The Algorithmic Landscape of Priority-Respecting Allocations

Matthew Eichhorn

Based on joint work with Sid Banerjee and David Kempe
Paper: https://arxiv.org/abs/2204.13019

A Motivating Example: Pandemic Response

Supply-chain constraints place limits on available resources

- Ventilators, Vaccines, Anti-viral treatments

Many considerations for who to prioritize

- Healthcare / essential workers
- Individuals with comorbidities
- Residents of high-density housing

A Motivating Example: Pandemic Response

Supply-chain constraints place limits on available resources

- Ventilators, Vaccines, Anti-viral treatments

Many considerations for who to prioritize

- Healthcare / essential workers
- Individuals with comorbidities
- Residents of high-density housing

What is a “fair” way to allocate care?

Commonly used priority schemes have issues

Formalizing the Reserve Allocation Setting

Agents : $\mathcal{A}, \quad n = |\mathcal{A}|$

- *Unit demand* for the resource
- *Indifferent* about how they are allocated

Categories : $\mathcal{C}, \quad m = |\mathcal{C}|$

Each category $c \in \mathcal{C}$ has:

Quota : $q_c \in \mathbb{N}, \quad q = \sum_{c \in \mathcal{C}} q_c$

Eligibility : $\mathcal{E}_c \subseteq \mathcal{A}$

Priorities : *Total pre-order* $\succeq_c$ over $\mathcal{E}_c$

- $\succeq_c$ separates agents into ranked *priority tiers*
- $a \succeq_c a' \implies c$ gives priority to a over a'

Visualizing an Instance

α (2)	β (1)	γ (1)
a	b	b
b	c , e	a
c	d	
d		
e		

Feasible Allocations

Goal: Select an allocation map $\varphi : \mathcal{A} \to \mathcal{C} \cup \{\varnothing\}$

Determines recipient set $\mathcal{A} \setminus \varphi^{-1}(\varnothing)$

Feasible Allocations

Goal: Select an allocation map $\varphi : \mathcal{A} \to \mathcal{C} \cup \{\varnothing\}$

Determines recipient set $\mathcal{A} \setminus \varphi^{-1}(\varnothing)$

What properties should φ have?

Feasible Allocations

Goal: Select an allocation map $\varphi : \mathcal{A} \to \mathcal{C} \cup \{\varnothing\}$

Determines recipient set $\mathcal{A} \setminus \varphi^{-1}(\varnothing)$

What properties should φ have?

Quota Respecting [QR]: Categories allocate at most their quotas

$$|\varphi^{-1}(c)| \leq q_c$$

Feasible Allocations

Goal: Select an allocation map $\varphi : \mathcal{A} \to \mathcal{C} \cup \{\varnothing\}$

Determines recipient set $\mathcal{A} \setminus \varphi^{-1}(\varnothing)$

What properties should φ have?

Quota Respecting [QR]: Categories allocate at most their quotas

$$|\varphi^{-1}(c)| \leq q_c$$

Eligibility Respecting [ER]: Categories only allocate to eligible agents

$$\varphi^{-1}(c) \subseteq \mathcal{E}_c$$

Feasible Allocations

Goal: Select an allocation map $\varphi : \mathcal{A} \to \mathcal{C} \cup \{\varnothing\}$

Determines recipient set $\mathcal{A} \setminus \varphi^{-1}(\varnothing)$

What properties should φ have?

Quota Respecting [QR]: Categories allocate at most their quotas

$$|\varphi^{-1}(c)| \leq q_c$$

Eligibility Respecting [ER]: Categories only allocate to eligible agents

$$\varphi^{-1}(c) \subseteq \mathcal{E}_c$$

Priority Respecting [PR]: A category allocates to an agent only if all higher-priority agents have been allocated

$$\varphi(a') = c \ \land \ a \succeq_c a' \implies \varphi(a) \neq \varnothing$$

Visualizing an Allocation

α (2)	β (1)	γ (1)
a	b	b
b	c , e	a
c	d	
d		
e		

Visualizing an Allocation

α (2)	β (1)	γ (1)
a	b	b
b	c , e	a
c	d	
d		
e		

Locating Good Allocations

Pareto Efficient [PE]: No alternate allocation satisfying [ER], [QR], [PR], allocates to a strict superset of agents

$$\neg\exists\, \psi \;:\; \psi^{-1}(\varnothing) \subsetneq \varphi^{-1}(\varnothing)$$

Locating Good Allocations

Pareto Efficient [PE]: No alternate allocation satisfying [ER], [QR], [PR], allocates to a strict superset of agents

$$\neg\exists\, \psi \;:\; \psi^{-1}(\varnothing) \subsetneq \varphi^{-1}(\varnothing)$$

Is there an efficient algorithm to find allocations with these properties?

Existing Approaches

Pathak et al (2021) [1]: Variant of Deferred Acceptance [2]

- Agents have arbitrary preferences over eligible categories
- Run DA with agents proposing to categories
- [QR], [ER], [PR], not necessarily [PE]

Delacrétaz (2021) [3]: Simultaneous Reserves Algorithm

- "Water-filling" down priority lists determines who gets allocated
- [QR], [ER], [PR], not necessarily [PE]

Aziz and Brandl (2021) [4]: Reverse Rejecting Algorithm

- Iteratively certifies whether a maximal allocation can be found without allocating to a particular agent
- All four properties, but requires $O(n)$ max matching problems

Toward an Efficient Algorithm

Decision variables: $\mathbf{x} = \{x_{a,c}\}_{a \in \mathcal{A}, c \in \mathcal{C}}$. $x_{a,c} = \mathbb{I}\big(\varphi(a) = c\big)$,

(P_0)

$$
\begin{aligned}
\max \quad & V(\mathbf{x}) := \sum_{a \in \mathcal{A}} \sum_{c \in \mathcal{C}} x_{a,c} & & \text{[PE]} \\
\text{s.t.} \quad & \sum_{a \in \mathcal{A}} x_{a,c} \leq q_c & \forall\, c \in \mathcal{C} & \quad \text{[QR]} \\
& \sum_{c \in \mathcal{C}} x_{a,c} \leq 1 & \forall\, a \in \mathcal{A} & \quad \text{[UD]} \\
& x_{a,c} = 0 & \forall\, a, c \;:\; a \notin \mathcal{E}_c & \quad \text{[ER]} \\
& x_{a,c} \in \{0, 1\} & \forall\, a \in \mathcal{A}, c \in \mathcal{C} &
\end{aligned}
$$

(P_0) encodes a bipartite b-matching problem

LP-relaxation is totally unimodular $\implies$ integer corner points

(P_0) Doesn't Account for Priorities

α (2)	β (1)	γ (1)
a	b	b
b	c , e	a
c	d	
d		
e		

(P_0) Doesn't Account for Priorities

α (2)	β (1)	γ (1)
a	*b*	*b*
b	*c* , *e*	*a*
c	*d*	
d		
e		

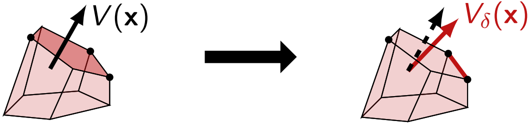
(P_0) Doesn't Account for Priorities

α (2)	β (1)	γ (1)
a	b	b
b	c , e	a
c	d	
d		
e		

To incorporate priorities, we'll modify the IP objective.

Adding Priorities

Idea: Tilt the objective so remaining optima respect priorities

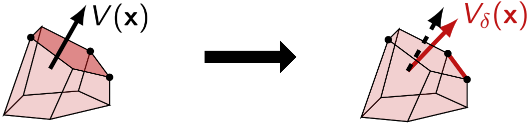


Replace $V(\mathbf{x})$ with $V_\delta(\mathbf{x}) = \sum_{a\in\mathcal{A}}\sum_{c\in\mathcal{C}}\left(1-\delta_{a,c}\right)x_{a,c}$.

Interpreting $\delta_{a,c}$ as the cost of allocating *a* through *c*, a *valid* δ satisfies:

Adding Priorities

Idea: Tilt the objective so remaining optima respect priorities



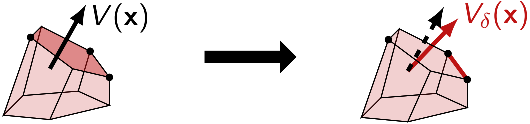
Replace $V(\mathbf{x})$ with $V_\delta(\mathbf{x}) = \sum_{a \in \mathcal{A}} \sum_{c \in \mathcal{C}} \left(1 - \delta_{a,c}\right) x_{a,c}$.

Interpreting $\delta_{a,c}$ as the cost of allocating a through c, a *valid* δ satisfies:

Small Effect: Costs don't disincentivize allocation $\sum_{a \in \mathcal{A}} \sum_{c \in \mathcal{C}} \delta_{a,c} \leq \frac{1}{2}$

Adding Priorities

Idea: Tilt the objective so remaining optima respect priorities



Replace $V(\mathbf{x})$ with $V_\delta(\mathbf{x}) = \sum_{a \in \mathcal{A}} \sum_{c \in \mathcal{C}} \left(1 - \delta_{a,c}\right) x_{a,c}$.

Interpreting $\delta_{a,c}$ as the cost of allocating a through c, a *valid* δ satisfies:

Small Effect: Costs don't disincentivize allocation $\sum_{a \in \mathcal{A}} \sum_{c \in \mathcal{C}} \delta_{a,c} \leq \frac{1}{2}$

Consistent: Prioritized agents have lower cost $a \succeq_c a' \iff \delta_{a,c} \leq \delta_{a',c}$

Our Perturbed LP

Given any δ, define the LP

(P_δ)

$$\begin{aligned}
\max \quad & V_\delta(\mathbf{x}) & \\
\text{s.t.} \quad & \sum_{a \in \mathcal{A}} x_{a,c} \leq q_c & \forall\, c \in \mathcal{C} \\
& \sum_{c \in \mathcal{C}} x_{a,c} \leq 1 & \forall\, a \in \mathcal{A} \\
& x_{a,c} = 0 & \forall\, a, c \ : \ a \notin \mathcal{E}_c \\
& x_{a,c} \geq 0 & \forall\, a \in \mathcal{A}, c \in \mathcal{C}
\end{aligned}$$

Our Perturbed LP

Given any δ, define the LP

(P_δ)

$$
\begin{aligned}
\max \quad & V_\delta(\mathbf{x}) & \\
\text{s.t.} \quad & \sum_{a \in \mathcal{A}} x_{a,c} \le q_c & \forall\, c \in \mathcal{C} \\
& \sum_{c \in \mathcal{C}} x_{a,c} \le 1 & \forall\, a \in \mathcal{A} \\
& x_{a,c} = 0 & \forall\, a, c \;:\; a \notin \mathcal{E}_c \\
& x_{a,c} \ge 0 & \forall\, a \in \mathcal{A}, c \in \mathcal{C}
\end{aligned}
$$

Theorem

Let $\mathbf{x}^*$ be a solution of (P_δ) for any valid δ. Then, $\mathbf{x}^*$ corresponds to an allocation satisfying $[ER], [QR], [PR], [PE]$.

Realizability of Good Allocations

Converse result:

Theorem (Informal)

Every recipient set determined by a fair allocation can be located by solving (P_δ) for some valid δ.

- Our perturbed matching framework is a standard setting
- The restrictions we've placed on δ are minimal

How far can we extend our techniques to handle related problems?

3 Case Studies: "Computational knife's edge" of priority-respecting allocation

1. Reasoning about a Particular Agent

Must agent *a* be allocated?

Remove *a* and all lower-ranked agents from instance.

Check if max matching size decreases

α (2)	β (1)	γ (1)
w	*x*	*x*
x	*a* , *z*	*w*
a	*y*	
y		

Can agent *a* be allocated?

A *serviceable* agent is a recipient in *some* good allocation.

Deciding whether an agent *a* is serviceable is NP-Hard.

Proof Idea: Reduction from X3C problem.

1. Reasoning about a Particular Agent

Must agent *a* be allocated?

Remove *a* and all lower-ranked agents from instance.

Check if max matching size decreases

α (2)	β (1)	γ (1)
w	*x*	*x*
x	*a* , *z*	*w*
a	*y*	
y		

Can agent *a* be allocated?

A *serviceable* agent is a recipient in *some* good allocation.

Deciding whether an agent *a* is serviceable is NP-Hard.

Proof Idea: Reduction from X3C problem.

1. Reasoning about a Particular Agent

Must agent *a* be allocated?

Remove *a* and all lower-ranked agents from instance.

Check if max matching size decreases

α (2)	β (1)	γ (1)
w	*x*	*x*
x	*z*	*w*

Can agent *a* be allocated?

A *serviceable* agent is a recipient in *some* good allocation.

Deciding whether an agent *a* is serviceable is NP-Hard.

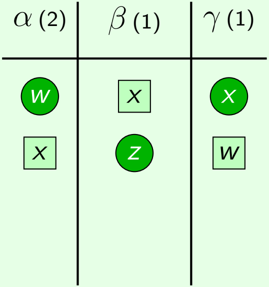
Proof Idea: Reduction from X3C problem.

1. Reasoning about a Particular Agent

Must agent *a* be allocated?

Remove *a* and all lower-ranked agents from instance.

Check if max matching size decreases



Can agent *a* be allocated?

A *serviceable* agent is a recipient in *some* good allocation.

Deciding whether an agent *a* is serviceable is NP-Hard.

Proof Idea: Reduction from X3C problem.

2. Incorporating Agent Utility

Agent a has a utility function $u_a : \mathcal{C} \to (0, 1]$ encoding preference for certain categories. ($u_a(\varnothing) = 0$.)

Utility *Pareto-Efficient* Allocation

Run our algorithm twice

First run uses arbitrary δ to determine recipient set.

Second run removes agents outside of recipient set and sets δ according to agent utilities.

Utility *Maximizing* Allocation

NP-Hard via a reduction from serviceable problem.

Proof Idea: One agent has high utility in all categories, others have low utility.

*Hardness reduction can be generalized to other optimization objectives (e.g. Nash Social Welfare)

3. Global Fairness Heuristics

For each eligible agent $a \in \mathcal{E}_c$, let $r_c(a)$ be their priority tier in c ($1 =$ highest priority, $2 =$ next priority tier, etc.)

Minimizing Maximum Allocated Rank

Run algorithm with "geometric" perturbation

$$\delta_{a,c} \propto n^{r_c(a)}$$

Proof Idea: Cost of highest ranked allocation dominates all others.

Maximizing Minimum Unallocated Rank

NP-Hard via an XC3 reduction similar to serviceable problem.

Proof Idea: Serviceable candidate has low rank in their only category. Other categories fill more ranks.

Conclusion

- Reserve Allocation is a reasonable modeling framework for assignment problems with “competing” objectives
- Can locate good allocations via a weighted matching LP
  - More efficient than existing approaches
  - Provides flexibility to many problem extensions

Open Questions:

Natural desiderata that locate a unique (fractional) allocation?

Can this allocation be computed efficiently?

- Our perturbation technique seems useful in other related problems

Thank You!

References

P. A. Pathak, T. Sönmez, M. U. Ünver, and M. B. Yenmez, "Fair allocation of vaccines, ventilators and antiviral treatments: leaving no ethical value behind in health care rationing," in *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 785–786, 2021.

D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *The American Mathematical Monthly*, vol. 69, no. 1, pp. 9–15, 1962.

D. Delacrétaz, "Processing reserves simultaneously," in *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 345–346, 2021.

H. Aziz and F. Brandl, "Efficient, fair, and incentive-compatible healthcare rationing," in *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 103–104, 2021.

Proving the Main Theorem

Theorem

Let $\mathbf{x}^*$ be a solution of (P_δ) for any valid δ. Then, $\mathbf{x}^*$ corresponds to an allocation satisfying $[ER], [QR], [PR], [PE]$.

Proof Sketch.

[ER],[QR]: Ensured by (P_δ) constraints.

[PR]: δ is Consistent.

[PE]: Small Effect of δ and integrality:

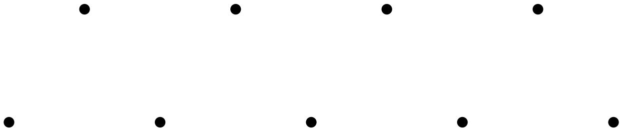
$$V(\hat{\mathbf{x}}) \geq V(\mathbf{x}^*) \geq V_\delta(\mathbf{x}^*) \geq V_\delta(\hat{\mathbf{x}}) = V(\hat{\mathbf{x}}) - \sum_{a,c} \delta_{a,c} \geq V(\hat{\mathbf{x}}) - \tfrac{1}{2}.$$

so $V(\hat{\mathbf{x}}) = V(\mathbf{x}^*)$ for a solution $\hat{\mathbf{x}}$ to (P_0). □

Exact Cover by 3-Sets (X3C)

Input: Ground set $E = \{e_1, e_2, \ldots, e_{3n}\}$.
Collection of subsets $\mathcal{S} = \{S_1, \ldots, S_m\}$, each $|S_i| = 3$.

Decide: Is there a collection of subsets $\{S_{i_1}, \ldots, S_{i_n}\}$ such that $E = \bigcup_{j=1}^{n} S_{i_j}$?



Lemma

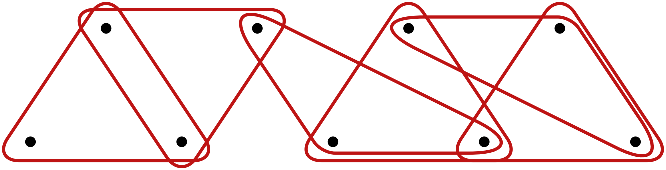
X3C *is NP-Complete.*

Exact Cover by 3-Sets (X3C)

Input: Ground set $E = \{e_1, e_2, \ldots, e_{3n}\}$.

Collection of subsets $\mathcal{S} = \{S_1, \ldots, S_m\}$, each $|S_i| = 3$.

Decide: Is there a collection of subsets $\{S_{i_1}, \ldots, S_{i_n}\}$ such that $E = \bigcup_{j=1}^{n} S_{i_j}$?



Lemma

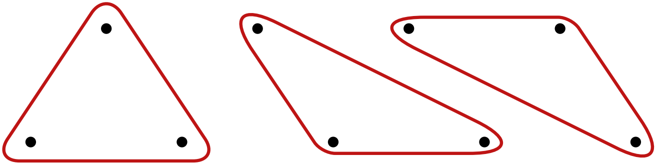
X3C *is NP-Complete.*

Exact Cover by 3-Sets (X3C)

Input: Ground set $E = \{e_1, e_2, \ldots, e_{3n}\}$.

Collection of subsets $\mathcal{S} = \{S_1, \ldots, S_m\}$, each $|S_i| = 3$.

Decide: Is there a collection of subsets $\{S_{i_1}, \ldots, S_{i_n}\}$ such that $E = \bigcup_{j=1}^{n} S_{i_j}$?



Lemma

X3C *is NP-Complete.*

The Reduction

X3C Input: $E = \{e_1, e_2, \ldots, e_{3n}\}$, $\mathcal{S} = \{S_1, \ldots, S_m\}$

$S_j = \{e_{i_{j,1}}, e_{i_{j,2}}, e_{i_{j,3}}\}$

Allocation Instance: $\mathcal{A} = E \cup \{s_1, \ldots, s_m\} \cup \left\{f_1, \ldots, f_{4(m-n)}\right\} \cup a$

set categories

α_1 (4)	$\cdots$	α_m (4)	β (1)
f_1		f_1	e_1
$\vdots$		$\vdots$	$\vdots$
$f_{4(m-n)}$		$f_{4(m-n)}$	e_{3n}
s_1		s_m	a
$e_{i_{1,1}}$		$e_{i_{m,1}}$	
$e_{i_{1,2}}$		$e_{i_{m,2}}$	
$e_{i_{1,3}}$		$e_{i_{m,3}}$	