Mathematical Programming (2024) 203:901–930
https://doi.org/10.1007/s10107-022-01902-8

FULL LENGTH PAPER

Series B

Truthful facility assignment with resource augmentation: an exact analysis of serial dictatorship

Ioannis Caragiannis[1] · Aris Filos-Ratsikas[2] · Søren Kristoffer Stiil Frederiksen[1] · Kristoffer Arnsfelt Hansen[1] · Zihan Tan[3]

Received: 28 January 2021 / Accepted: 5 October 2022 / Published online: 2 November 2022

Abstract

We study the *truthful facility assignment* problem, where a set of agents with private most-preferred points on a metric space have to be assigned to facilities that lie on the metric space, under capacity constraints on the facilities. The goal is to produce such an assignment that minimizes the social cost, i.e., the total distance between the most-preferred points of the agents and their corresponding facilities in the assignment, under the constraint of truthfulness, which ensures that agents do not misreport their most-preferred points. We propose a *resource augmentation framework*, where a truthful mechanism is evaluated by its worst-case performance on an instance with enhanced facility capacities against the optimal mechanism on the same instance with the original capacities. We study a well-known mechanism, Serial Dictatorship, and provide an exact analysis of its performance. Among other results, we prove that Serial Dictatorship has approximation ratio $g/(g-2)$ when the capacities are multiplied by any integer $g \geq 3$. Our results suggest that with a limited augmentation of the resources we can achieve exponential improvements on the performance of the mechanism and in particular, the approximation ratio goes to 1 as the augmentation factor becomes large. We complement our results with bounds on the approximation ratio of Random Serial Dictatorship, the randomized version of Serial Dictatorship, when there is no resource augmentation.

Keywords Mechanism design without money · Serial dictatorship · Resource augmentation · Approximation ratio

A preliminary version of this paper appeared in the 12th Conference on Web and Internet Economics (WINE '16).

✉ Aris Filos-Ratsikas
Aris.Filos-Ratsikas@ed.ac.uk

1 Department of Computer Science, Aarhus University, Aarhus, Denmark

2 School of Informatics, University of Edinburgh, Edinburgh, UK

3 DIMACS, Rutgers University, Newark, USA

Mathematics Subject Classification 68W25 · 68W40 · 91A68

1 Introduction

We study the *facility assignment problem*, in which there is a set of agents and a set of *facilities* with finite capacities; facilities are located on a metric space at points F_i and each agent has a most-preferred point A_i, which is her private information. The goal is to produce an *assignment* of agents to facilities, such that no capacity is exceeded and the sum of distances between agents and their assigned facilities, *the social cost*, is minimized. A *mechanism* is a function that elicits the points A_i from the agents and outputs an assignment. We will be interested in *truthful* mechanisms, i.e., mechanisms that do not incentivize agents to misreport their most-preferred locations and we will be aiming to find mechanisms that achieve a social cost that is as close as possible to that of the optimal assignment when applied to the true points A_i of the agents.

Our setting has various applications such as assigning patients to personal GPs, vehicles to parking spots, children to schools and pretty much any matching environment where there is some notion of distance involved. Crucially, we make the standard assumption that the agents have *metric preferences*, i.e., preferences that are induced by their distances to the facilities. Concretely, the cost of an agent i for a facility F_j is her distance from the facility in the metric space; clearly these distances satisfy the triangle inequality. The metric preference setting has been studied extensively in classic social choice theory (e.g., see [25, 44]) and in computational social choice (e.g., see [4, 7–10, 48] and references therein). The metric space domain restriction is in fact crucial for meaningful approximations to the optimal social cost to be possible; it is not hard to see that in completely unrestricted preference spaces, truthful mechanisms are bound to perform poorly.

Our work falls under the umbrella of *approximate mechanism design without money*, a term coined by [48] to describe problems where some objective function is optimized under the hard constraints imposed by the requirement of truthfulness. The standard measure of performance for truthful mechanisms is the *approximation ratio*, which for our objective, is the worst-case ratio between the social cost of the truthful mechanism in question over the minimum social cost, calculated over all input instances of the problem.

However, it is arguably unfair to compare the performance of a mechanism that is severely limited by the requirement of truthfulness to that of an omnipotent mechanism that operates under no restrictions and has access to the real inputs of the agents, without giving the truthful mechanism any additional capabilities. This is even more evident in general settings, where strong impossibility results restrict the performance of all truthful mechanisms to be rather poor. The need for a departure from the worst-case approach has been often advocated in the literature, but the suggestions mainly involve some average case analysis or experimental evaluations.

Instead, we will adopt a different approach, that has been made popular in the field of online algorithms and competitive analysis [37, 52]; the approach suggests enhancing the capabilities of the mechanism operating under some very limiting requirement (such as truthfulness or lack of information) before comparing to an optimal solution.

Our main conceptual contribution is the adoption of a *resource augmentation approach to approximate mechanism design*. In the resource augmentation framework, we evaluate the performance of a truthful mechanism on an input with additional resources, when compared to an optimal solution for the set of original resources. For our problem, we consider the social cost achievable by a truthful mechanism on some input with augmented facility capacities against the optimal assignment under the original capacities given as input.

More precisely, let I be an input instance to the facility assignment problem and let I_g be the same instance where each capacity has been multiplied by some integer constant g, that we call the *augmentation factor*. Then, the *approximation ratio with augmentation* g of a truthful mechanism M is the worst-case ratio of the social cost achievable by M on I_g over the social cost of the optimal assignment on I, over all possible inputs of the problem. The idea is that if the ratio achievable by a mechanism with small augmentation is much better when compared to the standard approximation ratio, it might make sense to invest in additional resources. At the same time, such a result would imply that the set of "bad" instances in the worst-case analysis is rather pathological and not very likely to appear in practice. To the best of our knowledge, this is the first time that such a resource augmentation framework has been explicitly proposed in algorithmic mechanism design.

From a technical perspective, our contribution in this paper is centered around two well-known truthful mechanisms for matching and assignment problems, Serial Dictatorship (SD) and Random Serial Dictatorship (RSD). Both of these mechanisms have been extensively studied in the context of social choice theory (e.g., see [18, 29, 41, 53]) and serve as excellent starting points for investigating the capabilities of truthful mechanisms in the facility assignment problem. In particular, in a nutshell we show how to apply the resource augmentation approach to SD, in order to obtain much improved approximation ratio bounds, and we compare those bounds to what is achievable by RSD without any resource augmentation. Exploring truthful mechanisms for facility assignment more generally is left for future work, although we do provide some preliminary results and potentially useful observations on that front in Sect. 6.

1.1 Our results

Here we highlight our results in more detail. For SD, we provide an *exact analysis*, obtaining tight bounds on the approximation ratio of the mechanism for all possible augmentation factors g. Specifically, we prove that when n is the number of agents, the approximation ratio with augmentation factor $g = 2$ is exactly $\log(n+1)$ whereas for $g \geq 3$, the approximation ratio is $g/(g-2)$, i.e., a small constant. This is in contrast to the case where there is no augmentation, for which it was known that the approximation ratio of SD is $2^n - 1$. In particular, our results imply that as the augmentation factor becomes large, the approximation ratio of SD with augmentation goes to 1 and the convergence is rather fast. Our results for SD improve and extend some results in the field of online algorithms [36].

To prove the approximation ratios for all augmentation factors, we use an interesting technique based on linear programming. Specifically, we first provide a directed graph interpretation of the assignment produced by SD and the optimal assignment, and then prove that the worst-case instances appear on g-trees, i.e., trees where (practically) every vertex has exactly g successors. Then, we formulate the problem of calculating the worst ratio on such trees as a linear program and bound the ratio by obtaining feasible solutions to its dual. Such a solution can be seen as a "path covering" of the assignment graph and we obtain the bounds by constructing appropriate path coverings of low cost.

We also consider randomized mechanisms and the very well-known Random Serial Dictatorship mechanism. We prove that for augmentation factor 1 (i.e., no resource augmentation), the approximation ratio of the mechanism is between $n^{0.29}$ and n; the result suggests that even a small augmentation ($g = 2$) is a more powerful tool than randomization.

1.2 Related work

Assignment problems are central in the literature of economics and computer science. The literature on one-sided matchings dates back to the seminal paper by [33] and includes many very influential papers, e.g. [18, 53] in economics as well as a rich recent literature in computer science [3, 29, 32, 41]. Serial Dictatorships (or their randomized counterparts) have been in the focus of much of this literature, mainly due to their simplicity and the fragile nature of truthfulness, which makes it quite hard to construct more involved truthful mechanisms. In a celebrated result, [53] characterized a large class of truthful mechanisms by serial dictatorships. Random Serial Dictatorship has also been extensively studied [1, 41] and in fact it was proven [29] that is asymptotically the best truthful mechanism for one-sided matchings under the general (normalized) cardinal preference domain.

The facility assignment problem can be interpreted as a matching problem; somewhat surprisingly, matching problems in metric spaces have only fairly recently been considered in the mechanism design literature. Emek et al. [24] study a setting very closely related to ours, where the goal is to find matchings on metric spaces, but they are interested in how well a mechanism that produces a stable matching can approximate the cost of the optimal matching. In a conceptually similar work, [5] study the performance of *ordinal* matching mechanisms on metric spaces, when the limitation is the lack of information. The fundamental difference between those works and ours is that we consider truthful mechanisms and bound their performance due to the truthfulness requirement. In a paper that was published independently and at the same time as the conference version of the present paper, [6] studied matching and clustering mechanisms in metric spaces, under the constraints of both limited information and truthfulness. Crucially however, their model involves weights and their objectives are solutions (e.g. matchings) of maximum weight, rather than min-cost matchings like we do here, which makes the problem substantially different. Another difference between our work and the aforementioned papers is that they do not consider resource augmentation and only bound the performance of mechanisms on the same set of resources.[1] However, given the generality of the augmentation framework, the same idea could be applied to their settings. In that sense, our paper proposes a *resource augmentation approach to algorithmic mechanism design* that could be adopted in most resource allocation and assignment settings.

Resource Augmentation: As we mentioned earlier, the idea of resource augmentation was popularized by the field of online algorithms and competitive analysis and is tightly related to the literature on *weak adversaries* where an online competitive algorithm is compared to the adversary that uses a smaller number of resources. The idea for this approach originated in the seminal paper by [52] and has been adopted by others ever since [39, 54]; the term "resource augmentation" was explicitly introduced in this context by [37].

As [37, p. 618-619] explain, a high competitive ratio is typically interpreted as a definitive inability of the algorithm to perform well against the omnipotent optimal ("the power of clairvoyance"), but in reality, often a small increase in processing power can actually have notable effects on how well the online algorithm fares against the adversary. A very similar argument can be made for our setting where instead of "online algorithm" we have a "truthful mechanism" and the "clairvoyant adversary" is one that knows the real preferences of the agents, rather than the future.

An additional interpretation of the bounds obtained for augmented resources is with regard to the worst-case instances: a resource augmentation analysis allows "to exclude those abnormal instances where the value of objective function changes drastically in response to a small change in processor speed" [37], or, in our setting, a small change in the capacity of the facilities. Indeed, upon inspection of the worst-case instance for Serial Dictatorship (without resource augmentation), one can see that the very high approximation ratio is due to a "mistake" in the assignment of the first agent to the first facility, which triggers a chain of inefficient assignments, leading to a high social cost in the end. It can be easily seen that by increasing the capacity of the first facility on the chain by 1, we obtain an optimal social cost; this demonstrates that the worst-case bound is given by a pathological instance. For additional discussion of the resource augmentation approach in the context of "beyond worst-case analysis", we refer the reader to the note by [51].

Most closely related to our problem is the *online metric matching* problem [35, 40, 45], also known as the *online transportation problem* [36]. In this problem, there is a set of points F that lie on a metric space and a set of points A that arrive in an online fashion, i.e., at unknown times in the future. At each time that a point in A arrives, it has to be matched to a point in F. The performance of an online algorithm is measured by its competitive ratio, i.e., the worst-case ratio over all inputs of the total cost of the algorithm over the total cost of the optimal matching, that knows the exact sequence of arriving points in advance. In relation to our setting, results about the greedy algorithm in the online metric matching problem imply bounds for the facility assignment problem and vice versa. However, contrary to the resource augmentation results in [36], our analysis is *exact*, i.e., our results involve no asymptotics. Additionally, we remark that our analysis is substantially different due to the use of linear programming;

[1] With the exception of the bi-criteria result in [5].

our primal-dual technique could be applicable for greedy assignment mechanisms on other resource augmentation settings, beyond the problem studied here. We discuss the relation between the two settings as well as the implications of our results to the online setting in more detail in Sect. 6.1.

Finally, there is some resemblance between our problem and the facility location problem [48] that has been studied extensively in the literature of approximate mechanism design, in the sense that in both settings, agents specify their most preferred positions on a metric space. Note that the settings are fundamentally different however, since in the facility location problem, the task is to identify the appropriate point to locate a facility whereas in our setting, facilities are already in place and we are looking for an assignment of agents to them.

2 Preliminaries

In the *facility assignment* problem, there is a set $N = \{1, \ldots, n\}$ of agents and a set $M = \{1, \ldots, m\}$ of *facilities*, where agents and facilities are located on a metric space, equipped with a distance function d. Each facility has a *capacity* $c_i \in \mathbb{N}_+$, which is the number of agents that the facility can accommodate. We assume that $\sum_{i=1}^{m} c_i \geq n$, i.e., all agents can be accommodated by some facility. Each agent has a most preferred position A_i on the space and his cost $d_i(j)$ from facility j is the distance $d(A_i, F_j)$ between A_i and the position F_j of the facility. Let $A = (A_1, \ldots, A_n)$ be a vector of preferred positions and call it a *location profile*. Let $F = (F_1, \ldots, F_m)$ be the corresponding set of points of the facilities. A pair of agents' most preferred points and facility points (A, F) is called an *instance* of the facility assignment problem and is denoted by I.

The locations of the facilities are known but the location profiles are not known; agents are asked to report them to a central planner, who then decides on an *assignment* S, i.e., a pairing of agents and facilities such that each agent is assigned to exactly one facility and no facility capacity is exceeded. Let S_i be the restriction of the assignment to the i'th coordinate, i.e., the facility to which agent i is assigned in S and let $\mathcal{S}$ be the set of all assignments. The *social cost* of an assignment S on input I is the sum of the agents' costs from their facilities assigned by S i.e., $\sum_{i=1}^{n} d_i(S_i)$. A deterministic mechanism maps instances to assignments whereas a randomized mechanism maps instances to probability distributions over assignments.

Definition 1 (Truthfulness) A mechanism M is *truthful* if no agent has an incentive to misreport his most preferred location. Formally, this is guaranteed when for every location profile A, any report A'_i, and any reports A_{-i} of all agents besides agent i, it holds that

$$d_i(S_i) \leq d_i(S'_i), \text{ where } S = M(I) \text{ and } S' = M(I')$$
$$\text{with } I = (A, F) \text{ and } I' = ((A'_i, A_{-i}), F).$$

For randomized mechanisms, the corresponding notion is *truthfulness-in-expectation*, where an agent cannot decrease her expected distance from the assigned facilities by deviating, i.e., it holds that $\mathbb{E}_{S\sim D}[d_i(S_i)] \leq \mathbb{E}_{S\sim D'}[d_i(S_i)]$, where D and D' are the probability distributions output by the mechanism on inputs I and I' respectively. A stronger notion of truthfulness for randomized mechanisms is that of *universal truthfulness*, which guarantees that for every realization of randomness, there will not be any agent with an incentive to deviate. Alternatively, one can view a universally truthful mechanism as a mechanism that runs a deterministic truthful mechanism at random, according to some distribution.

Resource Augmentation: As our main conceptual contribution, we will consider a *resource augmentation* framework where the minimum social cost of any assignment will be compared with the social cost achievable by a mechanism on a location profile with augmented facility capacities. Given an instance I, we will use the term *g-augmented instance* to refer to an instance of the problem where the input is I and the capacity of each facility has been multiplied by g. We will denote that instance by I_g and we will call g the *augmentation factor* of I. For example, when $g = 2$, we will compare the minimum social cost with the social cost of a mechanism on the same inputs but with double capacities.

For the facility assignment problem, the optimal mechanism computes a minimum cost matching (which can be computed using an algorithm for maximum weight bipartite matching) and it can be easily shown that it is not truthful; in order to achieve truthfulness, we have to output suboptimal solutions. As performance measure, we define the *approximation ratio with augmentation* of a mechanism M as

$$ratio_g(M) = \sup_I \frac{SC_M(I_g)}{SC_{OPT}(I)}$$

where $SC_M(I_g) = \sum_{i=1}^n d_i(M(I_g)_i)$ is the social cost of the assignment produced by mechanism M on input instance I with augmentation factor g and $SC_{OPT}(I)$ is the minimum social cost of any assignment on I i.e., $SC_{OPT}(I) = \min_{S\in\mathcal{S}} \sum_{i=1}^n d_i(S_i)$. More generally, we will use $SC_X(I) = \sum_{i=1}^n d_i(X_i)$ to denote the social cost of assignment X on I. For randomized mechanisms, the definitions involve the expected social cost and are very similar. Obviously, if we set $g = 1$, we obtain the standard notion of the approximation ratio for truthful mechanisms [48]. For consistency with the literature, we will denote $ratio_1(M)$ by $ratio(M)$.

SD and RSD: We will be interested in two natural truthful mechanisms that assign agents to facilities in a greedy nature. A *serial dictatorship* (SD) is a mechanism that first fixes an ordering of the agents and then assigns each agent to his most preferred facility, from the set of facilities with non-zero residual capacities. The fact that the mechanism is truthful is straightforward; an agent does not affect the ordering in which it is chosen and once chosen, the agent will "select" the best possible facility, according to its preference. The randomized counterpart of SD, *Random Serial Dictatorship* (RSD), is the mechanism that first fixes the ordering of agents uniformly at random and then assigns them to their favorite facilities that still have capacities left. In other words, RSD runs one of the $n!$ possible serial dictatorships uniformly at random and hence it is universally truthful.

We conclude the section with a simple example showing that the optimal assignment is not truthful.

Example 1 (The optimal assignment is not truthful) *Consider an instance with two agents and two facilities with capacities* $c_1 = c_2 = 1$, *such that* $A_1 = A_2$, $d(A_1, F_1) = 2$ *and* $d(A_1, F_2) = 1$. *Assume without loss of generality that in the optimal assignment* $S_1 = 1$ *and* $S_2 = 2$, *i.e., agent* $i \in \{1, 2\}$ *is assigned to facility* i. *Now, assume that agent* 1 *reports* $A'_1 = F_2$ *instead, i.e.,* $d(A'_1, F_2) = 0$ *and* $d(A'_1, F_2) = 3$. *Then, the optimal assignment would be such that* $S_1 = 2$ *and* $S_2 = 1$. *The real cost of agent* 1 *has therefore reduced from* 2 *to* 1, *and the agent has successfully manipulated the mechanism that outputs the optimal assignment.*

3 Approximation guarantees for serial dictatorship

In this section we provide our main results, the upper bounds on the approximation ratio with augmentation of Serial Dictatorship, for all possible augmentation factors. In Sect. 5, we state the theorem that ensures that the bounds proven here are tight.

Theorem 1 *The approximation ratio of SD with augmentation factor g in facility assignment instances with n agents is*

1. $ratio(SD) \leq 2^n - 1$,
2. $ratio_2(SD) \leq \log(n + 1)$,
3. $ratio_g(SD) \leq \frac{g}{g-2}$ *when* $g \geq 3$.

In order to prove the theorem,[2] we first need to introduce a different interpretation of the assignment produced by SD and the optimal assignment, in terms of a directed graph. We begin with a roadmap of the proof of Theorem 1.

1. We show how to represent an instance of facility assignment together with an optimal solution and a solution computed by the SD mechanism as a directed graph and argue that the instances in which the SD mechanism has the worst approximation ratio are specifically structured as directed trees (see Lemma 1).
2. We then observe that the cost of the SD mechanism in these instances is upper-bounded by the objective value of a maximization linear program defined over the corresponding directed trees.
3. We use duality to upper-bound the objective value of this LP by the value of a feasible solution for the dual LP. This reveals a direct relation of the approximation ratio of the SD mechanism to a graph-theoretic quantity defined on a directed tree, which we call the cost of a path covering (see Definition 2 and Lemma 2).
4. Our last step is to prove bounds on this quantity; these might be of independent interest and could find applications in other contexts (see Lemmas 3, 4, and 5).

Consider an instance I of facility assignment. Recall the interpretation of the problem as a metric bipartite matching and note that without loss of generality, each facility

[2] We point out here that statement 1 and a weaker version of statement 2 in Theorem 1 can be obtained as corollaries of results in the literature for the online metric matching problem (see [35, 36]). However, we will prove the three statements of Theorem 1 as part of our more general framework.

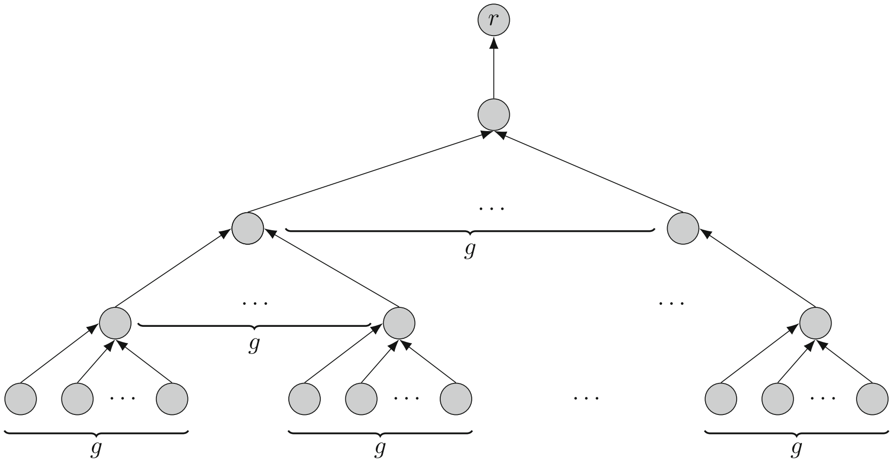

Fig. 1 An illustration of a g-tree. Nodes correspond to facilities and edges correspond to agents. An edge between two nodes means that the corresponding agent was assigned to the facility corresponding to the source at the optimal assignment, and to the facility corresponding to the target in the assignment of the SD mechanism

can be assumed to have capacity 1, and $m \geq n$. Unless otherwise specified, agents and facilities are identified by the integers in $[n]$ and $[m]$, respectively.

Now, let O be any assignment on input I, and let S be an assignment returned by the SD mechanism when applied on the instance I_g (where each facility has capacity g). We use a directed graph to represent the triplet I, O, and S as follows. The graph has a node for each facility. Each directed edge corresponds to an agent. A directed edge from a node corresponding to facility j_1 to a node corresponding to facility j_2 indicates that the agent corresponding to the edge is assigned to facility j_1 in O and to facility j_2 in S. Observe that there is at most one edge outgoing from each node; this edge corresponds to the agent that is assigned to the facility corresponding to the node in solution O. Furthermore, a node may have up to g incoming edges, corresponding to agents assigned to the facility by the SD mechanism.

Representations as directed g-trees are of particular importance. A *directed g-tree* T (or simply g-tree), is an acyclic directed graph that has a root node r of in-degree 1 and out-degree 0, leaves with in-degree 0 and out-degree 1, and intermediate nodes with in-degree g and out-degree 1 (see Fig. 1 for an illustration). We now show that it suffices to restrict our attention to directed g-trees as graph representations of instances in which the SD mechanism achieves its worst performance.

Lemma 1 *Given an integer $g \geq 1$, an instance I with n agents, with an optimal solution O and a solution S consistent with the SD mechanism when applied to instance I_g, there is another instance I', with an optimal solution O' and a solution S' consistent with the application of the SD mechanism on the instance I'_g such that the representation graph of the triplet (I', O', S') is a directed g-tree and such that*

$$\frac{SC_S(I_g)}{SC_O(I)} \leq \frac{SC_{S'}(I'_g)}{SC_{O'}(I')}.$$

In addition, if $g \in \{1, 2\}$, the number of agents in I' is at most n.

Proof Let o_i and s_i denote the facility to which agent i is connected in assignments O and S, respectively. Starting from (I, O, S), we construct a new triplet (I', O', S') satisfying the conditions of the lemma as follows:

Step 1 First, we transform the representation graph so that it consists of just a single weakly-connected component. The transformation does not increase the number of agents and is done as follows. Assume that the current representation graph has $k \geq 2$ weakly-connected components. For $j = 1, ..., k$, we define the triplet $(I^{(j)}, S^{(j)}, O^{(j)})$ as follows. The instance $I^{(j)}$ consists of the agents and facilities that correspond to the edges and nodes of the j-th weakly-connected component of the representation graph, respectively. The assignment $S^{(j)}$ is an assignment returned by the SD mechanism when applied on the instance $I_g^{(j)}$ (where each facility has capacity g), assuming that the order in which SD decides the assignment for each agent in $I^{(j)}$ is the same with the order in which SD decides their assignment in instance I. Also, $O^{(j)}$ is the restriction of assignment O to the agents of instance $I^{(j)}$. Now, among the triplets $(I^{(j)}, O^{(j)}, S^{(j)})$, we keep only the one that maximizes the ratio $\frac{SC_{S^{(j)}}(I_g^{(j)})}{SC_{O^{(j)}}(I^{(j)})}$. This ratio is not smaller than $\frac{SC_S(I_g)}{SC_O(I)}$. To see why, observe that $SC_S(I_g) = \sum_{j=1}^k SC_{S^{(j)}}(I_g^{(j)})$ and $SC_O(I) = \sum_{j=1}^k SC_{O^{(j)}}(I^{(j)})$.

Step 2 Next, we eliminate cycles from the graph representation. We do so by introducing new facilities but without increasing the number of agents. Let $i_0, i_1, ..., i_{k-1}$ be the $k \geq 2$ agents corresponding to the edges in such a cycle, with $s_{i_j} = o_{i_{j+1 \bmod k}}$ such that agent i_{k-1} is the last agent in the cycle who is assigned to a facility by SD. We introduce a new facility f, set $d(A_{i_0}, F_f) = d(A_{i_0}, F_{o_{i_0}})$ and change the assignment of agent i_0 in O to f. Agent i_0 is still assigned to facility s_{i_0} in S; this is consistent to SD due to the definition of distance $d(A_{i_0}, F_f)$. After repeating this for all cycles, the representation graph is an acyclic directed graph, that has a root node r of in-degree at most g and out-degree 0, leaves with in-degree 0 and out-degree 1, and intermediate nodes with in-degree at most g and out-degree 1. The quantity $SC_S(I_g)/SC_O(I)$ has not changed.

To make sure that SD will not assign any agent to the new facility f, we update the distances between agents and facilities they do not use in O and S to the *shortest path distance* in the graph representation as follows. Let i be an agent and j a facility. Let j' be the facility that is closest to j in the graph representation, among the two facilities that agent i uses in O and S. Let P be the set of agents that correspond to the edges in the path from node j to node j'. We set the distance $d(A_i, F_j)$ to $d(A_i, F_{j'}) + \sum_{t \in P} (d(A_t, F_{o_t}) + d(A_t, F_{s_t}))$. Notice that this transformation does not change the distance between an agent and a facility the agent uses in S or in O. Furthermore, the shortest path distance can only increase the distance of the agent to a facility she uses neither in S nor in O. Hence, the transformation does not affect the behavior of SD, nor the social cost of the solutions S and O. We maintain this shortest path distance property in the following steps.

Step 3 We perform a similar transformation with step 1 to get a graph representation in which the root node has in-degree 1. Assume that the current representation graph has a root node of in-degree $k \geq 2$. For $j = 1, ..., k$, we define the triplet $(I^{(j)}, S^{(j)}, O^{(j)})$

as follows. The instance $I^{(j)}$ consists of the agents and facilities that correspond to the edges and nodes of the j-th subtree of the root, including a copy of the root facility. The assignment $S^{(j)}$ is an assignment returned by the SD mechanism when applied on the instance $I_g^{(j)}$ (where each facility has capacity g), assuming that the order in which SD decides the assignment for each agent in $I^{(j)}$ is the same with the order in which SD decides their assignment in instance I. Also, $O^{(j)}$ is the restriction of assignment O to the agents of instance $I^{(j)}$. Like above, among the triplets $(I^{(j)}, O^{(j)}, S^{(j)})$, we keep only the one that maximizes the ratio $\frac{SC_{S^{(j)}}(I_g^{(j)})}{SC_{O^{(j)}}(I^{(j)})}$. Again, this ratio will not be smaller than $\frac{SC_S(I_g)}{SC_O(I)}$, since $SC_S(I_g) = \sum_{j=1}^{k} SC_{S^{(j)}}(I_g^{(j)})$ and $SC_O(I) = \sum_{j=1}^{k} SC_{O^{(j)}}(I^{(j)})$.

Step 4a Notice that if $g = 1$, the current representation graph will be a single directed path, i.e., a 1-tree. Also, recall that no new agents were introduced so far. So, the proof of the lemma is complete for $g = 1$.

For $g \geq 2$, the only difference the current graph representation may have with a g-tree is that intermediate nodes may have in-degree smaller than g. We take care of this in the last step, distinguishing between the case $g = 2$ and $g \geq 3$.

Step 4b If $g = 2$, the current graph representation may have "lacking" paths, which originate from nodes of in-degree either 0 or 2 and are destined either for the root node or for nodes of in-degree 2, so that all their intermediate nodes have in-degree 1. Consider such a lacking path and assume that it consists of edges corresponding to the $k \geq 2$ agents $i_1, i_2, ..., i_k$. Hence, the lacking path originates from node o_{i_1} and is destined for node s_{i_k}. Consider two triplets $(I^{(1)}, O^{(1)}, S^{(1)})$ and $(I^{(2)}, O^{(2)}, S^{(2)})$ defined as follows. Instance $I^{(1)}$ consists of agent i_k, all agents whose path to the root node does not contain the edge corresponding to agent i_k, and the facilities all these agents use in assignments O and S. The assignments $O^{(1)}$ and $S^{(1)}$ are the restrictions of the assignments O and S to the agents and facilities of instance $I^{(1)}$. We remark that the assignment of agent i_k to facility s_{i_k} (as opposed to o_{i_k}) in $S^{(1)}$ is consistent to SD. This is due to the fact that agent i_k is assigned to facility s_{i_k} by SD in S, even though facility o_{i_k} is used by only one agent (recall that node o_{i_k} has degree 1) and, hence, $d(A_{i_k}, F_{s_{i_k}}) \leq d(A_{i_k}, F_{o_{i_k}})$. Instance $I^{(2)}$ consists of agent i_1, the agents corresponding to edges in the subtrees of node o_{i_1} and the facilities all these agents use in assignments O and S. The assignments $O^{(2)}$ and $S^{(2)}$ are the restrictions of the assignments O and S to the agents and facilities of instance $I^{(2)}$.

Notice that the graph representation of both triplets $(I^{(1)}, O^{(1)}, S^{(1)})$ and $(I^{(2)}, O^{(2)}, S^{(2)})$ have one lacking path fewer than triplet (I, O, S). Notice that, if there are agents that are not included in $I^{(1)}$ and $I^{(2)}$ (i.e., if $k > 2$), their contribution to $SC_S(I_g)$ is not greater than their contribution to $SC_O(I)$, i.e., $\sum_{i=2}^{k-1} d(A_i, F_{s_i}) \leq \sum_{i=2}^{k-1} d(A_i, F_{o_i})$. This is due to the fact that when SD decided the assignment for such an agent $i \in \{2, ..., k-1\}$, facility o_i had less than g agents assigned to it. Since agent i is assigned to facility $s_i \neq o_i$ by SD, it must be that $d(A_i, F_{s_i}) \leq d(A_i, F_{o_i})$. Using this observation, we can furthermore see that one of the ratios $\frac{SC_{S^{(1)}}(I_g^{(1)})}{SC_{O^{(1)}}(I^{(1)})}$ and $\frac{SC_{S^{(2)}}(I_g^{(2)})}{SC_{O^{(2)}}(I^{(2)})}$ is not smaller than

$$\frac{SC_S(I_g) - \sum_{i=2}^{k-1} d(A_i, F_{s_i})}{SC_O(I) - \sum_{i=2}^{k-1} d(A_i, F_{o_i})} \geq \frac{SC_S(I_g)}{SC_O(I)}.$$

After repeating this transformation for all lacking paths in the graph representation, we will have the desired 2-tree. Again, observe that no new agents have been introduced. So, the proof of the lemma for the case $g = 2$ is complete.

Step 4c If $g \geq 3$, consider any non-root, non-leaf node f of the representation graph with in-degree $k < g$. We increase the in-degree of node f to g by introducing $g - k$ new facilities $f_1, f_2, ..., f_{g-k}$, $g-k$ new agents $i_1, ..., i_{g-k}$, and setting $d(A_{i_j}, F_{f_j}) = d(A_{i_j}, F_f) = 0$ for $j = 1, ..., g - k$. The assignments S and O are updated so that $s_{i_j} = f_j$ and $o_{i_j} = f$ for $j = 1, ..., g - k$. Distances of the old agents to the new facilities and of the new agents to the old facilities follow the shortest path distance metric; this guarantees that the updated assignment S is consistent to SD. Notice that an old agent i is not assigned to a new facility f_j by SD. Indeed, the shortest path distance from agent i to facility f_j is at least as high as the distance from agent i to facility f in the original instance. Hence, as agent i is assigned to facility s_i instead of f by SD in S, she must be assigned to s_i in the updated assignment S as well. After repeating this transformation for all intermediate nodes in the graph representation, we will have the desired g-tree. As the distance of the new agents from their assigned facility in S and O is 0, the ratio $\frac{SC_S(I_g)}{SC_O(I)}$ remains unchanged. This completes the proof of the lemma for the case $g \geq 3$ as well. □

So, in the following, we will focus on triplets (I, O, S) of a facility assignment instance I, with an optimal solution O, and with an SD solution S for instance I_g that have a directed g-tree T as graph representation. Below, we use $\mathcal{P}$ to denote the set of all paths that originate from leaves. Given an edge e of a g-tree, we use $\mathcal{P}_e$ (respectively, $\tilde{\mathcal{P}}_e$) to denote the set of all paths that originate from a leaf and cross (respectively, terminate with) edge e. We always use e_r to denote the edge incident to the root of a g-tree.

Our next observation is that $SC_S(I_g)$ is upper-bounded by the objective value of the following linear program.

$$\begin{aligned}
\text{maximize} \quad & \sum_{e \in T} z_e \\
\text{subject to:} \quad & z_e - \sum_{a \in p \setminus \{e\}} z_a \leq \sum_{a \in p} d(A_a, F_{o_a}), e \in T, p \in \tilde{\mathcal{P}}_e \\
& z_e \geq 0, e \in T
\end{aligned}$$

To see why, interpret variable z_e as the distance of the agent corresponding to edge e of T to the facility it is connected to under assignment S. Then, clearly, the objective $\sum_{e \in T} z_e$ represents $SC_S(I_g)$. Now, how high can $SC_S(I_g)$ be? The LP essentially answers this question (partially, because it does not use all constraints of the SD mechanism, but sufficiently for our purposes). In particular, the LP takes into account the fact that the distance of agent e to the facility to which it is connected in S is not higher than the distance of the agent to any leaf facility in its subtree; this follows by the definition of the SD mechanism since leaf facilities are by definition available throughout the execution of the SD mechanism. Indeed, consider agent e and a path $p \in \tilde{P}_e$. Since agent e is connected to facility s_e under SD and not to the facility corresponding to the leaf from which path p originates from, this means that the distance $d(A_e, F_{s_e})$ is not higher than the distance of A_e from the location of the facility corresponding to that leaf. Since d is a metric, this distance is at most

$$d(A_e, F_{o_e}) + \sum_{a \in p \setminus \{e\}} d(F_{s_a}, F_{o_a}) \leq d(A_e, F_{o_e}) + \sum_{a \in p \setminus \{e\}} (d(A_a, F_{s_a}) + d(A_a, F_{o_a})).$$

So, the constraint associated with path $p \in \tilde{P}_e$ in the LP captures the inequality

$$\begin{aligned} d(A_e, F_{s_e}) &\leq d(A_e, F_{o_e}) + \sum_{a \in p \setminus \{e\}} d(F_{s_a}, F_{o_a}) \\ &\leq d(A_e, F_{o_e}) + \sum_{a \in p \setminus \{e\}} (d(A_a, F_{s_a}) + d(A_a, F_{o_a})), \end{aligned}$$

by replacing $d(A_e, F_{s_e})$ with z_e and $d(A_a, F_{s_a})$ with z_a and rearranging the terms.

By duality, the cost $SC_S(I_g)$ of solution S is upper-bounded by the objective value of the dual linear program, defined as follows:

$$\begin{aligned} \text{minimize} \quad & \sum_{p \in \mathcal{P}} x_p \sum_{e \in p} d(A_e, F_{o_e}) \\ \text{subject to:} \quad & \sum_{p \in \mathcal{P}_{e_r}} x_p \geq 1 \\ & \sum_{p \in \tilde{\mathcal{P}}_e} x_p - \sum_{p \in \mathcal{P}_e \setminus \tilde{\mathcal{P}}_e} x_p \geq 1, e \in T, e \neq e_r \\ & x_p \geq 0, p \in \mathcal{P} \end{aligned}$$

Actually, for any feasible solution x of the dual LP, $SC_S(I_g)$ is upper bounded by the quantity $\sum_{p \in \mathcal{P}} x_p \sum_{e \in p} d(A_e, F_{o_e})$. We will refer to any assignment x over the paths of $\mathcal{P}$ that satisfies the constraints of the dual LP as a *path covering* of the directed g-tree T and will denote its cost by $c(x) = \max_{e \in T} \sum_{p \in \mathcal{P}_e} x_p$. We repeat these definitions for clarity:

Definition 2 Let T be a directed tree. A function $x : \mathcal{P} \rightarrow \mathbb{R}^+$ is called a path covering of T if the following conditions hold:
- $\sum_{p \in \mathcal{P}_{e_r}} x_p \geq 1$ for the edge e_r incident to the root of T;
- $\sum_{p \in \tilde{\mathcal{P}}_e} x_p - \sum_{p \in \mathcal{P}_e \cap \mathcal{P}_f} x_p \geq 1$ if $e \neq e_r$ and f denotes the parent edge of e.[3]

The cost $c(x)$ of x is equal to $\max_{e \in T} \sum_{p \in \mathcal{P}_e} x_p$.

The following lemma establishes that the approximation ratio with augmentation factor g of SD is bounded by the cost of the path covering.

[3] Note that $\mathcal{P}_e \setminus \tilde{\mathcal{P}}_e$ and $\mathcal{P}_e \cap \mathcal{P}_f$ are the same set.

Lemma 2 *Let $g \geq 1$ be an integer, I be a facility assignment instance with an optimal solution O, S be a solution of the SD mechanism when applied on instance I_g, so that the triplet (I, O, S) is represented as a directed g-tree T which has a path covering x. Then, $SC_S(I_g) \leq c(x) \cdot SC_O(I)$.*

Proof Using the interpretation of the variables of the primal LP, duality, and the definition of the cost of path covering x, we have that

$$\begin{aligned} SC_S(I_g) &\leq \sum_{e \in T} z_e \leq \sum_{p \in \mathcal{P}} x_p \sum_{e \in p} d(A_e, F_{o_e}) = \sum_{e \in T} d(A_e, F_{o_e}) \cdot \sum_{p \in \mathcal{P}_e} x_p \\ &\leq c(x) \cdot \sum_{e \in T} d(A_e, F_{o_e}) = c(x) \cdot SC_O(I) \end{aligned}$$

as desired. □

In order to establish the upper bounds in Theorem 1, it remains to show that path coverings with low cost do exist; this is what we do in the next three lemmas. We start with the statement for no augmentation.

Lemma 3 *Let T be a 1-tree with n edges. Then, there is a path covering of T of cost $2^n - 1$.*

Proof First, observe that a directed 1-tree consists of a single branch, where the first node (the leaf) has out-degree 1 and in-degree zero, the last node (the root) has in-degree 1 and out-degree 0 and all other nodes have precisely one incoming edge and one outgoing edge. Therefore, for each edge e in the tree, the set $\tilde{\mathcal{P}}_e$ consists of a single path.

Rename the edge e_r that is incident to the root of T as e_1 and for $i = 2, \ldots, n$, let e_i be the edge that is at depth i from the root. Hence, the edge e_n is incident to the leaf.

Now for every path p_{e_i}, let $x_{p_{e_i}} = 2^{i-1}$. This is a complete assignment to all paths, since every path originates from the single leaf, and ends at either the root or some intermediate node. It is not hard to see that the assignment is a path covering, since $\sum_{p \in P_{e_r}} \geq 1$ (due to $p_{e_r} = 1$), and for every path p_{e_i} it holds that $x_{p_{e_i}} = \sum_{j=1}^{i-1} x_{p_{e_j}} + 1$ and hence, $\sum_{p \in \tilde{\mathcal{P}}_{e_i}} x_p - \sum_{p \in \mathcal{P}_{e_i} \setminus \tilde{\mathcal{P}}_{e_i}} x_p \geq 1$. The cost of the path covering is $c(x) = \sum_{i=1}^n x_{p_{e_i}} = \sum_{i=1}^n 2^{i-1} = 2^n - 1$. □

In the following, we identify path coverings of low cost for the case of $g \geq 3$ and $g = 2$. The next two lemmas complete the proof of Theorem 1 that regards the upper bounds.

Lemma 4 *Let $g \geq 3$ be an integer and T be a g-tree. Then, there is a path covering of T of cost $\frac{g}{g-2}$.*

Proof We prove the lemma using the following assignment x: for every path p of length ℓ, we set $x_p = \frac{1}{g-2} g^{2-\ell}$ if it contains the edge that is incident to the root and $x_p = \frac{g-1}{g-2} g^{1-\ell}$ otherwise.

We will first show that $\sum_{p\in\mathcal{P}_e} x_p = \frac{g}{g-2}$ for every edge e using induction. We will do so by visiting the edges in a bottom-up manner (i.e., an edge will be visited only after its child-edges have been visited) and prove the equality for edge e using the information that the equality holds for its child-edges. As the basis of our induction, consider an edge e that is incident to a leaf at depth $\ell \geq 1$ from the root. If $\ell = 1$, this means that the tree consists of a single edge and there is a single path p with $x_p = \frac{g}{g-2}$. If $\ell \geq 2$, then the paths that contain edge e are those which end at each ancestor of the leaf adjacent to e. Hence,

$$\sum_{p\in\mathcal{P}_e} x_p = \sum_{i=1}^{\ell-1} \frac{g-1}{g-2} g^{1-i} + \frac{1}{g-2} g^{2-\ell} = \frac{g}{g-2}.$$

Now, let us focus on an edge e that is not incident to a leaf. Assume that $\sum_{p\in\mathcal{P}_{e_i}} x_p = \frac{g}{g-2}$ for each child-edge e_i (for $i \in [g]$) of e (this is the induction hypothesis). Let u be the node to which edges e and e_i with $i \in [g]$ are incident. The set of paths in $\mathcal{P}_e$ consists of the following disjoint sets of paths: for each edge e_i and for each path $p \in \tilde{\mathcal{P}}_{e_i}$, set $\mathcal{P}_e$ contains all super-paths of p, i.e., paths originating from the leaf-node reached by p and ending at each ancestor of node u; we use the notation $\sup(p)$ to denote the set of super-paths of p; $\sup(p)$ consists of strict super-paths of p, i.e., it does not contain p. Observe that, the definition of x implies that a super-path q of p that is longer than p by j has $x_q = \frac{1}{g-1} g^{1-j} x_p$ if q is adjacent to the root and $x_q = g^{-j} x_p$ otherwise. Hence, assuming that node u is at depth $\ell \geq 1$ from the root, we have that

$$\begin{aligned}
\sum_{p\in\mathcal{P}_e} x_p &= \sum_{i=1}^{g} \sum_{p\in\tilde{\mathcal{P}}_{e_i}} \sum_{q\in\sup(p)} x_q = \left(\sum_{j=1}^{\ell-1} g^{-j} + \frac{1}{g-1} g^{1-\ell}\right) \sum_{i=1}^{g} \sum_{p\in\tilde{\mathcal{P}}_{e_i}} x_p \\
&= \frac{1}{g-1}\left(\sum_{i=1}^{g} \sum_{p\in\mathcal{P}_{e_i}} x_p - \sum_{p\in\mathcal{P}_e} x_p\right),
\end{aligned}$$

which yields $\sum_{p\in\mathcal{P}_e} x_p = \frac{g}{g-2}$ as desired, since $\sum_{p\in\mathcal{P}_{e_i}} x_p = \frac{g}{g-2}$ by the induction hypothesis.

It remains to show feasibility. Clearly, $\sum_{p\in\mathcal{P}_e} x_p = \frac{g}{g-2} \geq 1$ if e is adjacent to the root. Otherwise, consider an edge e, its parent edge f, and their common endpoint u. Assuming that u is at depth ℓ from the root (and using definitions and observations we used above), we have

$$\sum_{p\in\mathcal{P}_e\cap\mathcal{P}_f} x_p = \sum_{p\in\tilde{\mathcal{P}}_e} \sum_{q\in\sup(p)} x_q = \left(\sum_{j=1}^{\ell-1} g^{-j} + \frac{1}{g-1} g^{1-\ell}\right) \sum_{p\in\tilde{\mathcal{P}}_e} x_p = \frac{1}{g-1} \sum_{p\in\tilde{\mathcal{P}}_e} x_p,$$

which, together with the fact that

$$\frac{g}{g-2} = \sum_{p\in\mathcal{P}_e} x_p = \sum_{p\in\mathcal{P}_e\cap\mathcal{P}_f} x_p + \sum_{p\in\tilde{\mathcal{P}}_e} x_p,$$

yields

$$\sum_{p\in\mathcal{P}_e\cap\mathcal{P}_f} x_p = \frac{1}{g-2} \quad \text{and} \quad \sum_{p\in\tilde{\mathcal{P}}_e} x_p = \frac{g-1}{g-2}$$

and, consequently, $\sum_{p\in\tilde{\mathcal{P}}_e} x_p - \sum_{p\in\mathcal{P}_e\cap\mathcal{P}_f} x_p = 1$ as desired. □

Finally, we state the lemma for augmentation factor $g = 2$.

Lemma 5 *Let T be a 2-tree with n edges. Then, there is a path covering of T of cost at most* $\log(n+1)$.

Proof Again, as in the proof of Lemma 4, we will construct the path covering x by visiting the edges of the tree in a bottom-up manner, i.e., first visiting edges that are incident to leaves and in such a way that an edge that is not adjacent to a leaf is visited only after its two child-edges have been visited. The assignment x will be defined using a temporary assignment y.

When visiting an edge e that is adjacent to a leaf, we determine the temporary value $y_p = \log(n+1)$ associated with the path p consisting of edge e only. When visiting an edge e that is not adjacent to a leaf, we set the temporary value for each path that originates from a leaf and terminates with edge e and determine the final value for each path of $\tilde{\mathcal{P}}_{e_i}$ that originates from a leaf and terminates with the child-edge e_i (with $i \in \{1, 2\}$) of e. In particular, let p be a path of $\tilde{\mathcal{P}}_{e_i}$ and let y_p be the temporary value assigned to it during our previous visit to edge e_i. During the phase associated with edge e, for the super-path q of p that terminates with edge e, we set the temporary value

$$y_q = \frac{\sum_{p'\in\tilde{\mathcal{P}}_{e_i}} y_{p'} - 1}{2\sum_{p'\in\tilde{\mathcal{P}}_{e_i}} y_{p'}} y_p \tag{1}$$

and determine the final value

$$x_p = \frac{\sum_{p'\in\tilde{\mathcal{P}}_{e_i}} y_{p'} + 1}{2\sum_{p'\in\tilde{\mathcal{P}}_{e_i}} y_{p'}} y_p \tag{2}$$

of path p. After we have visited all edges, we set the final value x_p for each path p that originates from a leaf and terminates with edge e_r to be equal to its temporary value y_p.

Consider an edge e that has at least one non-leaf child-edge e_i with $i \in \{1, 2\}$. Let $p \in \tilde{\mathcal{P}}_{e_i}$ and q be the super-path of p that terminates with edge e. Observe that Eqs.

(1) and (2) imply that $x_p + y_q = y_p$, which means that the temporary value of a path p is redistributed as final value of the path and temporary value of its super-path that terminates with its parent edge e. This argument can be repeated for all super-paths of p, and, together with the way we determine the final values of paths in $\mathcal{P}_{e_r}$ at the end of the above process, it implies that the temporary value of a path p in $\tilde{\mathcal{P}}_{e_i}$ is redistributed as total final value of path p and all its super-paths. By denoting the set of super-paths of path p by $\sup(p)$, we get

$$y_p = \sum_{q \in \{p\} \cup \sup(p)} x_q.$$

Clearly, this equality holds if $p \in \mathcal{P}_{e_r}$ as well. Now, observe that Eqs. (1) and (2) imply

$$x_p - y_q = \frac{y_p}{\sum_{p' \in \tilde{\mathcal{P}}_{e_i}} y_{p'}}.$$

Hence, by summing over all paths p of $\tilde{\mathcal{P}}_{e_i}$ and their corresponding super-paths, we have that

$$\sum_{p \in \tilde{\mathcal{P}}_{e_i}} x_p - \sum_{p \in \mathcal{P}_{e_i} \cap \tilde{\mathcal{P}}_e} y_p = 1.$$

Since the temporary value y_p on a path p is redistributed as final value on p and the paths of $\sup(p)$, we have that

$$\sum_{p \in \mathcal{P}_{e_i} \cap \tilde{\mathcal{P}}_e} y_p = \sum_{p \in \mathcal{P}_{e_i} \cap \mathcal{P}_e} x_p,$$

and the feasibility condition

$$\sum_{p \in \tilde{\mathcal{P}}_{e_i}} x_p - \sum_{p \in \mathcal{P}_{e_i} \cap \mathcal{P}_e} x_p = 1 \tag{3}$$

on edge e_i follows by the last two equalities.

We still have to prove the bound on the cost of x as well as the feasibility condition for the edge adjacent to the root. In order to do so, we will prove that for every edge e which defines a subtree with N_e nodes (including both its endpoints), it holds that

$$\log(n+1) \geq \sum_{p \in \mathcal{P}_e} x_p \geq \log(n+1) - \log N_e + 1.$$

The leftmost inequality yields the bound on the cost of x and the rightmost inequality yields the feasiblity constraint $\sum_{p \in \mathcal{P}_{e_r}} x_p \geq 1$ for the edge e_r that is incident to the root (recall that $N_{e_r} = n+1$). We prove the inequality inductively, starting from the edges that are incident to leaves, and proceeding to the remaining edges in a bottom-up manner. To prove the inequality for an edge e that is not adjacent to a leaf, we use the fact that the inequality has been proved for its child-edges before.

Since $N_e = 2$ for every edge e incident to a leaf, both inequalities hold (and are essentially the same equality) in this case. Now consider an edge e that is not incident to a leaf and is such that

$$\log(n+1) \geq \sum_{p\in\mathcal{P}_{e_i}} x_p \geq \log(n+1) - \log N_{e_i} + 1$$

for each child-edge e_i (with $i \in \{1,2\}$) of e. Using the feasibility condition (Eq. 3) for edges e_1 and e_2 (which recall that it holds with equality), we have

$$\begin{aligned}
\sum_{p\in\mathcal{P}_e} x_p &= \sum_{p\in\mathcal{P}_e\cap\mathcal{P}_{e_1}} x_p + \sum_{p\in\mathcal{P}_e\cap\mathcal{P}_{e_2}} x_p \\
&= \frac{1}{2}\sum_{p\in\mathcal{P}_{e_1}\cap\mathcal{P}_e} x_p + \frac{1}{2}\left(\sum_{p\in\tilde{\mathcal{P}}_{e_1}} x_p - 1\right) + \frac{1}{2}\sum_{p\in\mathcal{P}_{e_2}\cap\mathcal{P}_e} x_p + \frac{1}{2}\left(\sum_{p\in\tilde{\mathcal{P}}_{e_2}} x_p - 1\right) \\
&= \frac{1}{2}\sum_{p\in\mathcal{P}_{e_1}} x_p + \frac{1}{2}\sum_{p\in\mathcal{P}_{e_2}} x_p - 1. \qquad (4)
\end{aligned}$$

By the induction hypothesis, we have $\sum_{p\in\mathcal{P}_{e_1}} x_p \leq \log(n+1)$ and $\sum_{p\in\mathcal{P}_{e_2}} x_p \leq \log(n+1)$. So, (4) implies that $\sum_{p\in\mathcal{P}_e} x_p \leq \log(n+1)$ for any edge e as well, and the bound on the cost of x follows.
Using (4) and the assumption on the total final value of paths in $\mathcal{P}_{e_1}$ and $\mathcal{P}_{e_2}$, we get

$$\begin{aligned}
\sum_{p\in\mathcal{P}_e} x_p &\geq \frac{1}{2}\big(\log(n+1) - \log N_{e_1} + 1\big) + \frac{1}{2}\big(\log(n+1) - \log N_{e_2} + 1\big) - 1 \\
&= \log(n+1) - \log\sqrt{N_{e_1}\cdot N_{e_2}} \geq \log(n+1) - \log\left(\frac{N_{e_1}+N_{e_2}}{2}\right) \\
&= \log(n+1) - \log N_e + 1.
\end{aligned}$$

The second inequality follows by the relation of the geometric and arithmetic mean and the last equality is due to the fact that $N_e = N_{e_1} + N_{e_2}$. This completes the proof of the lemma. □

4 Approximation guarantees for random serial dictatorship

In the previous section, we showed that the performance of SD significantly improves even with a small augmentation factor. A natural next question is to study its randomized counterpart, RSD. Could randomization help in achieving much better ratios? In the following, we state an approximation guarantee for RSD, when there is no resource augmentation.

Theorem 2 *The approximation ratio of RSD without resource augmentation in facility assignment instances with n agents is* $ratio(RSD) \leq n$.

Proof Here again, we will use the alternative interpretation of the problem, where there are n agents and n facilities (without loss of generality, capacity slots are interpreted as different facilities of unit capacity which coincide on the same point). We will prove the lemma by induction on n.

When $n = 1$, RSD outputs an optimal solution. For the induction step, assume that for $n = k$, it holds that $ratio(RSD) \leq k$ and consider the case when $n = k + 1$. Let $d_i = \min_{j \in [k+1]} d(A_i, F_j)$ and $t_i = \arg\min_{j \in [k+1]} d(A_i, F_j)$. With a slight abuse of notation, let $SC_{RSD}(L, T)$ be the expected social cost of Random Serial Dictatorship on any instance with agents in L and facilities in T. Similarly, let $SC_{OPT}(L, T)$ denote the social cost of the optimal assignment between agents in L and facilities in T. Then, we have that

$$\begin{aligned}
SC_{RSD}(N, M) &= \frac{1}{k+1} \sum_{i=1}^{k+1} (d_i + SC_{RSD}(N - \{i\}, M - \{t_i\})) \\
&\leq \frac{1}{k+1} \sum_{i=1}^{k+1} d_i + \frac{1}{k+1} \sum_{i=1}^{k+1} k \cdot SC_{OPT}(N - \{i\}, M - \{t_i\}) \\
&\leq \frac{1}{k+1} \sum_{i=1}^{k+1} d_i + \frac{1}{k+1} \sum_{i=1}^{k+1} k \cdot (SC_{OPT}(N, M) + d_i) \\
&= \frac{1}{k+1} \sum_{i=1}^{k+1} d_i + \frac{k}{k+1} \sum_{i=1}^{k+1} (d_i + SC_{OPT}(N, M)) \\
&\leq (k+1) \cdot SC_{OPT}(N, M)
\end{aligned}$$

where the first inequality follows from the induction hypothesis and the last inequality follows from the fact that $SC_{OPT}(N, M) \geq \sum_{i=1}^{k+1} d_i$. For the second inequality, observe first that if in the optimal assignment, agent i is matched with t_i, then the inequality clearly holds. Hence, assume without loss of generality that in the optimal assignment, agent i is matched with some facility $j \neq t_i$ and facility t_i is matched with some agent i^*. Then, if we remove agent i and facility t_i from the optimal assignment on N and M and add the pair i^* and j, we obtain an assignment on $N - \{i\}$, $M - \{t_i\}$. Let S be that assignment and let $SC_S(N - \{i\}, M - \{t_i\})$ be its social cost. By the definition of $SC_{OPT}(N - \{i\}, M - \{t_i\})$, we have

$$\begin{aligned}
SC_{OPT}(N - \{i\}, M - \{t_i\}) &\leq SC_S(N - \{i\}, M - \{t_i\}) \\
&\leq SC_{OPT}(N, M) - d(A_{i^*}, F_{t_i}) - d(A_i, F_j) + d(A_{i^*}, F_j) \\
&\leq SC_{OPT}(N, M) + d(A_i, F_{t_i}) \\
&= SC_{OPT}(N, M) + d_i
\end{aligned}$$

where the last inequality follows from the triangle inequality. This completes the proof of the lemma. □

5 Lower bounds

In this section, we provide lower bounds on the approximation ratio with augmentation of the mechanisms that we study. Interestingly, the constructed instances are all on a simple metric space, the *real line metric*. For SD, the lower bounds that we prove show that our analysis in Sect. 3 is tight. For RSD and augmentation $g = 1$, while the bound is not tight, it shows that even if there is a more involved analysis that potentially yields better upper bounds, it is not possible to obtain a much better approximation ratio and in particular, it is not possible to match the logarithmic approximation guarantee of SD with augmentation $g = 2$. The lower bounds will be established by the following theorem.

Theorem 3 *The approximation ratio of SD with augmentation factor g in facility assignment instances with n agents is*

1. $ratio(SD) \geq 2^n - 1$
2. $ratio_2(SD) \geq \log(n + 1)$
3. $ratio_g(SD) \geq \frac{g}{g-2}$ *when* $g \geq 3$.

The approximation ratio of RSD is at least $ratio(RSD) \geq n^{0.29}$ *(without resource augmentation).*

All the statements in Theorem 3 will follow by the same construction, with agents and facilities lying on the real line. Note that similar instances for proving the simplest cases of Theorem 3 have appeared in the related literature in the past [36, 38, 45]; here we include those instances as part of a more general construction that allows us to obtain lower bounds for different augmentation factors as well as for Random Serial Dictatorship.

Let $k > 0$ be a positive integer and $\epsilon > 0$. There are $k + 2$ points of interest that will host agents and facilities; these have the coordinates $-\epsilon$, 1, 2, ..., and 2^k. For $i = 0, 1, ..., k - 1$, there are ℓ_i agents at level i and are located at point 2^i. For $i = 0, 1, ..., k - 1$, we use $n_i = \sum_{j=0}^{i} \ell_j$. Facilities are partitioned into $k + 1$ levels; each level has a single facility. The facility of level 0 has capacity c_0 and is located at point $-\epsilon$. For $i = 1, 2, ..., k$, the facility of level i is located at point 2^i and has capacity c_i. The different lower bounds will be obtained by setting the values of the quantities ℓ_i and c_i appropriately, but in all cases, we will set $c_i = \ell_i$. Note that then, the optimal cost is at most $\ell_0(1 + \epsilon)$ which is obtained by assigning the agents of level i to the facility of level i for $i = 0, 1, ..., k - 1$. Clearly, the optimal cost can become arbitrarily close to ℓ_0 by selecting ϵ to be sufficiently small.

Proof of Statements (1), (2), and (3) in Theorem 3: We will set the parameters of the construction appropriately and will consider the execution of SD using any ordering of the agents that is *non-decreasing* in terms of level. Let $g \geq 1$ be the augmentation factor. We set $\ell_i = c_i = g^{k-i-1}$ for $i = 0, 1, ..., k - 1$ and $c_k = 1$ (see Fig. 2). Note

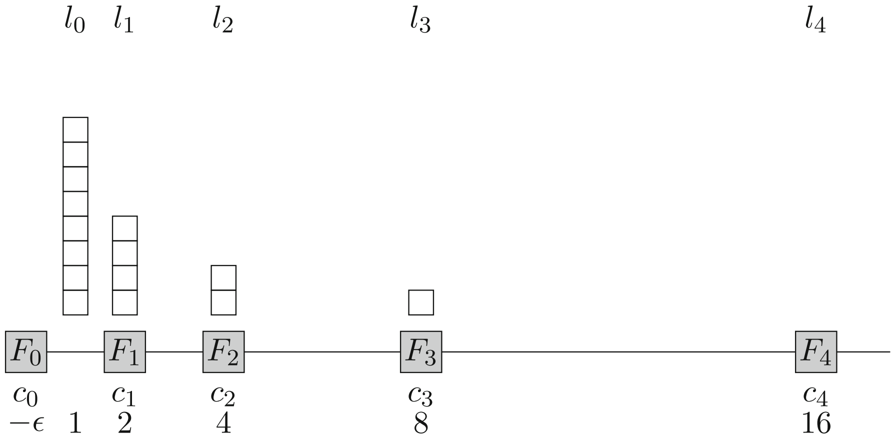

Fig. 2 The lower bound construction of Theorem 3 for 5 facilities ($k = 4$). The gray boxes correspond to facilities, the white boxes correspond to agents. For example, by setting $l_0 = c_0 = 8$, $l_1 = c_1 = 4$, $l_2 = c_2 = 2$ and $l_3 = c_3 = 1$, we obtain the instance for the lower bound when $g = 2$

that the g^{k-1} agents of level 0 that are considered first will be assigned to the facility of level 1 which is their closest one; it is at distance 1 from the agents of level 0, clearly closer compared to facilities of higher levels but also closer compared to the facility of level 0 which is at distance $1+\epsilon$ from the agents of level 0. Note that the (augmented) capacity of the facility of level 1 is exactly g^{k-1} which means that the agents of level 0 occupy it in full. The agents of level 1 appear next in the ordering and are assigned to facility of level 2 (since it is the closest facility that has available space). Again, the agents of level 1 occupy the facility in full. Continuing in this way, we have that the agents of level i (located at point 2^i) are assigned to the facility of level $i+1$ (at point 2^{i+1}) for $i = 0, 1, \ldots, k-1$.

The social cost is then $\sum_{i=0}^{k-1} g^{k-i-1} 2^i = \ell_0 \sum_{i=0}^{k-1} (2/g)^i$ while the number of agents is $n = \sum_{i=0}^{k-1} g^{k-i-1}$. If $g = 1$, we have $n = k$ agents and a social cost of $(2^k - 1)\ell_0 = (2^n - 1)\ell_0$. If $g = 2$, we have $n = 2^k - 1$ and a social cost of $k\ell_0 = \ell_0 \log(n+1)$. Finally, for $g \geq 3$, we have a social cost of $\ell_0 \frac{1-(2/g)^k}{1-2/g} \geq \ell_0 \left(\frac{g}{g-2} - \delta\right)$, where the inequality holds for every positive δ by selecting k to be sufficiently large. This completes the proof of the first three statements.

Proof of the lower bound for *RSD* in Theorem 3: For the lower bound of RSD, we set the parameters of the construction as follows: $\ell_0 = c_0 = 1$ and $\ell_i = c_i = (\gamma - 1)\cdot\gamma^{i-1}$ for $i = 1, \ldots, k-1$, and $c_k = 1$, where γ will be determined later. The proof is slightly more involved. We will need a definition and two technical lemmas.

Definition 3 An ordering has the "chain of levels" property if, for $i = 1, 2, \ldots, k-1$, at least one agent of level i appears after all agents of levels $0, 1, \ldots, i-1$.

Recall that $n_i = \sum_{j=0}^{i} \ell_j$. We now have the following lemma.

Lemma 6 *The probability that a random ordering of the agents has a chain of levels is* $\prod_{i=1}^{k-1}\left(1 - \frac{n_{i-1}}{n_i}\right)$.

Proof We can view the generation of a uniformly random ordering of all agents as a process that proceeds level by level. At level 0, the process simply computes a uniformly random ordering of the agents of level 0. At level $i > 0$, it computes a uniformly random ordering of the agents in levels $0, 1, ..., i$ as follows. It uses the random ordering of the agents in levels $0, 1, ..., i-1$, computes a uniformly random ordering of the agents of level i and picks one among the possible merges of the two orderings uniformly at random.

Now, in each step $i > 0$, the number of possible merges of the two orderings is equal to $\binom{n_i}{n_{i-1}}$ while the number of merged orderings in which the last agent belongs to level i (as the "chain of levels" property requires) is $\binom{n_i-1}{n_{i-1}}$. Since the random events at the different steps are independent, we obtain that the probability that the resulting ordering will have a chain of levels is $\prod_{i=1}^{k-1} \binom{n_i-1}{n_{i-1}} / \binom{n_i}{n_{i-1}} = \prod_{i=1}^{k-1} \left(1 - \frac{n_{i-1}}{n_i}\right)$. □

Lemma 7 *Consider the application of SD on the above instance using an ordering of the agents that has a chain of levels. Then, for* $i = 0, 1, ..., k-1$, *at least one agent of level* i *is assigned to the facility of level* $i+1$.

Proof We first claim that an agent of level i cannot be assigned to a facility of a lower level than i. Assume by contradiction that this is not the case and consider the first agent a in the ordering which is assigned to a facility of a lower level; in particular, let i be the level to which agent a belongs and let $i' < i$ be the level of the facility to which agent a is assigned to. This means that all facilities in levels $i'+1, ..., i$ are full by agents that appear before agent a in the ordering. Note that these facilities cannot contain any agent from levels higher than i (since agent a is the first one that is assigned to a facility of a lower level) or any agent in level i' or lower (since the facility of level i' which is closer to them has free space). Since agent a belongs to level i as well, we obtain that the total capacity of the facilities in levels $i'+1, ..., i$ is strictly smaller than the total number of agents in these levels; this contradicts the definition of the instance.

Now, we will prove the lemma by considering the last agent from each level in an ordering with a chain of levels. When the agent of level 0 is considered, the chain of levels property guarantees that some of the agents of level 1 has not appeared yet. Furthermore, the fact that no agent is ever assigned to a facility of lower level implies that the facility of level 1 (which is closer to the agent compared to the facility of level 0) has free space and the agent of level 0 will be assigned to it. Now, consider the last agent of level 1. When it appears, the facility of level 1 is full; it contains the agent of level 0 and the agents of level 1 before the last one. Again, the chain of levels property guarantees that some of the agents of level 2 has not appeared yet. Together with the fact that no agent is ever assigned to a facility of lower level, this leads again to the conclusion that the facility of level 2 (which is again closer to the agent compared to the facility of level 0 which is still empty) has free space and the agent of level 0 will be assigned to it. Continuing this reasoning completes the proof of the lemma. □

We now complete the proof as follows. Observe that the parameters are such that $n_0 = 1$, $n_i = 1 + (\gamma - 1)\sum_{j=1}^{i} \gamma^{j-1} = \gamma^i$ for $i = 1, ..., k-1$, and the number of agents is $n = n_{k-1} = \gamma^{k-1}$. By Lemma 7, we have that if the random ordering used by RSD happens to have a chain of levels, then some agent of level i will be assigned to the facility of level $i+1$, for $i=0,1,...,k-1$. The social cost in this case is $2^k-1 \geq 2^{k-1}$. By Lemma 6, the probability that a random ordering has a chain of levels is $\left(\frac{\gamma-1}{\gamma}\right)^{k-1}$ and hence, the expected social cost of RSD is at least $\left(2-\frac{2}{\gamma}\right)^{k-1} = n^{\log_\gamma(2-2/\gamma)}$, which using $\gamma=4$, is at least $(\ell_0-\varepsilon)n^{0.29}$. The bound for RSD follows, as ε can become arbitrarily small.

6 Discussion, challenges and future directions

In this paper, we proposed a resource augmentation framework for algorithmic mechanism design, where a mechanism, severely limited by the need for truthfulness is given some additional allocative power before being compared to the optimal mechanism, which operates under no restrictions. We conclude with a discussion of some interesting special cases, some possible future directions and some connections of our results with the related literature and their implications to those settings. We start with the relation of our problem with the literature on online algorithms.

6.1 The online metric matching problem

As we mentioned in the introduction, there is a connection between the facility assignment problem and the *online metric matching problem* (also known as the minimum online metric bipartite matching or the *online transportation problem* [36]), which has been studied in the literature of online algorithms [35, 38, 40, 45].

In the *online metric matching problem*, there is a set of points F on a metric space and a set of points A that arrive in an online fashion. At each time that a point in A arrives, it has to be matched to a point in F. The performance of an online algorithm is measured by its *competitive ratio*, i.e., the worst-case ratio over all inputs of the social cost of the algorithm over the social cost of the optimal matching, that knows the exact sequence of arriving points in advance. Our setting can be interpreted as a similar metric matching problem, by "splitting" facilities with capacity $c_i > 1$ to facilities of unit capacity that coincide on the metric space and by interpreting facilities as single, indivisible objects. Given this interpretation, SD and RSD can be thought of as greedy algorithms for the problem above. In particular, SD corresponds to the greedy algorithm in the setting with adversarial arrivals and RSD corresponds to the greedy algorithm when the arrival of points in A is according to a uniform random permutation.

For the online metric matching problem, it was known since the early 90s that without augmentation, Greedy achieves a competitive ratio of 2^n-1 [35]. Later on, [36] proved that when the online algorithm operates on doubled capacities, Greedy is $\Theta(\log n)$-competitive; given the discussion above, this implies a $\Theta(\log n)$-approximation bound for SD with $g=2$ in our setting. Note however that unlike the result in [36], our analysis is *exact*, i.e., our $\log(n+1)$ bound involves no asymptotics. Furthermore, we extend the result by proving exact bounds for any augmentation factor

$g \geq 3$; the bounds are all small constants and in fact the ratio goes to 1 as the augmentation factor grows large. These results naturally extend to the online transportation problem and confirm a conjecture by [22], namely that a constant competitive ratio can be achieved with augmentation factor 3. Our results for RSD also imply upper and lower bounds for the performance of Greedy in the online transportation problem with uniform random arrivals. Specifically, Theorems 1 and 2 give rise to the following corollary. We state the corollary using the terminology of the online problem for consistency.

Corollary 1 *The double-competitive ratio of* Greedy *for the online metric matching problem is* $\log(n+1)$. *The* g*-competitive ratio of* Greedy *is* $g/(g-2)$, *whenever* $g \geq 3$. *The competitive ratio of* Greedy *for the online metric matching problem with uniform random arrivals is between* $n^{0.29}$ *and* n.

Compared to the related result in [36], we remark that our analysis is substantially different due to the use of linear programming. This technique for the analysis of purely combinatorial algorithms has found applications in many different contexts such as facility location [34], set cover [13], online matching [43], maximum directed cut [26], wavelength routing [19], revenue optimization [2], as well as the price of anarchy of non-cooperative games [17, 42, 46]. Like in our case, these techniques usually lead to tight analysis.

We emphasize that while the connection between SD and RSD and the greedy algorithm for the online metric matching problem is straightforward, the two problems are fundamentally different and hence non-greedy online competitive algorithms do not imply any bounds for our setting and non-serial truthful mechanisms do not imply any bounds for the online setting. For example, while [36] also prove that a modification of the greedy algorithm called Balance in fact achieves a $O(1)$ competitive ratio with augmentation factor 2 for the online metric matching problem, this result does not have any implications to our setting, because Balance cannot be translated to a truthful mechanism in any meaningful way. For some more recent results on the online metric matching problem and its variants, we refer the reader to the works of [11, 12, 15, 16, 22, 47, 49, 50] and references therein.

6.2 Beyond greedy mechanisms

Our main results developed in earlier sections regard the performance of SD and RSD for several augmentation factors. Ideally, we would like to know whether there exist truthful mechanisms that achieve better approximation ratios either for $g = 1$ or for larger values of g. Answering these questions seems like a quite demanding task. On one hand, proving lower bounds for all truthful mechanisms is hard given that, besides truthfulness, we are not aware of any structural properties of those mechanisms. On the other hand for the upper bounds, we do not have any promising candidate truthful mechanisms in the literature.

In this section, we provide some results in this direction. First, we settle the best achievable approximation ratio for the case of two facilities and arbitrary capacities, when there is no resource augmentation in Sect. 6.2.1 below. Then in Sect. 6.2.2 we discuss how some seemingly promising approaches that have been effective in other related settings in the literature turn out to be insufficient for our setting.

6.2.1 Tight bounds for two facilities without resource augmentation

We start from the observation that for $m = 2$, the approximation ratio of SD is in fact 3. This follows from [35, Theorem 2.5], where it is proven that the competitive ratio of the greedy algorithm is $2^m - 1$ (where m is the number of "servers", corresponding to the number of facilities in our setting). Given the discussion above in Sect. 6.1, this directly implies a $2^m - 1$ upper bound for our problem.

Therefore, it suffices to prove that no truthful-in-expectation mechanism can achieve a better ratio. We start with the following simple lemma, which is similar to lemmas that have been proved before [28, 29, 32] in different settings. The statement of the lemma refers to the class of anomymous mechanisms. A mechanism is *anonymous* if the assignment does not depend on the names of the agents; formally if for any location profile $A = (A_1, A_2, \ldots, A_n)$, every agent i and any permutation $\pi : \mathbb{N} \to \mathbb{N}$, it holds that $M(A, F)_i = M(A_{\pi(1)}, A_{\pi(2)}, \ldots A_{\pi(n)}, F)_{\pi(i)}$. By this definition, in an anonymous mechanism, agents with exactly the same locations must have the same probabilities of being assigned to each facility.

Lemma 8 *For any truthful-in-expectation mechanism M, there exists an anonymous truthful-in-expectation mechanism M', such that ratio$(J') \leq$ ratio(J) for any augmentation factor g.*

Proof The proof is similar to those of the corresponding lemmas in [28, 29, 32]. Let M' be the mechanism that given any instance I_g applies a uniformly random permutation to the set of indices of the agents and then applies M on I_g. The mechanism is clearly anonymous. Furthermore, since I_g is a valid input to M, the approximation ratio of M' cannot be worse than that of M, since the approximation ratio is calculated over all possible input instances. For the same reason, if M is truthful-in-expectation and since the permutation is independent of the reports, M' is truthful-in-expectation. $\square$

We now can use Lemma 8 to prove our main lower bound for $m = 2$.

Theorem 4 *Let M be any truthful-in-expectation mechanism and let $m = 2$. Then, $ratio(M) \geq 3$.*

Proof For the proof, we will construct two instances I and I', both of which will be defined on the real line, meaning that the positions of the agents and the facilities will lie on the real line for both instances. Instance I will be the "base instance" on which we will lower-bound the approximation ratio of any mechanism M, and I' will be the "deviation instance" that we will use via arguments involving truthfulness to bound certain probabilities in the base instance. From Lemma 8, it suffices to lower-bound the approximation ratio of anonymous mechanisms only. In particular, we will use the corollary of anonymity that if all agents' locations coincide, then their probabilities of being assigned to any of the two facilities are also the same; instance I' will in fact be an instance where all of the agents' locations coincide.

Concretely, consider the following instances:

- The "base instance" $I = (A, F)$ will be an instance such that $d(F_2, F_1) = 2 + \epsilon$ for some $\epsilon > 0$ sufficiently small. Furthermore, for the positions of the agents, let $A_1 = \ldots = A_{n-1} = F_1$ and $A_n = F_1 + 1$, and for the capacities, let $c_1 = n - 1$ and $c_2 = 1$.
- The "deviation instance" $I' = (A', F)$ will be such that the locations of the facilities are the same as in I, for the positions of the agents we have $A' = (A_1, \ldots, A_{n-1}, A'_n)$ with $A'_n = F_1$, and for the capacities we have $c_1 = n-1$ and $c_2 = 1$, exactly as in I. In other words, I' is exactly the same as I except for the fact that the position of agent n is now $A'_n = F_1$ rather than $A_n = F_1 + 1$.

Let $p_n(I)$ and $p_n(I')$ be the probabilities that agent n is assigned to facility 2 on instance I and I' respectively. By the anonymity of M, since the positions of all agents in I' coincide, it holds that $p_n(I') = p_j(I')$ for any $j \in N$, where $p_j(I')$ is the probability that agent j is assigned to facility 2 on instance I', from which it follows immediately that $p_n(I') = 1/n$. Via the truthfulness of M, we will prove that $p_n(I) = 1/n$ as well.

To see this, consider the deviation A'_n of agent n on the instance I, resulting in instance I' (where the true position of the agent is in fact A_n, but the agent is feigning position A'_n). The cost of the agent after deviating is

$$p_n(I') \cdot (1 + \epsilon) + (1 - p_n(I')) \cdot 1 = \frac{1}{n} \cdot (1 + \epsilon) + \frac{n-1}{n},$$

whereas the agent's cost before deviating was $p_n(I) \cdot (1+\epsilon) + (1 - p_n(I)) \cdot 1$. Since M is truthful, the cost after deviating cannot be smaller than before, which only holds when $p_n(I) \leq 1/n$, by a simple calculation. Intuitively, agent n can ensure that she is treated exactly the same as all of the other agents by the mechanism, by pretending to have the same position as them.

Since $p_n(I) = 1/n$, we have that on instance I, the expected social cost of M is at most

$$\frac{1}{n} \cdot (1 + \epsilon) + \frac{n-1}{n} \cdot (3 + \epsilon).$$

This is because when agent n is not assigned to facility 2, which happens with probability at least $(n-1)/n$, some other agent i (with position $A_i = F_1$) is assigned to facility 2. In that case, the cost of agent n is 1 and the cost of agent i is $2 + \epsilon$, resulting in a social cost of $3 + \epsilon$. At the same time, the optimal social cost on instance I is $1 + \epsilon$. As $\epsilon \to 0$ and $n \to \infty$, the ratio goes to 3. □

6.2.2 Serial unilateral mechanisms

Back to the case of a general number m of facilities, we explore a seemingly promising idea that has been used for improved welfare guarantees in related but different contexts (e.g. see [28] or [30]), which is based on extending the greedy solutions to allow for more flexibility. For example, consider the following class of mechanisms, parameterized by a set of vectors q^i and the choice of the permutation of agents.

Mechanism[**Serial Unilateral Mechanisms**]. Fix an ordering of the agents (arbitrarily or uniformly at random). Whenever an agent i is selected according to the ordering, let $\mathcal{F}^i = \{F_1^i, \ldots, F_k^i\}$ be the set of facilities with non-zero residual capacities. Assign agent i to facility F_j^i with probability q_j^i, for $j = 1, \ldots, k$, according to some probability vector $(q_1^i, \ldots, q_k^i)$, such that $q_j^i \geq q_\ell^i$, whenever $d(A_i, F_j^i) \leq d(A_i, F_\ell^i)$.
In other words, every time that a serial unilateral mechanism selects an agent, that agent is assigned to each of the available facilities with probabilities such that it is not less likely to be assigned to facilities that are closer to the agent's most preferred position. Note that if we let j_i^* be the facility (among the available ones) that is closest to A_i for every agent i, then by setting $q_{j_i^*}^i = 1$ and $q_k^i = 0$ for all $k \neq j_i^*$, and by fixing the ordering of agents arbitrarily, we recover exactly SD. If we fix the ordering uniformly at random, we recover RSD.

For serial unilateral mechanisms to be relevant for our purposes, they have to be truthful. One way to guarantee that is to ensure that the probability vectors $(q_1^i, \ldots, q_k^i)$ (for each agent i) of the mechanism only depend on *ordinal* information. Formally, we will say that agent i *prefers* facility j over facility j', if $d(A_i F_j) \leq d(A_i, F_{j'})$, and we will denote it by $j \succeq_i j'$. From this, we can derive a *preference ordering* $(\succeq)_i$ for agent i. With this at hand, we will refer to a serial unilateral mechanism as *ordinal* if the probability vectors $(q_1^i, \ldots, q_k^i)$ are the same for any two instances I and I' in which the agents' costs induce the same preference orderings. Note that both SD and RSD are ordinal. From standard arguments from the literature on closely related settings (see [27, 28, 31]), it follows that ordinal serial unilateral mechanisms are truthful.

As a matter of fact, while ordinality is sufficient to guarantee that serial unilateral mechanisms are truthful, it is not necessary. It is possible to design *cardinal* serial unilateral mechanisms, in which the probabilities $(q_1^i, \ldots, q_k^i)$ can differ for two instances I and I' that induce the same preference orderings for all the agents. Intuitively, these probabilities depend not only on the preference orderings induced by the costs, but on the actual costs themselves. With this additional information, it seems conceivable that these mechanisms could achieve better approximation ratios. To this end, in a different but related setting, [28] showed that there are mechanisms of this type that outperform all ordinal ones. Ensuring truthfulness for these mechanisms is rather intricate, and we refer the reader to [27] and [28] for examples.

However, unlike environments like the ones studied in [29] or [28] where the objective is the maximization of utilities, our objective is the minimization of costs, which poses the following additional inherent complication. Consider an example where for every agent, her most preferred position A_i coincides with a different facility F_{j_i}, and there is enough capacity to assign every such agent to their corresponding facility. Then, this allocation should be outputted by the mechanism unequivocally (i.e., with probability 1), as otherwise the minimum social cost would be 0, the social cost of the mechanism would be positive, and the approximation ratio would be infinity. It is not hard to see that among the serial unilateral mechanisms described above, whether ordinal or cardinal, the only ones that satisfy this requirement are SD and RSD, and as a consequence, any other mechanism in this class is bound to fail in our setting.

6.3 Future work

From the discussion above, there are some clear open problems related to the facility assignment problem. The general question that we would like to answer seems challenging:

Given any augmentation factor $g \geq 1$, *what is the best truthful (deterministic or randomized) mechanism for the problem and what is its approximation ratio?*

Since our results for augmentation factors $g \geq 3$ are small constants, this question is most interesting for the case of doubling the capacities or applying no augmentation at all. A more immediate question is to prove a tight bound on the approximation ratio of RSD. A different approach could be to consider non-truthful mechanisms for the problem, and study their equilibrium inefficiency, in the same vein as [21].

More generally, our resource augmentation framework is applicable to other related problems in the literature of resource allocation and artificial intelligence; for example, the bi-criteria algorithms of [5] can be seen as instances of resource augmentation and the same holds for the auction setting of [23]. The framework can also be applied to broader settings where the loss in performance is due to restrictions other than truthfulness, such as fairness [20], stability [24] or ordinality [14, 29]; all the problems in those papers can be studied through the resource augmentation lens. It is not hard to imagine that similar notions like the *price of fairness* [20], could be redefined in terms of resource augmentation.

References

1. Abdulkadiroğlu, A., Sönmez, T.: Random serial dictatorship and the core from random endowments in house allocation problems. Econometrica **66**, 689–701 (1998)
2. Abed, F., Caragiannis, I., Voudouris, A.A.: Near-optimal asymmetric binary matrix partitions. Algorithmica **80**(1), 48–72 (2018)
3. Anshelevich, E., Das, S.: Matching, cardinal utility, and social welfare. ACM SIGECom Exch. **9**(1), 4 (2010)
4. Anshelevich, E., Postl, J.: Randomized social choice functions under metric preferences. J. Artif. Intell. Res. **58**, 797–827 (2017)
5. Anshelevich, E., Sekar, S.: Blind, greedy, and random: algorithms for matching and clustering using only ordinal information. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI), pp. 390–396 (2016)
6. Anshelevich, E., Sekar, S.: Truthful mechanisms for matching and clustering in an ordinal world. In: Proceedings of the 12th International Conference on Web and Internet Economics (WINE), pp. 265–278 (2016)

7. Anshelevich, E., Zhu, W.: Ordinal approximation for social choice, matching, and facility location problems given candidate positions. ACM Trans. Econ. Comput. **9**(2), 1–24 (2021)
8. Anshelevich, E., Bhardwaj, O., Elkind, E., Postl, J., Skowron, P.: Approximating optimal social choice under metric preferences. Artif. Intell. **264**, 27–51 (2018)
9. Anshelevich, E., Filos-Ratsikas, A., Shah, N., Voudouris, A.A.: Distortion in social choice problems: the first 15 years and beyond. In: Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI), pp. 4294–4301 (2021)
10. Anshelevich, E., Filos-Ratsikas, A., Voudouris, A.A.: The distortion of distributed metric social choice. Artif. Intell. **308**, 103713 (2022)
11. Antoniadis, A., Fischer, C., Tönnis, A.: A collection of lower bounds for online matching on the line. In: Proceedings of the 13th Latin American Symposium on Theoretical Informatics (LATIN), pp. 52–65 (2018)
12. Antoniadis, A., Barcelo, N., Nugent, M., Pruhs, K., Scquizzato, M.: A $o(n)$-competitive deterministic algorithm for online matching on a line. Algorithmica **81**(7), 2917–2933 (2019)
13. Athanassopoulos, S., Caragiannis, I., Kaklamanis, C.: Analysis of approximation algorithms for k-set cover using factor-revealing linear programs. Theory Comput. Syst. **45**(3), 555–576 (2009)
14. Aziz, H., Chen, J., Filos-Ratsikas, A., Mackenzie, S., Mattei, N.: Egalitarianism of random assignment mechanisms. In: Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 1267–1268 (2016)
15. Bansal, N., Buchbinder, N., Gupta, A., Naor, J.: A randomized $o(\log^2 k)$-competitive algorithm for metric bipartite matching. Algorithmica **2**, 390–403 (2014)
16. Bansal, N., Buchbinder, N., Gupta, A., Naor, J.S.: A randomized o (log 2 k)-competitive algorithm for metric bipartite matching. Algorithmica **68**(2), 390–403 (2014)
17. Bilò, V.: A unifying tool for bounding the quality of non-cooperative solutions in weighted congestion games. Theory Comput. Syst. **62**(5), 1288–1317 (2018)
18. Bogomolnaia, A., Moulin, H.: A new solution to the random assignment problem. J. Econ. Theory **100**, 295–328 (2001)
19. Caragiannis, I.: Wavelength management in WDM rings to maximize the number of connections. SIAM J. Discrete Math. **23**(2), 959–978 (2009)
20. Caragiannis, I., Kaklamanis, C., Kanellopoulos, P., Kyropoulou, M.: The efficiency of fair division. Theory Comput. Syst. **50**(4), 589–610 (2012)
21. Christodoulou, G., Filos-Ratsikas, A., Frederiksen, S.K., Goldberg, P.W., Zhang, J., Zhang, J.: Social welfare in one-sided matching mechanisms. In: Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 1297–1298 (2016)
22. Chung, C., Pruhs, K., Uthaisombut, P.: The online transportation problem: on the exponential boost of one extra server. In: Proceedings of the 8th Latin American Symposium on Theoretical Informatics (LATIN), pp. 228–239 (2008)
23. Eden, A., Feldman, M., Friedler, O., Talgam-Cohen, I., Weinberg, S.M.: The competition complexity of auctions: a Bulow-Klemperer result for multi-dimensional bidders. In: Proceedings of the 18th ACM Conference on Economics and Computation (EC), p. 343 (2017)
24. Emek, Y., Langner, T., Wattenhofer, R.: The price of matching with metric preferences. In: Proceedings of the 23rd Annual European Symposium on Algorithms (ESA), pp. 459–470 (2015)
25. Enelow, J.M., Hinich, M.J.: The Spatial Theory of Voting: An Introduction. Cambridge University Press, Cambridge (1984)
26. Feige, U., Jozeph, S.: Oblivious algorithms for the maximum directed cut problem. Algorithmica **71**(2), 409–428 (2015)
27. Feige, U., Tennenholtz, M.: Responsive lotteries. In: Proceedings of the 3rd International Symposium on Algorithmic Game Theory (SAGT), pp. 150–161 (2010)
28. Filos-Ratsikas, A., Miltersen, P.B.: Truthful approximations to range voting. In: Proceedings of the 10th International Conference on Web and Internet Economics (WINE), pp. 175–188 (2014)
29. Filos-Ratsikas, A., Frederiksen, S.K.S., Zhang, J.: Social welfare in one-sided matchings: random priority and beyond. In: Proceedings of the 7th International Symposium on Algorithmic Game Theory (SAGT), pp. 1–12 (2014)
30. Filos-Ratsikas, A., Frederiksen, S.K., Zhang, J.: Social welfare in one-sided matchings: random priority and beyond. arXiv preprint arXiv:1403.1508 (2014b)
31. Gibbard, A.: Manipulation of schemes that mix voting with chance. Econometrica **45**(3), 665–81 (1977)

32. Guo, M., Conitzer, V.: Strategy-proof allocation of multiple items between two agents without payments or priors. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 881–888 (2010)
33. Hylland, A., Zeckhauser, R.: The efficient allocation of individuals to positions. J. Polit. Econ. **87**(2), 293–314 (1979)
34. Jain, K., Mahdian, M., Markakis, E., Saberi, A., Vazirani, V.V.: Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. J. ACM **50**(6), 795–824 (2003)
35. Kalyanasundaram, B., Pruhs, K.: Online weighted matching. J. Algorithms **14**(3), 478–488 (1993)
36. Kalyanasundaram, B., Pruhs, K.: The online transportation problem. SIAM J. Discrete Math. **13**(3), 370–383 (2000)
37. Kalyanasundaram, B., Pruhs, K.: Speed is as powerful as clairvoyance. J. ACM **47**(4), 617–643 (2000)
38. Khuller, S., Mitchell, S.G., Vazirani, V.V.: On-line algorithms for weighted bipartite matching and stable marriages. Theor. Comput. Sci. **127**(2), 255–267 (1994)
39. Koutsoupias, E.: Weak adversaries for the k-server problem. In: Proceedings of the 40th Annual Symposium on Foundations of Computer Science (FOCS), pp. 444–449 (1999)
40. Koutsoupias, E., Nanavati, A.: The online matching problem on a line. In: Proceedings of the 1st International Workshop on Approximation and Online Algorithms (WAOA), pp. 179–191 (2003)
41. Krysta, P., Manlove, D.F., Rastegari, B., Zhang, J.: Size versus truthfulness in the house allocation problem. Algorithmica **81**(9), 3422–3463 (2019)
42. Kulkarni, J., Mirrokni, V.: Robust price of anarchy bounds via LP and Fenchel duality. In: Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1030–1049 (2015)
43. Mahdian, M., Yan, Q.: Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In: Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC), pp. 597–606 (2011)
44. Merrill, S., III., Merrill, S., Grofman, B.: A Unified Theory of Voting: Directional and Proximity Spatial Models. Cambridge University Press, Cambridge (1999)
45. Meyerson, A., Nanavati, A., Poplawski, L.: Randomized online algorithms for minimum metric bipartite matching. In: Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 954–959 (2006)
46. Nadav, U., Roughgarden, T.: The limits of smoothness: a primal-dual framework for price of anarchy bounds. In: Proceedings of the 6th International Workshop on Internet and Network Economics (WINE), pp. 319–326 (2010)
47. Nayyar, K., Raghvendra, S.: An input sensitive online algorithm for the metric bipartite matching problem. In: Proceedings of the 58th Annual Symposium on Foundations of Computer Science (FOCS), pp. 505–515 (2017)
48. Procaccia, A.D., Tennenholtz, M.: Approximate mechanism design without money. ACM Trans. Econ. Comput. **1**(4), 1–26 (2013)
49. Raghvendra, S.: A robust and optimal online algorithm for minimum metric bipartite matching. In: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM), vol. 18, pp. 1–16
50. Raghvendram, S.: Optimal analysis of an online algorithm for the bipartite matching problem on a line. In: Proceedings of the 34th International Symposium on Computational Geometry (SoCG), vol. 67, pp. 1–14 (2018)
51. Roughgarden, T.: Beyond worst-case analysis lecture #3: online paging and resource augmentation. https://theory.stanford.edu/~tim/f14/l/l3.pdf (2014)
52. Sleator, D.D., Tarjan, R.E.: Amortized efficiency of list update and paging rules. Commun. ACM **28**(2), 202–208 (1985)
53. Svensson, L.-G.: Strategy-proof allocation of indivisble goods. Soc. Choice Welf. **16**(4), 557–567 (1999)
54. Young, N.: The k-server dual and loose competitiveness for paging. Algorithmica **11**(6), 525–541 (1994)