Separability and decomposition in mechanism design with transfers *

Debasis Mishra, Swaprava Nath, and Souvik Roy †

August 9, 2017

Abstract

In private values quasi-linear environment, we consider problems where allocation decisions along multiple components have to be made. Every agent has additively separable valuation over the components. We show that every unanimous and dominant strategy implementable allocation rule in this problem is a *component-wise weighted utilitarian* rule, which assigns non-negative weight vectors to agents in each component and chooses an alternative in each component by maximizing the weighted sum of valuations in that component. A corollary of our result is that every unanimous and dominant strategy implementable allocation rule can be *almost* decomposed (modulo tie-breaking) into dominant strategy implementable allocation rules along each component.

JEL Codes: D82, D44, D47

Keywords: separable types, affine maximizer, Roberts' theorem.

*We thank two anonymous referees and an associate editor for detailed comments that have significantly improved the paper. We are also grateful to Arunava Sen and participants at a seminar in Indian Statistical Institute for useful discussions. Nath is supported by the Fulbright-Nehru Postdoctoral Fellowship, 2015-17.

†Debasis Mishra, Indian Statistical Institute, New Delhi, Email: `dmishra@isid.ac.in`; Swaprava Nath, Indian Institute of Technology (IIT), Kanpur, Email: `swapravn@cs.cmu.edu`; Souvik Roy, Indian Statistical Institute, Kolkata, Email: `souvik_roy2004@yahoo.com`

1 Introduction

Suppose a planner needs to choose various kinds of public goods for a city: a stadium location among a set of locations, a school with a particular medium of teaching among a set of teaching mediums, a mode of public transport among various modes of transport etc. Transfers (in the form of taxes or subsidies) are allowed and agents have (i) additively separable value over different kinds of public good and (ii) quasi-linear utility over transfers. How should the planner choose different kinds of public good?

We investigate the class of dominant strategy incentive compatible mechanisms in such environments - our focus is exclusively on deterministic mechanisms. These kind of settings are typically characterized by alternatives which have multiple dimensions or components - kinds of public good in the above example. Agents have additively separable valuations, which are defined by a valuation function over alternatives in each component, and then, the value for an alternative is just the sum of valuations of individual components. For instance, in the example, value for an alternative which consists of (a^1) a stadium in downtown, (a^2) a French medium school, and (a^3) a monorail transport system is calculated as the sum of values of a^1, a^2, and a^3.

A natural solution to such a problem in quasi-linear private values environment is to use a mechanism in the Groves class of mechanisms. Though efficient, Groves mechanisms are known to have undesirable features like not being able to cover cost and not maximizing surplus of participating agents. This is the motivation for studying the set of all incentive compatible mechanisms in this setting.

In a strategic voting environment, where transfers are precluded, such separability usually implies decomposition of the mechanism, i.e., we can employ a separate mechanism on each component that only considers preference information on that component and then the decisions on various components can be aggregated (Barberà et al., 1991; Barbera et al., 1993; Le Breton and Sen, 1999; Le Breton and Weymark, 1999; Weymark, 1999; Svensson and Torstensson, 2008) - usually, these results require additional mild conditions such as *unanimity* or *onto-ness*.

We show that such decomposition is not possible in our setting with transfers. In particular, in quasi-linear environment with additively separable valuations over components, an *onto* dominant strategy implementable allocation rule is not necessarily decomposable.[1] This is not surprising because introduction of transfers usually expands the set of incentive compatible mechanisms, and in the process, we get new mechanisms which are not decom-

[1] An allocation rule is dominant strategy implementable if there exists a payment function such that the corresponding mechanism is dominant strategy incentive compatible.

posable. Our main result shows that the decomposability result can be restored if we assume *unanimity* of the allocation rule. In particular, we show that every dominant strategy implementable and unanimous allocation rule must be a *component-wise weighted utilitarian (CWU)*. Conversely, every CWU allocation rule is unanimous and implementable under a mild tie-breaking condition. Unanimity requires that if *all* the agents have higher valuation for an alternative than *every* other alternative, then the allocation rule must choose that alternative.

A CWU allocation rule specifies for each component a vector of non-negative weights (not all zero) for the agents. Then, for each component, it computes weighted sum of valuations of agents and chooses an alternative that maximizes this weighted sum. Because of tie-breaking issues, this does not give us complete decomposability of the allocation rule but implies decomposability in a generic sense.

It is worth noting that in strategic voting environment (where transfers are not used), unanimity and ontoness are equivalent conditions under dominant strategy incentive compatibility.[2] This is no longer the case in quasilinear type space. This explains that we cannot get (almost) decomposability of onto dominant strategy implementable allocation rules but it is possible to decompose unanimous dominant strategy implementable allocation rules. We give two more conditions on allocation rules, both equivalent to unanimity *under dominant strategy implementability*, under which our result goes through: (1) *Pareto.* this requires that if there is a pair of alternative a and b such that valuation for *every* agent is lower for a then for b, then a should not be chosen;[3] (2) *neutrality.* this requires that valuations of alternatives are permuted (using a permutation over alternatives), then the outcome at the permuted valuation profile is the permutation of the outcome at the original profile.

Unanimity is a sufficient condition for our main result - we give an example to show that there are non-unanimous, onto, and implementable allocation rules that cannot be decomposed. We feel that unanimity is a natural condition in the examples we consider. It is a compelling normative axiom, extensively used in the social choice theory literature (Barbera, 2010). It imposes a minimal form of unbiasedness for the social planner - if the society agrees to some alternative being the best, then the social planner cannot choose a different alternative. Of course, unanimity is silent on what a social planner can do if the society cannot agree on a common best alternative. Potentially, the planner can discriminate among alternatives (say, based on their costs or other characteristics) at such profiles. Our result says that no implementable and unanimous allocation rule can do that.

[2] This result requires some richness of the type space - see for instance Reny (2001).

[3] We remind that an alternative a here consists of multiple components $a^1, a^2, \dots$ and the valuation for an alternative is sum of the valuations over individual components.

We also show that for every CWU allocation rule, there is a decomposed payment rule (i.e., total payment is the sum of the payments of each component) that implements such CWU allocation rule. Standard revenue equivalence result then pins down the entire class of payment rules that can implement CWUs - these are payment rules which will differ from the decomposed payment rule by an additive constant, which does not depend on agent's own valuation (Chung and Olszewski, 2007). However, this implies that there may be other non-decomposable payment rules that may implement a CWU allocation rule. Hence, we cannot be assured of the decomposition of *every* dominant strategy incentive compatible *mechanism* even though the allocation rule can be decomposed.

Our result contributes to two strands of literature. As discussed earlier, it complements the literature on decomposition of mechanisms in separable environments without transfers - for instance, see Barberà et al. (1991) for the case of two alternatives in each component, Barbera et al. (1993) for multidimensional single-peaked preferences, Le Breton and Sen (1999) for a general result covering many preference domains, Le Breton and Weymark (1999) for a model with a continuum of alternatives, Svensson and Torstensson (2008) for a component-wise dictatorship result when preferences in each component is unrestricted, and Weymark (1999) for a for a detailed survey with a general result. Probably, the closest model in this literature is the one of Svensson and Torstensson (2008), who consider the public good provision problem but do *not* allow for transfers. As a result, every incentive compatible and onto mechanism in their model is a *component-wise dictatorship* - see also Le Breton and Sen (1999). Our result can be thought as a generalization of their result in the multiple public good provision model with transfers but with quasilinear utility. Indeed, transfers are a natural tool for a planner in this environment and our result clarifies the exact nature of possibility and decomposition in the presence of transfers.

Next, it contributes to a new literature in mechanism design about extending a seminal result of Roberts (1979). In an important paper, Roberts (1979) showed that every onto and dominant strategy implementable allocation rule in an unrestricted type space in quasi-linear utility environment must be an *affine maximizer*.[4] Several authors establish simpler proofs (Lavi et al., 2009; Dobzinski and Nisan, 2009; Vohra, 2011) or extend his result to other restricted type space - for instance, Mishra and Sen (2012) show that Roberts theorem holds in certain bounded type spaces if neutrality is assumed; Carbajal et al. (2013) extend Roberts' result to infinite set of alternatives; Jehiel et al. (2008) extend it to certain interdependent valuation environment; Nath and Sen (2015) extend Roberts' result to a specific private good

[4] This result requires at least three alternatives. If there are two alternatives, there are implementable allocation rules that are not affine maximizers. Marchant and Mishra (2015) provide a complete characterization for the two-alternatives case.

allocation problem assuming an additional condition called *non-bossiness*. All these papers find new type spaces where implementability along with other reasonable conditions imply affine maximization. Mishra and Quadir (2014) provide an analogue of Roberts' theorem in the private values single object auction environment. Their result captures a larger class of allocation rules than just affine maximizers. Similarly, Dobzinski and Nisan (2011) show the existence of non-affine maximizers in multi-unit auction environment.

The richness of type space required in these papers is absent in the type space we consider. An important aspect of our type space (due to additive separability) is that altering the valuation of one alternative involves modifying values for one or more components. This in turn changes the valuations of many alternatives. As a result, we lose the freedom of modifying valuations along one of the alternatives without disturbing the valuations of other alternatives - this is a crucial element of getting to Roberts' result, which is exploited in the literature. As a consequence of absence of this richness, we get our *new* class of implementable allocation rules - CWU allocation rules. Further, none of the existing results imply our result and the set of CWU allocation rules captures a *larger* class of allocation rules than the affine maximizers satisfying unanimity in Roberts (1979) - we discuss this issue in detail later.

The rest of the paper is organized as follows. We introduce our separable type space in Section 2 along with a motivating example in Section 2.1. The main result is presented in Section 3 followed by some remarks in Section 3.2. Section 3.3 introduces a variety of axioms that can replace unanimity in our main result. The proof of the main result is given in Section 4. All the missing proofs are provided in an Appendix at the end. We conclude the paper with some discussions in Section 5.

2 The Separable Type Space

We begin by formally defining our model of separable type space with transfers. Let $N = \{1, \ldots, n\}$ be the set of agents. Agents need to decide on various kinds of public goods. Let $J = \{1, \ldots, m\}$ denote the kinds of public goods. We will also call the elements of J *components*. An alternate interpretation of our model is a m-period "planning problem", where a planner decides on the public good to provide in each of the $1, \ldots, m$ periods. Here, the components are periods and the decision for each of the periods is made simultaneously before the start of the periods.[5]

[5] As will be clear later, if we assume this interpretation of our model, we do not allow for any *dynamics* across periods. This is because we will assume that the values of the agents for alternatives in each period are known. A single decision is taken before the start of the periods about the public good to be provided in each of the periods.

For each component $j \in J$, let A^j be the (finite) set of available public goods of kind j. We assume that for all $j \in J$, we have $|A^j| \geq 3$. A public good along each component must be chosen. Hence, the set of alternatives is denoted by

$$A := A^1 \times \ldots \times A^m.$$

An alternative $a \in A$ will be equivalently denoted as $(a^1, \ldots, a^m)$, where for every j, $a^j \in A^j$ is the alternative of component j in a.

The *type* of an agent $i \in N$ is a valuation function $v_i : A \to \mathbb{R}$. We assume that the valuations of agents are separable in the following sense.

DEFINITION 1 *A valuation function v_i of agent i is* **additively separable** *if there exists a collection of vectors $(v_i^1, \ldots, v_i^m)$, where $v_i^j \in \mathbb{R}^{|A^j|}$ for each $j \in J$, such that for every $a \in A$,*

$$v_i(a) = \sum_{j \in J} v_i^j(a^j).$$

Here, v_i^j will be referred to as the valuation of agent i for the alternatives in the j-th component.

A profile of valuation functions $(v_1, \ldots, v_n)$ will be denoted by $\mathbf{v}$. Further, for every $i \in N$, we will sometimes denote its valuation v_i as $(v_i^1, \ldots, v_i^m)$.

The type space of each agent can be restricted by restricting the values of the valuation functions along each component. In general, we assume that for every $i \in N$ and for every $j \in J$, the valuations of alternatives in the j-component lie in $\mathbb{R}_{++}$.[6] The space of valuations for agent i for component j is thus $\mathbb{R}_{++}^{|A^j|}$. Hence, the **type space** of each agent i is given by

$$V \equiv \mathbb{R}_{++}^{|A^1|} \times \ldots \times \mathbb{R}_{++}^{|A^m|}.$$

It is important to note that because of the additive nature of the valuation, the type space is a strict subset of $\mathbb{R}_{++}^{|A|}$ - for instance, the following type from $\mathbb{R}_{++}^{|A|}$ can never be in our type space: for some $a \in A$, $v_i(a) > M$ for some large M and $v_i(b)$ is arbitrarily close to zero for all $b \neq a$. This is because if $v_i(a) > M$, then there is some $j \in J$ such that $v_i^j(a^j) > \frac{M}{m}$. But, then there is some $b \neq a$ but $b^j = a^j$ such that $v_i(b) > \frac{M}{m}$, which contradicts the fact that $v_i(b)$ is arbitrarily close to zero.

[6] We denote the set of non-positive, non-negative and positive real numbers by $\mathbb{R}_-$, $\mathbb{R}_+$ and $\mathbb{R}_{++}$ respectively. We emphasize that it is not necessary to exclude zero valuations for our results to hold. However, our proof requires the flexibility to be able to increase the values to an arbitrary amount. In that regard, the valuations can be drawn from an arbitrary open interval (x, ∞), where $x \in \mathbb{R}_- \cup \{-\infty\}$. We require the fact that such intervals are open due to technical reasons.

We refer to our type space as **separable type space** - without using the "additively" qualifier before separability. Sometimes, we will look at the valuations of all the agents $\mathbf{v}$ restricted to a particular component j. In that case, we will refer this restricted profile along the j-th component as v^j. The separable type space automatically induces a set of such restricted valuation profiles along the j-th component, which is $\mathbb{R}^{|A^j| \times n}$.

We assume quasilinear preferences. Suppose alternative $a \equiv (a^1, \ldots, a^m)$ is chosen and agent i gets a transfer of p_i, then the net utility of agent i with valuation v_i is

$$\sum_{j \in J} v_i^j(a^j) + p_i.$$

An **allocation rule** is a map $F : V^n \to A$. At any valuation profile $\mathbf{v}$, the alternative chosen for component j by the allocation rule F is denoted by $F^j(\mathbf{v})$.

A **payment function** of agent i is a map $p_i : V^n \to \mathbb{R}$. Note that the transfer of agent i need *not* be separable across components.

A mechanism is an allocation rule and a collection of payment functions: $(F, p_1, \ldots, p_n)$.

Definition 2 *An allocation rule F is* **implementable** *(in dominant strategies) if there exists $(p_1, \ldots, p_n)$ such that for every $i \in N$, for every $v_{-i} \in \times_{k \neq i} V_k$, for every $v_i, v_i' \in V$, we have*

$$\sum_{j \in J} v_i^j(F^j(v_i, v_{-i})) + p_i(v_i, v_{-i}) \geq \sum_{j \in J} v_i^j(F^j(v_i', v_{-i})) + p_i(v_i', v_{-i}).$$

In this case, we say $(F, p_1, \ldots, p_n)$ is a **dominant strategy incentive compatible (DSIC)** *mechanism.*

From a planner's perspective, it can have a separate allocation rule for each component to decide on the alternative of that component. It chooses alternatives in each component using a *marginal allocation rule* on that component. Formally, a **marginal allocation rule** for component j is a map $f^j : \mathbb{R}_{++}^{|A^j| \times n} \to A^j$. The planner can then aggregate the decisions on each component j using the marginal allocation rule f^j. Decomposability of an allocation rule requires the existence of such marginal allocation rules over each component.

Definition 3 *An allocation rule F is* **decomposable** *if there exists a collection of marginal allocation rules $(f^1, \ldots, f^m)$ such that for all $\mathbf{v}$, we have*

$$F^j(\mathbf{v}) = f^j(v_1^j, \ldots, v_n^j) \quad \forall\ j \in J.$$

Given the additive separability of valuations, it is natural to suspect that an implementable allocation rule may be decomposable. This is the main question we aim to answer:

Is every implementable allocation rule in the separable type space decomposable?

2.1 A Motivating Example

We start with an example which shows that implementability need not imply decomposability, and this in turn motivates us to ask the same question under an additional condition. Our example has two agents and two components. Component 1 has alternatives $\{a^1, b^1, c^1\} = A^1$ and component 2 has alternatives $\{a^2, b^2, c^2\} = A^2$. Hence, $A = A^1 \times A^2$. We will define an implementable allocation rule in this problem which is not decomposable.

Define a map $\kappa : A \to \mathbb{R}$ as follows: $\kappa(a^1, a^2) = 0.95, \kappa(b^1, b^2) = 1$ and $\kappa(x) = 0$ for all alternatives $x \notin \{(a^1, a^2), (b^1, b^2)\}$. Now, at every valuation profile $\mathbf{v} \equiv (v_1, v_2)$,

$$F(\mathbf{v}) \in \arg\max_{x \in A} \big[v_1(x) + v_2(x) + \kappa(x)\big],$$

where for every $i \in N$, $v_i(x) = v_i^1(x^1) + v_i^2(x^2)$. We break ties by using a strict ordering over the set of alternatives - in particular, in case of multiple alternatives maximizing, we pick the highest ranked alternative according to this tie-breaking ordering which belongs to the set of maximizers. Such allocation rules are called *affine maximizers* (we give a formal definition later). They are implementable using payment rules which are general versions of Groves payments (Roberts, 1979).

We argue that such an allocation rule is not decomposable. Indeed, if F was decomposable, then the change in valuations in one component would not change the outcome in the other component. We show that this property is violated. Consider the following pair of valuation profiles which differs in the valuation in the second component. Denote the first valuation profile as $\mathbf{v} \equiv (v_1, v_2)$ and the second one as $\bar{\mathbf{v}} \equiv (\bar{v}_1, \bar{v}_2)$. In particular, all valuations in both the profiles are arbitrarily close to zero except agent 1's valuation in $\bar{v}_1$ on alternative a^2 of component 2, which is set at

$$\bar{v}_1^2(a^2) = 0.1.$$

By construction,

$$F(\mathbf{v}) = (b^1, b^2) \text{ and } F(\bar{\mathbf{v}}) = (a^1, a^2).$$

Note that $\mathbf{v}$ and $\bar{\mathbf{v}}$ only differ in the valuation in the second component, but the outcome of F in the first component changed from $\mathbf{v}$ to $\bar{\mathbf{v}}$. This shows that F is implementable but not decomposable. Note that the set of valuation profiles at which F is not decomposable has a non-negative Lebesgue measure, which hints that the violation of decomposability is not a result of some particular tie-breaking rule. Also, note that F is onto.

Our main result below shows that in the class of additively separable valuations, if the allocation rule is *unanimous*, then implementability implies decomposability in a generic sense (i.e, except for some tie-breaking issues).

3 Component-wise weighted utilitarianism

In this section, we will state our main decomposition result and provide some remarks on it. The main result will say that implementability along with an additional condition will imply the following class of allocation rules. Further, these allocation rules are *almost* decomposable (modulo tie-breaking).

Definition 4 *An allocation rule F is a* **component-wise weighted utilitarian (CWU)** *allocation rule if for every $j \in J$, there exists weight vectors $\lambda^j \in \mathbb{R}^n_+ \setminus \{\mathbf{0}\}$ such that for all $\mathbf{v}$, we have*

$$F^j(\mathbf{v}) \in \arg\max_{a^j \in A^j} \sum_{i \in N} \lambda^j_i v^j_i(a^j).$$

We now describe the main result after stating the additional condition required for the result.

3.1 The main result

The additional condition required can be *any* of the following two axioms. We describe them next. Later, we provide more equivalent additional conditions which also give our main result.

Unanimity. Unanimity requires that if an alternative has higher value than every other alternative for all the agents, then it should be chosen. Formally, unanimity requires the following.

Definition 5 *An alternative a is* **unanimous** *at a valuation profile $\mathbf{v}$ if for every alternative $b \neq a$*

$$\sum_{j \in J} v^j_i(a^j) > \sum_{j \in J} v^j_i(b^j) \;\forall\; i \in N.$$

An allocation rule F satisfies **unanimity** *if for every $a \in A$ and for every $\mathbf{v}$ such that a is unanimous at $\mathbf{v}$, we have $F(\mathbf{v}) = a$.*

Another way to define unanimity is to define it at *component-level*, i.e., if valuations of all the agents for an alternative $a^j \in A^j$ is the highest among all the alternatives in component j, then a^j must be chosen by the allocation rule in that component. Later, we discuss such component-level unanimity and its connection to unanimity in Definition 5 in detail.

Pareto. Pareto requires that if an alternative a has lower value than some other alternative b for every agent, then a *cannot* be chosen. Formally, Pareto requires the following.

DEFINITION **6** *An alternative a is* **dominated** *at a valuation profile* $\mathbf{v}$ *if there exists an alternative $b \neq a$ such that*

$$\sum_{j \in J} v_i^j(a^j) < \sum_{j \in J} v_i^j(b^j) \;\forall\; i \in N.$$

An allocation rule F satisfies **Pareto** *if for every $a \in A$ and for every $\mathbf{v}$ such that a is dominated at $\mathbf{v}$, we have $F(\mathbf{v}) \neq a$.*

It is not difficult to see that Pareto implies unanimity.[7]

Our main result is the following, whose proof is postponed to Section 4.

THEOREM **1** *Suppose F is an allocation rule defined on a separable type space. Consider the following statements.*

1. *F is implementable and satisfies unanimity.*

2. *F is implementable and satisfies Pareto.*

3. *F is a CWU allocation rule.*

Statements (1) and (2) are equivalent, and each of them implies Statement (3).

A CWU allocation rule satisfies Pareto and unanimity, but it may fail to be implementable if ties are not broken carefully. We skip a formal example to illustrate this fact - one can easily verify that some dictatorships can be manipulated at valuation profiles where the dictator has more than one highest valued alternative. The issue is similar to the non-implementability of affine maximizers of Roberts (1979) if ties are not broken carefully - see detailed discussions in Carbajal et al. (2013). We introduce a mild tie-breaking condition that ensures implementability of CWU allocation rules.

DEFINITION **7** *A CWU allocation rule F with weights $\{\lambda^j\}_{j \in J}$ satisfies* **independence of non-influential agents (INA)** *if for every $j \in J$ and for every $i \in N$ with $\lambda_i^j = 0$, we have $F^j(\mathbf{v}) = F^j(\mathbf{v}')$ if $\mathbf{v}$ and $\mathbf{v}'$ differ only in the valuation of agent i.*

[7] The Pareto property of an allocation rule should not be confused with the Pareto efficiency of a *mechanism* in quasilinear settings. Pareto efficiency of a mechanism implies that the underlying allocation rule must be *utilitarian*, i.e., a CWU allocation rule with all weights equal to 1. On the other hand, Pareto property of allocation rule is much weaker than this and covers many allocation rules.

Now, we define a payment function that implements a CWU allocation rule satisfying INA. First, we define a payment function for each component j as follows: for every $i \in N$ and for every $\mathbf{v}$

$$p_i^j(\mathbf{v}) = \begin{cases} \frac{1}{\lambda_i^j} \sum_{k \neq i} \lambda_k^j v_k^j(F^j(\mathbf{v})) & \text{if } \lambda_i^j > 0 \\ 0 & \text{if } \lambda_i^j = 0 \end{cases}$$

Now, we can define a payment function for agent i as: for every $\mathbf{v}$,

$$p_i(\mathbf{v}) = \sum_{j \in J} p_i^j(\mathbf{v}).$$

The lemma below shows that this payment function implements a CWU satisfying INA. The proof follows standard arguments and is given in the Appendix at the end.

LEMMA **1** *If F is a CWU allocation rule satisfying INA, then $(p_1, \ldots, p_n)$ implements F.*

Unfortunately, a CWU allocation rule is not completely decomposable. The issue is the existence of multiple maximizers at certain valuation profiles, i.e., how we handle tie-breaking in the maximization above. Since Theorem 1 cannot pin down how to select an alternative for a component if there are multiple maximizers, it leaves room for the failure of decomposability. We give an example to illustrate this.

Consider two agents and two components. The alternatives in component 1 are a^1, b^1, c^1. Consider a valuation profile whose component 1 has the following valuations:

$$v_1^1(a^1) = 2, v_1^1(b^1) = 0, v_1^1(c^1) = 1,$$

$$v_2^1(a^1) = 4, v_2^1(b^1) = 5, v_2^1(c^1) = 1.$$

The CWU weights for component 1 are $\lambda_1^1 = 1, \lambda_2^1 = 2$. Both a^1 and b^1 maximize the weighted sum in the CWU. By Lemma 1, we can pick either a^1 or b^1 on this component, and the CWU will still be implementable (note that the CWU trivially satisfies INA in this case). In particular, we can choose a^1 on component 1 when valuations of other components is v^2 but b^1 when valuations of other components is $\bar{v}^2 \neq v^2$. This will violate our definition of decomposability. However, we argue that such failure of decomposability happens only in these kind of valuation profiles which has ties in the CWU maximization, and forms a set of measure zero over the set of all valuation profiles.

This example motivates us to consider a slightly weaker version of decomposability.

DEFINITION **8** *An allocation rule F is* **decomposable almost everywhere** *if there exists a collection of marginal allocation rules $(f^1, \ldots, f^m)$, where $f^j : \mathbb{R}_{++}^{|A^j| \times n} \to A^j$ for every*

$j \in J$, such that for every $j \in J$ there exists $X^j \subseteq \mathbb{R}_{++}^{|A^j|\times n}$ with $\mathbb{R}_{++}^{|A^j|\times n} \setminus X^j$ having zero measure and for all v^{-j}, we have

$$F^j(v^j, v^{-j}) = f^j(v^j) \quad \forall\ v^j \in X^j.$$

A simple corollary of Theorem 1 is the following almost decomposability result.

Corollary **1 (Almost Decomposition)** *Suppose F is an allocation rule defined on a separable type space. If F is implementable and satisfies unanimity, then it is decomposable almost everywhere.*

Proof: By Theorem 1, for every $j \in J$, there exists weight vectors $\lambda^j \in \mathbb{R}_+^n \setminus \{\mathbf{0}\}$ such that for all $\mathbf{v}$, we have

$$F^j(\mathbf{v}) \in \arg\max_{a^j \in A^j} \sum_{i \in N} \lambda_i^j v_i^j(a^j).$$

Fix a component $j \in J$. Let $X^j \subseteq \mathbb{R}_{++}^{|A^j|\times n}$ be defined as follows:

$$X^j := \{v^j \in \mathbb{R}_{++}^{|A^j|\times n} : |\arg\max_{a^j \in A^j} \sum_{i \in N} \lambda_i^j v_i^j(a^j)| = 1\}.$$

Note that $\mathbb{R}_{++}^{|A^j|\times n} \setminus X^j$ has zero (Lebesgue outer) measure. To see this, for any $a^j, b^j \in A^j$ with $a^j \neq b^j$, define $I(a^j, b^j) := \{v^j : \sum_{i \in N} \lambda_i^j v_i^j(a^j) = \sum_{i \in N} \lambda_i^j v_i^j(b^j)\}$. It is clear that $I(a^j, b^j)$ has lower dimension than X^j, and hence, has measure zero. Since $\mathbb{R}_{++}^{|A^j|\times n} \setminus X^j \subseteq \Big(\cup_{a^j, b^j} I(a^j, b^j)\Big)$, we conclude that it has measure zero.

Now, fix any $\bar{v}^{-j}$, and consider the restriction of F to the j-th component by fixing $\bar{v}^{-j}$. In particular, define $f^j : \mathbb{R}^{|A^j|\times n} \to A^j$ as follows. For every $v^j \in \mathbb{R}_{++}^{|A^j|\times n}$, let

$$f^j(v^j) := F^j(v^j, \bar{v}^{-j}).$$

By construction, for every $v^j \in X^j$ and any v^{-j}, v'^{-j}, we have

$$F^j(v^j, v^{-j}) = F^j(v^j, v'^{-j}).$$

Hence, for all $v^j \in X^j$ and all v^{-j}, we have

$$f^j(v^j) = F^j(v^j, v^{-j}),$$

establishing the claim that F is decomposable almost everywhere. ■

3.2 Remarks

Parallels in strategic voting. Our result can be thought of as a generalization of decomposability results in strategic voting literature. In the strategic voting models, allocation rules have to be implemented *without* transfers, and preferences of agents are *separable* orderings over alternatives.[8] The main results in that literature say that every *onto* and implementable allocation rule is decomposable - see a very general result in (Le Breton and Sen, 1999).[9] As remarked earlier, in those models, ontoness, unanimity, and Pareto are equivalent conditions under implementability. The example in Section 2.1 showed that our decomposition result (Corollary 1) is not true if we use ontoness in place of unanimity or Pareto. Hence, Theorem 1 does not hold if we drop unanimity or weaken it to ontoness.

Roberts' Theorem. Roberts (1979) worked in an environment where each alternative cannot be separated into components (or alternatively, there is only one component for every alternative). He showed the following theorem. **Notation.** Below, when we write $v_i(a)$, we mean valuation of agent i for alternative a, and since alternatives do not have components in Roberts' model, this notation is clear.

Fact **1 (Roberts (1979))** *Suppose $|A| \geq 3$ and type space of each agent is $\mathbb{R}^{|A|}$ (i.e., the set of all possible type vectors), then every onto and implementable allocation rule is an* affine maximizer*, i.e., there exists weight vectors $w \in \mathbb{R}^n_+ \setminus \{0\}$ and a map $\kappa : A \to \mathbb{R}$ such that at every valuation profile $\mathbf{v}$, we have*

$$F(\mathbf{v}) \in \arg\max_{a \in A} \big[\sum_{i \in N} w_i v_i(a) + \kappa(a)\big]. \tag{1}$$

Under a mild tie-breaking condition, similar to the INA condition in Definition 7, every affine maximizer is also implementable (Mishra and Sen, 2012) - see Carbajal et al. (2013) for a complete characterization.

As we have argued earlier, in our model, type space of every agent is

$$\mathbb{R}^{|A_1|}_{++} \times \ldots \times \mathbb{R}^{|A_m|}_{++} \subsetneq \mathbb{R}^{|A|}_{++} \subsetneq \mathbb{R}^{|A|}.$$

Hence, Roberts' theorem does not apply to our separable type space. The key restriction imposed by additive separable valuations is that for every alternative a, changing the value

[8] Usually, this literature is concerned with a broader definition of separability that captures additive separability and other forms of separability.

[9] Unlike our result, this result in strategic voting model, works under various ordinal restrictions of preferences.

of alternative a also changes the values of all alternatives which have some component alternatives common with a. This destroys the richness required to get the affine maximizer result in Roberts (1979).

A smaller domain means that we get new implementable allocation rules along with the affine maximizers. For instance, consider the class of rules that we call **component-wise affine maximizers**, which is defined by having weights vectors $w^j \in \mathbb{R}^n_+ \setminus \{0\}$ for each component j and a map $\kappa^j : A^j \to \mathbb{R}$ for each j such that at every valuation profile $\mathbf{v}$, we have

$$F^j(\mathbf{v}) \in \arg\max_{a^j \in A^j} \big[\sum_{i \in N} w_i^j v_i^j(a^j) + \kappa^j(a^j)\big].$$

Component-wise affine maximizers and (overall) affine maximizers are not the same. An affine maximizer need not be decomposable (as we showed in the example in Section 2.1). But a component-wise affine maximizer is always decomposable (almost everywhere). The non-decomposability of affine maximizers stem from the fact that their κ maps are not decomposable along components.

Notice that a *unanimous* affine maximizer is a CWU allocation rule - unanimity implies that the κ terms in Equation 1 are all zero, and this gives us a CWU allocation rule where weights on each component is the same for any agent. However, CWU allocation rule has more allocation rules than unanimous affine maximizer. This is because a CWU allocation rule allows us to choose different weights for different components - $w_i^j \neq w_i^{j'}$ for any j, j'.

We describe these various classes of allocation rules and where the CWU allocation rule lies using a figure in Figure 1. As it shows, the CWU allocation rules do not capture the entire set of decomposable allocation rules - component-wise affine maximizers are a larger class of decomposable allocation rules. The set of decomposable and implementable allocation rules are shown to be superset of the set of component-wise affine maximizers - we do not know whether this is a *strict* superset or not.

Payment decomposition. While Theorem 1 establishes that under unanimity an implementable allocation rule can be almost decomposed, can we also decompose payment decisions (almost everywhere)? For a unanimous and implementable allocation rule, there will always exist one almost decomposable payment function that implements it - this was shown in Lemma 1. Notice that for every $j \in J$, the payment function defined in Lemma 1, p_i^j depends on the valuation of other components since it is a function of $F^j(\mathbf{v})$. But almost everywhere, F^j can be computed by information of component j (Theorem 1). Hence, almost everywhere, p_i^j only depends on the information of component j. In that sense, the payment function in Lemma 1 is decomposable almost everywhere.

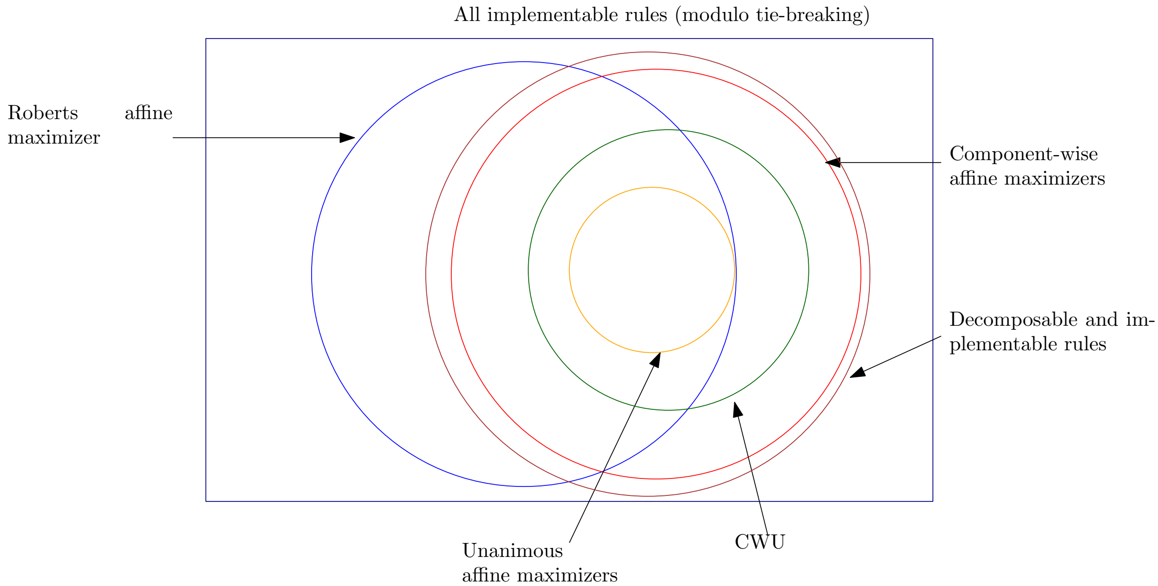

Figure 1: Various classes of allocation rules

Standard revenue equivalence formula (Chung and Olszewski, 2007) gives a complete description of the set of all payment functions that can implement a CWU. These are characterized by a map $h_i : V_{-i} \to \mathbb{R}$ for every agent i. Any payment functions $(q_1, \ldots, q_n)$ that implements a CWU allocation rule must satisfy for every agent i and every (v_i, v_{-i}),

$$q_i(v_i, v_{-i}) = p_i(v_i, v_{-i}) + h_i(v_{-i}),$$

where p_i is the payment function identified in Lemma 1. Even though, p_i is decomposable (almost everywhere), q_i need not be decomposable since h_i can be chosen in a non-decomposable manner by making it depend on *all* the components of other agents.

Other forms of separability. Our result crucially relies on the fact that valuations are additively separable. The following example shows that other forms of separability may not give us the result. Suppose there are two agents and two components. Further, suppose the valuations satisfy

$$v_i(x) = v_i^1(x^1)v_i^2(x^2), \forall x^1 \in A^1, x^2 \in A^2, i = 1, 2.$$

Clearly, the domain of valuations is separable in the sense that if at a valuation profile v^2 at component 2 agent i prefers alternative x^1 over x'^1, then she continues to prefer the same even when the valuation profile at component 2 changes to v'^2 (because the valuations are positive). However, such valuations are clearly not additively separable.

As is well known, the following affine maximizer allocation rule is implementable (with a fixed order over the alternatives in each component to break ties):

$$F(v) \in \underset{x \in A}{\operatorname{argmax}} \sum_{i=1}^{2} \lambda_i v_i(x) = \underset{x \in A}{\operatorname{argmax}} \sum_{i=1}^{2} \lambda_i v_i^1(x^1) v_i^2(x^2).$$

Let $\lambda_1 = 3, \lambda_2 = 2$. Consider the following two valuations:

	A^1			A^2		
v	a^1	b^1	c^1	a^2	b^2	c^2
v_1	2	3	5	1	2	3
v_2	5	2	1	4	2	1

	A^1			A^2		
$\tilde{v}$	a^1	b^1	c^1	a^2	b^2	c^2
v_1	2	3	5	1	2	3
v_2	5	2	1	4	2	100

The allocation at v is (c^1, c^2), but at $\tilde{v}$, it is (a^1, c^2). Even though, the valuations in component 1 did not change in these two profiles, the outcome changed. This shows that we cannot hope to get the kind of decomposability we got in Theorem 1 beyond additively separable valuations.

3.3 Component-wise axioms

Theorem 1 uses unanimity or Pareto in conjunction with implementability. Both these axioms are imposed on the *overall* allocation rule. Below, we provide alternate axioms which are imposed on *components*. As we show below, they are stronger than the axioms on the overall allocation rule, but become equivalent to them in the presence of implementability. Further, they provide a different perspective on our results. Our proof of the main result uses the fact that these axioms are equivalent to unanimity/Pareto.

We start off by stating the component-wise versions of unanimity and Pareto.

DEFINITION **9** *An alternative $a^j \in A^j$ is* **unanimous on component** *j at a valuation profile $\mathbf{v}$ if $v_i^j(a^j) > v_i^j(b^j)$ for all $b^j \neq a^j$ and for all $i \in N$.*
An allocation rule F satisfies **component-wise unanimity** *if for every $j \in J$, for every $a^j \in A^j$, and for every $\mathbf{v}$ such that a^j is unanimous on component j at $\mathbf{v}$, we have $F^j(\mathbf{v}) = a^j$.*

Component-wise unanimity implies unanimity. Under implementability, component-wise unanimity and unanimity are equivalent.

LEMMA **2** *Suppose F is an allocation rule defined on a separable type space.*

1. *If F satisfies component-wise unanimity, then it satisfies unanimity.*

2. *If F is implementable and satisfies unanimity, then it satisfies component-wise unanimity.*

Similarly, we can define a component-wise Pareto, which is weaker than the overall Pareto.

DEFINITION **10** *An alternative $a^j \in A^j$ is* **dominated on component** *j at valuation profile $\mathbf{v}$ if $v_i^j(a^j) < v_i^j(b^j)$ for some $b^j \neq a^j$ and for all $i \in N$.*
An allocation rule F satisfies **component-wise Pareto** *if for every $j \in J$, for every $a^j \in A^j$, and for every $\mathbf{v}$ such that a^j is dominated on component j at $\mathbf{v}$, we have $F^j(\mathbf{v}) \neq a^j$.*

Again, under implementability, component-wise Pareto and Pareto are equivalent.

LEMMA **3** *Suppose F is an allocation rule defined on a separable type space.*

1. *If F satisfies component-wise Pareto, then it satisfies Pareto.*

2. *If F is implementable and satisfies Pareto, then it satisfies component-wise Pareto.*

The proof of Lemma 3 is similar to Lemma 2, and is skipped - Proof of Lemma 2 is in the Appendix.

We now state two more component-wise axioms. Both the axioms will use a notation that we define next. For every $j \in J$, for every $a^j \in A^j$, for every $\mathbf{v}$, and for every $\epsilon \in \mathbb{R}^n_{++}$, define the valuation profile $\mathbf{v}'$ as:

$$\begin{aligned} v'^k(a^k) &= v^k(a^k) \qquad \forall\ a^k \in A^k, \forall\ k \neq j \\ v'^j(a^j) &= v^j(a^j) + \epsilon \\ v'^j(b^j) &= v^j(b^j) \qquad \forall\ b^j \in A^j \setminus \{a^j\}. \end{aligned}$$

We denote $\mathbf{v}'$ as $(\mathbf{v} + \mathbf{1}_\epsilon^{a^j})$. For every $j \in J$ and for every $\mathbf{v}$, define

$$C_j^F(\mathbf{v}) := \{a^j : F^j(\mathbf{v} + \mathbf{1}_\epsilon^{a^j}) = a^j\ \forall\ \epsilon \in \mathbb{R}^n_{++}\}.$$

In words, $C_j^F(\mathbf{v})$ are all the alternatives of component j that will be chosen by F if only valuations of that alternative is increased. In the case of CWU allocation rules, this will capture all the alternatives that achieve the maximum on component j. We show below that this set is always non-empty for *any* implementable F.

LEMMA **4** *Suppose F is an allocation rule defined on a separable type space. Then*

$$F^j(\mathbf{v}) \in C_j^F(\mathbf{v})\ \forall\ j \in J\ \forall\ \mathbf{v}.$$

We define a neutrality property below using this. For every $j \in J$, let π^j be a permutation of A^j. Given a profile of valuations $\mathbf{v}$, we can permute it along j by applying the permutation π^j - note, we only permute the component j of all the agents. With a small abuse of notation, we also denote the permutation of the valuation profile induced by the permutation of the alternative with the same symbol, i.e., $\pi^j(\mathbf{v})$ denotes a valuation profile where the valuations at $\pi^j(a^j) \in A^j$ are the same as the valuations in $\mathbf{v}$ at a^j. For example, suppose that there are two agents and two components and the alternative sets are $A^1 = \{a^1, b^1, c^1\}, A^2 = \{a^2, b^2, c^2\}$. If $\pi^1(a^1) = b^1, \pi^1(b^1) = a^1, \pi^1(c^1) = c^1$, then an example of $\mathbf{v}$ and $\pi^1(\mathbf{v})$ will be:

	A^1			A^2		
$\mathbf{v}$	a^1	b^1	c^1	a^2	b^2	c^2
v_1	2	3	4	7	6	5
v_2	5	1	2	8	4	3

	A^1			A^2		
$\pi^1(\mathbf{v})$	a^1	b^1	c^1	a^2	b^2	c^2
v_1	3	2	4	7	6	5
v_2	1	5	2	8	4	3

Note that the separable type space assumption implies that if $\mathbf{v}$ is in the type profile space, then $\pi^j(\mathbf{v})$ is also in the type profile space.

DEFINITION **11** *An allocation rule F is* **neutral** *if for all $j \in J$, for all permutations π^j, and for all $\mathbf{v}$, we have*

$$C_j^F(\pi^j(\mathbf{v})) = \pi^j(C_j^F(\mathbf{v}))$$

Neutrality says that if the names of the alternatives are changed along a component, then the outcome along that component must be changed accordingly.[10] It is not difficult to see that every CWU allocation rule satisfies neutrality.[11]

Our final component-wise axiom is the following.

DEFINITION **12** *An allocation rule F satisfies* **component-wise equal treatment of equal alternatives (CETEA)** *if for every $j \in J$, every $a^j, b^j \in A^j$, and every $\mathbf{v}$ with $v_i^j(a^j) = v_i^j(b^j)$ for all $i \in N$, we have*

$$\Big[a^j \in C_j^F(\mathbf{v})\Big] \Leftrightarrow \Big[b^j \in C_j^F(\mathbf{v})\Big].$$

[10] A reader may relate this definition of neutrality to a more commonly used definition, where the permutation $\pi : A \to A$ is defined over all alternatives and not restricted to any components. Since any such permutation π can be constructed by a sequence of component-wise permutations, our definition of neutrality is *equivalent* to this definition.

[11] A more conventional way of defining neutrality would require that for all permutations π^j and for all $\mathbf{v}$ with $\pi^j(\mathbf{v}) \neq \mathbf{v}$, we have $F^j(\pi^j(\mathbf{v})) = \pi^j(F^j(\mathbf{v}))$. Because of tie-breaking issues, a CWU allocation rule may fail to be neutral in this sense. Our definition of neutrality is weaker than this, and it avoids tie-breaking issues by using the notion of $C_j^F(\cdot)$.

CETEA requires that if two alternatives on some component are *equal* (in the sense that each agent has same valuation for them), then the allocation rule must choose one at slightly higher valuations if and only if it chooses the other at slightly higher valuations.

We are now ready to state the main result of this section.

PROPOSITION **1** *Suppose F is an allocation rule defined on a separable type space. If F is implementable, the following statements are equivalent.*

1. *F satisfies neutrality.*
2. *F satisfies component-wise Pareto.*
3. *F satisfies component-wise unanimity.*
4. *F satisfies CETEA.*
5. *F satisfies Pareto.*
6. *F satisfies unanimity.*

Proposition 1 allows us to read and interpret Theorem 1 and Corollary 1 in a variety of ways. It also allows us to contrast our results with the Roberts' theorem (Fact 1) in several ways.

4 Proof of Theorem 1

We present the proof of Theorem 1 below. Before doing so, we document some results from the literature and some preliminary results first. These results will be used repeatedly in the proofs.

4.1 Preparations for the proofs

We now state some elementary results for our proof. Some of these results will be for $|J| = 1$, and we will drop the component superscripts/notations for stating those results.

The first result is a well-known necessary condition for implementability.

FACT **2 (2-cycle monotonicity - Rochet (1987))** *If F is implementable, then for every $i \in N$, for every $\mathbf{v}$ and $\mathbf{v}'$ with $v_{-i} = v'_{-i}$ and $F(\mathbf{v}) = a$ and $F(\mathbf{v}') = b$ we have,*

$$\sum_{j=1}^{m} \left[v_i^j(a^j) - v_i^j(b^j)\right] \geq \sum_{j=1}^{m} \left[v_i'^j(a^j) - v_i'^j(b^j)\right].$$

The proof can be done by adding the pair of incentive constraints for $\mathbf{v}$ and $\mathbf{v}'$. A straightforward consequence of this necessary result is the following necessary condition for implementability.

Definition 13 *An allocation rule F satisfies* **component-wise positive association of differences (CPAD)** *if for every $j \in J$, every v^{-j}, and every v^j, v'^j with $F^j(v^j, v^{-j}) = a^j$, the following holds:*

$$\Big[v_i'^j(a^j) - v_i'^j(b^j) > v_i^j(a^j) - v_i^j(b^j)\ \forall\ b^j \in A^j \setminus \{a^j\},\ \forall\ i \in N\Big] \Rightarrow \Big[F^j(v'^j, v^{-j}) = a^j\Big].$$

When $|J| = 1$, CPAD collapses to the following condition identified by Roberts (1979), which he refers to as PAD. Adapted to our setting ($|J| \geq 1$), PAD is the following.

Definition 14 *An allocation rule F satisfies* **positive association of differences (PAD)** *if for every $\mathbf{v}$ and $\mathbf{v}'$ with $F(\mathbf{v}) = a$, the following holds:*

$$\Big[\sum_{j \in J} \big[v_i'^j(a^j) - v_i'^j(b^j)\big] > \sum_{j \in J} \big[v_i^j(a^j) - v_i^j(b^j)\big]\ \forall\ b \in A \setminus \{a\},\ \forall\ i \in N\Big] \Rightarrow \Big[F(\mathbf{v}') = a\Big].$$

Notice that PAD is an "overall" condition and CPAD is a "component-wise" condition. PAD is necessary for implementability (Roberts, 1979). In the Appendix, we give an example of an allocation rule which satisfies CPAD but fails PAD. Hence, CPAD does not imply PAD. We do not know if PAD implies CPAD. However, implementability implies CPAD.

Lemma 5 *Every implementable F satisfies CPAD.*

Next, we state an important result from the literature that we use in our proof.

Fact 3 (**Mishra and Sen (2012)**) *Suppose $|J| = 1$ and F is an allocation rule on $(\alpha_1, \beta_1)^{|A|} \times \ldots \times (\alpha_n, \beta_n)^{|A|}$, where $\alpha_i < \beta_i$ for all $i \in N$. Then, the following statements are equivalent.*

1. *F satisfies neutrality and PAD.*

2. *There exist non-negative weights $(\lambda_1, \ldots, \lambda_n)$, not all equal to zero, such that*

$$F(\mathbf{v}) \in \arg\max_{a \in A} \Big[\sum_{i \in N} \lambda_i v_i(a)\Big].$$

Fact 1, which is the Robert's theorem, uses ontoness and and Fact 3 uses neutrality. On the other hand, the type space in Fact 1 is $\mathbb{R}^{|A|}$ and it is any $|A|$-dimensional open interval in Fact 3.

4.2 The proof

Proof of Theorem 1 involves proving an analogue of Fact 3 when $|J| > 1$ in separable type space. In particular, we will prove the following theorem.

THEOREM **2** *Suppose F is an allocation rule defined on a separable type space. Then, the following statements are equivalent.*

1. *F is a neutral allocation rule satisfying CPAD.*

2. *F is a CWU allocation rule.*

It is easy to see that Theorem 1 follows from Theorem 2: (i) implementability implies CPAD (Lemma 5); (ii) Proposition 1 shows that neutrality is equivalent to unanimity/Pareto; and (iii) Theorem 2 then implies Theorem 1.

Hence, we will only prove Theorem 2. Also, since every CWU allocation rule is neutral, we only need to show that any allocation rule satisfying neutrality and CPAD is a CWU allocation rule. We do that below.

Main idea of the proof. In this section, we prove our main result. Before proceeding to the details of the proof, we give a sketch of the main idea of the proof. In the proof, we fix a component j and fix two profiles v^{-j} and v'^{-j}. Now, F restricted to this component at these two profiles gives two *marginal* allocation rules on component j. These marginal allocation rules satisfy PAD since F satisfies CPAD. Because of separable type space structure, the type space restricted to component j satisfies the required richness to invoke Fact 3 for both the marginal allocation rules. Hence, we get a pair of weight vectors. The crux of the proof lies in showing that these weight vectors are *equivalent*, i.e., one is obtained by a uniform scaling of the other. This establishes the result. The main idea for establishing this uniform scaling is that if these two weight vectors are not *equivalent*, then we show that an inequality implied by Fact 2 can never hold, giving us a contradiction.

We now present the proof of Theorem 2.

Proof: Suppose F is a neutral allocation rule satisfying CPAD. Fix a component j and consider two profiles v^{-j} and v'^{-j}. Denote the restriction of F to profiles of agents where components besides j are fixed at v^{-j} and v'^{-j} as f^j and f'^j respectively. Note that the domain of f^j and f'^j are $V_1^j \times \ldots \times V_n^j$. Since F satisfies CPAD, f^j and f'^j satisfy PAD. Further, the valuation of each agent $i \in N$ for component j lie in $\mathbb{R}_{++}^{|A^j|}$ and $|A^j| \geq 3$.

By Fact 3, there exists $\lambda^j, \lambda'^j \in \mathbb{R}^n_+ \setminus \{\mathbf{0}\}$ such that for all $v^j \in V^j$,

$$f^j(v^j) \in \arg\max_{a^j \in A^j} \sum_{i \in N} \lambda^j_i v^j_i(a^j).$$
$$f'^j(v^j) \in \arg\max_{a^j \in A^j} \sum_{i \in N} \lambda'^j_i v^j_i(a^j).$$

Observation. Notice that it is without loss of generality to assume that λ^j (and λ'^j) is such that for some agent $\ell \in N$, we have $\lambda^j_\ell = 1$. We will use this observation repeatedly at various points in the proof.

We show that there exists a positive number $\rho > 0$ such that $\lambda^j_h = \rho \lambda'^j_h$ for all $h \in N$. As a consequence of this, the set of maximizers at every v^j is the same for λ^j and λ'^j. Hence, it is without loss of generality to write

$$f'^j(v^j) \in \arg\max_{a^j \in A^j} \sum_{i \in N} \lambda^j_i v^j_i(a^j).$$

This will conclude the proof.

We begin by noting that it is without loss of generality to assume that v^{-j} and v'^{-j} differ in only one agent's valuation - if they differ in more than one agent's valuation, we can repeat this argument to show the desired property. Suppose v^{-j} and v'^{-j} differ in the valuation of agent i, i.e., for all $k \neq j$, we have $v^k_i \neq v'^k_i$ but $v^k_h = v'^k_h$ for all $h \neq i$.

For any alternative a, define a^{-j} as all the alternatives from components other than j-th component. Define Δ as follows:

$$\Delta := \max_{a^{-j}, b^{-j} \in A^{-j}} \sum_{k \neq j} \Big(\big[v^k_i(a^k) - v^k_i(b^k)\big] - \big[v'^k_i(a^k) - v'^k_i(b^k)\big] \Big).$$

Now pick any v^j and v'^j which only differ in valuation of agent i. Denote $F(v^j, v^{-j}) \equiv a$ and $F(v'^j, v'^{-j}) \equiv b$. Since F is implementable and (v^j, v^{-j}) and (v'^j, v'^{-j}) only differ in the valuation of agent i, we can apply Fact 2 to get

$$\begin{aligned}\big[v^j_i(a^j) + \sum_{k \neq j} v^k_i(a^k)\big] - \big[v^j_i(b^j) + \sum_{k \neq j} v^k_i(b^k)\big]& \\ + \big[v'^j_i(b^j) + \sum_{k \neq j} v'^k_i(b^k)\big] - \big[v'^j_i(a^j) + \sum_{k \neq j} v'^k_i(a^k)\big] &\geq 0.\end{aligned}$$

Using the definition of Δ, we get

$$\Delta + \big[v^j_i(a^j) - v^j_i(b^j)\big] - \big[v'^j_i(a^j) - v'^j_i(b^j)\big] \geq 0. \tag{2}$$

Note here that this applies to any v^j and v'^j - below, we choose specific v^j and v'^j to reach contradictions.

Notation. In the construction of valuation profiles below, we assign zero valuations in many cases. Since 0 is not in the type space (to remind, each $v_i^j \in \mathbb{R}_{++}^{|A^j|}$), what we mean is a valuation arbitrarily close to zero - this is possible since 0 lies in the closure of our type space. We write 0 only to avoid extra notations.

We also drop the notation for component j completely because the proof only involves working in that component and it will not create any confusion.

Now, we complete the step in various cases. Assume for contradiction that there is no $\rho > 0$ such that $\lambda_\ell \neq \rho\lambda'_\ell$ for all $\ell \in N$. In various cases below, we choose v and v' appropriately and use Inequality (2) to get a contradiction.

Case 1. For every $h \in N$, $\lambda_h\lambda'_h = 0$ (i.e, either $\lambda_h = 0$ or $\lambda'_h = 0$) - note that by definition there is at least one $\ell \in N$ such that $\lambda_\ell > 0$ and at least one $\ell' \in N$ such that $\lambda'_{\ell'} > 0$. Without loss of generality, we can assume that $\lambda_\ell = 1$ and $\lambda'_{\ell'} = 1$. Note that (by assumption for this case), $\lambda'_\ell = \lambda_{\ell'} = 0$. We consider some sub-cases.

Case 1a. Suppose $\lambda_i = \lambda'_i = 0$. In that case, we construct two valuation profiles (of the j-th component) - v and v' such that v and v' differ from each other by agent i's valuation. Moreover, the valuation of all the agents except agent i and ℓ, ℓ' are zero for all the alternatives, i.e., for all $c \in A$ we have

$$v_h(c) = v'_h(c) = 0 \ \forall\ h \in N \setminus \{i, \ell, \ell'\}.$$

Now, choose $a, b \in A$ and for all $c \notin \{a, b\}$, we have

$$v_i(c) = v'_i(c) = v_\ell(c) = v'_\ell(c) = v_{\ell'}(c) = v'_{\ell'}(c) = 0.$$

Finally, choose $\alpha, \delta, \gamma, \gamma' > 0$ with $\alpha > \delta$ and $K > 0$.

$$v_i(a) = K(\alpha - \delta), v_i(b) = v'_i(b) = 0, v'_i(a) = K\alpha.$$

$$v_\ell(a) = v'_\ell(a) = \gamma, v_\ell(b) = v_{\ell'}(a) = v'_\ell(b) = v'_{\ell'}(a) = 0, v_{\ell'}(b) = v'_{\ell'}(b) = \gamma'.$$

The valuations of agent i, ℓ, ℓ' for alternatives a and b are shown in Table 1.

By construction, $f(v) = a$ and $f'(v') = b$. Now, substituting the values of v_i and v'_i in Inequality (2), we get

$$0 \leq \Delta + K(\alpha - \delta) - K(\alpha) = \Delta - K\delta.$$

	a	b	Weights
v_i	$K(\alpha-\delta)$	0	$\lambda_i = 0$
v_ℓ	γ	0	$\lambda_\ell = 1$
$v_{\ell'}$	0	γ'	$\lambda_{\ell'} = 0$
v'_i	$K\alpha$	0	$\lambda'_i = 0$
v'_ℓ	γ	0	$\lambda'_\ell = 0$
$v'_{\ell'}$	0	γ'	$\lambda'_{\ell'} = 1$

Table 1: Illustration

Equivalently, we get

$$K\delta \leq \Delta,$$

which is impossible since $K, \delta > 0$ and K can be chosen arbitrarily large. Hence, we get a contradiction for this case.

Case 1b. Suppose $\lambda_i > 0$ but $\lambda'_i = 0$ (alternatively, we can also suppose that $\lambda'_i > 0$ but $\lambda_i = 0$). As before, there is some $\ell' \in N$ such that $\lambda'_{\ell'} > 0$ but $\lambda_{\ell'} = 0$. Without loss of generality, we assume that $\lambda'_{\ell'} = \lambda_i = 1$. In that case, we construct v and v' very similar to Case 1a with slight modifications.

For all $c \in A$ we have

$$v_h(c) = v'_h(c) = 0 \ \forall \ h \in N \setminus \{i, \ell'\}.$$

Now, choose $a, b \in A$ and for all $c \notin \{a, b\}$, we have

$$v_i(c) = v'_i(c) = v_{\ell'}(c) = v'_{\ell'}(c) = 0.$$

Finally, choose $\alpha, \delta, \gamma' > 0$ with $\alpha > \delta$ and $K > 0$.

$$v_i(a) = K(\alpha - \delta), v_i(b) = v'_i(b) = 0, v'_i(a) = K\alpha.$$

$$v_{\ell'}(a) = v'_{\ell'}(a) = 0, v_{\ell'}(b) = v'_{\ell'}(b) = \gamma'.$$

Valuations of agent i and ℓ' for a and b are shown in Table 2.

Now, by construction, $f(v) = a$ and $f'(v') = b$. Substituting the values of v_i and v'_i in Inequality (2), we get

$$0 \leq \Delta + K(\alpha - \delta) - K(\alpha) = \Delta - K\delta.$$

Equivalently, we get

$$K\delta \leq \Delta,$$

	a	b	Weights
v_i	$K(\alpha - \delta)$	0	$\lambda_i = 1$
$v_{\ell'}$	0	γ'	$\lambda_{\ell'} = 0$
v'_i	$K\alpha$	0	$\lambda'_i = 0$
$v'_{\ell'}$	0	γ'	$\lambda'_{\ell'} = 1$

Table 2: Illustration

which is impossible since $K, \delta > 0$ and K can be chosen arbitrarily large. Hence, we get a contradiction for this case.

The contradictions in these two sub-cases imply that Case 1 is impossible.

CASE 2. There is some agent ℓ such that $\lambda_\ell > 0$ and $\lambda'_\ell > 0$. Without loss of generality, we assume that $\lambda_\ell = \lambda'_\ell = 1$. Since there is no $\rho > 0$ such that $\lambda_h = \rho\lambda'_h$ for all $h \in N$, there is some $\ell' \neq \ell$ such that $\lambda_{\ell'} \neq \lambda'_{\ell'}$. We now consider sub-cases.

Case 2a. Suppose $\lambda_i \neq \lambda'_i$. Clearly, either $\lambda_i > 0$ or $\lambda'_i > 0$. Assume for contradiction $\lambda_i > \lambda'_i$ - an analogous argument works if $\lambda'_i > \lambda_i$.

Now, we construct two valuation profiles - v and v' such that v and v' differ from each other by agent i's valuation. Moreover, the valuation of all the agents except agent i and ℓ are zero for all the alternatives, i.e., for all $c \in A$ we have

$$v_h(c) = v'_h(c) = 0 \ \forall \ h \in N \setminus \{i, \ell\}.$$

Now, choose $a, b \in A^j$ and for all $c \notin \{a, b\}$, we have

$$v_i(c) = v'_i(c) = v_\ell(c) = v'_\ell(c) = 0.$$

Finally, choose $\alpha, \beta, \delta > 0$ with $\alpha > \delta$ and $K > 0$.

$$v_i(a) = K(\alpha - \delta), v_i(b) = v'_i(b) = 0, v'_i(a) = K\alpha.$$

$$v_\ell(a) = v'_\ell(a) = 0, v_\ell(b) = v'_\ell(b) = K\beta.$$

The valuations of agent i and ℓ for alternatives a and b are shown in Table 3.

Now, since $\lambda_i > 0$, by construction of v, if we ensure $\lambda_i K(\alpha - \delta) > K\beta$ or $\lambda_i(\alpha - \delta) > \beta$, we get $f(v) = a$. Similarly, if we ensure $\lambda'_i K\alpha < K\beta$ or $\lambda'_i\alpha < \beta$, we get $f'(v') = b$. Note that by choosing α, β, δ appropriately, we can ensure

$$\lambda_i(\alpha - \delta) > \beta > \lambda'_i\alpha,$$

	a	b	Weights
v_i	$K(\alpha-\delta)$	0	$\lambda_i > 0$
v_ℓ	0	$K\beta$	$\lambda_\ell = 1$
v'_i	$K\alpha$	0	$\lambda_i > \lambda'_i$
v'_ℓ	0	$K\beta$	$\lambda'_\ell = 1$

Table 3: Illustration

since $\lambda_i > \lambda'_i$. [12] Now, substituting the values of v_i and v'_i in Inequality (2), we get

$$0 \leq \Delta + K(\alpha-\delta) - K\alpha = \Delta - K\delta.$$

Equivalently, we get

$$K\delta \leq \Delta,$$

which is impossible since $K, \delta > 0$ and K can be chosen to be arbitrarily large. This gives us a contradiction.

Case 2b. Suppose $\lambda_i = \lambda'_i = 0$. Then, choose ℓ' such that $\lambda_{\ell'} \neq \lambda'_{\ell'}$. Without loss of generality, assume that $\lambda_{\ell'} > \lambda'_{\ell'}$.

Now, we construct two valuation profiles - v and v' such that v and v' differ from each other by agent i's valuation. Moreover, the valuation of all the agents except agent i, ℓ, ℓ' are zero for all the alternatives, i.e., for all $c \in A$ we have

$$v_h(c) = v'_h(c) = 0 \ \forall \ h \in N \setminus \{i, \ell, \ell'\}.$$

Now, choose $a, b \in A$ and for all $c \notin \{a, b\}$, we have

$$v_i(c) = v'_i(c) = v_\ell(c) = v'_\ell(c) = v_{\ell'}(c) = v'_{\ell'}(c) = 0.$$

Finally, choose $\alpha, \delta, \gamma, \gamma' > 0$ with $\alpha > \delta$ and $K > 0$.

$$v_i(a) = K(\alpha-\delta), v_i(b) = v'_i(b) = 0, v'_i(a) = K\alpha,$$

$$v_\ell(a) = v'_\ell(a) = 0, v_\ell(b) = v'_\ell(b) = \gamma,$$

$$v_{\ell'}(a) = v'_{\ell'}(a) = \gamma', v_{\ell'}(b) = v'_{\ell'}(b) = 0.$$

The valuations of agent i and ℓ for alternatives a and b are shown in Table 4.

[12] A similar argument can be made by switching the roles of v and v' if $\lambda'_i > \lambda_i$.

	a	b	Weights
v_i	$K(\alpha-\delta)$	0	$\lambda_i = 0$
$v_{\ell'}$	γ'	0	$\lambda_{\ell'} > 0$
v_ℓ	0	γ	$\lambda_\ell = 1$
v'_i	$K\alpha$	0	$\lambda'_i = 0$
$v'_{\ell'}$	γ'	0	$\lambda'_{\ell'} < \lambda_{\ell'}$
v'_ℓ	0	γ	$\lambda'_\ell = 1$

Table 4: Illustration

Now, if $\lambda_{\ell'}\gamma' > \gamma$, we have $f(v) = a$. Similarly, if $\lambda'_\ell\gamma' < \gamma$, we have $f'(v') = b$. Since $\lambda_{\ell'} > \lambda'_{\ell'}$, it is possible to choose γ, γ' such that

$$\lambda_{\ell'}\gamma' > \gamma > \lambda'_\ell\gamma'.$$

Now, substituting the values of v_i and v'_i in Inequality (2), we get

$$0 \leq \Delta + K(\alpha-\delta) - K\alpha = \Delta - K\delta.$$

Equivalently, we get

$$K\delta \leq \Delta,$$

which is impossible since $K, \delta > 0$ and K can be chosen to be arbitrarily large. This gives us a contradiction.

Case 2c. Suppose $\lambda_i = \lambda'_i = \lambda > 0$. Then, choose ℓ' such that $\lambda_{\ell'} \neq \lambda'_{\ell'}$ - again, such ℓ' exists because there is no $\rho > 0$ such that $\lambda_h = \rho\lambda'_h$ for all $h \in N$. Without loss of generality, assume that $\lambda_{\ell'} < \lambda'_{\ell'}$. Then, define (whenever $\lambda_{\ell'} > 0$)

$$\mu := \frac{\lambda}{\lambda_{\ell'}}, \mu' := \frac{\lambda}{\lambda'_{\ell'}}.$$

Note that $\mu > \mu'$.

Now, we construct two valuation profiles - v and v' such that v and v' differ from each other by agent i's valuation. Moreover, the valuation of all the agents except agent i and ℓ' are zero for all the alternatives, i.e., for all $c \in A$ we have

$$v_h(c) = v'_h(c) = 0 \;\forall\; h \in N \setminus \{i, \ell'\}.$$

Now, choose $a, b \in A$ and for all $c \notin \{a, b\}$, we have

$$v_i(c) = v'_i(c) = v_{\ell'}(c) = v'_{\ell'}(c) = 0.$$

Finally, choose $\alpha, \beta, \delta > 0$ with $\alpha > \delta$ and $K > 0$.

$$v_i(a) = K(\alpha - \delta), v_i(b) = v_i'(b) = 0, v_i'(a) = K\alpha.$$

$$v_{\ell'}(a) = v_{\ell'}'(a) = 0, v_{\ell'}(b) = v_{\ell'}'(b) = K\beta.$$

The valuations of agent i and ℓ' for alternatives a and b are shown in Table 5.

	a	b	Weights
v_i	$K(\alpha - \delta)$	0	$\lambda_i = \lambda > 0$
$v_{\ell'}$	0	$K\beta$	$\lambda_{\ell'}$
v_i'	$K\alpha$	0	$\lambda_i' = \lambda$
$v_{\ell'}'$	0	$K\beta$	$\lambda_{\ell'}' > \lambda_{\ell'}$

Table 5: Illustration

Now, if $\lambda K(\alpha - \delta) > \lambda_{\ell'} K\beta$, which is trivially possible if $\lambda_{\ell'} = 0$, or, equivalently, $\mu(\alpha - \delta) > \beta$, then we have $f(v) = a$. Similarly, if $\lambda K\alpha < \lambda_{\ell'}' K\beta$ or, equivalently, $\mu'\alpha < \beta$, then we have $f'(v') = b$. Since $\mu > \mu'$, it is possible to satisfy

$$\mu(\alpha - \delta) > \beta > \mu'\alpha,$$

by appropriately choosing α, β, δ with $\alpha > \delta$.

Now, substituting the values of v_i and v_i' in Inequality (2), we get

$$0 \leq \Delta + K(\alpha - \delta) - K\alpha = \Delta - K\delta.$$

Equivalently, we get

$$K\delta \leq \Delta,$$

which is impossible since $K, \delta > 0$ and K can be chosen to be arbitrarily large. This gives us a contradiction.

This exhausts all the cases and we have thus completed the proof. ■

5 Discussions

We conclude by discussing some open problems related to our main result.

Desirability of unanimity. As discussed earlier, unanimity is a normatively appealing axiom. Indeed, if an allocation rule violates unanimity, then the social planner must have strong reasons to discriminate between alternatives, and probably, should not bother about aggregating private valuations of agents. However, it will be useful to formally show that if a planner has a particular objective in mind (say, maximizing expected utilities with respect to some prior over valuations of agents), then he should use a unanimous allocation rule. Such a result will give a strong foundation to the use of unanimity.

Relaxing unanimity. We do not have an answer how our main result changes if we replace unanimity by ontoness (as in Roberts' theorem). A plausible conjecture is that implementability and ontoness will imply the following class of (non-decomposable) allocation rules. Informally, these allocation rules partition the set of components J into $(J^1, \ldots, J^\ell)$. Inside each J^k, we do an affine maximization, i.e., have weight vector λ^k and a map κ^k as in Roberts' theorem to find the alternatives to be chosen for components in J^k. If each J^k contains one component, then we get component-wise affine maximizers and if $\ell = 1$ we get Roberts' affine maximizers.[13] We leave this issue for future research.

Randomized mechanisms. We have assumed that the mechanisms we consider are deterministic mechanisms. The counterpart of Roberts' theorem is not known if we consider randomized mechanisms. We do not know how our results generalize with randomized mechanisms. A remarkable recent result by Chen et al. (2016) shows that for every Bayesian incentive compatible randomized mechanism there exists a *Bayesian incentive compatible deterministic* mechanism generating the same *interim* expected utilities/allocation probabilities to agents. This result cannot be applied to our problem directly because: (a) the result is silent for dominant strategy incentive compatible mechanisms; and (b) it is not clear if unanimity/Pareto/neutrality can be preserved. We plan to investigate this issue in future research.

Bayesian incentive compatibility. We have restricted attention to dominant strategy incentive compatible mechanisms. The literature is silent on an analogue of Roberts' theorem when we consider Bayesian incentive compatible mechanisms.[14] Finding the extension of Roberts' theorem and our main result using Bayesian incentive compatibility remains an open question.

[13] The CWU allocation rules are obtained by setting each $|J^k| = 1$ *and* without the κ map.

[14] By the result in Chen et al. (2016), there is little loss of generality if we consider deterministic mechanisms when the solution concept is Bayesian incentive compatibility.

Applications. Similar to Roberts' theorem, our result helps us in understanding the scope and structure of dominant strategy incentive compatible mechanisms. An immediate corollary of our result is that every implementable, unanimous, and *anonymous* allocation rule must be *utilitarianism* (weighted utilitarianism with all weights equal), where anonymity refers to the usual definition that if agents' valuations are permuted then the outcome does not change. Other applications of our results (without imposing anonymity), where one determines the optimal values of weights of weighted utilitarianism based on some ex-ante objective of the planner is left as a topic for future research.

Appendix: Omitted proofs

Proof of Lemma 1

Proof: Consider (v_i, v_{-i}) and (v_i', v_{-i}). Let F be a CWU allocation rule satisfying INA with weights $\{\lambda^j\}_{j\in J}$. Further, suppose $F(v_i, v_{-i}) = a$ and $F(v_i', v_{-i}) = b$. Let $\tilde{J}$ be the set of components such that $\lambda_i^j = 0$ for all $j \in \tilde{J}$. By INA, $a^j = b^j$ for all $j \in \tilde{J}$. Then,

$$
\begin{aligned}
\sum_{j\in J} v_i^j(a^j) + p_i(v_i, v_{-i}) &= \sum_{j\in J} \big[v_i^j(a^j) + p_i^j(v_i, v_{-i})\big] \\
&= \sum_{j\in\tilde{J}} v_i^j(a^j) + \sum_{j\notin\tilde{J}} \big[v_i^j(a^j) + \frac{1}{\lambda_i^j}\sum_{k\neq i}\lambda_k^j v_k^j(a^j)\big] \\
&= \sum_{j\in\tilde{J}} v_i^j(a^j) + \sum_{j\notin\tilde{J}} \frac{1}{\lambda_i^j}\big[\sum_{k\in N}\lambda_k^j v_k^j(a^j)\big] \\
&\geq \sum_{j\in\tilde{J}} v_i^j(b^j) + \sum_{j\notin\tilde{J}} \frac{1}{\lambda_i^j}\big[\sum_{k\in N}\lambda_k^j v_k^j(b^j)\big] \\
&\text{(Using the definition of CWU and the fact that } a^j = b^j \text{ for all } j\in\tilde{J}) \\
&= \sum_{j\in J} v_i^j(b^j) + \sum_{j\notin\tilde{J}} \big[\frac{1}{\lambda_i^j}\sum_{k\neq i}\lambda_k^j v_k^j(b^j)\big] \\
&= \sum_{j\in J} \big[v_i^j(b^j) + p_i^j(v_i', v_{-i})\big],
\end{aligned}
$$

which is the desired inequality for DSIC. ■

Proof of Lemma 2

Proof: Proof of (1). Suppose F satisfies component-wise unanimity. Consider a valuation profile $\mathbf{v}$ which is unanimous at an alternative $a \equiv (a^1, \ldots, a^m)$. Fix an arbitrary component j and an arbitrary agent i. Denote $V_i^{-j} := \sum_{k\neq j} v_i^k(a^k)$. Notice that for every alternative $b^j \in A^j$, (a^{-j}, b^j) is an alternative and since a is unanimous at $\mathbf{v}$, we have

$$V_i^{-j} + v_i^j(a^j) > V_i^{-j} + v_i^j(b^j).$$

This implies that $v_i^j(a^j) > v_i^j(b^j)$. Since i and j were arbitrary, we conclude that $\mathbf{v}$ is unanimous on component j at a^j for every $j \in J$. Using component-wise unanimity on each component, we get $F(\mathbf{v}) = a$. Hence, F satisfies unanimity.

PROOF OF (2). Suppose F satisfies unanimity. Consider a valuation profile $\mathbf{v}$ which is unanimous on component j at a^j. Assume for contradiction $F^j(\mathbf{v}) \neq a^j$. Suppose $F(\mathbf{v}) = b$ with $b^j \neq a^j$. Consider a valuation profile $\mathbf{v}'$ as follows. Choose $\epsilon > 0$ but arbitrarily close to zero and let

$$\begin{aligned}
v_i'^k(c^k) &= \epsilon \ \forall\ k \in J \setminus \{j\},\ \forall\ c^k \in A^k \setminus \{b^k\},\ \forall\ i \in N \\
v_i'^j(c^j) &= \epsilon \ \forall\ c^j \in A^j \setminus \{a^j, b^j\},\ \forall\ i \in N \\
v_i'^k(b^k) &= v_i^k(b^k) \ \forall\ k \in J,\ \forall\ i \in N \\
v_i'^j(a^j) &= v_i^j(a^j) - \epsilon \ \forall\ i \in N
\end{aligned}$$

Notice that $\mathbf{v}'$ is unanimous at (b^{-j}, a^j), and hence, $F(\mathbf{v}') = (b^{-j}, a^j)$. However, values of all alternatives except b decrease from $\mathbf{v}$ to $\mathbf{v}'$. Hence, PAD implies that $F(\mathbf{v}') = b$, which is a contradiction. ■

We now give a proof of Lemma 5 before giving the subsequent missing proofs.

Proof of Lemma 5

Proof: Suppose F is an implementable allocation rule. Fix $j \in J$, v^{-j}. Without loss of generality, consider v^j and v'^j that differ in the valuation of just one agent, say i, i.e., $v_i^j \neq v_i'^j$ but $v_k^j = v_k'^j$ for all $k \neq i$. Further, suppose $F(v^j, v^{-j}) = a \equiv (a^1, \ldots, a^m)$ and

$$v_i'^j(a^j) - v_i'^j(b^j) > v_i^j(a^j) - v_i^j(b^j) \qquad \forall\ b^j \neq a^j. \tag{3}$$

Suppose $F(v'^j, v^{-j}) = c \equiv (c^1, \ldots, c^m)$.

By 2-cycle monotonicity (Fact 2)

$$\sum_{\ell \neq j} \Big[v_i^\ell(a^\ell) - v_i^\ell(c^\ell) \Big] + \big[v_i^j(a^j) - v_i^j(c^j) \big] \geq \sum_{\ell \neq j} \Big[v_i^\ell(a^\ell) - v_i^\ell(c^\ell) \Big] + \big[v_i'^j(a^j) - v_i'^j(c^j) \big].$$

Hence, we get $\big[v_i^j(a^j) - v_i^j(c^j) \big] \geq \big[v_i'^j(a^j) - v_i'^j(c^j) \big]$. By Inequality 3, we get $c^j = a^j$.

If v^j and v'^j differ in more than one agents' valuation, then we can repeatedly apply this argument. ■

Proof of Lemma 4

Proof: By Lemma 5, an implementable F satisfies CPAD. Hence, by CPAD, for every $\mathbf{v}$ and every $j \in J$, $F^j(\mathbf{v}) \in C_j^F(\mathbf{v})$. ■

Proof of Proposition 1

Proof: We first show equivalence of (1), (2), (3), (4). Since by Lemma 2 and 3, (5) is equivalent to (2) and (6) to (3), we will be done.

For establishing the equivalence of (1), (2), (3), and (4), we first assume that $|J| = 1$. Hence, we drop any notation involving components. Further, we denote the one component allocation rule as f. Note that f satisfies (C)PAD by Lemma 5. We prove each implication one by one.

(1) $\Rightarrow$ (2). Suppose f neutral. Consider a type profile $\mathbf{v}$ and $a \in A$ such that $v(b) > v(a)$ for some $b \neq a$. Assume for contradiction $f(\mathbf{v}) = a$. Hence, $a \in C^f(\mathbf{v})$. Consider v' such that $v'(a) = v(b), v'(b) = v(a) + \epsilon$, where $\epsilon \in \mathbb{R}^n_{++}$ but arbitrarily close to $\mathbf{0}$, and $v'(c) = v(c)$. Since ϵ is arbitrarily close to $\mathbf{0}$, by PAD, $f(\mathbf{v}') = a$. Hence, $b \notin C^f(\rho(\mathbf{v}))$, where ρ is the permutation satisfying $\rho(a) = b, \rho(b) = a$, and $\rho(x) = x$ for all $x \notin \{a, b\}$. This contradicts neutrality because $a \in C^f(\mathbf{v})$.

(2) $\Rightarrow$ (3). This is trivial.

The next two implications will use the following fact.

Fact 4 **(Proposition 1 in Mishra and Sen (2012))** *Suppose $f : (\alpha, \beta)^{|A| \times n} \to A$ satisfies PAD. Consider two type profiles $\mathbf{v}, \mathbf{v}'$ such that $v'(a) = v(a), v'(b) = v(b)$ for some $a, b \in A$. Then, the following are true.*

1. *Suppose $a, b \in C^f(\mathbf{v})$. Then, $\left[a \in C^f(\mathbf{v}')\right] \Leftrightarrow \left[b \in C^f(\mathbf{v}')\right]$.*
2. *Suppose $a \in C^f(\mathbf{v}), b \notin C^f(\mathbf{v})$. Then, $b \notin C^f(\mathbf{v}')$.*

The proof of this fact follows from PAD - though simple, the arguments are somewhat tedious.

(3) $\Rightarrow$ (4). Suppose f satisfies unanimity. Assume for contradiction that it fails ETEA. Then, there is some profile $\mathbf{v}$ with $v(a) = v(b)$ for some $a, b \in A$ such that $a \in C^f(\mathbf{v})$ but $b \notin C^f(\mathbf{v})$. Consider a valuation profile $\mathbf{v}'$ such that $v'(a) = v(a) = v'(b) = v(b)$ and $v'(c) = \alpha + \epsilon$ for all $c \notin \{a, b\}$, where $\epsilon \in \mathbb{R}^n_{++}$ but arbitrarily close to $\mathbf{0}$.

The two profiles $\mathbf{v}$ and $\mathbf{v}'$ are shown in Table 6. Notice that by unanimity, $a, b \in C^f(\mathbf{v}')$. This immediately contradicts (1) in Fact 4.

Valuations	a	b	c
$\mathbf{v}$	$v(a)$	$v(b) = v(a)$	$v(c)$
$\mathbf{v}'$	$v'(a) = v(a)$	$v'(b) = v(a)$	$v'(c) = \alpha + \epsilon$

Table 6: Type profiles $\mathbf{v}$ and $\mathbf{v}'$.

(4) $\Rightarrow$ (1). Suppose f satisfies ETEA. Pick a pair of alternatives $a, b \in A$ and a permutation ρ such that $\rho(a) = b, \rho(b) = a, \rho(c) = c$ for all $c \notin \{a, b\}$. Such permutations are called *transpositions*, and it is well known that every permutation can be written as *product/composition* of transpositions. Hence, it is enough to establish neutrality for such transpositions.

Fix a type profile $\mathbf{v}$ and denote $\mathbf{v}' \equiv \rho(\mathbf{v})$. We do the proof in couple of steps.

Step 1. Pick any $c \notin \{a, b\}$. We first show that if $c \in C^f(\mathbf{v})$, then $c \in C^f(\mathbf{v}')$. Assume for contradiction $c \notin C^f(\mathbf{v}')$. This means, there is $\epsilon \in \mathbb{R}^n_{++}$ such that $f(\mathbf{v}' + \mathbf{1}^c_\epsilon) = x \neq c$. We consider two cases.

Case 1. Suppose $x \notin \{a, b\}$. Consider the following type profile $\mathbf{v}''$:

$$v''(x) = v(x), v''(c) = v(c), v''(y) = \alpha + \epsilon' \;\forall\; y \notin \{x, c\},$$

where $\epsilon' \in \mathbb{R}^n_{++}$ but arbitrarily close to $\mathbf{0}$. Type profiles $\mathbf{v}, \mathbf{v}', \mathbf{v}''$ are shown in Table 7.

Valuations	a	b	c	x	$y \notin \{a, b, c, x\}$
$\mathbf{v}$	$v(a)$	$v(b)$	$v(c)$	$v(x)$	$v(y)$
$\mathbf{v}'$	$v'(a) = v(b)$	$v'(b) = v(a)$	$v'(c) = v(c)$	$v'(x) = v(x)$	$v'(y) = v(y)$
$\mathbf{v}''$	$v''(a) = \alpha + \epsilon'$	$v''(b) = \alpha + \epsilon'$	$v''(c) = v(c)$	$v''(x) = v(x)$	$v''(y) = \alpha + \epsilon'$

Table 7: Type profiles $\mathbf{v}, \mathbf{v}'$, and $\mathbf{v}''$.

Since $f(\mathbf{v}' + \mathbf{1}^c_\epsilon) = x$, PAD implies that there is $\epsilon'' < \epsilon$ such that $f(\mathbf{v}'' + \mathbf{1}^c_{\epsilon''}) = x$ - notice that from $\mathbf{v}' + \mathbf{1}^c_\epsilon$ to $\mathbf{v}'' + \mathbf{1}^c_{\epsilon''}$, valuations of all alternatives decrease except x, and hence, PAD can be applied. But $c \in C^f(\mathbf{v})$, implies that for some $\hat{\epsilon} < \epsilon''$, we have $f(\mathbf{v} + \mathbf{1}^c_{\hat{\epsilon}}) = c$. Using PAD again, we get $f(\mathbf{v}'' + \mathbf{1}^c_{\epsilon''}) = c$, which is a contradiction.

Case 2. Suppose $x \in \{a, b\}$ - without loss of generality, suppose $x = a$. Then, $a \in C^f(\mathbf{v}')$. Consider the following type profile $\mathbf{v}''$:

$$v''(c) = v(c), v''(y) = v(b) \;\forall\; y \neq c.$$

Valuations	a	b	c	$y \notin \{a,b,c\}$
$\mathbf{v}$	$v(a)$	$v(b)$	$v(c)$	$v(y)$
$\mathbf{v}'$	$v'(a) = v(b)$	$v'(b) = v(a)$	$v'(c) = v(c)$	$v'(y) = v(y)$
$\mathbf{v}''$	$v''(a) = v(b)$	$v''(b) = v(b)$	$v''(c) = v(c)$	$v''(y) = v(b)$

Table 8: Type profiles $\mathbf{v}, \mathbf{v}'$, and $\mathbf{v}''$.

Type profiles $\mathbf{v}, \mathbf{v}', \mathbf{v}''$ are shown in Table 8.

Notice that if $c \notin C^f(\mathbf{v}'')$, then ETEA implies that for every $y \neq c$, we have $y \in C^f(\mathbf{v}'')$. Now, we considering $\mathbf{v}$ and $\mathbf{v}''$ and using $c \in C^f(\mathbf{v})$, we get a contradiction to (2) of Fact 4. Hence, $c \in C^f(\mathbf{v}'')$.

Similarly, considering $\mathbf{v}''$ and $\mathbf{v}'$, and using $a \in C^f(\mathbf{v}'), c \notin C^f(\mathbf{v}')$, we get a contradiction to (2) of Fact 4.

Combining these two cases, we get that if $c \in C^f(\mathbf{v})$, then $c \in C^f(\mathbf{v}')$. If $c \notin C^f(\mathbf{v})$ and $c \in C^f(\mathbf{v}')$, then we can swap the roles of $\mathbf{v}$ and $\mathbf{v}'$ in the above argument to get a contradiction. Hence,

$$\big[c \in C^f(\mathbf{v})\big] \Leftrightarrow \big[c \in C^f(\mathbf{v}')\big].$$

Step 2. Now, we will show that for every $x \in \{a, b\}$,

$$\big[x \in C^f(\mathbf{v})\big] \Leftrightarrow \big[\rho(x) \in C^f(\mathbf{v}')\big].$$

Pick $x = a$ - the case $x = b$ can be done similarly. Suppose $a \in C^f(\mathbf{v})$ but assume for contradiction that $b \notin C^f(\mathbf{v}')$. We consider two cases.

Case 1. Suppose $a \in C^f(\mathbf{v}')$. We consider three type profiles $\mathbf{u}, \mathbf{u}', \mathbf{u}''$ as follows:

$$u(b) = v(b),\ u(y) = v(a)\ \forall\ y \neq b$$
$$u'(a) = u'(b) = v(b),\ u(y) = v(a)\ \forall\ y \notin \{a,b\}$$
$$u''(a) = v(b),\ u(y) = v(a)\ \forall\ y \neq a$$

Type profiles $\mathbf{v}, \mathbf{v}', \mathbf{u}, \mathbf{u}', \mathbf{u}''$ are shown in Table 9.

We first argue that every $x \neq b$, we have $x \in C^f(\mathbf{u})$. If that is not the case, then ETEA implies every $x \neq b$, we have $x \notin C^f(\mathbf{u})$. But that will mean that $C^f(\mathbf{u}) = \{b\}$. Hence, we have $b \in C^f(\mathbf{u})$ and $a \notin C^f(\mathbf{u})$ but $a \in C^f(\mathbf{v})$. This contradicts (2) of Fact 4.

Hence, for every $x \neq b$, we have $x \in C^f(\mathbf{u})$. Next, we argue that for every $x \notin \{a, b\}$, we have $x \in C^f(\mathbf{u}')$. If that is not the case, ETEA will imply that every $x \notin \{a, b\}$, we have

Valuations	a	b	$y \notin \{a,b\}$
$\mathbf{v}$	$v(a)$	$v(b)$	$v(y)$
$\mathbf{u}$	$u(a) = v(a)$	$u(b) = v(b)$	$u(y) = v(a)$
$\mathbf{u}'$	$u'(a) = v(b)$	$u'(b) = v(b)$	$u'(y) = v(a)$
$\mathbf{u}''$	$u''(a) = v(b)$	$u''(b) = v(a)$	$u''(y) = v(a)$
$\mathbf{v}'$	$v'(a) = v(b)$	$v'(b) = v(a)$	$v'(y) = v(y)$

Table 9: Type profiles $\mathbf{v}, \mathbf{v}'$, $\mathbf{u}$, $\mathbf{u}'$ and $\mathbf{u}''$.

$x \notin C^f(\mathbf{u}')$. Further, ETEA implies that $C^f(\mathbf{u}') = \{a,b\}$. But then for some $x \notin \{a,b\}$, we have $b \in C^f(\mathbf{u}'), x \notin C^f(\mathbf{u}')$, but $x \in C^f(\mathbf{u})$. This is a contradiction to (2) of Fact 4.

Hence, for every $x \notin \{a,b\}$, we have $x \in C^f(\mathbf{u}')$. Next, we argue that for all $x \neq a$, we have $x \in C^f(\mathbf{u}'')$. If that is not the case, ETEA will imply that $C^f(\mathbf{u}'') = \{b\}$. But for some $x \notin \{a,b\}$, $b \in C^f(\mathbf{u}''), x \notin C^f(\mathbf{u}'')$ and $x \in C^f(\mathbf{u}')$. This contradicts (2) of Fact 4.

Hence, for all $x \neq a$, we have $x \in C^f(\mathbf{u}'')$. But then, $a \in C^f(\mathbf{v}'), b \notin C^f(\mathbf{v}')$ and $b \in C^f(\mathbf{u}'')$. This contradicts (2) of Fact 4 and concludes the proof for this case.

Case 2. Suppose $a \notin C^f(\mathbf{v}')$. Then, there is some $x \notin \{a,b\}$ such that $x \in C^f(\mathbf{v}')$. By Step 1, we have $x \in C^f(\mathbf{v})$. In this case, we consider one more type profile $\mathbf{v}''$:

$$v''(a) = v''(b) = v(a), v''(y) = v(y) \;\forall\; y \notin \{a,b\}.$$

Type profiles $\mathbf{v}, \mathbf{v}', \mathbf{v}''$ are shown in Table 10.

Valuations	a	b	$y \notin \{a,b\}$
$\mathbf{v}$	$v(a)$	$v(b)$	$v(y)$
$\mathbf{v}''$	$v''(a) = v(a)$	$v''(b) = v(a)$	$v''(y) = v(y)$
$\mathbf{v}'$	$v'(a) = v(b)$	$v'(b) = v(a)$	$v'(y) = v(y)$

Table 10: Type profiles $\mathbf{v}, \mathbf{v}'$, and $\mathbf{v}''$.

Assume for contradiction $a \notin C^f(\mathbf{v}'')$. By ETEA, $b \notin C^f(\mathbf{v}'')$. So, some $y \notin \{a,b\}$ will satisfy $y \in C^f(\mathbf{v}'')$. But this will contradict Fact 4 since $a \in C^f(\mathbf{v})$. So, using ETEA, $a, b \in C^f(\mathbf{v}'')$. Since $x \in C^f(\mathbf{v})$, (1) of Fact 4 implies that $x \in C^f(\mathbf{v}'')$. But $x \in C^f(\mathbf{v}')$ along with (1) of Fact 4 implies that $b \in C^f(\mathbf{v}')$. This is a contradiction.

This concludes the proof that for every $x \in \{a,b\}$, we have $x \in C^f(\mathbf{v})$ implies $\rho(x) \in C^f(\mathbf{v}')$. If $x \notin C^f(\mathbf{v})$ but $\rho(x) \in C^f(\mathbf{v}')$, then we can just replace the roles of $\mathbf{v}$ and $\mathbf{v}'$ along with x

and $\rho(x)$ in the above argument to get a similar contradiction. Hence, for every $x \in \{a, b\}$,

$$\big[x \in C^f(\mathbf{v})\big] \Leftrightarrow \big[\rho(x) \in C^f(\mathbf{v}')\big].$$

We will now argue that this equivalence argument goes through even when $|J| > 1$ with some additional notation. We only illustrate the argument for (1) $\Rightarrow$ (2), and the rest of the implications are similar. Suppose F is implementable and neutral. Suppose it fails component-wise Pareto. Then, there exists some $j \in J$ and some v^{-j} such that for all $i \in N$, $v_i^j(b^j) > v_i^j(a^j)$ for some $b^j, a^j \in A^j$ and $F^j(\mathbf{v}) = a^j$. Define the *marginal* allocation rule $f : \mathbb{R}_{++}^{|A_j| \times n} \to A^j$ as follows: $f(v'^j) = F^j(v'^j, v^{-j})$ for all $v'^j \in \mathbb{R}_{++}^{|A^j| \times n}$. Notice that since f defines a one-component allocation rule. Further, since F satisfies CPAD, f satisfies PAD and since F satisfies neutrality, f satisfies neutrality. Hence, our earlier equivalence implies that f satisfies Pareto. This is a contradiction since $f(v^j) = a^j$. ■

CPAD does not imply PAD

We give an example of an allocation rule which satisfies CPAD but not PAD. Suppose $N = \{1\}$ and $J = \{1, 2\}$ with $A^1 = \{a^1, b^1\}$ and $A^2 = \{a^2, b^2\}$ - the example can be easily extended to have more than one agent. At any valuation profile $\mathbf{v}$, the allocation rule chooses

$$F^1(\mathbf{v}) = \begin{cases} a^1 & \text{if } v_1^1(a^1) > v_1^1(b^1) \\ b^1 & \text{otherwise} \end{cases}$$

For component 2,

$$F^2(\mathbf{v}) = \begin{cases} a^2 & \text{if } v_1^2(a^2) - v_1^2(b^2) > v_1^1(a^1) - v_1^1(b^1) \\ b^2 & \text{otherwise} \end{cases}$$

Notice that the choice in component 2 depends on the valuations of component 1. It is also not difficult to see that this allocation rule satisfies CPAD: if a particular alternative is picked on a component, and its difference with respect to the other alternative increases (keeping the values on other component fixed), the same alternative will be picked on this component. However, this allocation rule violates PAD. Consider two valuations of agent 1 as shown below in Table 11. Verify that $F(v_1) = (a_1, a_2)$ and $F(v_1') = (a_1, b_2)$. Hence, PAD is violated.

References

Barbera, S. (2010): "Strategy-proof Social Choice," in *Handbook of Social Choice and Welfare*, ed. by K. J. Arrow, A. K. Sen, and K. Suzumura, North-Holland, 441–449.

	a_1	b_1	a_2	b_2
v_1	5	4	6	4
v_1'	10	4	7	4

Table 11: Valuations of agent 1

Barbera, S., F. Gul, and E. Stacchetti (1993): "Generalized Median Voter Schemes and Committees," *Journal of Economic Theory*, 61, 262–289.

Barberà, S., H. Sonnenschein, and L. Zhou (1991): "Voting by committees," *Econometrica*, 595–609.

Carbajal, J. C., A. McLennan, and R. Tourky (2013): "Truthful Implementation and Aggregation in Restricted Domains," *Journal of Economic Theory*, 148, 1074–1101.

Chen, Y.-C., W. He, J. Li, and Y. Sun (2016): "Equivalence of stochastic and deterministic mechanisms," Working paper, National University of Singapore.

Chung, K.-S. and W. Olszewski (2007): "A Non-Differentiable Approach to Revenue Equivalence," *Theoretical Economics*, 2, 1–19.

Dobzinski, S. and N. Nisan (2009): "A Modular Approach to Roberts' Theorem," in *In Proceedings of the 2nd International Symposium on Algorithmic Game Theory (SAGT 2009)*, Springer (Lecture Notes in Computer Science).

——— (2011): "Multi-Unit Auctions: Beyond Roberts," in *Forthcoming, Proceedings of* 12^{th} *ACM Conference on Electronic Commerce*, ACM Press.

Jehiel, P., M. M. ter Vehn, and B. Moldovanu (2008): "Ex-Post Implementation and Preference Aggregation via Potentials," *Economic Theory*, 37, 469–490.

Lavi, R., A. Mualem, and N. Nisan (2009): "Two Simplified Proofs of Roberts' Theorem," *Social Choice and Welfare*, 32, 407–423.

Le Breton, M. and J. A. Weymark (1999): "Strategy-proof social choice with continuous separable preferences," *Journal of Mathematical Economics*, 32, 47–85.

Le Breton, M. L. and A. Sen (1999): "Separable preferences, strategyproofness, and decomposability," *Econometrica*, 67, 605–628.

Marchant, T. and D. Mishra (2015): "Mechanism design with two alternatives in quasi-linear environments," *Social Choice and Welfare*, 44, 433–455.

Mishra, D. and A. Quadir (2014): "Non-bossy single object auctions," *Economic Theory Bulletin*, 2, 93–110.

Mishra, D. and A. Sen (2012): "Roberts' Theorem with Neutrality: A Social Welfare Ordering Approach," *Games and Economic Behavior*, 75, 283–298.

Nath, S. and A. Sen (2015): "Affine maximizers in domains with selfish valuations," *ACM Transactions on Economics and Computation*, 3, 26.

Reny, P. (2001): "Arrow's Theorem and the Gibbard-Satterthwaite Theorem: A Unified Approach," *Economics Letters*, 70, 99–105.

Roberts, K. (1979): *The Characterization of Implementable Choice Rules*, North Holland Publishing, chap. Aggregation and Revelation of Preferences, 321–348, editor: J-J. Laffont.

Rochet, J. C. (1987): "A Necessary and Sufficient Condition for Rationalizability in a Quasi-linear Context," *Journal of Mathematical Economics*, 16, 191–200.

Svensson, L.-G. and P. Torstensson (2008): "Strategy-proof allocation of multiple public goods," *Social Choice and Welfare*, 30, 181–196.

Vohra, R. V. (2011): *Mechanism Design: A Linear Programming Approach*, Cambridge University Press.

Weymark, J. A. (1999): "Decomposable Strategy-Proof Social Choice Functions," *Japanese Economic Review*, 50, 343–355.