Processing Reserves Simultaneously*

David Delacrétaz†

Job Market Paper

October 16, 2020

Latest Version: www.daviddelacretaz.net/jmp

Abstract

Policymakers frequently use reserve categories to combine competing objectives in allocating scarce resources based on priority. For example, schools may prioritize students from underprivileged backgrounds for some of their seats while allocating the rest of them based solely on academic merit. The order in which different categories are processed has been shown to have an important, yet subtle impact on allocative outcomes—and has led to unintended consequences in practice. I introduce a new, more transparent way of processing reserves, which handles all categories simultaneously. I provide an axiomatic characterization of my solution, showing that it satisfies basic desiderata as well as *category neutrality*: if an agent qualifies for n categories, she takes $1/n$ units from each of them. A practical advantage of this approach is that the relative importance of categories is entirely captured by their quotas.

Keywords: rationing problem, reserve system, simultaneous processing, category neutrality

JEL Classifications: C62, C78, D47, D61, D63.

*I am grateful to Ravi Jagadeesan, Bernhard Kasberger, Paul Klemperer, Scott Kominers, Simon Loertscher, Alex Teytelboym, Tayfun Sönmez, M. Bumin Yenmez, Özgür Yılmaz, and (online) seminar participants from the Lab for Economic Design at Harvard University and the Microeconomic Theory Group at the University of Oxford for valuable comments and suggestions.

†Department of Economics and Nuffield College, University of Oxford; Nuffield College, New Road, Oxford OX1 1NF; david.delacretaz@economics.ox.ac.uk.

1 Introduction

If a good is in short supply, who should have access to it? The COVID-19 crisis has highlighted the importance of rationing rules—for example, to allocate ventilators or vaccine doses—in situations where demand exceeds supply and it is not possible to use a price mechanism to equate them. The simplest approach is to use a priority order and allocate the good to whomever has the highest priority. For example, medical practitioners have guidelines to determine who should receive a treatment based, among other factors, on who is likely to benefit from it the most. Likewise, many cities (e.g., New York and Boston) use a priority system to allocate seats in public schools, based on a range of administrative rules.[1] *Reserve systems* constitute a more flexible approach, as they allow multiple priority orders, each of which applies to part of the capacity. For example, in an attempt to ensure classroom diversity, each public school in Chicago reserves 70% of its seats for students from specific neighborhoods (Dur, Pathak, and Sönmez, 2020). Since 2005, the US Customs and Immigration Service has reserved 20,000 H-1B visas each year for applicants with an advanced degree, while the remaining 65,000 are open to all applicants (Pathak, Rees-Jones, and Sönmez, 2020a). Such a system allows those applicants who have an advanced degree to be favored without closing the door to those who do not. Recently, Pathak, Sönmez, Ünver, and Yenmez (2020) have proposed using a reserve system to allocate medical resources (e.g., ventilators, ICU beds, drugs, and vaccines) to reconcile various ethical values. For example, there is a view in the medical community that essential personnel, such as frontline healthcare workers, should be given higher priority for medical resources, but doing so would risk denying goods to those patients who need them most. Reserving some capacity for essential personnel and allocating the rest based on need would allow both objectives to be reconciled.

In this paper, I propose a new solution for allocating a scarce resource through a reserve system. I consider a standard rationing problem in which a certain number of *units* (e.g., ventilators, school places, or visas) have to be allocated to *agents* (e.g., patients, students, or visa applicants) and the units are split into *(reserve) categories* (e.g., essential personnel, neighborhood tiers, advanced degree holders, or an open category), each of which has its own priority order over the agents. The novelty of my proposed solution is that the categories are processed simultaneously rather than sequentially. In a sequential reserve system, categories are processed one at a time following a *precedence order.* Each category allocates its *quota* (the number of units reserved for that category) to its highest-priority agents who have not yet received a unit. The precedence order impacts the allocation because an agent who qualifies

[1]See Pathak, Sönmez, Ünver, and Yenmez (2020) and Pathak (2011) for a comprehensive overview of triage and school choice practices, respectively, in the United States.

for multiple categories counts toward the quota of whichever category is processed first; thus, the other categories for which that agent qualifies have an additional unit to allocate to their next highest-priority agent. For that reason, all else equal, categories processed later tend to matter more. In contrast, the simultaneous reserve system I propose treats categories symmetrically and does not rely on any precedence order. Categories simultaneously allocate units to their highest-priority agents until their quotas are filled. If an agent is allocated a unit from, say, n categories, she only takes $1/n$ units of capacity from each, allowing these categories to allocate more capacity to agents further down their respective priority orders.

The effect of changing the precedence order on the allocation is not negligible; in fact, it can be of similar magnitude to the size of the quotas (Dur, Kominers, Pathak, and Sönmez, 2018). However, the role played by the precedence order in determining the outcome is counterintuitive and often misunderstood by policymakers and participants. In an experimental study, Pathak, Rees-Jones, and Sönmez (2020b) find that a large proportion of subjects reacted optimally to a change in quotas but ignored the impact of the precedence order in which categories are processed. Such mistakes are also well documented in the field. As Dur, Kominers, Pathak, and Sönmez (2018) report, the City of Boston established in 1999 a 50-50 seat split for its public schools: half of each school's seats were reserved for students living within walking distance, while the other half were open to all students. The policy was seen as a compromise between a neighborhood school and a school choice system. However, in practice, the "walk zone" reserve had almost no impact because it was processed first. Pathak, Rees-Jones, and Sönmez (2020a) document a similar story for the H-1B visa program. Procedure changes made for logistical reasons in 2005 and 2009 had unintended consequences due to the precedence order of the "advanced degree" and "all applicants" categories. Even if policymakers are made aware of the issue, finding the right combination of quotas and precedence order to achieve a given distributional goal and ensuring that market participants understand how the system works remain challenging tasks (Pathak, Rees-Jones, and Sönmez, 2020b, pp.4-5). In fact, once Boston Public Schools was informed of the reason the reserve system was not producing the intended outcome, it abandoned the system altogether, in large part due to concerns over the lack of transparency associated with the precedence order (Dur, Kominers, Pathak, and Sönmez, 2018). As the solution presented in this paper does not rely on a precedence order, the relative importance of categories is entirely determined by the quotas. Thus, the system is easier to design for policymakers and more transparent for market participants.

Theoretical Contribution

I introduce the *simultaneous reserve (SR) algorithm* (Algorithm 1). In each round, categories allocate their quotas to their respective highest-priority agents. If an agent is allocated more than one unit in aggregate (over all categories), then the amount that she receives from each category is reduced until she is allocated exactly one unit in aggregate. As a result, some categories have additional capacity, which they can allocate in the next round to agents further down their respective priority orders. Once no category has any additional capacity to allocate, the algorithm has found an allocation. While the SR algorithm may run for infinitely many rounds without finding an allocation, I show that it always converges to one (Theorem 1). I call the allocation to which the SR algorithm converges the *simultaneous reserve (SR) allocation*. The SR allocation is well defined and specifies how much capacity each category allocates to each agent. In contrast to most of the literature, these numbers do not have to be binary, so an agent can be allocated one unit in aggregate but receive parts of that unit from different categories.

I analyze the properties of the SR allocation. I show that it satisfies three standard axioms introduced by Pathak, Sönmez, Ünver, and Yenmez (2020)—*compliance with eligibility requirements*, *nonwastefulness*, and *respect of priorities*—as well as a fourth one that I call *category neutrality* (Theorem 2). An allocation is category neutral if every agent who qualifies for multiple categories receives the same amount of capacity from all of them. I show that the SR allocation may not be the only allocation to satisfy all four axioms; however, any other allocation that does generates the same *aggregate allocation*. That is, at any two allocations that satisfy the four axioms, every agent is allocated the same amount of capacity in aggregate (Theorem 3). Therefore, differences among allocations that satisfy all four axioms amount to a matter of accounting and do not have any tangible impact for agents. Among the allocations that satisfy all four axioms, I characterize the SR allocation as the one in which the maximum capacity that each agent is allocated from a single category is largest (Theorem 4).

The fact that the SR algorithm may run for infinitely many rounds constitutes a clear impediment to practical application. Following a similar approach to that of Kesten and Ünver (2015), I remedy the situation by using linear programming. The resulting *simultaneous reserve with linear programming (SRLP)* algorithm produces the SR allocation in finitely many rounds and polynomial time (Theorem 5).

Related Literature

This paper builds upon a rich literature on allocation problems with distributional constraints. In the school choice setting of Abdulkadiroğlu and Sönmez (2003), Abdulkadiroğlu (2005) proposes a solution to incorporate affirmative action through maximum quotas on specific types of students, and Kojima (2012) shows that maximum quotas on majority students can have unintended adverse consequences for the minority students that the policy aims to favor. Hafalir, Yenmez, and Yildirim (2013) propose using minimum quotas instead, which they show provide better outcomes on average for minority students. Their solution can be thought of as a reserve system with two categories: a "minority" category that ranks minority students first and an "open" category that ranks students based on the usual priorities, with the minority category processed first. Westkamp (2013) develops a solution based on minimum quotas for the German university admission system. Ehlers, Hafalir, Yenmez, and Yildirim (2014) and Echenique and Yenmez (2015) extend the approach to include multiple "minorities". As each student belongs to at most one minority, the precedence order of minority categories does not impact the outcome; however, the minimum quota approach implies that the "open" category must be processed last. Gonczarowski, Kovalio, Nisan, and Romm (2019) use a combination of minimum and maximum quotas in their design of a centralized matching market for Israeli gap-year programs.

Kominers and Sönmez (2016) formally introduce a reserve system with sequential processing. Their setting generalizes those of Kojima (2012) and Hafalir, Yenmez, and Yildirim (2013), as it allows for any priorities and precedence order. Dur, Kominers, Pathak, and Sönmez (2018) identify the importance of the precedence order and show that it explains why the walk zone reserve in Boston's public schools had almost no impact, a finding that led to the elimination of that system. Pathak, Rees-Jones, and Sönmez (2020a) document how seemingly innocuous changes to the H-1B visa program in the United States had important consequences because of sequential processing. Sequential reserve systems have been studied in various contexts, including Chicago's public schools (Dur, Pathak, and Sönmez, 2020) as well as university admissions in India (Sönmez and Yenmez, 2019a,b; Aygün and Turhan, 2020a,b) and Brazil (Aygün and Bó, 2020). Pathak, Sönmez, Ünver, and Yenmez (2020) propose using such a system for the allocation of medical resources. Pathak, Rees-Jones, and Sönmez (2020b) experimentally study the decision-making of subjects who must optimize over combinations of reserve quotas and precedence orders and show that very few participants do so optimally. The present paper departs from the literature on sequential reserve systems by proposing a simultaneous reserve system that does not rely on any precedence order and treats all categories identically.

Two approaches that do not rely on a sequential reserve system have recently been proposed. Yılmaz (2020) develops a solution concept independent of any precedence order that satisfies basic axioms and is as egalitarian as possible in terms of the share (between zero and one) that each agent is allocated in aggregate. The SR allocation satisfies Yılmaz's (2020) axioms but pursues a different goal of treating categories identically. In fact, most agents (all but at most as many agents as there are categories) are allocated either zero units or one unit (Proposition 5). The most closely related approach to that of the current paper is the *horizontal envelope* algorithm of Sönmez and Yenmez (2020), which Pathak, Sönmez, Ünver, and Yenmez (2020) generalize to develop the *smart reserves* algorithm. I discuss the differences and complementarity between this approach and my own in Section 4.5.

From a methodological point of view, the present paper also relates to the literature on random and probabilistic serial assignment, initiated by Hylland and Zeckhauser (1979) and Bogomolnaia and Moulin (2001) and generalized by Budish, Che, Kojima, and Milgrom (2013). The SR allocation resembles a random assignment in the sense that each category allocates to each agent a capacity between zero and one. However, agents do not "eat" categories in hopes of being assigned one of them. Rather, categories allocate capacity to agents based on their priority orders, and when an agent is allocated one unit in aggregate, that unit may be shared among different categories. Kesten and Ünver (2015) consider a school choice model with priority ties. As a result, a school may assign to a student an amount of capacity strictly between zero and one, and a student may be allocated parts of a seat by different schools. Students have preferences over schools and must ultimately be assigned to one of them; in contrast, in this paper, agents do not have preferences over categories and can be allocated parts of a unit from different categories.

The remainder of the paper is organized as follows. Section 2 presents a motivating example. Section 3 introduces the setup and the four axioms. Section 4 develops the SR algorithm and analyzes the properties of the SR allocation. Section 5 presents the SRLP algorithm and shows that it produces the SR allocation in polynomial time. Section 6 concludes, and all proofs are in the appendix.

2 Motivating Example

I present a simple example to illustrate how, in contrast to sequential processing, simultaneous processing ensures that all categories matter equally.[2] A school has four seats, two of which

[2]The example is inspired by the system in place for Boston's public schools until 2013, and for concreteness, I use the terminology of that application (see Dur, Kominers, Pathak, and Sönmez (2018) for a study of that

Walk	(2)
Amy	1
Bob	1
Eric	
Fiona	
Claire	
Dan	

Open	(2)
Amy	
Bob	
Claire	1
Dan	1
Eric	
Fiona	

(a) Walk category processed first.

Walk	(2)
Amy	
Bob	
Eric	1
Fiona	1
Claire	
Dan	

Open	(2)
Amy	1
Bob	1
Claire	
Dan	
Eric	
Fiona	

(b) Open category processed first.

Table 1: Sequential processing in the motivating example.

Walk	(2)
Amy	1/2
Bob	1/2
Eric	1
Fiona	
Claire	
Dan	

Open	(2)
Amy	1/2
Bob	1/2
Claire	1
Dan	
Eric	
Fiona	

Table 2: Simultaneous processing in the motivating example.

are reserved for students living within walking distance of the school, while the other two are open to all students. There are six students (Amy, Bob, Claire, Dan, Eric, and Fiona), and four of them (Amy, Bob, Eric, and Fiona) live within walking distance of the school. There is general priority order over the students, which I assume to be alphabetical. The "walk" category gives higher priority to students who live within walking distance of the school and then ranks students based on the general priority order. The "open" category ranks students according to their general priority. Thus, the priority order for the walk category is Amy, Bob, Eric, Fiona, Claire, and Dan, and the priority order for the open category is Amy, Bob, Claire, Dan, Eric, and Fiona.

Consider sequential processing, and suppose that the walk category is processed first. That category allocates its seats to its two highest-priority students, Amy and Bob. We then move to the open category, whose highest-priority students (Amy and Bob) have already been allocated a seat. Hence, the open category allocates its two seats to its next highest-priority students, Claire and Dan. Table 1a summarizes the outcome. The four students with the highest general priority—Amy, Bob, Claire, and Dan—are allocated a seat; hence, the same outcome would have been achieved without a reserve.[3] Suppose now that the open category

specific problem); however, the same example could be framed in different contexts, such as a hospital allocating ventilators to patients and reserving some of them for essential personnel.

[3]This example illustrates why the walk zone reserve in Boston had very little effect on the outcome as a

is processed first. That category allocates its two seats to Amy and Bob, its highest-priority students. As Amy and Bob have already been allocated a seat, the walk category allocates its two seats to Eric and Fiona. Table 1b summarizes the outcome. The students who live within walking distance of the school—Amy, Bob, Eric, and Fiona—are all allocated a seat; hence, the same outcome would have been achieved by reserving all four seats.

The precedence order has a large impact in this example, as it determines the allocation of half of the seats. Moreover, both outcomes are extreme in the sense that they each follow the priority of one category and ignore the other category. In contrast, as I next show, simultaneous processing yields an intermediate solution that accounts for both categories. The two categories simultaneously allocate their two seats to their highest-priority students. Hence, Amy and Bob each receive two seats, one from each category. As each student only requires one seat, Amy and Bob only keep half a seat from each category. Of course, seats are indivisible, and Amy and Bob each receive one; however, for accounting purposes, the seats they receive are split between the two categories. Each category has a quota of two and allocates one seat overall (1/2 to Amy and 1/2 to Bob); therefore, they each have one seat left for their respective third-priority students. The walk category allocates its second seat to Eric, and the open category allocates its second seat to Claire. The resulting allocation is displayed in Table 2. Amy, Bob, Claire, and Eric are allocated a seat. Both categories are equally important in the outcome: Amy and Bob are allocated a seat because they qualify for both categories, Claire is allocated a seat through the open category, and Eric is allocated a seat through the walk category.

3 Preliminaries

3.1 Setup

There are a set of **agents** A with typical element a, a set of **(reserve) categories** C with typical element c, and $q \in \mathbb{Z}_{>0}$ identical and indivisible **units**. Each category c has a **quota** $q_c \in \mathbb{R}_{\geq 0}$ with $\sum_{c \in C} q_c = q$ as well as a linear **priority order** π_c over the set of agents and an eligibility threshold $\emptyset$. Agent a is **eligible** for category c if $a \pi_c \emptyset$. For every agent a and every category c, I denote by $\hat{A}_{a,c} = \{a' \in A : a' \pi_c a\}$ the set of agents who have a higher priority than a for c and by $\check{A}_{a,c} = \{a' \in A : a \pi_c a'\}$ the set of agents who have a lower priority than a for c. A **rationing problem** is a tuple $R = (A, C, (\pi_c)_{c \in C}, (q_c)_{c \in C})$ specifying a set of agents, a set of categories, and, for each category, a priority order and quota. I say that the rationing problem R has **soft reserves** if every agent is eligible for every category, i.e.,

result of being processed before the open category (see Dur, Kominers, Pathak, and Sönmez, 2018).

if $a \pi_c \emptyset$ for every $a \in A$ and every $c \in C$; otherwise, I say that R has **hard reserves**.

A **(random) allocation** is an $|A| \times |C|$ matrix $\xi = (\xi_{a,c})_{a \in A, c \in C}$ such that for every agent a and every category c, (i) $\xi_{a,c} \in [0,1]$, (ii) $\sum_{a' \in A} \xi_{a',c} \leq q_c$, and (iii) $\sum_{c' \in C} \xi_{a,c'} \leq 1$. In words, each element $\xi_{a,c}$ specifies the amount of capacity (between zero and one) that category c allocates to agent a; each category allocates a total amount of capacity no larger than its quota; and each agent is allocated at most one unit in aggregate. For every agent a, I denote by $\xi_a = \sum_{c \in C} \xi_{a,c}$ the aggregate amount of capacity allocated to a at the allocation ξ. A **(random) aggregate allocation** is a vector $\rho = (\rho_a)_{a \in A}$ such that $\rho_a \in [0,1]$ for every agent a and $\sum_{a \in A} \rho_a \leq q$. In words, each element specifies the amount of capacity allocated to agent a in aggregate.[4] I denote by $\rho(\xi) = (\xi_a)_{a \in A}$ the aggregate allocation generated by the allocation ξ.

3.2 Reserve Systems in Practice

I consider three main applications: the allocation of medical resources, school seats, and immigration visas.

In the health care rationing problem of Pathak, Sönmez, Ünver, and Yenmez (2020), each unit is a unit of a medical resource, for example, a ventilator or a dose of a vaccine, while each agent is a patient who requires that medical resource. There is a general category in which patients are prioritized based on their medical situation (typically, expected health outcome and survival probability). Pathak, Sönmez, Ünver, and Yenmez (2020) also propose three possible categories, each of which would prioritize a group of patients. An *essential personnel* category would prioritize those patients whose activity is essential during a health emergency, for example, frontline healthcare workers. A *disadvantaged* category would prioritize patients in groups that are particularly affected by the crisis, and a *Good Samaritan reciprocity* category would prioritize agents whose selfless acts—for example, donating a kidney to a stranger, donating a large amount of blood, or participating in clinical trials—have saved lives in the past. In each of those categories, patients would first be ranked based first on whether they are part of the target group and then their medical situation.[5] Last, Pathak, Sönmez, Ünver, and Yenmez (2020) argue that prioritizing patients based on their medical situation may be discriminatory to certain groups, who might not have access to the medical resource at all. The authors propose creating a *disabled* category that does not take the general medical situation into account but rather prioritizes patients who have a disability

[4] Ultimately, each agent must be allocated either zero units or one unit. In Section 4.4, I consider solutions for the case in which some agents are allocated an amount of capacity strictly between zero and one.

[5] For the Good Samaritan reciprocity category, one could instead consider ranking patients based on the size of their past contribution.

and breaks ties with a lottery. For the purpose of allocating COVID-19 vaccines, Persad, Peek, and Emanuel (2020) propose reserving half of the doses for frontline healthcare workers and a quarter for people working or living in high-risk settings and leaving the remaining quarter unreserved. It is natural to think of a health care rationing problem as one with soft reserves because it is typically better to allocate a unit to a low-priority patient than not at all.

The allocation of seats in a public school constitutes another application: each unit is a seat at a given public school, and each agent is a student. Until 2013, Boston had a reserve system with two categories, each of which had a quota of half the total number of seats (Dur, Kominers, Pathak, and Sönmez, 2018). The open category prioritized students who had a sibling attending the school and then broke ties with a lottery, while the *walk zone* category prioritized students who lived within walking distance of the school and broke ties with the open category priority. In Chicago (Dur, Pathak, and Sönmez, 2020), the city's neighborhoods are split into four *tiers* based on socioeconomic factors. There is an open category that prioritizes students based on academic merit and four tier-specific categories (one per tier), which prioritize students who live in one of the tier's neighborhoods. The open category quota is equal to 30% of the school's seats, and each of the tier-specific categories has a quota of 17.5% of the seats. This application constitutes another example of a rationing problem with soft reserves: all seats are allocated as long as there are at least as many students as there are seats.

Pathak, Rees-Jones, and Sönmez (2020a) study a reserve system for the allocation of H-1B visas in the United States. Each unit is an H-1B visa, each agent is an applicant, and there are two categories: an open category with a quota of $65,000$ visas and an *advanced degree* category with a quota of $20,000$ visas. There are two separate lotteries, each of which determines the priority order of one of the categories. Only applicants with an advanced degree are eligible for the advanced degree category; therefore, the allocation of H-1B visas is an example of a rationing problem with hard reserves: even if the advanced degree category allocates fewer than $20,000$ visas, the remaining ones cannot be allocated to applicants who do not have an advanced degree.

Finally, a practically relevant aspect of the model is worth mentioning. While the total number of units is an integer, there is no such restriction on the category quotas. To the best of my knowledge, this paper is the first to provide that feature in the context of reserves. The additional flexibility may prove useful in practice, particularly when the number of units is small; for instance, if the policy is to allocate 30% of ventilators to essential personnel and 70% to the general population, a hospital with 5 ventilators can set the quotas to 1.5 and 3.5.

3.3 Desirable Properties of an Allocation

Pathak, Sönmez, Ünver, and Yenmez (2020) introduce three axioms that an allocation should satisfy: *compliance with eligibility requirements*, *nonwastefulness*, *and respect of priorities.* An important difference between my setting and that of Pathak, Sönmez, Ünver, and Yenmez (2020) is that they consider allocations (or *matchings* in their terminology) in which each element is either zero units or one; that is, each agent is either not allocated anything or is allocated one unit from one category. I generalize the three properties of Pathak, Sönmez, Ünver, and Yenmez (2020) to my setting and introduce a fourth one—*category neutrality*—that lies at the heart of the solution I propose.

Axiom 1. *An allocation ξ* ***complies with eligibility requirements*** *if for every agent a and every category c such that a is not eligible for c, $\xi_{a,c} = 0$.*

The first axiom requires that agents be allocated capacity only by categories for which they are eligible. In a rationing problem with soft reserves (e.g., medical resource rationing or school seat allocation), every allocation trivially complies with eligibility requirements; however, Axiom 1 matters in the presence of hard reserves. For instance, in the H-1B visa program, Axiom 1 precludes applicants who do not have an advanced degree from being allocated one of the $20,000$ visas reserved for advanced degree applicants.

Axiom 2. *An allocation ξ is* ***nonwasteful*** *if for every category c such that $\sum_{a' \in A} \xi_{a',c} < q_c$ and every agent a such that $\xi_a < 1$, a is not eligible for c.*

The second axiom states that whenever a category has not allocated its full quota, then none of the remaining capacity can benefit an agent who is eligible for that category, as that capacity would then be wasted. In a rationing problem with soft reserves, Axiom 2 requires that either all agents be allocated a unit or all units be allocated, that is, $\sum_{a \in A} \xi_a = \min\{|A|, q\}$. With hard reserves, some categories may allocate an amount of capacity smaller than their quotas as long as every eligible agent is allocated one unit in aggregate.

Axiom 3. *An allocation ξ* ***respects priorities*** *if for every agent a such that $\xi_a < 1$, every category c and every lower-priority agent $a' \in \check{A}_{a,c}$, $\xi_{a',c} = 0$.*

The third axiom ensures that each category allocates its capacity based on priority; that is, an agent can be allocated capacity from a category only if all higher-priority agents have been allocated one unit in aggregate.

As Pathak, Sönmez, Ünver, and Yenmez (2020, p.13) note, Axioms 1-3 are widely accepted as properties that an allocation should possess.

> "As far as we know, in every real-life application of a reserve system each of these three axioms are either explicitly or implicitly required. Hence, we see these three axioms as a minimal requirement for reserve systems."

While Axioms 1-3 narrow down the set of allocations to be considered, they leave many possible candidates. In particular, these axioms are silent on a key question: if an agent qualifies for multiple categories, from which one(s) should she receive a unit? The most common solution both in practice and in the literature is to use a *sequential reserve algorithm* in which categories are processed one at a time and allocate, until their quotas are filled, one unit of capacity to the highest-priority eligible agents who have not yet been allocated a unit.[6] The implication is that if an agent qualifies for multiple categories, she receives a unit from whichever is processed first; hence, categories processed early tend to allocate units to agents who also qualify for other categories. At the heart of my proposed solution is the idea that while units are ultimately indivisible, how much capacity categories allocate to agents is merely an accounting exercise; therefore, an agent allocated one unit overall can receive parts of that unit from multiple categories. The fourth axiom, which is newly introduced in this paper, stipulates that the unit an agent is allocated should be shared equally among the categories for which she qualifies.

Axiom 4. *An allocation ξ is* ***category neutral*** *if for every agent a and every category c such that a is eligible for c and $\xi_{a,c} < \max_{c' \in C}\{\xi_{a,c'}\}$, $\xi_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} = q_c$.*

Axiom 4 ensures that each agent receives the same amount of capacity from every category with available capacity. In the motivating example, the category neutrality condition dictates that Amy and Bob be allocated half a unit from each of the two categories. However, it does not prevent Claire from being allocated one unit of capacity from the open category and none from the walk category because all of the walk category's quota is allocated to higher-priority agents. From a normative perspective, the category neutrality condition is needed to ensure that all categories are treated the same so that their relative importance only depends on their quotas.

In the next section, I show that processing all categories simultaneously yields an allocation that satisfies Axioms 1-4 and that any other allocation satisfying those properties generates the same aggregate allocation.

[6] See Pathak, Sönmez, Ünver, and Yenmez (2020, p.17) for a full description of that procedure.

Algorithm 1: SIMULTANEOUS RESERVE (SR)

Initialization Set every agent's demand to one: $d_a^0 = 1$ for every agent a.

Round $i \geq 1$:

Capacity Allocation For every agent a and every category c, if a is eligible for c, then set $x_{a,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}, 0\}\}$, and otherwise set $x_{a,c}^i = 0$.

Demand Adjustment For every agent a such that $x_a^i < 1$, set $d_a^i = 1$. For every agent a such that $x_a^i = 1$, set $d_a^i = \max_{c \in C}\{x_{a,c}^i\}$. For every agent a such that $x_a^i > 1$, set d_a^i such that $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = 1$.

4 Simultaneous Reserve (SR) Allocation

4.1 Simultaneous Reserve (SR) Algorithm

The simultaneous reserve (SR) algorithm is formally defined in Algorithm 1. To describe the algorithm and analyze its properties, it is useful to define the concept of a **preallocation**, which is identical to an allocation but allows agents to be allocated more than one unit overall. Formally, a preallocation is an $|A| \times |C|$ matrix $x = (x_{a,c})_{a \in A, c \in C}$ such that for every agent a and every category c, (i) $x_{a,c} \in [0, 1]$ and (ii) $\sum_{a' \in A} x_{a',c} \leq q_c$. I denote by $x_a = \sum_{c \in C} x_{a,c}$ the aggregate amount of capacity allocated to agent a at the preallocation x. Axioms 1-4 are defined analogously over preallocations.

At the start of the SR algorithm, each agent has a *demand* of 1. This can be interpreted as the largest amount that an agent could require from any category; as every agent requires one unit overall, demand starts at 1 but may fall throughout the algorithm as agents are allocated capacity.

The first round starts with the capacity allocation stage: each category allocates one unit of capacity to one agent at a time in decreasing order of priority until it has less than one unit of capacity left or has allocated a unit to every eligible agent, whichever comes first. The next agent receives the remaining capacity (which could be 0 or any number smaller than 1), and the remaining agents are not allocated any capacity.

The capacity allocation stage generates a preallocation x^1, where for any agent-category pair (a, c), $x_{a,c}^1$ is the amount of capacity that c has allocated to a. An agent may be allocated more than one unit overall, so x^1 is a preallocation but not necessarily an allocation. The demand adjustment stage reduces the amount of capacity that agents demand to turn x^1 into an allocation. The demand adjustment stage does not affect agents who have not yet been allocated a unit in aggregate (i.e., a such that $x_a^1 < 1$); these agents continue to demand one unit. The demand of an agent who has been allocated exactly one unit in aggregate (i.e., a such that $x_a^1 = 1$) falls to the maximum capacity she is allocated from any category (i.e., $\max_{c \in C}\{x_{a,c}^1\}$). The rationale is that she does not require more from any category to be allocated one unit in aggregate so any additional capacity for which she qualifies can be allocated to the next agent on the priority order. The demand of an agent who has been allocated more than one unit in aggregate (i.e., a such that $x_a^1 > 1$) falls even further so that she abandons any capacity she does not require and is only allocated one unit in aggregate (as $\sum_{c \in C} \min\{d_a^1, x_{a,c}^1\} = 1$).

Every subsequent Round i starts with a demand vector d^{i-1} and in the capacity allocation stage, a preallocation x^i is calculated. The highest-priority agents are allocated their demand until there is not enough capacity for the next agent. That agent receives whatever capacity remains, and lower-priority agents are not allocated any capacity. For any agent a, the expression $q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}$ represents the amount of capacity remaining once all higher-priority agents have been allocated their demand. If this amount is weakly negative, then there is no capacity left for agent a so she is not allocated any capacity. If it is equal to or larger than d_a^{i-1}, then agent a is allocated her demand. If it is anything in between, then agent a is allocated that remaining capacity. In the demand adjustment stage, d^i is calculated from x^i, and the algorithm continues in Round $i+1$, in which x^{i+1} and d^{i+1} are calculated.

I next illustrate how the SR algorithm can generate an allocation.

Example 1. There are five agents and four categories, each with a quota of 1. Every agent is eligible for every category, and the priorities are

$$\pi_{c_1} : a_1, a_2, a_3, \ldots \quad \pi_{c_2} : a_1, a_2, a_4, \ldots \quad \pi_{c_3} : a_1, a_3, \ldots \quad \pi_{c_4} : a_4, a_5, \ldots.$$

The preallocation calculated in each of the first four rounds of the SR algorithm is displayed in Table 3. In Round 1, each category allocates one unit of capacity to its highest-priority agent. As agent a_1 is allocated a unit from three different categories, her demand drops to 1/3. In Round 2, categories c_1, c_2, and c_3 only allocate 1/3 to a_1, which leaves 2/3 to allocate to their second highest-priority agents. As a result, a_2 is allocated 4/3 in aggregate (2/3 from each of c_1 and c_2), so her demand drops to 1/2. In Round 3, c_1 and c_2 allocate 1/3 to a_1 and 1/2 to a_2; hence, they have 1/6 left to allocate to their third highest-priority agents a_3 and a_4, respectively. Agent a_4 is now allocated 7/6 in aggregate (1/6 from c_2 and 1 from c_4), so her demand drops to 5/6. In Round 4, c_4 only needs to allocate 5/6 to a_4 and can therefore allocate 1/6 to its second highest-priority agent a_5. Every agent is now allocated at most one unit, so the Round 4 preallocation x^4 is in fact an allocation. It is easy to see that

Round 1

c_1	(1)	c_2	(1)	c_3	(1)	c_4	(1)
a_1	1	a_1	1	a_1	1	a_4	1
a_2	0	a_2	0	a_3	0	a_5	0
a_3	0	a_4	0				

Round 2

c_1	(1)	c_2	(1)	c_3	(1)	c_4	(1)
a_1	1/3	a_1	1/3	a_1	1/3	a_4	1
a_2	2/3	a_2	2/3	a_3	2/3	a_5	0
a_3	0	a_4	0				

Round 3

c_1	(1)	c_2	(1)	c_3	(1)	c_4	(1)
a_1	1/3	a_1	1/3	a_1	1/3	a_4	1
a_2	1/2	a_2	1/2	a_3	2/3	a_5	0
a_3	1/6	a_4	1/6				

Round 4

c_1	(1)	c_2	(1)	c_3	(1)	c_4	(1)
a_1	1/3	a_1	1/3	a_1	1/3	a_4	5/6
a_2	1/2	a_2	1/2	a_3	2/3	a_5	1/6
a_3	1/6	a_4	1/6				

Table 3: SR algorithm applied to Example 1.

in any subsequent round, the SR algorithm continues to produce the same (pre)allocation and demand vector. Hence, in Example 1, the SR algorithm produces the allocation

$$x^4 = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{array} \begin{array}{c} \begin{array}{cccc} c_1 & c_2 & c_3 & c_4 \end{array} \\ \begin{pmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/6 & 0 & 2/3 & 0 \\ 0 & 1/6 & 0 & 5/6 \\ 0 & 0 & 0 & 1/6 \end{pmatrix} \end{array},$$

which generates the aggregate allocation

$$\rho(x^4) = \begin{array}{c} \begin{array}{ccccc} a_1 & a_3 & a_3 & a_4 & a_5 \end{array} \\ \begin{pmatrix} 1 & 1 & 5/6 & 1 & 1/6 \end{pmatrix} \end{array}.$$

It is easy to verify that the allocation x^4 satisfies Axioms 1-4. At first sight, it might look as if x^4 is not category neutral because a_4 is allocated 1/6 from c_2 and 5/6 from c_4. However, this does not violate Axiom 4, as c_2 can only allocate 1/6 to a_4 after having allocated 1/3 to a_1 and 1/2 to a_2; formally, $x^4_{a_4,c_2} + \sum_{a \in \hat{A}_{a_4,c_2}} x^4_{a,c_2} = 1/6 + 1/3 + 1/2 = 1 = q_{c_2}$. The fact that the SR algorithm produces an allocation that satisfies Axioms 1-4 is not a coincidence. In the next two subsections, I formally define the outcome of the SR algorithm and show that it is an allocation satisfying Axioms 1-4.

4.2 Outcome of the SR Algorithm

In Example 1, the SR algorithm finds an allocation after four rounds. In general, the SR algorithm may never reach that point; however, its outcome is nevertheless well defined.[7]

Theorem 1. *The SR algorithm converges to an allocation* $\xi^{SR} = \lim_{i\to\infty} x^i$.

I call ξ^{SR} the **simultaneous reserve (SR) allocation** and discuss its properties in Section 4.3. The remainder of this subsection is devoted to proving and providing the intuition for Theorem 1. In Example 1, the SR algorithm finds an allocation after four rounds and continues to return the same allocation and demand vector in every subsequent round. As the next result shows, this always occurs once the SR algorithm has found an allocation.

Proposition 1. *Suppose that in some Round* i *of the SR algorithm,* x^i *is an allocation. Then, for every* $j > i$, $x^j = x^i$ *and* $d^j = d^i$.

The intuition for Proposition 1 is as follows. If x^i is an allocation, then no agent is allocated more than one unit, so the demand of every agent is at least what she is allocated from any category, and so, in the next round, every category continues to allocate the same capacity to every agent, i.e., $x^{i+1} = x^i$. In each round, the demand vector depends on the current round's preallocation, and the preallocation depends on the previous round's demand vector; hence, the SR algorithm continues to return the same (pre)allocation and demand vector in every subsequent round.

Proposition 1 implies that the SR algorithm can stop once it finds an allocation, as that allocation is ξ^{SR}. Unfortunately, the SR algorithm may never reach an allocation.

Example 2. There are four agents and three categories. The priorities and quotas are

$$\pi_{c_1} : a_1, a_2, a_3, a_4, \emptyset \quad \pi_{c_2} : a_3, a_2, a_1, a_4, \emptyset \quad \pi_{c_3} : a_1, a_3, a_2, a_4, \emptyset \quad q_{c_1} = 1 \quad q_{c_2} = 1 \quad q_{c_3} = 2.$$

The working of the SR algorithm is displayed in Table 4. In Round 1, c_1 and c_2 each allocate one unit to their highest-priority agent, a_1 and a_3, respectively. Category c_3 has two units and allocates them to its two highest-priority agents, a_1 and a_3. Agents a_1 and a_3 are each allocated a unit from two different categories; hence, the demand of both agents drops to 1/2. In Round 2, c_1 and c_2 each have an extra half-unit to allocate, which goes to their second highest-priority agent, a_2, while c_3 has an extra unit to allocate to its third highest-priority agent, who is also a_2. As a result, a_2's demand drops to 1/3. In Round 3, c_1 allocates 1/6 to a_3, c_2 allocates 1/6 to a_1, and c_3 allocates 2/3 to a_4.

[7] In Section 5, I propose an outcome-equivalent algorithm that works in polynomial time and finds an allocation after fewer than $4|A||C|$ rounds.

Round 1

c_1	(1)	c_2	(1)	c_3	(2)
a_1	1	a_3	1	a_1	1
a_2	0	a_2	0	a_3	1
a_3	0	a_1	0	a_2	0
a_4	0	a_4	0	a_4	0

Round 2

c_1	(1)	c_2	(1)	c_3	(2)
a_1	1/2	a_3	1/2	a_1	1/2
a_2	1/2	a_2	1/2	a_3	1/2
a_3	0	a_1	0	a_2	1
a_4	0	a_4	0	a_4	0

Round 3

c_1	(1)	c_2	(1)	c_3	(2)
a_1	1/2	a_3	1/2	a_1	1/2
a_2	1/3	a_2	1/3	a_3	1/2
a_3	1/6	a_1	1/6	a_2	1/3
a_4	0	a_4	0	a_4	2/3

Round 4

c_1	(1)	c_2	(1)	c_3	(2)
a_1	5/12	a_3	5/12	a_1	5/12
a_2	1/3	a_2	1/3	a_3	5/12
a_3	3/12	a_1	3/12	a_2	1/3
a_4	0	a_4	0	a_4	5/6

	Round 4	Round 5	Round 6	Round 7	$\cdots$	Round $i \geq 3$
$x^i_{a_3,c_1}$	3/12	7/24	15/48	31/96	$\cdots$	$(2^{i-2}-1)/(3 \cdot 2^{i-2})$
$x^i_{a_1,c_2}$	3/12	7/24	15/48	31/96	$\cdots$	$(2^{i-2}-1)/(3 \cdot 2^{i-2})$
$x^i_{a_4,c_3}$	5/6	11/12	23/24	47/48	$\cdots$	$(3 \cdot 2^{i-3}-1)/(3 \cdot 2^{i-3})$
$d^i_{a_1}$	9/24	17/48	33/96	65/192	$\cdots$	$(2^{i-1}+1)/(3 \cdot 2^{i-1})$
$d^i_{a_3}$	9/24	17/48	33/96	65/192	$\cdots$	$(2^{i-1}+1)/(3 \cdot 2^{i-1})$

Table 4: SR algorithm applied to Example 2.

At this point, the SR algorithm begins to cycle. Agent a_1 is allocated 7/6 in aggregate (i.e., $x^3_{a_1} = 7/6$), so her demand drops in Round 3. However, she can only be allocated 1/6 from c_2, meaning that she needs to be allocated 5/12 from each of c_1 and c_3. It follows that a_1's demand drops to 5/12. Analogously, a_3 is allocated 7/6 in aggregate, and her demand drops to 5/12. In Round 4, as a result of the drop in a_1's and a_3's demand (by 1/12 each), c_1 allocates an extra 1/12 to a_3, c_2 allocates an extra 1/12 to a_1, and c_3 allocates an extra 1/6 to a_4. The extra 1/6 of capacity that a_1 releases benefits a_4 (in c_3) by half and a_3 (in c_1) by half, while the extra 1/6 of capacity that a_3 releases benefits a_4 (in c_3) by half and a_1 (in c_2) by half. As a result, a_1 and a_3 are each allocated 13/12 in aggregate in Round 4, so their demand drops to 9/24. In Round 5, as in Round 4, half of the capacity released by a_1 and a_3 goes to a_4 (c_3 allocates an extra 1/12 to a_4), and the other half comes back to a_1 and a_3 (c_1 allocates an extra 1/24 to a_3 and c_2 allocates an extra 1/24 to a_1). The SR algorithm continues to cycle forever, with the amount of reallocated capacity halving in each round.

However, even though the SR algorithm never reaches an allocation, it converges to one:

$$\xi^{SR} = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \!\!\begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 0 & 0 & 1 \end{pmatrix} \end{array}.$$

I now show that the SR algorithm always converges to an allocation even when it never reaches one. For every Round i of the SR algorithm, I construct the allocation $\xi^i = (\xi^i_{a,c})_{a\in A, c\in C}$ such that for every agent a and every category c, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\}$. By construction, $\xi^i_a \leq 1$ for every agent a so ξ^i is indeed an allocation.[8] I also define the matrix $z^i = x^i - \xi^i$ to be the **excess supply** in Round i of the SR algorithm. For every agent a, $z^i_a = \sum_{c\in C} z^i_{a,c}$ can be interpreted as the capacity that a is allocated in addition to the one unit that she requires. I denote the total excess supply by $|z^i| = \sum_{a\in A} z^i_a = \sum_{a\in A}\sum_{c\in C} z^i_{a,c}$. In Example 2, the total excess supply is 1/3 in Round 3 (a_1 and a_3 are each allocated 7/6) and is halved in every subsequent round; hence, it converges to zero.

Proposition 2. *For every Round $i \geq 1$ of the SR algorithm, $|z^{i+1}| \leq |z^i| \leq |A|(|C|-1)/i$.*

There is a relatively simple intuition for why the total excess supply decreases throughout the SR algorithm. If in some round an agent is allocated more than one unit in aggregate, that extra capacity is reallocated in the next round. It may be reallocated to an agent who was already allocated one unit in aggregate, in which case it continues to count toward the total excess supply, or to an agent who was not yet allocated one unit in aggregate. In the latter case, as agents always keep the capacity that they are allocated up to one unit, the extra capacity no longer counts toward the excess supply in any subsequent round. In Example 2, the excess supply is halved in every Round $i \geq 4$ because half of the excess supply is reallocated to a_1 and a_3, who have already been allocated one unit, and the other half is allocated to a_4, who has not. The intuition for the upper bound is that as excess supply is reallocated, categories allocate capacity to agents further down their priority order. Eventually, categories must reach the bottom, so there is a bound on how much excess supply can be reallocated throughout the algorithm.

Proposition 2 implies that the total excess supply converges to zero. As every element of z^i is weakly positive, each element must also converge to zero. Therefore, x^i and ξ^i must converge to each other, and we have the following corollary.

[8] I prove formally that ξ^i is an allocation in Appendix A (Lemma A.3).

Corollary 1. $\lim_{i\to\infty} z^i = \mathbf{0}$ *and* $\xi^{SR} = \lim_{i\to\infty} \xi^i = \lim_{i\to\infty} x^i$.

Corollary 1 guarantees that ξ^{SR} is well defined. As ξ^i is an allocation in every Round i of the SR algorithm, it is natural to think that its limit ξ^{SR} is an allocation as well. I formally show this in Appendix B, which completes the proof of Theorem 1.

Example 2 has three categories, which leaves open the question of whether an example exists that has only two categories and for which the SR algorithm runs indefinitely without finding an allocation. The next result answers this question in the negative.

Proposition 3. *Suppose that* $|C| = 2$. *Then, the SR algorithm finds the SR allocation after fewer than* $8|A|$ *rounds.*

The reason the SR algorithm does not find an allocation in Example 2 is that in each round, half of the capacity that a_1 and a_3 release goes to a_4 through c_2, while the other half goes back to a_1 and a_3. Such a situation cannot occur with only two categories because the amount of capacity than an agent releases cannot be split among several categories.

Proposition 3 implies that the SR algorithm works in polynomial time when there are only two categories, as the number of rounds required increases linearly with the number of agents. In Section 5, I show that linear programming can be added to the SR algorithm to generalize that lower bound to an arbitrary number of categories.[9]

4.3 Properties of the SR Allocation

Having defined the SR allocation, I now turn to its properties in regard to the axioms defined in Section 3.3.

Theorem 2. *The SR allocation satisfies Axioms 1-4.*

The capacity allocation stage of the SR algorithm is constructed in such a way that, in every Round i, x^i satisfies Axioms 1-3 because each category allocates its capacity to its eligible agents in order of priority. The reason x^i also satisfies Axiom 4 is found in the demand adjustment stage. An agent's demand sets an upper bound on how much capacity each category can allocate to that agent in subsequent rounds; thus, it ensures that all categories that would allocate at least that upper bound allocate the same amount to that agent. Theorem 2 follows from the fact that these properties continue to hold in the limit.

Theorem 2 makes the SR allocation an appealing solution for a rationing problem with reserves. The allocation satisfies the natural requirements in Axioms 1-3 and ensures that

[9] The case in which $|C| = 1$ is trivial: the SR algorithm ends after one round, and the q agents with the highest priority for the unique category are allocated a unit.

an agent who qualifies for multiple categories affects the quotas of each of those categories equally. This is in contrast to most existing solutions, which assign each agent to one category. A natural question at this point is whether the SR allocation is the only one to possess those properties. The next example shows that this is not the case; however, I show that any alternative allocation that satisfies Axioms 1-4 generates the same aggregate allocation.

Example 3. There are two agents and two categories, each with a quota of 1. The priorities are $\pi_{c_1} : a_1, a_2, \emptyset$ and $\pi_{c_2} : a_2, a_1, \emptyset$.

In Example 3, the SR algorithm reaches the allocation

$$\xi^{SR} = \begin{array}{c c} & \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \end{array} & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{array}$$

in Round 1: each category allocates one unit to its highest-priority agent, and no demand adjustment is required. However, for any $\lambda \in [0.5, 1]$, the allocation

$$\xi^{\lambda} = \begin{array}{c c} & \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \end{array} & \begin{pmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{pmatrix} \end{array}$$

satisfies Axioms 1-4. To see this, notice that ξ^{λ} trivially satisfies Axioms 1-3 since every agent is eligible for every category and $\xi^{\lambda}_{a_1} = \xi^{\lambda}_{a_2} = 1$. If $\lambda = 0.5$, then ξ^{λ} is also trivially category neutral since all four elements of ξ^{λ} are equal to 0.5. If $\lambda > 0.5$, then $\xi^{\lambda}_{a_1,c_2} < \xi^{\lambda}_{a_1,c_1}$ and $\xi^{\lambda}_{a_2,c_1} < \xi^{\lambda}_{a_2,c_2}$; however, Axiom 4 is not violated since $\xi^{\lambda}_{a_1,c_2} + \xi^{\lambda}_{a_2,c_2} = 1 = q_{c_2}$ and $\xi^{\lambda}_{a_2,c_1} + \xi^{\lambda}_{a_1,c_1} = 1 = q_{c_1}$.[10]

The SR allocation may not be the sole allocation that satisfies Axioms 1-4; in fact, there may be infinitely many such allocations. However, one aspect of Example 3 is worth noting: for every $\lambda \in [0.5, 1]$, ξ^{λ} allocates one unit of capacity each to a_1 and a_2. That is, $\rho(\xi^{\lambda}) = \rho(\xi^{SR})$ for every $\lambda \in [0.5, 1]$. As the next result shows, this is not specific to Example 3. I call the aggregate allocation $\rho(\xi^{SR})$ generated by the SR allocation the **SR aggregate allocation** and for any allocation ξ, I call ξ **SR equivalent** if it generates the SR aggregate allocation, that is, if $\rho(\xi) = \rho(\xi^{SR})$.

Theorem 3. *Every allocation that satisfies Axioms 1-4 is SR equivalent.*

The significance of Theorem 3 is that even though many allocations may satisfy Axioms 1-4, any difference among them is immaterial, as every agent is allocated the same capacity in

[10]If $\lambda < 0.5$, ξ^{λ} is no longer category neutral since $\xi^{\lambda}_{a_1,c_1} < \xi^{\lambda}_{a_1,c_2}$ and $\xi^{\lambda}_{a_1,c_1} < 1 = q_{c_1}$.

aggregate. Moreover, Theorem 3 implies the following corollary, which characterizes the SR aggregate allocation as the only one to be generated by an allocation satisfying Axioms 1-4.

Corollary 2. *An aggregate allocation is generated by an allocation that satisfies Axioms 1-4 if and only if it is the SR aggregate allocation.*

Finally, Theorem 3 and Corollary 2 are sharp in the sense that each of the four axioms is needed to characterize the SR aggregate allocation.

Proposition 4. *For each of Axioms 1-4, there exists a rationing problem in which an allocation that is not SR equivalent satisfies the other three axioms.*

Having characterized the SR aggregate allocation, I return to the SR allocation and show that it is characterized by Axioms 1-4 and an additional simple property. For any preallocation x and every agent a, I define $d_a(x)$ as the **demand of agent a associated with** x, as defined in the SR algorithm. That is, if $x_a < 1$, then $d_a(x) = 1$; if $x_a = 1$, then $d_a(x) = \max_{c \in C}\{x_{a,c}\}$; and if $x_a > 1$, then $d_a(x)$ is such that $\sum_{c \in C} \min\{d_a(x), x_{a,c}\} = 1$. I denote by $d(x) = (d_a(x))_{a \in A}$ the vector containing all of the agents' demand associated with x. Note that for an allocation ξ, a's demand simplifies to

$$d_a(\xi) = \begin{cases} 1 & \text{if } \xi_a < 1 \\ \max_{c \in C}\{\xi_{a,c}\} & \text{if } \xi_a = 1. \end{cases}$$

Theorem 4. *For every allocation $\xi^* \neq \xi^{SR}$ that satisfies Axioms 1-4, $d(\xi^*) < d(\xi^{SR})$.*

Theorem 4 allows full characterization of the SR allocation: it is the allocation satisfying Axioms 1-4 with the largest demand associated with it. The intuition is as follows. The SR algorithm initially sets every agent's demand to one, the largest possible level. In each round, it calculates a preallocation that satisfies Axioms 1-4 and reduces the demands to eliminate the excess supply. Thus, the SR algorithm finds in each round an upper bound for the demand vector in any allocation that satisfies Axioms 1-4. The algorithm continues until the demand vector has been reduced just enough to find an allocation satisfying Axioms 1-4; hence, it identifies the largest demand vector for which such an allocation exists.

4.4 Pure Aggregate Allocations

As units are indivisible, ultimately, each agent must be allocated either zero units or one unit in aggregate. Formally, an aggregate allocation ρ is **pure** if $\rho_a \in \{0, 1\}$ for every agent a. The SR aggregate allocation is not necessarily pure; in fact, in Example 1, a_3 and a_5 are allocated 5/6 and 1/6, respectively, in aggregate. In such a situation, one needs to decide which of the agents allocated an amount of capacity strictly between zero and one in aggregate end up receiving a unit. Formally, a pure aggregate allocation ρ is **consistent with the SR aggregate allocation** $\rho^{SR} = (\xi_a^{SR})_{a\in A}$ if $\rho_a = \xi_a^{SR}$ for every agent a such that $\xi_a^{SR} \in \{0, 1\}$. If ρ^{SR} is not pure, then one needs to choose among those pure allocations that are consistent with ρ^{SR}. I remain agnostic as to how that choice should be made as this is application dependent; however, I outline two possible solutions.

A common approach in the literature (see, e.g., Budish, Che, Kojima, and Milgrom, 2013; Kesten and Ünver, 2015) is, for each agent a, to treat ξ_a^{SR} as a probability. The Birkhoff-von Neuman theorem (Birkhoff, 1946; Von Neumann, 1953) guarantees the existence of a lottery over pure aggregate allocations (all of which are consistent with ρ^{SR}) such that each agent a is allocated a unit with probability ξ_a^{SR}. In Example 1, a lottery would allocate the last unit to a_3 with probability 5/6 and to a_5 with probability 1/6.

An alternative approach consists of allocating the remaining units to whomever has the largest aggregate allocation. The agents whose aggregate SR allocation lies strictly between zero and one are ordered from highest to lowest aggregate allocation (ties could be broken randomly or through a master ranking of agents). A pure aggregate allocation consistent with ρ^{SR} is then constructed by allocating one unit to every agent a such that $\rho_a^{SR} = 1$ and allocating remaining units according to the constructed order. In Example 1, the last unit would be allocated to a_3.

Whatever rule is used to choose among the pure aggregate allocations that are consistent with ρ^{SR}, the next result shows that the impact of that choice is limited.

Proposition 5. *At the SR aggregate allocation, at most $|C|$ agents are allocated an amount of capacity strictly between zero and one.*

The intuition for Proposition 5 is that, as ξ^{SR} respects priorities, each category allocates capacity to at most one agent who is not allocated one unit in aggregate; hence, the number of agents who are allocated some capacity but less than one unit in aggregate cannot exceed the number of categories. In practice, the number of categories is typically much smaller than the number of agents; therefore, the vast majority of agents are allocated either zero units or one unit at ρ^{SR}.

4.5 Horizontal Envelope and Smart Reserves

I conclude this section by discussing the relationship between the SR allocation and the most closely related solution concept in the literature. Sönmez and Yenmez (2020) consider a special case of my model in which there is a *baseline priority order* over the agents, one category ranks agents solely based on that order, and each of the remaining categories prioritizes a set of *beneficiaries* and breaks ties with the baseline priority order. Their *horizontal envelope* algorithm yields the unique allocation that maximizes the number of units allocated to beneficiaries while respecting the baseline priority order. Pathak, Sönmez, Ünver, and Yenmez (2020) generalize that idea and develop the concept of *smart reserves*, which allow for an arbitrary number of units to be allocated by the baseline priority order before the other categories are considered.

There are at least three important differences between that approach and the one developed in this paper. First, these two approaches pursue a different objective. The horizontal envelope and smart reserve algorithms aim to maximize the number of beneficiaries who are allocated a unit; otherwise, the allocation follows a baseline priority order. The aim of the SR allocation is to respect the priority of each category and treat all categories identically. Second, the horizontal envelope and smart reserve algorithms are only defined for a special case of the model considered in this paper, which may not fit every application. For example, one category considered by Pathak, Sönmez, Ünver, and Yenmez (2020) would prioritize patients with a disability and rank those agents randomly rather than following the baseline priority order. The results presented in this paper are more general, as they do not rely on any assumption about priorities. Third, the outcome of the smart reserves algorithm depends on how many unreserved units are allocated before or after the reserved units. The horizontal envelope algorithm constitutes a special case in which all unreserved units are allocated after the reserved units; with two categories, this is equivalent to processing the categories sequentially, starting with the one that has beneficiaries. In contrast, the SR results from processesing all categories simultaneously, and only depends on the category quotas and priorities.

The two approaches are complementary, as they make different solutions available for market designers to choose from depending on the specificities of the application at hand. Moreover, an interesting question for future research is whether and how the two approaches can be combined in rationing problems with hard reserves.[11] If efficiency—in the sense of allocating as many units as possible—is a desideratum, one might consider relaxing the category neutrality condition so that agents can be allocated more capacity from categories that have not assigned their entire quota. For example, if fewer than $20,000$ workers with an advanced degree apply for an H-1B visa, one might allocate to all of them one unit from the advanced degree category even if they also qualify for the open category, thus leaving all remaining $65,000$ visas available for applicants without an advanced degree.

[11] A similar approach may be useful for rationing problems with soft reserves in which a desideratum is that each category allocates as much capacity as possible to its beneficiaries.

Algorithm 2: SIMULTANEOUS RESERVE WITH LINEAR PROGRAMMING (SRLP)

Initialization Set every agent's demand to one: $d_a^0 = 1$ for every agent a.

Round $i \geq 1$:

Linear Programming If either $i = 1$ or $i > 1$ and there exists an agent-category pair (a, c) such that either $x_{a,c}^{i-1} > x_{a,c}^{i-2} = 0$, or $x_{a,c}^{i-1} \geq d_a^{i-1}$ and $x_{a,c}^{i-2} < d_a^{i-2}$, set $\delta_a^i = d_a^{i-1}$ for every agent a. Otherwise, set $\delta_a^i = \delta_a^{LP}(x^{i-1}, d^{i-1})$ (calculated by Algorithm 3) for every agent a.

Capacity Allocation For every agent a and every category c, if a is eligible for c, then set $x_{a,c}^i = \min\{\delta^i, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} \delta_{a'}^i, 0\}\}$ and otherwise set $x_{a,c}^i = 0$.

Termination If $x_a^i \leq 1$ for every agent a, end and output x^i.

Demand Adjustment For every agent a such that $x_a^i < 1$, set $d_a^i = 1$. For every agent a such that $x_a^i = 1$, set $d_a^i = \max_{c \in C}\{x_{a,c}^i\}$. For every agent a such that $x_a^i > 1$, set d_a^i such that $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = 1$.

5 Simultaneous Reserve with Linear Programming

A practical shortcoming of the SR algorithm is that it may run for infinitely many rounds. In this section, I propose an alternative algorithm that is outcome equivalent but runs in polynomial time. The simultaneous reserve algorithm with linear programming (SRLP) algorithm is formally defined in Algorithm 2. Its structure is similar to that of the SR algorithm, but in some rounds, it solves a linear programming problem (defined in Algorithm 3) to update the demand of some agents.

Theorem 5. *The SRLP algorithm produces ξ^{SR} after fewer than $4|A||C|$ rounds.*

In the remainder of this section, I describe the SRLP algorithm and illustrate how it works using Example 2. Along the way, I provide some intuition for Theorem 5, whose formal proof can be found in Appendix D. First, a note about the running time of the SRLP algorithm is in order. By Theorem 5, the number of rounds required is polynomial in $|A||C|$. For a round in which linear programming is not used, $|A|(|C|+2)$ operations are required (each category allocates some amount of capacity to each agent and then the total capacity allocated and demand of each agent is calculated). As linear programming can be solved in polynomial time (Khachiyan, 1979), it follows that the SR algorithm works in polynomial time.

At the high level, the idea behind the SRLP algorithm is to identify when the SR algorithm is at risk of slowing down and speed up the process by using linear programming. For this

Algorithm 3: LINEAR PROGRAMMING (LP)

.

Input Take as input a preallocation x and a demand vector d.

Linear Program Construction For every agent a, construct the set of categories for which a is qualified and marginal: $C_Q(a) = \{c \in C : x_{a,c} \geq d_a\}$ and $C_M(a) = \{c \in C : x_{a,c} \in (0, d_a)\}$. Let $\widetilde{A} = \{a \in A : \min\{|C_Q(a)|, |C_M(a)|\} \geq 1\}$ be the set of agents who are qualified and marginal for at least one category.

Let $\widetilde{C} = \{c \in C : c \in C_M(a) \text{ for some } a \in \widetilde{A}\}$ be the set of categories that have a marginal agent who is qualified for another category. For every such category $c \in \widetilde{C}$, let $a(c)$ be that category's marginal agent, construct the set $A_Q(c) = \{a \in A, x_{a,c} \geq d_a\}$ of agents who are qualified for c and the subset $\widetilde{A}_Q(c) = \{a \in \widetilde{A}, x_{a,c} \geq d_a\}$ of them who are marginal for another category, and adapt the quota to dismiss the agents who are not marginal for another category: $\widetilde{q}_c = q_c - \sum_{a \in A_Q(c) \setminus \widetilde{A}_Q(c)} d_a$.

Linear Program Solving Solve the following linear programming problem:

$$\begin{aligned}
&\max_{(\xi_{a(c),c})_{c\in\widetilde{C}}} \sum_{c \in \widetilde{C}} \xi_{a(c),c} \\
\text{subject to} \quad & \xi_{a(c),c} \leq \frac{1 - \sum_{c' \in C_M(a(c)) \setminus \{c\}} \xi_{a(c),c'}}{|C_Q(a(c))| + 1} \\
\text{and} \quad & \xi_{a(c),c} \leq \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi_{a,c'}}{|C_Q(a)|} \quad \text{for every } c \in \widetilde{C}.
\end{aligned} \tag{LP 1}$$

Output Let the vector $(\xi^*_{a(c),c})_{c \in \widetilde{C}}$ be the solution to the linear program (LP 1). For every agent a, set

$$\delta_a^{LP}(x,d) = \begin{cases} \frac{1 - \sum_{c' \in C_M(a(c)) \setminus \{c\}} \xi^*_{a(c),c'}}{|C_Q(a(c))|+1} & \text{if } a \in \widetilde{A} \\ d_a & \text{if } a \in A \setminus \widetilde{A}. \end{cases}$$

purpose, in every Round i of the SRLP algorithm, I split the agent-category pairs into three groups. I say that agent a is **qualified** for category c if $x^i_{a,c} \geq d^i_a$. The term captures the idea that a has a high enough priority for c to obtain the amount of capacity she requires from it. In contrast, if $x^i_{a,c} = 0$, I say that agent a is **unqualified** for category c in Round i: a's priority for c is not high enough to obtain any capacity. In the intermediate case in which $0 < x^i_{a,c} < d^i_a$, I say that a is **marginal** for c. I refer to agent a's quality as qualified, marginal, or unqualified as a's **status** for c. Using the convention that $x^0 = \mathbf{0}$, initially every agent is unqualified for every category. Throughout the SRLP algorithm, an agent's status for a category may change to marginal or qualified. However, any such change is by construction irreversible; therefore, there can be at most $2|A||C|$ status changes throughout the entire algorithm. Moreover, once $2|A||C|$ status changes have occurred, every agent is qualified for every category, so the only possible allocation is one in which each agent is allocated $1/|C|$ from each category, and the SRLP algorithm ends. The SRLP algorithm works identically to the SR algorithm until either it finds an allocation—in which case the SRLP algorithm ends and returns that allocation—or a round occurs in which no status changes. In the latter case, the SRLP algorithm uses the linear program defined in Algorithm 3 in the following round, which ensures that either an allocation is found or at least one status changes. As a result, at least one status changes every second round so the SRLP algorithm finds an allocation within $4|A||C|$ rounds.[12]

Whenever no change of status has occurred in the previous round (i.e., there is no agent-category pair (a, c) such that either $x^{i-1}_{a,c} > x^{i-2}_{a,c}$, or $x^{i-1}_{a,c} > d^{i-1}_a$ and $x^{i-2}_{a,c} < d^{i-2}_a$), the SRLP algorithm uses Algorithm 3 to calculate a new demand vector. Algorithm 3 takes a preallocation x and the associated demand vector d (given by the SRLP algorithm) as inputs and returns a demand vector δ^{LP}. I next describe how Algorithm 3 works and illustrate it with Example 2.

What the SR algorithm does is allocate the excess supply to marginal agents in each round until either a marginal agent is allocated her demand—in which case a change of status occurs—or there is no more excess supply—in which case an allocation has been found. As Example 2 shows, this may take infinitely many rounds; however, linear programming allows it to be done in just one round. The idea is to maximize the amount of capacity allocated to marginal agents under the constraints that agents cannot be allocated more than their demand from any category and more than one unit overall and that categories may not allocate more than their quotas.

[12] As at least two status changes occur in the first round but one more round may be required once all agents are qualified for all categories, an upper bound on the number of rounds after which the SRLP algorithm ends is $4|A||C| - 2$; see the proof of Lemma D.2 in Appendix D for a formal argument.

Consider a Round i in which the SRLP algorithm uses linear programming. Then, $x = x^{i-1}$ and $d = d^{i-1}$ are the input to Algorithm 3, which calculates a demand vector $\delta^{LP}(x, d)$. The idea behind Algorithm 3 is to first calculate an allocation ξ that satisfies Axioms 1-4 and satisfies that $\xi_{a,c} = 0$ for every agent-category pair (a, c) such that $x_{a,c} = 0$; that is, an agent who was unqualified for a category in Round $i - 1$ does not receive any capacity from that category at ξ. The output of Algorithm 3 turns out to be the demand vector associated with that allocation:

$$\delta_a^{LP} = \begin{cases} 1 & \text{if } \xi_a < 1 \\ \max_{c\in C}\{\xi_{a,c}\} & \text{if } \xi_a = 1. \end{cases}$$

Every agent a who was unqualified for every category in Round $i-1$ is not allocated any capacity in aggregate: $\xi_a = 0$. Her demand thus remains one: $\delta_a^{LP} = d_a = 1$. Consider next an agent a who was qualified for some categories in Round $i-1$ but not marginal for any. Then, the category neutrality condition dictates that $\xi_{a,c} = x_{a,c} = 1/|C_Q(a)|$ for every $c \in C_Q(a)$ (where $C_Q(a)$ is the set of categories for which a is eligible, as calculated in Algorithm 3). It follows that such an agent's demand does not change either: $\delta_a^{LP} = d_a = 1/|C_Q(a)|$. Finally, consider an agent a who was marginal for some categories in Round $i-1$ but not qualified for any. That agent's demand in Round $i-1$ was then one; hence, $d_a = 1$. For every category for which a was marginal, all lower-priority agents were unqualified;[13] therefore, regardless of a's demand, those agents are not allocated any capacity from c. Then, what a is allocated does not affect any other agent, and one can simply set $\xi_{a,c} = 0$ for every $c \in C$ and $\delta_a^{LP} = d_a = 1$.

The agents for which linear programming is required are those in set $\widetilde{A}$ (defined in Algorithm 3) who are qualified for at least one category and marginal for at least one category. Those agents receive one unit in aggregate, so $\sum_{c\in C_Q(a)} \xi_{a,c} + \sum_{c\in C_M(a)} \xi_{a,c} = 1$ for every $a \in \widetilde{A}$. The challenge is to determine how the unit allocated to a is shared among categories. The category neutrality condition dictates that a receive her demand from each category for which she is qualified, so we have

$$|C_Q(a)|\delta_a^{LP} + \sum_{c\in C_M(a)} \xi_{a,c} = 1 \quad \text{for every } a \in \widetilde{A}. \tag{1}$$

Moreover, the category neutrality condition also dictates that every category $c \in C_M(a)$ allocates to a either all of its remaining capacity or a's demand:

$$\xi_{a,c} = \min\{\delta_a^{LP}, q_c - \sum_{a'\in\hat{A}_{a,c}} \xi_{a',c}\} \quad \text{for every } c \in C_M(a).$$

[13] See Claim 6 in Appendix D for a formal statement

By construction, $\hat{A}_{a,c} = A_Q(c)$;[14] therefore, as $\xi_{a',c} = \delta_a^{LP}$ for every $a' \in A_Q(c)$ and $\delta_a^{LP} = d_a$ for every $a' \in A_Q(c) \setminus \tilde{A}_Q(c)$, we have

$$\sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} = \sum_{a' \in A_Q(c)} \xi_{a',c} = \sum_{a' \in \widetilde{A}_Q(c)} \xi_{a',c} + \sum_{a' \in A_Q(c) \setminus \widetilde{A}_Q(c)} \xi_{a',c} = \sum_{a' \in \widetilde{A}_Q(c)} \delta_{a'}^{LP} + \sum_{a' \in A_Q(c) \setminus \widetilde{A}_Q(c)} d_{a'}.$$

Then, by definition, we have

$$q_c - \sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} = q_c - \sum_{a' \in A_Q(c) \setminus \widetilde{A}_Q(c)} d_{a'} - \sum_{a' \in \widetilde{A}_Q(c)} \delta_{a'}^{LP} = \widetilde{q}_c - \sum_{a' \in \widetilde{A}_Q(c)} \delta_{a'}^{LP}.$$

It follows that

$$\xi_{a,c} = \min \left\{ d_a^{LP}, \widetilde{q}_c - \sum_{a' \in \widetilde{A}_Q(c)} \delta_{a'}^{LP} \right\} \quad \text{for every } c \in C_M(a). \tag{2}$$

The goal is then to find an allocation ξ that satisfies (1) and (2) for every agent $a \in \widetilde{A}$ and every category $c \in C_M(a)$. The agents' demands can be substituted out by using (1), which yields

$$\xi_{a,c} = \min \left\{ \frac{1 - \sum_{c \in C_M(a)} \xi_{a,c}}{|C_Q(a)|}, \widetilde{q}_c - \sum_{a' \in \widetilde{A}_Q(c)} \delta_{a'}^{LP} \right\} \quad \text{for every } a \in \widetilde{A} \text{ and every } c \in C_M(a).$$

Recalling (see Algorithm 3) that $\widetilde{C}$ is the set of categories that have a marginal agent who is qualified for another category and that for every $c \in \widetilde{C}$, c's marginal agent is denoted $a(c)$, it follows that

$$\xi_{a(c),c} = \min \left\{ \frac{1 - \sum_{c' \in C_M(a(c))} \xi_{a(c),c'}}{|C_Q(a(c))|}, \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi_{a,c'}}{|C_Q(a)|} \right\} \quad \text{for every } c \in \widetilde{C}.$$

Finally, by definition, $c \in C_M(a(c))$, so the previous equation can be simplified to

$$\xi_{a(c),c} = \min \left\{ \frac{1 - \sum_{c' \in C_M(a(c)) \setminus \{c\}} \xi_{a(c),c'}}{|C_Q(a(c))| + 1}, \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi_{a,c'}}{|C_Q(a)|} \right\} \quad \text{for every } c \in \widetilde{C}.$$

There are $|\widetilde{C}|$ variables and $|\widetilde{C}|$ equations, one for each category $c \in \widetilde{C}$; however, those

[14] See again Claim 6 in Appendix D for a formal statement

equations are not linear. What the linear program (LP 1) does is, for each $c \in \widetilde{C}$, turn that category's equation into two constraints and maximize $\xi_{a(c),c}$ subject to those constraints.

I illustrate how this works using Example 2. Round 4 is the first round in which no status changes occur (see Table 4), so linear programming is used in Round 5. The inputs are

$$x = x^4 = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \!\!\begin{array}{c} \begin{array}{ccc} \;\;c_1 & \quad c_2 & \quad c_3 \end{array} \\ \begin{pmatrix} 5/12 & 3/12 & 5/12 \\ 1/3 & 1/3 & 1/3 \\ 3/12 & 5/12 & 5/12 \\ 0 & 0 & 5/6 \end{pmatrix} \end{array} \quad \text{and} \quad d = d^4 = \begin{array}{c} \begin{array}{cccc} \;\;a_1 & \quad a_2 & \quad a_3 & \quad a_4 \end{array} \\ \begin{pmatrix} 5/12 & 1/3 & 5/12 & 1 \end{pmatrix} \end{array}.$$

In the linear program construction stage, we have

$$\begin{array}{llll} C_Q(a_1) = \{c_1, c_3\} & C_Q(a_2) = \{c_1, c_2, c_3\} & C_Q(a_3) = \{c_2, c_3\} & C_Q(a_4) = \emptyset \\ C_M(a_1) = \{c_2\} & C_M(a_2) = \emptyset & C_M(a_3) = \{c_1\} & C_M(a_4) = \{c_3\}. \end{array}$$

Therefore, $\widetilde{A} = \{a_1, a_2\}$ and $\widetilde{C} = \{c_1, c_2\}$, so we have $a(c_1) = a_3$ and $a(c_2) = a_1$. It follows that $A_Q(c_1) = \{a_1, a_2\}$, $\widetilde{A}_Q(c_1) = \{a_1\}$, $A_Q(c_2) = \{a_2, a_3\}$, and $\widetilde{A}_Q(c_2) = \{a_3\}$; hence, $\widetilde{q}_{c_1} = \widetilde{q}_{c_2} = 1 - d_{a_2} = 2/3$.

In the linear program solving stage, the linear program that must be solved is

$$\begin{aligned} \max_{(\xi_{a_3,c_1}, \xi_{a_1,c_2})} \quad & \xi_{a_3,c_1} + \xi_{a_1,c_2} \\ \text{subject to} \quad & \xi_{a_3,c_1} \leq 1/3, \\ & \xi_{a_1,c_2} \leq 1/3, \\ & \xi_{a_3,c_1} \leq 2/3 - (1 - \xi_{a_1,c_2})/2, \\ & \xi_{a_1,c_2} \leq 2/3 - (1 - \xi_{a_3,c_1})/2. \end{aligned}$$

Setting $\xi_{a_3,c_1} = \xi_{a_1,c_2} = 1/3$ makes all four constraints hold with an equality; hence, the vector $(\xi^*_{a_3,c_1}, \xi^*_{a_1,c_2}) = (1/3, 1/3)$ is the unique solution to the linear program. Then, the output of Algorithm 3 is the demand vector

$$\delta^{LP} = \begin{array}{c} \begin{array}{cccc} \;a_1 & \;\; a_2 & \;\; a_3 & \;\; a_4 \end{array} \\ \begin{pmatrix} 1/3 & 1/3 & 1/3 & 1 \end{pmatrix} \end{array}.$$

Returning to Round 5 of Algorithm 2, the linear programming stage has produced the demand vector $\delta^5 = \delta^{LP}$. The capacity allocation stage produces the SR allocation, $x^5 = \xi^{SR}$, and so the SRLP algorithm ends in Round 5 and outputs the SR allocation.

6 Conclusion

In this paper, I have proposed a new solution to rationing problems with reserves. In contrast to existing solutions, in which reserve categories are processed sequentially, I propose an algorithm that processes them simultaneously. A key advantage of this approach is transparency: the SR allocation depends solely on the category quotas and priorities. The key idea is to allow an agent who is allocated one unit in aggregate to receive parts of that unit from different categories. In fact, I show that the SR allocation is *category neutral*: if an agent qualifies for multiple categories, she receives the same amount of capacity from each of them. This is in stark contrast to sequential processing, in which an agent who qualifies for multiple categories receives one unit from whichever is processed first. In addition to being category neutral, the SR allocation satisfies three standard conditions: compliance with eligibility criteria, nonwastefulness, and respect for priorities. I show that any other allocation satisfying those four properties allocates in aggregate the same amount of capacity to every agent. Finally, I show that linear programming allows the SR allocation to be computed in polynomial time.

This paper opens up various opportunities for future research; I conclude by briefly describing four of them. First, it might be possible to tweak the SR algorithm to handle ties in the priority profile.[15] Priority ties are often present in real-world applications, and such a solution would avoid having to break them through a lottery. Second, it would be valuable to explore how the SR algorithm can be combined with the deferred acceptance mechanism (or any other mechanism) so it can be used in matching markets. Third, it may be possible to generalize the approach to sharing rules beyond category neutrality. If an agent qualifies for two categories, with sequential processing, the category processed first allocates one unit to that agent, while with the category neutrality condition, each category allocates half a unit to the agent. One might consider any sharing rule in between, which would convexify of the set of solutions provided by sequential allocation. Last, as I discuss in Section 4.5, it may be possible to relax the category neutrality condition in rationing problems with hard reserves to allocate more units overall. Ultimately, I hope that the ideas presented in this paper provide a new perspective on rationing problems with reserves and pave the way toward developing and applying new solutions in a wide range of contexts.

[15]See Kesten and Ünver (2015) for a similar approach without reserves.

References

Abdulkadiroğlu, A. (2005): "College admissions with affirmative action," *International Journal of Game Theory*, 33(4), 535–549.

Abdulkadiroğlu, A., and T. Sönmez (2003): "School Choice: A Mechanism Design Approach," *American Economic Review*, 93(3), 729–747.

Aygün, O., and I. Bó (2020): "College admission with multidimensional privileges: The Brazilian affirmative action case," *American Economic Journal: Microeconomics*, forthcoming.

Aygün, O., and B. Turhan (2020a): "Designing Direct Matching Mechanisms for India with Comprehensive Affirmative Action," *SSRN Working Paper, May 2020.*

——— (2020b): "Dynamic Reserves in Matching Markets," *Journal of Economic Theory*, 188, 105069.

Birkhoff, G. (1946): "Three observations on linear algebra," *Univ. Nac. Tacuman, Rev. Ser. A*, 5, 147–151.

Bogomolnaia, A., and H. Moulin (2001): "A new solution to the random assignment problem," *Journal of Economic Theory*, 100(2), 295–328.

Budish, E., Y.-K. Che, F. Kojima, and P. Milgrom (2013): "Designing random allocation mechanisms: Theory and applications," *American Economic Review*, 103(2), 585–623.

Dur, U., S. D. Kominers, P. A. Pathak, and T. Sönmez (2018): "Reserve design: Unintended consequences and the demise of Boston's walk zones," *Journal of Political Economy*, 126(6), 2457–2479.

Dur, U., P. A. Pathak, and T. Sönmez (2020): "Explicit vs. statistical targeting in affirmative action: Theory and evidence from Chicago's exam schools," *Journal of Economic Theory*, 187, 104996.

Echenique, F., and M. B. Yenmez (2015): "How to control controlled school choice," *American Economic Review*, 105(8), 2679–2694.

Ehlers, L., I. E. Hafalir, M. B. Yenmez, and M. A. Yildirim (2014): "School choice with controlled choice constraints: Hard bounds versus soft bounds," *Journal of Economic Theory*, 153, 648–683.

Gonczarowski, Y. A., L. Kovalio, N. Nisan, and A. Romm (2019): "Matching for the Israeli "Mechinot" Gap-Year Programs: Handling Rich Diversity Requirements," arXiv preprint arXiv:1905.00364.

Hafalir, I. E., M. B. Yenmez, and M. A. Yildirim (2013): "Effective affirmative action in school choice," *Theoretical Economics*, 8(2), 325–363.

Hylland, A., and R. Zeckhauser (1979): "The efficient allocation of individuals to positions," *Journal of Political Economy*, 87(2), 293–314.

Kesten, O., and M. U. Ünver (2015): "A theory of school-choice lotteries," *Theoretical Economics*, 10(2), 543–595.

Khachiyan, L. G. (1979): "A polynomial algorithm in linear programming," in *Doklady Akademii Nauk*, vol. 244, pp. 1093–1096. Russian Academy of Sciences.

Kojima, F. (2012): "School choice: Impossibilities for affirmative action," *Games and Economic Behavior*, 75(2), 685–693.

Kominers, S. D., and T. Sönmez (2016): "Matching with slot-specific priorities: Theory," *Theoretical Economics*, 11(2), 683–710.

Pathak, P. A. (2011): "The mechanism design approach to student assignment," *Annual Reiew of Economics*, 3(1), 513–536.

Pathak, P. A., A. Rees-Jones, and T. Sönmez (2020a): "Immigration Lottery Design: Engineered and Coincidental Consequences of H-1B Reforms," NBER Working Paper 26767.

——— (2020b): "Reversing Reserves," NBER Working Paper 26963.

Pathak, P. A., T. Sönmez, M. U. Ünver, and M. B. Yenmez (2020): "Fair Allocation of Vaccines, Ventilators and Antiviral Treatments: Leaving No Ethical Value Behind in Health Care Rationing," Working paper, July 2020.

Persad, G., M. E. Peek, and E. J. Emanuel (2020): "Fairly Prioritizing Groups for Access to COVID-19 Vaccines," *JAMA*, https://doi.org/10.1001/jama.2020.18513.

Sönmez, T., and M. B. Yenmez (2019a): "Affirmative action in India via vertical and horizontal reservations," Working paper, October 2019.

Sönmez, T., and M. B. Yenmez (2019b): "Constitutional implementation of vertical and horizontal reservations in India: A unified mechanism for civil service allocation and college admissions," Working paper, April 2019.

Sönmez, T., and M. B. Yenmez (2020): "Affirmative Action with Overlapping Reserves," Boston College Working Paper 990.

Von Neumann, J. (1953): "A certain zero-sum two-person game equivalent to the optimal assignment problem," *Contributions to the Theory of Games*, 2, 5–12.

Westkamp, A. (2013): "An analysis of the German university admissions system," *Economic Theory*, 53(3), 561–589.

Yilmaz, O. (2020): "Rationing Medical Units in a Pandemic: Fair Allocation through Reserves," Working paper.

Appendix: Proofs

The appendix is organized as follows. I first prove a series of properties of the SR algorithm in Appendix A, which I use in Appendix B to prove the results from Section 4. Appendix C contains properties of the SRLP algorithm that mirror those of the SR algorithm. In Appendix D, I use these properties to prove Theorem 5.

A Properties of the SR Algorithm

I start with a series of regularity conditions.

Lemma A.1. *For every agent a and every Round i such that $x^i_a > 1$, there exists a unique d^i_a such that $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1$. Moreover, $d^i_a \in (0, \max_{c\in C}\{x^i_{a,c}\})$.*

Proof. If $d^i_a \leq 0$, then $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} \leq 0 < 1$. If $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$, then $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = \sum_{c\in C} x^i_{a,c} = x^i_a > 1$. The expression $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\}$ is continuous at every d^i_a, strictly increasing in d^i_a at every $d^i_a \leq \max_{c\in C}\{x^i_{a,c}\}$, and constant in d^i_a at every $d^i_a \geq \max_{c\in C}\{x^i_{a,c}\}$. Therefore, there exists a unique value of d^i_a such that $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1$ and that value is an element of $(0, \max_{c\in C}\{x^i_{a,c}\})$. □

Lemma A.2. *For every Round $i \geq 1$, x^i is a preallocation and, for every agent a, $d^i_a \in [1/|C|, 1]$.*

Proof. ($d^i_a \in [0,1]$) Toward a contradiction, suppose that $d^i_a \notin [0,1]$. By definition, $x^i_a \geq 1$ as otherwise $d^i_a = 1$. If $x^i_a = 1$, then $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$. If $x^i_a > 1$, then $d^i_a \in (0, \max_{c\in C}\{x^i_{a,c}\})$ by Lemma A.1. In both cases, it follows that there exists a category $c \in C$ such that $x^i_{a,c} \notin [0,1]$. Then, a is eligible for c, as otherwise $x^i_{a,c} = 0$; therefore, $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$, which implies that $d^{i-1}_a \notin [0,1]$. By induction, it follows that $d^0_a \notin [0,1]$, a contradiction since $d^0_a = 1$.

(x^i is a preallocation) What needs to be shown is that $x^i_{a,c} \in [0,1]$ for all $a \in A$ and all $c \in C$ and that $\sum_{a\in A} x^i_{a,c} \leq q_c$ for all $c \in C$. Consider any agent a and any category c. If $x^i_{a,c} \notin [0,1]$, it was established in the first part of this proof that $d^{i-1}_a \notin [0,1]$, a contradiction.

It remains to show that $\sum_{a\in A} x^i_{a,c} \leq q_c$ for all $c \in C$. Consider any category c and suppose toward a contradiction that $\sum_{a\in A} x^i_{a,c} > q_c$. Then, there exists an agent a such that $x_{a,c} > 0$ and $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} > q_c$. By definition, $x^i_{a,c} \leq d^{i-1}_{a'}$ for all $a' \in \hat{A}_{a,c}$ so $x^i_{a',c} + \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'} > q_c$ or, equivalently, $x^i_{a',c} > q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}$. Again by definition, $x^i_{a,c} \leq \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$; therefore it must be that $x_{a',c} = 0$, a contradiction.

($d^i_a \geq 1/|C|$) If $x^i_a < 1$, the statement is trivially satisfied as $d^i_a = 1$ by definition. If $x^i_a = 1$, then $d^i_a = \max\{c \in C\}\{x^i_{a,c}\}$. As $\sum_{c\in C} x^i_{a,c} = 1$ and $x^i_{a,c} \in [0,1]$, we have

$\max_{c\in C}\{x^i_{a,c}\} \geq 1/|C|$ so $d^i_a \geq 1/|C|$. If $x^i_a > 1$, then $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1$; therefore, we have $\sum_{c\in C} d^i_a \geq 1$ so $|C|d^i_a \geq 1$, which is equivalent to $d^i_a \geq 1/|C|$. □

For every Round $i \geq 1$, let ξ^i to be the Round i allocation (as opposed to the Round i preallocation x^i) defined as follows. For every agent a and every category c, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\}$.

Lemma A.3. *For every Round $i \geq 1$, ξ^i is an allocation and, for every agent a, $\xi^i_a = \min\{x^i_a, 1\}$.*

Proof. ($\xi^i_a = \min\{x^i_a, 1\}$) *Case 1*: $x^i_a \leq 1$. If $x^i_a < 1$, then by definition $d^i_a = 1$ and $x^i_{a,c} < 1$ for all $c \in C$. If $x^i_a = 1$, then by definition $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$. It follows that $x^i_{a,c} \leq d^i_a$ for all $c \in C$; therefore,

$$\xi^i_a = \sum_{c\in C} \xi^i_{a,c} = \sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = \sum_{c\in C} x^i_{a,c} = x^i_a = \min\{x^i_a, 1\}.$$

Case 2: $x^i_a > 1$. By definition, d^i_a satisfies $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1$. It follows that

$$\xi^i_a = \sum_{c\in C} \xi^i_{a,c} = \sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1 = \min\{x^i_a, 1\}.$$

(ξ^i is an allocation) By definition, for every agent a and every category c, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\}$. As $d^i_a, x^i_{a,c} \in [0, 1]$, it follows that $\xi^i_{a,c} \in [0, 1]$. Moreover, $\sum_{a\in A} \xi^i_{a,c} \leq \sum_{a\in A} x^i_{a,c} \leq q_c$. Therefore, ξ^i is a preallocation and it remains to show that $\xi_a \leq 1$ for all $a \in A$, which follows from the previously established result that $\xi^i_a = \min\{x^i_a, 1\}$ for all $a \in A$. □

The next lemma states that the total amount that each agent is allocated weakly increases throughout the algorithm while each agent's demand decreases throughout the algorithm. For notational convenience, let $\xi^0 = \mathbf{0}_{|A|\times|C|}$.

Lemma A.4. *For every Round i of the SR algorithm and every agent a, $\xi^i_a \geq \xi^{i-1}_a$ and $d^i_a \leq d^{i-1}_a$.*

Proof. By definition, $\xi^0_a = 0$ and $d^0_a = 1$ and, by Lemmas A.2 and A.3, $\xi^i_a, d^i_a \in [0, 1]$; therefore the statement holds for Round 1: $\xi^1_a \geq \xi^0_a$ and $d^1_a \leq d^0_a$.

The remainder of the proof is by induction. For some $i \geq 2$, suppose that $\xi^{i-1}_a \geq \xi^{i-2}_a$ and $d^{i-1}_a \leq d^{i-2}_a$ for all $a \in A$ (induction hypothesis). I show that $\xi^i_a \geq \xi^{i-1}_a$ and $d^i_a \leq d^{i-1}_a$.

($\xi^i_a \geq \xi^{i-1}_a$) Consider any category c. If a is not eligible for c, then by definition $x^{i-1}_{a,c} = x^i_{a,c} = 0$ and $\xi^{i-1}_{a,c} = \min\{d^{i-1}_a, x^{i-1}_{a,c}\} = 0$; hence $x^i_{a,c} = \xi^{i-1}_{a,c} = 0$. If a is eligible for c, then by definition

$$x^{i-1}_{a,c} = \min\{d^{i-2}_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-2}_{a'}, 0\}\};$$

therefore,

$$\xi_{a,c}^{i-1} = \min\{d_a^{i-1}, d_a^{i-2}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-2}, 0\}\}.$$

By the induction hypothesis, $d_a^{i-1} \leq d_a^{i-2}$; hence

$$\xi_{a,c}^{i-1} = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-2}, 0\}\}. \tag{3}$$

By definition,

$$x_{a,c}^{i} = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}, 0\}\}. \tag{4}$$

The induction hypothesis implies that $d_{a'}^{i-1} \leq d_{a'}^{i-2}$ for all $a' \in \hat{A}_{a,c}$; therefore the right-hand side of (4) is weakly larger than the right-hand side of (3) and $x_{a,c}^i \geq \xi_{a,c}^{i-1}$.

The previous argument has established that $x_{a,c}^i \geq \xi_{a,c}^{i-1}$ for all $c \in C$; hence $x_a^i \geq \xi_a^{i-1}$. Combining Lemma A.3 with that result and the fact that $\xi_a^{i-1} \leq 1$ yields

$$\xi_a^i = \min\{x_a^i, 1\} \geq \min\{\xi_a^{i-1}, 1\} = \xi_a^{i-1},$$

which implies that $\xi_a^i \geq \xi_a^{i-1}$.

($d_a^i \leq d_a^{i-1}$) *Case 1*: $x_a^i < 1$. Lemma A.3 and the previously established result that $\xi_a^i \geq \xi_a^{i-1}$ imply that

$$\min\{x_a^i, 1\} = \xi_a^i \geq \xi_a^{i-1} = \min\{x_a^{i-1}, 1\}.$$

It follows that $\min\{x_a^i, 1\} \geq \min\{x_a^{i-1}, 1\}$, which combined with the case assumption that $x_a^i < 1$ implies that $x_a^{i-1} < 1$. By definition, it can therefore be concluded that $d_a^i = d_a^{i-1} = 1$.

Case 2: $x_a^i \geq 1$. If $x_a^i = 1$, then by definition $d_a^i = \max_{c \in C}\{x_{a,c}^i\}$. If $x_a^i > 1$, then by definition $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = 1$. Supposing that $d_a^i > \max_{c \in C}\{x_{a,c}^i\}$ yields

$$\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = \sum_{c \in C} x_{a,c}^i = x_a^i > 1,$$

a contradiction. Therefore, the case assumption that $x_a^i \geq 1$ implies that $d_a^i \leq \max_{c \in C}\{x_{a,c}^i\}$.

By definition, for every $c \in C$, $x_{a,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}, 0\}\} \leq d_a^{i-1}$; therefore $\max_{c \in C}\{x_{a,c}^i\} \leq d_a^{i-1}$, which means that $d_a^i \leq d_a^{i-1}$. $\square$

Lemma A.5. *For every Round i and every agent a,*

$$d_a^i = \begin{cases} 1 & \text{if } \xi_a^i < 1 \\ \max_{c \in C}\{\xi_{a,c}^i\} & \text{if } \xi_a^i = 1. \end{cases}$$

Proof. *Case 1*: $x_a^i < 1$. In that case, by Lemma A.3, $\xi_a^i = x_a^i < 1$ and, by definition, $d_a^i = 1$.

Case 2: $x_a^i = 1$. In that case, by Lemma A.3, $\xi_a^i = x_a^i = 1$. By definition, $d_a^i = \max_{c \in C}\{x_{a,c}\}$ and, for every $c \in C$, $\xi_{a,c}^i = \min\{d_a^i, x_{a,c}^i\}$. Combining those two results implies that $\xi_{a,c}^i = x_{a,c}^i$ for all $c \in C$, and therefore $d_a^i = \max\{\xi_{a,c}\}$.

Case 3: $x_a^i > 1$. In that case, by Lemma A.3, $\xi_a^i = 1$ so it remains to show that $d_a^i = \max_{c \in C}\{\xi_{a,c}^i\}$. If $d_a^i < \max_{c \in C}\{\xi_{a,c}^i\}$, then there exists $c \in C$ such that $d_a^i < \xi_{a,c}^i$. However, by definition, $\xi_{a,c}^i = \min\{d_a^i, x_{a,c}^i\} \leq d_a^i$, a contradiction. If $d_a^i > \max_{c \in C}\{\xi_{a,c}^i\}$, then by definition $d_a^i > \max_{c \in C}\{\min\{d_a^i, x_{a,c}^i\}\}$; therefore $d_a^i > x_{a,c}^i$ for all $c \in C$ so $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = \sum_{c \in C} x_{a,c}^i = x_a^i > 1$. However, by definition, $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = 1$, a contradiction. □

Lemma A.6. *For every Round i and every agent a, $\xi_a^i = 1$ if and only if there exists a category c such that $\xi_{a,c}^i = d_a^i$.*

Proof. If $\xi_a^i < 1$, then $\xi_{a,c}^i < 1$ for all $c \in C$ and, by Lemma A.5, $d_a^i = 1$; therefore, $\xi_{a,c}^i < d_a^i$ for all $c \in C$. If $\xi_a^i = 1$, then $d_a^i = \max_{c \in C}\{\xi_{a,c}^i\}$ by Lemma A.5; hence there exists $c \in C$ such that $\xi_{a,c}^i = d_a^i$. □

Lemma A.7. *For every Round i, every agent a, and every category c, either $x_{a,c}^i < d_a^{i-1}$ or $\xi_{a,c}^i < d_a^i$ implies that $x_{a',c}^i = \xi_{a',c}^i = 0$ for every lower-priority agent $a' \in \check{A}_{a,c}$.*

Proof. By definition, $\xi_{a,c}^i < d_a^i$ implies that $\xi_{a,c}^i = x_{a,c}^i$; hence, as $d_a^i \leq d_a^{i-1}$ by Lemma A.4, it follows that $x_{a,c}^i < d_a^{i-1}$. Moreover, again by definition, we have $\xi_{a',c}^i = 0$ if and only if $x_{a',c}^i = 0$. Therefore, it is sufficient to show that $x_{a,c}^i < d_a^{i-1}$ implies that $x_{a',c}^i = 0$ for every lower-priority agent $a' \in \check{A}_{a,c}$.

Suppose that $x_{a,c}^i < d_a^{i-1}$ and consider an arbitrary lower-priority agent $a' \in \check{A}_{a,c}$. I show that $x_{a',c}^i = 0$. If a' is not eligible for c, the desired result holds trivially since, by definition, $x_{a',c}^i = 0$. For the remainder of the proof, I assume that a' is eligible for c, which implies that a is eligible for c as well.

As $a\pi_c a'$, the assumption that a' is eligible for c implies that a is also eligible for c. Therefore, by definition, we have

$$x_{a,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1}, 0\}\} < d_a^{i-1},$$

which implies that $q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1} < d_a^{i-1}$ or, equivalently, $q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1} - d_a^{i-1} < 0$. As $a\pi_c a'$, it follows that

$$q_c - \sum_{\tilde{a} \in \hat{A}_{a',c}} d_{\tilde{a}}^{i-1} \leq q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1} - d_a^{i-1} < 0. \tag{5}$$

Moreover, as a' is eligible for c, we have

$$x^i_{a',c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{\tilde{a}\in \hat{A}_{a',c}} d^{i-1}_{\tilde{a}}, 0\}\}. \tag{6}$$

Combining (5) and (6) yields $x^i_{a',c} = 0$. $\square$

Lemma A.8. *For every Round i, every agent a, and every category c, $\xi^i_{a,c} > 0$ implies that $x^i_{a',c} = d^{i-1}_{a'}$ and $\xi^i_{a',c} = d^i_{a'}$ for every higher-priority agent $a' \in \hat{A}_{a,c}$.*

Proof. Suppose that $\xi^i_{a,c} > 0$. By definition, it must be that $x^i_{a,c} > 0$ so a is eligible for c and

$$q_c > \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}. \tag{7}$$

Consider any higher-priority agent $a' \in \hat{A}_{a,c}$. It needs to be shown that $\xi^i_{a',c} = d^i_{a'}$. By (7), $q_c > d^{i-1}_{a'} + \sum_{\tilde{a}\in \hat{A}_{a',c}} d^{i-1}_{\tilde{a}}$, which is equivalent to

$$q_c - \sum_{\tilde{a}\in \hat{A}_{a',c}} d^{i-1}_{\tilde{a}} > d^{i-1}_{a'}. \tag{8}$$

As a is eligible for c, so is a'; hence by definition $x^i_{a',c} = \min\{d^{i-1}_{a'}, \max\{q_c - \sum_{\tilde{a}\in \hat{A}_{a',c}} d^{i-1}_{\tilde{a}}, 0\}\}$. By (8), it follows that $x^i_{a',c} = d^{i-1}_{a'}$. By Lemma A.4, $d^i_{a'} \leq d^{i-1}_{a'}$; hence $x^i_{a',c} \geq d^i_{a'}$. Using that inequality in conjunction with the definition of $\xi^i_{a',c}$ yields $\xi^i_{a',c} = \min\{d^i_{a'}, x^i_{a',c}\} = d^i_{a'}$. $\square$

Lemma A.9. *For every agent a and category c such that a is eligible for c, and for every Round i, $x^i_{a,c} < d^{i-1}_a$ implies that $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} = q_c$.*

Proof. Let $\tilde{a}$ be the highest-priority agent such that $x^i_{\tilde{a},c} < d^{i-1}_{\tilde{a}}$. That is, $x^i_{\tilde{a},c} < d^{i-1}_{\tilde{a}}$ and, for every $a \in \hat{A}_{\tilde{a},c}$, $x^i_{a',c} = d^{i-1}_{a'}$. The assumption that $x^i_{a,c} < d^{i-1}_a$ ensures that $\tilde{a}$ exists and either $\tilde{a} = a$ or $\tilde{a}\pi_c a$. Then, as a is eligible for c, so is $\tilde{a}$ and we have $x^i_{\tilde{a},c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a'\in \hat{A}_{\tilde{a},c}} d^{i-1}_{a'}, 0\}\}$. As $x^i_{\tilde{a},c} < d^{i-1}_{\tilde{a}}$ and $x^i_{a',c} = d^{i-1}_{a'}$ for all $a' \in \hat{A}_{\tilde{a},c}$, it follows that $x^i_{\tilde{a},c} = \max\{q_c - \sum_{a'\in \hat{A}_{\tilde{a},c}} x^i_{a',c}, 0\}$. As x^i is a preallocation (by Lemma A.2), it must be that $\sum_{a'\in \hat{A}_{\tilde{a},c}} x^{i-1}_{a',c} \leq q_c$; therefore we can conclude that $x^i_{\tilde{a},c} = q_c - \sum_{a'\in \hat{A}_{\tilde{a},c}} x^i_{a',c}$ or, equivalently, $x^i_{\tilde{a},c} + \sum_{a'\in \hat{A}_{\tilde{a},c}} x^i_{a',c} = q_c$.

On the one hand, as either $\tilde{a} = a$ or $\tilde{a}\pi_c a$, we have $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} \geq x^i_{\tilde{a},c} + \sum_{a'\in \hat{A}_{\tilde{a},c}} x^i_{a',c} = q_c$. On the other hand, as x^i is an allocation, we have $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} \leq q_c$. Combining the two statements yields $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} = q_c$. $\square$

Lemma A.10. *In every Round i, x^i satisfies Axioms 1-4 and ξ^i satisfies Axioms 1 and 3.*

Proof. (x^i and ξ^i comply with eligibility requirements) By definition, if an agent a is not eligible for a category c, then $x^i_{a,c} = \xi^i_{a,c} = 0$.

(x^i is nonwasteful) Consider any category c such that $\sum_{a \in A} x^i_{a,c} < q_c$ and any agent a who is eligible for c. It needs to be shown that $x^i_a \geq 1$.

Case 1: $x^i_{a,c} = d^{i-1}_a$. By the case assumption and Lemma A.4, $x^i_{a,c} = d^{i-1}_a \geq d^i_a$; hence, by definition, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\} = d^i_a$. By Lemma A.6, it follows that $\xi^i_a = 1$ so, by definition, $x^i_a \geq \xi^i_a = 1$.

Case 2: $x^i_{a,c} < d^{i-1}_a$. In that case, Lemma A.9 applies and yields $x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x^i_{a',c} = q_c$. Then, we have $\sum_{a \in A} x^i_{a,c} \geq x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x^i_{a',c} = q_c$, which contradicts the assumption that $\sum_{a \in A} x^i_{a,c} < q_c$.

(x^i and ξ^i respect priorities) Arbitrarily fix an agent a and a category c. By Lemma A.3, $x^i_a < 1$ if and only if $\xi^i_a < 1$ and, by definition, $x^i_{a,c} = 0$ if and only if $\xi^i_{a,c}$. It follows that x^i respects priorities if and only if ξ^i respects priorities; hence it is enough to show that ξ^i respects priorities. Suppose that $\xi^i_a < 1$. To establish that ξ^i respects priorities, I need to show that, for every lower-priority agent $a' \in \check{A}_{a,c}$, $\xi^i_{a',c} = 0$. By Lemma A.6, the assumption that $\xi^i_a < 1$ implies that $\xi^i_{a,c} < d^i_a$ so, by Lemma A.7, $\xi^i_{a',c} = 0$.

(x^i is category neutral) Consider any agent a and any category c such that a is eligible for c and $x^i_{a,c} < \max_{c' \in C}\{x_{a,c'}\}$. It needs to be shown that $x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x_{a',c} = q_c$. By definition, $\max_{c' \in C}\{x^i_{a,c'}\} \leq d^{i-1}_a$; hence we have $x^i_{a,c} < d^{i-1}_a$. Then, by Lemma A.9, we have $x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x^i_{a',c} = q_c$. □

Lemma A.11. *For every agent a and every category c, $z^i_{a,c} \in [0,1]$. Moreover, $|z^i| = 0$ if and only if $x^i = \xi^i$.*

Proof. ($z^i_{a,c} \in [0,1]$) By definition, $z^i_{a,c} = x^i_{a,c} - \xi^i_{a,c} = x^i_{a,c} - \min\{d^i_a, x^i_{a,c}\} = \max\{x^i_{a,c} - d^i_a, 0\}$. As $x^i_{a,c}, d^i_a \in [0,1]$ (by Lemma A.2), it follows that $z^i_{a,c} \in [0,1]$.

($|z^i| = 0$ if and only if $x^i = \xi^i$) If $x^i = \xi^i$, then, for every $a \in A$ and every $c \in C$, $x_{a,c} = \xi_{a,c}$ so $z_{a,c} = x_{a,c} - \xi_{a,c} = 0$. It follows that $|z^i| = \sum_{a \in A} \sum_{c \in C} z^i_{a,c} = 0$. If $x^i \neq \xi^i$, then there exist $a \in A$ and $c \in C$ such that $x^i_{a,c} \neq \xi^i_{a,c}$ so $z_{a,c} \neq 0$. As $z^i_{a,c} \in [0,1]$ for all $a \in A$ and all $c \in C$, it follows that $|z^i| = \sum_{a \in A} \sum_{c \in C} z^i_{a,c} > 0$. □

Lemma A.12. *Suppose that, for some Round $i \geq 1$, $|z^i| = 0$. Then, for every Round $j \geq i$, $|z^j| = 0$, $x^j = \xi^j = x^i = \xi^i$, and, for every agent a, $d^j_a = d^i_a$.*

Proof. Fix an agent a and a category c arbitrarily. The main part of the proof consists in showing that $x^{i+1}_{a,c} = x^i_{a,c}$. By definition, if a is not eligible for c, then $x^{i+1}_{a,c} = x^i_{a,c} = 0$; therefore I focus throughout on the case in which a is eligible for c.

($x^{i+1}_{a,c} \geq x^i_{a,c}$) By definition, $x^{i+1}_{a,c} = \min\{d^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}\}$. By assumption, $|z^i| = 0$ so, by Lemma A.11, $x^i_{a,c} = \xi^i_{a,c}$. As, by definition, $\xi^i_{a,c} = \min\{x^i_{a,c}, d^i_a\}$, it follows that $x^i_{a,c} \leq d^i_a$. Therefore, it remains to show that $x^i_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}$.

By definition, $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$; hence $x^i_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$. By Lemma A.4, $\sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'} \geq \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}$; therefore, we have $x^i_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}$.

($x^{i+1}_{a,c} \leq x^i_{a,c}$) *Case 1*: $a \pi_c \tilde{a}$. By the case assumption, $x^i_{a,c} = d^{i-1}_a$. By definition, $x^{i+1}_{a,c} = \min\{d^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}\}$ so $x^{i+1}_{a,c} \leq d^i_a$. By Lemma A.4, $d^i_a \leq d^{i-1}_a$; therefore, we have $x^{i+1}_{a,c} \leq d^i_a \leq d^{i-1}_a = x^i_{a,c}$.

Case 2: $a = \tilde{a}$. By the case assumption, $x^i_{a,c} < d^{i-1}_a$ and $x^i_{a',c} = d^{i-1}_{a'}$ for every $a' \in \hat{A}_{a,c}$. By definition, $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$ so, as $x^i_{a,c} < d^{i-1}_a$, we have

$$x^i_{a,c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}. \tag{9}$$

Again by definition, we have

$$x^{i+1}_{a,c} = \min\{d^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}\}. \tag{10}$$

For every $a' \in \hat{A}_{a,c}$, $x^i_{a',c} = d^{i-1}_{a'}$ by the case assumption, $x^i_{a',c} = x^{i+1}_{a',c}$ by the result shown in Case 1, $x^{i+1}_{a',c} \leq d^i_{a'}$ by definition, and $d^i_{a'} \leq d^{i-1}_{a'}$ by Lemma A.4. Therefore, it can be concluded that $d^i_{a'} = d^{i-1}_{a'}$ for all $a' \in \hat{A}_{a,c}$. Combining that result with (9) and (10) yields

$$x^{i+1}_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\} = x^i_{a,c}.$$

Case 3: $\tilde{a} \pi_c a$. By definition, as $x^i_{\tilde{a},c} < d^{i-1}_{\tilde{a}}$, $x^i_{\tilde{a},c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$. As was shown in Case 2, $d^i_{a'} = d^{i-1}_{a'}$ for all $a' \in \hat{A}_{\tilde{a},c}$; therefore we have $x^i_{\tilde{a},c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}$, which implies that $q_c - x^i_{\tilde{a},c} - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'} \leq 0$. As was shown in Case 2, $x^i_{\tilde{a},c} = x^{i+1}_{\tilde{a},c}$ and, by definition, $x^{i+1}_{\tilde{a},c} \leq d^i_{\tilde{a}}$; therefore, we have $q_c - d^i_{\tilde{a}} - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'} \leq 0$. As $\tilde{a} \pi_c a$, $\sum_{a' \in \hat{A}_{a,c}} d^i_{a'} \geq d^i_{\tilde{a}} + \sum_{a' \in \hat{A}_{\tilde{a},c}} d^i_{a'}$; hence it follows that $q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'} \leq 0$. Combining that last inequality with the definition of x^{i+1}_a yields $x^{i+1}_a = 0$. By Lemma A.2, $x^i_a \geq 0$ so it can be concluded that $x^{i+1}_a \leq x^i_a$.

As a and c were chosen arbitrarily, $x^{i+1}_{a,c} = x^i_{a,c}$ holds for every agent and every category; therefore we have $x^{i+1} = x^i$. Then, by definition, $d^{i+1}_a = d^i_a$ for every agent a, and $\xi^{i+1} = \xi^i$. The result extends to every $j > i + 1$ by induction. □

Lemma A.13. *Suppose that, for some agent a, some category c, and some Round $i \geq 1$, $x^i_{a,c} < d^i_a$. Then, for every $j \leq i$, $x^j_{a,c} \leq x^i_{a,c} < d^j_a$.*

Proof. If a is not eligible for c, then $x^j_{a,c} = 0$ for every $j \geq 1$ and the result holds as, by Lemma A.2, $d^j_a > 0$. The remainder of the proof focuses on the case in which a is eligible for c.

By definition, $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$ and, by Lemma A.4, the assumption that $x^i_{a,c} < d^i_a$ implies that $x^i_{a,c} < d^{i-1}_a$. It follows that $x^i_{a,c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$. Again by definition, $x^{i-1}_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-2}_{a'}, 0\}$ and, by Lemma A.4, $\sum_{a' \in \hat{A}_{a,c}} d^{i-2}_{a'} \geq \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}$; hence we have $x^{i-1}_{a,c} \leq x^i_{a,c} < d^{i-1}_a$. By induction, the statement holds for every $j \leq i$. □

Lemma A.14. *Suppose that, for some agent a, some category c and some Round $i \geq 1$, $x^i_{a,c} \geq d^i_a$. Then, for every $j > i$, $x^j_{a,c} = d^{j-1}_a \geq d^j_a$.*

Proof. By definition, $x^i_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$, which, combined with the assumption that $x^i_{a,c} \geq d^i_a$, implies that $\max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\} \geq d^i_a$. By Lemma A.4, it follows that $\max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\} \geq d^i_a$. By definition, we have $x^{i+1}_{a,c} = \min\{d^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\}\} = d^i_a$. Then, by Lemma A.4, it follows that $x^{i+1}_{a,c} = d^i_a \geq d^{i+1}_a$ and the statement holds for all $j > i$ by induction. □

Lemma A.15. *For every allocation ξ^* that satisfies Axioms 1-4, every Round i of the SR algorithm, every agent a, and every category c, $\xi^*_a \geq \xi^i_a$ and $\xi^*_{a,c} \leq d^i_a$.*

Proof. By definition, for every agent a and every category c, $d^0_a = 1$ and ξ^* is an allocation so $\xi^*_{a,c} \leq 1 = d^0_a$. The remainder of the proof is by induction. Arbitrarily fixing a Round $i \geq 1$, I assume that $\xi^*_{a,c} \leq d^{i-1}_a$ for every agent a and every category c (induction hypothesis) and show that $\xi^*_a \geq \xi^i_a$ and $\xi^*_{a,c} \leq d^i_a$ for every agent a and every category c.

($\xi^*_a \geq \xi^i_a$ for every $a \in A$) Arbitrarily fix an agent a. By Lemma A.3, $\xi^*_a \leq 1$ and $\xi^i_a \leq 1$ so the desired result holds trivially if $\xi^*_a = 1$, and therefore only the case in which $\xi^*_a < 1$ needs to be considered. Arbitrarily fixing a category c, I show that, in this case, $\xi^*_{a,c} \geq \xi^i_{a,c}$. That result holds trivially if $\xi^i_{a,c} = 0$; hence I assume for the remainder of the argument that $\xi^i_{a,c} > 0$.

As ξ^i complies with eligibility requirements (by Lemma A.10), the assumption that $\xi^i_{a,c} > 0$ implies that a is eligible for c; hence, by definition, we have

$$\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\} = \min\{d^i_a, d^{i-1}_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}.$$

It follows that $\xi^i_{a,c} \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$ so the assumption that $\xi^i_{a,c} > 0$ implies that

$$\xi^i_{a,c} \leq q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}. \tag{11}$$

By the induction hypothesis, $\xi^*_{a',c} \leq d^{i-1}_{a'}$ for every $a' \in \hat{A}_{a,c}$; therefore (11) implies that $\xi^i_{a,c} \leq q_c - \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a,c}$, which is equivalent to

$$\xi^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a,c} \leq q_c. \tag{12}$$

By assumption, $\xi^*_{a,c}$ is nonwasteful and $\xi^*_a < 1$; hence, as a is eligible for c, we have

$$\xi^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a,c} = q_c. \tag{13}$$

Combining (12) and (13) implies that $\xi^*_{a,c} \geq \xi^i_{a,c}$. As c was chosen arbitrarily, that inequality holds for all categories and it can be concluded that $\xi^*_a = \sum_{c \in C} \xi^*_{a,c} \geq \sum_{c \in C} \xi^i_{a,c} = \xi^i_a$. As a was chosen arbitrarily, it follows that $\xi^*_a \geq \xi^i_a$ for every $a \in A$.

($\xi^*_{a,c} \leq d^i_a$ for every $a \in A$ and every $c \in C$) Arbitrarily fix an agent a and a category c. Toward a contradiction, suppose that $\xi^*_{a,c} > d^i_a$. As $\xi^*_{a,c} \leq 1$ (by Lemma A.3), it follows that $d^i_a < 1$. Then, Lemma A.5 implies that $d^i_a = \max_{c' \in C}\{\xi^i_{a,c'}\}$ and $\xi^i_a = 1$. It follows that

$$\xi^i_{a,c} \leq \max_{c' \in C}\{\xi^i_{a,c'}\} = d^i_a < \xi^*_{a,c} \leq \max_{c' \in C}\{\xi^*_{a,c'}\}. \tag{14}$$

Moreover, as $\xi^*_a \leq 1$ (by Lemma A.3) and as it was established in the previous part of the proof that $\xi^*_a \geq \xi^i_a$, the fact that $\xi^i_a = 1$ implies that

$$\xi^i_a = \sum_{c' \in C} \xi^i_{a,c'} = \sum_{c' \in C} \xi^*_{a,c'} = \xi^*_a = 1. \tag{15}$$

As (14) implies that $\xi^i_{a,c} < \xi^*_{a,c}$, it follows by (15) that there exists a category $\tilde{c} \in C$ such that

$$\xi^*_{a,\tilde{c}} < \xi^i_{a,\tilde{c}}. \tag{16}$$

By definition, $\xi^i_{a,\tilde{c}} \leq \max_{c' \in C}\{\xi^i_{a,c'}\}$ and, by (14), $\max_{c' \in C}\{\xi^i_{a,c'}\} \leq \max_{c' \in C}\{\xi^*_{a,c'}\}$; therefore, (16) implies that $\xi^*_{a,\tilde{c}} < \max_{c' \in C}\{\xi^*_{a,c'}\}$. As ξ^* is category neutral (by assumption), it follows that

$$\xi_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,\tilde{c}}} \xi^*_{a',\tilde{c}} = q_{\tilde{c}}. \tag{17}$$

Next, (16) implies that $\xi^i_{a,\tilde{c}} > 0$. As ξ^i complies with eligibility requirements (by Lemma A.10), it follows that a is eligible for $\tilde{c}$ so, by definition,

$$\xi^i_{a,\tilde{c}} = \min\{d^i_a, x^i_{a,\tilde{c}}\} = \min\{d^i_a, d^{i-1}_a, \max\{q_{\tilde{c}} - \sum_{a' \in \hat{A}_{a',\tilde{c}}} d^{i-1}_{a'}, 0\}\}.$$

It follows that $\xi^i_{a,\tilde{c}} \leq \max\{q_{\tilde{c}} - \sum_{a' \in \hat{A}_{a',\tilde{c}}} d^{i-1}_{a'}, 0\}$; hence $\xi^i_{a,\tilde{c}} > 0$ implies that $\xi^i_{a,\tilde{c}} \leq q_{\tilde{c}} - \sum_{a' \in \hat{A}_{a',\tilde{c}}} d^{i-1}_{a'}$ or, equivalently, $\xi^i_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a',\tilde{c}}} d^{i-1}_{a'} \leq q_{\tilde{c}}$. By the induction hypothesis, $\xi^*_{a,\tilde{c}} \leq d^{i-1}_{a'}$ for every $a' \in \hat{A}_{a,\tilde{c}}$; hence we have $\xi^i_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a',\tilde{c}}} \xi^*_{a',\tilde{c}} \leq q_{\tilde{c}}$. By (17), it follows that $\xi^i_{a,\tilde{c}} \leq \xi^*_{a,\tilde{c}}$, which contradicts (16). □

Lemma A.16. *For every agent a and every allocation ξ^* that satisfies Axioms 1-4,* $\max_{c \in C}\{\xi^*_{a,c}\} \leq \max_{c \in C}\{\xi^{SR}_{a,c}\}$.

Proof. Toward a contradiction, suppose to the contrary that $\max_{c \in C}\{\xi^*_{a,c}\} > \max_{c \in C}\{\xi^{SR}_{a,c}\}$. Then, there exists a category c' such that $\xi^*_{a,c'} > \xi^{SR}_{a,c'}$. By Theorem 3, we have $\xi^*_a = \xi^{SR}_a$ so, by definition, $\sum_{c \in C} \xi^*_{a,c} = \sum_{c \in C} \xi^{SR}_{a,c}$. Then, the fact that $\xi^*_{a,c'} > \xi^{SR}_{a,c'}$ implies there exists a category $\tilde{c}$ such that

$$\xi^*_{a,\tilde{c}} < \xi^{SR}_{a,\tilde{c}}. \tag{18}$$

It follows that $\xi^*_{a,\tilde{c}} < \max_{c \in C}\{\xi^*_{a,c}\}$ so, as ξ^* is category neutral, we have $\xi^*_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',\tilde{c}} = q_{\tilde{c}}$. Moreover, $\xi^{SR}_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,c}} \xi^{SR}_{a',\tilde{c}} \leq q_{\tilde{c}}$ by definition since ξ^{SR} is an allocation. It follows that

$$\xi^{SR}_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,c}} \xi^{SR}_{a',\tilde{c}} \leq \xi^*_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',\tilde{c}},$$

which combined with (18) implies that .

$$\sum_{a' \in \hat{A}_{a,c}} \xi^{SR}_{a',\tilde{c}} < \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',\tilde{c}}. \tag{19}$$

By (18), we have $\xi^{SR}_{a,\tilde{c}} > 0$; hence, by definition, $\lim_{i \to \infty} \xi^i_{a,\tilde{c}} > 0$. Similarly, (19) implies by definition that $\lim_{i \to \infty} \sum_{a' \in \hat{A}_{a,c}} \xi^i_{a',\tilde{c}} < \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',\tilde{c}}$. Then, there exists a Round i such that

$$\xi^i_{a,\tilde{c}} > 0 \quad \text{and} \tag{20}$$

$$\sum_{a' \in \hat{A}_{a,\tilde{c}}} \xi^i_{a',\tilde{c}} < \sum_{a' \in \hat{A}_{a,\tilde{c}}} \xi^*_{a',\tilde{c}}. \tag{21}$$

By Lemma A.8, (20) implies that $\xi^i_{a,\tilde{c}} = d^i_{a'}$ for all $a' \in \hat{A}_{a,c}$, which combined with (21) implies that $\sum_{a'\in\hat{A}_{a,\tilde{c}}} d^i_{a'} < \sum_{a'\in\hat{A}_{a,\tilde{c}}} \xi^*_{a',\tilde{c}}$. Therefore, there exists an agent $a' \in \hat{A}_{a,\tilde{c}}$ such that $d^i_{a'} < \xi^*_{a',\tilde{c}}$, which contradicts Lemma A.15. $\square$

B Proof of the Results from Section 4

Proof of Theorem 1

I have shown in the main text that ξ^{SR} is well defined and equal to both $\lim_{i\to\infty} x^i$ and $\lim_{i\to\infty} \xi^i$. It remains to show that ξ^{SR} is an allocation. Arbitrarily fix an agent a and a category c. By definition, it needs to be shown that (i) $\xi^{SR}_{a,c} \in [0,1]$, (ii) $\sum_{a'\in A} \xi^{SR}_{a',c} \leq q_c$, and (iii) $\xi^{SR}_a \leq 1$.

($\xi^{SR}_{a,c} \in [0,1]$) *Case 1: $\xi^i_{a,c} < d^i_a$* for every $i \geq 1$. By definition, the case assumption implies that $x^i_{a,c} < d^i_a$ for every $i \geq 1$, which by Lemma A.13 implies that $x^j_{a,c} \leq x^i_{a,c} < d^j_a$ for every $i, j \geq 1$ with $j \leq i$. By definition, it follows that the series $\{\xi^i_{a,c}\}_{i=1}^\infty$ is weakly increasing. By Lemma A.2, that series is bounded. Then, the Monotone Convergence Theorem implies that $\lim_{i\to\infty} \xi^i_{a,c}$ is equal to the series' supremum. By Lemma A.2, every element of the series $\{\xi^i_{a,c}\}_{i=1}^\infty$ is an element of $[0,1]$; hence so is its supremum.

Case 2: $\xi^i_{a,c} = d^i_a$ for some $i \geq 1$. By definition, the case assumption implies that $x^i_{a,c} \geq d^i_a$; hence Lemma A.14 implies that $x^j_{a,c} \geq d^j_a$ for every $j \geq i$. Again by definition, it follows that $\xi^j_{a,c} = d^j_a$ for all $j \geq i$, which implies that $\lim_{i\to\infty} \xi^i_{a,c} = \lim_{i\to\infty} d^i_a$ so it remains to show that $\lim_{i\to\infty} d^i_a \in [0,1]$.

The series $\{d^i_a\}_{i=1}^\infty$ is weakly decreasing by Lemma A.4 and bounded below by Lemma A.2. By the Monotone Convergence Theorem, $\lim_{i\to\infty} d^i_a$ is then equal to the infimum of the series $\{d^i_a\}_{i=1}^\infty$. By Lemma A.2, every element of that series is an element of $[1/|C|, 1]$; hence so is its infimum.

($\sum_{a'\in A} \xi^{SR}_{a,c} \leq q_c$) As $\lim_{i\to\infty} \xi^i_{a',c} \in [0,1]$ for every $a' \in A$, $\sum_{a'\in A} \xi^{SR}_{a',c} = \lim_{i\to\infty}(\sum_{a'\in A} \xi^i_{a',c}) = \sum_{a'\in A}(\lim_{i\to\infty} \xi^i_{a',c})$ converges to a real number. By Lemma A.3, ξ^i is an allocation for every $i \geq 1$; therefore every element of the series $\{\sum_{a'\in A} \xi^i_{a',c}\}_{i=1}^\infty$ is weakly smaller than q_c. Then, the number to which the series converges cannot exceed q_c.

($\xi^{SR}_a \leq 1$) As $\lim_{i\to\infty} \xi^i_{a,c'} \in [0,1]$ for every $c' \in C$, $\xi^{SR}_a = \sum_{c'\in C} \xi^{SR}_{a,c'} = \lim_{i\to\infty}(\sum_{c'\in C} \xi^i_{a,c'}) = \sum_{c'\in C}(\lim_{i\to\infty} \xi^i_{a,c'})$ is equal to a real number. By Lemma A.3, ξ^i is an allocation for every $i \geq 1$; therefore every element of the series $\{\xi^i_a\}_{i=1}^\infty$ is weakly smaller than 1. Then, the number to which the series converges cannot exceed 1. $\square$

Proof of Proposition 2

($|z^{i+1}| \leq |z^i|$) By definition, $|z^{i+1}| = |x^{i+1}| - |\xi^{i+1}|$ and $|z^i| = |x^i| - |\xi^i$ and, by Lemma A.4, $|\xi^{i+1}| \geq |\xi^i|$; therefore, it remains to show that $|x^{i+1}| \leq |x^i|$.

Consider first any category c such that $\sum_{a \in A} x^i_{a,c} < q_c$. I show that, for every $a \in A$,

$$x^i_{a,c} = \begin{cases} 0 & \text{if } a \text{ is not eligible for } c \\ d^{i-1}_a & \text{if } a \text{ is eligible for } c. \end{cases} \tag{22}$$

If a is not eligible for c, then $x^i_{a,c} = 0$ by definition; therefore, it remains to show that, if a is eligible for c, then $x^i_{a,c} = d^{i-1}_a$. Toward a contradiction, suppose that a is eligible for c and $x^i_{a,c} \neq d^{i-1}_a$. Let $\tilde{a}$ be the highest-priority agent in that situation; that is, $\tilde{a}$ is eligible for c, $x^i_{\tilde{a},c} \neq d^{i-1}_{\tilde{a}}$, and, for every $a' \in \hat{A}_{\tilde{a},c}$, $x^i_{a',c} = d^i_{a'}$. By definition, as $\tilde{a}$ is eligible for c, $x^i_{\tilde{a},c} = \min\{d^{i-1}_{\tilde{a}}, \max\{q_c - \sum_{a' \in \hat{A}_{\tilde{a},c}} d^{i-1}_{a'}, 0\}\}$ so the assumption that $x^i_{\tilde{a},c} \neq d^{i-1}_{\tilde{a}}$ implies that $x^i_{\tilde{a},c} < d^{i-1}_{\tilde{a}}$. It follows that $x^i_{\tilde{a},c} = \max\{q_c - \sum_{a' \in \hat{a}_{\tilde{a},c}} d^{i-1}_{a'}, 0\}$; therefore, $q_c - x^i_{\tilde{a},c} - \sum_{a' \in \hat{A}_{\tilde{a},c}} d^{i-1}_{a'} \leq 0$. Combining the last inequality with the assumption that $x^i_{a',c} = d^{i-1}_{a'}$ for every $a' \in \hat{A}_{\tilde{a},c}$ yields $q_c - x^i_{\tilde{a},c} - \sum_{a' \in \hat{A}_{\tilde{a},c}} x^i_{a',c} \leq 0$. It can then be concluded that $\sum_{a \in A} x^i_{a,c} \geq x^i_{\tilde{a},c} + \sum_{a' \in \hat{A}_{\tilde{a},c}} x^i_{a',c} \geq q_c$, a contradiction; hence (22) holds for every agent a. Letting $A_c = \{a \in A : a \pi_c \emptyset\}$ denote the set of agents that are acceptable for c, it follows that $\sum_{a \in A} x^i_{a,c} = \sum_{a \in A_c} d^{i-1}_a$.

Consider now any category c'. By Lemma A.2, $\sum_{a \in A} x^i_{a,c'} \leq q_{c'}$ and, by the previous argument, $\sum_{a \in A} x^i_{a,c'} < q_{c'}$ implies that $\sum_{a \in A} x^i_{a,c'} = \sum_{a \in A_{c'}} d^{i-1}_a$. It follows that $\sum_{a \in A} x^i_{a,c'} = \min\{q_{c'}, \sum_{a \in A_{c'}} d^{i-1}_a\}$ for every $c' \in C$. Therefore, we have $|x^i| = \sum_{c' \in C} \min\{q_{c'}, \sum_{a \in A_{c'}} d^{i-1}_a\}$ and, analogously, $|x^{i+1}| = \sum_{c' \in C} \min\{q_{c'}, \sum_{a \in A_{c'}} d^i_a\}$; hence, Lemma A.4 implies that $|x^{i+1}| \leq |x^i|$.

($|z^i| \leq |A|(|C| - 1)/i$)) Fix an agent a and a category c. I first show that

$$\sum_{j=1}^{i} z^j_{a,c} \leq 1 - 1/|C|. \tag{23}$$

By definition,

$$\sum_{j=1}^{i} z^j_{a,c} = \sum_{j=1}^{i} (x^j_{a,c} - \xi^j_{a,c}) = \sum_{j=1}^{i} (x^j_{a,c} - \min\{d^j_a, x^j_{a,c}\}) = \sum_{j=1}^{i} \max\{x^j_{a,c} - d^j_a, 0\}. \tag{24}$$

If $x^j_{a,c} \leq d^j_a$ for all $j = 1, \ldots, i$, then (23) holds trivially as (24) implies that $\sum_{j=1}^{i} z^j_{a,c} = 0$. Otherwise, let $k = 1, \ldots, i$ be the first round in which a receives her demand from c; that is, $x^k_{a,c} \geq d^k_a$ and $x^j_{a,c} < d^j_a$ for all $j = 1, \ldots, k-1$. Then, by Lemma A.14, $x^j_{a,c} = d^{j-1}_a$ for all

$j = k+1, \ldots, i$. By (24), we have

$$\sum_{j=1}^{i} z_{a,c}^j = \sum_{j=1}^{i} \max\{x_{a,c}^j - d_a^j, 0\} = \sum_{j=k}^{i} (x_{a,c}^j - d_a^j) = x_{a,c}^k - d_a^i + \sum_{j=k+1}^{i} (x_{a,c}^j - d_a^{j-1}) = x_{a,c}^k - d_a^i.$$

Then, (23) is satisfied as, by Lemma A.2, $x_{a,c}^k \leq 1$ and $d_a^i \geq 1/|C|$.

As a and c were chosen arbitrarily, (23) holds for every agent and every category; therefore, we have $\sum_{j=1}^{i} |z^j| \leq |A||C|(1-1/|C|) = |A|(|C|-1)$. The first part of the proof has established that $|z^{i+1}| \leq |z^i|$ and that result holds in every round; hence $|z^i| \leq |z^j|$ for all $j \leq i$. It can then be concluded that $i|z^i| \leq \sum_{j=1}^{i} |z^j| \leq |A|(|C|-1)$, which implies that $|z^i| \leq |A|(|C|-1)/i$. □

Proof of Proposition 3

I use for the SR algorithm the terminology introduced in Section 5 for the SRLP algorithm. That is, for any Round i of the SR algorithm, any agent a, and any category c, I say that that a is *qualified* for c is $x_{a,c}^i \geq d_a^i$, a is *marginal* for c is $0 < x_{a,c}^i < d_a^i$, and a is *unqualified* for c if $x_{a,c}^i = 0$. I refer to agent a's quality as either qualified, marginal or qualified as a's *status* for c. Using the convention that $x^0 = \mathbf{0}$, every agent is initially unqualified for every category. By Lemmas A.13 and A.14, a change of status is irreversible; therefore, throughout the SR algorithm, the status of any agent a for any category c can change at most twice: once from unqualified to marginal and once from marginal to qualified. As $|C| = 2$, it follows that there may be at most $4|A|$ status changes throughout the SR algorithm. The key part of the proof is to show that at least one status changes every second round until the SR algorithm finds an allocation.

Claim 1. *Suppose that, for some $i \geq 2$, the status of every agent a for every category c is the same in Rounds $i-1$, i, and $i+1$. Then, x^{i+1} is an allocation.*

Proof. Consider an agent-category pair (a, c) such that a is marginal for c in Round i, i.e., $0 < x_{a,c}^i < d_a^i$. Consider another agent a' and suppose first that a' has a lower priority for c, i.e., $a\pi_c a'$. By Lemma A.4, $x_{a,c}^i < d_a^{i-1}$ so, by Lemma A.7, $x_{a',c}^i = 0$ and a' is unqualified for c. Suppose instead that a' has a lower priority for c, i.e., $a'\pi_c a$. By Lemma A.8, $\xi_{a',c} = d_{a'}^i$; hence by definition $x_{a',c}^i \geq d_{a'}^i$ and a' is qualified for c. It can then be concluded that, for any category, there is at most one marginal agent and, if such an agent exists, all higher-priority agents are qualified while all lower-priority agents are unqualified for that category.

Next, consider an agent a who is not marginal for either category in Rounds $i-1$, i, and $i+1$ (by assumption, a's status for each category is the same in all three rounds). If a is unqualified for at least one category, then she receives capacity from at most one category so her aggregate allocation cannot exceed one: $x_a^{i-1}, x_a^i, x_a^{i+1} \leq 1$. By definition it follows that $d_a^{i-1} = d_a^i = d_a^{i+1} = 1$. Therefore, for each category c, $x_{a,c}^{i-1} = x_{a,c}^i = x_{a,c}^{i+1} \in \{0, 1\}$. Suppose next that a is qualified for both categories in all three rounds. Then, $x_{a,c}^{i-1} \geq d_a^{i-1}$ for each c and, as $d_a^{i-1} \geq 0.5$ by Lemma A.2, it must be that $x_{a,c}^{i-1} \geq 0.5$ for each c. By definition, it follows that $d_a^{i-1} = 0.5$. As the demand decreases throughout the SR algorithm (Lemma A.4) but cannot fall below 0.5 (Lemma A.2), we then have that t $d_a^{i-1} = d_a^i = d_a^{i+1} = 0.5$, which by Lemma A.14 implies that $x_{a,c}^i = x_{a,c}^{i+1} = 0.5$ for each c. It can then be concluded that, for every agent a who is not marginal for either category and for every category c,

$$x_{a,c}^i = x_{a,c}^{i+1} \in \{0, 0.5, 1\}, \quad x_a^{i+1} \leq 1, \quad \text{and} \quad d_a^{i-1} = d_a^i = d_a^{i+1} \in \{0.5, 1\}. \tag{25}$$

I now show that x^{i+1} is an allocation; that is, I show that no agent is allocated more than one unit, i.e., $x_a^{i+1} \leq 1$ for every agent a. If no agent is marginal for either category, (25) implies that result. As there are two categories and at most one agent is marginal for each, there are at most two agents who are marginal for at least one of the categories. I consider separately the case in which there is one and the case in which there are two such agents.

Case 1: one marginal agent. Let a_1 be the only agent who is marginal for at least one category in Rounds $i-1$, i, and $i+1$. Toward a contradiction, suppose that $x_{a_1}^{i+1} > 1$. If a_1 is marginal for both categories, she receives from each an amount of capacity smaller than her demand; therefore, by Lemma A.6, $\xi_a^{i+1} < 1$, which by Lemma A.3 implies that $x_a^{i+1} < 1$, a contradiction. Therefore, a_1 is marginal for one category but not the other. Let c_1 be the category for which a_1 is marginal and c_2 be the other category. If a_1 is unqualified for c_2, then $x_{a_1}^{i+1} = x_{a_1,c_1}^{i+1} \leq 1$, a contradiction; therefore, a_1 is qualified for c_2 in Rounds $i-1$, i, and $i+1$. Then, for every $j = i-1, i, i+1$, we have $x_{a_1,c_1}^j < d_{a_1}^j \leq x_{a_1,c_2}^j$ so, by Lemma A.6, $\xi_{a_1}^j = 1$ and, by Lemma A.3, $x_{a_1}^j \geq 1$. By definition, if $x_{a_1}^j = 1$ then $d_{a_1}^j = \max\{x_{a_1,c_1}^j, x_{a_1,c_2}\} = x_{a_1,c_2}$ and, if $x_{a_1}^j > 1$, then $\min\{d_{a_1}^j, x_{a_1,c_1}^j\} + \min\{d_{a_1}^j, x_{a_1,c_2}^j\} = 1$. Therefore, in both cases, we have $x_{a_1,c_1}^j + d_{a_1}^j = 1$ for every $j = i-1, i, i+1$. Next, as a_1 is marginal for c_1, Lemma A.9 implies that $x_{a_1,c_1}^j + \sum_{a' \in \hat{A}_{a_1,c_1}} x_{a',c_1}^j = q_{c_1}$; hence, by (25), we have $x_{a_1,c_1}^i = x_{a_1,c_1}^{i+1}$, which implies that $d_{a_1}^i = d_{a_1}^{i+1}$. Last, Lemma A.14 implies that $x_{a_1,c_2}^{i+1} = d_{a_1}^i$. It follows that

$$x_{a_1,c_1}^{i+1} + x_{a_1,c_2}^{i+1} = x_{a_1,c_1}^{i+1} + d_{a_1}^i = x_{a_1,c_1}^{i+1} + d_{a_1}^{i+1} = 1,$$

a contradiction. By (25), it can then be concluded that x^{i+1} does not allocate more than one unit to any agent in aggregate; hence x^{i+1} is an allocation.

Case 2: two marginal agents. Denote by a_1 the agent who is marginal for category c_1 and by a_2 the agent who is marginal for category c_2. If a_2 is unqualified for c_1, then $x_{a_2}^{i+1} =$

$x^{i+1}_{a_2,c_2} \leq 1$ and analogous reasoning to that in Case 1 establishes that $x^{i+1}_{a_1} \leq 1$ so, by (25), x^{i+1} is an allocation. Analogously, x^{i+1} is an allocation if a_1 is unqualified for c_2. Therefore, I focus on the case in which a_1 is qualified for c_2 and a_2 is qualified for c_1.

Arbitrarily fix $j = i, i+1$. Analogous reasoning to that in Case 1 establishes that $x^{j-1}_{a_1,c_1} + d^{j-1}_{a_1} = 1$; moreover, as $x^{j-1}_{a_1,c_2} \geq d^{j-1}_{a_1}$, Lemma A.14 implies that $x^{j}_{a_1,c_2} - d^{j-1}_{a_1}$. It follows that $x^{j-1}_{a_1,c_1} + x^{j}_{a_1,c_2} = 1$. As analogous reasoning for c_2 yields that $x^{j-1}_{a_2,c_2} + x^{j}_{a_2,c_1} = 1$, we have

$$x^{j-1}_{a_1,c_1} + x^{j-1}_{a_2,c_2} + x^{j}_{a_1,c_2} + x^{j}_{a_2,c_1} = 2. \tag{26}$$

By Lemma A.9, $x^{j}_{a_1,c_1} + \sum_{a' \in \hat{A}_{a_1,c_1}} x^{j}_{a',c_1} = q_{c_1}$; moreover, as c_1 does not allocate any capacity to any agent with a lower priority than a_1, we have $\sum_{a \in A} x^{j}_{a,c_1} = q_{c_1}$. Analogous reasoning for c_2 yields $\sum_{a \in A} x^{j}_{a,c_2} = q_{c_2}$; hence it can be concluded that $\sum_{a \in A} \sum_{c \in C} x^{j}_{a,c} = q$. By (25), the sum $\sum_{a \in A \setminus \{a_1,a_2\}} \sum_{c \in C} x^{j}_{a,c}$ is equal to either zero or a multiple of 0.5 (since each element of the sum is either 0, 0.5, or 1). As q is by definition an integer, it follows that $x^{j}_{a_1,c_1} + x^{j}_{a_2,c_2} + x^{j}_{a_1,c_2} + x^{j}_{a_2,c_1}$ is a multiple of 0.5. As $x^{j}_{a_1,c_1} < d^{j}_{a_1}$ and $x^{j}_{a_2,c_2} < d^{j}_{a_2}$, Lemma A.13 implies that $x^{j-1}_{a_1,c_1} \leq x^{j}_{a_1,c_1}$ and $x^{j-1}_{a_2,c_2} \leq x^{j}_{a_2,c_2}$. By (26), it follows that

$$x^{j}_{a_1,c_1} + x^{j}_{a_2,c_2} + x^{j}_{a_1,c_2} + x^{j}_{a_2,c_1} \in \{2, 2.5, \ldots\}. \tag{27}$$

If $x^{j-1}_{a_1,c_1} = x^{j}_{a_1,c_1}$ and $x^{j-1}_{a_2,c_2} = x^{j}_{a_2,c_2}$, then $x^{j}_{a_1,c_1} + x^{j}_{a_1,c_2} = 1 + x^{j}_{a_2,c_2} + x^{j}_{a_2,c_1} = 1$. Then, (25) implies that x^j is an allocation, which by Lemma A.12 implies that x^{i+1} is an allocation and completes the proof.

If either $x^{j-1}_{a_1,c_1} < x^{j}_{a_1,c_1}$ or $x^{j-1}_{a_2,c_2} < x^{j}_{a_2,c_2}$, then (26) and (27) imply that $(x^{j}_{a_1,c_1} + x^{j}_{a_2,c_2}) - (x^{j-1}_{a_1,c_1} + x^{j-1}_{a_2,c_2}) \geq 0.5$. As j was picked arbitrarily, it follows that $(x^{i+1}_{a_1,c_1} + x^{i+1}_{a_2,c_2}) - (x^{i-1}_{a_1,c_1} + x^{i-1}_{a_2,c_2}) \geq 1$. Then, it can be concluded that $x^{i+1}_{a_1,c_1} + x^{i+1}_{a_2,c_2} \geq 1$; hence either $x^{i+1}_{a_1,c_1} \geq 0.5$ or $x^{i+1}_{a_2,c_2} \geq 0.5$. If $x^{i+1}_{a_1,c_1} \geq 0.5$, then analogous reasoning to that in Case 1 establishes that $x^{i+1}_{a_1,c_1} + d^{i+1}_{a_1} = 1$ so $d^{i+1}_{a_1} \leq 0.5 \leq x^{i+1}_{a_1,c_1}$, which contradicts the assumption that a_1 is marginal for c_1 in Round $i+1$. Assuming that $x^{i+1}_{a_2,c_2} \geq 0.5$ analogously implies that $d^{i+1}_{a_2} \leq 0.5 \leq x^{i+1}_{a_2,c_2}$, which contradicts the assumption that a_2 is marginal for c_2 in Round $i+1$. □

If the SR algorithm finds an allocation in one of the $8|A|-2$ first rounds, the proof is complete; therefore I focus on the case in which the SR algorithm has not found an allocation after $8|A|-2$ rounds and show that, in that case, the SR algorithm finds an allocation in Round $8|A|-1$. In Round 1, at least one unit is allocated (since $q \geq 1$) so either an agent becomes qualified for a category or one agent becomes marginal for each category. It follows that at most $4|A|-2$ status changes occur after Round 1. By Claim 1, at least one status changes every two rounds; therefore, in Round $8|A|-3$, every agent is qualified for every category. Then, by Claim 1, the SR algorithm finds an allocation in Round $8|A|-1$.

Proof of Theorem 2

By Lemma A.10, ξ^i complies with eligibility requirements and respects priorities and x^i satisfies all four properties in every Round i, and by Corollary 1, $\xi^{SR} = \lim_{i\to\infty}\xi^i = \lim_{i\to\infty} x^i$. I use those two results to show that ξ^{SR} satisfies all four properties.

(ξ^{SR} complies with eligibility requirements) For every agent a. every category c for which a is not eligible, and every Round i, as x^i complies with eligibility requirements we have $x^i_{a,c} = 0$ for all $i \geq 1$. It follows that $\xi^{SR}_{a,c} = \lim_{i\to\infty} x^i_{a,c} = 0$.

(ξ^{SR} is nonwasteful) Consider any category c such that $\sum_{a\in A}\xi^{SR}_{a,c} < q_c$ and any agent a who is eligible for c. It needs to be shown that $\xi^{SR}_a \geq 1$. As $\sum_{a\in A}\xi^{SR}_{a,c} < q_c$, it must be that $\lim_{i\to\infty}\sum_{a\in A} x^i_{a,c} < q_c$; hence there exists a Round j such that, for all $i \geq j$, $\sum_{a\in A} x^i_{a,c} < q_c$. As x^i is nonwasteful and a is eligible for c,we have $x^i_a \geq 1$. Then, $x^{SR}_a = \lim_{i\to\infty} x^i_a \geq 1$.

(ξ^{SR} respects priorities) Arbitrarily fix an agent a such that $\xi^{SR}_a < 1$, a category c, and a lower-priority agent $a' \in \check{A}_{a,c}$. It needs to be shown that $\xi^{SR}_{a',c} = 0$. By Lemma A.4, for every Round i, $\xi^i_a \leq \xi^{SR}_a < 1$. As ξ^i respects priorities, it follows that $\xi^i_{a',c} = 0$ for all $i \geq 1$; hence we have $\xi^{SR}_{a',c} = \lim_{i\to\infty}\xi^i_{a',c} = 0$.

(ξ^{SR} is category neutral) Consider an agent a and a category c such that a is eligible for c and $\xi^{SR}_{a,c} < \max_{c'\in C}\{\xi^{SR}_{a,c'}\}$. It needs to be shown that $\xi^{SR}_{a,c} + \sum_{a'\in\hat{A}_{a,c}}\xi^{SR}_{a',c} = q_c$. By assumption, we have $\lim_{i\to\infty} x^i_{a,c} < \lim_{i\to\infty}\max_{c'\in C}\{x^i_{a,c'}\}$. Then, there exists a Round j such that, for all $i \geq j$, $x^i_{a,c} < \max_{c'\in C}\{x^i_{a,c'}\}$. For every $i \geq j$, x^i is nonwasteful; hence $x^i_{a,c} + \sum_{a'\in\hat{A}_{a,c}} x^i_{a',c} = q_c$ for all $i \geq j$. We can then conclude that $\xi^{SR}_{a,c} + \sum_{a'\in\hat{A}_{a,c}}\xi^{SR}_{a',c} = \lim_{i\to\infty}(x^i_{a,c} + \sum_{a'\in\hat{A}_{a,c}} x^i_{a',c}) = q_c$. □

Proof of Theorem 3

Let ξ^* be an allocation that satisfies Axioms 1-4; it needs to be shown that $\xi^*_a = \xi^{SR}_a$ for every agent a. Toward a contradiction, suppose to the contrary that $\xi^*_{\tilde{a}} \neq \xi^{SR}_{\tilde{a}}$ for some agent $\tilde{a}$. Lemma A.15 implies that $\xi^*_a \geq \xi^i_a$ for every agent a and every Round i; therefore, for every agent a, we have $\xi^*_a \geq \lim_{i\to\infty}\xi^i_a = \xi^{SR}_a$. It follows that $\xi^*_{\tilde{a}} > \xi^{SR}_{\tilde{a}}$ and, for every agent $a \neq \tilde{a}$, $\xi^*_a \geq \xi^{SR}_a$; hence we have $|\xi^*| > |\xi^{SR}|$. Consequently, there must exist a category c such that

$$\sum_{a\in A}\xi^*_{a,c} > \sum_{a\in A}\xi^{SR}_{a,c}. \tag{28}$$

By definition (as ξ^* is an allocation), $\sum_{a\in A}\xi^*_{a,c} \leq q_c$; hence (28) implies that $\sum_{a\in A}\xi^{SR}_{a,c} < q_c$. By Corollary 1, $\lim_{i\to\infty}\sum_{a\in A} x^i_{a,c} < q_c$; therefore, there exists a Round j such that

$\sum_{a\in A} x^i_{a,c} < q_c$ for every $i \geq j$. Then, by Lemma A.9, for every agent a who is eligible for c, we have $x^i_{a,c} = d^{i-1}_a$. By definition, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\} = \min\{d^i_a, d^{i-1}_a\}$ so, by Lemma A.4, $\xi^i_{a,c} = d^i_a$. As $\xi^*_{a,c} \leq d^i_a$ by Lemma A.15, we have $\xi^*_{a,c} \leq \xi^i_{a,c}$. Moreover, for any agent a who is not eligible for c, $\xi^i_{a,c} = 0$ by definition and $\xi^*_{a,c} = 0$ as ξ^* complies with eligibility requirements. Therefore, we can conclude that $\xi^*_{a,c} \leq \xi^i_{a,c}$ for every Round $i \geq j$ and every agent a. Then, for every agent a, $\xi^*_{a,c} \leq \lim_{i\to\infty} \xi^i_{a,c} = \xi^{SR}_{a,c}$, which contradicts (28). $\square$

Proof of Proposition 4

Let there be three agents a_1, a_2, and a_3 as well as two categories c_1 and c_2. For each of the four axioms, I construct quotas and priorities such that an allocation that is not SR equivalent satisfies the other three axioms.

(Complies with eligibility requirements) Let the quotas and priorities be

$$q_{c_1} = 2 \quad q_{c_2} = 1 \quad \pi_{c_1} : a_1, \emptyset, a_3, a_2 \quad \pi_{c_2} : a_2, \emptyset, a_3, a_1.$$

The SR algorithm finds the SR allocation after just one round: each category allocates one unit of capacity to its highest-priority agent, respectively a_1 and a_2. Hence, we have

$$\xi^{SR} = \begin{array}{c c} & \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} & \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \end{array} \quad \text{and} \quad \rho^{SR} = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 \end{pmatrix} \end{array}.$$

Consider the alternative allocation

$$\xi = \begin{array}{c c} & \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} & \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \end{array} \quad \text{with} \quad \rho(\xi) = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \end{array}.$$

The allocation ξ is not SR equivalent since $\rho(\xi) \neq \rho^{SR}$. However, ξ is nonwasteful since the amount of capacity allocated by each category is equal to its quota, ξ respects priority since every agent is allocated one unit, and ξ is category neutral since, for every agent-object pair (a, c) such that a is eligible for c, we have $\xi_{a,c} = \max_{c'\in C}\{\xi_{a,c'}\}$ (there are two such pairs, (a_1, c_1) and (a_2, c_2), and $\xi_{a_1,c_1} = \xi_{a_2,c_2} = 1$).

(nonwasteful) Let the quotas and priorities be

$$q_{c_1} = 2 \quad q_{c_2} = 1 \quad \pi_{c_1} : a_1, a_3, \emptyset, a_2 \quad \pi_{c_2} : a_2, \emptyset, a_3, a_1.$$

The SR algorithm finds the SR allocation after just one round: c_1 allocates its two units to its two highest-priority agents—a_1 and a_3—and c_2 allocates its unique unit to its highest-priority agent—a_2. Hence, we have

$$\xi^{SR} = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \!\!\begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \end{array} \quad \text{and} \quad \rho^{SR} = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \end{array}.$$

Consider the alternative allocation

$$\xi = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \!\!\begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \end{array} \quad \text{with} \quad \rho(\xi) = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 \end{pmatrix} \end{array}.$$

The allocation ξ is not SR equivalent since $\rho(\xi) \neq \rho^{SR}$. However, ξ complies with eligibility requirements since agents only receive capacity from categories for which they are eligible, ξ respects priorities as a_3 is the only agent who is not allocated one unit in aggregate and every agent with a lower priority than a_3 at either category is allocated 0 from that category, and a_3 is category neutral as a_1 is not eligible for c_2, a_2 is not eligible for c_1, and $\xi_{a_3,c_1} = \xi_{a_3,c_2}$.

(Respects priorities) Let the quotas and priorities be

$$q_{c_1} = 1 \quad q_{c_2} = 1 \quad \pi_{c_1} : a_1, a_3, a_2, \emptyset \quad \pi_{c_2} : a_1, a_2, a_3, \emptyset.$$

The SR algorithm finds the SR allocation after two rounds. In Round 1, both categories allocate one unit to a_1, which has the highest-priority for both categories. Therefore, a_1's demand decreases to $1/2$ and in Round 2 each category allocates half a unit to its second highest-priority agent, respectively a_3 and a_2. Hence, we have

$$\xi^{SR} = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \!\!\begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 1/2 & 1/2 \\ 0 & 1/2 \\ 1/2 & 0 \end{pmatrix} \end{array} \quad \text{and} \quad \rho^{SR} = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1/2 & 1/2 \end{pmatrix} \end{array}.$$

Consider the alternative allocation

$$\xi = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 0 & 0 \\ 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \end{array} \quad \text{with} \quad \rho(\xi) = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 \end{pmatrix} \end{array}.$$

The allocation ξ is not SR equivalent since $\rho(\xi) \neq \rho^{SR}$. However, ξ complies with eligibility requirements as agents only receive capacity from categories for which they are eligible, ξ is nonwasteful as each category allocates overall an amount of capacity equal to its quota, and ξ is category neutral as each agent is allocated the same amount of capacity from both categories.

(Category neutrality) Let the quotas and priorities be identical to the previous example: $q_{c_1} = q_{c_2} = 1$, $\pi_{c_1} : a_1, a_3, a_2, \emptyset$ and $\pi_{c_2} : a_1, a_2, a_3, \emptyset$. We have again that

$$\xi^{SR} = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 1/2 & 1/2 \\ 0 & 1/2 \\ 1/2 & 0 \end{pmatrix} \end{array} \quad \text{and} \quad \rho^{SR} = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 1/2 & 1/2 \end{pmatrix} \end{array}.$$

Consider the alternative allocation

$$\xi = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \begin{array}{c} \begin{array}{cc} c_1 & c_2 \end{array} \\ \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \end{array} \quad \text{with} \quad \rho(\xi) = \begin{array}{c} \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\ \begin{pmatrix} 1 & 0 & 1 \end{pmatrix} \end{array}.$$

The allocation ξ is not SR equivalent since $\rho(\xi) \neq \rho^{SR}$. However, ξ complies with eligibility requirements as agents only receive capacity from categories for which they are eligible, ξ is nonwasteful as each category allocates overall an amount of capacity equal to its quota, and ξ respects priorities as a_2 is the only agent not to be allocated one unit in aggregate and has a lower priority than a_1 for c_1 and a lower priority than a_2 for c_2. $\square$

Proof of Theorem 4

It needs to be shown that $d_a(\xi^*) \leq d_a(\xi^{SR})$ for every agent a and $d_a(\xi^*) < d_a(\xi^{SR})$ for some agent a. I prove each of the two statements separately.

($d_a(\xi^*) \leq d_a(\xi^{SR})$ for every agent a) Consider any agent a. If $\xi_a^{SR} < 1$, then by definition

$d_a(\xi^*) \leq 1 = d_a(\xi^{SR})$. If $\xi_a^{SR} = 1$, then $\xi_a^* = 1$ by Theorem 3. Therefore, by definition we have $d_a(\xi^*) = \max_{c \in C}\{\xi_{a,c}^*\}$ and $d_a(\xi^{SR}) = \max_{c \in C}\{\xi_{a,c}^{SR}\}$. By Lemma A.16, it follows that $d_a(\xi^*) = \max_{c \in C}\{\xi_{a,c}^*\} \leq \max_{c \in C}\{\xi_{a,c}^{SR}\} = d_a(\xi^{SR})$.

($d_a(\xi^*) < d_a(\xi^{SR})$ for some agent a) As $\xi^* \neq \xi^{SR}$, there exists an agent a and a category c such that $\xi_{a,c}^* \neq \xi_{a,c}^{SR}$. By Theorem 3, $\xi_a^* = \xi_a^{SR}$ so $\xi_{a,c}^* > \xi_{a,c}^{SR}$ implies that $\xi_{a,\tilde{c}}^* < \xi_{a,\tilde{c}}^{SR}$ for some category $\tilde{c}$. Therefore, without loss of generality, I assume that

$$\xi_{a,c}^* < \xi_{a,c}^{SR} \tag{29}$$

and show that $d_a(\xi^*) \leq d_a(\xi^{SR})$.

First, observe that (29) implies that a is eligible for c; otherwise, as ξ^* and ξ^{SR} comply with eligibility requirements, we would have that $\xi_{a,c}^* = \xi_{a,c}^{SR} = 0$. Second, I show the following intermediate result:

Claim 2. $\sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^* \leq \sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^{SR}$.

Proof. Suppose to the contrary that $\sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^* > \sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^{SR}$. Then, there exists an agent $\tilde{a} \in \hat{A}_{a,c}$ such that $\xi_{\tilde{a},c}^* > \xi_{\tilde{a},c}^{SR}$. As ξ^{SR} is an allocation, by definition $\sum_{a' \in A} \xi_{a',c}^{SR} \leq q_c$ and, by (29), $\xi_{a,c}^{SR} > 0$; therefore, as $\tilde{a}\pi_c a$, it can be concluded that $\xi_{\tilde{a},c}^{SR} + \sum_{a' \in \hat{A}_{\tilde{a},c}} \xi_{a',c}^{SR} < q_c$. As ξ^{SR} is category neutral and $\tilde{a}$ is eligible for c (since a is eligible for c and $\tilde{a}$ has a higher priority), it follows that $\xi_{\tilde{a},c}^{SR} = \max_{c' \in C}\{\xi_{\tilde{a},c'}^{SR}\}$. Then, the fact that $\xi_{\tilde{a},c}^* > \xi_{\tilde{a},c}^{SR}$ implies that

$$\max_{c' \in C}\{\xi_{\tilde{a},c'}^*\} \geq \xi_{\tilde{a},c}^* > \xi_{\tilde{a},c}^{SR} = \max_{c' \in C}\{\xi_{\tilde{a},c'}^{SR}\},$$

which contradicts Lemma A.16. $\square$

Having established Claim 2, I now use it to show that $d_a(\xi^*) \leq d_a(\xi^{SR})$. I consider separately the cases in which $\xi_a^{SR} < 1$ and $\xi_a^{SR} = 1$.

Case 1: $\xi_a^{SR} < 1$. In that case, by Theorem 3, $\xi_a^* = \xi_a^{SR} < 1$. As ξ^* and ξ^{SR} are nonwasteful, $\xi_a^* = \xi_a^{SR} < 1$, and a is eligible for c, we have $\sum_{a \in A} \xi_{a,c}^* = \sum_{a \in A} \xi_{a,c}^{SR} = q_c$. Moreover, as ξ^* and ξ^{SR} respect priorities and $\xi_a^* = \xi_a^{SR} < 1$, we have $\xi_{a',c}^* = \xi_{a',c}^{SR} = 0$ for every lower-priority agent $a' \in \check{A}_{a,c}$. It follows that

$$\xi_{a,c}^* + \sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^* = \xi_{a,c}^{SR} + \sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^{SR} = q_c.$$

Then, (29) implies that $\sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^* > \sum_{a' \in \hat{A}_{a,c}} \xi_{a,c}^{SR}$, which contradicts Claim 2. Therefore, we must be in Case 2.

Case 2: $\xi^{SR}_a = 1$. In that case, by Theorem 3, $\xi^*_a = \xi^{SR}_a = 1$. I consider separately two subcases: $\xi^*_{a,c} < \max_{c' \in C}\{\xi^*_{a,c'}\}$ and $\xi^*_{a,c} = \max_{c' \in C}\{\xi^*_{a,c'}\}$.

Subcase 2.1: $\xi^*_{a,c} < \max_{c' \in C}\{\xi^*_{a,c'}\}$. As ξ^* is category neutral and a is eligible for c, we have $\xi^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',c} = q_c$. Moreover, as ξ^{SR} is an allocation, we have $\xi^{SR}_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^{SR}_{a',c} \leq q_c$. Then, (29) implies that $\sum_{a' \in \hat{A}_{a,c}} \xi^*_{a,c} > \sum_{a' \in \hat{A}_{a,c}} \xi^{SR}_{a,c}$, which contradicts Claim 2. Therefore, we must be in Subcase 2.2.

Subcase 2.2: $\xi^*_{a,c} = \max_{c' \in C}\{\xi^*_{a,c'}\}$. As $\xi^*_a = \xi^{SR}_a = 1$, by definition we have $d_a(\xi^*) = \max_{c' \in C}\{d_a(\xi^*)\}$ and $d_a(\xi^{SR}) = \max_{c' \in C}\{d_a(\xi^{SR})\}$. Using those two results in conjunction with the subcase assumption and (29) yields

$$d_a(\xi^*) = \max_{c' \in C}\{\xi^*_{a,c'}\} = \xi^*_{a,c} < \xi^{SR}_{a,c} \leq \max_{c' \in C}\{\xi^{SR}_{a,c'}\} = d_a(\xi^{SR});$$

hence, it can be concluded that $d_a(\xi^*) \leq d_a(\xi^{SR})$, as required. □

Proof of Proposition 5

Let $A^* = \{a \in A : \xi^{SR}_a \in (0,1)\}$ be the set of agents who are allocated an amount of capacity strictly between zero and one at the SR aggregate allocation. It needs to be shown that $|A^*| \leq |C|$. Suppose toward a contradiction that $|A^*| > |C|$. At the SR allocation, every agent in A^* must be allocated a positive amount of capacity by at least one category (otherwise that agent would not be allocated any capacity in aggregate and would therefore not be an element of A^*). Then, the assumption that $|A^*| > |C|$ implies that there exists a category that allocates a positive amount of capacity to at least two agents. That is, there exist a category c and two agents $a_1, a_2 \in A^*$ with $a_1 \pi_c a_2$ such that $\xi^{SR}_{a_1,c} > 0$ and $\xi^{SR}_{a_2,c} > 0$. As $\xi^{SR}_{a_1} < 1$, ξ^{SR} does not respect priorities, which contradicts Theorem 2. □

C Properties of the SRLP Algorithm

Lemma C.1. *For every Round $i \geq 1$, ξ^i is an allocation and, for every agent a, $\xi^i_a = \min\{x^i_a, 1\}$.*

Proof. Lemma C.1 is a counterpart to Lemma A.3 for the SRLP algorithm and its proof is completely analogous to that of Lemma A.3. □

Lemma C.2. *For every Round i of the SRLP algorithm and every agent a, we have*

$$d^i_a = \begin{cases} 1 & \text{if } \xi^i_a < 1 \\ \max_{c \in C}\{\xi^i_{a,c}\} & \text{if } \xi^i_a = 1. \end{cases}$$

and $\xi_a^i = 1$ *if and only if there exists a category* c *such that* $\xi_{a,c}^i = d_a^i$.

Proof. Lemma C.2 is a counterpart to Lemmas A.5 and A.6 for the SRLP algorithm and its proof is completely analogous to those of Lemmas A.5 and A.6. $\square$

Lemma C.3. *For every Round* i *of the SRLP algorithm and every agent* a, $\xi_a^i \geq \xi_a^{i-1}$ *and* $d_a^i \leq \delta_a^i \leq d_a^{i-1}$.

Proof. Lemma C.3 is a counterpart to Lemma A.4 for the SRLP algorithm. By an analogous reasoning to that of Lemma A.4, the statement holds for Round 1: $\xi_a^1 \geq \xi_a^0$ and $d_a^1 \leq d_a^0$.

The remainder of the proof is by induction. For some $i \geq 2$, suppose that $\xi_a^{i-1} \geq \xi_a^{i-2}$ and $d_a^{i-1} \leq d_a^{i-2}$ for all $a \in A$ (induction hypothesis). I show that $\xi_a^i \geq \xi_a^{i-1}$ and $d_a^i \leq d_a^{i-1}$. The part that differs form the proof of Lemma A.4 is that one needs to show that $\delta_a^i \leq d_a^{i-1}$. The result is obtained directly if the SRLP algorithm does not use linear programming in Round i since, in that case, $\delta_a^i = d_a^{i-1}$. If the SRLP does use linear programming, then $\delta_a^i = d_a^{i-1}$ for every $a \notin \widetilde{A}$ so I focus on the case in which $a \in \widetilde{A}$. As shown in the proof of Lemma D.1, the vector $(\xi_{a(c),c}^{i-1})_{c \in \widetilde{C}}$ satisfies the constraints of the linear program (LP 1); therefore, the vector $(\xi_{a,c}^{i-1})_{c \in C_M(a)}$ satisfies the constraints of the linear program (LP 3). Let $(\xi_{a(c),c}^{*})_{c \in \widetilde{C}}$ be the solution to the linear program (LP 1); then, the vector $(\xi_{a,c}^{*})_{c \in C_M(a)}$ is the solution to the linear program (LP 3). It follows that $\sum_{c \in C_M(a)} \xi_a^{i-1} \leq \sum_{c \in C_M(a)} \xi_a^{*}$. By construction,

$$|C_Q(a)| d_a^{i-1} + \sum_{c \in C_M(a)} \xi_{a,c}^{i-1} = |C_Q(a)| \delta_a^i + \sum_{c \in C_M(a)} \xi_{a,c}^{*} = 1;$$

hence it can be concluded that $\delta_a^i \leq d_a^{i-1}$. As the SRLP algorithm constructs ξ^i from δ^i as well as d^i and ξ^i from x^i identically to the SR algorithm, analogous reasoning to that in the proof of Lemma A.4 implies that $\xi_a^i \geq \xi_a^{i-1}$ and $d_a^i \leq \delta_a^i$. $\square$

Lemma C.4. *For every Round* i *of the SRLP algorithm, every agent* a*, and every category* c*, either* $x_{a,c}^i < \delta_a^i$ *or* $\xi_{a,c}^i < d_a^i$ *implies that* $x_{a',c}^i = \xi_{a',c}^i = 0$ *for every lower-priority agent* $a' \in \check{A}_{a,c}$.

Proof. Lemma C.4 is a counterpart to Lemma A.7 for the SRLP algorithm. The proof is completely analogous to that of Lemma A.9, the only difference is that d_a^{i-1} needs to be replaced throughout by δ_a^i. $\square$

Lemma C.5. *For every agent* a *and category* c *such that* a *is eligible for* c*, and for every Round* i, $x_{a,c}^i < \delta_a^i$ *implies that* $x_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} x_{a',c}^i = q_c$.

Proof. Lemma C.5 is a counterpart to Lemma A.9 for the SRLP algorithm. The proof is completely analogous to that of Lemma A.9, the only difference is that d_a^{i-1} needs to be replaced throughout by δ_a^i. $\square$

Lemma C.6. *Suppose that, for some agent a, some category c, and some Round $i \geq 1$ of the SRLP algorithm, $x^i_{a,c} < d^i_a$. Then, for every $j \leq i$, $x^j_{a,c} \leq x^i_{a,c} < d^j_a$.*

Proof. Lemma C.6 is a counterpart to Lemma A.13 for the SRLP algorithm. The proof is completely analogous to that of Lemma A.13, the only difference is that Lemma C.3 needs to be used instead of Lemma A.4. □

Lemma C.7. *Suppose that, for some agent a, some category c and some Round $i \geq 1$ of the SRLP algorithm, $x^i_{a,c} \geq d^i_a$. Then, for every $j > i$, $x^j_{a,c} = d^{j-1}_a \geq d^j_a$.*

Proof. Lemma C.7 is a counterpart to Lemma A.14 for the SRLP algorithm. The proof is completely analogous to that of Lemma A.14, the only difference is that Lemma C.3 needs to be used instead of Lemma A.4. □

D Proof of Theorem 5

I present an argument to establish Theorem 5 that relies on four lemmas, whose proof can be found immediately after this proof. The first lemma establishes that the output of Algorithm 3 is well defined.

Lemma D.1. *The linear program* (LP 1) *in Algorithm 3 has a unique solution.*

Lemma D.1 ensures that each Round $i \geq 1$ of the SRLP algorithm, δ^i, x^i, and d^i are well defined. The next step is to show that, unlike the SR algorithm, the SRLP algorithm eventually terminates.

Lemma D.2. *The SRLP algorithm ends after fewer than $4|A||C|$ rounds.*

Lemma D.2 guarantees that the SRLP algorithm produces an allocation in finitely many rounds. Letting $N < 4|A||C|$ be the number of rounds after which the SRLP algorithm ends, the **outcome of the SRLP algorithm** is then the allocation x^N. (By construction, x^N must be an allocation, otherwise the SRLP algorithm would not end in Round N.) The next result ensures that the outcome of the SRLP algorithm satisfies all four axioms.

Lemma D.3. *In every Round i of the SRLP algorithm, x^i satisfies Axioms 1-4.*

Lemmas D.1-D.3 imply that, after $N < 4|A||C|$ rounds, the SRLP algorithm produces an allocation x^N that satisfies Axioms 1-4. Then, by Theorem 3, the outcome of the SRLP algorithm is SR equivalent. That is, the outcomes of the SR and SRLP algorithms yield the same aggregate allocation. However, there may be multiple allocations satisfying Axioms 1-4 so the last step is to show that x^N is indeed the SR allocation.

Lemma D.4. $x^N = \xi^{SR}$.

Combining Lemmas D.2 and D.4 completes the proof. $\square$

Proof of Lemma D.1

I first show that (LP 1) has a solution and then proceed to showing that there cannot be multiple solutions. For the first part of the proof, I show that the previous round allocation ξ^{i-1} satisfies all $2|\widetilde{C}|$ constraints, which guarantees that the (LP 1) has a solution. That is, I show that, for every $c \in \widetilde{C}$,

$$\xi^{i-1}_{a(c),c} \leq \frac{1 - \sum_{c' \in C_M(a(c)) \setminus \{c\}} \xi^{i-1}_{a(c),c'}}{|C_Q(a(c))| + 1} \tag{30}$$

$$\text{and} \quad \xi^{i-1}_{a(c),c} \leq \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi^{i-1}_{a,c'}}{|C_Q(a)|}. \tag{31}$$

Arbitrarily fix an agent $a \in \widetilde{A}$. By construction, a is qualified for at least one category in Round $i-1$ so, by Lemma A.6, $\xi^{i-1}_a = 1$. It follows that $\sum_{c \in C_Q(a)} \xi^{i-1}_{a,c} + \sum_{c \in C_M(a)} \xi^{i-1}_{a,c} = 1$. As $\xi^{i-1}_{a,c} = d_a$ for every $c \in C_Q(a)$ by definition, we have $d_a = (1 - \sum_{c \in C_M(a)} \xi_{a,c})/|C_Q(a)|$. Moreover, again by definition, we have $\xi^{i-1}_{a,c} \leq d_a$ for every $c \in C_M(a)$. As a was chosen arbitrarily, it follows that

$$\xi^{i-1}_{a,c} \leq \frac{1 - \sum_{c \in C_M(a)} \xi^{i-1}_{a,c}}{|C_Q(a)|} \quad \text{for every } a \in \widetilde{A} \text{ and every } c \in C_M(a),$$

which by construction is equivalent to

$$\xi^{i-1}_{a(c),c} \leq \frac{1 - \sum_{c' \in C_M(a(c))} \xi^{i-1}_{a(c),c'}}{|C_Q(a(c))|} \quad \text{for every } c \in \widetilde{C}.$$

Through simple algebra (as $c \in C_M(a(c))$, $\xi^{i-1}_{a(c),c}$ can be moved out of the sum and to the left-hand side), it follows that the last inequality is equivalent to (30).

To show that (31) holds, arbitrarily fix a category $c \in \widetilde{C}$. As ξ^{i-1} is an allocation (by Lemma A.3), we have $\xi_{a(c),c} + \sum_{a \in A_Q(c)} \xi^{i-1}_{a,c} \leq q_c$. By definition, $\xi^{i-1}_{a,c} = d_a$ for every $a \in A_Q(c)$; hence it follows that $\xi_{a(c),c} + \sum_{a \in A_Q(c)} d_a \leq q_c$, which by definition is equivalent to $\xi_{a(c),c} + \sum_{a \in \widetilde{A}_Q(c)} d_a \leq \widetilde{q}_c$. Moreover, for every $a \in A_Q(c)$, we have $|C_Q(a)| d_a + \sum_{c' \in C_M(a)} \xi^{i-1}_{a,c} =$

1 so $d_a = (1 - \sum_{c' \in C_M(a)} \xi^{i-1}_{a,c'})/|C_Q(a)|$. It follows that

$$\xi^{i-1}_{a(c),c} + \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi^{i-1}_{a,c'}}{|C_Q(a)|} \leq \widetilde{q}_c,$$

which is equivalent to (31).

Having established that the linear program in Algorithm 3 has a solution, I now show that the solution is unique. I being by introducing some notation that will be useful throughout the proof. Given a vector $(\xi_{a(c),c})_{c \in \widetilde{C}}$, for every agent $a \in \widetilde{A}$ let $S_a = \sum_{c \in C_M(a)} \xi_{a,c}$. Arbitrarily fix an agent $a \in \widetilde{A}$ and a vector $S_{-a} = (S_{a'})_{a' \in \widetilde{A} \setminus \{a\}}$. For every $c \in C_M(a)$, let

$$\theta_{a,c} = \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi_{a,c'}}{|C_Q(a)|} = \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - S_{a'}}{|C_Q(a)|}. \tag{32}$$

Note that $\theta_{a,c}$ is the right-hand side of the second constraint of the linear program (LP 3) and is fixed by S_{-a}. Label the categories for which a is marginal such that $C_M(a) = \{c_1, c_2, \ldots, c_{|C_M(a)|}\}$ with $\theta_{a,c_1} \geq \theta_{a,c_2} \geq \ldots \geq \theta_{a,c_{|C_M(a)|}}$. For every $i = 1, \ldots, |C_M(a)|$, let

$$T_i = \frac{1 - \sum_{j>i} \theta_{a,c_j}}{|C_Q(a)| + i}. \tag{33}$$

Finally, define the number $n = 0, 1, \ldots, |C_Q(a)|$ as follows. If $T_i > \theta_{a,c_i}$ for every $i = 1, \ldots, |C_M(a)|$, then $n = 0$. Otherwise, n is the largest number $i = 1, \ldots, |C_M(a)|$ such that $T_i \leq \theta_{a,c_i}$; that is, $T_n \leq \theta_{a,c_n}$ and $T_i > \theta_{a,c_i}$ for all $i > n$. Having introduced the required notation, I next introduce the first intermediate result.

Claim 3. *For every $i \geq n$, $T_i > T_{i+1}$.*

Proof. It needs to be shown that

$$\frac{1 - \sum_{j>i} \theta_{a,c_j}}{|C_Q(a)| + i} > \frac{1 - \sum_{j>i+1} \theta_{a,c_j}}{|C_Q(a)| + i + 1},$$

which is equivalent to

$$
\begin{aligned}
|C_Q(a)| + i + 1 - (|C_Q(a)| + i + 1) \sum_{j>i} \theta_{a,c_j} &> |C_Q(a)| + i - (|C_Q(a)| + i) \sum_{j>i+1} \theta_{a,c_j} \\
\Leftrightarrow \quad 1 + (|C_Q(a)| + i) \sum_{j>i+1} \theta_{a,c_j} &> (|C_Q(a)| + i + 1) \sum_{j>i} \theta_{a,c_j} \\
\Leftrightarrow \quad 1 &> (|C_Q(a)| + i + 1)\theta_{a,c_{i+1}} + \sum_{j>i+1} \theta_{a,c_j} \\
\Leftrightarrow \quad \theta_{a,c_{i+1}} &< \frac{1 - \sum_{j>i+1} \theta_{a,c_j}}{(|C_Q(a)| + i + 1)}.
\end{aligned}
$$

The right-hand side of the last inequality is equal to T_{i+1}; therefore, $T_i > T_{i+1}$ is equivalent to $\theta_{a,c_{i+1}} < T_{i+1}$, which is satisfied by the definition of n since, by assumption, $i + 1 > n$. □

Next, arbitrarily fix a vector $(y_i)_{i>n}$ such that $y_i \leq \theta_{a,c_i}$ for every $i > n$ and consider the following linear program:

$$
\begin{aligned}
&\max_{(\xi_{a,c_i})_{i=1}^{|C_M(a)|}} \sum_{i=1}^{|C_M(a)|} \xi_{a,c_i} \\
\text{subject to (i)} \quad &\xi_{a,c_i} \leq \frac{1 - \sum_{j \neq i} \xi_{a,c_j}}{|C_Q(a)| + 1} \quad \text{for every } i = 1, \ldots, |C_M(a)| \\
\text{and (ii)} \quad &\xi_{a,c_i} = y_i \quad \text{for every } i > n.
\end{aligned}
\tag{LP 2}
$$

The linear program (LP 2) can be interpreted as follows. For every $i > n$, ξ_{a,c_i} is set to y_i so only the first n elements (ξ_{a,c_i} for $i \leq n$) have to be chosen to maximized the sum, subject to constraint (i).

Claim 4. *For any vector $(y_i)_{i>n} \leq (\theta_{a,c_i})_{i>n}$, the unique solution to the linear program* (LP 2) *is the vector $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ such that*

$$
\xi^*_{a,c_i} = \begin{cases} \frac{1 - \sum_{j>n} y_j}{|C_Q(a)| + n} & \text{if } i \leq n \\ y_i & \text{if } i > n. \end{cases}
$$

Proof. I first show that $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ satisfies all constraints. Constraint (ii) is satisfied for all $i > n$ by definition; hence I focus on constraint (i).

Case 1: $i \leq n$. It needs to be shown that

$$
\frac{1 - \sum_{j>n} y_j}{|C_Q(a)| + n} \leq \frac{1 - \sum_{j \neq i} \xi^*_{a,c_j}}{|C_Q(a)| + 1}. \tag{34}
$$

By definition, we have

$$1-\sum_{j\neq i}\xi^*_{a,c_j}=1-(n-1)\frac{1-\sum_{j>n}y_j}{|C_Q(a)|+n}-\sum_{j>n}y_j=\frac{(|C_Q(a)|+1)(1-\sum_{j>n}y_j)}{|C_Q(a)|+n}.$$

Therefore, the right-hand side of (34) is equal to $(1-\sum_{j>n}y_j)/(|C_Q(a)|+n)$ and (34) holds with an equality.

Case 2: $i>n$. It needs to be shown that

$$\xi^*_{a,c_i}\leq\frac{1-\sum_{j\neq i}\xi^*_{a,c_j}}{|C_Q(a)|+1},$$

which is equivalent to

$$|C_Q(a)|\xi^*_{a,c_i}\leq 1-\sum_{j=1}^{|C_M(a)|}\xi^*_{a,c_j}. \tag{35}$$

By the definition of $(\xi_{a,c_i})_{i=1}^{|C_M(a)|}$, (35) is equivalent to

$$\begin{aligned}|C_Q(a)|y_i&\leq 1-n\frac{1-\sum_{j>n}y_j}{|C_Q(a)|+n}-\sum_{j>n}y_j\\ y_i&\leq\frac{1-\sum_{j>n}y_j}{|C_Q(a)|+n}\end{aligned} \tag{36}$$

As $y_i\leq\theta_{a,c_i}$ by definition, $\theta_{a,c_i}<T_i$ by the definition of n, $T_i<T_n$ by Claim 3, and T_n is equal to the right-hand side of (35), it can be concluded that (35) holds.

Having shown that the vector $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ satisfies all constraints, I proceed to show that it maximizes the objective, which makes it a solution to (LP 2). Consider any vector $(\xi_{a,c_i})_{i=1}^{|C_M(a)|}$ that satisfies constraints (i) and (ii); I show that $\sum_{i=1}^{C_M(A)}\xi_{a,c_i}\leq\sum_{i=1}^{C_M(A)}\xi^*_{a,c_i}$. Constraint (i) implies that, for every $i\leq n$,

$$|C_Q(a)|\xi_{a,c_i}\leq 1-\sum_{j=1}^{|C_M(a)|}\xi_{a,c_j}.$$

Summing up over all $i \leq n$ yields

$$\begin{aligned}
\sum_{i\leq n} |C_Q(a)|\xi_{a,c_i} &\leq \sum_{i\leq n}(1 - \sum_{j=1}^{|C_M(a)|} \xi_{a,c_j}) \\
\Leftrightarrow \quad |C_Q(a)| \sum_{i\leq n} \xi_{a,c_i} &\leq n - n \sum_{i=1}^{|C_M(a)|} \xi_{a,c_i} \\
\Leftrightarrow \quad (|C_Q(a)| + n) \sum_{i\leq n} \xi_{a,c_i} &\leq n - n \sum_{i>n} \xi_{a,c_i} \\
\Leftrightarrow \quad \sum_{i\leq n} \xi_{a,c_i} &\leq n \frac{1 - \sum_{i>n} \xi_{a,c_i}}{|C_Q(a)| + n} \\
\Leftrightarrow \quad \sum_{i=1}^{|C_M(a)|} \xi_{a,c_i} &\leq n \frac{1 - \sum_{i>n} \xi_{a,c_i}}{|C_Q(a)| + n} + \sum_{i>n} \xi_{a,c_i}.
\end{aligned}$$

As constraint (ii) holds for every $i > n$, we obtain

$$\sum_{i=1}^{|C_M(a)|} \xi_{a,c_i} \leq n \frac{1 - \sum_{i>n} y_i}{|C_Q(a)| + n} + \sum_{i>n} y_i. \tag{37}$$

By definition, we have

$$\sum_{i=1}^{|C_M(a)|} \xi^*_{a,c_i} = n \frac{1 - \sum_{j>n} y_j}{|C_Q(a)| + n} + \sum_{j>n} y_j,$$

which combined with (37) implies that $\sum_{i=1}^{C_M(A)} \xi_{a,c_i} \leq \sum_{i=1}^{C_M(A)} \xi^*_{a,c_i}$.

Having shown that is a solution to the linear program (LP 2), I finally show that it is the unique solution. Let $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)}$ be a solution to (LP 2), it needs to be shown that $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)} = (\xi^*_{a,c_i})_{i=1}^{|C_M(a)}$. As $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)}$ is a solution to (LP 2), it maximizes the objective so

$$\sum_{i=1}^{|C_M(a)|} \xi^\sharp_{a,c_i} = \sum_{i=1}^{|C_M(a)|} \xi^*_{a,c_i} = n \frac{1 - \sum_{j>n} y_j}{|C_Q(a)| + n} + \sum_{j>n} y_j.$$

Moreover, as $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)}$ satisfies constraint (i), for every $i \leq n$, we have

$$|C_Q(a)|\xi^\sharp_{a,c_i} \leq 1 - \sum_{j=1}^{|C_M(a)|} \xi^\sharp_{a,c_j}.$$

Therefore, it follows that, for every $i \leq n$

$$|C_Q(a)|\xi^{\sharp}_{a,c_i} \leq 1 - n\frac{1-\sum_{j>n} y_j}{|C_Q(a)|+n} - \sum_{j>n} y_j$$
$$\xi^{\sharp}_{a,c_i} \leq \frac{1-\sum_{j>n} y_j}{|C_Q(a)|+n}.$$

Then, by definition, it follows that $\xi^{\sharp}_{a,c_i} \leq \xi^{*}_{a,c_i}$ for every $i \leq n$. As $\xi^{\sharp}_{a,c_i} = \xi^{*}_{a,c_i}$ for all $i > n$ (by constraint (ii)) and $\sum_{i=1}^{|C_M(a)|} \xi^{\sharp}_{a,c_i} = \sum_{i=1}^{|C_M(a)|} \xi^{*}_{a,c_i}$ (as both vectors maximize the objective), we have $(\xi^{\sharp}_{a,c_i})_{i=1}^{|C_M(a)} = (\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$. $\square$

I next use Claim 4 to find the solution to the following linear program.

$$\begin{aligned} &\max_{(\xi_{a,c_i})_{i=1}^{|C_M(a)|}} \sum_{i=1}^{|C_M(a)|} \xi_{a,c_i} \\ \text{subject to (i)} \quad &\xi_{a,c_i} \leq \frac{1-\sum_{j\neq i}\xi_{a,c_j}}{|C_Q(a)|+1} \\ \text{and (ii)} \quad &\xi_{a,c_i} \leq \theta_{a,c_i} \qquad\qquad \text{for every } i = 1,\ldots,|C_M(a)|. \end{aligned} \tag{LP 3}$$

The linear program (LP 3) can be thought of as the linear program (LP 1) from Algorithm 3 in which S_{-a} has been fixed so it remains to choose the vector $(\xi_{a,c_i})_{i=1}^{|C_M(a)|}$ to maximize S_a.

Claim 5. *The unique solution to the linear program* (LP 3) *is the vector* $(\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$ *such that, for every* $i = 1,\ldots,C_M(a)$,

$$\xi^{*}_{a,c_i} = \begin{cases} T_n & \text{if } i \leq n \\ \theta_{a,c_i} & \text{if } i > n. \end{cases}$$

Proof. By Claim 4, $(\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$ satisfies constraint (i); otherwise the unique solution to (LP 2) would not satisfy that linear program's constraints. By definition, for every $i \leq n$, $\xi^{*}_{a,c_i} = T_n \leq \theta_{a,c_i}$ and, for every $i > n$, $\xi^{*}_{a,c_i} = \theta_{a,c_i}$; hence $(\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$ satisfies constraint (ii).

Having shown that $(\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$ satisfies all constraints (which implies that (LP 3) has a solution), I now show that it is the unique solution to (LP 3). Let $(\xi^{\sharp}_{a,c_i})_{i=1}^{|C_M(a)}$ be a solution to (LP 3), I show that $(\xi^{\sharp}_{a,c_i})_{i=1}^{|C_M(a)} = (\xi^{*}_{a,c_i})_{i=1}^{|C_M(a)}$. By constraint (ii), $\xi^{\sharp}_{a,c_i}\theta_{a,c_i}$ for every $i > n$; therefore Claim 4 implies that, for every $i \leq n$,

$$\xi^{\sharp}_{a,c_i} = \frac{1-\sum_{j>n}\xi^{\sharp}_{a,c_j}}{|C_Q(a)|+n}, \tag{38}$$

as otherwise $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)}$ would not be optimal. Then,

$$\sum_{i=1}^{|C_M(a)|} \xi^\sharp_{a,c_i} = n\frac{1-\sum_{i>n}\xi^\sharp_{a,c_i}}{|C_Q(a)|+n} + \sum_{i>n}\xi^\sharp_{a,c_j} = \frac{n+|C_Q(a)|\sum_{i>n}\xi^\sharp_{a,c_i}}{|C_Q(a)|+n}$$

so the objective is increasing in $(\xi^\sharp_{a,c_i})_{i>n}$; therefore, the unique maximizer is obtained by setting $\xi^\sharp_{a,c_i} = \theta_{a,c_i}$ for every $i > n$, which by (38) implies that $\xi^\sharp_{a,c_i} = T_n$ for every $i \leq n$. It follows that $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|} = (\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$. $\square$

I finally go back to the linear program (LP 1) in Algorithm 3 and use Claim 5 to show that (LP 1) has a unique solution. For any agent $a \in \widetilde{A}$ and any S_{-a}, let $S_a(S_{-a})$ be the maximized objective function of the linear program (LP 3); in words, $S_a(S_{-a})$ is the largest sum that can be reached for the elements involving agent a given S_{-a}. If S_{-a} increases, then by (32), so does $\theta_{a,c}$ for every $c \in C_M(a)$. Therefore, constraint (ii) of the linear program (LP 3) is relaxed, meaning that the largest sum that can be reached increases as well. It follows that $S_a(S_{-a})$ is increasing in S_{-a}. Suppose toward a contradiction that (LP 1) has two solutions giving two distinct sum vectors $S^* = (S^*_a)_{a\in\widetilde{A}}$ and $S^\sharp = (S^\sharp_a)_{a\in\widetilde{A}}$. Then, for every $a \in \widetilde{A}$, $S^*_a = S_a(S^*_{-a})$ and $S^\sharp_a = S_a(S^\sharp_{-a})$. Consider the sum vector $\overline{S} = (\overline{S}_a)_{a\in\widetilde{A}}$ with $\overline{S}_a = \max\{S^*_a, S^\sharp_a\}$ for every $a \in \widetilde{A}$. As S^* and $S^\sharp$ are distinct and derive from solutions of (LP 1), it must be that $\sum_{a\in\widetilde{A}}\overline{S}_a > \sum_{a\in\widetilde{A}} S^*_a = \sum_{a\in\widetilde{A}} S^\sharp_a$; hence the allocation underpinning $\overline{S}$ must violate some constraint of (LP 1). Consequently, there exists an agent $a \in \widetilde{A}$ such that $\overline{S}_a > S_a(\overline{S}_{-a})$. By definition, $\overline{S}_{-a} \geq S^*_{-a}$; hence, as $S_a(S_{-a})$ is increasing in S_{-a}, it follows that $S_a(\overline{S}_{-a}) \geq S_a(S^*_{-a})$. As $S^*_a = S_a(S^*_a)$, it can be concluded that $\overline{S}_a > S^*_a$. Analogous reasoning yields that $\overline{S}_a > S^\sharp_a$ so $\overline{S}_a > \max\{S^*_a, S^\sharp_a\}$, a contradiction.

The preceding reasoning implies that every solution to (LP 1) yields the same sum vector, which I denote by S^*. By Claim 5, for every $a \in \widetilde{A}$, there exists a unique vector $(\xi^*_{a,c})_{c\in C_M(a)}$ such that $\sum_{c\in C_M(a)}\xi^*_{a,c} = S^*_a = S_a(S^*_{-a})$. Therefore, the vector $(\xi^*_{a,c})_{a\in\widetilde{A},c\in C_M(a)} = (\xi^*_{a(c),c})_{c\in\widetilde{C}}$ is the unique solution to (LP 1). $\square$

Proof of Lemma D.2

Consider any Round i, any agent a and any category c. Suppose that a is qualified for c in Round i, that is $x^i_{a,c} \geq d^i_a$. By Lemma C.7, for every $j > i$, $x^j_{a,c} \geq d^j_a$; therefore, a remains qualified for c in every subsequent round. Suppose next that a is marginal for c in Round i, that is $0 < x^i_{a,c} < d^i_a$. By Lemma C.6, in any Round $j > i$, either $x^j_{a,c} \geq x^i_{a,c}$ or $x^j_{a,c} = d^j_a$. In both cases, $x^j_{a,c} > 0$; therefore, a is either marginal or qualified for c in every subsequent round. It follows that throughout the SRLP algorithm, for every agent-object pair (a, c), a's status for c changes at most twice: once from unqualified to marginal and once from marginal to qualified.

Consider next a Round i in which the SRLP algorithm uses linear programming, i.e., $\delta^i = \delta^{LP}(x^{i-1}, d^{i-1})$. I show that either the algorithm ends in Round i or there exists an agent-object pair (a, c) such that a's status for c changes in Round i. If $x^i_a \leq 1$ for every agent a, the algorithm ends in Round i so the remainder of the argument focuses on the case in which $x^i_a > 1$ for some agent a. Suppose first that a was not qualified for any category in Round $i-1$. The assumption that $x^i_a > 1$ implies by definition that $\xi^i_a = 1$ and by Lemma C.2 that $\xi^i_{a,c} = d^i_a$ for some category c. It follows that a's status for c has changed from either unqualified or marginal to qualified in Round i. Suppose next that a was qualified for some categories but not qualified for any. Then, by assumption, $\xi^{i-1}_{a,c} = d^{i-1}_a$ for all $c \in C_Q(a)$ and $\xi^{i-1}_{a,c} = 0$ for all $c \in C \setminus C_Q(a)$. By Lemma C.2, $\xi^{i-1}_a = 1$ so $|C^{i-1}_Q(a)| d^{i-1}_a = 1$. As a is not marginal for any category, $a \notin \widetilde{A}$ in the LP algorithm; hence $\delta^i = d^{i-1}$. It follows that $|C^{i-1}_Q(a)| \delta^i_a = 1$; moreover, by definition $x^i_{a,c} \leq \delta^i_a$ for every category c so $\sum_{c \in C^{i-1}_Q(a)} x^i_{a,c} \leq 1$. Then, the assumption that $x^i_a > 1$ implies that there is a category c such that $x^i_{a,c} > x^{i-1}_{a,c} = 0$ so a's status for c has changed in Round i from unqualified to either marginal or qualified. Last, consider the remaining case in which a is qualified for at least one category and marginal for at least one category. In that case, $a \in \widetilde{A}$ in the LP algorithm so $\delta^i = \delta^{LP}(x^{i-1}, d^{i-1})$. Let $(\xi_{a(c),c})_{c \in \widetilde{C}}$ be the solution to the linear program (LP 1) in the LP algorithm. By construction, for every $c \in C_M(a)$, $\xi^*_{a,c} = \min\{\delta^i_a, \widetilde{q}_c - \sum_{a' \in \widetilde{A}_Q(c)} \delta^i_{a'}\}$; otherwise, one constraint in (LP 1) would hold with a strict inequality and $(\xi_{a(c),c})_{c \in \widetilde{C}}$ would not be the solution to (LP 1). If $\xi^*_{a,c} = \delta^i_a$, then $x^i_{a,c} \leq \xi^*_{a,c}$ since $x^i_{a,c} \leq \delta^i_a$ by definition. If $\xi^*_{a,c} = \widetilde{q}_c - \sum_{a' \in \widetilde{A}_Q(c)} \delta^i_{a'}$, then, as $\widetilde{q}_c = q_c - \sum_{a' \in A_Q(c) \setminus \widetilde{A}_Q(c)} d^{i-1}_{a'}$ and $\delta^i_{a'} = d^{i-1}_{a'}$ for every $a' \in A_Q(c) \setminus \widetilde{A}_Q(c)$, we have $\xi^*_{a,c} = q_c - \sum_{a' \in A_Q(c)} \delta^i_{a'}$. Moreover, as a is marginal for c, $A_Q(c) = \hat{A}_{a,c}$ so $\xi^*_{a,c} = q_c - \sum_{a' \in \hat{A}_{a,c}} \delta^i_{a'}$. Then, for every agent $a' \in \hat{A}_{a,c}$, $\delta^i_{a'} + \sum_{\tilde{a} \in \hat{A}_{a',c}} \delta_{\tilde{a}} < q_c$ so by definition $x^i_{a',c} = \delta^i_{a'}$ for every $a' \in \hat{A}_{a,c}$. It follows that $\xi^*_{a,c} = q_c - \sum_{a' \in \hat{A}_{a,c}} \xi^i_{a',c}$ or, equivalently, $\xi^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^i_{a',c} = q_c$. As $x_{a,c}$ is a preallocation, $x^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^i_{a',c} \leq q_c$ so it can be concluded that $\xi^*_{a,c} \leq \xi^*_{a,c}$. By construction, $|C_Q(a)| \delta^i_a + \sum_{c \in C_M(a)} \xi^*_{a,c} = 1$; therefore the fact that $x^i_{a,c} \leq \xi^*_{a,c}$ for every $c \in C_M(a)$ implies that $\sum_{c \in C_Q(a) \cup C_M(a)} x^i_{a,c} \leq 1$. Then, the assumption that $x^i_a > 1$ implies that $x^i_{a,c} > 0$ for some $c \in C_U(a)$ so a's status for c changes in Round i from unqualified to either marginal or qualified.

Having established that the status of each agent for each category changes at most twice throughout the SRLP algorithm and that in every round in which the SRLP algorithm uses linear programming the status of at least one agent for at least one category change, I am now in a position to prove that the SRLP algorithm ends after fewer than $4|A||C|$ rounds. In fact, I will show that $4|A||C| - 2$ is an upper bound for the number of rounds of the SRLP algorithm. First, suppose that fewer than two status changes occur in Round 1. In that case, no agent is qualified for any category in Round 1, that is $x^1_{a,c} < d^1_a$ for every agent-category pair (a, c). Then, by Lemma C.2, $\xi^1_a < 1$ and $d^1_a = 1$ for every agent a. By definition, it follows that $x^1_a < 1$ for every agent a so the SRLP algorithm ends in Round 1. Therefore, if the SRLP algorithm lasts more than one round, then at least two changes of status occur in Round 1. Suppose that, in some subsequent Round $i > 1$, no change of status occurs. Then, by construction, the SRLP algorithm uses linear programming in Round $i + 1$, which guarantees that a change of status occurs in Round $i + 1$. It follows that at least one change of status occurs every second round. Then, by the end of Round $4|A||C| - 3$, $4|A||C|$ changes of status must have occurred: 2 in Round 1 and $2|A||C| - 2$ in the $4|A||C| - 4$ subsequent rounds. As the status of each agent for each category can change at most twice, it follows that, at the end of Round $4|A||C| - 3$, every agent is qualified for every category. Then, $\xi^{4|A||C|-3}_{a,c} = d^{4|A||C|-3}_a$ for every agent-category pair (a, c) so, by Lemma C.2, $\xi^{4|A||C|-3}_a = 1$ for every agent a. It follows that $d^{4|A||C|-3}_a = 1/|C|$ for every agent a. In Round $4|A||C| - 2$, by Lemma C.7, $x^{4|A||C|-2}_{a,c} = d^{4|A||C|-3}_a = 1/|C|$ for every agent-object pair (a, c). Then, for every agent a, $x^{4|A||C|-2}_a = \sum_{c\in C} 1/|C| = 1$ so the SRLP algorithm ends. □

Proof of Lemma D.3

While not entirely analogous, the reasoning is very similar to the proof of Lemma A.10.

(Complies with eligibility requirements) By definition, if an agent a is not eligible for a category c, then $x^i_{a,c} = 0$.

(nonwasteful) Consider any category c such that $\sum_{a\in A} x^i_{a,c} < q_c$ and any agent a who is eligible for c. It needs to be shown that $x^i_a \geq 1$.

Case 1: $x^i_{a,c} = \delta^i_a$. By the case assumption and Lemma C.3, $x^i_{a,c} = \delta^i_a \geq d^i_a$; hence, by definition, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\} = d^i_a$. By Lemma C.2, it follows that $\xi^i_a = 1$ so, by definition, $x^i_a \geq \xi^i_a = 1$.

Case 2: $x^i_{a,c} < \delta^i_a$. In that case, Lemma C.5 applies and yields $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} = q_c$. Then, we have $\sum_{a\in A} x^i_{a,c} \geq x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} = q_c$, which contradicts the assumption that $\sum_{a\in A} x^i_{a,c} < q_c$.

(Respects priorities) Consider an agent a such that $\xi^i_a < 1$ and arbitrarily fix a category c and a lower-priority agent $a' \in \check{A}_{a,c}$. It needs to be shown that $x^i_{a',c} = 0$. By Lemma C.2, the assumption that $\xi^i_a < 1$ implies that $\xi^i_{a,c} < d^i_a$ so, by Lemma C.4, $\xi^i_{a',c} = 0$.

(Category neutrality) Consider any agent a and any category c such that a is eligible for c and $x^i_{a,c} < \max_{c'\in C}\{x_{a,c'}\}$. It needs to be shown that $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x_{a',c} = q_c$. By definition, $\max_{c'\in C}\{x^i_{a,c'}\} \leq \delta^i_a$; hence we have $x^i_{a,c} < \delta^i_a$. Then, by Lemma C.5, we have $x^i_{a,c} + \sum_{a'\in \hat{A}_{a,c}} x^i_{a',c} = q_c$. □

Proof of Lemma D.4

Consider the allocation ξ^{SR} produced by the SR algorithm and the associated demand vector $d(\xi^{SR})$. By definition, for every agent a, $d_a(\xi^{SR}) = 1$ if $\xi^{SR}_a < 1$ and $d_a(\xi^{SR}) = \max_{c\in C}\{\xi_{a,c}\}$ if $\xi^{SR}_a = 1$. For every category c, it is possible to identify the agents who are *qualified*, *marginal*, and *unqualified* for c at the allocation ξ^{SR}. For every category c, I denote by $A^{SR}_Q(c) = \{a \in A : \xi^{SR}_{a,c} = d_a(\xi^{SR})\}$ the set of agents who are qualified for c at ξ^{SR}, by $A^{SR}_M(c) = \{a \in A : \xi^{SR}_{a,c} \in (0, d_a(\xi^{SR}))\}$ the set of agents who are marginal for c at ξ^{SR}, and by $A^{SR}_U(c) = \{a \in A : \xi^{SR}_{a,c} = 0\}$ the set of agents who are unqualified for a at ξ^{SR}. Similarly, for every agent a, I denote by $C^{SR}_Q(a) = \{c \in C : \xi^{SR}_{a,c} = d_a(\xi^{SR})\}$ the set of categories for which a is qualified, by $C^{SR}_M(a) = \{c \in C : \xi^{SR}_{a,c} \in (0, d_a(\xi^{SR}))\}$ the set of categories for which a is marginal, and by $C^{SR}_U(a) = \{c \in C : \xi^{SR}_{a,c} = 0\}$ the set of categories for which a is unqualified. The next result formalizes the properties of the preceding definitions.

Claim 6. *For every category c, $|A_M(c)| \leq 1$ and, for any two agents a and a', either $a \in A_Q(c)$ and $a' \in A_M(c) \cup A_U(c)$ or $a \in A_M(c)$ and $a' \in A_U(c)$ implies that $a\pi_c a'$.*

Proof. ($|A_M(c)| \leq 1$.) Toward a contradiction, suppose that there exist two distinct agents $a, a' \in A_M(c)$ with $a \neq a'$. By assumption, we have $0 < \xi^{SR}_{a,c} < d_a(\xi^{SR})$ and $0 < \xi^{SR}_{a',c} < d_{a'}(\xi^{SR})$; as ξ^{SR} complies with eligibility requirements, it follows that both a and a' are eligible for c. If $\xi^{SR}_a < 1$, then, as ξ^{SR} respects priorities, $\xi^{SR}_{a',c} = 0$, a contradiction. If $\xi^{SR}_a = 1$, then by definition $\xi^{SR}_{a,c} < d_a(\xi^{SR}) = \max_{c'\in C}\{\xi^{SR}_{a,c'}\}$. As ξ^{SR} is category neutral, it follows that $\xi^{SR}_{a,c} + \sum_{\tilde{a}\in\hat{A}_{a,c}} \xi_{\tilde{a},c} = q_c$ so $\xi_{a',c} = 0$, a contradiction.

($a \in A_Q(c)$ and $a' \in A_M(c) \cup A_U(c)$ implies that $a\pi_c a'$.) By assumption, $a \neq a'$ and $\xi^{SR}_{a,c} = d_a(\xi^{SR}) > 0$. Toward a contradiction, suppose that $a\pi_c a'$. If a' is not eligible for c, then neither is a since $a'\pi a$; hence, as ξ^{SR} complies with eligibility requirements, $\xi^{SR}_{a,c} = 0$, a contradiction. For the remainder of the argument, I assume that a' is eligible for c. If $\xi^{SR}_{a'} < 1$, then, as ξ^{SR} respects priorities, the assumption that $a'\pi_c a$ implies that $\xi^{SR}_{a,c} = 0$, a contradiction. If $\xi^{SR}_{a'} = 1$, then by definition $d^{SR}_{a'} = \min_{c'\in C}\{\xi^{SR}_{a',c'}\}$; as $a' \in A_M(c) \cup A_U(c)$, $\xi^{SR}_{a',c} < d^{SR}_{a'}$ so it follows that $\xi^{SR}_{a',c} < \min_{c'\in C}\{\xi^{SR}_{a',c'}\}$. As ξ^{SR} is category neutral, we have $\xi^{SR}_{a',c} + \sum_{\tilde{a}\in\hat{A}_{a',c}} \xi_{\tilde{a},c} = q_c$ so $\xi^{SR}_{a,c} = 0$, a contradiction.

($a \in A_M(c)$ and $a' \in A_U(c)$ implies that $a\pi_c a'$.) The reasoning is almost analogous to that of the preceding argument. By assumption, we have $a \neq a'$ and $\xi^{SR}_{a,c} > 0$. Toward a contradiction, suppose that $a\pi_c a'$. If a' is not eligible for c, neither is a so $\xi^{SR}_{a,c} = 0$. Otherwise, if $\xi^{SR}_{a'} < 1$, we have $\xi^{SR}_{a,c} = 0$ since ξ^{SR} respects priorities and if $\xi^{SR}_{a'} < 1$, we have $\xi^{SR}_{a,c} = 0$ since ξ^{SR} is category neutral. Therefore, in all cases, $\xi^{SR}_{a,c} = 0$, a contradiction. $\square$

Next, I construct an alternative rationing problem $\overline{R} = (A, C, (\overline{\pi}_c))c \in C, (q_c)_{c\in C})$ that is identical to the original rationing problem R except that every agent a who is unqualified for a category c at the SR allocation is not eligible for c in $\overline{R}$ (whether or not a is eligible for c in R). That is, for every category c, $\overline{\pi}_c$ is constructed as follows: for any two agents a and a', $a\overline{\pi}_c a'$ if and only if $a\pi_c a'$ and for every agent a, $a\overline{\pi}_c\emptyset$ if and only if $\xi^{SR}_{a,c} > 0$. In the original rationing problem R, the SR algorithm produces the allocation ξ^{SR} and the SRLP algorithm terminates after N rounds and produces the allocation x^N. In the alternative rationing problem $\overline{R}$, I denote by $\overline{x}^{SR}_{a,c}$ the allocation produced by the SR algorithm, by $\overline{N}$ the number of rounds after which the SRLP algorithm ends, and by $\overline{x}^{\overline{N}}$ the allocation produced by the SRLP algorithm. To prove that $\xi^{SR} = x^N$, I show successively that $\xi^{SR} = \overline{\xi}^{SR}$, $\overline{\xi}^{SR} = \overline{x}^{\overline{N}}$, and $\overline{x}^{\overline{N}} = x^N$.

($\xi^{SR} = \overline{\xi}^{SR}$) For the original rationing problem R, I denote by x^i, ξ^i, and d^i the preallocation, the allocation, and the demand vector calculated by the SR algorithm in any given Round i. Similarly, for the alternative rationing problem $\overline{R}$, I denote by $\overline{x}^i$, $\overline{\xi}^j$, and $\overline{d}^i$ the preallocation, the allocation, and the demand vector calculated by the SR algorithm in any given Round i. I also denote by d^0 and $\overline{d}^0$ the initial demand vectors in R and $\overline{R}$, respectively.

By definition, we have $d^0 = \overline{d}^0 = \mathbf{1}$. Consider any Round $i \geq 1$ of the SR algorithm and suppose, toward an inductive argument, that $d^{i-1} = \overline{d}^{i-1}$. I show that $x^i = \overline{x}^i$ and $d^i = \overline{d}^i$. Arbitrarily fix an agent a and a category c. I first show that $x^i_{a,c} = \overline{x}^i_{a,c}$, considering separately the cases in which $\xi^{SR}_{a,c} > 0$ and $\xi^{SR}_{a,c} = 0$.

Case 1: $\xi^{SR}_{a,c} > 0$. As ξ^{SR} complies with eligibility requirements, the case assumption implies that a is eligible for c in the rationing problem R. By definition, the case assumption also implies that a is eligible for c in the alternative rationing problem $\overline{R}$. Again by definition, it follows that $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a'\in\hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$ and $\overline{x}^i_{a,c} = \min\{\overline{d}^{i-1}_a, \max\{q_c - \sum_{a'\in\hat{A}_{a,c}} \overline{d}^{i-1}_{a'}, 0\}\}$ so the induction hypothesis that $d^{i-1} = \overline{d}^{i-1}$ implies that $x^i_{a,c} = \overline{x}^i_{a,c}$. (Note that, by definition, the priority among agents is the same in both rationing problems so $\hat{A}_{a,c}$ can be used to calculate both $x^i_{a,c}$ and $\overline{x}^i_{a,c}$.)

Case 2: $\xi^{SR}_{a,c} = 0$. By definition, the case assumption implies that a is not eligible for c in the original rationing problem $\overline{R}$; hence, $\overline{x}^i_{a,c} = 0$ and it remains to show that $x^i_{a,c} = 0$. If a is not eligible either in the alternative rationing problem R, it follows by definition that $x^i_{a,c} = 0$; therefore, for the remainder of the argument, I assume that a is eligible for c in R.

If $\xi^{SR}_a < 1$, as a is eligible for c and ξ^{SR} is nonwasteful, we have $\sum_{a'\in A} \xi^{SR}_{a,c} = q_c$. Moreover, as ξ^{SR} respects priorities, $\xi^{SR}_{a',c} = 0$ for every $a' \in \check{A}_{a,c}$ and, by the case assumption, $\xi^{SR}_{a,c} = 0$. It follows that $\sum_{a'\in\hat{A}_{a,c}} \xi^{SR}_{a',c} = q_c$. If $\xi^{SR}_a = 1$, then $\xi^{SR}_{a,c} + \sum_{a'\in\hat{A}_{a,c}} \xi^{SR}_{a',c} = q_c$ as ξ^{SR} is category neutral so the case assumption implies that $\sum_{a'\in\hat{A}_{a,c}} \xi^{SR}_{a',c} = q_c$. Therefore, it has been established that

$$\sum_{a'\in\hat{A}_{a,c}} \xi^{SR}_{a',c} = q_c. \tag{39}$$

Consider any agent $a' \in \hat{A}_{a,c}$. By Lemma A.5, for every Round $j \geq 1$ of the SR algorithm, we have $d^j_{a'} \geq \max_{c' \in C}\{\xi_{a,c'}\}$, which implies that $d^j_{a'} \geq \xi^j_{a',c}$. By Lemma A.4, it follows that, for every $j \geq 1$, $d^j_{a'} \geq \lim_{j\to\infty} \xi^j_{a',c}$. Therefore, by Corollary 1, we have $d^j_{a'} \geq \xi^{SR}_{a',c}$ for every $j \geq 1$, which implies that $d^{i-1}_{a'} \geq \xi^{SR}_{a',c}$. As the last inequality holds for every $a' \in \hat{A}_{a,c}$, (39) implies that $\sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'} \geq q_c$. As a is eligible for c, it can then be concluded that $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\} = 0$.

As a and c were chosen arbitrarily, it has been established that $x^i_{a,c} = \overline{x}^i_{a,c}$ for every agent a and every category c; hence we have $x^i = \overline{x}^i$. Then, by construction, it follows that $d^i = \overline{d}^i$. By induction, it can then be concluded that $x^i = \overline{x}^i$ for every $i \geq 1$. Therefore, by Corollary 1, we have $\xi^{SR} = \lim_{i\to\infty} x^i = \lim_{i\to\infty} \overline{x}^i = \overline{\xi}^{SR}$.

($\overline{\xi}^{SR} = \overline{x}^N$) As each of $\overline{\xi}^{SR}$ and $\overline{x}^N$ is an allocation of the alternative rationing problem $\overline{R}$ that satisfies Axioms 1-4, it is sufficient to show that $\overline{\xi}^{SR}$ is the unique allocation of $\overline{R}$ that satisfies Axioms 1-4. Let $\overline{\xi}^*$ be an allocation of $\overline{R}$ that satisfies Axioms 1-4. I show that $\overline{\xi}^* = \overline{\xi}^{SR}$.

First, by Theorem 3, $\overline{\xi}^*$ and $\overline{\xi}^{SR}$ generate the same aggregate allocation; moreover, by Theorem 4, the demand vector associated with $\overline{\xi}^*$ is weakly smaller than the one associated with ξ^{SR}. It follows that

$$d_a(\overline{\xi}^*) \leq d_a(\overline{\xi}^{SR}) \quad \text{and} \quad \overline{\xi}^*_a = \overline{\xi}^{SR}_a \quad \text{for every } a \in A. \tag{40}$$

Consider any agent-object pair (a,c) such that a is not qualified for c at $\overline{\xi}^{SR}$, i.e., $\overline{\xi}^{SR}_{a,c} = 0$. As $\overline{\xi}^{SR} = \xi^{SR}$, a is not qualified for c at ξ^{SR}; hence, by definition, a is not eligible for c in the alternative rationing problem $\overline{R}$. As $\overline{\xi}^*$ complies with eligibility requirements, it follows that $\xi^*_{a,c} = 0$ so we have

$$\overline{\xi}^*_{a,c} = 0 \quad \text{for every } (a,c) \in A \times C \text{ such that } \overline{\xi}^{SR}_{a,c} = 0. \tag{41}$$

Consider next any agent-object pair (a,c) such that a is qualified for c at $\overline{\xi}^{SR}$, i.e., $\overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR})$. By definition, $\overline{\xi}^*_{a,c} \leq d_a(\overline{\xi}^*)$ and, by (40), $d_a(\overline{\xi}^*) \leq d_a(\overline{\xi}^{SR})$; therefore, we have

$$\overline{\xi}^*_{a,c} \leq d_a(\overline{\xi}^*) \leq d_a(\overline{\xi}^{SR}) = \overline{\xi}^{SR}_{a,c}. \tag{42}$$

Consider any agent $a' \in \hat{A}_{a,c}$ and suppose, toward a contradiction, that $\overline{\xi}^{SR}_{a',c} < d_{a'}(\xi^{SR})$. If $\overline{\xi}^{SR}_{a'} < 1$, then the fact that $\overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR}) > 0$ implies that $\overline{\xi}^{SR}$ does not respect priorities, a contradiction. If $\overline{\xi}^{SR}_{a'} = 1$, then by definition $d_{a'}(\overline{\xi}^{SR}) = \max_{c' \in C}\{\xi^{SR}_{a,c'}\}$ so we have $\overline{\xi}^{SR}_{a',c} < \max_{c \in C}\{\overline{\xi}^{SR}_{a',c'}\}$. As $\overline{\xi}^{SR}$ is category neutral, it must then be that $\overline{\xi}^{SR}_{a',c} + \sum_{\tilde{a} \in \hat{A}_{a',c}} \overline{\xi}^{SR}_{\tilde{a},c} = q_c$. However, as $\overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR}) > 0$, we have $\overline{\xi}^{SR}_{a,c} + \sum_{\tilde{a} \in \hat{A}_{a,c}} \overline{\xi}^{SR}_{\tilde{a},c} > q_c$, a contradiction. It can then be concluded that $\overline{\xi}^{SR}_{a',c} = d_{a'}(\xi^{SR})$; hence (42) applies to a' and we have $\overline{\xi}^{*}_{a',c} \leq \overline{\xi}^{SR}_{a',c}$ for all $a' \in \hat{A}_{a,c}$. It follows that

$$\overline{\xi}^{*}_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^{*}_{a',c} \leq \overline{\xi}^{SR}_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^{SR}_{a',c}. \tag{43}$$

Suppose toward a contradiction that $\overline{\xi}^{*}_{a,c} < d_a(\overline{\xi}^{*})$. By (40), we have $\overline{\xi}^{*}_{a,c} < \overline{\xi}^{SR}_{a,c}$ so (43) holds with a strict inequality, which implies that $\overline{\xi}^{*}_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^{*}_{a',c} < q_c$. If $\overline{\xi}^{*}_{a} < 1$, then $\overline{\xi}^{*}$ either is wasteful or does not respect priorities while if $\overline{\xi}^{*}_{a} = 1$, then $d_a(\overline{\xi}^{*}) = \max_{c' \in C}\{\overline{\xi}^{*}_{a,c'}\}$ so $\overline{\xi}^{*}_{a,c} < \max_{c' \in C}\{\overline{\xi}^{*}_{a,c'}\}$ and $\overline{\xi}^{*}$ is not category neutral. As $\overline{\xi}^{*}$ satisfies Axioms 1-4 by assumption, both cases yield a contradiction. It follows that $\overline{\xi}^{*}_{a,c} = d_a(\overline{\xi}^{*})$. Then, if $d_a(\overline{\xi}^{*}) = 1$, $\overline{\xi}^{*}_{a,c} = 1$ so $\overline{\xi}^{*}_{a} = 1$ and if $d_a(\overline{\xi}^{*}) < 1$, $\overline{\xi}^{*}_{a} = 1$ by definition. As a and c were chosen arbitrarily, it can then be concluded that

$$\overline{\xi}^{*}_{a,c} = d_a(\overline{\xi}^{*}) \quad \text{and} \quad \overline{\xi}^{*}_{a} = 1 \quad \text{for every } (a,c) \in A \times C \text{ such that } \overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR}). \tag{44}$$

Let $A^{SR}_{Q} = \cup_{c \in C}\{A^{SR}_{Q}(c)\}$ be the set of agents who are qualified for at least one category at $\overline{\xi}^{SR}$. For every agent $a \in A^{SR}_{Q}$, (44) implies that $\overline{\xi}^{*}_{a} = 1$ so $\sum_{a \in C^{SR}_{Q}(a)} \overline{\xi}^{*}_{a,c} + \sum_{a \in C^{SR}_{M}(a)} \overline{\xi}^{*}_{a,c} + \sum_{a \in C^{SR}_{U}(a)} \overline{\xi}^{*}_{a,c} = 1$. By (41) and (44), it follows that $|C^{SR}_{Q}(a)| d_a(\overline{\xi}^{*}) + \sum_{a \in C^{SR}_{M}(a)} \overline{\xi}^{*}_{a,c} = 1$; hence we have

$$d_a(\overline{\xi}^{*}) = \frac{1 - \sum_{c \in C^{SR}_{M}(a)} \overline{\xi}^{*}_{a,c}}{|C^{SR}_{Q}(a)|} \quad \text{for every } a \in A^{SR}_{Q}. \tag{45}$$

Next, let $C^{SR}_{M} = \cup_{a \in A} C^{SR}_{M}(a)$ be the set of categories that have a marginal agent and, for every $c \in C^{SR}_{M}$, let $a(c)$ be the agent who is marginal for c (by Claim 6, $a(c)$ is unique). Consider any category $c \in C^{SR}_{M}$. By definition, $\overline{\xi}^{SR}_{a,c} = 0$ for every $a \in A^{SR}_{U}(c)$ so $\sum_{a \in A} \overline{\xi}^{SR}_{a,c} = \overline{\xi}^{SR}_{a(c),c} + \sum_{a' \in A^{SR}_{Q}(c)} \overline{\xi}^{SR}_{a',c}$. As Claim 6 implies that $A^{SR}_{Q}(c) = \hat{A}_{a(c),c}$, we have $\sum_{a \in A} \overline{\xi}^{SR}_{a,c} = \overline{\xi}^{SR}_{a(c),c} + \sum_{a' \in \hat{A}_{a(c),c}} \overline{\xi}^{SR}_{a',c}$. If $\overline{\xi}^{SR}_{a(c)} < 1$, then, as $\overline{\xi}^{SR}$ is nonwasteful and respects priorities, it must be that $\overline{\xi}^{SR}_{a(c),c} + \sum_{a' \in \hat{A}_{a(c),c}} \overline{\xi}^{SR}_{a',c} = q_c$. If $\overline{\xi}^{SR}_{a(c)} = 1$, then by definition $d_a(\overline{\xi}^{SR}) = \max_{c' \in C}\{\overline{\xi}^{SR}_{a(c),c'}\}$ so $\overline{\xi}^{SR}_{a(c),c} < \max_{c' \in C}\{\overline{\xi}^{SR}_{a(c),c'}\}$. As $\overline{\xi}^{SR}$ ris category neutral, it must then be that $\overline{\xi}^{SR}_{a(c),c} + \sum_{a' \in \hat{A}_{a(c),c}} \overline{\xi}^{SR}_{a',c} = q_c$. The preceding argument has established that

$$\sum_{a \in A} \overline{\xi}^{SR}_{a,c} = q_c \quad \text{for every } c \in C^{SR}_{M}. \tag{46}$$

As $\overline{\xi}^{*}$ is an allocation, it must be that $\sum_{a \in A} \overline{\xi}^{*}_{a,c} \leq q_c$, and therefore (46) implies that $\sum_{a \in A} \overline{\xi}^{*}_{a,c} \leq \sum_{a \in A} \overline{\xi}^{SR}_{a,c}$.

Consider next a category $c \in C \setminus C^{SR}_{M}$. For every $a \in A^{SR}_{U}(c)$, $\overline{\xi}^{SR}_{a,c} = 0$ by definition and

$\overline{\xi}^*_{a,c} = 0$ by (41). As $A^{SR}_M(c) = \varnothing$ by assumption, it follows that $\sum_{a\in A} \overline{\xi}^{SR}_{a,c} = \sum_{a\in A^{SR}_Q(c)} \overline{\xi}^{SR}_{a,c}$ and $\sum_{a\in A} \overline{\xi}^*_{a,c} = \sum_{a\in A^{SR}_Q(c)} \overline{\xi}^*_{a,c}$. For every $a \in A^{SR}_Q(c)$, $\overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR})$ by definition and $\overline{\xi}^*_{a,c} = d_a(\overline{\xi}^*)$ by (44), which implies that $\sum_{a\in A} \overline{\xi}^{SR}_{a,c} = |A^{SR}_Q(c)| d_a(\overline{\xi}^{SR})$ and $\sum_{a\in A} \overline{\xi}^*_{a,c} = |A^{SR}_Q(c)| d_a(\overline{\xi}^*)$. As $d_a(\overline{\xi}^*) \leq d_a(\overline{\xi}^{SR})$, it must then be that $\sum_{a\in A} \overline{\xi}^*_{a,c} \leq \sum_{a\in A} \overline{\xi}^{SR}_{a,c}$. The argument in the last two paragraphs allows concluding that

$$\sum_{a\in A} \overline{\xi}^*_{a,c} \leq \sum_{a\in A} \overline{\xi}^{SR}_{a,c} \quad \text{for every } c \in C. \tag{47}$$

By (40), $|\overline{\xi}^*| = \sum_{a\in A} \overline{\xi}^*_a = \sum_{a\in A} \overline{\xi}^{SR}_a = |\overline{\xi}^{SR}|$.; hence, by definition, we have $\sum_{c\in C} \sum_{a\in A} \overline{\xi}^*_{a,c} = \sum_{c\in C} \sum_{a\in A} \overline{\xi}^{SR}_{a,c}$, which combined with (47) implies that

$$\sum_{a\in A} \overline{\xi}^*_{a,c} = \sum_{a\in A} \overline{\xi}^{SR}_{a,c} \quad \text{for every } c \in C. \tag{48}$$

For every category $c \in C^{SR}_M$, combining (46) and (48) yields $\sum_{a\in A} \overline{\xi}^*_{a,c} = q_c$. As $\overline{\xi}^*_{a,c} = 0$ for every $a \in A^{SR}_U(c)$ by (41) and $\overline{\xi}^*_{a,c} = d_a(\overline{\xi}^*)$ for every $a \in A^{SR}_Q(c)$ by (44), it must be that

$$\overline{\xi}^*_{a(c),c} + \sum_{a'\in A^{SR}_Q(c)} d_{a'}(\overline{\xi}^*) = q_c \quad \text{for every } c \in C^{SR}_M. \tag{49}$$

Combining (45) and (49), it follows that any allocation $\overline{\xi}^*$ that satisfies Axioms 1-4 must satisfy the following linear system of equations:

$$\overline{\xi}^*_{a(c),c} + \sum_{a'\in A^{SR}_Q(c)} \frac{1 - \sum_{c\in C^{SR}_M(a')} \overline{\xi}^*_{a',c}}{|C^{SR}_Q(a')|} = q_c \quad \text{for every } c \in C^{SR}_M. \tag{50}$$

The linear system of equations defined in (50) has $|C^{SR}_M|$ variables and $|C^{SR}_M|$ equations. For any $\overline{\xi}^*$ that satisfies Axioms 1-4, $(\overline{\xi}^*_{a(c),c})_{c\in C^{SR}_M}$ must be a solution to the linear system of equations defined in (50); moreover, for every agent-category pair (a, c) such that $a \in A^{SR}_Q(c)$, $\overline{\xi}^*_{a,c}$ must be determined by (44) and (45), and for every agent-category pair (a, c) such that $a \in A^{SR}_U(c)$, it must be that $\overline{\xi}^*_{a,c} = 0$, as per (41). As $\overline{\xi}^{SR}$ satisfies Axioms 1-4, $(\overline{\xi}^{SR}_{a(c),c})_{c\in C^{SR}_M}$ is a solution to the linear system of equations defined in (50). If all $|C^{SR}_M|$ equations in (50) are linearly independent, then $(\overline{\xi}^{SR}_{a(c),c})_{c\in C^{SR}_M}$ is the unique solution so $\overline{\xi}^{SR}$ is the unique allocation in the alternative allocation problem $\overline{R}$ to satisfy Axioms 1-4; hence the proof is complete. In the remainder of the proof, I show that the opposite case leads to a contradiction.

Toward a contradiction, suppose that the linear system of equations defined in (50) has strictly fewer than $|C^{SR}_M|$ linearly independent equations. Then, there is at least one degree of freedom; hence, arbitrarily fixing a category $c \in C_M^{SR}$, for any value of $\overline{\xi}^*_{(a(c),c)}$ there exists a vector $(\overline{\xi}^*_{a(c'),c'})_{c' \in C_M^{SR} \setminus \{c\}}$ such that $(\overline{\xi}^*_{a(c'),c'})_{c' \in C_M^{SR}}$ is a solution to the linear system of equations defined in (50).

Given an arbitrarily small positive number $\epsilon > 0$, I construct an allocation $\overline{\xi}^\epsilon$ as follows. Let $\overline{\xi}^\epsilon_{a(c),c} = \overline{\xi}^{SR}_{a(c),c} + \epsilon$ and, for every $c' \in C_M^{SR} \setminus \{c\}$, let $\overline{\xi}^\epsilon_{a(c'),c'}$ be such that $(\overline{\xi}^\epsilon_{a(c'),c'})_{c' \in C_M^{SR}}$ is a solution to the system of equations defined in (50). Then, for every agent-category pair (a, c) such that $a \in A_Q^{SR}(c)$, let $\overline{\xi}^\epsilon_{a,c}$ be determined by (44) and (45) and, for every agent-category pair (a, c) such that $a \in A_U^{SR}(c)$, let $\overline{\xi}^\epsilon_{a,c} = 0$ (as per (41)). By definition, for every $c' \in C_M^{SR}$, $0 < \overline{\xi}^{SR}_{a(c'),c'} < d_{a(c')}(\overline{\xi}^{SR})$. As all equations in (44) and (50) are linear, there exists a value $\epsilon > 0$ small enough so that $0 < \overline{\xi}^\epsilon_{a(c'),c'} < d_{a(c')}(\overline{\xi}^\epsilon)$ for every $c' \in C_M^{SR}$. Fixing such an ϵ, I next show that $\overline{\xi}^\epsilon$ satisfies Axioms 1-4.

By definition, $\overline{\xi}^\epsilon_{a,c} = 0$ for every agent-category pair (a, c) such that a is not eligible for c in the alternative problem $\overline{R}$ so $\overline{\xi}^\epsilon$ complies with eligibility requirements. I next introduce a small result that is useful to prove that $\overline{\xi}^\epsilon$ satisfies the other three axioms. For every agent-object pair (a, c) such that $a \in A_Q^{SR}(c)$, by the definition $\overline{\xi}^\epsilon$ satisfies the first part of (44): $\overline{\xi}^\epsilon_{a,c} = d_a(\overline{\xi}^\epsilon)$. I show that the second part of (44) also holds. If $d_a^(\overline{\xi}^\epsilon) = 1$, then $\overline{\xi}^\epsilon_{a,c} = 1$ so $\overline{\xi}^\epsilon_a = 1$. If $d_a^(\overline{\xi}^\epsilon) < 1$, then by definition $\overline{\xi}^\epsilon_a = 1$. Therefore, we have

$$\overline{\xi}^\epsilon_a = 1 \quad \text{for every } (a, c) \in A \times C \text{ such that } \overline{\xi}^{SR}_{a,c} = d_a(\overline{\xi}^{SR}). \tag{51}$$

Suppose toward a contradiction that $\overline{\xi}^\epsilon$ is wasteful. Then, there exists an agent-category pair (a, c) such that $\sum_{a' \in A} \overline{\xi}^\epsilon_{a',c} < q_c$, $\overline{\xi}^\epsilon_a < 1$, and a is eligible for c in $\overline{R}$. By (50) the fact that $\sum_{a' \in A} \overline{\xi}^\epsilon_{a',c} < q_c$ implies that $c \notin C_M^{SR}$ so $A_M^{SR}(a) = \varnothing$; hence it must be that either $a \in A_Q^{SR}(c)$ or $a \in A_U^{SR}(c)$. If $a \in A_Q^{SR}(c)$, then by (51), $\overline{\xi}^\epsilon_a = 1$, a contradiction. If $a \in A_U^{SR}(c)$, then by definition a is not eligible for c in $\overline{R}$, a contradiction. It can then be concluded that $\overline{\xi}^\epsilon$ is nonwasteful. Next, suppose toward a contradiction that $\overline{\xi}^\epsilon$ does not respect priorities. Then, there exists an agent-category pair (a, c) and a lower-priority agent $a' \in \check{A}_{a,c}$ such that $\overline{\xi}^\epsilon_a < 1$ and $\overline{\xi}^\epsilon_{a',c} > 0$. If $a \in A_Q^{SR}(c)$, then (51) implies that $\overline{\xi}^\epsilon_a = 1$, a contradiction. If $a \notin A_Q^{SR}(c)$, then by Claim 6, $a' \in A_U^{SR}(c)$ so, by (41), $\overline{\xi}^\epsilon_{a',c} = 0$, a contradiction. It follows that $\overline{\xi}^\epsilon$ respects priorities. Finally, suppose toward a contradiction that $\overline{\xi}^\epsilon$ is not category neutral. Then, there exists an agent-category pair (a, c) such that a is eligible for c in $\overline{R}$, $\overline{\xi}^\epsilon_{a,c} < \max_{c' \in C}\{\overline{\xi}^\epsilon_{a,c'}\}$, and $\overline{\xi}^\epsilon_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^\epsilon_{a',c} < q_c$. By definition, $\max_{c' \in C}\{\overline{\xi}^\epsilon_{a,c'}\} \leq d_a(\overline{\xi}^\epsilon)$, which by (44) implies that $a \notin A_Q^{SR}(c)$. Moreover, the assumption that a is eligible for c in $\overline{R}$ implies by definition that $a \notin A_U^{SR}(c)$ so it must be that $a \in A_M^{SR}(c)$. In that case, however, we have $\overline{\xi}^\epsilon_{a,c} + \sum_{a' \in A_Q^{SR}(c)} \overline{\xi}^\epsilon_{a',c} = q_c$ by (50) and that $A_Q^{SR}(c) = \hat{A}_{a,c}$ by Claim 6. It follows that $\overline{\xi}^\epsilon_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^\epsilon_{a',c} = q_c$, a contradiction.

The preceding argument has established that there exists an allocation $\overline{\xi}^{\epsilon}$ in the alternative rationing problem $\overline{R}$ that satisfies Axioms 1-4. By definition, there exists a category $c \in C_M^{SR}$ such that $\overline{\xi}^{\epsilon}_{a(c),c} = \overline{\xi}^{SR}_{a(c),c} + \epsilon > \overline{\xi}^{SR}_{a(c),c}$. Moreover, by construction, both $\overline{\xi}^{\epsilon}$ and $\overline{\xi}^{SR}$ satisfy (49) so we have

$$\overline{\xi}^{\epsilon}_{a(c),c} + \sum_{a' \in A_Q^{SR}(c)} d_{a'}(\overline{\xi}^{\epsilon}) = \overline{\xi}^{SR}_{a(c),c} + \sum_{a' \in A_Q^{SR}(c)} d_{a'}(\overline{\xi}^{SR}) = q_c.$$

Then, as $\overline{\xi}^{\epsilon}_{a(c),c} > \overline{\xi}^{SR}_{a(c),c}$, there must exist an agent $a' \in A_Q^{SR}(c)$ such that $d_{a'}(\overline{\xi}^{\epsilon}) < d_{a'}(\overline{\xi}^{SR})$, which contradicts (40).

($\overline{x}^{\overline{N}} = x^N$) Though not entirely analogous, the reasoning is this last part of the proof is similar to that of the first part (which shows that $\xi^{SR} = \overline{\xi}^{SR}$). The main difference is that I follow the SRLP algorithm instead of the SR algorithm. For the original rationing problem R, I denote by x^i, δ^i, and d^i the preallocation, the LP demand vector, and the demand vector calculated by the SRLP algorithm in any given Round i. Similarly, for the alternative rationing problem $\overline{R}$, I denote by $\overline{x}^i$, $\overline{\delta}^i$, and $\overline{d}^i$ the preallocation, the LP demand vector, and the demand vector calculated by the SRLP algorithm in any given Round i. I also denote by d^0 and $\overline{d}^0$ the initial demand vectors in R and $\overline{R}$, respectively.

By definition, we have $d^0 = \overline{d}^0 = \delta^1 = \overline{\delta}^1 = \mathbf{1}$. Consider any Round $i = 1, \ldots, \min\{N, \overline{N}\}$ of the SRLP algorithm and suppose, toward an inductive argument, that $\delta^i = \overline{\delta}^i$. I show that $x^i = \overline{x}^i$ and, if $i < \min\{N, \overline{N}\}$, $\delta^{i+1} = \overline{\delta}^{i+1}$. Arbitrarily fix an agent a and a category c. I first show that $x^i_{a,c} = \overline{x}^i_{a,c}$, considering separately the cases in which $\xi^{SR}_{a,c} > 0$ and $\xi^{SR}_{a,c} = 0$.

Case 1: $\xi^{SR}_{a,c} > 0$. As ξ^{SR} complies with eligibility requirements, the case assumption implies that a is eligible for c in the rationing problem R. By definition, the case assumption also implies that a is eligible for c in the alternative rationing problem $\overline{R}$. Again by definition, it follows that $x^i_{a,c} = \min\{\delta^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} \delta^i_{a'}, 0\}\}$ and $\overline{x}^i_{a,c} = \min\{\overline{\delta}^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} \overline{\delta}^i_{a'}, 0\}\}$ so the induction hypothesis that $\delta^i = \overline{\delta}^i$ implies that $x^i_{a,c} = \overline{x}^i_{a,c}$.

Case 2: $\xi^{SR}_{a,c} = 0$. By definition, the case assumption implies that a is not eligible for c in the alternative rationing problem $\overline{R}$; hence, $\overline{x}^i_{a,c} = 0$ and it remains to show that $x^i_{a,c} = 0$. If a is not eligible either in the alternative rationing problem R, it follows by definition that $x^i_{a,c} = 0$; therefore, for the remainder of the argument, I assume that a is eligible for c in R.

As $\overline{x}^{\overline{N}} = \overline{\xi}^{SR} = \xi^{SR}$, (39) implies that

$$\sum_{a' \in \hat{A}_{a,c}} \overline{x}^{\overline{N}}_{a',c} = q_c. \tag{52}$$

Consider any agent $a' \in \hat{A}_{a,c}$. By construction, $\overline{x}^{\overline{N}}$ is an allocation (otherwise the SRLP algorithm would not end in Round $\overline{N}$) so by definition $d_{a'}(\overline{x}^{\overline{N}}) = \max_{c' \in C}\{x^{\overline{N}}_{a',c'}\}$ if $\overline{x}^{\overline{N}}_a = 1$ and $d(\overline{x}^{\overline{N}}) = 1$ if $\overline{x}^{\overline{N}}_a < 1$. It follows that $\overline{d}^{\overline{N}}_{a'} = d_{a'}(\overline{x}^{\overline{N}}) \geq \max_{c' \in C}\{x^{\overline{N}}_{a',c'}\} \geq \overline{x}^{\overline{N}}_{a',c}$. As a' was chosen arbitrarily, we have $\overline{d}^{\overline{N}}_{a'} \geq \overline{x}^{\overline{N}}_{a',c}$ for every $a' \in \hat{A}_{a,c}$; hence (52) implies that $\sum_{a' \in \hat{A}_{a,c}} \overline{d}^{\overline{N}}_{a'} \geq q_c$. Then, by Lemma C.3, we have $\sum_{a' \in \hat{A}_{a,c}} \overline{\delta}^i_{a'} \geq q_c$. As a is eligible for c, it follows by definition that $\overline{x}^i_{a,c} = \min\{\overline{\delta}^i_a, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} \overline{\delta}^i_{a'}, 0\}\} = 0$.

As a and c were chosen arbitrarily, it has been established that $x^i_{a,c} = \overline{x}^i_{a,c}$ for every agent a and every category c; hence we have $x^i = \overline{x}^i$. Then, by construction, it follows that $d^i = \overline{d}^i$. Moreover, if $i < \min\{N, \overline{N}\}$, then by construction the fact that $x^i = \overline{x}^i$ and $d^i = \overline{d}^i$ implies that $\delta^{i+1} = \overline{\delta}^{i+1}$. By induction, it follows that $x^{\min\{N,\overline{N}\}} = \overline{x}^{\min\{N,\overline{N}\}}$ so it must be that, in both rationing problems, the SRLP algorithm ends in the same Round $N = \overline{N}$ and produces the same allocation $x^N = \overline{x}^{\overline{N}}$. $\square$