Efficient Parking Allocation as Online Bipartite Matching with Posted Prices

Reshef Meir*
Hebrew University of Jerusalem
reshef.meir@mail.huji.ac.il

Yiling Chen
Harvard University
yiling@seas.harvard.edu

Michal Feldman
Harvard University and Hebrew University of Jerusalem
mfeldman@seas.harvard.edu

ABSTRACT

We study online bipartite matching settings inspired by parking allocation problems, where rational agents arrive sequentially and select their most preferred parking slot. In contrast to standard online matching setting where edges incident to each arriving vertex are revealed upon its arrival, agents in our setting have private preferences over available slots. Our focus is on natural and simple pricing mechanisms, in the form of posted prices. On the one hand, the restriction to posted prices imposes new challenges relative to standard online matching. On the other hand, we employ specific structures on agents' preferences that are natural in many scenarios including parking. We construct optimal and approximate pricing mechanisms under various informational and structural assumptions, and provide approximation upper bounds under the same assumptions. In particular, one of our mechanisms guarantees a better approximation bound than the classical result of Karp et al. [10] for unweighted online matching, under a natural structural restriction.

Categories and Subject Descriptors: I.2.11 Artificial Intelligence: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms: Theory, Algorithms, Economics

Keywords: Online Matching, Parking, Mechanism design, Posted prices

Error correction.

Unfortunately, Theorem 18 (which is one of our main results) is wrong. Please do not cite.

1. INTRODUCTION

In recent years, smart parking systems are being deployed in an increasing number of cities. Such systems allow commuters and visitors to see in real time, using cellphone applications or other digital methods, all available parking slots and their prices.[1] At the same time, dynamic pricing becomes more popular in various domains [12], including for example congestion tolls [19], smart grids [16], and electric vehicle charging [7, 17].

Parking allocation as matching. We consider the problem of maximum online bipartite matching with dynamic posted prices, motivated by the real-world challenge of efficient parking allocation. As in standard online matching setting, one side of the bipartite graph (representing the parking slots) is known in advance. The vertices of the other side (representing commuters or agents) arrive sequentially and each demand a slot. However, in contrast to standard online matching setting where edges incident to each arriving vertex are revealed upon its arrival, the preference of an agent over available slots is private and not completely revealed. Future systems may provide communication interfaces that will allow commuters to report their parking preferences [6]. As such interfaces are not yet available, and to avoid unnecessary complication, we focus on natural posted price mechanisms. Agents are assumed to be rational and select a slot based on their private preference and prices of slots at the time of arrival.

While private preferences of agents and the restriction to posted price mechanisms impose additional challenges relative to standard online matching, in the parking allocation domain there are some natural structures on agent preferences that can be exploited to achieve more efficient allocation. Specifically, we assume that every agent has a goal (e.g. her office building), and prefers parking slots closer to her goal, ceteris paribus. An agent's valuation of a parking slot depends on its distance to her goal. In this paper, we consider two natural single-parameter valuation schemes: MAXDISTANCE and LINEARCOST. In MAXDISTANCE, an agent is willing to accept any slot within a certain distance from her goal, while in LINEARCOST, an agent's valuation of a slot decreases linearly with the distance between the slot and her goal.

The objective of a system designer is to set up (dynamic) prices for available parking slots to prompt the most efficient allocation, in terms of *social welfare* — the total value of all agents who are allocated a slot. That is, the sole purpose of payments is to align the incentives of the agents with that of the society, rather than to make a profit. In cases where the optimal allocation cannot be achieved in the online setting, we seek the best possible approximation ratio that can be attained by posted price mechanisms.

Although our problem is motivated by the application of parking allocation, the general setup is applicable to other domains with private preferences that have similar structural restrictions. An example is online procurement, where each agent has some ideal product or service in mind (the goal), but must select from a limited range of available options based on their similarity to her goal and current prices. However, in such domains, the system designer may arguably be more interested in maximizing revenue than optimizing social welfare, which is the focus of this paper.

*Part of the research took place while the author was a visitor at the Center for Research on Computation and Society in Harvard University.

[1] San Francisco and other cities in California are already supporting such an application for some time now. Other cities are rapidly catching up [15, 14].

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

1.1 Related work

"Smart parking" has attracted much attention in urban planning. For example, Geng and Cassandras [6] proposed a system asking each agent to report her maximum acceptable distance to her goal and maximum parking cost and leveraging integer programing to decide an allocation. Such systems do not consider the strategic nature of agents and have not yet provided theoretical guarantees on efficiency. Some related online allocation problems such as charging of electrical vehicles [7, 17] and WiFi bandwidth allocation [5] use auction-like mechanisms that are based on agents' reported type. The main difference in our approach is that it uses posted prices, which come with their pros and cons (in particular, we require no input or almost no input from the agents).

Matching. The parking allocation problem we study closely relates to maximum online matching in unweighted bipartite graphs, as defined by Karp et al. [10].[2] In fact, one variant of our problem coincides with it exactly. In this case, we can easily implement their well-known RANKING algorithm, using random posted prices. Karp et al. proved that RANKING achieves an approximation ratio of $1 - 1/e$, and that no online algorithm (and thus no pricing mechanism) can do better.

Some later work on online bipartite matching studied the best possible approximation ratio that can be guaranteed in several variants of the original problem, typically by varying the informational and distributional assumptions on arriving vertices [13, 4, 9]. The motivation behind some of these comes from the AdWords assignment problem. A setting where all slots reside on a line was also studied, albeit with a focus on *minimum matching* [11].

Weighted matchings. While the general problem of online matching with weights is quite difficult (even in bipartite graphs), better algorithms exist if certain restrictions are made. Aggarwal et al. [1] extended the result of Karp et al. [10] to vertex-weighted matchings, where every vertex on the *known* side (the parking slots in our case) has a weight. In one of our models, there are values (weights) attributed to the *unknown* vertices (the agents), in which case the approximation ratio may be unbounded. A different restriction on weights that has been considered - namely triangle inequality - has led to a $\frac{1}{3}$-approximation mechanism [8].

Allocation with posted prices. Chawla et al. [3] recently tackled a much more general challenge of resource allocation (not necessarily matching) using posted prices. They gave constant approximation bounds (between $\frac{1}{8}$ and $\frac{2}{3}$) for maximum revenue in a range of allocation problems.[3] Among other differences from our model, their model assumes that each arriving agent is sampled from some known distribution.

1.2 Our contribution

We study the parking allocation problem under MAXDISTANCE and LINEARCOST valuation schemes respectively and with various informational and structural assumptions.

For MAXDISTANCE, our contribution is two-fold. At the conceptual level, we isolate explicit structural and informational assumptions inspired by real-world parking allocation and establish connections to the well-studied online bipartite matching problem.

At the technical level, we provide several powerful, yet simple to implement, pricing mechanisms. We show that when the population (but not the order of arrival) is known in advance, an optimal mechanism exists provided that we have access to each agent's goal. For other variants of the problem we provide approximation mechanisms and approximation upper bounds. Our results for the MAXDISTANCE valuation scheme are summarized in Table 1.

Notably, we show that under a plausible structural restriction, there is a mechanism that guarantees a 0.682 approximation ratio in the "unweighted" variant of the problem. It is better than the $1 - 1/e \cong 0.632$ approximation ratio provided by Karp et al. [10] for general unweighted matching problems. Further, in contrast to Karp et al., our pricing mechanism is deterministic.

For the more intricate LINEARCOST scheme, we focus on the case where both the population and the goals are known. Using results from ad auction theory, we provide a pricing mechanism that guarantees the optimal social welfare.

Some proofs are omitted due to space constraints, but can be found in the full version of this paper.Some proofs have been deferred to the appendix to allow continuous reading.

2. MODEL

An instance of a parking allocation problem is a tuple $H = \langle \mathcal{S}, N, \pi \rangle$, consisting of a *structure*, a *population* and an arrival order. Specifically, the structure $\mathcal{S}$ is given by a tuple $\mathcal{S} = \langle S, G, d \rangle$, where S is a finite set of parking slots, G is a finite set of goals, and d is a distance metric over $S \cup G$. We denote $m = |S|, k = |G|$. The population consists of a set of agents N with their preferences, where $n = |N|$. Finally, π is a permutation of $[n]$, indicating the order of arrival.

The preference of an agent $j \in N$ is given by (g_j, v_j), where $g_j \in G$ is the goal of agent j, and $v_j : S \to \mathbb{R}$ is a function specifying the valuation of agent j for being allocated some slot s. We assume that $v_j(s)$ is *distance based*. More specifically, $v_j(s) = \phi_j - C_j(s)$, where ϕ_j is a constant and $C_j(s)$ is some non-decreasing function of $d(g_j, s)$. ϕ_j can be interpreted as the cost of using a default option, in case the agent j is not allocated any slot. Such a default option might be a large parking lot that is always available, but is either expensive or inconveniently located. Throughout the paper we assume that this cost depends only on the goal and not on the identity of the agent, i.e., $\phi_j = \phi_g$ whenever $g_j = g$. A special case is $\phi_j = \phi_g = \phi$ for all agents, for example, when there is a single default option available for all goals.

An allocation of slots S to agents N is a matching $\sigma : N \to S \cup \{\emptyset\}$, specifying for each agent her allocated slot (or in the case of $\emptyset$, no slot is allocated). Given any $i, j \in N$, the allocation satisfies $\sigma(i) \neq \sigma(j)$ unless $\sigma(i) = \sigma(j) = \emptyset$. That is, each slot can be allocated to at most one agent.

A mechanism M maps an instance of a parking allocation problem to an allocation. Let M_H denote the allocation outputted by M for instance H. The *social welfare* achieved by mechanism M at instance H is defined as the sum of agent valuations at the allocation, i.e. $SW(M, H) = \sum_{j \in N} v_j(M_H(j))$. We emphasize that although the agent's decision is eventually based on the prices of slots, the social welfare is not influenced by monetary transfers.[4] For randomized mechanisms, we treat $SW(M, H)$ as the expected value over all realizations of prices and allocations.

We consider posted price mechanisms. Agents with private preferences arrive in sequence according to π and are presented with a posted price $p(s)$ for every available parking slot $s \in S$. The utility of an agent for selecting a slot s is quasi-linear, defined as $u_j(s) = v_j(s) - p(s)$. Agents are assumed to be rational; they select a slot to maximize their utility or reject all slots and use the

[2] This is yet another difference from online allocation settings such as EV charging, where the underlying optimization problem does not always resemble matching.

[3] Chawla et al. [3] claimed that the same bounds hold for maximum social welfare.

[4] Equivalently, we can count payments in and sum over all agents and the parking authority in calculating the social welfare.

default option if none of the slots provides nonnegative utility. The goal of the system designer is to design a posted price mechanism that maximizes social welfare.

Approximation ratio. We adapt the competitive model of Karp et al. [10] to evaluate pricing mechanisms. The structure $\mathcal{S}$ is common knowledge among all agents and the system designer. We allow our mechanism to flip coins when setting prices. An adversary who knows the mechanism selects a set of agents N (i.e. their preferences) and an arrival order π.[5] The performance of the mechanism is compared to that of the optimal allocation, in the worst selected instance. Formally, the *approximation ratio*[6] of a pricing mechanism M over a structure $\mathcal{S}$ is

$$AR_{\mathcal{S}}(M) = \min_{N} \min_{\pi} \frac{SW(M, \langle \mathcal{S}, N, \pi \rangle)}{opt(\langle \mathcal{S}, N \rangle)},$$

where SW is the *expected social welfare* taken over the randomization of the mechanism, and $opt(\langle \mathcal{S}, N \rangle)$ is the social welfare achieved by the optimal allocation for structure $\mathcal{S}$ and agents N. Note that $AR(M) \leq 1$, with equality if and only if the mechanism is optimal. Optimal allocations are w.l.o.g. deterministic.

Unless explicitly stated otherwise, we assume that the instances are "large enough". That is, the number of allocated slots in the optimal allocation goes to infinity. In particular, n and m are sufficiently large to ignore rounding issues.

Valuation Schemes. We will consider two valuation schemes of agents in this paper. An agent's preference under each scheme can be characterized by a single parameter on top of her goal.

MAXDISTANCE: In this scheme each agent j has a parameter m_j, specifying the maximum distance she is willing to walk. Thus if agent j is allocated slot $s_i \in S$, her valuation is $v_j(s_i) = \phi_j$ if $d(g_j, s_i) \leq m_j$, and 0 otherwise.

LINEARCOST: Each agent j incurs a cost c_j for walking a unit of distance. Thus, the valuation of a parking slot $s_i \in S$ for agent j is $v_j(s_i) = \phi_j - c_j d(g_j, s_i)$.

Informational assumptions. In some cases, we make simplified assumptions on what the system designer knows.

Assumption KP (Known Population): The size of N and the distribution of agent preferences are public information. That is, the system designer knows how many agents exist for what preference, but does not know the preference of any arriving agent.

Assumption KG (Known Goal): g_j is public information. E.g., each commuter has a chip in her car to identify her employer.

Assumption UV (Uniform Values): $\phi_j = \phi$ for all j.

For the purpose of comparing our results with standard results on online matching algorithms, we also define Assumption KT (Known Type), which means that the full preference of each arriving agent is public information. Clearly KT entails KG.

It is easy to see that under Assumptions KT+UV the parking allocation problem in the MAXDISTANCE setting is a special case of online bipartite matching. We will later see (Cor. 3) that the reverse is also true.

Structural restrictions. We will consider the following four classes of structures, ordered by their level of generality.

1. Structures with a single goal.
2. Structures with two goals, where all slots are scattered along an interval between them. That is, $d(s, g_1) = R - d(s, g_2)$ for all $s \in S$ and some constant R.
3. Layered structures. Coarsely speaking, this means every slot has several duplicates or near-duplicates (see details in Section. 4.3). Layered structures include for example slots scattered along the edges of sparse graphs, and structures where all slots are concentrated in several large parking lots.
4. General structures.

[5]Following Karp et al., this is a non-adaptive adversary.

[6]This ratio is sometimes referred to as *competitive ratio*.

3. GENERAL OBSERVATIONS

It is sometimes useful to decompose the allocation problem into two steps: find the right partition of space for the goals and optimally allocate space assigned to each goal to agents with that goal.

OBSERVATION 1. *Finding an optimal offline allocation is a special case of maximum weighted bipartite matching. Thus, under Assumption KP, the optimal allocation (and in particular an optimal partition of the space) can be found in polynomial time.*

To see this, suppose we define agents to be the vertices of the left side of the graph, and slots to be the vertices of the right side. We add an edge between every agent j and slot i, whose weight is the valuation of j for slot i. Then, an allocation is a matching and its social welfare is exactly the total weight of the matching. Maximum weighted matching can be found in polynomial time, e.g. by the Edmond-Karp algorithm.

According to Observation 1, the MAXDISTANCE model under Assumption UV is a special case of maximum cardinality (unweighted) matching in bipartite graphs. Our next result shows that they are equivalent.

LEMMA 2. *Let (I, S, E) be a bipartite graph with vertex sets I and S and edge set E. Then there is an instance of* MAXDISTANCE *where $d(s, g_i) \leq m_i$ if and only if $(i, s) \in E$.*

Given the last lemma, we have the equivalence of the online problems under Assumptions KT+UV.

COROLLARY 3. *Under the* MAXDISTANCE *model with Assumptions KT+UV, the parking allocation problem is equivalent to the online maximum cardinality matching problem.*

This is simply because if the preference of an arriving agent is known, we have the same information as in online matching. We can allocate any desired slot s to this agent by setting the price of s to 0, and prices of other slots to infinity. It follows that any algorithm or approximation upper bound for online algorithms in one domain (bipartite matching/parking allocation) immediately applies to the other as well.

Our next observation is that given a partition of space to k goals, $P = (S_1, \ldots, S_k)$, the online allocation problem reduces to a single goal problem provided that we have access to agents' goals.

OBSERVATION 4. *Suppose we have a pricing mechanism that finds an optimal allocation for a single goal. Then under Assumption KG, we have a pricing mechanism that implements the optimal allocation for any given partition P.*

Upon the arrival of an agent with goal g, we block all slots of $S_{g'}, g' \neq g$, and price the slots of S_g as if this is the entire space. Since our pricing for every set S_g yields an optimal allocation of these slots, we get the best possible allocation for P.

Thus, under Assumption KG we have the following recipe:

- Design an optimal online pricing mechanism for a single goal.
- Based on prior information, find a good partition of slots to goals (either optimal or approximately optimal). For example, by Observation 1, an optimal offline partition can be found under Assumption KP.

- Run the single goal mechanism for the appropriate goal whenever an agent arrives.

4. RESULTS FOR MAXDISTANCE

We first note that by the equivalence to matching, even trivial mechanisms work reasonably well if all agents have the same ϕ.

OBSERVATION 5. *Under Assumption UV, any* maximal *matching is a $\frac{1}{2}$-approximation. A maximal matching can be easily attained by setting prices of all slots to 0.*

A single goal. We next consider a restricted setting, in which there is a single goal g, with value ϕ. In this case, we sort all slots according to non-decreasing distance from g. Thus $d(g, s_i) \leq d(g, s_{i'})$ for all $i < i'$.

It is easy to see that if we take agents in an arbitrary order, and place j on the highest (i.e. most distant) slot s_i s.t. $d(g, s_i) \leq m_j$, then this allocation would be optimal. Indeed, if some m^* slots are allocated, then either all agents got slots; or all m^* slots closest to g are allocated, in which case there are no agents with $m_j > m^*$. The allocation in both cases is clearly optimal.

There is a very simple pricing mechanism that implements such an optimal allocation: We sort slots according to nondecreasing distance from g, and set prices to $p_i = (m - i)\epsilon$ for all i, with some $\epsilon < \phi/m$. We refer to this mechanism as *monotone pricing scheme*. Under these prices, each agent prefers the most distant slot s.t. $d(g, s_i) \leq m_i$.

By Observations 1 and 4, monotone pricing can be easily extended to any number of goals in arbitrary spaces.

COROLLARY 6. *Under Assumptions KP+KG, there is an optimal pricing mechanism.*

In the remainder of this section, we study the best approximation ratio that can still be guaranteed when these assumptions are relaxed. For easy comparison, the results are summarized in Table 1. Throughout this section, we use the notation $\alpha = \frac{\max_g \phi_g}{\min_{g'} \phi_{g'}} \geq 1$.

4.1 Two goals on an interval

Our next setting involves two goals, residing on the two ends of an interval containing all slots. We sort all slots by non-decreasing distance from g_1, and this is also a non-increasing distance from g_2. We assume, w.l.o.g. $\phi_1 \geq \phi_2$, thus $\phi_1 = \alpha\phi_2$.

We say that $P_t = (S_1, S_2)$ is a *threshold partition* for threshold t if $s_i \in S_1$ for all $i \leq t$ and $s_i \in S_2$ for all $i > t$.

LEMMA 7. *There is always an optimal threshold partition P_{t^*}. Moreover, w.l.o.g. t^* is exactly the maximal number of agents with goal g_1 that can be placed in the optimal allocation.*

PROOF. If there is an agent with goal g_1 getting a higher slot than some agent with goal g_2, we could just switch them. Also, if there are gaps on both sides of the threshold, we could push down the threshold t^*. If we could assign a slot to one more agent from goal g_1 when $\alpha > 1$, this would shift the threshold up by one, which would displace at most one agent of goal g_2. Since $\phi_1 \geq \phi_2$, this would weakly increase welfare. $\square$

LEMMA 8. *For any threshold partition P_t, we can implement with posted prices an allocation that is at least as good as P_t.*

PROOF. The mechanism THRESHOLD is defined as follows. We need each agent to select the most distant slot s_i from her goal g, s.t. $d(g, s_i)$ is bounded by both m_j and the threshold t. In other words, the slot closest to t s.t. $d(g, s_i) \leq \min\{m_j, d(g, t)\}$.

On arrival, we price available slot s_i by ϵd_i, where d_i is the *number of available slots* between s_i and t (not the distance); and ϵ is small enough such that, for all i, ϵd_i is less than ϕ_2 and if $\alpha > 1$ it is also less than $\phi_1 - \phi_2$. Moreover, if all slots in S_2 are full and $\alpha > 1$, we add ϕ_2 to the price of all slots in S_1.

Now, suppose that an agent with goal g_1 and maximum distance to walk m_j, denoted (g_1, m_j), arrives and selects s_i. There are three cases: (a) There is one cheapest slot closer than m_j, below t (i.e. on the "correct" side). Then this is the slot assigned to j by the optimal allocation anyway. (b) There are two cheapest slots, one on each side of t. Then one of these is the one from case (a), which is preferred by default since it is closer to g_1. (c) $s_i > t$ (but below m_j). This means that all slots $s < s_i$ are taken, and it cannot prevent future agents from being allocated slots above s_i. Thus, this new allocation is still optimal for the threshold t.

A similar argument works for agents with goal g_2, except that in case (c) all available slots belong to S_1 and thus cost more than ϕ_2. Therefore, agents with goal g_2 are never allocated slots $i \leq t$. $\square$

Given a threshold t, we can still implement an optimal allocation for t without knowing agents' goals. By Lemma 7, the optimal partition is indeed a threshold partition, we thus have the following.

COROLLARY 9. *Under Assumption KP, there is an optimal mechanism for two goals on an interval.*

We will later see that this no longer holds even in slightly more complex structures.

Unknown population. By Observation 5 there is a simple $\frac{1}{2}$-approximation mechanism under Assumption UV. However, when agents have different values, larger inefficiencies may occur. This holds even if agents' preference is known on arrival (i.e. the difficulty arises from the online setting).

PROPOSITION 10. Every *online algorithm under Assumption UV has a worst-case approximation ratio of at most $\frac{3}{4}$, even on an interval. If we relax Assumption UV, then the bound is at most $\frac{1}{2}$.*

PROOF. Consider the following two sequences of n agents, where $n = m$. The first $n/2$ agents (denoted N') are of type (g_1, n), with goal g_1 and maximum distance to walk n. They can be allocated any slot. Our two instances differ in the next $n/2$ agents (denoted N''). In H_1, we have $n/2$ agents of type $(g_2, \frac{1}{2}n)$. In H_2, we have $n/2$ agents of type $(g_1, \frac{1}{2}n)$. Note that $opt(H_1) = opt(H_2) = n$.

Let r_1, r_2 be the expected number of agents from N' that are allocated slots by the mechanism in half of the interval that is closer to g_1 and g_2, respectively. Since $r_1 + r_2 \leq |N'| = n/2$, at least one of them is at most $n/4$. We divide into two cases: (a) if $r_1 \leq n/4$, then on instance H_1 all of N'' are placed closer to g_2; (b) if $r_2 \leq n/4$, then on instance H_2 all of N'' are placed closer to g_1.

In both cases, the total number of allocated slots is at most $\frac{1}{2}n + n/4 = \frac{3}{4}n$. Thus for any mechanism M either $SW(M, H_1) \leq \frac{3}{4}n = \frac{3}{4}opt(H_1)$, or $SW(M, H_2) \leq \frac{3}{4}n = \frac{3}{4}opt(H_2)$.

Next, suppose that we drop Assumption UV, and set $\phi_1 \gg \phi_2$. Let N' contain n agents of type (g_2, n), and N'' contain n agents of type (g_1, n). Once again we define two instances H_1, H_2. In H_1, only N' arrive. In H_2, N' arrive and then N''.

Denote by r the expected number of agents from N' placed by the mechanism. If $r \leq n/2$ we are done since $SW(M, H_1) \leq \frac{1}{2}|N'| = \frac{1}{2}opt$. Otherwise, consider the performance of M on H_2. Note that since $\phi_1 \gg \phi_2$, we can practically ignore the type 2 agents in the welfare computation. However strictly less than $n - r \leq \frac{1}{2}n$ of the type 1 agents are placed in expectation, so the approximation is at most $\frac{1}{2}$. $\square$

To conclude the section, we present a mechanism that matches the upper bound on the interval without any assumption.

PROPOSITION 11. *There is a $\frac{1}{2}$-approximation mechanism for two goals on the interval.*

PROOF. We will prove that running the THRESHOLD mechanism for the threshold $\hat{t} = m/2$ provides us with a $\frac{1}{2}$-approximation. Let t^* be the minimal true optimal threshold. Let N_1, N_2 be the sets of agents from each goal that are allocated in the optimal allocation. By the Lemma 7, $|N_1| = t^*$. We divide in two cases. Note that $opt = \alpha|N_1| + |N_2|$.

If $t^* < \hat{t}$, then our mechanism will allocate to all of N_1, as none of them is restricted by $\hat{t}$. Also, all of N_2 will be allocated unless all top $m/2$ slots are full. Thus the total utility in our mechanism is

$$\alpha|N_1| + \min\{|N_2|, m/2\} \geq \alpha|N_1| + |N_2|/2 \geq opt/2.$$

If $t^* \geq \hat{t}$, then $|N_1| \geq m/2$, and thus our mechanism allocates all of the bottom $m/2 \geq |N_1|/2$ lower slots to g_1. Also, all of N_2 are allocated. Thus the total utility is at least

$$\alpha|N_1|/2 + |N_2| \geq \frac{\alpha|N_1| + |N_2|}{2} = opt/2. \qquad \square$$

4.2 General structures

The RANKING algorithm by Karp et al. [10] assigns a random priority over slots, and matches every arriving node to its highest-priority neighbor. They prove that the algorithm has an approximation ratio of $1-1/e \cong 0.632$ in expectation, and that no mechanism can do better on general unweighted bipartite graphs. By Corollary 3, it follows that no better mechanism exists for the general parking allocation problem either.

The RANKING algorithm can easily be implemented with posted prices without any additional assumption (except Assumption UV), by assigning random prices to slots and keep these prices fixed.

In contrast, when ϕ_g's significantly differ, no constant approximation can be guaranteed even under Assumption KT.

PROPOSITION 12. *No online algorithm can guarantee an approximation ratio better than $1/\alpha$.*

PROPOSITION 13. *Setting fixed prices at 0 guarantees a $1/2\alpha$ approximation.*

Thus, the approximation ratio on general structures without further assumptions is $\Theta(1/\alpha)$. Another bound we can get is in terms of the number of goals. Consider the RANDOM-PARTITION mechanism, which generates a random partition of space $P = (S_1, \ldots, S_k)$ to the k goals. We know by Observation 4 that any partition including P can be optimally implemented with posted prices under Assumption KG.

PROPOSITION 14. *Under Assumption KG, for any number of goals k, RANDOM-PARTITION is a $\frac{1}{k}$-approximation mechanism. Moreover, it can be derandomized.*

PROOF SKETCH. A random partition allocates every goal roughly $1/k$ of the slots in every possible distance (in expectation). Further, with a deterministic queuing algorithm, we can make sure that *at least* $1/k$ of the slots at distance *at most* d are allocated to goal g - for every goal g and distance d.

Suppose that in the optimal allocation some set $N_i \subseteq N$ of goal g_i's agents are allocated slots. Then such a partition guarantees that at least $1/k$ of the agents in N_i can still be allocated. $\square$

Known population. Our upper bounds thus far relied on the inherent difficulty of the online matching problem. When the population is known, the online matching problem (which is equivalent to parking allocation with Assumption KT) is trivial by Corollary 6, and thus this setting highlights the mechanism design challenge. That is, how does the fact that the allocation is done by a pricing mechanism affects the approximation ratio.

We next show that if agents' goals are unknown, then no pricing mechanism can implement the optimal allocation even if the population is initially known. Further, this holds even if the space is a mild variation of the interval setting from Section 4.1. We still use two goals on a one-dimensional line. However, the goals may not be located on the ends of an interval, and there can be slots on either side of each goal.

PROPOSITION 15. *For the structure of two goals on a line, under Assumption KP, there exists no pricing mechanism that implements the optimal allocation.*

PROOF SKETCH. Consider a structure $\mathcal{S}$ over a line of size 8, $\{s_1, \ldots, s_8\}$, with two goals $g_1 = s_4, g_2 = s_7$, and four vacant slots $\{s_1, s_5, s_6, s_8\}$. All other slots are blocked. We set $\phi_1 = 2, \phi_2 = 1$. The population N has five agents: $(g_1, 1); (g_1, 8); (g_2, 1); (g_2, 1); (g_2, 8)$. Note that in the optimal solution we can place both agents with goal g_1 and two other agents, thus $opt = 2\phi_1 + 2\phi_2 = 6$. Our proof shows that for any *deterministic* mechanism M, $\min_\pi SW(M, \langle\mathcal{S}, N, \pi\rangle) \leq 5 = \frac{5}{6}opt$. Since there is only a finite number of outcomes, it follows that the approximation of any randomized mechanism is also bounded away from 1. $\square$

While no optimal mechanism exists, the knowledge of the population can be exploited to achieve a constant approximation ratio. The mechanism computes an optimal offline allocation. Then it blocks low-value agents from getting slots that should be allocated to high-value goals, by using appropriate pricing.

PROPOSITION 16. *Under Assumption KP, the described mechanism has an approximation ratio of $\frac{1}{2}$.*

4.3 Layered structures

Our next result shows that for structures that are "well-shaped" in a sense, we can actually break the 0.632 bound and get a better approximation ratio than the RANKING algorithm. While the formal definition of layered structures require some lengthy notations, the intuition behind it is quite simple. Suppose that parking slots are clustered in large underground parking lots around the city. Each parking lot has a single pedestrian exit, so all slots in the lot are equivalent in terms of their distance to goals. Then our structure can be split into (say) five parts, where every part contains 20% of each parking lot. We call these parts "layers". All layers in this example are essentially equivalent. The following definitions are required to handle more general structures where layers are only roughly equivalent.

DEFINITION 1. *We write $A \subseteq_r B$ if B contains all elements in A except at most r. Similarly, $A =_r B$ if $A \subseteq_r B$ and $B \subseteq_r A$.*

Given a structure $\mathcal{S}$, let $R_{i,d} \subseteq S$ be the set of all slots which are at distance at most d from g_i.

DEFINITION 2. *Two disjoint sets of slots S', S'' are* r-equivalent *if there is a bijection $f : S' \to S''$, such that for every goal i and every distance d, $R_{i,d} \cap S'$ and $R_{i,d} \cap S''$ can be mapped onto one another. Formally, if $R_{i,d} \cap S'' =_r R_{i,d} \cap f(S')$ and $R_{i,d} \cap S' =_r R_{i,d} \cap f^{-1}(S'')$, where $f(S) \equiv \bigcup_{s \in S} f(s)$.*

DEFINITION 3. *We say that a structure*[7] $\mathcal{S}$ *is* β-layered, *if all but at most* r *slots in* S *can be partitioned to* β *subsets that are* r*-equivalent, for some* $r = O(\beta)$.

Note that in a β-layered structure, f is in fact a partition to equivalence classes of size β. Intuitively, this means that all slots in this class are roughly in the same location (w.r.t the goals). Any structure is β-layered for $\beta = 1$ and also for sufficiently large β (e.g. $\beta = |S|$), but the definition will turn out to be useful for intermediate values that are much smaller than the number of slots.

It is not hard to see that β-layered structure with low β are quite common. A trivial case is when every slot has β duplicates, as with the underground parking lots example above (holds even for $r = 0$). Another common case is when slots are scattered along the edges of a graph. Figure 1 gives an example and Lemma 17 shows it formally.

LEMMA 17. *Let* Q *be a fixed graph with* q *edges, and suppose that all slots and goals of* $\mathcal{S}$ *are placed along the edges of* Q*, s.t.* $|S| \gg q$*. Then* $\mathcal{S}$ *is* β*-layered for any* β*.*

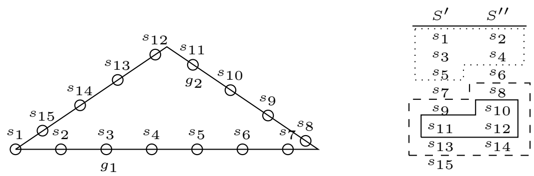

Figure 1: A partition of a structure into two layers S', S''. We outlined the sets $R_{1,2}$ (dotted), $R_{2,1}$ (solid), and $R_{2,3}$ (dashed).

Given a β-layered structure, the LAYERS mechanism is defined as follows.

1. Choose an arbitrary order over layers.
2. Give all slots in layer j price $\frac{j}{\beta+1} \min_g \phi_g$.

The behavior of the agents under the mechanism is straightforward. On arrival, each agent selects a slot in the first layer, if one in her range is available. Otherwise, she is looking for a slot in the second layer, and so on.

THEOREM 18. *Consider* β*-layered instances where* $\beta \gg 1$*, and opt* $\gg \beta^2$*. Under Assumption UV, the worst-case approximation ratio of mechanism LAYERS is* 0.682*.*

Correction: the theorem is wrong (thanks to Yossi Azar, Niv Buchbinder and Moran Feldman for helping is spotting the errors).

PROOF SKETCH. To simplify the proof, we will assume that the layers $S_1, \ldots, S_\beta$ are identical copies. I.e. that they are 0-equivalent rather than $O(\beta)$-equivalent. Thus there are $\gamma \cdot \beta$ slots, where $\gamma = |S_j|$ for all j. That is, there are γ equivalence classes, with β equivalent slots in each. Given some $S' \subseteq S_j$, we denote by $f_{j'}(S') \subseteq S_{j'}$ the locations corresponding to S' in $S_{j'}$. That is, the entries in column j' that are in the same rows as S'.

Denote by N^*_j the set of agents that are assigned by our mechanism to layer j, and by $S^*_j \subseteq S_j$ the slots N^*_j occupy. Denote, $N^* = \bigcup N^*_j; h = |S^*_\beta|$. By the way our mechanism works, it is guaranteed that $S^*_j \subseteq f_{j'}(S^*_j)$ for all $j > j'$. We can therefore enumerate the equivalence classes $1, 2, \ldots, \gamma$, s.t. every S^*_j intersects classes 1 to $|S^*_j|$. Therefore, slots can be organized in a matrix

[7] Formally, this should be a family of structures since the definition is asymptotic in $m = |S|$.

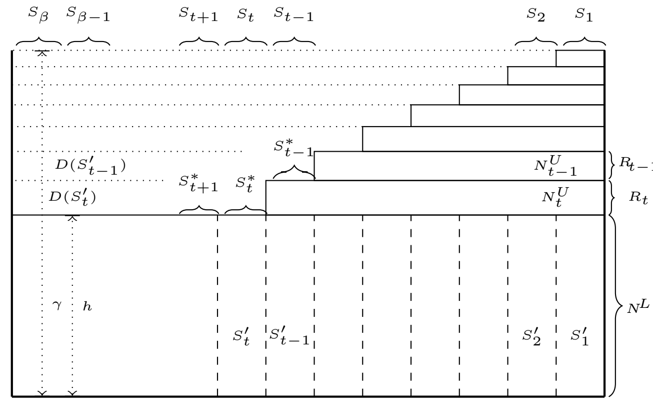

Figure 2: The worst case allocation for LAYERS. The area under the solid line is occupied. In the optimal allocation, all agents located in S'_j are displaced to $D(S'_j)$, and S'_j is occupied by agents of Z_j.

of $\gamma \times \beta$, where all of the first h rows are occupied. Also, by our assumptions, $\gamma \gg \beta$.

Given a suboptimal allocation, we can w.l.o.g. allocate slots to additional agents by replacing some of the current agents to other available slots. Let Z be a maximum-size set of agents that could be assigned by replacing currently assigned agents. Note that $opt = |Z| + |N^*|$.

Suppose we now allocate slots to Z, where in each S_j there is a set of agents $Z_j \subseteq Z$ displacing a currently allocated set $N'_j \subseteq N^*_j$. The agents of N'_j are displaced from slots $S'_j \subseteq S^*_j$ to some other locations $D(S'_j) = D(N'_j)$. Clearly $|Z_j| = |N'_j| = |S'_j| = |D(S'_j)|$. It can be shown that in any outcome:

- For all $j \leq \beta$, $S'_j \subseteq f_j(S^*_\beta)$.
- For all $j' < j$, $D(S'_j) \cap S_{j'} = \emptyset$.
- For all $j < \beta$, $f_j(S^*_{j+1} \uplus D(S'_{j+1})) \subseteq S^*_j$.

The proof first shows that w.l.o.g. the structure of the worst-case allocation is as in Figure 2. Then, we compute the ratio of the instance from the figure, showing that $|N^*| \geq 2.146|Z|$, which entails the stated approximation ratio.

We partition the occupied part S^*_j of each column to a lower part L_j (bottom h rows), and an upper part $U_j = S^*_j \setminus L_j$. W.l.o.g. $S'_j = L_j$ for all $j \leq t$ for some t, and $S'_j = \emptyset$ for all $j \geq t+1$. In particular, $|Z| = \sum_{j=1}^t |Z_j| = \sum_{j=1}^t |S'_j| = ht$.

In order to count N^*, we split it to disjoint sets as follows. The set N^L contains all agents in the first h rows. Clearly $|N^L| = h\beta$.

Consider the set of slots S'_t and the displaced locations $D(S'_t)$. Let R_t be the set of rows that intersect $D(S'_t)$. By the points above, all slots in the block $R_t \times \{1, \ldots, t-1\}$ are occupied. Therefore $|N^*|$ is minimized when $|R_t|$ is minimized. Thus w.l.o.g. $D(S'_t)$ is the rectangle $R_t \times \{t, \ldots, \beta\}$. Now, since $|D(S'_t)| = |S'_t| = h$, we have that $|R_t| = \frac{h}{\beta - t}$, which entails $|N^U_t| \geq (t-1)|R_t| = \frac{t-1}{\beta-t}h$.

We can similarly define R_j, N^U_j for every $j \leq t$. In the worst case, all of R_j are minimal and disjoint, $D(S'_j)$ are rectangles, and thus N^U_j are also disjoint. By a simple calculation as above, $|N^U| \geq \sum_{j=1}^t |N^U_j| \geq \sum_{j=1}^t \frac{j-1}{\beta-j}h \geq \sum_{j=1}^{t-1} \frac{j}{\beta-j}h$.

We can now write the $|N^*|/|Z|$ ratio as

$$\frac{|N^*|}{|Z|} = \frac{|N^L|+|N^U|}{|Z|} \geq \frac{1}{ht}(h\beta + h\sum_{j=1}^{t-1}\frac{j}{\beta-j})$$
$$\cong \frac{1}{x}(1-\ln(1-x)) - 1 \geq 2.146 \qquad \text{(for } x = \tfrac{t}{\beta})$$

Finally, $opt = |N^*|+|Z| \leq (1+\frac{1}{2.146})|N^*| \cong 1.466|N^*|$, which means a 0.682-approximation. □

As the bound in Theorem 18 is asymptotic in β, one may wonder whether a large number of layers is required for a good approximation ratio. For $\beta = 3$ we will have $t = 2$, and one "step" N_1^U of height $h/2$. Thus the approximation ratio (when $opt \to \infty$) will be $7/11 \cong 0.636$. That is, already better than $1 - 1/e$. The approximation then gradually improves as β increases, but not necessarily monotonically due to rounding.

5. RESULTS FOR LINEARCOST

We begin by characterizing the optimal offline allocation for a single goal under the LINEARCOST scheme. Suppose that in the optimal allocation there are m' occupied slots. Then it is clear that (a) these are the m' slots closest to the goal; and (b) each of the m' agents with lowest cost c_j gets a slot. Assume slots are sorted by non-decreasing distance from g, and agent j gets slot $\sigma(j)$. Then the social welfare of this allocation is

$$\sum_{i\leq m'}(\phi - d(g,s_i)c_{\sigma^{-1}(i)}) = m'\cdot\phi - \sum_{i\leq m'} d(g,s_i)c_{\sigma^{-1}(i)}.$$

Sort agents by cost c_j in non-decreasing order. In order to minimize $\sum_{i\leq m'} d(g,s_i)c_{\sigma^{-1}(i)}$ (and thus maximize welfare), we need to assign $s_{m'}$ (farthest occupied slot) to agent 1 who has the lowest cost c_1, assign $s_{m'-1}$ to agent 2 and so on. Thus to find the optimal allocation we can try all $m' \leq \min\{m,n\}$, and for each m' apply the optimal allocation of m' agents described above.

5.1 Parking as a position auction

We will leverage results for the standard generalized second price (GSP) auction [18] to set posted prices for our parking allocation problem under the LINEARCOST scheme and with Assumptions KP+KG.

In a GSP auction, there are a set of slots (e.g. advertising slots) with quality $(x_s)_{s\in S}$, and a set of agents (e.g. advertisers) with valuation $(V_i)_{i\in N}$. The utility that agent i extracts from slot s at price p_s is $U(i,s) = V_i x_s - p_s$. Agents each submit a bid $(b_i)_{i\in N}$. The GSP auction allocates the slot of the highest quality to the agent with the highest bid and so on. It then charges agent assigned to slot s a price $p_s = x_s b_{\sigma^{-1}(s+1)}$. Hence, $U(i,s) = x_s(V_i - b_{\sigma^{-1}(s+1)})$.

Varian [18] characterized the Symmetric Nash Equilibria (SNE) of GSP auctions and provided closed-form expressions of agent's bid b_i at an SNE in terms of $(x_s)_{s\in S}$ and $(V_i)_{i\in N}$. He showed that these SNEs are envy-free, that is, for any two agents i and i' it holds that $U(i,\sigma(i)) \geq U(i,\sigma(i'))$. These results suggest that if we can calculate p_s (without engaging the bidding process) and use them as posted prices for the slots, we can achieve the same allocation as the GSP auction at an SNE. Varian's expressions of b_i make it possible to remove the actual bidding process. Given $(x_s)_{s\in S}$ and $(V_i)_{i\in N}$, we can "simulate" bids at an SNE and then calculate p_s.

We now map a single-goal parking allocation instance to a GSP auction. Let $D = \max_s d(g,s)$, and set the quality as $x_s = D - d(g,s)$. To determine the valuation of each agent, we set $V_i = c_i$.

Suppose that every agent $i \in N$ submits a bid b_i, and is allocated slot $s = \sigma(i)$ at price p_s (GSP prices). The utility of i is

$$u_i(s) = \phi - d(g,s)c_i - p_s = \phi - (D - x_s)c_i - p_s$$
$$= \phi - Dc_i + x_s V_i - x_s b_{\sigma^{-1}(s+1)} = \mu_i + U(i,s).$$

That is, the utility of i in the parking allocation is exactly the utility of i in the induced ad-auction allocation, plus a constant $\mu_i = \phi - Dc_i$ that does not depend on the allocation.

For slots S and agents N, let $\mathbf{p} = \mathbf{p}(S,N)$ be a vector of SNE prices (there are usually more than one). As $u_i(s)$ is an affine transformation of $U(i,s)$, $\mathbf{p}$ induces an envy-free parking allocation.

LEMMA 19. *Suppose we assign m' slots to m' agents using SNE prices $\mathbf{p}$. Agent utility is non-decreasing in the distance from g. That is, if $d(g,\sigma(i)) > d(g,\sigma(j))$, then $u_i(\sigma(i)) \geq u_j(\sigma(j))$.*

We now define mechanism GSP-PARK for a single goal under Assumption KP.

1. Given the population N and structure $\mathcal{S}$, sort agents by *increasing* cost, and slots by increasing distance from g.
2. Compute an optimal offline allocation σ, and extract m^*, which is the optimal number of agents allocated a slot. Note that $\sigma(m^*) = 1, \sigma(m^*-1) = 2$, etc.
3. Simulate some SNE bids $b_1,\ldots,b_{m^*}$ for these agents (in the induced GSP auction) given $V_i = c_i$ and $x_s = D - d(g,s)$.
4. If $m^* \geq n$, set the price of slot s_i as $p_i = x_i b_{i+1}$. Otherwise, define $M = u_{m^*}(s_1,p_1) - \epsilon$ (for some low ϵ), and set prices as $p'_i = p_i + M$.

PROPOSITION 20. *Under Assumption KP, GSP-PARK is optimal for a single goal.*

PROOF. Due to envy-freeness, we know that each agent $j \leq m^*$ prefers the slot allocated to her over any other slot at these prices.[8]

The translation M is required to prevent high-cost agents from selecting a slot on arrival. Indeed, for every j s.t. $c_j > c_{m^*}$,

$$u_j(s_1,p'_1) = u_j(s_1,p_1) - M = u_j(s_1,p_1) - u_{m^*}(s_1,p_1) + \epsilon < 0.$$

Thus no such agent will be interested in the first slot. Moreover, since agent m^* prefers the first slot to any other, this must apply for any agent whose cost c_j is higher. Thus at prices $\mathbf{p}'$, any agent with $c_j > c_{m^*}$ will avoid all slots. A slight complication is when $c_j = c_{m^*}$ for some $j > m^*$. It can be similarly shown that these agents will forgo any slot assigned to agents of lower-cost types.

It remains to prove that the new mechanism is individually rational, i.e. all agents $j \leq m^*$ want their slot at the modified price p'. Indeed, since $c_{m^*+1} > c_{m^*}$ (and the distance of every slot is non-zero), $u_{m^*}(s_1,p'_1) = u_{m^*}(s_1,p_1) - M = \epsilon > 0$. By Lemma 19, $u_j(s_{\sigma(j)}, p_{\sigma(j)}) \geq u_{m^*}(s_1,p'_1) > 0$ for all $j \leq m^*$. □

An immediate corollary from Observations 1 and 4, is that under Assumptions KP+KG, there is an optimal pricing mechanism for LINEARCOST for any structure and number of goals.

6. DISCUSSION

In this work we established a firm link between online bipartite matching mechanisms and practical parking allocation problems. We then provided pricing mechanisms that can exploit the rising popularity of advanced city-wide parking systems in order to increase the social welfare of the population.

[8]Indifference between slots might pose a problem in principle. However, it can be shown that this is solved if a particular SNE solution is chosen.

Structure \ Assumptions	Uniform values for goals (UV)		Different values for goals		
	KP, ¬KG	¬KP	KP + KG	KP,¬KG	¬KP
Interval (2 goals)	opt	UB: 0.75 (P.10) LB: 0.682 (T.18)	opt	opt (P.9)	0.5 (P.11,P.10)
Layered	LB: 0.682		opt	UB: < 1 (P.15)	UB: $O(1/\alpha)$ (P.12)
Any	LB: 0.632	0.632 (Karp et al. [10])	opt (P.6)	LB: 0.5 (P.16)	LB: $\Omega(1/\alpha)$ (P.13)*

Table 1: Summary of results for MAXDISTANCE. KP = Known Population, KG = Known Goal, $\alpha = \frac{\max_g \phi_g}{\min_{g'} \phi_{g'}}$. The results in the ¬KP columns hold regardless of whether there is information on the full type, the goal only, or none at all. Proposition numbers appear in brackets. Entries with no reference follow from entries to their right or bottom. * LB: $\max\{\Omega(1/\alpha), 1/k\}$ under KG (P. 14).

Our mechanisms also advance the state-of-the-art in "standard" online bipartite matching by taking advantage of structural restrictions of the matching graph. For unweighted graphs with layered structure we improve the $1 - \frac{1}{e} \cong 0.632$ bound of Karp et al. [10] (which is tight in the general case) to 0.682. Moreover, in contrast to Karp's RANKING algorithm, our LAYERS algorithm is deterministic. We conjecture that the bound could be further improved by determining a random priority (equivalently, random prices) for each layer independently, similar to what is used by Karp et al. [10].

Our result for the LINEARCOST scheme reveals interesting connection between parking allocation and ad auctions. While in this paper we showed how known techniques from GSP auctions can be applied to parking allocation, the other direction is interesting too: The multi-goal version of our problem can be interpreted as a generalization of the ad auction setting. That is, where the value of a slot to different advertisers may depend on different spatial attributes. As a concrete example, think of ads that are displayed from right to left. While advertisers in English value ads by their proximity to the left end of the screen, advertisers in languages that are written from right to left (like Hebrew and Arabic) value ads by their closeness to the right end. Interestingly, this motivating example exactly coincides with our interval structure from Section 4.1.

From a mechanism design perspective, the welfare criterion should be weighted against other properties of the assignment mechanism such as stability (as in the recent and lucid survey of Eric Budish [2]). The choice of using posted prices mechanisms eliminates the need to deal with agents' incentives, and allow the designer to focus on welfare optimization. In other words, the inherent constraints of today's parking systems settle the tradeoff between stability and welfare. As future applications for parking allocation will collect more information from the agents themselves, the tradeoff between the various properties of the allocation will become more important, and mechanisms will have to deal with it explicitly.

Future directions. While the assumption that the goals of agents are known is often realistic, the population itself is unlikely to be completely known in advance. Future work will focus on weakening Assumption KP by considering only partial knowledge about the population (e.g. its distribution).

Our current model characterizes the situation where parking slots are allocated for the entire day (or month/year), which is suitable for commuter parking but not, for example, for parking at shopping malls and entertainment places. A possible extension to accommodate these scenarios is to allow agents arrive and leave dynamically, and thus their preferences include the amount of time they plan to use a slot. The latter may in turn depends on prices of slots. Hence, such a model presents a much more complicated challenge.

Acknowledgements

This research was supported in part by the Google Inter-university center for Electronic Markets and Auctions.

7. REFERENCES

[1] G. Aggarwal, G. Goel, C. Karande, and A. Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. SODA '11, pages 1253–1264, 2011.

[2] E. Budish. Matching "versus" mechanism design. *SIGecom Excehanges*, 11(2):4–15, 2012.

[3] S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. STOC '10, pages 311–320, 2010.

[4] J. Feldman, A. Mehta, V. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. FOCS '09, pages 117 –126, oct. 2009.

[5] E. Friedman and D. Parkes. Pricing wifi at starbucks: issues in online mechanism design. In *Proc. of 4th ACM-EC*, pages 240–241. ACM, 2003.

[6] Y. Geng and C. G. Cassandras. A new "smart parking" system based on optimal resource allocation and reservations. ITSC '11, pages 979–984, 2011.

[7] E. Gerding, V. Robu, S. Stein, D. Parkes, A. Rogers, and N. Jennings. Online mechanism design for electric vehicle charging. In *Proc. of 10th AAMAS*, 2011.

[8] B. Kalyanasundaram and K. Pruhs. On-line weighted matching. SODA '91, pages 234–240, 1991.

[9] C. Karande, A. Mehta, and P. Tripathi. Online bipartite matching with unknown distributions. STOC '11, pages 587–596, 2011.

[10] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. STOC '90, pages 352–358, 1990.

[11] E. Koutsoupias and A. Nanavati. The online matching problem on a line. In R. Solis-Oba and K. Jansen, editors, *Approximation and Online Algorithms*, volume 2909 of *LNCS*, pages 327–328. 2004.

[12] P. R. Lewis, P. Marrow, and X. Yao. Resource allocation in decentralised computational systems: an evolutionary market-based approach. *Autonomous Agents and Multi-agent Systems*, 21:143–171, 2010.

[13] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized online matching. *J. ACM*, 54(5), Oct. 2007.

[14] E. S. Povich. Mobile apps revolutionize municipal parking. AolGovernment (Innovation), `http://tinyurl.com/dylyp3p`, 2012.

[15] M. Richtel. Now, to find a parking spot, drivers look on their

phones. The New York Times (Technology), http://tinyurl.com/d9nvcew, 2011.
[16] M. Roozbehani, M. A. Dahleh, and S. K. Mitter. Volatility of power grids under real-time pricing. *CoRR*, abs/1106.1401, 2011.
[17] S. Stein, E. Gerding, V. Robu, and N. Jennings. A model-based online mechanism with pre-commitment and its application to electric vehicle charging. In *Proc. of 11th AAMAS*, 2012.
[18] H. R. Varian. Position auctions. *Int. Journal of Industrial Organization*, 25(6):1163–1178, 2007.
[19] M. B. Yildirim. Congestion toll pricing models and methods for variable demand networks. Ph.D. thesis, University of Florida, 2001.

APPENDIX

A. PROOFS

LEMMA 2. *Let (I, S, E) be a bipartite graph. Then there is an instance of MaxDistance where $d(s, g_i) \leq m_i$ if and only if $(i, s) \in E$.*

PROOF. For every $i \in I$, denote by $\Gamma(i) \subseteq S$ the neighbors of i. Let $m = |S|$, and consider the $(m-1)$-dimensional regular simplex (with unit side length) over vertices S. Let d_k be the radius of a k-dimensional face in this simplex, then $d_k = \sqrt{\frac{k}{2(k+1)}}$. Note that $0 = d_0 < d_1 < \cdots < d_{m-1}$. Let $x_s \in \mathbb{R}^{m-1}$ denote the coordinates of slot s. For every agent $i \in I$ we define a single goal g_i, and place it in the middle of the face defined by $\Gamma(i)$. That is $x_{g_i} = \frac{1}{|\Gamma(i)|}\sum_{s\in\Gamma(i)} x_s$. Finally, we define the type of agent i as $(g_i, d_{|\Gamma(i)|})$. Thus $d(i, s) \leq m_i$ iff $s \in \Gamma(i)$. □

We present a mechanism called TWO-THRESHOLDS (omitted from the main text), which guaranteed as $\frac{2}{3}$-approximation. This is worse than the LAYERS mechanism, but demonstrates the power of simple threshold mechanisms on the interval. Denote by T_i the $m/3$ slots closest to g_i. Whenever an agent with goal i arrives, the mechanism sets infinite price on all slots in T_{-i}, and applies monotone pricing for the remaining slots. This mechanism can be shown to guarantee an approximation ratio of $\frac{2}{3}$ on the interval, under Assumptions KG+UV.

PROPOSITION 21. *Under Assumptions KG+UV, the TWO-THRESHOLDS mechanism guarantees an approximation ratio of $\frac{2}{3}$ on the interval.*

PROOF. At the end of the sequence, if T_i is not full - then all agents of type i were allocated. If both T_1, T_2 are full or both are non-full, then the allocation must be optimal (either all slots are taken, or all agents are assigned). Thus suppose that exactly one of T_i (w.l.o.g. T_1) is full. Denote by T' the middle segment. If T' is full, then at least $\frac{2}{3}m$ slots are allocated and we are done. Let x_1, x_2 be the number of type 1 and type 2 agents in T', and let y be the number of un-allocated type 1 agents (that could be allocated in *opt*). There are no unallocated agents of type 1 with range above $\frac{2}{3}m$, as they would go the gaps in T'. However, every type 2 agent in T' prevents the allocation of at most one type 1 agent. Thus $y \leq x_2$.

Our mechanism allocated all of T_1, plus x_1+x_2 slots from T', plus z_1+z_2 slots from T_2 (to agents of types 1,2). That is $m/3+x_1+x_2+z_1+z_2$ in total. The optimal allocation would assign at most x_2+z_2 slots to type 2 agents (as all were allocated), plus $m/3+x_1+y+z_1$ slots to type 1 agents. Thus the approximation ratio is at least

$$\frac{m/3+x_1+x_2+z_1+z_2}{m/3+x_1+y+z_1+x_2+z_2} \geq \frac{m/3+x_2}{m/3+x_2+y} \geq \frac{m/3+x_2}{m/3+2x_2} \geq \frac{m/3+m/3}{m/3+2m/3} = \frac{2}{3}.$$

□

PROPOSITION 12. *No online algorithm can guarantee an approximation ratio better than $1/\alpha$.*

PROOF. We start with a single goal g' with a low value ϕ', and for every slot $s \in S$ add another goal g_s with a high value $\phi_s = \alpha\phi'$. For every subset of slots $T \subseteq S$ (including $T = \emptyset$), we define an instance H_T as follows. First, m agents of type (g', ∞) arrive (i.e. they can be placed anywhere). Then for every $s \in T$, arrives an agent of type $(g_s, 0)$, where $g_s = s$.

Suppose that after the first sequence of agents, some slots $T^* \neq \emptyset$ are allocated. Then on instance H_{T^*}, we have $SW = |T^*|\phi'$ and $opt = |T^*|\alpha\phi' + (m-|T^*|)\phi' \geq \alpha SW$. On the other hand, if $T^* = \emptyset$, then on instance $H_\emptyset$ we have $SW = 0 < opt/\alpha$. □

PROPOSITION 13. *Setting fixed prices at 0 guarantees a $1/2\alpha$ approximation.*

PROOF. By setting all prices to 0, we get a *maximal* cardinality matching of size t, whose welfare is at least $t \cdot \min_j \phi_j$. By Observation 5, the *maximum* cardinality matching is of size at most $2t$. Thus the approximation is at least $\frac{t\cdot\min_j \phi_j}{2t\cdot\max_i \phi_i} = \frac{1}{2\alpha}$. □

PROPOSITION 14. *Under Assumption KG, for any number of goals k, RANDOM-PARTITION is a $\frac{1}{k}$-approximation mechanism. Moreover, it can be derandomized.*

PROOF. Let $P = (N_1, \ldots, N_k)$ be the allocated agents in an optimal partition, where $N_{id} \subseteq N_i$ is the set of agents s.t. $m_j \leq d$. Denote by $S'_i \subseteq S_i$ the set of slots allocated by the mechanism to agents following goal g_i. We argue that in expectation at least $|S'_i| \geq |N_i|/k$.

Let $R_{i,d}$ be the set of slots of distance at most d from g_i. Consider some goal g_i and let m' be the smallest distance s.t. $(S_i \setminus S'_i) \cap R_{i,m'} \neq \emptyset$. That is, the smallest distance of an available slot in our allocation. All agents of type (g_i, m_j) with $m_j \geq m'$ are allocated, as otherwise we would assign j to an available slot. For every $d < m'$ (and in particular $d = m'-1$), we have $S_i \cap R_{i,d} = S'_i \cap R_{i,d}$. By our random assignment, $|S'_i \cap R_{i,d}| = |S_i \cap R_{i,d}| = |R_{i,d}|/k$ in expectation. Clearly $|N_{i,d}| \leq |R_{i,d}|$. Finally,

$$|S'_i| = |S'_i \cap R_{i,d}| + |S'_i \setminus R_{i,d}| \geq |N_{i,d}|/k + |N_i \setminus N_{i,d}| \geq |N_i|/k.$$

The total welfare at out allocation is thus

$$\sum_{i\leq k} \phi_i |S'_i| \geq \sum_{i\leq k} \phi_i |N_i|/k = \frac{1}{k}\sum_{i\leq k}\phi_i|N_i| = \frac{opt}{k}.$$

If we want to de-randomize the allocation of S_i and thus the mechanism, this is not too hard. We sort the slots by their distance from each of the k goals. Thus we have k priority queues, each containing all of the slots. We iteratively traverse all slots in a round-robin over goals, when in each iteration t we take the next slot in queue $i = t \mod k + 1$ and assign in to S_i. We then remove the assigned slot from all queues. Using the same notations as in the randomized algorithm, it now holds that $|S'_i \cap R_{i,d}| = |S_i \cap R_{i,d}| \geq |R_{i,d}|/k - k$, since all slots of distance at most d from g_i have been divided either equally (up to a rounding factor of k), or in favor of goal i. Thus it still holds that

$$|S'_i| = |S'_i \cap R_{i,d}| + |S'_i \setminus R_{i,d}| \geq |N_{i,d}|/k - k + |N_i \setminus N_{i,d}| \geq |N_i|/k - k,$$

and thus

$$\sum_{i \le k} \phi_i |S'_i| \ge \sum_{i \le k} \phi_i (|N_i|/k - k) = \frac{1}{k} \sum_{i \le k} \phi_i |N_i| - k^2 = \frac{opt}{k} - k^2.$$

As the number of agents increases as k remains constant, the additive factor is negligible. □

PROPOSITION 15. *For the case of two goals on a line, under Assumption KP, there exists no optimal SP mechanism.*

PROOF. We set $\phi_1 = 2, \phi_2 = 1$. Our proof shows that an adaptive adversary can force an approximation ratio of at least $\frac{5}{6}$. It follows that the approximation of any *deterministic* mechanism against a static aversary is also at least $\frac{5}{6}$. While some randomized mechanisms might do better, clearly none can guarantee a perfect performance (e.g. against an adversary that plays a random sequence).

Consider a line with 8 slots, $\{s_1, \dots, s_8\}$, where $g_1 = s_4, g_2 = s_7$. The vacant slots are $a = s_1, b = s_5, c = s_6$, and $d = s_8$. All other slots are blocked. The population has five agents: $(g_1, 1); (g_1, 8); (g_2, 1); (g_2, 1); (g_2, 8)$. Note that in the optimal solution we can place both type 1 agents and two other agents, thus $opt = 2\phi_1 + 2\phi_2 = 6$.

We can model the arrival process as a zero-sum game in extensive form between the mechanism (which is setting prices), and an adversary (which is setting the order of arrival). The goal of the mechanism is to maximize the welfare. We will show that the adversary can coerce a situation such that either not all slots are full, or only one high value agent is placed.

While the adversary has full information, the mechanism only observes its own actions and the currently occupied slots. That is, if at a given pricing there are several agents that would pick s_1, then all states where one of these agents arrived and occupied s_1 belong to the same information set of the mechanism player. Crucially, the mechanism cannot condition the pricing in the next step based on the identity of the agent occupying s_1.

Therefore, for every sequence of prices, the adversary can choose (part of) the order in retrospect, as long as selections are consistent with the information sets. We denote by p_x^i the price of slot $x \in \{a, b, c, d\}$ in step i. Note that agent $(g_1, 1)$ can only be placed in b. Thus if b is being occupied by any other agent, the adversary wins. Thus we can assume that $p_a < p_b$ as long as there is at least one active agent with range m (and in particular in the first step).

Consider the first step. Case A: p_a^1 is strictly lower than all other prices. Then the adversary sends either $(g_1, 8)$ or $(g_2, 8)$, which occupy slot a. Note that both states are in the same information set, thus the next action of the mechanism must be the same for both.

Case A-A: Suppose p_b^2 is the cheapest price. Then the adversary places $(g_1, 8)$ in a (i.e. chooses from the information set of case A), and sends $(g_2, 8)$, which occupies slot b (and thus wins).

Case A-B: Either $p_c^2 \ge p_b^2$, or $p_d^2 \ge p_b^2$. Then the adversary sends a $(g_2, 1)$ agent, which occupies one of $\{c, d\}$. Note that the adversary does not select an option from the information set yet.

In the 3rd step, then once again if b is cheapest, we are in case A-A. Otherwise, b remains the only empty slot.

Case A-B-A: $p_b^4 \le \phi_2$. As in case A-A, the adversary sets $(g_1, 8)$ in a, and $(g_2, 8)$ in b.

Case A-B-B: $p_b^4 > \phi_2$. The adversary selects $(g_2, 8)$ from the information set, and sends $(g_1, 8)$, which occupies b. In all cases b is occupied by an agent other than $(g_1, 1)$.

Case B: $p_x^1 \le p_a^1$, for $x \in \{c, d\}$. Then the adversary sends either $(g_1, 8)$ or $(g_2, 1)$, which occupy slot x. As in Case A, both states belong to the same information set. Denote by y the slot from $\{c, d\}$ that is not x.

Case B-A: $p_a^2 < p_y^2$. The adversary selects $(g_2, 1)$ from the information set (which occupies x), and continue as if we are in Case A (only with one less available slot). We saw that in Case A the adversary wins.

Case B-B: $p_a^2 \ge p_y^2$. The adversary selects $(g_1, 8)$ from the information set (which occupies x), and sends $(g_2, 8)$, which will occupy y. However, now there is no agent that can occupy slot a, and thus the adversary wins again.

We saw that the adversary can always enforce a suboptimal outcome. We next compute the attained approximation. In Case B-B, the welfare is at most $2\phi_1 + \phi_2 = 5$. In all other cases, the welfare is at most $\phi_1 + 3\phi_2 = 5$ as well. Thus any pricing mechanism has an approximation ratio of at most $\frac{5}{6}$. □

PROPOSITION 16. *Under Assumption KP, there is a $\frac{1}{2}$-approximation mechanism.*

PROOF. We enumerate the distinct values of goals s.t. $\phi_1 > \phi_2 > \dots > \phi_{k'}$. It is possible that $k' < |G|$ if some goals have the same value. We compute an optimal partition offline. Denote by S_i the sets of all slots allocated to goals with value ϕ_i. Then we apply arbitrary prices on any S_i separately, where all prices in S_i are between ϕ_{i+1} and ϕ_i (thus no agent wants a slot of a higher value goal).

Let N_j be all agents whose value is equal to ϕ_j (i.e. we merge groups with the same value). Consider the set of allocated agents $A \subseteq N$, and denote by $T \subseteq N$ the agents participating in the original optimal (offline) allocation. Suppose we want to allocates slots to all of T. For every $j \in T \setminus A$, j's original slot is occupied by an agent with same or higher value. Denote $A_i = \bigcup_{j \le i} (A \cap N_j)$, and $O_i = \bigcup_{j \le i} (T \cap N_j)$.

Thus $\sum_{j \le i} |A \cap N_j| \ge \sum_{j \le i} |(T \setminus A) \cap N_j|$ for all i, which means

$$|A_i| = \sum_{j \le i} |A \cap N_j| \ge \frac{1}{2} \sum_{j \le i} |T \cap N_j| = \frac{1}{2} |O_i.|$$

Finally, let $a_j = |A \cap N_j|$. Note that $\phi_j - \phi_{j+1} > 0$.

$$
\begin{aligned}
SW(A) &= \sum_{i \le k'} a_i \phi_i = \sum_{i \le k'} (|A_i| - |A_{i-1}|)\phi_i \\
&= \sum_{i \le k'} |A_i|\phi_i - \sum_{i \le k'} |A_{i-1}|\phi_i = \sum_{i < k'} |A_i|(\phi_i - \phi_{i+1}) + |A_{k'}| \\
&\ge \sum_{i < k'} \frac{1}{2}|O_i|(\phi_i - \phi_{i+1}) + \frac{1}{2}|O_{k'}| \\
&\frac{1}{2}\sum_{i \le k'} (|O_i| - |O_{i-1}|)\phi_i = \frac{1}{2} opt.
\end{aligned}
$$

To see that the analysis of this mechanism is tight, consider an instance with two goals on the edges of an internal, $\phi_1 = \phi_2 + \epsilon$. There are $m/2$ agents of type (g_1, m) (arriving first), and $m/2$ of type $(g_2, m/2)$. While clearly $opt = m(1 + \frac{\epsilon}{2})$, in this mechanism the first $m/2$ agents will occupy all cheap slots of goal 2, and thus $SW(A) = \frac{m}{2}(1 + \epsilon)$. $\square$

LEMMA 17. *Let Q be a fixed graph with q edges, and suppose that all slots and goals of $\mathcal{S}$ are placed along the edges of Q, s.t. $|S| \gg q$. Then $\mathcal{S}$ is β-layered for any β.*

PROOF. We map as equivalent sets of β nearby slots each time. The difference between $R_{i,d} \cap S''$ and $R_{i,d} \cap f(S')$ can only be in slots whose equivalence classes fall along the cut in Q defined by $R_{i,d}$. Since any cut is of size at most q, there can be at most $q \cdot \beta = O(\beta)$ such slots. In addition, even if we create equivalence classes independently on every edge, at most β slots on every edge ($O(\beta)$ in total) escape the mapping. $\square$

THEOREM 18. *Consider β-layered instances where $\beta \gg 1$, and $opt \gg \beta^2$. Under Assumption UV, the worst-case approximation ratio of mechanism LAYERS is* 0.682. Note that $0.682 > \frac{2}{3} > 1 - \frac{1}{e}$.

PROOF. To simplify the proof, we will assume that the layers $S_1, \dots, S_\beta$ are identical copies. I.e. that they are 0-equivalent rather than $O(\beta)$-equivalent.[9] Thus there are $\gamma \cdot \beta$ slots, where $\gamma = |S_j|$ for all j. That is, there are γ equivalence classes, with β slots in each.

Denote by N^*_j the set of agents that are assigned by our mechanism to layer j, and by $S^*_j \subseteq S_j$ the slots N^*_j occupy. Denote, $N^* = \bigcup N^*_j; h = |S^*_\beta|$. By the way our mechanism works, it is guaranteed that $S^*_j \subseteq f_{j'}(S^*_j)$ for all $j > j'$. We can therefore enumerate the equivalence classes $1, 2, \dots, \gamma$, s.t. every S^*_j intersects classes 1 to $|S^*_j|$. Therefore, slots can be organized in a matrix of $\gamma \times \beta$, where all of the first h rows are occupied. Also, by our assumptions, $\gamma \gg \beta$.

Given a suboptimal allocation, we can w.l.o.g. allocate slots to additional agents by replacing some of the current agents to other available slots (if longer chains are required, we can replace the instance with another instance where every agent is displaced at most once). Let Z be a maximum-size set of agents that could be assigned by replacing currently assigned agents. Note that $opt = |Z| + |N^*|$.

Given some $S' \subseteq S_j$, we denote by $f_{j'}(S') \subseteq S_{j'}$ the locations corresponding to S'. That is, the entries in column j' that are in the same rows as S'. Suppose we now allocate slots to Z, where in each S_j there is a set of agents $Z_j \subseteq Z$ displacing a currently allocated set $N'_j \subseteq N^*_j$. The agents of N'_j are displaced from slots $S'_j \subseteq S^*_j$ to some other locations $D(S'_j) = D(N'_j)$. Clearly $|Z_j| = |N'_j| = |S'_j| = |D(S'_j)|$.

We argue that $|N^*| \ge 2.146|Z|$, which will entail the stated approximation ratio. Our proof proceeds in four steps. First, we highlight some characteristics of the allocation. Second, we show that w.l.o.g. the slots S'_j are scattered as in Figure 2 by altering the instance. Third, we show that w.l.o.g. the slots $D(S'_j)$ are scattered as in Figure 2 so as to minimize $|N^*|$. Fourth, we compute the ratio of the "worst case" instance from the figure.

We can see that

1. $\forall j \le \beta,\ S'_j \subseteq f_j(S^*_\beta)$, otherwise our mechanism would have assigned agents from Z_j to valid slots in S_β.
2. An agent a displaced from $s \in S'_j$ can only be reallocated to a slot s' in the same layer or higher. Otherwise a would have preferred s' over s in the first place. Thus $D(S'_j) \cap S_{j'} = \emptyset$ for all $j' < j$.
3. $\forall j \le \beta,\ f_j(S^*_{j+1} \uplus D(S'_{j+1})) \subseteq S^*_j$. Otherwise an agent placed in S_{j+1} would have selected a valid location in S_j.

We next turn to show that all slots S'_j are w.l.o.g. organized in the bottom right rectangle as in Figure 2. To that end, we will define a sequence of matching instances and legal LAYERS allocations. Every allocation will be slightly different than the previous one, but they will all preserve $|Z|$ and $|N^*|$, and thus will have the same approximation ratio.

Formally, S^*_j can be partitioned to a lower part $L_j = f_j(S^*_\beta)$, and an upper part $U_j = S^*_j \setminus L_j$. In other words, L_j is the first h rows of column j. Suppose there are slots $s \in S'_j, s' \in L_i \setminus S'_i$, for $j > i$. Denote by $a \in N'_j, a' \in N_i$ the agents occupying slots s and s', respectively. Let $s'' = d(s) \in D(S'_j)$ be the new slot of a in the optimal allocation, after being displaced from s by some other agent $y \in Z_j$. Note that a' is not displaced.

We now define a new instance, where instead of agents a, a', y we have agents b, b', z. Agent b can only go to s, agent b' can only go to s' or s'', and agent z can go to s'. In the corresponding allocation, b is in s, and b' is in s'. Note that z can displace b' (which will be assigned to s'') in the optimal allocation of the new instance, thus both $|Z|$ and $|N^*|$ remain the same. Also, since $j > i$, the properties above and in particular (2) are kept.

[9] When layers are $O(\beta)$-equivalent, this may add a factor of $O(\beta^2)$ to the sizes of the sets we compute. Since $opt \gg \beta^2$, this becomes negligible in the limit.

By repeating the process, we can concentrate all of the "displacement slots" S'_j in the lower parts of the first layers. That is, w.l.o.g. $S'_j = L_j$ for all $j < t$ for some t, and $S'_j = \emptyset$ for all $j \geq t+1$.

$$|Z| = \sum_{j=1}^{t} |Z_j| = \sum_{j=1}^{t} |S'_j| = \sum_{j=1}^{t-1} h + |S'_t| \in [h(t-1), ht].$$

Since we only care about a lower bound for the ratio $|N^*|/|Z|$, we assume $S'_t = L_t$, which means $|Z| = ht$. This completes the second part of the proof. In order to count N^*, we split it to disjoint sets as follows. The first set N^L contains all agents in the first h rows. Clearly $|N^L| = h \cdot \beta$.

Consider the set of slots S'_t and the displaced locations $D(S'_t)$. Let R_t be the set of rows that intersect $D(S'_t)$. By (2)+(3), all slots in the block $R_t \times \{1, \dots, t-1\}$ are occupied. Therefore $|N^*|$ is minimized when $|R_t|$ is minimized. Thus w.l.o.g. $D(S'_t)$ is the rectangle $R_t \times \{t, \dots, \beta\}$. Now, since $|D(S'_t)| = |S'_t| = h$, we have that $|R_t| = \frac{h}{\beta - t}$. Then we have that $|N^U_t|$, which is the number of agents in rows R_t, is at least $(t-1)|R_t| = \frac{t-1}{\beta-t}h$.

We can similarly define R_j, N^U_j for every $j \leq t$. In the worst case, all of R_j are minimal and disjoint, $D(S'_j)$ are rectangles, and thus N^U_j are also disjoint. This is since such a partition would minimize the total number of rows $|\bigcup_{j \leq t} R_j|$, and minimizes the hight of the high index rectangles (which are wider) first. By a simple calculation as above, $|N^U| \geq \sum_{j=1}^{t} |N^U_j| \geq \sum_{j=1}^{t} \frac{j-1}{\beta-j}h \geq \sum_{j=1}^{t-1} \frac{j}{\beta-j}h$. This completes the third part of the proof.

We can now write the $|N^*|/|Z|$ ratio as

$$\begin{aligned}
\frac{|N^*|}{|Z|} &= \frac{|N^L| + |N^U|}{|Z|} \geq \frac{1}{ht}(h\beta + h\sum_{j=1}^{t-1} \frac{j}{\beta - j}) \\
&= \frac{\beta}{t} + \frac{1}{t}\sum_{j=1}^{t-1} \frac{j}{\beta - j} = \frac{\beta}{t} + \frac{1}{t}\sum_{j=\beta-t+1}^{\beta-1} \frac{\beta - j}{j} \\
&= \frac{1}{t}(\beta + \beta \sum_{j=\beta-t+1}^{\beta-1} \frac{1}{j} - \sum_{j=\beta-t+1}^{\beta-1} 1) \cong \frac{1}{t}(\beta + \beta \ln \frac{\beta}{\beta - t} - t) \\
&= \frac{1}{x}(1 - \ln(1-x)) - 1 \geq 2.146 \qquad (\text{for } x = \tfrac{t}{\beta})
\end{aligned}$$

The last inequality holds since the ultimate term is convex in the range $x \in [0,1]$ and has a minimum at $x^* \cong 0.682$. Finally, $opt = |N^*| + |Z| \leq (1 + \frac{1}{2.146})|N^*| \cong 1.466|N^*|$, which means a 0.682-approximation. [10] To see that this approximation ratio is tight for the LAYERS mechanism, observe that the worst case instance described above can be constructed as follows. For every $j = 1, 2, \dots, t$, agents of N'_j arrive and are allocated slots L_j, followed by $N_j \setminus N'_j$ which occupy U_j. Then all of $N_j, j = t+1, \dots, \beta$ arrive and occupy S^*_j of the last layers. Finally, Z arrive but cannot be allocated any slot since all of $S^*_j, j \leq \beta$ are occupied. □

LEMMA 19. *Suppose we assign m' slots to m' agents using SNE prices* $\mathbf{p}$. *Agent utility is non-decreasing in the distance from g. That is, if $d(g, \sigma(i)) > d(g, \sigma(j))$, then $u_i(\sigma(i)) \geq u_j(\sigma(j))$.*

PROOF. Suppose that slots are ordered by increasing distance $d_s = d(g, s)$, and the m' agents are sorted by decreasing cost c_i. Then in the optimal allocation $\sigma(i) = i$.

In every SNE (see [18]), we have $p_{m'} = 0$, and for any $i < m'$,

$$p_i \in [p_{i+1} + (x_i - x_{i+1})c_{i+1}, p_{i+1} + (x_i - x_{i+1})c_i].$$

Since $x_i > x_{i+1}$, then this range is non-singleton whenever $c_i > c_{i+1}$. The utility of agent i is

$$\begin{aligned}
&u_i(s_i) = \phi - d_i c_i - p_i \leq \phi - d_i c_i - (p_{i+1} + (x_i - x_{i+1})c_{i+1}), \\
&u_{i+1}(s_{i+1}) = \phi - d_{i+1}c_{i+1} - p_{i+1}. \quad \Rightarrow \\
&u_{i+1}(s_{i+1}) - u_i(s_i) \geq -d_{i+1}c_{i+1} + d_i c_i + (x_i - x_{i+1})c_{i+1} \\
&= (D - x_i)c_i - (D - x_{i+1})c_{i+1} + (x_i - x_{i+1})c_{i+1} \\
&= (D - x_i)(c_i - c_{i+1}) = d_i(c_i - c_{i+1}) \geq 0. \qquad \square
\end{aligned}$$

[10] While an analytic solution is not at hand, computing the approximation ratio to a higher degree of accuracy reveals that it exactly equals x^*.