Berlin Mathematics Research Center MATH+

Fast algorithms for Packing Linear Feasibility and its dual

Francisco Criado (TUB), David Martínez-Rubio (University of Oxford), Sebastian Pokutta (ZIB)

α-fairness and proportional fairness

Suppose we have a resource allocation problem where a single resource has to be allocated to n agents. One could try and maximize the amount of the resource allocated, but this could strongly favour some agents in the detriment of others.
A resource allocation would be **proportionally unfair** if it is possible to transfer an amount of the resource from an agent to another in such a way the proportional increase one agent experience is larger than the proportional decrease the other agent faces.
More generally, a resource allocation satisfies **proportional fairness** if it maximizes the sum of the log-utilities of each agent: $f(x) = \sum_{i=1}^{n} \log x_i$. Proportional fairness is a particular case of α-**fairness**, where the objective function is

$$f_\alpha(x) = \begin{cases} \frac{x^{1-\alpha}}{1-\alpha}, & \text{if } \alpha \geq 0, \alpha \neq 1 \\ \log(x) & \text{if } \alpha = 1, \end{cases}$$

for $\alpha = 1$. Note that the 0-fair utility corresponds to the sum of the utilities.
Proportional fairness is the only utility function satisfying several expected *fairness axioms*. [3] Here we are interested in studying this fairness criterion for the case of positive polytopes, this is a set of the form

$$\mathcal{P} = \{x \in \mathbb{R}^n_{\geq 0} : Ax \leq \mathbb{1}_m\}$$
For $A \in \mathbb{R}^{m\times n}_{\geq 0}$,

These feasible regions are what naturally appear in various resource allocation problems, particularly network flows.

Definition (Packing proportional fairness and its dual)

Let $A \subseteq \mathbb{R}^{m\times n}_{\geq 0}$ be a nonnegative matrix. We study the following two problems:

1-fair packing [4]: $\max_{x\in\mathbb{R}^n_{\geq 0}} \left\{ f(x) \stackrel{\text{def}}{=} \sum_{i=1}^{n} \log x_i : Ax \leq \mathbb{1}_m \right\}$. (1)

Dual 1-fair packing: $\min_{\lambda\in\Delta^n} \left\{ g(\lambda) \stackrel{\text{def}}{=} -\sum_{i=1}^{n} \log(A^T\lambda)_i - n\log n \right\}$. (2)

Primal problem: Exponential reparametrization and linear coupling

We imitate the reparametrization for the 0-fair packing problem in [4]. First, consider Problem 1 in exponential space, that is, replace the variable x_i with e^{x_i} to get:

$$\max_{x\in\mathbb{R}^n} \left\{ \hat{f}(x) \stackrel{\text{def}}{=} \sum_{i\in n} x_i : A\exp(x) \leq \mathbb{1}_m \right\}.$$

Now, we remove the constraints and add a barrier function to the objective funciton:

$$f_r(x) \stackrel{\text{def}}{=} -\sum_{i\in n} x_i + \frac{\beta}{1+\beta} \sum_{i=1}^{m} (A\exp(x))_i^{\frac{1+\beta}{\beta}}.$$

β is a technical parameter. If β is small, then the second summand becomes arbitrarily large when $\exp(x) \notin \mathcal{P}$. In particular, we use $\beta = \frac{\varepsilon}{6n\log(2mn^2/\varepsilon)}$.

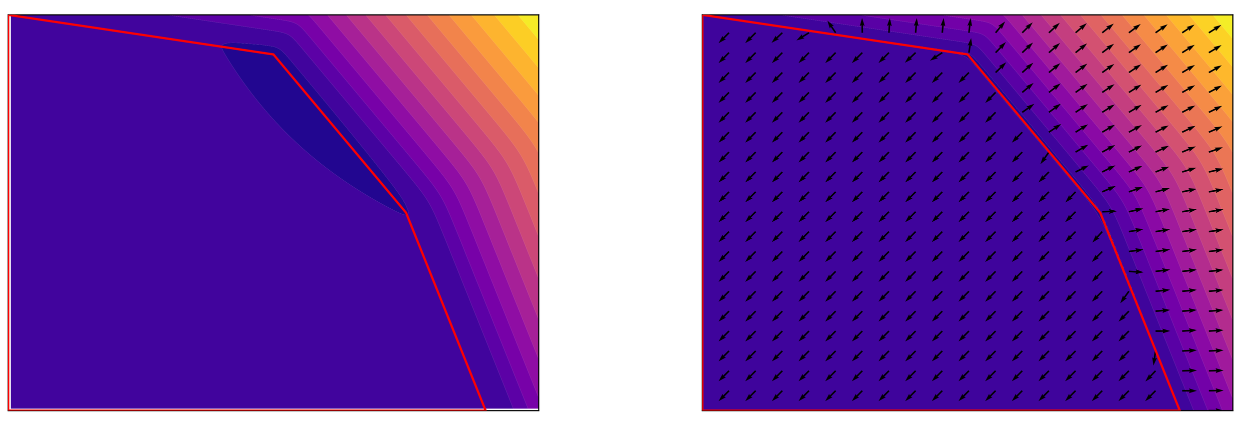


Figure 1. Regularized objective f_r (left) and its gradient (right), for a sample matrix $A \in \mathcal{M}_{3\times 2}(\mathbb{R}_{\geq 0})$. For visualization purposes we show $\log(f_r(x))$ and $\log(\|\nabla f_r(x)\|)$, represented by color, and we indicate the direction of the gradient with normalized arrows. Also, note that we show the results in the original space (before reparametrizing) but the gradient shown is computed in the reparametrized space.

We solve this second problem via *linear coupling* [1]. Linear coupling is a first order optimization technique that combines a gradient descent step (primal) with a mirror descent step. The next point to compute will be a convex combination of the primal and dual iterations. It is possible to show that the new point will either reduce the value of the primal functon greatly, or increase the best known lower bound on the problem. In either case the dual gap decreases, guaranteeing fast convergence. With this technique we can find a solution to the reparametrized problem with a low objective value, and furthermore, we can reconstruct a solution to the original problem:

Theorem (Criado,Martínez,Pokutta 2021)

Let $\varepsilon \leq n/2$ and let $\bar{x}^*$ be the optimum solution of Problem 1. Our algorithm computes a point $y^{(T)} \in B$ such that $f_r(y^{(T)}) - f_r(x_r^*) \leq \varepsilon$ in a number of iterations $T = \tilde{O}(n/\varepsilon)$. Besides, $\hat{x} \stackrel{\text{def}}{=} \exp(y^{(T)})/(1+\varepsilon/n)$ is a feasible point of Problem 1, i.e., $A\hat{x} \leq \mathbb{1}_m$, and $f(\bar{x}^*) - f(x) \leq 5\varepsilon = O(\varepsilon)$.

Our algorithm outperforms the best known algorithms for general α [4]. Furthermore its running time is independent on the *width* of A, unlike most α-fair algorihtms.

Dual problem: The centroid map and the PST oracle

Intuitively, Problem 2 is about finding the simplex minimizing volume with a fixed corner in the positive orthant that covers $\mathcal{P}$.
In this section we identify the constraint $\langle h, x\rangle \leq 1$ with the (dual) point $h \in \mathbb{R}^n_{\geq 0}$. We can do this as all the constraints we work with have strictly positive right hand side.

Definition

- $\mathcal{D} = \text{conv}\{A_i : i \in [m]\}$,
- $\mathcal{D}^+ = (\text{conv}\{A_i : i \in [m]\} + [-\infty, 0]^n) \cap \mathbb{R}^n_{\geq 0}$.

$\mathcal{D}^+$ is the set of constraints (i.e., dual points) feasible in all $\mathcal{P}$.

Consider the **centroid map** $c : \mathbb{R}^n_{\geq 0} \to \mathbb{R}^n_{\geq 0}$, $c(h) = \left(\frac{1}{nh_1}, \ldots, \frac{1}{nh_n}\right)$. It maps a constraint h with the centroid of the simplex resulting from its intersection with the positive orthant.

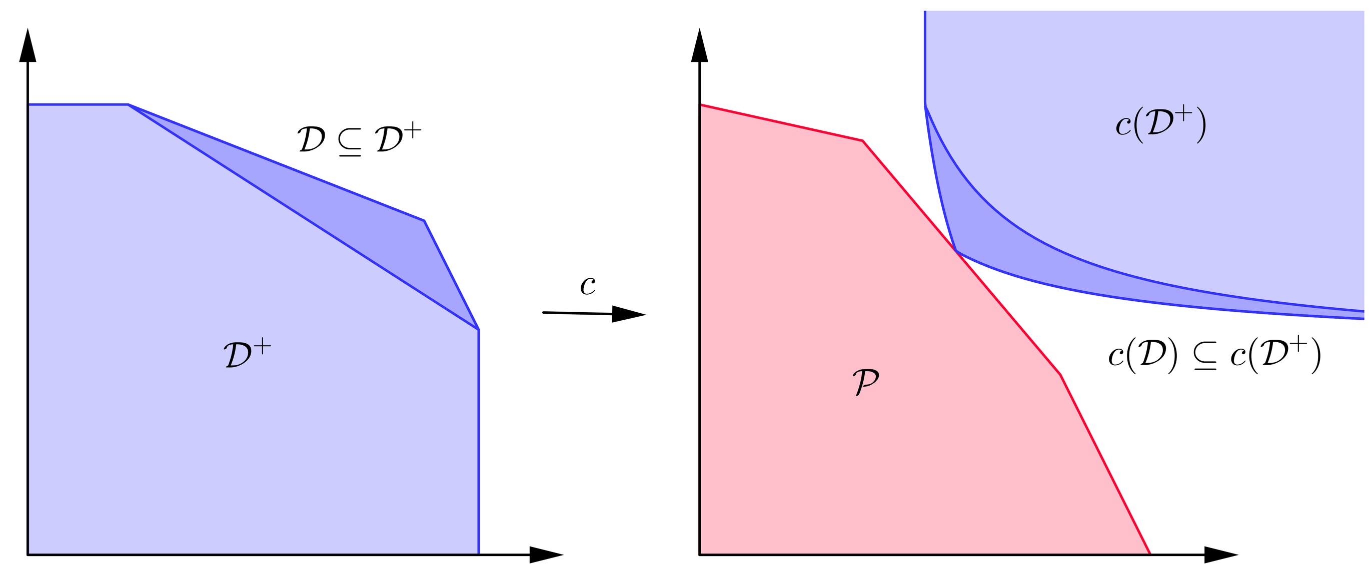


Figure 2. Left: The (extended) dual polytope $\mathcal{D}^+$ contains $\mathcal{D}$. c maps dual points (i.e., feasible constraints in $\mathcal{P}$) to primal points. Right: $\mathcal{P}$ with the image of $\mathcal{D}^+$ and $\mathcal{D}$ under c. The intersection $\mathcal{P} \cap c(\mathcal{D}^+)$ is exactly one point which is also in $c(\mathcal{D})$. Note that $c(\mathcal{D})$ is not convex but $c(\mathcal{D}^+)$ is.

The solution is the unique point in the intersection $\mathcal{P} \cap c(\mathcal{D}^+)$. This motivates the study of the following **proxy problem**:

$$\min_{p\in c(\mathcal{D}^+)} \left\{ \hat{g}(p) \stackrel{\text{def}}{=} \max_{i\in[m]} \langle A_i, p\rangle \right\}. \quad (3)$$

The minimum is known to be 1 and the solution is unique. This is a linear packing feasibility problem of the constraints $Ax \leq \mathbb{1}_m$ over the convex set $c(\mathcal{D}^+)$. We solve this via a variant of the Plotkin-Shmoys-Tardos (PST) algorithm as explained in [2]. This algorithm requires a *feasibility oracle* to guide it.
The feasibility oracle receives a convex combination of the rows of A, $s = \lambda_s^T A$, $\lambda_s \in \Delta^n$ and returns a point $x \in c(\mathcal{D}^+)$ with $\lambda_s^T Ax \leq 1$. In our case we made an **adaptive feasibility oracle** that performs better if equipped with good solutions of Problem 3.

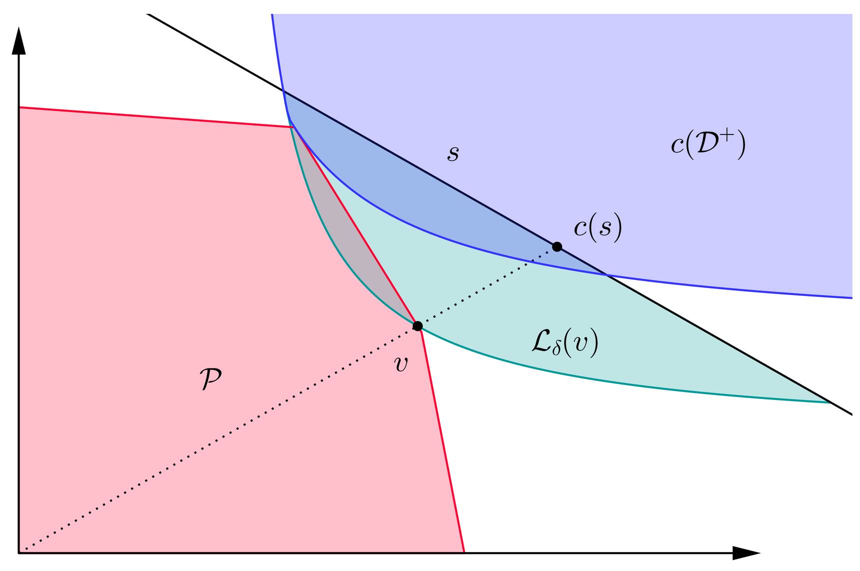


Figure 3. The **lens** given by a *good* feasible solution s. Our oracle returns points in this region satisfying any input constraint $h \in \mathcal{D}^+$ feasible over $\mathcal{P}$. Observe that the intersection $\mathcal{P} \cap c(\mathcal{D}^+)$ is contained in $\mathcal{L}_\delta(v)$. As δ becomes smaller, the lens becomes smaller around the optimum primal solution. Thus the oracle gives better approximations of it.

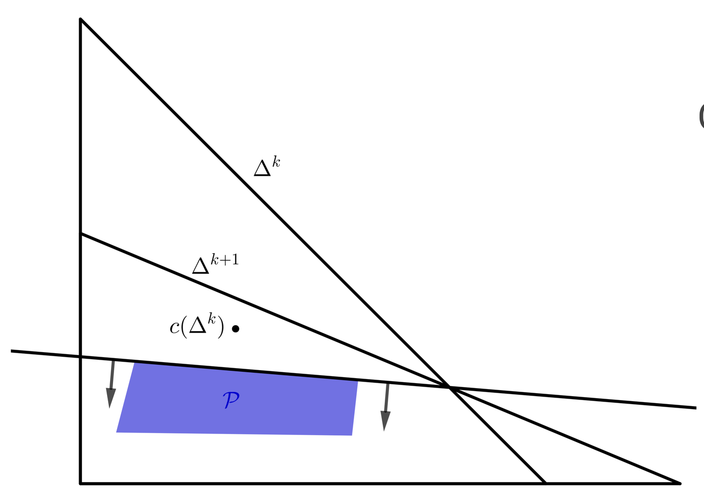
Thanks to this adaptive oracle and a restarting scheme, we obtain the following result:

Theorem (Criado, Martínez, Pokutta, 2021)

Let $\varepsilon \in (0, n(n-1)]$ be an accuracy parameter. There is an algorithm that finds a linear combination of the rows of A, $\lambda \in \Delta^m$ such that $\hat{g}(c(\lambda^T A)) \leq 1+\varepsilon$ (i.e., an ε-approximate solution of Problem 3) after $\tilde{O}(n^2/\varepsilon)$ iterations. Furthermore, this same solution is a (ε/n)-approximate solution of Problem 2.

Dual problem motivation: Yamnitsky and Levin's simplices algoritm

The following algorithm [5] is a discrete version of the Ellipsoid method for convex/linear feasibility that runs in polynomial time for some choice of initial simplex and final volume:



Input: A face description of a polyhedron $\mathcal{P} = \{x \in \mathbb{R}^n : Ax \leq b\}$ for $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$. Also a simplex $\Delta^0 = \subseteq \mathbb{R}^n$ with $\mathcal{P} \in \Delta^0$ and a lower bound on the volume of $\mathcal{P}$, V_{min}
Output: Either a point $x \in \mathcal{P}$ or **emtpy**.

0. Let $k = 0$.
1. If $vol(\Delta^k) < V_{min}$ return **empty**.
2. Compute the centroid of Δ, c^k.
3. If $c^k \in \mathcal{P}$ then return c^k.
4. Select a row A_j such that $A_j c^k > b_j$.
5. Obtain a new simplex Δ^{k+1} by combining one facet of Δ^k with A_j, b_j. The other facets remain unchanged. $vol(\Delta^{k+1}) \leq (1 - \frac{1}{n^2}) vol(\Delta^k)$. Go to step 1.

The Yamnitski-Levin algorithm does not specify which hyperplane to change in case of multiple options or multiple separating hyperplanes. If we fix one corner and we try to find the minimum simplex with that corner fixed that covers the positive constraints of $\mathcal{P}$, then we have exactly the dual 1-fair packing problem.
We do not know right now if fast algorithms for the dual 1-fair problem give a faster version of Yamnitsky-Levin, either asymptotically or practically.

References

[1] Z. Allen-Zhu and L. Orecchia.
Nearly linear-time packing and covering LP solvers - achieving width-independence and 1/epsilon-convergence.
Math. Program., 175(1-2):307–353, 2019.
URL: https://doi.org/10.1007/s10107-018-1244-x, doi:10.1007/s10107-018-1244-x.

[2] S. Arora, E. Hazan, and S. Kale.
The multiplicative weights update method: a meta-algorithm and applications.
Theory Comput., 8(1):121–164, 2012.
URL: https://doi.org/10.4086/toc.2012.v008a006, doi:10.4086/toc.2012.v008a006.

[3] D. Bertsimas, V. F. Farias, and N. Trichakis.
The price of fairness.
Oper. Res., 59(1):17–31, 2011.
URL: https://doi.org/10.1287/opre.1100.0865, doi:10.1287/opre.1100.0865.

[4] J. Diakonikolas, M. Fazel, and L. Orecchia.
Fair packing and covering on a relative scale.
SIAM J. Optim., 30(4):3284–3314, 2020.
URL: https://doi.org/10.1137/19M1288516, doi:10.1137/19M1288516.

[5] B. Yamnitsky and L. A. Levin.
An old linear programming algorithm runs in polynomial time.
In *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982*, pages 327–328. IEEE Computer Society, 1982.
URL: https://doi.org/10.1109/SFCS.1982.63, doi:10.1109/SFCS.1982.63.

TU berlin ZIB ZUSE INSTITUTE BERLIN