several fictitious nodes with degree of one, is relatively trivial. This is because the ultimate network is formed if this condition is satisfied in Stage 2, and there is no need for Stage 3 and 4. For this trivial case, we can derive the overall change of agent v's utility in the following claim.

Claim 12. *If $\tilde{G}^2 = G^*$, $\Lambda^2 = \{v^1, \cdots, v^d\}$, and $\boldsymbol{w}_{\Lambda^2} = (w^*_{v^1}, \cdots, w^*_{v^d})$, then $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}) = U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) = U_v(G; \boldsymbol{w})$.*

Now we focus on the non-trivial case that after applying the Adjusting Technique in Stage 2, $x_{\hat{v}^2 u^i} < w^*_{v^i}$, for any $u^i \in \Gamma(\hat{v}^2)$, and $x_{\check{v}^2 u^i} > w^*_{v^i}$, for any $u^i \in \Gamma(\check{v}^2)$. Furthermore, each fictitious node $v' \in \Lambda^2 \setminus \{\check{v}^2, \hat{v}^2\}$, must be a leaf with fixed weight $w^*_{v^i}$, on $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2})$. Then in Stage 3, we continue to execute the Increasing Process (Algorithm 2) on $(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2})$ by setting threshold $\eta' = 0$, and $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$ is the resulting network.

Proof of Claim 3. Recall the Claim 11-(4), we know that $\hat{v}^2$ is a B-class with $\alpha_{\hat{v}^2}(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) \leq \alpha_v(G; \boldsymbol{w}) < 1$. For each other fictitious node $v' \in \Lambda^2 \setminus \{\hat{v}^2\}$, v' is a B-class by Claim 11-(2) and (3).
Applying Proposition A.5-(2), we have

$$U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}) \leq U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) + \eta_{\hat{v}^2}(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) \cdot (w(\Lambda^3) - w(\Lambda^2)).$$

Note that $w(\Lambda^3) - w(\Lambda^2) = \sum_{u^i \in \Gamma(\hat{v}^2)} (w^*_{v^i} - x_{\hat{v}^2 u^i}) \leq w_v$, and $\eta_{\hat{v}^2}(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) = \alpha_{\hat{v}^2}(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) \leq \alpha_v(G; \boldsymbol{w})$ by Claim 11-(4). we have

$$\begin{aligned} U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3}) &\leq U_v(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) + \eta_{\hat{v}^2}(\tilde{G}^2; \boldsymbol{w}_{\Lambda^2}) \cdot (w(\Lambda^3) - w(\Lambda^2)) \\ &\leq U_v(G; \boldsymbol{w}) + \alpha_v(G; \boldsymbol{w}) \cdot w_v \leq 2 \cdot U_v(G; \boldsymbol{w}). \end{aligned}$$

The desired result is proved. □

In addition, there are more properties of $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$ as follows.

Claim 13. *Let $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$ be the network after Stage 3, then*

(1) $\forall v' \in \Lambda^3 \setminus \{\check{v}^3\}$, $d_{v'} = 1$ and $w_{v'} = w^*_{v^i}$ on $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$, where u^i is the neighbor of v^i.

(2) $\forall u^i \in \Gamma(\check{v}^3)$, $x_{\check{v}^3 u^i} \geq w^*_{v^i}$ on $(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$.

At the end of Increasing Process, Λ^3 only contains some fictitious nodes with degree of one and $\check{v}^3$, since $\hat{v}^2$ has been split into several nodes $\{v^i\}$. As a result, Claim 13-(1) holds. In addition, the decreasing fictitious node is not impacted during the Increasing Process, which leads to the allocation from $\check{v}^3$ to its each neighbor remaining unchanged, so that Claim 13-(2) holds.

Claim 14. *Let $(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4})$ be the network after Stage 4, then*

(1) $\tilde{G}^4 = G^*$, $\Lambda^4 = \{v^1, \cdots, v^d\}$ and $\boldsymbol{w}_{\Lambda^4} = (w^*_{v^1}, \cdots, w^*_{v^d})$;

(2) $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}) = U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4})$.

At the end of decreasing process, all fictitious nodes with degree of one are spilt out. Therefore, the ultimate network is formed after the Stage 4, and Claim 14 holds.

Proof of Claim 4. By the the monotonicity of v's utility when applying Decreasing Process on $\check{v}^3$ (Proposition A.15-(2)), we have $U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4}) \leq U_v(\tilde{G}^3; \boldsymbol{w}_{\Lambda^3})$. Combined with $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}) = U_v(\tilde{G}^4; \boldsymbol{w}_{\Lambda^4})$ (Claim 14-(2)), the desired result has been proved. □

This completes the proof of Lemma 3.4.

D PROOF OF LEMMA 3.5

Claim 5 and Claim 6 are directly from Lemma 3.2 and the properties of the Adjusting Technique in Proposition A.13, respectively. In addition, we have some other properties of $(\tilde{G}^1;\boldsymbol{w}_{\Lambda^1})$ and $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$ as follows.

Claim 15. *Let $(\tilde{G}^1;\boldsymbol{w}_{\Lambda^1})$ be the network after Stage 1, then*
(1) $w(\Lambda^1) = w_v$.
(2) $\eta_{\hat{v}^1}(\tilde{G}^1;\boldsymbol{w}_{\Lambda^1}) \le \eta_{\check{v}^1}(\tilde{G}^1;\boldsymbol{w}_{\Lambda^1}) = 1/\alpha_v(G;\boldsymbol{w})$.

Claim 16. *Let $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$ be the network after Stage 2, then*
(1) $w(\Lambda^2) = w(\Lambda^1)$.
(2) $\forall v' \in \Lambda^2 \setminus \{\check{v}^2, \hat{v}^2\}$, v' is a C-class vertex with $\alpha_{v'} = \alpha_v(G;\boldsymbol{w})$, $d_{v'} = 1$, $w_{v'} = w^*_{v^i}$ on $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$.
(3) $\forall u^i \in \Gamma(\check{v}^2)$, $x_{\check{v}^2 u^i} \ge w^*_{v^i}$ and $\check{v}^2$ is a C-class vertex with $\alpha_{\check{v}^2} = \alpha_v(G;\boldsymbol{w})$ on $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$.
(4) $\forall u^i \in \Gamma(\hat{v}^2)$, $x_{\hat{v}^2 u^i} \le w^*_{v^i}$ and $\eta_{\hat{v}^2}(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2}) \le \eta_{\check{v}^2}(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2}) = \eta_v(G;\boldsymbol{w})$.

Similar to the analysis in Lemma 3.4, if the trivial stop condition of the Adjusting Technique, $\hat{v}^1$ and $\check{v}^1$ are both replaced by fictitious nodes with degree of one, is satisfied in Stage 2, then the ultimate network is formed. For this case, we can derive the overall change of agent v's utility in the following claim.

Claim 17. *If $\tilde{G}^2 = G^*$, $\Lambda^2 = \{v^1, \cdots, v^d\}$, and $\boldsymbol{w}_{\Lambda^2} = (w^*_{v^1}, \cdots, w^*_{v^d})$, then $U_v(G^*; w^*_{v^1}, \cdots, w^*_{v^d}) = U_v(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2}) = U_v(G;\boldsymbol{w})$.*

Now we focus on the non-trivial case that after applying the Adjusting Technique in Stage 2, for any $u^i \in \Gamma(\hat{v}^2)$, $x_{\hat{v}^2 u^i} < w^*_{v^i}$, and for any $u^i \in \Gamma(\check{v}^2)$ $x_{\check{v}^2 u^i} > w^*_{v^i}$. Furthermore, each fictitious node $v' \in \Lambda^2 \setminus \{\check{v}^2, \hat{v}^2\}$, must be a leaf with fixed weight $w^*_{v^i}$, on $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$.

By the monotonicity of v's utility when execute the Decreasing Process (Proposition A.15-(2)), Claim 7 holds. Moreover, there are other properties of $(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$ as follows.

Claim 18. *Let $(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$ be the network after Stage 3, then*
(1) $\forall v' \in \Lambda^3 \setminus \{\hat{v}^3\}$, $\alpha_{v'}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \le \alpha_v(G;\boldsymbol{w})$, $d_{v'} = 1$, $w_{v'} = w^*_{v^i}$.
(2) $\forall u^i \in \Gamma(\hat{v}^3)$, $x_{\hat{v}^3 u^i} < w^*_{v^i}$ on $(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$ and $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \le 1/\alpha_v(G;\boldsymbol{w})$.

Proof. According to the Decreasing Process, $\check{v}^2$ is totally replaced by $\{v^i\}$ and $\hat{v}^3 = \hat{v}^2$ at the end of Stage 3 (Proposition A.15-(1)). So Λ^3 only contains $\hat{v}^3$ and $\{v_i\}$, each v^i having weight of $w^*_{v^i}$. So Claim 18-(1) holds.

Note that before the Decreasing Process, by Proposition A.13, we have $\eta_{\hat{v}^2} \le \eta_{\check{v}^2} = \eta_v(G;\boldsymbol{w}) = 1/\alpha_v(G;\boldsymbol{w})$ and $\check{v}^2$ is in C class on the network $(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$.

For the case $\eta_{\hat{v}^2} \ne \eta_{\check{v}^2}$, if $\hat{v}^2$ is a B-class vertex (including $\alpha_{\hat{v}^2} = 1$), then $\hat{v}^2$ will always be in the B-class during the Decreasing Process (Proposition A.4-(1)), which means $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \le 1 \le 1/\alpha_v(G;\boldsymbol{w})$. If $\hat{v}^2$ is a C class vertex with $\alpha_{\hat{v}^2} > \alpha_{\check{v}^2}$, then the bottleneck pair containing $\hat{v}^2$ remains unchanged (Proposition A.15-(3)). Thus $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) = \eta_{\hat{v}^2}(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2}) \le 1/\alpha_v(G;\boldsymbol{w})$.

For the case $\eta_{\hat{v}^2} = \eta_{\check{v}^2}$, if $\hat{v}^2$ is a C class vertex, then $\alpha_{\check{v}^2} = \alpha_{\hat{v}^2} < 1$. By Proposition A.14, for any $\epsilon > 0$, $\alpha_{\check{v}^2}(\tilde{G}^2; w_{\check{v}^2} - \epsilon, \boldsymbol{w}_{\Lambda^2\setminus\{\check{v}^2\}}) < \alpha_{\hat{v}^2}(\tilde{G}^2; w_{\check{v}^2} - \epsilon, \boldsymbol{w}_{\Lambda^2\setminus\{\check{v}^2\}}) = \alpha_{\hat{v}^2}(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2})$. Then $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) = \eta_{\hat{v}^2}(\tilde{G}^2;\boldsymbol{w}_{\Lambda^2}) \le 1/\alpha_v(G;\boldsymbol{w})$. If $\hat{v}^2$ is a B class vertex, then $\hat{v}^2$ will always be in the B-class during the Decreasing Process, which means $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \le 1 \le 1/\alpha_v(G;\boldsymbol{w})$. □

As we described in Section 3.4, on the network $(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$, construct a neighbor set of $\hat{v}^3$: $N' = \{u^i \in \Gamma(\hat{v}^3) | \eta^*_{v^i} \le \eta_v(G;\boldsymbol{w}) = 1/\alpha_v(G;\boldsymbol{w})\}$, where v^i corresponds to u^i and $\eta^*_{v^i}$ is the η-ratio of v^i on $(G^*; w^*_{v^1}, \cdots, w^*_{v^d})$.

We first show N' is not empty. Note that $\eta_{\hat{v}^3} \leq \eta_v(G;\boldsymbol{w})$ and for each $u^i \in \Gamma(\hat{v}^3)$, $x_{\hat{v}^3 u^i} < w^*_{v^i}$, which means the weight of $\hat{v}^3$ needs to increase. By proposition A.3, we know the η-ratio of a vertex is non-increasing when its weight increases. The high-level intuition of the following lemma is, there is at least one v^i such that v^i has a low exchange ratio ($\eta^*_{v^i} \leq \eta_v(G;\boldsymbol{w})$).

Lemma D.1. $N' \neq \emptyset$

Proof. Note that $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \leq 1/\alpha_v(G;\boldsymbol{w})$ by Claim 18-(2). Let $(\tilde{G}^0,\boldsymbol{w}_{\Lambda^0},\hat{v}^0)$ be the output of Increase$(\tilde{G}^3,\boldsymbol{w}_{\Lambda^3},\hat{v}^3,0)$. Similar to the proof of Claim 14-(1), we have $\tilde{G}^0 = G^*, \Lambda^0 = \{v^1,\cdots,v^d\}, \boldsymbol{w}_{\Lambda^0} = (w^*_{v^1},\cdots,w^*_{v^d})$. Obviously, the neighbor of $\hat{v}^0$ is in N'. Thus it suffices to prove $\eta_{\hat{v}^0} \leq 1/\alpha_v(G;\boldsymbol{w})$.

Now we shall show that $\eta_{\hat{v}} \leq 1/\alpha_v(G;\boldsymbol{w})$ at each step of the Algorithm 2. At the beginning of the Algorithm 2, $\eta_{\hat{v}} = \eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \leq 1/\alpha_v(G;\boldsymbol{w})$. In the While loop (line 3-5), we have $\eta_{\hat{v}}$ is non-increasing by Lemma 3.2-(4) and (5). In the line 6, $w_{\hat{v}}$ increases and $\eta_{\hat{v}}$ is non-increasing by the monotonicity of $\eta_{\hat{v}}(x)$ (Proposition A.3). □

Then if $\eta^*_{v^i} \geq 1/\eta_v(G;\boldsymbol{w})$ for any $u^i \in N'$, we derive the desired upper bound of the incentive ratio by Lemma 3.6. Note that under this condition, we consider simply increasing $w_{\hat{v}}$ and execute RRB-split until $\hat{v}$ becomes several leaves with $w_{v^i} = w^*_{v^i}$ on the network $(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$, which leads to the ultimate network (we call it simple transform process, see Figure 4 (d) to (e)).

Proof of Lemma 3.6. Let $\Lambda^t = \{v^i | v^i \in \Lambda^3 \setminus \{\hat{v}^3\} \wedge \alpha_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) < \alpha_v(G;\boldsymbol{w})\}$. This proof includes two parts: (i) for each $v^i \in \Lambda^t$, $\eta^*_{v^i} = \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})$; (ii) for each $v^i \in \{v^1,\cdots,v^d\} \setminus \Lambda^t$, $\eta^*_{v^i} \leq \eta_v(G;\boldsymbol{w})$. Then

$$
\begin{aligned}
& U_v(G^*;w^*_{v^1},\cdots,w^*_{v^d}) - U_v(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \\
= & \sum_{v^i \in \Lambda^t} \eta^*_{v^i} w^*_{v^i} + \sum_{v^i \in \{v^1,\cdots,v^d\}\setminus\Lambda^t} \eta^*_{v^i} w^*_{v^i} \\
& - \left(\sum_{v^i \in \Lambda^t} \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) w^*_{v^i} + \left(U_v(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) - \sum_{v^i \in \Lambda^t} \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) w^*_{v^i} \right)\right) \\
= & \sum_{v^i \in \Lambda^t} \left(\eta^*_{v^i} - \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \right) \cdot w^*_{v^i} + \sum_{v^i \in \{v^1,\cdots,v^d\}\setminus\Lambda^t} \eta^*_{v^i} w^*_{v^i} \\
& - \left(U_v(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) - \sum_{v^i \in \Lambda^t} \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) w^*_{v^i} \right) \\
\leq & \sum_{v^i \in \Lambda^t} (\eta^*_{v^i} - \eta_{v^i}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3})) \cdot w^*_{v^i} + \sum_{v^i \in \{v^1,\cdots,v^d\}\setminus\Lambda^t} \eta^*_{v^i} w^*_{v^i} \\
= & \sum_{v^i \in \{v^1,\cdots,v^d\}\setminus\Lambda^t} \eta^*_{v^i} w^*_{v^i} \\
\leq & \sum_{v^i \in \{v^1,\cdots,v^d\}\setminus\Lambda^t} \eta_v(G;\boldsymbol{w}) w^*_{v^i} \leq U_v(G;\boldsymbol{w})
\end{aligned}
$$

Combining $U_v(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \leq U_v(G;\boldsymbol{w})$ by Claim 5, Claim 6, Claim 7, we have $U_v(G^*;\boldsymbol{w}^*_v) \leq 2 \cdot U_v(G;\boldsymbol{w})$.

These two parts of this proof are essentially using Proposition A.6.

Let $(G^*,\boldsymbol{w}^*_v,\hat{v}^0)$ be the output of Increase$(\tilde{G}^3,\boldsymbol{w}_{\Lambda^3},\hat{v}^3,0)$. Recall the proof of Lemma D.1, $\hat{v}^0 \in N'$. Combined with the condition $\forall u^i \in N', \eta^*_{v^i} \geq 1/\eta_v(G;\boldsymbol{w})$, we have $\alpha_{\hat{v}^0}(G^*;w^*_{v^1},\cdots,w^*_{v^d}) \geq \alpha_v(G;\boldsymbol{w})$.

Next we show that (i) and (ii) hold at each step of the Increasing Process. Firstly, since $\eta_{\hat{v}}$ is non-increasing at each step and $\alpha_{\hat{v}^0} \geq \alpha_v(G;\boldsymbol{w})$ at the end, it must be $\eta_{\hat{v}} \geq \alpha_v(G;\boldsymbol{w})$ at each step of the Increasing Process. Combining the fact that $\hat{v}^3$ is a C-class vertex and $\alpha_{\hat{v}^3} \geq \alpha_v(G;\boldsymbol{w})$, we have $\alpha_{\hat{v}} \geq \alpha_v(G;\boldsymbol{w})$ at each step of the Increasing Process.

At the beginning of the Increasing Process, (i) and (ii) hold obviously. In the While loop (line 3-5), note that $w^*_{v^i} > 0$, we have $\alpha_{v^i} = \alpha_{\hat{v}} \geq \alpha_v(G;\boldsymbol{w})$. Furthermore, the η-ratio of each other vertex remains unchanged by the properties of RRB-split (Lemma 3.2).

In line 6, $w_{\hat{v}}$ increases. Note that $\alpha_{\hat{v}}(w_{\hat{v}}) \geq \alpha_v(G;\boldsymbol{w})$ and $\alpha_{\hat{v}}(w_{\hat{v}} + z) \geq \alpha_v(G;\boldsymbol{w})$. For each $v' \in \Lambda$, if $\alpha_{v'}(w_{\hat{v}}) < \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$, then $(B_{v'}(w_{\hat{v}} + z), C_{v'}(w_{\hat{v}} + z)) = (B_{v'}(w_{\hat{v}}), C_{v'}(w_{\hat{v}}))$ and $\alpha_{v'}(w_{\hat{v}} + z) = \alpha_{v'}(w_{\hat{v}})$ by Proposition A.6-(1). Particularly, $\forall v' \in \Lambda^t, \alpha_{v'}(w_{\hat{v}}) < \alpha_v(G;\boldsymbol{w}) \leq \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$ which means $\eta_{v'}$ remains unchanged. For each $v' \in \Lambda$, if $\alpha_{v'}(w_{\hat{v}}) \geq \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$, then $\alpha_{v'}(w_{\hat{v}} + z) \geq \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$ by Proposition A.6-(2). Thus we have for each $v' \in \Lambda \backslash \Lambda^t$, either $\alpha_{v'}(w_{\hat{v}}) < \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$ then $\alpha_{v'}(w_{\hat{v}} + z) = \alpha_{v'}(w_{\hat{v}}) \geq \alpha_v(G;\boldsymbol{w})$, either $\alpha_{v'}(w_{\hat{v}}) \geq \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\}$, then $\alpha_{v'}(w_{\hat{v}} + z) \geq \min\{\alpha_{\hat{v}}(w_{\hat{v}}), \alpha_{\hat{v}}(w_{\hat{v}} + z)\} \geq \alpha_v(G;\boldsymbol{w})$. □

On the other hand, if the condition in Lemma 3.6 does not hold, we need to go to Stage 4 to play RRB-split. That is, if $\eta^*_{v^i} > 1/\alpha_v(G;\boldsymbol{w})$, then play RRB-split to split node v^i out from $\hat{v}^3$ with weight of $x_{\hat{v}^3 u^i}$, the allocation from $\hat{v}^3$ to u^i, one by one. We use H to denote the set containing these fictitious nodes. In the following proofs, we shall focus on the case $H \neq \emptyset$, as the case $H = \emptyset$ can also be tackled by the techniques developed in the case $H \neq \emptyset$.

Claim 8 is directly from Claim 18 and the properties of RRB-split (Lemma 3.2). Before analyzing other properties of the network obtained in Stage 4, we first prove a lemma related to the relationship between the weight assignment and η-ratio.

Lemma D.2. *Given a weight profile $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ satisfying (1) $\forall i \in \{1, \cdots, d\}, w'_{v^i} \leq w^*_{v^i}$; (2) if $w'_{v^i} < w^*_{v^i}$, then v^i is a C-class vertex with $\alpha^*_{v^i} < 1$ on $(G^*;\boldsymbol{w}^*_v)$. Then $\forall i \in \{1, \cdots, d\}$, if $w'_{v^i} < w^*_{v^i}$, it must be $\eta_{v^i}(G^*;\boldsymbol{w}^*_v) \leq \eta_{v^i}(G^*;\boldsymbol{w}'_v)$.*

Proof. Without loss of generality, assume that $w'_{v^1} < w^*_{v^1}, \cdots, w'_{v^h} < w^*_{v^h}$ in this proof, where $h \geq 1$. We shall prove this lemma by contradiction: if there is a $v^i \in \{v^1, \cdots, v^d\}$ with $w'_{v^i} < w^*_{v^i}$ and $\eta_{v^i}(G^*;\boldsymbol{w}^*_v) > \eta_{v^i}(G^*;\boldsymbol{w}'_v)$, then there exists $v^i \in \{v^1, \cdots, v^d\}$ where v^i is a B-class vertex on $(G^*;\boldsymbol{v}^*)$.

Without loss of generality, assume that $\eta_{v^1}(G^*;\boldsymbol{w}'_v) < \eta_{v^1}(G^*;\boldsymbol{w}^*_v)$. Let us transform $(G^*;\boldsymbol{w}'_v)$ into $(G^*; w^*_{v^1}, \cdots, w^*_{v^d})$ by increasing w_{v^i} iteratively, where $w'_{v^i} < w^*_{v^i}$. We shall prove after each weight increase, there is at least one v^i with $\eta_{v^i} < \eta_{v^i}(G^*;\boldsymbol{w}^*_v)$. When the weight of all nodes has finished increasing, there is also at least one v^i with $\eta_{v^i} < \eta_{v^i}(G^*;\boldsymbol{w}^*_v)$, which leads to a contradiction.

Increase w_{v^i} from w'_{v^i} to $w^*_{v^i}$ from $i = 1$ to h iteratively. Since η_{v^1} is non-increasing by Proposition A.3, $\eta_{v^1}(G^*; w^*_{v^1}, w'_{v^2}, \cdots, w'_{v^h}, w^*_{v^{h+1}}, \cdots, w^*_{v^d}) \leq \eta_{v^1}(G^*; w'_{v^1}, w'_{v^2}, \cdots, w'_{v^h}, w^*_{v^{h+1}}, \cdots, w^*_{v^d})$. For simplicity, we use the notation $(G^*; w^*_{v^1}, w'_{v^2})$ to denote $(G^*; w^*_{v^1}, w'_{v^2}, \cdots, w'_{v^h}, w^*_{v^{h+1}}, \cdots, w^*_{v^d})$. Other notations are similar. The marked fictitious node is the fictitious node $v^i \in H$ with $\eta_{v^i} < \eta_{v^i}(G^*;\boldsymbol{w}^*_v)$ currently. So v^1 is the first marked fictitious node.

Now let us increase w_{v^2} to $w^*_{v^2}$. For the case v^1 is a B-class vertex (including $\alpha_{v^1} = 1$) on $(G^*; w^*_{v^1}, w'_{v^2})$. If v^2 is a B-class vertex (including $\alpha_{v^2} = 1$) on $(G^*; w^*_{v^1}, w^*_{v^2})$, then let v^2 be the marked fictitious node. If v^2 is C-class vertex with $\alpha_{v^2} < 1$, we have the class that v^1 is in remains unchanged by Proposition A.4-(1), i.e., v^1 is still in B-class. We shall keep v^1 as the marked fictitious node. For the case v^1 is a C-class vertex with $\alpha_{v^1} < 1$ on $(G^*; w^*_{v^1}, w'_{v^2})$, we have $\alpha_{v^1} > \alpha_{v^1}(G^*;\boldsymbol{w}^*_v)$. If v^2 is a B-class vertex (including $\alpha_{v^2} = 1$) on $(G^*; w^*_{v^1}, w^*_{v^2})$, then let v^2 be the marked fictitious node.

If v^2 is C-class vertex with $\alpha_{v^2} < 1$, we have the α-ratio of v^1 is non-decreasing by Proposition A.4-(3) and $\alpha_{v^1} < 1$ by Proposition A.4-(4), i.e., $\alpha_{v^1}(G^*;\boldsymbol{w}_v^*) < \alpha_{v^1} < 1$ on $(G^*; w_{v^1}^*, w_{v^2}^*)$. We shall keep v^1 as the marked fictitious node. For the $v^i, 3 \leq i \leq h$ is similar, we can prove there is at least one marked fictitious node, which is contradiction. Thus the desired result has been proved. □

Based on Lemma D.2, we show some properties of $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$ as follows.

Claim 19. *Let $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$ be the network after Stage 4, then*

(1) $w(\Lambda^4) = w(\Lambda^3)$.
(2) $\forall u^i \in \Gamma(\hat{v}^4), x_{\hat{v}^4u^i} < w_{v^i}^*$ on $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$.
(3) For each $v^i \in H$, $\eta_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \leq 1/\alpha_v(G;\boldsymbol{w})$.
(4) $\forall v' \in \Lambda^4$, if $\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < 1/\alpha_v(G;\boldsymbol{w})$, then $w_{v'}^4 = 0$.
(5) If $H \neq \emptyset$, $\alpha_{\hat{v}^4}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$.

Proof. (1) and (2) can be proved directly by Claim 18 and the properties of RRB-split (Lemma 3.2). Note that each $v^i \in \Lambda^4 \setminus \{\hat{v}^4\}$ with $w_{v^i}^4 < w_{v^i}^*$ is split from $\hat{v}^3$ by RRB-split. Combined with the fact $\eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \leq 1/\alpha_v(G;\boldsymbol{w})$, we have (3).

For (4), note that $\forall v' \in \Lambda^3 \setminus \{\hat{v}^3\}, \alpha_{v'}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) \leq \alpha_v(G;\boldsymbol{w})$, we only need to consider $\hat{v}^4$ and H. If $w_{\hat{v}^3} = 0$, since each $v' \in H \cup \{\hat{v}^4\}$ is split from $\hat{v}^3$, then $w_{v'}^4 = 0$. If $w_{\hat{v}^3} > 0$, then $\eta_{\hat{v}^3} = 1/\alpha_v(G;\boldsymbol{w})$. So for each $v' \in H \cup \{\hat{v}^4\}$, if $w_{v'}^4 > 0$, we have $\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) = \eta_{\hat{v}^3}(\tilde{G}^3;\boldsymbol{w}_{\Lambda^3}) = 1/\alpha_v(G;\boldsymbol{w})$. This completes the proof for (4).

Finally, we prove (5) by and contradiction. Assume that $\alpha_{\hat{v}^4}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$. Note that $\hat{v}^4$ is also split from $\hat{v}^3$ by RRB-split, we have $\eta_{\hat{v}^4}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \leq 1/\alpha_v(G;\boldsymbol{w})$. Thus $\hat{v}^4$ is a B-class vertex with $\alpha_{\hat{v}^4} < 1$ on $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$.

Let $(\tilde{G}^0;\Lambda^0,\boldsymbol{w}_{\Lambda^0},\hat{v}^0)$ be the output of Increase$(\tilde{G}^4;\Lambda^4,\boldsymbol{w}_{\Lambda^4},\hat{v}^4,0)$. Similar to the proof of Claim 14-(2), we can prove that $\tilde{G}^0 = G^*, \Lambda^0 = \{v^1,\cdots,v^d\}, \forall v^i \in \{v^i,\cdots,v^d\} \setminus H, w_{v^i}^0 = w_{v^i}^*, \forall v^i \in H, w_{v^i}^0 = w_{v^i}^4 < w_{v^i}^*$. We shall show that for each $v^i \in H$, $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) \leq 1/\alpha_v(G;\boldsymbol{w})$ by two cases. And then we have $w_{v^i}^0 < w_{v^i}^*$ and $\eta_{v^i}(G^*;\boldsymbol{w}_v^*) > 1/\alpha_v(G;\boldsymbol{w}) \geq \eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0})$, which leads to a contradiction to Lemma D.2.

To prove for each $v^i \in H$, $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) \leq 1/\alpha_v(G;\boldsymbol{w})$, if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$, we have $\alpha_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) = \alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$ and (B_{v^i}, C_{v^i}) on $(G^*;\boldsymbol{w}_{\Lambda^0})$ is equal to (B_{v^i}, C_{v^i}) on $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$ by Proposition A.16-(3), thus $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) = \eta_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \leq 1/\alpha_v(G;\boldsymbol{w})$. If $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, same as $\hat{v}^4$, then v^i is a B-class vertex on $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$. It is easy to verify v^i is always in the B-class and α_{v^i} is non-increasing at each step of the Increasing Process. Thus $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) = \alpha_{v^i}(G^*;\boldsymbol{w}_{\Lambda^0}) < 1 < 1/\alpha_v(G;\boldsymbol{w})$. By contradiction, it must be $\alpha_{\hat{v}^4}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$. This completes the proof of Claim 19. □

Next, we consider the Increasing Process in Stage 5.

Claim 20. *Let $(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})$ be the network after Stage 5, then*

(1) $\forall u^i \in \Gamma(\hat{v}^5), x_{\hat{v}^5u^i} < w_{v^i}^*$ on $(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})$.
(2) $\forall v^i \in \Lambda^5 \setminus \{\hat{v}^5\}$, $w_{v^i} \leq w_{v^i}^*$.
(3) If $H \neq \emptyset$, then $\hat{v}^5$ is a B-class vertex and $\alpha_{\hat{v}^5} = \alpha_v(G;\boldsymbol{w})$ on $(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})$.
(4) If $H \neq \emptyset$, then for each $v^i \in H$, v^i is a C-class vertex and $\alpha_{\hat{v}^5} = \alpha_v(G;\boldsymbol{w})$ on $(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})$.

Proof. (1) and (2) are straightforward. For (3), we shall prove, in the case $\alpha_{\hat{v}^5} > \alpha_v(G;\boldsymbol{w})$ or $\alpha_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$, there must be at least one $v^i \in H$ with $w_{v^i}^5 < w_{v^i}^*$ and $\eta_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) \leq 1/\alpha_v(G;\boldsymbol{w}) < \eta_{v^i}(G^*;\boldsymbol{w}_v^*)$, which leads to a contradiction to Lemma D.2.

For the case $\alpha_{\hat{v}^5} > \alpha_v(G;\boldsymbol{w})$, note that $\forall v^i \in \Lambda^4 \setminus \{\hat{v}^4\}, d_{v'} = 1$. During the Increasing Process, each new fictitious node is split out from $\hat{v}$ in line 3-6, so the degree of these new fictitious nodes is 1. Moreover, we have $d_{\hat{v}^5} = 1$ by Proposition A.16-(1), which deduces $\tilde{G}^5 = G^*$. Note that before Increasing Process $\alpha_{\hat{v}^4} \geq \alpha_v(G;\boldsymbol{w})$ and after Increasing Process $\alpha_{\hat{v}^5} \geq \alpha_v(G;\boldsymbol{w})$. Similar to the proof of Lemma 3.6, we have for each $v^i \in H$, if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, then $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^5}) = \alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$; if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$, then $\alpha_{v^i}(G^*;\boldsymbol{w}_{\Lambda^5}) \geq \alpha_v(G;\boldsymbol{w})$, which leads to $\eta_{v^i}(G^*;\boldsymbol{w}_{\Lambda^5}) \leq 1/\alpha_v(G;\boldsymbol{w})$. This is a contradiction to Lemma D.2.

The idea of case $\alpha_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$ is essentially the same. Since $\eta_{\hat{v}^4} \leq 1/\alpha_v(G;\boldsymbol{w})$ and $\eta_{\hat{v}}$ is non-increasing during the Increasing Process, then $\hat{v}^5$ is a B-class vertex with $\alpha_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$, which also means $\eta_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$. We shall stress why $\eta_{\hat{v}^5}$ can be strictly less than $\alpha_v(G;\boldsymbol{w})$, since the threshold of function Increase is $\alpha_v(G;\boldsymbol{w})$. The only reason that $\eta_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$ could happen is after line 3-6, $\eta_{\hat{v}}$ decreased from greater than or equal to $\alpha_v(G;\boldsymbol{w})$ to less than $\alpha_v(G;\boldsymbol{w})$ (Actually this could not happen when $H \neq \emptyset$). Then the Algorithm ends in line 7-8. When RRB-split, for each $v^i \in \Lambda \setminus \{\hat{v}\}$ with $w_{v^i} < w^*_{v^i}$, η_{v^i} remains unchanged. Thus we can ignore the last RRB-split step and focus on $\eta_{\hat{v}} \geq \alpha_v(G;\boldsymbol{w})$. Similar to the proof of Lemma 3.6, we have for each $v^i \in H$, if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, then $\eta_{v^i}(G^5;\boldsymbol{w}_{\Lambda^5}) = \alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$; if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$, then $\alpha_{v^i}(G^5;\boldsymbol{w}_{\Lambda^5}) \geq \alpha_v(G;\boldsymbol{w})$, which means that $\eta_{v^i}(G^5;\boldsymbol{w}_{\Lambda^5}) \leq 1/\alpha_v(G;\boldsymbol{w})$.

Let $(\tilde{G}^0;\Lambda^0,\boldsymbol{w}_{\Lambda^0},\hat{v}^0)$ be the output of Increase$(\tilde{G}^5;\Lambda^5,\boldsymbol{w}_{\Lambda^5},\hat{v}^5,0)$.It is easy to see $\tilde{G}^0 = G^*$, $\forall v^i \in \{v^1,\cdots,v^d\} \setminus H, w^0_{v^i} = w^*_{v^i}$, $\forall v^i \in H, w^0_{v^i} < w^*_{v^i}$. Note that for each $v^i \in \Lambda^4$, if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, v^i must be a B-class vertex on $(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})$ by Claim 19-(4). Since $\hat{v}^5$ is a B-class vertex with $\alpha_{\hat{v}^5} < \alpha_v(G;\boldsymbol{w})$, we have $\forall v^i \in H$, if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \geq \alpha_v(G;\boldsymbol{w})$, α_{v^i} remains unchanged by Proposition A.16-(3); if $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, α_{v^i} is non-increasing, so v^i is still a B-class vertex on $(\tilde{G}^0;\boldsymbol{w}_{\Lambda^0})$. There is also a contradiction to Lemma D.2.

The key idea of proof of the fourth property is essentially same as the proof above. If there is $v^i \in H$ with $\alpha_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) < \alpha_v(G;\boldsymbol{w})$, it must be a B-class vertex, then it will be always in the B-class after Increase$(\tilde{G}^5;\Lambda^5,\boldsymbol{w}_{\Lambda^5},\hat{v}^5,0)$. If there is $v^i \in H$ with $\alpha_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) > \alpha_v(G;\boldsymbol{w})$, then α_{v^i} remains unchanged after Increase$(\tilde{G}^5;\Lambda^5,\boldsymbol{w}_{\Lambda^5},\hat{v}^5,0)$. In summary, there is at least one v^i in H and $\eta_{v^i} \leq 1/\alpha_v(G;\boldsymbol{w})$ after Increase$(\tilde{G}^5;\Lambda^5,\boldsymbol{w}_{\Lambda^5},\hat{v}^5,0)$. Similar to the proof above, this leads to a contradiction to Lemma D.2. □

Proof of Claim 9. The proof for Claim 9 is similar to the proof of Lemma 3.6. During the Increasing Process, $\eta_{\hat{v}}$ decrease from at most $1/\alpha_v(G;\boldsymbol{w})$ to $\alpha_v(G;\boldsymbol{w})$, thus for each $v^i \in \Lambda^4$ with $\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) < \alpha_v(G;\boldsymbol{w})$, w_{v^i} and U_{v^i} remains unchanged. Furthermore, for each $v^i \in \Lambda^4$ with $\eta_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) > 1/\alpha_v(G;\boldsymbol{w})$, we have $w_{v^i} = 0$ by Claim 19-(4), i.e., $U_{v^i} = 0$. Formally,

$$
\begin{aligned}
& U_v(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \\
= & \sum_{\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) + \sum_{\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})=1/\alpha_v(G;\boldsymbol{w})} U_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \\
+ & \sum_{\alpha_v(G;\boldsymbol{w})\leq\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})<1/\alpha_v(G;\boldsymbol{w})} U_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \\
= & \sum_{\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) + \Big(\sum_{\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})=1/\alpha_v(G;\boldsymbol{w})} w^4_{v'}\Big)/\alpha_v(G;\boldsymbol{w}) \\
= & \sum_{\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) + \Big(\sum_{\alpha_v(G;\boldsymbol{w})\leq\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})\leq1/\alpha_v(G;\boldsymbol{w})} w^4_{v'}\Big)/\alpha_v(G;\boldsymbol{w})
\end{aligned}
$$

and

$$
\begin{aligned}
& U_v(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) \\
= & \sum_{\alpha_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) + \sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})\le 1/\alpha_v(G;\boldsymbol{w})} U_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) \\
\le & \sum_{\alpha_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) + (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})\le 1/\alpha_v(G;\boldsymbol{w})} w^5_{v'})/\alpha_v(G;\boldsymbol{w})
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
& U_v(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) - U_v(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) \\
\le & \sum_{\alpha_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5}) + (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})\le 1/\alpha_v(G;\boldsymbol{w})} w^5_{v'})/\alpha_v(G;\boldsymbol{w}) \\
- & (\sum_{\alpha_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})<\alpha_v(G;\boldsymbol{w})} U_{v^i}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4}) + (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})\le 1/\alpha_v(G;\boldsymbol{w})} w^4_{v'})/\alpha_v(G;\boldsymbol{w})) \\
= & (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})\le 1/\alpha_v(G;\boldsymbol{w})} w^5_{v'})/\alpha_v(G;\boldsymbol{w}) - (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})\le 1/\alpha_v(G;\boldsymbol{w})} w^4_{v'})/\alpha_v(G;\boldsymbol{w}) \\
= & (\sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})\le 1/\alpha_v(G;\boldsymbol{w})} w^5_{v'} - \sum_{\alpha_v(G;\boldsymbol{w})\le\eta_{v'}(\tilde{G}^4;\boldsymbol{w}_{\Lambda^4})\le 1/\alpha_v(G;\boldsymbol{w})} w^4_{v'})/\alpha_v(G;\boldsymbol{w}) \\
= & (w(\Lambda^5) - w(\Lambda^4))/\alpha_v(G;\boldsymbol{w}) = \eta_v(G;\boldsymbol{w}) \cdot (w(\Lambda^5) - w(\Lambda^4)).
\end{aligned}
$$

□

The final part for the proof of Lemma A.11 requires complicated definitions, thus we present it in a separate subsection.

D.1 Proof of Lemma 3.7

Our proof relies on the analysis on a new increasing process in Stage 6. In this subsection, we first formally introduce this Updated Version of Increasing Process. Then we introduce a group of conditions, and any weight profile satisfying them is called a candidate weight profile. We show that the ultimate network for the Sybil attack can be obtained by applying any candidate weight profile as the input of the Updated Version of Increasing Process. Finally, we prove the upper bound of agent v's utility change during the process by induction. Combining with the fact that the weight profile of the fictitious nodes after Stage 5 is candidate weight profile, we complete the proof of Lemma 3.7.

Updated Version of Increasing Process (Increase2 for short).

On $(\tilde{G};\boldsymbol{w}_{\Lambda^5})$, $\hat{v}^5$ is an increasing fictitious node and the weights of all $v^i \in H$ are smaller than $w^*_{v^i}$, so we shall increase them to reach the ultimate network. There are two kinds of increasing processes. One is to increase $w_{\hat{v}^5}$ to $w_{\hat{v}^5} + z$, and thus its α-ratio non-increases in line 15. Once $\hat{v}^5$'s updated α-ratio is equal to the α-ratio of one $v^j \in H$, we can prove $\hat{v}^5$ and v^j are in a same bottleneck pair, in which $\hat{v}^5$ and v^j are in B-class and C-class respectively. At this time, by simultaneously increasing $w_{\hat{v}^5}$ to $w_{\hat{v}^5} + z$ and increasing w_{v^j} to $w_{v^j} + \alpha'_{v^j} \cdot z$ in line 11, the bottleneck decomposition and the α-ratios of $\hat{v}^5$ and v^j remain unchanged, until w_{v^j} reaches $w^*_{v^j}$. During these two kinds of increasing processes, if the current allocation on edge $(\hat{v}^5, u^i)$ is equal to the ultimate weight $w^*_{v^i}$, then Increase2 executes RRB-split to split v^i out and let its weight $w^*_{v^i}$ be fixed (line 3-6, line 9-12). The detailed process is summarized in Algorithm 4.

Correctness of Increase2.

ALGORITHM 4: Increase2($\tilde{G}'$, Λ', $\boldsymbol{w}_{\Lambda'}$, $\boldsymbol{w}'_v$, $\hat{v}'$, η')

Input: $\tilde{G}$, Λ', $\boldsymbol{w}_{\Lambda'}$, $\hat{v}' \in \Lambda'$, a threshold η' satisfying $\hat{v}'$ is a B-class vertex with $\alpha_{\hat{v}'} < 1$ on $(\tilde{G}';\boldsymbol{w}_{\Lambda'})$; $\boldsymbol{w}'_v$ is the ultimate weight profile of $\{v^1, \cdots, v^d\}$.

Output: $\tilde{G}$, Λ, $\boldsymbol{w}_\Lambda$, $\hat{v}$.

Let $\tilde{G} \leftarrow \tilde{G}'$, $\Lambda \leftarrow \Lambda'$, $\boldsymbol{w}_\Lambda \leftarrow \boldsymbol{w}_{\Lambda'}$, $\hat{v} \leftarrow \hat{v}'$.
do
  while $(d_{\hat{v}} > 1) \wedge (\exists u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} = w'_{v^i})$ **do**
    Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\hat{v}) \setminus \{u^i\}$.
    Let $(\tilde{G}, \hat{v}^1, \hat{v}^2, (w_{\hat{v}^1}, w_{\hat{v}^2})) \leftarrow$ RRB-split$(\tilde{G}, (\boldsymbol{w}_\Lambda), \hat{v}, N_1, N_2)$.
    Let $\Lambda \leftarrow \Lambda \setminus \{\hat{v}\} \cup \{\hat{v}^1, \hat{v}^2\}$, $v^i \leftarrow \hat{v}^1$, $\hat{v} \leftarrow \hat{v}^2$.
  end
  for $v^j \in \Lambda \setminus \{\hat{v}\} \wedge (w_{v^j} < w'_{v^j}) \wedge (1/\eta'_{v^j} = \alpha_{v^j}(\tilde{G};\boldsymbol{w}_\Lambda) = \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda))$ **do**
    do
      while $(d_{\hat{v}} > 1) \wedge (\exists u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} = w'_{v^i})$ **do**
        Let $N_1 \leftarrow \{u^i\}$, $N_2 \leftarrow \Gamma(\hat{v}) \setminus \{u^i\}$.
        Let $(\tilde{G}, \hat{v}^1, \hat{v}^2, (w_{\hat{v}^1}, w_{\hat{v}^2})) \leftarrow$ RRB-split$(\tilde{G}, (\boldsymbol{w}_\Lambda), \hat{v}, N_1, N_2)$.
        Let $\Lambda \leftarrow \Lambda \setminus \{\hat{v}\} \cup \{\hat{v}^1, \hat{v}^2\}$, $v^i \leftarrow \hat{v}^1$, $\hat{v} \leftarrow \hat{v}^2$.
      end
      if $\alpha_{v^j}(\tilde{G};\boldsymbol{w}_\Lambda) \neq \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda)$ **then**
        Break
      end
      Find the largest $z \geq 0$ such that:
        (1) $\alpha'_{v^j} \cdot z \leq w'_{v^j} - w_{v^j}$.
        (2) $\forall u^j \in \Gamma(\hat{v}), x_{\hat{v}u^j} \leq w'_{v^j}$ on $(\tilde{G}; w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{v^j,\hat{v}\}})$.
        (3) $\mathcal{B}(\tilde{G}; w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{v^j,\hat{v}\}}) = \mathcal{B}(\tilde{G}, \boldsymbol{w}_\Lambda)$.
      Let $w_{v^j} \leftarrow w_{v^j} + \alpha'_{v^j} \cdot z$, $w_{\hat{v}} \leftarrow w_{\hat{v}} + z$.
    while $z > 0$;
  end
  Find the largest $z \geq 0$ such that:
    (1) $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} \leq w'_{v^i}$ on $(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$.
    (2) $\eta_{\hat{v}}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) \geq \eta'$.
    (3) $\forall v^i \in \Lambda$, if $w_{v^i} < w'_{v^i}$, then $\alpha_{v^i}(G^*;\boldsymbol{w}'_v) \leq \alpha_{v^i}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})$.
  Let $w_{\hat{v}} \leftarrow w_{\hat{v}} + z$.
while $z > 0$;
return $\tilde{G} = (\tilde{V}, \tilde{E})$, Λ, $\boldsymbol{w}_\Lambda$, $\hat{v}$.

As this new increasing process is less intuitive, we need to prove that the ultimate network can indeed be obtained by applying increase2 on the network $(\tilde{G}^5;\boldsymbol{w}_{\Lambda^5})$ in Stage 6. Precisely, Increase2 has two inputs, the network and the ultimate weight profile, while in the desired ultimate network, each fictitious node must only connect to one distinguished neighbor. We extend such a correctness by showing that applying Increase2 for any network with a weight profile satisfying the following condition leads to the ultimate network.

Definition D.3 (Candidate Weight Profile). Given a Sybil network $(\tilde{G};\boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose that (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G};\boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$.

For a weight profile $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ of $\{v^1, \cdots, v^d\}$, denote $\eta_{v^i}(G^*; \boldsymbol{w}'_v)$ by η'_{v^i}. Then $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ is called a candidate weight profile of $(\tilde{G}; \boldsymbol{w}_\Lambda)$ if and only if it satisfies the following conditions:

- $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} \le w'_{v^i}$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$.
- $\forall v^i \in \Lambda \setminus \{\hat{v}\}$, $w_{v^i} \le w'_{v^i}$.
- $\forall v^i \in \Lambda \setminus \{\hat{v}\}$, if $w_{v^i} < w'_{v^i}$, then v^i is a C-class vertex on both $(\tilde{G}; \boldsymbol{w}_\Lambda)$ and $(G^*; \boldsymbol{w}'_v)$. Furthermore, $\alpha_{v^i}(\tilde{G}; \boldsymbol{w}_\Lambda) \ge \alpha_{v^i}(G^*; \boldsymbol{w}'_v)$.

Lemma D.4. *Given a Sybil network $(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$ with a fictitious node $\hat{v}' \in \Lambda'$. Suppose (1) $\hat{v}'$ is a B-class vertex with $\alpha_{\hat{v}'} < 1$ on $(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$; (2) $\forall v' \in \Lambda' \setminus \{\hat{v}'\}, d_{v'} = 1$. Let $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ be a candidate weight profile of $(\tilde{G}'; \boldsymbol{w}_{\Lambda'})$.*

Let $(\tilde{G}, \Lambda, \boldsymbol{w}_\Lambda, \hat{v})$ be the output of Increase2$(\tilde{G}', \Lambda', \boldsymbol{w}_{\Lambda'}, \hat{v}', \boldsymbol{w}'_v, 0)$, then $(\tilde{G}; \boldsymbol{w}_\Lambda) = (G^*; \boldsymbol{w}'_v)$ with $\Lambda = \{v^1, \cdots, v^d\}$.

Proof. This proof includes three parts. Firstly we prove that at each step of Increase2, for each vertex $u \in \tilde{V}$, the class that u is in remains unchanged and α_u is non-increasing.

At the beginning of Increase2, $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$. In the While loop (line 3-6), we have $\eta_{\hat{v}^1} \le \eta_{\hat{v}} < 1$ and $\eta_{\hat{v}^2} \le \eta_{\hat{v}} < 1$ by Lemma 3.2-(4) and (5), i.e., the new fictitious node v^i is a B-class vertex, $\hat{v}$ is always in the B-class and $\alpha_{\hat{v}}$ is non-increasing. For each other vertex $u \in \tilde{V} \setminus \{\hat{v}\}$, η_u remains unchanged, i.e., α_u and the class that u is in remains unchanged. In the For loop (line 7-16), if there exists one fictitious node $v^j \in \Lambda \setminus \{\hat{v}\}$ satisfying $w_{v^j} < w'_{v^j}$ and $\alpha'_{v^j} = \alpha_{v^j}(\tilde{G}; \boldsymbol{w}_\Lambda) = \alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$, we shall increase w_{v^j} and $w_{\hat{v}}$ simultaneously and keep each vertex's α-ratio the same. The While loop (line 8-16) is working for a specific fictitious node v^j chosen in line 7. Similar above, each other vertex $u \in \tilde{V} \setminus \{\hat{v}\}$, η_u remains unchanged, i.e., α_u and the class that u is in remain unchanged in the While loop (line 9-12). Note that v^j is in C-class and $\hat{v}$ is in B-class. By the condition $\mathcal{B}(\tilde{G}; w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{v^j, \hat{v}\}}) = \mathcal{B}(\tilde{G}, \boldsymbol{w}_\Lambda)$, we know that the bottleneck decomposition remains unchanged.

For the sake of convenience, we shall use $(w_{v^j}, w_{\hat{v}})$ and $(w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z)$ to denote $(\tilde{G}, \boldsymbol{w}_\Lambda)$ and $(\tilde{G}; w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{v^j, \hat{v}\}})$. Suppose that $\mathcal{B}(w_{v^j}, w_{\hat{v}}) = \{(B_1, C_1), \cdots, (B_k, C_k)\}$ and $v^j, \hat{v} \in B_j \cup C_j$. Since only the weights of v^j and $\hat{v}$ change, let us focus on the pair (B_j, C_j). Note that $\alpha_{\hat{v}} < 1$, we have $\alpha'_j(w_{v^j} + \alpha'_{v^j} \cdot z, w_{\hat{v}} + z) = \frac{w(C_j) + \alpha_j \cdot z}{w(B_j) + z} = \alpha_j$. In addition, for each vertex $u \in \tilde{V}$, α_u and the class that u is in remain unchanged, which has been proved before. Note that $0 < \eta_{\hat{v}} < 1$, Increase2 will not end on line 18. In line 19, since $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$, for each $u \in \tilde{V}$, α_u is non-increasing by Proposition A.5-(4), the class that u is in is not impacted by Proposition A.5-(1). The first claim has been proved.

Secondly, we show that at each step of Increase2, for each $v^i \in \Lambda \setminus \{\hat{v}\}$, if $w_{v^i} < w'_{v^i}$, then $\alpha_{\hat{v}} \ge \alpha'_{v^i}$.

At the beginning of Increase2, this result holds by the definition of Candidate Weight Profile, in particular, the third condition. In the While loop (line 3-6), for each vertex $u \in \tilde{V} \setminus \{\hat{v}\}$, η_u remains unchanged, i.e., α_u and the class that u is in remains unchanged. So this property is maintained. In the While loop (line 7-16), we have proved above, for each iteration, the α-ratio of all vertices remains unchanged. Now we show that if $\exists v^j \in \Lambda \setminus \{\hat{v}\} \wedge (w_{v^j} < w'_{v^j}) \wedge (\alpha'_{v^j} = \alpha_{v^j}(\tilde{G}; \boldsymbol{w}_\Lambda) = \alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda))$ in line 7, then $w_{v^j} = w'_{v^j}$ after the While loop (line 8-16). Based on this observation, we can conclude after the For loop (7-16), for each $v^i \in \Lambda \setminus \{\hat{v}\}$, if $w_{v^i} < w'_{v^i}$, then $\alpha_{\hat{v}} > \alpha'_{v^i}$.

The subsequent analysis for the While loop (line 8-16) is based on a specific fictitious node v^j chosen in line 7. In the While loop (9-12), similar as above, the α-ratio of each vertex remains unchanged. So the second claim is still maintained. We shall show that if $w_{v^j} < w'_{v^j}$, then $\alpha_{v^j}(\tilde{G};\boldsymbol{w}_\Lambda) = \alpha_{\hat{v}}(\tilde{G};\boldsymbol{w}_\Lambda)$ at line 13. The key idea is the contradiction to Lemma D.2 similar as the proof of Claim 19-(6). Since $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$, if $\alpha_{\hat{v}}$ changed in While loop (9-12), then $\alpha_{\hat{v}}$ must decrease, which means $\alpha_{\hat{v}} < \alpha_{v^j}$. Let $(\tilde{G}^0;\Lambda^0,\boldsymbol{w}_{\Lambda^0},\hat{v}^0)$ be the output of Increase$(\tilde{G};\Lambda,\boldsymbol{w}_\Lambda,\hat{v},0)$ (where $\boldsymbol{w}'_v$ is the ultimate weight assignment while not $\boldsymbol{w}^*_v = (w^*_{v^1},\cdots,w^*_{v^d})$), we can prove that $\tilde{G}^0 = G^*, \Lambda^0 = \{v^1,\cdots,v^d\}, \forall v^i \in \{v^i,\cdots,v^d\}, w^0_{v^i} \le w'_{v^i}$ and $w^0_{v^j} < w'_{v^j}$. Since $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < \alpha_{v^j}$, then $\alpha_{v^j}(G^*;\boldsymbol{w}_{\Lambda^0}) = \alpha_{v^j}(\tilde{G};\boldsymbol{w}_\Lambda) = \alpha'_{v^j}$ by Proposition A.16-(3). Note that $\alpha_{v^j}(G^*;\boldsymbol{w}_{\Lambda^0}) < 1$, then $\alpha_{v^j}(G^*;w^0_{v^j}+\epsilon,\boldsymbol{w}_{\Lambda^0\setminus\{v^j\}}) > \alpha_{v^j}(G^*;\boldsymbol{w}_{\Lambda^0}) = \alpha'_{v^j}$ and $w^0_{v^j}+\epsilon < w'_{v^j}$ for any sufficiently small $\epsilon > 0$. This leads to a contradiction to Lemma D.2. Based on this observation, when $w_{v^j} < w'_{v^j}$, the While loop (8-16) must break in line 16, which means $z = 0$.

Next, we show it is also impossible that the While loop (8-16) break in line 16 with $z = 0$. If this happens, one of the three conditions in line 15 cannot hold for any $z > 0$. If the first condition leads to $z = 0$, we have $w'_{v^j} = w_{v^j}$, thus the second claim still holds. If the second condition leads to $z = 0$, we have $d_{\hat{v}} = 1$ and $w_{\hat{v}} = w'_{v^i}$, where u^i is the neighbor of v^i. Then $\tilde{G} = G^*$ and $\forall v^i \in \{v^1,\cdots,v^d\}, w_{v^i} \le w'_{v^i}$. Note that $\alpha_{v^j} = \alpha'_{v^j}$, if $w_{v^j} < w'_{v^j}$, then this leads to a contradiction to Lemma D.2 similar as above, so w_{v^j} must equal to w'_{v^j}. If only the third condition $\mathcal{B}(\tilde{G};w_{v^j}+\alpha'_{v^j}\cdot z, w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{v^j,\hat{v}\}}) = \mathcal{B}(\tilde{G},\boldsymbol{w}_\Lambda)$, leads to $z = 0$, we have $\mathcal{B}(w_{v^j},w_{\hat{v}}) \neq \mathcal{B}(w_{v^j}+\alpha'_{v^j}\cdot\epsilon, w_{\hat{v}}+\epsilon)$ for any sufficiently small $\epsilon > 0$. Similar as the proof of Proposition A.14, we have $\alpha_{\hat{v}}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) < \alpha_{v^j}(\tilde{G};w_{\hat{v}}+\epsilon,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) = \alpha_{v^j}(\tilde{G};\boldsymbol{w}_\Lambda) = \alpha'_{v^j}$. If $w_{v^j} < w'_{v^j}$, then let $w_{\hat{v}}$ increase a sufficiently small $\epsilon > 0$ and execute the function Increase (Algorithm 3), we get $(G^*,\boldsymbol{w}_{\Lambda^0})$, where $\forall v^i \in \{v^i,\cdots,v^d\}, w^0_{v^i} \le w'_{v^i}$ and $w^0_{v^j} < w'_{v^j}$. Combining with the fact that v^j is a C-class vertex with $\alpha'_{v^j} = \alpha_{v^j} < 1$, this leads to a contradiction to Lemma D.2 similar as above. This completes the proof of the second claim.

Lastly, we claim the output $\tilde{G} = G^*$ and for each new fictitious node split from $\hat{v}$ in line 3-6, i.e., $v^i \in \{v^1,\cdots,v^d\}\setminus(\Lambda'\setminus\{\hat{v}'\}), w_{v^i} = w'_{v^i}$.

To prove the third claim, we only focus on $\hat{v}$. Note that for each new fictitious split from $\hat{v}$ in line 3-6 must satisfy $d_{v^i} = 1$ and $w_{v^i} = w'_{v^i}$, so it is sufficient to prove $d_{\hat{v}} = 1$ and $w_{\hat{v}} = w'_{v^i}$ after the Algorithm ends, where u^i is the neighbor of $\hat{v}$. Note that $\eta_{\hat{v}} > 0$ all the time, so the Algorithm must end with $z = 0$. Look at the three conditions in line 11. For the first condition, after line 3-6, either $d_{\hat{v}} = 1$ or $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} < w'_{v^i}$ on $(\tilde{G};\boldsymbol{w}_\Lambda)$. If $d_{\hat{v}} = 1$ and $w_{\hat{v}} = w'_{v^i}$ then the third claim has been proved. For other case, $\forall u^i \in \Gamma(\hat{v}), x_{\hat{v}u^i} < w'_{v^i}$, so the condition 1 does not restrict z must be equal to zero. The second condition can not restrict z since for any $z > 0$, $\eta_{\hat{v}}$ must be greater than zero. For the third condition, recall that in the proof of second claim, we have after the For loop (7-16), for each $v^i \in \Lambda\setminus\{\hat{v}\}$, if $w_{v^i} < w'_{v^i}$, then $\alpha_{\hat{v}} > \alpha'_{v^i}$. Thus $w_{\hat{v}}$ must increase $z > 0$ until $\alpha_{\hat{v}}$ equal to the largest α'_{v^i}, where v^i satisfying $w_{v^i} < w'_{v^i}$. So the third can not restrict z. In summary, there is only one case that $d_{\hat{v}} = 1$ and $w_{\hat{v}} = w'_{v^i}$, where u^i is the neighbor of $\hat{v}$. This completes the proof of the third claim.

Actually, when Increase2 finds the largest $z > 0$, if the first condition satisfies, then Increase2 will go back to the While loop (line 3-6) to split these fictitious nodes out. If the first condition is not satisfied but the third condition is satisfied, then Increase2 will skip the While loop (line 3-6) and execute the For loop (line 7-16) to increase w_{v^j} to w'_{v^j}. This v^j must be the one whose α'_{v^j} is the largest α-ratio among those fictitious nodes $v^i \in \Lambda\setminus\{\hat{v}\}$ with $w_{v^i} < w'_{v^i}$. Meanwhile, $\alpha_{\hat{v}} = \alpha_{v^j} = \alpha'_{v^j}$.

Combining the above three claims, we have $\tilde{G} = G^*$, $\Lambda = \{v^1, \cdots, v^d\}$. Finally we prove $\boldsymbol{w}_\Lambda = \boldsymbol{w}'_v$ by contradiction. Note that $\forall v^i \in \{v^1, \cdots, v^d\}, w_{v^i} \le w'_{v^i}$. If there exists one node v^i with $w_{v^i} < w'_{v^i}$, then $\alpha_{v^i} \ge \alpha'_{v^i}$ by the second claim. v^i is a C-class vertex with $\alpha'_{v^i} \le \alpha_{v^i} < 1$, this leads to a contradiction to Lemma D.2 similar as above. Thus, $\boldsymbol{w}_\Lambda = \boldsymbol{w}'_v$. □

Changes of agent v's utility during Increase2.

We first present a property of Increase2.

Proposition D.5. *Given a Sybil network $(\tilde{G}; \boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$. Suppose $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ is a candidate weight profile of $(\tilde{G}; \boldsymbol{w}_\Lambda)$.*

Let $(G^*, \{v^1, \cdots, v^d\}, \boldsymbol{w}'_v, \hat{v}^t)$ be the output of Increase2$(\tilde{G}, \Lambda, \boldsymbol{w}_\Lambda, \hat{v}, \boldsymbol{w}'_v, 0)$. For each vertex $v^i \in \Lambda \setminus \{\hat{v}\}$, if $\alpha_{v^i}(G^*; \boldsymbol{w}'_v) > \alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$ or $\alpha_{v^i}(G^*; \boldsymbol{w}'_v) < \alpha_{\hat{v}^t}(G^*; \boldsymbol{w}'_v)$, then $w_{v^i} = w'_{v^i}$ and $\eta_{v^i}(\tilde{G}; \boldsymbol{w}_\Lambda, \boldsymbol{w}_v) = \eta_{v^i}(G^*; \boldsymbol{w}'_v)$.

Proof. Note that during Process2, $\alpha_{\hat{v}}$ monotonously decreases from $\alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$ to $\alpha_{\hat{v}^t}(G^*; \boldsymbol{w}'_v)$. So we have for each vertex $v^i \in \Lambda \setminus \{\hat{v}\}$, if $\alpha_{v^i}(G^*; \boldsymbol{w}'_v) > \alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)$, then $\alpha_{v^i} > \alpha_{\hat{v}}$ at each step of Increase2. So the w_{v^i} can not increase in the For loop (line 7-16). It is easy to see in the While loop (line 3-6) and For loop (line 7-10), η_{v^i} remains unchanged. Furthermore, η_{v^i} remains unchanged by Proposition A.5-(1) and (2). The case that $\alpha_{v^i}(G^*; \boldsymbol{w}'_v) < \alpha_{\hat{v}^t}(G^*; \boldsymbol{w}'_v)$ is similar. □

Based on Proposition D.5, we define a set of fictitious node, called Impacted Fictitious Node, consists of those whose α-ratio on G^* is between the one of $\hat{v}^t$ on G^* and the one of $\hat{v}$ on $\tilde{G}$.

Definition D.6 (Impacted Fictitious Node). Given a Sybil network $(\tilde{G}; \boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$. Let $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ be a candidate weight profile of $(\tilde{G}; \boldsymbol{w}_\Lambda)$. Let $(G^*, \boldsymbol{w}'_v, \hat{v}^*)$ be the output of Increase2$(\tilde{G}, \boldsymbol{w}_\Lambda, \hat{v}, \boldsymbol{w}'_v, 0)$.

The impacted fictitious nodes set is defined as : $Q := \{v^i | \alpha_{\hat{v}^*}(G^*; \boldsymbol{w}'_v) \le \alpha_{v^i}(G^*; \boldsymbol{w}'_v) \le \alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)\}$.

Next, we upper bound the change of agent v's utility by the total amount of weight increased during Increase2 and the weight of those impacted fictitious nodes who is in C-class.

Lemma D.7 (Main lemma). *Given a Sybil network $(\tilde{G}; \boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$. Let $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ be a candidate weight profile of $(\tilde{G}; \boldsymbol{w}_\Lambda)$. Let Q be the impacted fictitious nodes set and Q' be the C-class vertices in Q, such that any $v^i \in Q'$ has $\eta_{v^i}(G^*; \boldsymbol{w}'_v) \ge 1/\alpha_{\hat{v}}(\tilde{G}; \boldsymbol{w}_\Lambda)\}$. So*

$$U_v(G^*; \boldsymbol{w}'_v) - U_v(\tilde{G}; \boldsymbol{w}_\Lambda) \le \sum_{i=1}^{d} w'_{v^i} - \sum_{v' \in \Lambda} w_{v'} + \sum_{v^i \in Q'} w_{v^i}.$$

The key idea for proving Lemma D.7 is the utility bound of Lemma A.10, Lemma A.11, Lemma A.12 and the mathematical induction. Specifically, Lemma A.10 and Lemma A.11 show the changes of utility is upper bounded by the changes of weights. Lemma A.12 shows that the utility of B-class vertices is upper bounded by the weight of C-class vertices.

Before proving Lemma D.7 in detail, we need to introduce two necessary definitions.

Definition D.8 (Number of Q's α-ratio). Given a Sybil network $(\tilde{G}; \boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G}; \boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$. Let $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ be a candidate weight profile of $(\tilde{G}; \boldsymbol{w}_\Lambda)$ and Q be the impacted fictitious nodes set. The number of different α-ratios of $\{v^i\}$ on $(G^*; \boldsymbol{w}'_v)$ is named as the number of Q's α-ratio.

The following definition is a technique to divide the impacted fictitious nodes set Q.

Definition D.9 (BC Interval). Given a Sybil network $(\tilde{G};\boldsymbol{w}_\Lambda)$ with a fictitious node $\hat{v} \in \Lambda$. Suppose (1) $\hat{v}$ is a B-class vertex with $\alpha_{\hat{v}} < 1$ on $(\tilde{G};\boldsymbol{w}_\Lambda)$; (2) $\forall v' \in \Lambda \setminus \{\hat{v}\}, d_{v'} = 1$. Let $\boldsymbol{w}'_v = (w'_{v^1}, \cdots, w'_{v^d})$ be a candidate weight profile of $(\tilde{G};\boldsymbol{w}_\Lambda)$ and Q be the impacted fictitious nodes set.

Define $M := |Q|$. Note that the α-ratio of each impacted fictitious node is less than one, so each impacted fictitious node either only belongs to B class or only C-class. Without loss of generality, re-index the impacted fictitious nodes as $Q = \{v^1, \cdots, v^M\}$, satisfying (1) $\alpha'_{v^1} \leq \cdots \leq \alpha'_{v^M}$; (2) if v^i is a B-class vertex and v^j is a C-class vertex with $\alpha'_{v^i} = \alpha'_{v^j}$ on $(G^*;\boldsymbol{w}'_v)$, then let $i < j$.

The BC intervals are defined as follows: Start with $q_0 = 0$ and $i = 1$. Find the largest p_i satisfying $\forall j \in \{q_{i-1}+1, q_{i-1}+2, \cdots, p_i\}$, v^j is a B-class vertex. If $p_i = M$, then set $q_i = p_i = M$. Otherwise, find the largest q_i satisfying $\forall j \in \{p_i+1, p_i+2, \cdots, q_i\}$, v^j is a C-class vertex. Then $[q_{i-1}+1, q_i]$ is defined as the ith BC interval. Repeat the construction of BD interval if $q_i \neq M$. If $q_i = M$, then set $m = i$, and thus m is the number of BC intervals.

It is worth to note that $\alpha_{\hat{v}^*}(G^*;\boldsymbol{w}'_v)$ is the smallest among all α-ratios of the impacted fictitious nodes and $\hat{v}^*$ is a B-class vertex. So BC Intervals are well-defined.

Proof of Lemma D.7. Note that $(G^*;\boldsymbol{w}'_v)$ is the output of Increase2$(\tilde{G};\boldsymbol{w}_\Lambda)$. Let m_1 is the number of BC Interval of Q and m_2 is the number of Q's α-ratio. We prove this lemma by induction on this ordered pair (m_1, m_2). Precisely, we first check the result for (m_1, m_2) with $m_1 = 1, \forall m_2 > 0$ in **Basis**. Then in **Induction step**, we show that if this result holds for $(m_1 - 1, m_2)$ $(\forall m_2 > 0)$ and (m_1, k_2) $(\forall k_2 < m_2)$ for any $(\tilde{G};\boldsymbol{w}_\Lambda)$ and a candidate weight profile $\boldsymbol{w}'_v$ of $(\tilde{G};\boldsymbol{w}_\Lambda)$, then it also holds for (m_1, m_2) for any $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$ and a candidate weight profile $\boldsymbol{w}'_v$ of $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$.

Basis: We shall prove for any $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$ and candidate weight profile $\boldsymbol{w}'_v$ of $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$, if $m_1 = 1$ and $m_2 > 0$, i.e., there is only one BC Interval, this utility bound holds $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q'} w^s_{v^i}$. Recall that $\forall j \in \{1, \cdots, p_1\}, v^j$ is a B-class vertex and $\forall j \in \{p_1+1, \cdots, q_1\}$, v^j is a C-class vertex on $(G^*;\boldsymbol{w}'_v)$.

We shall analyze the change of utility in each step of the Increase2$(\tilde{G}^s, \Lambda^s, \boldsymbol{w}_{\Lambda^s}, \boldsymbol{w}'_v, \hat{v}^s, 0)$. ('s' stands for 'start'.) Note that $\sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'}$ is the change of the weights of all fictitious nodes and $\sum_{v^i \in Q'} w^s_{v^i}$ is the sum of the weights of C-class impacted fictitious nodes. Thus we focus on the relationship between ΔU_v and Δw_v. For the trivial case $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'}$, i.e., $\Delta U_v \leq \Delta w_v$, the desired result has been proved. So there must be at least one step in Increase2, such that $\Delta U_v > \Delta w_v$. Next we shall analyze this case and use $\sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q'} w^s_{v^i}$ to bound the difference of utility.

We use ΔU_v and Δw_v to denote the change of agent v's utility and the change of all fictitious nodes' total weight respectively. For the sake of convenience, we shall always use these simple notations to represent the changes but specify the exact step(s) when the changes happen if necessary.

We first claim that in each iteration of the While loop (line 3-6) and the DoWhile loop (line 8-16), $\Delta U_v = \Delta w_v$. Precisely, in the While loop (line 3-6), it is easy to see $\Delta U_v = 0$ and $\Delta w_v = 0$. In the While loop (line 8-16), similarly we have $\Delta U_v = \Delta w_v = 0$ in the line 9-12. For the line 15, recall that the bottleneck decomposition and the α-ratio of each pair remain unchanged in the proof of Lemma D.4, we thus only focus on v^j and $\hat{v}$. Note that at each small iteration of the While loop (line 8-16), $\Delta w_v = \Delta w_{v^j} + \Delta w_{\hat{v}} = \alpha'_{v^j} \cdot z + z$, $\Delta U_v = \Delta U_{v^j} + \Delta U_{\hat{v}} = \Delta w_{v^j}/\alpha'_{v^j} + \Delta w_{\hat{v}} \cdot \alpha'_{v^j} = \alpha'_{v^j} \cdot z / \alpha'_{v^j} + z \cdot \alpha'_{v^j} = z + z \cdot \alpha'_{v^j}$, so we have $\Delta U_v = \Delta w_v$.

Based on this observation we know that there must be at least one iteration of the While loop (line 2-18), precisely line 17, such that $\Delta U = U_v(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\hat{v}\}}) - U_v(\tilde{G};\boldsymbol{w}_\Lambda) > z = \Delta w_v$, where z is the one find in line 17. Let $(\tilde{G}^t; w_{\hat{v}^t} + z, \boldsymbol{w}_{\Lambda^t \setminus \{\hat{v}^t\}})$ be network after the last iteration that $\Delta U_v > \Delta w_v$.

('t' stands for 'temporary'.) Then we have $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t; w_{\hat{v}^t} + z, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}}) \leq \sum_{i=1}^{d} w'_{v^i} - (\sum_{v'\in\Lambda^t} w^t_{v'} + z)$. To prove the desired result, we only need to prove $U_v(\tilde{G}^t; w_{\hat{v}^t} + z, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}}) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq (\sum_{v'\in\Lambda^t} w^t_{v'} + z) - \sum_{v'\in\Lambda^s} w^s_{v'} + \sum_{v^i\in Q'} w^s_{v^i}$.

Now we focus on this last step that $w_{\hat{v}^t}$ is increased to $w_{\hat{v}^t} + z$. Note that when increasing $w_{\hat{v}^t}$ to $w_{\hat{v}^t} + z$, the bottleneck decomposition changes with respect to a single parameter. Similar to Appendix A.1, partition $[w_{\hat{v}^t}, w_{\hat{v}^t} + z]$ into a number of disjoint subintervals $\{\langle a_i, b_i\rangle\}_i$. When $x \in \langle a_i, b_i\rangle$ the bottleneck decomposition is represented as $\mathcal{B}(\tilde{G}^t; x, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}}) = \mathcal{B}^i = \{(B^i_1, C^i_1), \cdots, (B^i_{k^i}, C^i_{k^i})\}$. For the sake of convenience, simplify the notations $(\tilde{G}^t; x, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}})$ as (x), which only depends on a single parameter x. Meanwhile, the condition that $U_v(w_{\hat{v}^t} + z) - U_v(w_{\hat{v}^t}) > z$ means there must be at least one subinterval such that $U_v(b_i) - U_v(a_i) > b_i - a_i$. Let ℓ be the largest index satisfying $U_v(b_\ell) - U_v(a_\ell) > b_\ell - a_\ell$. Thus $U_v(w_{\hat{v}^t} + z) - U_v(b_\ell) \leq w_{\hat{v}^t} + z - b_\ell$. So we only need to prove $U_v(b_\ell) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq (\sum_{v'\in\Lambda^t} w^t_{v'} - w_{\hat{v}^t} + b_\ell) - \sum_{v'\in\Lambda^s} w^s_{v'} + \sum_{v^i\in Q'} w^s_{v^i}$.

Suppose $\hat{v}^t \in B^\ell_j \cup C^\ell_j$. We first claim there are some impacted fictitious nodes in C^ℓ_j. If not, it must be $U_v(b_\ell) - U_v(a_\ell) \leq \eta_{\hat{v}^t}(a_\ell)\cdot(b_\ell - a_\ell) < b_\ell - a_\ell$, which leads to a contradiction to the definition of ℓ. Then we claim for each $v^i \in \{v^1, \cdots, v^{p_1}\} \cap \Lambda^t$, $v^i \in B^\ell_j$. To prove the claim, assume, for the sake of convenience, $v^1 \notin B^\ell_j$, we know that $\hat{v}^t$ is the impacted fictitious node with the smallest α-ratio among the B-class impacted fictitious nodes, so $\alpha_{v^1} > \alpha_{\hat{v}^t}$. At the same time, there must be at least one C-class impacted fictitious node, for example v^{q_1}, such as $\alpha_{v^{q_1}} < \alpha_{v^1}$. In each of following iterations of the While loop (line 2-18), the α-ratio of each vertex in non-increasing and α_{v^1} is remains unchanged, so at the end of Increase2, $\alpha_{v^{q_1}} < \alpha_{v^1}$, which leads to a contradiction to the definition of BC Interval.

Then we consider $U_v(\cdot) = U_{\Lambda^t\cap B}(\cdot) + U_{\Lambda^t\cap C}(\cdot)$, where $\Lambda^t \cap B$ $(\Lambda^t \cap C)$ denotes all B class (C class respectively) nodes in Λ^t. As $m_1 = 1$, $\Lambda^t \cap B = \Lambda^t \cap B^\ell_j$. By Lemma A.12, we have $U_{\Lambda^t\cap B^\ell_j}(b_\ell) < w(\Lambda^t \cap C^\ell_j)$ directly. Now we show that $w(\Lambda^t \cap C^\ell_j) \leq \sum_{v^i\in Q'} w^s_{v^i}$. Clearly, for each $v^i \in \Lambda^t \cap C^\ell_j$, v^i is a C-class impacted fictitious node, i.e., $v^i \in Q'$. Recall that we are considering the network $(\tilde{G}^t; b_\ell, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}})$, which is an intermediate network when increasing $w_{\hat{v}^t}$ to $w_{\hat{v}^t} + z$ on the network $(\tilde{G}^t; w_{\hat{v}^t}, \boldsymbol{w}_{\Lambda^t\setminus\{\hat{v}^t\}})$. Then for each $v^i \in \Lambda^t \cap C^\ell_j$, $w_{v^i} = w^s_{v^i}$, because if the weight of v^i has been increased, v^i will not be in the same bottleneck pair as $\hat{v}^t$. As a result, $U_{\Lambda^t\cap B}(b_\ell) - U_{\Lambda^s\cap B}(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq U_{\Lambda^t\cap B}(b_\ell) = U_{\Lambda^t\cap B^\ell_j}(b_\ell) < w(\Lambda^t \cap C^\ell_j) \leq \sum_{v^i\in Q'} w^s_{v^i}$.

Finally, we consider $U_{\Lambda^t\cap C}(b_\ell)$. We shall show that at each step of line 15, $U_{\Lambda^t\cap C}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}}) - U_{\Lambda^t\cap C}(\tilde{G};\boldsymbol{w}_\Lambda) \leq z$. Since $\hat{v}$ is a B-class vertex we only focus on the utility of C-class vertices, this result can be deduced by Lemma A.11. Furthermore, in each iteration of the While loop (line 8-16), we have $\Delta U_{v^j} = z \leq z + \alpha'_{v^j}\cdot z = \Delta w_v$. So we have $U_{\Lambda^t\cap C}(b_\ell) - U_{\Lambda^s\cap C}(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq \Delta w_v = (\sum_{v'\in\Lambda^t} w^t_{v'} - w_{\hat{v}^t} + b_\ell) - \sum_{v'\in\Lambda^s} w^s_{v'}$.

Induction step: We now proceed to prove the inductive step by showing that this result holds for $(m_1 - 1, m_2)$ $(\forall m_2 > 0)$ and (m_1, k_2) $(\forall k_2 < m_2)$ for any $(\tilde{G};\boldsymbol{w}_\Lambda)$ and a candidate weight profile $\boldsymbol{w}'_v$ of $(\tilde{G};\boldsymbol{w}_\Lambda)$, then it also holds for (m_1, m_2) for any $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$ and a candidate weight profile $\boldsymbol{w}'_v$ of $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$.

Intuitively, we divide the process of executing Increase2 into two stages through setting a positive threshold. Thus, we can first deal with the increasing process which never influence those fictitious nodes belonging to the first BC interval, by leveraging the result of the inducted pair $(m_1 - 1, m_2)$ (one less intervals) or $(m_1, k_2 < m_2)$ (at least one less α-ratio). Then we deal with the rest process which only influence those belonging to the first BC interval, with similar analysis as **Basis**.

Note that $m_1 > 1$. By the definition of BC Interval, for any $j \in \{1, \cdots, p_1\}$ and $j \in \{q_1+1, \cdots, p_2\}$, v^j is a B-class vertex; and for any $j \in \{p_1 + 1, \cdots, q_1\}$ and $j \in \{p_2 + 1, \cdots, q_2\}$, v^j is a C-class vertex. Now we divide Increase2 into two stages. Note that $\alpha'_{v^1} \le \cdots \le \alpha'_{v^{q_1}} < \alpha'_{v^{q_1+1}} \le \cdots \le \alpha'_{v^{q_2}}$. Let $(\tilde{G}^t, \Lambda^t, \boldsymbol{w}_{\Lambda^t}, \hat{v}^t)$ be the output of Increase2$(\tilde{G}^s, \Lambda^s, \boldsymbol{w}_{\Lambda^s}, \boldsymbol{w}'_v, \hat{v}^s, \alpha'_{v^{q_1+1}})$, i.e., $(\tilde{G}^t, \Lambda^t, \boldsymbol{w}_{\Lambda^t})$ is the intermediate state from $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$ to $(G^*;\boldsymbol{w}'_v)$ when $\alpha_{\hat{v}} = \alpha'_{v^{q_1+1}}$, the smallest α-ratio in the second BC Interval. It is possible that $\alpha^t_{\hat{v}^t} < \alpha'_{v^{q_1+1}}$ because of the jump of $\alpha_{\hat{v}}$ in the While loop (line 3-6) when $w_{\hat{v}^2} = 0$, but the analysis of this special case is essentially the same as the case $\alpha^t_{\hat{v}^t} = \alpha'_{v^{q_1+1}}$, so we only present the more general case $\alpha^t_{\hat{v}^t} = \alpha'_{v^{q_1+1}}$. The main task in induction step is to obtain

$$U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \le \sum_{i=1}^{d} w'_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q'} w^s_{v^i} \tag{7}$$

Consider $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) = U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) + U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^s}) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$. We deal with the case $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) \le \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^t} w^t_{v'}$ (Case 1, denoted by $\Delta U_v \le \Delta w_v$) and $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) > \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^t} w^t_{v'}$ (Case 2, denoted by $\Delta U_v > \Delta w_v$) separately.

Case 1 ($\Delta U_v \le \Delta w_v$): Under the condition of Case 1, that is $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) \le \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^t} w^t_{v'}$, it suffices to prove

$$U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \le \sum_{v' \in \Lambda^t} w^t_{v'} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q'} w^s_{v^i}.$$

Here on the network $(\tilde{G}^t; w_{\Lambda^t})$, we split $\hat{v}^t$ iteratively through RRB-split until each fictitious node has only one neighbor. The obtained network is denoted by $(G^*;\boldsymbol{w}^a_v)$, where $\boldsymbol{w}^a_v = (w^a_{v^1}, \cdots, w^a_{v^d})$, and $w^a_{v^i} := w^t_{v^i}$ if $v^i \in \Lambda^t \setminus \{\hat{v}^t\}$ and $w^a_{v^i} := x_{\hat{v}^t u^i}$ if $u^i \in \Gamma(\hat{v}^t)$. Obviously, $\sum_{v' \in \Lambda^t} w^t_{v'} = \sum_{i=1}^d w^a_{v^i}$ and $U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) = U_v(G^*;\boldsymbol{w}^a_v)$. Next we shall explain that $\boldsymbol{w}^a_v$ is a candidate weight profile of $(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$ by verifying the three conditions in the definition of Candidate Weight Profile. Note that $\forall u^i \in \Gamma(\hat{v}^s)$, if $u^i \in \Gamma(\hat{v}^t)$, then $w^a_{v^i} = x_{\hat{v}^t u^i} \ge x_{\hat{v}^s u^i}$ since $x_{\hat{v} u^i}$ is non-decreasing. If $u^i \notin \Gamma(\hat{v}^t)$, then v^i has been split from $\hat{v}$ in line 3-6. It must be $w^a_{v^i} = w'_{v^i} \ge x_{\hat{v}^s u^i}$, and then the first condition holds. Clearly, $\forall v^i \in \Lambda^s \setminus \{\hat{v}^s\}$, w_{v^i} is non-decreasing, implying $w^s_{v^i} \le w^a_{v^i}$, i.e., the second condition holds. During Increase2, for each $v^i \in \Lambda \setminus \{\hat{v}\}$, the class that v^i is in remains unchanged and α_{v^i} is non-increasing by the proof of Lemma D.4. The third condition holds.

W.l.o.g., assume for each fictitious node v^i, $w^a_{v^i} > 0$. This is because for those v^i with $w^a_{v^i} = 0$, $U_{v^i}(\tilde{G}^a;\boldsymbol{w}^a_v) = 0$, then $U_{v^i}(\tilde{G}^a;\boldsymbol{w}^a_v) - U_{v^i}(\tilde{G}^s;\boldsymbol{w}^s_\Lambda) \le 0$.

Let Q^a be the set of impacted fictitious nodes of $(G^*;\boldsymbol{w}^a_v)$. Note that $Q^a \subseteq Q$. Since for each $v^i \in Q^a$, i.e., $\alpha'_{v^{q_1+1}} \le \alpha_{v^i}(\tilde{G}^a;\boldsymbol{w}^a_v) \le \alpha_{\hat{v}^s}$, it must be $\alpha'_{v^1} \le \alpha_{v^i}(G^*;\boldsymbol{w}'_v) \le \alpha_{\hat{v}^s}$. So $v^i \in Q$. We shall prove the following two claims, which can help us to obtain the result by induction: (1) $\alpha^a_{v^i} = \alpha'_{v^i}$, for each $v^i \in \{v^{q_1+1}, \cdots, v^M\}$; (2) for each $v^i \in \{v^1, \cdots, v^{q_1}\}$, if $v^i \in Q^a$, i.e., $\alpha'_{v^{q_1+1}} \le \alpha_{v^i}(\tilde{G}^a;\boldsymbol{w}^a_v)$, then $\alpha'_{v^{q_1+1}} = \alpha_{v^i}(\tilde{G}^a;\boldsymbol{w}^a_v)$.

The proof of these two claims are essentially the same. Note that $\forall u^i \in \Gamma(\hat{v}^t), \alpha^a_{v^i} = \alpha^t_{\hat{v}^t} = \alpha'_{v^{q_1+1}}$, since v^i is split from $\hat{v}^t$ and $w^a_{v^i} > 0$. When $w_{\hat{v}^t}$ increases, $\alpha_{\hat{v}^t}$ will decrease, which means $\forall u^i \in \Gamma(\hat{v}^t), \alpha'_{v^i} < \alpha'_{v^{q_1+1}}$, i.e., $v^i \in \{v^1, \cdots, v^{q_1}\}$. Thus we can focus on the nodes v^i, where $u^i \notin \Gamma(\hat{v}^t)$. If there is $v^i \in \{v^{q_1+1}, \cdots, v^M\}$, $\alpha^t_{v^i} > \alpha'_{v^i} \ge \alpha'_{v^{q_1+1}}$, then we continue to execute Increase2$(\tilde{G}^t; \Lambda^t, \boldsymbol{w}_{\Lambda^t}, \boldsymbol{w}'_v, \hat{v}^t, 0)$, $\alpha^t_{v^i}$ remains unchanged, implying $\alpha_{v^i}(G^*;\boldsymbol{w}'_v) > \alpha'_{v^i}$. This leads to a contradiction. The proof of the second claim is similar.

There are two cases for induction. W.l.o.g., suppose Q^a contains one B-class node v^1 and one C-class node v^{q_1} (the analysis is the same if there are more than one C-class fictitious node and B-class fictitious node in Q^a). If $\alpha'_{v^{q_1+1}} = \cdots = \alpha'_{v^{p_2}}$, then $(v^1, v^{q_1+1}, \cdots, v^{p_2}, v^{q_1}, v^{p_2+1}, \cdots, v^{q_2})$ will be the first BC Interval and the other BC Intervals are same, indicating there are $m_1 - 1$ BC Intervals.

This case corresponds to the inducted pair $(m_1 - 1, m_2)$. If $\alpha'_{v^{q_1+1}} < \alpha'_{v^{p_2}}$, without loss of generality, assume $\alpha'_{v^{q_1+1}} < \alpha'_{v^{q_1+2}}$, then $(v^1, v^{q_1+1}, v^{q_1})$ will be the first BC Interval and $(v^{q_1+2}, \cdots, v^{q_2})$ will be the second BC Interval and the other BC Intervals are the same, implying there are m_1 BC Intervals. Furthermore, we have $\alpha^a_{v^1} = \alpha^a_{v^{q_1+1}} = \alpha_{v^{q_1}}$, then the number of Q's α-ratio decreases at least one. This case corresponds to the inducted pair (m_1, k_2) with $k_2 < m_2$. By applying the inductive hypothesis, we have

$$U_v(G^*;\boldsymbol{w}^a_v) - U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) \leq \sum_{i=1}^{d} w^a_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q^a} w^s_{v^i} \leq \sum_{i=1}^{d} w^a_{v^i} - \sum_{v' \in \Lambda^s} w^s_{v'} + \sum_{v^i \in Q'} w^s_{v^i}.$$

Note that $U_v(G^*;\boldsymbol{w}^a_v) = U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t})$ and $\sum_{i=1}^d w^a_{v^i} = \sum_{v' \in \Lambda^t} w^t_{v'}$, the desired result is obtained.

Case 2 ($\Delta U_v > \Delta w_v$): If the condition for Case 2 holds, we introduce another increasing process: first RRB-split the increasing fictitious node $\hat{v}^s$ into two nodes, $\hat{v}^0$ and $\hat{v}^b$, by connecting $\hat{v}^0$ to all the neighbors of those fictitious nodes in the first BC Interval while $\hat{v}^b$ to others (from $(\tilde{G}^s; w_{\Lambda^s})$ to $(\tilde{G}^0; w_{\Lambda^0})$); then execute Increase2 w.r.t. $\hat{v}^0$ (from $(\tilde{G}^0; w_{\Lambda^0})$ to $(\tilde{G}^b; w_{\Lambda^b})$); finally execute Increase2 w.r.t. $\hat{v}^b$ (from $(\tilde{G}^b; w_{\Lambda^0})$ to $(\tilde{G}^*; w'_v)$). Then we upper bound the change of agent v's utility in the first two stages, while the one for the last stage is proved by the induction condition directly.

Let $N^0 = \{u^i | v^i \in \{v^1, \cdots, v^{p_1}\}\}$. Note that all the B-class impacted fictitious nodes are split from $\hat{v}^s$, showing $N^0 \subsetneq \Gamma(\hat{v}^s)$. Let $(\tilde{G}^0, \hat{v}^0, \hat{v}^b, w_{\hat{v}^0}, w_{\hat{v}^b})$ be the output of RRB-split$(\tilde{G}^s, (\boldsymbol{w}_{\Lambda^s}), \hat{v}^s, N^0, \Gamma(\hat{v}^s) \setminus N^0)$. Denote $\Lambda^0 := \Lambda^s \setminus \{\hat{v}^s\} \cup \{\hat{v}^0, \hat{v}^b\}$. Here we emphasize that there are two specified increasing fictitious nodes on $\tilde{G}^0$. Thus we first execute Increase2$(\tilde{G}^0, \Lambda^0, \boldsymbol{w}_{\Lambda^0}, \boldsymbol{w}'_v, \hat{v}^0, 0)$ to increase the weight of $\hat{v}^0$ on purpose, and then $(\tilde{G}^b; \Lambda^b, \boldsymbol{w}_{\Lambda^b}, \hat{v}^1)$ is the output. Next we continue to execute Increase2$(\tilde{G}^b, \Lambda^b, \boldsymbol{w}_{\Lambda^b}, \boldsymbol{w}'_v, \hat{v}^b, 0)$ to purposefully increase the weight of $\hat{v}^b$ to achieve $(G^*;\boldsymbol{w}'_v)$.

To obtain the desired result in this lemma, we need following two inequalities:

$$U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) \leq \sum_{i=1}^{d} w'_{v^i} - \sum_{v' \in \Lambda^b} w^b_{v'} + \sum_{v^i \in Q' \setminus \{v^{p_1+1}, \cdots, v^{q_1}\}} w^s_{v^i}; \tag{8}$$

and

$$U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) - U_v(\tilde{G}^0;\boldsymbol{w}_{\Lambda^0}) \leq \sum_{v' \in \Lambda^b} w^b_{v'} - \sum_{v' \in \Lambda^0} w^0_{v'} + \sum_{v^i \in \{v^{p_1+1}, \cdots, v^{q_1}\}} w^s_{v^i}, \tag{9}$$

in which inequality (Equation (8)) can be obtained by induction hypothesis and the main idea of the proof for (Equation (9)) is similar to the one in **Basis**, while the analysis is much more complicated. Note that $\sum_{v' \in \Lambda^0} w^0_{v'} = \sum_{v' \in \Lambda^s} w^s_{v'}$ and $U_v(\tilde{G}^0;\boldsymbol{w}_{\Lambda^0}) = U_v(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s})$. Combining these two inequalities, (Line 23) can be proved.

We first prove inequality (Equation (8)). Recall $\hat{v}^0$ is a increasing fictitious node, split out from $\hat{v}^s$. After Increase2$(\tilde{G}^0, \Lambda^0, \boldsymbol{w}_{\Lambda^0}, \boldsymbol{w}'_v, \hat{v}^0, 0)$, it must be $\forall v^i \in \{v^1, \cdots, v^{q_1}\}$, $w^b_{v^i} = w'_{v^i}$ and $\alpha^b_{v^i} = \alpha'_{v^i}$. If not, during Increase2$(\tilde{G}^b, \Lambda^b, \boldsymbol{w}_{\Lambda^b}, \boldsymbol{w}'_v, \hat{v}^b, 0)$, we have $\alpha_{\hat{v}} \geq \alpha'_{v^{q_1+1}}$. Thus the vertices with α-ratio less than $\alpha'_{v^{q_1+1}}$ will not be impacted, which leads to a contradiction to the ultimate state $(G^*;\boldsymbol{w}'_v)$. Clearly, the impacted fictitious nodes set Q^b of $(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b})$ and $(G^*;\boldsymbol{w}'_v)$ is $Q^b = \{v^{q_1+1}, \cdots, v^M\}$. There are $m_1 - 1$ BC Intervals, which corresponds to the inducted pair $(m_1 - 1, m_2)$. By the induction hypothesis, the first inequality is obtained.

Now consider Equation (9). We divide Increase2$(\tilde{G}^0, \Lambda^0, \boldsymbol{w}_{\Lambda^0}, \boldsymbol{w}'_v, \hat{v}^0, 0)$ into two stages. Firstly, we execute Increase2$(\tilde{G}^0, \Lambda^0, \boldsymbol{w}_{\Lambda^0}, \boldsymbol{w}'_v, \hat{v}^0, \alpha'_{v^{q_1+1}})$ by setting threshold $\eta' = \alpha'_{v^{q_1+1}}$ to output an interim $(\tilde{G}^r, \Lambda^r, \boldsymbol{w}_{\Lambda^r}, \hat{v}^r)$. Secondly, we execute Increase2$(\tilde{G}^r, \Lambda^r, \boldsymbol{w}_{\Lambda^r}, \boldsymbol{w}'_v, \hat{v}^r, 0)$ by setting threshold $\eta' = 0$ to return $(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b})$.

Let $w_{\hat{v}^t}$ increase a sufficiently small $\epsilon_1 > 0$ and $w_{\hat{v}^r}$ increase a sufficiently small $\epsilon_2 > 0$ such that $\alpha_{\hat{v}^t}(\tilde{G}^t; w_{\hat{v}^t} + \epsilon_1, \boldsymbol{w}_{\Lambda^t \setminus \{\hat{v}^t\}}) = \alpha_{\hat{v}^r}(\tilde{G}^r; w_{\hat{v}^r} + \epsilon_2, \boldsymbol{w}_{\Lambda^r \setminus \{\hat{v}^r\}})$. For the sake of convenience, denote $(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t})$ by $(w_{\hat{v}^t})$ and $(\tilde{G}^t; w_{\hat{v}^t}+\epsilon_1, \boldsymbol{w}_{\Lambda^t \setminus \{\hat{v}^t\}})$ by $(w_{\hat{v}^t}+\epsilon_1)$. Similarly, denote $(\tilde{G}^r;\boldsymbol{w}_{\Lambda^r})$ by $(w_{\hat{v}^r})$ and $(\tilde{G}^r; w_{\hat{v}^r} + \epsilon_2, \boldsymbol{w}_{\Lambda^r \setminus \{\hat{v}^r\}})$ by $(w_{\hat{v}^r} + \epsilon_2)$. Note that each u^i is connected to exactly one fictitious node on any Sybil network, so denote the fictitious node connected to u^i by $v(u^i)$. Let $L^t = \{v(u^i) | u^i \in \tilde{G}^t \wedge u^i \in \{u^1, \cdots, u^{q_1}\}\} \subseteq \Lambda^t$ and $L^r = \{v(u^i) | u^i \in \tilde{G}^r \wedge u^i \in \{u^1, \cdots, u^{q_1}\}\} \subseteq \Lambda^r$.

Since $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t}) > \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^t} w^t_{v'}$ in Case 2, we shall prove a similar property $U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) - U_v(w_{\hat{v}^r} + \epsilon_2) > \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2)$ under this condition.

Claim 21. *We have the following properties.*

(1) $\alpha_{\hat{v}^t}(w_{\hat{v}^t} + \epsilon_1) < \alpha_{\hat{v}^t}(w_{\hat{v}^t}) = \alpha'_{v^{q_1+1}}$.
(2) $\alpha_{\hat{v}^r}(w_{\hat{v}^r} + \epsilon_2) < \alpha_{\hat{v}^r}(w_{\hat{v}^r}) = \alpha'_{v^{q_1+1}}$.
(3) $U_v(G^*;\boldsymbol{w}'_v) - U_v(\tilde{G}^t; w_{\hat{v}^t} + \epsilon_1, \boldsymbol{w}_{\Lambda^t \setminus \{\hat{v}^t\}}) > \sum_{i=1}^d w'_{v^i} - (\sum_{v' \in \Lambda^t} w^t_{v'} + \epsilon_1)$.
(4) $\forall u \in \tilde{G}^t \setminus \Lambda^t$, $\alpha_u(w_{\hat{v}^t} + \epsilon_1) > \alpha_{\hat{v}^t}(w_{\hat{v}^t} + \epsilon_1)$ *if and only if* $\alpha_u(G^*;\boldsymbol{w}'_v) \geq \alpha'_{v^{q_1+1}}$.
(5) $\forall u \in \tilde{G}^t$, $\alpha_u(w_{\hat{v}^t} + \epsilon_1) < \alpha_{\hat{v}^t}(w_{\hat{v}^t} + \epsilon_1)$ *if and only if* $\alpha_u(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) < \alpha'_{v^{q_1+1}}$.
(6) $\forall u \in \tilde{G}^r \setminus \Lambda^r$, $\alpha_u(w_{\hat{v}^r} + \epsilon_2) > \alpha_{\hat{v}^r}(w_{\hat{v}^r} + \epsilon_2)$ *if and only if* $\alpha_u(G^*;\boldsymbol{w}'_v) \geq \alpha'_{v^{q_1+1}}$.
(7) $\forall u \in \tilde{G}^r$, $\alpha_u(w_{\hat{v}^r} + \epsilon_2) < \alpha_{\hat{v}^r}(w_{\hat{v}^r} + \epsilon_2)$ *if and only if* $\alpha_u(\tilde{G}^s;\boldsymbol{w}_{\Lambda^s}) < \alpha'_{v^{q_1+1}}$.
(8) $\forall u^i \in \{u^1, \cdots, u^d\}$, $\alpha_{v(u^i)}(w_{\hat{v}^t} + \epsilon_1) > \alpha_{\hat{v}^t}(w_{\hat{v}^t} + \epsilon_1)$ *if and only if* $i \geq q_1 + 1$.
(9) $\forall u^i \in \{u^1, \cdots, u^d\}$, $\alpha_{v(u^i)}(w_{\hat{v}^r} + \epsilon_2) > \alpha_{\hat{v}^r}(w_{\hat{v}^r} + \epsilon_2)$ *if and only if* $i \geq q_1 + 1$.

Let us focus on $(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t})$ first. Note that for each $v^i \in \{v^{q_1+1}, \cdots, v^M\}$, it must be $w_{v^i}(w_{\hat{v}^t} + \epsilon_1) = w'_{v^i}$ and $\alpha_{v^i}(w_{\hat{v}^t} + \epsilon_1) = \alpha'_{v^i}$. So the utilities of these fictitious nodes are offset. Then we have

$$U_v(G^*;\boldsymbol{w}'_v) - U_v(w_{\hat{v}^t} + \epsilon_1) = \sum_{i=1}^{q_1} w'_{v^i} \cdot \eta'_{v^i} - \sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1) \cdot \eta_{v'}(w_{\hat{v}^t} + \epsilon_1)$$

and

$$\sum_{i=1}^{d} w'_{v^i} - \sum_{v' \in \Lambda^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1) = \sum_{i=1}^{q_1} w'_{v^i} - \sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1).$$

Thus $\sum_{i=1}^{q_1} w'_{v^i} \cdot \eta'_{v^i} - \sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1) \cdot \eta_{v'}(w_{\hat{v}^t} + \epsilon_1) > \sum_{i=1}^{q_1} w'_{v^i} - \sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1)$.

Then consider $(\tilde{G}^r;\boldsymbol{w}_{\Lambda^r})$. Note that $\alpha_{\hat{v}^r}(w_{\hat{v}^r} + \epsilon_2) < \alpha'_{v^{q_1+1}}$ and $\alpha_{\hat{v}}$ is non-increasing during the Increase2. Combining Claim 21-(9), expand the result of Proposition A.5, we have $\forall u^i \in \{u^{q_1+1}, \cdots, u^d\}$, $\alpha_{v(u^i)}(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) = \alpha_{v(u^i)}(w_{\hat{v}^r} + \epsilon_2)$ and $w_{v(u^i)}(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) = w_{v(u^i)}(w_{\hat{v}^r} + \epsilon_2)$. So

$$U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) - U_v(w_{\hat{v}^r} + \epsilon_2) = \sum_{i=1}^{q_1} w'_{v^i} \cdot \eta'_{v^i} - \sum_{v' \in L^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2) \cdot \eta_{v'}(w_{\hat{v}^r} + \epsilon_2)$$

and

$$\sum_{i=1}^{d} w'_{v^i} - \sum_{v' \in \Lambda^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2) = \sum_{i=1}^{q_1} w'_{v^i} - \sum_{v' \in L^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2).$$

Our goal is to prove $U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b}) - U_v(w_{\hat{v}^r} + \epsilon_2) > \sum_{i=1}^d w'_{v^i} - \sum_{v' \in \Lambda^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2)$. So we shall prove

$$\sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1) \cdot \eta_{v'}(w_{\hat{v}^t} + \epsilon_1) = \sum_{v' \in L^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2) \cdot \eta_{v'}(w_{\hat{v}^r} + \epsilon_2) \tag{10}$$

and

$$\sum_{v' \in L^t} w_{v'}(w_{\hat{v}^t} + \epsilon_1) = \sum_{v' \in L^r} w_{v'}(w_{\hat{v}^r} + \epsilon_2). \tag{11}$$

Note that $B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\cup C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)=\{u|\alpha_u(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t})=\alpha_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\}$ and $B_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)\cup C_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)=\{u|\alpha_u(\tilde{G}^r;\boldsymbol{w}_{\Lambda^r})=\alpha_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)\}$. Note that the difference between $\tilde{G}^t$ and $\tilde{G}^r$ is due to the split of $\hat{v}$, so the C-class vertices in $(\tilde{G}^t;\boldsymbol{w}_{\Lambda^t})$ are the same as the C-class vertices in $(\tilde{G}^r;\boldsymbol{w}_{\Lambda^r})$, which means $C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)=C_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)$. Similarly, we have $B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\setminus\Lambda^t=B_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)\setminus\Lambda^r$. By the Claim 21-(8) and (9), we know that $\Lambda^t\cap C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)=\Lambda^r\cap C_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)\subseteq\{v^{p_1+1},\cdots,v^{q_1}\}$, $\Lambda^t\cap B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)=\{v(u^i)|u^i\in\tilde{G}^t\wedge u^i\in\{u^1,\cdots,u^{p_1}\}\}$ and $\Lambda^r\cap B_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)=\{v(u^i)|u^i\in\tilde{G}^r\wedge u^i\in\{u^1,\cdots,u^{p_1}\}\}$.

Now we analyze the weight of each vertex. Clearly, for each $u\in B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\cup C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\setminus\Lambda^t$, it must be $w_u(w_{\hat{v}^t}+\epsilon_1)=w_u(w_{\hat{v}^r}+\epsilon_2)$, then we focus on the fictitious nodes Λ^t and Λ^r. We claim that $\forall v^i\in\Lambda^t\cap C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1), w^t_{v^i}(w_{\hat{v}^t}+\epsilon_1)=w^r_{v^i}(w_{\hat{v}^r}+\epsilon_2)=w^s_{v^i}$. The main reason is when the weight of C-class fictitious node v^i increases, $\alpha_{v^i}=\alpha'_{v^i}$ always holds. But $\alpha_{v^i}(w_{\hat{v}^t}+\epsilon_1)=\alpha_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\geq\alpha'_{v^{q_1+1}}-f(\epsilon_1)$, where $f(\epsilon_1)$ is upper bounded by constant times of ϵ_1. Combined with $\alpha'_{v^1}\leq\cdots\leq\alpha'_{v^{q_1}}<\alpha'_{v^{q_1+1}}$, we know that $\forall v^i\in\Lambda^t\cap C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)$ has not increased. Similarly, $\forall v^i\in\Lambda^r\cap C_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)$ has not increased. So we have $\forall v^i\in\Lambda^t\cap C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1), w_{v^i}(w_{\hat{v}^t}+\epsilon_1)=w_{v^i}(w_{\hat{v}^r}+\epsilon_2)=w^s_{v^i}$. Combined with $\alpha_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)=\alpha_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)$, we have $w(\Lambda^t\cap B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1))=w(\Lambda^r\cap B_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2))$. Furthermore, we conclude that $U_{\Lambda^t\cap(B_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1)\cup C_{\hat{v}^t}(w_{\hat{v}^t}+\epsilon_1))}(w_{\hat{v}^t}+\epsilon_1)=U_{\Lambda^r\cap(B_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2)\cup C_{\hat{v}^r}(w_{\hat{v}^r}+\epsilon_2))}(w_{\hat{v}^r}+\epsilon_2)$. For each $v^i\in\{v^{p_1+1}.\cdots,v^{q_1}\}\wedge v^i\in\tilde{G}^t$ with $\alpha_{v^i}(w'_{v\hat{q_1}+1}$, there must be $w_{v^i}(w_{\hat{v}^t}+\epsilon_1)=w^s_{v^i}$ and $\alpha_{v^i}(w_{\hat{v}^t}+\epsilon_1)=\alpha^s_{v^i}$. Similarly, for each $v^i\in\{v^{p_1+1}.\cdots,v^{q_1}\}\wedge v^i\in\tilde{G}^r$ with $\alpha_{v^i}(w_{\hat{v}^r}+\epsilon_2)<\alpha'_{v^{q_1+1}}$, there must be $w_{v^i}(w_{\hat{v}^r}+\epsilon_2)=w^s_{v^i}$ and $\alpha_{v^i}(w_{\hat{v}^r}+\epsilon_2)=\alpha^s_{v^i}$. So the Equations Equation (10) and Equation (11) hold. Then $U_v(\tilde{G}^b;\boldsymbol{w}_{\Lambda^b})-U_v(w_{\hat{v}^r}+\epsilon_2)>\sum_{i=1}^d w'_{v^i}-\sum_{v'\in\Lambda^r}w_{v'}(w_{\hat{v}^r}+\epsilon_2)$.

The subsequent analysis is essentially the same as the proof in **Basis** when $\Delta U_v>\Delta w_v$. Consider Increase2$(\tilde{G}^r,\Lambda^r,(w_{\hat{v}^r}+\epsilon_2,\boldsymbol{w}_{\Lambda^r\setminus\{\hat{v}^r\}}),\boldsymbol{w}'_v,\hat{v}^r,0)$. We know that there must be at least one iteration of the DoWhile loop (line 2-18), precisely line 17, such that $U_v(\tilde{G};w_{\hat{v}}+z,\boldsymbol{w}_{\Lambda\setminus\{\hat{v}\}})-U_v(\tilde{G};\boldsymbol{w}_\Lambda)>z$. Let $(\tilde{G}^c;w_{\hat{v}^c}+z,\boldsymbol{w}_{\Lambda^c\setminus\{\hat{v}^c\}})$ be the next network after the last iteration that $\Delta U_v>\Delta w_v$ is satisfied. Then we have $U_v(G^*;\boldsymbol{w}'_v)-U_v(\tilde{G}^c;w_{\hat{v}^c}+z,\boldsymbol{w}_{\Lambda^c\setminus\{\hat{v}^c\}})\leq\sum_{i=1}^d w'_{v^i}-(\sum_{v'\in\Lambda^c}w^c_{v'}+z)$. To prove the inequality (Equation (9))), we shall prove $U_v(\tilde{G}^c;w_{\hat{v}^c}+z,\boldsymbol{w}_{\Lambda^c\setminus\{\hat{v}^c\}})-U_v(\tilde{G}^0;\boldsymbol{w}_{\Lambda^0})\leq(\sum_{v'\in\Lambda^c}w^c_{v'}+z)-\sum_{v'\in\Lambda^0}w^0_{v'}+\sum_{v^i\in\{v^{p_1+1},\cdots,v^{q_1}\}}w^s_{v^i}$.

Note that when the weight of $\hat{v}^c$ increases from $w_{\hat{v}^c}$ to $w_{\hat{v}^c}+z$, the bottleneck decomposition is changed with respect to single parameter. Similar to analysis in Appendix A.1, partition $[w_{\hat{v}^c},w_{\hat{v}^c}+z]$ into a number of disjoint subintervals $\{\langle a_i,b_i\rangle\}_i$. When $x\in\langle a_i,b_i\rangle$ the bottleneck decomposition is represented as $\mathcal{B}(\tilde{G}^c;x,\boldsymbol{w}_{\Lambda^c\setminus\{\hat{v}^c\}})=\mathcal{B}^i=\{(B^i_1,C^i_1),\cdots,(B^i_{k^i},C^i_{k^i})\}$. For the sake of convenience, simplify the notations $(\tilde{G}^c;x,\boldsymbol{w}_{\Lambda^c\setminus\{\hat{v}^c\}})$ to (x) such as $\mathcal{B}(x)$. Note that there must be at least one subinterval such that $U_v(b_i)-U_v(a_i)>b_i-a_i$. Let ℓ be the largest index for which $U_v(b_\ell)-U_v(a_\ell)>b_\ell-a_\ell$. Then we have $U_v(w_{\hat{v}^c}+z)-U_v(b_\ell)\leq w_{\hat{v}^c}+z-b_\ell$. So we only need to prove $U_v(b_\ell)-U_v(\tilde{G}^0;\boldsymbol{w}_{\Lambda^0})\leq(\sum_{v'\in\Lambda^c}w^c_{v'}-w_{\hat{v}^c}+b_\ell)-\sum_{v'\in\Lambda^0}w^0_{v'}+\sum_{v^i\in\{v^{p_1+1},\cdots,v^{q_1}\}}w^s_{v^i}$.

Suppose that $\hat{v}^c\in B^\ell_j\cup C^\ell_j$. We first claim there are some impacted fictitious nodes in C^ℓ_j. If not, there must be $U_v(b_\ell)-U_v(a_\ell)\leq\eta_{\hat{v}^c}(a_\ell)\cdot(b_\ell-a_\ell)<b_\ell-a_\ell$, which leads to a contradiction to the definition of ℓ. Then we claim $\Lambda^c\cap B^\ell_j=\{v(u^i)|u^i\in\tilde{G}^c\wedge u^i\in\{u^1,\cdots,u^{p_1}\}\}$, the reason is essentially the same as the proof in **Basis**.

Let us focus on the utility of $\Lambda^c\cap B^\ell_j$. By Lemma A.12, we have $U_{\Lambda^c\cap B^\ell_j}(b_\ell)<w(\Lambda^c\cap C^\ell_j)$ directly. Now we show that $w(\Lambda^c\cap C^\ell_j)\leq\sum_{v^i\in\{v^{p_1+1},\cdots,v^{q_1}\}}w^s_{v^i}$. Note that $\Lambda^c\cap C^\ell_j\subseteq\{v^{p_1+1},\cdots,v^{q_1}\}$. So it is sufficient to show for each $v^i\in\Lambda^c\cap C^\ell_j$, $w_{v^i}=w^s_{v^i}$, i.e., the weight of v^i has not increased. Recall that only when the α-ratio of C-class impacted fictitious node v^i reaches α'_{v^i}, its weight will increase in the While loop (line 8-16). In the subinterval $\langle a_\ell,b_\ell\rangle$, for each $v^i\in\Lambda^c\cap C^\ell_j$, α_{v^i}

is still decreasing. If w_{v^i} has increased, then $\alpha_{v^i} < \alpha'_{v^i}$. Combining the α-ratio of each vertex is non-increasing in each following iteration of Increase2, we have $\alpha_{v^i} < \alpha'_{v^i}$ at the end of Increase2, which is a contradiction. Let $\Lambda^t \cap B$ ($\Lambda^t \cap C$) denotes all B class (C class respectively) nodes in Λ^t, then $U_v(b_\ell) = U_{\Lambda^c \cap B}(b_\ell) + U_{\Lambda^c \cap C}(b_\ell)$, and

$$\begin{aligned} & U_{\Lambda^c \cap B}(b_\ell) - U_{\Lambda^0 \cap B}(\tilde{G}^0; \boldsymbol{w}_{\Lambda^0}) \\ \leq \quad & U_{\Lambda^c \cap B_j^\ell}(b_\ell) - U_{\hat{v}^0}(\tilde{G}^0; \boldsymbol{w}_{\Lambda^0}) \leq U_{\Lambda^c \cap B_j^\ell}(b_\ell) < w(\Lambda^c \cap C_j^\ell) \leq \sum_{v^i \in \{v^{p_1+1}, \cdots, v^{q_1}\}} w^s_{v^i}. \end{aligned}$$

Now let us discuss the utility of $\Lambda^c \cap C$. Consider Increase2($\tilde{G}^0, \Lambda^0, \boldsymbol{w}_{\Lambda^0}, \boldsymbol{w}'_v, \hat{v}^0, \alpha_{\hat{v}^c}(b_\ell)$) directly. We shall show that at each step of line 15, $U_{\Lambda^c \cap C}(\tilde{G}; w_{\hat{v}} + z, \boldsymbol{w}_{\Lambda \setminus \{\hat{v}\}}) - U_{\Lambda^c \cap C}(\tilde{G}; \boldsymbol{w}_\Lambda) \leq z$. Since $\hat{v}$ is a B-class vertex we only consider the utility of C-class vertices, this result can be deduced by Lemma A.11. Furthermore, in the While loop (line 8-16), we have $\Delta U_{v^j} = z \leq z + \alpha'_{v^j} \cdot z = \Delta w_v$. So $U_{\Lambda^c \cap C}(b_\ell) - U_{\Lambda^0 \cap C}(\tilde{G}^0; \boldsymbol{w}_{\Lambda^0}) \leq \Delta w_v = (\sum_{v' \in \Lambda^c} w^c_{v'} - w_{\hat{v}^c} + b_\ell) - \sum_{v' \in \Lambda^0} w^0_{v'}$. □

Proof of Lemma 3.7. Let $w_{\hat{v}^5}$ increase a sufficiently small $\epsilon > 0$. We first prove $\boldsymbol{w}^*_v = (w^*_{v^1}, \cdots, w^*_{v^d})$ is a Candidate Weight Profile of $(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}})$. Note that $(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5})$ and $\hat{v}^5$ are obtained by Increase($\tilde{G}^4, \Lambda^4, \boldsymbol{w}_{\Lambda^4}, \hat{v}^4, \alpha_v(G; \boldsymbol{w})$). The threshold $\alpha_v(G; \boldsymbol{w})$ guarantees that $\eta_{\hat{v}^5}(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) < \alpha_{\hat{v}^5}(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}) \leq 1$ for any sufficiently small $\epsilon > 0$. So we have (1) $\hat{v}^5$ is a B-class vertex with $\alpha_{\hat{v}^5}(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) < 1$; (2) $\forall v' \in \Lambda^5 \setminus \{\hat{v}^5\}, d_{v'} = 1$. Furthermore, for the weight assignment $\boldsymbol{w}^*_v = (w^*_{v^1}, \cdots, w^*_{v^d})$, we have

- $\forall u^i \in \Gamma(\hat{v}^5), x_{\hat{v}^5 u^i} \leq w^*_{v^i}$ on $(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}})$ by Claim 20-(1).
- $\forall v^i \in \Lambda^5 \setminus \{\hat{v}^5\}, w_{v^i} \leq w^*_{v^i}$ by Claim 20-(2).
- $\forall v^i \in \Lambda^5 \setminus \{\hat{v}^5\}$, if $w^5_{v^i} < w^*_{v^i}$, then v^i is a C-class vertex on both $(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}})$ and $(G^*; \boldsymbol{w}^*_v)$. Additionally, $\alpha_{v^i}(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) \geq \alpha_{v^i}(G^*; \boldsymbol{w}^*_v)$ by Claim 20-(4).

Recall the definition of impacted fictitious nodes $Q := \{v^i | \alpha_{\hat{v}^t}(G^*; \boldsymbol{w}^*_v) \leq \alpha_{v^i}(G^*; \boldsymbol{w}^*_v) \leq \alpha_{\hat{v}^5}(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}})\}$. For each C-class impacted fictitious node, i.e., $\forall v^i \in Q'$, we have $\alpha_{v^i}(G^*; \boldsymbol{w}^*_v) \leq \alpha_{\hat{v}^5}(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) < \alpha_{\hat{v}^5}(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}) = \alpha_v(G; \boldsymbol{w})$. By the definition of H, for each fictitious nodes v^i split during Increase($\tilde{G}^4, \Lambda^4, \boldsymbol{w}_{\Lambda^4}, \hat{v}^4, \alpha_v(G; \boldsymbol{w})$) in Stage 5, $\eta_{v^i}(G^*; \boldsymbol{w}^*_v) \leq 1/\alpha_v(G; \boldsymbol{w})$, which means v^i is either a B-class node or a C-class node with $\alpha_{v^i}(G^*; \boldsymbol{w}^*_v) \geq \alpha_v(G; \boldsymbol{w})$. So $w^5_{v^i} = w^4_{v^i}$, for each $v^i \in Q'$.

Thus,

$$U_v(G^*; \boldsymbol{w}^*_v) - U_v(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) \leq w_v - (w(\Lambda^5) + \epsilon) + w(\Lambda^4),$$

where $\sum_{i=1}^d w'_{v^i} = w_v$, $\sum_{v' \in \Lambda^5} w_{v'} = w(\Lambda^5)$ and $\sum_{v^i \in Q'} w^5_{v^i} \leq w(\Lambda^4)$.

Note that $U_v(G^*; \boldsymbol{w}^*_v) - U_v(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) + \epsilon$ is a continuous function, take the limit on this formula, we have

$$\begin{aligned} U_v(G^*; \boldsymbol{w}^*_v) - U_v(\tilde{G}^5; \boldsymbol{w}_{\Lambda^5}) \quad &= \quad \lim_{\epsilon \to 0} (U_v(G^*; \boldsymbol{w}^*_v) - U_v(\tilde{G}^5; w_{\hat{v}^5} + \epsilon, \boldsymbol{w}_{\Lambda^5 \setminus \{\hat{v}^5\}}) + \epsilon) \\ &\leq \quad w_v - w(\Lambda^5) + w(\Lambda^4). \end{aligned}$$

□