Accepted by the 22nd International Conference on Game Theory

A Sealed-Bid Unit-Demand Auction with Put Options*

Chinmayi Krishnappa
chinmayi@cs.utexas.edu

C. Greg Plaxton
plaxton@cs.utexas.edu

Abstract

We introduce a variant of the classic sealed-bid unit-demand auction in which each item has an associated put option. The put option of an item is held by the seller of the item, and gives the holder the right to sell the item to a specified target bidder at a specified strike price, regardless of market conditions. Unexercised put options expire when the auction terminates. In keeping with the underlying unit-demand framework, we assume that each bidder is the target of at most one put option. The details of each put option — holder, target, and strike price — are fixed and publicly available prior to the submission of unit-demand bids. We motivate our unit-demand auction by discussing applications to the reassignment of leases, and to the design of multi-round auctions.

In the classic sealed-bid unit-demand setting, the VCG mechanism provides a truthful auction with strong associated guarantees, including efficiency and envy-freedom. In our setting, the strike price of an item imposes a lower bound on its price. The VCG mechanism does not accommodate such lower bound constraints, and hence is not directly applicable. Moreover, the strong guarantees associated with the VCG mechanism in the classic setting cannot be achieved in our setting. We formulate an appropriate solution concept for our setting, and devise a truthful auction for implementing this solution concept. We show how to compute the outcome of our auction in polynomial time.

*Department of Computer Science, University of Texas at Austin, Austin, TX 78712. Supported by NSF Grant CCF–0635203 and THECB NHARP Grant 003658–0235–2007.

1 Introduction

Consider an auction in which many different items are for sale. Assume that a bidding agent assigns a separate value to each item, and is interested in acquiring at most one item. Such an agent is said to have unit-demand preferences. In a unit-demand auction, the bid of an agent takes the same form as a unit-demand preference function: The agent specifies an offer for each item, with the understanding that the bid can win at most one item. A put option associated with an item is a commitment between two parties — the holder of the put and the target of the put. The holder of the put possesses the right to sell the item to the target of the put at a specified "strike price", regardless of the current market prices.

In this paper, we consider a new variant of the classic sealed-bid unit-demand auction in which each item is associated with a predetermined fixed put that expires when the auction terminates. The holder of an item's put is the seller of the item and the target is an agent in the auction under the constraint that no agent is the target of more than one put. We restrict attention to the case of no side-payments — the outcome of the auction consists of an allocation and pricing of the items, and each agent who is allocated an item pays the item's price to the seller of the item. We seek to formulate a suitable solution concept for this setting and to design a truthful auction that implements this solution concept.

Our auction finds motivation in several applications, some of which are described below. As a first application, we introduce the following "Lease Exchange" problem. Consider a number of leased apartments and a number of agents with unit demand preferences for renting the apartments. The lessees of some apartments seek to break their current leases. We would like to reallocate and reprice the apartments such that each lessor receives at least the monthly rent being paid by the current lessee for the remainder of the lease term. Our auction provides a suitable way to do so. We model each apartment as an item in our auction. The lessor of each apartment holds a put of the apartment whose target is the current lessee and whose strike price is equal to the current monthly rent of the apartment. In practice, a lease involves many other important considerations including varying lease terms and lessee specific adjustments that have been ignored in our simple example. Such factors are easily handled by allowing lessors to specify additive amounts for each option and incorporating these amounts into the bidding based on the options chosen by each agent.

In combinatorial auction design, it is often useful to follow a two-phase approach e.g. the clock-proxy auction proposed by Ausubel et al. [3]. A second application of our auction is in implementing the second phase of a two-phase combinatorial auction. In general, our auction is a candidate for implementing the last round of any dynamic unit-demand auction.

A natural third application of our auction is in the design of a dynamic unit-demand auction in which each round is resolved using the sealed-bid auction proposed in this paper. In the first round of the dynamic auction, the seller of each item holds a put whose target is a dummy agent and whose strike price is equal to the reserve price of the item. In each subsequent round, each item is associated with a put whose target is the agent who is tentatively allocated to the item in the previous round, and whose strike price is the price of the item determined by the auction in the previous round. Such a dynamic auction generalizes the eBay auction for the unit-demand setting.

In the standard sealed-bid unit-demand context, one can apply the well-known Vickrey-Clarke-Groves (VCG) mechanism [16, 4, 7] to obtain an auction that is truthful, efficient, and envy-free [17]. For an item in our setting, the strike price of the item imposes a lower bound on the auction price of the item — by exercising the item's put, the seller of the item can ensure that the auction price is at least as high as the strike price. Due to these lower bound constraints on prices, we find that the VCG mechanism is not well-suited for our setting. Moreover, in our setting, we cannot guarantee the strong properties that are achieved by VCG in the classic setting. For example, consider an auction instance in which no agent bids on a particular item. The auction would be forced to allocate the item to the target of its put at its associated strike price even if such an allocation violates the envy-freedom property of the target. As a result, we formulate a solution concept that is appropriate for our setting.

Solution concept. We now informally introduce the solution concept adopted in the present work. In Section 2.4, we provide a formal specification of the associated equilibrium conditions.

We say that an agent u is "satisfied" in an outcome if u satisfies the standard property of envy-freedom (see Section 2.3 for the formal definition). For the classic sealed-bid unit-demand auction, a solution is said to be Walrasian if all of the agents are satisfied. Moreover, the VCG mechanism returns a Walrasian solution where the pricing is given by the unique minimum price vector over all Walrasian solutions. For the present problem, we relax the Walrasian conditions by requiring only a certain subset of the agents in an outcome to be satisfied. For example, we enforce the natural requirement that if an agent u is not allocated, then u is satisfied; equivalently, each component of the unit-demand bid of u is required to be less than or equal to the price of the corresponding item. Additionally, we require that if a non-allocated agent u is indifferent to being allocated to an item v that is allocated to some agent u', then u' is satisfied. Continuing in this manner, we require that if a non-allocated agent u is indifferent to being allocated to an item v that is allocated to agent u', and u' is indifferent to being allocated to an item v' (not equal to v) that is allocated to agent u'', then u'' is satisfied, and so on. In the terminology of the well-known Hungarian algorithm [12] for weighted bipartite matching, the aforementioned sequence of requirements may be stated more concisely as follows: If an agent u belongs to the Hungarian tree rooted at some non-allocated agent, then u is satisfied. (In Section 2.3, we formalize this requirement as a reachability condition in a suitably defined digraph.) The class of solutions meeting the latter requirement — which clearly includes all Walrasian solutions — plays a central role in our work. We refer to such solutions as "semi-Walrasian" solutions (see Condition 1 in Section 2.4).

A key observation underlying the design of our solution concept is that a semi-Walrasian solution implicitly partitions the items into two sets: the set of all items v such that any positive decrease in the price of v (while leaving the prices of all other items unchanged) yields a solution that is no longer semi-Walrasian, and the remaining items. In Section 2.3, the former items are defined to be "priced at market", and the latter items are defined to be "priced above market". For an item that is priced at market, the associated put need not be exercised in order to justify the price. For such an item v, the price is required to be at least the strike price (see Condition 2 in Section 2.4); otherwise, the seller of item v would prefer to exercise the put associated with v. For an item that is priced above market, the price can only be justified via exercise of the associated put; for such an item we require the price to be equal to the strike price (see Condition 3 in Section 2.4).

We require that the set of items V' priced above market be purchased by the set of agents U' who are targets of the associated puts (see Condition 4(a) in Section 2.4); the motivation for this requirement is that the items in V' are too expensive to be of interest to any of the remaining agents. The problem of determining a suitable allocation of V' to U' may be viewed as an instance of the house allocation problem [13]; accordingly, we enforce standard desiderata related to envy-freedom (see Condition 4(b) in Section 2.4) and Pareto-efficiency (see Condition 5 in Section 2.4).

The main contribution of this paper is a truthful auction that returns a pricing and allocation satisfying all of the equilibrium conditions outlined above. Furthermore, our auction admits a polynomial-time implementation. We also establish a property related to privacy preservation that is critical for the dynamic unit-demand auction application mentioned earlier in this introduction.

A two-phase approach. We construct a two-phase auction that draws on two fundamental techniques, one from the realm of mechanism design for numerical preferences — the dynamic unit-demand approximate auction of Demange et al. [5] — and one from the realm of mechanism design for ordinal preferences — the Top Trading Cycles (TTC) algorithm [13]. In what follows, we will refer to the dynamic unit-demand approximate auction of Demange et al. as DGS-approximate. The DGS-approximate auction is an ascending-price auction that proceeds in rounds. In each round, agents that are not tentatively allocated are consulted in round-robin order and given the opportunity to either select an item, or pass. If an unallocated agent u selects an item v, the tentative price of item v is increased by a parameter δ, and the tentative allocation is updated to reflect that item v is allocated to agent u. The DGS-approximate algorithm terminates when all of the unallocated agents pass.

Informally, the first phase of our auction corresponds to the following proxy version of DGS-approximate. We fix an initial tentative allocation and pricing of the items as follows: each item is allocated to the target of its put and has a price equal to the strike price of its put. We associate with each agent u, a proxy agent u' who employs the following strategy to bid on behalf of u in each round of DGS-approximate: if the tentative price on every item exceeds u's offer on the item, then u' passes; otherwise, u' selects an item with the highest utility for u (difference between u's offer on the item and the tentative price of the item). On termination of the auction, we identify as "unhappy" each agent who is allocated to an item, but strictly prefers some other item at the current prices. It is easy to see that the set of unhappy agents are a subset of the agents in the initial tentative allocation. We note that in the limit as δ approaches zero, the proxy based DGS-approximate auction achieves a relaxed form of efficiency: the auction is efficient if the unit-demand bid of each unhappy agent is replaced with a single offer on its allocated item equal to the strike price of the item. The proxy based DGS-approximate auction also achieves a relaxed form of envy-freedom — the auction is envy-free for all agents other than the set of unhappy agents.

The second phase of our auction corresponds to a single application of the TTC algorithm on a suitably defined instance of the house allocation problem [13]. The second phase of our auction affects only the allocation, and keeps the item prices unchanged. The proposed two-phase auction computes an outcome in the weak core and achieves the relaxed forms of efficiency and envy-freedom that are described with respect to the first phase. Alternatively, by employing the $\text{TC}^{\prec}$ algorithm of Jaramillo and Manjunath [9] in the second phase, we achieve Pareto-efficiency of our two-phase auction. The $\text{TC}^{\prec}$ algorithm runs in polynomial time and produces a strategy-proof and Pareto-efficient outcome for the house allocation problem in the absence of strict preferences.

The two-phase approach proposed above has two shortcomings. Firstly, the auction is not truthful for any positive value of δ. Secondly, the δ parameter associated with the DGS-approximate auction leads to a trade-off between speed and efficiency in the first phase. The running time of the first phase increases as δ diminishes (it takes $O(1/\delta)$ time to increase the item prices by a constant). Furthermore, for large values of δ, the auction is not efficient, even in the relaxed form discussed above. We address these shortcomings in our work. We provide a polynomial time implementation of the first phase auction. By carefully breaking ties, we successfully obtain a truthful first phase auction. The composition of two truthful auctions is not necessarily truthful. We successfully show that the two-phase auction obtained by composing the truthful first and second phases is also truthful.

The theory of two-sided matchings has a rich history. In recent work, Fujishige and Tamura [6] show the existence of a stable matching in a generalized many to many matching model with upper and lower bounds on payments. The model proposed by Fujishige and Tamura generalizes various previous two-sided matching models; see [6] for a discussion of the relevant literature. Like much of the prior work in this line of research, the work of Fujishige and Tamura does not address issues related to incentive compatibility. In applying the theory of two-sided matchings to the design of auctions, a fundamental challenge is to identify two-sided matching models where truthfulness is achievable. Aggarwal et al. [1] address this challenge for a special case of Fujishige and Tamura's model with applications to sponsored search auctions. Specifically, for the unit-demand auction setting with bidder and item specific minimum and maximum prices, Aggarwal et al. provide a truthful auction that computes a bidder optimal stable matching. Similarly, for the setting considered in the present work, our central focus is to obtain a truthful auction.

In the auction of Aggarwal et al. , each agent submits a value and a maximum price that the agent is willing to pay for each item. We observe that the algorithm of Aggarwal et al. can be used to implement the first phase of our auction in the special case where each agent that is the target of a put values the associated item strictly below than the strike price of the put. This special case of our auction can be modeled in the framework of Aggarwal et al. as follows: For each item and agent pair where the agent is the target of the item's put, we submit a value of infinity and a maximum price equal to the strike price of the put; for every other agent and item pair, we submit the agent's offer for the item as the value and maximum price; we set the reserve price of each item to the strike price of its put.

Recall the lease exchange problem that was discussed earlier in this introduction. A lessee may sometimes value his leased apartment below its current rent. In such cases, the first phase of our auction can be implemented using the algorithm of Aggarwal et al. [1] as described above. However, it is not difficult to see that a lessee's value for his leased apartment may in certain cases be higher that the current rent. Note that the decision to put the apartment in the auction may not rest with the lessee; in such situations, the lessee may be willing to pay a higher rent to retain his current apartment. Even in situations where the lessee decides to put the apartment in the auction, it is not uncommon to have lessees who are willing to risk winning their current apartments at higher monthly rates.

Organization of the paper. We present and analyze our auction in two layers. Within the technical body of this paper, we present two unit-demand auctions — the bottom-level auction and the top-level auction. The top-level auction is our proposed sealed-bid unit-demand auction and consists of two phases as discussed above. The first phase, which affects both the allocation and pricing, is defined in terms of the bottom-level auction. The bottom-level auction is dynamic with each round corresponding to an agent raising all of its offers by exactly one unit.

The remainder of the paper is organized as follows. Section 2 provides a formal description of our problem. Section 3 provides a foundation for the technical presentation to follow. Sections 4 and 5 present the bottom-level and top-level auctions respectively. Section 6 offers some concluding remarks.

2 Problem Formulation

In formulating our problem, we first introduce the notions of *bid-graphs* and *configurations*. Bid-graphs and configurations model the inputs and outputs of our auctions.

2.1 Agents and Items

We refer to the bidders in our auction as agents. In order to break ties among agents, we identify each agent with a binary string. We define the maximum over an empty set of agents as the empty agent ϵ. An item v in our auction is a pair where the first component is a binary string identifier, denoted $id(v)$, and the second component is an integer lower bound on the price of v, denoted $min(v)$. We allow the price of an item in our auction to be negative in order to support procurement-type auctions.

2.2 Bid-Graphs

A *bid-graph* encapsulates a set of items and a set of agents having unit-demand bids on the items. Formally, a bid-graph is an edge-weighted complete bipartite graph $G = (U, V, w)$, where U is a set of agents, V is a set of items, w is a function from the set $U \times V$ to the set of integers, and the following conditions are satisfied: (1) the cardinality of U is at least the cardinality of V; (2) for any agent u in U, agent u is nonempty; (3) for any pair of distinct items v and v' in V, we have $id(v) \neq id(v')$.

2.3 Configurations

A *configuration* encapsulates a bid-graph along with an associated outcome (allocation and pricing of the items).

A configuration χ is a triple (G, M, Φ), where $G = (U, V, w)$ is a bid-graph, M is a maximum cardinality matching (MCM) of G, and Φ is a potential function that maps each item v in V to an integer $\Phi(v)$ such that $\Phi(v) \geq min(v)$. In the definitions that follow, let $\chi = (G, M, \Phi)$ be a configuration where $G = (U, V, w)$.

We define $matched(\chi)$ as the subset of agents in U that are matched in M, and we define $unmatched(\chi)$ as the set of agents in $U \setminus matched(\chi)$. For any item v in V, we define $match(\chi, v)$ as the agent u in U such that the edge (u, v) belongs to M. For any agent u in U, we define $gap(\chi, u)$ as $w(u, v) - \Phi(v)$ if $match(\chi, v) = u$, and as zero otherwise.

We say that an agent u in U satisfies *envy-freedom* if $gap(\chi, u)$ is nonnegative and $gap(\chi, u) \geq w(u, v) - \Phi(v)$ for all items v in V. We say χ is *Walrasian* if every agent u in U satisfies envy-freedom.

We now characterize a suitable directed graph on χ and formulate a reachability condition on this directed graph. We define $digraph(\chi)$ as the directed graph $(U \cup V, A)$, where A is the set of arcs that includes for each edge (u, v) in $U \times V$ such that $w(u, v) - \Phi(v) \geq w(u, v') - \Phi(v')$ for every item v' in $V - v$: (1) an arc (v, u) if edge (u, v) is in M; (2) an arc (u, v) if edge (u, v) is not in M. For any agent u in $unmatched(\chi)$, we define $items(\chi, u)$ as the set of items v in V such that there exists a directed path from agent u to item v in $digraph(\chi)$. In the terminology of the Hungarian algorithm, the set $items(\chi, u)$ is the set of items reachable from agent u in the Hungarian tree rooted at u.

We now formally introduce *semi-Walrasian* configurations that we referred to during the discussion of the solution concept in the introduction. A configuration χ is *semi-Walrasian* if for every agent u in $unmatched(\chi)$ and every item v in $items(\chi, u)$, the agent $match(\chi, v)$ satisfies envy-freedom.

A semi-Walrasian configuration χ induces a partition of the items into two sets: the set of items that belong to $items(\chi, u)$ for some agent u in $unmatched(\chi)$, and the remaining items. We say that the items in the former set are *priced at market*, and that the remaining items are *priced above market*. For the standard sealed-bid unit-demand auction, the VCG mechanism yields a Walrasian configuration in which every item is priced at market.

2.4 Problem Statement

We use configurations to represent both the inputs and outputs of our auction. For any configuration $\chi = (G, M, \Phi)$ as input where $G = (U, v, w)$, and any item v in V, the target of v's put is the agent $match(\chi, v)$, and the strike price of v's put is $\Phi(v)$.

Given a configuration $\chi_0 = (G, M_0, \Phi_0)$ as input where $G = (U, V, w)$, we seek to devise a truthful mechanism that computes a configuration $\chi = (G, M, \Phi)$ satisfying the equilibrium conditions listed below.

1. The configuration χ is semi-Walrasian.
2. For any item v in V that is priced at market, we have $\Phi(v) \geq \Phi_0(v)$.
3. For any item v in V that is priced above market, we have $\Phi(v) = \Phi_0(v)$.
4. Let V' denote the set of all items in V that are priced above market. Then there is a permutation π of V' such that the following conditions hold.
   (a) For any item v in V', $match(\chi, \pi(v)) = match(\chi_0, v)$.
   (b) For any item v in V' having $match(\chi_0, v) = u$, $gap(\chi, u) \geq gap(\chi_0, u)$.
5. For any configuration $\chi' = (G, M', \Phi)$, if there exists an agent u in U such that $gap(\chi, u) < gap(\chi', u)$, then there exists an agent u' in U such that: (strong version) $gap(\chi', u') < gap(\chi, u')$; (weak version) u' is matched differently in M and M', and $gap(\chi', u') \leq gap(\chi, u')$.

The reader will note that above conditions are stated in terms of an agent's *gap* rather than the utility. For a unit-demand auction where agents bid truthfully, the *gap* of an agent is equal to its utility, and (the weak version of) Condition 5 corresponds to a solution in the (weak) core. Our reference to the (weak) core is in the sense defined by Jaramillo and Manjunath [9]. Consequently, for a truthful auction, a solution in the core satisfies Pareto-efficiency, and a solution is in the weak core satisfies the following property: no subset of agents can exchange their allocated items amongst themselves such that every agent in the subset experiences a strict improvement in utility.

3 Additional Definitions

In Section 2, we introduced the notions of bid-graphs and configurations. Below we present additional definitions related to bid-graphs and configurations. We also introduce additional types that are useful in our analysis.

3.1 Bid-Graphs

In general, an agent's unit-demand bid may not include an offer for every item in the bid-graph. In this paper, we assume that an agent's unit-demand bid includes an integer offer for every item in the big-graph, and we choose to represent the absence of an offer by a negative integer that is sufficiently large in magnitude.

For any set of items V, we define a (unit-demand) *bid* on V as a function that maps each item in V to an integer. In the definitions that follow, let $G = (U, V, w)$ be a bid-graph. We define $bids(G)$ as the set of all possible bids on the set V. For any agent u in U, we define $bid(G, u)$ as the bid β in $bids(G)$ such that $\beta(v) = w(u, v)$ for any item v in V.

For any nonempty agent u not in U, and any bid β in $bids(G)$, we define $add(G, u, \beta)$ as the bid-graph $G' = (U + u, V, w')$ where $bid(G', u) = \beta$ and $bid(G', u') = bid(G, u')$ for any agent u' in U. For any nonempty agent u not in U, any item v in V, and any integer z, we define $add(G, u, v, z)$ as $add(G, u, \beta)$, where β is the bid in $bids(G)$ such that $\beta(v) = z$ and $\beta(v') = min(v') - 1$ for any item v' in $V - v$.

For any any agent u in U, and any integer z, we define $shift(G, u, z)$ as the bid-graph (U, V, w') where $w'(u, v) = w(u, v) + z$ for any item v in V, and $w'(u', v) = w(u', v)$ for any agent u' in $U - u$ and any item v in V. For any any agent u in U, and any bid β in $bids(G)$, we define $subst(G, u, \beta)$

as the bid-graph $G' = (U, V, w')$ where $bid(G', u) = \beta$, and $bid(G', u') = bid(G, u')$ for any agent u' in $U - u$.

3.2 Configurations

In the definitions that follow, let $\chi = (G, M, \Phi)$ be a configuration where bid-graph $G = (U, V, w)$.

We say χ is *efficient* if M is a maximum weight MCM (MWMCM) of G. The function $agents(\chi)$ is the set U and the function $items(\chi)$ is the set V. For any item v in V, we define $amount(\chi, v)$ as $w(match(\chi, v), v)$, and we define $amount(\chi)$ as the function that maps each item v in V to $amount(\chi, v)$. We define $positive(\chi)$ as the set of agents u in U such that $gap(\chi, u) > 0$ and we define the set $nonpositive(\chi)$ as $U \setminus positive(\chi)$.

For any bid β in $bids(G)$, we define $max\text{-}gap(\chi, \beta)$ as the maximum over all items v in V, of $\beta(v) - \Phi(v)$. For any bid β in $bids(G)$, we define $pseudo\text{-}demand(\chi, \beta)$ as the set of all items v in V such that $\beta(v) - \Phi(v) = max\text{-}gap(\chi, \beta)$. For any bid β in $bids(G)$, we define $demand(\chi, \beta)$ as $pseudo\text{-}demand(\chi, \beta)$ if $max\text{-}gap(\chi, \beta) \geq 0$, and as the empty set $\emptyset$ otherwise. For any item v in V, we define $bids(\chi, v)$ as the set of all bids β in $bids(G)$ such that v belongs to $demand(\chi, \beta)$. For any agent u in U, we define $max\text{-}gap(\chi, u)$ as $max\text{-}gap(\chi, bid(G, u))$, we define $pseudo\text{-}demand(\chi, u)$ as $pseudo\text{-}demand(\chi, bid(G, u))$, and we define $demand(\chi, u)$ as $demand(\chi, bid(G, u))$.

For any nonempty agent u not in U, and any bid β in $bids(G)$, we define $add(\chi, u, \beta)$ as the configuration $(add(G, u, \beta), M, \Phi)$. Similarly, for any nonempty agent u not in U, any item v in V, and any integer z, we define $add(\chi, u, v, z)$ as the configuration $(add(G, u, v, z), M, \Phi)$.

For any agent u in U and any bid β in $bids(G)$, we denote the configuration $(subst(G, u, \beta), M, \Phi)$ by $subst(\chi, u, \beta)$. For any agent u in U and any nonempty agent u' not in U, we define $subst(\chi, u, u')$ as the configuration obtained from χ by replacing all occurrences of agent u with agent u'. For any agent u in U and any integer z, we define $shift(\chi, u, z)$ as the configuration $subst(\chi, u, \beta)$ where β is the bid in $bids(G)$ such that $\beta(v) = w(u, v) + 1$ for any item v in V.

3.3 Agent Colors

We identify special classes of configurations (e.g. Walrasian configurations) by adopting a suitable coloring scheme of the agents. Every agent in a configuration is colored white, gray, or black according to certain rules. Informally, a non-black agent satisfies envy-freedom, and a white agent satisfies a certain tie-breaking convention described below.

For any configuration χ, the color of any agent u in $agents(\chi)$ is determined as follows. We first consider the case where agent u belongs to $matched(\chi)$. In this case, let v be the item such that $match(\chi, v) = u$. If v does not belong to $pseudo\text{-}demand(\chi, u)$, then agent u is black. If v belongs to $demand(\chi, u)$, then agent u is white. Otherwise, agent u is gray. Next, we consider the case where agent u belongs to $unmatched(\chi)$. In this case, if $max\text{-}gap(\chi, u) > 0$, then agent u is black. If $max\text{-}gap(\chi, u) = 0$, and there exists some item v in $items(\chi, u)$ such that either $match(\chi, v)$ is non-white, or $match(\chi, v) < u$ and $gap(\chi, match(\chi, v)) = 0$, then agent u is gray. Otherwise, agent u is white.

We define $white(\chi)$ as the set of white agents in χ. The sets $gray(\chi)$, $black(\chi)$, $nonblack(\chi)$, and $nonwhite(\chi)$ are defined similarly.

3.4 Walrasian configurations

In this section, we formalize the notion of Walrasian solutions that were discussed in Section 2. The results of the present section and the following section on white configurations follow from standard results in the literature; these sections have been included in order to provide a self-contained presentation.

A configuration $\chi = (G, M, \Phi)$ is *Walrasian* if $\mathit{matched}(\chi) \subseteq \mathit{white}(\chi)$ and $\mathit{unmatched}(\chi) \subseteq \mathit{nonblack}(\chi)$. A bid-graph G is Walrasian if it admits a Walrasian configuration of the form (G, M, Φ). The following is a list of definitions and lemmas related to Walrasian configurations.

Lemma 3.1. *For any bid-graph G' of the form* $\mathrm{add}(G, u, \beta)$*, if bid-graph G is Walrasian then the bid-graph G' is Walrasian.*

Proof. Since G is Walrasian, there exists some Walrasian configuration of the form (G, M, Φ). It follows that M is an MWMCM of G, and thus there exists some MWMCM M' of G'. Koopmans and Beckmann [10] show the existence of prices satisfying the Walrasian properties in the unit-demand setting; thus, there exists some Walrasian configuration of the form (G', M', Φ'). □

Lemma 3.2. *If $\chi = (G, M, \Phi)$ is a Walrasian configuration, then M is an MWMCM of G.*

Proof. By definition, every agent in $\mathit{matched}(\chi)$ is white; thus for every item v in $\mathit{items}(\chi)$, we have $w(\mathit{match}(\chi, v), v) \geq \Phi(v)$. Similarly, for every agent u in $\mathit{unmatched}(\chi)$ and every item v in $\mathit{items}(\chi)$, we have $w((u, v)) \leq \Phi(v)$. It follows that any MCM of G with higher weight than M must match the set of agents in $\mathit{matched}(\chi)$.

Suppose there exists an MCM M' of G that matches agents in $\mathit{matched}(\chi)$ and has higher weight than M. Since every agent in $\mathit{matched}(\chi)$ is white, it follows that $w((u, v)) - \Phi(v) \geq w((u, v')) - \Phi(v')$, where v and v' are the items matched to u in M and M' respectively. Thus, $\displaystyle\sum_{(u,v)\in M} w((u, v)) - \Phi(v) \geq \sum_{(u',v)\in M'} w((u', v)) - \Phi(v)$. Thus, M' cannot be of higher weight than M. □

For any Walrasian bid-graph G, we define $\mathit{potentials}(G)$ as the set of all potential functions Φ such that there exists a Walrasian configuration of the form (G, M, Φ).

Lemma 3.3. *Let (G, M, Φ) be a Walrasian configuration where bid-graph $G = (U, V, w)$, and let M^* be an MWMCM of G. Let P be a path in the undirected graph $(U \cup V, M \oplus M^*)$. Let A denote the set of edges of P that are in M and let B denote the set of edges of P that are in M^*. Then the configuration $(G, M - A + B, \Phi)$ is Walrasian.*

Proof. Let path P be defined by the sequence of vertices $u_0, v_0, u_1, v_1, \cdots v_{k-1}, u_k$. We use the fact that (G, M, Φ) is Walrasian to derive the following equations.

$$w((u_k, v_{k-1})) \geq \Phi(v_{k-1}), \tag{1}$$

$$w((u_0, v_0)) \leq \Phi(v_0), \tag{2}$$

$$w((u_i, v_{i-1})) - \Phi(v_{i-1}) \geq w((u_i, v_i)) - \Phi(v_i) \tag{3}$$

for all i such that $0 < i < k$. Rewriting inequalities 1 and 2 as $w((u_k, v_{k-1})) - \Phi(v_{k-1}) \geq 0$ and $0 \geq w((u_0, v_0)) - \Phi(v_0)$, respectively, and then adding the latter inequalities to those obtained from 3, we find that the potential terms all cancel out, and we are left with the inequality $\sum_{0<i\leq k} w((u_i, v_{i-1})) \geq \sum_{0\leq i<k} w((u_i, v_i))$. Since M and M^* are both MWMCMs, the above inequality is tight. It follows that all of the inequalities we summed in order to obtain the above inequality are also tight. In other words, we have

$$w((u_k, v_{k-1})) = \Phi(v_{k-1}), \tag{4}$$

$$w((u_0, v_0)) = \Phi(v_0), \tag{5}$$

and

$$w((u_i, v_{i-1})) - \Phi(v_{i-1}) = w((u_i, v_i)) - \Phi(v_i) \tag{6}$$

for all i such that $0 < i < k$. Armed with the above equations, we are now ready to establish that (G, M', Φ) is Walrasian.

First we show that for all edges $e = (u, v)$ in $M' \setminus M$, $w(e) \geq \Phi(v)$. The latter claim follows immediately from Equations 5 and 6. Next we show that for all nodes v such that (u_k, v) belongs to $M \oplus M^*$, $w((u_k, v)) \leq \Phi(v)$. (Notice that u_k is the only node that is matched under M and unmatched under M'.) For $v = v_{k-1}$, the desired inequality holds tightly by Equation 4. For $v \neq v_{k-1}$, Walrasian property of (G, M, Φ) implies $w((u_k, v_{k-1})) - \Phi(v_{k-1}) \geq w((u_k, v)) - \Phi(v)$, so the desired inequality follows from Equation 4.

Finally, we show that for any edges $e = (u, v)$ and $e' = (u, v')$ such that (u, v) belongs to M', $w(e) - \Phi(v) \geq w(e') - \Phi(v')$. We consider three subcases. In the first subcase, assume that u is not equal to one of the u_i's, $0 \leq i \leq k$. In this subcase, the claim follows from the Walrasian property of (G, M, Φ). In the second subcase, assume that $u = u_0$, which is unmatched in M and matched via edge $e = (u_0, v_0)$ in M'. Let edge $e' = (u_0, v')$ belong to E, where $v' \neq v_0$. Then Equation 5 implies that $w(e) - \Phi(v_0) = 0$, and the Walrasian property of (G, M, Φ) implies that $w(e') \leq \Phi(v')$; hence $w(e) - \Phi(v_0) \geq w(e') - \Phi(v')$, as required. In the third subcase, assume that $u = u_i$ for some i such that $0 < i < k$. (We do not need to consider $u = u_k$ because u_k is unmatched in M'.) Notice that u_i is matched in M' via edge $e = (u_i, v_i)$. Let edge $e' = (u_i, v')$ belong to E, where $v' \neq v_i$. If $v' = v_{i-1}$, the required inequality $w(e) - \Phi(v_i) \geq w(e') - \Phi(v')$ holds tightly by Equation 6. Otherwise, Equation 6 implies that $w(e) - \Phi(v_i) = w((u_i, v_{i-1})) - \Phi(v_{i-1})$, the Walrasian property of (G, M, Φ) implies that $w((u_i, v_{i-1})) - \Phi(v_{i-1}) \geq w(e') - \Phi(v')$, and hence the required inequality $w(e) - \Phi(v_i) \geq w(e') - \Phi(v')$ holds. □

Lemma 3.4. *Let (G, M, Φ) be a Walrasian configuration where bid-graph $G = (U, V, w)$, and let M^* be an MWMCM of G. Let C be a cycle in the undirected graph $(U \cup V, M \oplus M^*)$. Let A denote the set of edges of C that are in M and let B denote the set of edges of C that are in M^*. Then the configuration $(G, M - A + B, \Phi)$ is Walrasian.*

Proof. The proof is similar to that of Lemma 3.3. □

Lemma 3.5. *For any Walrasian bid-graph G, any MWMCM M of G, and any potential function Φ in* potentials(G)*, the configuration (G, M, Φ) is Walrasian.*

Proof. Let $\mathcal{M}$ denote the set of all MWMCMs M' of G such that (G, M', Φ) is Walrasian. Since Φ is in *potentials*(G), the set $\mathcal{M}$ is guaranteed to be nonempty. Fix an MWMCM M^* in $\mathcal{M}$ such that $|M \setminus M^*|$ is minimized. It is sufficient to prove that $M = M^*$. We prove this by contradiction. Assume that $|M \setminus M^*|$ is equal to some positive integer k. Consider the undirected graph $G' = (U \cup V, M \oplus M^*)$. Since no vertex in G' has degree greater than 2, it can be partitioned into isolated vertices, simple paths of positive length, and simple cycles of positive length. Since $k > 0$, we are assured that G' contains either a simple path of positive length or a simple cycle of positive length. We consider these two cases separately.

- Graph G' contains a simple path P of positive length

  Let A and B denote the sets of edges of P that are in M^* and M respectively. By Lemma3.3, the configuration (G, M', Φ) is Walrasian, where $M' = M^* - A + B$. Furthermore, $|M \setminus M'| = k - |A| < k$, contradicting the definition of M^*.

- Graph G' contains a simple cycle C of positive length

  Let A and B denote the sets of edges of P that are in M^* and M respectively. By Lemma 3.4, the configuration (G, M', Φ) is Walrasian, where $M' = M^* - A + B$. Furthermore, $|M \setminus M'| = k - |A| < k$, contradicting the definition of M^*.

□

Lemma 3.6. *For any Walrasian bid-graph G, the functions in* potentials(G) *form a lattice with meet and join operations given by pointwise minimum and maximum, respectively.*

Proof. Let $G = (U, V, w)$ and let Φ_0 and Φ_1 be potential functions in *potentials*(G). Let Φ and Φ' be potential functions such that for any item v in V, $\Phi(v) = \min(\Phi_0(v), \Phi_1(v))$ and $\Phi'(v) = \max(\Phi_0(v), \Phi_1(v))$. We are required to show that configurations (G, M, Φ) and (G, M, Φ') are Walrasian. In what follows, we will show that (G, M, Φ) is Walrasian. By a similar argument, it follows that (G, M, Φ') is also Walrasian.

Observe that (G, M, Φ_0) and (G, M, Φ_1) are Walrasian; thus, for any edge (u, v) in M, we have $w((u, v)) \geq \Phi_0(v)$ and $w((u, v)) \geq \Phi_1(v)$; thus $w((u, v)) \geq \Phi(v)$. Similarly, for any agent u unmatched in M and any item v in V, we have $w((u, v)) \leq \Phi_0(v)$ and $w((u, v)) \leq \Phi_1(v)$; thus $w((u, v)) \leq \Phi(v)$. It now remains to be shown the following condition: for any edge (u, v) in M and any item v' in $V - v$, we have $w((u, v)) - \Phi(v) \geq w((u, v')) - \Phi(v')$. We accomplish this by showing that the condition holds when $\Phi(v) = \Phi_0(v)$. It follows by symmetry that the condition also holds when $\Phi(v) = \Phi_1(v)$. Fix an arbitrary item v' in V, and consider the following two cases.

- $\Phi(v') = \Phi_0(v')$.

  Since (G, M, Φ_0) is Walrasian, we have $w((u, v)) - \Phi_0(v) \geq w((u, v')) - \Phi_0(v')$; using $\Phi(v) = \Phi_0(v)$ and $\Phi(v') = \Phi_0(v')$, we obtain the desired inequality $w((u, v)) - \Phi(v) \geq w((u, v')) - \Phi(v')$.

- $\Phi(v') = \Phi_1(v')$

  By the Walrasian property of configuration (G, M, Φ_1), we have $w((u, v)) - \Phi_1(v) \geq w((u, v')) - \Phi_1(v')$. Since $\Phi(v) = \Phi_0(v)$, we have $\Phi_0(v) \leq \Phi_1(v)$. Thus $w((u, v)) - \Phi_0(v) \geq w((u, v')) - \Phi_1(v')$; using $\Phi(v) = \Phi_0(v)$ and $\Phi(v') = \Phi_1(v')$, we obtain the desired inequality $w((u, v)) - \Phi(v) \geq w((u, v')) - \Phi(v')$.

□

For any Walrasian bid-graph G, we define *max-potential*(G) and *min-potential*(G) as the maximum and minimum functions in *potentials*(G); the existence of these functions is guaranteed by Lemma 3.6.

Lemma 3.7. *For any bid-graph G' of the form* add(G, u, v, z) *where bid-graph G is Walrasian, there exists a unique integer z_0 such that the following conditions hold:*

- *If $z > z_0$ and configuration $\chi = (G', M, \Phi)$ is Walrasian, then agent u belongs to* matched(χ).
- *If $z < z_0$ and configuration $\chi = (G', M, \Phi)$ is Walrasian, then agent u belongs to* unmatched(χ).
- *If $z = z_0$, then there exist Walrasian configurations $\chi = (G', M, \Phi)$ and $\chi' = (G', M', \Phi)$ such that agent u belongs to* matched$(\chi) \cap$ unmatched(χ').

Proof. Let $\Phi = \textit{max-potential}(G)$ and let M be some MWMCM of G. Since G is Walrasian, (G, M, Φ) is Walrasian. Thus the weight of any MWMCM of G is at least equal to $\sum_{v \in G} \Phi(v)$. By Lemma 3.1, G' is Walrasian and by Lemma 3.5, any configuration of the form (G', M', Φ') is Walrasian, where M' is an MWMCM of G' and Φ' is in $\textit{potentials}(G')$. It is easy to see that when $z < \Phi(v)$, (G', M, Φ) is Walrasian and thus u is unmatched in every Walrasian configuration of G'. We now consider the case when $z = \Phi(v)$. There exists some item v' in $\textit{items}(\chi, u)$ such that $\textit{gap}(\chi, \textit{match}(\chi, v')) = 0$ as otherwise $\Phi(v'')$ can be incremented for each item v'' in $\textit{items}(\chi, u)$. In this case, by Lemma 3.3, u is matched in some Walrasian configuration of G'. Further, it is easy to see that if u is matched in some Walrasian configuration χ of G', then u is matched in every Walrasian configuration of $\textit{shift}(\chi, u, 1)$. Thus, there exists a unique $z_0 = \textit{max-potential}(v)$ with the desired property. □

For any Walrasian bid-graph $G = (U, V, w)$ and any item v in V, we define $\textit{threshold}(G, v)$ as the unique integer z_0 of Lemma 3.7, and we define $\textit{threshold}(G)$ as the function that maps each item v in V to $\textit{threshold}(G, v)$.

Lemma 3.8. *For any Walrasian bid-graph G, we have* $\text{threshold}(G) = \text{max-potential}(G)$.

Proof. In the proof of Lemma 3.7, we showed that for any item v in G, $\textit{threshold}(G, v) = \textit{max-potential}(v)$. □

For any Walrasian bid-graph $G = (U, V, w)$, we define $\textit{price}(G)$ as $\textit{min-potential}(G)$, and for any item v in V, we define $\textit{price}(G, v)$ as $\Phi(v)$, where Φ is equal to $\textit{min-potential}(G)$.

Lemma 3.9. *For any bid-graph G' of the form* $\text{add}(G, u, \beta)$ *where bid-graph $G = (U, V, w)$ is Walrasian, if $\beta(v) \leq \text{price}(G, v)$ for every item v in V, then* $\text{price}(G') = \text{price}(G)$.

Proof. Let M be an MWMCM of G. By Lemma 3.5, $\chi = (G, M, \textit{price}(G))$ is Walrasian and thus the weight of M is at least $\sum_{v \in G} \textit{price}(G, v)$. Since $\beta(v) \leq \textit{price}(G, v)$ for every item v in V, M is an MWMCM of G'. It follows that $(G', M, \textit{price}(G))$ is a Walrasian configuration. Further, if there exists a potential function Φ in $\textit{potentials}(G')$ such that $\Phi < \textit{price}(G)$, then (G, M, Φ) is Walrasian; this contradicts the definition of $\textit{price}(G)$. Thus, $\textit{price}(G') = \textit{price}(G)$. □

Lemma 3.10. *For any Walrasian configuration $\chi = (G, M, \Phi)$, we have* $\text{price}(G) \leq \text{threshold}(G) \leq \text{amount}(\chi)$.

Proof. By definition, $\textit{price}(G) = \textit{min-potential}(G)$ and by Lemma 3.8, $\textit{threshold}(G) = \textit{max-potential}(G)$; thus $\textit{price}(G) \leq \textit{threshold}(G)$. Since $\textit{threshold}(G) = \textit{max-potential}(G)$, it follows from Lemma 3.5 that $(G, M, \textit{threshold}(G))$ is Walrasian. By the definition of Walrasian configurations, it follows that $\textit{amount}(\chi) \geq \textit{threshold}(G)$. □

Lemma 3.11. *Let G' be a bid-graph of the form* $\text{add}(G, u, \beta)$ *where bid-graph $G = (U, V, w)$ is Walrasian. Let Δ denote the maximum over all items v in V, of* $\beta(v) - \text{threshold}(G, v)$, *and let V' denote the set of all items v in V such that* $\beta(v) - \text{threshold}(G, v) = \Delta$. *Then the following conditions hold:*

- *If $\Delta > 0$ and configuration $\chi = (G', M, \Phi)$ is Walrasian, then* $\text{match}(\chi, v) = u$ *for some item v in V', and* $\text{price}(G', v) = \text{threshold}(G, v)$ *for for every item v in V'.*
- *If $\Delta < 0$ and configuration $\chi = (G', M, \Phi)$ is Walrasian, then agent u is unmatched in M.*
- *If $\Delta = 0$, then there exist Walrasian configurations $\chi = (G', M, \Phi)$ and $\chi' = (G', M', \Phi)$ such that agent u belongs to* $\text{matched}(\chi) \cap \text{unmatched}(\chi')$.

- *If* $\Delta \leq 0$, *then* threshold(G') = threshold(G).

Proof. Let M be an MWMCM of G and let $\Phi = \textit{max-potential}(G)$. By Lemma 3.5, $\chi_0 = (G, M, \Phi)$ is Walrasian; thus, by definition, the weight of M is at least $\sum_{v \in G} \Phi(v)$. We consider the following cases:

- $\Delta < 0$

  In this case, $\beta(v) < \Phi(v)$ for each item v in G, and thus, M is an MWMCM of G'. Thus, (G', M, Φ) is Walrasian, and it follows by the definition of Walrasian configurations that u is unmatched in M.

- $\Delta \leq 0$

  By the same argument as in the previous case, (G', M, Φ) is Walrasian. Further, if there exists a potential function Φ' in $\textit{potentials}(G')$ such that $\Phi' > \textit{max-potential}(G)$, then (G, M, Φ') is Walrasian; this contradicts the definition of $\textit{max-potential}(G)$. Thus, $\textit{max-potential}(G') = \textit{max-potential}(G)$. By Lemma 3.8, $\textit{max-potential}(G) = \textit{threshold}(G)$ and $\textit{max-potential}(G') = \textit{threshold}(G')$; thus $\textit{threshold}(G) = \textit{threshold}(G')$.

- $\Delta \geq 0$

  There exists at least one item v in $\textit{items}(\chi_0, u)$ such that $\textit{gap}(\chi_0, \textit{match}(\chi_0, v)) = 0$ as otherwise the potential associated with each item in $\textit{items}(\chi_0, u)$ can be incremented while χ remains Walrasian, violating the definition of $\textit{max-potential}(G)$. When $\Delta \geq 0$, v belongs to $\textit{items}(\chi_0, u)$; thus there exists an augmenting path in $\textit{digraph}(\chi_0)$. It follows that u belongs to some MWMCM of G'.

  When $\Delta = 0$, it is easy to see that (G', M, Φ) is Walrasian; thus, Φ is in $\textit{potentials}(G')$. By Lemma 3.5, $(G', M, \textit{price}(G'))$ is Walrasian. However, we know that u is not matched in M. It follows that $\textit{price}(G', v) \geq \Phi(v)$ for each item v in V' to satisfy the Walrasian property of $(G', M, \textit{price}(G'))$. Thus $\textit{price}(G', v) = \Phi(v) = \textit{threshold}(G, v)$ for each item v in V'. Above we showed that when $\Delta = 0$, u is matched in some MWMCM M' of G'; thus $(G', M', \textit{price}(G'))$ is Walrasian. It is easy to see that when $\Delta > 0$, $(G', M', \textit{price}(G'))$ remains Walrasian and thus, $\textit{price}(G', v) = \textit{threshold}(G, v)$ for each item v in V'.

□

3.5 White configurations

A configuration χ is *white* if $\textit{agents}(\chi) = \textit{white}(\chi)$. The following is a set of definitions and lemmas related to white configurations. The proofs of these lemmas are similar to those for the corresponding results established in Section 3.4 for Walrasian configurations.

Lemma 3.12. *For any Walrasian bid-graph G, there exists a white configuration of the form (G, M, Φ), and for any white configuration of the form (G, M, Φ), the bid-graph G is Walrasian.*

Proof. By definition, every white configuration is Walrasian. Thus, for any white configuration of the form (G, M, Φ), the bid-graph G is Walrasian. Consider any Walrasian configuration $\chi = (G, M, \Phi)$ that is not white. Then there exists at least one agent u in $\textit{unmatched}(\chi)$ such that for some item v in $\textit{items}(\chi, u)$, $\textit{gap}(\chi, \textit{match}(\chi, v)) = 0$ and $\textit{match}(\chi, v) < u$. By repeated application of Lemma 3.3, χ can be transformed to a white configuration. □

Lemma 3.13. *For any Walrasian bid-graph G and any potential function Φ in* potentials(G)*, there exists a white configuration of the form (G, M, Φ).*

Proof. By Lemma 3.5, any configuration $\chi = (G, M', \Phi)$ is Walrasian where M' is an MWMCM of G. Then there exists at least one agent u in $unmatched(\chi)$ such that $gap(\chi, match(\chi, v)) = 0$ and $match(\chi, v) < u$ for some item v in $items(\chi, u)$. By repeated application of Lemma 3.3, χ can be transformed to a white configuration. Thus, there exists some MWMCM M of G such that (G, M, Φ) is white. □

Lemma 3.14. *For any Walrasian bid-graph G and any pair of white configurations $\chi = (G, M, \Phi)$ and $\chi' = (G, M', \Phi')$, we have* matched(χ) = matched(χ')*.*

Proof. Let $\chi_0 = (G, M, \textit{max-potential}(G))$ and let $\chi_1 = (G, M', \textit{max-potential}(G))$. By Lemma 3.5, χ_0 and χ_1 are Walrasian. Further, since $\textit{max-potential}(G) \geq \Phi$ and $\textit{max-potential}(G) \geq \Phi'$, it follows that $white(\chi) \cap unmatched(\chi) = white(\chi_0) \cap unmatched(\chi_0)$ and $white(\chi') \cap unmatched(\chi') = white(\chi_1) \cap unmatched(\chi_1)$; thus χ_0 and χ_1 are white configurations. If $M \oplus M'$ consists of only cycles and no paths, then it follows that $matched(\chi) = matched(\chi')$. Suppose there exists some path P in $M \oplus M'$ with endpoints u in $matched(\chi_0) \cap unmatched(\chi_1)$ and u' in $matched(\chi_1) \cap unmatched(\chi_0)$. By Lemma 3.3, u belongs to $agents(\chi_0, u')$ and u' belongs to $agents(\chi_1, u)$. Since $u < u'$ or $u' < u$, this violates the assumption that χ_0 and χ_1 are both white. Thus, there is no such path P and $matched(\chi) = matched(\chi')$. □

By Lemmas 3.12 and 3.14, we can conclude that for any Walrasian bid-graph G, there exists a unique set of matched agents in any white configuration of the form (G, M, Φ). We denote this unique set of matched agents by $matched(G)$.

Lemma 3.15. *For any white configuration (G, M, Φ), and for any potential function Φ' in* potentials(G)*, the configuration (G, M, Φ') is white.*

Proof. By Lemma 3.13, there exists an MWMCM M' of G such that the configuration $\chi' = (G, M', \Phi')$ is white. Let $\chi = (G, M, \Phi)$ and let $\chi'' = (G, M, \Phi')$. By Lemma 3.5, χ'' is Walrasian and by Lemma 3.14, $matched(\chi) = matched(\chi')$. Thus, $matched(\chi) = matched(\chi'')$ and $matched(\chi'') \subseteq white(\chi'')$. Since χ'' and χ are Walrasian, for any agent u in $unmatched(\chi'')$, we have $agents(\chi'', u) = agents(\chi, u)$; since u is white in χ, it follows that u is white in χ''. □

In what follows, we sometimes compare amount-agent pairs. Such comparisons are resolved lexicographically.

Lemma 3.16. *For any Walrasian bid-graph $G = (U, V, w)$, any item v in V, and any white configurations $\chi = (G, M, \Phi)$ and $\chi' = (G, M', \Phi)$, we have* agents(χ, v) = agents(χ', v)*.*

Proof. By Lemma 3.14, we have $matched(\chi) = matched(\chi')$; thus $unmatched(\chi) = unmatched(\chi')$. Since χ and χ' are Walrasian, for any item v, $gap(\chi, match(\chi, v)) = \max_{v \in V} w(match(\chi, v), v) - \Phi(v)$. Further, by the definition of $digraph(\chi)$, arc $(match(\chi, v), v')$ belongs to $digraph(\chi)$ for every item v' in $demand(\chi, match(\chi, v))$. By a similar argument, arc $(match(\chi', v), v')$ belongs to $digraph(\chi')$ for every item v' in $demand(\chi', match(\chi', v))$. Thus it follows that $agents(\chi, v) = agents(\chi', v)$. □

For any Walrasian bid-graph $G = (U, V, w)$, any potential function Φ in $potentials(G)$, and any item v in V, we define $agents(G, \Phi, v)$ as the unique set $agents(\chi, v)$ of Lemma 3.16, where $\chi = (G, M, \Phi)$ is a white configuration whose existence is guaranteed by Lemma 3.13. For any Walrasian bid-graph $G = (U, V, w)$, and any item v in V, we define $agents(G, v)$ as $agents(G, price(G), v)$.

For any Walrasian bid-graph $G = (U, V, w)$, and any item v in V, we define $price^*(G, v)$ as $(price(G), u_0)$, where u_0 is the maximum agent in $agents(G, v)$. Recall that the maximum agent over an empty set is defined as ϵ. In addition, we define *price*$^*(G)$ as the function that maps each item v in V to *price*$^*(G, v)$.

For the following lemmas, we view bids and prices as pairs — if u has an offer of z on item v, we view the offer as the pair (z, u).

Lemma 3.17. *For any bid-graph G' of the form* $\text{add}(G, u, v, z)$ *where bid-graph $G = (U, V, w)$ is Walrasian, there exists a unique agent u_0 in U such that agent u belongs to* $\text{matched}(G')$ *if and only if* $(z, u) > (\text{threshold}(G, v), u_0)$.

Proof. The proof is similar to that of Lemma 3.9 when bids and prices are viewed as pairs. □

For any Walrasian bid-graph $G = (U, V, w)$ and any item v in V, we define *threshold*$^*(G, v)$ as the unique pair $(\textit{threshold}(G, v), u_0)$ of Lemma 3.17.

Lemma 3.18. *For any bid-graph G' of the form* $\text{add}(G, u, \beta)$ *where bid-graph G is Walrasian, if the pair* $(\beta(v), u) < \text{price}^*(G, v)$ *for all items v in V, then* $\text{price}^*(G') = \text{price}^*(G)$.

Proof. The proof is similar to that of Lemma 3.9 when bids and prices are viewed as pairs. □

For any configuration $\chi = (G, M, \Phi)$ where $G = (U, V, w)$, and any item v in V, we define *amount*$^*(\chi, v)$ as the pair $(\textit{amount}(\chi, v), \textit{match}(\chi, v))$, and we define *amount*$^*(\chi)$ as the function that maps each item v in V to *amount*$^*(\chi, v)$.

Lemma 3.19. *For any white configuration $\chi = (G, M, \Phi)$, we have* $\text{price}^*(G) \leq \text{threshold}^*(G) \leq \text{amount}^*(\chi)$.

Proof. The proof is similar to that of Lemma 3.10 when bids and prices are viewed as pairs. □

Lemma 3.20. *Let G' be a bid-graph of the form* $\text{add}(G, u, \beta)$ *where bid-graph $G = (U, V, w)$ is Walrasian. Let Δ denote the maximum, over all items v in V, of* $\beta(v) - \text{threshold}(G, v)$*, and let V' denote the set of all items v in V such that* $\beta(v) - \text{threshold}(G, v) = \Delta$. *Let u_0 denote the minimum, over all items v in V', of the second component of the pair* $\text{threshold}^*(G, v)$. *Then the following conditions hold:*

- *If the pair $(\Delta, u) > (0, u_0)$ and configuration $\chi = (G', M, \Phi)$ is white, then* $\text{match}(\chi, v) = u$ *for some item v in V', and* $\text{price}(G', v) = \text{threshold}(G, v)$ *for every item v in V'.*
- *If the pair $(\Delta, u) < (0, u_0)$ and configuration $\chi = (G', M, \Phi)$ is white, then agent u is unmatched in M and* $\text{threshold}^*(G') = \text{threshold}^*(G)$.

Proof. The proof is similar to that of Lemma 3.11 when bids and prices are viewed as pairs. □

3.6 Quiescent configurations

The inputs and outputs of the bottom-level auction of Section 4 are *quiescent* configurations. A configuration $\chi = (G, M, \Phi)$ is quiescent if *unmatched*$(\chi) \subseteq$ *white*(χ), and for any agent u in *black*(χ) where $\beta = \textit{bid}(G, u)$, we have $\beta(v) < \Phi(v)$ for all items v in *items*(χ). It is easy to see that quiescent configurations satisfy equilibrium conditions 1, 2, and 3 of Section 2.4.

For any configuration $\chi = (G, M, \Phi)$ where $G = (U, V, w)$, and any agent u in U, we say χ is u-quiescent if either (1) u belongs to *unmatched*$(\chi) \cap$ *gray*(χ) and (G', M, Φ) is quiescent, where $G' = (U - u, V, w)$, or (2) u belongs to *matched*(χ) and *shift*$(\chi, u, 1)$ is quiescent.

3.7 ECCs

We use tie-breaking to handle degeneracy in the bottom-level auction of Section 4; in each round, we break ties such that the set of allocated agents is uniquely determined. Below we identify equivalence classes of configurations (ECCs) that adhere to this tie breaking scheme.

For any pair of configurations $\chi = (G, M, \Phi)$ and $\chi' = (G, M', \Phi)$, we write $\chi \sim \chi'$ if $\mathit{matched}(\chi) = \mathit{matched}(\chi')$, $\mathit{nonwhite}(\chi) = \mathit{nonwhite}(\chi')$, and for any item v in $\mathit{items}(\chi)$ such that $\mathit{match}(\chi, v)$ is non-white, we have $\mathit{match}(\chi, v) = \mathit{match}(\chi', v)$. Observe that $\sim$ is an equivalence relation and thus partitions the set of all configurations into equivalence classes. We refer to an equivalence class of configurations as an *ECC*, and we use the notation $[\chi]$ to refer to the ECC of a given configuration χ. By definition, for any ECC X, there exists a unique bid-graph G_0 and a unique potential function Φ_0 such that every configuration in X is of the form (G_0, M, Φ_0). We define *bid-graph*(X) and $\mathit{potential}(X)$ as G_0 and Φ_0 respectively. We define $\mathit{potential}(X, v)$ as $\Phi_0(v)$, for any item v in V, where $G_0 = (U, V, w)$. An ECC X is *quiescent* if every configuration χ in X is quiescent. We define u-quiescent ECCs similarly.

It follows by definition that for any ECC X that every configuration χ in X is associated with the same set of agents; we define $\mathit{agents}(X)$ to be this unique set of agents. We define the following similarly: $\mathit{items}(X)$, $\mathit{matched}(X)$, and $\mathit{unmatched}(X)$. For any ECC X and any agent u in $\mathit{unmatched}(X)$, we define $\mathit{items}(X, u)$ to be the unique set of items given by Lemma 3.21. We define the following similarly: $\mathit{gray}(X)$, $\mathit{white}(X)$, $\mathit{black}(X)$, $\mathit{nonwhite}(X)$, $\mathit{nonblack}(X)$, $\mathit{enabled}(X)$, $\mathit{positive}(X)$, $\mathit{nonpositive}(X)$, $\mathit{gap}(X, u)$, *max-gap*(X, u), $\mathit{demand}(X, u)$, *pseudo-demand*(X, u), $\mathit{bids}(X, v)$, $\mathit{items}(X, u)$, $\mathit{agents}(X, u)$, and $\mathit{agents}(X, v)$.

For any ECC X, any agent u in $\mathit{agents}(X)$ and any integer z such that either u belongs to $\mathit{unmatched}(X)$ or $z \geq 0$, we define $\mathit{shift}(X, u, z)$ as $[\mathit{shift}(\chi, z, u)]$ where χ is any configuration in X. For any ECC X and any agent u in $\mathit{agents}(X)$ and any agent u' not in $\mathit{agents}(X)$, we define $\mathit{subst}(X, u, u')$ as the ECC $\cup_{\chi \in X}[\mathit{subst}(\chi, u, u')]$ given by Lemma 3.23. We define the following similarly: $\mathit{add}(X, u, \beta)$, and $\mathit{add}(X, u, v, z)$.

Lemma 3.21. *For any ECC X, any agent u in* unmatched(X)*, and any pair of configurations χ and χ' in X, we have* items$(\chi, u) =$ items(χ', u).

Proof. Configurations χ and χ' are associated with the same potential function. By definition, $\mathit{matched}(\chi) = \mathit{matched}(\chi')$, $\mathit{nonwhite}(\chi) = \mathit{nonwhite}(\chi')$, and for any agent u'' in $\mathit{matched}(\chi) \cap \mathit{nonwhite}(\chi)$, u is matched to the same item in χ and χ'. Thus, for any item v, $\mathit{gap}(\chi, \mathit{match}(\chi, v)) = \max_{v \in V} w(\mathit{match}(\chi, v), v) - \Phi(v)$. Further, by the definition of $\mathit{digraph}(\chi)$, arc $(\mathit{match}(\chi, v), v')$ belongs to $\mathit{digraph}(\chi)$ for every item v' in $\mathit{demand}(\chi, \mathit{match}(\chi, v))$. By a similar argument, arc $(\mathit{match}(\chi', v), v')$ belongs to $\mathit{digraph}(\chi')$ for every item v' in $\mathit{demand}(\chi', \mathit{match}(\chi', v))$. It follows that $\mathit{items}(\chi, u) = \mathit{items}(\chi', u)$. □

Lemma 3.22. *For any quiescent configuration χ, the ECC $[\chi]$ is quiescent.*

Proof. Let $\chi = (G, M, \Phi)$ and let χ' be any configuration in $[\chi]$. By definition, $\mathit{potential}([\chi']) = \Phi$, $\mathit{unmatched}(\chi') = \mathit{unmatched}(\chi)$, and for every agent u in $\mathit{matched}(\chi) \cap \mathit{nonwhite}(\chi)$, there exists an item v in $\mathit{items}(\chi)$ such that $\mathit{match}(\chi, v) = \mathit{match}(\chi', v) = u$. It follows that since χ is quiescent, χ' is quiescent. □

Lemma 3.23. *For any ECC X, any agent u in* agents(X)*, and any agent u' not in* agents(X)*, the set of configurations given by $\cup_{\chi \in X}[\mathrm{subst}(\chi, u, u')]$ is an ECC.*

Proof. The proof follows from the definition of ECCs and the definition of $\mathit{subst}(\chi, u, u')$ for any configuration χ in X. □

4 Bottom-Level Auction

As discussed in the introduction, we present our auction in two layers. The bottom-level auction constitutes the first layer and is a building block of the first phase of our top-level auction.

The bottom-level auction is dynamic. The input to each round of the bottom-level auction is a quiescent ECC. Recall that a quiescent ECC satisfies equilibrium conditions 1, 2, and 3 of Section 2.4. In each round of the bottom-level auction, a single agent increments its offers on all items by one unit, and the round is resolved by incorporating the bid increment while continuing to satisfy equilibrium conditions 1, 2,and 3. (We note that equilibrium conditions 4 and 5 are incorporated by the second phase of the top-level auction discussed in Section 5.2.)

4.1 Description

For any configuration χ, we define *enabled*(χ) as the set of agents u in *agents*(χ) such that either (1) u belongs to *white*(χ), or (2) u belongs to *nonwhite*(χ) and for all items v in *items*(χ), we have $\beta(v) < \Phi(v) - 1$, where $\beta = bid(G, u)$. We define *enabled*(X) similarly.

The bottom-level auction takes a quiescent ECC as input and updates the ECC over a sequence of rounds. In a general round of the bottom-level auction, a single enabled agent in the ECC invokes the function *raise* defined below. Informally, an invocation of *raise* by an agent corresponds to the agent incrementing all components of its bid by one unit. If two or more enabled agents wish to invoke *raise* in a round, then the auction chooses from amongst them arbitrarily. The auction terminates when no agent invokes *raise* in a round.

We now develop formalism leading to the definition of the function *raise*. For any ECC X and any agent u in *unmatched*(X), we define the predicate $P_0(X, u)$ to hold if X is either quiescent or u-quiescent. We now define *victim*(X, u, z) for any ECC X, any integer z in $\{0, 1\}$, and any agent u in *unmatched*(X) such that the predicate $P_0(X, u)$ holds. Let set U_0 denote *white*(X) and let set U_1 denote *agents*$(X, u) \cup \{u\} \cap$ *nonpositive*(X). Note that set U_1 is nonempty as it contains agent u. If $U_1 \setminus U_0 \neq \emptyset$, we define *victim*$(X, u, z)$ as the minimum agent in $U_1 \setminus U_0$. If $U_1 \setminus U_0 = \emptyset$, $z = 1$, and $U_1 - u \neq \emptyset$, then we define *victim*(X, u, z) as the minimum agent in $U_1 - u$. Otherwise, we define *victim*(X, u, z) as the minimum agent in U_1.

For any ECC X and any agent u in *enabled*(X), we define the predicate $P_1(X, u)$ to hold if either (1) agent u belongs to *matched*(X) and X is quiescent, or (2) agent u belongs to *unmatched*(X) and the predicate $P_0(X, u)$ holds. We now define $\text{augment}(X, u, z)$ for any ECC X, any integer z in $\{0, 1\}$, and any agent u in *enabled*(X) such that the predicate $P_1(X, u)$ holds. If agent u belongs to *matched*(X), then $\text{augment}(X, u, z)$ is the ECC X. Otherwise, $\text{augment}(X, u, z)$ is the ECC $[\chi']$, where χ' is constructed as follows: Let χ be an arbitrary configuration in X and let P be an arbitrary simple directed path from u to *victim*(X, u, z) in *digraph*(χ); for every item v' such that there exists an arc of the form (u', v') on path P, we set $match(\chi', v') = u'$, and for every item v' that is not on path P, we set $match(\chi', v') = match(\chi, v')$. By Lemma 4.1, it follows that $\text{augment}(X, u, z)$ is well defined.

For any ECC X and any agent u in *enabled*(X) such that either (1) X is quiescent and *agents*$(X, u) \cap$ *nonpositive*$(X) = \emptyset$, or (2) X is u-quiescent and u belongs to *matched*(X), we define $\text{inc}(X, u)$ as the $\cup_{(G,M,\Phi) \in X}[(G', M, \Phi')]$ where $G' = shift(bid\text{-}graph(X), u, 1)$, and Φ' is defined as follows: if agent u belongs to *matched*(χ), then $\Phi' = \Phi$; otherwise $\Phi'(v) = \Phi(v) + 1$ for any item v in *items*(χ, u) and $\Phi'(v) = \Phi(v)$ for any item v in *items*$(\chi) \setminus$ *items*(χ, u). The existence of such an ECC is established by Lemma 4.2.

For any quiescent ECC X and any agent u in *enabled*(X), we define *raise*$'(X, u)$ as $\text{augment}(X, u, 1)$. For any ECC X and any agent u in *enabled*(X) such that either X is quiescent, or X is u-quiescent and u belongs to *matched*(X), we define *raise*$''(X, u)$ as augment$(\text{inc}(X, u), u, 0)$ For any quiescent ECC X and any agent u in *enabled*(X), the function *raise*(X, u) is defined as *raise*$''($*raise*$'(X, u), u)$.

For any quiescent ECC X and any agent u in *unmatched*(X), we define *victim*(X, u) as follows: if *matched*$(X) \cap$ *unmatched*$($*raise*$(X, u)) = \{u'\}$, then *victim*$(X, u) = u'$; otherwise, *victim*$(X, u) = \emptyset$. Recall that by Fact 4.4, *matched*$(X) \cap$ *unmatched*(X) has a cardinality of at most 1.

The facts below follow from the definition of the function *raise*.

Fact 4.1. *For any quiescent ECC X and any agent u in* $\text{enabled}(X) \cap \text{matched}(X)$*, we have* $\text{raise}(X, u) = \text{shift}(X, u, 1)$*.*

Fact 4.2. *For any quiescent ECC X and any agent u in* $\text{enabled}(X)$*, we have* $\text{potential}(\text{raise}(X, u)) \geq \text{potential}(X)$*.*

Fact 4.3. *For any quiescent ECC X and any agent u in* $\text{enabled}(X)$ *such that* $\text{bid}(\text{bid-graph}(X), u) < \text{potential}(X)$*, we have* $\text{potential}(\text{raise}(X, u)) = \text{potential}(X)$*.*

Fact 4.4. *For any ECC X' of the form* $\text{raise}(X, u)$*, we have* $|S| \leq 1$*, where* $S = \text{matched}(X) \setminus \text{matched}(X')$*.*

The following lemmas establish that the output of the bottom-level auction is a quiescent ECC.

Lemma 4.1. *For any ECC X, any integer z in $\{0, 1\}$, and any agent u in* $\text{enabled}(X)$ *such that the predicate $P_1(X, u)$ holds, there is a unique ECC of the form augment(X, u, z). augment(X, u, z).*

Proof. If u belongs to *matched*(X), then by definition $\text{augment}(X, u, z) = X$. We now consider the case where u belongs to *unmatched*(X). Let χ be any configuration in X. By definition, irrespective of the choice of χ and the path P used, the agent *victim*(χ, u, z) is unmatched in $\text{augment}(X, u, z)$ and each agent in *matched*$(\chi) \cap$ *nonwhite*$(\chi) \setminus$ *victim*(χ, u, z) is matched to the same item in χ and $\text{augment}(X, u, z)$. Thus it follows that $\text{augment}(X, u, z)$ is an ECC. □

Lemma 4.2. *Any set of configurations of the form inc(X, u) is an ECC.*

Proof. If u belongs to *matched*(X), then by definition $\text{inc}(X, u) =$ *shift*$(X, u, 1)$. We now consider the case where u belongs to *unmatched*(X). By the preconditions on X required by $\text{inc}(X, u)$, it follows that X is quiescent and there exists no agent u' in *agents*(X, u) such that *gap*$(X, u') \leq 0$. Let (G, M, Φ) be any configuration in X. By definition, $\text{inc}(X, u)$ includes the configuration (G', M, Φ') where $G' =$ *shift*$($*bid-graph*$(X), u, 1)$ and $\Phi'(v) = \Phi(v) + 1$ for each item v in *items*(χ, u). Thus, every agent in *nonwhite*(χ) is matched to the same item in χ and χ'. It follows that the set of configurations given by $\text{inc}(X, u)$ is an ECC. □

Lemma 4.3. *For any quiescent ECC X and any agent u in* $\text{enabled}(X)$*, the predicate $P_1(X, u)$ holds.*

Proof. Since X is quiescent, by definition, $P_1(X, u)$ holds if u belongs to *matched*(X). Suppose u belongs to *unmatched*(X). Then, $P_1(X, u)$ holds if $P_0(X, u)$ holds. By definition, $P_0(X, u)$ holds when X is quiescent. □

Lemma 4.4. *For any ECC X' of the form inc(X, u), the predicate $P_1(X', u)$ holds.*

Proof. By the preconditions of $\text{inc}(X, u)$, we know that either *agents*$(X, u) \cap$ *nonpositive*$(X) = \emptyset$ and X is quiescent, or u belongs to *matched*(X) and *shift*$(X, u, 1)$ is quiescent. We first consider the case when u belongs to *matched*(X). In this case, $\text{inc}(X, u) =$ *shift*$(X, u, 1)$. Thus $\text{inc}(X, u)$ is quiescent and $P_1(X', u)$ holds. Next we consider the case when u belongs to *unmatched*(X). In this case, $\mathit{agents}(X,u) \cap \mathit{nonpositive}(X) = \emptyset$ and X is quiescent. By definition, $\mathrm{inc}(X,u)$ is an ECC X' whose bid-graph $G = \mathit{shift}(\mathit{bid\text{-}graph}(X), u, 1)$ and whose potential function has incremented $\mathit{potential}(X,v)$ by one for each item v in $\mathit{items}(X,u)$. It is easy to see that u is either white or gray in $\mathrm{inc}(X,u)$; thus $\mathrm{inc}(X,u)$ is either quiescent or u-quiescent. It follows that $P_1(\mathrm{inc}(X,u),u)$ holds □

Lemma 4.5. *For any quiescent ECC X and any agent u in* $\mathrm{enabled}(X)$*, either* $\mathrm{raise}'(X,u)$ *is quiescent, or* $\mathrm{raise}'(X,u)$ *is u-quiescent and u belongs to* $\mathrm{matched}(\mathrm{raise}'(X,u))$*.*

Proof. We first consider the case where u belongs to $\mathit{matched}(X)$. In this case $\mathit{raise}'(X,u) = X$ and thus $\mathit{raise}'(X,u)$ is quiescent.

Next, we consider the case where u belongs to $\mathit{unmatched}(X)$. Since X is quiescent, u belongs to $\mathit{white}(X)$. In this case, $\mathit{raise}'(X,u) = \mathrm{augment}(X,u,1)$. Thus, either u is unmatched and u is white in $\mathit{raise}'(X,u)$, or u belongs to $\mathit{matched}(\mathit{raise}'(X,u))$ and u is gray in $\mathit{raise}'(X,u)$. Thus, $\mathit{raise}'(X,u)$ is either quiescent or u-quiescent. □

Lemma 4.6. *Any ECC of the form* $\mathrm{raise}(X,u)$ *is quiescent.*

Proof. By definition $\mathit{raise}(X,u) = \mathit{raise}''(\mathit{raise}'(X,u),u)$. By Lemma 4.5, $\mathit{raise}'(X,u)$ satisfies the preconditions of raise''. We consider the following two cases. First, we consider the case where u belongs to $\mathit{matched}(\mathit{raise}'(X,u))$, In this case, by Lemma 4.5, $\mathit{raise}'(X,u)$ is u-quiescent. Further, by definition, raise'' for a matched agent results in incrementing the bid of the agent by one unit; thus, $\mathit{raise}(X,u) = \mathit{shift}(\mathit{raise}'(X,u),u,1)$ which is quiescent by definition.

Next we consider the case where u belongs to $\mathit{unmatched}(\mathit{raise}'(X,u))$. In this case, by Lemma 4.5, $\mathit{raise}'(X,u)$ is quiescent and thus $\mathit{raise}'(X,u)$ satisfies the precondition for invoking inc. By Lemma 4.4, the predicate $P_1(\mathrm{inc}(\mathit{raise}'(X,u),u),u)$ holds; thus, either u belongs to $\mathit{matched}(\mathrm{inc}(\mathit{raise}'(X,u),u))$, or $\mathrm{inc}(\mathit{raise}'(X,u),u)$ is either quiescent of u-quiescent. In the case where $\mathrm{inc}(\mathit{raise}'(X,u),u)$ is u-quiescent, it is easy to see from the definition of *augment* that $\mathit{raise}(X,u)$ is quiescent. For the remaining two cases, *augment* is a no-op. □

4.2 Basic Properties

Below we discuss some basic properties of the bottom-level auction that are useful in both proving lemmas of Section 4.5 and establishing properties of the top-level auction of Section 5.

Lemma 4.7. *For any ECC X' of the form* $\mathrm{raise}(X,u')$ *and any agent u in* $\mathrm{nonwhite}(X)$*, either* (1) *u belongs to* $\mathrm{unmatched}(X')$*, or* (2) *u belongs to* $\mathrm{nonwhite}(X')$*, and there exists an item v in* $\mathrm{items}(X)$ *such that* $\mathrm{potential}(X,v) = \mathrm{potential}(X',v)$ *and* $\mathrm{match}(\chi,v) = u$ *for any configuration χ in $X \cup X'$.*

Proof. Since u belongs to $\mathit{nonwhite}(X)$, there exists an item v in $\mathit{items}(X)$ such that for any configuration χ in X, we have $\mathit{match}(\chi,v) = u$. By definition, u does not belong to $\mathit{digraph}(X)$ and v is a leaf of $\mathit{digraph}(X)$. Since v is a leaf of $\mathit{digraph}(X)$, by the definition of the function raise' either implies that $u = \mathit{victim}(X,u',1)$ or $\mathit{match}(\chi,v) = u$ for any configuration χ in $X \cup \mathit{raise}'(X,u')$. If $u = \mathit{victim}(X,u',1)$, then u belongs to $\mathit{unmatched}(X')$ and the proof is complete.

We now consider the case where $u \neq \mathit{victim}(X,u',1)$; thus v does not belong to $\mathit{items}(X,u')$. By the definition of the function raise'', $\mathit{potential}(X',v') = \mathit{potential}(X,v') + 1$ for any item v' in $\mathit{items}(X,u')$ and $\mathit{potential}(X',v') = \mathit{potential}(X,v')$ for any item v' not in $\mathit{items}(X,u')$; thus $\mathit{potential}(X',v) = \mathit{potential}(X,v)$. Let $X'' = \mathrm{inc}(\mathit{raise}'(X,u'),u')$. It is easy to see that v is a leaf of $\mathit{digraph}(X'')$. Thus, either $u = \mathit{victim}(X'',u',0)$ or $\mathit{match}(\chi,v) = u$ for any configuration χ in $X \cup X'$. □

Lemma 4.8. *For any quiescent ECC X and any agent u in* $\mathrm{enabled}(X)$*, if* $X' = \mathrm{raise}(X, u)$*, then*

$$\mathrm{gray}(X) \subseteq \mathrm{nonblack}(X') \wedge \mathrm{white}(X) \subseteq \mathrm{white}(X').$$

Proof. By the definition of the function *raise*, if u belongs to $\mathit{gray}(X)$, then u belongs to $\mathit{gray}(X')$, and if u belongs to $\mathit{white}(X)$, then u belongs to $\mathit{white}(X')$. Consider any agent u_0 in $\mathit{agents}(X) - u$. By Lemma 4.6, X' is quiescent, and by the definition of a quiescent ECC, $\mathit{unmatched}(X') \subseteq \mathit{white}(X')$. Thus, if u_0 belongs to $\mathit{unmatched}(X')$, then u_0 belongs to $\mathit{white}(X')$ and hence u_0 belongs to $\mathit{enabled}(X')$. Now suppose that u_0 belongs to $\mathit{matched}(X')$. We consider the following two cases.

First we consider the case where u_0 belongs to $\mathit{gray}(X) \cap \mathit{matched}(X')$. By Fact 4.2, $\mathit{potential}(X') \geq \mathit{potential}(X)$ and by Lemma 4.7, it follows that there exists an item v_0 in $\mathit{items}(X)$ having $\mathit{potential}(X, v_0) = \mathit{potential}(X', v_0)$ and for any configuration χ in $X \cup X'$, we have $\mathit{match}(\chi, v_0) = u_0$. It follows that u_0 belongs to $\mathit{gray}(X')$.

Next we consider the case where u_0 belongs to $\mathit{white}(X) \cap \mathit{matched}(X')$. By our assumption, u_0 belongs to $\mathit{matched}(X)$. By the definition of *raise*, it follows that $\mathit{gap}(X', u_0) \geq 0$. Thus, u_0 belongs to $\mathit{white}(X')$. □

Lemma 4.9. *For any quiescent ECC X and any agent u in* $\mathrm{enabled}(X)$*, we have* $\mathrm{enabled}(X) - u \subseteq \mathrm{enabled}(\mathrm{raise}(X, u))$*.*

Proof. Let $X' = \mathit{raise}(X, u)$. By Lemma 4.6, X' is quiescent. Consider any agent u_0 in $\mathit{enabled}(X) - u$. Suppose u_0 belongs to $\mathit{white}(X)$; then by Lemma 4.8, u_0 belongs to $\mathit{white}(X')$, and hence u_0 belongs to $\mathit{enabled}(X')$.

Suppose u_0 belongs to $\mathit{nonwhite}(X)$. Since u_0 belongs to $\mathit{enabled}(X)$, we have $\beta(v) < \mathit{potential}(X, v)$ for every item v in $\mathit{items}(X)$, where $\beta = \mathit{bid}(X, u_0)$. By Fact 4.2, $\mathit{potential}(X') \geq \mathit{potential}(X)$ and by Lemma 4.7, either u_0 belongs to $\mathit{unmatched}(X')$ or there exists an item v_0 in $\mathit{items}(X)$ such that for any configuration χ in $X \cup X'$, we have $\mathit{match}(\chi, v_0) = u_0$. Thus, u_0 belongs to $\mathit{enabled}(X')$. □

Lemma 4.10. *For any quiescent ECC X and any agent u in* $\mathrm{enabled}(X)$*, if there exists an item v in* $\mathrm{items}(X)$ *such that* $\mathrm{potential}(X, v) = \mathrm{potential}(\mathrm{raise}(X, u), v)$*, then* $\mathrm{bids}(X, v) \subseteq \mathrm{bids}(\mathrm{raise}(X, u), v)$*.*

Proof. Let β be any bid in in $\mathit{bids}(X, v)$. By definition, for any item v' in $\mathit{items}(X) - v$, we have $\beta(v) - \mathit{potential}(X, v) \geq \beta(v') - \mathit{potential}(X, v')$. By Lemma 4.2, we have $\mathit{potential}(\mathit{raise}(X, u)) \geq \mathit{potential}(X)$. Thus, for any item v' in $\mathit{items}(X) - v$, we have $\beta(v) - \mathit{potential}(X, v) \geq \beta(v') - \mathit{potential}(\mathit{raise}(X, u), v')$. Thus, β belongs to $\mathit{bids}(\mathit{raise}(X, u), v)$. □

Lemma 4.11. *For any quiescent ECC X_0 and any quiescent ECC X_1 of the form* $\mathrm{subst}(X_0, u_0, u_1)$ *where u_0 belongs to* $\mathrm{unmatched}(X_0)$ *and $u_1 < u_0$, we have* $\mathrm{gap}(\mathrm{raise}(X_0, u_0), u_0) = \mathrm{gap}(\mathrm{raise}(X_1, u_1), u_1) = 0$*. Furthermore, either* (1) $\mathrm{raise}(X_1, u_1) = \mathrm{subst}(\mathrm{raise}(X_0, u_0), u_0, u_1)$*, or* (2) $\mathrm{raise}(\mathrm{raise}(X_1, u_1), u_1) = \mathrm{subst}(\mathrm{raise}(\mathrm{raise}(X_0, u_0), u_0), u_0, u_1)$*.*

Proof. Let $\beta = \mathit{bid}(\mathit{bid\text{-}graph}(X_0), u_0)$. Let $X_0' = \mathit{raise}'(X_0, u_0)$ and let $X_0'' = \mathit{raise}''(X_0', u_0)$. Let $X_1' = \mathit{raise}'(X_1, u_1)$ and let $X_1'' = \mathit{raise}''(X_1', u_1)$. Note that $\mathit{items}(X_0, u_0) = \mathit{items}(X_1, u_1)$. Thus, by the definition of the function raise' it follows that $X_1' = \mathit{subst}(X_0', u_0, u_1)$. If u_0 belongs to $\mathit{matched}(X_0')$, then it is easy to see that $X_1'' = \mathit{subst}(X_0'', u_0, u_1)$ and the proof is complete. We now consider the case where u_0 belongs to $\mathit{unmatched}(X_0')$. Note that $\mathit{items}(X_0', u_0) = \mathit{items}(X_1', u_1)$. Since $u_1 < u_0$, it follows from the definition of the function raise'' that if u_1 belongs to $\mathit{matched}(X_1'')$, then u_0 belongs to $\mathit{matched}(X_0'')$. Similarly, if u_0 belongs to $\mathit{unmatched}(X_0'')$, then u_1 belongs to *unmatched*(X_1''). Thus, either $X_1'' = \textit{subst}(X_0'', u_0, u_1)$, or u_0 belongs to *matched*(X_0'') and u_1 belongs to *unmatched*(X_1''). Thus, there exists an item v in $\textit{items}(\text{inc}(X_1', u_1), u_1)$ such that $\textit{match}(\text{inc}(X_1', u_1), v)$ belongs to $\textit{zero}(\text{inc}(X_1', u_1))$ and $u_1 < u' < u_0$. It is easy to see from the definition of the function *raise*′ that $\textit{raise}'(X_0'', u_0) = X_0''$ and $\textit{raise}'(X_1'', u_1) = \textit{subst}(X_0'', u_0, u_1)$. Thus, $\textit{raise}(X_1'', u_0) = \textit{subst}(\textit{raise}(X_0'', u_0), u_0, u_1)$. □

4.3 Commutativity of *raise* invocations

A key property of the bottom-level auction is the commutativity of *raise* invocations. This property is formalized in Lemma 4.20 and is used extensively in the following sections of the paper.

Lemma 4.12. *For any quiescent ECC X, any agents u_0 and u_1 in* $\text{unmatched}(X)$ *such that* $\text{agents}(X, u_0) \cap \text{nonpositive}(X) = \emptyset$, *and any item v in* $\text{items}(X, u_0)$, *we find that v belongs to* $\text{items}(\text{raise}(X, u_1), u_0)$ *if and only if* $\text{potential}(\text{raise}(X, u_1), v) = \text{potential}(X, v)$.

Proof. Since $\textit{agents}(X, u_0) \cap \textit{nonpositive}(X) = \emptyset$, it follows that $\textit{victim}(X, u_1, 1)$ does not belong to $\textit{agents}(X, u_0)$, thus $\textit{agents}(X, u_0) = \textit{agents}(\textit{raise}'(X, u_1), u_0)$ and $\textit{items}(X, u_0) = \textit{items}(\textit{raise}'(X, u_1), u_0)$. Let $\chi = (G, M, \Phi)$ be any configuration in X and let $\chi' = (G', M', \Phi')$ be any configuration in $\textit{raise}(X, u_1)$. By Lemma 3.21, $\textit{items}(\chi, u_0) = \textit{items}(X, u_0)$ and $\textit{items}(\chi', u_0) = \textit{items}(\textit{raise}(X, u_1), u_0)$. By definition, v belongs to $\textit{items}(\chi, u_0)$ if and only if there exists a directed path from u_0 to v in $\textit{digraph}(\chi)$, where every edge of the form (u', v') in $\textit{digraph}(\chi)$ is such that v' belongs to $\textit{demand}(\chi, \textit{bid}(\textit{bid-graph}(X), u'))$.

It is easy to see that if $\textit{potential}(\textit{raise}(X, u_1), v) > \textit{potential}(X, v)$, then there is no directed path from u_0 to v in $\textit{digraph}(\chi)$. We now consider the case where $\textit{potential}(\textit{raise}(X, u_1), v) = \textit{potential}(X, v)$. It follows from the definition of the *raise* function that if $\Phi'(v') > \Phi(v')$ for some item v' on a directed path from u_0 to v, then $\Phi'(v) > \Phi(v)$, and this would contradict our assumption that $\textit{potential}(\textit{raise}(X, u_1), v) = \textit{potential}(X, v)$. Thus every item v' on every directed path from u_0 to v has $\Phi'(v') = \Phi(v')$; it follows that all such directed paths are preserved in $\textit{digraph}(\chi')$, and thus, v belongs to $\textit{items}(\textit{raise}(X, u_1), u_0)$. □

Lemma 4.13. *For any quiescent ECC X and any agents u_0 in* $\text{unmatched}(X)$ *and u_1 in* $\text{enabled}(X)$, *if* $\text{agents}(X, u_0) \cap \text{nonpositive}(X) = \emptyset$, *then* $\text{agents}(X_1, u_0) \subseteq \text{agents}(X, u_0)$, *and* $\text{agents}(X_1, u_0) \cap \text{nonpositive}(X_1) = \emptyset$, *where* $X_1 = \text{raise}(X, u_1)$.

Proof. If u_1 belongs to *matched*(X), then by Fact 4.1, we have $X_1 = \textit{shift}(X, u_1, 1)$; in this case it is easy to see that $\textit{agents}(X_1, u_0) \subseteq \textit{agents}(X, u_0)$ and $\textit{agents}(X_1, u_0) \cap \textit{nonpositive}(X_1) = \emptyset$.

We now consider the case where u_1 belongs to *unmatched*(X). By Lemma 4.12, we have $\textit{items}(X_1, u_0) \subseteq \textit{items}(X, u_0)$ and $\textit{potential}(X_1, v) = \textit{potential}(X, v)$ for any item v in $\textit{items}(X_1, u_0)$. Thus, we have $\textit{agents}(X_1, u_0) \subseteq \textit{agents}(X, u_0)$ and $\textit{agents}(X_1, u_0) \cap \textit{nonpositive}(X_1) = \emptyset$. □

Lemma 4.14. *Let X_0 and X_1 be quiescent ECCs such that* $\text{bid-graph}(X_0) = \text{bid-graph}(X_1)$, $\text{potential}(X_0) = \text{potential}(X_1)$ *and for any agent u in* $\text{nonwhite}(X) \cap \text{matched}(X)$, *there exists an item v such that* $\text{match}(\chi, v) = u$ *for any configuration χ in $X_0 \cup X_1$. For any agents u_0 and u_1 such that* $\text{matched}(X_0) \setminus \text{matched}(X_1) = \{u_1\}$ *and* $\text{matched}(X_1) \setminus \text{matched}(X_0) = \{u_0\}$, *if u_1 belongs to* $\text{agents}(X_0, u_0)$ *and u_0 belongs to* $\text{agents}(X_1, u_1)$, *either* $\text{victim}(X_0, u_0, 1) = \text{victim}(X_1, u_1, 1)$ *or* $\text{victim}(X_0, u_0, 0) = \text{victim}(X_1, u_1, 0)$.

Proof. Let $U = \textit{matched}(X_0) - u_1 = \textit{matched}(X_1) - u_0$. We have $\textit{potential}(X_0) = \textit{potential}(X_1)$ and for any agent u in $\textit{nonwhite}(X) \cap \textit{matched}(X)$, there exists an item v such that $\textit{match}(\chi, v) = u$ for any configuration χ in $X_0 \cup X_1$; thus we have $\textit{nonpositive}(X_0) = \textit{nonpositive}(X_1)$ and for any agent

u in U, we have $agents(X_0, u) = agents(X_1, u)$ and $items(X_0, u) = items(X_1, u)$. Additionally, since u_1 belongs to $agents(X_0, u_0)$ and u_0 belongs to $agents(X_1, u_1)$, we have $nonpositive(X_0) \cap agents(X_0, u_0) = nonpositive(X_1) \cap agents(X_1, u_1)$. By the definition of the function *victim*, it is easy to see that either $victim(X_0, u_0, 1) = victim(X_1, u_1, 1)$ or $victim(X_0, u_0, 0) = victim(X_1, u_1, 0)$. □

Lemma 4.15. *For any quiescent ECC X and any agents u_0 and u_1 in* $\text{unmatched}(X)$*, if* $\text{victim}(X, u_0) = \text{victim}(X, u_1, 1)$*, then u_0 belongs to* $\text{agents}(\text{raise}(X, u_0), u_1)$*.*

Proof. Let $X_0 = raise(X, u_0)$ and let $victim(X, u_0) = victim(X, u_1, 1) = u$. Since $u = victim(X, u_1, 1)$, we have u_1 belongs to $nonpositive(X)$ and $V \subseteq items(X, u_1)$ where $V = demand(X, u)$.

Suppose $V \cap items(X_0, u_1) = \emptyset$. Then, by Lemma 4.12, we have $potential(X_0, v) = potential(X, v) + 1$ for every item in V and $V \subseteq items(X, u_0)$; thus u belongs to $agents(X, u_0)$ and by the definition of the function $raise'$, $potential(X_0) = potential(X)$, which is a contradiction. Thus, we have $V \cap items(X_0, u_1) \neq \emptyset$. Additionally, since $u = victim(X, u_0)$, we have u_0 belongs to $agents(X_0, u')$ for any agent u' having $V \cap items(X_0, u') = \emptyset$. Thus, u_0 belongs to $agents(raise(X, u_0), u_1)$. □

Lemma 4.16. *For any quiescent ECC X and any agents u_0 and u_1 in* $\text{unmatched}(X)$ *such that* $u_1 = \text{victim}(X, u_1, 1)$*, if* $\text{victim}(X, u_0) = \text{victim}(X'_1, u_1, 0)$ *where* $X'_1 = inc(\text{raise}'(X, u_1), u_1)$*, then agent u_0 belongs to* $\text{agents}(X'_{01}, u_1)$ *where* $X'_{01} = inc(\text{raise}'(\text{raise}(X, u_0), u_1), u_1)$*.*

Proof. Let $X_0 = raise(X, u_0)$ and let $victim(X, u_0) = victim(X'_1, u_1, 0) = u$. Note that the case where $victim(X, u_0, 1) = u$ is symmetric to the case handled by Lemma 4.15; thus the proof of this case follows from Lemma 4.15.

We now focus on the case where $victim(X, u_0, 1) = u_0$. By the definition of the function $raise''$, we have $potential(X_0, v) = potential(X, v) + 1$ for any item v in $items(X, u_0)$ and $potential(X, v) = potential(X, v)$ for any item v in $items(X) \setminus items(X, u_0)$. Since $victim(X, u_0) = u$, we have u_0 belongs to $agents(X_0, u')$ for any agent u' such that $demand(X_0, u) \cap items(X_0, u') \neq \emptyset$. Since $victim(X, u_1, 1) = u_1$, we have $nonpositive(X) \cap agents(X, u_1) = \emptyset$; thus by Lemmas 4.12 and 4.13, we have $potential(X'_{01}, v) = potential(X, v) + 1$ for any item v in $items(X, u_0) \cup items(X, u_1)$. Since $victim(X'_1, u_1, 0) = u$, we have $V \cap items(X'_{01}, u_1) \neq \emptyset$. It follows that u_0 belongs to $agents(X'_{01}, u_1)$. □

Lemma 4.17. *Let X be a quiescent ECC and let u_0 and u_1 be agents in* $\text{unmatched}(X)$*. Let* $X_0 = \text{raise}(X, u_0)$ *and let* $X_1 = \text{raise}(X, u_1)$*. If* $\text{victim}(X, u_0) \neq \text{victim}(X, u_1)$*, then we have* $\text{victim}(X_0, u_1) = \text{victim}(X, u_1)$ *and* $\text{potential}(\text{raise}(X_0, u_1), v) = \text{potential}(X_1, v)$ *for any item v in* $\text{items}(X)$ *such that* $\text{potential}(X_1, v) = \text{potential}(X, v) + 1$*.*

Proof. First we consider the case where $victim(X, u_1, 1) \neq u_1$. In this case, we have $victim(X, u_1) = victim(X, u_1, 1)$ and $potential(X_1) = potential(X)$; thus $victim(X, u_1)$ belongs to $nonpositive(X)$. The statement of the lemma assumes that $victim(X, u_0) \neq victim(X, u_1)$; thus, by the definition of the function *raise*, we find that $victim(X, u_1)$ belongs to $agents(X_0, u_1) \cap nonpositive(X_0)$. If $victim(X_0, u_1) = victim(X, u_1)$, then the proof is complete. Suppose that $victim(X_0, u_1) \neq victim(X, u_1)$. Then there is an agent u' in $agents(X, u_1)$ such that $u' = victim(X, u_0)$, and hence u_0 belongs to $agents(X_0, u_1)$. Since $u' = victim(X_0, u_0)$, u' belongs to $agents(X, u_1)$, and $u' \neq victim(X, u_1, 1)$, the definition of the function *victim* implies that $victim(X_0, u_1, 1) = victim(X, u_1, 1)$.

Next we consider the case where $victim(X, u_1, 1) = u_1$; thus, by the definition of the function *raise*, we have $nonpositive(X) \cap agents(X, u_1) = \emptyset$ and $potential(X_1, v) = potential(X, v) + 1$ for any item v in $items(X, u_1)$. Thus, $victim(X, u_1) = victim(X'_1, u_1, 0)$, where $X'_1 = \text{inc}(raise'(X, u_1), u_1)$.

By Lemma 4.12, we have $\mathit{items}(X_0, u_1) = \mathit{items}(X, u_1) \setminus \mathit{items}(X, u_0)$ and $\mathit{potential}(X_0, v) = \mathit{potential}(X, v)+1$ for any item v in $\mathit{items}(X, u_0)$. By Lemmas 4.12 and 4.13, we have $\mathit{nonpositive}(X_0) \cap \mathit{agents}(X_0, u_1) = \emptyset$; thus, we have $\mathit{potential}(\mathit{raise}(X_0, u_1), v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X_0, u_1)$. Since $\mathit{items}(X_0, u_1) = \mathit{items}(X, u_1) \setminus \mathit{items}(X, u_0)$, we have $\mathit{potential}(\mathit{raise}(X_1, u_1), v) = \mathit{potential}(X, v)+1$ for any item v in $\mathit{items}(X, u_0) \cup \mathit{items}(X, u_1)$. Let $X'_{01} = \mathrm{inc}(\mathit{raise}'(X_0, u_1), u_1)$. If $\mathit{victim}(X'_{01}, u_1, 0) = \mathit{victim}(X'_1, u_1, 0)$, then the proof is complete. Suppose that $\mathit{victim}(X'_{01}, u_1, 0) \neq \mathit{victim}(X'_1, u_1, 0)$; then there is an agent u' in $\mathit{agents}(X'_1, u_1)$ such that $u' = \mathit{victim}(X, u_0)$, and hence u_0 belongs to $\mathit{agents}(X'_{01}, u_1)$. Since $u' = \mathit{victim}(X_0, u_0)$, u' belongs to $\mathit{agents}(X'_1, u_1)$, and $u' \neq \mathit{victim}(X'_1, u_1, 0)$. The definition of the function *victim* implies that $\mathit{victim}(X'_{01}, u_1, 0) = \mathit{victim}(X'_1, u_1, 0)$.

Thus, $\mathit{victim}(X_0, u_1) = \mathit{victim}(X, u_1)$ and $\mathit{potential}(\mathit{raise}(X_0, u_1), v) = \mathit{potential}(X_1, v)$ for any item v in $\mathit{items}(X)$ such that $\mathit{potential}(X_1, v) = \mathit{potential}(X, v) + 1$.

□

Lemma 4.18. *For any quiescent ECC X and any agents u_0 and u_1 in* $\mathrm{agents}(X)$*, if* $\mathrm{matched}(X) \cap \{u_0, u_1\} \neq \emptyset$*, then* $\mathrm{potential}(X_{01}) = \mathrm{potential}(X_{10})$ *and* $\mathrm{matched}(X_{01}) = \mathrm{matched}(X_{10})$*, where* $X_{01} = \mathrm{raise}(\mathrm{raise}(X, u_0), u_1)$ *and* $X_{10} = \mathrm{raise}(\mathrm{raise}(X, u_1), u_0)$*.*

Proof. Let $X_0 = \mathit{raise}(X, u_0)$ and let $X_1 = \mathit{raise}(X, u_1)$.

We first consider the case where $|\{u_0, u_1\} \cap \mathit{matched}(X)| = 2$; thus, $\{u_0, u_1\} \subseteq \mathit{matched}(X)$. By Fact 4.1, we have $X_{01} = \mathit{shift}(\mathit{shift}(X, u_0, 1), u_1, 1)$, and $X_{10} = \mathit{shift}(\mathit{shift}(X, u_1, 1), u_0, 1)$; thus, $X_{01} = X_{10}$.

We now focus on the case where $|\{u_0, u_1\} \cap \mathit{matched}(X)| = 1$. Without loss of generality, we assume that $\{u_0, u_1\} \cap \mathit{matched}(X) = \{u_1\}$; thus, u_1 belongs to $\mathit{matched}(X)$. Since u_1 belongs to $\mathit{enabled}(X) \cap \mathit{matched}(X)$ and $X_1 = \mathit{shift}(X, u_1, 1)$, either u_1 belongs to $\mathit{nonwhite}(X) \cap \mathit{nonwhite}(X_1)$ or u_1 belongs to $\mathit{white}(X) \cap \mathit{white}(X_1)$. If u_1 belongs to $\mathit{nonwhite}(X) \cap \mathit{nonwhite}(X_1)$, then for every item v in $\mathit{items}(X)$, we have $\beta(v) < \mathit{potential}(X, v) - 2$, where $\beta = \mathit{bid}(\mathit{bid\text{-}graph}(X), u_1)$. Thus we have $\mathit{victim}(X, u_0) = \mathit{victim}(X_1, u_0) = u_1$ and $\mathit{raise}(X_0, u_1) = X_0$. Using these facts, it is straightforward to argue that $\mathit{potential}(X_{01}) = \mathit{potential}(X_{10})$ and $\mathit{matched}(X_{01}) = \mathit{matched}(X_{10})$. It remains to address the case where u belongs to $\mathit{white}(X) \cap \mathit{white}(X_1)$. We proceed via the following case analysis.

- Case 1: $\mathit{victim}(X, u_0) \neq u_1$.
  - Case 1.1: $\mathit{victim}(X, u_0, 1) \neq u_0$.
    We have $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_0, 1)$. In this case, u_0 belongs to $\mathit{matched}(\mathit{raise}'(X, u_0))$; thus, by the definition of the function *raise*, we have $\mathit{matched}(X_0) = \mathit{matched}(X) + u_0 - \mathit{victim}(X, u_0)$ and $\mathit{potential}(X_0) = \mathit{potential}(X)$. By Fact 4.1, we have $X_{01} = \mathit{shift}(X_0, u_1, 1)$; thus $\mathit{potential}(X_{01}) = \mathit{potential}(X)$ and $\mathit{matched}(X_{01}) = \mathit{matched}(X) + u_0 - \mathit{victim}(X, u_0)$.
    Since $\mathit{victim}(X, u_0) \neq u_1$, there exists an agent u' in $\mathit{nonwhite}(X) \cap \mathit{agents}(X, u_0)$ such that $\mathit{victim}(X, u_1, 1) = u'$. By Fact 4.1, we have $X_1 = \mathit{shift}(X, u_1, 1)$ and thus, $\mathit{nonwhite}(X) \cap \mathit{agents}(X, u_0) - u_1 = \mathit{nonwhite}(X_1) \cap \mathit{agents}(X_1, u_0) - u_1$; it follows that $\mathit{victim}(X_1, u_0) = \mathit{victim}(X, u_0)$. Thus, $\mathit{matched}(X_{10}) = \mathit{matched}(X) + u_0 - \mathit{victim}(X, u_0)$. Since $X_1 = \mathit{shift}(X, u, 1)$ and u_0 belongs to $\mathit{matched}(\mathit{raise}'(X, u_0))$, we have $\mathit{potential}(X_{10}) = \mathit{potential}(X)$.
    Thus, $\mathit{matched}(X_{01}) = \mathit{matched}(X_{10})$ and $\mathit{potential}(X_{01}) = \mathit{potential}(X_{10})$.
  - Case 1.2: $\mathit{victim}(X, u_0, 1) = u_0$.

In this case, $\mathit{victim}(X, u_0) = \mathit{victim}(X_0', u_0, 0)$ where $X_0' = \mathrm{inc}(\mathit{raise}'(X, u_0), u_0)$. Since $\mathit{victim}(X, u_0, 1) = u_0$, we have $\mathit{nonwhite}(X) \cap \mathit{agents}(X, u_0) = \emptyset$; thus, $\mathit{potential}(X_0, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0)$. By Fact 4.1, we have $X_{01} = \mathit{shift}(X_0, u_1, 1)$; thus $\mathit{potential}(X_{01}) = \mathit{potential}(X_0)$ and $\mathit{matched}(X_{01}) = \mathit{matched}(X) + u_0 - \mathit{victim}(X, u_0)$. We established above that $X_1 = \mathit{shift}(X, u_1, 1)$; thus we have $\mathit{potential}(X_1) = \mathit{potential}(X)$ and $\mathit{matched}(X_1) = \mathit{matched}(X)$. Further, since $\mathit{agents}(X, u_0) \cap \mathit{nonpositive}(X) = \emptyset$ and $\mathit{potential}(X_1) = \mathit{potential}(X_0)$, by Lemmas 4.12 and 4.13 can be used to argue that we have $\mathit{items}(X_1, u_0) = \mathit{items}(X, u_0)$ and $\mathit{agents}(X_1, u_0) \cap \mathit{nonpositive}(X_1) = \emptyset$; thus it follows that $\mathit{potential}(X_{10}, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0)$.

Let $X_1' = \mathrm{inc}(\mathit{raise}'(X_1, u_0), u_0)$; it is easy to see that $X_1' = \mathit{shift}(X_0', u_1, 1)$. Since $u_1 \neq \mathit{victim}(X_0', u_0)$ and $X_1' = \mathit{shift}(X_0', u_1, 1)$, we have $u_1 \neq \mathit{victim}(X_1', u_0)$; thus, $\mathit{victim}(X_1', u_0) = \mathit{victim}(X_0', u_0) = \mathit{victim}(X, u_0)$, and $\mathit{matched}(X_{10}) = \mathit{matched}(X) + u_0 - \mathit{victim}(X, u_0)$.

It follows that $\mathit{matched}(X_{01}) = \mathit{matched}(X_{10})$ and $\mathit{potential}(X_{01}) = \mathit{potential}(X_{10})$.

- Case 2: $\mathit{victim}(X, u_0) = u_1$.

  - Case 2.1: $\mathit{victim}(X, u_0, 1) \neq u_0$.

    In this case, $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_0, 1) = u_1$; thus, $\mathit{potential}(X_0) = \mathit{potential}(X)$ and $\mathit{matched}(X_0) = \mathit{matched}(X) + u_0 - u_1$. By the definition of the function *raise*, we have $\mathit{gap}(X, u_1) = 0$ and $\mathit{gap}(X_0, u_0) = 1$. We consider two sub-cases.

    First we consider the sub-case where $\mathit{agents}(X_0, u_1) \cap \mathit{nonpositive}(X_0) \neq \emptyset$. In this case, we have $\mathit{potential}(X_{01}) = \mathit{potential}(X_0)$ and by Lemma 4.13, we have $\mathit{agents}(X_0, u_1) \cap \mathit{nonpositive}(X_0) = \mathit{nonpositive}(X) \cap (\mathit{agents}(X, u_1) \cup \mathit{agents}(X, u_0))$. Since $\mathit{gap}(X, u_1) = 0$ and $X_1 = \mathit{shift}(X, u_1, 1)$, we have $\mathit{gap}(X_1, u_1) = 1$; by the definition of the function *raise*′, we find that u_1 does not belong to $\mathit{nonpositive}(X_0)$ and $\mathit{victim}(X_0, u_1) \neq u_1$. Thus, $\mathit{potential}(X_{01}) = \mathit{potential}(X)$ and $\mathit{matched}(X_{01}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X_0, u_1)$. Since u_1 belongs to $\mathit{agents}(X_1, u_0)$ and $X_1 = \mathit{shift}(X, u_1, 1)$, it follows that $\mathit{potential}(X_1) = \mathit{potential}(X_0)$ and $\mathit{nonpositive}(X_1) \cap \mathit{agents}(X_1, u_0) = \mathit{nonpositive}(X) \cap (\mathit{agents}(X, u_0) \cup \mathit{agents}(X, u_1)) - u_1$; thus $\mathit{victim}(X_1, u_0) = \mathit{victim}(X_0, u_1)$. Therefore, we have $\mathit{potential}(X_{10}) = \mathit{potential}(X)$ and $\mathit{matched}(X_{10}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X_0, u_1)$.

    Next we consider the sub-case where $\mathit{agents}(X_0, u_1) \cap \mathit{nonpositive}(X_0) = \emptyset$, In this case, we have $\mathit{potential}(X_{01}, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X_0, u_1)$ and $\mathit{victim}(X_0, u_1) = \mathit{victim}(X_0', u_1, 0)$ where $X_0' = \mathrm{inc}(\mathit{raise}'(X, u_1), u_1)$. Since u_0 belongs to $\mathit{agents}(X_0, u_1)$ and $\mathit{gap}(X_0, u_0) = 1$, we have u_0 belongs to $\mathit{nonpositive}(X_0') \cap \mathit{agents}(X_0', u_1)$; thus u_1 belongs to $\mathit{matched}(X_{01})$. Thus, we have $\mathit{matched}(X_{01}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X_0', u_1, 0)$ and $\mathit{potential}(X_{01}, v) = \mathit{potential}(X, v) + 1$ for any item in $\mathit{items}(X, u_1)$. Since $\mathit{potential}(X_0) = \mathit{potential}(X)$ and $\mathit{victim}(X, u_0, 1) = u_1$, and since $\mathit{agents}(X_0, u_1) \cap \mathit{nonpositive}(X_0) = \emptyset$, we have $\mathit{agents}(X, u_0) \cap \mathit{nonpositive}(X_0) = \{u_1\}$. Since $X_1 = \mathit{shift}(X, u, 1)$, we have $\mathit{agents}(X_1, u_0) \cap \mathit{nonpositive}(X_1) = \emptyset$; thus $\mathit{potential}(X_{10}, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X_1, u_0)$, where $\mathit{items}(X_1, u_0) = \mathit{items}(X_0, u_1)$. Let $X_1' = \mathrm{inc}(\mathit{raise}'(X_1, u_1), u_1)$; thus we have $X_1' = \mathit{shift}(X_0', u_1, 1)$, $\mathit{agents}(X_1', u_0) \cap \mathit{nonpositive}(X_1') = \mathit{agents}(X_0', u_1) \cap \mathit{nonpositive}(X_0') - u_1$, and we have $\mathit{victim}(X_1', u_0, 0) = \mathit{victim}(X_0', u_1, 0)$. Therefore, we have $\mathit{potential}(X_{10}) = \mathit{potential}(X)$ and $\mathit{matched}(X_{10}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X_0', u_1, 0)$.

– Case 2.2: $\mathit{victim}(X, u_0, 1) = u_0$.

In this case, $\mathit{victim}(X, u_0) = \mathit{victim}(X'_0, u_0, 0) = u_1$ where $X'_0 = \mathrm{inc}(\mathit{raise}'(X, u_0), u_0)$. Since $\mathit{victim}(X, u_0, 1) = u_0$, we have $\mathit{nonwhite}(X) \cap \mathit{agents}(X, u_0) = \emptyset$; thus, $\mathit{potential}(X_0, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0)$, and by the definition of the function raise'', we have $\mathit{gap}(X_0, u_0) = 0$. Since u_0 belongs to $\mathit{agents}(X_0, u_1)$ and $\mathit{gap}(X_0, u_0) = 0$, we have u_1 belongs to $\mathit{matched}(\mathit{raise}'(X_0, u_1))$; thus $\mathit{potential}(X_{01}) = \mathit{potential}(X_0)$ and $\mathit{matched}(X_{01}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X_0, u_0, 1)$.

We established above that $X_1 = \mathit{shift}(X, u_1, 1)$; thus we have $\mathit{potential}(X_1) = \mathit{potential}(X)$ and $\mathit{matched}(X_1) = \mathit{matched}(X)$. Further, since $\mathit{agents}(X, u_0) \cap \mathit{nonpositive}(X) = \emptyset$ and $\mathit{potential}(X_1) = \mathit{potential}(X)$, Lemmas 4.12 and 4.13 imply that $\mathit{items}(X_1, u_0) = \mathit{items}(X, u_0)$ and $\mathit{agents}(X_1, u_0) \cap \mathit{nonpositive}(X_1) = \emptyset$; thus $\mathit{potential}(X'_1, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0)$, where $X'_1 = \mathrm{inc}(\mathit{raise}'(X_1, u_0), u_0)$. Since $\mathit{gap}(X_0, u_1) = 0$, $\mathit{potential}(X'_1) = \mathit{potential}(X_0)$ and $X_1 = \mathit{shift}(X, u_1, 1)$, we have $\mathit{gap}(X'_1, u_1) = 1$; thus $u_1 \neq \mathit{victim}(X'_1, u_0, 0)$. By the definition of the function raise, we have $X'_1 = \mathit{shift}(X_0, u_1, 1)$; thus, $\mathit{victim}(X'_1, u_0) = \mathit{victim}(X_0, u_0, 0)$. It is now easy to see that $\mathit{matched}(X_{10}) = \mathit{matched}(X) + u_0 + u_1 - \mathit{victim}(X, u_0, 0)$ and $\mathit{potential}(X_{10}) = \mathit{potential}(X_0)$.

□

Lemma 4.19. *For any quiescent ECC X and any agents u_0 and u_1 in* $\mathrm{unmatched}(X)$*, if* $\mathrm{victim}(X, u_0) = \mathrm{victim}(X, u_1) = u$*, then* $\mathrm{potential}(X_{01}) = \mathrm{potential}(X_{10})$ *and* $\mathrm{matched}(X_{01}) = \mathrm{matched}(X_{10})$ *where* $X_{01} = \mathrm{raise}(\mathrm{raise}(X, u_0), u_1)$ *and* $X_{10} = \mathrm{raise}(\mathrm{raise}(X, u_1), u_0)$*.*

Proof. Let $X_0 = \mathit{raise}(X, u_0)$ and let $X_1 = \mathit{raise}(X, u_1)$. Let $X'_0 = \mathrm{inc}(\mathit{raise}'(X, u_0), u_0)$ and let $X'_{01} = \mathrm{inc}(\mathit{raise}'(X_0, u_1), u_1)$. Let $X'_1 = \mathrm{inc}(\mathit{raise}'(X, u_1), u_1)$ and let $X'_{10} = \mathrm{inc}(\mathit{raise}'(X_1, u_0), u_0)$. We consider the following cases.

- Case 1: $\mathit{victim}(X, u_1, 1) \neq u_1$

  – Case 1.1 $\mathit{victim}(X, u_0, 1) \neq u_0$. We begin by establishing the following sequence of claims.

    1. $\mathit{potential}(X_0) = \mathit{potential}(X)$. Follows from the fact that $\mathit{victim}(X, u_0, 1) \neq u_0$ and the definition of the function raise''.
    2. u_0 belongs to $\mathit{agents}(X_0, u_1)$. Follows from the fact that $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_1, 1) = u$ and Lemma 4.15.
    3. $\mathit{potential}(X_1) = \mathit{potential}(X)$. Follows from the fact that $\mathit{victim}(X, u_1, 1) \neq u_1$ and the definition of the function raise''.
    4. u_1 belongs to $\mathit{agents}(X_1, u_0)$. Follows from the fact that $\mathit{victim}(X, u_1) = \mathit{victim}(X, u_0, 1) = u$ and Lemma 4.15.

    We now consider two sub-cases.

    (a) Case 1.1.2 $\mathit{victim}(X_0, u_1, 1) \neq u_1$.

    By claims 1 and 2, we have $\mathit{potential}(X_0) = \mathit{potential}(X_1)$ and by claims 2 and 4, we have u_0 belongs to $\mathit{agents}(X_0, u_1)$ and u_1 belongs to $\mathit{agents}(X_1, u_0)$; thus we have $\mathit{victim}(X_1, u_0, 1) \neq u_0$. Since $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_1) = u$, we have $\mathit{matched}(X_0) \setminus \mathit{matched}(X_1) = \{u_0\}$ and $\mathit{matched}(X_1) \setminus \mathit{matched}(X_0) = \{u_1\}$. Further, by the definition of the function raise, we have $\mathit{bid\text{-}graph}(X_0) = \mathit{bid\text{-}graph}(X_1)$ and for any agent u' in $\mathit{nonwhite}(X)$, there exists an item v' in $\mathit{items}(X)$ such that $\mathit{match}(\chi, v) = u'$ for any configuration χ in $X_0 \cup X_1$. Thus, by Lemma 4.14, we have $victim(X_0, u_1, 1) = victim(X_1, u_0, 1)$. Further, by the definition of the function $raise''$, we have $potential(X_{01}) = potential(X_{10})$. Since $victim(X, u_0) = victim(X, u_1)$ and $victim(X_0, u_1, 1) = victim(X_1, u_0, 1)$, we have $matched(X_{01}) = matched(X_{10})$.

(b) Case 1.1.2 $victim(X_0, u_1, 1) = u_1$.
By claims 1 and 2, we have $potential(X_0) = potential(X_1)$ and by claims 2 and 4, we have u_0 belongs to $agents(X_0, u_1)$ and u_1 belongs to $agents(X_1, u_0)$; thus we have $agents(X_0, u_1) = agents(X_1, u_0)$ and by the definition of the function $raise''$, we have $potential(X'_{01}) = potential(X'_{10})$. Since $victim(X, u_0) = victim(X, u_1) = u$, we have $matched(X'_{01}) \setminus matched(X'_{10}) = \{u_0\}$ and $matched(X'_{10}) \setminus matched(X'_{01}) = \{u_1\}$. Further, by the definition of the function *raise*, we have $bid\text{-}graph(X'_{01}) = bid\text{-}graph(X'_{10})$ and for any agent u' in $nonwhite(X)$, there exists an item v' in $items(X)$ such that $match(\chi, v) = u'$ for any configuration χ in $X'_{01} \cup X'_{10}$. Thus, by Lemma 4.14, we have $victim(X'_{01}, u_1, 0) = victim(X'_{10}, u_0, 0)$. By the definition of the function $raise''$, we have $potential(X_{01}) = potential(X_{10})$. Since $victim(X, u_0) = victim(X, u_1)$ and $victim(X'_{01}, u_1, 0) = victim(X_{10}, u_0, 0)$, we have $matched(X_{01}) = matched(X_{10})$.

– Case 1.2. $victim(X, u_0, 1) = u_0$. We begin by establishing the following sequence of claims.

1. $potential(X_0, v) = potential(X, v)+1$ for any item v in $items(X, u_0)$ and $potential(X_0, v) = potential(X, v)$ for any item v in $items(X) \setminus items(X, u_0)$. Follows from the fact that $victim(X, u_0, 1) = u_0$ and the definition of the function $raise''$.
2. $gap(X_0, u_0) = 0$. We have $victim(X, u_0, 1) = u_0$ and $victim(X, u_0) = u$; thus, u_0 is matched by a $raise''$ invocation and $gap(X_0, u_0) = 0$.
3. u_0 belongs to $agents(X_0, u_1)$. Since $victim(X, u_0) = victim(X, u_1, 1)$, by Lemma 4.15, we have u_0 belongs to $agents(X_0, u_1)$.
4. $potential(X'_{01}) = potential(X_0)$. By 2 and 3, we have u_0 belongs to $nonpositive(X_0) \cap agents(X_0, u_1)$; thus by the definition of the function $raise'$, we have $potential(X'_{01}) = potential(X_0)$.
5. $potential(X_1) = potential(X)$. Follows from the fact that $victim(X, u_1, 1) = u$ and the definition of the function $raise'$.
6. $potential(X'_{10}) = potential(X_0)$. Since $victim(X, u_0, 1) = u$, we have $nonpositive(X) \cap agents(X, u_0) = \emptyset$, thus by 5 and Lemma 4.12, we have $items(X_1, u_0) = items(X, u_0)$, and by Lemma 4.13, we have $nonpositive(X_1) \cap agents(X_1, u_0) = \emptyset$; thus by the definition of the function $raise''$ and 1, we have $potential(X'_{10}) = potential(X_0)$.
7. u_1 belongs to $agents(X'_{10}, u_0)$. Since $victim(X, u_0, 1) = u_0$, we have $victim(X, u_0) = victim(X'_0, u_0, 0) = u$ where $X'_0 = \text{inc}(raise'(X, u_0), u_0)$. We have $victim(X, u_1, 1) = u$. Thus, by Lemma 4.16, we have u_1 belongs to $agents(X'_{10}, u_0)$.
8. $victim(X'_{10}, u_0, 0) = victim(X_0, u_1, 1)$. By claims 4 and 6, we have $potential(X'_{01}) = potential(X'_{10})$ and by claims 3 and 7 we have u_0 belongs to $agents(X_0, u_1)$ and u_1 belongs to $agents(X_{10'}, u_0)$. It is easy to see that $matched(X_0) \setminus matched(X'_{10}) = \{u_0\}$ and $matched(X'_{10}) \setminus matched(X_0) = \{u_1\}$, and by the definition of the function *raise*, for any agent in $nonwhite(X'_{01})$, there exists an item in $items(X)$ such that $match(\chi, v) = u$ for any configuration in $X_0 \cup X'_{10}$. Thus, it follows from Lemma 4.16 that $victim(X'_{10}, u_0, 0) = victim(X_0, u_1, 1)$.

By claims 4 and 6, we have $\mathit{potential}(X_{01}) = \mathit{potential}(X_{10})$. The statement of the lemma assumes that $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_1)$ and by claim 8, we have $\mathit{victim}(X'_{10}, u_0, 0) = \mathit{victim}(X_0, u_1, 1)$; thus $\mathit{matched}(X_{01}) = \mathit{matched}(X_{10})$.

- Case 2: $\mathit{victim}(X, u_1, 1) = u_1$
  - Case 2.1: $\mathit{victim}(X, u_0, 1) \neq u_0$
    This case is symmetric to case 1.2.
  - Case 2.1: $\mathit{victim}(X, u_0, 1) = u_0$. We begin by establishing the following sequence of claims.
    1. $\mathit{potential}(X_0, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0)$ and $\mathit{potential}(X_0, v) = \mathit{potential}(X, v)$ for any item v in $\mathit{items}(X) \setminus \mathit{items}(X, u_0)$. Follows from the fact that $\mathit{victim}(X, u_0, 1) = u_0$ and the definition of the function raise''.
    2. $\mathit{potential}(X'_{01}, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_0) \cup \mathit{items}(X, u_1)$ and $\mathit{potential}(X'_{01}, v) = \mathit{potential}(X, v)$ for any item v in $\mathit{items}(X) \setminus \mathit{items}(X, u_0) \cup \mathit{items}(X, u_1)$. Since $\mathit{victim}(X, u_1, 1) = u_1$, we have $\mathit{nonpositive}(X) \cap \mathit{agents}(X, u_1) = \emptyset$; by 1 and Lemma 4.12, we have $\mathit{items}(X_0, u_1) = \mathit{items}(X, u_1) \setminus \mathit{items}(X, u_0)$, and by Lemma 4.13, we have $\mathit{nonpositive}(X_0) \cap \mathit{agents}(X_0, u_1) = \emptyset$; thus, claim 2 follows by the definition of the function raise'' and claim 1.
    3. u_0 belongs to $\mathit{agents}(X'_{01}, u_1)$. By claim 2, we have $\mathit{potential}(X'_{01}) > \mathit{potential}(X_0)$; thus $\mathit{victim}(X_0, u_1, 1) = u_1$ and $\mathit{victim}(X_0, u_1) = \mathit{victim}(X'_{01}, u_1, 0)$; and by Lemma 4.16, we find that u_0 belongs to $\mathit{agents}(X'_{01}, u_1)$.
    4. $\mathit{potential}(X_1, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_1)$ and $\mathit{potential}(X_0, v) = \mathit{potential}(X, v)$ for any item v in $\mathit{items}(X) \setminus \mathit{items}(X, u_1)$. Follows from the fact that $\mathit{victim}(X, u_0, 1) = u_0$ and the definition of the function raise''.
    5. $\mathit{potential}(X'_{10}, v) = \mathit{potential}(X, v) + 1$ for any item v in $\mathit{items}(X, u_1) \cup \mathit{items}(X, u_0)$ and $\mathit{potential}(X'_{10}, v) = \mathit{potential}(X, v)$ for any item v in $\mathit{items}(X) \setminus \mathit{items}(X, u_1) \cup \mathit{items}(X, u_0)$. The analysis is similar to claim 2.
    6. u_1 belongs to $\mathit{agents}(X'_{10}, u_0)$. By claim 5, we have $\mathit{potential}(X'_{10}) > \mathit{potential}(X_1)$; thus $\mathit{victim}(X_1, u_0, 1) = u_0$; and $\mathit{victim}(X_1, u_0) = \mathit{victim}(X'_{10}, u_0, 0)$; and by Lemma 4.16, we find that u_1 belongs to $\mathit{agents}(X'_{10}, u_0)$.
    7. $\mathit{victim}(X'_{10}, u_0, 0) = \mathit{victim}(X'_{01}, u_1, 0)$. By claims 4 and 5, we have $\mathit{potential}(X'_{01}) = \mathit{potential}(X'_{10})$ and by claims 3 and 6 we have u_0 belongs to $\mathit{agents}(X'_{01}, u_1)$ and u_1 belongs to $\mathit{agents}(X'_{10}, u_0)$. Since $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_1)$, we have $\mathit{matched}(X'_{01}) \setminus \mathit{matched}(X'_{10}) = \{u_1\}$ and $\mathit{matched}(X'_{10}) \setminus \mathit{matched}(X'_{10}) = \{u_0\}$, and by the definition of the function raise, for any agent in $\mathit{nonwhite}(X'_{01})$, there exists an item in $\mathit{items}(X)$ such that $\mathit{match}(\chi, v) = u$ for any configuration in $X'_{01} \cup X'_{10}$. It follows from Lemma 4.16 that $\mathit{victim}(X'_{10}, u_0, 0) = \mathit{victim}(X'_{01}, u_1, 0)$.

    By claims 2 and 5, we have $\mathit{potential}(X_{01}) = \mathit{potential}(X_{10})$. The statement of the lemma assumes that $\mathit{victim}(X, u_0) = \mathit{victim}(X, u_1)$ and by claim 7, we have $\mathit{victim}(X'_{10}, u_0, 0) = \mathit{victim}(X'_{01}, u_1, 0)$; thus, $\mathit{matched}(X_{01}) = \mathit{matched}(X_{10})$.

□

Lemma 4.20. *For any quiescent ECC X and any agents u_0 and u_1 in* enabled(X)*, we have*

$$\text{raise}(\text{raise}(X, u_0), u_1) = \text{raise}(\text{raise}(X, u_1), u_0).$$

Proof. Let $X_{01} = raise(raise(X, u_0), u_1)$ and let $X_{10} = raise(raise(X, u_1), u_0)$.

We first prove the following claim: $potential(X_{01}) = potential(X_{10})$ and $matched(X_{01}) = matched(X_{10})$. By Lemma 4.18, the claim holds when $matched(X) \cap \{u_0, u_1\} \neq \emptyset$. It remains to show that the claim holds when $\{u_0, u_1\} \subseteq unmatched(X)$. By Lemma 4.19, the claim holds when $\{u_0, u_1\} \subseteq unmatched(X)$ and $victim(X, u_0) = victim(X, u_1)$. By Lemma 4.17, the claim holds when $\{u_0, u_1\} \subseteq unmatched(X)$ and $victim(X, u_0) \neq victim(X, u_1)$.

It now remains to be shown that if $potential(X_{01}) = potential(X_{10})$ and $matched(X_{01}) = matched(X_{10})$, then $X_{01} = X_{10}$. Consider any agent u in $nonwhite(X)$; thus, there exists an item v in $items(X)$ such that $match(\chi, v) = u$ for every configuration χ in X. Note that if u belongs to $unmatched(raise(X, u_0))$, then by the definition of the function *raise*, it follows that u belongs to $unmatched(X_{01})$. Using this fact and by repeated application of Lemma 4.7, it follows that either u belongs to $unmatched(X_{01})$ or $match(\chi, v) = u$ for every configuration χ in $X \cup X_{01}$. By an identical argument, we find that either u belongs to $unmatched(X_{10})$ or $match(\chi, v) = u$ for every configuration in $X \cup X_{10}$. However, since we established above that $matched(X_{01}) = matched(X_{10})$, it follows that $match(\chi, v) = u$ for every configuration χ in $X \cup X_{01} \cup X_{10}$, and hence, $X_{01} = X_{10}$. □

4.4 A restricted class of bidding strategies

The first phase of our top-level auction is defined in terms of the bottom-level auction. We associate with each agent u in the top-level auction, a proxy agent u' who bids on behalf of u in the bottom-level auction. The bid of agent u in the top-level auction restricts the number of *raise* invocations of agent u' in the bottom-level auction. Accordingly, we analyze the bottom-level auction when each agent has a restricted "target" number of *raise* invocations.

We define a *target* as a function from the set of all agents to the set of nonnegative integers. For any target α, any agent u, and any integer z such that $\alpha(u) + z \geq 0$, we define $shift(\alpha, u, z)$ as the target α' where $\alpha'(u) = \alpha(u) + z$ and $\alpha'(u') = \alpha(u')$ for any agent u' different from u. For any configuration $\chi = (G, M, \Phi)$ where $G = (U, V, w)$, and any target α, we define $shift(\chi, \alpha)$ as the configuration (G', M, Φ) where $G' = (U, V, w')$ and $w'(u, v) = w(u, v) + \alpha(u)$ for any agent u in U and any item v in V. For any ECC X and any target α, we define $shift(X, \alpha)$ as $\cup_{\chi \in X}[shift(\chi, \alpha)]$.

We view the bottom-level auction as taking a pair (X, α) as input, where X is a quiescent ECC and α is a target, and updating this pair over a sequence of rounds. For any agent u in X, the nonnegative integer $\alpha(u)$ represents the number of additional *raise* invocations desired by agent u. In a general round of the auction with input (X_0, α_0), a single agent u in $enabled(X_0)$ having $\alpha_0(u) > 0$ invokes *raise*, and the output of the round, denoted by $raise(X_0, u, \alpha_0)$ is given by $(raise(X_0, u), shift(\alpha_0, u, -1))$. The auction terminates when no enabled agent has pending *raise* invocations.

We define $bottom(X, \alpha)$ as the output of the bottom-level auction when given the pair (X, α) as input. By Lemma 4.9 and Lemma 4.20, it follows that $bottom(X, \alpha)$ is uniquely defined.

In Section 4.5, we establish various properties of $bottom(X, \alpha)$. These properties are crucial for describing and analyzing the top-level auction of Section 5.

The facts below follow from the definition of the function *raise* and the commutativity of *raise* invocations established in Lemma 4.20.

Fact 4.5. *For any quiescent ECC X, any target α, and any agent u in* $\text{enabled}(X)$*, we have*

$$\text{bottom}(\text{raise}(X, u), \alpha) = \text{bottom}(X, \text{shift}(\alpha, u, 1)).$$

Fact 4.6. *For any quiescent ECC X_0 of the form* $\text{add}(X, u, \beta)$ *and any target α, we have* $\text{bottom}(X_0, \alpha) = \text{bottom}(\text{add}(X', u, \beta), \alpha')$ *where* $(X', \alpha') = \text{bottom}(X, \alpha)$.

Fact 4.7. *For any quiescent ECC X, any agent u in* $\text{white}(X)$*, and any target α, if* $(X_0, \alpha_0) = \text{bottom}(X, \alpha)$*, then*

$$\text{bottom}(X, \text{shift}(\alpha, u, 1)) = \text{bottom}(X_0, \text{shift}(\alpha_0, u, 1)).$$

4.5 Truthfulness-related properties

The goal of this section is to establish Lemma 4.30. We use Lemma 4.30 to establish Lemma 5.9 (see Section 5.4.1) on the truthfulness of the top-level auction.

For any quiescent ECC X and any target α, we define $\mathit{matched}(X, \alpha)$ as the set of agents in $\mathit{matched}(X')$, where $(X', \alpha') = \mathit{bottom}(X, \alpha)$. For any quiescent ECC X, any target α, and any item v in $\mathit{items}(X)$, we define $\mathit{agents}(X, \alpha, v)$ as $\mathit{agents}(X', v)$, where $(X', \alpha') = \mathit{bottom}(X, \alpha)$.

Lemma 4.21. *For any quiescent ECC X, any quiescent ECC X' of the form* $\text{subst}(X, u, u')$*, and any target α such that $\alpha(u) = \alpha(u')$, if u belongs to* $\text{matched}(X) \cap \text{white}(X)$*, then* $\text{gap}(X_0, u) = \text{gap}(X_1, u')$*, where* $(X_0, \alpha_0) = \text{bottom}(X, \alpha)$ *and* $(X_1, \alpha_1) = \text{bottom}(X', \alpha)$*.*

Proof. By Lemma 4.20, the *raise* invocations of the bottom-level auction instances with inputs X and $\mathit{subst}(X, u, u')$ can be reordered such that at each round, either the same agent invokes *raise* in both executions, or agents u and u' invoke *raise* in their corresponding executions. By the definitions of the functions *raise*$'$ and *raise*$''$, the executions treat agents u and u' identically until both agents attain a utility of zero. By Lemma 4.8, we know that u and u' remain white in every round of their corresponding executions, and by Fact 4.2, we know that the potentials are nondecreasing over the rounds of both executions. Thus, agents u and u' continue to have zero utility for the remainder of the executions, and we have $\mathit{gap}(X_0, u) = \mathit{gap}(X_1, u')$. □

Lemma 4.22. *Let X be a quiescent ECC and let X' be a quiescent ECC of the form* $\text{subst}(X, u, u')$ *such that for any agent u'' in* $\text{agents}(X)$*, we have $u'' < u$ if and only if $u'' < u'$. Let α and α' be targets such that $\alpha(u) = \alpha'(u')$ and $\alpha(u'') = \alpha'(u'')$ for any agent u'' in* $\text{agents}(X) - u$*. If* $(X_0, \alpha_0) = \text{bottom}(X, \alpha)$ *and* $(X_1, \alpha_1) = \text{bottom}(X', \alpha')$*, then* $\text{gap}(X_0, u) + \alpha_0(u) = \text{gap}(X_1, u') + \alpha_1(u')$*.*

Proof. By Lemma 4.20, the *raise* invocations of the bottom-level auction instances with inputs X and $\mathit{subst}(X, u, u')$ can be reordered such that at each round, either the same agent invokes *raise* in both executions, or agents u and u' invoke *raise* in their corresponding executions. Since agents u and u' have the same relative ordering with respect to the agents in $\mathit{agents}(X) - u$, it is easy to see that if X_0 and X_0' are the output ECCs corresponding to the same round in both executions, then we have $X_0' = \mathit{subst}(X_0, u, u')$. □

Lemma 4.23. *For any quiescent ECC X' of the form* $\text{add}(X, u, v, z)$ *and any target α, there exists a unique integer z^* and a unique agent u^* in* $\text{agents}(X) + \epsilon$ *such that u belongs to* $\text{matched}(X', \alpha)$ *if and only if $(z + \alpha(u), u) > (z^*, u^*)$. Moreover, if u belongs to* $\text{matched}(X', \alpha)$*, then* $\text{potential}(X'', v) = z^*$ *where* $(X'', \alpha'') = \text{bottom}(X', \alpha)$*.*

Proof. Let S be the ordered sequence of all pairs of the form (z', u') where z' is an integer and u' is an agent that does not belong to $\mathit{agents}(X) \cup \epsilon$. Consider any pair (z_0, u_0) in S such that $z_0 + \alpha_0(u) < \mathit{potential}(X_0, v)$, where $\alpha_0 = \mathit{subst}(\alpha, u_0, \alpha(u))$ and $(X_0, \alpha_0') = \mathit{bottom}(\mathit{add}(X, u_0, v, z_0), \alpha_0)$. By repeated application of Fact 4.2, we know that $\mathit{potential}(X_0) \geq \mathit{potential}(X)$ and by repeated application of Lemma 4.8, we have u_0 belongs to $\mathit{white}(X_0)$. Thus, u_0 does not belong to $\mathit{matched}(X_0)$. Further, since u_0 belongs to $\mathit{white}(X_0)$, it follows that $\mathit{potential}(X_0, v) \geq z_0$. Since prices cannot grow indefinitely, there must be a first pair $(z_1, u_1) > (z_0, u_0)$ in S such that u_1 belongs to $\mathit{matched}(X_1, \alpha_1)$ where $X_1 = \mathit{add}(X, u_1, v, z_1)$ and $\alpha_1 = \mathit{subst}(\alpha, u_1, \alpha(u))$. Consider the pair

(z_1, u_2) where u_2 is the maximum agent such that $u_2 < u_1$. By Lemma 4.22, if u_2 does not belong to $\mathit{agents}(X) \cup \epsilon$, then u_2 and u_1 have the same relative ordering with respect to the remaining agents in X and thus, u_2 belongs to $\mathit{matched}(\mathit{subst}(X_1, u_1, u_2), \mathit{subst}(\alpha, u_2, \alpha(u)))$. However, we know that (z_1, u_1) is the first pair in S such that u_1 belongs to $\mathit{matched}(X_1, \alpha_1)$. Thus, it follows that u_2 belongs to $\mathit{agents}(X) \cup \epsilon$. Consider any pair $(z_3, u_3) > (z_1, u_1)$ in S; by the definition of the bottom-level auction, we find that u_3 belongs to $\mathit{matched}(\mathit{subst}(X_1, u_1, u_3), \mathit{subst}(\alpha, u_3, \alpha(u)))$. Thus $u^* = u_2$ and $z^* = z_1$.

We now show that if u belongs to $\mathit{matched}(X', \alpha)$, then $\mathit{potential}(X'', v) = z^*$ where $(X'', \alpha'') = \mathit{bottom}(X', \alpha)$. Suppose that $\mathit{potential}(X'', v) < z^*$. Then consider the case where $u < u^*$ and $z + \alpha(u) = z^*$. Since $(z + \alpha(u), u) < (z^*, u^*)$, we have u belongs to $\mathit{unmatched}(X'')$ and since $z + \alpha(u) > \mathit{potential}(X'', v)$, we have u belongs to $\mathit{matched}(X'')$; a contradiction. Suppose that $\mathit{potential}(X'', v) > z^*$. Then consider the case where $u > u^*$ and $z + \alpha(u) = z^*$. Since $(z + \alpha(u) + z, u) > (z^*, u^*)$, we have u belongs to $\mathit{matched}(X'')$ and since $z + \alpha(u) < \mathit{potential}(X'', v)$, we have u belongs to $\mathit{unmatched}(X'')$; a contradiction. It follows that $\mathit{potential}(X'', v) = z^*$. □

For any quiescent ECC X, any target α, and any item v in $\mathit{items}(X)$, we define $\mathit{threshold}^*(X, \alpha, v)$ as the unique pair (z^*, u^*) of Lemma 4.23, and we define $\mathit{threshold}^*(X, \alpha)$ as the function that maps each item v in $\mathit{items}(X)$ to $\mathit{threshold}^*(X, \alpha, v)$. In addition, we define $\mathit{threshold}(X, \alpha, v)$ as the integer z^* and we define $\mathit{threshold}(X, \alpha)$ as the function that maps each item v in $\mathit{items}(X)$ to $\mathit{threshold}(X, \alpha, v)$.

Lemma 4.24. *For any quiescent ECC X, any target α, and any agent u in* $\mathrm{enabled}(X)$*, we have*

$$\mathrm{threshold}^*(X, \alpha) \leq \mathrm{threshold}^*(\mathrm{raise}(X, u), \alpha).$$

Proof. Assume $\mathit{threshold}^*(\mathit{raise}(X, u), \alpha, v_0) < \mathit{threshold}^*(X, \alpha, v_0)$ for some item v_0 in $\mathit{items}(X)$. Let X_0 be an ECC of the form $\mathit{add}(X, u_0, v_0, \mathit{min}(v_0))$ and let α_0 be a target such that (1) $\alpha_0(u') = \alpha(u')$ for any agent u' in $\mathit{agents}(X)$, and (2) $\mathit{threshold}^*(\mathit{raise}(X, u), \alpha, v_0) < \alpha_0(u_0) + \mathit{min}(v_0) < \mathit{threshold}^*(X, \alpha, v_0)$. Note that X_0 is quiescent. We have $\mathit{threshold}^*(X, \alpha) = \mathit{threshold}^*(X, \alpha_0)$, and $\mathit{threshold}^*(\mathit{raise}(X, u), \alpha) = \mathit{threshold}^*(\mathit{raise}(X, u), \alpha_0)$; thus, $\mathit{threshold}^*(\mathit{raise}(X, u), \alpha_0, v_0) < \alpha_0(u_0) + \mathit{min}(v_0) < \mathit{threshold}^*(X, \alpha_0, v_0)$.

Let $(X_1, \alpha_1) = \mathit{bottom}(X_0, \alpha_0)$; since $\alpha_0(u_0) + \mathit{min}(v_0) < \mathit{threshold}^*(X, \alpha_0, v_0)$, by Lemma 4.23 we find that u_0 belongs to $\mathit{unmatched}(X_1)$. Since X_0 is quiescent and u belongs to $\mathit{unmatched}(X_0)$, we have u belongs to $\mathit{white}(X_0)$, and by repeated application of Lemma 4.8, we find that u_0 belongs to $\mathit{white}(X_1)$. We conclude that $\alpha_1(u_0) = 0$.

By Fact 4.5, we have $\mathit{bottom}(\mathit{raise}(X_0, u), \alpha_0) = \mathit{bottom}(X_0, \mathit{shift}(\alpha_0, u, 1))$, and by Fact 4.7, we have $\mathit{bottom}(X_0, \mathit{shift}(\alpha_0, u, 1)) = \mathit{bottom}(X_1, \mathit{shift}(\alpha_1, u, 1))$.

Since $\mathit{threshold}^*(\mathit{raise}(X, u), \alpha_0, v_0) < \alpha_0(u_0) + \mathit{min}(v_0)$, by Lemma 4.23, we find that u_0 belongs to $\mathit{matched}(\mathit{raise}(X_0, u), \alpha_0)$, and since we established above that $\alpha_1(u_0) = 0$, we have u_0 does not belong to $\mathit{matched}(X_1, \mathit{shift}(\alpha_1, u, 1))$. Since $\mathit{bottom}(\mathit{raise}(X_0, u), \alpha_0) = \mathit{bottom}(X_1, \mathit{shift}(\alpha_1, u, 1))$, this yields a contradiction. Thus, we have $\mathit{threshold}^*(X, \mathit{shift}(\alpha, u, 1)) \leq \mathit{threshold}^*(\mathit{raise}(X, u), \alpha)$. □

Lemma 4.25. *For any quiescent ECC X_0 of the form* $\mathrm{add}(X, u, \beta)$ *and any target α, if u does not belong to* $\mathrm{matched}(X_0, \alpha)$*, then* $\mathrm{threshold}^*(X_0, \alpha) = \mathrm{threshold}^*(X, \alpha)$.

Proof. Let $(X', \alpha') = \mathit{bottom}(X, \alpha)$ and let $(X'_0, \alpha'_0) = \mathit{bottom}(X_0, \alpha)$. By Fact 4.6, we have $\mathit{bottom}(X_0, \alpha) = \mathit{bottom}(\mathit{add}(X', u, \beta), \alpha')$. By repeated application of Lemma 4.24, it follows that $\mathit{threshold}^*(X'_0, \alpha'_0) \geq \mathit{threshold}^*(X', \alpha')$. Suppose $\mathit{threshold}^*(X', \alpha', v_1) < \mathit{threshold}^*(X'_0, \alpha'_0, v_1)$ for some v_1 in $\mathit{items}(X)$. Let $X_1 = \mathit{add}(X_0, u_1, v_1, \mathit{min}(v_1))$ for some agent u_1, and let α_1 be a target such that (1) $\mathit{threshold}^*(X', \alpha', v_1) < (\alpha_1(u_1) + \mathit{min}(v_1), u_1) < \mathit{threshold}^*(X'_0, \alpha'_0, v_1)$, and

(2) $\alpha_1(u') = \alpha(u')$ for any u' in *agents*(X_0). Note that *threshold*$^*(X_0, \alpha_0) =$ *threshold*$^*(X_0, \alpha_1)$. Similarly, *threshold*$^*(X_1, \alpha_0) =$ *threshold*$^*(X_1, \alpha_1)$.

Let $X_2 = add(X, u_1, v_1, min(v_1))$; then $X_1 = add(X_2, u, \beta)$. Let $(X_2', \alpha_2') = bottom(X_2, \alpha_1)$; by Fact 4.6, we find that $bottom(X_1, \alpha_1) = bottom(add(X_2', u, \beta), \alpha_2')$. By Lemma 4.23, since $min(v_0) + \alpha_1(u_1) > threshold^*(X', \alpha', v_1)$ and $threshold^*(X_0', \alpha_0') \geq threshold^*(X', \alpha')$, u_0 is in $matched(X_2', \alpha_2')$. By Lemma 4.24, we have $threshold^*(X_2', \alpha_2') \geq threshold^*(X, \alpha_1)$, and since u is not in $matched(X_0, \alpha_1)$, we have u does not belong to $matched(X_1, \alpha_1)$; thus u_1 belongs to $matched(X_1, \alpha_1)$.

Since u belongs to $enabled(X_1)$, by Fact 4.6, it follows that $bottom(add(X_0', u_1, v_1, min(v_1)), \alpha_1') = bottom(X_1, \alpha_1)$. By Lemma 4.23, since $min(v_0) + \alpha_0(u_0) < threshold^*(X_0', \alpha_0', v_1)$, agent u_0 does not belong to $matched(X_0', \alpha_1')$; thus u and u_1 do not belong to $matched(X_1, \alpha_1)$, a contradiction. □

For any quiescent ECC X, any target α, and any item v in *items*(X), we define *price*(X, α, v) as *potential*(X', v) where $(X', \alpha') = bottom(X, \alpha)$, and we define *price*$(X, \alpha)$ as the function that maps every item v in *items*(X) to *price*(X, α, v).

For any quiescent ECC X, any target α, and any item v in *items*(X), we define *price*$^*(X, \alpha, v)$ as $(price(X, \alpha, v), u_0)$, where u_0 is the maximum agent in *agents*(X, α, v). In addition, we define *price*$^*(X, \alpha)$ as the function that maps each item v in *items*(X) to *price*$^*(X, \alpha, v)$.

Lemma 4.26. *For any quiescent ECC X and any target α, we have* $\text{price}^*(X, \alpha) \leq \text{threshold}^*(X, \alpha)$.

Proof. Assume that there exists an item v in *items*(X) such that $price^*(X, \alpha, v) > threshold^*(X, \alpha, v)$. Let $(X', \alpha') = bottom(X, \alpha)$. Let $X_0 = add(X', u, v, min(v))$ for some agent u, and let α_0 be a target such that (1) $threshold^*(X, \alpha, v) < \alpha_0(u) + min(v) < price^*(X, \alpha, v)$, and (2) $\alpha_0(u') = \alpha'(u')$ for any agent u' in *agents*(X). Note that X_0 is quiescent. It is easy to see that $threshold^*(X, \alpha) = threshold^*(X', \alpha') = threshold^*(X', \alpha_0)$; thus $threshold^*(X', \alpha_0, v) < \alpha_0(u) + min(v)$.

We have $bottom(X_0, \alpha_0) = bottom(add(X', u, v, min(v)), subst(\alpha', u, \alpha_0(u)))$; thus, by repeated application of Fact 4.2, we have $price^*(X_0, \alpha_0) \geq price^*(X', \alpha') \geq price^*(X, \alpha)$. Thus, $threshold^*(X', \alpha_0, v) < \alpha_0(u) + min(v) < price^*(X_0, \alpha_0, v)$.

Let $(X_0', \alpha_0') = bottom(X_0, \alpha_0)$. By Lemma 4.23, since $\alpha_0(u) + min(v) > threshold^*(X', \alpha_0, v)$ we find that u belongs to $matched(X_0')$. Since u belongs to $unmatched(X_0)$, we find that u belongs to $white(X_0)$; thus, by repeated application of Lemma 4.8, we have u belongs to $white(X_0')$. However, since $\alpha_0(u) + min(v) < price^*(X, \alpha, v)$, it follows that u belongs to $nonwhite(X_0')$, thus yielding a contradiction. Thus, $price^*(X, \alpha) \leq threshold^*(X, \alpha)$. □

Lemma 4.27. *Let X_0 be a quiescent ECC and let u_0 be an agent in* $\text{unmatched}(X_0)$*. Let X_1 be a quiescent ECC of the form* $\text{subst}(X_0, u_0, u_1)$*, where $u_1 < u_0$. Then for any target α such that $\alpha(u_0) = \alpha(u_1)$, we have* $\text{gap}(X_0', u_0) = \text{gap}(X_1', u_1)$*, where* $(X_0', \alpha_0') = \text{bottom}(X_0, \alpha)$ *and* $(X_1', \alpha_1') = \text{bottom}(X_1, \alpha)$.

Proof. Let $\beta = bid(bid\text{-}graph(X_0), u)$ and let X be the ECC such that $X_0 = add(X, u, \beta)$. Observe that X and X_1 are quiescent. Let $(X', \alpha') = bottom(X, \alpha)$. By Fact 4.6, we have $bottom(X_0, \alpha) = bottom(add(X', u_0, \beta), \alpha')$ and $bottom(X_1, \alpha) = bottom(subst(X_0, u_0, u_1), \alpha')$. We refer to the instance of the bottom-level auction with inputs X_0 and α as execution A and we refer to the instance of the bottom-level auction with inputs X_1 and α as execution B. By Lemma 4.20, *raise* invocations of executions A and B can be reordered such that agents u_0 and u_1 exhaust their *raise* invocations before any other agent invokes the function *raise*. If u_0 and u_1 are unmatched when they exhaust their *raise* invocations, then by the description of the bottom-level auction, agents u_0 and u_1 have zero utility in executions A and B respectively, and they continue to have zero utility for the rest of the corresponding executions; thus $gap(X_0', u_0) = gap(X_1', u_1) = 0$.

For the remainder of this proof, we may assume that consider the following cases at least one of agents u_0 and u_1 is matched by a *raise* invocation in either execution A or execution B. Let k be the first round in which either u_0 or u_1 is matched and let X_k and X'_k be the output ECCs of round k of executions A and B. By repeated application of Lemma 4.11, we have $\mathit{gap}(X_k, u_0) = \mathit{gap}(X'_k, u_1) = 0$, and either $X_k = \mathit{subst}(X'_k, u_1, u_0)$, or $\mathit{raise}(X_k, u_0) = \mathit{raise}(X'_k, u_1)$.

First we consider the case where $X_k = \mathit{subst}(X'_k, u_1, u_0)$. In this case, u_0 belongs to $\mathit{matched}(X_k) \cap \mathit{white}(X_k)$ and u_1 belongs to $\mathit{matched}(X'_k) \cap \mathit{white}(X'_k)$; thus, by Lemma 4.21 we have $\mathit{gap}(X'_0, u_0) = \mathit{gap}(X'_1, u_1)$.

Next we consider the case where $X_k \neq \mathit{subst}(X'_k, u_1, u_0)$. If agents u_0 and u_1 have exhausted their *raise* invocations, then by the description of the bottom-level auction, they continue to have zero utility for the rest of the auction; if u_0 and u_1 have one or more pending *raise* invocations, then by Lemma 4.11, $\mathit{raise}(X_k, u_0) = \mathit{subst}(\mathit{raise}(X'_k, u_0), u_1, u_0)$, and by Lemma 4.21, we have $\mathit{gap}(X'_0, u_0) = \mathit{gap}(X'_1, u_1)$. □

Lemma 4.28. *Let X be a quiescent ECC of the form* $\mathrm{add}(X_0, u, \beta)$ *and for each item v in* $\mathrm{items}(X)$, *let $X_v = \mathrm{add}(X_0, u, v, z)$ where $z = \beta(v)$. Then for any target α, agent u belongs to* $\mathrm{matched}(X, \alpha)$ *if and only if u belongs to* $\mathrm{matched}(X_v, \alpha)$ *for some item v in* $\mathrm{items}(X)$.

Proof. We refer to the bottom-level auction instance with inputs (X, α) as execution A, and for each item v, we refer to the bottom-level auction instance with input (X_v, α) as execution A_v. We represent the output of round i of execution A by (X_i, α_i), and for any v in V, we represent the output of round i of execution A_v by $(X_{v,i}, \alpha_{v,i})$. Note that agent u is unmatched and therefore enabled in all rounds of all executions under consideration. By Lemma 4.20, we choose to defer the *raise* invocations of agent u in each execution to a round j in which u is the only enabled agent. Further, we choose to allow the same agent to invoke *raise* in each round of every execution.

We now allow agent u to exhaust its *raise* invocations in rounds j to k of all executions, where $k = j + \alpha(u)$. We consider the following two cases.

- Case (1) : $(\beta(v) + \alpha(u), u) < \mathit{threshold}^*(X, \alpha(u), v)$ for every item v in $\mathit{items}(X)$.

  By Lemma 4.23, since $(\beta(v) + \alpha(u), u) < \mathit{threshold}^*(X, \alpha(u), v)$ for every item v in $\mathit{items}(X)$, we find that u does not belong to $\mathit{matched}(X_v, \alpha)$ and thus u belongs to $\mathit{unmatched}(X_{v,k})$ for every item v in $\mathit{items}(X)$. Assume that u belongs to $\mathit{matched}(X, \alpha)$; thus, u belongs to $\mathit{matched}(X_k)$. Let α' be a target such that $\alpha'(u') = \alpha_k(u')$ for any u' in $\mathit{agents}(X_k)$ and for any agent of the form u_v where v is an item in $\mathit{items}(X)$, we have $(\beta(v) + \alpha(u), u) < (\alpha'(u_v) + \mathit{min}(v), u_v) < \mathit{threshold}^*(X, \alpha, v)$. By Lemma 4.24, we have $\mathit{threshold}^*(X, \alpha) \leq \mathit{threshold}^*(X_k, \alpha')$; thus, we have $(\beta(v) + \alpha(u), u) < (\alpha'(u_v) + \mathit{min}(v), u_v) < \mathit{threshold}^*(X_k, \alpha', v)$ for any item v in $\mathit{items}(X)$. Let X' be an ECC that is constructed from X_k as follows: initialize $X' = X_k$, and for each item v in $\mathit{items}(X)$, set $X' = \mathit{add}(X', u_v, v, \mathit{min}(v))$. Consider the execution A' of the bottom-level auction with input (X', α'), and for any round i of execution A', let (X'_i, α'_i) represent the output of round i of execution A'. We now use Lemma 4.20, to allow all agents in $\cup_{v \in \mathit{items}(X)} u_v$ to exhaust their *raise* invocations. If m is the last round of the *raise* invocations by agents in $\cup_{v \in \mathit{items}(X)} u_v$, then by Lemmas 4.23, since $(\alpha'(u_v) + \mathit{min}(v), u_v) < \mathit{threshold}^*(X_k, \alpha', v)$ for every item v, we find that agent u_v belongs to $\mathit{unmatched}(X'_m)$ for every v in $\mathit{items}(X)$, and by Lemma 4.25, we have $\mathit{threshold}^*(X'_m, \alpha'_m) = \mathit{threshold}^*(X_k, \alpha_k)$. Since every agent u_v belongs to $\mathit{unmatched}(X'_m)$ and X'_m is quiescent, we have $\mathit{potential}(X'_m, v) \geq (\alpha'(u_v) + \mathit{min}(v), u_v)$ for every item v in $\mathit{items}(X)$; thus by Fact 4.2, we have $\mathit{price}^*(X', \alpha', v) \geq (\alpha'(u_v) + \mathit{min}(v), u_v)$ for every item v. Since $(\beta(v) + \alpha(u), u) < (\alpha'(u_v) + \mathit{min}(v), u_v)$ for every item v, we have $\mathit{price}^*(X', \alpha', v) \geq (\beta(v) + \alpha(u), u)$ for every item v. However, by repeated use of Lemma 4.8, agent u is white at the end of execution A', and by our assumption that u belongs to $matched(X, \alpha)$, we have $price^*(X', \alpha', v) < (\beta(v) + \alpha(u), u)$ for some item v; this yields a contradiction. Thus, we have u does not belong to $matched(X, \alpha)$.

- Case (2) : $(\beta(v) + \alpha(u), u) > threshold^*(X, \alpha, v)$ for some item v in $items(X)$.

  By Lemma 4.23, since $(\beta(v) + \alpha(u), u) > threshold^*(X, \alpha, v)$ for some item v in $items(X)$, we find that u belongs to $matched(X_v, \alpha)$ for some item v in $items(X)$. Assume that u does not belong to $matched(X, \alpha)$. Consider the execution A' defined as in Case 1 above. By Lemma 4.25, we have $threshold^*(X_k, \alpha_k) = threshold^*(X'_m, \alpha'_m)$. By Lemma 4.26, we have $price^*(X'_m, \alpha'_m) \leq threshold^*(X_k, \alpha_k)$. Thus, there exists some item v in $items(X)$ such that u belongs to $unmatched(X'_m)$ and $(\beta(v) + \alpha(u), u) > price^*(X'_m, \alpha'_m, v)$; this violates the quiescent property of X'_m. Thus, u belongs to $matched(X, \alpha)$.

We conclude that agent u belongs to $matched(X, \alpha)$ if and only if u belongs to $matched(X_v, \alpha)$ for some item v in $items(X)$, as required. □

Lemma 4.29. *Let X be a quiescent ECC of the form* $\mathrm{add}(X_0, u, \beta)$*, let α be a target, and for each item v in* $\mathrm{items}(X)$*, let* $X_v = \mathrm{add}(X_0, u, v, z)$*, where $z = \beta(v)$. Then, we have* $\mathrm{gap}(X', u) + \alpha'(u) = \max_{v \in \mathrm{items}(X)}\{\mathrm{gap}(X'_v, u) + \alpha_{v'}(u)\}$*, where* $(X', \alpha') = \mathrm{bottom}(X, \alpha)$ *and* $(X'_v, \alpha'_v) = \mathrm{bottom}(X_v, \alpha)$ *for each item v in* $\mathrm{items}(X)$*.*

Proof. By Lemma 4.27, if $(X^*, \alpha^*) = bottom(add(X_0, u', \beta), \alpha)$ for any agent u', then $gap(X', u) = gap(X^*, u')$. Thus, without loss of generality, we can assume that $u > u'$ for any agent u' in $agents(X)$. By Lemma 4.28, u belongs to $matched(X')$ if and only if u belongs to $matched(X'_v)$ for some v in $items(X_0)$. Thus, if u belongs to $unmatched(X')$, we have $gap(X', u) = gap(X'_v, u) = 0$ for all v in $items(X)$.

We now focus on the case where u belongs to $matched(X')$. Let z be the largest integer such that u belongs to $matched(shift(X', u, -z))$. By Fact 4.7, we have $(X', \alpha') = (X'', shift(\alpha'', u, z))$ where $(X'', \alpha'') = bottom(shift(X, u, -z), \alpha)$; thus $gap(X', u) + \alpha'(u) = gap(X'', u) + \alpha''(u) + z$. By Lemma 4.28, agent u belongs to $matched(X''_v)$ for some item v in $items(X)$, where $(X''_v, \alpha''_v) = bottom(shift(X_v, u, -z), \alpha)$. By Fact 4.7, we have $(X'_v, \alpha'_v) = bottom(X''_v, shift(\alpha'_v, u, z))$; thus we have $gap(X'_v, u) + \alpha'_v = gap(X''_v, u) + \alpha''_v(u) + z$. Since u belongs to $white(X) \cap white(X_v)$, we have $\alpha''(u) = \alpha''_v(u) = 0$. To complete the proof, it remains to be shown that $gap(X''_v, u) = gap(X''_v, u)$.

We refer to the bottom-level auction instance with inputs $(shift(X, u, -z), \alpha)$ as execution A, and for each item v, we refer to the bottom-level auction instance with input $(shift(X_v, u, -z), \alpha)$ as execution A_v. We represent the output of round i of execution A by (X_i, α_i), and for any v in V, we represent the output of round i of execution A_v by $(X_{v,i}, \alpha_{v,i})$. Since u belongs to $unmatched(X)$, it follows that u belongs to $enabled(X) \cap enabled(X_v)$. By Lemma 4.20, we choose to allow agent u to first exhaust its *raise* invocations in all executions. Since $u > u'$ for any agent u' in $agents(X)$, it follows that for each round j in which u invokes *raise*, either u belongs to $unmatched(X_j) \cap unmatched(X_{v,j})$ or there exists a first round j such that u belongs to $matched(X'_j)$ and u belongs to $matched(X'_{v,j})$ for some item v in $items(X)$. Since z was chosen to be the largest integer such that u belongs to $matched(shift(X', u, -z))$, we have $gap(X_j, u) = gap(X_{v,j}, u) = 0$, and thus $gap(X'', u) = gap(X''_v, u) = 0$. □

Lemma 4.30. *Let X' be a quiescent ECC of the form* $\mathrm{add}(X, u, \beta)$ *and let* $(X'', \alpha'') = \mathrm{bottom}(X', \alpha)$ *for some target α. Let Δ denote the maximum, over all items v in* $\mathrm{items}(X)$*, of* $\beta(v) + \alpha(u) - \mathrm{threshold}(X, \alpha, v)$*, and let V denote the set of all items v in* $\mathrm{items}(X)$ *such that* $\beta(v) + \alpha(u) - \mathrm{threshold}(X, \alpha, v) = \Delta$*. Let u_0 denote the minimum, over all items v in V of the second component of the pair given by* $\mathrm{threshold}^*(X, \alpha, v)$*. Then the following conditions hold:*

- *If the pair* $(\Delta, u) < (0, u_0)$*, then agent* u *belongs to* unmatched(X'')*, and* threshold$^*(X', \alpha) =$ threshold$^*(X, \alpha)$*.*

- *If the pair* $(\Delta, u) > (0, u_0)$*, then agent* u *belongs to* matched(X'') *and,* (1) *for every configuration* χ *in* X''*, there exists an item* v *in* V *such that* match$(\chi, v) = u$*, and* (2) potential$(X'', v) =$ threshold(X, α, v) *for any item* v *in* V*.*

Proof. First, we show that u belongs to $matched(X'')$ if and only if $(\Delta, u) > (0, u_0)$.

Let v_0 be any item in $items(X)$; thus, we find that $\beta(v_0) + \alpha(u) - threshold(X, \alpha, v_0) = \Delta$. If $(\Delta, u) < (0, u_0)$, then by adding $threshold(X, \alpha, v_0)$ to the first component of both pairs, we find that $(\beta(v_0) + \alpha(u), u) < (threshold(X, \alpha, v_0), u_0)$. Similarly, if $(\Delta, u) > (0, u_0)$, we find that $(\beta(v_0) + \alpha(u), u) > (threshold(X, \alpha, v_0), u_0)$. By Lemma 4.23, it follows that agent u belongs to $matched(add(X, u, v_0, \beta(v_0)), \alpha)$ if and only if $(\beta(v_0) + \alpha(u), u) > (threshold(X, \alpha, v_0), u_0)$, and by Lemma 4.28, we find that agent u belongs to $matched(X'') = matched(X', \alpha)$ if and only if there exists some item v' in $items(X)$ such that agent u belongs to $matched(add(X, u, v', \beta(v')), \alpha)$; thus, we find that agent u belongs to $matched(X'')$ if and only if $(\Delta, u) > (0, u_0)$.

Next we show that if u does not belongs to $matched(X'')$, then $threshold^*(X', \alpha) = threshold^*(X, \alpha)$. The result follows directly from Lemma 4.25.

Finally, we show that if u belongs to $matched(X'')$, then (1) and (2) stated above hold. By Lemma 4.29, if u belongs to $matched(X'')$, then $gap(X'', u) = max_{v \in items(X)}\{gap(X'_v, u)\}$, where X'_v is equal to $bottom(add(X, u, v, \beta(v)), \alpha)$, and by Lemma 4.23, we have $max_{v \in items(X)}\{gap(X'_v, u)\} = \Delta$; thus $gap(X'', u) = \Delta$. Let v_0 be any item in V. By Lemma 4.26, we have $potential(X'', v_0) \leq threshold(X, \alpha, v_0)$, and since $gap(X'', u) = \Delta$, we have $potential(X'', v_0) \geq threshold(X, \alpha, v_0)$; thus $potential(X'', v_0) = threshold(X, \alpha, v_0)$ and condition (1) holds. Now consider any item v not in V. By definition, we have

$$\beta(v) + \alpha(u) - threshold^*(X, \alpha, v) < \Delta. \tag{3}$$

By Lemma 4.26, we have

$$potential(X'', v) \leq threshold^*(X, \alpha, v). \tag{4}$$

By subtracting (4) from (3), we have $\beta(v) + \alpha(u) - potential(X'', v) < \Delta$; since u belongs to $white(X'')$, agent u attains its highest utility by being matched to some item in V in every configuration of X'' and condition (2) holds. □

4.6 Determinized bottom-level auction

We have been dealing with ECCs in the discussion of the bottom-level auction in the previous sections. Our top-level auction of Section 5 works with configurations; accordingly, we define a suitable determinization of the bottom-level auction.

For any quiescent configuration χ and any agent u in $enabled(\chi)$, we would like to define $raise(\chi, u)$ as a specific configuration in $raise([\chi], u)$ such that for any agent u_0 in $enabled(\chi)$, if u does not belong to $matched(\chi) \cap unmatched(raise(\chi, u_0))$, then $raise(raise(\chi, u_0), u) = raise(raise(\chi, u), u_0)$. In order to do so, we determinize the choice of the augmenting path in function *augment* defined in Section 4.1. Specifically, we pick a lexicographically first (with respect to item identifiers) shortest path.

We view the bottom-level auction as taking a pair (χ, α) as input, where χ is a quiescent configuration and α is a target, and updating this pair over a sequence of rounds. A general round of the auction with input (χ_0, α_0) is defined as follows: if $enabled(\chi_0) = \emptyset$, then the auction terminates; if the minimum agent in $matched(\chi_0) \cap enabled(\chi_0) = \epsilon$ then the minimum agent in *enabled*(χ_0) invokes *raise*; otherwise, the minimum agent in *matched*$(\chi_0) \cap$ *enabled*(χ_0) invokes *raise*. We define *bottom*(χ, α) as the output of the bottom-level auction when given the pair (χ, α) as input.

Below we provide determinized versions of some key lemmas of the bottom-level auction that were discussed in previous sections.

Lemma 4.31. *For any configuration* χ' *of the form* $\text{raise}(\chi, u')$*, and any agent* u *in* $\text{nonwhite}(\chi)$*, either* (1) u *belongs to* $\text{unmatched}(\chi')$*, or* (2) u *belongs to* $\text{nonwhite}(\chi')$*, and there exists an item* v *in* $\text{items}(\chi)$ *such that* $\text{potential}(\chi, v) = \text{potential}(\chi', v)$ *and* $\text{match}(\chi, v) = \text{match}(\chi', v) = u$.

Proof. Let $X' = \textit{raise}([\chi], u')$. By definition, χ' belongs to X'. The result follows from Lemma 4.7. ☐

Lemma 4.32. *For any quiescent configuration* χ *and any agent* u *in* $\text{enabled}(\chi)$*, if* $\chi' = \text{raise}(\chi, u)$*, then*

$$\text{gray}(\chi) \subseteq \text{nonblack}(\chi') \wedge \text{white}(\chi) \subseteq \text{white}(\chi').$$

Proof. Let $X' = \textit{raise}([\chi], u)$. By definition, χ' belongs to X'. The result follows from Lemma 4.8. ☐

Lemma 4.33. *For any quiescent configuration* χ *and any agent* u *in* $\text{enabled}(\chi)$*, we have* $\text{enabled}(\chi) - u \subseteq \text{enabled}(\text{raise}(\chi, u))$.

Proof. Let $X' = \textit{raise}([\chi], u)$. By definition, $\textit{raise}(\chi, u)$ belongs to X'. The result follows from Lemma 4.9. ☐

Lemma 4.34. *For any quiescent configuration* χ*, any quiescent configuration* χ' *of the form* $\text{subst}(\chi, u, u')$*, and any target* α *such that* $\alpha(u) = \alpha(u')$*, if* u *belongs to* $\text{matched}(\chi) \cap \text{white}(\chi)$*, then* $\text{gap}(\chi_0, u) = \text{gap}(\chi_1, u')$*, where* $(\chi_0, \alpha_0) = \text{bottom}(\chi, \alpha)$ *and* $(\chi_1, \alpha_1) = \text{bottom}(\chi', \alpha)$.

Proof. Let $X = [\chi]$ and let $X' = [\chi']$. Let $(X_0, \alpha_0) = \textit{bottom}(X, \alpha)$ and let $(X_1, \alpha_1) = \textit{bottom}(X', \alpha)$. By definition, χ_0 belongs to X_0 and χ_1 belongs to X_1. The result follows from Lemma 4.21. ☐

Lemma 4.35. *Let* χ *be a quiescent configuration and let* χ' *be a quiescent configuration of the form* $\text{subst}(\chi, u, u')$ *such that for any agent* u'' *in* $\text{agents}(\chi)$*, we have* $u'' < u$ *if and only if* $u'' < u'$*. Let* α *and* α' *be targets such that* $\alpha(u) = \alpha'(u')$ *and* $\alpha(u'') = \alpha'(u'')$ *for any agent* u'' *in* $\text{agents}(\chi) - u$*. If* $(\chi_0, \alpha_0) = \text{bottom}(\chi, \alpha)$ *and* $(\chi_1, \alpha_1) = \text{bottom}(\chi', \alpha')$*, then* $\text{gap}(\chi_0, u) + \alpha_0(u) = \text{gap}(\chi_1, u') + \alpha_1(u')$.

Proof. Let $X = [\chi]$ and let $X' = [\chi']$. Let $(X_0, \alpha_0) = \textit{bottom}(X, \alpha)$ and let $(X_1, \alpha_1) = \textit{bottom}(X', \alpha')$. By definition, χ_0 belongs to X_0 and χ_1 belongs to X_1. The result follows from Lemma 4.22. ☐

Lemma 4.36. *Let* χ' *be a quiescent configuration of the form* $\text{add}(\chi, u, \beta)$ *and let* $(\chi'', \alpha'') = \text{bottom}(\chi', \alpha)$ *for some target* α*. Let* Δ *denote the maximum, over all items* v *in* $\text{items}(\chi)$*, of* $\beta(v) + \alpha(u) - \text{threshold}(\chi, \alpha, v)$*, and let* V *denote the set of all items* v *in* $\text{items}(\chi)$ *such that* $\beta(v) + \alpha(u) - \text{threshold}(\chi, \alpha, v) = \Delta$*. Let* u_0 *denote the minimum, over all items* v *in* V *of the second component of the pair given by* $\text{threshold}^*(\chi, \alpha, v)$*. Then the following conditions hold:*

- *If the pair* $(\Delta, u) < (0, u_0)$*, then agent* u *belongs to* $\text{unmatched}(\chi'')$*, and* $\text{threshold}^*(\chi', \alpha) = \text{threshold}^*(\chi, \alpha)$.
- *If the pair* $(\Delta, u) > (0, u_0)$*, then agent* u *belongs to* $\text{matched}(\chi'')$ *and,* (1) *for every configuration* χ *in* χ''*, there exists an item* v *in* V *such that* $\text{match}(\chi, v) = u$*, and* (2) $\text{potential}(\chi'', v) = \text{threshold}(\chi, \alpha, v)$ *for any item* v *in* V.

Proof. Let $X = [\chi]$ and let $X' = [\chi']$. Let $(X'', \alpha'') = bottom(X', \alpha)$. By definition, χ'' belongs to X''. The result follows from Lemma 4.30. □

Lemma 4.37. *For any quiescent configuration χ and any agents u_0 and u_1 in* $\text{enabled}(\chi)$*, we have*

$$\text{raise}(\text{raise}(\chi, u_0), u_1) = \text{raise}(\text{raise}(\chi, u_1), u_0).$$

Proof. By the definition of the function *raise* that takes an ECC as an argument, either u_0 does not belong to $matched(\chi) \cap unmatched(raise([\chi], u_1))$, or u_1 does not belong to $matched(\chi) \cap unmatched(raise([\chi], u_0))$. The result follows from the definition of the function *raise* that takes a configuration as argument. □

5 Top-Level Auction

The top-level auction is our proposed sealed bid unit-demand auction with put options and proceeds in two phases. The first phase corresponds to running an instance of the bottom-level auction and the second phase corresponds to solving an instance of the house allocation problem [13]. We establish strong properties related to truthfulness, efficiency, and privacy for our auction.

We provide a formal description of the first and second phases of the top-level auction in Sections 5.1 and 5.2. Here we briefly mention some of the high-level ideas underlying the design of the first phase. To ensure that the price of an item v does not decrease, at the outset of the first phase, we tentatively impose the following obligation on the agent u who is the target of item v's put: Agent u will remain allocated to v at the strike price of v. Next, we drop the bids of all agents sufficiently until equilibrium properties 1, 2, and 3 of Section 2.4 are satisfied. The first phase then proceeds to update the tentative allocation and pricing in an iterative manner by invoking the bottom-level auction. In section 5.7, we discuss a fast implementation of this iterative procedure. In our fast implementation, each iteration either permanently releases an initially tentatively allocated agent from its obligation, or eliminates an unallocated agent whose unit-demand bid is too low to ever be allocated. The latter property ensures termination of the first phase.

5.1 First phase

For any configuration χ, we define $targets(\chi)$ as the set of all targets α such that there exists a quiescent configuration χ_0 satisfying the following conditions: (1) $shift(\chi_0, \alpha) = \chi$, (2) $white(\chi_0) \cap matched(\chi_0) = white(\chi) \cap matched(\chi)$, and (3) for any agent u in $unmatched(\chi)$ we have $items(\chi_0, u) = \emptyset$. For any configuration χ, we define $target(\chi)$ as the unique pointwise minimum target in $targets(\chi)$.

Consider any instance of the top-level auction with a configuration χ as input and let $\chi = shift(\chi_0, target(\chi))$. Recall from Section 2.4, that each item v in $items(\chi)$ is associated with a put whose strike price is $min(v)$, and whose target is the agent $match(\chi, v)$. The output of the first phase of the top-level auction, denoted $top'(\chi)$, is given by $shift(\chi', \alpha')$, where $(\chi', \alpha') = bottom(\chi_0, target(\chi))$.

5.2 Second phase

The second phase of the top-level auction affects only the allocation and uses a single application of either the TTC algorithm [13] or the $\text{TC}^{\prec}$ algorithm of Jaramillo and Manjunath [9] to exchange items within a certain subset of the allocated agents.

For any configuration $\chi = (G, M, \Phi)$, we define an instance of the house allocation problem on χ as follows. Each agent in $black(\chi)$ represents a house owner and the item matched to u in M represents the house owned by u. Each agent u in $black(\chi)$ is associated with a preference ordering over the items as follows, where $\beta = bid(\chi, u)$: for any pair of items v and v', if $\beta(v) - \Phi(v) > \beta(v') - \Phi(v')$, then agent u prefers item v over item v'; ties, if any, are broken using item identifiers.

For any configuration χ, we define $top''(\chi)$ as the configuration obtained by using the TTC algorithm to resolve the house allocation problem defined on χ. Alternatively, the second phase of the top-level auction can be implemented using the polynomial time $\text{TC}^{\prec}$ algorithm. The $\text{TC}^{\prec}$ algorithm has a slower running time than the TTC algorithm but yields an outcome with stronger efficiency-related properties than the TTC algorithm. (see Section 5.5 for details.)

For any instance of the top-level auction with configuration χ as input, the second phase of the top-level auction takes the configuration $\chi' = top'(\chi)$ as input and produces the configuration $top''(\chi')$ as output. For any instance of the top-level auction with configuration χ as input, we define $top(\chi)$ as $top''(top'(\chi))$.

Recall that equilibrium properties 1, 2, and 3 of Section 2.4 are satisfied by the first phase of the top-level auction. In the second phase, item prices and the allocation of non-black agents remain unchanged. Thus, it is easy to see that equilibrium properties 1, 2, and 3 are retained in the second phase. The second phase involves resolving an instance of the house allocation problem on the subset of black agents; thus, by definition, equilibrium properties $4(a)$ and $4(b)$ are satisfied. Finally, it follows from known results on the $\text{TC}^{\prec}$ (TTC) algorithm that the solution computed in the second phase is in the (weak) core. This establishes equilibrium property 5 of Section 2.4.

Fact 5.1. *For any configuration χ' of the form $\text{top}''(\chi)$ where $\chi = (G, M, \Phi)$ and $\chi' = (G, M', \Phi')$, we have $\Phi' = \Phi$, $\text{unmatched}(\chi') = \text{unmatched}(\chi)$, and $\text{white}(\chi) \subseteq \text{white}(\chi')$.*

Fact 5.2. *For any configuration $\chi = (G, M, \Phi)$, if $\text{top}(\chi) = (G, M', \Phi')$, then $\Phi' \geq \Phi$.*

5.3 Properties

The following lemmas establish basic properties of the top-level auction.

Lemma 5.1. *For any configuration χ and any targets α_0 and α_1 in $\text{targets}(\chi)$ such that $\chi = \text{shift}(\chi_0, \alpha_0) = \text{shift}(\chi_1, \alpha_1)$, we have $\text{bottom}(\chi_0, \alpha_0) = \text{bottom}(\chi_1, \alpha_1)$.*

Proof. Let $\chi = shift(\chi^*, \alpha^*)$, where $\alpha^* = target(\chi)$. Since α^* is the pointwise minimum target in $targets(\chi)$, we have $\alpha_0(u) \geq \alpha^*(u)$ for any agent u in $agents(\chi)$. Let S be the set of agents u in $agents(\chi)$ such that $\alpha_0(u) > \alpha^*(u)$. By the definitions of $targets(\chi)$ and α^*, for any agent u in S, we find that u belongs to $enabled(\chi_0)$ and $raise(\chi_0, u) = shift(\chi_0, u_0, 1)$; by repeated use of this fact and Lemma 4.37, agents in S can commute their *raise* invocations forward until each agent u in S has $\alpha^*(u)$ pending *raise* invocations and the resulting configuration is χ^*; thus, $bottom(\chi_0, \alpha_0) = bottom(\chi^*, \alpha^*)$. By a similar argument, we have $bottom(\chi_1, \alpha_1) = bottom(\chi^*, \alpha^*)$. Thus, $bottom(\chi_0, \alpha_0) = bottom(\chi_1, \alpha_1)$. □

Lemma 5.2. *Let χ be any configuration and let χ^* be the configuration such that $\chi = \text{shift}(\chi^*, \text{target}(\chi))$. If $(\chi_0, \alpha_0) = \text{bottom}(\chi^*, \text{target}(\chi))$, then $\text{unmatched}(\text{top}'(\chi)) \subseteq \text{white}(\text{top}'(\chi))$ and $\text{nonwhite}(\text{top}'(\chi)) \subseteq \text{nonwhite}(\chi_0)$.*

Proof. Let $\alpha^* = target(\chi)$. By definition, we have $top'(\chi) = shift(\chi_0, \alpha_0)$. Since χ_0 is quiescent, we have $unmatched(\chi_0) \subseteq white(\chi_0)$ and for any agent u in $unmatched(\chi_0)$, we find that $agents(\chi_0, u) \cap nonwhite(\chi_0) = \emptyset$. Moreover, these facts imply that by definition of the bottom-level auction, we have $\alpha_0(u) = 0$ for any agent u in *white*(χ_1); we conclude that u belongs to *white*(*shift*(χ_0, α_0)), where *top*$'(\chi) =$ *shift*(χ_0, α_0).

It remains to show that *nonwhite*(*top*$'(\chi)$) $\subseteq$ *nonwhite*(χ_0). Consider any agent u in *nonwhite*(*top*$'(\chi)$); since $\alpha_0(u) \geq 0$, we conclude that u belongs to *nonwhite*(χ_0). □

Lemma 5.3. *For any configuration χ and any agent u in* nonwhite(top$'(\chi)$)*, the following conditions hold:* (1) *u belongs to* nonwhite(χ)*,* (2) *there exists an item v in* items(χ) *such that* potential$(\chi, v) =$ potential(top$'(\chi), v$)*, and* (3) match$(\chi, v) =$ match(top$'(\chi), v) = u$*.*

Proof. Let $\chi =$ *shift*(χ^*, α^*), where $\alpha^* =$ *target*(χ), and let $(\chi_0, \alpha_0) =$ *bottom*(χ^*, α^*). By definition, we have *top*$'(\chi) =$ *shift*(χ_0, α_0). By Lemma 5.2, u belongs to *nonwhite*(χ_0); since χ_0 is quiescent, we find that u belongs to *matched*(χ_0); further, since *top*$'(\chi) =$ *shift*(χ_0, α_0), there exists an item v in *items*(χ) such that *potential*(*top*$'(\chi), v$) $=$ *potential*(χ_0, v) and *match*$(\chi, v) =$ *match*(*top*$'(\chi), v$). By repeated application of Lemma 4.31, we find that u belongs to *nonwhite*(χ^*), *potential*$(\chi^*, v) =$ *potential*(χ_0, v) and *match*$(\chi^*, v) =$ *match*$(\chi_0, v) = u$. By the description of the top-level auction it follows that u belongs to *nonwhite*(χ), *potential*$(\chi, v) =$ *potential*(χ^*, v), and *match*$(\chi, v) = u$. These facts imply that u belongs to *nonwhite*(χ) and there exists an item v in *items*(χ) such that *potential*$(\chi, v) =$ *potential*(*top*$'(\chi), v$), and *match*$(\chi, v) =$ *match*(*top*$'(\chi), v$) $=$ u. □

Lemma 5.4. *For any configuration χ, we have* nonwhite(top(χ)) $\subseteq$ nonwhite(χ) *and* unmatched$(\chi) \cup$ white$(\chi) \subseteq$ white(top(χ))*.*

Proof. Let $\chi =$ *shift*(χ^*, α^*) where $\alpha^* =$ *target*(χ), and let $(\chi_0, \alpha_0) =$ *bottom*(χ^*, α^*). We have *unmatched*$(\chi) =$ *unmatched*$(\chi^*) \cap$ *white*(χ^*) and *white*$(\chi) \cap$ *matched*$(\chi) =$ *white*$(\chi^*) \cap$ *matched*(χ^*); thus *unmatched*$(\chi) \cup$ *white*$(\chi) \subseteq$ *white*(χ^*). By repeated application of Lemma 4.32, we have *white*$(\chi^*) \subseteq$ *white*(χ_0), and since *top*$'(\chi) =$ *shift*(χ_0, α_0), we have *white*$(\chi^*) \subseteq$ *white*(*top*$'(\chi)$). Finally, by Fact 5.1, we have *white*(*top*$'(\chi)$) $\subseteq$ *white*(*top*(χ)). Thus, we have *unmatched*$(\chi) \cup$ *white*$(\chi) \subseteq$ *white*(*top*(χ)). By Fact 5.1, we have *white*(*top*$'(\chi)$) $\subseteq$ *white*(*top*(χ)); thus *nonwhite*(*top*(χ)) $\subseteq$ *nonwhite*(*top*$'(\chi)$). □

Lemma 5.5. *Let χ_0 and χ_1 be configurations and let α_0 and α_1 be targets such that* (1) *configurations χ_0 and χ_1 are quiescent, and $\chi =$* shift$(\chi_0, \alpha_0) =$ shift(χ_1, α_1)*,* (2) white$(\chi_0) \cap$ matched$(\chi_0) =$ white$(\chi_1) \cap$ matched$(\chi_1) = \emptyset$*, and* (3) *for any agent u in* unmatched(χ)*, we have* items$(\chi_0, u) =$ items$(\chi_1, u) = \emptyset$*. Then, we have* bottom$(\chi_0, \alpha_0) =$ bottom(χ_1, α_1)*.*

Proof. Let α^* be the target such that for any agent u in *agents*(χ), we have $\alpha^*(u) =$ *min*$(\alpha_0(u), \alpha_1(u))$, and let $\chi =$ *shift*(χ^*, α^*); thus $\alpha_0(u) \geq \alpha^*(u)$. Let S be the set of agents u in *agents*(χ) such that $\alpha_0(u) > \alpha^*(u)$. By the definitions of α_0 and α^*, for any agent u in S, we find that u belongs to *enabled*(χ_0) and *raise*$(\chi_0, u) =$ *shift*$(\chi_0, u_0, 1)$; by repeated use of this fact and Lemma 4.37, agents in S can commute their *raise* invocations forward until each agent u in S has $\alpha^*(u)$ pending *raise* invocations and the resulting configuration is χ^*; thus, *bottom*$(\chi_0, \alpha_0) =$ *bottom*(χ^*, α^*). By a similar argument, we have *bottom*$(\chi_1, \alpha_1) =$ *bottom*(χ^*, α^*). Thus, *bottom*$(\chi_0, \alpha_0) =$ *bottom*(χ_1, α_1). □

Lemma 5.6. *The second phase of the top-level auction is truthful.*

Proof. Consider any configuration χ' of the form *top*$''(\chi)$. We first The second phase of the top-level auction, which is implemented using an application of the TTC algorithm or the TC$^{\prec}$, updates only the matching of black agents. By known results on the truthfulness of the TTC and TC$^{\prec}$ algorithms, the second phase of the top-level auction is truthful for agents in *black*(χ). By Fact 5.1, we have *potential*(*top*$''(\chi)$) $=$ *potential*(χ); thus no agent u in *nonblack*(χ) can achieve a utility higher than $gap(\chi, u)$ by submitting a false bid. Thus, the second phase of the top-level auction is truthful. $\square$

Lemma 5.7. *For any configuration* χ_0 *of the form* $\mathrm{subst}(\chi, u, \beta)$*, if* $\beta \neq \mathrm{bid}(\mathrm{bid\text{-}graph}(\chi), u)$*, then either* $\mathrm{top}'(\chi_0) = \mathrm{subst}(\mathrm{top}'(\chi), u, \beta)$ *or* u *belongs to* $\mathrm{white}(\mathrm{top}'(\chi)) \cap \mathrm{white}(\mathrm{top}'(\chi_0))$.

Proof. Let $\chi = \mathit{shift}(\chi^*, \mathit{target}(\chi))$ and let $\chi_0 = \mathit{shift}({\chi_0}^*, \mathit{target}(\chi_0))$. By definition, we have $\mathit{top}'(\chi) = \mathit{shift}(\chi', \alpha')$ where $(\chi', \alpha') = \mathit{bottom}(\chi^*, \mathit{target}(\chi))$ and we have $\mathit{top}'(\chi_0) = \mathit{shift}(\chi'_0, \alpha'_0)$ where $(\chi'_0, \alpha'_0) = \mathit{bottom}({\chi_0}^*, \mathit{target}(\chi_0))$. We consider the following cases. First we consider the case where u belongs to $\mathit{white}(\chi^*) \cap \mathit{white}({\chi_0}^*)$. By repeated application of Lemma 4.32, we find that u belongs to $\mathit{white}(\chi')$, and since $\alpha'(u) \geq 0$, we find that u belongs to $\mathit{white}(\mathit{top}'(\chi)) \cap \mathit{white}(\mathit{top}'(\chi_0))$.

Next we consider the case where u belongs to $\mathit{nonwhite}(\chi^*) \cap \mathit{nonwhite}({\chi_0}^*)$. It follows from the description of the bottom-level auction, that either u is unmatched in the same round of both bottom-level auction instances and hence u belongs to $\mathit{white}(\mathit{top}'(\chi)) \cap \mathit{white}(\mathit{top}'(\chi_0))$, or u remains matched throughout to the same item in both auction instances and $\mathit{top}'(\chi_0) = \mathit{subst}(\mathit{top}'(\chi), u, \beta)$.

Finally we look at the case where u either belongs to $\mathit{nonwhite}(\chi^*)$ or belongs to $\mathit{nonwhite}(\chi_0^*)$. Without loss of generality, assume that u belongs to $\mathit{nonwhite}(\chi^*)$. It follows from the description of the bottom-level auction, that either u is unmatched in some round of the auction instance with input $(\chi^*, \mathit{target}(\chi))$ and hence u belongs to $\mathit{white}(\mathit{top}'(\chi)) \cap \mathit{white}(\mathit{top}'(\chi_0))$, or u remains matched throughout in the auction instance with input $(\chi^*, \mathit{target}(\chi))$ and hence $\mathit{gap}(\mathit{top}'(\chi), u) = \mathit{gap}(\chi, u)$. It follows that $\mathit{top}'(\chi_0) = \mathit{subst}(\mathit{top}'(\chi), u, \beta)$. $\square$

Lemma 5.8. *For any configuration* χ_0 *of the form* $\mathrm{subst}(\chi, u, \beta)$*, if* $\beta \neq \mathrm{bid}(\mathrm{bid\text{-}graph}(\chi), u)$*, then either* $\mathrm{top}'_0(\chi_0) = \mathrm{subst}(\mathrm{top}'_0(\chi), u, \beta)$ *or* u *belongs to* $\mathrm{white}(\mathrm{top}'_0(\chi)) \cap \mathrm{white}(\mathrm{top}'_0(\chi_0))$.

Proof. The proof is identical to the proof of Lemma 5.7 for the case where u belongs to $\mathit{nonwhite}(\chi^*) \cap \mathit{nonwhite}({\chi_0}^*)$. $\square$

5.4 Truthfulness

A sealed-bid auction is said to be truthful if it is a weakly dominant strategy for every agent in the auction to bid truthfully. Formally, we say the first phase of the top-level auction is truthful if it satisfies the following condition: for any configuration χ and any agent u in $\mathit{agents}(\chi)$, if $\chi' = \mathit{subst}(\chi, u, \beta)$ for some bid β in $\mathit{bids}(\mathit{bid\text{-}graph}(\chi))$, then $\mathit{gap}(\mathit{top}(\chi), u) \geq \mathit{gap}(\chi'', u)$, where $\chi'' = \mathit{subst}(\mathit{top}(\chi'), u, \mathit{bid}(\chi, u))$.

For any configuration χ, we define $\mathit{top}'_0(\chi)$ as follows. Let χ_0 be a quiescent configuration and let α_0 be a target such that $\chi = \mathit{shift}(\chi_0, \alpha_0)$, $\mathit{white}(\chi_0) \cap \mathit{matched}(\chi_0) = \emptyset$, and for any agent u in $\mathit{unmatched}(\chi)$, we have $\mathit{items}(\chi_0, u) = \emptyset$. We define $\mathit{top}'_0(\chi)$ as the configuration $\mathit{shift}(\chi', \alpha')$, where $(\chi', \alpha') = \mathit{bottom}(\chi_0, \alpha_0)$. The uniqueness of $\mathit{top}'_0(\chi)$ is established by Lemma 5.5.

The auction that takes an configuration χ as input and produces $\mathit{top}'_0(\chi)$ as output does not immediately incorporate bid revision requests of tentatively allocated agents at the beginning of each round. We find it useful to first establish Lemma 5.11 on the truthfulness of $\mathit{top}'_0(\chi)$ in Section 5.4.1. We then use the claims of Section 5.4.1 to establish Lemma 5.21 on the truthfulness of the top-level auction.

5.4.1 Truthfulness of top'_0

The goal of this section is to establish Lemma 5.11 on the truthfulness of the auction that takes a configuration χ as input and produces the configuration $\mathit{top}''(\mathit{top}'_0(\chi))$ as output. We establish

Lemma 5.9 based on the claim of Lemma 4.36 of Section 4.5. Lemma 5.10 follows from Lemma 5.9. The proof of Lemma 5.11 follows from Lemma 5.10 and known results on the truthfulness of the TTC and $\mathrm{TC}^{\prec}$ algorithms.

Lemma 5.9. *For any configuration χ, any agent u in* agents(χ) *where* $\mathrm{bid}(\mathrm{bid\text{-}graph}(\chi), u) = \beta'$, *and any configuration χ' of the form* $\mathrm{subst}(\chi, u, \beta)$ *where β is a bid in* $\mathrm{bids}(\mathrm{bid\text{-}graph}(\chi))$*, we have*

$$\mathrm{gap}(\mathrm{top}_0'(\chi), u) \geq \mathrm{gap}(\mathrm{subst}(\mathrm{top}_0'(\chi'), u, \beta'), u).$$

Proof. Let $\chi = \mathit{shift}(\chi_0, \alpha_0)$, where χ_0 is a quiescent configuration and α_0 is a target such that $\mathit{white}(\chi_0) \cap \mathit{matched}(\chi_0) = \emptyset$, and for any agent u in $\mathit{unmatched}(\chi)$, we have $\mathit{items}(\chi_0, u) = \emptyset$. By definition, $\mathit{top}_0'(\chi) = \mathit{shift}(\chi_0', \alpha_0')$, where $(\chi_0', \alpha_0') = \mathit{bottom}(\chi_0, \alpha_0)$. Note that $\mathit{threshold}^*(\chi, \alpha_0) = \mathit{threshold}^*(\chi, \alpha_0)$. Similarly, let $\chi' = \mathit{shift}(\chi_1, \alpha_1)$, where χ_1 is a quiescent configuration and α_1 is a target such that $\mathit{white}(\chi_1) \cap \mathit{matched}(\chi_1) = \emptyset$, and $\mathit{items}(\chi_1, u) = \emptyset$ for any agent u in $\mathit{unmatched}(\chi)$. By definition, $\mathit{top}_0'(\chi') = \mathit{shift}(\chi_1', \alpha_1')$, where $(\chi_1', \alpha_1') = \mathit{bottom}(\chi_1, \alpha_1)$.

Let $\beta_T = \mathit{bid}(\mathit{bid\text{-}graph}(\chi), u)$ and let $\chi'' = \mathit{subst}(\chi_1', u, \beta_T)$. Assume that $\mathit{gap}(\mathit{top}_0'(\chi), u) < \mathit{gap}(\mathit{subst}(\mathit{top}_0'(\chi'), u, \beta_T), u)$; thus $\mathit{gap}(\chi_0', u) + \alpha_0'(u) < \mathit{gap}(\chi'', u) + \alpha_1'(u)$.

We first consider the case where u belongs to $\mathit{unmatched}(\chi)$. By repeated application of Lemma 4.32, we find that u belongs to $\mathit{white}(\chi_0') \cap \mathit{white}(\chi_1')$ and $\alpha_0'(u) = \alpha_1'(u) = 0$; thus, by our assumption, $\mathit{gap}(\chi_0', u) < \mathit{gap}(\chi'', u)$. Since $\mathit{gap}(\chi_0', u) < \mathit{gap}(\chi'', u)$ and u belongs to $\mathit{white}(\chi_0')$, we have $\mathit{gap}(\chi'', u) \geq 1$; thus u belongs to $\mathit{matched}(\chi_1')$. By Lemma 4.37, we choose to defer the *raise* invocations of u until a round in which u is the only remaining enabled agent with pending *raise* invocations. Let (χ_a, α_a) be the input of the first round in which u invokes the function *raise* in the bottom-level auction instance with input (χ_0, α_0), and let χ_a' be the configuration such that $\chi_a = \mathit{add}(\chi_a', u, \beta_T)$. Let (χ_b, α_b) be the input of the first round in which u invokes the function *raise* in the bottom-level auction instance with input (χ_1, α_1), and let ${\chi_b}'$ be the configuration such that $\chi_b = \mathit{add}(\chi_b', u, \beta)$. Let S' be the set of items v in $\mathit{items}(\chi)$ for which $\beta(v) - \mathit{threshold}^*(\chi_b', \alpha_b, v)$ is maximized. By Lemma 4.36, we have $\mathit{potential}(\chi_1', v) = \mathit{threshold}^*(\chi_b', \alpha_b, v)$; thus, $\mathit{gap}(\chi_1', u) = \beta(v) - \mathit{threshold}^*(\chi_b', \alpha_b, v)$ and $\mathit{gap}(\chi_1'', u) = \beta_T(v) - \mathit{threshold}^*(\chi_b', \alpha_b, v)$. By Lemma 4.36, $\mathit{gap}(\chi_0', u) = \mathit{max}_{v \in \mathit{items}(\chi)}(\beta_T(v) - \mathit{threshold}^*(\chi_a', \alpha_a, v))$; since $\mathit{threshold}^*(\chi_b', \alpha_b) = \mathit{threshold}^*(\chi_a', \alpha_a)$, we have $\mathit{gap}(\chi_0', u) \geq \mathit{gap}(\chi'', u)$; a contradiction. Thus, it follows that $\mathit{gap}(\mathit{subst}(\mathit{top}_0'(\chi'), u, \beta_T), u) \leq \mathit{gap}(\mathit{top}_0'(\chi), u)$.

Next we consider the case where agent u belongs to $\mathit{matched}(\chi)$. Since $\mathit{matched}(\chi_0) \cap \mathit{white}(\chi_0) = \mathit{matched}(\chi_1) \cap \mathit{white}(\chi_1) = \emptyset$, the definition of the function *shift* implies that there exists an item v in $\mathit{items}(\chi)$ such that $\mathit{match}(\chi_0, v) = \mathit{match}(\chi_1, v) = u$. Since u belongs to $\mathit{nonwhite}(\chi_0) \cap \mathit{nonwhite}(\chi_1)$, by the description of the bottom-level auction, either $\mathit{match}(\chi_0', v) = \mathit{match}(\chi_1', v) = u$, or u is unmatched in some round of the bottom-level auction instances with inputs (χ_0, α_0) and (χ_1, α_1). In the case where $\mathit{match}(\chi_0', v) = \mathit{match}(\chi_1', v) = u$, it is easy to see that $\mathit{gap}(\mathit{top}_0'(\chi), u) \geq \mathit{gap}(\mathit{subst}(\mathit{top}_0'(\chi'), u, \beta_T), u)$. In the case where u is unmatched in some round of the bottom-level auction instances with inputs (χ_0, α_0) and (χ_1, α_1), the analysis is similar to the previous case in which u belongs to $\mathit{unmatched}(\chi)$.

Thus, we have $\mathit{gap}(\mathit{top}_0'(\chi), u) \geq \mathit{gap}(\mathit{subst}(\mathit{top}_0'(\chi'), u, \mathit{bid}(\mathit{bid\text{-}graph}(\chi), u)), u)$. ☐

Lemma 5.10. *Any auction that takes an configuration χ as input and produces the configuration* $\mathrm{top}_0'(\chi)$ *as output is truthful.*

Proof. Follows from Lemma 5.9 and the definition of truthfulness. ☐

Lemma 5.11. *Any auction that takes an configuration χ as input and produces the configuration* $\mathrm{top}''(\mathrm{top}_0'(\chi))$ *as output is truthful.*

Proof. For any configuration χ_0, let $f(\chi_0)$ denote $top''(top'_0(\chi_0))$. Consider any instance of the top-level auction with configuration χ as input and let u be an agent in $agents(\chi)$. Let $\beta = bid(bid\text{-}graph(\chi), u)$ and let $\beta_T \neq \beta$ be the truthful bid of u. Let $\chi_T = subst(\chi, u, \beta_T)$. We are required to show that $gap(subst(f(\chi), u, \beta_T), u) \leq gap(f(\chi_T), u)$. By Lemma 5.8, it follows that either agent u belongs to $white(top'_0(\chi)) \cap white(top'_0(\chi_T))$ or $top'_0(\chi_T) = subst(top'_0(\chi), u, \beta_T)$.

First, we consider the case where u belongs to $white(top'_0(\chi)) \cap white(top'_0(\chi_T))$. By Fact 5.1, we have $potential(f(\chi)) = potential(top'_0(\chi))$ and u belongs to $white(f(\chi))$; thus we have $gap(f(\chi), u) = gap(top'_0(\chi), u)$ and $gap(f(\chi_T), u) = gap(top'_0(\chi_T), u)$. By Lemma 5.9 that $gap(subst(top'_0(\chi), u, \beta_T), u) \leq gap(top'_0(\chi_T), u)$; thus, we have $gap(subst(f(\chi), u, \beta_T), u) \leq gap(\chi_T, u)$.

Next, we consider the case where $top'_0(\chi_T) = subst(top'_0(\chi), u, \beta_T)$. By Lemma 5.6, the second phase of the top-level auction is truthful; thus, we have $gap(subst(f(\chi), u, \beta_T), u) \leq gap(f(\chi_T), u)$. Thus, any auction that takes an configuration χ as input and produces the configuration $f(\chi) = top''(top'_0(\chi))$ as output is truthful.

□

5.4.2 Truthfulness of the top-level auction

The goal of this section is to establish Lemma 5.21 on the truthfulness of the the top-level auction. We first establish that for any configuration χ and any white agent u in $top'(\chi)$, agent u has the same utility in $top'(\chi)$ as it does in $top'_0(\chi)$ (see Lemma 5.18). Lemmas 5.19 and 5.20 follow easily from Lemma 5.18. We use Lemmas 5.20 and known results on the truthfulness of the TTC algorithm to establish Lemma 5.21.

Lemma 5.12. *For any configuration* χ' *of the form* $\text{subst}(\chi, u, u')$*, if* u *is an agent in* $\text{matched}(\chi) \cap \text{white}(\chi)$*, then* $\text{gap}(\text{top}'(\chi), u) = \text{gap}(\text{top}'(\chi'), u')$*.*

Proof. Let $target(\chi) = \alpha$ and let $target(\chi') = \alpha'$. We first show that $\alpha = \alpha'$. Since u belongs to $matched(\chi) \cap white(\chi)$, we have $\alpha(u) = 0$, and since α is the pointwise minimum target in $targets(\chi)$ and u' does not belong to $agents(\chi)$, we have $\alpha(u') = 0$. Similarly, we have $\alpha'(u') = \alpha'(u) = 0$. Since α and α' are the pointwise minimum targets in $targets(\chi)$ and $targets(\chi')$ respectively, we have $\alpha(u'') = \alpha'(u'')$ for any agent u'' in $agents(\chi) - u$. It follows that $\alpha = \alpha'$.

Let $top'(\chi) = shift(\chi'_0, \alpha'_0)$ where $(\chi'_0, \alpha'_0) = bottom(\chi_0, \alpha)$ and let $top'(\chi') = shift(\chi'_1, \alpha'_1)$ where $(\chi'_1, \alpha'_1) = bottom(\chi_1, \alpha)$. By Lemma 4.34, we have $gap(\chi_0, u) = gap(\chi_1, u')$. Further, since $\alpha(u) = \alpha(u') = 0$, we have $\alpha'_0(u) = \alpha'_1(u') = 0$. Thus, $gap(top'(\chi), u) = gap(top'(\chi'), u')$. □

Lemma 5.13. *Let* χ *be a configuration and let* χ_A *be the quiescent configuration such that* $\chi = \text{shift}(\chi_A, \text{target}(\chi))$*. Let* u *be an agent in* $\text{matched}(\chi) \cap \text{white}(\chi)$*, and let* z *be an integer such that* u *belongs to* $\text{gray}(\text{shift}(\chi_A, u, -z))$*. If* u *is the minimum agent in* $\text{agents}(\chi)$*, then* $\text{gap}(\text{top}'(\chi), u) = \text{gap}(\chi_B, u) + \alpha_B(u)$*, where* $(\chi_B, \alpha_B) = \text{bottom}(\text{shift}(\chi'_A, u, -z), \text{shift}(\text{target}(\chi), u, z))$*.*

Proof. Let $(\chi'_B, \alpha'_B) = bottom(\chi_A, target(\chi))$. We refer to the executions of the bottom-level auction with inputs $(shift(\chi'_A, u, -z), shift(target(\chi), u, z))$ and $(\chi_A, target(\chi))$ as executions R and R' respectively. Let (χ_i, α_i) and (χ'_i, α'_i) be the outputs of round i of executions R and R' respectively. By Lemma 4.37, the *raise* invocations of enabled agents commute. Thus, by repeated application of Lemma 4.37, executions R and R' can be reordered such that for any round i, if S is a nonempty set of agents in $enabled(\chi_i) \cap enabled(\chi'_i)$ such that $\alpha_i(u') = \alpha'_i(u') > 0$, then some agent u' in S invokes *raise* in round $i + 1$ of executions R and R'. For any round i, we define the predicate $P(i)$ to hold if $(\chi'_i, \alpha'_i) = (shift(\chi_i, u, -z), shift(\alpha_i, u, z))$.

First we consider the case where $P(i)$ holds for every round of executions R and R'. In this case, we have $(\chi'_B, \alpha'_B) = (shift(\chi_B, u, -z), shift(\alpha_B, u, z))$. Thus, we have $gap(top'(\chi), u) = gap(\chi'_B, u)+$

$\alpha'_B = gap(\chi_B, u) - z + \alpha_B(u) + z$. Since u belongs to $white(top(\chi))$, we have $\alpha'_B(u) = 0$. Thus, $gap(top'(\chi), u) = gap(\chi'_B, u) + \alpha'_B$.

Next we consider the case where there exists a first round k such that $P(k)$ does not hold. In this case, it is easy to see that either u belongs to $unmatched(\chi_k)$ or u belongs to $unmatched(\chi'_k)$; further, since u belongs to $nonwhite(\chi'_k)$, we find that u belongs to $unmatched(\chi_k)$. Let u' be the agent in $matched(\chi'_{k-1}) \cap unmatched(\chi'_k)$. We now allow u to exhaust all its *raise* invocations in rounds $(k+1)\cdots(k+z)$ of execution B; thus, we have $\alpha_B(u) = 0$. Since u belongs to $white(top'(\chi))$, we have $agents(\chi'_j, u) \neq \emptyset$ for some round j of execution B where $(k+1) \leq j \leq (k+z)$, and thus, $u' = victim(\chi'_j, u)$. It is straightforward to see that for any round $j > k$ of execution R, we have $\chi_j = \chi_{j+z}$; thus, $\chi'_B = \chi_B$. Since $\alpha(u) = 0$, we have $\alpha'_B(u) = 0$, and we established above that $\alpha_B(u) = 0$. Thus, $gap(top'(\chi), u) = gap(\chi_B, u) = gap(\chi_B, u) + \alpha_B(u_0)$. □

For any configuration χ, any agent u in $matched(\chi) \cap white(\chi)$, and any agent u' such that $u < u'$ and there exists no agent u'' in $agents(\chi)$ such that $u' < u'' < u$, we define $split(\chi, u, u')$ as the configuration $add(shift(\chi, u, -z), u', \beta)$ where z is the integer such that $max\text{-}gap(shift(\chi, u, -z), u) = -1$, and $\beta = bid(bid\text{-}graph(\chi), u)$.

Lemma 5.14. *Let* χ' *be a configuration of the form* $\mathrm{split}(\chi, u, u')$ *and let* $\chi = \mathrm{shift}(\chi_A, \mathrm{target}(\chi))$. *For any integer* z *such that* u *belongs to* $\mathrm{gray}(\mathrm{shift}(\chi_A, u, -z))$*, we have* $\mathrm{gap}(\mathrm{top}'(\chi'), u') = \mathrm{gap}(\chi_B, u) + \alpha_B(u)$ *where* $(\chi_B, \alpha_B) = \mathrm{bottom}(\mathrm{shift}(\chi_A, u, -z), \mathrm{shift}(\mathrm{target}(\chi), u, z))$.

Proof. Let $\chi'' = shift(\chi_A, u, -z^*)$, where z^* is the integer such that $max\text{-}gap(\chi'', u) = -1$. Let $\alpha = target(\chi)$. By the definition of the function *split*, we find that u belongs to $matched(\chi) \cap white(\chi)$; thus, $\alpha(u) = 0$. Note that u belongs to $white(shift(\chi'', u, 1))$; thus, $z^* \leq z$. By repeated application of Fact 4.5, it follows that $(\chi_B, \alpha_B) = bottom(\chi'', shift(target(\chi), u, z^*))$. Let χ'_A be the quiescent configuration such that $\chi' = shift(\chi'_A, target(\chi'))$ and let $(\chi'_B, \alpha'_B) = bottom(\chi'_A, target(\chi'))$.

We refer to the bottom-level auction instance with inputs $(\chi'', shift(target(\chi), u, z^*))$ as execution A and we refer to the bottom-level auction instance with inputs $(\chi'_A, target(\chi'))$ as execution B. By Lemma 4.37, we can assume that the same agent invokes *raise* in both executions whenever possible. From the description of the bottom-level auction, we find that either u becomes unmatched in the same round of both executions, or u remains matched in both executions until u' is the only remaining enabled agent with pending *raise* invocations in execution B.

We first consider the case where u is unmatched in the same round of both executions. In this case, we immediately process the *raise* invocations of agent u in execution A and agent u' in execution B. If χ_B and χ'_B are the resultant configurations of executions A and B after agents u and u' have exhausted their *raise* invocations, then by Lemma 4.35, we have $gap(\chi'_B, u') + \alpha'_B(u') = gap(\chi_B, u) + \alpha_B(u)$, and the proof is complete.

Next we consider the case where u remains matched to some item v in both executions, all enabled agents have exhausted their *raise* invocations in execution A, and u' is the only remaining enabled agent with pending *raise* invocations in execution B. We now allow agent u' to exhaust its *raise* invocations. While u remains allocated for the rest of execution B, the potential of v remains unchanged; thus u' attains its highest utility of $gap(\chi_B, u) + \alpha_B(u)$. If u is unmatched by some *raise* invocation, then the auction terminates with the potential of item v unchanged, and with u' attaining its highest utility of $gap(\chi_B, u) + \alpha_B(u)$. Thus, we have $gap(top'(\chi'), u') = gap(\chi_B, u) + \alpha_B(u)$ □

Lemma 5.15. *Let* χ_0 *be an configuration of the form* $\mathrm{subst}(\chi, u, u_0)$ *where* u *is an agent in* $\mathrm{white}(\chi) \cap \mathrm{matched}(\chi)$*, and* u_0 *is an agent such that* $u_0 > u$ *and there exists exactly one agent* u''

in agents(χ) *such that* $u < u'' < u_0$. *Then* $\text{gap}(\text{top}'(\chi'), u') = \text{gap}(\text{top}'(\chi'_0), u'_0)$ *where* χ' *is any configuration of the form* $\text{split}(\chi, u, u')$ *and* χ'_0 *is any configuration of the form* $\text{split}(\chi_0, u_0, u'_0)$.

Proof. Let χ_A be the quiescent configuration such that $\chi' = \mathit{shift}(\chi_A, \mathit{target}(\chi'))$ and let χ_B be the quiescent configuration such that $\chi'_0 = \mathit{shift}(\chi_B, \mathit{target}(\chi'_0))$. We refer to the executions of the bottom-level auctions with inputs $(\chi_A, \mathit{target}(\chi'))$ and $(\chi_B, \mathit{target}(\chi'_0))$ as executions A and B respectively. By Lemma 4.37, the *raise* invocations by enabled agents commute. Thus, we choose to defer the *raise* invocations of agents in the set $S = \{u', u'_0, u''\}$ in executions A and B until S is the only set of enabled agents. Further, we commute *raise* invocations in both executions such that whenever possible, the same agent invokes *raise* in each round.

For any nonnegative integer i, let χ_i and χ'_i be the output configurations of round i of executions A and B respectively. We define the predicate $\mathit{sync}(A, B, i)$ to hold if $\chi'_i = \mathit{subst}(\mathit{subst}(\chi_i, u, u_0), u', u'_0)$. We define the predicate $\mathit{coupled}(A, B, i)$ to hold if there exists exactly one maximal set of items V_i in $\mathit{items}(\chi)$ and agents u_a in $\mathit{matched}(\chi_i) \cap \mathit{unmatched}(\chi'_i)$ and u_b in $\mathit{unmatched}(\chi_i) \cap \mathit{matched}(\chi'_i)$ such that for any agent u_c in $\mathit{unmatched}(\chi_i) \cap \mathit{unmatched}(\chi'_i)$ such that $\mathit{items}(\chi_i, u^*) \cap V_i \neq \emptyset$, we have $\mathit{victim}(\chi_i, u_c, 1) = u_a$ and $\mathit{victim}(\chi'_i, u_c, 1) = u_b$.

Consider the first round j such that $\mathit{coupled}(A, B, j)$ holds; then, by the definition of the bottom-level auction, it is easy to see that either: (1) execution A evicts u' in round j and execution B evicts u'' in round j, or (2) execution A evicts u in round j and execution B evicts u'' in round j. In case (1), since u' is evicted by execution A in round j, and $\mathit{sync}(A, B, j-1)$ holds, agents u' and u'_0 have zero utility in round j, and by the definition of the bottom-level auction, u' and u'_0 continue to have zero utility for the rest of the auction; this completes the proof for case (1).

We now consider case (2). Consider each round $i > j$ of executions A and B where some agent u_1 in $\mathit{agents}(\chi) - \{u''\}$ invokes the function *raise*. If $\mathit{items}(\chi_i, u_1) \cap V_i \neq \emptyset$, then it follows that $\mathit{coupled}(A, B, i+1)$ holds. Consider the first round $k > j$ in which some agent u_1 invokes *raise* and $\mathit{victim}(\chi_k, u_1, 1) = u''$, thus we have $\mathit{victim}(\chi'_k, u_1, 1) = u_0$. and $\mathit{sync}(A, B, k+1)$ holds; further, it is easy to see that $\mathit{sync}(A, B, i)$ holds for every round $i > k$ in which some agent in $\mathit{agents}(\chi) \setminus S$ invokes the function *raise*.

We now look at executions A and B in a round k when u'' is the only enabled agent with pending *raise* invocations. We consider two cases. We first consider the case where $\mathit{sync}(A, B, k-1)$ holds; in this case, u'' is the only enabled agent with pending *raise* invocations in both executions A and B; thus for any $i \geq k$, if $\mathit{coupled}(A, B, i)$ holds, then condition (1) holds where execution A evicts u' and execution B evicts u'' and the proof follows from the analysis of case (1) discussed above. Next, we consider the case where $\mathit{coupled}(A, B, k-1)$ holds; in this case execution A has terminated, and u'' is the only enabled agent with pending *raise* invocations in execution B, and agents u'' and u' are matched in executions A and B respectively. While u'' and u' remains allocated for the rest of executions A and B, the potentials of items on P remain unchanged; thus u' and u'_0 both attain zero utility. If u' is unmatched by some *raise* invocation of execution B, then execution B terminates and thus, agents u' and u'_0 attain zero utility. □

Lemma 5.16. *Let* χ_0 *be an configuration of the form* $\text{subst}(\chi, u, u_0)$ *where* u *is an agent in* $\text{white}(\chi) \cap \text{matched}(\chi)$. *Then* $\text{gap}(\text{top}'(\chi'), u') = \text{gap}(\text{top}'(\chi'_0), u'_0)$ *where* χ' *is any configuration of the form* $\text{split}(\chi, u, u')$ *and* χ'_0 *is any configuration of the form* $\text{split}(\chi_0, u_0, u'_0)$.

Proof. Without loss of generality, we can assume that $u < u_0$. If there is no agent u_1 in $\mathit{agents}(\chi)$ such that $u < u_1 < u_0$, then the result follows by repeated application of Lemma 4.35. If there is an agent u_1 in $\mathit{agents}(\chi)$ such that $u < u_1 < u_0$, then the result follows by induction using Lemma 5.15. □

Lemma 5.17. *For any configuration* χ *and any agent* u *in* $\mathrm{matched}(\chi) \cap \mathrm{white}(\chi)$*, if configuration* χ' *is of the form* $\mathrm{subst}(\chi, u, u')$*, then* $\mathrm{gap}(\mathrm{top}'(\chi), u) = \mathrm{gap}(\mathrm{top}'(\chi'), u')$*.*

Proof. Let χ_0 be an configuration of the form $split(\chi, u, u_0)$ and let χ_1 be an configuration of the form $split(\chi', u', u_1)$. By Lemma 5.14, we have $gap(top'(\chi), u) = gap(top'(\chi_0), u_0)$ and $gap(top'(\chi'), u') = gap(top'(\chi_1), u_1)$. Since $\chi' = subst(\chi, u, u')$, by Lemma 5.16, we have $gap(top'(\chi_0), u_0) = gap(top'(\chi_1), u_1)$. Thus, we have $gap(top'(\chi), u) = gap(top'(\chi'), u')$. □

Lemma 5.18. *For any configuration* χ *such that* $\chi = \mathrm{shift}(\chi_0, \mathrm{target}(\chi))$*, any agent* u *in* $\mathrm{matched}(\chi) \cap \mathrm{white}(\chi)$*, and any integer* z *such that* u *belongs to* $\mathrm{gray}(\mathrm{shift}(\chi_0, u, -z))$*, we have* $\mathrm{gap}(\mathrm{top}'(\chi), u) = \mathrm{gap}(\chi_1, u) + \alpha_1(u)$ *where* $(\chi_1, \alpha_1) = \mathrm{bottom}(\mathrm{shift}(\chi_0, u, -z), \mathrm{shift}(\mathrm{target}(\chi), u, z))$*.*

Proof. Let u' be an agent such that $u' < u''$ for every agent u'' in $agents(\chi)$. Let χ'_1 be a configuration and let α'_1 be a target such that $(\chi'_1, \alpha'_1) = bottom(shift(subst(\chi_0, u, u'), u', -z), shift(target(\chi), u', z))$. By Lemma 5.13, we have $gap(\chi'_1, u') + \alpha'_1(u') = gap(top'(subst(\chi, u, u')), u')$.

By Lemma 5.17, we have $gap(\chi_1, u) + \alpha_1(u) = gap(\chi'_1, u') + \alpha'_1(u')$. By Lemma 5.12, we have $gap(top'(\chi), u) = gap(top'(subst(\chi, u, u')), u')$. Thus, we have $gap(top'(\chi), u) = gap(\chi_1, u) + \alpha_1(u)$. □

Lemma 5.19. *For any configuration* χ*, any agent* u *in* $\mathrm{agents}(\chi)$*, and any configuration* χ' *of the form* $\mathrm{subst}(\chi, u, \beta)$ *where* β *is a bid in* $\mathrm{bids}(\text{bid-graph}(\chi))$*, we have*

$$\mathrm{gap}(\mathrm{top}'(\chi), u) \geq \mathrm{gap}(\mathrm{subst}(\mathrm{top}'(\chi'), u, \mathrm{bid}(\text{bid-graph}(\chi), u)), u).$$

Proof. The analysis for agents in $unmatched(\chi)$ is identical to the analysis for unmatched agents in the proof of Lemma 5.9, and the analysis for agents in $matched(\chi) \cap nonwhite(\chi)$ is identical to the analysis for matched nonwhite agents in the proof of Lemma 5.9.

We now consider any agent u in $matched(\chi) \cap white(\chi)$. Let $\chi = shift(\chi_0, target(\chi))$ and let z be any integer such that u belongs to $gray(shift(\chi_0, u, -z))$. By Lemma 5.18, we have $gap(top'(\chi), u) = gap(\chi_1, u) + \alpha_1(u)$ where $(\chi_1, \alpha_1) = bottom(shift(\chi_0, u, -z), shift(target(\chi), u, z))$. Thus, u obtains the same utility as it would have obtained if its bid had been shifted down sufficiently to make u gray; it follows that we can restrict attention to bottom-level auction instances that have no white matched agents in their input configurations. Let $\chi = shift(\chi_1, \alpha_1)$, where χ_1 is the configuration obtained by shifting down the bid of every white agent in χ_0 such that $white(\chi_1) \cap matched(\chi_1) = \emptyset$. If $(\chi'_1, \alpha'_1) = bottom(\chi_1, \alpha_1)$, then by definition, we have $top'_0(\chi) = shift(\chi'_1, \alpha'_1)$. The proof now follows from Lemma 5.9. □

Lemma 5.20. *The first phase of the top-level auction is truthful.*

Proof. Follows from Lemma 5.19 and the definition of truthfulness. □

We use Lemma 5.20 and Lemma 5.6 on the truthfulness of the second phase to establish Lemma 5.21.

Lemma 5.21. *The top-level auction is truthful.*

Proof. By Lemma 5.20 and Lemma 5.6, the first and second phases of the top-level auction are individually truthful. We now show that the top-level auction which combines the two phases is truthful. Consider any instance of the top-level auction with configuration χ as input and let u be an agent in $agents(\chi)$. Let $\beta = bid(\textit{bid-graph}(\chi), u)$ and let $\beta_T \neq \beta$ be the truthful bid of u. Let $\chi_T = subst(\chi, u, \beta_T)$. We wish to show that $gap(subst(top(\chi), u, \beta_T), u) \leq gap(top(\chi_T), u)$. By Lemma 5.7, either u belongs to $white(top'(\chi)) \cap white(top'(\chi_T))$ or $top'(\chi_T) = subst(top'(\chi), u, \beta_T)$.

First, we consider the case where u belongs to $white(top'(\chi)) \cap white(top'(\chi_T))$. By Fact 5.1, we have $potential(top(\chi)) = potential(top'(\chi))$ and u belongs to $white(top(\chi))$; thus $gap(top(\chi), u) = gap(top'(\chi), u)$ and $gap(top(\chi_T), u) = gap(top'(\chi_T), u)$. Further, it follows from Lemma 5.19 that $gap(top'(\chi), u) \leq gap(subst(top'(\chi_T), u, \beta_T), u)$. Thus, we conclude that $gap(top(\chi), u) \leq gap(subst(top(\chi_T), u, \beta_T), u)$.

Next, we consider the case where $top'(\chi_T) = subst(top'(\chi), u, \beta_T)$. By Lemma 5.6 on the truthfulness of the second phase of the top-level auction, we have $gap(top(\chi), u) \leq gap(subst(top(\chi_T), u, \beta_T), u)$. □

5.5 Efficiency

Recall that put options impose lower bound constraints on item prices. As a result, we cannot in general achieve efficiency in our auction setting. Lemmas 5.24 and 5.25 establish a relaxed form of efficiency for our auction — the outcome is efficient if the target of every put that is exercised satisfies envy-freedom.

For any configuration χ such that $unmatched(\chi) \subseteq white(\chi)$ and any agent u in $nonwhite(\chi)$, we define $admissible(\chi, u)$ as the set of all bids β in $bids(bid\text{-}graph(\chi))$ such that u belongs to $white(subst(\chi, u, \beta))$. For any configuration χ such that $unmatched(\chi) \subseteq white(\chi)$, we define $admissible(\chi)$ as the set of all possible configurations that can be obtained from χ by replacing the bid of every agent u in $black(\chi)$ by a bid in $admissible(\chi, u)$.

Lemma 5.22. *For any configuration χ and any agent u in* $\text{matched}(\text{top}(\chi)) \cap \text{nonwhite}(\text{top}(\chi))$*, if u belongs to* $\text{white}(\text{subst}(\chi, u, \beta))$ *for some bid β in* $\text{bids}(\text{bid-graph}(\chi))$*, then u belongs to* $\text{white}(\text{subst}(\text{top}(\chi), u, \beta))$.

Proof. By Lemma 5.3, there exists an item v in $items(\chi)$ such that $potential(top(\chi), v) = potential(\chi, v)$ and $match(\chi, v) = match(top(\chi), v) = u$. Let β be any bid in $admissible(\chi, u)$. By definition, $\beta(v) - potential(\chi, v) \geq \beta(v') - potential(\chi, v')$ for any item v' in $items(\chi)$. By Lemma 5.2, it follows that $potential(top(\chi), v') \geq potential(\chi, v)$. Thus, $\beta(v) - potential(top(\chi), v) \geq \beta(v') - potential(top(\chi), v')$ for any item v' in $items(\chi)$. Thus, β belongs to $admissible(top(\chi), u)$. □

Lemma 5.23. *If χ is an configuration such that* $\text{unmatched}(\chi) \subseteq \text{white}(\chi)$*, then any configuration in* $\text{admissible}(\chi)$ *is white.*

Proof. Let χ' be any configuration in $admissible(\chi)$. By the definition of $admissible(\chi)$, for every agent u in $nonwhite(\chi)$, we have $bid(\chi', u)$ belongs to $admissible(\chi, u)$; thus we have $nonwhite(\chi) \subseteq white(\chi')$. Further, for any agent u in $white(\chi)$, we have $bid(\chi, u) = bid(\chi', u)$, and thus u belongs to $white(\chi')$. Thus, configuration χ' is white. □

Lemma 5.24. *For any configuration χ and any agent u in* $\text{agents}(\chi)$*, if u belongs to* $\text{nonwhite}(\text{top}(\chi))$*, then u belongs to* $\text{nonwhite}(\chi)$ *and* $\text{admissible}(\chi, u) \subseteq \text{admissible}(\text{top}(\chi), u)$.

Proof. By Lemma 5.2, we have $nonwhite(top(\chi)) \subseteq nonwhite(\chi)$. Thus, u belongs to $nonwhite(\chi)$. Let β be any bid in $admissible(\chi, u)$; then by definition, u belongs to $white(subst(\chi, u, \beta))$. By Lemma 5.22, u belongs to $white(subst(top(\chi), u, \beta))$. Thus, β belongs to $admissible(top(\chi), u)$. □

Lemma 5.25. *For any configuration χ, we have* $\text{unmatched}(\text{top}(\chi)) \subseteq \text{white}(\text{top}(\chi))$*, and every configuration in* $\text{admissible}(\text{top}(\chi))$ *is efficient.*

Proof. By definition, we have $top(\chi) = top''(top'(\chi))$. By Fact 5.1, $unmatched(top'(\chi)) = unmatched(top(\chi))$ and $potential(top(\chi)) = potential(top'(\chi))$, and by Lemma 5.2 we have $unmatched(top'(\chi)) \subseteq white(top'(\chi))$; thus $unmatched(top(\chi)) \subseteq white(top(\chi))$.

Let $\chi' = (G, M, \Phi)$ be any configuration in $admissible(top(\chi))$. By Lemma 5.23, χ' is white, and by Lemmas 3.12, χ' is Walrasian; thus, it follows from Lemma 3.2 that M is an MWMCM of G. Thus, χ' is efficient. □

We define a sealed-bid auction to be Pareto-efficient if it is truthful — so that no agent has an incentive to lie — and it satisfies the strong version of Condition 5. Lemmas 5.26 and 5.27 establish the strong and weak versions of equilibrium condition 5 of Section 2.4.

Lemma 5.26. *The top-level auction is Pareto-efficient when the second phase of the auction is implemented using the* $TC^{\prec}$ *algorithm.*

Proof. Consider any configuration χ and let $\chi' = top(\chi)$. Suppose by way of contradiction that there is a nonempty set of agents U_0 who can trade their allocated items amongst themselves such that every agent in U_0 experiences an increase in utility. By definition, for any agent u in $white(\chi')$ and any item v in $items(\chi)$, we have $gap(\chi', u) \geq \beta(v) - potential(\chi', v)$, where $\beta = bid(bid\text{-}graph(\chi), u)$. Thus, $white(\chi') \cap U_0 = \emptyset$. It follows that $U_0 \subseteq nonwhite(\chi')$; this contradicts the Pareto-efficient property of the $\text{TC}^{\prec}$ algorithm. Thus, $U_0 = \emptyset$ and the top-level auction is Pareto-efficient. □

Lemma 5.27. *The top-level auction produces an outcome in the weak core when the second phase of the auction is implemented using the TTC algorithm.*

Proof. Consider any configuration χ and let $\chi' = top(\chi)$. Suppose by way of contradiction that there is a nonempty set of agents U_0 who can trade their allocated items amongst themselves such that every agent in U_0 experiences an increase in utility. By definition, for any agent u in $white(\chi')$ and any item v in $items(\chi)$, we have $gap(\chi', u) \geq \beta(v) - \Phi'(v)$, where $\beta = bid(bid\text{-}graph(X), u)$ and $\chi' = (G, M', \Phi')$. Thus, $white(\chi') \cap U_0 = \emptyset$. It follows that $U_0 \subseteq nonwhite(\chi')$; this is a contradiction to the well established property that in the absence of strict preferences, the TTC algorithm produces an outcome in the weak core. Thus, $U_0 = \emptyset$. □

5.6 Privacy Preservation

A motivating application of the sealed-bid unit-demand auction proposed in this paper is the design of a dynamic unit-demand auction in which each round is implemented using the proposed sealed-bid auction. If the seller of an item in the dynamic auction has access to the maximum price that an agent who is tentatively allocated to the item is willing to pay for the item, then the seller can extract this price by submitting a "shill" offer just below the agent's offer. Many dynamic auctions including the popular eBay auction suffer from shill bidding [14]. Thus, a goal of our proposed dynamic auction is to ensure bid privacy for tentatively allocated agents. Below we establish Lemmas 5.28 and 5.29 which are useful in showing a certain privacy preserving property of the dynamic auction — no seller can artificially raise the price of an item by more than one unit without risking forfeiture of sale. Ideally, we would want to prevent a seller from raising the price of an item even by a single unit; however, our adoption of tie-breaking to handle degeneracy limits our auction to giving up one unit in being shill proof.

Consider an agent u with bid β, and let u be white and allocated in an outcome of the top-level auction. Lemma 5.28 establishes that the outcome of the top-level auction remains unchanged for any bid β' of u that exceeds β in all its components. Lemma 5.29 establishes that if u has a positive utility, then the outcome of the top-level auction remains unchanged if u drops all components of its bid by one unit. It follows from Lemmas 5.28 and 5.29 that the seller of an item cannot deduce if the offer of an agent who wins the item exceeds the price of the item by more than one unit.

Lemma 5.28. *For any configuration* χ *and any agent* u *in* $\text{matched}(\text{top}(\chi)) \cap \text{white}(\text{top}(\chi))$*, we have* $\text{top}(\text{shift}(\chi, u, 1)) = \text{shift}(\text{top}(\chi), u, 1)$.

Proof. Let $\chi' = \textit{shift}(\chi, u, 1)$. Let $\chi = \textit{shift}(\chi_A, \textit{target}(\chi))$ and let $\chi' = \textit{shift}(\chi'_A, \textit{target}(\chi'))$. We refer to the execution of the bottom-level auction with inputs χ_A and $\textit{target}(\chi)$ as execution R and we refer to the execution of the bottom-level auction with inputs χ'_A and $\textit{target}(\chi')$ as execution S. Let (χ_i, α_i) and (χ'_i, α'_i) be the outputs of round i of executions R and S respectively. Let S be the sequence of agents where the ith element of sequence S, denoted $S(i)$, is the agent that invoked *raise* in round i of execution R. Similarly, we define S' to be the sequence of agents that invoked *raise* in execution S. Let j be the round in which u makes its last *raise* invocation in execution R.

We first claim that u has the same color in configurations χ and χ' and that u is not gray in either configuration. If u belongs to $\textit{gray}(\chi)$, then by the definition of the top-level auction, u belongs to $\textit{gray}(\chi_0)$ and $\alpha(u) = 0$, where $\alpha = \textit{target}(\chi)$; thus, u either belongs to $\textit{gray}(\textit{top}(\chi))$ or u belongs to $\textit{unmatched}(\textit{top}(\chi))$, which is a contradiction. Thus, u does not belong to $\textit{gray}(\chi)$. Since $\chi' = \textit{shift}(\chi, u, 1)$, we find that u does not belong to $\textit{gray}(\chi')$; it is thus straightforward to argue that u has the same color in configurations χ and χ'.

We now show that (1) for all $i \leq j$, we have $S(i) = S'(i)$ and $\chi_i = \chi'_i$, and (2) $S'_{j+1} = u$.

By definition of the top-level auction, $\chi'_0 = \textit{shift}(\chi_0, u, 1)$. Further, we established above that agent u is non-gray has the same color in configurations χ_0 and χ'_0. Thus, it follows from the definition of the determinized bottom-level auction that for all $i \leq j$, we have $S(i) = S'(i)$ and $\chi_i = \chi'_i$. Since u belongs to $\textit{matched}(\textit{top}(\chi))$ and u invoked its last *raise* in round j, we find that u belongs to $\textit{matched}(\chi_j) \cap \textit{white}(\chi_j)$. Further, since $\chi_j = \chi_{j'}$, we find that u belongs to $\textit{matched}(\chi'_j) \cap \textit{white}(\chi'_j)$ and $\textit{enabled}(\chi_j) = \textit{enabled}(\chi'_j)$. Since $S(j) = u$, and by the definition of the function *raise*, no matched agents were enabled in round j of both executions, we find that u belongs to $\textit{enabled}(\chi_{j+1})$; it follows from these facts that $S'(j+1) = u$.

Next we show that $S(i) = S'(i+1)$ for any $i > j$. We established above that u belongs to $\textit{matched}(\chi'_j)$ and that u makes its last *raise* invocation of execution S in round $j+1$; thus $\alpha'_{j+1}(u) = 0$. By the definition of the function *raise*, we have $\chi'_{j+1} = \textit{shift}(\chi'_j, u, 1)$ and $\textit{enabled}(\chi'_j) = \textit{enabled}(\chi'_{j+1})$. Since $\chi_j = \chi'_j$, we have $\chi'_{j+1} = \textit{shift}(\chi_j, u, 1)$ and $\textit{enabled}(\chi'_{j+1}) = \textit{enabled}(\chi_j)$; further, we established that $\alpha'_{j+1}(u) = 0$; thus, $S(i) = S'(i+1)$ for $i \geq j$.

We now show that if $S'(j+2) = u'$, then $\textit{raise}(\textit{raise}(\chi'_j, u), u') = \textit{raise}(\textit{raise}(\chi'_j, u'), u)$. From the preceding claim, we have $S(j+1) = u'$. Since u belongs to $\textit{matched}(\textit{top}(\chi)) \cap \textit{white}(\textit{top}(\chi))$ and u invoked its last *raise* in round j of execution R, it follows that $u \neq \textit{victim}(\chi_{j+1}, u')$. Since $\chi'_{j+1} = \textit{shift}(\chi_j, u, 1)$ and $S'(j+2) = u'$, it follows that $u \neq \textit{victim}(\chi'_{j+1}, u')$; thus, by the definition of the determinized function *raise*, we have $\textit{raise}(\textit{raise}(\chi'_j, u), u') = \textit{raise}(\textit{raise}(\chi'_j, u'), u)$.

By repeated application of the preceding argument, the last *raise* invocation of u in execution S can be commuted to the last round k of execution S. Thus, it follows that $(\chi'_{k-1}, \alpha'_{k-1}) = (\chi_B, \textit{shift}(\alpha_B, u, 1))$, where $(\chi_B, \alpha_B) = \textit{bottom}(\chi_A, \textit{target}(\chi))$.

By the definition of the function *raise*, we have $\textit{bottom}(\chi_B, \textit{target}(\chi')) = (\textit{shift}(\chi_B, u, 1), \alpha_B)$. By the description of the second phase of the top-level auction, it follows that $\textit{top}(\chi') = \textit{shift}(\textit{top}(\chi), u, 1)$. □

Lemma 5.29. *For any configuration* χ *and any agent* u *in* $\text{agents}(\chi)$*, if* $\text{gap}(\text{top}(\chi), u) > 1$*, then we have* $\text{gap}(\text{top}(\chi_0), u) \geq 1$*, where* $\chi_0 = \text{shift}(\chi, u, -1)$.

Proof. Let $\chi = (G, M, \Phi)$. It follows that $\chi_0 = (G_0, M, \Phi)$, where $G_0 = \mathit{shift}(G, u, -1)$. Let $\beta = \mathit{bid}(G_0, u)$. Let $\mathit{top}(\chi_0) = (G_0, M_0', \Phi_0')$ and let $\mathit{top}(\chi) = (G, M', \Phi')$. We consider the following cases.

Suppose u belongs to $\mathit{white}(\mathit{top}(\chi_0)) \cap \mathit{matched}(\mathit{top}(\chi_0))$. By Lemma 5.28, it follows that $\mathit{top}(\chi) = \mathit{shift}(\mathit{top}(\chi_0), u, 1)$. Thus, $\mathit{gap}(\mathit{top}(\chi_0), u) \geq 1$.

Suppose u belongs to $\mathit{white}(\mathit{top}(\chi_0)) \cap \mathit{unmatched}(\mathit{top}(\chi_0))$. It follows that $\mathit{gap}(\mathit{top}(\chi_0), u) = 0$. Thus, $\beta(v) \leq \Phi_0'(v)$ for every item v in $\mathit{items}(\chi_0)$. However, since $\mathit{gap}(\mathit{top}(\chi), u) > 1$, there exists an item v in $\mathit{items}(\chi_0)$ such that $\mathit{match}(\mathit{top}(\chi), v) = u$ and $\beta(v) - \Phi'(v) \geq 1$. Thus, $\mathit{gap}(\mathit{top}(\chi_0), u) < \mathit{gap}(\chi'', u)$, where $\chi'' = \mathit{subst}(\mathit{top}(\chi), u, \beta)$. This contradicts Lemma 5.19.

Suppose u belongs to $\mathit{nonwhite}(\mathit{top}(\chi_0))$. By Lemma 5.4, u belongs to $\mathit{nonwhite}(\chi_0)$, and by Lemma 5.7, we have $\mathit{top}'(\chi_0) = \mathit{subst}(\mathit{top}'(\chi), u, \beta)$; thus, it follows from the definition of the second phase of the top-level auction that $\mathit{top}(\chi_0) = \mathit{subst}(\mathit{top}(\chi), u, \beta)$. It is now straightforward to see that since $\mathit{gap}(\mathit{top}(\chi), u) > 1$, we have $\mathit{gap}(\mathit{top}(\chi_0), u) \geq 1$. □

5.7 Scalability

In this section we briefly sketch a fast implementation of the top-level auction. We say that a bid component is "active" if it is at least equal to the price (viewing the bid components and prices as pairs, as in the discussion on white configurations in Section 3.5) of the corresponding item. We only need to maintain information concerning the active bid components. We first define an initial tentative pricing and allocation at the start of the auction: each item is allocated to the seller of its put and has a price equal to the strike price of its put. The agents that are not tentatively allocated do not have any active bid components, and so we do not need to maintain any information concerning such agents. We do not maintain an explicit color value (black, gray, or white) for each tentatively allocated agent. Instead, when we need to determine the color of an agent, we do so by examining its active bid components along with the current prices of the associated items.

We now iteratively process bids of unallocated agents. At the start of an iteration, our auction state specifies the current pricing and allocation, the target bid of each tentatively allocated agent, and a set of unallocated agents for which the associated bids have yet to be processed. We pick an arbitrary unallocated agent u from the latter set, and in the style of the well-known Hungarian algorithm for weighted bipartite matching [12], or the closely related successive shortest paths algorithm [2, Chapter 9], we proceed to update the tentative pricing and allocation to account for the bid of u. The high-level strategy is to grow a Hungarian tree (which involves increasing certain prices, while maintaining the allocation) rooted at u until one of the following two conditions occurs: (1) one or more nonwhite tentatively allocated agents enter the tree; (2) the utility of u or one or more of the white tentatively allocated agents drops to zero.

If (2) occurs before (1), then we update the allocation via an augmentation that unallocates (and discards) the minimum zero-utility agent, and allocates u. (If agent u is itself the minimum zero-utility agent, then no augmentation is performed, and the allocation remains unchanged.) Using a standard primal-dual approach, it is possible to update the pricing and allocation in time proportional to the time required to solve a single-source shortest paths (SSSP) problem on the active subgraph of the current bid-graph. For a directed graph with n vertices and m edges, Thorup presents an $O(m + n \log \log n)$ algorithm for the SSSP problem [15]. Thus the time complexity of the update is close to linear in the number of active bid components.

If (1) occurs before (2), then we update the allocation via an augmentation that unallocates the minimum nonwhite tentatively allocated agent, call it u', and allocates u. The time complexity for performing this update is the same as in the case of the preceding paragraph. The difference is that here we cannot necessarily discard agent u'. In particular, if agent u' was black before the update, then it may still have one or more active bid components; if so, we add agent u' to the set of unallocated agents for which the associated bids have yet to be processed. While the size of the latter set does not decrease (because we removed u and added u'), we are able to prove that the number of black tentatively allocated agents has decreased by at least one. Consequently, in any execution of the main auction, the total number of SSSP computations performed is at most the total number of agents in the auction.

Recall that our main auction consists of two phases. The foregoing discussion has focused on the implementation of the first phase. In the second phase, any black tentatively allocated agents are given the opportunity to exchange items with one another. As discussed in Section 5.2, either the TTC algorithm or the $\mathrm{TC}^{\prec}$ algorithm is used to update the allocation, and the item prices are left unchanged. The second phase of the top-level auction can be implemented in linear time in the size of the active bid-graph using the TTC algorithm [13], and in polynomial time using the $\mathrm{TC}^{\prec}$ algorithm [9].

6 Concluding Remarks

For the classic sealed-bid unit-demand framework, the celebrated VCG mechanism yields a truthful, efficient, and envy-free outcome. In this paper, we introduce a generalization of the classic sealed-bid unit-demand setting that is motivated by practical applications. We show that our auction for this setting retains, to the extent possible, the strong properties of the VCG mechanism.

Acknowledgements. We are grateful to anonymous referees for their valuable comments and pointers to related work.

References

[1] G. Aggarwal, S. Muthukrishnan, D. Pál, and M. Pál. General auction mechanism for search advertising. *World Wide Web*, 18, 2009.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows.* Prentice Hall, New York, NY, 1993.

[3] L. M. Ausubel, P. Cramton, and P. Milgrom. *The Clock-Proxy Auction: A Practical Combinatorial Auction Design.* MIT Press, Cambridge, Massachusetts, 2005.

[4] H. Clarke, E. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.

[5] G. Demange, D. Gale, and M. Sotomayor. Multi-item auctions. *Journal of Political Economy*, 94(4):863–872, 1986.

[6] S. Fujishige and Akihisa. Tamura. A two-sided discrete-concave market with possibly bounded side payments: An approach by discrete concave analysis. *Mathematics of Operations Research*, 32(1):136–155, 2007.

[7] T. Groves. Incentives in teams. *Econometrica*, 41(4):617–631, 1973.

[8] F. Gul and E. Stacchetti. Walrasian equilibrium with gross substitutes. *Journal of Economic Theory*, 87(1):95–124, 1999.

[9] P. Jaramillo and V. Manjunath. The difference indifference makes in strategy-proof allocation of objects. *Under review*, 2009.

[10] T. Koopmans and M. Beckmann. Assignment problem and the location of economic activities. *Econometrica*, 25:53–76, 1957.

[11] C. Krishnappa and C. G. Plaxton. A sealed-bid unit-demand auction with put options. Technical Report TR–11–04, Department of Computer Science, University of Texas at Austin, February 2011.

[12] H. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 3:253–258, 1955.

[13] L. Shapley and H. Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1(1):23–37, 1974.

[14] K. Steiglitz. *Snipers, Shills & Sharks: eBay and Human Behavior.* Princeton University Press, Princeton, New Jersey, 2007.

[15] M. Thorup. Integer priority queues with decrease key in constant time and the single source shortest paths problem. *Journal of Computer and System Sciences*, 69(3):330–353, 2004.

[16] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, 1961.

[17] L. Walras. *Elements of Pure Economics.* Allen and Unwin, 1954.