LOCAL VS. GLOBAL INCENTIVE COMPATIBILITY ON ORDINAL TYPE SPACES*

Souvik Roy[1] and Ujjwal Kumar[1]

[1]Economic Research Unit, ISI Kolkata

September, 2019

Abstract

We consider locally incentive compatible (LIC) mechanisms with deterministic allocation rules and transfers with quasilinear utility. We use a "uniform" notion of local incentive compatibility. A mechanism is LIC if there is some (small) $\epsilon > 0$ such that it is IC on every pair of types s and t with the (Euclidean) distance between them at most ϵ. We show that LIC and IC are equivalent on a strict ordinal type-space if the ordinal domain satisfies the no-restoration property (Sato (2013)). A large class of strict ordinal type-spaces such as single-peaked, single-dipped, single-crossing, etc., satisfy the no-restoration property. Next, we show that the same does not hold if we allow the type-space to be weak. We introduce the notion of almost everywhere IC. A mechanism is almost everywhere IC if it IC outside a set of (Lebesgue) measure zero. We provide a sufficient condition on weak type-spaces for the equivalence of LIC and almost everywhere IC. Various subsets of single-peaked and single-crossing domains satisfy our sufficient condition. Finally, we provide results on how to check whether a given mechanism is IC or not. It follows from our result that in addition to local types, only a few types on the "boundary" of the type-space needed to be checked additionally. Our results apply to several non-convex type-spaces, and thereby generalize the results in Carroll (2012) considerably.

KEYWORDS: local incentive compatibility, (global) incentive compatibility, ordinal type spaces

JEL CLASSIFICATION CODES: D82, D44, D47

This is a preliminary and incomplete draft. Please do not quote.

*The authors would like to thank Debasis Mishra for his valuable suggestions.

1. INTRODUCTION

We consider standard mechanism design problem where a set of agents have valuations for each object in a finite set of objects. Based on these valuations, the planner has to select an object to be shared by all the agents and some payment for each agent. Such a decision scheme is called a mechanism.

Agents evaluate their net utilities by means of quasilinear utility functions. A mechanism is incentive compatible (IC) if no agent can increase his/her net utility by misreporting his/her type. A mechanism is locally IC (LIC) if no agent can increase his/her net utility by misreporting to a type that is "close" to his/her sincere types.

An important problem in mechanism design is to characterize the IC mechanisms for a given type-space. Except from the case when the type-space is $\mathbb{R}^{|A|}$, where A is the set of objects, this turns out to be a hard problem. As an intermediate step, researchers have got interested in exploring if the requirement of IC can be reduced considerably. Local IC (LIC) turns out to be a way.

LIC ensures that a mechanism is IC on the types that are sufficiently close (with respect to Euclidean distance) to each other. Carroll (2012) worked with a slightly weaker notion of LIC and showed that if the type-space is convex, then LIC is equivalent to IC. The objective of this paper is to extend the results in Carroll (2012) to non-convex type-spaces.

First, we consider type-spaces that are strict, that is, no two objects have the same valuation in any type. A domain is said to satisfy the no-restoration property if between any two preferences of the domain, there is a path of preferences in which no two alternatives change their relative ordering more than once. Almost all well-known domains such as single-peaked, single-crossing, single-dipped, etc., satisfies the no-restoration property. The notion of no-restoration property is introduced in Sato (2013), where it is shown that if a domain satisfies this property then every locally strategy-proof social choice function is strategy-proof. We extend this result for cardinal set-up and show that if an ordinal domain satisfies the no-restoration property, then every LIC mechanism on its type-space is IC. It is worth mentioning that our result applies to domains that are not necessarily convex, not even connected.

Next, we consider weak type-spaces. We introduce the notion of almost everywhere IC. A mechanism is almost everywhere IC if it is IC outside a set of (Lebesgue) measure zero. We provide a sufficient condition on a weak ordinal domain so that every LIC mechanism on its type-space is almost everywhere IC. The closure of single-peaked or single-crossing type-spaces, single-plateaued type-spaces, etc., satisfy our sufficient condition.

Finally, we consider the problem of checking whether a given mechanism is IC or not on an arbitrary domain. We show that apart from checking the local types, one needs to check only the boundary types to ensure IC of a mechanism.

As we have mentioned, our results apply to domains that are not necessarily convex, not even connected. Thus, they generalize the results in Carroll (2012) in a considerable manner. Mishra et al. (2016) consider the same problem for a particular type of mechanisms, called payments-only mechanisms. We extend their results by dropping the assumption of payments-onlyness.

In the auction design literature with transfers and quasilinear utility, a long standing research agenda has been to identify a minimal set of incentive constraints that will imply overall incentive compatibility - see discussions on relaxed problem in Chapter 7 of Fudenberg and Tirole (1991), Armstrong (2000) and Chapter 6 in Vohra (2011).

2. Model

We present the basic model in two further subsections. First, we present the notion of ordinal preferences, the notion of types and discuss the notion of a type representing an ordinal preference. Second, we present the notion of mechanisms and discuss their properties.

2.1 Ordinal domains and type-spaces

Let A be a finite set of alternatives. A strict preference on A is a linear order, that is, an antisymmetric, complete, and transitive binary relation on A. For two alternatives $a, b \in A$ and a strict preference P, we write aPb to mean that a is strictly preferred to b according to P. We denote by $\widehat{\mathcal{P}}(A)$ the set of all strict preferences on A. A strict ordinal domain $\widehat{\mathcal{D}}$ is a subset of $\widehat{\mathcal{P}}(A)$.

A weak preference on A is a weak order, that is, a complete and transitive binary relation on A. For two alternatives $a, b \in A$ and a weak preference R, we write aRb to mean that a is "as good as" b according to R. We denote by $\overline{\mathcal{P}}(A)$ the set of all weak preferences on A. A weak ordinal domain $\overline{\mathcal{D}}$ is a subset of $\overline{\mathcal{P}}(A)$.

A type t is a mapping from A to $\mathbb{R}$ that represents the utility/valuation of each alternative in A. A subset T of $\mathbb{R}^{|A|}$ is called a type-space. A type-space T is called strict if $t(a) \neq t(b)$ for all $t \in T$ and all $a, b \in A$, otherwise it is called weak.

In this paper, we deal with type-spaces that have some additional structure. We say that a type t represents a strict preference P if for all $a, b \in A$, aPb if and only if $t(a) > t(b)$. For a domain $\mathcal{D}$ (strict or weak), we denote by $V(\mathcal{D})$ the set of all types that represent some preference in $\mathcal{D}$, that is, $V(\mathcal{D}) = \{t \in \mathbb{R}^{|A|} \mid t \text{ represents } P \text{ for some } P \in \mathcal{D}\}$. A type-space T is ordinal if there exists a domain $\mathcal{D}$ such that $T = V(\mathcal{D})$.

2.2 MECHANISMS AND THEIR PROPERTIES

We consider one-agent model in this paper. This is without loss of generality for our analysis. In fact, as it is well-known in the literature, all the results of this paper can be generalized to the case of more than one agents in a systematic manner.

An allocation rule is a map $f : T \to A$ and a payment rule is a map $p : T \to \mathbb{R}$. A (direct) mechanism μ is a pair consisting of an allocation rule f and a payment rule p.

If the agent has type t but (falsely) reports s to the mechanism, then his/her net utility is given by

$$t(f(s)) - p(s).$$

In particular, if the agent with type t reports truthfully, then his/her net utility is $t(f(t)) - p(t)$. Note that here we assume quasilinearity to evaluate utility from payments. We say a type t cannot manipulate to a type s if the net utility from reporting t is at least as much as that from reporting s, that is, if

$$t(f(t)) - p(t) \geq t(f(s)) - p(s).$$

Definition 2.1. A mechanism μ is incentive compatible (IC) on a pair of types (t, s) if t cannot manipulate to s. It is called IC on a type-space T if it is IC on every pair of types $(t, s) \in T \times T$.

Next, we introduce the notion of local incentive compatibility. It requires that a mechanism is incentive compatible on pairs of types that are "very close" to each other. More formally, for a given (arbitrarily small) $\epsilon > 0$, a mechanism is called ϵ-locally IC if it is IC on every pair of types having (Euclidean) distance at most ϵ. A mechanism is called locally IC if it is ϵ-locally IC for some $\epsilon > 0$.

Definition 2.2. A mechanism is said to be locally IC if there exists $\epsilon > 0$ such that it is IC on every pair of types (t, s) with $d(t, s) < \epsilon$.

3. Local incentive compatibility vs. incentive compatibility on strict ordinal type-spaces

In this section, we explore the connection between local incentive compatibility and incentive compatibility on strict ordinal type spaces. More precisely, we investigate under what condition on a domain, every local IC mechanism is IC.

We present a condition on a strict ordinal domain $\widehat{\mathcal{D}}$ called the no-restoration property (see Sato (2013)). For some $1 \leq k \leq m$, we denote the k-th ranked alternative of P by $P(k)$. Two preferences P and P' are said to be adjacent local if they differ by the ranking of two consecutively ranked alternatives, that is, there is $1 \leq k < m$ such that $P(k) = P'(k+1)$, $P(k+1) = P'(k)$, and $P(l) = P'(l)$ for all $l \neq k$. A sequence of strict preferences $(P^1, \ldots, P^k)$ (or a path from P^1 to P^k) is called an adjacent local path if P^l and P^{l+1} are adjacent local preferences for all $1 \leq l < k$. An adjacent local path $(P^1, \ldots, P^k)$ is said to satisfy the no-restoration property if every pair of alternatives change their relative ordering at most once along the path, that is, for every distinct $a, b \in A$, there are no $1 \leq r < s < t \leq k$ such that $aP^r b, bP^s a$, and $aP^t b$. A strict ordinal domain $\widehat{\mathcal{D}}$ is said to satisfy no-restoration property if for all $P, P' \in \widehat{\mathcal{D}}$, there is an adjacent local path in the domain satisfying the no-restoration property.

We are now ready to present the main result of this section. It says that if a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the no-restoration property, then every locally IC mechanism on $V(\widehat{\mathcal{D}})$ is IC.

Theorem 3.1. *Let a strict ordinal domain $\widehat{\mathcal{D}}$ satisfy no-restoration property. Then, every locally IC mechanism on $V(\widehat{\mathcal{D}})$ is IC.*

The proof of Theorem 3.1 can be found in Appendix .1.

A large class of strict ordinal domains of practical importance such as single-peaked, single-dipped, single-crossing, etc., satisfy the no-restoration property. Therefore, Theorem 3.1 implies that locally IC and IC are equivalent on the cardinal versions of these domains.

3.1 A generalization in the direction of Carroll (2012)

Carroll (2012) introduced a slightly different notion of local IC. He calls a mechanism locally IC if for every $t \in T$ there exists $\epsilon > 0$ such that for all $s \in T$ with $d(t, s) < \epsilon$, it is IC on both the pairs of types (t, s) and (s, t). Note that according to this notion, one has the freedom to chose different $\epsilon > 0$ for different types, whereas in our notion one has to chose the same $\epsilon > 0$ for all types. If the infimum value of the ϵs' chosen for different types (in Carrol's.... notion) is positive, then that infimum value can be taken as the choice of ϵ in our notion, and consequently these two notions will become equivalent. However, if the said infimum is zero, then our notion is slightly stronger than that of Carroll (2012). To see this, consider the situation where there are just two alternatives and the type-space $T = \{t \in \mathbb{R}^2 \mid t(a) \neq t(b)\}$. Thus, T is disconnected and can be written as a union of two disjoint open spaces $T^1 = \{t \in \mathbb{R}^2 \mid t(a) < t(b)\}$ and $T^2 = \{t \in \mathbb{R}^2 \mid t(a) > t(b)\}$. In such situations, one can define neighbourhoods of the points, say in T^1, such that none of them intersects T^2. This means local IC as defined in Carroll (2012) does not impose IC on a pair of types (s, t) where $s \in T^1$ and $t \in T^2$, and consequently such notion of local IC can never ensure IC. However, in our notion, local IC ensures IC on pair of types that are very close but still come from different partitions of T. Thus, there are scopes that one can achieve IC by means of our notion of local IC.

It is worth mentioning that our notion of local IC is as useful as that of Carroll (2012) for all practical purposes. In reality, if one wants to check (perhaps by means of a computer or so) whether some mechanism is locally IC or not, he/she can only check it for some given neighbourhood of each type. In other words, one cannot consider a sequence of neighbourhoods the size of which converges to zero.

As we have discussed, Theorem 3.1 (or some version of it) cannot be achieved by using the notion of local IC defined in Carroll (2012). In what follows, we discuss how this notion can be modified to obtain Theorem 3.1. A mechanism μ is said to be adjusted locally IC on a strict type-space T if (i) for every type t in T, there is a neighborhood around t such that μ is IC on both (t, s) and (s, t) for all types s in that neighborhood, and (ii) for every type $\bar{t}$ that lies on the boundary of T (that is, in $cl(T) \setminus T$), there is a neighborhood of $\bar{t}$ such that μ is IC on every pair of strict types in that neighborhood.

Roughly speaking, part (i) says that a locally IC mechanism μ cannot be manipulated from a strict type t to a strict type s, when t and s are "arbitrarily close". Implication of part (ii) is somewhat involved. Note that a strict ordinal type-space is by definition disconnected. For instance, if there are two alternatives a and b, and $\mathcal{D}$ contains both the strict preferences over these two alternatives, then there are two (connected) components of $V(\mathcal{D})$: one contains all types that represent the preference where a is preferred to b, and the other contains all types that represent the preference where b is preferred to a. Now, suppose that a mechanism is IC on pairs of types that are arbitrarily close to each other as required by part (i). Since one can always find a small neighborhood $N(t)$ around a type t in one component of $V(\mathbb{D})$ such that $N(t)$ does not intersect the other component, such a mechanism need not be IC on a pair (t, s) where t and s come from different components. In other words, part (i) can no way ensure IC for a mechanism on a type-space that is disconnected. To overcome this problem, we introduce part (ii). As desired, part (ii) imposes IC on pairs of types that are in different components, but still close enough. To model this closeness, we consider a type that lies on the boundary between two components. For instance, with two alternatives, such a type is one where both the alternatives have equal utilities. Consider an arbitrarily small neighborhood of such a type. Since $\bar{t}$ lies on the boundary, such a neighborhood will intersect both the components. Part (ii) requires that the mechanism is IC on any pair of strict types in this neighborhood. Technically, part (ii) ensures that the local IC property can cross boundaries of a disconnected type-space. If one wants to ensure IC by local IC, some such condition is clearly necessary for disconnected type-spaces.

Definition 3.1. A mechanism μ on a strict type space $\hat{T}$ is said to be adjusted locally IC if

(i) for every $t \in \hat{T}$, there exists an open neighborhood $N(t) \subseteq \hat{T}$ of t such that for all $t' \in N(t)$, μ IC is on both (t, t') and (t', t), and

(ii) for every $\bar{t} \in cl(\hat{T}) \setminus \hat{T}$, there exists an open neighborhood $N(\bar{t}) \subseteq cl(\hat{T})$ of $\bar{t}$ such that for all $t', t'' \in N(t) \cap \hat{T}$, μ is IC on (t', t'').

Note that the notion of local IC in Carroll (2012) requires only part (i) of the notion of adjusted locally IC. As we have justified, part (ii) is necessary if, for instance, one is dealing with non-connected type-spaces. Additionally, it is worth mentioning that the (Lebesgue) measure of the boundary of strict ordinal type-spaces is zero, and hence, the additional constraint required by part (ii) can be considered as a mild requirement.

3.2 A GENERALIZATION OF THEOREM 3.1

Note that Theorem 3.1 applies to strict type-spaces in which, for some strict ordinal domain, *all* types representing each preference in the domain are present. This requirement is somewhat strong–one can have situations where, for some ordinal preference, only a few types (but not all) are present in the type-space. In what follows, we present a generalization of Theorem 3.1 where this requirement is weakened. In contrast to providing sufficient conditions on ordinal domains, here we directly look for sufficient conditions on strict type-spaces. We say an alternative a

overtakes another alternative b from a strict preference P to another strict preference P' if bPa and $aP'b$.

We say a strict type-space $\hat{T}$ satisfies the no-restoration property if for every two types t and t' in it, there is an adjacent local path $(P^1, \ldots, P^k)$ of strict preferences, each representing some type in the type-space, such that (i) the path satisfies the no-restoration property, and (ii) if an alternative a overtakes another alternative b from a preference P^l to P^{l+1}, then for each t^{l+1} representing P^{l+1}, we have that (a) for each alternative c other than a, there exists a type t^l in $\hat{T}$ representing P^l such that from t^l to t^{l+1}, the utility of c decreases relative to every other alternative, and (b) there is a type $\hat{t}^l$ representing the preference P^l such that from $\hat{t}^l$ to t^{l+1}, the utility of a decreases relative to every other alternative except b, and from t to $\hat{t}^l$, the utility of a increases relative to b.

Clearly, this condition is quite technical. However, we show that it has significant contributions, particularly for domains that are "far away" from being convex.

For a strict type-space $\hat{T}$, we denote the set of all strict preferences that represent some types in $\hat{T}$ by $\mathcal{D}(\hat{T})$, that is, $\mathcal{D}(\hat{T}) = \{P \mid \text{ there exists } t \in \hat{T} \text{ such that } t \text{ represents } P\}$.

Definition 3.2. A strict type space $\hat{T}$ satisfies the no restoration property if for all $t, t' \in \hat{T}$, there exists an adjacent path of strict preferences $(P^1, \ldots, P^k)$ in $\mathcal{D}(\hat{T})$ with $P^1 \in \mathcal{D}(t)$ and $P^k \in \mathcal{D}(t')$ such that

(i) $(P^1, \ldots, P^k)$ is a no restoration path in $\mathcal{D}(\hat{T})$, and

(ii) for all $l < k$ and all $t^{l+1} \in \mathcal{D}(P^{l+1})$, if some alternative a overtakes some other alternative b from P^l to P^{l+1}, then

  (a) for all $c \neq a$, there exists $t^l \in \mathcal{D}(P^l)$ such that $t^l(c) - t^l(x) > t^{l+1}(c) - t^{l+1}(x)$ fo all $x \in A \setminus \{c\}$,

  (b) there exists $\hat{t}^l \in \mathcal{D}(P^l)$ such that $\hat{t}^l(a) - \hat{t}^l(y) > t^{l+1}(a) - t^{l+1}(y)$ for all $y \in A \setminus \{a, b\}$ and $t(a) - t(b) \leq \hat{t}^l(a) - \hat{t}^l(b)$.

Theorem 3.2. *Let $\hat{T}$ be a strict type space that satisfies the no restoration property. Let μ be a mechanism that is IC on $V(P, P') \cap \hat{T}$ for all adjacent local preferences P and P' in $\mathcal{D}(\hat{T})$. Then, μ is IC on $\hat{T}$.*

The proof of Theorem 3.2 can be found in Appendix .1.

3.3 General local incentive compatibility vs. incentive compatibility on strict ordinal type-spaces

In this section, we consider a general notion of local preferences in a strict ordinal domain. More formally, we assume that the pairs of local preferences in a strict ordinal domain are given a priori, which can be totally arbitrary. Thus, two preferences need not be adjacent in order to be local. We call two types local if the preferences they represent are local. We investigate the connection between local IC and IC for this general notion of localness. Kumar et al. [2019, working paper] provide a necessary and sufficient condition on a strict ordinal domain so that every local IC ordinal (without any transfer) mechanism on it is IC. They present the structure of local preferences in an ordinal domain by means of a graph.

Let $\widehat{\mathcal{D}}$ be a strict ordinal domain and let $G = \langle \widehat{\mathcal{D}}, E \rangle$ be an (undirected) graph on $\widehat{\mathcal{D}}$. Two preferences in $\widehat{\mathcal{D}}$ are called local in G if they form an edge in G. We redefine the notion of no-restoration path with respect to a graph in a natural way. A path $(P^1, \ldots, P^k)$ from P^1 to P^k is local in G if every two consecutive preferences in it are local in G. A path $(P^1, \ldots, P^k)$ local in G, is called a no-restoration path if for every distinct $a, b \in A$, there are no $1 \leq r < s < t \leq k$ such that $aP^r b$, $bP^s a$, and $aP^t b$. We say a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the no-restoration property with respect to G if for all $P, P' \in \widehat{\mathcal{D}}$, there is a no restoration local path in G from P to P'.

We introduce the notion of ordinal localness for two types. Two types $\hat{t}$ and $\hat{t}'$ are said to be ordinally local in G if the preferences they represent, (that is, $\mathcal{D}(\hat{t})$ and $\mathcal{D}(\hat{t}')$) are local in G. Note that the (Euclidean) distance between two ordinally local types need not be very small.

Our next theorem generalizes Theorem 3.1 for general local structures. It says that if a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the no-restoration property with respect to a graph, then every mechanism, that is IC on each pair of ordinally local types, is IC on $V(\widehat{\mathcal{D}})$.

Theorem 3.3. *Let $\widehat{\mathcal{D}}$ be a strict ordinal domain satisfying the no restoration property with respect to a graph G. If a mechanism is IC on $V(P, P')$ for all preferences P and P' in $\widehat{\mathcal{D}}$ that are local in G, then it is IC on $V(\widehat{\mathcal{D}})$.*

The proof of Theorem 3.3 can be found in Appendix .1.

Note that Theorem 3.3 is slightly different in nature from Theorem 3.1. While in Theorem 3.1, we impose IC on types that are "close" (with respect to Euclidean distance) in the type-space, here we impose it on types that come from local ordinal preferences. Clearly, when the notion of localness for preferences is given by an arbitrary graph, it does not make sense to define the same for types by means of Euclidean distances.

4. Local incentive compatibility vs. incentive compatibility on weak type-spaces

In this section, we explore the relation between local incentive compatibility and incentive compatibility on weak type-spaces. We begin with the simpler case where the type-space is the closure of some strict type-space.

We use the following notations to ease our presentation. For a weak ordinal domain $\overline{\mathcal{D}}$, we denote its maximal strict ordinal subset by $strict(\overline{\mathcal{D}})$, that is, $strict(\overline{\mathcal{D}}) = \{R \in \overline{\mathcal{D}} \mid R \text{ is a strict preference}\}$.

4.1 Closure of type-spaces of strict ordinal domains

Let $\widehat{\mathcal{D}}$ be a strict ordinal domain and let $cl(V(\widehat{\mathcal{D}}))$ be the closure of $V(\widehat{\mathcal{D}})$. We use the same notion of local IC as in Definition 3.1 for such type-spaces. Since these spaces are closed, part (ii) of Definition 3.1 is vacuously true, and consequently, our notion of local IC boils down to that of Carroll (2012). However, Theorem 3.1 does not hold anymore. In order to obtain a version of it, we need to strengthen our no-restoration property.

A no-restoration adjacent local path $(P^1, \ldots, P^k)$ in a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the consistency property if whenever an alternative a overtakes another one from some preference P^l to P^{l+1} along this path, it must be that (i) the alternatives that are ranked strictly below a in P^1 are also ranked strictly below a at P^l, and (ii) in the preference P^l, every alternative that is ranked above a in P^1 is preferred to every alternative that is ranked below a in P^1. Loosely put, consistency says that whenever an alternative moves up from a preference P^l in a no-restoration path, it must be the case that the alternative is (i) already in a "higher position" in P^l in comparison with its position in the initial preference P^1 of that path, and (ii) the alternatives that are ranked above a in the initial preference P^1 appear as a "top-set" (that is, an upper contour set) in P^l. Below, we present a formal definition.

Definition 4.1. A no-restoration adjacent local path $(P^1, \ldots, P^k)$ in a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the consistency property if for all $l \in \{1, \ldots, k-1\}$ and all $a \in A$, a overtakes some alternative from P^l to P^{l+1} implies

(i) $L(a, P^1) \subseteq L(a, P^l)$, and

(ii) $\bar{U}(a, P^1) = \bar{U}(b, P^l)$ for some $b \in A$.

A strict ordinal domain is said to satisfy the consistent no-restoration property if between every two preferences in it, there is a consistent no-restoration adjacent local path. Note that a large class of single-peaked and single-crossing domains satisfy the consistent no-restoration property.

In what follows, we introduce the notion of almost everywhere IC. We use the following notation to ease the presentation. For a weak type-space $\bar{T}$, we denote its maximal strict subset by $strict(\bar{T})$, that is, $strict(\bar{T}) = \{t \in \bar{T} \mid t(a) \neq t(b)$ for all distinct $a, b \in A\}$.

Definition 4.2. A mechanism on a weak type-space $\bar{T}$ is said to be almost everywhere IC, if it is IC on every pair of types in $\bar{T} \times strict(\bar{T})$.

Thus, an almost everywhere IC mechanism might fail to become IC on a pair of types $(t, \bar{t})$ only if $\bar{t}$ lies on the boundary of $\bar{T}$. As we have already mentioned, the (Lebesgue) measure of the boundary $\bar{T} \setminus strict(\bar{T})$ is zero. Therefore, the measure (in the product space) of the pairs on which an almost everywhere IC mechanism may fail to be IC is also zero.

Our next theorem says that if a strict ordinal domain $\widehat{\mathcal{D}}$ satisfies the consistent no-restoration property, then every local IC mechanism on its type-space $cl(V(\widehat{\mathcal{D}}))$ is almost everywhere IC.

Theorem 4.1. *Let a strict ordinal domain $\hat{\mathcal{D}}$ satisfy the consistent no-restoration property. If a mechanism on $cl(V(\widehat{\mathcal{D}}))$ is locally IC, then it is almost everywhere IC.*

The proof of Theorem 4.1 can be found in Appendix .2.

4.2 TYPE-SPACES OF WEAK ORDINAL DOMAINS

In this section, we consider a more general class of weak type-spaces where the type-space need not be the closure of some strict ordinal type-space, and investigate the connection between local IC and IC on such spaces.

A type $\bar{t}$ is said to represent a weak preference R if aRb if and only if $\bar{t}(a) \geq \bar{t}(b)$. For a weak ordinal domain $\overline{\mathcal{D}}$, we denote the set of all types that represent some weak preference in $\overline{\mathcal{D}}$ by $V(\overline{\mathcal{D}})$, that is, $V(\overline{\mathcal{D}}) = \{\bar{t} \in \mathbb{R}^{|A|} \mid \bar{t}$ represents R for some $R \in \overline{\mathcal{D}}\}$. Note that for arbitrary weak ordinal domain $\overline{\mathcal{D}}$, the type-space $V(\overline{\mathcal{D}})$ need not be connected.

Note that if a weak type-space is the closure of some strict ordinal type-space, then it automatically includes types of certain weak ordinal preferences. In other words, one does not get much control on restricting the weak ordinal preferences. For an illustration, suppose that there are three alternatives a, b, and c, and consider the (singleton) domain $\widehat{\mathcal{D}}$ containing the single preference $P = abc$. If we take the closure of $V(\widehat{\mathcal{D}})$, then all types representing the preferences in $\{abc, [ab]c, a[bc], [abc]\}$ are included.[1] However, one might be interested in the type-space representing only the preferences, for instance, $\{abc, [ab]c\}$, or $\{abc, [ab]c, a[bc]\}$, etc. Clearly, such a type-space cannot be represented as a closure of some strict ordinal type-space.

In order to generalize Theorem 4.1 for weak ordinal type-spaces, we introduce the notions of weak-compatibility and dichotomous richness on a weak ordinal domain. For a weak preference R, we say a strict preference $\hat{P}$ is compatible with R if aPb implies $a\hat{P}b$ for all $a, b \in A$. For instance, if $R = [ab]c[de]f$, then the following preferences are compatible with R: $abcdef$, $abcedf$, $bacdef$, and $bacedf$. Weak compatibility says that for every weak preference R in $\overline{\mathcal{D}}$, there exists a strict preference in $\overline{\mathcal{D}}$ that is compatible with R.

A weak ordinal domain $\overline{\mathcal{D}}$ satisfies the dichotomous richness property if for all strict preference $\hat{P} \in \mathit{strict}(\overline{\mathcal{D}})$ and all $a \in A$, there exists a weak preference $R \in \overline{\mathcal{D}}$ in which alternatives that are weakly preferred to a in $\hat{P}$ form a top indifference class and the ones that are less preferred to a in $\hat{P}$ form a bottom indifference class, that is, xIy for all $x, y \in A$ such that either $x, y \in \bar{U}(a, \hat{P})$ or $x, y \notin \bar{U}(a, \hat{P})$, and xPy for all $x \in \bar{U}(a, \hat{P})$ and $y \notin \bar{U}(a, \hat{P})$. For instance, if $\hat{P} = wxyz$, then for $a = y$ the preference described above is $R = [wxy]z$, and for $a = x$ we have $R = ...$ Note that the preference R has two indifference classes, except when a is the bottom ranked alternative in $\hat{P}$ (in which case R has one indifference class containing all alternatives). Such a preference is called dichotomous.

To ease our presentation, we say that a weak ordinal domain $\overline{\mathcal{D}}$ satisfies the consistent no-restoration property (as defined in Section 4.1) if $\mathit{strict}(\overline{\mathcal{D}})$ satisfies it. The implication of our next theorem is as follows. Consider a weak ordinal domain $\overline{\mathcal{D}}$ that satisfies the consistent no-restoration property, the weak compatibility property, and the dichotomous richness property, and consider a mechanism on its type-space. As in Theorem 3.3, for every pair of adjacent local strict preferences $\hat{P}$ and $\hat{P}'$ in the domain, consider the types in $cl(V(\hat{P}, \hat{P}'))$. However, since all such types might not be present in $V(\overline{\mathcal{D}})$, consider only those that are present, that is, the types $cl(V(\hat{P}, \hat{P}')) \cap V(\overline{\mathcal{D}})$. Suppose that the mechanism is IC on all such types. Then, Theorem 4.2 says that it will be almost everywhere IC on the whole type-space $V(\overline{\mathcal{D}})$.

Theorem 4.2. *Let a weak ordinal domain $\overline{\mathcal{D}}$ satisfy the consistent no-restoration property, the weak-*

[1] By $[ab]c$, we denote a weak preference where a and b are indifferent, and are preferred to c.

compatibility property, and the dichotomous richness property. Suppose that a mechanism μ is IC on $cl(V(\hat{P}, \hat{P}')) \cap V(\overline{\mathcal{D}})$ for all adjacent strict preferences $\hat{P}, \hat{P}' \in strict(\overline{\mathcal{D}})$. Then, μ is almost everywhere IC on $V(\overline{\mathcal{D}})$.

The proof of Theorem 4.2 can be found in Appendix .2.

Note that Theorem 4.2 is quite similar in nature to Theorem 3.3, except the fact that in contrast to Theorem 3.3 where we consider arbitrary notion of localness, here we consider adjacent localness.

5. Local IC vs. IC for given mechanisms

It follows from our earlier results that on a large class of domains, local IC implies IC or almost everywhere IC. Recall that the notion of almost everywhere IC leaves the possibility of violating IC on a pair of types $(t, \bar{t})$, where $\bar{t}$ lies on the boundary of the type-space. Although the set of such types has measure zero, in this section we investigate how to check if a given mechanism is IC on such pair of types.

Consider a weak type-space $\bar{T}$. Let $\bar{t}$ be a weak type that lies on the boundary of $\bar{T}$. Consider an almost everywhere IC mechanism $\mu = (f, p)$ on $\bar{T}$. We want to check if it is IC on every pair of types $(t, \bar{t})$, where $t \in \bar{T}$ and $\bar{t} \in \bar{T} \setminus strict(\bar{T})$. Our next theorem says that we do not need to check this for every $t \in \bar{T}$, it is enough to do it for some particular $\hat{t} \in$ *strict* $(\bar{T})$. This particular type $\hat{t}$ satisfies the property that the utility of the outcome $f(\bar{t})$ relative to any other alternative strictly increases from $\bar{t}$ to $\hat{t}$, that is, $\hat{t}(f(\bar{t})) - \hat{t}(x) > \bar{t}(f(\bar{t})) - \bar{t}(x)$ for all $x \in A \setminus \{f(\bar{t})\}$.

Theorem 5.1. *Let $\bar{T}$ be a weak type-space. Suppose that a mechanism $\mu = (f, p)$ is almost everywhere IC on $\bar{T}$. Let a weak type $\bar{t} \in \bar{T} \setminus strict(\bar{T})$ and a strict type $\hat{t} \in strict(\bar{T})$ be such that*

(i) $\hat{t}(f(\bar{t})) - \hat{t}(x) > \bar{t}(f(\bar{t})) - \bar{t}(x)$ for all $x \in A \setminus \{f(\bar{t})\}$, and

(ii) μ is IC on $(\hat{t}, \bar{t})$.

Then, μ is IC on $\bar{T} \times \{\bar{t}\}$.

The proof of Theorem 5.1 can be found in Appendix .3.

Suppose a weak ordinal domain $\overline{\mathcal{D}}$ satisfies the consistent no-restoration property, the weak-compatibility property, and the dichotomous richness property. Suppose that a mechanism μ is IC on $cl(V(\hat{P}, \hat{P}')) \cap V(\overline{\mathcal{D}})$ for all adjacent strict preferences $\hat{P}, \hat{P}' \in strict(\overline{\mathcal{D}})$. Then by Theorem 4.2, we know that μ is almost everywhere IC on $V(\overline{\mathcal{D}})$. Consider a weak type $\bar{t} \in V(\overline{\mathcal{D}}) \setminus strict(V(\overline{\mathcal{D}}))$

such that at $\bar{t}$, the utility of the outcome $f(\bar{t})$ is different from that of every other alternative, i.e., $\bar{t}(f(\bar{t})) \neq \bar{t}(x)$ for all $x \neq f(\bar{t})$. It follows from Theorem 5.1 that an almost everywhere IC mechanism will be IC on $(t, \bar{t})$ for all $t \in V(\overline{\mathcal{D}})$. In other words, one does not have to check IC on such pairs.

Corollary 5.1. *Let a weak ordinal domain $\overline{\mathcal{D}}$ satisfy the consistent no-restoration property, the weak-compatibility property, and the dichotomous richness property. Suppose that a mechanism $\mu = (f, p)$ is IC on $cl(V(\hat{P}, \hat{P}')) \cap V(\overline{\mathcal{D}})$ for all adjacent strict preferences $\hat{P}, \hat{P}' \in strict(\overline{\mathcal{D}})$. Let a type $\bar{t} \in V(\overline{\mathcal{D}}) \setminus strict\ V(\overline{\mathcal{D}})$. If $f(\bar{t})$ is such that $\bar{t}(f(\bar{t})) \neq \bar{t}(x)$ for all $x \in A \setminus f(\bar{t})$, then μ is IC on $V(\overline{\mathcal{D}}) \times \{\bar{t}\}$.*

The proof of Corollary 5.1 can be found in Appendix .4

For a set $\bar{T} \subseteq \mathbb{R}^n$, by $\bar{T}^o$ we denote the interior of the set $\bar{T}$ i.e. $\bar{T}^o = \{t \in \bar{T} \mid \text{ there exists } \epsilon > 0 \text{ such that } s \in \bar{T} \text{ for every } s \text{ with } d(t, s) < \epsilon\}$. By $\partial\bar{T}$ we denote the points in $\bar{T}$ that lie on the boundary of $\bar{T}$ i.e. $\partial\bar{T} = \bar{T} \setminus \bar{T}^o$. Using Theorem 5.1, we can then show that an almost everywhere IC mechanism on $\bar{T}$ is IC on $\bar{T} \times \bar{T}^o$.

Appendix

.1 Proofs of Theorem 3.1, 3.2 and 3.3

Proofs of Theorem 3.1 and 3.2 follows from proof of Theorem 3.3. So we provide only the proof of Theorem 3.3 below.

Proof. Let (f, p) be a locally IC mechanism. Consider $P, P' \in \widehat{\mathcal{D}}$, and types $t \in \hat{cl}(P), t' \in \hat{cl}(P')$. We need to show $t(f(t)) - p(t) \geq t(f(t')) - p(t')$. Since the domain $\mathcal{D}$ is a "no restoration" domian, there exists a no retoration path $\pi(P, P') = (P^1, \ldots, P^k)$ from P to P'.

Claim .1. *We have $t(f(t)) - p(t) \geq t(f(s^2)) - p(s^2)$ for all $s^2 \in \hat{cl}(P^2)$.*

Proof of this claim follows from the fact that f is IC on $\hat{cl}(P^1, P^2)$.

Claim .2. *Let $2 \leq l < k$ and suppose that*

$$t(f(t)) - p(t) \geq t(f(s^l)) - p(s^l) \text{ for all } s^l \in \hat{cl}(P^l). \tag{1}$$

Then $t(f(t)) - p(t) \geq t(f(s^{l+1})) - p(s^{l+1})$ for all $s^{l+1} \in \hat{cl}(P^{l+1})$.

Proof. Assume for contradiction that there exists $\bar{s}^{l+1} \in \hat{cl}(P^{l+1})$ such that

$$t(f(t)) - p(t) < t(f(\bar{s}^{l+1})) - p(\bar{s}^{l+1}). \tag{2}$$

f is IC on $\hat{cl}(P^l, P^{l+1})$. Therefore,

$$\bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(f(s^l)) \geq s^l(f(\bar{s}^{l+1})) - s^l(f(s^l)) \text{ for all } s^l \in \hat{cl}(P^l). \tag{3}$$

Claim .3. *$t(f(\bar{s}^{l+1})) - t(f(s^l)) > s^l(f(\bar{s}^{l+1})) - s^l(f(s^l))$ for all $s^l \in \hat{cl}(P^l)$.*

Proof. Assume for contradiction that there exist $s^l \in \hat{cl}(P^l)$ such that

$$t(f(\bar{s}^{l+1})) - t(f(s^l)) \leq s^l(f(\bar{s}^{l+1})) - s^l(f(s^l)) \tag{4}$$

By equation 1 and 2, we have $p(\bar{s}^{l+1}) - p(s^l) < t(f(\bar{s}^{l+1})) - t(f(s^l))$ for all $s^l \in \hat{cl}(P^l)$. By 4, this means $p(\bar{s}^{l+1}) - p(s^l) < s^l(f(\bar{s}^{l+1})) - s^l(f(s^l))$, or $s^l(f(s^l)) - p(s^l) < s^l(f(\bar{s}^{l+1})) - p(\bar{s}^{l+1})$. This means f violates IC from s^l to $\bar{s}^{l+1}$, which contradicts the fact that f is IC on $\hat{cl}(P^l, P^{l+1})$. ■

Claim .4. *$f(s^l) \neq f(\bar{s}^{l+1})$ for all $s^l \in \hat{cl}(P^l)$.*

Proof. Asssume for contradiction that there exists $s^l \in \hat{cl}(P^l)$ such that $f(s^l) = f(\bar{s}^{l+1})$. Sinc e$f$ is IC on $\hat{cl}(P^l, P^{l+1})$, this means $p(s^l) = p(s^{l+1})$. Plugging these values in 2, we have $t(f(t)) - p(t) < t(f(s^l)) - p(s^l)$, which violates 1. ■

Claim .5. *There does not exist $s^l \in \hat{cl}(P^l)$ such that $f(\bar{s}^{l+1})P^l f(s^l)$ and $f(s^l)P^{l+1}f(\bar{s}^{l+1})$.*

Proof. Assume for contradiction that there exist $s^l \in \hat{cl}(P^l)$ such that $f(\bar{s}^{l+1})P^l f(s^l)$ and $f(s^l)P^{l+1}f(\bar{s}^{l+1})$. Since $f(\bar{s}^{l+1})P^l f(s^l)$, we have $s^l(f(\bar{s}^{l+1})) - s^l(f(s^l)) > 0$. Again, since $f(s^l)P^{l+1}f(\bar{s}^{l+1})$, we have $\bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(f(s^l)) < 0$. Combining the two equations, we have $s^l(f(\bar{s}^{l+1})) - s^l(f(s^l)) > \bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(f(s^l))$ which contradicts 3. ■

We now come back to the proof of Claim 0.2. Partition the alternatives in $A \setminus f(\bar{s}^{l+1})$ into three sets A_1, A_2, A_3 and A_4 where $A_1 = \{x \in A \mid xP^l f(\bar{s}^{l+1}) \text{ and } xP^{l+1}f(\bar{s}^{l+1})\}, A_2 = \{x \in A \mid xP^l f(\bar{s}^{l+1}) \text{ and } f(\bar{s}^{l+1})P^{l+1}x\}, A_3 = \{x \in A \mid f(\bar{s}^{l+1})P^l x \text{ and } f(\bar{s}^{l+1})P^{l+1}x\}, A_4 = \{x \in A \mid f(\bar{s}^{l+1})P^l x \text{ and } xP^{l+1}f(\bar{s}^{l+1})\}$. In what follows, we construct a type $\tilde{s}^l \in \hat{cl}(P^l)$ in the following way. Let $\kappa_1 = max_{x \in A_1}\{\bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(x)\}, \kappa_2 = max_{x \in A_2}\{t(f(\bar{s}^{l+1})) - t(x)\}, \kappa_3 =$

$max_{x \in A_3}\{\bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(x)\}$. Clearly κ_1 is negative. We argue that κ_2 is also negative. Since $f(\bar{s}^{l+1})$ overtakes the alternatives in A_2 from P^l to P^{l+1} and the path $\pi(P, P')$ is a no restoration path, it must be that these alternatives are above $f(\bar{s}^{l+1})$ in P, that is $xP^l f(\bar{s}^{l+1})$ for all $x \in A_2$. This, in particular means κ_2 is negative. Consider a type $\tilde{s}^l \in \hat{cl}(P^l)$ such that $max(\kappa_1, \kappa_2) < \tilde{s}^l(f(\bar{s}^{l+1})) - \tilde{s}^l(x) < 0$ for all $x \in A_1 \cup A_2$, $\tilde{s}^l(f(\bar{s}^{l+1})) - \tilde{s}^l(x) > \kappa_3$, and $\tilde{s}^l(x)$ is arbitrary(subject to the constraints by the preference P^l) for all $x \in A_4$. It is left to the reader that such a type $\tilde{s}^l$ can be found in $\hat{cl}(P^l)$. By Claim .5 and .4, $f(\tilde{s}^l) \notin A_4 \cup \{f(\tilde{s}^l)\}$. We show that for every possible value of $f(\tilde{s}^l)$ in $A_1 \cup A_2 \cup A_3$, we lead to a contradiction.

Case 1. Suppose $f(\tilde{s}^l) \in A_1$. Then, by the construction of $\tilde{s}^l, \tilde{s}^l(f(\bar{s}^{l+1})) - \tilde{s}^l(f(\tilde{s}^l)) > \kappa_1 \geq \bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(f(\tilde{s}^l))$, which is a contradiction to 3.

Case 2. Suppose $f(\tilde{s}^l) \in A_2$. Then, by the construction of $\tilde{s}^l, \tilde{s}^l(f(\bar{s}^{l+1})) - \tilde{s}^l(f(\tilde{s}^l)) > \kappa_2 \geq t(f(\bar{s}^{l+1})) - t(f(\tilde{s}^l))$, which is a contradiction to Claim .3.

Case 3. Suppose $f(\tilde{s}^l) \in A_3$. Then, by the construction of $\tilde{s}^l, \tilde{s}^l(f(\bar{s}^{l+1})) - \tilde{s}^l(f(\tilde{s}^l)) > \kappa_3 \geq \bar{s}^{l+1}(f(\bar{s}^{l+1})) - \bar{s}^{l+1}(f(\tilde{s}^l))$, which is a contradiction to 3. ■

Since ,$2 \leq l < k$ is arbitrary, by Claim .2, it must be that $t(f(t)) - p(t) \geq t(f(t')) - p(t')$. This completes the proof of the Theorem.

■

.2 Proofs of Theorem 4.1 and 4.2

Proof of Theorem 4.1 follows from proof of Theorem 4.2. So we provide only the proof of Theorem 4.2.

Proof. Let (f, p) be a locally IC mechanism. Let $t \in cl(V(\widehat{\mathcal{D}}))$ and $t' \in strictcl(V(\widehat{\mathcal{D}}))$. Then there exists $P, P' \in \widehat{\mathcal{D}}$, and types $t \in cl(P), t' \in \hat{cl}(P')$. We need to show $t(f(t)) - p(t) \geq t(f(t')) - p(t')$. Since the domain $\mathcal{D}$ satisfies consistent no restoration property, there exists a consistent no retoration path $\pi(P, P') = (P^1, \ldots, P^k)$ from P to P'.

Since $cl(P, P')$ is convex, it follows from Carroll (2012) that $t(f(t)) - p(t) \geq t(f(s^2)) - p(s^2)$ for all $s^2 \in \hat{cl}(P^2)$.

It is sufficient to prove the following:

Claim .6. *Let $2 \leq l < k$ and suppose that*

$$t(f(t)) - p(t) \geq t(f(s^l)) - p(s^l) \text{ for all } s^l \in \hat{cl}(P^l). \tag{5}$$

Then $t(f(t)) - p(t) \geq t(f(s^{l+1})) - p(s^{l+1})$ *for all* $s^{l+1} \in \hat{cl}(P^{l+1})$.

Proof. Assume for contradiction that there exists $\bar{s}^{l+1} \in \hat{cl}(P^{l+1})$ such that

$$t(f(t)) - p(t) < t(f(\bar{s}^{l+1})) - p(\bar{s}^{l+1}). \tag{6}$$

Assume that $cP^l a$ and $aP^{l+1}c$ for some $a, c \in A$. Then it must be the case that $f(\bar{s}^{l+1}) = a$. Since, $\pi(P, P') = (P^1, \ldots, P^k)$ is a consistent no restoration path from P to P', $L(a, P^1) \subseteq L(a, P^l)$ and there exists $b \in A$ such that $\bar{U}(a, P^1) = \bar{U}(b, P^l)$ for some $b \in A$. In what follows, we construct a type $\tilde{s}^l \in cl(P^l)$ in the following way. Consider a type $\tilde{s}^l \in cl(P^l)$ such that $\tilde{s}^l(x) = \tilde{s}^l(y)$ for all $x, y \in \bar{U}(b, P^l)$, $\tilde{s}^l(u) = \tilde{s}^l(v)$ for all $u, v \in L(b, P^l)$, and $\tilde{s}^l(a) - \tilde{s}^l(u) > max_{v \in L(b,P^l)}\{\bar{s}^{l+1}(a) - \bar{s}^{l+1}(v)\}$ for some $u \in L(b, P^l)$. $\tilde{s}^l(a) - \tilde{s}^l(z) > \bar{s}^{l+1}(a) - \bar{s}^{l+1}(z)$ for all $z \in L(b, P^l)$. By the construction of $\tilde{s}^l$ and the fact that $cl(P, P')$ is convex, it must be the case that $f(\tilde{s}^l) \in \bar{U}(b, P^l)$. By the construct of $\tilde{s}^l$ and the fact that the path $\pi(P, P') = (P^1, \ldots, P^k)$ is a consistent no restoration path, it follows that $\tilde{s}^l \in cl(P^1)$. Since, $t, \tilde{s}^l \in cl(P)$, (f, p) must be IC on (t, s). Also, since $f(\tilde{s}^l) \in \bar{U}(b, P^l)$, and by the construction of $\tilde{s}^l$, it must be the case that $\tilde{s}^l(a) - \tilde{s}^l(f(\tilde{s}^l)) = 0$. Since, $f(\tilde{s}^l) \in \bar{U}(b, P^l)$, it must be the case that $t(a) - t(f(\tilde{s}^l)) < 0$. Hence we get $t(a) - t(f(\tilde{s}^l)) < \tilde{s}^l(a) - \tilde{s}^l(f(\tilde{s}^l))$. Since $t(a) - t(f(\tilde{s}^l)) < \tilde{s}^l(a) - \tilde{s}^l(f(\tilde{s}^l))$ and (f, p) is IC on $(t, \tilde{s}^l)$ and $(\tilde{s}^l, \bar{s}^{l+1})$, it leads to a contradiction to 6. This completes the proof of the claim. ■

This completes the proof of the theorem. ■

.3 Proof of Theorem 5.1

Proof. Assume for contradiction that there exists $t \in \bar{T}$ such that $\mu = (f, p)$ is not IC on $(t, \bar{t})$ i.e., $t(f(t)) - p(t) < t(f(\bar{t})) - p(\bar{t})$. By condition (ii) of the Theorem and the fact that μ is almost everywhere IC, it must be the case that μ is IC on $(\bar{t}, \hat{t})$ and $(\hat{t}, \bar{t})$. Since $\hat{t}(f(\bar{t})) - \hat{t}(x) > \bar{t}(f(\bar{t})) - \bar{t}(x)$ for all $x \in A \setminus \{f(\bar{t})\}$, it follows that $f(\hat{t}) = f(\bar{t})$. Since μ is IC on $(\bar{t}, \hat{t})$ and $(\hat{t}, \bar{t})$, it must be the case that $p(\hat{t}) = p(\bar{t})$. By our assumption for contradiction, we have $t(f(t)) - p(t) < t(f(\bar{t})) - p(\bar{t})$. Since $f(\hat{t}) = f(\bar{t})$ and $p(\hat{t}) = p(\bar{t})$, it follows that $t(f(t)) - p(t) < t(f(\hat{t})) - p(\hat{t})$. Since $\hat{t} \in strict(\bar{(T)})$, which is a contradiction to μ being almost everywhere IC. This completes the proof of the Theorem. ■

.4 PROOF OF COROLLARY 5.1

Proof. The proof of this Corollary is a direct application of Theorem 5.1. Since $\bar{t}(f(\bar{t})) \neq \bar{t}(x)$ for all $x \in A$, there exists $\hat{t} \in strict(V(\bar{(}\mathcal{D})))$ such that $\hat{t}(f(\bar{t})) - \hat{t}(x) > \bar{t}(f(\bar{t})) - \bar{t}(x)$ for all $x \in A \setminus \{f(\bar{t})\}$ and μ is IC on $(\hat{t}, \bar{t})$. Hence by Theorem 5.1, it follows that μ is IC on $V(\bar{(}\mathcal{D})) \times \{\bar{t}\}$. This completes the proof of this Corollary. ■

REFERENCES

ARMSTRONG, M. (2000): "Optimal multi-object auctions," *The Review of Economic Studies*, 67, 455–481.

CARROLL, G. (2012): "When are local incentive constraints sufficient?" *Econometrica*, 80, 661–686.

FUDENBERG, D. AND J. TIROLE (1991): "Game theory mit press," *Cambridge, MA*, 86.

MISHRA, D., A. PRAMANIK, AND S. ROY (2016): "Local incentive compatibility with transfers," *Games and Economic Behavior*, 100, 149–165.

SATO, S. (2013): "A sufficient condition for the equivalence of strategy-proofness and nonmanipulability by preferences adjacent to the sincere one," *Journal of Economic Theory*, 148, 259–278.

VOHRA, R. V. (2011): *Mechanism design: a linear programming approach*, vol. 47, Cambridge University Press.