Size versus fairness in the assignment problem

Anna Bogomolnaia and Herve Moulin*
Adam Smith Business School, University of Glasgow

October 28, 2013

Abstract

When not all objects are acceptable to all agents, maximizing the number of objects actually assigned is an important design concern. We compute the guaranteed size index of the Probabilistic Serial mechanism, i.e., the worst ratio of the actual expected size to the maximal feasible size. It converges decreasingly to $1 - \frac{1}{e} \simeq 63.2\%$ *as the maximal size increases. It is the best index of any Envy-Free assignment mechanism.*

We are especially grateful to our colleagues David Manlove and Baharak Rastegari for introducing us to their research project EPSRC Grant # EP/K010042/1, and sharing the results already obtained with their colleagues at the University of Liverpool. Special thanks also to Jay Sethuraman for guiding us through the literature on online matching, and to Bettina Klaus, Tadashi Hashimoto, and seminar participants in Warwick, Manchester and Toulouse, for stimulating conversations.

1 Introduction

1.1 The context

The assignment of indivisible objects to economic agents by means of lotteries is an important example of a "market without money", where randomizing the allocation of objects, or, equivalently in some contexts, implementing time sharing, is the only way to achieve a fair outcome. The familiar real life examples include assigning workers to jobs, jobs to time slots, classes or dormitory rooms to students, school choice ([2], [19]), etc.. See [24] for a survey.

The three normative goals of mechanism design, efficiency, incentive compatibility and fairness, lead the discussion of the assignment problem in the economic literature. The recent literature on algorithmic mechanism design introduces the fourth goal of maximizing a simple measure of social optimality. One of the earliest instances of this approach is [23], discussing the tradeoff

*corresponding author: phone 44(0)1413302004; fax 44(0)1413302020; herve.moulin@glasgow.ac.uk

between Strategy-Proofness and the utilitarian minimization of aggregate cost. Another seminal example, closer to home, is in the bilateral matching problem. When preferences have ties and are incomplete (remaining single is preferred to some potential partners) not all stable matchings are of the same size (the "rural hospital theorem" does not apply), so it is natural to look for a stable matching of maximal size ([16]), or for a maximal cardinality matching with the smallest number of blocking pairs ([5]): both questions turn out to be NP-hard. The project [20], from which the present work is born, explores the tradeoffs between Strategy-Proofness on one hand, and maximizing the size of the match on the other, in a variety of assignment and matching problems.

1.2 The problem and the punchline

In most practical instances of the assignment problem, incomplete preferences are the norm: in school choice, parents can opt out of the public system; jobs have deadlines which render certain time slots useless; students can live off campus, and so on. Even with strict preferences, efficient assignments can then have very different sizes (number of agents who receive an object), so the goal of maximizing the size of the actual assignment becomes important in its own right: filling the largest possible number of slots/seats/jobs, is a component of the system performance, to which public shool administrators, the housing office on campus, etc.., are paying attention. We define the *size index* of an assignment as the ratio of the size of the actual assignment to the maximal feasible size.

Note that the largest feasible size of an assignment only depends upon the bipartite graph of acceptability, and ignores the finer information in the profile of individual preferences. This implies that size maximization frequently conflicts with the goals of fairness and incentive compatibility, as is obvious in the following elementary example with two objects a, b and two agents Ann, Bob, who both prefer a to b. If both objects are acceptable to Bob but Ann only accepts a, then assigning a to Ann and b to Bob is the only assignment of maximal size, but it is obviously unfair to Bob, and makes it profitable for him to report that only a is acceptable, if he prefers a 50% chance of getting a to a 100% chance of b. Here, as in [8] and much of the subsequent literature, we interpret fairness as the well-known Envy-Freeness property, and incentive compatibility as Strategy-Proofness (both defined in section 4).

We define the *guaranteed m-size index* of a random assignment mechanism as its worst size index over all assignment problems such that the maximal size of a feasible assignment is m. We compute the largest guaranteed m-size index r_m of any Envy-Free mechanism, and show that it is achieved by the *Probabilistic Serial* mechanism ([8], [7], [15]; see section 4), the only known mechanism to date combining Envy-Freeness with Ordinal Efficiency. Moreover the sequence r_m converges decreasingly to $1 - \frac{1}{e}$ as m grows.

Although the Probabilistic Serial mechanism is not strategy-proof, this result throws some light on the tradeoff between size maximization and Strategy-Proofness. Indeed the familiar assignment mechanism *Random Priority* (a.k.a.

serial dictatorship, see [1]) is strategy-proof, and it was shown in [12] to have a smaller size index than Probabilistic Serial precisely for those problems where the latter achieves its worst case index r_m (see section 6 for details). Therefore r_m is also an upper bound for the guaranteed m-size index of Random Priority. Moreover, unlike PS, RP is *wasteful* ([14]), i.e., it may fail to fully distribute some desirable objects. Nevertheless [21] show that its index is at least $1-(1-\frac{1}{m+1})^m - \frac{1}{m}$, a sequence converging increasingly to $1-\frac{1}{e}$.[1] This confirms the earlier results in [11] about the asymptotic equivalence of these two benchmark mechanisms. It may well be that the worst case performance of RP cannot be improved by any other strategy-proof mechanism, despite the fact that RP is dominated by some less wasteful strategy-proof mechanisms ([14]).

2 Random assignment with outside options

Fix N the set of agents and A of objects, with respective cardinalities n and q. A preference R_i of agent $i \in N$ is a *possibly empty* ordered subset of A, written $R_i = (a_1, a_2, \cdots, a_k)$ where a_1 is the best object for i and a_k her least preferred acceptable object. Abusing notation, $a \in R_i$ means that a is an acceptable object for i, and $R_i = \varnothing$ means that no object is acceptable to i. We write $\mathcal{R}(A)$ for the set of individual preferences.

A profile of preferences $R \in \mathcal{R}(A)$ defines a *compatibility* bipartite graph $E \subseteq N \times A$: $ia \in E(R) \Leftrightarrow a \in R_i$, describing which objects are acceptable to which agents. An assignment problem is a triple $\Delta = (N, A, R)$, and its compatibility graph is written $E(\Delta)$.

An **assignment** is a $N \times A$ *substochastic* matrix $P = [p_{ia}] \in \mathbb{R}_+^{N\times A}$: $\sum_N p_{ia} \leq 1$ for all a and $\sum_A p_{ia} \leq 1$ for all i. It is feasible at R if, in addition, $p_{ia} > 0 \Rightarrow ia \in E(\Delta)$. We write $\mathcal{P}(E(\Delta))$, or simply $\mathcal{P}(E)$, for the set of feasible assignments at Δ, and $\mathcal{P}^d(E)$ for the subset of *deterministic* feasible assignments ($p_{ia} = 0, 1$ for all i, a). A well known fact (a variant of Birkhof's Theorem) is that the convex hull of $\mathcal{P}^d(E)$ is $\mathcal{P}(E)$.

The expected number of objects (or agents) assigned at P is $s(P) = \sum_{N\times A} p_{ia}$, we call it the **size** of P. Note that $s(P) \leq \min\{n, q\}$. The following nice fact refines Birkhof's Theorem. A random assignment is implemented by deterministic assignments of (almost) equal size: any $P \in \mathcal{P}(E)$ is a convex combination of deterministic assignments of size $\lfloor s(P) \rfloor$ or $\lceil s(P) \rceil$ (lower and upper integral part).[2]

In particular the program

$$s^*(E) = \max_{P\in\mathcal{P}(E)} s(P) \tag{1}$$

[1] This result is closely related to the well known online algorithm maximizing the guaranteed size of a bilateral matching, relative to the maximal size feasible offline. The Ranking algorithm of [18] selects randomly and uniformly an ordering of the objects, then assign to the incoming agent the highest acceptable object in that ordering; its m-guaranteed size is no less than $1-(1-\frac{1}{m+1})^m$ (see also [4] for a simpler proof and [17] for a generalization to multiple objects).

[2] This follows from the results in [10].

has at least one deterministic solution, and every solution is a convex combination of such deterministic assignments. We call $s^*(E(\Delta))$ the **size of the problem** Δ, i.e., the maximal number of objects/agents it is feasible to assign. The set of assignment problems of size m is denoted $\mathcal{A}^m$.

An **assignment mechanism** F associates to every assignment problem Δ a feasible assignment $F(\Delta) = P \in \mathcal{P}(E(\Delta))$. We focus in this paper on the worst possible match size that a mechanism can reach, relative to the size of the problem. Define the **guaranteed** m**-size index** of F as follows

$$\sigma_m(F) = \min_{\Delta \in \mathcal{A}^m} \frac{1}{m} s(F(\Delta))$$

The **absolute** guaranteed size index of F is $\sigma_\infty(F) = \inf_{m \geq 1} \sigma_m(F)$.

3 Efficiency and guaranteed size

Given a problem Δ and two *deterministic* assignments $P, P' \in \mathcal{P}^d(E(\Delta))$, we say that P' is Pareto superior to P if $P \neq P'$ and

$$\{p'_{ia} = 1 \text{ and } p_{ib} = 1\} \Rightarrow aR_i b$$

$$\{p'_{ia} = 0 \text{ for all } a\} \Rightarrow \{p_{ia} = 0 \text{ for all } a\}$$

An *efficient* (Pareto optimal) deterministic assignment is one that is not Pareto dominated.

In any problem $\Delta \in \mathcal{A}^m$ there is at least one efficient deterministic assignment of maximal size m. This follows because if an assignment $P \in \mathcal{P}^d(E)$ is Pareto dominated by P', then $s(P) \leq s(P')$. On the other hand it is easy to construct problems with efficient deterministic assignments of size $\frac{m}{2}$. The example in subsection 1.2 is the simplest one:

Ann	Bob
a	a
$\varnothing$	b

Here $m = 2$ yet $\{a \to \text{Bob}, \varnothing \to \text{Ann}\}$ is an efficient assignment. If m is even (resp. odd), we can replicate this two-agent$\times$two-object pattern to get a problem in $\mathcal{A}^m$ with an efficient assignment of size $\frac{m}{2}$ (resp. $\frac{m+1}{2}$).

A useful and well known observation is that *in any problem of size m, any efficient deterministic assignment is of size at least $\frac{m}{2}$.*[3] Therefore any efficient deterministic mechanism has a size index of at least $\frac{1}{2}$.

For a general (random) assignment mechanism F, the weakest efficiency requirement is **Ex Post Efficiency (EPE)**, requiring that the assignment P be a convex combination of efficient deterministic assignments. The above observation implies that any ex post efficient assignment mechanism has a guaranteed

[3] If $P \in \mathcal{P}^d(E)$ is efficient and of size m', and both agent i and object a are not matched at P, then $ia \notin E$, otherwise assigning a to i would be a Pareto improvement of P. It follows that any edge used by a matching feasible at E has at least one endnode matched in P, and there are $2m'$ such nodes.

size of at least $\frac{1}{2}$. This good news is mitigated by the fact, to which we now turn, that other normative requirements of fairness and incentive compatibility place an upper bound on the guaranteed size of the match.

4 Three main axioms

Given a problem Δ, agent i compares two feasible assignments $P, P' \in \mathcal{P}(E(\Delta))$ by means of her own allocations $p(i) = (p_{ia})_{a \in A}$ and $p'(i)$, the i-th rows of P and P' respectively. We define a familiar incomplete preference relation for agent i such that $R_i = (a_1, \cdots, a_k)$, $1 \leq k \leq r$ (this relation is empty if $R_i = \varnothing$). We say that $p(i)$ is **sd-preferred** to $p'(i)$ (where sd stands for stochastic dominance) if

$$\sum_1^t p_{ia_t} \geq \sum_1^t p'_{ia_t} \text{ for all } t, 1 \leq t \leq k$$

and we write $p(i) \overset{sd_i}{\succeq} p'(i)$. Note that sd-indifference is just equality. We say that $p(i)$ is **strictly sd-preferred** to $p'(i)$ if $p(i) \overset{sd_i}{\succeq} p'(i)$ and $p'(i) \neq p(i)$, so that at least one of the inequalities above is strict; then we write $p(i) \overset{sd_i}{\succ} p'(i)$. We now define the three normative properties key to the discussion of random assignment mechanisms.

The feasible assignment $P \in \mathcal{P}(E(\Delta))$ is

Ordinally Efficient (OE) if for all $P' \in \mathcal{P}(E(\Delta))$, $\{p'(i) \overset{sd_i}{\succeq} p(i)$ for all $i \in N\} \Longrightarrow P' = P$

Envy-Free (EF) if $p(i) \overset{sd_i}{\succeq} p(j)$ for all $i, j \in N$

For a deterministic assignment, OE and EPE are the same thing, but for general random assignments OE is a strictly stronger requirement than EPE.

A deterministic mechanism (i.e., selecting $F(\Delta) = P \in \mathcal{P}^d(E(\Delta))$ for any problem) cannot be Envy-Free, so EF requires randomization. The Probabilistic Serial mechanism, explained in the next section, is Ordinally eficient and Envy-Free, and the only example to date of a random mechanism with these two properties.

The assignment mechanism F is

Strategy-proof (SP) if for all Δ, all $i \in N$, and all $R'_i \in \mathcal{R}(A)$ we have $p(i) \overset{sd_i}{\succeq} p'(i)$, where $F(N, A, R) = P$ and $F(N, A, (R'_i, R_{-i})) = P'$

The simplest example of a strategyproof mechanism is the Fixed π-Priority mechanism, where π is an arbitrary ordering $\pi = \{i_1, i_2, \cdots, i_n\}$ of the agents in N: agent i_1 gets her best acceptable object in R_{i_1}, next agent i_2 gets his best remaining acceptable object in R_{i_2}, if any, and so on. This mechanism is clearly Strategy-Proof and Ordinally Efficient, thus its guaranteed size is at least $\frac{m}{2}$. By replicating the $\{\text{Ann,Bob}\} \times \{a, b\}$ example in the previous section, we see that its guaranteed size is exactly $\frac{m}{2}$ if m is even, and $\frac{m+1}{2}$ if it is odd. Moreover, the same example also shows that *the guaranteed size of any* *deterministic strategyproof mechanism cannot be more than* $\frac{1}{2}$ *if* m *is even, or* $\frac{1}{2}(1+\frac{1}{m})$ *if it is odd.*[4].

There is in fact no assignment mechanism meeting OE, EF, and SP (Theorem 2 in [8]). However the two benchmark mechanisms known as **Random Priority** (RP) and **Probabilistic Serial** (PS) almost fit the bill. Here we only discuss PS, postponing until section 7 the discussion of RP.

The simplest definition of the Probabilistic Serial (PS) mechanism PS is recursive.[5] Each agent fills his allocation by eating at constant speed 1, from time $t = 0$ until at most time $t = 1$, from her best acceptable object still available. At time 0, one unit of each object is available. For brevity we only illustrate the definition by an example with 5 agents and 4 objects. Here a is the best object for agents $1, 2, 3$, b is best for $4, 5$, and c, d for nobody. Then a is fully eaten at time $t = \frac{1}{3}$, and $1, 2, 3$ each get a $\frac{1}{3}$ share of it. Suppose agent 1 only accepts a, then she is done; say the next acceptable object is b for agent 2 and c for agent 3. Then starting from $t = \frac{1}{3}$ we have $2, 4, 5$ eating the remaining $\frac{1}{3}$ unit of b, thus b is exhausted at $t' = \frac{1}{3} + \frac{1}{9}$, and is divided in $\frac{4}{9}$ for each of 4 and 5, and $\frac{1}{9}$ for agent 2; and so on.

The PS mechanism is Ordinally Efficient, Envy-Free, but not Strategy-Proof. That is, under the premises of this axiom, the preference $p(i) \overset{sd_i}{\succeq} p'(i)$ may not hold; however the reverse strict preference $p'(i) \overset{sd_i}{\succ} p(i)$ does not hold either: based on her ordinal preferences only, an agent never has a compelling incentive to misreport his preferences. This latter property is called *Weak Strategy-Proofness.*

5 Size versus Envy-Freeness: the result

We compute first the guaranteed m-size index $\sigma_m(PS)$ of the PS mechanism. Then we show that this is the best feasible guaranteed size index for any Envy-Free mechanism.

The main clue comes from considering the following **canonical diagonal problem** *of size* m, denoted Δ^*_m. This problem already played a role in three relevant earlier papers: [18] , [12], and [9]. There are m agents $N = \{1, \cdots, m\}$ and m objects $A = \{a_1, \cdots, a_m\}$, and agent i's preferences are $R_i = (a_m, a_{m-1}, \cdots, a_i)$. One interpretation is of a scheduling problem where objects are time slots (higher label means earlier time) and agents are jobs that are processed in exactly one time slot; each job prefers an earlier slot, and job i has a deadline at time i (cannot be processed later than i). Here is Δ^*_5:

[4] At the profile where both Ann and Bob report that only a is acceptable, if a is not assigned, the size index is 0; if a is given to agent Bob, say, then by SP Bob still gets a at the canonical example.

[5] See [6] for another, more compact, though somewhat less transparent definition.

5	4	3	2	1
a_5	a_5	a_5	a_5	a_5
$\varnothing$	a_4	a_4	a_4	a_4
	$\varnothing$	a_3	a_3	a_3
		$\varnothing$	a_2	a_2
			$\varnothing$	a_1

Problem Δ_m^* is in $\mathcal{A}_m$ because we can assign object a_i to agent i for all i. In the PS eating algorithm, object a_m is eaten first by all agents, who each get a share $\frac{1}{m}$; next object a_{m-1} is eaten by agents $1, \cdots, m-1$, , who each get a share $\frac{1}{m-1}$; and so on until the critical object a_{k_m} such that

$$\frac{1}{k_m+1} + + \cdots + \frac{1}{m} \leq 1 < \frac{1}{k_m} + \frac{1}{k_m+1} + \cdots + \frac{1}{m}$$

Object a_{k_m+1} is eaten in full, but not so object a_{k_m}: agents $k_m, k_m-1, \cdots, 1$, can only eat a full unit, therefore their share of a_{k_m} is $1-(\frac{1}{k_m+1}+\frac{1}{k_m+2}+\cdots+\frac{1}{m})$ (which is less than $\frac{1}{k_m}$). Objects $a_{k_m-1}, \cdots, a_1$, are not eaten (consumed) at all.

Define for any integers $1 \leq k < m$

$$S(m,k) = \frac{1}{k+1} + \frac{1}{k+2} + \cdots + \frac{1}{m}$$

so that k_m is defined by the inequalities $S(m, k_m) \leq 1 < S(m, k_m - 1)$. We just saw that the assignment matrix of $PS(\Delta_m^*)$ is as follows. For all $i \in N, a_j \in A$

$$\begin{aligned} p_{ia_j} &= 0 \text{ if } i > j \text{ and/or } j < k_m \\ p_{ia_j} &= \frac{1}{j} \text{ if } i \leq j \text{ and } j \geq k_m + 1 \\ p_{ia_j} &= 1 - S(m, k_m) \text{ if } i \leq j \text{ and } j = k_m \end{aligned}$$

so that

$$s(PS(\Delta_m^*)) = \sum_{1 \leq i,j \leq m} p_{ia_j}$$

$$= m - k_m + k_m(1 - S(m, k_m)) = m - k_m S(m, k_m)$$

$$\Rightarrow \frac{1}{m} s(PS(\Delta_m^*)) = 1 - \frac{k_m}{m} S(m, k_m) \overset{def}{=} r_m$$

Recalling $\Delta_m^* \in \mathcal{A}_m$, this implies $\sigma_m(PS) \leq r_m$.

Lemma *The sequence r_m is decreasing and converges to $1 - \frac{1}{e} = 0.632$ at the speed $O(\frac{1}{n})$. For instance $r_2 = 0.750$, $r_3 = 0.722$, $r_4 = 0.708$, $r_5 = 0.687$, $r_{10} = 0.662$, $r_{20} = 0.648$.*

It turns out that the canonical diagonal profile achieves the worst possible size index for the PS mechanism, on all problems of $\mathcal{A}_m$.

Theorem

i) $\sigma_m(PS) = r_m$

ii) The m-size index of any Envy-Free mechanism is at most r_m.

There are inefficient Envy-Free mechanisms with a worst performance than PS: for instance we can assign objects sequentially, uniformly among all the still unmatched agents, throwing the object away if it is not acceptable to the winner; this gives the index $\frac{m+1}{2}$ at Δ_m^*.

We conjecture that the following refinement of statement *ii*) is true: *the m-size index of any Ordinally Efficient and Envy-Free mechanism is r_m.* The intuition comes from the following result about the class $\mathcal{D}^m$ of problems such that, for a common ordering $\{a_1, \cdots, a_m\}$ of the objects, all individual preferences take the form $R^k = (a_m, a_{m-1}, \cdots, a_k)$; thus $\mathcal{D}^m$ contains Δ_m^*, as well as problems with different numbers of preferences R^k for each k. Theorem 1 in [9] states that if F is Ordinally Efficient and Envy-Free, it coincides with PS on $\mathcal{D}^m$. The conjecture is that the problems Δ_m^* are also the worst case configuration for F.

6 The Random Priority mechanism

The RP mechanism runs the Fixed π-Priority mechanism after selecting π randomly and with uniform probability on all orderings of N. It is StrategyProof and Ex Post Efficient, but not Ordinally Efficient. Moreover, RP is not Envy-Free, that is to say in the assignment $P = RP(\Delta)$, the sd-preference $p(i) \overset{sd_i}{\succeq} p(j)$ may fail for some i, j; however $p(j) \overset{sd_i}{\succ} p(i)$ cannot hold either. In other words, based on her ordinal preferences R_i only, agent i never has a compelling reason to envy agent j's allocation. This latter property is called *Weak Envy-Freeness.*

Our Theorem is helpful to place an upper bound on the guaranteed m-size index of RP. Recall that Theorem 1 in [12] states that $s(RP(\Delta)) \leq s(PS(\Delta))$ for all m and all $\Delta \in \mathcal{D}^m$ (defined in the previous paragraph). In particular $s(RP(\Delta_m^*)) \leq s(PS(\Delta_m^*))$, and this inequality is strict as soon as $m \geq 4$. Combined with our Theorem, this implies $\sigma_m(RP) \leq r_m$.

Next [18] show that their Ranking algorithm yields the lower bound $1-(1-\frac{1}{m+1})^m$, which converges to $1-\frac{1}{e}$, precisely at the canonical diagonal profile Δ_m^*. Now Ranking is the same algorithm as Random Priority when preferences are identical (but acceptable objects vary across agents). From there [21] deduces the general lower bound $1-(1-\frac{1}{m+1})^m - \frac{1}{m}$ for RP.

We conclude that the performance of RP is inferior to that of PS, but not asymptotically so.

7 Concluding comments

1. Other worst case indices to measure the welfare performance of RP, PS, and other random assignment mechanisms, are proposed in [3]. Their *linear welfare factor* uses Borda scores as a proxy for cardinal utilities; the performance of PS

is nearly $\frac{2}{3}$, and is superior to that of RP. More work is needed to understand the connection of those results to ours.

2. Many concrete instances of assignments to jobs, schools, etc.., forces participants to report only a fixed number q_0 of acceptable objects, while other objects are deemed unacceptable by the mechanism. It is therefore natural to look for the the guaranteed sizes of RP and PS in this context.

3. In many assignment instances, there are exogenous differences between the agents so that it matters more to match some agents, or some objects, than others. An example is the assignment of overdemanded slots in Dutch universities, where a student record increases her probability of admission. The design objective is now to maximize a weighted sum of the matches, as discussed in [22] for bilateral matching, and in [13] for the assignment problem. The hard question is how should we adapt RP and PS to take this new objective into account?

8 Appendix: proofs

8.1 Lemma

Step 1 $\frac{k}{m}S(m,k) \leq \frac{1}{e}$ for all $k, 1 \leq k \leq m-1$
The Euler constant is the positive number C such that $\lim_m \varepsilon_m = 0$ where $\varepsilon_m \overset{def}{=} \ln(m) + C - (\sum_{j=1}^{m} \frac{1}{j})$. It is easy to check that ε_m decreases to zero, as $\varepsilon_{m+1} < \varepsilon_m \Leftrightarrow \ln(1+\frac{1}{n}) > \frac{1}{n+1}$, which follows from $\ln(1+x) > \frac{x}{x+1}$ for $x > 0$. This implies

$$S(m,k) = \ln(m) - \varepsilon_m - (\ln(k) - \varepsilon_k) \leq \ln(\frac{m}{k}) \tag{2}$$

Now for $x \in]0,1]$ we have $|x\ln(x)| \leq \frac{1}{e}$, hence $\frac{k}{m}S(m,k) \leq \frac{k}{m}\ln(\frac{m}{k}) \leq \frac{1}{e}$ as desired.

Step 2 $\frac{k_m}{m}S(m,k_m)$ increases strictly in m.
Compare k_m and k_{m+1}. We have $S(m+1, k_m-1) > S(m, k_m-1) > 1$ hence $k_{m+1} \geq k_m$. Moreover $S(m+1, k_m+1) \leq S(m, k_m) \leq 1$ implies $k_{m+1} \leq k_m+1$. We distinguish two cases. If $k_{m+1} = k_m = k$ we want to prove $\frac{1}{m+1}S(m+1,k) > \frac{1}{m}S(m,k)$ which easily reduces to $S(m+1,k) < 1$, and the latter is true by definition of k_{m+1}, and the fact that $S(m,k) = 1$ holds only for $m = 1, k = 0$. If $k_{m+1} = k_m + 1$, and we write simply $k_m = k$, a straighforward computation gives

$$\frac{k+1}{m+1}S(m+1,k+1) > \frac{k}{m}S(m,k)$$
$$\Leftrightarrow \frac{m-k}{m(m+1)}S(m,k+1) > \frac{k}{m(k+1)} - \frac{k+1}{(m+1)^2}$$
$$\Leftrightarrow S(m+1,k) > 1$$

and the latter inequality follows from the assumption $k_{m+1} > k$.

Step 3 $\lim_m \frac{k_m}{m}S(m,k_m) = \frac{1}{e}$

Set $\alpha_m = \frac{k_m}{m} S(m, k_m)$. By definition of k_m we have $1 - \frac{1}{k_m} \leq S(m, k_m) \leq 1$, implying $\frac{k_m}{m} - \frac{1}{m} \leq \alpha_m \leq \frac{k_m}{m}$. We know from Steps 1,2 that α_m converges to some $\alpha \leq \frac{1}{e}$, so that $\lim_m \frac{k_m}{m} = \alpha$ as well. In particular $\lim_m k_m = \infty$, therefore $\lim_m S(m, k_m) = 1$. From the equality in (2) we deduce $\lim_m \ln(\frac{m}{k_m}) = 1$, and the conclusion $\alpha = \frac{1}{e}$ follows.

8.2 Theorem

8.2.1 Statement $i)$

It remains to prove $\sigma_m(PS) \geq m \times r_m$.

Step 1 an auxiliary result

In this step we consider the variant of the model where in addition to N, A, R, a problem specifies a common positive capacity γ for each agent, and a profile of non negative capacities $\delta = (\delta_a)_{a \in A}$ for the objects. An *augmented* assignment problem is now $\widetilde{\Delta} = (N, A, R, \gamma, \delta)$, and an assignment is a $N \times A$ non negative matrix $P = [p_{ia}] \in \mathbb{R}_+^{N \times A}$ such that $\sum_N p_{ia} \leq \delta_a$ for all a and $\sum_A p_{ia} \leq \gamma$ for all i. We drop the probabilistic interpretation of P, where p_{ia} was the probabilty that agent i is assigned to object a, and think instead of the deterministic assignment of q divisible commodities, such that the initial endowment of good a is δ_a and agent i cannot consume more than γ units in total.The size of P is $s(P) = \sum_{N \times A} p_{ia}$ as before, and represents now the total capacity assigned at P. Note that $s(P) \leq \min\{n\gamma, \sum_A \delta_a\}$.

Although the RP mechanism is no longer defined, the eating algorithm runs for γ units of time and works as before, thus defining a feasible assignment $PS(\widetilde{\Delta})$.

Lemma 2 *Fix $\varepsilon > 0$ and two augmented problems $\widetilde{\Delta} = (N, A, R, \gamma, \delta)$, $\widetilde{\Delta}' = (N, A, R, \gamma, \delta')$, such that $\delta \leq \delta'$. Then*

$$s(PS(\widetilde{\Delta})) \leq s(PS(\widetilde{\Delta}')) \leq s(PS(\widetilde{\Delta})) + \sum_A (\delta'_a - \delta_a)$$

Proof By induction on the number of objects. The statement is obvious if $q = 1$. Fix now q and assume the property holds until $q-1$. Choose $\widetilde{\Delta}$, $\widetilde{\Delta}'$, two augmented problems with q objects, that only differ in that $\delta'_a = \delta_a + \varepsilon$ for a single object a and $\varepsilon > 0$. We must prove $s(P) \leq s(P') \leq s(P) + \varepsilon$, where P, P' are the corresponding assignments under PS. We write D, D' for the two corresponding eating algorithms, and $\delta_b(z)$, $\delta'_b(z)$ for the remaining capacity of object b at time z in D, D'.

If in D object a is fully consumed at time γ, then $D' = D$ and we are done. Now we assume that a "dies" at some time $t, t < \gamma$. If any other object dies at t in D, then D and D' coincide up to t, and the restriction of $D_{[t,\gamma]}, D'_{[t,\gamma]}$ to $[t, \gamma]$ is simply PS applied to two augmented problems with at most $q - 1$ objects, capacities $(\gamma - t)$ for agents, $\delta(t)$ and $\delta'(t)$ for objects, that only differ in that $\delta'_a(t) = \varepsilon$ while $\delta_a(t) = 0$, so we can apply the inductive assumption. Thus we assume now that only object a dies at t, and we define t' to be the first time after t where an object dies in D', or $t' = \gamma$ if there is no such object. Note that in D', a is not dead at t, and no agent can die or switch objects during the interval $[t, t']$, because this only happens when some object dies.

We check that $\delta_b(t') \leq \delta'_b(t')$ for all $b \in A$. This is clear for a because $\delta_a(t) = 0$, and also for any b that nobody is eating at t in D (and D'): in D' nobody switches object in $[t, t']$, thus nobody eats b in that interval. Consider finally $b, b \neq a$, that the agents in the subset N_b are eating at t in D (and D'): in D' the agents in N_b and only them continue to do so in $[t, t']$; in D the agents in N_b may be joined by new agents switching to b, and if b does not die before t' nobody switches in N_b, thus $\delta_b(t') \leq \delta'_b(t')$ as desired; this is also true if b dies in $[t, t']$.

We compare now $D_{[t',\gamma]}$ and $D'_{[t',\gamma]}$: they are PS applied to two augmented problems with at most $q - 1$ objects (for b dying at t' in D', we just showed $\delta_b(t') = 0$ as well), so by the inductive assumption

$$s(D_{[t',\gamma]}) \leq s(D'_{[t',\gamma]}) \leq s(D_{[t',\gamma]}) + \sum_{b \in A} (\delta'_b(t') - \delta_b(t')) \tag{3}$$

$$= s(D_{[t',\gamma]}) + \delta'_a(t') + \sum_{b \in A \smallsetminus \{a\}} (\delta'_b(t') - \delta_b(t'))$$

We also have two accounting identities

$$s(D_{[t,t']}) = \sum_{b \in A} (\delta_b(t) - \delta_b(t')) = \sum_{b \in A \smallsetminus \{a\}} (\delta_b(t) - \delta_b(t'))$$

$$s(D'_{[t,t']}) = \sum_{b \in A} (\delta'_b(t) - \delta'_b(t'))$$

$$= \varepsilon - \delta'_a(t') + \sum_{b \in A \smallsetminus \{a\}} (\delta'_b(t) - \delta'_b(t'))$$

and the equalities $D_{[0,t]} = D'_{[0,t]}$, $\delta_b(t) = \delta'_b(t)$ for all $b \neq a$. Combining those and the two previous equalities gives

$$s(D'_{[0,t']}) - s(D_{[0,t']}) = \varepsilon - \delta'_a(t') + \sum_{b \in A \smallsetminus \{a\}} (\delta_b(t') - \delta'_b(t'))$$

Plugging this in the right hand inequality in (3) gives $s(D') \leq s(D) + \varepsilon$. For inequality $s(D) \leq s(D')$, recall that in D', no agent still alive at t dies in $[t, t']$, and the agents still alive at t in D are a subset of those, therefore $s(D_{[t,t']}) \leq s(D'_{[t,t']})$ completing the proof.■

A useful consequence of Lemma 2 is the following monotonicity result:

Lemma 3 *Consider two (non augmented) problems* $\Delta = (N, A, R), \Delta' = (N, A, R')$ *where for all* $i \in N$, R'_i *is a truncation of* R_i*: for all* i *we have* $\{R'_i = R_i\}$ *or* $\{R_i = (a_1, \cdots, a_k),\ k \geq 2,$ *and* $R'_i = (a_1, \cdots, a_{k'})$ *with* $k' < k\}$ *or* $\{R_i = (a_1)$ *and* $R'_i = \varnothing\}$. *Then* $s(PS(\Delta')) \leq s(PS(\Delta))$.

Proof We use the the notation of the previous proof. It is enough to assume that a single agent i truncates her preferences from $R_i = (a_1, \cdots, a_k)$, $k \geq 2$, to $R'_i = (a_1, \cdots, a_{k-1})$, or from $R_i = (a_1)$ to $R'_i = \varnothing$. If in the PS algorithm D at R agent i eats no a_k, then the PS algorithm D' at R' is identical. If i eats α_k units of object a_k starting at time t, then it is the last object she eats. Therefore the restriction $\widetilde{D}$ of D to $N \diagdown \{i\}$ and to interval $[t, 1]$ is PS applied to the augmented problem $\widetilde{\Delta}$ with capacities $\gamma = 1 - t$ for agents, $\delta_b(t)$ for each $b \neq a_k$, and $\delta_{a_k}(t) - \alpha_k$ for object a_k. On the other hand agent i dies in D' at time t, and the restriction $\widetilde{D}'$ of D' to $[t, 1]$ is PS applied to the augmented problem $\widetilde{\Delta}'$ on $N \diagdown \{i\}$ with capacities $\gamma = 1 - t$, and $\delta_b(t)$ for all b. Therefore Lemma 2 implies

$$s(D'_{[t,1]}) = s(\widetilde{D}'_{[t,1]}) \leq s(\widetilde{D}_{[t,1]}) + \alpha_k = s(D_{[t,1]})$$

and the conclusion follows from combining this inequality with $D'_{[0,t]} = D_{[0,t]}$.■

Step 2 proof of statement $i)$

We fix now an arbitrary (non augmented) problem $\Delta_0 = (N, A, R)$ of size m, and we must prove $s(PS(\Delta_0)) \geq mr_m$. We construct first another problem Δ resembling the canonical diagonal problem Δ^*_m, and such that $s(PS(\Delta)) \leq s(PS(\Delta_0))$. Pick an efficient deterministic assignment $P \in \mathcal{P}^d(E(\Delta_0))$ where m agents are matched to m objects. It is well known, and easy to check, that we can order these agents $\{1, \cdots, m\}$ and these objects $\{a_m, \cdots, a_1\}$ in such a way that P assigns object a_i to agent i, so $a_i \in R_i$, and a_i is the best object for agent i among $\{a_i, \cdots, a_1\}$ (some of which may not be acceptable to i). By Lemma 3 if we fix $R_i = \varnothing$ for all agents unmatched at P, and for each $i \in \{1, \cdots, m\}$ we truncate R_i at a_i, thus making all objects $\{a_i, \cdots, a_1\}$ unacceptable, then the expected size of the resulting problem Δ is weakly smaller than at Δ_0.

We now show $s(PS(\Delta)) \geq mr_m$. Let $\{i_1, i_2, \cdots, i_H\}$ the set of agents in $\{1, \cdots, m\}$ who do not get a full allocation in $PS(\Delta)$ ($\sum_A p_{ia} < 1$), ordered according to the time $t_1 \leq t_2 \leq \cdots \leq t_H$ at which they die in the PS algorithm. Set $\tau_h = t_h - t_{h-1}$, with the convention $t_0 = 0$. Then agent i_h eats $\sum_{l=1}^{h} \tau_l$, therefore

$$s(PS(\Delta)) = m - H + \sum_{h=1}^{H} (H + 1 - h)\tau_h$$

We set $k = m - H$ and list H inequalities that the non negative numbers τ_h must satisfy:

$(k + H)\tau_1 \geq 1$, because at least object a_{i_1} is dead at t_1;

$(k + H)\tau_1 + (k + H - 1)\tau_2 \geq 2$, because at least objects a_{i_1}, a_{i_2} are dead at t_2, and in $[t_1, t_2]$ one agent is absent;

and for all $h, 1 \leq h \leq H$:

$$\sum_{l=1}^{h} (k + H + 1 - l)\tau_l \geq h \tag{4}$$

because objects $a_{i_1}, \cdots, a_{i_h}$ are dead at t_h, and $l - 1$ agents are dead in $[t_{l-1}, t_l]$.

Define $\Theta = \{\tau = (\tau_h) \in \mathbb{R}_+^H | \tau \text{ meets (4) for all } h, 1 \leq h \leq H\}$. Then $s(PS(\Delta)) \geq k + \min_{\tau \in \Theta} \sum_{h=1}^{H} (H+1-h)\tau_h$. We claim that the value of the latter program is $\sum_{h=H}^{1} \frac{h}{k+h}$. To check this, we change variables to $\lambda_h = (k+H+1-h)\tau_h$, so the program becomes

$$\min \sum_{h=1}^{H} \frac{(H+1-h)}{k+H+1-h} \lambda_h$$

$$\text{such that } \lambda \geq 0 \text{ and } \sum_{l=1}^{h} \lambda_l \geq h \text{ for all } h, 1 \leq h \leq H$$

Its optimal solution is $\lambda_h = 1$ for all h. Indeed if $\lambda_1 > 1$, a transfer from λ_1 to λ_2 lowers the objective, so λ_1 must be 1; and so on.

We just proved $s(PS(\Delta)) \geq k + \sum_{h=H}^{1} \frac{h}{k+h}$, and this sum is $k + \sum_{h=H}^{1}(1 - \frac{k}{k+h}) = m - kS(m,k)$. Finally we check that the sequence $k \to kS(m,k)$ is single-peaked with its peak at k_m, implying $s(PS(\Delta)) \geq m - k_m S(m, k_m) = mr_m$. This is because the inequality $kS(m,k) \geq (k+1)S(m,k+1)$ (resp. $<$) is rearranged as $S(m,k) \leq 1$ (resp. $S(m,k) > 1$).

8.2.2 Statement *ii)*

Consider the canonical diagonal profile Δ_m^* and an Envy-Free assignment $P \in \mathcal{P}(E(\Delta_m^*))$. We check $s(P) \leq mr_m$.

Because a_m is the top object for everyone, EF implies $p_{ia_m} = p_{ja_m} = x_m$ for all i,j. Because a_{m-1} is the second best object for agents $1, \cdots, m-1$, and they all eat the same amount of a_m, EF implies $p_{ia_{m-1}} = p_{ja_{m-1}} = x_{m-1}$ for all $i,j \leq m-1$. Repeating the argument we see that for all k, $p_{ia_k} = x_k$ is independent of $i \leq k$. Feasibility w.r.t objects gives $kx_k \leq 1$, and w.r.t. agent 1 it gives $\sum_{k=1}^{m} x_k \leq 1$. Moreover $s(P) = \sum_{k=1}^{m} kx_k$. Now we claim

$$mr_m = \max_{x \in \mathbb{R}_+^m} \{\sum_{k=1}^{m} kx_k | \sum_{k=1}^{m} x_k \leq 1 \ ; \ kx_k \leq 1 \text{ all } k\}$$

If x is optimal, $x_k > 0$ and $x_{k+1} < \frac{1}{k+1}$ cannot both be true, otherwise a transfer from x_k to x_{k+1} improves the objective. Hence there is at most one k^* such that $0 < x_{k^*} < \frac{1}{k^*}$, and then $x_k = 0$ for $k < k^*$, and $x_k = \frac{1}{k}$ for $k > k^*$. Call this case 1. Case 2 is when no such k^* exists, then $x_k = 0$ up to some $\widetilde{k}$, after which $x_k = \frac{1}{k}$.

In Case 1 we have $\sum_{k=1}^{m} x_k = S(m,k^*) + x_{k^*} \leq 1$, in particular $S(m,k^*) \leq 1$. Moreover this constraint must be tight, else we can improve the objective by raising x_{k^*}. Therefore $1 - S(m,k^*) = x_{k^*} < \frac{1}{k^*} \Leftrightarrow S(m,k^*-1) > 1 \Rightarrow k^* = k_m$. Now $\sum_{k=1}^{m} kx_k = m - k^* + k^* x_{k^*} = m - k_m S(m,k_m)$ as desired.

In Case 2 we have $\sum_{k=1}^{m} x_k = S(m,\widetilde{k}) \leq 1$, implying $\widetilde{k} \geq k_m$. Moreover $\sum_{k=1}^{m} kx_k = m - \widetilde{k} \Rightarrow \sum_{k=1}^{m} kx_k \leq m - k_m \leq m - k_m S(m,k_m)$.

References

[1] A. Abdulkadiroğlu and T. Sonmez , *Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems,* Econometrica 66: 689-701, May 1998

[2] A. Abdulkadiroğlu and T. Sonmez , *School Choice: A Mechanism Design Approach* (joint with Atila Abdulkadiroğlu), American Economic Review 93-3: 729-747, June 2003

[3] A. Bhalgat, D. Chakrabarty and S. Khanna, *Social Welfare in One-Sided Matching Markets without Money,* Approx/Random 2011, pp. 87-98, 2011

[4] Birnbaum B. and C. Mathieu, *On-line Bipartite Matching Made Simple*, ACM SIGACT News, vol 39, issue 1, March 2008

[5] P. Biró, D.F. Manlove and S. Mittal, *Size versus stability in the Marriage problem*, Theoretical Computer Science, 411 : 1828-1841, 2010

[6] A. Bogomolnaia, *Random Assignment: Redefining the Serial Rule*, mimeo, Glasgow University

[7] A. Bogomolnaia, and E. J. Heo, *Probabilistic Assignment of Objects: Characterizing the Serial Rule*, Journal of Economic Theory, vol. 147, 5, 2072-2082, 2012

[8] A. Bogomolnaia and H. Moulin, *A New Solution to the Random Assignment Problem*, Journal of Economic Theory, 100, 295-328, 2001

[9] A. Bogomolnaia and H. Moulin, *A Simple Random Assignment Problem with A Unique Solution*, Economic Theory, 19, 3, 623-636, 2002

[10] E. Budish, Y-K Che, F. Kojima, and P. Milgrom, *Designing Random Allocation Mechanisms: Theory and Applications*, American Economic Review, 2013

[11] Y-K Che and F. Kojima, *Asymptotic equivalence of probabilistic serial and random priority mechanisms*, Econometrica, Vol. 78, No. 5 (September, 2010), 1625–1672

[12] H. Cres and H. Moulin, *Scheduling with Opting Out: Improving upon Random Priority*, Operations Research, 49, 4, 565-577, 2001

[13] S. Dughmi and A. Ghosh. *Truthful Assignment without Money*, Proceedings of the 11th ACM conference on Electronic commerce (EC 10), 325-334, 2010

[14] A. Erdil, *Strategy-proof Stochastic Assignment,* Journal of Economic Theory, forthcoming 2014

[15] T. Hashimoto, D. Hirata, O. Kesten, M. Kurino, and U. Unver, *Two Axiomatic Approaches to the Probabilistic Serial Mechanism*, Theoretical Economics, forthcoming

[16] R.W. Irving and D.F. Manlove, *Finding large stable matchings,* ACM Journal of Experimental Algorithmics, volume 14, section 1, article 2, 30 pages, 2009

[17] B. Kalyanasundaram and K. R. Pruhs, *An optimal deterministic algorithm for online b-matching*, Theoretical Computer Science 233, 319-325, 2000

[18] R. Karp, U. Vazirani, V. Vazirani, *An optimal algorithm for online bipartite matching*, ACM Symposium on the Theory of Computing, pp. 352-358, 1990

[19] F. Kojima and U. Unver *The "Boston" School-Choice Mechanism: An Axiomatic Approach*, Economic Theory, forthcoming

[20] P. Krysta, and D. Manlove, *Efficient Algorithms for Mechanism Design Without Monetary Transfer,* EPSRC Grants # EP/K01000X/1 and EP/K010042/1, June 2013

[21] P. Krysta, D. Manlove, B. Rastegari and J. Zhang, *private communication,* September 2013.

[22] D.F. Manlove, *Algorithmics of Matching Under Preferences*, World Scientific, 2013.

[23] A. Procaccia and M. Tennenholtz , *Approximate mechanism design without money*, Proceedings of the 10th ACM conference on Electronic commerce (EC 09), 177-186, 2009

[24] T. Sonmez and U. Unver, *Matching, Allocation, and Exchange of Discrete Resources*, Handbook of Social Economics, J. Benhabib, A. Bisin, and M. Jackson (eds.), vol 1A 781-852, North-Holland,2011