PROCESSING RESERVES SIMULTANEOUSLY

DAVID DELACRÉTAZ

Abstract. Policymakers frequently use reserve categories to combine competing objectives in allocating a scarce resource 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 it has led to unintended consequences in practice. I introduce a new, more transparent way of processing reserves, which handles all categories simultaneously. I characterize my solution, showing that it satisfies basic desiderata and is *category neutral*: 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, reserve system, simultaneous processing, category neutrality

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. *Reserve systems* constitute a more flexible alternative, as they allow multiple priority orders, each of which applies to part of the capacity. For example, several US states followed a proposal by Pathak, Sönmez, Ünver, and Yenmez (2021) and initially reserved a

---

Date: First version: August 15, 2020. This version: August 13, 2021.
I am grateful to Marcelo Fernandez, Ravi Jagadeesan, Bernhard Kasberger, Paul Klemperer, Scott Kominers, Simon Loertscher, David Ronayne, Alex Teytelboym, Tayfun Sönmez, M. Bumin Yenmez, Özgür Yılmaz, and audiences at various conferences and seminars for their valuable feedback.
Affiliation: Department of Economics, University of Manchester, Oxford Road, Manchester M13 9PL, United Kingdom. *Email*: david.delacretaz@manchester.ac.uk.

proportion of their COVID-19 vaccines for disadvantaged communities,[1] Chicago's exam schools reserve 70% of their seats for students from specific neighborhoods (Dur, Pathak, and Sönmez, 2020), and 20,000 of the 65,000 H-1B visas delivered every year by the US Customs and Immigration Service are reserved for applicants with an advanced degree (Pathak, Rees-Jones, and Sönmez, 2020a).

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., vaccine doses, school places, visas) have to be allocated to *agents* (e.g., patients, students, applicants) and are split into *(reserve) categories* (e.g., disadvantaged communities, neighborhoods, advanced degree, open), each of which has its own priority order over the agents. Both in practice and in the literature, categories are processed sequentially, following a *precedence order*. Each category allocates its *quota* (the number of units allocated through 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 for multiple categories receives one from 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. My proposed solution is to process categories simultaneously rather than sequentially, thus eliminating the precedence order and ensuring that the relative importance of a category only depends on its quota. Categories simultaneously allocate capacity 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 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

[1]The use of reserve systems for medical rationing is documented at www.covid19reservesystem.org.

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. 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 as they reversed the precedence order. 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, in large part because of concerns over the lack of transparency associated with the precedence order, Boston Public Schools abandoned their reserve system when they realized why it was not producing its intended outcome (Dur, Kominers, Pathak, and Sönmez, 2018). The solution presented in this paper eliminates the precedence order and the problems it creates, thus making reserve systems easier to design for policymakers and more transparent for participants.

Theoretical contribution. I introduce the *simultaneous reserve (SR)* algorithm. In each round, categories simultaneously allocate their quotas to their respective highest-priority agents. If an agent is allocated more than one unit in aggregate (i.e., over all categories), then the maximum amount she receives from any category is reduced until she is allocated exactly one unit in aggregate. It turns out that the SR algorithm may run for infinitely many rounds without finding an allocation; however, I show that it converges to one (Theorem 1), which I call the *simultaneous reserve (SR)* allocation. The SR allocation satisfies three standard axioms introduced by Pathak, Sönmez, Ünver, and Yenmez (2021)—*compliance with eligibility requirements*, *nonwastefulness*, and *respect of priorities*—as well as a fourth axiom 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 every allocation satisfying these four axioms allocates to each agent the same amount of capacity in aggregate (Theorem 3). Finally, I propose a polynomial-time algorithm to compute the SR allocation (Theorem 4).

Related Literature. This paper builds upon a rich literature on allocation problems with distributional constraints. Abdulkadiroğlu (2005) proposes a solution to incorporate affirmative action through maximum quotas on specific types of students, and Kojima (2012) shows how maximum quotas can have unintended consequences. Hafalir, Yenmez, and Yildirim (2013) propose using minimum quotas instead, which Westkamp (2013) adapts to the German university admission system. Ehlers, Hafalir, Yenmez, and Yildirim (2014) and Echenique and Yenmez (2015) extend the approach to include more categories. Gonczarowski, Kovalio, Nisan, and Romm (2020) use a combination of minimum and maximum quotas to design a centralized matching market for Israeli gap-year programs.

Kominers and Sönmez (2016) formally introduce a reserve system with sequential processing, generalizing previous models by allowing any priorities and precedence order. Sequential reserve systems have been studied in various contexts, including Boston and Chicago's public schools (Dur, Kominers, Pathak, and Sönmez, 2018; Dur, Pathak, and Sönmez, 2020), medical rationing (Pathak, Sönmez, Ünver, and Yenmez, 2021), the H-1B visa program (Pathak, Rees-Jones, and Sönmez, 2020a), and university admissions in India (Sönmez and Yenmez, 2021, 2020b; Aygün and Turhan, 2020a,b) and Brazil (Aygun and Bó, 2021). Pathak, Rees-Jones, and Sönmez (2020b) provide experimental evidence of how difficult it is to account for the precedence order. The present paper departs from this literature by proposing a simultaneous reserve system that does not rely on any precedence order.

Sönmez and Yenmez (2020a) consider a reserve system with a *baseline priority order*. Each category 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' (2021) *smart reserves* allow for an arbitrary number of units to be allocated by the baseline priority order before the other categories are considered. The *reverse rejecting* and *smart reverse rejecting* rules of Aziz and Brandl (2021) generalize both procedures by allowing the categories' priorities to differ from the baseline. An important feature of the SR allocation is that it is entirely pinned down by the categories' priorities and quotas: it does not rely on any baseline priority order over agents or precedence order over reserved and unreserved units. Moreover, each of the aforementioned solutions matches agents with categories while the SR allocation typically splits units across categories.

The SR algorithm can be interpreted as a DA procedure in which categories propose to agents, and subject to small differences in the setups, it is equivalent to the *fractional deferred acceptance* algorithm of Kesten and Ünver (2015). I provide new results for that family of algorithms; in particular, I show that the SR algorithm finds an allocation in polynomial time when there are only two categories and otherwise can be turned into a polynomial-time algorithm by using linear programming.

Last, the present paper is connected 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) and Aziz and Brandl (2020). While the SR algorithm and probabilistic serial rule may appear similar, I show in Section 4.4 that they yield different outcomes.

Organization of the paper. Section 2 presents a motivating example. Section 3 introduces the setup and the four axioms. Section 4 introduces the SR algorithm and analyzes the properties of the SR allocation. Section 5 presents an algorithm that produces the SR allocation in polynomial time. Section 7 concludes and all proofs are in the appendix.

2. Motivating Example

I illustrate sequential and simultaneous processing with a simple example.[2] A school has four seats. Two of them are reserved for students living within walking distance and the other two are open to all students. There are six students—Amy, Bob, Claire, Dan, Eric, Fiona—and four of them—Amy, Bob, Eric, Fiona—live within walking distance of the school. There is a general priority order over the students, which I assume to be alphabetical. The "walk" category prioritizes students living within walking distance and breaks ties alphabetically; hence, its priority is Amy, Bob, Eric, Fiona, Claire, Dan. The "open" category ranks students alphabetically.

Consider sequential processing, starting with the walk category. That category allocates its seats to its two highest-priority students, Amy and Bob. The open

[2]This example models the system used by Boston Public Schools until 2013 and uses the school choice terminology for concreteness. However, the same example can be framed in different contexts, such as allocating a vaccine and reserving some doses for members of disadvantaged communities.

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.

category is processed next, but its highest-priority students—Amy and Bob—have already been allocated a seat. Therefore, the two seats are allocated to the next highest-priority students, Claire and Dan. Table 1a summarizes the outcome. The four students with the highest general priority—Amy, Bob, Claire, Dan—are allocated a seat; hence, the same outcome would have been achieved without a reserve. Suppose now that the open category 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, 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, 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 students only require one seat, Amy and Bob only keep half a seat from each category; hence, they each obtain one seat in aggregate. Each category has one seat left to allocate to their respective third-priority students: Eric and Claire. Table 2 summarizes the outcome. Both categories are equally important in the outcome: Amy and Bob are each allocated a seat through both categories, Claire is allocated a seat through the open category, and Eric is allocated a seat through the walk category.

3. 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 $\sum_{c \in C} q_c = q$ and a strict **priority order** π_c over the 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 $\hat{A}_{a,c} = \{a' \in A : a' \pi_c a\}$ and $\check{A}_{a,c} = \{a' \in A : a \pi_c a'\}$ the set of agents who have a higher and a lower priority than a for c, respectively. A **rationing problem** is a tuple $R = (A, C, (\pi_c)_{c \in C}, (q_c)_{c \in C})$.

In practice, reserve systems often have an open category and one or more specific categories that prioritize agents from a target group. For example, each of Chicago's exam schools has an open category that allocates 30% of the seats and ranks students based on their exam scores, and four specific categories, each of which allocates 17.5% of the seats and prioritizes students living in certain neighborhoods (Dur, Pathak, and Sönmez, 2020). In the context of allocating a medical good (e.g., a vaccine), an open category could rank patients based on medical needs and specific categories could prioritize healthcare workers or patients from disadvantaged communities (Pathak, Sönmez, Ünver, and Yenmez, 2021). In some applications, agents who are not in a category's target group are not eligible for it; for example, applicants without an advanced degree can only receive an H-1B visa from the open category. The model presented is completely flexible as it allows for any number of categories with any quotas and priority orders. Moreover, to the best of my knowledge, this paper is the first to allow noninteger category quotas. This can be useful in problems where the number of units is small; for example, if there are 5 units and half of them need to be reserved for a target group, each category's quotas can be set to 2.5.

An **allocation** is a 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) $\sum_{c'\in C} \xi_{a,c'} \leq 1$. Thus, each element $\xi_{a,c}$ specifies the amount of capacity that c allocates to 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 allocation ξ, which can be interpreted as the probability that a obtains a unit. An **aggregate allocation** is a vector $\rho = (\rho_a)_{a\in A}$ such that $\rho_a \in [0,1]$ for every agent a and $\sum_{a\in A} \rho_a \leq q$. I denote by $\rho(\xi) = (\xi_a)_{a\in A}$ the aggregate allocation generated by the allocation ξ.

I introduce four axioms. The first three are common both in practice and in the literature (see Pathak, Sönmez, Ünver, and Yenmez, 2021) but are generalized here to a setting in which units can be split over categories.

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

The first axiom only matters when some agents are not eligible for some categories; for example, it would preclude H-1B visa applicants without an advanced degree to obtain any capacity from the advanced degree category.

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

If every agent is eligible for every category, the second axiom requires that $\sum_{a\in A} \xi_a = \min\{|A|, q\}$. In general, a category may not allocate its entire quota as long as all of its eligible agents are allocated one unit in aggregate.

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

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

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, a unit to each of their highest-priority eligible agents who have not yet been allocated a unit.[3] As a result, an agent who qualifies for multiple categories receives a unit from whichever is processed first. At the heart of my proposed solution is the idea that, while units are ultimately indivisible, how they are divided across categories is merely an accounting exercise; hence, the unit that an agent receives may be split across multiple categories. The fourth axiom, which is newly introduced in this paper, stipulates that the unit an agent is allocated should be split 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 the walk category's entire quota is allocated to higher-priority students. From a normative perspective, Axiom 4 ensures that all categories are treated the same in regard to sharing an agent so that their relative importance only depends on their quotas.

4. Simultaneous Reserve (SR) Allocation

In this section, I introduce an algorithm that processes categories simultaneously and show that it converges to an allocation that satisfies Axioms 1-4. Moreover, I show that every allocation satisfying Axioms 1-4 generates the same aggregate 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 in aggregate. Formally, a preallocation is a matrix $x = (x_{a,c})_{a \in A, c \in C}$ such that, for every

[3]See Pathak, Sönmez, Ünver, and Yenmez (2021, p.21) for a full description of that procedure.

ALGORITHM 1: Simultaneous Reserve (SR)

Initialization Set $x^0 = \mathbf{0}_{|A|\times|C|}$ and $d^0 = \mathbf{1}_{|A|}$.

Round $i \geq 1$:

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

Demand Adjustment For every agent a such that $x^i_a < 1$, set $d^i_a = 1$. For every agent a such that $x^i_a = 1$, set $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$. For every agent a such that $x^i_a > 1$, set d^i_a such that $\sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = 1$.

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. For any demand vector $d = (d_a)_{a\in A}$, it is also useful to define $x(d)$ to be the preallocation calculated by the capacity allocation stage; that is, for every agent-category pair (a, c), $x_{a,c}(d) = \min\{d_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d_{a'}, 0\}\}$ if a is eligible for c and $x_{a,c}(d) = 0$ otherwise. Similarly, for any preallocation x, let $d(x)$ be the demand vector calculated by the demand adjustment stage. That is, for every agent a, $d_a(x) = 1$ if $x_a < 1$, $d_a(x) = \max_{c\in C}\{x_{a,c}\}$ if $x_a = 1$, and $\sum_{c\in C} \min\{d_a(x), x_{a,c}\} = 1$ if $x_a > 1$.

At the start of the SR algorithm, no agent is allocated any capacity and each agent has a *demand* of 1. An agent's demand can be interpreted as the amount of capacity that an agent requires from any category to be allocated one unit in aggregate. Throughout the algorithm, demands fall as agents are allocated capacity.

The first round starts with the capacity allocation stage, which calculates a preallocation $x^1 = x(d^0)$ as follows. Each category allocates one unit of capacity to each of its highest-priority agents 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.

At x^1, some agents may be allocated more than one unit in aggregate. To turn x^1 into an allocation, the demand adjustment stage updates the demand vector to $d^1 = d(x^1)$. The demand of an agent who has not yet been allocated a unit in aggregate remains one (i.e., $d^1_a = 1$ if $x^1_a < 1$). The demand of an agent who has been allocated exactly one unit in aggregate falls to the maximum capacity she is allocated from any category (i.e., $d^1_a = \max_{c\in C}\{x^1_{a,c}\}$ if $x^1_a = 1$). The demand of an agent who

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.

has been allocated more than one unit in aggregate falls to the level that ensures this agent keeps exactly one unit (i.e., $\sum_{c\in C} \min\{d_a^1, x_{a,c}^1\} = 1$ if $x_a^1 > 1$).

Every subsequent Round i starts with a demand vector d^{i-1} and calculates a preallocation $x^i = x(d^{i-1})$ in the capacity allocation stage. 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. The demand adjustment stage calculates $d^i = d(x^i)$ and the algorithm continues in Round $i+1$, in which $x^{i+1} = d(x^i)$ and $d^{i+1} = d(x^{i+1})$ are calculated.

I next illustrate the SR algorithm with an example.

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

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); therefore, 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. Agent a_4 is now allocated 7/6 in aggregate (1/6 from c_2 and 1 from c_4); hence, 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; therefore, the SR algorithm has found the following allocation:

$$x^4 = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{array} \!\!\begin{array}{c} \begin{array}{cccc} c_1 & c_2 & c_3 & c_4 \end{array} \\ \begin{pmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/6 & 0 & 2/3 & 0 \\ 0 & 1/6 & 0 & 5/6 \\ 0 & 0 & 0 & 1/6 \end{pmatrix} \end{array} \quad \text{with} \quad \rho(x^4) = \begin{array}{c} \begin{array}{ccccc} a_1 & a_3 & a_3 & a_4 & a_5 \end{array} \\ \begin{pmatrix} 1 & 1 & 5/6 & 1 & 1/6 \end{pmatrix} \end{array}.$$

It is easy to verify that 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, there is no violation 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, $x^4_{a_4,c_2} + \sum_{a \in \hat{A}_{a_4,c_2}} x^4_{a,c_2} = 1/6 + 1/3 + 1/2 = 1 = q_{c_2}$.

4.2. **Outcome of the SR Algorithm.** I first illustrate how, in contrast to Example 1, the SR allocation may never find an allocation. I then show that the SR algorithm still converges to an allocation, even when it does not reach one.

Example 2. There are four agents and three categories. The priorities and quotas are $\pi_{c_1} : a_1, a_2, a_3, a_4, \emptyset$, $\pi_{c_2} : a_3, a_2, a_1, a_4, \emptyset$, $\pi_{c_3} : a_1, a_3, a_2, a_4, \emptyset$, $q_{c_1} = q_{c_2} = 1$, $q_{c_3} = 2$.

The operation 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, their demands drop 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. At this point, the SR algorithm begins to cycle. Agent a_1 (a_3) is allocated 7/6 in aggregate but can only be allocated 1/6 from c_2 (c_1) so her demand is adjusted to 5/12. In Round 4, as a_1 and a_3's demands have each dropped by 1/12, 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. As a result, a_1 and a_3 are each allocated 13/12 in aggregate in Round 4, so their demands drop to 9/24. In Round 5, again half of the capacity

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

TABLE 4. SR algorithm applied to Example 2.

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 and c_2 allocate an extra 1/24 to a_3 and a_1, respectively). The SR algorithm continues to cycle forever, with the amount of reallocated capacity halving in each round. The SR algorithm never reaches an allocation in Example 2; however, it converges to one in which a_4 receives 1 from c_4 and each of a_1, a_2, and a_3 receives 1/3 from each category. As the next result shows, this property generalizes.

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.[4] To understand the intuition behind Theorem 1, it is useful to define for every Round i the allocation $\xi^i = (\xi^i_{a,c})_{a\in A, c\in C}$ such that, for every agent a and every category c, $\xi^i_{a,c} = \min\{d^i_a, x^i_{a,c}\}$.[5] I also define the matrix $z^i = x^i - \xi^i$ to be the **excess supply** in Round i of the SR algorithm. I denote the total excess supply by $|z^i| = \sum_{a\in A} z^i_a = \sum_{a\in A}\sum_{c\in C} z^i_{a,c}$. The excess supply corresponds to the amount of capacity that agents are allocated in addition to the unit they require; therefore, the

[4]In Section 5, I propose a polynomial-time algorithm to calculate the SR allocation.

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

total excess supply is the amount of capacity that will be reallocated in the next round. In Example 2, the total excess supply is 1/3 in Round 3 (a_1 and a_3 are each allocated 7/6 in aggregate) and is halved in every subsequent round; hence, it converges to zero. This property generalizes for two reasons. First, until they are allocated one unit in aggregate, agents permanently keep the capacity allocated to them; therefore, the excess supply weakly decreases from one round to the next. Second, as excess supply is reallocated, categories allocate capacity further down their priority order; eventually, categories must reach the bottom so there is an upper bound on how much excess supply can be reallocated throughout the algorithm. As the amount of capacity reallocated converges to zero, in the limit every agent is allocated at most one unit so x^i converges to an allocation.

Example 2 has three categories, which leaves open the question of whether the SR algorithm finds an allocation in finitely many rounds when there are only two categories. The next result provides an affirmative answer.

Proposition 1. *Suppose that* $|C| = 2$*. Then, the SR algorithm finds the SR allocation after fewer than* $8|A|$ *rounds.*

Proposition 1 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. Intuitively, the reason the SR algorithm does not find an allocation in Example 2 is as follows. In each round, a_1 obtains additional capacity from c_2. That capacity is reallocated in the next round: half of it goes to a_3 through c_1 and the other half goes through a_4 through c_3. Similarly, half of the extra capacity that a_3 obtains from c_1 is reallocated to a_1 through c_2 and the other half is reallocated to a_4 through c_3. Such a situation cannot occur when there are only two categories because the additional amount of capacity that an agent obtains from one category can only be reallocated through the other category.

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.

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

In every Round i of the SR algorithm, 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 are able to allocate that upper bound do so. Then, these categories allocate the same amount of capacity to that agent. As I show in Appendix B, these properties continue to hold in the limit.

A question arising from Theorem 2 is whether the SR allocation is the only one to satisfy Axioms 1-4. The next example provides a negative answer; however, I show that all allocations satisfying Axioms 1-4 generate 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, for any $\lambda \in [0.5, 1]$, the following allocation satisfies Axioms 1-4:

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

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 $\xi^\lambda_{a_2,c_1} + \xi^\lambda_{a_1,c_1} = 1 = q_{c_1}$. In the special case where $\lambda = 1$, ξ^λ is the SR allocation.

Note one aspect of Example 3: for every $\lambda \in [0.5, 1]$, a_1 and a_2 are each allocated one unit in aggregate; that is, $\rho(\xi^\lambda) = \rho(\xi^{SR})$ for every $\lambda \in [0.5, 1]$. The next result shows that this property generalizes. I call the aggregate allocation $\rho(\xi^{SR})$ generated by the SR allocation the **SR aggregate allocation** and I call any allocation 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 aggregate. Moreover, Theorem 3 characterizes the SR aggregate allocation as the only aggregate allocation that is generated by an allocation satisfying

Axioms 1-4. Finally, Theorem 3 is sharp in the sense that each of the four axioms is needed to characterize the SR aggregate allocation.

Proposition 2. *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.

Proposition 3. *For every allocation $\xi^* \neq \xi^{SR}$ satisfying Axioms 1-4, $d(\xi^*) < d(\xi^{SR})$.*

Proposition 3 characterizes the SR allocation as the allocation satisfying Axioms 1-4 with the largest demand. 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. Therefore, the SR algorithm finds in each round an upper bound for the demand in any allocation satisfying Axioms 1-4 and converges to an allocation whose demand has been reduced just enough to satisfy Axioms 1-4.

A novelty of this paper is that an agent may receive parts of an indivisible unit from different categories, which does not cause any practical problem as long as such an agent is allocated one unit in aggregate. However, some agents may obtain an amount of capacity strictly between zero and one at the SR aggregate allocation, as seen in Example 1: a_3 and a_5 obtain 5/6 and 1/6, respectively. A common approach (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 such that each agent a is allocated a unit with probability ξ_a^{SR}. In Example 1, the last unit would be allocated to a_3 with probability 5/6 and to a_5 with probability 1/6. The next result shows that the number of agents affected by that lottery is limited.

Proposition 4. *At the SR aggregate allocation, at most $|C|$ agents are allocated an amount of capacity strictly between zero and one.*

The intuition for Proposition 4 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 or one unit at ρ^{SR}.

In practice, agents often have to declare (and provide evidence) that they meet criteria that would give them a higher priority; hence, they might be able to lower their priority for some categories. Some reserve systems have been shown to be manipulable in that way; for example, university applicants in Brazil and India may gain from not revealing all categories for which they are eligible (Aygun and Bó, 2021; Sönmez and Yenmez, 2021). It seems natural to think that the SR allocation does not suffer from this drawback because the higher an agent's priority for a category, the more capacity she obtains from that category, and therefore the more capacity she obtains in aggregate. The last result formalizes this intuition. Let $\widehat{R} = (A, C, (\hat{\pi}_c)_{c\in C}, (q_c)_{c\in C})$ be a rationing problem that is identical to R except that the priority of some agent a is lower for some categories. Formally, for every category c and for any two agents $b \in (A \cup \{\emptyset\}) \setminus \{a\}$ and $b' \in (A \cup \{\emptyset\}) \setminus \{b\}$ with $b\pi_c b'$, the new priority profile $(\hat{\pi}_c)_{c\in C}$ satisfies $b\hat{\pi}_c b'$. I denote by $\widehat{\xi}^{SR}$ the SR allocation of $\widehat{R}$.

Proposition 5. $\xi^{SR}_a \geq \widehat{\xi}^{SR}_a$.

4.4. **Random Precedence Order and Probabilistic Serial.** As the SR allocation lies in between "extreme" solutions found by sequential processing, one might think it is equivalent to the ex ante allocation obtained by randomizing the precedence order. The motivating example readily disproves this conjecture: Randomizing with equal probability over the two possible precedence orders gives Claire, Dan, Eric, and Fiona each an ex ante probability of 0.5, which differs from the SR aggregate allocation. One might then wonder whether the SR aggregate allocation can be replicated by allocating one unit at a time (see Kominers and Sönmez, 2016). In the motivating example, either sequence walk-open-walk-open or open-walk-open-walk yields an SR-equivalent allocation. However, in general, the answer is again negative.

Example 4. There are seven agents and two categories, each with a quota of 1. The priorities are $\pi_{c_1} : a_1, a_3, a_5, a_7, \ldots$ and $\pi_{c_2} : a_2, a_1, a_4, a_3, a_6, \ldots$.

c_1	(3)
a_1	1/2
a_3	1/2
a_5	1
a_7	1

c_2	(3)
a_2	1
a_1	1/2
a_4	1
a_3	1/2
a_6	0

(A) SR allocation in Example 4.

c_1	(3)
a_1	1
a_3	1
a_5	1
a_7	0

c_2	(3)
a_2	1
a_1	0
a_4	1
a_3	0
a_6	1

(B) Alternative allocation in Example 4.

TABLE 5. Two allocations in Example 4.

Table 5a displays the SR allocation in Example 4. In aggregate, every agent but a_6 is allocated a unit. Allocating one unit at a time—starting with either category—yields the alternative allocation displayed in Table 5b, in which every agent but a_7 is allocated one unit in aggregate. In general, sequentially allocating any amount of capacity $\epsilon \in (0, 1]$ in either order yields the allocation from Table 5b. In the limit, as ϵ approaches zero, this procedure converges to the *probabilistic serial* algorithm of Bogomolnaia and Moulin (2001), in which categories "eat" agents. Example 4 therefore shows that the probabilistic serial and SR algorithms are not equivalent. The key difference between sequential processing (even allocating a vanishingly small amount of capacity at a time) and the SR algorithm is that, in the former, any capacity allocation is final, while the latter allows the amount of capacity that a category allocates to an agent to fall when the agent is allocated capacity from other categories. In Example 4, the SR algorithm has c_1 tentatively allocating one unit each to a_1 and a_3, but this falls to half a unit once a_1 and a_3 are allocated capacity from c_2. In contrast, a sequential or probabilistic serial procedure has c_1 allocating one unit each to a_1 and a_3 permanently.

5. Simultaneous Reserve with Linear Programming

In this section, I show that the SR allocation can be computed in polynomial time by adding linear programming to the SR algorithm.

5.1. **Notation and Terminology.** Fix a preallocation x that satisfies Axioms 1-4. Agent a's **status** for category c at x is **qualified** if $x_{a,c} \geq d_a(x)$, **marginal** if $0 < x_{a,c} < d_a(x)$, and **unqualified** if $x_{a,c} = 0$. For every agent a, I denote by $C^x_Q(a) = \{c \in C : x_{a,c} \geq d_a(x)\}$, $C^x_M(a) = \{c \in C : 0 < x_{a,c} < d_a(x)\}$, and

$C_U^x(a) = \{c \in C : x_{a,c} = 0\}$ the set of categories for which a is qualified, marginal, and unqualified at x, respectively. For every category c, I denote by $A_Q^x(c) = \{a \in A : x_{a,c} \geq d_a(x)\}$, $A_M^x(c) = \{a \in A : 0 < x_{a,c} < d_a(x)\}$, and $A_U^x(c) = \{a \in A : x_{a,c} = 0\}$ the set agents that are qualified, marginal, and unqualified for c at x, respectively.

I call each agent a an **agent of interest** at preallocation x if a is qualified for a category *and* marginal for another. I denote by $\widetilde{A}^x = \{a \in A : C_Q^x(a) \cap C_M^x(a) \neq \varnothing\}$ the set of agents of interest, and for every category c and every status $S \in \{Q, M, U\}$, I denote by $\widetilde{A}_S(c) = \{a \in A : A_S^x(c) \cap \widetilde{A}^x\}$ the set of agents of interest whose status for c at x is S. As x satisfies Axioms 1-4, each category c has at most one marginal agent, i.e., $|A_M^x(c)| \leq 1$ (see Lemma A.14 for a formal statement). For every category c such that $A_M^x(c) \neq \varnothing$, let $a(c)$ be the agent who is marginal for c (i.e., $A_M^x(c) = \{a^x(c)\}$) and let $\widetilde{C}^x = \{C \in C : a^x(c) \in \widetilde{A}\}$ be the set of categories whose marginal agent is an agent of interest. Finally, it will prove useful to adjust the quota of each category $c \in \widetilde{C}^x$ by removing the capacity allocated to agents who are not of interest: $\widetilde{q}_c^x = q_c - \sum_{a \in A_Q^x(c) \setminus \widetilde{A}^x} x_{a,c}$.

5.2. **SRLP Algorithm.** The simultaneous reserve with linear programming (SRLP) algorithm is formally defined in Algorithm 2. I first present the main result and then provide intuition about how the SRLP algorithm works.

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

Theorem 4 implies that the number of rounds required to find the SR allocation is polynomial in $|A||C|$. As linear programming can be solved in polynomial time (Khachiyan, 1979), it follows that the SRLP algorithm calculates the SR allocation in polynomial time.

In both the SR and the SRLP algorithms, every agent is initially unqualified for every category. Throughout both algorithms, agents are allocated capacity and their status for some categories may change to marginal or qualified. As status changes are irreversible, there can be at most $2|A||C|$ of them overall.[6] The SRLP algorithm is identical to the SR algorithm until either it finds an allocation (in which case it terminates) or a round occurs without any change of status. In the latter case, the SRLP algorithm updates the demands by solving a linear program. This step

[6]See Lemmas A.12, A.13, and C.7 for formal statements that status changes are irreversible.

ALGORITHM 2: Simultaneous Reserve with Linear Programming (SRLP)

Initialization Set $x^0 = \mathbf{0}_{|A|\times|C|}$ and $d^0 = \mathbf{1}_{|A|}$.
Round $i \geq 1$:
Capacity Allocation Set $x^i = x(d^{i-1})$.
Termination If x^i is an allocation, terminate and output x^i.
Demand Adjustment If the status of at least one agent for one category has changed between x^{i-1} and x^i, set $d^i = d(x^i)$ and continue to Round $i+1$.
Linear Programming Solve the following linear program:

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

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

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

guarantees that, in the next round, either an allocation is found or a status has changed. Thus, a change of status occurs every second round until an allocation is found; hence, an allocation is found within $4|A||C|$ rounds.

The linear programming stage builds an allocation ξ that is identical to x^i for all agents that are not of interest. Agents of interest are allocated one unit in aggregate and (LP 1) determines how that unit is split among categories by maximizing the amount of capacity that agents receive from categories for which they are marginal, with the maximization subject to two constraints: an agent cannot be allocated more than her demand by any category and categories cannot allocate more than their quotas. More precisely, consider a Round i in which the SRLP algorithm uses linear programming and an agent-category pair $(a^{x^i}(c), c)$ with $c \in \widetilde{C}$. Agent $a^{x^i}(c)$ is allocated her demand from each category for which she is qualified and some amount of capacity from each category for which she is marginal: $|C_Q^{x^i}(a^{x^i}(c))|d_{a^{x^i}(c)}(\xi) + \sum_{c'\in C_M^{x^i}(a^{x^i}(c))} \xi_{a^{x^i}(c),c'} = 1$. Cardinal monotonicity requires that $\xi_{a^{x^i}(c),c} \leq d_{a^{x^i}(c)}(\xi)$, which by the previous equation gives the first constraint of (LP 1). Feasibility requires that $\xi_{a^{x^i}(c),c}+\sum_{a'\in C_Q^{x^i}(c)} d_{a'}(\xi) \leq \widetilde{q}_c^{x^i}$, which again by using the previous equation gives the second constraint of (LP 1). The solution to (LP 1) pins down an allocation ξ and the demand vector d^i is set to $d(\xi)$. By construction, the next round preallocation $x^{i+1} = x(d^i)$ is such that every agent who qualifies for at least one category at x^i is allocated exactly one unit in aggregate. Therefore, either x^{i+1} is an allocation or there is an agent a who is not qualified for any category at x^i and is allocated more than one unit in aggregate at x^{i+1}. In the latter case, a's demand at $d_a(x^{i+1})$ falls below what she receives from at least one category, so she qualifies for a category at x^{i+1}. Therefore, a status has changed between x^i and x^{i+1}.

5.3. **Example.** I illustrate the SRLP algorithm using Example 2. All statuses are the same at x^3 and x^4 (see Table 4), so the SRLP algorithm enters the linear programming stage in Round 4. The agents of interest are a_3 and a_1, who are marginal for c_2 and c_1, respectively, and qualified for the other two categories. The linear program is

$$\max_{(\xi_{a_3,c_1},\xi_{a_1,c_2})} \xi_{a_3,c_1} + \xi_{a_1,c_2} \quad \text{subject to}$$

$$\xi_{a_3,c_1} \leq 1/3,\ \xi_{a_1,c_2} \leq 1/3,\ \xi_{a_3,c_1} \leq 2/3 - (1-\xi_{a_1,c_2})/2,\ \xi_{a_1,c_2} \leq 2/3 - (1-\xi_{a_3,c_1})/2.$$

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 and the resulting demand vector is $d^4 = (1/3, 1/3, 1/3, 1)$. In Round 5, the capacity allocation stage produces $x^5 = \xi^{SR}$, and therefore the SRLP algorithm ends and outputs the SR allocation.

6. Conclusion

This paper proposes a new solution to reserve systems that processes reserve categories simultaneously rather than sequentially. 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 the same amount of capacity to every agent in aggregate. Finally, I show that the SR allocation can be computed in polynomial time.

I conclude by briefly describing four avenues for future research that arise from this paper. First, it may be possible to tweak the SR algorithm to handle ties in the priority profile. 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 could be combined with the deferred acceptance mechanism (or any other mechanism) such that it could 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 may consider any sharing rule in between, which would convexify the set of solutions provided by sequential allocation. Last, when a category does not allocate its entire quota due to a lack of eligible agents, it may be possible to increase efficiency by relaxing the category neutrality condition to allow eligible agents to receive more capacity from that category. Ultimately, I hope that the ideas presented in this paper provide a new perspective on reserve systems and pave the way toward developing and applying new solutions in a wide range of contexts.

References

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

Aygun, O., and I. Bó (2021): "College admission with multidimensional privileges: The Brazilian affirmative action case," *American economic journal: Microeconomics*, 13(3), 1–28.

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.

Aziz, H., and F. Brandl (2020): "The Vigilant Eating Rule: A General Approach for Probabilistic Economic Design with Constraints," *arXiv preprint arXiv:2008.08991.*

——— (2021): "Efficient, fair, and incentive-compatible healthcare rationing," *arXiv preprint arXiv:2102.04384*.

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 (2020): "Matching for the Israeli "Mechinot" Gap-Year Programs: Handling Rich Diversity Requirements," arXiv preprint arXiv:1905.00364v2.

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., 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 (2021): "Fair Allocation of Vaccines, Ventilators and Antiviral Treatments: Leaving No Ethical Value Behind in Health Care Rationing," Working paper, January 2021.

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

——— (2020b): "Constitutional implementation of vertical and horizontal reservations in India," Working paper, May 2020.

——— (2021): "Can Economic Theory be Informative for the Judiciary? Affirmative Action in India via Vertical and Horizontal Reservations," Working paper, January 2021.

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.

Appendix A. Properties of the SR Algorithm

Throughout this appendix, fix an agent a and a Round $i \geq 1$ of the SR algorithm.

Lemma A.1. *If $x_a^i > 1$, then there exists a unique d_a^i such that $\sum_{c\in C} \min\{d_a^i, x_{a,c}^i\} = 1$. Moreover, $d_a^i \in (0, \max_{c\in C}\{x_{a,c}^i\})$.*

Proof. If $d_a^i \leq 0$, then $\sum_{c\in C} \min\{d_a^i, x_{a,c}^i\} \leq 0 < 1$. If $d_a^i \geq \max_{c\in C}\{x_{a,c}^i\}$, then $\sum_{c\in C} \min\{d_a^i,\ x_{a,c}^i\} = \sum_{c\in C} x_{a,c}^i = x_a^i > 1$. The expression $\sum_{c\in C} \min\{d_a^i, x_{a,c}^i\}$ is continuous and strictly increasing in d_a^i at every $d_a^i < \max_{c\in C}\{x_{a,c}^i\}$. Therefore, there exists a unique value of $d_a^i \in (0, \max_{c\in C}\{x_{a,c}^i\})$ such that $\sum_{c\in C} \min\{d_a^i, x_{a,c}^i\} = 1$. $\square$

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

Proof. ($d^i_a \in [0,1]$) Suppose that $d^i_a \notin [0,1]$. Then, by definition, $x^i_a \geq 1$. 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, there is a category c such that $x^i_{a,c} \notin [0,1]$. Then, a is eligible for c; hence, $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$, 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) Fix a category c. If $x^i_{a,c} \notin [0,1]$, it was established above that $d^{i-1}_a \notin [0,1]$, a contradiction. It remains to show that $\sum_{a'\in A} x^i_{a',c} \leq q_c$. If $\sum_{a'\in A} x^i_{a',c} > q_c$, then there exists an agent b such that $x_{b,c} > 0$ and $x^i_{b,c} + \sum_{a'\in \hat{A}_{b,c}} x^i_{a',c} > q_c$. By definition, $x^i_{b,c} \leq d^{i-1}_{a'}$ for all $a' \in \hat{A}_{b,c}$ so $x^i_{a',c} + \sum_{a'\in \hat{A}_{b,c}} d^{i-1}_{a'} > q_c$. By definition, $x^i_{b,c} \leq \max\{q_c - \sum_{a'\in \hat{A}_{b,c}} d^{i-1}_{a'}, 0\}$; therefore, it must be that $x_{b,c} = 0$, a contradiction.

($d^i_a \geq 1/|C|$) If $x^i_a < 1$, then $d^i_a = 1$ 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 $d^i_a \geq 1/|C|$. □

Lemma A.3. *ξ^i is an allocation and $\xi^i_a = \min\{x^i_a, 1\}$.*

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 $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$. Then, $x^i_{a,c} \leq d^i_a$ for all $c \in C$ so $\xi^i_a = \sum_{c\in C} \min\{d^i_a, x^i_{a,c}\} = \sum_{c\in C} x^i_{a,c} = x^i_a = \min\{x^i_a, 1\}$.

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

(ξ^i is an allocation) By definition, for every 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 $\xi^i_{a,c} \in [0,1]$ and $\sum_{b\in A} \xi^i_{b,c} \leq \sum_{b\in A} x^i_{b,c} \leq q_c$ for every c. Finally, $\xi_a \leq 1$ follows from the above result that $\xi^i_a = \min\{x^i_a, 1\}$. □

Lemma A.4. *$\xi^i_a \geq \xi^{i-1}_a$ 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 $d^1_a \leq d^0_a$. (As a was fixed arbitrarily, the statement holds for every agent.) The remainder of the proof is by induction. Suppose that $\xi^{i-1}_a \geq \xi^{i-2}_a$ and $d^{i-1}_a \leq d^{i-2}_a$ (induction hypothesis). I show that $\xi^i_a \geq \xi^{i-1}_a$ and $d^i_a \leq d^{i-1}_a$.

($\xi_a^i \geq \xi_a^{i-1}$) Fix a category c. If a is not eligible for c, then by definition $x_{a,c}^{i-1} = x_{a,c}^i = 0$ so $\xi_{a,c}^i = \xi_{a,c}^{i-1} = 0$. If a is eligible for c, then by definition $x_{a,c}^{i-1} = \min\{d_a^{i-2}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-2}, 0\}\}$; 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 $\xi_{a,c}^{i-1} = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-2}, 0\}\}$. By definition, $x_{a,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}, 0\}\}$. The induction hypothesis implies that $d_{a'}^{i-1} \leq d_{a'}^{i-2}$ for all $a' \in \hat{A}_{a,c}$; therefore, the last two equations imply that $x_{a,c}^i \geq \xi_{a,c}^{i-1}$. As c was fixed arbitrarily, this holds for every category and we have $x_a^i \geq \xi_a^{i-1}$. Combining Lemma A.3 with that result and the fact that $\xi_a^{i-1} \leq 1$ yields $\xi_a^i = \min\{x_a^i, 1\} \geq \min\{\xi_a^{i-1}, 1\} = \xi_a^{i-1}$, as required.

($d_a^i \leq d_a^{i-1}$) *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 $\min\{x_a^i, 1\} = \xi_a^i \geq \xi_a^{i-1} = \min\{x_a^{i-1}, 1\}$. Combining this result with the case assumption that $x_a^i < 1$ yields $x_a^{i-1} < 1$. 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, $\max_{c \in C}\{x_{a,c}^i\} \leq d_a^{i-1}$, which means that $d_a^i \leq d_a^{i-1}$. □

Lemma A.5. *If $\xi_a^i < 1$, then $d_a^i = 1$ and, if $\xi_a^i = 1$, then $d_a^i = \max_{c \in C}\{\xi_{a,c}^i\}$.*

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

Case 2: $x_a^i = 1$. 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 all $c \in C$, $\xi_{a,c}^i = \min\{d_a^i, x_{a,c}^i\}$. Combining those two results implies that $\xi_{a,c}^i = x_{a,c}^i$ for all $c \in C$, and therefore $d_a^i = \max_{c \in C}\{\xi_{a,c}\}$.

Case 3: $x_a^i > 1$. 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 $c \in C$ and $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = \sum_{c \in C} x_{a,c}^i = x_a^i > 1$. However, by definition, $\sum_{c \in C} \min\{d_a^i, x_{a,c}^i\} = 1$, a contradiction. □

Lemma A.6. *$\xi_a^i = 1$ 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 $\xi_{a,c}^i = d_a^i$. □

For the remainder of this appendix, fix a category c.

Lemma A.7. *If either $x_{a,c}^i < d_a^{i-1}$ or $\xi_{a,c}^i < d_a^i$, then, $x_{a',c}^i = 0$ for every $a' \in \check{A}_{a,c}$.*

Proof. By definition, $\xi_{a,c}^i < d_a^i$ implies that $\xi_{a,c}^i = x_{a,c}^i$; hence, by Lemma A.4, $x_{a,c}^i < d_a^{i-1}$. Fix an agent $a' \in \check{A}_{a,c}$. If a' is not eligible for c, then $x_{a',c}^i = 0$ so I assume that a' (hence a) is eligible for c. By definition, $x_{a,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1}, 0\}\} < d_a^{i-1}$, which implies that $q_c - \sum_{\tilde{a} \in \hat{A}_{a,c}} d_{\tilde{a}}^{i-1} < d_a^{i-1}$. As $a \pi_c a'$, it follows that $q_c - \sum_{\tilde{a} \in \hat{A}_{a',c}} d_{\tilde{a}}^{i-1} < 0$ which by definition implies that $x_{a',c}^i = 0$. □

Lemma A.8. *If $\xi_{a,c}^i > 0$, then, for every $a' \in \hat{A}_{a,c}$, $x_{a',c}^i = d_{a'}^{i-1}$ and $\xi_{a',c}^i = d_{a'}^i$.*

Proof. By assumption, a is eligible for c and $q_c > \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}$. Fix an agent $a' \in \hat{A}_{a,c}$. The last inequality implies that $q_c > d_{a'}^{i-1} + \sum_{\tilde{a} \in \hat{A}_{a',c}} d_{\tilde{a}}^{i-1}$, which is equivalent to $q_c - \sum_{\tilde{a} \in \hat{A}_{a',c}} d_{\tilde{a}}^{i-1} > d_{a'}^{i-1}$. As a is eligible for c, so is a'; hence, by definition, $x_{a',c}^i = d_{a'}^{i-1}$. By Lemma A.4, $d_{a'}^i \leq d_{a'}^{i-1}$, hence $x_{a',c}^i \geq d_{a'}^i$. By definition, it follows that $\xi_{a',c}^i = \min\{d_{a'}^i, x_{a',c}^i\} = d_{a'}^i$. □

Lemma A.9. *If a is eligible for c and $x_{a,c}^i < d_a^{i-1}$, then $x_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} x_{a',c}^i = q_c$.*

Proof. Let b be the highest-priority agent such that $x_{b,c}^i < d_b^{i-1}$. That is, $x_{b,c}^i < d_b^{i-1}$ and, for every $a' \in \hat{A}_{b,c}$, $x_{a',c}^i = d_{a'}^{i-1}$. The assumption that $x_{a,c}^i < d_a^{i-1}$ ensures that b exists and either $b = a$ or $b \pi_c a$. Then, as a is eligible for c, so is b and we have $x_{b,c}^i = \min\{d_a^{i-1}, \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} d_{a'}^{i-1}, 0\}\}$. As $x_{b,c}^i < d_b^{i-1}$ and $x_{a',c}^i = d_{a'}^{i-1}$ for all $a' \in \hat{A}_{b,c}$, it follows that $x_{b,c}^i = \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} x_{a',c}^i, 0\}$. As x^i is a preallocation (by Lemma A.2), it must be that $\sum_{a' \in \hat{A}_{b,c}} x_{a',c}^{i-1} \leq q_c$; therefore, we can conclude that $x_{b,c}^i = q_c - \sum_{a' \in \hat{A}_{b,c}} x_{a',c}^i$ or, equivalently, $x_{b,c}^i + \sum_{a' \in \hat{A}_{b,c}} x_{a',c}^i = q_c$. On the one hand, as either $b = a$ or $b \pi_c a$, we have $x_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} x_{a',c}^i \geq x_{b,c}^i + \sum_{a' \in \hat{A}_{b,c}} x_{a',c}^i = q_c$. On the other hand, as x^i is an allocation, we have $x_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} x_{a',c}^i \leq q_c$. Combining the two statements yields $x_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} x_{a',c}^i = q_c$. □

Lemma A.10. *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 a is not eligible for c, then $x^i_{a,c} = \xi^i_{a,c} = 0$.

(x^i is nonwasteful) Suppose that a is eligible for c and $\sum_{b \in A} x^i_{b,c} < q_c$. It needs to be shown that $x^i_a \geq 1$. If $x^i_{a,c} < d^{i-1}_a$, then Lemma A.9 yields $x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x^i_{a',c} = q_c$, which contradicts the assumption that $\sum_{b \in A} x^i_{b,c} < q_c$. Therefore, we have $x^i_{a,c} = d^{i-1}_a$, in which case Lemma A.4 implies that $x^i_{a,c} = d^{i-1}_a \geq d^i_a$. Then, by definition, $\xi^i_{a,c} = d^i_a$ and, by Lemma A.6, $\xi^i_a = 1$; by definition, it follows that $x^i_a \geq \xi^i_a = 1$.

(x^i and ξ^i respect priorities) By Lemma A.3, $x^i_a < 1 \Leftrightarrow \xi^i_a < 1$ and, by definition, $x^i_{a,c} = 0 \Leftrightarrow \xi^i_{a,c} = 0$. It follows that x^i respects priorities if and only if ξ^i does; hence, it is enough to show that ξ^i respects priorities. If $\xi^i_a < 1$, Lemma A.6 implies that $\xi^i_{a,c} < d^i_a$ so, by Lemma A.7, $\xi^i_{a',c} = 0$ for every $a' \in \check{A}_{a,c}$; hence, ξ^i respects priorities.

(x^i is category neutral) Suppose that a is eligible for c and $x^i_{a,c} < \max_{c' \in C}\{x_{a,c'}\}$. 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 $x^i_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x^i_{a',c} = q_c$ so x^i is category neutral. □

Lemma A.11. *$z^i_{a,c} \in [0,1]$. 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 $c' \in C$, $x_{a',c'} = \xi_{a',c'}$ 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 every $a' \in A$ and $c' \in C$, we have $|z^i| = \sum_{a' \in A} \sum_{c' \in C} z^i_{a',c'} > 0$. □

Lemma A.12. *If $x^i_{a,c} < d^i_a$, then, for every Round $j \leq i$, $x^j_{a,c} \leq x^i_{a,c} < d^j_a$.*

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, $d^j_a > 0$. I assume henceforth that a is eligible for c. By Lemma A.4, the assumption that $x^i_{a,c} < d^i_a$ implies that $x^i_{a,c} < d^{i-1}_a$; therefore, by definition, $x^i_{a,c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}$. 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, $d^{i-2}_{a'} \geq d^{i-1}_{a'}$ for every $a' \in \hat{A}_{a,c}$; hence, we have $x^{i-1}_{a,c} \leq x^i_{a,c} < d^{i-1}_a$. By induction, the statement holds for every $j < i - 1$. □

Lemma A.13. *If $x^i_{a,c} \geq d^i_a$, then, for every $j > i$, $x^j_{a,c} = d^{j-1}_a \geq d^j_a$.*

Proof. By the definition of, $x^i_{a,c}$, 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 $\max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d^i_{a'}, 0\} \geq d^i_a$, which by the definition of $x^{i+1}_{a,c}$ implies that $x^{i+1}_{a,c} = d^i_a$. 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. $\square$

For the next two lemmas, fix a preallocation x that satisfies Axioms 1-4.

Lemma A.14. *(i) $x_a \geq 1$ if and only if $C^x_Q(a) \neq \varnothing$,*
(ii) $|A^x_M(c)| \leq 1$,
(iii) For any $a_1 \in A^x_Q(c)$, $a_2 \in A^x_M(c)$, and $a_3 \in A^x_U(c)$, $a_1 \pi_c a_2 \pi_c a_3$.

Proof. (i) If $x_a < 1$, then $d_a(x) = 1$ so $x_{a,c} < d_a(x)$ for every $c \in C$, hence $C^x_Q(a) = \varnothing$. If $x_a = 1$, then $d_a(x) = \max_{c \in C}\{x_{a,c}\}$. If $x_a > 1$, then $\sum_{c \in C} \min\{d_a(x), x_{a,c}\} = 1$ so $d_a(x) < \max_{c \in C}\{x_{a,c}\}$. If follows that $x_a \geq 1$ implies that there exists $c \in C$ such that $x_{a,c} \geq d_a(x)$, hence $C^x_Q(a) \neq \varnothing$.

((ii) & (iii)) Fix an agent a. I show that $a \in A^x_M(c) \cup A^x_U(c)$ implies $b \in A^x_U(c)$ for every $b \in \check{A}_{a,c}$ and that $a \in A^x_Q(c) \cup A^x_M(c)$ implies $b \in A^x_Q(c)$ for every $b \in \hat{A}_{a,c}$. Taken together, these statements imply (ii) and (iii). Suppose that $a \in A^x_M(c) \cup A^x_U(c)$ and consider an agent $b \in \check{A}_{a,c}$. It needs to be shown that $x_{b,c} = 0$. If a is not eligible for c, then neither is b so $x_{b,c} = 0$ so I assume for the rest of the argument that a is eligible for c. If $x_a < 1$, then Axiom 3 implies that $x_{b,c} = 0$. If $x_a \geq 1$, then by (i) there exists a category c' such that $x_{a,c'} \geq d_a(x) > x_{a,c}$ so Axiom 4 implies that $x_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x_{a',c} = q_c$, hence $x_{b,c} = 0$. Suppose that $a \in A^x_Q(c) \cup A^x_M(c)$ and consider an agent $b \in \check{A}_{a,c}$. It needs to be shown that $x_{b,c} \geq d_b(x)$. As $x_{a,c} > 0$, Axiom 3 implies that $x_b = 1$. Then, by (i), there exists $c' \in C$ such that $x_{b,c'} \geq d_b(x)$. As $b \pi_c a$, $x_{b,c} + \sum_{a' \in \hat{A}_{b,c}} x_{b,c} < q_c$ and Axiom 4 implies that $x_{b,c} \geq x_{b,c'} \geq d_b(x)$. $\square$

Lemma A.15. *If a is eligible for c and $x_{a,c} < d_a(x)$, then $x_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x_{a',c} = q_c$.*

Proof. If $x_a < 1$, then Axioms 2 and 3 imply that $x_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x_{a',c} = q_c$. If $x_a = 1$, then by Lemma A.14(i) there exists a category c' such that $x_{a,c'} \geq d_a(x)$. It follows that $x_{a,c} < \max_{c' \in C}\{x_{a,c'}\}$; hence, Axiom 4 implies that $x_{a,c} + \sum_{a' \in \hat{A}_{a,c}} x_{a',c} = q_c$. $\square$

For the remaining two lemmas, fix an allocation ξ that satisfies Axioms 1-4.

Lemma A.16. $\xi_a \geq \xi^i_a$ *and* $\xi_{a,c} \leq d^i_a$.

Proof. By definition, $\xi_{a,c} \leq 1 = d^0_a$. The remainder of the proof is by induction. I assume that $\xi_{a,c} \leq d^{i-1}_a$ (induction hypothesis) and show that $\xi_a \geq \xi^i_a$ and $\xi_{a,c} \leq d^i_a$.

($\xi_a \geq \xi_a^i$) As the result holds trivially when $\xi_a = 1$, I focus on the case in which $\xi_a < 1$. I show that $\xi_{a,c} \geq \xi_{a,c}^i$, which is sufficient since c was picked arbitrarily. As this result holds trivially when $\xi_{a,c}^i = 0$, I focus on the case in which $\xi_{a,c}^i > 0$. As ξ^i complies with eligibility requirements (Lemma A.10), it follows that a is eligible for c. By definition, $\xi_{a,c}^i \leq \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}, 0\}$; hence, the assumption that $\xi_{a,c}^i > 0$ implies that $\xi_{a,c}^i \leq q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^{i-1}$. By the induction hypothesis, $\xi_{a',c} \leq d_{a'}^{i-1}$ for every $a' \in \hat{A}_{a,c}$; therefore, the previous inequality implies that $\xi_{a,c}^i + \sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} \leq q_c$. By assumption, ξ is nonwasteful and $\xi_a < 1$; hence, as a is eligible for c, we have $\xi_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} = q_c$. Combining the last two results yields $\xi_{a,c} \geq \xi_{a,c}^i$.

($\xi_{a,c} \leq d_a^i$) Suppose that $\xi_{a,c} > d_a^i$. Then, $d_a^i < 1$ so Lemma A.5 implies that $d_a^i = \max_{c' \in C}\{\xi_{a,c'}^i\}$ and $\xi_a^i = 1$. Combining these results and recalling that $\xi_a \geq \xi_a^i$ yields $\xi_{a,c}^i \leq \max_{c' \in C}\{\xi_{a,c'}^i\} = d_a^i < \xi_{a,c} \leq \max_{c' \in C}\{\xi_{a,c'}\}$ and $\xi_a^i = \xi_a = 1$. Then, there exists $\tilde{c} \in C$ such that $\xi_{a,\tilde{c}} < \xi_{a,\tilde{c}}^i$. Moreover, by definition, $\xi_{a,\tilde{c}}^i \leq \max_{c' \in C}\{\xi_{a,c'}^i\}$ so the previous result that $\max_{c' \in C}\{\xi_{a,c'}^i\} < \max_{c' \in C}\{\xi_{a,c'}\}$ implies that $\xi_{a,\tilde{c}} < \max_{c' \in C}\{\xi_{a,c'}\}$. As ξ is category neutral, it follows that $\xi_{a,\tilde{c}} + \sum_{a' \in \hat{A}_{a,\tilde{c}}} \xi_{a',\tilde{c}} = q_{\tilde{c}}$. By definition, $\xi_{a,\tilde{c}}^i \leq \max\{q_{\tilde{c}} - \sum_{a' \in \hat{A}_{a',\tilde{c}}} d_{a'}^{i-1}, 0\}$. As $\xi_{a,\tilde{c}}^i > \xi_{a,\tilde{c}}$, we have that $\xi_{a,\tilde{c}}^i > 0$, and therefore $\xi_{a,\tilde{c}}^i \leq q_{\tilde{c}} - \sum_{a' \in \hat{A}_{a',\tilde{c}}} d_{a'}^{i-1}$. By the induction hypothesis, $\xi_{a',\tilde{c}} \leq d_{a'}^{i-1}$ for every $a' \in \hat{A}_{a,\tilde{c}}$; hence, we have $\xi_{a,\tilde{c}}^i + \sum_{a' \in \hat{A}_{a',\tilde{c}}} \xi_{a',\tilde{c}} \leq q_{\tilde{c}}$. It follows that $\xi_{a,\tilde{c}}^i \leq \xi_{a,\tilde{c}}$, a contradiction. □

Lemma A.17. *If a is eligible for c, then $\xi_{a,c} = \min\{d_a(\xi), \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\xi), 0\}\}$.*

Proof. If $\xi_{a,c} = d_a(\xi)$, then $\xi_{a',c} = d_{a'}(\xi)$ for every $a' \in \hat{A}_{a,c}$ by Lemma A.14(iii) so $d_a(\xi) + \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\xi) \leq q_c$. If $0 < \xi_{a,c} < d_a(\xi)$, then again $\xi_{a',c} = d_{a'}(\xi)$ for every $a' \in \hat{A}_{a,c}$ by Lemma A.14(iii). Therefore, by Lemma A.15, $\xi_{a,c} + \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\xi) = q_c$. If $\xi_{a,c} = 0$, then $\sum_{a' \in \hat{A}_{a,c}} \xi_{a',c} = q_c$ by Lemma A.15 and $\xi_{a',c} \leq d_{a'}(\xi)$ for every $a' \in \hat{A}_{a,c}$ by definition since ξ is an allocation. Therefore, $q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\xi) \leq 0$. □

Appendix B. Proof of the Results from Section 4

Proof of Theorem 1. Recall that, for every Round i, $|z^i| = \sum_{a \in A} \sum_{c \in C} z_{a,c}^i$. I use analogous notation for x^i and ξ^i: $|x^i| = \sum_{a \in A} \sum_{c \in C} x_{a,c}^i$ and $|\xi^i| = \sum_{a \in A} \sum_{c \in C} \xi_{a,c}^i$. I first show that the excess demand converges to zero. Fixing a Round i of the SR algorithm, I show that $|z^{i+1}| \leq |z^i|$ and $|z^i| \leq |A|(|C| - 1)/i)$.

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

c be a category such that $\sum_{a\in A} x^i_{a,c} < q_c$. If agent a is eligible for c, then, by Lemma A.9, $x^i_{a,c} = d^i_a$. Otherwise, $x^i_{a,c} = 0$ as x^i satisfies Axiom 1 (Lemma A.10). Letting $A_c = \{a \in A : a\pi_c\emptyset\}$, it follows that $\sum_{a\in A} x^i_{a,c} = \sum_{a\in A_c} d^{i-1}_a$. As categories do not allocate more than their quotas at x^i (Lemma A.2), it can be concluded that $|x^i| = \sum_{c'\in C} x^i_{a,c'} = \sum_{c'\in C} \min\{q_{c'}, \sum_{a\in A_{c'}} d^{i-1}_a\}$. Analogous reasoning yields $|x^{i+1}| = \sum_{c'\in C} \min\{q_{c'}, \sum_{a\in A_{c'}} d^i_a\}$; therefore, 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. By definition, $\sum_{j=1}^i z^j_{a,c} = \sum_{j=1}^i (x^j_{a,c} - \xi^j_{a,c}) = \sum_{j=1}^i (x^j_{a,c} - \min\{d^j_a, x^j_{a,c}\}) = \sum_{j=1}^i \max\{x^j_{a,c} - d^j_a, 0\}$. If $x^j_{a,c} \leq d^j_a$ for all $j = 1, \ldots, i$, then $\sum_{j=1}^i z^j_{a,c} = 0$. Otherwise, let $k = 1, \ldots, i$ be the first round such that $x^k_{a,c} > d^k_a$ (hence, $x^j_{a,c} \leq d^j_a$ for all $j = 1, \ldots, k-1$). Then, by Lemma A.13, $x^j_{a,c} = d^{j-1}_a$ for all $j = k+1, \ldots, i$, and therefore $\sum_{j=1}^i z^j_{a,c} = \sum_{j=1}^i \max\{x^j_{a,c} - d^j_a, 0\} = \sum_{j=k}^i (x^j_{a,c} - d^j_a) = x^k_{a,c} - d^i_a$. As $x^k_{a,c} \leq 1$ and $d^i_a \geq 1/|C|$ (Lemma A.2), it follows that $\sum_{j=1}^i z^j_{a,c} \leq 1 - 1/|C|$. As the latter bound is the larger one of the two cases, it holds for every agent-category pair. Therefore, we have $\sum_{j=1}^i |z^j| \leq |A||C|(1 - 1/|C|) = |A|(|C|-1)$. As the excess demand decreases throughout the algorithm, it follows that $i|z^i| \leq \sum_{j=1}^i |z^j| \leq |A|(|C|-1)$, which implies that $|z^i| \leq |A|(|C|-1)/i$.

The last result established that $\lim_{i\to\infty} |z^i| = 0$. As every element of z^i is by definition weakly positive, it follows that $\lim_{i\to\infty} z^i = \mathbf{0}$; therefore, $\lim_{i\to\infty} x^i = \lim_{i\to\infty} \xi^i$ and $\xi^{SR} = \lim_{i\to\infty} \xi^i$. I use the latter result to show that ξ^{SR} is an allocation.

($\xi^{SR}_{a,c} \in [0,1]$) *Case 1:* $\xi^i_{a,c} < d^i_a$ for every $i \geq 1$. By definition, the case assumption yields $x^i_{a,c} < d^i_a$, which by Lemma A.12 implies that $x^j_{a,c} \leq x^i_{a,c} < d^j_a$ for every $j \leq i$. By definition, it follows that the series $\{\xi^i_{a,c}\}_{i=1}^\infty$ is weakly increasing. By Lemma A.3, that series is also bounded; therefore, the Monotone Convergence Theorem implies that $\lim_{i\to\infty} \xi^i_{a,c}$ is equal to the series' supremum. Again by Lemma A.3, 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 yields $x^i_{a,c} \geq d^i_a$; hence, Lemma A.13 implies that $x^j_{a,c} \geq d^j_a$ for every $j \geq i$. Again by definition, we have $\xi^j_{a,c} = d^j_a$ 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.

($\sum_{a'\in A}\xi^{SR}_{a',c}\le q_c$) As $\lim_{i\to\infty}\xi^i=\xi^{SR}$, we have $\lim_{i\to\infty}(\sum_{a'\in A}\xi^i_{a',c})=\sum_{a'\in A}\xi^{SR}_{a',c}$. Then, the series $\{\sum_{a'\in A}\xi^i_{a',c}\}_{i=1}^{\infty}$ converges to a finite number and, as ξ^i is an allocation for every i (Lemma A.3), is bounded above by q_c. Therefore, the series converges to a number no greater than q_c and $\sum_{a'\in A}\xi^{SR}_{a',c}=\lim_{i\to\infty}(\sum_{a'\in A}\xi^i_{a',c})\le q_c$.

($\xi^{SR}_a\le 1$) As $\lim_{i\to\infty}\xi^i=\xi^{SR}$, we have $\lim_{i\to\infty}(\xi^i_a)=\xi^{SR}_a$. Then, the series $\{\xi^i_a\}_{i=1}^{\infty}$ converges to a finite number and, as ξ^i is an allocation for every i (Lemma A.3), is bounded above by 1. Therefore, the series converges to $\xi^{SR}_a=\lim_{i\to\infty}(\xi^i_a)\le 1$. □

Proof of Proposition 1. This proof uses the notation and terminology introduced in Section 5.1. The main part of the proof consists of showing the following.

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

Proof. Let a be an agent who is not marginal for either category at x^i. If a is unqualified for a category, then she receives either 0 or 1 from the other, as otherwise she would be marginal for it. It follows that $d^{i-1}_a=d^i_a=d^{i+1}_a=1$, $x^{i-1}_a=x^i_a=x^{i+1}_a\in\{0,1\}$, and, for each c, $x^{i-1}_{a,c}=x^i_{a,c}=x^{i+1}_{a,c}\in\{0,1\}$. If instead a is qualified for both categories, then, as $d^{i-1}_a\ge 0.5$ by Lemma A.2, it must be that $x^{i-1}_{a,c}\ge 0.5$ for each c. By definition, it follows that $d^{i-1}_a=0.5$. By Lemmas A.2 and A.4, we then have $d^{i-1}_a=d^i_a=d^{i+1}_a=0.5$, which, by Lemma A.13, yields $x^i_{a,c}=x^{i+1}_{a,c}=0.5$ for each c. We conclude that, for every agent a who is not marginal for either category, and for every category c,

$$x^i_{a,c}=x^{i+1}_{a,c}\in\{0,0.5,1\},\quad x^i_a=x^{i+1}_a\in\{0,1\},\quad d^{i-1}_a=d^i_a=d^{i+1}_a\in\{0.5,1\}. \tag{1}$$

By (1), every agent a who is not marginal for either category is allocated at most one unit in aggregate. It remains to show that this is also the case for those agents who are marginal for at most one category. As there are only two categories, by Lemma A.14 there are at most two agents of interest at x^i. I consider three cases.

Case 0: zero agents of interest. Let b be an agent who is marginal for a category. We need to show that $x^{i+1}_b\le 1$. If b is only marginal for one category, then by the case assumption she is unqualified for the other and $x^{i+1}_b\le 1$. If b is marginal for both categories, then $x^{i+1}_{b,c}\le d^{i+1}_b$ for each c; hence, $x^{i+1}_b<1$ by Lemmas A.3 and A.6.

Case 1: one agent of interest. Let agent a_1 be marginal for category c_1 and qualified for category c_2. If c_2 has a marginal agent a_2, by the case assumption a_2 is unqualified for c_1 so $x_{a_2}^{i+1} \leq 1$. Therefore, it remains to show that $x_{a_1}^{i+1} \leq 1$. As a_1 is marginal for c_1, $x_{a_1,c_1}^{i-1} < d_{a_1}^{i-1}$ so, by Lemma A.9, $x_{a_1,c_1}^{j} = q_{c_1} - \sum_{a' \in \hat{A}_{a_1,c_1}} x_{a',c_1}^{j}$ for $j = i, i+1$. As, by (1), $x_{a',c_1}^{i} = x_{a',c_1}^{i+1}$ for every $a' \in \hat{A}_{a_1,c_1}$, it follows that $x_{a_1,c_1}^{i} = x_{a_1,c_1}^{i+1}$. As a_1 is qualified for c_2, $x_{a_1,c_2}^{i} \geq d_{a_1}^{i}$ so Lemma A.13 implies that $x_{a_1,c_2}^{i+1} = d_{a_1}^{i}$. Moreover, by Lemma A.6, $\xi_a^i = 1$, which by definition is equivalent to $\min\{x_{a_1,c_1}^{i}, d_{a_1}^{i}\} + \min\{x_{a_1,c_2}^{i}, d_{a_1}^{i}\} = 1$. Then, the assumption that $x_{a_1,c_1}^{i} < d_{a_1}^{i} \leq x_{a_1,c_2}^{i}$ implies that $x_{a_1,c_1}^{i} + d_{a_1}^{i} = 1$, which combined with our two previous results yields $x_{a_1,c_1}^{i+1} + x_{a_1,c_2}^{i+1} = 1$.

Case 2: two agents of interest. Let a_1 (respectively a_2) be the agent who is marginal for c_1 (c_2) and qualified for c_2 (c_1). I assume w.l.o.g. that $x_{a_1}^{i+1} \geq x_{a_2}^{i+1}$ and show that $x_{a_1}^{i+1} \leq 1$. By Lemma A.9, $x_{a_1,c_1}^{i+1} + \sum_{a' \in \hat{A}_{a_1,c_1}} x_{a',c_1}^{i+1} = q_{c_1}$ so $\sum_{a \in A} x_{a,c_1}^{i+1} = q_{c_1}$ (since x^{i+1} is a preallocation). Analogous reasoning for c_2 yields $\sum_{a \in A} x_{a,c_2}^{i+1} = q_{c_2}$; hence, it can be concluded that $\sum_{a \in A} x_a^{i+1} = q$. As q is an integer and, by (1), $x_a^{i+1} \in \{0, 1\}$ for every $a \neq a_1, a_2$, $x_{a_1}^{i+1} + x_{a_2}^{i+1}$ must also be an integer. Suppose that $x_{a_1}^{i+1} > 1$. As $x_{a_2,c_1}^{i+1} \geq d_{a_2}^{i+1}$, Lemmas A.3 and A.6 imply that $x_{a_2}^{i+1} \geq 1$; hence, as $x_{a_1}^{i+1} + x_{a_2}^{i+1}$ is an integer, $x_{a_1}^{i+1} + x_{a_2}^{i+1} \geq 3$. It follows that $x_{a_1}^{i+1} \geq 1.5$; hence $x_{a_1,c_1}^{i+1}, x_{a_1,c_2}^{i+1} \geq 0.5$. By definition, we have that $d_{a_1}^{i+1} = 0.5$ so $x_{a_1,c_1}^{i+1} \geq d_{a_1}^{i+1}$, a contradiction. □

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; hence, at least two statuses change. Thereafter, by Claim 1, every second round either a status changes or an allocation is found. By Lemmas A.12 and A.13, status changes are irreversible, therefore there can be at most $4|A|$ of them throughout the SR algorithm. It follows that an allocation is found by Round $8|A| - 1$. □

Proof of Theorem 2. In every Round i, ξ^i satisfies Axioms 1 and 3 and x^i satisfies Axioms 1-4 by Lemma A.10. As was shown in the proof of Theorem 1, $\xi^{SR} = \lim_{i \to \infty} \xi^i = \lim_{i \to \infty} x^i$. Arbitrarily fixing an agent-category pair (a, c) and using those two results, I show that ξ^{SR} satisfies Axioms 1-4.

(Axiom 1) If a is not eligible for c, then $x_{a,c}^i = 0$ in every Round i as x^i satisfies Axiom 1. Therefore, $\xi_{a,c}^{SR} = \lim_{i \to \infty} x_{a,c}^i = 0$.

(Axiom 2) If a is eligible for c and $\sum_{b \in A} \xi_{b,c}^{SR} < q_c$, then $\lim_{i \to \infty} \sum_{b \in A} x_{b,c}^i < q_c$ so there exists a Round j such that, for all $i \geq j$, $\sum_{b \in A} x_{b,c}^i < q_c$. As x^i is nonwasteful and a is eligible for c, $x_a^i \geq 1$ so $\xi_a^i = 1$ by Lemma A.3. Then, $\xi_a^{SR} = \lim_{i \to \infty} \xi_a^i = 1$.

(Axiom 3) Suppose that $\xi_a^{SR} < 1$ and consider a lower-priority agent $a' \in \check{A}_{a,c}$. By Lemma A.4, for every Round i, $\xi_a^i \leq \xi_a^{SR} < 1$. As ξ^i respects priorities, it follows that $\xi_{a',c}^i = 0$, and therefore $\xi_{a',c}^{SR} = \lim_{i\to\infty} \xi_{a',c}^i = 0$.

(Axiom 4) If a is eligible for c and $\xi_{a,c}^{SR} < \max_{c'\in C}\{\xi_{a,c'}^{SR}\}$, then $\lim_{i\to\infty} x_{a,c}^i < \lim_{i\to\infty} \max_{c'\in C}\{x_{a,c'}^i\}$. Therefore, there exists a Round j such that, for all $i \geq j$, $x_{a,c}^i < \max_{c'\in C}\{x_{a,c'}^i\}$. As x^i is category neutral, $x_{a,c}^i + \sum_{a'\in \hat{A}_{a,c}} x_{a',c}^i = q_c$; therefore, $\xi_{a,c}^{SR} + \sum_{a'\in \hat{A}_{a,c}} \xi_{a',c}^{SR} = \lim_{i\to\infty}(x_{a,c}^i + \sum_{a'\in \hat{A}_{a,c}} x_{a',c}^i) = q_c$. □

Proof of Theorem 3. Consider an allocation ξ^* that satisfies Axioms 1-4 and suppose that, for some agent b, $\xi_b^* \neq \xi_b^{SR}$. By Lemma A.16, $\xi_a^* \geq \xi_a^i$ for every agent a and Round i, which implies that $\xi_a^* \geq \lim_{i\to\infty} \xi_a^i = \xi_a^{SR}$. It follows that $\xi_b^* > \xi_b^{SR}$, and therefore $|\xi^*| > |\xi^{SR}|$. Consequently, there exists a category c such that $\sum_{a\in A} \xi_{a,c}^* > \sum_{a\in A} \xi_{a,c}^{SR}$. By definition, $\sum_{a\in A} \xi_{a,c}^* \leq q_c$ so $\sum_{a\in A} \xi_{a,c}^{SR} < q_c$. By Theorem 1, $\lim_{i\to\infty} \sum_{a\in A} x_{a,c}^i < q_c$; therefore, there exists a Round j such that $\sum_{a\in A} x_{a,c}^i < q_c$ for every $i \geq j$. Then, by Lemma A.9, for every agent a who is eligible for c, we have $x_{a,c}^i = d_a^{i-1}$. By definition, $\xi_{a,c}^i = \min\{d_a^i, x_{a,c}^i\} = \min\{d_a^i, d_a^{i-1}\}$ so, by Lemma A.4, $\xi_{a,c}^i = d_a^i$. As $\xi_{a,c}^* \leq d_a^i$ by Lemma A.16 and ξ^* satisfies Axiom 1, we conclude that $\xi_{a,c}^* \leq \xi_{a,c}^i$ for every a and every $i \geq j$, which implies that $\xi_{a,c}^* \leq \xi_{a,c}^{SR}$ for every a. It follows that $\sum_{a\in A} \xi_{a,c}^* \leq \sum_{a\in A} \xi_{a,c}^{SR}$, a contradiction. □

Proof of Proposition 2. 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.

(Axiom 1) Let the quotas and priorities be $q_{c_1} = 2$, $q_{c_2} = 1$ $\pi_{c_1} : a_1, \emptyset, a_3, a_2$, and $\pi_{c_2} : a_2, \emptyset, a_3, a_1$. The SR allocation ξ^{SR} is such that $\xi_{a_1,c_1}^{SR} = \xi_{a_2,c_2}^{SR} = 1$ and all other elements are 0. The alternative allocation ξ such that $\xi_{a_3,c_1} = 1$ and all other elements are identical to ξ^{SR} is not SR equivalent and satisfies Axioms 2-4.

(Axiom 2) Let the quotas and priorities be $q_{c_1} = 2$, $q_{c_2} = 1$, $\pi_{c_1} : a_1, a_3, \emptyset, a_2$, $\pi_{c_2} : a_2, \emptyset, a_3, a_1$. The SR allocation ξ^{SR} is such that $\xi_{a_1,c_1}^{SR} = \xi_{a_2,c_2}^{SR} = \xi_{a_3,c_1}^{SR} = 1$ and all other elements are 0. The alternative allocation ξ such that $\xi_{a_3,c_1} = 0$ and all other elements are identical to ξ^{SR} is not SR equivalent and satisfies Axioms 1, 3, and 4.

(Axiom 3) Let the quotas and priorities be $q_{c_1} = 1$, $q_{c_2} = 1$, $\pi_{c_1} : a_1, a_3, a_2, \emptyset$, $\pi_{c_2} : a_1, a_2, a_3, \emptyset$. The SR allocation ξ^{SR} is such that $\xi_{a_2,c_1}^{SR} = \xi_{a_3,c_2}^{SR} = 0$ and all other elements are 1/2. The alternative allocation ξ such that $\xi_{a_1,c_1} = \xi_{a_1,c_2} = 1$ and all other elements are 0 is not SR equivalent and satisfies Axioms 1, 2, and 4.

(Axiom 4) Let the quotas and priorities be identical to the previous example. The allocation ξ such that $\xi_{a_1,c_1} = \xi_{a_2,c_2} = 1$ and all other elements are 0 is not SR equivalent and satisfies Axioms 1-3. $\square$

Proof of Proposition 3. This proof uses some of the notation and terminology introduced in Section 5.1. For some Round i of the SR algorithm, suppose that $d(\xi^*) \leq d^{i-1}$ (induction hypothesis). I show that $d(\xi^*) \leq d^i$. Fix an agent a and suppose that $d_a(\xi^*) > d_a^i$. For every $c \in C_Q^{\xi^*}(a)$, $\xi^*_{a,c} = d_a(\xi^*)$ so the assumption that $d_a(\xi^*) > d_a^i$ yields $\min\{d_a^i, x^i_{a,c}\} < \xi^*_{a,c}$. For every $c \in C_M^{\xi^*}(a) \cup C_U^{\xi^*}(a)$, $\xi^*_{a,c} < d_a(\xi^*)$ so Lemma A.17 yields $\xi^*_{a,c} = \max\{q_c - \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\xi^*), 0\}$. By definition, $x^i_{a,c} \leq q_c - \sum_{a' \in \hat{A}_{a,c}} d_a^{i-1}$ so the induction hypothesis implies that $x^i_{a,c} \leq \xi^*_{a,c}$, and therefore $\min\{d_a^i, x^i_{a,c}\} \leq \xi^*_{a,c}$. Combining the last two results yields $\sum_{c\in C} \min\{d_a^i, x^i_{a,c}\} < \sum_{c \in C} \xi^*_{a,c} = \xi^*_a$. As ξ^* is an allocation, $\xi^*_a \leq 1$ so $\min\{d_a^i, x^i_{a,c}\} < 1$. It follows that $x^i_a \leq 1$; however, in that case $d_a^i = 1$, which contradicts the assumption that $d_a(\xi^*) > d_a^i$. As $d^0 = \mathbf{1}$, the induction hypothesis holds in Round 1 so the preceding argument implies that $d(\xi^*) \leq d^i$ in every Round i of the SR algorithm. It follows that the series $\{d^i\}_{i=1}^{\infty}$ is bounded below by $d(\xi^*)$. As the series is decreasing (by Lemma A.4), the Monotone Convergence Theorem implies that $\lim_{i\to\infty} \xi^i_{a,c}$ is equal to the series' infimum, which cannot be smaller than $d(\xi^*)$; therefore, $d(\xi^*) \leq d(\xi^{SR})$. If $d(\xi^*) = d(\xi^{SR})$, then Lemma A.17 implies that $\xi^* = \xi^{SR}$, a contradiction. We conclude that $d(\xi^*) < d(\xi^{SR})$. $\square$

Proof of Proposition 4. Let a be an agent such that $0 < \xi_a^{SR} < 1$. By Lemma A.14(i), a is not qualified for any category at ξ^{SR}; hence, by assumption, a is marginal for a category. By Lemma A.14(ii), there are at most $|C|$ agents in that situation. $\square$

Proof of Proposition 5. This proof uses some of the notation and terminology introduced in Section 5.1. Construct a rationing problem $\widetilde{R} = (A, C, (\tilde{\pi}_c)_{c\in C}, (q_c)_{c\in C})$, which is identical to R except that a's priority rank drops by one rank for some category $\tilde{c}$. Formally, consider $\tilde{a} \in A \cup \{\emptyset\}$ such that $a\pi_c\tilde{a}$ and there is no $a' \in A \cup \{\emptyset\}$ with $a\pi_c a'\pi_c\tilde{a}$. The priority profile $(\tilde{\pi}_c)_{c\in C}$ is identical to $(\pi_c)_{c\in C}$ except that a and $\tilde{a}$'s priorities for $\tilde{c}$ are reversed: $\tilde{a}\tilde{\pi}_c a$ and, for every tuple $(b, \tilde{b}, c) \neq (a, \tilde{a}, \tilde{c})$, $b\tilde{\pi}_c b'$ if and only if $b\pi_c b'$. Denote by $\widetilde{\xi}^{SR}$ the SR allocation of $\widetilde{R}$. As $\widehat{R}$ can be constructed from R through a series of priority reversals, by induction it is sufficient to show that $\xi_a^{SR} \geq \widetilde{\xi}_a^{SR}$. This result holds trivially if $\xi_a^{SR} = 1$ so I assume throughout that $\xi_a^{SR} < 1$. By construction, every agent $b \neq a$ is eligible for the same subset of categories in R and $\widetilde{R}$, which I denote by $C_E(b) = \{c \in C : b\pi_c\emptyset\} = \{c \in C : b\tilde{\pi}_c\emptyset\}$. I first show that $d(\widetilde{\xi}^{SR}) \leq d(\xi^{SR})$ and then use this result to show that $\xi_a^{SR} \geq \widetilde{\xi}_a^{SR}$.

For every $i \geq 1$, denote by x^i and $\tilde{x}^i$ (d^i and $\tilde{d}^i$) the preallocation (demand vector) obtained in Round i of the SR algorithm over R and $\widetilde{R}$, respectively. Fixing a Round i, suppose that $\tilde{d}^{i-1} \leq d^{i-1}$ (induction hypothesis). I show that $\tilde{d}^i \leq d^i$. Fix an agent $b \neq a, \tilde{a}$ and a category $c \in C_E(b)$. As b's priority for c is the same in R and $\widetilde{R}$, $\{a' \in A : a'\tilde{\pi}_c b\} = \{a' \in A : a'\pi_c b\} = \hat{A}_{b,c}$; therefore, $\tilde{x}^i_{b,c} = \min\{\tilde{d}^{i-1}_b, \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} \tilde{d}^{i-1}_{a'}, 0\}\}$ and $x^i_{b,c} = \min\{d^{i-1}_b, \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} d^{i-1}_{a'}, 0\}\}$. If $x^i_b < 1$, then $\tilde{d}^i_b \leq d^i_b = 1$. If $\tilde{x}^i_b < 1$, then by Lemma A.6 $\tilde{x}^i_{b,c} < \tilde{d}^i_a$; hence, by Lemma A.4 and the induction hypothesis, $\tilde{x}^i_{b,c} \geq x^i_{b,c}$. As this holds for every category, $x^i_b \leq \tilde{x}^i_b < 1$ so $\tilde{d}^i_b \leq d^i_b = 1$. If $\tilde{x}^i_b, x^i_b \geq 1$, then $\sum_{c \in C_E(b)} \min\{\tilde{d}^i_b, \tilde{x}^i_{a,c}\} = \sum_{c \in C_E(b)} \min\{d^i_b, x^i_{a,c}\} = 1$. By Lemma A.4, it follows that $\sum_{c \in C_E(b)} \min\{\tilde{d}^i_b, \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} \tilde{d}^{i-1}_{a'}, 0\}\} = \sum_{c \in C_E(b)} \min\{d^i_b, \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} d^{i-1}_{a'}, 0\}\}$; therefore, the induction hypothesis implies that $\tilde{d}^i_b \leq d^i_b$. The argument for $\tilde{d}^i_{\tilde{a}} \leq d^i_{\tilde{a}}$ is almost analogous, the only difference is that $\tilde{a}\tilde{\pi}_{\tilde{c}} a$, hence $\tilde{x}^i_{\tilde{a},\tilde{c}} = \min\{\tilde{d}^{i-1}_{\tilde{a}}, \max\{q_{\tilde{c}} - \sum_{a' \in \hat{A}_{\tilde{a},\tilde{c}} \setminus \{a\}} \tilde{d}^{i-1}_{a'}, 0\}\}$. Finally, as $\xi_a^{SR} < 1$, Lemma A.4 implies that $x^i_a < 1$, hence $\tilde{d}^i_a \leq d^i_a = 1$. As the induction hypothesis holds in Round 1, $\tilde{d}^i \leq d^i = 1$ for every i, and therefore $d(\widetilde{\xi}^{SR}) \leq d(\xi^{SR})$.

Consider next a category c and let $A_E(c) = \{b \in A : b\pi_c\emptyset\}$ and $\widetilde{A}_E(c) = \{b \in A : b\tilde{\pi}_c\emptyset\}$. By construction, $\widetilde{A}_E(c) = A_E(c) \setminus \{a\}$ if $c = \tilde{c}$ and $\tilde{a} = \emptyset$, and $\widetilde{A}_E(c) = A_E(c)$ otherwise; hence $\widetilde{A}_E(c) \subseteq A_E(c)$. If $\sum_{b \in A} \xi^{SR}_{b,c} < q_c$, then by Lemma A.15 $\sum_{b \in A} \xi^{SR}_{b,c} = \sum_{b \in A_E(c)} d_b(\xi^{SR})$. As $d(\widetilde{\xi}^{SR}) \leq d(\xi^{SR})$ and $\widetilde{A}_E(c) \subseteq A_E(c)$, it follows that $\sum_{b \in \widetilde{A}_E(c)} d_b(\widetilde{\xi}^{SR}) \leq \sum_{b \in A_E(c)} d_b(\xi^{SR}) < q_c$. Then, by the same argument $\sum_{b \in A} \widetilde{\xi}^{SR}_{b,c} = \sum_{b \in \widetilde{A}_E(c)} d_b(\widetilde{\xi}^{SR})$, and therefore $\sum_{b \in A} \widetilde{\xi}^{SR}_{b,c} \leq \sum_{b \in A} \xi^{SR}_{b,c}$. This result holds trivially if $\sum_{b \in A} \xi^{SR}_{b,c} = q_c$; hence, it holds for all categories and $|\widetilde{\xi}^{SR}| \leq |\xi^{SR}|$. Consider finally an agent $b \neq a, \tilde{a}$, I show that $\widetilde{\xi}^{SR}_b \geq \xi^{SR}_b$. This is trivially satisfied if $\widetilde{\xi}^{SR}_b = 1$ so I assume that $\widetilde{\xi}^{SR}_b < 1$. By Lemma A.17, for every $c \in C_E(b)$, $\widetilde{\xi}^{SR}_{b,c} = \min\{d_b(\widetilde{\xi}^{SR}), \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} d_{a'}(\widetilde{\xi}^{SR}), 0\}\}$ and $\xi^{SR}_{b,c} = \min\{d_b(\xi^{SR}), \max\{q_c - \sum_{a' \in \hat{A}_{b,c}} d_{a'}(\xi^{SR}), 0\}\}$. As $\widetilde{\xi}^{SR}_b < 1$, Lemma A.14(i) implies that $d_b(\widetilde{\xi}^{SR}) = 1 \geq d_b(\xi^{SR})$; therefore, as $d(\widetilde{\xi}^{SR}) \leq d(\xi^{SR})$, we have $\widetilde{\xi}^{SR}_b \geq \xi^{SR}_b$. The argument for $\widetilde{\xi}^{SR}_{\tilde{a}} \geq \xi^{SR}_{\tilde{a}}$ is almost analogous, the only difference is that $\tilde{a}\tilde{\pi}_{\tilde{c}}a$ so $\widetilde{\xi}^{SR}_{\tilde{a},\tilde{c}} = \min\{d_{\tilde{a}}(\widetilde{\xi}^{SR}), \max\{q_{\tilde{c}} - \sum_{a' \in \hat{A}_{\tilde{a},\tilde{c}} \setminus \{a\}} d_{a'}(\widetilde{\xi}^{SR}), 0\}\}$. As $|\widetilde{\xi}^{SR}| \leq |\xi^{SR}|$ and $\widetilde{\xi}^{SR}_b \geq \xi^{SR}_b$ for every $b \neq a$, it must be that $\xi^{SR}_a \geq \widetilde{\xi}^{SR}_a$. □

Appendix C. Properties of the SRLP Algorithm

I first show that the solution to (LP 1) is unique as long as its input x^i is a preallocation (Lemma C.1) and then show that x^i is indeed an allocation (Lemma C.2). Taken together, Lemmas C.1 and C.2 ensure that the SRLP algorithm is well defined.

Lemma C.1. *For any preallocation x,* (LP 1) *has a unique solution.*

Proof. For notational simplicity and as there is no risk of confusion, I omit the dependency on x throughout the proof. First, note that the vector $(\xi_{a(c),c})_{c \in \widetilde{C}} = \mathbf{0}_{|\widetilde{C}|}$ satisfies all constraints so (LP 1) has a solution and it remains to show that there cannot be multiple ones. For this purpose, I introduce some notation. Given a vector $(\xi_{a(c),c})_{c \in \widetilde{C}}$, for every agent $a' \in \widetilde{A}$ let $S_{a'} = \sum_{c \in C_M(a)} \xi_{a,c}$. 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

$$\theta_{a,c} = \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - \sum_{c' \in C_M(a)} \xi_{a,c'}}{|C_Q(a)|} = \widetilde{q}_c - \sum_{a \in \widetilde{A}_Q(c)} \frac{1 - S_{a'}}{|C_Q(a)|}. \tag{2}$$

Note that $\theta_{a,c}$ is the right-hand side of (LP 1)'s second constraint 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 $\theta_{a,c_1} \geq \ldots \geq \theta_{a,c_{|C_M(a)|}}$. For every $i = 1, \ldots, |C_M(a)|$, let $T_i = \frac{1 - \sum_{j > i} \theta_{a,c_j}}{|C_Q(a)| + i}$ and define $n = 0, 1, \ldots, |C_M(a)|$ as follows. If $T_i > \theta_{a,c_i}$ for every $i = 1, \ldots, |C_M(a)|$, then $n = 0$. Otherwise, n is the such that $T_n \leq \theta_{a,c_n}$ and $T_i > \theta_{a,c_i}$ for every $i > n$.

Claim 2. *For every $i \geq n$, $T_i > T_{i+1}$.*

Proof. Fix $i \geq n$. It needs to be shown that $T_i > T_{i+1}$, which is equivalent to $\theta_{a,c_{i+1}} < \frac{1 - \sum_{j > i+1} \theta_{a,c_j}}{C_Q(a)| + i + 1}$. The latter inequality is satisfied as its right-hand side is equal to T_{i+1} and, since $i + 1 > n$, $\theta_{a,c_{i+1}} < T_{i+1}$. □

Consider the following linear program.

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

The linear program (LP 2) can be thought of as a version of (LP 1) 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. To show that (LP 2) has a unique solution, I first consider a restricted version. 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:

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

(LP 3) is a restricted version of (LP 2), in which every ξ_{a,c_i} with $i>n$ is fixed.

Claim 3. *For any $(y_i)_{i>n} \leq (\theta_{a,c_i})_{i>n}$, the unique solution to* (LP 3) *is $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ such that $\xi^*_{a,c_i} = \frac{1-\sum_{j>n} y_j}{|C_Q(a)|+n}$ for every $i \leq n$ and $\xi^*_{a,c_i} = y_i$ for every $i>n$.*

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

Case 1: $i \leq n$. It needs to be shown that $\frac{1-\sum_{j>n} y_j}{|C_Q(a)|+n} \leq \frac{1-\sum_{j\neq i}\xi^*_{a,c_j}}{|C_Q(a)|+1}$, which holds with an equality since, by definition,

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

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, using the definition of $(\xi_{a,c_i})_{i=1}^{|C_M(a)|}$ and rearranging, is equivalent to $y_i \leq \frac{1-\sum_{j>n} y_j}{|C_Q(a)|+n}$. The latter inequality is satisfied as $y_i \leq \theta_{a,c_i} < T_i$ by definition, $T_i < T_n$ by Claim 2, and T_n is equal to the right-hand side.

($(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ is a solution to (LP 3)) 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 $i \leq n$ and rearranging yields $\sum_{i=1}^{|C_M(a)|}\xi_{a,c_i} \leq n\frac{1-\sum_{i>n}\xi_{a,c_i}}{|C_Q(a)|+n} + \sum_{i>n}\xi_{a,c_i}$, which by constraint (ii) and the definition of $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ implies that $\sum_{i=1}^{|C_M(a)|}\xi_{a,c_i} \leq n\frac{1-\sum_{i>n}y_i}{|C_Q(a)|+n} + \sum_{i>n}y_i = \sum_{i=1}^{|C_M(a)|}\xi^*_{a,c_i}$.

((LP 3) does not have any other solution) Let $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|}$ be a solution to (LP 3), I show that $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|} = (\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$. As $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|}$ is a solution to (LP 3),

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

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

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

Rearranging yields $\xi^\sharp_{a,c_i} \leq \frac{1-\sum_{j>n}y_j}{|C_Q(a)|+n}$; therefore, $\xi^\sharp_{a,c_i} \leq \xi^*_{a,c_i}$ for every $i \leq n$. As $\xi^\sharp_{a,c_i} = \xi^*_{a,c_i}$ for every $i > n$ (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 conclude that $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|} = (\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$. □

Claim 4. *The unique solution to* (LP 2) *is* $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ *such that* $\xi^*_{a,c_i} = T_n$ *for every* $i \leq n$ *and* $\xi^*_{a,c_i} = \theta_{a,c_i}$ *for every* $i > n$.

Proof. By Claim 3, $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ satisfies constraint (i); otherwise the unique solution to (LP 3) would not satisfy its constraints in the special case where $y_i = \theta_{a,c_i}$ for every $i > n$. As $T_n \leq \theta_{a,c_i}$ for every $i \leq 1$, $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ satisfies constraint (ii) by definition. Having shown that $(\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$ satisfies all constraints (which implies that (LP 2) has a solution), I now show that it is the unique solution to (LP 2). Let $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|}$ be a solution to (LP 2), I show that $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|} = (\xi^*_{a,c_i})_{i=1}^{|C_M(a)|}$. Fixing $\xi^\sharp_{a,c_i} \leq \theta_{a,c_i}$ for every $i > n$; Claim 3 implies that, for every $i \leq n$, $\xi^\sharp_{a,c_i} = \frac{1-\sum_{j>n}\xi^\sharp_{a,c_j}}{|C_Q(a)|+n}$, as otherwise $(\xi^\sharp_{a,c_i})_{i=1}^{|C_M(a)|}$ would not be a solution. 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 implies that $\xi^\sharp_{a,c_i} = T_n$ for every $i \leq n$. $\square$

Finally, I use Claim 4 to show that (LP 1) has a unique solution. For any $a \in \widetilde{A}$ and any S_{-a}, let $S_a(S_{-a})$ be the maximized objective function of (LP 2). If S_{-a} increases, then by (2), so does $\theta_{a,c}$ for every $c \in C_M(a)$. Therefore, constraint (ii) of (LP 2) is relaxed so $S_a(S_{-a})$ is increasing in S_{-a}. Suppose 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 $\overline{S}_a > S^\sharp_a$; hence $\overline{S}_a > \max\{S^*_a, S^\sharp_a\}$, a contradiction. By the preceding argument, every solution to (LP 1) yields the same sum vector, which I denote by S^*. By Claim 4, for every $a \in \widetilde{A}$, there is 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). $\square$

Fix an agent a, a category c, and a Round $i \geq 1$ of the SRLP algorithm.

Lemma C.2. *x^i is a preallocation and $d^i_a \in [1/|C|, 1]$.*

Proof. If the SRLP algorithm uses linear programming in Round i, then $d^i_a \in [1/|C|, 1]$ by construction. The statement then follows from Lemma A.2. $\square$

Define $\xi^i = (\xi^i_{a,c})_{a\in A,c\in C}$ with $\xi^i_{a,c} = \min\{x^i_{a,c}, d^i_a\}$ to be the allocation found in Round i of the SRLP algorithm.

Lemma C.3. *(i) ξ^i is an allocation and $\xi^i_a = \min\{x^i_a, 1\}$*
(ii) $d^i_a = 1$ if $\xi^i_a < 1$ and $d^i_a = \max_{c\in C}\{x^i_{a,c}\}$ if $\xi^i_a = 1$
(iii) $\xi^i_a = 1$ if and only if there exists a category c such that $\xi^i_{a,c} = d^i_a$.

Proof. Analogous to Lemmas A.3, A.5 and A.6. $\square$

Lemma C.4. *$\xi^i_a \geq \xi^{i-1}_a$ and $d^i_a \leq d_a(x^i) \leq d^{i-1}_a$.*

Proof. By definition, $\xi^1_a \geq \xi^0 = 0$ and $d_a(x^1) \leq d^0_a = 0$. Moreover, at x^0 every agent is unqualified for every category while at x^1 at least one agent is marginal for a category since $q \geq 1$ units are allocated. (This does not hold if no agent is eligible for any category, but in that case the result is trivial as the SRLP algorithm ends in Round 1 and outputs $x^1 = \mathbf{0}$.) Therefore, the SRLP algorithm does not use linear programming in Round 1 and $d^i_a = d_a(x^i)$ so the statement holds in Round 1. I proceed by induction, assuming that the statement holds for every $j < i$ (induction hypothesis) and showing that $\xi^i_a \geq \xi^{i-1}_a$ and $d^i_a \leq d_a(x^i) \leq d^{i-1}_a$. Analogous reasoning to Lemma A.4 implies that $d_a(x^i) \leq d^{i-1}_a$. If either the SRLP algorithm does not use linear programming in Round i or it does but $a \notin \widetilde{A}^{x^i}$, then $d^i_a = d_a(x^i) \leq d^{i-1}_a$. It remains to consider the case in which the SRLP algorithm uses linear programming in Round i and $a \in \widetilde{A}^{x^i}$. Define the allocation $\tilde{\xi}^i = (\tilde{\xi}^i_{b,c})_{b\in A, c\in C}$ such that $\tilde{\xi}^i_{b,c} = \min\{x^i_{b,c}, d_b(x^i)\}$; that is, $\tilde{\xi}^i$ is defined analogously to ξ^i but uses the demand vector $d(x^i)$ rather than d^i. Observe that the vector $(\tilde{\xi}^i_{a^{x^i}(c),c})_{c\in\widetilde{C}^{x^i}}$ satisfies the constraints of (LP 1); therefore, the vector $(\xi^i_{a,c})_{c\in C^{x^i}_M(a)}$ satisfies the constraints of (LP 2) (see the proof of Lemma C.1). Let $(\xi^*_{a^{x^i}(c),c})_{c\in\widetilde{C}}$ be the solution to (LP 1); then the vector $(\xi^*_{a,c})_{c\in C^{x^i}_M(a)}$ is the solution to (LP 2) (see again the proof of Lemma C.1). It follows that $\sum_{c\in C^{x^i}_M(a)} \tilde{\xi}^i_{a,c} \leq \sum_{c\in C^{x^i}_M(a)} \xi^*_{a,c}$. For every $c \in C^{x^i}_M$, by definition $x^i_{a,c} < d_a(x^i)$ so $\tilde{\xi}^i_{a,c} = x^i_{a,c}$; hence $\sum_{c\in C^{x^i}_M(a)} x^i_{a,c} \leq \sum_{c\in C^{x^i}_M(a)} \xi^*_{a,c}$. Again by definition, $\sum_{c\in C} \min\{d_a(x^i), x^i_{a,c}\} = 1$ and $|C^{x^i}_Q(a)|d^i_a = 1 - \sum_{c\in C^{x^i}_M(a)} \xi^*_{a,c}$. Combining these two results yields $|C^{x^i}_Q(a)|d_a(x^i) + \sum_{c\in C^{x^i}_M(a)} x^i_{a,c} = |C^{x^i}_Q(a)|d^i_a + \sum_{c\in C^{x^i}_M(a)} \xi^*_{a,c} = 1$. As $\sum_{c\in C^{x^i}_M(a)} x^i_{a,c} \leq \sum_{c\in C^{x^i}_M(a)} \xi^*_{a,c}$, it follows that $d_a(x^i) \geq d^i_a$. Having established that $d^i_a \leq d_a(x^i) \leq d^{i-1}_a$, $\xi^i_a \geq \xi^{i-1}_a$ follows from analogous reasoning to Lemma A.4. □

Lemma C.5. *(i) If either $x^i_{a,c} < d^{i-1}_a$ or $\xi^i_{a,c} < d^i_a$, then $x^i_{a',c} = 0$ for every $a' \in \check{A}_{a,c}$.*
(ii) If a is eligible for c and $x^i_{a,c} < d^{i-1}_a$, then $x^i_{a,c} + \sum_{a'\in\hat{A}_{a,c}} x^i_{a',c} = q_c$.

Proof. Analogous to Lemmas A.7 and A.9. □

Lemma C.6. *x^i satisfies Axioms 1-4.*

Proof. Analogous to Lemma A.10. □

Lemma C.7. *(i) If $x^i_{a,c} < d^i_a$, then, for every Round $j \leq i$, $x^j_{a,c} \leq x^i_{a,c} < d^j_a$.*
(ii) If $x^i_{a,c} \geq d^i_a$, then, for every $j > i$, $x^j_{a,c} = d^{j-1}_a \geq d^j_a$.

Proof. Analogous to Lemmas A.12 and A.13. □

Lemma C.8. *If the SRLP algorithm uses linear programming in Round i, then either $x_a^{i+1} \leq 1$ or a's status for a category changes between x^i and x^{i+1}.*

Proof. Case 1: $C_Q^{x^i}(a) \neq \varnothing$ and $C_M^{x^i}(a) = \varnothing$. By construction, $d_a^i = 1/|C_Q^{x^i}(a)|$ so $\sum_{c \in C_Q^{x^i}(a)} x_{a,c}^{i+1} \leq |C_Q^{x^i}(a)| d_a^i = 1$. Therefore, $x_a^{i+1} > 0$ implies that there exists $c \in C_U^{x^i}(a)$ such that $x_{a,c}^{i+1} > 0$, hence a's status for c changes between x^i and x^{i+1}.

Case 2: $C_Q^{x^i}(a) \neq \varnothing$ and $C_M^{x^i}(a) \neq \varnothing$. By assumption, a is an agent of interest. The solution to (LP 1) ξ^* is such that, for each element, at least one of the two constraints holds, otherwise that element (hence the sum) can be increased. Therefore, for every $c \in C_M^{x^i}(a)$, $\xi_{a,c}^* = \min\{d_a^i, \widetilde{q}_c^{x^i} - \sum_{a' \in \widetilde{A}_Q^{x^i}(c)} d_{a'}^i\}$. By the definition of $\widetilde{q}_c^{x^i}$ and Lemma A.14(iii), we have $\xi_{a,c}^* = \min\{d_a^i, q_c^{x^i} - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^i\}$ and, as $c \in C_M^{x^i}(a)$, $q_c^{x^i} - \sum_{a' \in \hat{A}_{a,c}} d_{a'}^i > 0$ so $x_{a,c}^{i+1} = \xi_{a,c}^*$. For every $c \in C_Q^{x^i}(a)$, $x_{a,c}^i \geq d_a(x^i)$ by definition so $x_{a,c}^i \geq d_a^i$ by Lemma C.4 and $x_{a,c}^{i+1} = d_a^i$ by Lemma C.7(ii). It follows that

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

Then, $x_a^{i+1} > 1$ implies that there exists $c \in C_U^{x^i}(a)$ such that $x_{a,c}^{i+1} > 0$, hence a's status for c changes between x^i and x^{i+1}.

Case 3: $C_Q^{x^i}(a) = \varnothing$. By Lemma A.14(i), $x_a^{i+1} > 1$ implies that $C_Q^{x^{i+1}}(a) \neq \varnothing$, hence a's status for some category changes between x^i and x^{i+1}. □

Proof of Theorem 4. 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; hence, at least two statuses change. Thereafter, by Claim 1, every second round either a status changes or an allocation is found. By Lemma C.7, status changes are irreversible, therefore there can be at most $2|A||C|$ of them throughout the SRLP algorithm. It follows that an allocation is found by Round $4|A||C| - 1$.

Let $N < 4|A||C|$ be the number of rounds after which the SRLP algorithm terminates. Then, the SRLP algorithm produces the allocation x^N, which by Lemma C.6 satisfies Axioms 1-4. It remains to show that $x^N = \xi^{SR}$. 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 ξ^{SR} 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 such that: (i) for any two agents a and a', $a\overline{\pi}_c a'$ if and only if $a\pi_c a'$ and (ii) $a\overline{\pi}_c\emptyset$ if and only if $\xi^{SR}_{a,c} > 0$. In this alternative problem $\overline{R}$, I denote the SR allocation by $\overline{\xi}^{SR}$, the number of rounds after which the SRLP algorithm terminates by $\overline{N}$, and the output of the SRLP algorithm by $\overline{x}^{\overline{N}}$. To prove that $\xi^{SR} = x^N$, I show successively that $\xi^{SR} = \overline{\xi}^{SR}$, $\overline{\xi}^{SR} = \overline{x}^{\overline{N}}$, and $\overline{x}^{\overline{N}} = x^N$.

($\xi^{SR} = \overline{\xi}^{SR}$) Consider any agent-category pair (a, c) such that $\xi^{SR}_{a,c} = 0$. If $x^i_{a,c} > 0$ in any Round i of the SR algorithm, then for every $j > i$, either $x^j_{a,c} < d^j_a$ and $x^j_{a,c} \geq x^i_{a,c} > 0$ by Lemma A.12, or $x^j_{a,c} \geq d^j_a > 0$ by Lemma A.2; hence $\xi^{SR}_{a,c} = 0$, a contradiction. It follows that $x^i_{a,c} = 0$ for every $i \geq 1$; therefore, making a not eligible for c does not impact the SR algorithm and $\xi^{SR} = \overline{\xi}^{SR}$.

($\overline{\xi}^{SR} = \overline{x}^{\overline{N}}$) As $\overline{x}^{\overline{N}}$ is an allocation of $\overline{R}$ 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. For notational convenience, for every agent a, category c, and status $S \in \{Q, M, U\}$, I denote by $C_S(a) = C^{\xi^{SR}}_S(a) = C^{\overline{\xi}^{SR}}_S(a)$ the set of categories for which a's status at the SR allocation is S (in both R and $\overline{R}$ since $\xi^{SR} = \overline{\xi}^{SR}$), by $A_S(c) = A^{\xi^{SR}}_S(c) = A^{\overline{\xi}^{SR}}_S(c)$ the set of agents whose status for c at the SR allocation is S, and, if $A_M(c) \neq \varnothing$, by $a(c) = a^{\xi^{SR}}(c) = a^{\overline{\xi}^{SR}}(c)$ the agent who is marginal for c at the SR allocation ($a(c)$ is unique by Lemma A.14(ii)). Fix an allocation $\overline{\xi}^*$ of $\overline{R}$ that satisfies Axioms 1-4 and an agent-category pair (a, c). I consider three cases.

Case 1: $a \in A_U(c)$. By definition, $\overline{\xi}^{SR}_{a,c} = \xi^{SR}_{a,c} = 0$ so a is not eligible for c in $\overline{R}$ and $\overline{\xi}^*_{a,c} = 0$ as $\overline{\xi}^*$ satisfies Axiom 1.

Case 2: $a \in A_Q(c)$. By definition, $\overline{\xi}^{SR}_{a,c} = d(\overline{\xi}^{SR}_{a,c})$ and by Lemma A.14(iii), $\overline{\xi}^{SR}_{a',c} = d_{a'}(\overline{\xi}^{SR})$ for every $a' \in \hat{A}_{a,c}$; hence, as $\overline{\xi}^{SR}$ is an allocation, $d_a(\overline{\xi}^{SR}) + \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\overline{\xi}^{SR}) \leq q_c$. By Proposition 3, it follows that $d_a(\overline{\xi}^*) + \sum_{a' \in \hat{A}_{a,c}} d_{a'}(\overline{\xi}^*) \leq q_c$. If $\overline{\xi}^*_{a,c} < d_a(\overline{\xi}^*)$, then Lemma A.15 implies that $\overline{\xi}^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \overline{\xi}^*_{a',c} = q_c$ so there exists $a' \in \hat{A}_{a,c}$ such that $\overline{\xi}^*_{a',c} > d_{a'}(\overline{\xi}^*)$, a contradiction. We conclude that $\overline{\xi}^*_{a,c} = d_a(\overline{\xi}^*)$.

Case 3: $a \in A_M(c)$. By Lemma A.15, $\overline{\xi}^*_{a,c} + \sum_{a' \in \hat{A}_{a,c}} \xi^*_{a',c} = q_c$. By Lemma A.14(iii), $\hat{A}_{a,c} = A_Q(c)$ and by Case 2, $\overline{\xi}^*_{a',c} = d_{a'}(\overline{\xi}^*)$ for every $a' \in A_Q(c)$; therefore, $\overline{\xi}^*_{a,c} + \sum_{a' \in A_Q(c)} d_{a'}(\overline{\xi}^*) = q_c$. By Lemma A.14(i), $\overline{\xi}^*_a = 1$ for every $a' \in \hat{A}_{a,c}$ so $\sum_{c' \in C_Q(a')} \overline{\xi}^*_{a',c'} + \sum_{c' \in C_M(a')} \overline{\xi}^*_{a',c'} = 1$, which by Case 2 is equivalent to $|C_Q(a')| d_{a'}(\overline{\xi}^*) + \sum_{c' \in C_M(a')} \overline{\xi}^*_{a',c'} = 1$. Combining the last two results yields $\overline{\xi}^*_{a,c} + \sum_{a' \in A_Q(c)} \frac{1 - \sum_{c' \in C_M(a')} \overline{\xi}^*_{a',c'}}{|C_Q(a')|} = q_c$.

Let $C_M = \{c \in C : C_M(c) \neq \varnothing\}$ be the set of categories that have a marginal agent. Cases 1-3 jointly imply that, for any allocation $\overline{\xi}^*$ of $\overline{R}$ that satisfies Axioms 1-4,

$$\overline{\xi}^*_{a(c),c} + \sum_{a' \in A_Q(c)} \frac{1 - \sum_{c' \in C_M(a')} \overline{\xi}^*_{a',c'}}{|C_Q(a')|} = q_c \quad \text{for every } c \in C_M \text{, and} \tag{3}$$

$$\overline{\xi}^*_{a,c} = \begin{cases} \frac{1 - \sum_{c' \in C_M(a)} \overline{\xi}^*_{a,c'}}{|C_Q(a)|} & \text{if } a \in A_Q(c) \\ 0 & \text{if } a \in A_U(c) \end{cases} \quad \text{for every } (a,c) \text{ with } a \notin A_M(c). \tag{4}$$

Note that (3) is a linear system of equations with $|C_M|$ variables and $|C_M|$ equations. Once a vector $(\overline{\xi}^*_{a(c),c})_{c \in C_M}$ satisfying (3) has been found, the corresponding allocation is pinned down by (4). As $\overline{\xi}^{SR}$ satisfies Axioms 1-4, (3) has a solution. If that solution is unique, then $\overline{\xi}^{SR}$ is the unique allocation of $\overline{R}$ that satisfies Axioms 1-4 and the proof is complete. It remains to show that (3) does not have multiple solutions.

Suppose that (3) has multiple solutions. Then, (3) has strictly fewer than $|C_M|$ linearly independent equations so there is at least one degree of freedom. Arbitrarily fixing a category $c \in C_M$, for any value of $\overline{\xi}^*_{a(c),c}$ there exists a vector $(\overline{\xi}^*_{a(c'),c'})_{c' \in C_M \setminus \{c\}}$ such that $(\overline{\xi}^*_{a(c'),c'})_{c' \in C_M}$ is a solution to (3). Given an arbitrarily small positive number $\epsilon > 0$, I construct an allocation $\overline{\xi}^\epsilon$ as follows. First, set $\overline{\xi}^\epsilon_{a(c),c} = \overline{\xi}^{SR}_{a(c),c} - \epsilon$. Second, for every $c' \in C_M \setminus \{c\}$, set $\overline{\xi}^\epsilon_{a(c'),c'}$ be such that $(\overline{\xi}^\epsilon_{a(c'),c'})_{c' \in C_M}$ is a solution to (3). Third, for every (a,c) with $a \notin A_M(c)$, set $\overline{\xi}^\epsilon_{a,c}$ using (4). As all equations in (3) and (4) 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 $c' \in C_M$. Fixing such an ϵ, I next show that $\overline{\xi}^\epsilon$ satisfies Axioms 1-4.

(Axiom 1) By definition, $\overline{\xi}^\epsilon_{a,c} = 0$ if a is not eligible for c in $\overline{R}$.

(Axiom 2) If $\overline{\xi}^\epsilon$ 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 $\overline{R}$. If $a \in A_Q(c)$, then, by (4), $\overline{\xi}^\epsilon_{a,c}|C_Q(a)| + \sum_{c' \in C_M(a)} \overline{\xi}^\epsilon_{a,c'} = 1$. Moreover, (4) implies that $\overline{\xi}^\epsilon_{a,c} = \overline{\xi}^\epsilon_{a,c'}$ for every $c' \in C_Q(a)$ so we have that $\sum_{c' \in C_Q(a)} \overline{\xi}^\epsilon_{a,c'} + \sum_{c' \in C_M(a)} \overline{\xi}^\epsilon_{a,c'}$; hence $\overline{\xi}^\epsilon_a = 1$, a contradiction. If $a \in A_M(c)$, then (3) and (4) imply that $\overline{\xi}^\epsilon_{a,c} + \sum_{a' \in A_Q(c)} \overline{\xi}^\epsilon_{a',c} = q_c$, a contradiction. If $a \in A_U(c)$, then $\xi^{SR}_{a,c} = 0$ so a is not eligible for c in $\overline{R}$, a contradiction.

(Axiom 3) If $\overline{\xi}^\epsilon$ does not respect priorities, then there exist an agent-category pair (a,c) and an agent $a' \in \check{A}_{a,c}$ such that $\overline{\xi}^\epsilon_a < 1$ and $\overline{\xi}^\epsilon_{a',c} > 0$. If $a \in A_Q(c)$, then analogous reasoning to Axiom 2 yields $\xi_a = 1$, a contradiction. If $a \notin A_Q(c)$, then $a' \in A_U(c)$ so (4) implies that $\overline{\xi}^\epsilon_{a',c} = 0$, a contradiction.

(Axiom 4) If $\overline{\xi}^\epsilon$ 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, $\max_{c'\in C}\{\overline{\xi}^\epsilon_{a,c'}\} \leq d_a(\overline{\xi}^\epsilon)$ so $\overline{\xi}^\epsilon_{a,c} < d_a(\overline{\xi}^\epsilon)$, which by (4) implies that $a \notin A_Q(c)$. Moreover, the assumption that a is eligible for c in $\overline{R}$ implies that $a \notin A_U(c)$ so it must be that $a \in A_M(c)$. In that case, however, we have $\overline{\xi}^\epsilon_{a,c} + \sum_{a'\in A_Q(c)} \overline{\xi}^\epsilon_{a',c} = q_c$ by (3) and (4). By Lemma A.14(iii), $A_Q(c) = \hat{A}_{a,c}$ so we conclude that $\overline{\xi}^\epsilon_{a,c} + \sum_{a'\in \hat{A}_{a,c}} \overline{\xi}^\epsilon_{a',c} = q_c$, a contradiction.

By the preceding argument, $\overline{R}$ has an allocation $\overline{\xi}^\epsilon$ that satisfies Axioms 1-4. Recall that, by construction, there is a category $c \in C_M$ such that $\overline{\xi}^\epsilon_{a(c),c} = \overline{\xi}^{SR}_{a(c),c} - \epsilon$ (with $\epsilon > 0$) and $0 < \overline{\xi}^\epsilon_{a(c),c} < d_{a(c)}(\overline{\xi}^\epsilon)$. Lemma A.15 implies that $\overline{\xi}^\epsilon_{a(c),c} + \sum_{a'\in \hat{A}_{a(c),c}} \overline{\xi}^\epsilon_{a',c} = q_c$. Then, for every $a' \in \hat{A}_{a(c),c}$, $\overline{\xi}^\epsilon_{a',c} + \sum_{b\in \hat{A}_{a',c}} \overline{\xi}^\epsilon_{b,c} < q_c$ so Lemma A.15 implies that $\overline{\xi}^\epsilon_{a',c} = d_{a'}(\overline{\xi}^\epsilon)$. It follows that $\overline{\xi}^\epsilon_{a(c),c} + \sum_{a'\in \hat{A}_{a(c),c}} d_{a'}(\overline{\xi}^\epsilon) = q_c$. As $\overline{\xi}^{SR}$ satisfies Axioms 1-4, analogous reasoning yields $\overline{\xi}^{SR}_{a(c),c} + \sum_{a'\in \hat{A}_{a(c),c}} d_{a'}(\overline{\xi}^{SR}) = q_c$. As $\overline{\xi}^\epsilon_{a(c),c} < \overline{\xi}^{SR}_{a(c),c}$, there exists $a' \in \hat{A}_{a(c),c}$ with $d_{a'}(\overline{\xi}^\epsilon) > d_{a'}(\overline{\xi}^{SR})$, which contradicts Proposition 3.

($\overline{x}^{\overline{N}} = x^N$) Suppose that, for some $i = 1, \ldots, \min\{N, \overline{N}\}$, $x^{i-1} = \overline{x}^{i-1}$ and $d^{i-1} = \overline{d}^{i-1}$ (induction hypothesis). I show that $x^i = \overline{x}^i$ and $d^i = \overline{d}^i$. Fixing any agent-category pair (a,c), I show that $x^i_{a,c} = \overline{x}^i_{a,c}$. If a is not eligible for c in R, then by definition $x^i_{a,c} = \overline{x}^i_{a,c} = 0$. If $\xi^{SR}_{a,c} > 0$, then a is eligible for c in both R and $\overline{R}$ so $x^i_{a,c} = \min\{d^{i-1}_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'}, 0\}\}$ and $\overline{x}^i_{a,c} = \min\{\overline{d}^{i-1}_a, \max\{q_c - \sum_{a'\in \hat{A}_{a,c}} \overline{d}^{i-1}_{a'}, 0\}\}$; hence, implies that $x^i_{a,c} = \overline{x}^i_{a,c}$ by the induction hypothesis. It remains to consider the case in which a is eligible for c in R and $\xi^{SR}_{a,c} = 0$. As $\xi^{SR}_{a,c} = 0$, a is not eligible for c in $\overline{R}$ so $\overline{x}^i_{a,c} = 0$. By Lemma A.15, $\xi^{SR}_{a,c} = 0$ implies that $\sum_{a'\in \hat{A}_{a,c}} \xi^{SR}_{a',c} = q_c$; therefore, as $\xi^{SR} = \overline{\xi}^{SR} = \overline{x}^{\overline{N}}$, we have $\sum_{a'\in \hat{A}_{a,c}} \overline{x}^{\overline{N}}_{a',c} = q_c$. For every $a' \in \hat{A}_{a,c}$, $\overline{x}^{\overline{N}}_{a',c} \leq d_{a'}(\overline{x}^{\overline{N}})$ as $\overline{x}^{\overline{N}}$ is an allocation and $d_{a'}(\overline{x}^{\overline{N}}) \leq \overline{d}^{i-1}_{a'}$ by Lemma C.4. It follows that $\sum_{a'\in \hat{A}_{a,c}} \overline{d}^{i-1}_{a'} \geq q_c$, which by the induction hypothesis implies that $\sum_{a'\in \hat{A}_{a,c}} d^{i-1}_{a'} \geq q_c$; hence $x^i_{a,c} = 0$.

As $x^{i-1} = \overline{x}^{i-1}$ and $x^i = \overline{x}^i$, the SRLP algorithm uses linear programming in Round i when run over R if and only if it does so when run over $\overline{R}$. In either case, the zero elements of the round preallocation do not impact the construction of the demand vector; hence $d^i = \overline{d}^i$. As the induction hypothesis holds in Round 1, it follows by induction that $x^i = \overline{x}^i$ for every $i = 1, \ldots, \min\{N, \overline{N}\}$ so the SRLP algorithm over R ends in Round $N = \overline{N}$ and produces $x^N = \overline{x}^{\overline{N}}$. $\square$