Fair Allocations of Service Tasks with Two-Dimensional Costs

Hadi Hosseini, Mingming Li, Shivika Narang, Zhehan Yu

Abstract

Recent work has explored fair allocations of delivery tasks where delivery agents incur costs based on the distance they travel. We generalize this setting to one of service tasks where an agent's cost has two dimensions: the *time* spent completing each task and the *distance* traveled. Thus, the input includes a graph where all nodes other than the hub correspond to a unique order/task, and must be allocated to some agent. We model the cost incurred by an agent as a linear function of the edges traversed and the nodes they need to service. In this setting, we explore two well motivated fairness concepts: Envy-Freeness up to One Order (EF1) and Minimax Share (MMS). We show several surprising results, including the fact that in our setting an MMS allocation can be found in polynomial time. We also show conditions which can guarantee the existence of allocations that are both EF1 and MMS. We further show that these allocations can be found in polynomial time. We also provide tight upper bounds on the price of fairness. We complement our theoretical results with an experimental analysis demonstrating the effect of various input parameters on the MMS cost.

1 Introduction

Fair allocations have been widely explored by the EconCS community. Extensive work has gone into finding fair allocations for divisible items [Aziz and Mackenzie, 2016; Brams and Taylor, 1996; Steen, 1999; Segal-Halevi *et al.*, 2017; Barman and Kulkarni, 2023], indivisible items [Lipton *et al.*, 2004a; Budish, 2010; Aziz *et al.*, 2022b; Barman *et al.*, 2018; Barman and Krishnamurthy, 2020; Wei *et al.*, 2023], and their combinations [Bei *et al.*, 2021a,b; Li *et al.*, 2023; Liu *et al.*, 2024]. While various different types of valuation functions have been considered, most of the work in this space assumes that items are independent of each other, leading to additive valuations. The majority of the work on fair division does not effectively capture settings where the values/costs from the items are dictated by a strong joint structure.

Recently, Hosseini *et al.* [2024] introduced the model of fair distribution of orders – where each order lies on a unique node on a tree – among a set of delivery agents. Each agent is responsible for collecting items from the hub, traversing through each assigned order, and ultimately returning to the hub. Their model only considers costs incurred due to the distance traveled. As a result, in their setting, an agent traveling to a specific node can be assigned all the nodes on the way for no additional cost. While this may model delivery settings where minimal time is spent making the actual deliveries, there is a large variety of alternate settings this does not capture. For instance when delivering large electronics or furniture, a delivery agent will incur the costs from the distance traveled to reach the address, as well as the time taken to drop the item inside the houses/premises. As a result, in our setting we assume a more general model with both distance and time-based costs. This significantly increases the intricacy of the problem. Consequently, we first assume that the underlying graph is a path. Paths are an important building block for building more complicated models and have thus been previously explored within fair division [Misra *et al.*, 2021; Suksompong, 2019; Truszczynski and Lonc, 2020; Hosseini and Schierreich, 2025] as well as other social choice contexts such as seat assignment games Ceylan *et al.* [2023]; Aziz *et al.* [2024]; Berriaud *et al.* [2023], topological distance games [Bullinger and Suksompong, 2024; Deligkas *et al.*, 2024b] and facility location (see [Chan *et al.*, 2021] for a survey).

The assumption of two-dimensional costs is also well motivated beyond delivery tasks. Consider a central office in charge of the repair and maintenance of all buildings along a fixed portion of the road it is located on. Many university campuses and suburban neighborhoods follow similar models. Here, the repair technicians will not only incur the costs of the fuel and time spent to reach their assigned buildings, they also will incur a cost in terms of the time spent servicing each assigned building. Other real-life examples include: energy and electric providers that dispatch repair teams after an outage (possibly due to storm, tornado, or other natural disruptions). Internet/cell phone providers have similar issues for maintenance within neighboring grids. In all these cases, in order to achieve worker satisfaction, it is important to ensure that the task allocation is fair.

Fairness can be captured in multiple ways, with envy-freeness (EF) [Foley, 1967; Varian, 1974; Stromquist, 1980] and maximin share (MMS) [Budish, 2010; Barman and Krishnamurthy, 2020; Wei *et al.*, 2023] being two of the most prominent criteria. Given the demanding nature of envy-freeness for

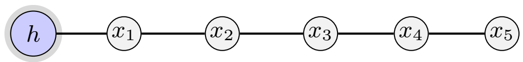

Figure 1: An example graph with the hub at h.

indivisible goods, we consider its relaxation, known as envy freeness up to one item (EF1) [Lipton *et al.*, 2004b; Budish, 2010; Caragiannis *et al.*, 2019].

Example 1. *To illustrate the challenges in ensuring fairness, consider a stylized example with two agents and five vertices illustrated in Figure 1. If costs were only incurred from the distance traveled (one-dimensional), it would be sufficient to have one agent service all orders, as they can drop off orders on nodes x_1 to x_4 on the way to x_5. In the two dimensional case, we can clearly minimize the maximum cost incurred by splitting the nodes among the two agents.*

Minimizing the maximum cost is not enough to ensure envy-freeness or even EF1. When costs are only from distance, any allocation where nodes x_4 and x_5 are assigned to different agents would be EF1, this includes an allocation where one agent is assigned nodes $\{x_1, x_2, x_3, x_4\}$ and the other is assigned only node $\{x_5\}$. When costs are only from time, this clearly would not be EF1, or even minimize maximum cost. In this case, any allocation that equitably divides the nodes would be both EF1 and MMS such as when one agent is assigned vertices $\{x_1, x_2\}$ and the other agent is assigned vertices $\{x_3, x_4, x_5\}$. Under two dimensional costs, where both costs are positive this allocation cannot be EF1 as the agent doing the larger set of vertices is also traveling further.

As shown in Example 1, the ideas and solutions for the one-dimensional case do not extend trivially to the two-dimensional case. While there may be cases where one dimension *dominates* the other: one incurs significantly higher cost, the cost in the other dimension may be non-zero and affect the structure of fair outcomes. For instance, when conducting repairs within a university campus, the repair tasks would likely take more time than the travel time, but for both MMS and EF1 agents being assigned larger bundles must have lower travel time. Consequently, we study arbitrary two-dimensional settings as well as cases where one dimension dominates.

1.1 Our Contributions

In this paper, we introduce a two-dimensional cost model for allocating service tasks and explore the existence of fair solutions for this setting. We are given a path graph, with one node marked as the hub and all remaining nodes corresponding to unique orders/service tasks which need to be allocated among the agents. An agent's cost is the sum of the distance based cost and the time based cost they incur. We pursue *Envy-freeness up to one item* (EF1) and *Minimax Costs* (MMS) as our fairness concepts. As part of our technical contributions, we present multiple polynomial time algorithms with strong guarantees on fairness, give tight upper bounds on the price of fairness and conduct a parameterized numerical analysis to build intuition on how various graph parameters affect agent costs under fair allocations.

Existence and Computation of Fair Allocations. We first discuss the existence of fair allocations in our setting in Section 3. These results are summarized in Table 1. EF1 allocations can be found by adapting the standard envy-graph algorithm of Lipton *et al.* [2004a].

		EF1	**MMS+PO**	**EF1+MMS**
General Costs	existence	✓	✓	?
	computation	P (Prop. 1)	P (Thm. 1)	?
Time - Dominant	existence	✓	✓	✓
	computation	$O(m)$	$O(m)$ (Thm. 2)	$O(m)$ (Thm. 2)
Distance - Dominant	existence	✓	✓	✓
	computation	$O(m)$	$O(m)$ (Thm. 3)	$O(m)$ (Thm. 3)

Table 1: The summary of our results on the existence and computation of fair allocations. ✓ denotes that the allocation always exists ? denotes that the problem remains open.

In our main result, we show that *an MMS and PO allocation* can be found in polynomial time given an arbitrary instance of our problem. This algorithm is non-trivial and makes use of several structural results on Pareto optimality which we prove. This comes in stark contrast to most standard settings, including that of [Hosseini *et al.*, 2024; Hosseini and Schierreich, 2025] where computing an MMS allocation is always NP-hard. Assuming that the underlying graph is a path is what makes the problem tractable, but still requires some non-trivial machinery to find the solution.

We also give additional results on the structure of fair allocations. For instances where one dimension of costs dominates the other, we present linear time algorithms for finding allocations that are *simultaneously* MMS, EF1 and Pareto Optimal. In fact, we show that when the time costs dominate the distance costs, *every* MMS allocation must satisfy EF1.

Price of Fairness. In Section 4 we discuss the price incurred by insisting on fairness. Informally, the price of fairness under a given instance $\mathcal{I}$, is the ratio of the minimum possible sum of agent costs under a fair allocation to the minimum possible sum of agent costs under any allocation. We give a tight upper bound on the price of EF1 for arbitrary instances. We then show that when one dimension of costs dominates, the price of EF1 reduces further. We show that in these settings, the price of MMS is even lower than the price of EF1.

Experimental Analysis. We complement our theoretical results by studying the MMS cost in a variety of instances in Section 5. In order to better understand the effect of parameters like the time and distance costs and the hub location on the MMS cost, we conduct an extensive experimental analysis where we vary these parameters for different graph sizes and differing number of agents.

1.2 Relevant Work

The fair division of indivisible items has been well studied. The fairness notions such as EF1 [Lipton *et al.*, 2004a; Budish, 2010] and MMS [Budish, 2010; Amanatidis *et al.*, 2017] have been widely explored for goods [Barman *et al.*, 2018; Chaudhury *et al.*, 2020; Bouveret *et al.*, 2016; Lipton *et al.*, 2004a], chores [Huang and Lu, 2021; Bhaskar *et al.*, 2021; Sun *et al.*, 2023] and their combinations [Aziz *et al.*, 2022a; Bhaskar *et al.*, 2021; Hosseini *et al.*, 2023b,a]. For both goods and chores, the price of fairness is quite important and has been well studied [Barman *et al.*, 2020; Bei *et al.*, 2021c; Li *et al.*, 2024; Caragiannis *et al.*, 2012]. We defer an extended cover of the relevant work to Appendix A.

While several papers have considered fair division on a graph, most of the models explored are unrelated to ours. The closest papers to ours are those of Hosseini *et al.* [2024] and Hosseini and Schierreich [2025]. Hosseini *et al.* [2024] introduce the problem of fairly allocating delivery orders, where each order corresponds to a unique node on a tree. They assume costs only come from distance, not time, and pursue EF1 and MMS alongside efficiency via PO. They show that finding MMS allocations, even without PO, proves to be hard, even for trees. They give an exponential time algorithm to find fair and efficient outcomes, which Hosseini and Schierreich [2025] show to be the best possible time-complexity for the case of trees. In contrast, we assume two-dimensional costs (time and distance) and assume the underlying graph to be a path, showing overwhelmingly positive results.

Hosseini and Schierreich [2025] extend the model studied in [Hosseini *et al.*, 2024] where they still assume costs only come from distance, but now different edges may have different weights. They also pursue MMS but relax the notion of Pareto optimality to *non-wastefulness* where each non-leaf node must only be serviced by an agent who is also servicing a descendant of it. Among various types of trees, they also study paths, and show that an MMS and non-wasteful allocation can be computed trivially. When costs come only from distances, every PO allocation must be non-wasteful. This is no longer true when one has time and distance based costs. Consequently, their approach does not extend to our model. Further, we prove the existence of MMS and PO which is stronger than non-wastefulness.

2 Model

We use $[k]$ to denote the set of positive integers up to k, that is $[k] = \{1, 2, \cdots, k\}$.

Let $M = [m]$ be the set of customers/orders distributed along a path, and $N = [n]$ the set of agents, with $h \in M$ representing the hub. Each vertex other than the hub h corresponds to a unique order. We assume that $m - 1 \geq n$. Each order must be assigned to one agent. We say that an allocation $A = (A_1, \cdots, A_n)$ is an n-partition of the set of orders. That is, A_i is the bundle assigned to agent $i \in N$ and we have that for any two distinct $i, j \in N$, $A_i \cap A_j = \varnothing$ and $\cup_{i \in N} A_i = M$.

Each agent starts from the hub, visits all the nodes in their assigned bundle A_i, and returns to the hub. We assume that the underlying graph is an unweighted path. That is the distance between any pair of adjacent nodes is the same. Let c^d denote the distance between two adjacent orders. Upon arriving at an order, the agent must spend time servicing this node, which incurs a cost of c^t. We assume that the time costs are identical for all nodes. Consequently, an instance of our problem can be represented by the tuple $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$.

We say that an agent i *services* node x when assigned bundle $S \subseteq M \setminus \{h\}$ if and only if $x \in S$. We say that i visits node x when assigned S if x lies on the path between the hub and some node in S. We now define $L(S) = \min_S x$ and $R(S) = \max_S x$, where $L(S)$ and $R(S)$ indicate the leftmost and rightmost nodes assigned under bundle S, respectively.

The total cost of servicing the bundle S is denoted by $c(S)$ and has two components. The first component is distance based cost incurred by the agent traveling from the hub to the assigned nodes and returning to the hub. Here, the agent must traverse each edge twice making the distance cost of S as $c^d(S) = 2c^d(\max(R(S), h) - \min(L(S), h))$. The second component accounts for the time based cost incurred to actually service each assigned node, given by $c^t(S) = c^t|S|$.

Definition 1 (Agent Costs)**.** *The cost of an agent for servicing a bundle $S \subseteq M \setminus \{h\}$ is denoted by $c(S) = c^d(S) + c^t(S) = 2c^d(\max(R(S), h) - \min(L(S), h)) + c^t|S|$.*

Thus, given two bundles of equal size, an agent would always prefer the one where they need to travel less. To this end, it would be ideal if the bundle only comprised of nodes adjacent to each other.

Definition 2 (Contiguous Bundles)**.** *We say that a bundle $S \subseteq M \setminus \{h\}$ is contiguous if for every node x s.t. $L(S) \leq x \leq R(S)$, either i) $x = h$ or ii) $x \in S$.*

That is, a bundle is contiguous if it forms a connected subgraph, with the inclusion of the hub if necessary. An empty bundle is vacuously contiguous. Analogously, a non-contiguous bundle is one where some node (other than the hub) between the leftmost and rightmost nodes in the bundle is not contained within it.

2.1 Fairness Notions.

One of the first definitions of fairness to be studied is Envy-Freeness (EF) where each agent weakly prefers their own assignment to that of any other. An EF allocation may not exist in many discrete settings, including that of ours. In this case, a prominent relaxation of EF is Envy-Freeness up to One Item (EF1) [Budish, 2010; Lipton *et al.*, 2004a], which requires that in an allocation, if agent i envies agent j, we can remove one node from agent i's bundle to eliminate the envy.

Definition 3 (Envy-Freeness up to One Order (EF1))**.** *An allocation A is EF1 if for every pair $i, j \in N$, either $A_i = \varnothing$ or there exits $x \in A_i$ such that $c(A_i \setminus \{x\}) \leq c_j(A_j)$.*

Another widely studied fairness notion is Minimax Share (MMS), which requires that the maximum cost incurred under an allocation be minimized across the set of all allocations. In order to define this, we first define the minimax share. Here we use Π^n to denote the set of all possible n-partitions of the nodes in $M \setminus \{h\}$.

Definition 4 (Minimax Share)**.** *Given an instance $\mathcal{I}$, the minimax share of agent i is $MMS(\mathcal{I}) = \min_{A \in \Pi^n} \max_{j \in N} c(A_j)$.*

An allocation A is MMS if for each $i \in N$, we have that $c(A_i) \leq MMS(\mathcal{I})$. That is, the cost of each bundle A_i is at most the Minimax Share cost.

In addition to fairness, economic efficiency is a very desirable and natural objective. We seek efficient outcomes via Pareto Optimality.

Definition 5 (Pareto Optimality (PO))**.** *An allocation A Pareto dominates $A^{'}$ if for all agent $i \in N$ $c(A_i) \leq c(A_i^{'})$, and there exists $j \in N$ such that $c(A_j) < c(A_j^{'})$. An allocation is Pareto optimal if it is not Pareto dominated by any other allocation.*

One-Dimension Dominant Costs. We find that when agent costs are such that one dimension is much more costly than the other, we are able to provide much stronger fairness guarantees. Such costs are well motivated and have been explored previously in more typical settings [Hosseini *et al.*, 2023b,a; Shao and Guo, 2024] under the guise of lexicographic valuations. Such costs are not one-dimensional as the less costly does influence fair allocations. Rather, they impose useful structures which we can exploit to find allocations with strong fairness guarantees particularly quickly.

Definition 6 (Time-Dominant Costs)**.** *An instance $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$ is said to have time-dominant costs if we have that $c^t > 2c^d(m-1)$.*

That is, an instance has time-dominant costs if an agent would rather go to either end of the path than service an additional node. We can similarly define distance-dominant costs:

Definition 7 (Distance-dominant Costs)**.** *An instance $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$ is said to have distance-dominant costs if we have that $2c^d > c^t(m-1)$.*

3 Fair Allocations

In this section, we discuss algorithms for finding fair allocations. We first present our results for two-dimensional costs. All omitted proofs from this section are deferred to Appendix B.

3.1 General Two-Dimensional Costs

We first consider the notion of envy-freeness up to One Item (EF1). EF1 requires that for any pair of agents, if one agent envies the allocation of another, this envy can be eliminated by removing a single item. Proposition 1 shows that, in a given instance, an EF1 allocation always exists and can be computed in polynomial time.

Proposition 1. *Given an instance $\mathcal{I}$, an EF1 allocation always exists and can be computed in polynomial time.*

MMS allocations. We now turn to fairness via MMS. Observe that in our model, all agents bear the same cost when assigned a fixed bundle $S \subseteq M \setminus \{h\}$. As a result, the MMS share value would be the same and be given by the same allocation. As a result, an MMS allocation must always exist. Further, an allocation that satisfies and MMS and PO must always exist, via a leximin optimal allocation.

In the model of Hosseini *et al.* [2024], the problem of finding an MMS allocation is NP-hard, even without requiring efficiency. However, in our main result, we show that we can find an MMS and PO allocation in our setup in polynomial time. To this end, we first make the following observation.

Lemma 1. *Given instance $\mathcal{I}$ and a Pareto optimal allocation $A = (A_1, \cdots, A_n)$ for each $i \in N$, we have that either:*

1. *A_i is contiguous or*
2. *$L(A_i) < h < R(A_i)$ and there exists $j \neq i$ such that $L(A_i) < L(A_j) \leq R(A_j) < R(A_i)$ and the bundles $\{x \in A_i | x < h\}$ and $\{x \in A_i | x > h\}$ are contiguous.*

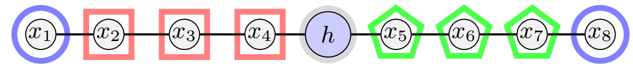

Figure 2: Instance in Example 2. An MMS allocation is depicted, the three bundles are shown by circles, squares and pentagons resp.

Whenever we have an allocation where there exist distinct $i, j \in N$ s.t. $L(A_i) < L(A_j) \leq R(A_j) < R(A_i)$ we say that A_j *surrounds* A_i. A pareto optimal allocation having non-contiguous bundles seems rather counter-intuitive. However, there are many cases where an MMS and PO allocation must have at least one non-contiguous bundle.

Example 2. *Consider an instance with $m = 9$ that is 8 orders and $n = 3$ agents with the hub at $h = 5$. This instance is illustrated in Figure 2. Let $c^d = 0.5$ and $c^t = 100$, that is, this is an instance of time-dominant costs. For this instance, for any allocation A, the agent with maximum cost will also be one with the largest sized bundle. As we have to allocate 8 nodes among 3 agents, at least one agent will have a bundle of size at least 3. The most equitable split of the nodes would be to have two agents service 3 nodes each and one agent service 2 nodes.*

Now, in order to minimize the maximum cost, we need to minimize the distance traveled by any agent. Thus, any MMS allocation must allocate the two furthest nodes to a single agent and the remaining nodes must be split into two contiguous bundles among the other two agents. It is easy to see that such an allocation is also PO. This allocation is also illustrated in Figure 2 where circles, squares and pentagons represent the three bundles.

In contrast, we now show that whenever we have only two agents, an MMS and PO allocation must be contiguous.

Proposition 2. *Given an instance $\mathcal{I}$ with $n = 2$, a Pareto optimal and MMS allocation must assign contiguous bundles to both agents.*

Armed with the structure on MMS and PO allocations, we can now present a polynomial time algorithm to find MMS and PO allocations using dynamic programming.

Theorem 1. *Given an instance $\mathcal{I}$, an MMS and Pareto Optimal allocation can be calculated in polynomial time.*

Proof Sketch. We develop an algorithm to minimize the maximum cost incurred. We only consider allocations where each bundle is either contiguous or surrounds another bundle. Recall that, given allocation A, A_i surrounds A_j if $L(A_i) < L(A_j) \leq R(A_j) < R(A_i)$. Intuitively, the algorithm proceeds by considering all possible ways of splitting the given instance into three pieces: a left piece where all bundles are contiguous, a central piece where bundles may be non-contiguous and a right piece where again all bundles will be contiguous. For each of the three pieces different combinations of agents are assigned and each piece is solved recursively.

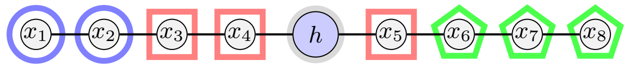

Figure 3: For the instance in Example 2 an EF1 but not MMS allocation is depicted. The three bundles are shown by circles, squares and pentagons.

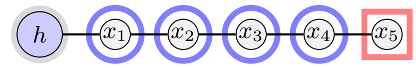

Figure 4: Under distance dominant costs with $n = 3$, an MMS and PO but not EF1 allocation is depicted. The two non-empty bundles are shown by circles and squares.

Specifically, we proceed by solving two dynamic programs: one for all agents receiving contiguous bundles, and the other where some agents may receive non-contiguous bundles. These dynamic programs require significant setup, consequently, we defer their statement to Appendix B. The correctness of these algorithms follows from our earlier results. From Proposition 2 and Lemma 1, we have certain cases where all agents must receive contiguous bundles: i) when the hub is at either end of the path or ii) when the number of agents is at most 2. As a result, we can first resolve the contiguous case, and use it to solve the non-contiguous case. □

3.2 Stronger Fairness and Efficiency Guarantees

We now show that when agents have time-dominant or distance-dominant costs, we can get both faster algorithms as well as stronger fairness guarantees.

Time-Dominant Costs. We first consider time-dominant costs where $c^t > (m-1)2c^d$. Here we find that an EF1 and PO allocation must always exist and can be found in linear time via an MMS and PO allocation via Algorithm 3.

Theorem 2. *Given an instance $\mathcal{I}$ with time dominant costs, we have i) every MMS allocation is also EF1 and ii) an allocation that satisfies MMS, EF1 and PO can be found in linear time.*

Unfortunately, every EF1 allocation need not be MMS under this setting. Figure 3 shows an EF1 allocation for this instance which is not MMS. We know that any MMS allocation must allocate the two furthest nodes to a single agent with no other nodes assigned.

Distance-dominant Costs. We now consider the case where distance costs dominate: $2c^d > (m-1)c^t$. In this case, MMS allocations need not all be EF1, but we can still find allocations that satisfy MMS and EF1 in linear time. We first show that the MMS share cost in this case can be determined in constant time.

Lemma 2. *Given an instance $\mathcal{I}$ with distance-dominant costs, we have that the MMS share cost is $MMS(\mathcal{I}) = 2c^d(\max(m-h,h) + c^t$ whenever $n > 3$.*

Proof. Under distance-dominant costs, the bundle containing the furthest node from the hub will always have maximum cost. It is easy to see that the furthest node from the hub must be one or both of the end points of the path (either 1 or m). The distances of these nodes from the hub are h and $m-h$ respectively.

Let $S \subseteq M \setminus \{h\}$ be a bundle containing at least one of the furthest nodes. As a result, $c(S) \geq 2c^d \max(m-h,h) + c^t(S) \geq 2c^d \max(m-h,h) + c^t$. That is, for any allocation A, we have that $\max_{i \in N} c(A_i) \geq 2c^d \max(h, m-h) + c^t$.

To minimize this cost, while still servicing the furthest node, we need to ensure that S only contains the furthest node. As a result, under an MMS allocation, a bundle containing the furthest node cannot contain any other node. In this case, the cost of this bundle will be exactly $2c^d(\max(m-h,h) + c^t$. Consequently, $MMS(\mathcal{I}) = 2c^d(\max(m-h,h) + c^t$. □

This leads us to the following corollaries over some significant MMS and PO allocations.

Corollary 1. *Given an instance $\mathcal{I}$ with distance-dominant costs, an MMS allocation with minimum sum of agent costs must be such that each edge on the path is traversed by at most two agents.*

Corollary 2. *Given an instance $\mathcal{I}$ with distance-dominant costs and $c^t > 0$, under a leximin optimal allocation the n furthest nodes from the hub must all be serviced by distinct agents. The agent servicing the closest of these nodes to the hub must also service all other nodes.*

Theorem 3. *Given an instance $\mathcal{I}$ with distance-dominant costs, an allocation that satisfies MMS and EF1 can be found in linear time.*

Through the proof of Theorem 3, we get the following corollary.

Corollary 3. *Whenever $n \leq 2$, under distance-dominant costs, every MMS allocation is EF1.*

When $n > 2$, MMS and EF1 do not always coincide. Figure 4 shows an example of an MMS and PO allocation for an instance with distance-dominant costs with $n = 3$ agents and $m = 6$. For any choice of cost parameters satisfying distance dominance, the allocation depicted will be MMS and PO but not EF1.

In the proof of Theorem 3, we also show that when $n \leq 2$, every MMS allocation will be PO by default. Further, whenever $c^t > 0$, the allocation returned by Algorithm 4 will be PO. As a result, we have the following corollary.

Corollary 4. *Given an instance $\mathcal{I}$ with distance-dominant costs and $c^t > 0$, an allocation that is MMS, EF1 and PO can be found in linear time.*

4 Price of Fairness

The *Price of Fairness* measures the efficiency loss incurred when imposing fairness constraints on allocations. In this subsection, we analyze the price of fairness for EF1 and MMS. We first formally define the price of fairness where we compare the minimum possible sum of agent costs under a fair allocation to the minimum possible sum of agent costs under any allocation. We defer all omitted proofs from this section to Appendix C.

Recall that we denote the space of all allocations among n agents as Π^n.

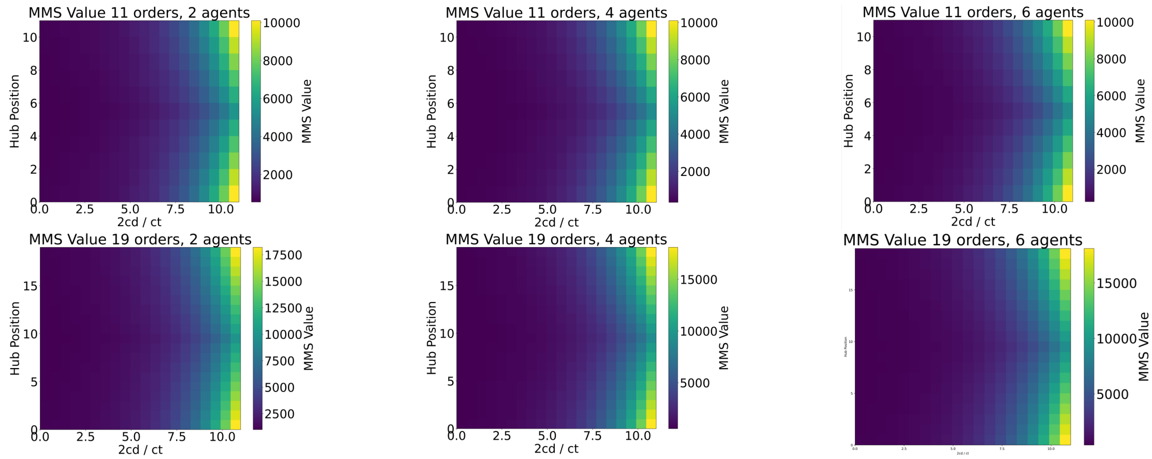

Figure 5: Experimental results on MMS cost varying with different hub locations and values of $\frac{2c^d}{c^t}$

Definition 8 (Price of Fairness)**.** *Given an instance $\mathcal{I}$ and a fairness criterion $\mathcal{F}$, the price of $\mathcal{F}$ under $\mathcal{I}$ is given as:*

$$\mathrm{PoF}(I) = \frac{\min_{A \in \prod^n, A \textit{ satisfies } \mathcal{F}} \sum_{i=1}^n c(A_i)}{\min_{A' \in \prod^n} \sum_{i=1}^n c(A'_i).}$$

First, we observe that the denominator is fixed for all instances. For this, recall that for a bundle $S \subseteq M \setminus \{h\}$, we have that $c^d(S) = 2c^d(\max(R(S), h) - \min(L(S), h))$, that is, $c^d(S)$ denote the cost from the distance traveled to service S. Analogously, $c^t(S)$ denotes the time cost from bundle S.

Observation 1. *Given instance $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$, for any allocation A, $\sum_{i \in N} c^t(A_i) = (m-1)c^t$. Consequently, the minimum possible sum of costs is minimized based on the minimum sum of distances.*

Observation 2. *Given instance $\mathcal{I}$, as each node must be serviced, we can have only one agent service all nodes and hence,* $\min_{A \in \prod^n} \sum_{i=1}^n c(A_i) = (m-1)(c^t + 2c^d)$.

4.1 Price of EF1

Theorem 4. *Given an instance $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$, we have that $PoEF1(\mathcal{I}) \leq \frac{(m-1)c^t + (2m-n-1)nc^d}{(m-1)c^t + 2(m-1)c^d}$.*

Observe that, Theorem 4 shows that PoEF1 increases as the ratio of $2c^d/c^t$ increases. As a result, it would be maximum for distance-dominant costs and minimum for time-dominant costs. Thus, applying Theorems 2 and 3 to Theorem 4 we get the following result.

Proposition 3. *Given an instance $\mathcal{I}$, we have that*

1. *If $\mathcal{I}$ has distance-dominant costs, $PoEF1 \leq \frac{(2m-n-1)n+2}{2(m-1)}$*
2. *If $\mathcal{I}$ has time-dominant costs, $PoEF1 \leq \frac{m+n}{m-1}$*

An interesting observation here is that if the smaller dimension vanishes, the impact on the price of EF1 is very different.

Corollary 5. *Given an instance $\mathcal{I}$, we have that i) if $c^d = 0$, $PoEF1(\mathcal{I}) = 1$ and alternately ii) if $c^t = 0$, $PoEF1 \leq \frac{(2m-n-1)n}{2(m-1)}$.*

4.2 Price of MMS

The upper bound on the Price of EF1 over arbitrary instances actually also applies to MMS, but it is not a tight upper bound. We unfortunately do not know of a tight upper bound on arbitrary instances. We now give upper bounds on Price of MMS under one-dimension dominant costs. In these cases, the price of MMS is quite low.

Proposition 4. *Given an instance $\mathcal{I}$, we have that*

1. *Under time-dominant costs, $PoMMS(\mathcal{I}) \leq \frac{m}{m-1}$,*
2. *Under distance-dominant costs $PoMMS(\mathcal{I}) \leq 2$ and*
3. *Whenever either $c^d = 0$ or $c^t = 0$, $PoMMS(\mathcal{I}) = 1$.*

5 Experimental Analysis

In this section, we present the results of our experiments studying the effect of various parameters on MMS share cost. Specifically, we compare the MMS value under a variety of hub locations, values of $\frac{2c^d}{c^t}$, number of agents and number of nodes/orders. We present some of the results here and defer the rest to Appendix D.

Experimental Methods. We create instances where the number of agents is such that $2 \leq n \leq 6$ and $m \in \{11, 15, 19\}$. For each combination of m and n, we allow the hub to be on any of the positions $\{1, 2, \cdots, m\}$. We fix the value c^t to be 100 and vary choose c^d to be $2c^t\tau$. Here, the coefficient τ – a critical parameter that balances inspection time and transportation costs – is logarithmically sampled (sampled values are uniformly spaced on the logarithmic scale) over 20 values that span $[0.1, 10]$.

Thus our instances span distance-dominant instances ($\tau \ll 1$) to time-dominant ones ($\tau \gg 1$). For each instance generated, we compute the MMS share cost using our algorithm described in Theorem 1. As a result, we can compare the effect of three parameters on the MMS cost: hub position, ratio of the costs $\frac{2c^d}{c^t}$ and the number of orders/nodes. We defer the results for $n = 3$ and $n = 5$ and $m = 15$ to Appendix D along with a more detailed discussion. We present the remaining results here.

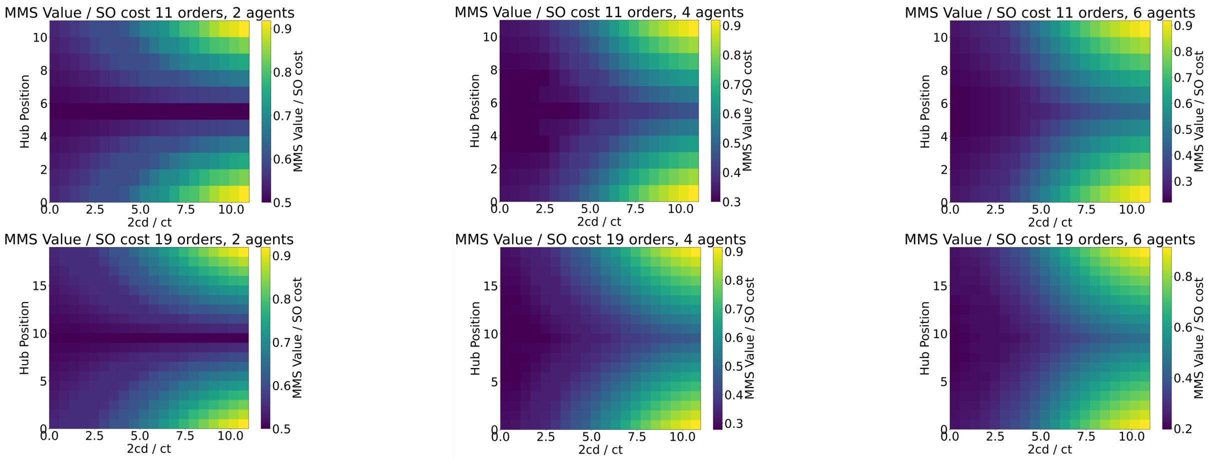

Figure 6: Experimental results on normalized MMS cost varying with different hub locations and values of $2c^d/c^t$

5.1 MMS Share

We first observe the effect on the MMS value of two parameters: the hub position and the ratio of $\frac{2c^d}{c^t}$. We present our results in Figure 5. We first observe that the range of the MMS share costs varies based on the number of orders, but is similar when we vary the number of agents. Keeping the value of $\frac{2c^d}{c^t}$ fixed the MMS share cost seems to be minimized when the hub is at the center. We find that as the hub moves further from the middle, the MMS share cost increases. Further, this increase is symmetric on either side. However, this effect is less visible for instances with a larger number of orders.

Keeping the hub position fixed, we find that initially there is noticeable little effect on the MMS share costs and these seem to increase only after $\frac{2c^d}{c^t} > 6.0$. The increase is also sharper, further away from the hub. Further, for larger number of orders, the increase in the MMS share cost is less visible for the same value of $\frac{2c^d}{c^t}$. This makes it hard to see anything interesting for the smaller values of $\frac{2c^d}{c^t}$.

Observe that the heatmaps for a fixed number of orders look very similar even if we change the number of agents. That is, we observe similar relative tradeoffs of hub position and $\frac{2c^d}{c^t}$ for all the chosen values of n. However, on closer inspection, fixing m, hub position and $\frac{2c^d}{c^t}$, as the number of agents increases, the MMS value decreases.

5.2 Normalized MMS Share

In order to better observe the effect of varying $\frac{2c^d}{c^t}$, we study the effect of $\frac{2c^d}{c^t}$ and the hub position on the ratio of MMS share cost and the socially optimal cost. We present these results in Figure 6 and we can observe some marked differences from the results in Figure 5.

Firstly, the range of the ratio of the MMS share cost to the socially optimal cost remains the same for all three choices of the number of orders. Further for the two extreme hub positions, the values seem to be the same for 11 and 19 orders. Interestingly, this is not the case when the hub is at the center. When the hub is at the center, the MMS share cost is smaller for the same ratio of $\frac{2c^d}{c^t}$ when there are more orders.

For a fixed hub position, as $\frac{2c^d}{c^t}$ increases, ratio of the MMS share cost to SO cost increases much more evenly, and there is a noticeable increase even for the smaller values of $\frac{2c^d}{c^t}$. And in this experiment, for the same τ, the changes brought about by different hub positions are more obvious than in the previous experiment. Further, for the extreme hub positions, the MMS share cost is smaller.

It is important to note that for this experiment, we do see noticeable differences in the heatmaps generated as we change the number of agents for a fixed number of orders. This is particularly noticeable when the hub is in the center. In fact, on closer inspection, we find that as we increase the number of agents, the ratio of the MMS share to the SO cost decreases.

6 Concluding Remarks

In this paper, we present a generalized model of allocating tasks to agents, where agents incur two dimensions of costs: time and distance. We study this under the assumption that each order/task lies on a unique node on a path. For this setting, we show that fair allocations can be found in polynomial time, even MMS allocations, which are typically intractable to find. We then show that when one dimension of cost is significant more costly, allocations that are both EF1 and MMS can be found in time linear in the number of nodes.

We then provide a tight upper bound on EF1 allocations and give smaller upper bounds for one-dimension dominant costs for the price of EF1 and MMS. We then complement our theoretical results by showing the effect of various parameters on the MMS share costs.

Introducing a new model, we open may avenues for future work, including finding EF1 and PO allocations whenever they exist and tight bounds on the price of MMS. Further, for paths, weighted graphs where different nodes and edges have different costs need to be studied. Another direction would be to study more complicated graphs like cycles and trees. Other fairness notions like proportionality and equitability can also be pursued here.

References

Mahyar Afshinmehr, Alireza Danaei, Mehrafarin Kazemi, Kurt Mehlhorn, and Nidhi Rathi. Efx allocations and orientations on bipartite multi-graphs: A complete picture. *arXiv preprint arXiv:2410.17002*, 2024.

Georgios Amanatidis, Evangelos Markakis, Afshin Nikzad, and Amin Saberi. Approximation algorithms for computing maximin share allocations. *ACM Transactions on Algorithms (TALG)*, 13(4):1–28, 2017.

Haris Aziz and Simon Mackenzie. A discrete and bounded envy-free cake cutting protocol for any number of agents. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 416–427. IEEE, 2016.

Haris Aziz, Ioannis Caragiannis, Ayumi Igarashi, and Toby Walsh. Fair allocation of indivisible goods and chores. *Autonomous Agents and Multi-Agent Systems*, 36(1):1–21, 2022.

Haris Aziz, Bo Li, Herve Moulin, and Xiaowei Wu. Algorithmic fair allocation of indivisible items: A survey and new questions. *arXiv preprint arXiv:2202.08713*, 2022.

Haris Aziz, Grzegorz Lisowski, Mashbat Suzuki, and Jeremy Vollen. Neighborhood stability in assignments on graphs. *arXiv preprint arXiv:2407.05240*, 2024.

Siddharth Barman and Sanath Kumar Krishnamurthy. Approximation algorithms for maximin fair division. *ACM Transactions on Economics and Computation*, 8(1):1–28, 2020.

Siddharth Barman and Pooja Kulkarni. Approximation algorithms for envy-free cake division with connected pieces. In *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261, page 16. Schloss Dagstuhl–Leibniz-Zentrum f {\" u} r Informatik, 2023.

Siddharth Barman, Arpita Biswas, Sanath Kumar Krishnamurthy, and Yadati Narahari. Groupwise maximin fair allocation of indivisible goods. *AAAI Conference on Artificial Intelligence*, pages 9921–9922, 2017.

Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In *Proceedings of the 2018 ACM Conference on Economics and Computation (EC-18)*, pages 557–574, 2018.

Siddharth Barman, Umang Bhaskar, and Nisarg Shah. Optimal bounds on the price of fairness for indivisible goods. In *Proceedings of the 16th International Conference on Web and Internet Economics (WINE)*, pages 356–369. Springer, 2020.

Siddharth Barman, Anand Krishna, Pooja Kulkarni, and Shivika Narang. Sublinear approximation algorithm for nash social welfare with xos valuations. *arXiv preprint arXiv:2110.00767*, 2021.

Siddharth Barman, Vishnu Narayan, and Paritosh Verma. Fair chore division under binary supermodular costs. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 2863–2865, 2023.

Xiaohui Bei, Zihao Li, Jinyan Liu, Shengxin Liu, and Xinhang Lu. Fair division of mixed divisible and indivisible goods. *Artificial Intelligence*, 293:103436, 2021.

Xiaohui Bei, Shengxin Liu, Xinhang Lu, and Hongao Wang. Maximin fairness with mixed divisible and indivisible goods. *Autonomous Agents and Multi-Agent Systems*, 35(2):34, 2021.

Xiaohui Bei, Xinhang Lu, Pasin Manurangsi, and Warut Suksompong. The price of fairness for indivisible goods. *Theory of Computing Systems*, 65(7):1069–1093, 2021.

Nawal Benabbou, Mithun Chakraborty, Ayumi Igarashi, and Yair Zick. Finding fair and efficient allocations when valuations don't add up. *Symposium on Algorithmic Game Theory*, pages 32–46, 2020.

Kristóf Bérczi, Erika R Bérczi-Kovács, Endre Boros, Fekadu Tolessa Gedefa, Naoyuki Kamiyama, Telikepalli Kavitha, Yusuke Kobayashi, and Kazuhisa Makino. Envy-free relaxations for goods, chores, and mixed items. *Theoretical Computer Science*, 1002:114596, 2024.

Damien Berriaud, Andrei Constantinescu, and Roger Wattenhofer. Stable dinner party seating arrangements. In *International Conference on Web and Internet Economics*, pages 3–20. Springer, 2023.

Umang Bhaskar and Yeshwant Pandit. Efx allocations on some multi-graph classes. *arXiv preprint arXiv:2412.06513*, 2024.

Umang Bhaskar, AR Sricharan, and Rohit Vaish. On approximate envy-freeness for indivisible chores and mixed resources. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

Umang Bhaskar, Neeldhara Misra, Aditi Sethia, and Rohit Vaish. The price of equity with binary valuations and few agent types. In *Proceedings of the 16th International Symposium on Algorithmic Game Theory (SAGT)*, pages 271–289. Springer, 2023.

Vittorio Bilò, Ioannis Caragiannis, Michele Flammini, Ayumi Igarashi, Gianpiero Monaco, Dominik Peters, Cosimo Vinci, and William S Zwicker. Almost envy-free allocations with connected bundles. *Games and Economic Behavior*, 131:197–221, 2022.

Sylvain Bouveret and Jérôme Lang. Efficiency and envy-freeness in fair division of indivisible goods: logical representation and complexity. *J. Artif. Int. Res.*, 32(1):525–564, jun 2008.

Sylvain Bouveret, Yann Chevaleyre, Nicolas Maudet, and Hervé Moulin. Fair allocation of indivisible goods., 2016.

Sylvain Bouveret, Katarína Cechlárová, Edith Elkind, Ayumi Igarashi, and Dominik Peters. Fair division of a graph. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

Steven J Brams and Alan D Taylor. *Fair Division: From cake-cutting to dispute resolution*. Cambridge University Press, 1996.

Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. In *Proceedings of the Behavioral and Quantitative Game Theory: Conference on Future Directions*, BQGT '10, New York, NY, USA, 2010. Association for Computing Machinery.

Martin Bullinger and Warut Suksompong. Topological distance games. *Theoretical Computer Science*, 981:114238, 2024.

Ioannis Caragiannis and Shivika Narang. Repeatedly matching items to agents fairly and efficiently. In *Proceedings of the 16th International Symposium on Algorithmic Game Theory (SAGT)*, pages 347–364. Springer, 2023.

Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. *Theory of Computing Systems*, 50(4):589–610, 2012.

Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum Nash welfare. *ACM Transactions on Economics and Computation*, 7(3):1–32, 2019.

Esra Ceylan, Jiehua Chen, and Sanjukta Roy. Optimal seat arrangement: what are the hard and easy cases? In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 2563–2571, 2023.

Hau Chan, Aris Filos-Ratsikas, Bo Li, Minming Li, and Chenhao Wang. Mechanism design for facility location problems: a survey. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. Efx exists for three agents. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 1–19, 2020.

George Christodoulou, Amos Fiat, Elias Koutsoupias, and Alkmini Sgouritsa. Fair allocation in graphs. In *Proceedings of the 24th ACM Conference on Economics and Computation*, pages 473–488, 2023.

Argyrios Deligkas, Eduard Eiben, Tiger-Lily Goldsmith, and Viktoriia Korchemna. Ef1 and efx orientations. *arXiv preprint arXiv:2409.13616*, 2024.

Argyrios Deligkas, Eduard Eiben, Dušan Knop, and Šimon Schierreich. Individual rationality in topological distance games is surprisingly hard. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 2782–2790, 2024.

DC Foley. Resource Allocation and the Public Sector. *Yale Economic Essays*, 7(1):73–76, 1967.

Jugal Garg, Pooja Kulkarni, and Rucha Kulkarni. Approximating nash social welfare under submodular valuations through (un) matchings. *fourteenth annual ACM-SIAM symposium on discrete algorithms*, pages 2673–2687, 2020.

Anjali Gupta, Rahul Yadav, Ashish Nair, Abhijnan Chakraborty, Sayan Ranu, and Amitabha Bagchi. Fairfoody: Bringing in fairness in food delivery. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pages 11900–11907, 2022.

Hadi Hosseini and Šimon Schierreich. The algorithmic landscape of fair and efficient distribution of delivery orders in the gig economy. *arXiv preprint arXiv:2503.16002*, 2025.

Hadi Hosseini and Andrew Searns. Guaranteeing maximin shares: Some agents left behind. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 238–244, 2021.

Hadi Hosseini, Aghaheybat Mammadov, and Tomasz Wąs. Fairly allocating goods and (terrible) chores. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 2738–2746, 2023.

Hadi Hosseini, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Fairly dividing mixtures of goods and chores under lexicographic preferences. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 152–160, 2023.

Hadi Hosseini, Shivika Narang, and Tomasz Wąs. Fair distribution of delivery orders. In *33rd International Joint Conference on Artificial Intelligence, IJCAI 2024*, pages 2825–2833. International Joint Conferences on Artificial Intelligence, 2024.

Xin Huang and Pinyan Lu. An algorithmic framework for approximating maximin share allocation of chores. In *Proceedings of the 22nd ACM Conference on Economics and Computation (EC)*, pages 630–631, 2021.

Wenzheng Li and Jan Vondrák. Estimating the nash social welfare for coverage and other submodular valuations. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1119–1130. SIAM, 2021.

Zihao Li, Shengxin Liu, Xinhang Lu, and Biaoshuai Tao. Truthful fair mechanisms for allocating mixed divisible and indivisible goods. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 2808–2816, 2023.

Zihao Li, Shengxin Liu, Xinhang Lu, Biaoshuai Tao, and Yichen Tao. A complete landscape for the price of envy-freeness. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 1183–1191, 2024.

R. J. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, EC '04, page 125–131, New York, NY, USA, 2004. Association for Computing Machinery.

Richard J Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. *Electronic commerce*, pages 125–131, 2004.

Shengxin Liu, Xinhang Lu, Mashbat Suzuki, and Toby Walsh. Mixed fair division: A survey. *Journal of Artificial Intelligence Research*, 80:1373–1406, 2024.

Neeldhara Misra and Aditi Sethia. Envy-free and efficient allocations for graphical valuations. In *International Con-*

ference on Algorithmic Decision Theory, pages 258–272. Springer, 2024.

Neeldhara Misra, Chinmay Sonar, PR Vaidyanathan, and Rohit Vaish. Equitable division of a path. *arXiv preprint arXiv:2101.09794*, 2021.

Ashish Nair, Rahul Yadav, Anjali Gupta, Abhijnan Chakraborty, Sayan Ranu, and Amitabha Bagchi. Gigs with guarantees: Achieving fair wage for food delivery workers. *arXiv preprint arXiv:2205.03530*, 2022.

Ariel D Procaccia and Junxing Wang. Fair enough: Guaranteeing approximate maximin shares. In *Proceedings of the fifteenth ACM conference on Economics and computation (EC)*, pages 675–692, 2014.

Erel Segal-Halevi, Shmuel Nitzan, Avinatan Hassidim, and Yonatan Aumann. Fair and square: Cake-cutting in two dimensions. *Journal of Mathematical Economics*, 70:1–28, 2017.

Kuncheng Shao and Hao Guo. Complete bipartite graph division under weakly lexicographic preferences. In *International Conference on Parallel and Distributed Computing: Applications and Technologies*, pages 327–338. Springer, 2024.

Daman Deep Singh, Syamantak Das, and Abhijnan Chakraborty. Fairassign: Stochastically fair driver assignment in gig delivery platforms. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 753–763, 2023.

Lynn Arthur Steen. Cake-cutting algorithms: Be fair if you can. *The American Mathematical Monthly*, 106(3):283, 1999.

Walter Stromquist. How to Cut a Cake Fairly. *The American Mathematical Monthly*, 87(8):640–644, 1980.

Warut Suksompong. Fairly allocating contiguous blocks of indivisible items. *Discrete Applied Mathematics*, 260:227–236, 2019.

Ankang Sun, Bo Chen, and Xuan Vinh Doan. Fairness criteria for allocating indivisible chores: connections and efficiencies. *Autonomous Agents and Multi-Agent Systems*, 37(2):39, 2023.

Miroslaw Truszczynski and Zbigniew Lonc. Maximin share allocations on cycles. *Journal of Artificial Intelligence Research*, 69:613–655, 2020.

Hal R Varian. Equity, Envy, and Efficiency. *Journal of economic theory*, 9(1):63–91, 1974.

Vignesh Viswanathan and Yair Zick. Weighted notions of fairness with binary supermodular chores. *arXiv preprint arXiv:2303.06212*, 2023.

Tianze Wei, Bo Li, and Minming Li. Maximin-aware allocations of indivisible chores with symmetric and asymmetric agents. In *32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*, pages 2897–2905. International Joint Conferences on Artificial Intelligence, 2023.

Jinghan A Zeng and Ruta Mehta. On the structure of envy-free orientations on graphs. *arXiv preprint arXiv:2404.13527*, 2024.

Yu Zhou, Tianze Wei, Minming Li, and Bo Li. A complete landscape of efx allocations on graphs: goods, chores and mixed manna. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 3049–3056, 2024.

A Additional Related Work

Fair division. The fair division of indivisible goods has been a subject of intense inquiry in various academic disciplines, including economics, computer science, and operations research. Key concepts that have emerged in this area of study include envy freeness up to one item (EF1) and max-min share (MMS), both of which have been thoroughly explored alone [Budish, 2010; Lipton *et al.*, 2004a; Bouveret *et al.*, 2016; Procaccia and Wang, 2014] and alongside efficiency [Barman *et al.*, 2018; Caragiannis and Narang, 2023; Bouveret and Lang, 2008]. Fairness has been explored for goods [Barman *et al.*, 2018; Chaudhury *et al.*, 2020; Bouveret *et al.*, 2016; Lipton *et al.*, 2004a], chores [Huang and Lu, 2021; Bhaskar *et al.*, 2021; Sun *et al.*, 2023] and their combinations [Aziz *et al.*, 2022a; Bhaskar *et al.*, 2021; Hosseini *et al.*, 2023b,a]. Likewise a variety of preference types have been explored including additive valuations/costs [Barman *et al.*, 2018, 2017; Aziz *et al.*, 2022a; Hosseini and Searns, 2021] and submodular or subadditive valuations/costs [Garg *et al.*, 2020; Barman *et al.*, 2021; Benabbou *et al.*, 2020; Li and Vondrák, 2021]. Recently even supermodular and superadditive costs [Barman *et al.*, 2023; Viswanathan and Zick, 2023] along with non-monotone valuations [Bhaskar *et al.*, 2023; Bérczi *et al.*, 2024] have been explored. Major results in this field have been summarized in a survey paper Aziz *et al.* [2022b].

Price of Fairness. The reason so much work has pursued fair and efficient allocations is to minimize the efficiency loss from fairness. Consequently, this has also been separately studied as the price of fairness, beginning with the work of Caragiannis *et al.* [2012], which explored divisible settings. The first paper to pursue the price of fairness under indivisible settings was by Barman *et al.* [2020]. Since then much work has gone into finding tight upper bounds on the price of fairness as a function of the input parameters for additive as well as subadditive settings [Bei *et al.*, 2021c; Li *et al.*, 2024; Bhaskar *et al.*, 2023; Sun *et al.*, 2023].

Graph-based settings. Fair division on graphs has been studied previously [Bouveret *et al.*, 2017; Bilò *et al.*, 2022; Truszczynski and Lonc, 2020; Misra *et al.*, 2021] where items are the nodes on a graph and each agent must receive a bundle that forms a connected subgraph. Recent work, initiated by Christodoulou *et al.* [2023] introduces *graphical valuations* where agents are nodes on a graph and the edges are the items [Zhou *et al.*, 2024; Bhaskar and Pandit, 2024; Zeng and Mehta, 2024; Misra and Sethia, 2024; Deligkas *et al.*, 2024a; Afshinmehr *et al.*, 2024; Zhou *et al.*, 2024]. An incident edge can only be assigned to one of its two endpoints. It is easy to see that there is little overlap between this setting and the one pursued in this paper. The model we study is very different from such settings, as while our items are indeed nodes, firstly there is no requirement that each agent receive a connected bundle and secondly, costs come from both the edges as well as the nodes. Further, in our setup, there is a unique node called the hub that must not be allocated, but informs the costs.

Fairness in Delivery Settings. Hosseini *et al.* [2024] introduced the problem of fair distribution of delivery tasks, where given a tree rooted at the *hub* all other nodes correspond to a delivery/task and must be allocated to some agent. The agent costs come from the distance they travel to start from the hub, service all their assigned nodes and then return to the hub. For this model they characterize the space of instances where fair and efficient allocations exist and give an exponential time algorithm to find them. All combinations of fairness and efficiency that they study prove to be NP-hard to check and find. Specifically, Hosseini *et al.* [2024] give an XP algorithm to find the Pareto frontier and this paired with their characterization results enables us to find fair and efficient allocations

Hosseini and Schierreich [2025] follow up on this model and show that in fact no FPT algorithm can exist to find fair and efficient allocations, making an XP algorithm the best possible. They further relax efficiency to non-wastefulness which they define as no agent should service a non-leaf node when they don't also service a descendant of it. They pursue MMS and non-wasteful allocations for arbitrary trees, as well as special types of trees including stars and paths. In their case, an MMS and non-wasteful allocation can be computed in a straightforward manner. Unfortunately, these results do not extend to our model, as they assume one-dimensional costs. Further, in our case a non-wasteful and MMS allocation need not exist in the first place. An argument can even be made that non-wastefulness, in the way they define it, is only meaningful for the case when costs only come from distances. Further, we give positive results for MMS and PO, which is stronger than non-wastefulness, even in their setting.

Some other recent work has also explored fairness in delivery settings [Gupta *et al.*, 2022; Nair *et al.*, 2022; Singh *et al.*, 2023]. Here the approach towards fairness is to ensure equal income distribution. This work is largely empirical however, and does not provide any significant theoretical guarantees. Further the costs incurred by agents from the delivery tasks tare independent of each other. In contrast, the costs in our case are interdependent, as an agent has lower marginal cost for nodes on the way to other nodes already assigned to them.

B Omitted Material From Section 3

We now present all omitted proofs on the existence and computation of fair allocations, in the order they appear in the paper.

Proposition 1. *Given an instance $\mathcal{I}$, an EF1 allocation always exists and can be computed in polynomial time.*

Proof. Given an arbitrary instance $\mathcal{I}$, observe that for any $S \subseteq S' \subseteq M \setminus \{h\}$ we have that the costs satisfy $c(S) \leq c(S')$. That is, our cost functions are monotone increasing. Consequently, we can use the envy-graph approach from Lipton *et al.* [2004a].

Here, starting from an empty allocation, we choose an agent with minimum cost and give them an unassigned node in $M \setminus \{h\}$. We do this repeatedly till all the nodes in $M \setminus \{h\}$ are assigned.

This procedure maintains an EF1 allocation throughout. The initial empty allocation is envy-free and thus EF1. In each iteration, as cost functions are identical, there always exists at least one agent with minimum cost. This agent will not envy any other agent. If the additional node increases their cost, EF1 will still be maintained, as removing this item will make the cost less than or equal to the cost from all other bundles. □

B.1 MMS allocations

We now turn to MMS allocations. We shall show that an MMS and PO allocation can always be found in polynomial time. To this end, we first prove an important necessary condition for Pareto optimal allocations.

Lemma 1. *Given instance $\mathcal{I}$ and a Pareto optimal allocation $A = (A_1, \cdots, A_n)$ for each $i \in N$, we have that either:*

1. *A_i is contiguous or*
2. *$L(A_i) < h < R(A_i)$ and there exists $j \neq i$ such that $L(A_i) < L(A_j) \leq R(A_j) < R(A_i)$ and the bundles $\{x \in A_i | x < h\}$ and $\{x \in A_i | x > h\}$ are contiguous.*

Proof. Given an instance $\mathcal{I} =< M, N, h, c^d, c^t >$, for an allocation A, we shall show that whenever an allocation doesn't satisfy conditions (1) or (2) there must exist a Pareto dominating allocation. Recall that A' pareto dominates A, if for all $i \in N$, $c(A_i) \geq c(A'_i)$ and for at least one $j \in N$, $c(A_j) > c(A'_j)$. To this end, consider an allocation A where there exists A_i which is non-contiguous.

Case 1: $L(A_i) \leq R(A_i) < h$ or $h < L(A_i) \leq R(A_i)$. Without loss of generality assume that i) $L(A_i) \leq R(A_i) < h$ and ii) i is the left most agent with a non-contiguous bundle. That is, all bundles containing a node $x < L(A_i)$ are contiguous. As a result, there must exist at least one agent i' s.t. $L(A_i) < L(A_{i'}) < R(A_i)$.

Consider alternate allocation A' obtained from swapping $L(A_{i'})$ for $R(A_i)$. That is, $A'_i = (A_i \setminus \{R(A_i)\}) \cup L(A_{i'})$, $A'_{i'} = (A_{i'} \setminus L(A'_i) \cup R(A_i)$ and for all $j \in N \setminus \{i, i'\}$ set $A'_j = A_j$.

Observe that under A', for each $j \in N \setminus \{i, i'\}$, $c(A_j) = c(A'_j)$. Further, $|A_i| = |A'_i|$ and $|A_{i'}| = |A'_{i'}|$, and thus, the time costs are the same. With respect to distance, we have that $c^d(A_i) = c^d(A'_i)$. For agent i', we now have that $L(A'_{i'}) > L(A_{i'})$ and thus, $c^d(A'_{i'}) < c_d(A_{i'})$. As a result, A' Pareto dominates A.

Case 2: Let there exist $i, i' \in N$ s.t. $L(A_i) < L(A_{i'}) < h < R(A_i) < R(A_{i'})$. Consider alternate allocation A' obtained from swapping $L(A_{i'})$ for $R(A_i)$. That is, $A'_i = (A_i \setminus \{R(A_i)\}) \cup L(A_{i'})$, $A'_{i'} = (A_{i'} \setminus L(A'_i) \cup R(A_i)$ and for all $j \in N \setminus \{i, i'\}$ set $A'_j = A_j$.

We again have that for all $j \in N \setminus \{i, i'\}$, $c(A_j) = c(A'_j)$. Meanwhile, $|A_i| = |A'_i|$ and $|A_{i'}| = |A'_{i'}|$, but $R(A'_i) < R(A_i)$ and $L(A'_{i'}) > L(A_{i'})$. Consequently, both i and i' travel less far and as a result, $c^d(A'_i) < c_d(A_i)$ and $c^d(A'_{i'}) < c_d(A_{i'})$. Thus A' Pareto dominates A.

Hence, under any Pareto optimal allocation A, for every $i \in N$, either A_i is contiguous or $L(A_i) < h < R(A_i)$ and there exists $i' \in N$ s.t. $L(A_i) < L(A_{i'}) \leq R(A_{i'}) < R(A_i)$.

Analogous to Case 1, for any non-contiguous bundle A_i, under a PO allocation, the bundles $\{x \in A_i | x < h\}$ and $\{x \in A_i | x > h\}$ are contiguous. □

Proposition 2. *Given an instance $\mathcal{I}$ with $n = 2$, a Pareto optimal and MMS allocation must assign contiguous bundles to both agents.*

Proof. Given an instance $\mathcal{I}$ with 2 agents, consider an arbitrary allocation A. We shall show that if even one bundle under A is non-contiguous, A is either not PO or not MMS. Let $L(A_1) < L(A_2)$.

Case 1: First consider the case where neither of the two bundles is contiguous under A. For this setting, if $L(A_1) < L(A_2) < R(A_1) < R(A_2)$, we have from Lemma 1 that A is not PO. The only remaining possibility for both bundles to be non-contiguous is when $L(A_1) < L(A_2) < R(A_2) < R(A_1)$ and there exists $x \neq h$ s.t. $L(A_2) < x < R(A_2)$ and $x \in A_1$. In this case, we can swap all such x for the furthest nodes from h in A_2 to get a Pareto dominating allocation.

Case 2: The remaining case is when exactly one bundle is non-contiguous. Without loss of generality, let A_1 be non-contiguous and A_2 be contiguous. As a consequence of Lemma 1, it must be that $L(A_1) < L(A_2) \leq R(A_2) < R(A_1)$. In this case, we have that $c^d(A_1) > c^d(A_2)$. Now consider the allocation A' where A'_2 contains the leftmost $|A_2|$ nodes and A'_1 contains the rightmost $|A_1|$ nodes.

Case 2a: $|A_1| \geq |A_2|$. In this case, we have that $c(A_1) > c(A_2)$. Now, we have that in terms of bundle size, $|A_1| = |A'_1|$ and $|A_2| = |A'_2|$. In terms of distance, both $c^d(A'_1)$ and $c^d(A'_2)$ are strictly less than $c^d(A_1)$. As a result, A' has lower maximum cost than A. Thus, A cannot be MMS.

Case 2b: $|A_1| < |A_2|$. In this case, A_2 contains at least half the nodes. Without loss of generality, assume that the hub is in the left half of the path, i.e., $2h < m + 1$. The bundle sizes are identical in both allocations. Observe that here, $L(A'_2) < h < R(A'_2)$. Consequently, $c^d(A'_2) \leq c^d(A_2)$ whereas, $c^d(A'_1) < c^d(A_1)$. As a result, A' Pareto dominates A. □

Hence, we now have that when $n = 2$, given an allocation A where even one bundle is non-contiguous, there always exists an alternate allocation A' s.t. either $\max_i c(A_i) > \max_i c(A'_i)$ or $\max_i c(A_i) = \max_i c(A'_i)$ and $\min_i c(A_i) > \min_i c(A'_i)$. We say that A' leximin dominates A. It is easy to see that an allocation that is not leximin dominated by any other allocation must be MMS under our model. We shall now show that we can find such an allocation in poly time.

Theorem 1. *Given an instance $\mathcal{I}$, an MMS and Pareto Optimal allocation can be calculated in polynomial time.*

Proof. We shall now show how to find a specific type of MMS and PO allocation: a leximin optimal allocation. To this end, let us first define some useful notation. **Setup.** Given an allocation A, we can *sort it in non-increasing cost order* to obtain allocation $B = sort(A)$ such that $c(B_1) \geq c(B_2) \geq \cdots \geq c(B_n)$ and $B_i = A_{\pi(i)}$ for every agent $i \in [n]$ and some permutation of agents π.

Definition 9 (Leximin Optimality)**.** *An allocation A leximin dominates an allocation A' if there is agent $i \in [n]$ such that $c(B_i) < c(B'_i)$ and $c(B_j) = c(B'_j)$ for every $j \in [i-1]$, where $B = sort(A)$ and $B' = sort(A')$. An allocation is*

ALGORITHM 1: CTG

Input: $n, left, right, h, c^d, c^t$
Output: A leximin optimal cost vector for the instance

- **if** $left > right$ **then**
  - **Return** $\{2c^d m^2 + c^t m^2\}$ ▷ Invalid instance
- **if** $left = right$ **then**
  - **if** $left = H = right$ **then**
    - Let A be a n length vector where all entries are 0 ▷ No nodes to allocate
  - **else**
    - Let A be a n length vector where $A_1 = c^t + 2c^d \max(H - left, right - H)$ and all other entries are 0 ▷ Only one node to allocate
  - **Return** A
- ▷ Here, $left < right$
- **if** $n = 1$ **then**
  - Initialize $s \leftarrow right - left$ and $k \leftarrow right - left$
  - **if** $h < left$ **then**
    - $k \leftarrow right - h$
    - $s \leftarrow s + 1$
  - **if** $h > right$ **then**
    - $k \leftarrow h - left$
    - $s \leftarrow s + 1$
  - **Return** $(2c^d k + c^t s)$
- **else**
  - **Return** $\text{Lmin}_{l' > left}\, CTG(m-1, l', right, h, c^d, c^t)$ $\bigoplus 2c^d \max(h - left, l' - h) + c^t(l' - left + \mathbb{I}(H > l'$ OR $h < left))$

ALGORITHM 2: NTG

Input: $n, left, right, h, c^d, c^t$
Output: A leximin optimal cost vector for the instance

- **if** $n = 0$ **then**
  - **if** $left \leq right \neq h$ **then**
    - **Return** $\{2c^d m^2 + c^t m^2\}$ ▷ Invalid instance
  - **else**
    - **Return** $\varnothing$ ▷ Instance with no agents and no nodes
- **if** $left > right$ **then**
  - **Return** $\{2c^d m^2 + c^t m^2\}$ ▷ Invalid instance
- **if** $left = right$ *OR* $h \leq left$ *OR* $h \geq right$ *OR* $n \leq 2$ **then**
  - **Return** $CTG(n, left, right, h, c^d, c^t)$ ▷ Optimal allocation must be contiguous
- ▷ Here, $left < h < right$
- ▷ Build leximin opt over contiguous solutions
- $\mathcal{L}_C \leftarrow CTG(n, left, right, h, c^d, c^t)$
- ▷ Build leximin opt over non-contiguous solutions
- $\mathcal{L}_N \leftarrow \text{Lmin}_{l', r' : left \leq l' < h < r' \leq right}\, \mathcal{I}_1$
- where $\mathcal{I}_1 \leftarrow \text{Lmin}_{n_L + n' + n_R = n-1 : n_L > 0 \Leftrightarrow l' > left, n_R > 0 \Leftrightarrow r' < right}\, I_2$
- where $\mathcal{I}_2 \leftarrow CTG(n_L, left, l'-1, h, c^d, c^t) \bigoplus \mathcal{I}_3 \bigoplus$
- $CTG(n_r, r'+1, right, h, c^d, c^t)$
- where $\mathcal{I}_3 \leftarrow \text{Lmin}_{l'', r'' : l' < l'' \leq r'' \leq r'}\, NTG\,(n', l'', r'', h, c^d, c^t) \bigoplus (2c^d((h - l') + (r' - h)) + c^t((r' - r'') + (l'' - l')))$
- **Return** $\text{Lmin}(\mathcal{L}_C, \mathcal{L}_N)$

leximin optimal if it is not leximin dominated by any other allocation.

Observe that given two allocations A and A', unless they give identical costs, one must leximin dominate the other. Between A and A' we can decide in linear time, which allocation leximin dominates the other. With regards to our goal of finding MMS and PO allocations, it is straightforward to see that every leximin optimal allocation must be MMS. Further, if A pareto dominates A', it implies that A also leximin dominates A' but the reverse may not be true. As a result, every leximin optimal allocation is PO but not vice versa. Consequently, every leximin optimal allocation must satisfy the conditions in Lemma 1.

Specifically, under a leximin optimal allocation A^*, for each $i \in N$, either A_i^* is contiguous or it spans either side of the hub and surrounds at least one other bundle. Consequently, given any two bundles, A_i^* and A_j^* where $L(A_i^*) < L(A_j^*)$, we have that either $R(A_i^*) < L(A_j^*)$ or if A_i is non-contiguous, $L(A_i^*) < L(A_j^*) \leq R(A_j^*) < R(A_i^*)$. As every non-contiguous bundle must surround the hub, given two non-contiguous bundles, A_i^* and A_j^* where $L(A_i^*) < L(A_j^*)$ it must be that $L(A_i^*) < L(A_j^*) \leq R(A_j^*) < R(A_i^*)$. Let there be at least one non-contiguous bundle under A^*. Choose i s.t. A_i^* is the non-contiguous bundle with the largest distance-cost. Thus, all bundles with nodes to the left or right of $L(A_i^*)$ and $R(A_i^*)$, respectively, must be contiguous.

Contiguous Algorithm Overview. We can now show how to find a cost vector of an MMS and PO allocation via the leximin optimal. Through this procedure, an MMS and PO allocation can also be found. Firstly, we show that when restricted to the space of allocations where all bundles are contiguous, we can find a leximin optimal allocation (over contiguous allocations) via dynamic programming. Informally, given instance $\mathcal{I}$, we consider all possible lengths for the leftmost bundle, conditioned on which, consider all possible lengths for the second leftmost bundle and so on. We give this algorithm in Algorithm 1. Before discussing the algorithm details, we first setup some operators:

- Lmin: This operator, takes as input multiple allocations/sorted cost vectors, and returns one leximin optimal among this set.
- $\bigoplus$: This operator takes as input two cost vectors and merges them into a single sorted cost vector. For instance cost vectors: $(10, 5, 2)$ and $(8, 5, 3, 2)$ would be merged as $(10, 5, 2) \bigoplus (8, 5, 3, 2) = (10, 8, 5, 5, 3, 2)$.
- $\mathbb{I}$: This operator is an indicator (variable) and takes as input a logical statement/event and returns a value of 1 if the logical statement is true and 0 if it is false.

We are now prepared to describe our dynamic program for allocations with contiguous bundles. The CTG dynamic program takes as input n agents, a sub-path (of the path on M) between the nodes $left$ and $right$, a hub location and the cost parameters c^d and c^t. CTG then returns the leximin optimal solution for dividing this sub-path among the n agents. In order to do this, there are some base cases to consider:

- Invalid instance $left > right$: in this case, we have stumbled into an invalid instance, and thus we return a high cost which is more than what could be attained under any instance.
- $left = right$: in this case at most one node needs to be allocated which can be done trivially.
- $n = 1$: in this case, only one agent is there so they must service all nodes.

If none of these cases occur, that is there are multiple nodes, to be allocated amongst multiple agents, we consider the leximin optimal of the allocations given by different lengths of the leftmost contiguous bundle and solve the remaining instance among the remaining $n-1$ agents. As a result, CTG can find the leximin optimal solution among allocations with all contiguous bundles.
Running time: For each agent, at most m different bundles are considered and finding the leximin optimal among a set of cost vectors is done in linear time. As a result, CTG terminates in time $O(mn^2)$.

We now show how to find a leximin optimal over all allocations by developing a dynamic program that considers non-contiguous bundles as well.

Non-contiguous Algorithm Overview. Recall that under a leximin optimal allocation, given two non-contiguous bundles, one must surround the other. As a result, given an instance $\mathcal{I}$, we can break it into three parts: left contiguous, center non-contiguous and right contiguous, where the hub lies in the center. That is, given a path, we can consider all possible non-contiguous bundles by starting with a non-contiguous bundle with largest distance cost, and solving the sub paths to the right and left of it as contiguous solutions. To this end there are some base cases to consider:

- No agent and no nodes: In this case we return an empty set showing no cost.
- Invalid instance: here either there are no agents, or $left > right$. In either case we return a high cost that could not be attained under *any* allocation.
- Only contiguous solutions: this can happen is the hub is to the side of the sub-path or if $n \leq 2$. In this case, we use the CTG dynamic program as a subroutine.

For any other instance, where $n \geq 3$ and $left < h < right$, we consider all contiguous solutions via CTG. For non-contiguous solutions, we first split the given sub-path into three pieces $left$ to $l'-1$, l' to r' and $r'+1$ to $right$ where $left \leq l' < h < r' \leq right$. For a fixed choice of l' and r', we now consider how to split the agents across these sub-instances. To this end, we choose $n_L + n' + n_R = n-1$ where $n_L > 0$ if and only if $l' > left$ and $n_R > 0$ if and only if $r' < right$. We then solve the left and right sub-instances via CTG.

For the central sub-instance, we know from Lemma 1 that for a non-contiguous bundle, the nodes on one side of the hub should still be contiguous. As a result, we consider l'' and r'', such that one agent service the nodes from l' to $l''-1$ and those from $r''+1$ to r', and then use NTG for the remaining n' agents and the sub-path between l'' and r''. We take the leximin optimal over all such allocations and compare it with the leximin optimal over all contiguous solutions, and ultimately find the leximin optimal.
Running Time: For each of our agent variables n_L, n' and n_R we have $O(n)$ choices and for the path variables we have at most $O(m)$ choices for each. Merging two cost vectors and finding the leximin can be done in linear time and as a result NTG runs in polynomial time.

□

ALGORITHM 3: Fairness under Time-Dominant Costs

Input: $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$ with time-dominant costs
Output: An MMS, EF1 and PO allocation A

Let $k \leftarrow (m-1) \mod n$ and $p \leftarrow \lfloor \frac{m-1}{n} \rfloor$
Choose $(l, r) \in \arg\min\{|h-l| + |r-h| : 1 \leq l \leq r \leq m-1 \text{ and } r-h = k(p+1)\}$
Let $l' \leftarrow l$
for $i = 1$ *to* k **do**
  $A_i \leftarrow \{l', \cdots, l' + (p+1)\}$ ▷ Allocate larger bundles
  $l' \leftarrow l' + p + 2$
  if $h \in A_i$ **then**
    $A_i \leftarrow (A_i \setminus \{h\}) \cup l'$
    $l' \leftarrow l' + 1$
$l' \leftarrow 1$
for $i = k+1$ *to* n **do**
  if $l' \leq l \leq l' + p$ **then**
    $A_i \leftarrow \{l', \cdots, l-1\} \cup \{r+1, \cdots r+p-(l-l')\}$ ▷ Non-contiguous bundle
    $l' \leftarrow r + p - (l - l') + 1$
  else
    $A_i \leftarrow \{l', \cdots, l'+p\}$
    $l' \leftarrow l' + p + 1$

B.2 Stronger Guarantees for Fairness and Efficiency

We now turn to the case where one dimension of cost dominates the other. For such instances, we can find MMS and EF1 allocations in linear time.

Time-Dominant Costs. We shall constructively prove that under time-dominant costs, EF1, MMS and PO allocations must exist. In particular, Algorithm 3 finds one in linear time.

Theorem 2. *Given an instance $\mathcal{I}$ with time dominant costs, we have i) every MMS allocation is also EF1 and ii) an allocation that satisfies MMS, EF1 and PO can be found in linear time.*

Proof. Firstly, observe that under any (complete) allocation, there will always be one bundle which contains at least $\lceil \frac{m-1}{n} \rceil$ nodes. Under time-dominant costs, we have that the bundle with the largest cost will be the one that contains the most nodes and among those, goes the furthest.

As a result, under an MMS allocation, we need to minimize the number of nodes in each bundle. As a result every bundle under an MMS allocation must have size either $\lfloor \frac{m-1}{n} \rfloor$ or $\lceil \frac{m-1}{n} \rceil$. Further all bundles with $\lceil \frac{m-1}{n} \rceil$ nodes will travel less far than any bundle with $\lfloor \frac{m-1}{n} \rfloor$ nodes. Under time-dominant costs, this is sufficient to ensure EF1.

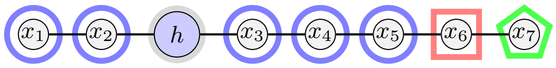

Figure 7: Outcome of Algorithm 4 on an instance with $n = 3$. An MMS, PO and EF1 allocation is depicted. The three bundles are shown by circles, squares and pentagons.

Given an MMS allocation A, fix an arbitrary $i, j \in [n]$. If $|A_i| > |A_j|$, as A satisfies MMS, it must be that $c^d(A_i) \leq c^d(A_j)$. As a result, removing *any* node from A_i will ensure equal time cost and thus EF1 is satisfied. Now if $|A_i| \leq |A_j|$ then removing any node from A_i will ensure that A_i contains strictly fewer nodes than A_j and as the costs are time-dominant, this is ensures EF1.

Consequently, under time-dominant costs, *every* MMS allocation is EF1.

Now we show that we can build an MMS+EF1+PO allocation in linear time. Let $k = (m-1) \mod n$ that is, $k \in [n-1] \cup \{0\}$ s.t. there exists some $p \in \mathbb{Z}^+$ s.t. $m - 1 = pn + k$. Thus, we have that under an MMS allocation, k agents receive a bundle of size $p + 1 = \lceil \frac{m-1}{n} \rceil$ and the remaining $n - k$ agents receive a bundle of size $p = \lfloor \frac{m-1}{n} \rfloor$.

We need to first minimize the distance traveled under any bundle with $|A_i| = p + 1$. To this end, we can select the $(n-k)p$ nodes furthest from the hub. The remaining nodes must form a connected sub-path of the original graph along with the hub. These remaining $k(p+1)$ nodes can be divided into k contiguous bundles. This is needed to ensure MMS.

To ensure PO, the $(n-k)p$ furthest nodes also need to be divided in a way that minimizes the sum of the distances traveled. To this end, at most one agent must receive a non-contiguous bundle of the closest p nodes to the hub from this set and all other $n - k - 1$ agents can receive contiguous bundles. Observe that if this is satisfied, in order to reduce the cost of any bundle, we need to either increase the number of nodes in a different bundle or increase the distance traveled. Consequently, this approach is sufficient to ensure PO.

Algorithm 3 does precisely this. It first identifies the $k(p+1)$ closest nodes to the hub and allocates them as contiguous bundles. The remaining $(n-k)p$ nodes are allocated, with at most one non-contiguous bundle, with consisting of the closest nodes to the hub from this set. It is straightforward to see that algorithm runs in time linear in the number of orders. □

Distance-dominant Costs. We shall now show that we can always construct an allocation that is EF1 and MMS under distance-dominant costs.

Theorem 3. *Given an instance $\mathcal{I}$ with distance-dominant costs, an allocation that satisfies MMS and EF1 can be found in linear time.*

Proof. We shall prove this result in two parts: a single algorithm for when $n \geq 3$ and a more tedious algorithm for the case of $n \leq 2$.

Linear time approach via Algorithm 4. We shall show that Algorithm 4 returns an MMS and EF1 allocation whenever $n > 2$. To this end, observe that l and r are chosen such that the nodes in $\{1, \cdots l-1\} \cup \{r+1, m\}$ are the n furthest nodes from h. As a result, it must be that whenever $1 < l \leq h \leq r < m$, it must be that $|(h-l) - (r-h)| \leq 1$, otherwise, the node further from h among l and r would violate the fact that $\{1, \cdots l-1\} \cup \{r+1, m\}$ are the n furthest nodes from h.

ALGORITHM 4: Fairness under Distance-Dominant Costs

```
Input: 𝓘 = ⟨M, N, h, c^d, c^t⟩ with distance-dominant costs
Output: An MMS and EF1 allocation A
Choose (l, r) ∈ arg min{|h − l| + |r − h| : 1 ≤ l ≤
    r ≤ m − 1 and r − h = m − n}
if exist x, x' s.t. 1 ≤ x < l ≤ h ≤ r < x' then
    for i = 1 to l − 2 do
        A_i ← {i}
    A_{l−1} ← {l − 1, ··· h − 1}
    A_l ← {h + 1, ··· , r + 1}
    for i = l + 1 to n do
        A_i ← {r + i + 1 − l}
else
        ▷ The furthest nodes from h are all on one side
    if l = 1 then
        for i = 1 to n − 1 do
            A_i ← {m − i + 1}
        A_n ← {1, ··· , m − n + 1} \ {h}
    else
        for i = 1 to n − 1 do
            A_i ← i
        A_n ← {n, ··· , m} \ {h}
```

Now, observe that the agents servicing the furthest node(s) in the path always service a single node. As a result, the allocation returned by Algorithm 4 must be MMS from Lemma 2. It only remains to prove that this allocation is EF1.

Let A be the allocation returned. For each i s.t. $|A_i| = 1$, EF1 is satisfied trivially. Now consider i s.t. $|A_i| > 1$. This agent does not envy any agent who services a node further than it. Whenever $l = 1$ or $r = m$, all agents other than i travel further and thus EF1 is satisfied.

Let $1 < l \leq h \leq r < m$. Without loss of generality, assume that $(h-l) \geq (r-h)$ (the other case follows analogously). In this case, we have that $A_{l-1} = \{l-1, \cdots, h-1\}$, $A_l = \{h+1, \cdots, r+1\}$ and $A_{l+1} = r + 2$. We know that $h - l \leq (r-h) + 1$. Thus, $c^d(A_{l-1}) \leq c^d(A_{l+1})$. Thus, agent $l - 1$ does not envy agents $l + 2, \cdots, n$ and agent l does not envy anyone. Finally, for agent $l - 1$, removing the node $l - 1$ would make their cost less than that of agent l and as a result EF1 is satisfied.

Thus, whenever $n > 2$, Algorithm 4 returns an MMS and EF1 allocation. It is straightforward to see that Algorithm 4 proceeds in linear time.

Remaining Cases. When $n = 1$, all allocations are trivially MMS and EF1. For $n = 2$, if $|h - (m-h)| > 0$ that is there is a unique furthest node, in order to achieve MMS, one agent must service the furthest node and the other agent services all remaining nodes. This must is EF1 as the agent servicing the furthest node has greater cost than the other agent, but on removing this node, the cost becomes 0. Finally the hub is in the center, that is, $h = m - h$, in order to be MMS, one agent must service the nodes $\{1, \cdots, h-1\}$ and the other service the nodes $\{h+1, \cdots, m\}$. The cost of both bundles is equal

Figure 8: Under distance-dominant costs, a minimum cost EF1 allocation is depicted as a tight example for Theorem 4. The four bundles are shown by circles, squares, pentagons and hexagons.

and as a result, an MMS allocation here would be EF1. □

C Omitted Material from Section 4

We now turn to our results on Price of Fairness. We first prove results on the price of EF1.

C.1 Price of EF1

We first consider arbitrary instances. Here, we prove an upper bound for all allocations and then show that there is a class of EF1 allocations for which this is tight.

Theorem 4. *Given an instance $\mathcal{I} = \langle M, N, h, c^d, c^t \rangle$, we have that $PoEF1(\mathcal{I}) \leq \frac{(m-1)c^t+(2m-n-1)nc^d}{(m-1)c^t+2(m-1)c^d}$.*

Proof. Given an instance $\mathcal{I}$, we know from Observation 1 that we need to maximize the distance traveled under a minimum cost EF1 allocation in order to find a tight upper bound on it. As each node must be serviced by a unique agent, exactly one agent visits the furthest node. As a result, in order to maximize the sum of distances, the second furthest node must be serviced by a different agent.

Consider an allocation where A_i consists only of the $i^t h$ furthest node from the hub from $i \in n-1$ and A_n contains the remaining $m-n$ nodes. Observe that, in order to maximize the distance traveled under such an allocation, we need to ensure that $h \in \{1, m\}$. Without loss of generality, we assume that $h = 1$.

Consequently, we consider the allocation where for $i < n$, $A_i = \{m - i\}$ and $A_n = \{1, \cdots, m-n\}$. Consider any alternate allocation A' s.t. $c^d(A'_1) \geq c^d(A'_2) \geq \cdots c^d(A'_n)$. We can see inductively that $c^d(A_1) = c^d(A'_1)$ and for all $i > 1$, we have that $c^d(A_i) \geq c^d(A'_i)$. Thus, from Observation 1, we have that $\sum_{i \in N} c(A_i) \geq \sum_{i \in N} c(A'_i)$.

Specifically, the sum of the costs is $\sum_{i \in N} c(A_i) = c^t(m-1) + \sum_{i \in N} c^d(A_i)$. In particular,

$$\begin{aligned}\sum_{i \in N} c^d(A_i) &= \sum_{i \in N} 2c^d(m-i) \\ &= 2c^d \left(\sum_{i=1}^{n} m - \sum_{i=1}^{n} i \right) \\ &= 2c^d \left(nm - \frac{n(n+1)}{2} \right) \\ &= 2c^d n \left(\frac{2m-n+1}{2} \right) \\ &= c^d n(2m-n+1).\end{aligned}$$

As a result, we have that $PoEF1(\mathcal{I}) \leq \frac{c^t(m-1)+c^d n(2m-n+1)}{(m-1)(c^t+2c^d)}$. In order to show that this bound is indeed tight, we shall show that there an allocation like A can indeed be the minimum cost EF1 allocation.

We find that whenever $4c^d \geq (m-n-2)c^t$, this allocation will be EF1. Observe that under this allocation, for $i < n$ $c(A_i) = c^t + 2(m-i)c^d$ and $c(A_n) = (m-n)(2c^d + c^t)$. For $i < n$, $|A_i| = 1$ and upon removing this node, all envy is eliminated. That is, $c(A_i \setminus \{m-i\}) = 0 \leq c(A_j)$ for all $j \in N$.

The only remaining case is that of A_n. It suffices to show that A_n satisfies EF1 with A_{n-1}. We have that

$$\begin{aligned}c(A_n \setminus \{m-n\}) &= 2c^d(m-n-1) + c^t(m-n-1) \\ &\leq 2c^d(m-n-1) + 4c^d + c^t \\ &\qquad \text{(As } 4c^d \geq (m-n-2)c^t\text{)} \\ &= 2c^d(m-n+1) + c^t \\ &= c(A_{n-1}).\end{aligned}$$

Consequently, whenever $\mathcal{I}$ is such that $h = 1$ and $4c^d > c^t(m-n-2)$, A is an EF1 allocation. In fact, due to the high distance cost, all EF1 allocations here must be of this form. As a result, we have that $A \in \arg\min_{A' \text{ satisfies } EF1} \sum_{i \in N} c(A_i)$. Thus, $PoEF1(\mathcal{I}) = \frac{c^t(m-1)+c^d n(2m-n+1)}{(m-1)(c^t+2c^d)}$. □

We now show that when one-dimension dominates, we can obtain tighter bounds, only in terms of m and n.

Proposition 3. *Given an instance $\mathcal{I}$, we have that*

1. *If $\mathcal{I}$ has distance-dominant costs, $PoEF1 \leq \frac{(2m-n-1)n+2}{2(m-1)}$*
2. *If $\mathcal{I}$ has time-dominant costs, $PoEF1 \leq \frac{m+n}{m-1}$*

Proof. **Distance-dominant costs.** Given an instance $\mathcal{I}$ with distance-dominant costs, we know from Theorem 4 that $PoEF1(\mathcal{I}) \leq \frac{(m-1)c^t+(2m-n-1)nc^d}{(m-1)c^t+2(m-1)c^d}$. We now take into account the fact that $2c^d > (m-1)c^t$, to get the desired upper bound. ‘

$$\begin{aligned}PoEF1(\mathcal{I}) &\leq \frac{(m-1)c^t + (2m-n-1)nc^d}{(m-1)c^t + 2(m-1)c^d} \\ &\leq \frac{(m-1)c^t + (2m-n-1)nc^d}{2(m-1)c^d} \\ &\leq \frac{2c^d + (2m-n-1)nc^d}{2(m-1)c^d} \\ &= \frac{(2m-n-1)n+2}{2(m-1)}\end{aligned}$$

Time-dominant costs. Here we have that $(m-1)2c^d < c^t$. Here, we can give an even tighter upper bound.

Given an instance $\mathcal{I}$ with time-dominant costs, it can be seen that an allocation A where there exist $i, j \in N$ s.t. $|A_i| - |A_j| > 1$ cannot be EF1. As a result, for an allocation to be EF1 under time-dominant costs, it must be that for each $i \in N$, $\lfloor \frac{m-1}{n} \rceil \leq |A_i| \leq \lceil \frac{m-1}{n} \rceil$.

Recall that in the price of EF1, we consider the EF1 allocation with minimum sum of costs and we know from Observation 1 that such an allocation must minimize the sum of the distances traveled by the agents. To ensure EF1, we cannot have agents with larger bundles travel further than agents with smaller bundles. Thus, under an EF1 allocation, the distance traveled by an agent with a larger bundle must be less than or equal to the distance traveled by an agent with a smaller bundle. Analogous to the proof of Theorem 2, the only thing we can now do to ensure minimizing the sum of distances is to allocate contiguous bundles.

As in the proof of Theorem 4, we have that the sum of the distances will be maximized when the hub is at the end of the path. We can assume without loss of generality that the hub is at $h = 1$. Consequently, under an EF1 allocation A with minimum sum of costs the sum of the distance costs will be at most

$$\sum_{i \in N} R(A_i) = \frac{m-1}{n} + 2\frac{m-1}{n} + \cdots + n\frac{m-1}{n} = \frac{m-1}{n}\frac{n(n+1)}{2} = \frac{(n+1)(m-1)}{2}$$

Thus, we have that the price of EF1 under an instance with time-dominant costs is

$$\begin{aligned} PoEF1(\mathcal{I}) &\le \frac{(m-1)c^t + (m-1)(n+1)c^d}{(m-1)c^t + (m-1)c^d} \\ &\le \frac{(m-1)c^t + (m-1)(n+1)c^d}{(m-1)c^t} \\ &\le \frac{(m-1)c^t + (n+1)c^t}{(m-1)c^t} \\ &\le \frac{m-n}{m-1}. \end{aligned}$$

□

C.2 Price of MMS.

We now prove tighter upper bounds on the price of MMS under one-dimension dominant costs.

Proposition 4. *Given an instance $\mathcal{I}$, we have that*

1. *Under time-dominant costs, $PoMMS(\mathcal{I}) \le \frac{m}{m-1}$,*
2. *Under distance-dominant costs $PoMMS(\mathcal{I}) \le 2$ and*
3. *Whenever either $c^d = 0$ or $c^t = 0$, $PoMMS(\mathcal{I}) = 1$.*

Proof. **Time-Dominant Costs.** We know from Theorem 2 that under time-dominant costs, under an MMS allocation A, for each $i \in N$, it must be that $\lfloor \frac{m-1}{n} \rfloor \le |A_i| \le \lceil \frac{m-1}{n} \rceil$. As in the proof of Proposition 3, the sum of the distances traveled by the agents can be at most $\frac{(n-1)(m-1)}{2}$. Consequently, we have that the price of MMS here is

$$\begin{aligned} PoMMS(\mathcal{I}) &\le \frac{(m-1)c^t + (m-1)(n+1)c^d}{(m-1)c^t + (m-1)c^d} \\ &\le \frac{(m-1)c^t + (m-1)(n+1)c^d}{(m-1)c^t} \\ &\le \frac{(m-1)c^t + (n+1)c^t}{(m-1)c^t} \\ &\le \frac{m-n}{m-1}. \end{aligned}$$

Note that this matches the Price of EF1, however, we know from Theorem 2 that every MMS allocation must be EF1 under time-dominant costs, so this is reasonable.

Distance-Dominant Costs. In this case, we know from Corollary 1 that under an MMS allocation with minimum sum of costs, every edge is traversed by at most 2 agents. As a result, the sum of the distances traveled by the agents can be at most $2(m-1)$. As a result, in this case,

$$\begin{aligned} PoMMS(\mathcal{I}) &\le \frac{(m-1)c^t + 4(m-1)c^d}{(m-1)c^t + (m-1)2c^d} \\ &= 1 + \frac{2c^d}{c^t + 2c^d} \\ &\le 2 \end{aligned}$$

This bound, is not tight, but in fact for every $\epsilon \in (0, 1)$, we can come up with a family of instances where $PoMMS = 2 - \epsilon$. Intuitively, whenever $c^t > 0$ but very small, we will require that the furthest node from the hub be serviced by one agent and all remaining nodes be serviced by a different agent. The small the c^t is here, the closer the Price of MMS will be to 2. We now discuss the case of $c^t = 0$.

Time-free tasks $c^t = 0$, in this case, under an MMS allocation A with minimum sum of costs, if $h > 1$, one agents services all nodes from $1, \cdots, h-1$ and if $h < m$, a different agent services all nodes from $h+1, \cdots, m$. As a result, the sum of the distances traveled under A is exactly $m-1$. Consequently, we have that $PoMMS(\mathcal{I}) = 1$.

Distance-free graphs $c^d = 0$. In this case, each agent virtually incurs no cost from the distance traveled, thus, from Observation 1, we have that $PoMMS(\mathcal{I}) = PoEF1(\mathcal{I}) = 1$. □

D Omitted Material from Section 5

We now present the complete experimental results.

Experimental Methods. We create instances where the number of agents is such that $2 \le n \le 6$ and $m \in \{11, 15, 19\}$. For each combination of m and n, we allow the hub to be on any of the positions $\{1, 2, \cdots, m\}$. We fix the value c^t to be 100 and vary choose c^d to be $2c^t\tau$. Here, the coefficient τ – a critical parameter that balances inspection time and transportation costs – is logarithmically sampled (sampled values are uniformly spaced on the logarithmic scale) over 20 values that span $[0.1, 10]$.

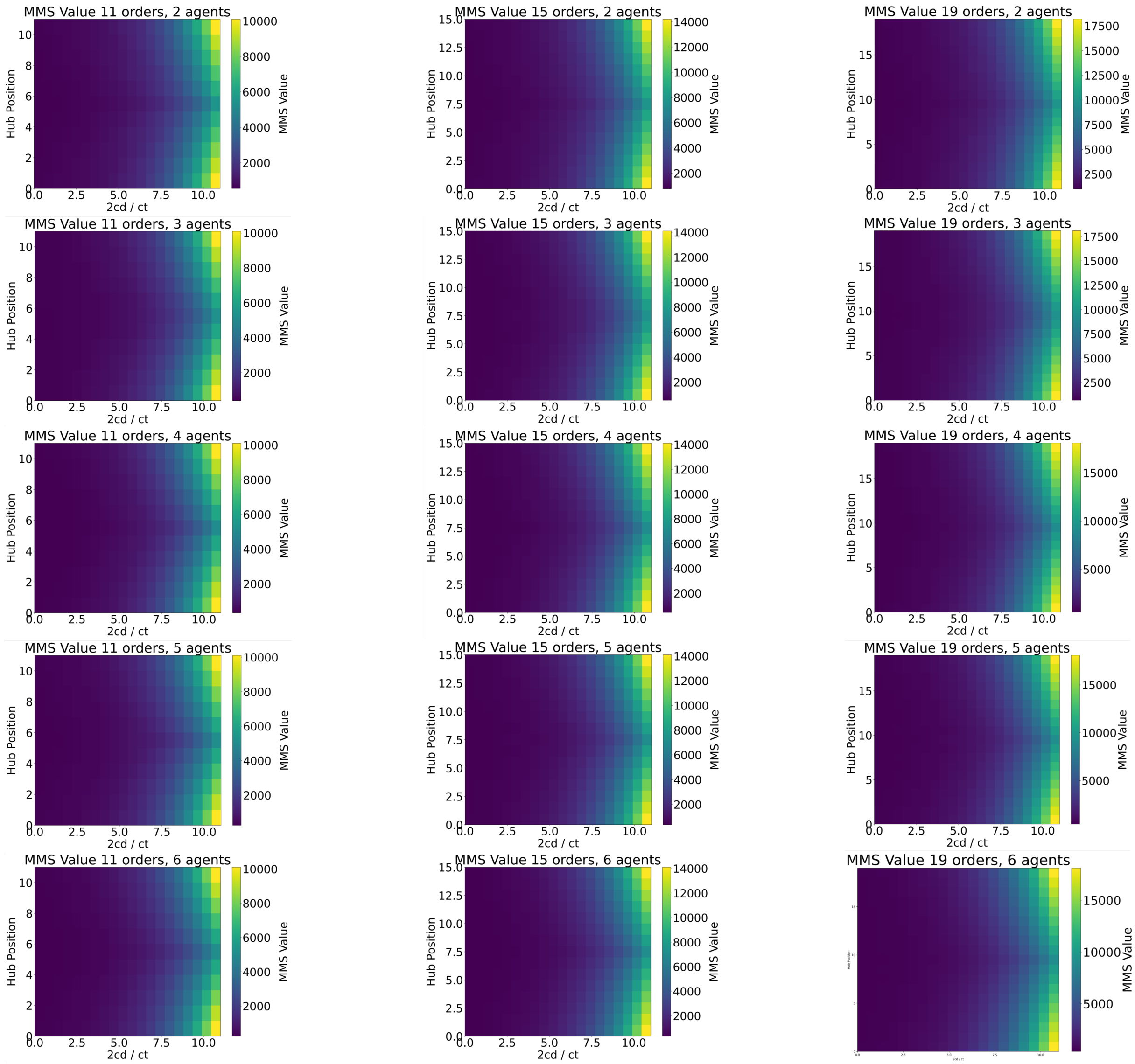

Figure 9: Experimental results on MMS cost varying with different hub locations and values of $\frac{2c^d}{c^t}$

Thus our instances span distance-dominant instances ($\tau \ll 1$) to time-dominant ones ($\tau \gg 1$). For each instance generated, we compute the MMS share cost using our algorithm described in Theorem 1. As a result, we can compare the effect of three parameters on the MMS cost: hub position, ratio of the costs $\frac{2c^d}{c^t}$ and the number of orders/nodes.

D.1 MMS Share Cost

We first observe the effect on the MMS share cost of two parameters: the hub position and the ratio of $\frac{2c^d}{c^t}$. We present our results in Figure 9. We find that the range of the MMS cost varies based on the number of orders. In contrast, conditioned on a specific number of orders, the range of the MMS cost stays the same even when we vary the number of agents.

Keeping the value of $\frac{2c^d}{c^t}$ fixed the MMS cost seems to be minimized when the hub is at the center. We find that as the hub moves further from the middle, the MMS cost increases. Further, this increase is symmetric on either side. This is consistent with our theoretical bound son the price of fairness, where we found the worst case to be when the hub was at either end of the path. However, this effect is less visible for instances with a larger number of orders.

Keeping the hub position fixed, we find that initially there is noticeable little effect on the MMS cost and these seem to increase only after $\frac{2c^d}{c^t} > 6.0$. The increase is also sharper further away from the hub. Further, for larger number of orders, the increase in the MMS value is less visible for the same value of $\frac{2c^d}{c^t}$. This makes it hard to see anything interesting for the smaller values of $\frac{2c^d}{c^t}$.

Observe that the heatmaps for a fixed number of orders look very similar even if we change the number of agents. That is, we observe similar relative tradeoffs of hub position and $\frac{2c^d}{c^t}$

for all the chosen values of n. However, on closer inspection, fixing m, hub position to be not at either extreme and $\frac{2c^d}{c^t}$, as the number of agents increases, the MMS share cost decreases.

D.2 Ratio of MMS Value and Social Optimal Cost

Observe that in Figure 9, not much discernible difference is observed in the MMS value for smaller values of $\frac{2c^d}{c^t}$. It is important to note that as the ratio of the $\frac{2c^d}{c^t}$ changes, the minimum sum of agent costs for the given instance would also change. That is, the socially optimal cost of $(m-1)(c^t+2c^d)$ will also change. Thus, it is important to normalize for this.

In order to better observe the effect of varying $\frac{2c^d}{c^t}$, we study the effect of $\frac{2c^d}{c^t}$ and the hub position on the ratio of MMS share cost and the socially optimal cost. These results are presented in Figure 10 and we can observe some marked differences from the results in Figure 9.

Firstly, the range of the ratio of the MMS share cost to the socially optimal cost remains the same for all three choices of the number of orders. Further for the two extreme hub positions, the values seem to be the same for 11, 15 and 19 orders. Interestingly, this is not the case for when the hub is at the center. When the hub is closer to the center, the MMS value is smaller for the same ratio of $\frac{2c^d}{c^t}$ when there are more orders. This can be by the center of the heatmaps being progressively less bright as we increase the number of orders.

For a fixed hub position, as $\frac{2c^d}{c^t}$ increases, ratio of MMS cost to SO cost increases much more evenly, and there is a noticeable increase even for the smaller values of $\frac{2c^d}{c^t}$. That is, in this experiment, for the same τ, the changes brought about by different hub positions are more obvious than in the previous experiment. Further, for the extreme hub positions, the MMS share cost is smaller.

It is important to note that for this experiment, we do see noticeable differences in the heatmaps generated as we change the number of agents for a fixed number of orders. This is particularly noticeable when the hub is in the center. In fact, on closer inspection, we find that as we increase the number of agents, the ratio of the MMS cost to the SO cost decreases. This is not directly apparent at first glance as the range of values is different, but this coincides with the observation that the MMS share cost decreases as we increase the number of agents.

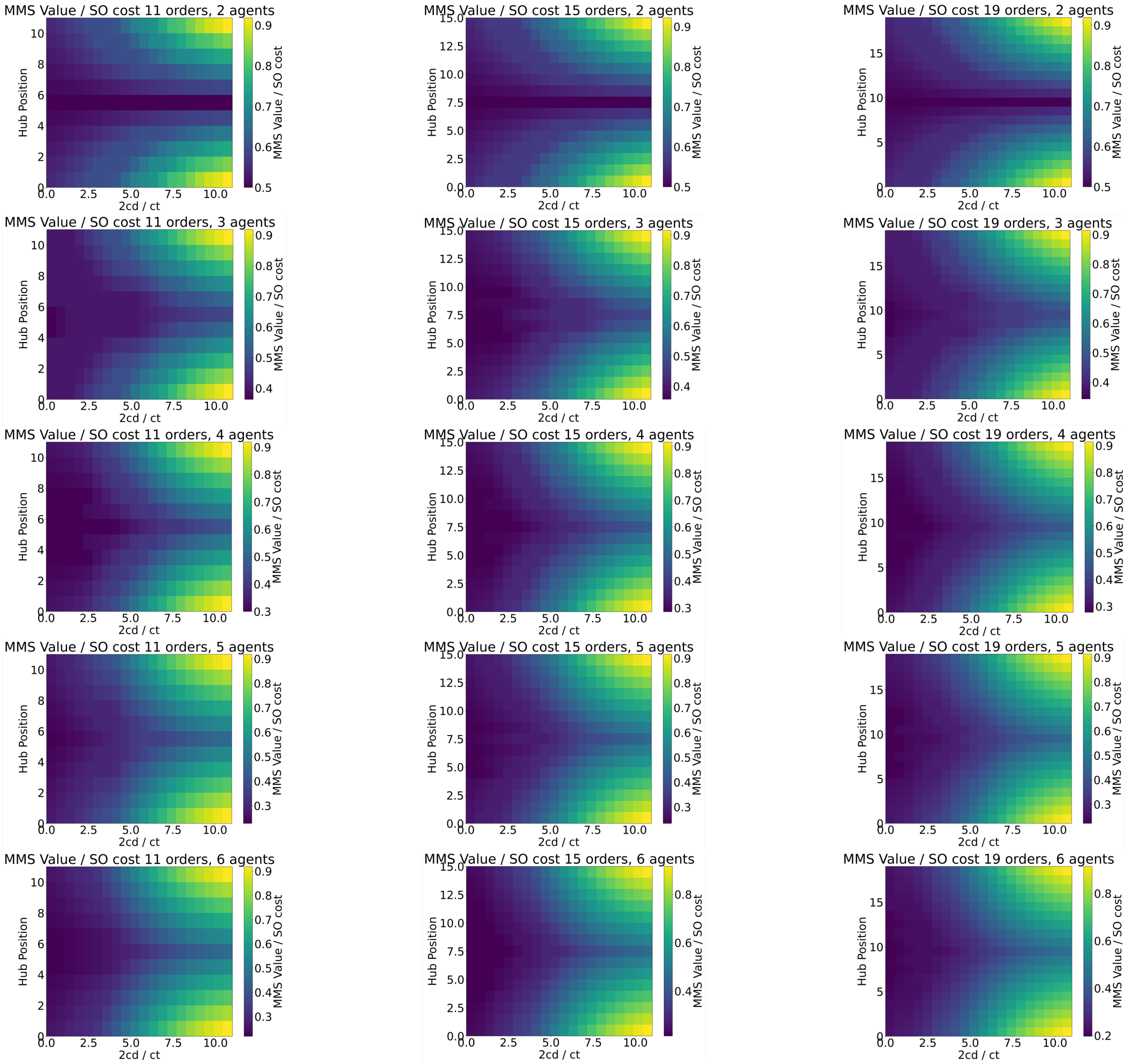

Figure 10: Experimental results on MMS cost varying with different hub locations and values of $\frac{2c^d}{c^t}$