# A general algorithm for convex fair partitions of convex polygons

Mathilda Campillo[1], Maria D. Gonzalez-Lima[1*] and Bernardo Uribe[1]

*Correspondence:
limad@uninorte.edu.co
[1]Department of Mathematics and
Statistics, Universidad del Norte, Km.
5 via Puerto Colombia, Barranquilla,
081007, Colombia

**Abstract**

A convex fair partition of a convex polygonal region is defined as a partition on which all regions are convex and have equal area and equal perimeter. The existence of such a partition for any number of regions remains an open question. In this paper, we address this issue by developing an algorithm to find such a convex fair partition without restrictions on the number of regions. Our approach utilizes the normal flow algorithm (a generalization of Newton's method) to find a zero for the excess areas and perimeters of the convex hulls of the regions. The initial partition is generated by applying Lloyd's algorithm to a randomly selected set of points within the polygon, after appropriate scaling. We performed extensive experimentation, and our algorithm can find a convex fair partition for 100% of the tested problem. Our findings support the conjecture that a convex fair partition always exists.

**Mathematics Subject Classification:** 52A38; 68U05

**Keywords:** Convex equipartition; Fair partition; Lloyd's algorithm; Voronoi partition; Centroidal Voronoi partition

## 1 Introduction

Mass partitions of convex regions satisfying specific properties are common optimization questions that have appeared in several contexts (Roldán-Pensado and Soberón [1]). In this note we are interested in the problem of partitioning a convex polygon into convex regions of equal area and equal perimeter. This problem was firstly proposed by Nandakumar and Ramana Rao [2] where such a partition was coined *convex fair partition*. The same authors showed that the intermediate value theorem implies that a convex fair partition for two regions always exists and gave a general argument on how to show that a convex fair partition for $n = 2^k$ regions always exists. Bárány et al. [3] gave a proof of the existence of a convex fair partition for three regions using equivariant topological properties of the space of 3-fans in $\mathbb{R}^2$, and later, Blagojević and Ziegler [4] and Karasev et al. [5] gave proofs of the existence of convex fair partitions for $n = p^k$ with $p$ any prime using obstruction theory in equivariant algebraic topology.

The proof of the existence of a convex fair partition for any number of regions seems still unresolved; although the recent preprint of Akopyan et al. [6] claims to have shown

its existence by applying a recursive argument on the proofs presented in (Blagojević and Ziegler [4]) and (Karasev et al. [5]).

All the works presented above tackle the existence problem using topological arguments generalizing the well-known Borsuk–Ulam theorem [7]. But none of them focuses on the practical question of finding such a partition for a given convex polygonal region and a given number of regions. The only algorithm that has appeared in the literature has been provided by Armaselu and Daescu [8], and they show how to implement a procedure to find a convex fair partition of a convex polygonal region into $n = 2^k$ regions by iterating the original idea of Nandakumar and Ramana Rao [2].

Our work presents a novel, simple yet practical algorithm designed to find a convex fair partition of any convex polygon into any number of regions, if such a partition exists. It is based on the generalization of Newton's method denoted *normal flow algorithm* that finds zeros of under-determined systems of equations [9, 10]. The convergence theory for the proposed algorithm uses the assumption that there exists a convex starting partition of the polygon such that the areas and perimeters are sufficiently close. We use the centroidal Voronoi partitions [11, 12] (using the well-known *Lloyd's algorithm*), as well as a scaling of the polygonal region, to find this starting partition. This is key to the practicality of the method because of the large number of configurations that must be considered for a general procedure. When applying the *normal flow algorithm*, the convexity is preserved by using the area of the convex hull instead of the area function. Our extensive experiments show that convergence of the proposed algorithm is reached, no matter the choice of the number of regions or the polygon considered.

In Sect. 2 we describe our general algorithm, in Sect. 3 we provide the theoretical background that explains the different choices that appear in our algorithm, and in Sect. 4 we provide numerical evidence of efficacy of our algorithm and discuss the properties that centroidal Voronoi partitions have. We end with Sect. 5 with some conclusions and further ideas that could be addressed. We have added an appendix where we present convex fair partitions of an equilateral triangle, a square, and a pentagon into different regions.

## 2  Method

Along the paper we denote by $\mathbb{K}$ the convex polygon to be considered for a convex fair partition. The main algorithm that we are presenting in this work is the following.

---

**Algorithm 1** Convex fair partition algorithm

---

**Input:** (convex polygon $\mathbb{K}$, number of partitions=n)

**Output:** (convex fair partition)

1. Scale the height of the polygon $\mathbb{K}$ along the direction perpendicular to its  diameter to a region whose height equals the diameter of $\mathbb{K}$.
2. Choose n random points in the interior of the scaled polygon.
3. Apply Lloyd's algorithm to find a centroidal Voronoi partition in the scaled polygon.
4. Scale back the centroidal Voronoi partition to the $\mathbb{K}$ polygon.
5. Apply the normal flow algorithm to the excess convex area and perimeter function that depends on the coordinates of the interior and boundary vertices of the partition, and the areas of the convex hull.

---

**Figure 1** Steps of Algorithm 1 on the prescribed convex polygon $\mathbb{K}$ presented in (a). The region $\mathbb{K}$ is scaled in (b) to match the diameter with the height. Random Voronoi centers are chosen in (c) to start with a random convex partition. Lloyd's algorithm is performed in (d) to obtain a centroidal Voronoi partition. The polygon, together with the centroidal Voronoi partition, is scaled back in (e) to the fit the polygon $\mathbb{K}$. Finally, the normal flow algorithm is used in the interior and boundary vertices of the partition in (e) to obtain a fair partition in (f)

The key new idea underlying the algorithm is the following. Fix a convex partition of the convex polygon $\mathbb{K}$ and look at the vertices the partition describes (see Fig. 1(e). These vertices are located on the interior of the polygon $\mathbb{K}$, on the boundary of the polygon $\mathbb{K}$, or are vertices of the polygon $\mathbb{K}$. The internal and boundary vertices, together with the edges of the regions the partition describes, define a combinatorial graph. Perturbing the coordinates of these internal and boundary vertices but keeping the combinatorial way on which they are attached to one another defines a new geometrical partition of the polygon $\mathbb{K}$. The total dimension of the perturbations that can be carried out this way is the number of boundary vertices (they can only be perturbed on the boundary of the polygon where they are located) plus twice the number of internal vertices (they may move freely around their position). This total dimension is always greater than or equal to the number of equations needed to solve the convex fair partition problem; moreover, the extra dimensions are equal to the number of internal vertices. So, if the fixed partition that we start with is located close to the solution, a generalized Newton's method such as the normal flow algorithm (Ben-Israel [9], Walker and Watson [10]) can find the desired partition (see Fig. 1(f)). The method is applied to the excess areas and perimeter functions, namely the difference between the area and perimeters from their means, and to achieve convexity, we take the area of the convex hull of the region. The zeroes of these functions provide a partition that is fair and convex.

To obtain a partition that is close to the solution, and that brings as many extra dimensions as possible for the normal flow algorithm, we proceed to using Lloyd's algorithm for obtaining a centroidal Voronoi partition (Lloyd [11], Du et al. [12]). Centroidal Voronoi partitions provide balanced Voronoi partitions where the centroids of the regions agree with the Voronoi centers (see Fig. 1(d)). The areas and perimeters of centroidal Voronoi partitions do not differ by much, and therefore they provide a good starting point for

the normal flow algorithm. But when the polygonal region $\mathbb{K}$ is too thin, the centroidal Voronoi partition may include very few internal vertices, and therefore the partition may not be the best choice for a starting point of the normal flow algorithm. To overcome this difficulty, we linearly scale the region of the polygon $\mathbb{K}$ on the direction perpendicular to the diameter in such a way that the new height equals the diameter (see Fig. 1(b)). On the scaled polygon the centroidal Voronoi partition always includes several vertices on the interior (see Fig. 1(d)), and then we scale back to the polygon $\mathbb{K}$ the whole partition (see Fig. 1(e)). This partition will be the starting point for the normal flow algorithm.

## 3 Theory

We start by describing the nonlinear equations problem once we have a convex partition of polygon $\mathbb{K}$. Then we describe the procedure that we propose to generate a convex partition of $\mathbb{K}$, which is close to the fair convex one.

Throughout the paper we denote by $|.|$ the Euclidean and the Frobenius norm (for vectors and matrices, respectively), unless otherwise specified. If $\mathcal{S}$ is an index set, denote by $|\mathcal{S}|$ its cardinality.
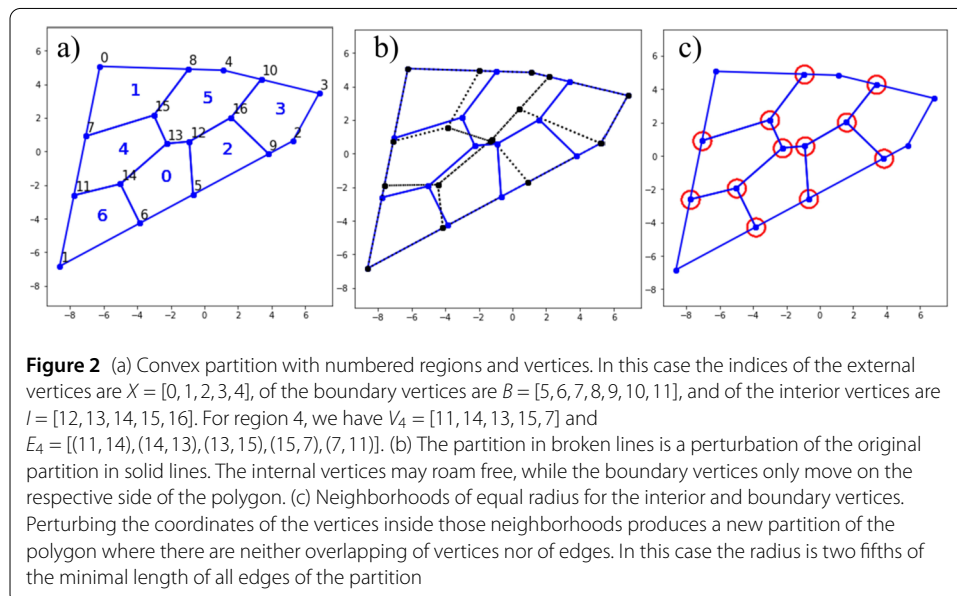
### 3.1 Excess convex area and perimeter function

Let us fix a convex polygon $\mathbb{K}$ and let us fix a convex partition of the polygon into $n$ regions. We want to move the coordinates of the interior and boundary vertices of the partition to find a fair and convex one. This will be an optimization problem that we will set up in this section.

Enumerate all vertices of the partition, starting with the vertices of the polygon $\mathbb{K}$, then the ones that lie on the boundary, and then the interior ones.

Make the following definitions and notations (as an example see Fig. 2):
- $n$: number of regions of the partition.
- $q_r = (x_r, y_r)$: coordinates of the $r$th vertex of the partition.
- $X$: list of indices of vertices of the polygon $\mathbb{K}$.
- $B$: list of indices of vertices lying on the boundary of the polygon $\mathbb{K}$.



**Figure 2** (a) Convex partition with numbered regions and vertices. In this case the indices of the external vertices are $X = [0, 1, 2, 3, 4]$, of the boundary vertices are $B = [5, 6, 7, 8, 9, 10, 11]$, and of the interior vertices are $I = [12, 13, 14, 15, 16]$. For region 4, we have $V_4 = [11, 14, 13, 15, 7]$ and $E_4 = [(11, 14), (14, 13), (13, 15), (15, 7), (7, 11)]$. (b) The partition in broken lines is a perturbation of the original partition in solid lines. The internal vertices may roam free, while the boundary vertices only move on the respective side of the polygon. (c) Neighborhoods of equal radius for the interior and boundary vertices. Perturbing the coordinates of the vertices inside those neighborhoods produces a new partition of the polygon where there are neither overlapping of vertices nor of edges. In this case the radius is two fifths of the minimal length of all edges of the partition

- $I$: list of indices of vertices lying on the interior of the polygon $\mathbb{K}$.
- $V_k$: list of indices of the vertices lying on the $k$th region, cyclicly ordered (counter clockwise).
- $E_k$: list of pairs of indices $(r, s)$ such that the oriented segment from $q_r$ to $q_s$ is an edge on the $k$-region (with counter clockwise orientation).

The area and perimeter functions of the $k$th region, in terms of the coordinates of its vertices, are defined as follows:

$$A_k := \frac{1}{2} \sum_{(r,s)\in E_k} (y_s x_r - y_r x_s), \qquad P_k := \sum_{(r,s)\in E_k} |q_s - q_r|. \tag{1}$$

Since we are interested in finding equal areas and equal perimeters, we may subtract the average of the areas and of the perimeters respectively:

$$\overline{A}_k := A_k - \frac{\text{Area of polygon}}{n}, \qquad \overline{P}_k := P_k - \frac{1}{n}\left(\sum_{j=1}^{n} P_j\right). \tag{2}$$

Bundle up these excess areas and perimeters into a functional

$$f : \mathbb{R}^{2(|I|+|B|+|X|)} \to \mathbb{R}^{2n}, \qquad f := \langle \overline{A}_1, \ldots, \overline{A}_n, \overline{P}_1, \ldots, \overline{P}_n \rangle^T, \tag{3}$$

and note that fair partitions are exactly the zeroes of this functional $f$. The domain of the partition problem for the excess areas and perimeters is determined by the free variables of the optimization problem. The internal vertices may roam free on the interior of the polygon, while the boundary vertices are restricted to move along the edge of the polygon where lie. Let us add the following notations and conventions:

- $q_r^0 = (x_r^0, y_r^0)$: fixed coordinates of the vertex $r$ of the fixed partition; $r \in I \cup B \cup X$.
- $\mathbf{q}_I^0 = (q_r^0)_{r\in I} \in (\mathbb{R}^2)^{|I|}$, $\mathbf{q}_B^0 = (q_s^0)_{s\in B} \in (\mathbb{R}^2)^{|B|}$, $\mathbf{q}_X^0 = (q_u^0)_{u\in X} \in (\mathbb{R}^2)^{|X|}$: fixed coordinates of interior, boundary, and exterior vertices respectively.
- $\mathbf{q}_I = (q_r)_{r\in I} \in (\mathbb{R}^2)^{|I|}$, $\mathbf{q}_B = (q_s)_{s\in B} \in (\mathbb{R}^2)^{|B|}$: coordinates of interior and boundary vertices respectively.
- $\mathbf{v} = (v^s)_{s\in B} \in (\mathbb{R}^2)^{|B|}$: unitary vector of the edge of the polygon where the vertex $q_s$ lies, taken in the counter clockwise direction; only for $s \in B$.
- $\mathbf{t} = (t_s)_{s\in B} \in \mathbb{R}^{|B|}$: variables that parametrize perturbations of the boundary vertices.
- $\mathbf{p} = (p_r)_{r\in I} \in (\mathbb{R}^2)^{|I|}$: variables that parametrize perturbations of the interior vertices.

Therefore we may consider the parametrization map

$$\iota : \mathbb{R}^{2|I|+|B|} \to \mathbb{R}^{2(|I|+|B|+|X|)} \tag{4}$$

$$(\mathbf{p}, \mathbf{t}) \mapsto \left(\mathbf{q}_I^0 + \mathbf{p}, \mathbf{q}_B^0 + \mathbf{tv}, \mathbf{q}_X^0\right),$$

which parametrizes perturbations of the internal vertices and boundary vertices

$$\mathbf{q}_I = \mathbf{q}_I^0 + \mathbf{p} \quad \text{and} \quad \mathbf{q}_B = \mathbf{q}_B^0 + \mathbf{tv} \tag{5}$$

respectively. Here $\mathbf{tv}$ will denote $(t_s v^s)_{s\in B} \in (\mathbb{R}^2)^{|B|}$.

**Definition 1** Composing the parametrization $\iota$ with the excess areas and perimeters functional $f$, we obtain $F := f \circ \iota$

$$F : \mathbb{R}^{2|I|+|B|} \to \mathbb{R}^{2n} \tag{6}$$

$$F(\mathbf{p}, \mathbf{t}) := f\left(\mathbf{q}_I^0 + \mathbf{p}, \mathbf{q}_B^0 + \mathbf{t}\mathbf{v}, \mathbf{q}_X^0\right).$$

The function $F$ will be called *excess area and perimeter function.*

A solution of $F = 0$ will provide the coordinates of a partition, which is fair, but we moreover need it to be convex.

To address the convexity of the iteration procedure on the normal flow algorithm, we modify the excess area and perimeter function to one that solves both issues at the same time. Let us elaborate.

Recall that the convex hull of a region consists of all points lying in the interior of line segments defined by any two points of the region. The convex hull of a polygonal region is also a polygonal region, and its vertices form a subset of the set of the original vertices.

**Definition 2** Let $V_k$, $E_k$, and $A_k$ denote the vertices, the edges, and the area of the $k$th region of the partition. Define by $V_k^c$, $E_k^c$, and $A_k^c$ the vertices, edges, and area of the convex hull of the region.

The area of the convex hull is defined as in equation (1), and its formula is

$$A_k^c := \frac{1}{2} \sum_{(r,s) \in E_k^c} (y_s x_r - y_r x_s). \tag{7}$$

Therefore we get the following straightforward result.

**Lemma 1** $V_k^c \subseteq V_k$, $A_k^c \geq A_k$ *and the equalities are reached if and only if the region is convex. Hence, the partition is convex if and only if*

$$\sum_{k=1}^{n} A_k^c = \text{Area of polygon.} \tag{8}$$

By the previous lemma we see that a solution of the equations

$$A_k^c - \frac{\text{Area of polygon}}{n} = 0 \tag{9}$$

implies on the one hand equation (8), and therefore the convexity of the partition, and on the other hand that all regions have equal area. Therefore we may modify our objective function as follows.

**Definition 3** Let

$$\overline{A}_k^c := A_k^c - \frac{\text{Area of polygon}}{n} \tag{10}$$

be the excess convex area of the $k$th region, define the functional

$$f^c : \mathbb{R}^{2(|I|+|B|+|X|)} \to \mathbb{R}^{2n}, \qquad f := \left\langle \overline{A}_1^c, \ldots, \overline{A}_n^c, \overline{P}_1, \ldots, \overline{P}_n \right\rangle^T, \tag{11}$$

and define the *excess convex area and perimeter function $F^c$* as the new *nonlinear function*

$$F^c : \mathbb{R}^{2|I|+|B|} \to \mathbb{R}^{2n}, \qquad F^c(\mathbf{p}, \mathbf{t}) := f^c \left( \mathbf{q}_I^0 + \mathbf{p}, \mathbf{q}_B^0 + \mathbf{tv}, \mathbf{q}_X^0 \right).$$

*Remarks* 1  Several remarks are in order:
- The excess functions $F$ and $F^c$ agree on convex partitions. In particular on the initial scaled centroidal Voronoi partition.
- A solution of the equation $F^c = 0$ is also a solution of $F = 0$. Hence a partition of equal areas and equal perimeters.
- A solution of the equation $F^c = 0$ defines a convex partition.
- All the bounds for the first and second derivatives of $F$ apply also to the first and second derivatives of $F^c$.

On the fifth step of our main Algorithm 1, we use the excess convex area and perimeter function $F^c$ and apply the normal flow algorithm as will be outlined in the following Sect. 3.2.

### 3.2  Normal flow algorithm

A widely used method for finding zeroes of functions $G : \mathbb{R}^{\overline{m}} \to \mathbb{R}^m$ with $\overline{m} \geq m$ is a generalization of Newton's algorithm called *normal flow algorithm*. The *normal flow algorithm* was first introduced by Ben-Israel (Ben-Israel [9]) for solving nonlinear system of equations when the derivative matrix is not square.

The algorithm is the following.

---

**Algorithm 2** Normal flow algorithm

---

*Normal flow algorithm:* Given $\overline{x}_0 \in \mathbb{R}^{\overline{m}}$, determine for $k = 0, 1, 2, \ldots$

$$\overline{x}_{k+1} = \overline{x}_k - G'(\overline{x}_k)^+ G(\overline{x}_k). \tag{12}$$

---

Here $G'(\overline{x})$ denotes the derivative matrix and $G'(\overline{x})^+$ is its pseudo-inverse. That is, for $b \in \mathbb{R}^n$, $G'(\overline{x})^+ b \in \mathbb{R}^{\overline{m}}$ is the solution of $G'(\overline{x})\overline{s} = b$ having minimal Euclidean norm.

Walker and Watson (Walker and Watson [10]) proved local convergence of the *normal flow algorithm* under the following hypotheses (see (Walker and Watson [10], Thm 2.1)):
- $G$ is differentiable and $G' \in \mathbb{R}^{m \times \overline{m}}$ is of full rank $m$ in an open convex set $\Omega$.
- There exist $\gamma \geq 0$ and $p \in (0, 1]$ such that $|G'(\overline{y}) - G'(\overline{x})| \leq \gamma |\overline{y} - \overline{x}|^p$ for all $\overline{y}, \overline{x} \in \Omega$.
- There is a constant $\mu$ for which $|G'(\overline{x})^+| \leq \mu$ for all $\overline{x} \in \Omega$.
- There is a constant $\eta$ such that $\Omega_\eta := \{\overline{x} \in \Omega : |\overline{y} - \overline{x}| < \eta \Rightarrow \overline{y} \in \Omega\}$ is a nonempty open set.

If these hypotheses are met, (Walker and Watson [10], Thm 2.1) assures that there is $\epsilon > 0$ depending on $\gamma$, $p$, $\mu$, and $\eta$ such that if $\overline{x}_0 \in \Omega_\eta$ and $|G(\overline{x}_0)| < \epsilon$, then the iterates $\{\overline{x}_k\}_{k=0,1,\ldots}$ determined by the normal flow algorithm converge to a point $\overline{x}^* \in \Omega$ such that

$G(\overline{x}^*) = 0$. Furthermore, it shows that there is a constant $\beta$ for which $|\overline{x}_{k+1} - \overline{x}^*| \leq \beta |\overline{x}_k - \overline{x}^*|^{1+p}$.

### 3.3 Properties of the excess convex area and perimeter function

Because of Remark 1, we are going to focus our analysis on the excess area and perimeter function $F : \mathbb{R}^{2|I|+|B|} \to \mathbb{R}^{2n}$ to determine how suited this function is for the hypothesis presented in the previous section.

First note that the total excess of the area and that of the perimeter are both zero. This simply says that

$$\sum_{i=1}^{n} \overline{A}_i = 0, \qquad \sum_{i=1}^{n} \overline{P}_i = 0. \tag{13}$$

Therefore the image of the excess area and perimeter function $F$ lands on the $2(n-1)$ dimensional subspace whose first $n$-coordinates add to zero, and the second $n$-coordinates add also to zero.

**Proposition 1** *The excess area and perimeter function $F$ is differentiable in the open set of points in $\mathbb{R}^{2|I|+|B|}$ where all induced vertices are different. Its derivative $F'$ is moreover of full rank $2(n-1)$ on this open set.*

*Proof* The derivatives of the area and perimeter functions are the following. With respect to vertex $q_r = (x_r, y_r)$, with $r \in V_k$ and $(s, r), (r, u) \in E_k$:

$$\frac{\partial A_k}{\partial x_r} = \frac{1}{2}(y_s - y_u) \qquad \frac{\partial A_k}{\partial y_r} = \frac{1}{2}(x_u - x_s) \tag{14}$$

$$\frac{\partial P_k}{\partial x_r} = \frac{x_r - x_s}{|q_r - q_s|} + \frac{x_r - x_u}{|q_r - q_u|} \qquad \frac{\partial P_k}{\partial y_r} = \frac{y_r - y_s}{|q_r - q_s|} + \frac{y_r - y_u}{|q_r - q_u|}. \tag{15}$$

Whenever $r \notin V_k$, the vertex $q_r$ is not a vertex of the $k$th region, and therefore the partial derivatives of $A_k$ and $P_k$ with respect to $x_r$ and $y_r$ vanish.

With respect to the variables $t_s$, parametrizing the boundary vertices with $s \in B$ and $s \in V_i \cap V_k$, we have

$$\frac{\partial A_i}{\partial t_s} = \left\langle \frac{\partial A_i}{\partial x_s}, \frac{\partial A_i}{\partial y_s} \right\rangle \cdot v^s \qquad \frac{\partial P_i}{\partial t_s} = \left\langle \frac{\partial P_i}{\partial x_s}, \frac{\partial P_i}{\partial y_s} \right\rangle \cdot v^s. \tag{16}$$

Note that $\frac{\partial A_i}{\partial t_s} + \frac{\partial A_k}{\partial t_s} = 0$.

The derivatives of the excess area and perimeter functions are the following. With respect to the internal vertex $q_r$ with $r \in V_i \cap V_j \cap V_k$, we have ($q_r$ is the common vertex of the regions $i, j$ and $k$):

$$\frac{\partial \overline{A}_k}{\partial x_r} = \frac{\partial A_k}{\partial x_r} \qquad \frac{\partial \overline{A}_k}{\partial y_r} = \frac{\partial A_k}{\partial y_r} \tag{17}$$

$$\frac{\partial \overline{P}_k}{\partial x_r} = \frac{n-1}{n}\frac{\partial P_k}{\partial x_r} - \frac{1}{n}\left(\frac{\partial P_i}{\partial x_r} + \frac{\partial P_j}{\partial x_r}\right) \qquad \frac{\partial \overline{P}_k}{\partial y_r} = \frac{n-1}{n}\frac{\partial P_k}{\partial y_r} - \frac{1}{n}\left(\frac{\partial P_i}{\partial y_r} + \frac{\partial P_j}{\partial y_r}\right). \tag{18}$$

Here we are using that the total area of the partition is constant. Therefore the derivatives of the areas and the excess areas are the same.

The derivatives for the excess areas and perimeters with respect to $t_s$ with $s \in B$ and $s \in V_i \cap V_k$ are:

$$\frac{\partial \overline{A}_i}{\partial t_s} = \frac{\partial A_i}{\partial t_s} \qquad \frac{\partial \overline{P}_i}{\partial t_s} = \frac{n-1}{n} \frac{\partial P_i}{\partial t_s} - \frac{1}{n} \frac{\partial P_k}{\partial t_s}. \tag{19}$$

All derivatives of the excess area and the excess perimeter are continuous and differentiable as long as the denominators of the expressions in (15) are not zero. This simply says that the excess areas and perimeters are $C^\infty$-functions on the open set of parameters that induce partitions without overlapping vertices.

Now let us show that $F'$ is of maximal rank. Denote by $\nabla \overline{A}_i$ the vector of partial derivatives of $\overline{A}_i$ with respect to the free variables $x_r, y_r$, and $t_s$ for $r \in I$ and $s \in B$; define similarly $\nabla \overline{P}_j$. Note that the vectors $\nabla \overline{A}_i$ and $\nabla \overline{P}_j$ are precisely the rows of the derivative matrix $F'$.

Let us assume that there is a linear relation among the rows of the derivative matrix $F'$:

$$\sum_{i=1}^{n} \alpha_i \nabla \overline{A}_i + \sum_{j=1}^{n} \beta_j \nabla \overline{P}_j = 0. \tag{20}$$

We will show that this implies that all $\alpha$s are equal and all $\beta$s are also equal. Hence we would have that the rank of the derivative matrix $F'$ is $2(n-1)$ in all the open set where the vertices do not overlap.

Equation (20) may be restricted to the coordinate $x_r$ for $r \in I$. In this case we obtain the equation

$$\alpha_i \frac{\partial A_i}{\partial x_r} + \alpha_j \frac{\partial A_j}{\partial x_r} + \alpha_k \frac{\partial A_k}{\partial x_r} + \beta_i \frac{\partial \overline{P}_i}{\partial x_r} + \beta_j \frac{\partial \overline{P}_j}{\partial x_r} + \beta_k \frac{\partial \overline{P}_k}{\partial x_r} = 0 \tag{21}$$

with $r \in V_i \cap V_j \cap V_k$. Denoting by $u, s, w$ the indices of the vertices joining $q_r$ in the counter clockwise order and assuming without loss of generality that $u \in V_j \cap V_k$, $s \in V_k \cap V_i$, and $w \in V_i \cap V_j$, we may replace equations (17), (18), and (19) in equation (21) to obtain the following equation:

$$\frac{1}{2} \big( y_u(\alpha_j - \alpha_k) + y_s(\alpha_k - \alpha_i) + y_w(\alpha_i - \alpha_j) \big) + \frac{x_r - x_u}{|q_r - q_u|}(\beta_j + \beta_k - 2\overline{\beta})$$
$$+ \frac{x_r - x_s}{|q_r - q_s|}(\beta_k + \beta_i - 2\overline{\beta}) + \frac{x_r - x_w}{|q_r - q_w|}(\beta_i + \beta_j - 2\overline{\beta}) = 0, \tag{22}$$

where $\overline{\beta} = \frac{1}{n} \sum_{l=1}^{n} \beta_l$. Whenever any of the coordinates $y_*$ or $x_*$ vary, the associated coefficient must vanish.

Since any common edge of a pair of adjacent regions sharing an internal vertex cannot be attached to an external vertex, the variables $x_*, y_*$ vary. Hence all coefficients in the equation must vanish. In particular, this implies that $\alpha_i = \alpha_j = \alpha_k$ and $\beta_i = \beta_j = \beta_k = \overline{\beta}$.

Therefore, for every internal vertex, the values of the $\alpha$s and the $\beta$s of the regions that touch this vertex are equal.

Now let us see what happens whenever we restrict equation (20) to a coordinate $t_s$ representing a vertex on the boundary. In this case we obtain equation

$$\alpha_i \frac{\partial A_i}{\partial t_s} + \alpha_j \frac{\partial A_j}{\partial t_s} + \beta_i \frac{\partial \overline{P}_i}{\partial t_s} + \beta_j \frac{\partial \overline{P}_j}{\partial t_s} = 0 \tag{23}$$

with $s \in B \cap V_i \cap V_j$. On the one hand, we get that $(\alpha_i - \alpha_j)\frac{\partial A_i}{\partial t_s} = 0$ because $\frac{\partial A_i}{\partial t_s} + \frac{\partial A_j}{\partial t_s} = 0$; on the other hand, we get that $(\beta_i - \beta_j)\frac{\partial P_i|_{(s,r)}}{\partial t_s} = 0$ where $P_i|_{(s,r)}$ denotes the perimeter of the $i$th region restricted to the edge $(r,s)$ where $r \in B \cap V_i$, in other words, the length of the side of the $i$-region lying on the boundary of the polygon with one vertex with index $s$. From both equations we also obtain that $\alpha_i = \alpha_j$ and $\beta_i = \beta_j$.

We have shown that adjacent regions in the partition have equal $\alpha$s and equal $\beta$s, and since the adjacency graph induced by the regions of the partition is connected, we conclude that all $\alpha$s are equal and all $\beta$s are equal. Therefore the rank of the derivative $F'$ is always $2(n-1)$ as claimed.                                                                  $\square$

By Proposition 1 we know that $\dim(\mathrm{Ker}(F')) = 2|I| + |B| - 2(n-1)$. Let us now see that these numbers equal the number of internal vertices.

**Proposition 2** *The dimension of the kernel of the derivative excess area and perimeter equals the number of internal vertices, i.e.,*

$$\dim\big(\mathrm{Ker}\big(F'\big)\big) = |I|. \tag{24}$$

*Proof* The partition has $|I| + |B| + |X|$ vertices, $n$ regions, and twice the number of edges equals the sum of the valences of all vertices. Since the valence of internal and boundary edges is three, and the valence of external vertices is two, by the Euler characteristic formula we obtain

$$\big(|I| + |B| + |X|\big) - \frac{1}{2}\big(3|I| + 3|B| + 2|X|\big) + n = 1. \tag{25}$$

We have therefore that the number of internal and boundary vertices equals twice the number of regions minus two:

$$|I| + |B| = 2n - 2. \tag{26}$$

Therefore we conclude

$$\dim\big(\mathrm{Ker}\big(F'\big)\big) = 2|I| + |B| - 2(n-1) = |I|. \tag{27}$$
$$\square$$

Since we have just shown that the excess area and perimeter function is differentiable, by the implicit function theorem we obtain the following corollary.

**Corollary 1** *If the solution set F = 0 is nonempty and lies inside the open set where F is differentiable, then the solution set is a differentiable manifold whose dimension equals the number of internal vertices of the original partition.*

Hence, the more internal points the original partition has, the better for the optimization problem.

Now let us bound the norm of the first and second derivatives of the excess area and perimeter function, and let us address the Lipschitz condition for the first derivative that appears in the hypothesis list.

Consider the open set

$$\Omega = \text{Interior}\big\{(\mathbf{p}, \mathbf{t}) \in \mathbb{R}^{2|I|+|B|} : \iota(\mathbf{p}, \mathbf{t}) \in \text{ConvexHull(Polygon)}\big\} \tag{28}$$

consisting of all parametrizations of configurations where the interior points roam inside the polygonal region and the boundary points roam on the side of the polygon where they lie.

Take any $\delta > 0$ and define the open set

$$\Omega(\delta) = \big\{(\mathbf{p}, \mathbf{t}) \in \Omega : |q_s - q_u| > \delta \text{ for all } s \in I \cup B \ \& \text{ for all } u \in (I \cup B \cup X)\big\} \tag{29}$$

of parametrizations of configurations where the distance between any pair of adjacent vertices in the partition is bounded below by $\delta$. Here we are not taking into account adjacent polygonal vertices that are still adjacent after the partition.

On the open set $\Omega(\delta)$ we may bound the first and second derivatives of $F$. Let us first calculate the second derivatives of the areas and the perimeters.

Let $(s, r), (r, u) \in E_k$, then

$$\frac{\partial^2 P_k}{\partial y_s \partial y_r} = -\frac{1}{|q_r - q_s|} + \frac{(y_r - y_s)^2}{|q_r - q_s|^3} \qquad \frac{\partial^2 A_k}{\partial y_r \partial x_s} = -\frac{1}{2} \tag{30}$$

$$\frac{\partial^2 P_k}{\partial x_s \partial x_r} = -\frac{1}{|q_r - q_s|} + \frac{(x_r - x_s)^2}{|q_r - q_s|^3} \qquad \frac{\partial^2 A_k}{\partial y_s \partial x_r} = \frac{1}{2} \tag{31}$$

$$\frac{\partial^2 P_k}{\partial y_r \partial x_r} = -\frac{(x_r - x_s)(y_r - y_s)}{|q_r - q_s|^3} - \frac{(x_r - x_u)(y_r - y_u)}{|q_r - q_u|^3} \tag{32}$$

$$\frac{\partial^2 P_k}{\partial y_s \partial x_r} = \frac{(x_r - x_s)(y_r - y_s)}{|q_r - q_s|^3} \tag{33}$$

$$\frac{\partial^2 P_k}{\partial x_r^2} = \frac{1}{|q_r - q_s|} + \frac{1}{|q_r - q_u|} - \frac{(x_r - x_s)^2}{|q_r - q_s|^3} - \frac{(x_r - x_u)^2}{|q_r - q_u|^3} \tag{34}$$

$$\frac{\partial^2 P_k}{\partial y_r^2} = \frac{1}{|q_r - q_s|} + \frac{1}{|q_r - q_u|} - \frac{(y_r - y_s)^2}{|q_r - q_s|^3} - \frac{(y_r - y_u)^2}{|q_r - q_u|^3}. \tag{35}$$

If we denote by $D$ the maximum width of the polygon, then $\delta$, $x_r - x_s$ and $y_r - y_s$ are all smaller than $D$, and therefore we may bound all second derivative as follows:

$$\left|\frac{\partial^2 P_k}{\partial x_r^2}\right|, \left|\frac{\partial^2 P_k}{\partial y_r^2}\right|, \left|\frac{\partial^2 P_k}{\partial x_r \partial y_r}\right|, \left|\frac{\partial^2 P_k}{\partial x_s \partial x_r}\right|, \left|\frac{\partial^2 P_k}{\partial y_s \partial x_r}\right|, \left|\frac{\partial^2 P_k}{\partial y_s \partial y_r}\right| < \frac{2}{\delta} + \frac{2D^2}{\delta^3} < \frac{4D^2}{\delta^3}. \tag{36}$$

By the calculations of the derivatives of the excess perimeter functions $\overline{P}_k$ done in equations (18), we see that the second derivatives of the excess perimeter functions are also bounded. Hence, on the open set $\Omega(\delta)$ the second derivative of $F$ is bounded. So we obtain the following result.

**Proposition 3** *For sufficiently small $\delta$, the second derivative of the excess area and perimeter function is bounded in $\Omega(\delta)$. Therefore there exists $\gamma$ such that for all points $(\bar{\mathbf{p}}, \bar{\mathbf{t}}), (\mathbf{p}, \mathbf{t}) \in \Omega(\delta)$ the following Lipschitz condition holds:*

$$\left| F'(\bar{\mathbf{p}}, \bar{\mathbf{t}}) - F'(\mathbf{p}, \mathbf{t}) \right| < \gamma \left| (\bar{\mathbf{p}}, \bar{\mathbf{t}}) - (\mathbf{p}, \mathbf{t}) \right|. \tag{37}$$

*Proof* We just need to show that $\Omega(\delta)$ is path connected for sufficiently small $\delta$. But this follows from the fact that one can construct a path in $\mathbb{R}^2$ joining two points with other two points, keeping their distance bounded from below.

By the mean value theorem and the Cauchy–Schwarz inequality, we have that

$$\left| F'(\bar{\mathbf{p}}, \bar{\mathbf{t}}) - F'(\mathbf{p}, \mathbf{t}) \right| \le \left| F''(\hat{\mathbf{p}}, \hat{\mathbf{t}}) \right| \left| (\bar{\mathbf{p}}, \bar{\mathbf{t}}) - (\mathbf{p}, \mathbf{t}) \right| \tag{38}$$

for some point $(\hat{\mathbf{p}}, \hat{\mathbf{t}}) \in \Omega(\delta)$. Then we may take the bound of $|F''|$ inside $\Omega(\delta)$ as the desired $\gamma$. □

Now let us choose an appropriate value of $\delta$:

$$\delta = \frac{1}{5} \min \left\{ \left| q_r^0 - q_s^0 \right| : (r, s) \right.$$

$$\left. \text{runs over the edges of the original partition with } (r, s) \notin X \times X \right\}. \tag{39}$$

That is, $5\delta$ is the minimum length among all edges of the original partition not including the sides of the polygon without boundary vertices.

It is then easy to see that the product of the balls of radius $\delta$ around each internal and boundary vertex is an open subset of $\Omega(\delta)$:

$$\prod_{r \in I} B_\delta^2(0) \times \prod_{s \in B} B_\delta^1(0) \subset \Omega(\delta), \tag{40}$$

and therefore the distance between adjacent points will be always bigger than $\delta$ (see Fig. 2 (c). Here $B_\delta^2(0)$ and $B_\delta^1(0)$ denote respectively the 2-dimensional and 1-dimensional balls of radius $\delta$ around zero.

Now let us bound the pseudo-inverse $(F')^+$. For this we will reduce our original open set $\Omega(\delta)$ to a subset where the pseudo-inverse $|(F')^+|$ is bounded, and it includes the origin.

Since the norm of the pseudo-inverse is not zero at the origin, and since $|F'((\mathbf{p}, \mathbf{t}))^+|$ is continuous on $\Omega(\delta)$, by local compactness around the origin inside $\Omega(\delta)$, there must exist a radius $\xi \le \delta$ such that

$$\left| F'(\mathbf{p}, \mathbf{t})^+ \right| < \mu \quad \text{for all } (\mathbf{p}, \mathbf{t}) \in \prod_{r \in I} B_\xi^2(0) \times \prod_{s \in B} B_\xi^1(0). \tag{41}$$

Then let us consider

$$\overline{\Omega} := \prod_{r \in I} B_\xi^2(0) \times \prod_{s \in B} B_\xi^1(0) \subset \Omega(\delta) \tag{42}$$

and note that on $\overline{\Omega}$ the Lipschitz condition for $F'$ holds and $|F'|$, $|F''|$, and $|(F')^+|$ are all bounded.

Defining

$$\overline{\Omega}_\eta = \left\{ (\mathbf{p}, \mathbf{t}) \in \overline{\Omega} : \left| (\bar{\mathbf{p}}, \bar{\mathbf{t}}) - (\mathbf{p}, \mathbf{t}) \right| < \eta \Rightarrow (\bar{\mathbf{p}}, \bar{\mathbf{t}}) \in \overline{\Omega} \right\} \tag{43}$$

and taking $\eta = \xi/2$, we see by the triangle inequality that

$$\prod_{r \in I} B_{\xi/2}^2(0) \times \prod_{s \in B} B_{\xi/2}^1(0) \subset \overline{\Omega}_{\xi/2}. \tag{44}$$

Therefore we can conclude the following.

**Theorem 1** *Let $5\delta$ be the minimum length of the edges of the original partition without including edges between vertices of the polygon, and let $\xi \le \delta$ be such that $|F'|$, $|F''|$, and $|(F')^+|$ are all bounded on the open set*

$$\overline{\Omega} := \prod_{r \in I} B_\xi^2(0) \times \prod_{s \in B} B_\xi^1(0) \subset \Omega(\delta). \tag{45}$$

*Then the origin is an interior point of $\overline{\Omega}_{\xi/2}$, and we have the following open set around the origin:*

$$\prod_{r \in I} B_{\xi/2}^2(0) \times \prod_{s \in B} B_{\xi/2}^1(0) \subset \overline{\Omega}_{\xi/2}. \tag{46}$$

*Moreover, since all the hypotheses presented in Sect. 3.2 of (Walker and Watson [10], Thm 2.1) are met, there exists $\epsilon > 0$ such that whenever the initial partition parametrized by $(\mathbf{p}, \mathbf{t})$ belongs to $\overline{\Omega}_{\xi/2}$ and $|F(\mathbf{p}, \mathbf{t})| < \epsilon$, the iterates of the normal flow algorithm converge to a partition $(\mathbf{p}^*, \mathbf{t}^*)$ of equal areas and equal perimeters.*

The following remark is in order. Considering the first derivative matrix in vector form, the infinity norm for the matrix $F''$ (and therefore the Lipschitz constant $\gamma$) is bounded by $\frac{8nD^2}{\delta^3}$. The larger this value is, the smaller (at least theoretically) is the region of convergence for the *normal flow algorithm*. Hence, the minimum distance between contiguous vertices may affect the performance of the algorithm, specially when $n$ is large.

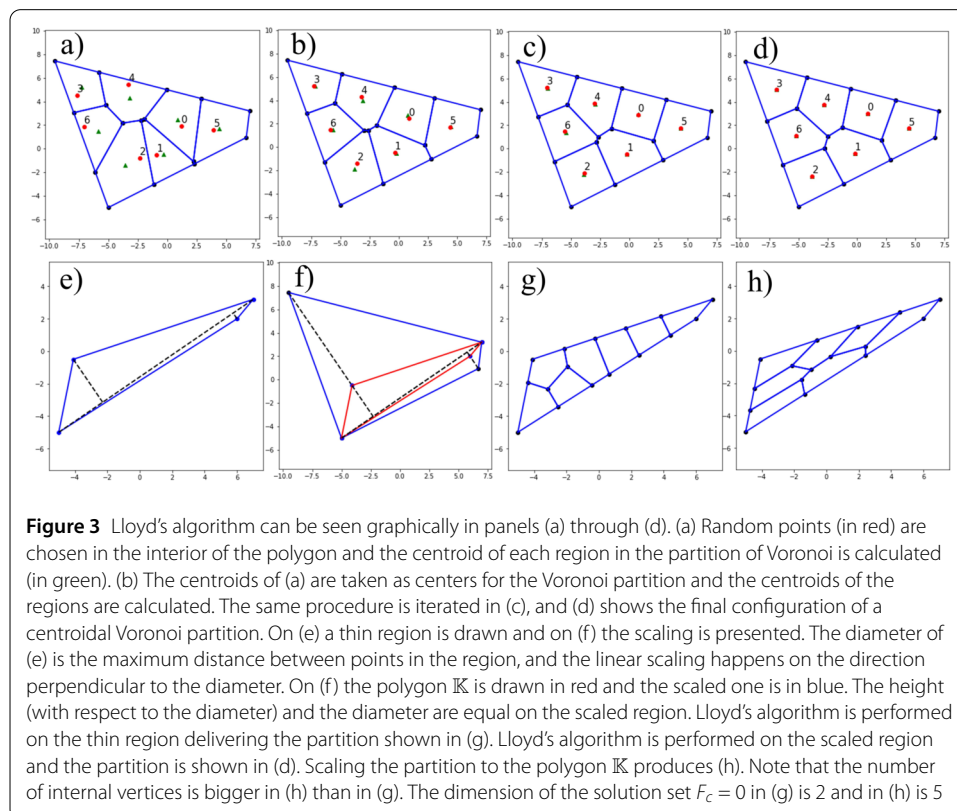### 3.4 Starting point: scaled centroidal Voronoi partitions

To apply the normal flow algorithm (Algorithm 2) to the excess area and perimeter function described in the previous section, we need to show the procedure that provides a starting convex partition that is close enough to a fair partition. Our proposal for this task is presented in the first four steps of our Algorithm 1, and it is a composition of a scaling of the polygon $\mathbb{K}$ with the use of Lloyd's algorithm to find a centroidal Voronoi partition.

Lloyd's original algorithm to find Voronoi partitions whose centroids match the Voronoi centers is as follows (Lloyd [11]). Start with a set of points inside a convex and compact region. Find the Voronoi partition that these points define and calculate the centroids of the regions in the partition. Take the centroids as Voronoi centers and iterate the procedure. Lloyd's algorithm converges to a centroidal Voronoi partition and it is extensively used because of its simplicity (Lloyd [11], Du et al. [12]). The naive $k$-means clustering algorithm is nothing else as the discrete version of Lloyd's algorithm.

Centroidal Voronoi partitions provide subdivisions of the polygon, which minimizes the sum of the polar second moment of area of all regions. Hence they all try to be very symmetrical with respect to the centroid; in some sense they are very round. The areas and perimeters of centroidal Voronoi partitions are not so different, and therefore they provide clear candidates for the starting partition of the normal flow algorithm (Algorithm 2). In Sect. 4.1 we provide numerical evidence of the proximity of the centroidal Voronoi partitions to the fair ones.

Since centroidal Voronoi partitions tend to be round, in thin regions the number of internal vertices might be very small (compare the centroidal Voronoi partition of Fig. 3(g) with the scaled centroidal Voronoi partition of Fig. 3(h)). By Corollary 1 we know that the number of internal vertices matches the number of free dimensions that the optimization problem has, and the number of internal vertices is bounded by the inequalities $0 \leq |I| \leq 2(n-1)$ since we have shown that $|I| + |B| = 2(n-1)$.

To find a starting partition that is close to the solution and with many internal vertices, we proceed to applying Lloyd's algorithm to a scaled polygon, which is also somewhat round.



**Figure 3** Lloyd's algorithm can be seen graphically in panels (a) through (d). (a) Random points (in red) are chosen in the interior of the polygon and the centroid of each region in the partition of Voronoi is calculated (in green). (b) The centroids of (a) are taken as centers for the Voronoi partition and the centroids of the regions are calculated. The same procedure is iterated in (c), and (d) shows the final configuration of a centroidal Voronoi partition. On (e) a thin region is drawn and on (f) the scaling is presented. The diameter of (e) is the maximum distance between points in the region, and the linear scaling happens on the direction perpendicular to the diameter. On (f) the polygon $\mathbb{K}$ is drawn in red and the scaled one is in blue. The height (with respect to the diameter) and the diameter are equal on the scaled region. Lloyd's algorithm is performed on the thin region delivering the partition shown in (g). Lloyd's algorithm is performed on the scaled region and the partition is shown in (d). Scaling the partition to the polygon $\mathbb{K}$ produces (h). Note that the number of internal vertices is bigger in (h) than in (g). The dimension of the solution set $F_c = 0$ in (g) is 2 and in (h) is 5

The *scaling of the polygon* procedure goes as follows. Take the diameter of the polygon, namely the longest distance on the hull of the region. Measure the longest heights on each side of the diameter, add them up, and call this number the height of the polygon. Scale the polygon linearly on the direction perpendicular to the diameter such that in the scaled polygon the height equals the diameter. In Fig. 3(e) the diameter and the two heights of the thin region are drawn in broken black lines. In Fig. 3(f) the scaled region is drawn using a solid line and its height matches the diameter of $\mathbb{K}$.

The scaled region is not thin anymore, and Lloyd's algorithm provides a centroidal Voronoi partition, which has more internal vertices than the centroidal Voronoi partition of the thin polygon $\mathbb{K}$. Scale back to $\mathbb{K}$ and notice that the induced partition provides more room to maneuver for the normal flow algorithm (see Fig. 3(g) and Fig. 3(h)).

Our Algorithm 1 performs pretty well with the scaling, while some thin polygons $\mathbb{K}$ cannot be solved by Algorithm 1 if the scaling is not performed. This is the main reason we included the scaling in the algorithm.

## 4  Numerical results

We have written a script in Python that carries out Algorithm 1 on any given convex polygonal region.

After several trials, we have noticed that two pieces of information that we have added to the script have made the algorithm more effective. Effective in terms of finding the convex fair partition in less trials. These two amendments are on the iteration of the normal flow algorithm (Algorithm 2) applied to $F^c$:

- The first ten iterations of the normal flow algorithm (Algorithm 2) shorten the size of the step by 1/10 on the first iteration, by 1/9 on the second, and so on, until the tenth iteration where the step is not modified anymore.
- Whenever the iteration sends any internal or boundary vertex outside the polygonal region, the step is shortened by half.
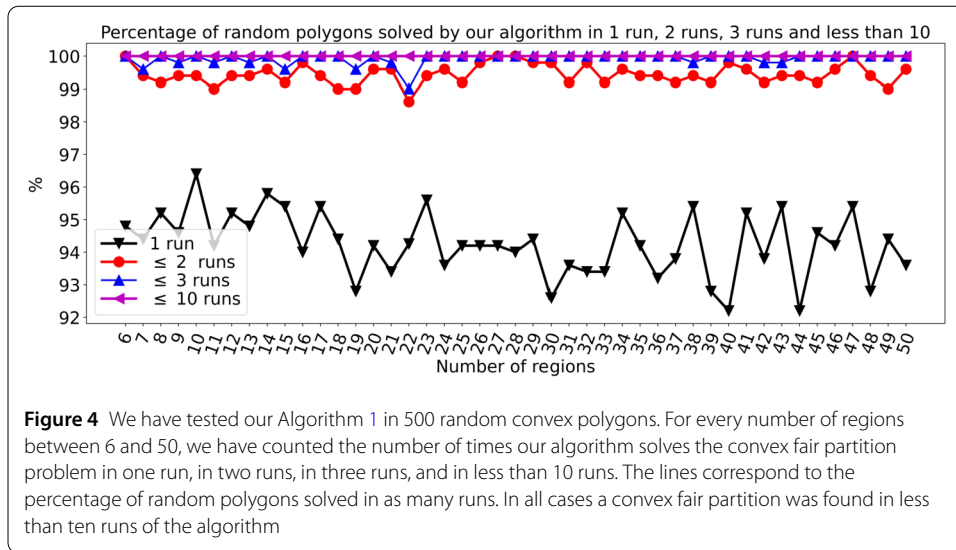
The first procedure avoids that the first steps change dramatically the convexity, therefore turning on the convex area functions of $F^c$, and the second restricts the internal and boundary vertices abandon the polygonal region.

Our convergence stopping criterion for the algorithm is $|\%F^c| \leq 10^{-6}$ with $|\%F^c|$ being the relative measure for the excess convex area and perimeter function, that is,

$$\left| \%F^c \right|^2 = \sum_{k=1}^{n} \left( \frac{A_k^c - \frac{1}{n}\sum_{i=1}^{n} A_i}{\frac{1}{n}\sum_{i=1}^{n} A_i} \right)^2 + \sum_{k=1}^{n} \left( \frac{P_k - \frac{1}{n}\sum_{i=1}^{n} P_i}{\frac{1}{n}\sum_{i=1}^{n} P_i} \right)^2. \tag{47}$$

This says that all areas and all perimeters have a portion of excess of at most $10^{-6}$ times the mean area or the mean perimeter respectively. In this case we say that the algorithm converges.

Now, by Theorem 1 we know that the normal flow algorithm will find a solution whenever the starting partition is close enough to the solution. Unfortunately, the scaled centroidal Voronoi partition that we propose to start the normal flow algorithm (Algorithm 2) may not always provide an initial configuration where Theorem 1 applies. Therefore our algorithm may not find the solution on only one run. Therefore the script is also stopped when the maximum number of iterations reaches the value of 100. In this case, Algorithm 1 is run again for the same polygon but with a different initial random point. We

**Figure 4** We have tested our Algorithm 1 in 500 random convex polygons. For every number of regions between 6 and 50, we have counted the number of times our algorithm solves the convex fair partition problem in one run, in two runs, in three runs, and in less than 10 runs. The lines correspond to the percentage of random polygons solved in as many runs. In all cases a convex fair partition was found in less than ten runs of the algorithm

follow this strategy at most 30 times. If the algorithm never reaches the convergence stopping criterion, we say that the algorithm failed to succeed.

## 4.1  Experiments and discussion

For a number of regions $n$ from 6 to 50, we have applied our Algorithm 1 to 500 random polygons. The random polygon is obtained by taking the convex hull of eight random points in the square $[-10, 10] \times [-10, 10]$ using a normal distribution on each variable. The results of Fig. 4 show the percentage of polygons where our Algorithm 1 succeeded in one run, in two runs, in three runs, and in less than ten runs. In all cases the algorithm found a convex fair partition in less than ten runs, and in 99.5% of cases, in three runs the convex fair partition was found.

This is strong evidence that our Algorithm 1 indeed finds a convex fair partition for polygonal regions.
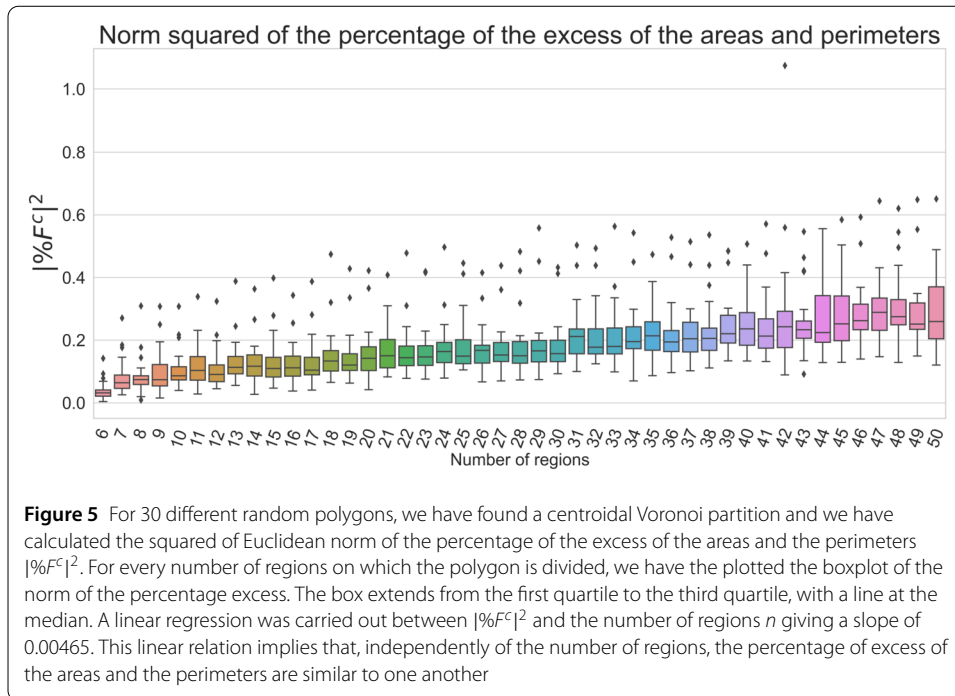
We have also carried out an analysis on how far centroidal Voronoi partitions are to convex fair ones.

In Fig. 5 we have plotted the distribution of the squared norm of the percentage of the excess of the areas and perimeters $|\%F^c|^2$ for the centroidal Voronoi partitions, with respect to the number of regions on which a polygon is divided. We have done this calculation on 30 random polygons for a number of regions between 6 and 50. We see a linear correlation between $|\%F^c|^2$ and the number of regions $n$ with slope 0.00465. This shows that the squares of the percentage of the excess of the areas and the perimeters are close to each other, independently of the number of regions.

One way to interpret this linear correlation is via the following argument. Take a polygon and find a centroidal Voronoi partition of $nm$ regions. Compare it with the partition obtained first by finding the centroidal Voronoi partition of $n$ regions, and then partitioning each one of them into $m$ by Lloyd's algorithm. Both may have similar geometrical shapes, and therefore they may have similar percentage of excess in the areas and the perimeters.

The closeness of the centroidal Voronoi partitions to the convex fair ones is the reason we decided to start the normal flow algorithm (Algorithm 2) on them.

A shortcoming of our algorithm has appeared whenever the polygonal region $\mathbb{K}$ is too thin. After many numerical experiments we conclude that too thin means that the height
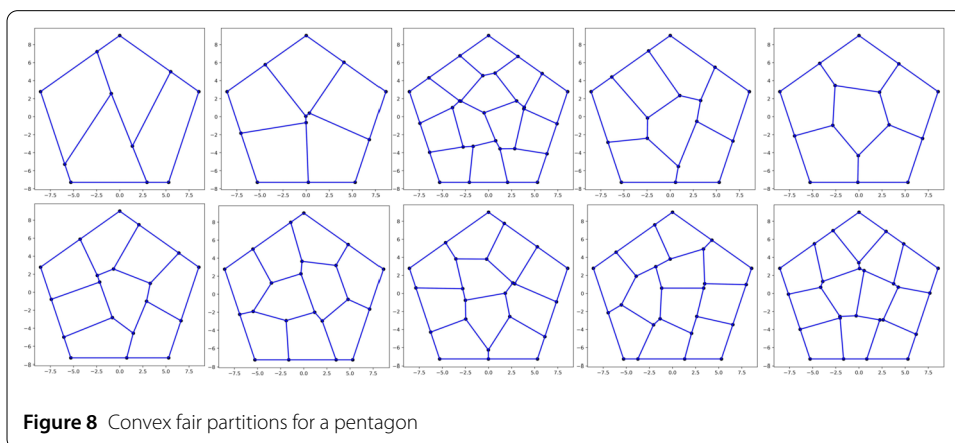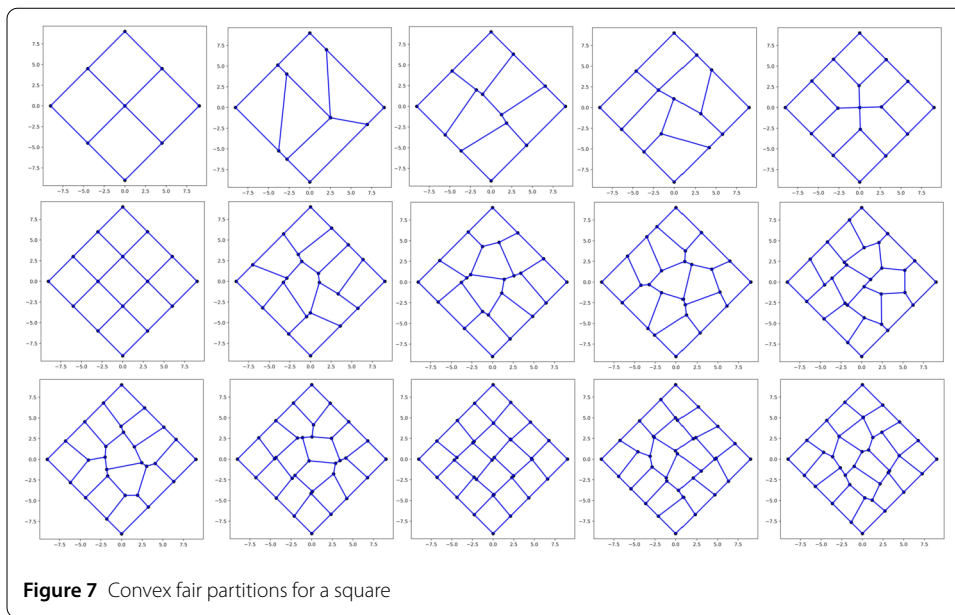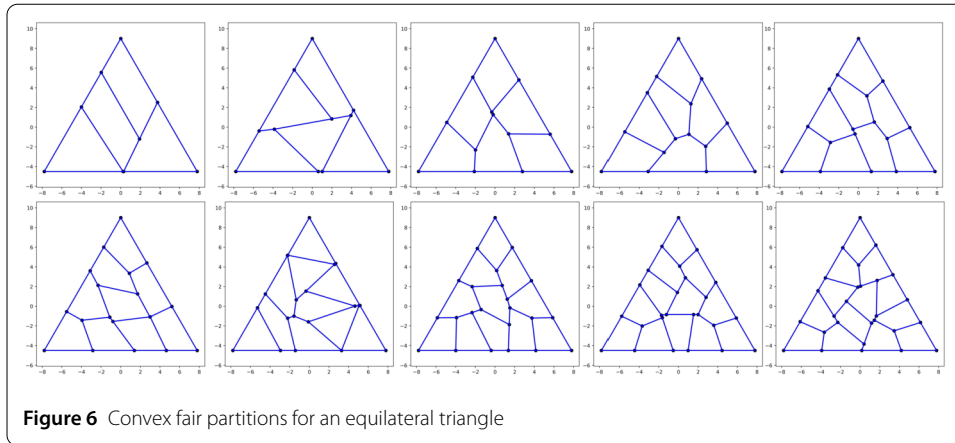
**Figure 5** For 30 different random polygons, we have found a centroidal Voronoi partition and we have calculated the squared of Euclidean norm of the percentage of the excess of the areas and the perimeters $|\%F^c|^2$. For every number of regions on which the polygon is divided, we have the plotted the boxplot of the norm of the percentage excess. The box extends from the first quartile to the third quartile, with a line at the median. A linear regression was carried out between $|\%F^c|^2$ and the number of regions $n$ giving a slope of 0.00465. This linear relation implies that, independently of the number of regions, the percentage of excess of the areas and the perimeters are similar to one another

is smaller than one twentieth of the diameter. In those cases the steps of the normal flow algorithm (Algorithm 2) tries to send the internal vertices out of the polygon and our script does not in general converge. A better suited partition to start the normal flow algorithm might be necessary in these cases. On the other hand, the convergence rate of Lloyd's algorithm for the centroidal Voronoi partition is affected by the number of regions considered. Its linear convergence rate of $1 - \mathcal{O}(h^2)$, where $h$ is the minimum diameter of the regions (Hateley et al. [13]), is slow when the number of regions is large. Therefore, better algorithm options for its implementation may be used for increasing efficiency when a large number of partitions is required (Hateley et al. [13]). We leave these questions open.

## 5  Conclusions

We have proposed an algorithm for finding a convex fair partition to any convex polygon into any number of regions. The algorithm is based on finding a centroidal Voronoi partition on a scaled region of the given polygon and on iterating the normal flow algorithm for the associated excess convex area and perimeter function. Our algorithm is the first one to appear in the literature designed to solve the convex fair partition problem without any restriction. The algorithm successfully identified the desired partition across a large and wide variety of tests. This outcome appears to support the conjecture that a convex fair partition exists for any convex polygon and any number of regions.

## Appendix

To have a graphical idea of how convex fair partitions look like on known polygonal regions, in Figs. 6, 7, and 8 we have used our Algorithm 1 to calculate convex fair partitions on an equilateral triangle, on a square, and on a pentagon respectively. Note that the par-

**Figure 6** Convex fair partitions for an equilateral triangle



**Figure 7** Convex fair partitions for a square



**Figure 8** Convex fair partitions for a pentagon

titions of the square into 4, 6, 8, 9, and 16 regions are configurations that are very symmetrical.

**Author contributions**

**Data availability**

Not applicable.

# Declarations

**Ethics approval and consent to participate**

Not applicable.

**Consent for publication**

Not applicable.

**Competing interests**

The authors declare no competing interests.

**References**

1. Roldán-Pensado, E., Soberón, P.: A survey of mass partitions. Bull. Am. Math. Soc. (N.S.) **59**, 227–267 (2022). https://doi.org/10.1090/bull/1725
2. Nandakumar, R., Ramana Rao, N.: Fair partitions of polygons: an elementary introduction. Proc. Indian Acad. Sci. Math. Sci. **122**, 459–467 (2012). https://doi.org/10.1007/s12044-012-0076-5
3. Bárány, I., Blagojević, P., Szűcs, A.: Equipartitioning by a convex 3-fan. Adv. Math. **223**, 579–593 (2010). https://doi.org/10.1016/j.aim.2009.08.016
4. Blagojević, P.V.M., Ziegler, G.M.: Convex equipartitions via equivariant obstruction theory. Isr. J. Math. **200**, 49–77 (2014). https://doi.org/10.1007/s11856-014-1006-6
5. Karasev, R., Hubard, A., Aronov, B.: Convex equipartitions: the spicy chicken theorem. Geom. Dedic. **170**, 263–279 (2014). https://doi.org/10.1007/s10711-013-9879-5
6. Akopyan, A., Avvakumov, S., Karasev, R.: Convex fair partitions into an arbitrary number of pieces (2018). https://arxiv.org/abs/1804.03057. https://doi.org/10.48550/ARXIV.1804.03057
7. Borsuk, K.: Drei Sätze über die *n*-dimensionale euklidische Sphäre. Fundam. Math. **20**, 177–190 (1933). https://doi.org/10.4064/fm-20-1-177-190
8. Armaselu, B., Daescu, O.: Algorithms for fair partitioning of convex polygons. Theor. Comput. Sci. **607**, 351–362 (2015). https://doi.org/10.1016/j.tcs.2015.08.003
9. Ben-Israel, A.: A Newton-Raphson method for the solution of systems of equations. J. Math. Anal. Appl. **15**, 243–252 (1966)
10. Walker, H.F., Watson, L.T.: Least-change secant update methods for underdetermined systems. SIAM J. Numer. Anal. **27**, 1227–1262 (1990). https://doi.org/10.1137/0727071
11. Lloyd, S.P.: Least squares quantization in PCM. IEEE Trans. Inf. Theory **28**, 129–137 (1982). https://doi.org/10.1109/TIT.1982.1056489
12. Du, Q., Emelianenko, M., Ju, L.: Convergence of the Lloyd algorithm for computing centroidal Voronoi tessellations. SIAM J. Numer. Anal. **44**, 102–119 (2006). https://doi.org/10.1137/040617364
13. Hateley, J., Wei, H., Chen, L.: Fast methods for computing centroidal Voronoi tessellations. J. Sci. Comput. **63**, 185–212 (2015). https://doi.org/10.1007/s10915-014-9894-1

## Publisher's Note