Rochester Institute of Technology

COMPUTATIONAL GEOMETRY

CSCI 710

FAIR CONVEX PARTITIONS OF CONVEX POLYGONS

Authors :
Paritosh Borkar (pb2982)
Vineet Kamat (vvk1199)

Contents

1 PROBLEM DESCRIPTION

Define a fair partitioning of a polygon as a partition of it into a finite number of pieces so that every piece has both the same area and the same perimeter. If all the resulting pieces are convex, call it a fair convex partitioning. Given any positive integer n, can any convex polygon be convex fair partitioned into n pieces? [1]

2 INTRODUCTION

This problem is easy to grasp and can be solved through some clever intuition, thus appealing to us. There are solutions for specific values of n, but no solution for a general value of n. It doesn't seem to have any applications but its analogy in 3 dimensions does make for a problem that can be used for practical industrial purposes. In 3 dimensions the problem converts into dividing a solid object into pieces of equal surface area and volume[2]. The problem can also have applications in topology or areas of mathematics that involve equally partitioning a domain of numbers.

3 BACKGROUND

This problem was first asked by R. Nandakumar and N. Ramana Rao[3] in 2008. They introduced the idea of fair convex partitions of polygons and proposed that it is possible to divide a polygon fairly for any number of partitions. Later on they provided proofs for $n = 2$ and $n = 4$ and discuss proofs for higher powers of 2. Later Boris Aronov and Alfredo Hubard provided proofs for $n = p^k$, where p is a prime number and k is any natural number. Their proof worked for any d-dimensional polygon partitioned into n pieces of equal "volume" and "surface area"[4].

4 PROJECT GOALS

1. Proof for $n = 6$:
   Currently there are proofs for this problem only when $n = p^k$ [4], where p is any prime number and $k \in \mathbf{N}$. The project's objective was to attempt to prove it for n that does not fall under this category. Proving it for all n is a much harder problem than proving it for a specific number so we decided to

tackle this problem for $n = 6$, which is the smallest number that does not have a proof yet.

2. Visualising solutions for the problem:
For $n = 2^k, k \in \mathbf{N}$ there are algorithms that allow us to find the fair partitions for a convex polygon. One of the objectives was to provide an online graphical tool that allows users to visualise the solutions for $n = 2^k$ for any given polygon.

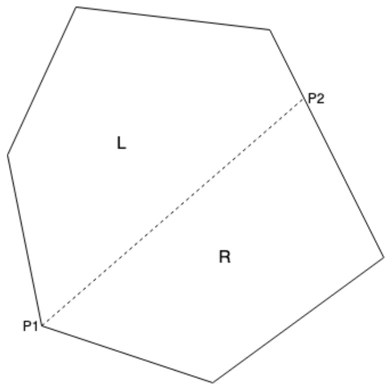
5 OBJECTIVE ATTEMPTS

Proving for n = 6

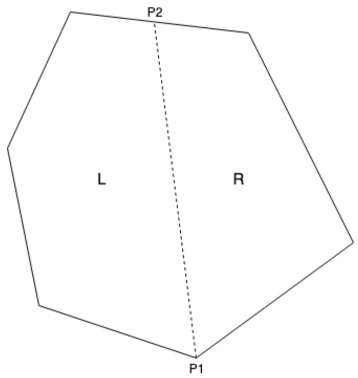
Our idea to tackle the problem for $n = 6$ involved using proofs for $n = 2$ and $n = 3$ simultaneously. We wanted to use the proof for $n = 3$ to get 3 fair partitions and then using the proof for $n = 2$ for each of the partitions to produce 6 fair partitions. This was easier said than done because of a couple of reasons:

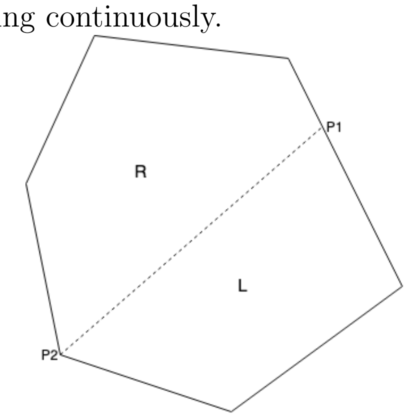
- Understanding the proof for $n = 3$ required extensive knowledge of topology, which neither of us had any experience with. The proof for $n = 2$ and $n = 3$ will be explained qualitatively in this section.
- Simply applying the solutions doesn't work as the perimeters are not conserved. The reason this problem is difficult to solve is because finding partitions with equal perimeters is not an easy task.

Proof for n = 2 [3]
We need to make the assumption that any convex polygon has a line segment that passes through it, dividing it into 2 pieces of equal areas. Let a line segment with endpoints $P1$ and $P2$, divide the polygon into 2 pieces called L an d R, the left and right pieces. Here $P1$ and $P2$ lie on the boundary of the polygon.



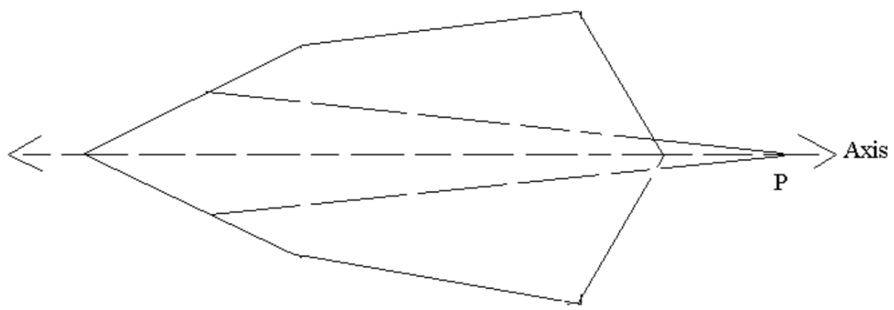
Without loss of generalisation assume that the perimeter of the L is greater than the perimeter of R. Now we slowly rotate this segment in anti-clockwise direction, maintaining the areas on both side. Because we are moving $P1$ and $P2$ continuously, the perimeters of L and R are also changing continuously.



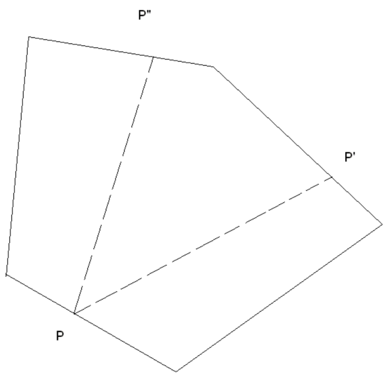


At some point $P1$ will reach the original position of $P2$, while $P2$ will reach the original position of $P1$. But in this case the perimeter of L is now smaller than the perimeter of R. This implies that at some point in the rotation, both perimeters of L and R must have been equal. Therefore it is possible to obtain a fair partition of a convex polygon for $n = 2$.

Proof for n = 3 [3]
This is not a conclusive proof, but gives you an idea of how the proof can be constructed. Any polygon can be divided into 3 equal areas by shooting 3 rays from an interior point to the polygon or by shooting 2 rays from an exterior point to the polygon. If the polygon is a reflective convex polygon(i.e. has one line of symmetry), then these points lie on an infinite axis called the reflective axis. Two of the pieces are guaranteed to be reflections of each other on the reflective axis.



As we move on this line continuously, it follows that the perimeters of the pieces also change continuously. Therefore there is a point on this axis that can divide the polygon into 3 pieces of equal areas and perimeters by shooting 2 or 3 rays, depending on whether the point is an exterior or interior point respectively.
For a general convex polygon, we need the existence of a poly-line from which shooting rays will give us 3 convex pieces of equal areas and perimeters. For every point on the polygon there exists exactly 2 rays that divide the polygon into pieces(left, middle and right) of equal areas and perimeters. Let P be a point on the polygon with 2 rays shooting from it that meet the polygon at P' and P''. For point P', the right piece of P is the left piece of P', while the left piece of P is the right piece of P''.



Let's call the perimeters of the left and right pieces P_l and P_r respectively. Plotting the values of P_l and P_r against the position of P gives us two curves. You'll observe that the . range of values for both curves is exactly the same, as well as that the curves are periodic with respect to the position of P on the polygon. These two properties imply that both curves necessarily intersect at least twice. This means there are at least two points on the polygon that equal areas for both the left and right sides.
Consider a slightly larger polygon created from the original polygon. This new polygon will also have a point P' such that by shooting 2 rays (meeting the polygon at

P_1' and P_2') from it you get 3 pieces of equal areas and perimeters. These points P_1' and P_2' will lie in the immediate neighborhood of P' and P''. By continuously expanding the polygon we can plot the trajectory of P_1' and P_2', which gives us the poly-lines that we required. To get a full axial poly-line we need to prove that there exists a bridge from the exterior of the polygon to its interior.
This shows that the existence of an axial poly-line will guarantee a set of points that can serve as the starting points for rays that divide the polygon into fair partitions.

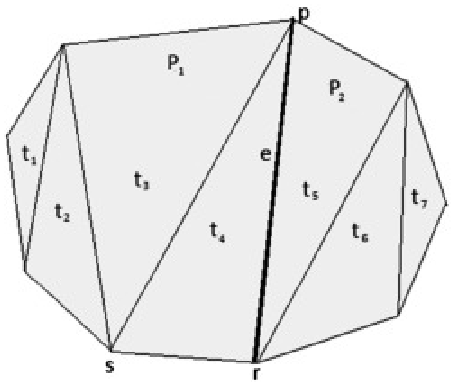
We tried to combine both proofs to see if the problem can be proved for $n = 6$. We weren't able to do it, but looking at these solutions gave us key insight into how one could go about tackling this problem.

Visualising the solution for $\mathbf{n = 2^k}$

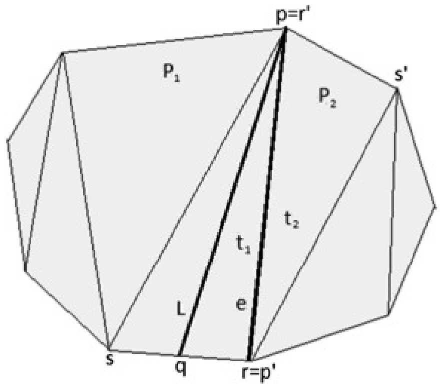
We found a paper by Bogdan Armaselu and Ovidiu Daescuthat[5] that describes a method for finding fair partitions when $n = 2^k$ in $\mathcal{O}(n)$ time. This section will describe how the algorithm works qualitatively as well as the problems we faced when trying to implement the algorithm.
Before describing the algorithm, we would like to point out that the paper is riddled with mistakes which made it difficult to implement the algorithm. The paper has 2 parts to it, the proof that shows how the algorithm works and the pseudo code for it. There are numerous discrepancies between the proof and the pseudo code including but not limited to notation mismatch, ambiguous language and irrelevant variables.

- **Algorithm Approach**:
  The algorithm tries to find a fair partition by first finding a single area bisector of the polygon that divides the polygon into its left and right parts. Once it has the area bisector, it is rotated around the polygon until both the left and right parts have equal perimeters too. This will give us a fair partition that divides the polygon into 2 fair pieces. To find the pieces for $n = 2^k$ apply this algorithm recursively for each of the partitions.

- **Finding the area bisector**:
  To find the first area bisector the polygon is first triangulated.

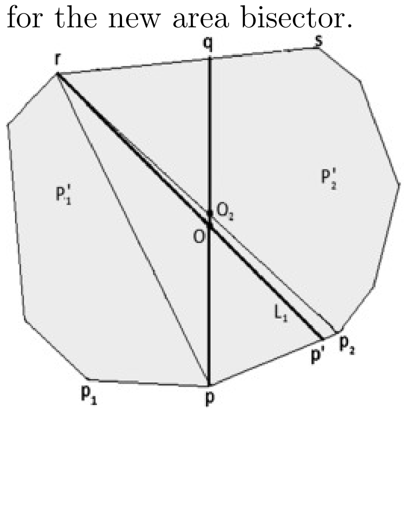


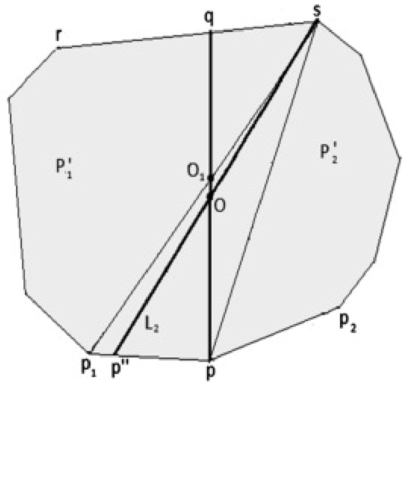
Looking at each triangle boundary from left to right, find the boundary at which the area of the triangles on the left side becomes greater than the ones on the right side. When this happens we know that the area bisector lies somewhere between the current boundary and the previous one.

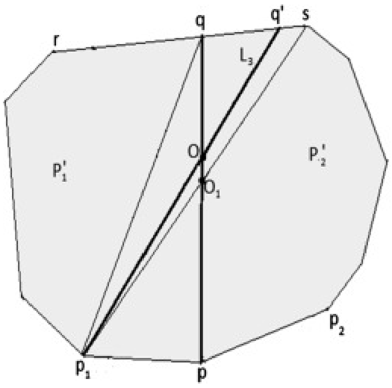


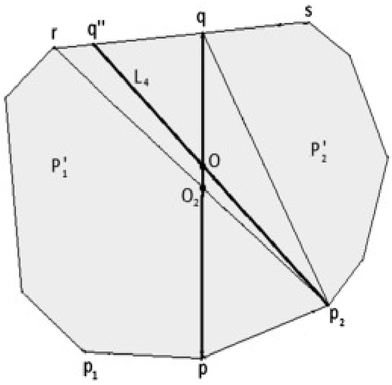
Equating the areas for P1 - t1 and P2 + t1 we arrive at the formula for q given by $q = \frac{D1}{2area(t1)}\vec{sr} + s$

- **Updating the area bisector**:
  Once we have an area bisector L with endpoints p and q, we need to update it till the left and right pieces have the same perimeter. There are 4 possibilities for the new area bisector.









In each of cases we find the difference in perimeters between the left and right sides and update the area bisector to the one with the least difference. To find the 4 area bisectors we do something similar to what we did to find q. We calculate p', p'', q' and q'' with:

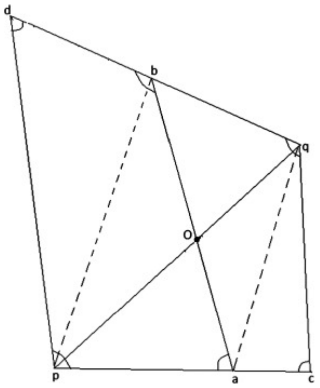
$$p' = \frac{area(\Delta pp_2O_2) - area(\Delta qrO_2)}{pp_2 \cdot rp_2 \cdot sin(\angle rp_2p)} \cdot \vec{p_2p} + p_2$$
$$p'' = \frac{area(\Delta pp_1O_1) - area(\Delta qsO_1)}{pp_1 \cdot sp_1 \cdot sin(\angle sp_1p)} \cdot \vec{p_1p} + p_1$$
$$q' = \frac{area(\Delta qsO_1) - area(\Delta pp_1O_1)}{sq \cdot sp_1 \cdot sin(\angle qsp_1)} \cdot \vec{sq} + s$$
$$q'' = \frac{area(\Delta qrO_2) - area(\Delta pp_2O_2)}{rq \cdot rp_2 \cdot sin(\angle qrp_2)} \cdot \vec{rq} + r$$

Because of the way these are derived, either only $L1$ or $L4$ exists and either $L2$ or $L3$ exists. We denote the area bisector $L1'$ among $L1$ or $L4$ and $L2'$ among $L2$ or $L3$.
Between $L1'$ and $L2'$ we need to update our area bisector L so that the difference between perimeters on the left and right side is minimized. Let the difference in perimeters be given by:

$$Dp = perimeter(P_1) - perimeter(P_2)$$
$$Dp_1 = perimeter(P_{11}) - perimeter(P_{12})$$
$$Dp_2 = perimeter(P_{21}) - perimeter(P_{22})$$

There are 3 cases to consider to find the endpoints of the updated area bisector L'. The endpoints are given by a and b:
Case 1:
If $Dp = 0$ then $a = p$ and $b = q$
Case 2:
If $Dp \cdot Dp_1 < 0$ or $Dp \cdot Dp_2 < 0$ then this implies that difference in perimeters

have changed sign. In this case we know that there has to be an area bisector L' between either L and $L1'$ or L and $L2'$ such that the left and right pieces created by L' have the same perimeter. To find a and b in this case first let's denote the c and d to be the endpoints of either $L1'$ or $L2'$, depending on which of the above two conditions are satisfied. Consider the figure:



We define the following variables to calculate the values of a and b: $\Delta^1 = area(dpq)$ and $\Delta^2 = area(cpq)$. Also $\Delta_1 = area(dpb)$, $\Delta_2 = area(pba)$, $\Delta_3 = area(baq)$, $\Delta_1 = area(aqc)$. In order for $area(P_1') = area(P_2')$ to hold as well as $Dp' = 0$, the following equations must be satisfied:

$$\Delta^1 = \Delta_1 + \Delta_2$$
$$\Delta^2 = \Delta_3 + \Delta_4$$
$$qb - ap = Dp$$

Solving these set of equations we get:

$$(\Delta^1 - \Delta^2) \cdot qb^2 + (\Delta^2 \cdot (Dp + qd) - \Delta^1 \cdot (Dp + pc)) \cdot qb - \Delta^2 \cdot Dp \cdot qd = 0$$

Solving this equation we get

$$qb_1 = \frac{-B + \sqrt{|B^2 - 4AC|}}{2A}, qb_2 = \frac{-B - \sqrt{|B^2 - 4AC|}}{2A}$$

where $A = \Delta^1 - \Delta^2$, $B = \Delta^2 \cdot (Dp + qd) - \Delta^1 \cdot (Dp + pc)$ and $C = -\Delta^2 \cdot Dp \cdot qd$
Assign qb to either $qb1$ or $qb2$, depending on whichever satisfies $area(\Delta aOp) = area(\Delta bOq)$. Once we qb we can find ap with $ap = qb - Dp$
We calculate a and b by:

$$b = q + (d - q) \cdot qb$$
$$a = p + (c - p) \cdot ap$$

This gives us the endpoints for the fair bisector L'.
Case 3:
Set L' to the area bisector corresponding to $Dp' = min(Dp_1, Dp_2)$

- It is possible that while updating the area bisector, we might end up updating it to an area bisector that we have previously visited. To avoid this a stack, Λ, of previously visited area bisectors is maintained. At every that step where we don't find the fair bisector we push L onto the stack and update it to one of $L1'$ or $L2'$, whichever has the smallest difference in perimeters between the left and right sides, as well as is not present in Λ.

We were able to implement the algorithm except for the part where it need to calculate the exact fair partition. There seems to be some error in calculating the values of qb and ap that gave us incorrect values for the endpoints of the fair bisector. There's a discrepancy between the proof and the pseudo code. Further study is required to figure out the bug.

6 RESULTS

1. Studied the proofs for the problem for values of n when n = 2 and 3 [3].
2. Attempted to solve the problem for n = 6.
3. Studied a paper[5] that describs an $\mathcal{O}(n)$ algorithm to find the fair partition for a convex polygon when $n = 2$ and $n = 2^k$.
4. Attempted to implement the fair partition algorithm for $n = 2$ in Python.
5. Created a Github page that hosts the algorithm implementation as well as this report.

7 REFERENCES

References

[1] Problem 67: Fair Partitioning of Convex Polygons
http://cs.smith.edu/ jorourke/TOPP/P67.htmlk-esm-10

[2] Open Problem 67: Fair Partitioning of Convex Polygons
https://parasol.tamu.edu/ amato/Courses/620/openProblems/csce620-openProblem-P67_FairPartitioning_NicolasCastet.pdf

[3] R. Nandakumar, N. Ramana Rao: 'Fair' Partitions of Polygons - an Introduction.
https://arxiv.org/abs/0812.2241

[4] Boris Aronov, Alfredo Hubard: Convex Equipartitions of volume and surface area
https://arxiv.org/abs/0812.2241

[5] Bogdan Armaselu, Ovidiu Daescu: Algorithms for fair partitioning of convex polygons
https://doi.org/10.1016/j.tcs.2015.08.003