Eranism: a Need-Based Aid Economy

Part 3: Algorithms for Sequential and Fixed-Path Allotment

Arnold J. Bomans*

June 29, 2021 draft†

Abstract

Two algorithms for allotment along a fixed path are given. One is equipped with a floor function to handle indivisible goods. This variation is shown to be a case of sequential allotment, for which a general algorithm is also given. The conclusion indicates which properties of both types of allotment carry over to each case of them. The assignment of "bads" is shown to be the allocation of negative goods.

*abomans@hotmail.com

†The margins indicate possible improvements. Syntactical, linguistic, and stylistic errors are to be corrected.

Contents

1 Introduction

Eranism (see Part 1) has a strategy-proof allocation already. For, if a person asks more goods than needed in reality then the time to produce them raises the general upper bound on labour time, which implies that the person might have to work longer than when the true amount of goods were asked. However, this encouragement is rather weak. Therefore, the allotment proper should be strategy-proof.

Candidate non-monetary mechanisms for this are partial allocation [8] or, in presence of scarcity and priorities, sequential allotment (as used here) if not any of its generalisations [19, **?**]. If goods are distributed along certain pathways, then they can be made to obey egalitarian transfer [5] or, more generally, the uniform gains rule [20]. If other natural desiderata are imposed, then the theorems below show that little choice is left: fixed-path or sequential alloation. improve overview

Allotment along a fixed path is characterized in [18]. Sequential allotment is characterized in [4] and, especially for indivisible goods, in [3]. This document presents two algorithms for fixed-path allotment, one using an index (the residual) and the other based on an iteration ("terminal adjustment") which is cast as sequential allotment. Also a quite general algorithm for sequential allotment is given to support an example. (No such algorithms are presented in the aforementioned literature.)

For indivisible items, the uniform rule has been implemented as a kind of round-robbin allocation, where goods are distributed in rounds along the agents and at each round, at most one item is allotted to each agent. (See Example 1, p.655 of [19].) Here, the iteration for fixed-path allotment is equiped with a floor function. The result can be cast as a sequential allotment, so it inherits some desirable properties, as summarized in the conclusion.

2 Preliminary Definitions

First, some general notation [18]. Let Z be either the real numbers or the integers, both positive and negative. Let $[a, b] := \{z \in Z \mid a \leq z \leq b\}$ for real numbers a and b. Let there be n *agents* for $n \geq 2$ and let $N := \{1, \ldots, n\}$. For z in Z^n let $1z := \sum_{i=1}^{n} z_i$, which is the inner product of the row vector z and the column vector having entries 1 of suitable dimension.[1] Further, let

[1] This notation is used in [21, p.595].

$z_{i,\ldots,j} := z|_{\{i,\ldots,j\}}$ be the projection (restriction of z as a function) to $i,\ldots,j$ for some i and j in N. So $1z_{i,\ldots,j} = \sum_{k=i}^{j} z_k$. Let Z^n be partially ordered as follows: For vectors x and y in Z^n let $x \leq y$ denote $x_i \leq y_i$ for all i in N. The inverse of a function f is denoted by $f^{\leftarrow}$ to avoid confusion with $f^{-1} = 1/f$.

2.1 Demands and Burdens

For i in N let $X_i \geq 0$ in Z be *bounds*. Let $\wp_+ := \prod_{i=1}^{n}[0, X_i]$ and $\wp_- := \prod_{i=1}^{n}[-X_i, 0]$. Let $\wp := \wp_+ \cup \wp_-$ be the vectors either having all elements non-negative or having all elements non-positive. Let x in $\wp$ be the *ideal points* of agents and r in $\wp$ their *reported* ideal points. Positive vectors x or r are a *demand* and vectors in $\wp_-$ are a *burden* mimicked by a negative "demand." Let r_0 in $[-1X, 1X]$ be an *artificial point* which is the same for all r. It need not always be known and is not a coordinate of r.

The remainder of this section is intended for the fixed-path mechanism but treated prior to it because it is independent of M, the amount to be allotted.

2.2 Indices

The following is only needed for the method of indices, not for any iterative computations. Let Θ be a subset of the real-valued numbers which contains a smallest and a greatest element. Suppose x_i in Θ for $i = 0, 1, \ldots, n$ are increasing: $x_0 \leq x_1 \leq \cdots \leq x_n$. Let $x = (x_1, x_2, \ldots, x_n)$ and suppose $x_0 = \min \Theta$, that is, x_0 is an artificial element independent of x. Let θ in Θ.

Definition 1 (Index) *If $\theta < x_n$ then let $b_x(\theta) := \min\{i = 1, \ldots, n \mid x_i > \theta\}$ be the* upper index *or* break index[2] *of x. For any θ the* lower index *or* residual[3] *of x is given by $\overline{x}(\theta) := \max\{i = 0, 1, \ldots, n \mid x_i \leq \theta\}$.*

Notice that x_0 is not derived from the parameter x, that b_x does not need x_0, and that $\theta \geq x_n$ would imply that b_x is undefined.[4] If $b_x(\theta)$ is mentioned then $\theta < x_n$ is presuppoed.

[2] The term 'first passage time' is reserved for the equal sign [14, pp.121, 264].

[3] In stochastics, it is the renewal process [14, p.279].

[4] Extending b_x for $x_{n+1} = \max \Theta$ would leave $b_x(x_{n+1})$ undefined.

Lemma 1 (Index) *The lower index is well-defined. If confined to* $\{x_i \mid i = 0, \ldots, n\}$ *it is the inverse of* x *restricted to* $\overline{x}(\Theta)$*, which is the range of* $\overline{x}$ *applied to* Θ*. So, with those restrictions,* $x_{\overline{x}(\theta)} = \theta$ *and* $\overline{x}(x_i) = i$*. For* $a = \overline{x}(\theta)$ *one has* $\theta \in [x_a, x_{a+1})$ *if* $a = 0, \ldots, n-1$ *and* $\theta \in [x_n, \max\Theta]$ *if* $a = n$*. For* $b = b_x(\theta)$ *one has* $\theta \in [x_{b-1}, x_b)$ *and* $b = a+1$ *if* $a = 0, \ldots, n-1$*.*

Proof From the appendix: $\overline{x}(\theta)$ is the residual of the series x; it is well-defined because the least element of Θ is in the range of x (this can also be checked directly.) By definition, $x_i \leq \theta$ for all $i = 0, \ldots, n$. Suppose $a = 0, \ldots, n-1$. Then $\theta < x_{a+1}$. For, if $\theta \geq x_{a+1}$ were the case, then $\overline{x}(\theta) \geq a+1$, a contradiction.

It is easily seen that $a \leq n-1$ is equivalent to $\theta < x_n$. As to the upper index, its existence ($\theta < x_n$) implies $a+1 \leq n$. From θ in $[x_a, x_{a+1})$ follows that x_i surpasses θ if i goes from a to $a+1$. So, $b = a+1$ by definition of b and this in turn yields the interval indexed by b. ■

Notice that the interval $[x_a, x_{a+1})$ is not empty as $x_a < x_{a+1}$ for $a = \overline{x}(\theta)$. For example, suppose $\min\Theta = x_0 = x_1 < x_2$ then $\overline{x}(x_0) = 1$ so x_0 in $[x_1, x_2)$.

3 Sum of Minima

The sum of minima is an expression depending on a number λ which is to be found for fixed-path allotment. Using a vector instead of a number λ is subject of sequential allotment.

Let $W := [-1X, 1X]$ and $g : W \to \wp$ be monotonous, that is, in every coordinate.

Example 1 For $n = 2$ let $g_1(\lambda) := \lfloor \frac{\lambda}{2} \rfloor$ and $g_2(\lambda) := \lceil \frac{\lambda}{2} \rceil$. Their sum is λ (distinguish even and odd λ) but agent 1 will consider g unfair for the allocation problem below. ■

Definition 2 (Sum of Minima) *For* λ *in* W *define* $z_i(\lambda) := \min\{r_i, g_i(\lambda)\}$ *and* $h(\lambda) := 1z(\lambda)$*. Also let* $y_i := z_i(\lambda)$*.*

So $h(\lambda) = \sum_{i=1}^{n} \min\{r_i, g_i(\lambda)\}$. Notice that h and z are parametrized by g and r. reconsider bounds

If λ is to solve the allocation problem, then r_n will not meet the condition $r_n \leq g_n(\lambda)$. Suppose λ exists so that $r_n \leq g_n(\lambda)$. Then $r_j \leq g_j(\lambda)$ for all j in N. Let $\Lambda_+(r) := \min\{\lambda \in W \mid r \leq g(\lambda)\}$. (According to the appendix, g is the residual of Λ_+, that is, $\overline{\Lambda_+} = g$.) Of course, $r \leq g(\lambda)$ is equivalent to $h(\lambda) = 1r$. The order of the coordinates of r implies that $\Lambda_+(r) = \min\{\lambda \in W \mid r_n \leq g_n(\lambda)\}$ once this minimum exists. Let $\Lambda_-(r) := \arg\min_{\lambda \in W} 1g(\lambda)$ be the λ for which $h(\lambda)$ is least. So $\Lambda_-(r) = -1X$. Let $H_\pm := h(\Lambda_\pm(r))$ so $H_- := 1g(-1X)$ and $H_+ := 1r$. Let $[H_-, H_+]$ be the *range* of h.

Definition 3 (Auxiliary Variables) *Let $g_0 : W \to \wp$ be a strictly monotonous function and let $r_0 := g_0(\Lambda_-(r))$. For $j = 0, \ldots, n$ define the function $G_j := \sum_{i=j}^{n} g_i$.*

For $j = 1, \ldots, n$ let $R_j := \sum_{i=1}^{j-1} r_i$ with the convention that $R_1 = 0$.

The definition of r_0 allows Definition 2, p.7 of a residual. Ordering the r_i for $i = 0, 1, \ldots, n$ for non-invertible g is only of theoretical interest so-far. It is helpful to think of the agents as runners i who have set personal goals r_j. They start running at the same time and follow a course $g_i(\lambda)$ as time λ proceeds. If they reach r_j then they stay there (their demand r_j is capped by $g_i(\lambda)$.) Runner 1 is the one who reaches some r_j first, number 2 who arrives at some r_k next, and so on. The strict order is characterized by the fact that runner 2 is still running if runner 1 is but there also is a time when runner 1 has stopped and runner 2 has not.

Definition 4 (Order) *Define the* shortage $\alpha_\lambda(i) := r_i - g_i(\lambda)$ *for $i = 0, 1, \ldots, n$. Let $D_i := \{\lambda \in W \mid \alpha_\lambda(i)\}$ contain the λ for which the demand r_i is honoured, that is, not capped by $g_i(\lambda)$. Let the order of r_0 and r be given by $r_0, \ldots, r_i, \ldots, r_j, \ldots, r_n$ (where $i \leq j$) exactly if $D_i \subset D_j$.*

So, the coordinates of r are ordered by the index i for which r_i is first surpassed by $g_i(\lambda)$ as λ increases. In particular, $(r_0) = g_0(\Lambda_-) \leq g_0(\lambda)$ for all λ'

A construction would be as follows. Suppose the r_i are still unordered for

i in N. Let

$$\begin{array}{ll}
\lambda_1 & := \inf\{\lambda \mid \exists i \in N : r_i < g_i(\lambda)\} \\
i_1 & \in \{i \in N \mid r_i < g_i(\lambda_1)\} \\
\lambda_2 & := \inf\{\lambda \mid \exists i \in N \setminus \{i_1\} : r_i < g_i(\lambda)\} \\
i_2 & \in \{i \in N \setminus \{i_1\} \mid r_i < g_i(\lambda_2)\} \\
\vdots & \vdots \\
\lambda_{n-1} & := \inf\{\lambda \mid \exists i \in N \setminus \{i_1, \ldots, i_{n-2}\} : r_i < g_i(\lambda)\} \\
i_{n-1} & \in \{i \in N \setminus \{i_1, \ldots, i_{n-2}\} \mid r_i < g_i(\lambda_{n-1})\}
\end{array}$$

and let i_n be the remaining element of $N \setminus \{i_1, \ldots, i_{n-1}\}$. Finally, renumber r_i as r_{i_k} for k in N. The r_0 is positioned first.

3.1 Expression for Sum of Minima

In the following, the method of indices is used to express the sum of minima. This method can be used in spread sheets which do not allow macros, but each cell expression would have to contain many absolute references, that need to be manually changed when defining a new allocation, like for the next week. So, the interval index is more useful for deriving new formulas.

Lemma 2 (Expression for Sum of Minima) *The residual (see appendix) of α_λ evaluated in 0 is:*

$$\overline{\alpha_\lambda}(0) = \max\{j = 0, \ldots, n \mid r_j \leq g_j(\lambda)\}$$

For $a := \overline{\alpha_\lambda}(0)$ one has $h(\lambda) = R_{a+1} + G_{a+1}(\lambda)$.

Proof The a is well-defined because of the definition r_0 and the order of r. Further, $h(\lambda) = \sum_{i=1}^{a} r_i + \sum_{i=a+1}^{n} g_i(\lambda)$. ∎

Without consulting the appendix, the expression $\overline{\alpha_\lambda}(0)$ can also be seen as a notation with a superfluous 0 attached. The inverse for the continuous case will be computed in the next section but its existence is announced in the following lemma.

Lemma 3 (Inverses) *The equation $h(\lambda) = \mu$ has a solution for μ in $[H_-, H_+]$ if g is continuous or g is defined on the integers. On the range $h[\Lambda_-, \Lambda_+)$ one such solution is $\overline{h}(\mu) = \max\{\lambda \in [\Lambda_-, \Lambda_+) \mid h(\lambda) \leq \mu\}$.*

If all g_i are invertible (in the continuous case: strictly monotonous) then the solution is unique for μ in $[H_-, H_+)$. For λ in $[\Lambda_+(r), 1X]$ the value of $h(\lambda)$ is the constant H_+.

Proof First, existence. For the continuous problem, this follows from the intermediate value theorem. Now for the discrete problem. From the feasibility condition follows that $\sum_{i=1}^{N} g_i(\lambda+1)-g_i(\lambda) = 1$. As every term is positive, for any L subset of $\{1,\ldots,n\}$ one has $\sum_{\ell\in L} g_\ell(\lambda+1)-g_\ell(\lambda) \leq 1$. Let I, J, and K be subsets of $\{1,\ldots,n\}$ so that $h(\lambda) = \sum_{i\in I} r_i+\sum_{j\in J} g_j(\lambda)$ and $g_k(\lambda) < r_k \leq g_k(\lambda+1)$ for all k in K. So $h(\lambda+1) = \sum_{i\in I\cup K} r_i + \sum_{j\in J\setminus K} g_j(\lambda+1)$. Then $h(\lambda+1)-h(\lambda) = y+\sum_{j\in J} g_j(\lambda+1)-g_j(\lambda)$ for $y := \sum_{k\in K} r_k - g_k(\lambda+1) \leq 0$. Therefore, h increases by at most 1 if λ grows by 1.

Second, uniqueness. For λ in $[-1X, \Lambda_+(r))$ there is an i in N so that $r_i \geq g_i(\lambda)$ for otherwise, $r \leq g(\lambda)$, that is, $\lambda \geq \Lambda_+(r)$, a contradiction. So, $h(\lambda) = \sum_{j\in N} z_j(\lambda)$ is the sum of monotonous functions z_j of which at least $z_i = g_i$ is strictly monotonous. Therefore, h is strictly monotonous and the inverse exists. ■

Notice that the discrete case has a solution by virtue of the feasibility constraint.

Example 2 For $n = 2$ let $g_1(\lambda) := \lfloor\lambda\rfloor$ $g_2(\lambda) := \lfloor 2\lambda\rfloor$ which does not meet the feasibility condition. Let $r = (9,9)$ so $h(\lambda) = g_1(\lambda)+g_2(\lambda)$. If λ in $[\frac{1}{2}i, \frac{1}{2}+\frac{1}{2}i)$ where $i = 0,1,2,3,4$ then $h(\lambda) = 0,1,3,4,6$ respectively. ■

3.2 Inverse of Sum of Minima

The method of indices is used again to invert the sum of maxima. Suppose in this section that g is continuous and strictly monotonous. For any real-valued n-vector x introduce $x'_i := g_i^{\leftarrow}(x_i)$ for all i in N. So, the order the entries r_i of r now is determined as follows:

$$r'_0 \leq r'_1 \leq \cdots \leq r'_n$$

This easily follows from Definition 4, p.6 by taking $\lambda = r'_j$. Note that $r'_0 = \Lambda_-(r)$.

Proposition 1 (Piecewise Graph) *If λ in $[-1X, r'_n)$ then $h(\lambda) = R_b + G_b(\lambda)$ where $b := b_{r'}(\lambda) = \min\{j \in N \mid r'_j > \lambda\}$. Also, $h(\lambda) = R_{a+1} + G_{a+1}(\lambda)$ for $a = \max\{i = 0, 1, \dots, n \mid r'_i \leq \lambda\}$.*

Proof As $\Lambda_+(r) = r'_n$ the range of λ is as indicated. The graph of h is easily drawn. This yields $h(\lambda) = R_k + G_k(\lambda)$ if $r'_{k-1} \leq \lambda < r'_k$ for some k in N which is unique because the intervals do not overlap. (Some intervals are empty, though.) The k equals b as indicated.

An alternative proof is to write $h(\lambda) = \sum_{i=1}^{n} g_i(\min\{r'_i, \lambda\})$ because a monotonous function respects the minimum, as is easy to see. So $h(\lambda) = \sum_{i=1}^{b-1} r_i + \sum_{i=b}^{n} g_i(\lambda)$.

Lemma 2, p.7 yields $a = \overline{a_\lambda}(0)$. Observe that $a = \max\{i = 0, \dots, n \mid r'_i \leq \lambda\}$. As in the appendix, $a = \overline{r'}(\lambda)$, the residual of r'_i as a function of i. Lemma 1, p.5 gives $b = a + 1$. ■

The following illustrates Proposition 1, p.9.

Example 3 For *weighted gains* (see below) $g_i(\lambda) = w_i \lambda$ for weights w_i. So $G_b(\lambda) = (w_b + \cdots + w_n)\lambda$. For example, let $r = (1, 2, 3)$ and $w = (2, 3, 4)$. If $\lambda = \frac{1}{2}$ then $b = 2$ and $h(\lambda) = \frac{9}{2}$ while if $\lambda = \frac{1}{4}$ then $b = 1$ and $h(\lambda) = \frac{9}{4}$, as should be. For $w_1 = \cdots = w_n$ (the *uniform rule*) $h(\lambda) = R_b + (n - b + 1)\lambda$. ■

As g is invertible, the following definition is allowed: $L_k := R_k + G_k(r'_k)$ for $k = 0, \dots, n$. For R_k and G_k see Definition 3, p.6. Notice that g_0 and r_0 are used for $k = 0$.

Lemma 4 (Index Shift) *For $k = 0, \dots, n$ one has $L_k = R_{k+1} + G_{k+1}(r'_k)$ and $h(r'_k) = L_k$.*

Proof The first term of $G_k(r'_k)$ is $g_k(r'_k) = r_k$. (Even for $k = n$ this first term exists. For $k = 0$ one can add r_0 to $R_0 = 0$ to yield $R_1 = r_0$.) Further, Proposition 1, p.9 yields $h(r'_k) = R_b + G_b(r'_k)$ for $b = b_{r'}(r'_k) = \min\{j \in N \mid r'_j > r'_k\} = k + 1$. So $h(r'_k) = R_{k+1} + G_{k+1}(r'_k)$. ■

The function L is used in the break index in the following.

Proposition 2 (Inverse) *For μ in $[H_-, H_+)$ the inverse of h is given by $h^{\leftarrow}(\mu) = G_b^{\leftarrow}(\mu - R_b)$ where $b = b_L(\mu)$ the break index, that is, $b = \min\{k \in N \mid R_k + G_k(r'_k) > \mu\}$.*

Proof The inverse of h is easily derived by expanding the above for different k and drawing a graph:

$$\begin{array}{ll} h([r'_0, r'_1)) & = [h(r'_0), h(r'_1)) \\ h([r'_1, r'_2)) & = [h(r'_1), h(r'_2)) \\ \vdots & \vdots \\ h([r'_{n-1}, r'_n)) & = [h(r'_{n-1}), h(r'_n)) \end{array}$$

The result is $h^{\leftarrow}(\mu) = G_k^{\leftarrow}(\mu - R_k)$ if $L_{k-1} \leq \mu < L_k$ for some k in N. This k is the break index, which is well-defined because $\mu < h(r'_n)$. ■

Lemma 4, p.9 yields that in this proof, the interval for μ reads $h(r'_{k-1}) \leq \mu < h(r'_k)$. This better corresponds to the graphical representation. Also $k = b_{h \circ r'}(\mu) = \min\{j \in N \mid h(r'_j) > \mu\}$ is an alternative index which returns the original interval $[L_{k-1}, L_k)$. Replace L_i with $L_\lambda(i)$

Let $\tilde{h}(\lambda) := \sum_{i=1}^{n} \min\{r'_i, \lambda\}$. Trivially, the constant $\tilde{M} := \tilde{h}(h^{\leftarrow}(M))$ has the property that $\tilde{h}(\lambda) = \tilde{M}$. So, the original problem can be cast as the uniform rule, but only after the original problem is solved.

Finally, here are two observations that little is gained so-far from order theory, as summarized in the appendix.

First, let $f : N \to Z$ be some function. So f preserves any supremum of the indices (which trivially is a maximum.) That is, $f(\sup V) = \sup f(V)$ for any subset V of the integers, where $f(V) = \{f(v) \mid v \in V\}$. Let $V := \{b\}$ for b as in Lemma 2, p.10. Let $f(i) := R_i$, which is monotonic. Then $R_b = \min\{R_i \mid i \in N \ \&\ L_i > \mu\}$. So, little is gained from this.

Second, let $a := \max\{i = 0, \ldots, n \mid L_i \leq \mu$. (Note: $a - 1 = b$ in Lemma 2, p.10.) So $a = \overline{L}(\mu)$, the residual of L, so L is monotonic in i, which is no news.

Example 4 Consider again weighted gains, that is, $g_k(\lambda) = w_k\lambda$ but this time for weights $w_k > 0$ where k in N. Then $L_k = R_k + (w_k + \cdots + w_n)r_k/w_k$ for $k = 0, \ldots, n$ and $w_0 \neq 0$ arbitrary. Removing the unnecessary division by w_k yields $L_k = \alpha_k + \beta_k r'_k$ where $\alpha_k := R_{k+1} = r_1 + \cdots + r_k$ and $\beta_k := w_{k+1} + \cdots + w_n$. So

$$\begin{aligned} h^{\leftarrow}(\mu) = \lambda &= (\mu - \alpha_b + r_b)/(w_b + \beta_b) \\ &= (\mu - r_1 - \cdots - r_{b-1})/(w_b + \cdots + w_n) \end{aligned}$$

where $b = \min\{j \in N \mid L_j > \mu\}$. ∎

The following applies to y_i as in Definition 2, p.5.

Lemma 5 *Suppose g is invertible. First, once $r'_i \geq \lambda$ for some i in N, then $y'_j = y'_i$ for all $j \geq i$. Second, $y'_1 \leq y'_2 \leq \cdots \leq y'_n$.*

Proof First, $\lambda \leq r'_i \leq r'_j$. So $g_j(\lambda) \leq r_j$ and therefore, $y_j = g_j(\lambda)$, whence $y'_j = \lambda = y'_i$. Second, consider i and $j \geq i$. If $r'_i > \lambda$ then $y'_i = y'_j$ by virtue of the previous result. Now consider $r'_i \leq \lambda$. Then $y'_i = r'_i$ so $y'_i \leq \lambda$. If $r'_j > \lambda$ then $y'_j = \lambda \geq y'_i$. If $r'_j \leq \lambda$ then $y'_j = r'_j \geq r'_i = y'_i$. ∎

This can probably be generalized to non-invertible g and also simplified.

4 Rationing and Assignment

Now for the problem at hand [4]. Let *allotment* mean rationing of a good or assignment of a burden. The mapping g (Paragraph 3, p.5) is called an *N-path* if it obeys a feasibility condition: $1g(\lambda) = \lambda$ for all λ in W. (This allows the possibility $1g(\lambda) = M$ for the allocation problem. The condition is used in Lemma 3, p.7.) As all coordinates of g are either all positive or all negative, this implies $g(0) = 0$.

Let M in W (Paragraph 3, p.5) be the amount to be distributed. An *allotment* is a vector from $A := \{a \in \wp \mid 1a = M\}$. An *allotment rule* is a function $f(N, M, \cdot) : \wp \to A$. If N is fixed, it can be suppressed, and so can M. The allotment rule turns a vector of reported ideal points r to allotments which add to M. The x_i, r_i and f_i are functions of a continuous, n-dimensional utility u but this has not been indicated here. The utility is single-peaked, to the effect that one can speak of the (reported) ideal points only. (This is the tops-only or peaks-only property.) Some properties which identify the methods are as follows.

The allotment f is (Pareto) *efficient* if $1r \geq M$ implies $f \leq r$ and $1r \leq M$ implies $f \geq r$. So, for rationing, nobody gets more than reported, for otherwise the excess allotment might as well be given to those in need. (Without single-peaked utilities, this is only single-sidedness.)

Further, f is *strategy-proof* if $\tilde{f}(r) \leq \tilde{f}(x)$ where $\tilde{f} := (u_1 \circ f_1, \ldots, u_n \circ f_n)$. So, no agent i can increase the utility of an allotment by misreporting r_i.

Also, f is *replacement-monotonic* if $f_i(r) \geq f_i(x)$ implies that $f_j(r) \leq f_j(x)$ for all i in N and all j in $N \setminus \{i\}$. That is, if i gets more by misreporting x_i then everybody else gets less. The remainder of this list is from [18].

As a variation, f is *resource monotonic* if $M' \geq M$ implies $f(M', r) \geq f(M, r)$ for all r in $\wp_+$ so if there is more, nobody gets less.

Finally, f is *consistent* if for all i in N,

$$(f(N, M, r))|_V = f(V, M - f_i(N, M, r), r|_V)$$

where $V := N \setminus \{i\}$. So, distributing to i first and repeating the procedure f for the rest amounts (for agents other than i) to applying f to the whole population.

4.1 Fixed Path Mechanism

If N has to be expressed then the N-path g is written as $g(N, \cdot)$. A *full path* is a g which is "consistent" in the sense that for $V \subset N$ the ranges $g(\cdot, W)$ obey $g(N, W)|_V = g(V, W)$ where the restriction to V holds for the first coordinate. In the following, the dependency of f on g has been made explicit by adding it as a variable.

To repeat Definition 2, p.5: For rationing, $z_i(\lambda) = \min\{r_i, g_i(\lambda)\}$ and $y_i = z_i(\lambda)$ as well as $\sum_{i=1}^{n} \min\{r_i, g_i(\lambda)\} = M$. Of course, the excess supply of goods, that is, their underdemand, $1r \leq M$, is not considered a problem. Neither does $M = 0$ pose a mathematical problem.

Definition 5 *The allotment f is a* fixed-path allotment *if g is a full path and $y = f(M, r, g)$ obeys $y = z(\lambda)$ where*[5] *λ in Z is the solution of $h(\lambda) = M$. The problem consists of the condition $1r > M$. The mechanism is an* allocation *or rationing of goods (there is* excess demand*) if $M > 0$ and $r \geq 0$ as well as $g \geq 0$. If $M < 0$ and $r \leq 0$ as well as $g \leq 0$ then there is an* excess burden, *that is, it is an* assignment *of 'bads' with a minus sign.*

[5]There seems to be no reason why λ should be integer for the discrete problem.

Only for invertible h the notation $y = f(M, r, g)$ is warranted.

Example 5 Consider for example $r = (2, 2, 3)$ and $M = 3$. First distribute $\frac{3}{2}$ evenly amongst $\{1, 2\}$ and 3 and then give priority to agent 1. This yields $y = (\frac{3}{2}, 0, \frac{3}{2})$. The path is not consistent [18, p.57]. ■

The following states that the problem for excess burden can be represented as the problem for excess demand by putting a minus sign before r and M. Define $Z_i(\lambda) := \max\{-r_i, -g_i(\lambda)\}$ and $Y := Z(\lambda)$.

Lemma 6 (Excess burden) *Suppose $r \leq 0$ and $M < 0$ as well as $g < 0$. Then the excess burden problem reads $1Z(\lambda) = -M$, that is:*

$$\sum_{i \in N} \max\{-r_i, -g_i(\lambda)\} = -M$$

Let λ solve $h(\lambda) = M$. Then λ solves the excess burden problem and $Y = -f(-|M|, -|r|, -|g|)$. Also, $-1r < -M$, that is, the absolute values of the burden is more than the absolute value of the total ideal capacity.

Proof As is easy to see, $\min\{u, v\} = -\max\{-u, -v\}$ for any u and v. So $Y_i = Z_i(\lambda) = -y_i = -f(M, r, g)$. ■

See Example 9, p.16 below. The solution of the allotment problem $h(\lambda) = M$ is $\lambda = \overline{h}(M)$, which can computed using the lower or upper index. For $g_i(\lambda) = w_i\lambda$ the fixed-path allotment has been coined *weighted gains* [19].

Example 6 Weighted gains using the upper index b, based on Example 4, p.10. Suppose $r = (1, 4, 2, 3)$ and $w = (1, 2, 1, 1)$ so $r' = (1, 2, 2, 3)$ and $L = (5, 9, 9, 10)$. If $M = 8$ then $b = 2$ and $\lambda = \frac{7}{4}$ so $y = (1, \frac{14}{4}, \frac{7}{4}, \frac{7}{4})$. ■

Theorem 1 (Fixed-path mechanism) *The allocation in Definition 5, p.12 is efficient, strategy-proof, consistent, and resource monotonous if and only if it is a fixed path mechanism: see [18, p.56] and a straightforward proof in [9]. The fixed path mechanism is also coalitionally strategy-proof [18, p.59] but not necessarily for single peaked preferences over multiple goods, though the multidimensional uniform rule actually is strategy-proof [17].*

The following is a basic result where M is involved.

Lemma 7 *If g is invertible, then $y'_n = \lambda$, that is, the last agent's requirement is capped.*

Proof Suppose, to derive a contradiction, that $r'_n < \lambda$. Then $r'_i < \lambda$ for all i in N, that is, $r_i < g_i(\lambda)$. So $1r = M$, a contradiction, and the statement easily follows. ■

As was illustrated above, the solution using the upper or lower index can not easily be made to work for integer domains. As a preparation for a workaround, a solution in the form of an iteration is exposed in the following. This also paves the way for replacing the single λ with multiple λ_i in the role of $g_i(\lambda)$.

An iteration is derived as follows. Start with stage 1, as indicated by superscripts between brackets. First, consider $\lambda^{(1)}$ so that $r'_1 > \lambda^{(1)}$. Then $r'_i > \lambda^{(1)}$ for all i in N because of the way the r_i are sorted. So, $\sum_{i=1}^{n} g_i(\lambda^{(1)}) = M$. Therefore, $\lambda^{(1)} = (g_1 + \cdots + g_n)^{\leftarrow}(M)$ and $y_1 = g_1(\lambda^{(1)})$. As remarked in Lemma 5, p.11, $y_1 = \cdots = y_n$. Now consider the alternative, $r'_1 \leq \lambda^{(1)}$ so $y_1 = r_1$. The process needs to be repeated to solve $y_2 + \cdots + y_n = M - y_1$ (consistency.) As before, if $r'_2 > \lambda^{(2)}$ then $\lambda^{(2)} = (g_2 + \cdots + g_n)^{\leftarrow}(M - y_1)$ and $y_2 = g_2(\lambda^{(2)})$ as well as $y_2 = \cdots = y_n$. Otherwise, $r'_2 \leq \lambda^{(2)}$ so $y_2 = r_2$. And so on until $\lambda^{(n)} = g_n^{\leftarrow}(M - y_1 - \cdots - y_{n-1})$. Finally, let $\lambda := \lambda^{(n)}$. The proof of Lemma 8, p.15 will show that replacing $g(\lambda^{(i)})$ with λ in the definitions $y_i = \min\{r_i, g_i(\lambda^{(i)})$ preserves y_i. This replacement is postponed until the end of the loop, so the procedure is called terminal adjustment.

In the following definition, the procedure is rather arbitrarily equipped with a floor function for the discrete problem, that is, for indivisible goods. Warning: the procedure including the floor function generally ceases to be a fixed-path allocation. (It has been inserted to maintain a single definition.) In the further analysis of terminal adjustment, the floor function will be ignored; its treatment is postponed to the section on sequential allotment.

Definition 6 (Terminal adjustment) *Let y be computed from M and r as follows. For stage $k = 1, \ldots, n$ let*

$$\lambda^{(k)} := (g_k + \cdots + g_n)^{\leftarrow}(M - y_1 - \cdots - y_{k-1})$$

and $y_k := \min\Big\{r_k, \lfloor g_k(\lambda^{(k)}) \rfloor\Big\}$. For the real-valued problem, the floor function is omitted. Finally, $\lambda := \lambda^{(n)}$ though y need not be recomputed (and should not for the discrete problem.)

For the discrete problem, $g_n(\lambda^{(n)}) = M - y_1 - \cdots - y_{n-1}$ so the final floor function has no effect and M is distributed completely. Once $g_k(\lambda^{(k)}) \leq r_k$ the loop can stop, set $\lambda := \lambda^{(k)}$ and let $y_i := \min\{r_i, g_i(\lambda)\}$ (possibly with a floor function) for $i = k+1, \ldots, n$. The first k for this to happen is $k = b_L(M)$ so $y_i = r_i$ for $i = 1, \ldots, b_L(M)-1$ and $y_i = g_k(\lambda)$ for $i = b_L(M), \ldots, n$. This, however, requires computation of $B_L(M)$. This short-cut is based on the following lemma, of which the proof shows why a "diagonal element" $g_k(\lambda^{(k)}$ is selected.

Lemma 8 *One has* $\lambda^{(k)} \leq \lambda^{(k+1)}$ *for* $k = 1, \ldots, n-1$. *Once* $g_k(\lambda^{(k)}) \leq r_k$ *for some* k *in* N *then* $\lambda^{(\ell)} = \lambda^{(k)}$ *for all* $\ell = k, \ldots, n$.

Proof Let $M^{(k)} := M - y_1 - \cdots - y_{k-1}$ and $\phi_k := g_{k+1} + \cdots + g_n$. Then $M^{(k)} = (g_k + \phi_k)(\lambda^{(k)})$ and $M^{(k)} - y_k = \phi_k(\lambda^{(k+1)})$. Eliminating $M^{(k)}$ yields $\phi_k(\lambda^{(k+1)}) - \phi_k(\lambda^{(k)}) = g_k(\lambda^{(k)}) - y_k \geq 0$ where the $\geq$ follows from $y_k = \min\{r_k, g_k(\lambda^{(k)})\}$. As ϕ_k is invertible and monotonic, $\lambda^{(k+1)} \geq \lambda^{(k)}$. The $\geq$ is an equality if $g_k(\lambda^{(k)}) \leq r_k$. In that case, $\lambda^{(k+1)} = \lambda^{(k)} \leq r'_k \leq r'_{k+1}$ and the argument can be repeated. ∎

The proof of the following proposition provides a first formal proof of correctness for the uniform rule. For, the proof in [22] does not indicate that the $\lambda^{(k)}$ increase with k.

Proposition 3 *For divisible goods and burdens, Definition 6, p.14 gives* y *as the solution of the fixed-path allotment problem,* $y = f(M, r, g)$.

Proof By induction on n. For $n = 1$ one has $\lambda^{(1)} = g_1^{\leftarrow}(M)$ so $y_1 = \min\{r_1, M\} = M$. Now assume the statement to be true for $n-1$ where $n \geq 2$. Let $\hat{r}_1 := r_2, \ldots, \hat{r}_{n-1} := r_n$ and let $\hat{M} := M - y_1$. By the induction hypothesis, there is a $\hat{\lambda}$ for which $\sum_{i=1}^{n-1} \hat{y}_i = \hat{M}$ where $\hat{y}_i := \min\{\hat{r}_i, g_i(\hat{\lambda})\}$. Consistency: $\hat{y}_{k-1} = y_k$ for $k = 2, \ldots, n$. By virtue of Lemma 7, p.14, $\hat{\lambda} = \hat{y}_{n-1} = y_n = \lambda$. So $y_k = \min\{r_k, g_k(\lambda)\}$ for $k = 2, \ldots, n$. The proposition is proved if $y_1 = \min\{r_1, g_1(\lambda)\}$. This is done using Lemma 8, p.15. If $r_1 > g_1(\lambda^{(1)})$ then $\lambda^{(1)} = \cdots = \lambda^{(n)} = \lambda$. So indeed $y_1 = g_1(\lambda)$. If $r_1 \leq g_1(\lambda^{(1)})$ then $r_1 \leq g_1(\lambda^{(2)}) \leq \cdots \leq g_1(\lambda^{(n)}) \leq g_1(\lambda)$. Again indeed $y_1 = r_1$. ∎

Example 7 For weighted gains, let there be weights $w_i > 0$ and let $g_i(\lambda) := w_i\lambda$ for i in N. The requirements are ordered by $r_1/w_1 \le \cdots \le r_n/w_n$. The solution is

$$\begin{array}{ll} \lambda^{(k)} & := (M - y_1 - \cdots - y_{k-1})/(w_k + \cdots + w_n) \\ y_k & := \min\left\{r_k, \lfloor w_k\lambda^{(k)}\rfloor\right\} \end{array}$$

for $k = 1, \ldots, n$ and for real numbers, without the floor function.

For a concrete illustration, let $r = (1, 2, 3)$ and $w = (2, 3, 4)$. So, r is in the right order. Take $M = 5$. For divisible goods:

$$\begin{array}{lll} \lambda^{(1)} = \frac{5}{9} & g_1(\lambda^{(1)}) = 1\frac{1}{9} & y_1 = 1 \\ \lambda^{(2)} = \frac{4}{7} & g_2(\lambda^{(2)}) = 1\frac{5}{7} & y_2 = 1\frac{5}{7} \\ \lambda^{(3)} = \frac{4}{7} & g_3(\lambda^{(3)}) = 2\frac{2}{7} & y_3 = 2\frac{2}{7} \end{array}$$

So, agents 2 and 3 get a ration. For indivisible goods:

$$\begin{array}{lll} \lambda^{(1)} = \frac{5}{9} & \lfloor g_1(\lambda^{(1)})\rfloor = \lfloor 1\frac{1}{9}\rfloor = 1 & y_1 = 1 \\ \lambda^{(2)} = \frac{4}{7} & \lfloor g_2(\lambda^{(2)})\rfloor = \lfloor 1\frac{5}{7}\rfloor = 1 & y_2 = 1 \\ \lambda^{(3)} = \frac{3}{4} & \lfloor g_3(\lambda^{(3)})\rfloor = 5 - 1 - 1 = 3 & y_3 = 3 \end{array}$$

So agent 2 is rationed but agent 3 is not. ■

Repeating the procedure for Example 6, p.13 yields $y = (1, \frac{7}{2}, \frac{7}{4}, \frac{7}{4})$ as is easily checked.

Example 8 For the *uniform rule* let $w_1 = \cdots = w_n = 1$ so $\phi_k = 1/(n-k+1)$. For $r = (1, 2, 3)$ and $M = 5$ one has

$$\begin{array}{ll} \lambda^{(1)} = \lfloor \frac{5}{3} \rfloor = 1 & y_1 = 1 \\ \lambda^{(2)} = \lfloor \frac{4}{2} \rfloor = 2 & y_2 = 2 \\ \lambda^{(3)} = 5 - 1 - 2 = 2 & y_3 = 2 \end{array}$$

This discrete case is only an approximation of the uniform rule. ■

As promised, here is an illustration of solving the excess burden problem using Lemma 6, p.13.

Example 9 Similar to Example 4, p.10 consider the problem of solving ν from

$$\max\{1, 2\nu\} + \max\{2, 3\nu\} + \max\{3, 4\nu\} = 6\frac{1}{2}$$

which is an allotment problem because 1+2+3 $< 6\frac{1}{2}$. Define $r = (-1, -2, -3)$ and $w = (-2, -3, -4)$ so $r' = (\frac{1}{2}, \frac{2}{3}, \frac{3}{4})$ which means that the coordinates of r are listed in the right order. Let $M := -6\frac{1}{2}$. So

$$\begin{array}{lll} \lambda^{(1)} = -\frac{13}{18} & g_1(\lambda^{(1)}) = -1\frac{4}{9} & y_1 = -1\frac{4}{9} \\ \lambda^{(2)} = -\frac{13}{18} & g_2(\lambda^{(2)}) = -2\frac{1}{6} & y_2 = -2\frac{1}{6} \\ \lambda^{(3)} = -\frac{26}{9} & g_3(\lambda^{(3)}) = -2\frac{7}{9} & y_3 = -3 \end{array}$$

where the repetition of $\lambda^{(2)}$ announces a cap. If $6\frac{1}{2}$ is replaced with 9 then there is no cap and $y = (-2, -3, -4)$ for $\lambda = -1$. ■

4.2 Sequential Allotment

The following procedure is from [4]. The word 'sequential' refers to the sequential adjustment of reference points (or endowments) and not to its iterative nature, which it shares with one of the solutions of the fixed path allotment. For *stage* $k = 0, \ldots, n$ let there be a vector of *reference points* $q^{(k)}$ in A. (So $1q^{(k)} = M$.) The *initial reference points* are

$$q^{(0)} = \begin{cases} q_{\mathrm{H}} & \text{if } 1r \geq M \\ q_{\mathrm{L}} & \text{if } 1r < M \end{cases}$$

for some q_{H} and q_{L} in A. (The r now are positive numbers.) For *stage* $k = 0, 1, \ldots, n$ (where $k = 0$ for a preparatory stage) let $\mathcal{Q}^{(k)}$ be the collection of subsets of N having $n - k$ elements. As usual, write $2^N = \bigcup_{k=0,\ldots,n} \mathcal{Q}^{(k)}$. Let $u \leq' v$ for real numbers u and v denote $u \leq v$ if $1r > M$ and $u \geq v$ if $1r < M$. Let $u <' v$ mean $u \leq' v$ and $u \neq v$. If $1r = M$ then the choice between these definitions is arbitrary. Sequential allotment is characterized as follows.

Definition 7 (Sequential allotment) *For stage k in N the k-th* adjustment *is a function*[6]

$$\Gamma^{(k)} : \mathcal{Q}^{(k-1)} \times A \times \wp \rightarrow \mathcal{Q}^{(k)} \times A \times \wp$$

which maps $(Q^{(k-1)}, q^{(k-1)}, r)$ to $(Q^{(k)}, q^{(k)}, r)$. Each set $Q^{(0)}, Q^{(1)}, \ldots, Q^{(n)}$ is a called a list *of agents for whom processing needs to be finished still,*

[6] A computer program will adjust a single array q whereas here, a version of q is retained at every stage.

where $Q^{(0)} = N$ *and* $Q^{(n)} = \emptyset$. *Let* $\Gamma^n := \Gamma^{(n)} \circ \Gamma^{(n-1)} \circ \cdots \circ \Gamma^{(1)}$. *The map* Γ^n *is a* sequential adjustment *if for all* i *in* N, *the following holds;*[7] *the expression '*$Q^{(k)}$*' has been omitted from '*$\Gamma^{(k)}$*'.*

1. $\exists k$: *If* $r_i \leq' q_i^{(k-1)}$ *then* $q_i^{(k)} = r_i$ *to ensure efficiency.*

2. $\forall k$: *If* $r_i >' q_i^{(k-1)}$ *then* $q_i^{(k)} \geq' q_i^{(k-1)}$ *to ensure efficiency.*

3. $\forall k$: *If* $\tilde{r}_i \geq' r_i >' q_i^{(k-1)}$ *then* $\Gamma(q^{(k-1)}, r) = \Gamma(q^{(k-1)}, \tilde{r})$ *to ensure that the mechanism is strategy-proof, where* $\tilde{r}_i$ *is an alternative to* r_i *and* $\tilde{r}_j = r_j$ *for all* $j \neq i$.

4. *If* $\tilde{r}_i \geq' r_i$ *then* $(\Gamma^n(q^{(0)}, \tilde{r}))_j \leq' (\Gamma^n(q^{(0)}, r))_j$ *for all* $j \neq i$ *to ensure replacement monotonicity.*

The allotment f *is* sequential[8] *if* $f(M, r) = \Gamma^n(N, q^{(0)}, r)$.

Note that r is only copied and serves as a parameter: $\Gamma^{(k)}(q, r)$ should read $\Gamma_r^{(k)}(q)$. If r is constant, write $\Gamma^{(k)}$ instead of $\Gamma_r^{(k)}$. So $\Gamma^{(k)} : A \to A$ if the list is not mentioned.

Theorem 2 (Sequential allotment) *An allotment rule is strategy-proof, efficient, and replacement-monotonous if and only if it is sequential; they are also coalitionally strategy-proof [4, p.16]. All sequential allotment rules are obviously-strategy-proof. (Consult [2] for divisible and [3] for indivisible goods.)*

Roughly speaking, a mechanism is obviously-strategy-proof if end-users can restrict attention to extreme cases when determining their strategies and correctly conclude that misreporting does not increase their utility.

A specific case is as follows. (It is not described, only ilustrated in [4].) For any Q in $2^N \setminus \emptyset$ and q in A let $p := (Q, q, r)$ be a parameter; constant elements of the parameter will not be mentioned. For j in N let $\phi_j(p)$ be the *share factors* or *apportionment factors* if it meets two conditions:[9]

[7] The quantifiers for i and k have been inserted.

[8] The expression $(q_i, u) = \ldots$ should read $(q, u) = \ldots$ at the top of [4, p.14].

[9] A stage k should only be mentioned to simplify implementation because 1. it can be derived as $k = n - |Q|$ and 2. one can not predict at which stage ϕ_j should share in a specific manner.

1. For all parameters p one has $\sum_{j\in Q}\phi_j(p)=1$.

2. Consider r and $\tilde{r}$ in $\wp$ so that there is an i in N for which $\tilde{r}_i \geq r_i > q_i$ and $\tilde{r}_j = r_j$ for all $j \neq i$. Then $\phi_j(Q,q,r)=\phi_j(Q,q,\tilde{r})$.

Often, a share is apportionned to agents whose requirement can be fulfilled already; to avoid this, restrict sharing to agents in need by imposing

$$\sum_{j\in Q: r_j>q_j}\phi_j(p)=1$$

instead. To distribute the available amount over the initial reference points in a way that is consistent with later such distributions, one may define $\phi_j(N\cup\{0\})$ where $N\cup\{0\}$ is a dummy so as to let $q_j^{(0)} := \phi_j(N\cup\{0\})M$.

Definition 8 (Sequential apportionment) *Suppose* $1r > M > 0$. *For stage* k *in* N *define the parameter* $p^{(k-1)} := (Q^{(k-1)}, q^{(k-1)}, r)$ *and let the* k*-th* sequential apportioning $q^{(k)} = \Gamma^{(k)}(p^{(k-1)})$ *be defined as follows. Locally define*

$$B^{(k)} := \{j \in Q^{(k-1)} \mid q_j^{(k-1)} > r_j\}$$

as the set of agents from whom to derive any surplus $q_j^{(k-1)} - r_j$. *If* $B^{(k)}=\emptyset$ *then output* $p^{(k-1)}$ *and let* $\ell := k$ *be the* number of stages, *otherwise continue as follows. Define the local variable* $i_k := \min B^{(k)}$ *(an arbitrary choice) denoting the agent whose requirement is to be fulfilled. Let the new list be* $Q^{(k)} := Q^{(k-1)} \setminus \{i_k\}$. *For* j *in* N *let*

$$q_j^{(k)} := \begin{cases} r_{i_k} & \text{if } j = i_k \\ q_j^{(k-1)} + (q_{i_k}^{(k-1)} - r_{i_k})\phi_j(Q^{(k)}, q_j^{(k-1)}, r) & \text{if } j \text{ in } Q^{(k)} \\ q_j^{(k-1)} & \text{otherwise} \end{cases}$$

and output $\Gamma^{(k)} := (Q^{(k)}, q^{(k)}, r)$ *as the new* $p^{(k)}$ *which can be input of* $\Gamma^{(k+1)}$. *Once the final stage* ℓ *has been reached,* $f(M,r) = q^{(\ell)}$ *is the* sequential apportionment.

The share factors can be conceived as priorities for sharing, but it is more customary to reserve the term ‘priority’ for a ranking which is applied before fixed-path or sequential allotment, though it may be defined as part of those allotments. Notice that $Q^{(n)}$ would be empty so $\phi_j(Q^{(n)})$ is not used. Stage n is even never reached because if it were, then for all j in N there would be a k so that $q_j^{(k-1)} > r_j$. Because $q^{(k)}$ increases with k one would find $q > r$. But $M = 1q > 1r > M$, which is a contradiction.

The list is not a queue because clients can be served little by little, and simultaneously, while on the list. To match the mathematical definition, ℓ can be raised to n after making some copies of $q^{(\ell)}$.

Lemma 9 (Sequential apportionment) *The sequential apportionment is a sequential allotment.*

Proof The domain and range A are respected: $\sum_{i \in N} q_i^{(k)} = M$ for all k in N, as is easily seen. Obviously, the algorithm has conditions 1 and 2 in Definition 7, p.17. Conditions 3 is also met: would $\tilde{r}_{i_k} \geq r_{i_k} > q_{i_k}^{(k-1)}$ as in that condition, then the second condition on ϕ guararantees that $\phi_j(Q^{(k)}, q_j^{(k-1)}, r)$ stays the same if r is replaced with $\tilde{r}$ and so does $q_{i_k}^{(k-1)} - r_{i_k}$ because i_k is in the role of j in condition 3, that is, $r_{i_k} = \tilde{r}_{i_k}$. Therefore, $q^{(k)}$ is unaffected and so is Γ, for it nowhere else depends on r. For property 4, there are two cases. 1. If $r_{i_k} \leq \tilde{r}_{i_k} \leq q_{i_k}^{(k-1)}$ then the surplus shared with the list is smaller, so all get less or the same. 2. If $q_{i_k}^{(k-1)} \leq \tilde{r}_{i_k}$ then the surplus even drops to zero. ■ formalize proof

Lemma 10 *For all j in N there exists k in N so that $y_j = \min\{r_j, q_j^{(k)}\}$. In particular, $k = \ell$ is an example: $y_j = \min\{r_j, q_j^{(\ell)}\}$.*

Proof If there is a stage k at which $j = i_k$ then $r_j < q_j^{(k-1)}$ and $r_j = q_j^{(k)} = q_j^{(k+1)} = \cdots = q_j^{(\ell)} = y_j$. If there is no such stage, then $r_j \geq q^{(k)}$ for all $k = 1, \ldots, \ell$ and as $q^{(k)}$ is non-decreasing, $r_j \geq q^{(\ell)} = y_j$. ■

Because of the minimum, the surplus $q_{i_k}^{(k-1)} - r_{i_k}$ for $i_k = \min B^{(k)}$ is best couched as a slightly modified variable $d_i^{(k-1)} := \max\{0, q_i^{(k-1)} - r_i\}$ for i and k in N. For, $q_i^{(k-1)} - d_i^{(k-1)} = \min\{r_i, q_i^{(k-1)}\}$. (This is because $u - \min\{u, v\} = \max\{0, u - v\}$ for any real numbers u and v.) The variable $d^{(k-1)} := d_{i_k}^{(k-1)}$ is used in the following example.

Example 10 Let $\phi_j(Q) := 1/|Q|$ for a uniform allocation, that is, independent of j. Superfluous parameters are henceforth omitted. Consider $r := (2, 1, 10)$ and $q^{(0)} := (4, 4, 4)$. So $M = 12$. (As it were, $q_j^{(0)} = q_j^{(-1)} + \phi d^{(-1)}$ for $q_j^{(-1)} = 0$ and $d^{(-1)} = M$.) For $k = 1$ one has $B^{(1)} = \{1, 2\}$ so $i_1 = 1$ while for the uniform rule, it would have been 2 because of the ordering on the basis of the coordinates of r. The surplus is $d^{(0)} = 4-2 = 2$ and the new list is $Q^{(1)} = \{2, 3\}$. As $\phi(Q^{(1)}) = \frac{1}{2}$ one finds $q^{(1)} = (2, 4+\frac{1}{2}2, 4+\frac{1}{2}2) = (2, 5, 5)$. Indeed, $1q^{(1)} = 12$. For $k = 2$ results $i_2 = 2$ and $d^{(1)} = q_2^{(1)} - r_2 = 5 - 1 = 4$. The new list is $Q^{(2)} = \{3\}$. From $\phi(Q^{(2)}) = 1$ follows $q^{(2)} = (2, 1, 5+4)$. (Notice that, just as with the uniform rule, there is a raise $q_2^{(1)} = q_1^{(0)} + 1$ while the next step recognizes that $1 = r_2 < q_2^{(1)} = 5$, which was apparent from $1 = r_2 < q_2^{(0)} = 4$ already. At the next stage, the surplus thereof is added to $q_3^{(2)}$. Would the surplus $d^{(0)} = 2$ have been added to $q_3^{(2)}$ rightaway, the number of manipulations would be the same and generaly, approximately the same. Notice also that the order by which elements are taken from $B^{(1)}$ is immaterial.) For $k = 3$ the set $B^{(2)}$ is empty so the output is $y = q^{(2)} = (2, 1, 9)$ which is the same as for the uniform rule. ■

The following example of a "uniform" (that is, "weightless") apportionment is from [4, p.15].

Example 11 Let $q^{(0)} := (\frac{1}{6}, \frac{1}{3}, \frac{1}{6}, \frac{1}{3})$ be initial endowments. Define

$$\phi_j(Q, q, r) := \begin{cases} 1/|Q \cap \{1, 2\}| & \text{if } j \in \{1, 2\} \text{ and } r_{1,2} \not\leq q_{1,2} \\ 1/|Q \cap \{3, 4\}| & \text{if } j \in \{3, 4\} \text{ and } r_{1,2} \leq q_{1,2} \\ 0 & \text{otherwise} \end{cases}$$

for j in Q. Note that $r_{1,2} \leq q_{1,2}$ means that $r_1 \leq q_1$ and $r_2 \leq q_2$ so agents 1 and 2 can both be satisfied. So, share evenly among agents 1 and 2 if any of them is in need, otherwise among the others, irrespective of whether they are in need. To check that $1\phi(Q, q, r) = 1$ is easy by distinguishing $r_{1,2} \leq q_{1,2}$ and its negation. To check condition 2 on ϕ_j: let j in $\{1, 2\}$ be so that $r_j > q_j$. (This is impossible for $j = 3$ and $j = 4$.) Then ϕ remains as is if r_j is increased to $\tilde{r}_j$.

As a first case, let $r_1 := \frac{1}{8}$ so $r_1 < q_1^{(0)}$ and agent 1 can be satisfied. Consider stage $k = 1$. So $B^{(1)} = \{1, \ldots\}$ where the dots can only be filled after a definition of r_2 and so on. In any case, $i_1 = 1$. (In the sequel, $i_k = k$ unless indicated otherwise.) As agent 1 is satisfied, $y_1 = q_1^{(1)} = r_1 = \frac{1}{8}$. The new list is $Q^{(1)} = \{2, 3, 4\}$. The way the surplus $d^{(0)} = \frac{1}{6} - \frac{1}{8} = \frac{1}{24}$ is shared depends on whether agent 2 is in need. For r_2 consider the following two cases.

1. Let $r_2 := 1$ so $r_2 > q_2^{(0)}$ and agent 2 is in need. Consequently,

$$\phi_j(Q^{(1)}, q^{(0)}, r) = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{otherwise} \end{cases}$$

   and $q_2^{(1)} = \frac{1}{3} + d^{(0)} = \frac{3}{8}$ so $q^{(1)} = (\frac{1}{8}, \frac{3}{8}, \frac{1}{6}, \frac{1}{3})$. Now for stage $k = 2$. As $r_2 > q_2^{(1)} = \frac{3}{8}$ agent 2 is not in $B^{(2)}$. Suppose $r_3 = r_4 = 1$. So $B^{(2)} = \emptyset$ and the iteration halts with outcome $y = q^{(1)}$.

2. Let $r_2 := \frac{1}{12}$ so $r_2 < q_2^{(0)} = \frac{1}{3}$. As agents 1 and 2 can both be satisfied,

$$\phi_j(Q^{(1)}, q^{(0)}, r) = \begin{cases} \frac{1}{2} & \text{if } j \in \{3, 4\} \\ 0 & \text{otherwise} \end{cases}$$

   and this yields $q^{(1)} = (r_1, \frac{1}{3}, \frac{1}{6} + \frac{1}{2}d^{(0)}, \frac{1}{3} + \frac{1}{2}d^{(0)})$. Remember that $d^{(0)} = \frac{1}{24}$ so $q^{(1)} = (\frac{1}{8}, \frac{1}{3}, \frac{3}{16}, \frac{17}{48})$. Note that nothing is added for agent 2, whose requirement could have been fulfilled already at stage 1. Now for stage $k = 2$. As agent 2 can be satisfied, $B^{(2)} = \{2, \ldots\}$ where the dots depend on r_3 and r_4. So $i_2 = 2$ and $y_2 = q_2^{(2)} = r_2$. Share $d^{(1)} = q_2^{(1)} - r_2 = \frac{1}{4}$ among the new list $Q^{(2)} = \{3, 4\}$ using the same factors $\phi_j(Q^{(2)}, q^{(1)}, r) = \frac{1}{2}$ for j in $Q^{(2)}$. So $q^{(2)} = (r_1, r_2, \frac{3}{16} + \frac{1}{2}d^{(1)}, \frac{17}{48} + \frac{1}{2}d^{(1)})$ which is $q^{(2)} = (\frac{1}{8}, \frac{1}{12}, \frac{5}{16}, \frac{23}{48})$. Consider $k = 3$. Let $r_3 := 1$ so 3 is not in $B^{(3)}$. For r_4 consider the following cases.

   (a) Let $r_4 := 1$ so $B^{(3)} = \emptyset$ and the iteration stops, outputting $y = q^{(2)}$.

   (b) Let $r_4 := \frac{1}{3} + \frac{1}{24} = \frac{3}{8}$. (If agent 4 were selected at stage 1 then r_4 would have been capped by $q_4^{(0)} = \frac{1}{3}$.) So $\frac{18}{48} = r_4 < q_4^{(2)} = \frac{23}{48}$ yielding $B^{(3)} = \{4\}$ and agent 4 is satisfied with $y_4 = r_4$. The surplus $d^{(2)} = q_4^{(2)} - r_4 = \frac{5}{48}$ is shared with $Q^{(3)} = \{3\}$ using the factor $\phi_3(Q^{(3)}, q^{(2)}, r) = 1$ to yield $q_3^{(3)} = q_3^{(2)} + d^{(2)} = \frac{5}{16} + \frac{5}{48} = \frac{5}{12}$. Consider $k = 4$. As $1 = r_3 > q_3^{(3)}$ the iteration stops with $y_3 = q_3^{(3)}$. So $y = (r_1, r_2, 1 - r_1 - r_2 - r_4, r_4)$.

As a second case, let $r_i > q_i^{(0)}$ for i in $\{1, 2, 3\}$ and $r_4 := \frac{1}{6}$ so $r_4 < q_4^{(0)} = \frac{1}{3}$. Let $k = 1$ so $B^{(1)} = \{4\}$ from which $i_1 = 4$ and $y_4 = r_4$. Further, $Q^{(1)} = \{1, 2, 3\}$

and $d^{(0)} = q_4^{(0)} - r_4 = \frac{1}{6}$. As agents 1 and 2 are in need and have priority, so $q^{(1)} = (\frac{1}{6} + \frac{1}{2}d^{(0)}, \frac{1}{3} + \frac{1}{2}d^{(0)}, \frac{1}{6}, r_4)$, that is, $q^{(1)} = (\frac{1}{4}, \frac{5}{12}, \frac{1}{6}, \frac{1}{6})$. For r_1 and r_2 consider the following cases.

1. Suppose $r_1 = r_2 = 1$. So $r_1 > q_1^{(1)}$ and $r_2 > q_2^{(1)}$. The iteration stops with $y = q^{(1)}$.

2. Suppose $r_1 = \frac{1}{5}$ and $r_2 = \frac{3}{8}$ which still obeys $r_2 > q_2^{(0)} = \frac{1}{3}$. So $r_1 < q_1^{(1)} = \frac{1}{4}$ and $r_2 < q_2^{(1)} = \frac{5}{12}$. (Initially, though, $r_1 > q_1^{(0)} = \frac{1}{6}$ and $r_2 > q_2^{(0)} = \frac{1}{3}$, as assumed at the outset.) Consider stage $k = 2$. Still, $r_3 > q_3^{(1)} = \frac{1}{6}$ without further specification yet. So $B^{(2)} = \{1, 2\}$ and $i_2 = 1$ as well as $y_1 = q_1^{(2)} = r_1$ and $d^{(1)} = q_1^{(1)} - r_1 = \frac{1}{20}$. So $Q^{(2)} = \{2, 3\}$ and as both agents 1 and 2 can be satisfied,

$$\phi_j(Q^{(2)}, q^{(1)}, r) = \begin{cases} 1 & \text{if } j = 3 \\ 0 & \text{if } j = 2 \end{cases}$$

and $q_3^{(2)} = \frac{1}{6} + \frac{1}{20} = \frac{13}{60}$. So $q^{(2)} = (r_1, \frac{5}{12}, \frac{13}{60}, r_4)$, that is, $q^{(2)} = (\frac{1}{5}, \frac{5}{12}, \frac{13}{60}, \frac{1}{6})$. Now suppose $r_3 = 1$. At $k = 3$, only surplus $d^{(2)} = q_2^{(2)} - r_2 = \frac{1}{24}$ can be got from $B^{(3)} = \{2\}$. The new list is $Q^{(3)} = \{3\}$ and the corresponding factor is 1. So $q_3^{(3)} = \frac{13}{60} + \frac{1}{24} = \frac{31}{120}$ and $q^{(3)} = (r_1, r_2, \frac{31}{120}, r_4)$, that is, $q^{(3)} = (\frac{1}{5}, \frac{3}{8}, \frac{31}{120}, \frac{1}{6})$. This is the output at $k = 4$, where the iteration stops and essentially outputs $y = q^{(3)} = (r_1, r_2, 1 - r_1 - r_2 - r_4, r_4)$.

3. Suppose $r_1 > q_1^{(1)} = \frac{1}{4}$ (still unspecified) and, as in the previous item, $r_2 := \frac{3}{8} < q_2^{(1)} = \frac{5}{12}$. Consider $k = 2$. Only from $B^{(2)} = \{2\}$ can a surplus be derived: $d^{(1)} = \frac{5}{12} - \frac{3}{8} = \frac{1}{24}$. So $y_2 = r_2$ and the new list is $Q^{(2)} = \{1, 3\}$. Agent 1 is in need, so the factor would be 1 for agent 1 and 0 for agent 3, were it not that a variation is introduced at this point: after sharing the surplus from agent 4 to the high priority agents 1 and 2, now share the new surplus with those who are still in need irrespective of the priority. So, the formula for ϕ_j would have to use the whole history $q^{(1)}, \ldots, q^{(k-1)}$ to determine whether a surplus was shared already with agents in need. (This seems to be ethically inconsistent.) So, divide $d^{(1)} = \frac{1}{24}$ equally amongst agents 1 and 3. This yields $q^{(2)} = (\frac{1}{4} + \frac{1}{2}\frac{1}{24}, r_2, \frac{1}{6} + \frac{1}{2}\frac{1}{24}, r_4)$, that is, $q^{(2)} = (\frac{13}{48}, \frac{3}{8}, \frac{3}{16}, \frac{1}{6})$. For stage $k = 3$ consider the following possibilities.

(a) Suppose either $r_1 = \frac{25}{96}$ or $r_3 = \frac{17}{96}$. So $r_1 < q_1^{(2)} = \frac{13}{48} = \frac{26}{96}$ or $r_3 < q_3^{(2)} = \frac{3}{16} = \frac{18}{96}$. (Both is impossible, for otherwise, all requirements would be fulfilled.) This is compatible with $r_1 > q_1^{(1)} = \frac{1}{4} = \frac{24}{96}$ and $r_3 > q_3^{(1)} = \frac{1}{6} = \frac{16}{96}$ as established above. So $y_1 = r_1$ or $y_3 = r_3$, respectively, and the excess is given to the other.

(b) Suppose $r_1 = 1 \geq q_1^{(3)}$ and $r_3 = 1 \geq q_3^{(3)}$ then $y = q^{(2)}$.

Many other variations are possible. ∎

This ends the treatment of sequential allotment.

4.3 Synthesis

To finish, the fixed-path mechanism is cast in the form of sequential allotment and the floor function is considered for the indivisible goods. check indices

A variable $q_i^{(k)}$ is sought for which $1q^{(k)} = M$ for all k in N. The expression for $\lambda^{(k)}$ in Definition 6, p.14 reads $\sum_{i=1}^{k-1} y_i + G_k(\lambda^{(k)}) = M$. (See Definition 3, p.6 for the auxiliary variables.) This means that initially, $1g(\lambda^{(1)}) = M$. Suppose agent 1 can be satisfied, so $y_1 = r_1$. For $k = 2$, the surplus $M - r_1$ is divided according to $r_1 + G_2(\lambda^{(2)}) = M$. Thanks to the ordering of r, the general identity is $R_b + G_b(\lambda^{(b)}) = M$ for $b = b_L(M)$. This suggests the following choice for $q_i^{(k)}$.

Definition 9 (Sequential adjustment along a fixed path) *For* sequential adjustment along a fixed path*, define the reference points, for ordered r, by*

$$q_j^{(k)} := \begin{cases} r_j & \text{if } j = 1, \ldots, k \\ g_j(G^{\leftarrow}_{k+1}(M - R_{k+1})) & \text{if } j = k+1, \ldots, n \end{cases}$$

for $k = 0, \ldots, \ell$ where $\ell := \max\{m \in N \mid R_{m+1} < M\}$, after which $y := q_j^{(m)}$. If the coordinates of r are not necessarily ordered, then define $\Gamma_j^{(k)}$ by replacing the expression for $q_j^{(k)}$ in Definition 8, p.19 with the initialization $q_j^{(0)} := g_j(M)$ and with

$$q_j^{(k)} := \begin{cases} r_{i_k} & \text{if } j = i_k \\ g_j\left(G^{\leftarrow}_{k+1}\left(-r_{i_k} + \sum_{i \in Q^{(k-1)}} q_i^{(k-1)}\right)\right) & \text{if } j \text{ in } Q^{(k)} \\ q_j^{(k-1)} & \text{otherwise} \end{cases}$$

for $k = 1, \ldots, n-1$ where j in N. For the discrete case, continu for $k = \ell+1, \ldots, n-1$ by replacing r_{i_k} with $\rho_{i_k} := \lfloor q^{(k)} \rfloor$.

The first equation is not an iteration because the outcome does not depend on the outcome of a previous stage.

Proposition 4 *Sequential adjustment along a fixed path is a sequential adjustment (Definition 8, p.19) and outputs the same solution as terminal adjustment, Definition 6, p.14.*

Proof The first definition follows from $\sum_{i=1}^{b-1} r_i + G_b(\lambda^{(b)}) = M$ where $b = b_L(M)$. This yields the second definition after the observation that for each k, the 'otherwise' case is just a copy of r_{i_κ} where $\kappa < k$. Formalize proof

Condition 1 is met as before, also for the discrete case. Condition 2 follows for real numbers because $g_j \circ G^{\leftarrow}_{k+1}$ is monotonous. For integer numbers, at stage $k = \ell + 1$ the original $r_{i_k} > q_{i_k}$ is replaced with $\rho_{i_k} = \lfloor q_{i_k} \rfloor$ so $-\rho_{i_k} > -r_{i_k}$ in the argument of $g_j \circ G^{\leftarrow}_{k+1}$. Condition 3 is fulfilled because the expression $q_j^{(k)} := g_j(G_{k+1}(\ldots))$ for j in $Q^{(k)}$ does not depend on r_j. Condition 4 is satisfied because this expression also shows that, if r_{i_k} is raised, where k the particular stage for which i_k is set, then every other agent will not get more. In the discrete case, taking the floor function is immune for any raise of r_{i_k} for $k = \ell + 1, \ldots, n$. ■

The consequences will be stated in the conclusion.

Example 12 As in Example 6, p.13 (weighted gains) suppose $r = (1, 4, 2, 3)$ and $w = (1, 2, 1, 1)$. However, assume $M = 9\frac{1}{2}$. The terminal adjustment of Definition 6, p.14 for divisible goods yields:

$$\begin{array}{lll} \lambda^{(1)} = \frac{5}{9} & g_1(\lambda^{(1)}) = \frac{19}{10} & y_1 = r_1 = 1 \\ \lambda^{(2)} = \frac{17}{8} & g_2(\lambda^{(2)}) = 4\frac{1}{4} & y_2 = r_2 = 4 \\ \lambda^{(3)} = 2\frac{1}{4} & g_3(\lambda^{(3)}) = 2\frac{1}{4} & y_3 = r_3 = 2 \\ \lambda^{(4)} = 2\frac{1}{2} & g_4(\lambda^{(4)}) = 2\frac{1}{2} & y_3 = 2\frac{1}{2} \end{array}$$

See also Example 7, p.16.

Now for Definition 9, p.24. For weighted gains, $g_j(G^{\leftarrow}_k(x)) = w_j x/(w_k + \cdots + w_n)$ for any x. So

$$\begin{array}{ll} q^{(0)} & = (1\frac{9}{10}, 3\frac{4}{5}, 1\frac{9}{10}, 1\frac{9}{10}) \\ q^{(1)} & = (r_1, 4\frac{1}{4}, 2\frac{1}{8}, 2\frac{1}{8}) \\ q^{(2)} & = (r_1, r_2, 2\frac{1}{4}, 2\frac{1}{4}) \\ q^{(3)} & = (r_1, r_2, r_3, 2\frac{1}{2}) \end{array}$$

and $y = q^{(3)}$ agrees with the outcome of the terminal adjustment.

For unordered r keep the order because it will be clear that it does not matter. Initially, $Q^{(0)} = N$ and $q^{(0)}$ as above. Consider $k = 1$ so $B^{(1)} = \{1\}$ and $Q^{(1)} = \{2, 3, 4\}$ because only $r_1 < q_1^{(0)}$. Notice that $\sum_{i \in Q^{(0)}} q_i^{(0)} = M$. So $q^{(1)}$ as before. For $k = 2$ compare $q_{2,3,4}^{(1)} = (4\frac{1}{4}, 2\frac{1}{8}, 2\frac{1}{8})$ to $r_{2,3,4} = (4, 2, 3)$ and conclude that $B^{(2)} = \{2, 3\}$. So $i_2 = 2$ although $i_2 = 3$ would also have been possible in case of a different order of r. Therefore, $\sum_{i \in Q^{(1)}} q_i^{(1)} = 8\frac{1}{2}$ which is just an alternative computation of $M - r_1$, from which r_2 is substracted. Further, $Q^{(2)} = \{3, 4\}$ so $q^{(2)}$ as indicated. Finally, for $k = 3$, compare $q_{3,4}^{(2)} = (2\frac{1}{4}, 2\frac{1}{4})$ to $r_{3,4} = (2, 3)$ and conclude $B^{(2)} = \{3\}$. From $\sum_{i \in Q^{(2)}} q_i^{(2)} = 4\frac{1}{2}$ substract r_3 and finish with $y = q^{(3)}$ as before.

For the discrete case, repeat Example 7, p.16, where $r = (1, 2, 3)$ and $w = (2, 3, 4)$ or $M = 5$. Initially, $q^{(0)} = (1\frac{1}{9}, 1\frac{2}{3}, 2\frac{1}{3})$. For $k = 1$ process $i_1 = 1$ so $Q^{(1)} = \{2, 3\}$ and $q^{(1)} = (1, 1\frac{5}{7}, 2\frac{2}{7})$. At this stage, the iteration would stop for real-valued numbers so $\ell = 1$. In this case, continue for $k = 2$ by letting $\rho_2 := \lfloor 1\frac{5}{7} \rfloor = 1$. So $y = q^{(2)} = (r_1, \rho_2, 3)$ as before. The step $k = 3$ can be skipped because it just turns $\rho_3 = q_3^{(2)} = 3$ into $q_3^{(3)} = 3$. ■

5 Conclusion and Further Research

In conclusion, fixed-path allotment has been implemented as a "terminal adjustment" (Definition 6, p.14) which in turn has been reworked to a case of sequential allotment. So, fixed path allotment is replacement monotonous. The terminal adjustment has been equipped with a floor function in the discrete case. In that case, it no longer is a fixed-path allotment but it still is a sequential allotment, so (Pareto) efficient (i.e. one-sided) and strategy-proof, while also replacement monotonous, but not necessarily consistent or resource monotonous anymore, as for the fixed-path allotment.

Suggestions for further research are: adapting fixed-path allotment to handle the case that agents already possess the desired goods [15]; extending it to vector-valued supply M and requirements r_i, that is, for multiple types of goods [1, 17]; or a mixture of divisible and indivisible [10]; generalizing sequential allotment as a case of a multicriteria choice mechanism, that is, as a generalized median voter scheme with a tie-breaking rule [16]; investigating the relation between these allotments and a generalisation of the Kettele algorithm which also handles a lexigraphic ordering [6]; research into lattices, as suggested by the appendix.

A Residuation Theory

First, here are some notions for a partially ordered set M. For a total ordering (like in the main document) the maximum is the same as the greatest but the distinction has been retained to bring out the structure of the ordering. For x in M the *principal ideal* is $(x] := \{m \in M \mid m \leq x\}$ and $[x) := \{m \in M \mid m \geq x\}$ is the *principal filter* [13]. As in [11] for T a subset of M the set of *upper bounds* of T is $U(T) := \{m \in M \mid \forall t \in T : m \geq t\}$ and the set of its *lower bounds* is $L(T) := \{m \in M \mid \forall t \in T : m \leq t\}$. So $U(T) = \bigcap_{t \in T}[t)$ and so on. As in [7] the *greatest* of T is the (unique) element $G(T)$ so that $\{G(T)\} = \{u \in T \mid \forall t \in T : u \geq t\}$ and the *smallest* element of T is the (unique) element $S(T)$ so that $\{S(T)\} = \{u \in T \mid \forall t \in T : u \leq t\}$. These definitions do not depend on M, that is, $\{G(M)\} = U(M)$ and so forth. The *infimum* of T is $\inf T := G(L(T))$ and the *supremum* of T is $\sup T := S(U(T))$. So $U(T) = [\inf T)$ etcetera. A function f is said to preserve the supremum of a set V if $f(\sup V) = \sup f(V)$ where $f(V) := \{f(v) \mid v \in V\}$. The *maximum* of T is $\max T := \{t \in T \mid \forall u \in T : u \geq t \text{ implies } u = t\}$ and $\min T := \{t \in T \mid \forall u \in T : u \leq t \text{ implies } u = t\}$.

The following is from chapter 3 of [12] but much can be found in literature about Galois connections, lattice theory, or universal algebra. Let P and Q be partially ordered sets.

A map $f : P \to Q$ is *monotone* if $p \leq \pi$ implies $f(p) \leq f(\pi)$ for all p and π in P, where '$\leq$' is interpreted in P and Q separately. The map f is *residuated* if there exists a map $g : Q \to P$ so that

$$f(p) \leq q \text{ if and only if } p \leq g(q)$$

for all p in P and all q in Q. The maps f and g are said to form a *residuated pair.*[10]

A characterization is as follows. The maps f and g form a residuated pair if, and only if, both are monotone (that is, preserving, or reflecting,[11] the order) and $p \leq g(f(p))$ for all p in P as well as $q \geq f(g(q))$ for all q in Q.

[10] In [13][p.15] this condition reads $f(p) \geq q$ if and only if $p \geq g(q)$ for the same order relation as [12]. Note: for a Galois connection, the order on Q is interpreted as the reverse order.

[11] The word 'reflect' is not preferred because it can be misinterpreted as 'reverse'.

It turns out that any g of a residuated pair (f, g) is uniquely determined by f. Therefore, $g = \overline{f}$ is said to be the *residual* on f. Its form is:

$$\overline{f} = \max\{p \in P \mid f(p) \leq q\}$$

The maximum is unique because it is a supremum, see the proof of Lemma 3.3 in [12]. Conversely,[12]

$$f(p) = \min\{q \in Q \mid p \leq g(q)\}$$

if g belongs to a residuated pair (f, g).

An additional condition[13] is that the maximum is taken of a non-empty set, that is, $\forall q \in Q \; \exists p \in P : f(p) \leq q$. Abbreviate the range of f as $W := f(P)$. As is easily seen, this condition is equivalent to $S(Q)$ exists and $S(Q) \in W$. Similarly, let $V := g(Q)$. The condition for the minimum is $\forall p \in P \; \exists q \in Q : p \leq g(q)$. That is, $G(P)$ exists and $G(P) \in V$.

Some important applications of this machinery are the following. The restrictions $f|_V$ and $g|_W$ are each other's inverse and preserve order. Further, in any residuated pair (f, g) the f preserves arbitrary existing suprema and g preserves arbitrary existing infima. (Here, 'arbitrary existing supremum' means: supremum of any set, if the supremum exists.) If existence is guaranteed (that is, P and Q are complete lattices) then a function $f : P \to Q$ is residuated if, and only if, f preserves arbitrary suprema; conversely, a function $g : Q \to P$ is the residual of some function if, and only if, g preserves arbitrary infima.

References

[1] Adachi, T. (2010) *The uniform rule with several commodities: A generalization of Sprumont's characterization*, Journal of Mathematical Economics, vol.46, no.6, pp.952-964.

[2] Arribillaga, R.P.; Massó, J.; Neme, A. (2020) *On obvious strategy-proofness and single-peakedness*, Journal of Economic Theory, Volume 186, pp.1-33.

[12] In [13] f is called *residuated* of g. The on-line correction to [13] should be ignored here, for otherwise these terms would be reversed. Note that it is now recognized that the residuated is 1. unique and 2. a maximum.

[13] This condition is missing from the literature.

http://pareto.uab.es/jmasso/pdf/
Arribillaga_Masso_Neme_JET_2020.pdf

[3] Arribillaga, R.P.; Massó, J.; Neme, A. (2019) *All Sequential Allotment Rules Are Obviously Strategy-Proof*, Barcelona GSE Working Paper 1108 and UFAE-IAE Working Paper 966.19. The discrete case is in Appendix 2. See

http://pareto.uab.es/wp/2019/96619.pdf

via

https://econpapers.repec.org/paper/aubautbar/966.19.htm

The Barcelona GSE Working Paper 1108 is not available at GSE.

[4] Barberà, S.; Jackson, M. O.; Neme, A. (1997). *Strategy-Proof Allotment Rules*, Games and Economic Behavior, vol.18, pp.1-21.

[5] Chandramouli, Sh.; Sethuraman, J. (November 2017) *Groupstrategyproofness of the egalitarian mechanism for constrained rationing problems*. Mathematical Social Sciences, vol.90, pp.111-118.

[6] Charnes, A.; Zhang, D. (March 1988) *A Further Generalized Kettele Algorithm with Multiple Constraints*, Center for Cybernetic Studies, report 593. (Also in Journal of Information and Optimization Sciences, Vol.11, 1990, no.3)

[7] Cohn, P.M. (1965) *Universal Algebra*, London etc.: Harper & Row.

[8] Cole, R.; Gkatzelis, V.; Goel, G. (2013). *Mechanism Design for Fair Division: Allocating Divisible Items Without Payments*. Proceedings of the Fourteenth ACM Conference on Electronic Commerce. New York: ACM, pp.251-268.

[9] Ehlers, L. (2002) *On Fixed-Path Rationing Methods*, Journal of Economic Theory, vol.106, pp.472-477.

[10] Flores-Szwagrzak, K. (Dec. 2015) *Priority classes and weighted constrained equal awards rules for the claims problem*, Journal of Economic Theory Vol.160, pp.36-55.

[11] Fuchs, L. (1966) *Teilweise geordnete algebraïsche Strukturen*, Göttingen: Vandenhoeck & Ruprecht.

[12] Galatos, N.; Jipsen, P.; Kowalski, T.; Ono, H. (2007) *Residuated Lattices: An Algebraic Glimpse at Substructural Logics*, Amsterdam: Elsevier.

[13] Ganter, B.; Wille, R. (1996) *Formale Begriffsanalyse. Mathematische Grundlagen*, Springer.

[14] Grimmett, G.R.; Stirzaker, D.R. (1990II) *Probability and Random Processes*, Oxford Science Publications.

[15] Klaus, B.; Peters, H.; Storcken, T. (1998) *Strategy-proof division with single-peaked preferences and individual endowments*, Social Choice and Welfare vol.15, pp.297-311.

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.528.1938&rep=rep1&type=pdf

[16] Korgin, N.A. (2014) *Representing a Sequential Allotment Rule in the Form of a Strategy-Proof Mechanism of Multicriteria Active Expertise*, Automation and Remote Control, Vol.72, No.8.

https://www.researchgate.net/publication/271949499_Representing_a_sequential_allotment_rule_in_the_form_of_a_strategy-proof_mechanism_of_multicriteria_active_expertise

[17] Morimoto, S.; Serizawa, S.; Ching, S. (March 2013) *A characterization of the uniform rule with several commodities and agents*, Social Choice and Welfare, vol.40, no.3, pp.871-911. Discussion paper:

https://www.iser.osaka-u.ac.jp/library/dp/2010/DP0769.pdf

Publication:

https://www.semanticscholar.org/paper/A-Characterization-of-the-Uniform-Rule-With-Several-Morimoto-Serizawa/27ee604e1c002ebaaf86598a62ad2e7a74e7dd3b

[18] Moulin, H. (1999) *Rationing a Commodity along Fixed Paths*, Journal of Economic Theory, vol.84, pp.41-72.

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=102615

[19] Moulin, H. (May 2000) *Priority Rules and Other Asymmetric Rationing Methods*, Econometrica, Vol.68, No.3, pp.643-684.

[20] Moulin, Hervé (2017) *One-dimensional mechanism design.* Theoretical Economics, vol.12, no.2, pp.587-619.

[21] Nemhauser, G.L.; Wolsey, L.A. (1988) *Integer and Combinatorial Optimization*, Wiley.

[22] Sönmez, Tayfun (1994) *Consistency, monotonicity, and the uniform rule*, Econ. Letters, vol.46, no.3, pp.229-235.