Computing buyer-optimal Walrasian prices in multi-unit matching markets via a sequence of max flow computations

Katharina Eickhoff\textsuperscript{1}, S. Thomas McCormick\textsuperscript{2}, Britta Peis\textsuperscript{1}, Niklas Rieken\textsuperscript{1}, and Laura Vargas Koch\textsuperscript{3}

\textsuperscript{1} School of Business and Economics, RWTH Aachen University
\textsuperscript{2} Sauder School of Business, University of British Columbia
\textsuperscript{3} Department of Mathematics, ETH Zurich

November 12, 2021

Abstract. In a matching market a set of objects is sold to a set of buyers, each equipped with a valuation function for the objects. The goal of the auctioneer is to determine reasonable prices together with a stable allocation. One definition of “reasonable” and “stable” is a Walrasian equilibrium, which is a tuple consisting of a price vector together with an allocation satisfying the following desirable properties: (i) the allocation is market-clearing in the sense that as much as possible is sold, and (ii) the allocation is stable in the sense that every buyer ends up with an optimal set with respect to the given prices. Moreover, “buyer-optimal” means that the prices are smallest possible among all Walrasian prices. In this paper, we investigate the computability of finding a buyer-optimal Walrasian equilibrium.

We present a combinatorial network flow algorithm to compute buyer-optimal Walrasian prices in a multi-unit matching market with linear valuation functions and buyer demands. The algorithm can be seen as a generalization of the classical housing market auction and mimics the very natural procedure of an ascending auction. We use our insights to prove sensitivity results when the demand or supply in the matching market changes. Moreover, we summarize known results for the more general case with strong substitute valuation functions. We discuss connections between previous results and our results, especially our algorithm is an implementation of Ausubel’s auction for linear valuation functions.

1 Introduction

We consider a multi-unit auction market where discrete indivisible items of \( n \) different types, denoted by \( \Omega = \{i_1, \ldots, i_n\} \), are sold to a set of \( m \) buyers \( B = \{j_1, \ldots, j_m\} \). We assume that each product type (or object) \( i \in \Omega \) is available in a quantity \( b_i \in \mathbb{Z}_+ \). Each buyer \( j \in B \) is interested in buying at most \( d_j \in \mathbb{Z}_+ \) items in total. We call this general case a multi-unit matching market, and the special case where \( b_i = 1 \) for all \( i \in \Omega \), and \( d_j = 1 \) for all \( j \in B \) a single-unit matching market. The goal of the auctioneer is to find a per-unit price \( p(i) \) for each object \( i \), together with an allocation (or assignment) \( x \in \mathbb{Z}_+^{\Omega \times B} \) of items to buyers such that the prices \( p \) and the allocation \( x \) satisfy certain desirable properties. Certainly, the allocation \( x \) should be feasible in the sense that at most \( b_i \) units are sold of each object \( i \in \Omega \), and each buyer \( j \) buys at most \( d_j \) items. That is, \( x \in \mathbb{Z}_+^{\Omega \times B} \) needs to satisfy the feasibility constraints

\[ \sum_{j \in B} x_{ij} \leq b_i \quad \text{for all } i \in \Omega, \quad \text{and} \quad \sum_{i \in \Omega} x_{ij} \leq d_j \quad \text{for all } j \in B. \]

Most of this paper is concerned with the case where the buyers have linear valuations of the objects, i.e., where buyer \( j \in B \) has a value of \( v_{ij} \in \mathbb{Z}_+ \) for each copy of object \( i \in \Omega \). This value is reduced by the price \( p(i) \) that the auctioneer charges, so the net value (or payoff) of one unit of object \( i \) to buyer \( j \) is \( v_{ij} - p(i) \).

(We consider non-linear valuations later in the paper.) For an assignment \( x \in \mathbb{Z}_+^{\Omega \times B} \) we denote the items assigned to buyer \( j \) by \( x \cdot j \). Then, in addition to the feasibility constraints, each buyer \( j \in B \) should be
happy with her allocation \( x_{i*} \), i.e., she should be assigned one of her preferred bundles under the current prices. That is, \( x_{i*} \) should be an optimal integral solution of the linear program

\[
\max_{x_{i*}} \left\{ \sum_{i \in \Omega} (v_{ij} - p(i)) \cdot x_{ij} \mid \sum_{i \in \Omega} x_{ij} \leq d_j, 0 \leq x_{ij} \leq b_i \text{ for all } i \in \Omega \right\}.
\]

A feasible assignment where each buyer achieves a preferred bundle is called stable. Prices which admit a stable allocation are called competitive. Note that competitive prices can easily be achieved. For example, if each price \( p(i) \) exceeds the maximum valuation \( \max_{j \in B} v_{ij} \) for this item, the prices are competitive since each buyer achieves a preferred bundle under the assignment where nothing is sold. Thus, the goal of an auctioneer lies in finding competitive prices which are market-clearing in the sense that the maximum possible amount \( D := \min \{ \sum_{i \in \Omega} b_i, \sum_{j \in B} d_j \} \) of items is sold. Such market-clearing competitive prices are also known as Walrasian prices. A tuple \((p^*, x^*)\) consisting of Walrasian prices \( p^* \) and an associated stable allocation \( x^* \) is called Walrasian equilibrium [20].

Prices are buyer-optimal if they minimize \( \sum_{i \in \Omega} p(i) \) among all Walrasian prices. Walrasian prices are known to form a lattice with respect to the component-wise ordering (see [6] for the single-unit case and [1] in general). That is, if \( p \) and \( q \) are Walrasian prices, then the two vectors \( p \land q := (\min \{ p_i, q_i \})_{i \in \Omega} \) and \( p \lor q := (\max \{ p_i, q_i \})_{i \in \Omega} \) as well. Due to [1] this holds even in the more general setting with strong substitute valuation functions (defined below). Thus, prices are buyer-optimal if and only if they are the unique component-wise minimal Walrasian prices.

**Special case: single-unit matching markets.** The model we consider generalizes the classical matching market (a.k.a. housing market) model. For such single-unit matching markets, a seminal paper of Demange et al. [3] describes an ascending auction which starts at the minimal possible selling prices (e.g. \( p_i = 0 \) for all \( i \in \Omega \)) and iteratively raises the prices on some overdemanded set (“Hall set”) until the prices are market-clearing. By always raising the prices on an inclusion-wise minimal overdemanded set, Demange et al. guarantee that the prices are the (unique) component-wise minimal competitive prices, and that they are market-clearing, thus buyer-optimal Walrasian prices. Kern et al. [10] propose a primal-dual algorithm to compute these prices in single-unit matching markets in polynomial time. They furthermore point out that those prices coincide with VCG prices (see [17] for an introduction), and so are incentive compatible; see [4] for a good overview on these topics.

**Disadvantages of the copy method.** A naïve approach to reduce our more general multi-unit auction to a single-unit auction is via the following copy method: we replace the \( b_i \) items of object \( i \) by \( b_i \) copies of a unit object, and replace the \( d_j \) items demanded by buyer \( j \) by \( d_j \) unit-item buyers, with the same valuations. Certainly, an ascending auction of the single-unit instance will return market-clearing prices, but these prices are in general not buyer-optimal:

**Example 1.** Consider one buyer with a demand of two and two different items with a supply of one which are valued differently by her, say \( v = (5, 1) \). If we copy the buyer, both copies will prefer object 1 until the price increased to 4. Now both copies of the sole buyer are indifferent between the objects and thus \( p = (4, 0) \) and \( x = (1, 1) \) is a Walrasian equilibrium. However, considering the original situation, since the buyer is alone \( p = (0, 0) \) and \( x = (1, 1) \) is the buyer-optimal Walrasian equilibrium. Thus, the prices computed by the copy method are not buyer-optimal.

**Remark 1.** Example [1] shows that the copy method is not incentive compatible and thus does not lead to VCG prices. For the buyer it would be strictly better to report valuation \( v = (1, 1) \).

### 1.1 Multi-unit auctions with strong substitute valuations

In this section we review some of what is known about non-linear valuation functions. The basic setup remains the same, where the auctioneer is looking for prices \( p \in \mathbb{Z}_+^{\Omega} \), and an allocation \( x \in \mathbb{Z}_+^{\Omega \times B} \) that
satisfies $\sum_{j \in B} x_{ij} \leq b_i$ for all $i \in \Omega$. Each buyer $j \in B$ has a valuation function $v_j$ that maps $j$’s allocation $x_{ij}^*$ to a value in $\mathbb{Z}_+$. Then the net value to buyer $j$ under prices $p$ of allocation $x_{ij}^*$ is $v_j(x_{ij}^*) - p^T x_{ij}^*$, and the set of preferred bundles of buyer $j \in B$ w.r.t. prices $p$ is
\[ D_j(p) := \arg \max \{v_j(x_{ij}^*) - p^T x_{ij}^* \mid 0 \leq x_{ij}^* \leq b_i, x_{ij}^* \in \mathbb{Z}^\Omega \}. \]
The maximum payoff buyer $j$ can obtain under prices $p$ when all other buyers are absent is
\[ V_j(p) := \max \{v_j(x_{ij}^*) - p^T x_{ij}^* \mid 0 \leq x_{ij}^* \leq b_i, x_{ij}^* \in \mathbb{Z}^\Omega \}. \]
Our main model with linear valuations corresponds to the special case of this where the valuation functions $v_j$ of all buyers $j \in B$ are of the form
\[ v_j(x) = \max \{ \sum_{i \in \Omega} v_{ij} \bar{x}_{ij} \mid \bar{x} \leq x, \sum_{i \in \Omega} \bar{x}_{ij} \leq d_j, \bar{x}_{ij} \leq b_i \} \tag{2} \]
for given values $v_{ij} \in \mathbb{Z}_+$.

A valuation function $v_j : \mathbb{Z}_+^\Omega \rightarrow \mathbb{Z}$ satisfies the \textit{gross substitute condition} if for all price vectors $p, q \in \mathbb{R}^\Omega$ with $p \leq q$ it holds that for all $x_{ij}^* \in D_j(p)$ there exists $y_{ij} \in D_j(q)$ such that $x_{ij} \leq y_{ij}$ for all $i \in \Omega$ with $p(i) = q(i)$. If, additionally, $\sum_{i \in \Omega} x_{ij} \geq \sum_{i \in \Omega} y_{ij}$ holds for these allocations $x, y$, then we say that $v_j$ satisfies the \textit{strong gross substitute condition}. That is, the gross substitute property states that the demand on object $i$ (with $p(i) = q(i)$) does not decrease if other objects become more expensive ($p(k) < q(k)$ for some $k \neq i$), while the strong gross substitute property additionally requires that with higher prices the total demand cannot increase. A function $v_j : \mathbb{Z}_+^\Omega \rightarrow \mathbb{Z}$ is called \textit{concave-extendible} if there exists a concave function $\tilde{v}_j : \mathbb{R}_+^\Omega \rightarrow \mathbb{R}$ with $\tilde{v}_j(x) = v_j(x)$ for all $x \in \mathbb{Z}_+^\Omega$. A valuation function $v_j : \mathbb{Z}_+^\Omega \rightarrow \mathbb{Z}$ which is both strong gross substitute and concave-extendible is called \textit{strong substitute}. For example, the linear valuation functions $v_j$ defined in (2) are strong substitute. Note that this definition of strong substitute differs from the one used by Ausubel [1]. In [1], a valuation function $v_j$ is said to be strong substitute if, after copying each item $b_i$ times, the resulting valuation function is gross substitute. These two definitions are, however, equivalent (see, e.g., Theorem 13 in [12]).

For the special case where the supply of each object is one, Gul and Stacchetti [6] and Ben-Zwi [2] provide further equivalent conditions for valuation functions to be gross substitute. Fujishige and Yang [3] pointed out that the gross substitute condition is equivalent to $M^2$-concavity (see [3] for the definitions of $M^2$-concavity and $L^2$-convexity). Murota, Shioura and Yang [15] generalize their result to multi-unit auctions and prove that valuation function $v_j$ is strong substitute if and only if it is $M^2$-concave (Proposition A.2 and A.3 in [15]).

It turns out that the $M^2$-concavity of the valuation functions $v_j$ is crucial for the computability of Walrasian equilibria. For this, consider the following potential (or Lyapunov) function
\[ L(p) = \sum_{j \in B} V_j(p) + \sum_{i \in \Omega} b_i p(i). \]
In case the valuation functions $v_j$ are strong substitute, it is known (see e.g. [19]) that the minimizers of $L$ are precisely the Walrasian prices which are guaranteed to exist due to [19]. Moreover, the subgradients of $L$ at price $p$ correspond exactly to the overdemanded sets (a.k.a. excess demand sets [11]). This motivates the common theme in ascending auctions of increasing the prices of objects in overdemanded sets by one unit in each iteration. Such price-raising steps can be implemented to run in strongly polynomial time at each iteration thanks to the fact that, when each valuation function $v_j$ is strong substitute, the Lyapunov function is an $L^2$-convex function ([15] Theorem 1.6). Thus, a Walrasian price vector can be computed in weakly polynomial time via, e.g., the steepest descent scaling algorithm (see [14] Section 4.2) for computing the minimizer of an $L^2$-convex function.

In this paper we are interested in the computability of \textit{buyer-optimal} Walrasian price vectors. As usual, let $\chi_X$ denote the incidence vector of set $X$. Ausubel [1] shows that an ascending auction which raises
prices on an inclusion-wise minimal minimizer of the function $X \mapsto L(p + \chi_X)$ in each iteration results in buyer-optimal Walrasian prices.

Murota et al. [15] investigate the computability of the ascending auction which raises in each iteration an arbitrary, not necessarily minimal, minimizer of $L(p + \chi_X)$. They show that the price-raising step in each iteration can be implemented in strongly-polynomial time via Submodular Function Minimization (SFM). Furthermore, they show that the ascending auction needs $\|p - p_0\|_\infty$ iterations to terminate, where $p_0$ is the initial price vector (note that $p_0$ has to be component-wise smaller than a Walrasian price vector). However, this leaves open the possibility that changing prices on overdemanded sets by more than one unit at each iteration might lead to a better iteration bound.

While in [15] the Walrasian prices computed by the auction are not necessarily buyer-optimal, since the authors did not choose the minimal minimizer, they adapt their procedure in [16]. Since $L$ is $L^s$-convex, it is possible to find the minimal minimizer of $L(p + \chi_X)$ in polynomial time, for example with Iwata and Orlin’s Algorithm for submodular function minimization ([8], [11] Section 3.4.2).

Moreover, [16] implies a weakly polynomial time algorithm to compute the buyer-optimal Walrasian prices directly by slightly perturbing the Lyapunov function $L$. With the right perturbation the minimal minimizer becomes the unique minimizer and can thus be computed by e.g. the steepest descent scaling algorithm (see [14] Section 4.2.; see Appendix A.7 for details).

### 1.2 Our contributions

In this paper, we focus on multi-unit auctions with linear valuation functions and demands. For this special case we provide a flow based ascending auction. We prove our results independent from the literature on auctions with strong substitute valuations and discrete convex analysis by using network flow properties. This enables us to show sensitivity regarding changes in supply and demand.

More concretely, in Section 2 we present an ascending auction which iteratively raises the prices on the objects in the left-most min cut in an associated auxiliary flow network, and prove that the algorithm terminates with component-wise minimal Walrasian prices. We show how to construct the corresponding stable allocation where as much as possible is sold, and where every object with positive price is completely sold (Theorem 1). Section 3 shows that structural insights obtained from our flow-based approach lead to several insights regarding the sensitivity analysis of our ascending auction. Here a basic ingredient is that the minimal competitive prices exist and coincide with the buyer-optimal Walrasian prices.

Finally, Section 4 compares our work to previous work. It shows how to compose known results to increase the step size of the algorithm such that fewer update steps are needed. For the single-unit case this results in a polynomial bound on the number of iterations.

### 2 A flow-based ascending auction for linear valuations

We first sketch our flow-based ascending auction, called the Price-Raising Algorithm. The auction starts with the all-zero vector $p_0(i) = 0$ for all $i \in \Omega$ (or with any initial price vector $p_0$ known to be a lower bound on the minimal Walrasian price vector $p^*$). In each iteration, given the current price vector $p \in \mathbb{R}^\Omega_+$, the algorithm computes an integral $s$-$t$-flow $f$ of maximum value in an auxiliary flow network $G(p)$ (described below). If the value $\text{val}(f)$ of flow $f$ equals the sum of capacities on the $s$-leaving arcs in $G(p)$, call it $D_p$, the algorithm stops. Otherwise (if $\text{val}(f) < D_p$), the prices on all objects in the left-most (minimal) min cut are raised by one unit, and the algorithm iterates with the updated price vector. Then Theorem 2 shows that the final price vector $p^*$ returned by the algorithm is the minimal (and thus buyer-optimal) competitive price vector.

For computing a corresponding stable and market-clearing allocation $x^*$ such that $(p^*, x^*)$ is a buyer-optimal Walrasian equilibrium, we modify $G(p^*)$ slightly to network $H(p^*)$, which allows us to find a stable

---

4 By searching for a maximal minimizer $Y$ of the submodular function $Y \mapsto L(p + \chi_{\Omega \setminus Y})$. Note that $X = \Omega \setminus Y$ is a minimal minimizer of $L(p + \chi_X)$ by the choice of $Y$. 

4
Lemma 1. The prices $p$ are competitive if and only if there is a flow $f$ in $G(p)$ satisfying all $s$-leaving arcs, i.e., of value $\text{val}(f) = \sum_{j \in B} (h_{j'} + h_{j''})$. Moreover, given a competitive price vector $p$, an associated stable allocation $x$ can be computed via a single max flow computation in $G(p)$.

Proof. There is a clear one-to-one correspondence between an integral $s$-$t$-flow and a feasible assignment of objects. Since all capacities in $G(p)$ are integral, there exists a max flow with integral values. Consider a path decomposition of an integral max flow $f$ into $s$-$t$-paths. Every path $P$ connects a buyer $j$ in form of a representative $j'$ or $j''$ and an object $i$ and carries flow of value $|f|_P$. By assigning $|f|_P$ items of object $i$ to buyer $j$ for every such path we obtain a feasible assignment. This assignment is competitive if and only if the demand of every buyer at prices $p$ is satisfied. This is equivalent to the requirement that the flow satisfies all $s$-leaving arcs, i.e., $\text{val}(f) = \sum_{j \in B} (h_{j'} + h_{j''})$. \qed

2.1 Structure of preferred bundles

A preferred bundle for buyer $j$ with respect to prices $p$ can be computed by the greedy procedure which starts with the empty set and, iteratively (until $d_j$ units are selected) adds an item $i$ of maximum payoff $v_{ij} - p(i)$ among those objects of positive residual supply. The preferred bundles of buyer $j$ are often not unique. To understand their structure better, we partition the objects which may occur in a preferred bundle of buyer $j$ with respect to prices $p$ into the sets $\Omega_j'(p)$, $\Omega_j''(p)$, and $\Omega_j'''(p)$ as detailed in Appendix A.1.

The procedure enforces that $\Omega_j'(p)$ is the set of objects that buyer $j$ values most and so which will appear in all her preferred bundles under the current prices $p$, whereas $\Omega_j''(p)$ is the set of objects among which she is indifferent for “filling up” her bundle to meet her demand. The objects in $\Omega_j'''(p)$ have payoff zero, so she is indifferent whether to buy any of them. To shorten notation, we write $\Omega_j'$ instead of $\Omega_j'(p)$ when $p$ is clear from the context, and similarly for $\Omega_j''$ and $\Omega_j'''$. Thus, in every preferred bundle, buyer $j$ buys $h_{j'} := \sum_{i \in \Omega_j'} b_i$ items from objects in $\Omega_j'$ and $h_{j''} := \min\{\sum_{i \in \Omega_j''} b_i, d_j - h_{j'}\}$ items from objects in $\Omega_j''$.

Note that the minimum is attained in the former term only if $\Omega_j'' = \emptyset$. In addition, there might be up to $h_{j'''} := \min\{\sum_{i \in \Omega_j'''} b_i, d_j - h_{j'}\}$ items of objects with zero payoff in a preferred bundle.

2.2 Construction of auxiliary flow networks

Each iteration uses the $\Omega_j'(p)$, $\Omega_j''(p)$, and $\Omega_j'''(p)$ to construct the auxiliary flow network $G(p)$. Our algorithm proceeds in two phases: in the first phase, it computes the minimal Walrasian price vector $p^*$, and in the second phase it computes the associated market-clearing competitive allocation $x^*$. Thus the first phase we don’t need to worry about allocating the objects in $\Omega_j'''(p)$ in $G(p)$, since buyers are indifferent between buying and not buying items from these sets. After the first phase has computed $p^*$, the second phase inserts vertices and arcs into $G(p^*)$ to get $H(p^*)$, and computes a max (maximum) flow in $H(p^*)$ to determine a market-clearing competitive allocation $x^*$.

Given the sets $\Omega_j'(p)$ and $\Omega_j''(p)$ with associated demands $h_j'$ and $h_j''$ for every buyer $j \in B$, we construct $G(p)$ as follows (see Figure 1 for an example of $G(p)$ with two buyers and three objects). The vertex set of $G(p)$ consists of source $s$, sink $t$, one vertex for each object $i \in \Omega$, and two vertices $j'$ and $j''$ for each buyer $j \in B$. Vertices $j'$ and $j''$ correspond to the sets $\Omega_j'(p)$ and $\Omega_j''(p)$, respectively. The arcs are defined as follows:

- $(s, j')$ with capacity $h_{j'}$ for all $j \in B$,
- $(s, j'')$ with capacity $h_{j''}$ for each $j \in B$,
- $(j', i)$ with capacity $g_{ij'} := b_i$ for all $j \in B$, $i \in \Omega_j'$,
- $(j'', i)$ with capacity $g_{ij''} := \min\{b_i, h_{j''}\}$ for all $j \in B$, $i \in \Omega_j''$, and
- $(i, t)$ with capacity $b_i$ for all $i \in \Omega$.

Lemma 1. The prices $p$ are competitive if and only if there is a flow $f$ in $G(p)$ satisfying all $s$-leaving arcs, i.e., of value $\text{val}(f) = \sum_{j \in B} (h_{j'} + h_{j''})$. Moreover, given a competitive price vector $p$, an associated stable allocation $x$ can be computed via a single max flow computation in $G(p)$. \qed
However, in this allocation, no item with payoff zero is included. Since we aim for an allocation where as much as possible is sold, we extend $G(p^*)$ to flow network $H(p^*)$ such that the assignments of buyers to objects of payoff zero are possible. To do so, we first balance the supply and the demand. If $\sum_{i \in \Omega} b_i < \sum_{j \in B} d_j$, we add a dummy object $i_0$ with supply $b_{i_0} = \sum_{j \in B} d_j - \sum_{i \in \Omega} b_i$ and valuations $v_{i_0j} = 0$ for all $j \in B$. If $\sum_{i \in \Omega} b_i > \sum_{j \in B} d_j$, we add a dummy buyer $j_0$ with demand $d_{j_0} = \sum_{i \in \Omega} b_i - \sum_{j \in B} d_j$ and valuations $v_{ij_0} = 0$ for all $i \in \Omega$. Now we can assume that $\sum_{i \in \Omega} b_i = \sum_{j \in B} d_j$. Note that the Price-Raising Algorithm computes the same prices with the dummy object or buyer as without. To construct $H(p^*)$ from $G(p^*)$, we add a vertex $j''$ for each buyer $j \in B$ and an arc $(s, j'')$ with capacity $h_{j''}$. Furthermore, we add for each $j \in B$ the arc $(j'', i)$ with capacity $\min\{b_i, h_{j''}\}$ for all $i \in \Omega''$.

**Proposition 1.** A max flow in network $H(p^*)$ and its corresponding allocation satisfy:
1. Buyers are assigned to subsets of their preferred bundles at prices $p^*$.
2. If for buyer $j \in B$ the flow on the arcs $(s, j')$ and $(s, j'')$ is saturated, $j$ is assigned to one of her preferred bundles at prices $p^*$.
3. It holds that $h_{j'} + h_{j''} \leq d_j$ for each $j \in B$.

### 2.3 Computation of a buyer-optimal Walrasian equilibrium

Here we formally describe the Price-Raising Algorithm. Each of its iterations can be done in polynomial time since the network can be constructed in polynomial time and only one max flow computation is needed. The number of iterations is $\|p^* - p_0\|_\infty$, where $p^*$ is the minimal Walrasian price vector (see [15]), which is pseudo-polynomial.

Given buyer-optimal prices $p^*$, the Allocation Algorithm constructs the auxiliary network $H(p^*)$ and its max flow, leading to allocation $x^*$. Hence $x^*$ can be found in polynomial time. Section 2.5 shows that the value of the maximum flow is $\max\{\sum_{i \in \Omega} b_i, \sum_{j \in B} d_j\}$, since we include a dummy buyer resp. a dummy object. With Proposition 1 this implies that each buyer is assigned to one of her preferred bundles.

If the supply does not exceed the demand, everything is sold. Otherwise, only the items which are allocated to the dummy buyer $j_0$ are not sold. Since $v_{ij_0} = 0$ for all $i \in \Omega$, the price of an object allocated to $j_0$ has to be zero. Thus all objects with positive price are completely sold.

**Theorem 1.** The prices $p^*$ computed by the Price-Raising Algorithm are the buyer-optimal Walrasian prices. Moreover, under the stable allocation $x^*$ returned by the Allocation Algorithm as much as possible is sold, and every item with positive price is sold.

---

**Fig. 1.** The auxiliary network with two buyers $j_1, j_2$ and three objects $\alpha, \beta, \gamma$. We have valuations $v_{j_1} = (3, 2, 1)$ and $v_{j_2} = (0, 2, 0)$, demands $d_{j_1} = 4, d_{j_2} = 2$, supply $b_{\alpha} = b_{\beta} = 1, b_\gamma = 4$, and current prices $p(\alpha) = p(\beta) = p(\gamma) = 0$. Appendix A.2 contains a more complicated example network.
We now analyze the Price-Raising Algorithm by considering the behavior of the buyers at given prices.

### 2.4 Properties of the prices

Section 3. Second, our proof is independent from the literature on strong substitute and $L^2$-convex functions such as [1]. However, our proof has two advantages. First, we show slightly more by proving that minimal competitive prices are buyer-optimal Walrasian prices, which is useful in Section 4. Second, our proof is independent from the literature on strong substitute valuation functions and uses only network flow arguments.

#### 2.4.1 We sketch the proof of Theorem 1 in subsequent sections, with the proofs of the lemmas in Appendices A.3 and A.4. Note that the theorem follows directly from our Section 4.1 and results in literature on strong substitute valuation functions such as [1]. However, our proof has two advantages. First, we show slightly more by proving that minimal competitive prices are buyer-optimal Walrasian prices, which is useful in Section 4. Second, our proof is independent from the literature on strong substitute valuation functions and uses only network flow arguments.

#### 2.4 Properties of the prices

We now analyze the Price-Raising Algorithm by considering the behavior of the buyers at given prices $p$ using the structure of $G(p)$. As usual, for a given digraph with arc set $A$, we use $\Gamma^-(v)$ to refer to all nodes that are the starting node of an incoming arc into $v$, i.e., $\Gamma^-(v) := \{ u \mid (u,v) \in A \}$, analogously for $\Gamma^+(v)$. We extend this definition also to sets, so that $\Gamma^-(I) := \{ u \mid \exists v \in I \text{ with } (u,v) \in A \}$.

**Definition 1.** Given a price vector $p$ and the associated network $G(p)$, a set $I \subseteq \Omega$ is overdemanded if

$$\sum_{i \in I} b_i < \sum_{j \in \Gamma^-(I)} \max \{ 0, h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \}. $$

The left hand side of this inequality denotes the total number of items in $I$. The right hand side is the sum of each buyer’s demand which cannot be fulfilled by items which are not in $I$. If the inequality holds there are not enough items of objects in $I$ to sell in order to meet all demands for that set. We use Lemma 1 and the following three lemmas to prove that such an overdemanded set exists if and only if the prices are not competitive. Let $B' := \bigcup_{j \in B} j'$ and $B'' := \bigcup_{j \in B} j''$.

**Lemma 2.** There is no overdemanded set for prices $p$ if there exists a flow $f$ in the network $G(p)$ of value $\text{val}(f) = \sum_{j \in B' \cup B''} h_j$.

**Lemma 3.** Let $C$ be a left-most min cut in $G(p)$ and $I := C \cap \Omega$. It holds that

$$C = \{ s \} \cup \left\{ j \in B' \cup B'' \mid h_j > \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\} \cup I.$$
Lemma 4. Given prices p, if there is any overdemanded set, then $I := C \cap \Omega$ is an overdemanded set, where $C$ is the left-most min cut of the network $G(p)$.

Now we can prove the first part of our main theorem. The following proof is based on Demange et al. but we use the left-most min cut and its structure instead of an inclusion-wise minimal overdemanded set.

Theorem 2. The prices $p^*$ returned by our Price-Raising Algorithm are the (unique) component-wise minimal competitive prices, i.e., if $q$ is a competitive price vector then $p^*(i) \leq q(i)$ for all $i \in \Omega$.

Proof. Assume for contradiction that for a competitive price vector $q$, there is an object $i \in \Omega$ such that $p^*(i) > q(i)$. Let $p_t$ denote the price vector in iteration $t$. Then, for the start vector $p_0$ we have $p_0 \leq q$. Let $t$ be the last iteration where $p_t \leq q$, i.e., there is an object $i \in \Omega$ such that $p_{t+1}(i) > q(i)$. Let $I$ be the overdemanded set chosen by the algorithm in iteration $t$. Further, let $I_1$ be the subset of $I$ where $p_t(i) = q(i)$ holds and $I_2$ be the subset of $I$ where $p_t(i) < q(i)$ holds. Thus, $I = I_1 \cup I_2$, and

$$I = \{ i \in \Omega \mid p_t(i) < p_{t+1}(i) \}, \quad I_1 = \{ i \in I \mid p_t(i) = q(i) \}, \quad I_2 = \{ i \in I \mid p_t(i) < q(i) \}.$$

We will derive a contradiction by showing that $I_2$ induces a min cut which is a strict subset of $C$, since $I_1$ is non-empty by choice of $t$. To do so, we start with analyzing the behavior of buyer $j \in B$ at prices $q$ by comparing it with the behavior at prices $p_t$. For this purpose, we fix the network properties such as $g_{ij}$, $h_j$ and also $j'$ and $j''$ of the prices $p_t$. With this in hand, we can show the following lemma by analyzing the difference of the cut values. Its proof is in Appendix A.3.

Lemma 5. The cut $\bar{C} = \{ s \} \cup \{ j \in B' \cap B'' \mid h_j > \sum_{i \in I^{+}(j) \cap I_2} g_{ij} \} \cup I_2$ is a min cut and $\bar{C} \subseteq C$.

This contradicts $C$ being a left-most min cut and thus the assumption that there is an object $i$ with price $p(i) > q(i)$. Therefore, the Price-Raising Algorithm finds the component-wise minimal competitive price vector $p^*$ if it starts at prices $p_0 \leq p^*$.

2.5 Properties of the allocation

It remains to show that the prices computed by the Price-Raising Algorithm are market-clearing, i.e., that an allocation exists where $D = \min \{ \sum_{i \in I} b_i, \sum_{j \in B} d_j \}$ is sold.

Lemma 6. Given a price vector $p$ computed in some iteration of the Price-Raising Algorithm starting with prices $p_0 = 0$, then there is a max flow $f$ in $G(p)$ with corresponding allocation $x \in \mathbb{Z}_{+}^{I \times B}$ such that

1. $\sum_{j \in B} x_{ij} = b_i$ if $p(i) > 0$,
2. $x_{ij}$ is a subset of a preferred bundle of buyer $j$.

We show this by iteratively constructing such an allocation using the steps of the Price-Raising Algorithm starting with $p_0 = 0$, see Appendix A.4.

Theorem 3. Given prices $p^*$ computed by the Price-Raising Algorithm, there exists an allocation $x^* \in \mathbb{Z}_{+}^{I \times B}$ such that:

1. Any buyer $j \in B$ gets a preferred bundle, i.e., $x^*_{ij} \in \mathbb{Z}_{+}^{D_j(p^*)}$.
2. As much as possible is sold, i.e., $\sum_{i \in \Omega} \sum_{j \in B} x^*_{ij} = D$.
3. If there is an item which is not sold, it has price zero, i.e., $\sum_{j \in B} x^*_{ij} = b_i$ for all $i \in \Omega$ with $p^*(i) > 0$.

The definition of an overdemanded set by Demange et al. differs slightly from ours because they consider the unit supply/demand case.
Proof. Lemma\textsuperscript{[6]} shows that there is an allocation $x^*$ such that every item with a positive price is completely sold and $x^*_j$ is a subset of a preferred bundle of buyer $j$. Moreover, the corresponding flow $f^*$ is a max flow in $G(p^*)$. Thus $x^*$ is a stable allocation since $p^*$ is a competitive price vector.

It remains to show that as much as possible is sold by the algorithm. But this follows directly since all unsold items have price 0 and all players with left over demand have payoff 0 on all items that they do not buy. This means we get a complete bipartite graph in which every maximal flow assigns either all remaining objects or satisfies all demands. Since all prices and payoffs are 0, any such flow can just be added to the computed assignment.

Note that this construction is only needed to prove the existence of such a flow, but it will actually be computed by the Allocation Algorithm. Now we have all properties needed to prove the main theorem.

Proof of Theorem\textsuperscript{[7]} The prices are the minimal competitive prices by Theorem\textsuperscript{[2]}. By Theorem\textsuperscript{[3]} there is an associated stable allocation where $D$ is sold and where each item with positive price is sold. This existence together with the construction of the network $H(p^*)$ gives us that the Allocation Algorithm computes such an allocation.

3 Sensitivity analysis, social optimality, and VCG

Sensitivity analysis. The Price-Raising Algorithm determines for each instance the unique component-wise minimal Walrasian price vector $p^*$. In this section, we analyze the sensitivity of the auction with respect to changes of the demand and supply. In particular, we show that, as intuitively expected, the auction is monotone in the sense that the returned prices can only increase if the demand increases or the supply decreases.

Theorem 4. Given an instance with valuations $v$, demands $d$, supplies $b$ and the corresponding buyer-optimal Walrasian prices $p$, consider a second instance with the same valuation functions but increased demands and decreased supplies, i.e., demands $d'$ with $0 \leq d_j \leq d'_j$ for all $j \in B$ and supplies $b'$ with $0 \leq b'_i \leq b_i$ for all $i \in \Omega$ with buyer-optimal Walrasian prices $p'$. Then we have $p(i) \leq p'(i)$ for all $i \in \Omega$ with $b'_i > 0$.

The proof idea is that prices remain competitive when the supply increases or the demand decreases. Since for linear valuation functions minimal competitive prices are unique and market-clearing, the buyer-optimal Walrasian prices are smaller or equal to any competitive prices. Achieving a similar result for strong substitute valuation functions turns out to be more difficult. The reason is that it is unclear whether the minimal competitive prices are unique and whether they are market-clearing.

To prove the theorem we show two lemmas, analyzing the change of the demand and the supply separately.

Lemma 7. Let $d$ and $d'$ be two demand vectors with $d_j \leq d'_j$ for all $j \in B$ (here it is possible that $d_j = 0$ for some $j$). Then the buyer-optimal Walrasian prices $p$ at demand $d$ are not greater than the buyer-optimal Walrasian prices $p'$ at demand $d'$, i.e., $p(i) \leq p'(i)$ for all $i \in \Omega$.

Proof. Consider an integral max flow $f'$ at prices $p'$ in the auxiliary flow network $G(p')$. Recall that flow $f'$ corresponds to a feasible allocation from buyers to their preferred bundles. Now adapt this flow as follows. For each buyer $j$ with $d_j < d'_j$, among the paths going through $j'$ or $j''$, select one with a currently lowest payoff and reduce flow on that path, until flow through $j'$ and $j''$ gets reduced to $d_j$. This procedure terminates with a flow meeting demands $d_j$. Furthermore, since we still allocate the items with the highest payoff to a buyer, each buyer is allocated to a preferred bundle at prices $p'$ at demand $d$. Thus $p'$ is a competitive price vector for demand $d$. By Theorem\textsuperscript{[2]} it follows that $p$ is not only the minimal Walrasian price vector but also the component-wise minimal competitive one, we get $p(i) \leq p'(i)$ for all $i \in \Omega$.

Next we show that the minimal competitive prices are bigger if the supply is smaller.
Lemma 8. Let \( b \) and \( b' \) be two supply vectors with \( b'_i \leq b_i \) for all \( i \in \Omega \) (here it is possible that \( b'_i = 0 \) for some \( i \)). Then for the corresponding buyer-optimal Walrasian prices \( p \) and \( p' \), it holds that \( p(i) \leq p'(i) \) for all \( i \) with \( b'_i > 0 \).

Proof. Assume without loss of generality that \( b \) and \( b' \) only differ in the supply of object \( \ell \) by one item. We fix the allocation computed by the minimal competitive prices at supply \( b \) and consider two cases.

First, we consider the case \( b'_\ell > 0 \). Given the assigned bundles, we analyze the behavior of the buyers when the additional item of object \( \ell \) arrives. If there is a buyer \( j \) who is not assigned to one of her preferred bundles at supply \( b \), we knew that she is the only one who is assigned to items of object \( \ell \) since otherwise the prices \( p' \) are not competitive. Thus all other buyers are assigned to one of their preferred bundles at supply \( b \). Hence we can change the preferred bundle of buyer \( j \) by assigning one more item of \( \ell \) to her, if necessary by omitting the least profitable item. This change does not harm the other buyers, thus in the new allocation everyone is assigned to a preferred bundle at prices \( p' \). Thus prices \( p' \) are competitive for the instance with supply \( b \). The prices \( p \) are the minimal competitive prices by Theorem 2 which yields \( p(i) \leq p'(i) \).

Next, we consider the case \( b'_\ell = 0 \). We adapt the prices \( p' \) to prices \( \bar{p} \) by setting \( \bar{p}(i) = p'(i) \) for \( i \in \Omega \setminus \{\ell\} \) and \( \bar{p}(\ell) = \max_{j \in B} v_{ij} + 1 \). Thus no buyer wants to buy an item of object \( \ell \). Therefore, the given allocation is an assignment of buyers to one of their preferred bundles at prices \( p \) for supply \( b \). Using again that \( p \) is the minimal competitive price vector at supply \( b \), we get \( p(i) \leq \bar{p}(i) = p'(i) \) for all \( i \in \Omega \setminus \{\ell\} \).

Proof of Theorem 4. For a given modified instance we can construct an intermediate instance where only the demand is changed and the supply remains as in the original instance. With Lemma 7 this implies that prices only increase compared to the original instance. Now applying Lemma 8 to the intermediate instance gives the statement of the theorem.

The monotonicity allows for faster reoptimization by starting with the old Walrasian price vector.

**Corollary 1.** Given prices \( p \) we can compute \( p' \) by applying at most \( \|p - p'\|_\infty \) iterations of the Price-Raising Algorithm with start prices \( p(i) \) for all \( i \in \Omega \) with \( b'_i > 0 \).

Theorem 1 allows starting with any initial price vector which is in every component at most as large as the minimal-competitive price vector. Thus, Theorem 4 allows us to start the Price-Raising Algorithm at the price \( p(i) \) for all \( i \in \Omega \) with \( b'_i > 0 \). Murota et al. show that the number of iterations is then bounded by \( \max\{p(i) - p'(i) \mid i \in \Omega, b'_i > 0\} \). However, the following example shows that we cannot bound \( \|p - p'\|_\infty \) even if the demand or supply is only slightly changed:

**Example 2.** Consider an instance with two buyers and two objects. The valuation of both buyers are \((M, M)\), the demand for both is two and the supply of both objects is two. In this instance \( p = (0, 0) \) are buyer-optimal competitive prices. Now assume the demand of one buyer is increased by one. In this case, the unique buyer-optimal prices are \( p' = (M, M) \) and thus \( \|p - p'\|_\infty = M \). The same happens if the supply of one item is decreased by one.

**Social optimality and VCG.** We show that the buyer-optimal Walrasian equilibrium \((p^*, x^*)\) is socially optimal. Appendix A.5 proves the following Theorem:

**Theorem 5.** The computed buyer-optimal Walrasian equilibrium \((p^*, x^*)\) is socially optimal in the sense that the total payoff of buyers and sellers together is maximized.

Social optimality is a necessary condition if we aim to show that the allocation and prices computed by a procedure correspond to the VCG mechanism. But while in the single-unit case the determined prices are VCG prices (see [10]), we cannot expect that for the prices in our model. Intuitively, VCG prices can only be achieved by a price-per-bundle mechanism, not with a price-per-good mechanism. The following example illustrates this.
Example 3. Let $B = \{1, 2, 3\}$ and $\Omega = \{1, 2\}$ with demands $d_1 = d_2 = 2, d_3 = 1$, supplies $b_1 = 3, b_2 = 2$, and valuations $v_1 = (3, 1), v_2 = (2, 0), v_3 = (0, 1)$. It is easy to see that both feasible allocations (buyer 1 or 2 gets good 1 twice, the other one gets one of each good, buyer 3 gets one item of good 2 in both allocations) are socially optimal. Say buyer 1 gets good 1 twice, buyer 2 gets good 1 only once. Then the net effect of buyer 1 on buyer 2 (i.e., the cost that buyer 1 imposes on buyer 2 by her presence) is 2, so the only possible price for good 1 is 1. Since buyer 2 also has to purchase good 1 once, good 2 needs to be subsidized (i.e., assigned a price of $-1$) to give her a total cost of 0. However, this would give buyer 3 total cost of $-1$, which is not VCG.

4 Comparison and extension of the algorithm

The goal of this section is to show the connection between our algorithm and the existing work. It turns out that computing a left-most min cut in our network directly corresponds to finding the minimal minimizer of the Lyapunov function in case of linear valuation functions.

4.1 Comparison

Observation 2. In our model for linear valuations, the Lyapunov function can be rewritten to

$$L(p) = \max \sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i))x_{ij} + \sum_{i \in \Omega} b_ip(i)$$

subject to $\sum_{i \in \Omega} x_{ij} \leq d_j$ and $x_{ij} \in [0, b_i]$ for all $j \in B$ and all $i \in \Omega$.

Next we compute the amount by which the Lyapunov function can decrease in an augmentation step.

Proposition 2. In our model with linear valuation functions the difference of the Lyapunov function in an augmentation step equals the difference between the capacity of the $s$-leaving arcs and the min cut value. More formally

$$L(p) - L(p + \chi_X) = \sum_{j \in F^-(X)} \max \{0, h_j - \sum_{i \in [i^+(j) \setminus X]} g_{ij}\} - \sum_{i \in X} b_i.$$

The first sum is the difference $V_j(p) - V_j(p + \chi_X)$ which corresponds to the total amount of items that $j$ wants to buy from $X$ under prices $p$ without any alternative outside of $X$. This yields the equation by definition of the Lyapunov function.

Lemma 9. Given prices $p$, the overdemanded set $I$ determined by the left-most min cut in $G(p)$ minimizes $L(p + \chi_X)$ among all $X \subseteq \Omega$.

For the proof we carefully compare the two values and use Proposition 2, see Appendix A.6.

4.2 Adapted Step-Length

One natural approach to speed up the computation of buyer-optimal prices is to increase the length of the augmentation steps. In the Price-Raising Algorithm, this means given a left-most min cut, increase the prices of the corresponding objects until the min cut changes. In the more general case of strong substitute valuation functions, this means a steepest descent direction of the Lyapunov function is used as long as it remains the steepest descent direction (see [18] Section 3, Theorem 4.17). The fact that increasing the prices on a left-most min cut as long as possible is indeed a special case of the algorithm of Shioura follows by Lemma 9.

The step length in this algorithm is determined by a binary search. This is possible since by Shioura ([18] Proposition 4.16) whenever a steepest descent direction and thus a min cut becomes infeasible one of the following two situations happens: either the support increases, or the slope with which the Lyapunov
function changes decreases. Due to the resulting monotonicity we can apply binary search to determine the step length.

This observation also implies a bound on the number of price raising steps, similar to [18], which is the number of different objects times the value of the slope in the first step. For linear valuation functions we can characterize this more concretely.

**Proposition 3.** The number of iterations of the adapted Price Raising Algorithm for linear valuation functions is upper bounded by $|\Omega| \cdot \sum_{j \in B} d_j$.

**Proof.** As noted above, whenever the left most min cut changes, either the support increases or the slope with which the Lyapunov function changes decreases. This yields that after the support changed at most $|\Omega|$ times, the slope needs to decrease by a value of at least one. Proposition 2 bounds this slope by $\sum_{j \in B} d_j$.

**Remark 2.** Note that this gives the polynomial time bound $O(|\Omega|^2)$ on the number of iterations for single-unit matching markets.

5 Conclusion and outlook

The literature on multi-unit auctions is broad and a lot of results are known for the general case with strong substitute valuations. In this work we aimed to show the connection of the existing work and especially focus on multi-unit auctions with linear valuation functions. For this model, we present a network interpretation of an ascending auction. We show that by iteratively raising the prices on a left-most min cut we can compute buyer-optimal Walrasian prices. The algorithm itself is pseudo-polynomial, but each update step runs in strongly polynomial time. Furthermore, we show that when increasing the step length as far as possible, the time bound improves, but remains pseudo-polynomial. We have not found an example that needs exponentially many iterations when using the adapted step length, so the question of whether the iterative ascending auction algorithm can be modified to run in polynomial time remains open.

Moreover, we provide a sensitivity analysis if the demand or the supply changes. It remains open whether a similar analysis is possible if the valuation functions are perturbed. However, the number of iterations necessary to reach buyer-optimal Walrasian prices for a slightly perturbed instance is not polynomially bounded for the perturbations we considered. Are there perturbations where the algorithm reaches buyer-optimal Walrasian prices again with only constantly many, or only polynomially many update steps? A generalization of our results to the strong substitute case is also open. One way to achieve this is to understand the connection between minimal competitive prices and buyer-optimal Walrasian prices.

References

20. Walras, L.: Éléments d’économie politique pure ou théorie de la richesse sociale. F. Rouge (1874)
A Appendix

A.1 Structure of preferred bundles

Note that a preferred bundle for buyer $j$ with respect to prices $p$ can easily be computed via the following greedy procedure: Initially, $j$’s preferred bundle is empty, i.e., $x_{ij} = 0$ for all $i \in \Omega$, and the residual demand $d_j'$ is equal to $d_j$. While $d_j' > 0$ and there exist items $i$ of non-negative payoff $v_{ij} - p(i)$, buyer $j$ selects an item $i$ of maximum payoff and buys as many copies of $i$ as possible until either the capacity $b_i$ or the residual demand $d_j'$ is reached. That is, the greedy procedure assigns $x_{ij} = \min\{b_i, d_j'\}$ and updates the residual demand to $d_j' = d_j - \sum_{i \in \Omega} x_{ij}$. This by no means generally results in a unique solution, on contrary a player might have multiple preferred bundles. To understand their structure better, we divide the objects in those which are part of every preferred bundle and those between which a player is indifferent.

We distinguish between three cases. In the first case the preferred bundle is unique, while in the other two cases there are multiple preferred bundles. We can determine if the constructed preferred bundle is unique if we consider the item $k$ that $j$ buys last in the greedy algorithm.

The preferred bundle is unique if $v_{kj} - p(k) > 0$ and if there is no item available with the same payoff as $k$, i.e., there is no unsold item of an object $i$ with $v_{ij} - p(i) = v_{kj} - p(k)$. Then we define

$$
\Omega'_j(p) := \{i \in \Omega \mid v_{ij} - p(i) \geq v_{kj} - p(k)\},
$$
$$
\Omega''_j(p) := \emptyset,
$$
$$
\Omega'''_j(p) := \emptyset.
$$

If there is another item available with payoff $v_{ij} - p(i) = v_{kj} - p(k) > 0$ which is not sold yet, there are multiple preferred bundles, since $j$ can choose any of these to “fill up” the bundle. Note that the available item could be one of object $k$ as well, so that the preferred bundle might be unique in the number of items of each object. We define

$$
\Omega'_j(p) := \{i \in \Omega \mid v_{ij} - p(i) > v_{kj} - p(k)\},
$$
$$
\Omega''_j(p) := \{i \in \Omega \mid v_{ij} - p(i) = v_{kj} - p(k)\},
$$
$$
\Omega'''_j(p) := \emptyset.
$$

If the item $k$ has payoff $v_{kj} - p(k) = 0$ there are multiple preferred bundles as well, since $j$ is indifferent between buying an item with payoff zero or not. We define

$$
\Omega'_j(p) := \{i \in \Omega \mid v_{ij} - p(i) > v_{kj} - p(k)\},
$$
$$
\Omega''_j(p) := \emptyset,
$$
$$
\Omega'''_j(p) := \{i \in \Omega \mid v_{ij} - p(i) = 0\}.
$$

In total this means, $\Omega'_j(p)$ is the set of objects that buyer $j$ values most and which she wants to buy completely under current prices. Whereas $\Omega''_j(p)$ is the set of objects among which she is indifferent and between which she will choose to “fill up” her bundle to meet her demand. The objects in $\Omega'''_j(p)$ yield payoff zero, so she is indifferent whether to buy them. Thus, in every preferred bundle, buyer $j$ buys $h_{j'} := \sum_{i \in \Omega'_j} b_i$ items in total from objects in $\Omega'_j$ and $h_{j''} := \min\{\sum_{i \in \Omega''_j} b_i, \, d_j - \sum_{i \in \Omega'_j} b_i\}$ items in total from objects in $\Omega''_j$. Note that the minimum is attained in the former term only if $\Omega''_j = \emptyset$. In addition, there might be up to $h_{j'''} := \min\{\sum_{i \in \Omega'''_j} b_i, \, d_j - \sum_{i \in \Omega'_j} b_i\}$ items of objects with zero payoff in a preferred bundle.
A.2 Sketch of the auxiliary network

Fig. 2. Sketch of the auxiliary network. Buyer $j_1$’s preferred bundle is completely unique or the preferred bundles may contain items with payoff 0; she wants to buy as much as she can of $i_1$ and $i_2$. Buyer $j_2$’s preferred bundles always include the maximum amount of object $i_1$ but she is indifferent between $i_2$ and $i_\ell$ to fill up her bundle. Similarly, buyer $j_k$’s preferred bundles all consist of as much items of type $i_2$ and $i_\ell$ and are filled up as much as possible with items of type $i_1$ or $i_n$. Buyer $j_m$’s preferred bundles are just any combination of items $i_\ell$ and $i_n$ that meet her demand.

A.3 Properties of the prices - proofs of Lemma 2-5

Lemma 2. There is no overdemanded set for prices $p$ if there exists a flow $f$ in the network $G(p)$ of value $\text{val}(f) = \sum_{j \in B' \cup B''} h_j$.

Proof. Consider a flow $f$ in $G(p)$ of value $\text{val}(f) = \sum_{j \in B' \cup B''} h_j$ and some set $I \subseteq \Omega$. According to the flow conservation, for each vertex $j \in B' \cup B''$ we have

$$\sum_{i \in \Gamma^+(j) \cap I} f_{ji} = f_{sj} - \sum_{i \in \Gamma^+(j) \setminus I} f_{ji} \geq \max \left\{ 0, h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\}.$$ 

If we sum over all $j \in \Gamma^-(I)$ we get

$$\sum_{j \in \Gamma^-(I)} \max \left\{ 0, h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\} \leq \sum_{j \in \Gamma^-(I)} \sum_{i \in \Gamma^+(j) \setminus I} f_{ji} = \sum_{(j,i) \in \Gamma^+(I)} f_{ji} = \sum_{i \in I} f_{it} \leq \sum_{i \in I} b_i.$$ 

Thus, $I$ is not an overdemanded set.

Lemma 3. Let $C$ be a left-most min cut in $G(p)$ and $I := C \cap \Omega$. It holds that

$$C = \{s\} \cup \left\{ j \in B' \cup B'' \mid h_j > \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\} \cup I.$$
Lemma 5. For $j \in \Gamma^+(s) \setminus I$.

**Proof.** Let $C \subseteq T$ be the left-most min cut of $G(p)$. Define $I := C \cap \Omega$ and $T := C \cap (B' \cup B'')$. The capacity of $C$ is given by

$$\text{cap}(C) = \sum_{i \in I} b_i + \sum_{j \in T} \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} + \sum_{j \in (B' \cup B'') \setminus T} h_j.$$  \hspace{1cm} (3)

By the Max-Flow-Min-Cut Theorem, and Lemma 2, it holds that

$$\text{cap}(C) < \sum_{j \in B' \cup B''} h_j.$$  \hspace{1cm} (4)

By combining and rearranging (3) and (4) we get the following chain of inequalities:

$$\sum_{i \in I} b_i < \sum_{j \in T} h_j - \sum_{j \in T} \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} = \sum_{j \in T} \left( h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right) \leq \sum_{j \in T} \max \left\{ 0, h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\}.$$  

For $j \in T$ it holds, by the structure of $C$ (Lemma 3), that $h_j \geq \sum_{i \in \Gamma^+(j) \setminus I} g_{ij}$. Since the outflow of $j$ in $G(p)$ is at least as large as the inflow by construction, there exist an $i \in I$ with $g_{ij} > 0$. Thus it holds that $T \subseteq \Gamma^+(I)$. Hence, $I$ is an overdemanded set by definition. \hfill \Box

**Lemma 4.** Given prices $p$, if there is any overdemanded set, then $I := C \cap \Omega$ is an overdemanded set, where $C$ is the left-most min cut of the network $G(p)$.

**Proof.** Let $C$ be the left-most min cut of $G(p)$. Define $I := C \cap \Omega$ and $T := C \cap (B' \cup B'')$. The capacity of $C$ is given by

$$\text{cap}(C) = \sum_{i \in I} b_i + \sum_{j \in T} \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} + \sum_{j \in (B' \cup B'') \setminus T} h_j.$$  \hspace{1cm} (3)

By the Max-Flow-Min-Cut Theorem, and Lemma 2, it holds that

$$\text{cap}(C) < \sum_{j \in B' \cup B''} h_j.$$  \hspace{1cm} (4)

By combining and rearranging (3) and (4) we get the following chain of inequalities:

$$\sum_{i \in I} b_i < \sum_{j \in T} h_j - \sum_{j \in T} \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} = \sum_{j \in T} \left( h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right) \leq \sum_{j \in T} \max \left\{ 0, h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\}.$$  

For $j \in T$ it holds, by the structure of $C$ (Lemma 3), that $h_j \geq \sum_{i \in \Gamma^+(j) \setminus I} g_{ij}$. Since the outflow of $j$ in $G(p)$ is at least as large as the inflow by construction, there exist an $i \in I$ with $g_{ij} > 0$. Thus it holds that $T \subseteq \Gamma^+(I)$. Hence, $I$ is an overdemanded set by definition. \hfill \Box

**Lemma 5.** The cut $\tilde{C} = \{ s \} \cup \left\{ j \in B' \cap B'' \mid h_j > \sum_{i \in \Gamma^+(j) \setminus I_2} g_{ij} \right\} \cup I_2$ is a min cut and $\tilde{C} \subseteq C$.

**Proof.** This claim is used to show Theorem 2. Remember, we assumed that $q$ is a competitive price vector and that $t$ is the last iteration of the Price-Raising Algorithm where $p_t \leq q$ and where $I = C \cap \Omega$ is given by the left-most min cut $C$ in $G(p_t)$. We defined

$$I = \{ i \in \Omega \mid p_t(i) < p_{t+1}(i) \}, \quad I_1 = \{ i \in I \mid p_t(i) = q(i) \}, \quad I_2 = \{ i \in I \mid p_t(i) < q(i) \}$$

and we fixed the network properties such as $g_{ij}$, $h_j$ and also $j'$ and $j''$ of the prices $p_t$.

Now consider the buyers who demand the items in $I$ such that $I$ becomes overdemanded. Therefore, we define the following sets:

$$T = \left\{ j \in B' \cup B'' \mid h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} > 0 \right\}, \quad T_1 = \left\{ j \in T \mid g_{ij} > 0 \text{ for all } i \in I_1 \right\}, \quad T_2 = T \setminus T_1.$$  

**Claim.** It holds

$$\sum_{i \in I_1} b_i \geq \sum_{j \in T_1} \min \left\{ h_j - \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} , \sum_{i \in \Gamma^+(j) \setminus I} g_{ij} \right\}.$$  \hspace{1cm} (5)
Proof. First, consider the demand that \( j' \) has on objects in \( I_1 \) at price \( p_t \), i.e., \( \sum_{i \in \Gamma^+(j') \cap I_1} g_{ij'} \). Comparing \( q \) and \( p_t \), remember that the price of any item only increases, i.e., \( p_t(i) \leq q(i) \), while the prices in \( I_1 \) remain the same. That is why the buyer \( j \) likes to buy at prices \( q \) at least the same amount of items from objects in \( I_1 \) as at prices \( p_t \).

Next, we consider the demand \( j'' \) has on objects in \( I_1 \) at prices \( p_t \). Recall that \( j'' \) gets the same utility from every item. Comparing \( p \) and \( q \) we have that the prices on \( I_1 \) remain constant while the prices on \( I_2 \) strictly increase. Thus \( j'' \) likes to fill up the preferred bundle with items in \( I_1 \) and maybe with items outside of \( I \) before buying the items in \( I_2 \). Note however that for prices \( q \) it is not clear to which copy of buyer \( j \) this demand is assigned. Since furthermore, we have that buyer \( j \) cannot buy more objects than available in \( I_1 \) we get the following lower bound on the demand reassigned from a buyer \( j'' \) for prices \( p_t \) to buyer \( j \) for prices \( q \):

\[
\min \left\{ h_{j''} - \sum_{i \in \Gamma^+(j'') \cap I_1} g_{ij''} , \sum_{i \in \Gamma^+(j'') \cap I_1} g_{ij''} \right\}.
\]

Finally, if we sum up the demands of all \( j' \in B' \) and \( j'' \in B'' \) at prices \( q \) we obtain the following lower bound of the total demand of all buyer in \( j \in T_1 \)

\[
\sum_{j \in T_1} \min \left\{ h_j - \sum_{i \in \Gamma^+(j) \cap I_1} g_{ij} , \sum_{i \in \Gamma^+(j) \cap I_1} g_{ij} \right\}.
\]

The set \( I_1 \) is not overdemanded at price \( q \). Thus the demand of items in \( I_1 \) at price \( q \) is smaller or equal than the total supply of all objects in \( I_1 \). Using the bound of the total demand of all buyers in \( T_1 \), we obtain the inequality of the claim.

Now, we can show that \( \tilde{C} \) is a min cut too by comparing the capacities of \( \tilde{C} \) and the cut \( C \). According to Lemma \( 3 \) the cut \( C \) is given by

\[
C = \{s\} \cup \left\{ j \in B' \cap B'' \mid h_j > \sum_{i \in \Gamma^+(j) \cap I_1} g_{ij} \right\} \cup I.
\]
By definition it follows that $\tilde{C} \subseteq C$. So it remains to show that $\tilde{C}$ is a min cut as well by comparing its value with the value of the cut $C$:

$$\text{cap}(C) - \text{cap}(\tilde{C}) = \left( \sum_{i \in I} b_i + \sum_{j \in B \cup B''} \min \left\{ h_j, \sum_{i \in I^+(j) \setminus I} g_{ij} \right\} \right) - \left( \sum_{i \in I_2} b_i + \sum_{j \in B' \cup B''} \min \left\{ h_j, \sum_{i \in I^+(j) \setminus I_2} g_{ij} \right\} \right)$$

$$= \sum_{i \in I_1} b_i + \sum_{j \in T} \left( \sum_{i \in I^+(j) \setminus I} g_{ij} - \min \left\{ h_j, \sum_{i \in I^+(j) \setminus I_2} g_{ij} \right\} \right)$$

$$= \sum_{i \in I_1} b_i - \sum_{j \in T} \min \left\{ h_j - \sum_{i \in I^+(j) \setminus I} g_{ij}, \sum_{i \in I^+(j) \setminus I_1} g_{ij} \right\} \geq 0.$$ 

Therefore $\tilde{C}$ is a min cut too. 

\[\qed\]

### A.4 Properties of the allocation - proof of Lemma 6

**Lemma 6.** Given a price vector $p$ computed in some iteration of the Price-Raising Algorithm starting with prices $p_0 = 0$, then there is a max flow $f$ in $G(p)$ with corresponding allocation $x \in \mathbb{N}^{I \times B}$ such that

1. $\sum_{j \in B} x_{ij} = b_i$ if $p(i) > 0$,
2. $x_{ij}$ is a subset of a preferred bundle of buyer $j$.

**Proof.** We prove the lemma by induction on the number of iterations $t$. Assume first for the induction base that $t = 0$ and thus also $p = 0$. In this case any feasible assignment given by a max flow fulfills both properties. For the induction step assume we are given a max flow $f_t$ in $G(p_t)$ and a corresponding extended allocation $x^t$ with the wished properties for the price vector $p_t$. Let $C$ be the left-most min cut in $G(p_t)$. Let further $I := C \cap \Omega$ and $T := C \cap (B' \cup B'')$. Remember that

$$p_{t+1}(i) = \begin{cases} p_t(i) & \text{for all } i \notin I, \\ p_t(i) + 1 & \text{for all } i \in I. \end{cases}$$

Thus, all items that have been in a preferred bundle of buyer $j$ in iteration $t$ and that are not part of set $I$ remain in a preferred bundle. More formally, there exists a preferred bundle $x^t_{ij} = \tilde{x}^{t+1}_{ij} |_{\Omega \setminus I}$ such that $\tilde{x}^{t+1}_{ij} |_{\Omega \setminus I} \geq x^t_{ij} |_{\Omega \setminus I}$.

Let $f_{\Omega \setminus I} := f_t |_{\Omega \setminus I}$ be the max flow in $G(p_t)$ restricted to the set $\Omega \setminus I$. Note that for all objects in $\Omega \setminus I$ with positive price which are not fully assigned by $f_{\Omega \setminus I}$ there are sellers willing to buy these items at prices $p_t$ by assumption. Thus these would buy the items at prices $p_{t+1}$ as well.

To assign the items in $I$ we consider the flow network $G(p_t)$ restricted to the set of buyers in $T$ and the set of items $I$. Moreover, the capacity on the $s$-leaving arcs is reduced. More formally, in the network $\tilde{G}$ we are given a source $s$, a sink $t$, one node per buyer $j \in T$ and one node per object $i \in I$. An arc $(s, j)$ gets capacity $h_j - \sum_{i \in I^+(j) \setminus I} g_{ij}$. By definition of $T$ we know that this is positive. The arcs $(i, j)$ get capacity $g_{ij}$ and the arcs $(i, t)$ get capacity $b_i$. Then we construct a max flow $\tilde{f}_I$ in $\tilde{G}$.

**Claim.** The max flow in $\tilde{G}$ saturates the arcs $(i, t)$ for all $i \in I$, i.e., $\text{val}(\tilde{f}) = \sum_{i \in I} b_i$. 

\[\qed\]
Proof. Assume for contradiction that such a flow does not exist. We then show a contradiction to the choice of $C$ in the iteration step of the algorithm. Clearly, if no such flow exists in $\tilde{G}$, there is a cut $\tilde{C}$ with $\text{cap}(\tilde{C}) < \sum_{i \in I} b_i$. Let $I := I \cap \tilde{C}$ and $T := T \cap \tilde{C}$. We can describe the value of $\tilde{C}$ in the following way

$$\sum_{i \in I} b_i - \sum_{j \in T \setminus \tilde{T}} h_j - \sum_{j \in \tilde{T}} \sum_{i \in I} g_{ij} + \sum_{j \in T \setminus \tilde{T}} \sum_{i \in I} g_{ij} > 0.$$ \hfill (6)

This yield

$$\sum_{i \in I \setminus \tilde{I}} b_i - \sum_{j \in T \setminus \tilde{T}} h_j - \sum_{j \in \tilde{T}} \sum_{i \in I} g_{ij} + \sum_{j \in T \setminus \tilde{T}} \sum_{i \in I \setminus \tilde{I}} g_{ij} > 0.$$ \hfill (6)

With this at hand we consider now network $G(p_t)$ and show that the existence of $\tilde{C}$ is a contradiction to the choice of $C$. Therefore we compare the capacities of both cuts in $G(p_t)$ (a sketch of the cuts can be found in Figure 4).

$$\text{cap}(C) - \text{cap}(\tilde{C}) = -\sum_{j \in T \setminus \tilde{T}} h_j - \sum_{j \in \tilde{T}} \sum_{i \in I} g_{ij} + \sum_{j \in T \setminus \tilde{T}} \sum_{i \in I \setminus \tilde{I}} g_{ij} + \sum_{i \in I \setminus \tilde{I}} b_i \geq 0.$$ \hfill (6)

Since this is contradiction, there is a flow $\tilde{f_t}$ of value $\sum_{i \in I} b_i$ in $\tilde{G}$. \hfill \blacksquare

The demands in $\tilde{G}$ are given by $h_j - \sum_{i \in I} g_{ij}$, so we can conclude that the corresponding buyers like to buy the items at prices $p_{t+1}$ as well. Note that at prices $p_{t+1}$ there might be payoff zero assignments. Now we combine the allocation $x^t|_{\Omega \setminus I}$ and the allocation $\tilde{x}$ given by the flow $\tilde{f_t}$, and obtain the feasible allocation

$$\bar{x}_{ij} := \begin{cases} x^t_{ij} & \text{for all } i \notin I, \\ \tilde{x}_{ij} & \text{for all } i \in I. \end{cases}$$

Since we use the restricted network to allocate the items in $I$, all buyers are assigned to a subset of a preferred bundle at prices $p_{t+1}$ in $\bar{x}$. This follows since to determine $\tilde{x}$ the capacity was reduced to $h_j - \sum_{i \in I} g_{ij}$. For prices $p_t$ these demands were only assigned to items in $I$. Thus, there is no interference between $x^t$ and $\bar{x}$ in the sense that a buyer exceeds her demand. Furthermore, all objects with positive price are fully assigned by definition of $x^t$ and $\bar{x}$.
We adapt the allocation $\bar{x}$ such that the corresponding flow is a max flow in $G(p_{t+1})$. Therefore we first construct the corresponding flow $\bar{f}$ in $G(p_{t+1})$ by leaving out the payoff zero assignments. Then we augment $\bar{f}$ to a max flow $f_{x^{t+1}}$. Let $x^{t+1}$ denote the corresponding allocation. Then, we use the zero-payoff assignments in $\bar{x}$ to fill up $x^{t+1}$ with assignments to unsold items of objects with positive price. Note that the flow on the $t$-entering arcs does not decreases in comparison to $\bar{f}$. Therefore all objects with positive price are sold, and thus the flow $f_{x^{t+1}}$ together with the allocation $x^{t+1}$ fulfills the desired properties.

\[\square\]

A.5 Social optimality - proof of Theorem 5

**Theorem 5.** The computed buyer-optimal Walrasian equilibrium $(p^*, x^*)$ is socially optimal in the sense that the total payoff of buyers and sellers together is maximized.

**Proof.** Let $p$ be the prices computed by the Price-Raising Algorithm. Assume that the demand and the supply is balanced, otherwise add a dummy object or a dummy buyer and note that this does not change the prices or the allocation. Let maximal value of $(LP)$ be $V_J$, i.e.,

$$V_J := \max \left\{ \sum_{i \in \Omega} (v_{ij} - p(i)) \cdot x_{ij} \mid \sum_{i \in \Omega} x_{ij} \leq d_j, 0 \leq x_{ij} \leq b_i \text{ for all } i \in \Omega \right\}$$

(7)

For each feasible assignment $x$ it holds $\sum_{i \in \Omega} (v_{ij} - p(i)) x_{ij} \leq V_J$. Hence, we have

$$\max \{ \sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i)) x_{ij} \mid x \text{ is a feasible solution} \} \leq \sum_{j \in B} V_J.$$  

Since every buyer gets one of her preferred bundles in the given allocation $x^*$, this allocation fulfills

$$\sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i)) x^*_{ij} = \sum_{j \in B} V_J.$$  

Thus $x^*$ maximizes $\sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i)) x_{ij}$ over all feasible solutions $x$. Furthermore, it holds

$$\sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i)) x_{ij} = \sum_{j \in B} \sum_{i \in \Omega} v_{ij} x_{ij} - \sum_{i \in \Omega} \left[ p(i) \cdot \sum_{j \in B} x_{ij} \right] = \sum_{j \in B} \sum_{i \in \Omega} v_{ij} x_{ij} - \sum_{i \in \Omega} p(i) \cdot \left. x_{ij} \right|_{=b_i}$$

for all perfect allocations. Thus, since $x^*$ is a perfect allocation, it maximizes

$$\sum_{j \in B} \sum_{i \in \Omega} v_{ij} x_{ij} = \sum_{j \in B} \sum_{i \in \Omega} (v_{ij} - p(i)) x_{ij} + \sum_{i \in \Omega} p(i) \cdot \sum_{j \in B} x_{ij}.$$  

Hence, $x^*$ maximizes the total payoff of buyers and seller together.\[\square\]

A.6 Comparison - proof of Lemma 9

**Lemma 9.** Given prices $p$, the overdemanded set $I$ determined by the left-most min cut in $G(p)$ minimizes $L(p + \chi X)$ among all $X \subseteq \Omega$.

**Proof.** A set $X \subseteq \Omega$ minimizes $L(p + \chi X)$ if and only if it maximizes

$$L(p) - L(p + \chi X) = \sum_{j \in B} (V_j(p) - V_j(p + \chi X)) - \sum_{i \in X} b_i.$$  

20
Now we consider again the constructed auxiliary networks $G(p)$ and $G(p + \chi_X)$ and the induced changes in capacities. With Proposition 3 we obtain that $X \subseteq \Omega$ maximizes

$$\sum_{j \in I^-}(X) \max \{0, h_j - \sum_{i \in I^+(j) \setminus X} g_{ij} \} - \sum_{i \in X} b_i \quad (8)$$

Given an $X$ that minimizes $L(p + \chi_X)$, we construct the cut

$$C_X = \{s\} \cup \{j \in B' \cup B'' : h_j > \sum_{i \in I^+ (j) \setminus I} g_{ij}\} \cup X.$$ 

The structure of the cut, determined by our algorithm from set $I$, is given by

$$C = \{s\} \cup \{j \in B' \cup B'' : h_j > \sum_{i \in I^+ (j) \setminus I} g_{ij}\} \cup I.$$ 

We can show that $C_X$ is also a min cut:

$$0 \geq \text{cap}(C) - \text{cap}(C_X)$$

$$= \sum_{i \in I} b_i + \sum_{j \in B' \cup B''} \min \{h_j, \sum_{i \in I^+(j) \setminus I} g_{ij}\} - \sum_{i \in X} b_i - \sum_{j \in B' \cup B''} \min \{h_j, \sum_{i \in I^+(j) \setminus X} g_{ij}\}$$

$$= \sum_{i \in I} b_i - \sum_{j \in B' \cup B''} \left( \max \{0, h_j - \sum_{i \in I^+(j) \setminus I} g_{ij}\} - h_j \right) - \sum_{i \in X} b_i + \sum_{j \in B' \cup B''} \left( \max \{0, h_j - \sum_{i \in I^+(j) \setminus X} g_{ij}\} - h_j \right)$$

$$= \left( \sum_{j \in B' \cup B''} \max \{0, h_j - \sum_{i \in I^+(j) \setminus X} g_{ij}\} - \sum_{i \in I} b_i \right) - \left( \sum_{j \in B' \cup B''} \max \{0, h_j - \sum_{i \in I^+(j) \setminus I} g_{ij}\} - \sum_{i \in I} b_i \right)$$

$$\geq 0,$$

where the first inequality follows from the fact that $C$ is a min cut and the last inequality follows from (8), i.e., that $X$ maximizes the term. Hence, $C_X$ is a min cut. Moreover the choice $X = C \cap \Omega$ minimizes $L(p + \chi_X)$, since equality holds in the chain of inequalities.

**A.7 Directly computing the prices**

Although this result is already implied by [16] we state and prove it here for completeness.

**Proposition 4.** In a multi-unit auction market instance where all buyers have strong substitute valuation functions, the buyer-optimal Walrasian prices can be computed in weakly polynomial time.

**Proof.** The buyer-optimal Walrasian prices are the minimal minimizer of the Lyapunov function. Since Murota et al. ([15] Theorem 1.6) show that the function is $L^2$-convex, it is possible to apply e.g. the steepest descent scaling algorithm (see [17] Section 4.2.) to find a minimizer of the Lyapunov function. Since in our case the function only attains non-negative integer values, we can use a slight perturbation such that the minimal minimizer is unique but no value is shifted by more than $\frac{1}{2}$. This can be done by adding an additive term of $\varepsilon \sum_{i \in \Omega} p(i)$, where we choose $\varepsilon := (2 \sum_{i \in \Omega} \max_{j \in B} v_{ij})^{-1}$. In more detail, we define a new function as follows

$$\bar{L}(p) = L(p) + \varepsilon \sum_{i \in \Omega} p(i).$$

Since we only add a modular function the function remains $L^2$-convex.

Moreover, since $\max_{j \in B} v_{ij}$ is an upper bound on the price of object $i$ as claimed, $\varepsilon$ is small enough such that $[\bar{L}(p)] = L(p)$. This means that the set of minimizers is perturbed in such a way that the old minimal minimizer is the unique minimizer of the new function. For this procedure it is important that the vector is component-wise minimal and thus the unique vector minimizing $\varepsilon \sum_{i \in \Omega} p(i)$. Moreover, it is important that the minimal minimizer is integral which follows by Ausubel ([11] Appendix B, Corollary to Proposition 1).