A Mathematical Approach to Paranormal Protection

Énora Hauduc

Tom Rocks Maths: Essay competition

Computational geometry

2025

1.1 Ghostbusters – An introduction

In the iconic billion-dollar franchise Ghostbusters, a team of parapsychologists in New York City makes it their mission to eliminate paranormal occurrences. Armed with a variety of cutting-edge gadgets, the team takes on the thrilling task of combating supernatural forces. The most notable gadget is the proton pack—a portable particle accelerator device used to control and capture ghosts by emitting a beam of charged particles. Weighing a hefty 100 pounds, the device is awkward and bulky, immobilising its user during operation.

Figure 1 – Ghostbusters (Reitman, Ivan. Ghostbusters. Columbia Pictures, 1984), Proton Pack (Wikipedia, 2025)

It is one of the most powerful devices in the Ghostbusters universe and, from a mathematical perspective, one of the most interesting. Beyond its cinematic appeal, it invites a curious mathematical question: In a room filled with ghosts, how can we optimally position the Ghostbusters – each equipped with a proton pack - to capture all ghosts in a room of any shape?

1.2 Underlying assumptions

As with any mathematical model, it is a good idea to lay some groundwork. Here are some assumptions we will establish:

1. Proton packs emit a continuous and infinitely long beam in a straight line. A ghost can be captured if and only if it lies along an unobstructed straight path (direct line of sight) from a Ghostbuster's position, meaning beams cannot capture ghosts through walls.
2. Due to its 100-pound weight, ghostbusters will be immobile while using the proton pack. However, they can freely rotate and aim the device in any direction from their fixed position.

3. All rooms will be modelled as simple polygons meaning their walls form a closed shape without self-intersections or holes. For simplicity (and because abandoned haunted buildings are conveniently empty), we will assume there are no internal obstacles such as furniture either.

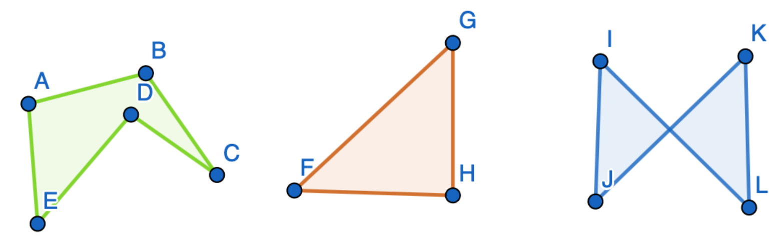


Figure 2 - Simple polygons (green and orange) and a complex self-intersecting blue polygon, which will not be considered in this model

2.1 Initial experiments

To start, it's a good idea to imagine the simplest possible room we could encounter: a triangle. By imagining the proton pack's beam rotating across the entire room, we can quite easily see that the beam has a direct line of sight to every point within the triangle from any position inside it. From this, we can see that one ghostbuster placed anywhere within the triangle would be sufficient to catch a ghost anywhere in the room.

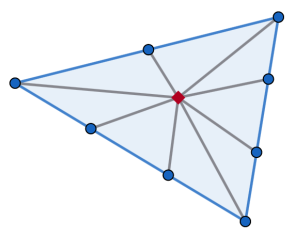


Figure 3 - A ghostbuster is represented by the red diamond and the grey spokes show their direct line of sight to all areas of the triangle

However, apartments in New York tend to be a bit more complicated and their architecture is less… well, triangular. Since our model should work for any simple polygon, let's try this 12-sided polygon to start with:

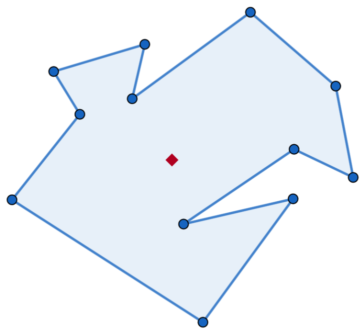


Figure 4 - An example room with a 12-sided polygon

By applying the same logic as with the triangular room, imagine the proton pack's beam rotating from the central spot around the room. By doing this, we can see an issue straight away: the beam cannot reach all areas of the room.

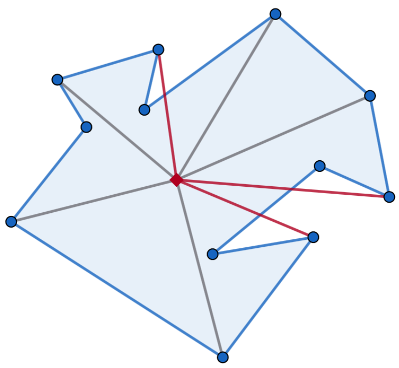


Figure 5 - Red spokes denote areas that the beam cannot directly access

The visible region from the central point in the polygon forms what's known as a 'star-shaped region' – a shape where all points are visible from a single interior point. We can highlight this area using an orange overlay. So, while the majority of the room *is* visible to the proton beam, if a pesky ghost happens to hide in one of the corners that aren't covered by the beam's visible region, then it's game over.

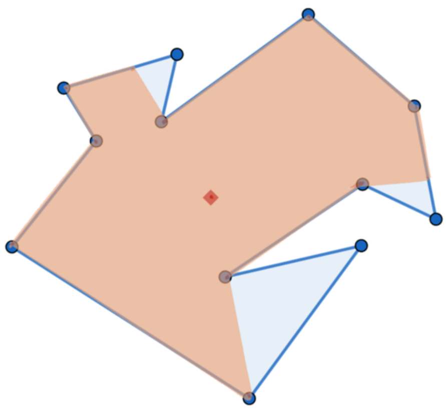


Figure 6 – Visible region of the Ghostbuster (red diamond) represented by the orange overlay

But let's just take a closer look at this star-shaped region. It's a particularly convenient shape because if the room *was* that exact shape, then one Ghostbuster would be enough to cover it entirely.

What this orange overlay is showing are all the points that, when connected to the central Ghostbuster using a segment, remain entirely inside the polygon P. Mathematically, we can represent all the points in the visible region using the definition:

A point $t \epsilon P$ is in the visible region when $\overline{tc} \subseteq P$ where c is the Ghostbuster's position and P is the polygon

If we want to be able to access the entire room, however, maybe we could think about splitting the room up into smaller rooms and tackling each section individually. By decomposing the problem into smaller sub-problems, we can try to tackle this task. This seems sensible because from what we have seen so far, as the number of sides in room increases, it becomes much harder to clearly see how many Ghostbusters we will need.

2.2 Splitting up the room

So, we know that one Ghostbuster can guard any star-shaped region. That suggests a strategy where we could split the room into several star-shaped regions. After placing a Ghostbuster in each one, we could take leftover regions and repeat the process by breaking them down into stars until everything is covered. It works… but things are quickly getting messy.

Let's take a step back. What made the star-shaped region so effective? It's the visibility: the fact that the central point had a direct line of sight to every point in the region.

So, are there any other polygons that always have this property? Yes! In fact *any* convex polygon, meaning all its interior angles are less than 180°, will have this property. In any convex shape, each point is visible from any other point inside it. Logically, this makes sense since all the polygon's boundaries bulge outwards, not inwards, meaning there aren't any indentations or holes that would block the line of sight. In other words, if there is a ghost and a ghostbuster in any convex shape then the ghostbuster will always be able to capture the ghost because it will have a direct line of sight.

Perfect! This provides us with a new strategy: instead of trying to build star-shaped regions, we can partition the room into convex polygons. Then, by placing a Ghostbuster in each sub-region, we're guaranteed to cover the entire room.

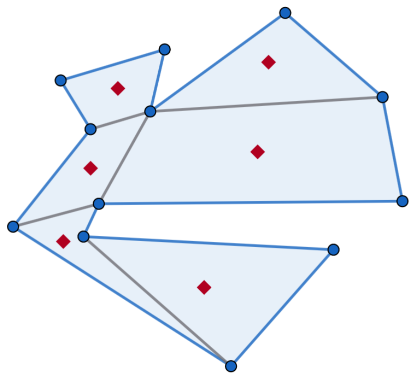


Figure 6 - Splitting our room into convex polygons and adding a Ghostbuster inside each sub-polygon

There! Problem solved!

... Or is it?

Ghostbusters are expensive to hire and we don't want to waste valuable resources by using more than necessary. In Figure 7, we can see that we can remove at least two of the Ghostbusters and the room is still equally guarded. So, while this method works, it's not optimal.

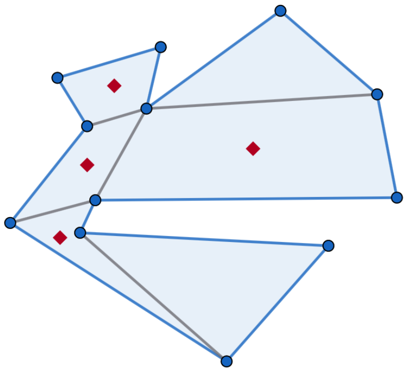


Figure 7 - Removing some Ghostbusters doesn't impact the coverage

Now we're getting somewhere. This approach of splitting into convex polygons is promising but we still need to find a way to minimise the number of Ghostbusters. Can we generalise this idea to find a method which gives us an optimal solution *every time?*

2.3 Returning to basics

Let's return to the simplest convex polygon room: the triangle. With just three sides, it's easy to see that placing a Ghostbuster at any one of these vertices would allow us to cover the entire room. But what happens if the room is more complex?

Through a process known as triangulation, we can split any simple polygon into a set of non-overlapping triangles whose union fills the entire shape. This is crucial because we saw that placing a Ghostbuster at one of the triangle's vertices can cover the entire area. Therefore, once a more complicated room has been triangulated, we're just left with the task of determining which Ghostbusters are necessary to keep.

First, let's prove that we *can* always triangulate a polygon.

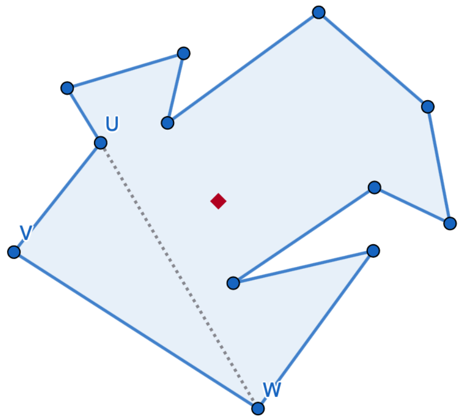
Using proof by induction:

Let n be the number of sides of a simple polygon P.

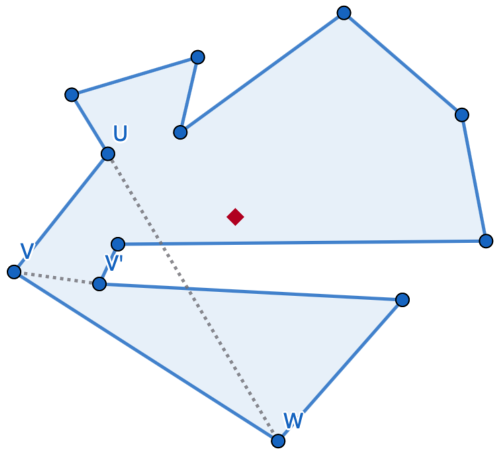
Our base case is when $n = 3$. Here, the polygon is already a triangle, so no further triangulation is required.

Consider the next case where $n > 3$. Because the sum of the interior angles in any simple polygon is $180(n - 2)°$, at least one of the interior angles must be less than $180°$ (meaning it is convex) otherwise the sum of interior angles would be $180n°$. Therefore, the polygon must contain at least one convex vertex V.

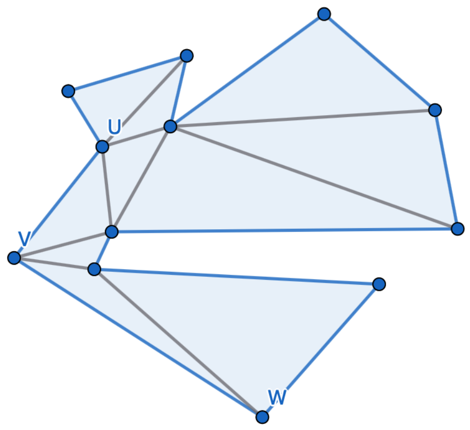
Since $n > 3$ there will be at least two adjacent vertices on either side of our convex angle. Let's call them W and U. If we can connect W and U, creating a segment $\overline{WU}$ which lies entirely within the polygon, then we have successfully created a valid triangle ΔWVU. Repeating this for the rest of the polygon would successfully triangulate the polygon.



If, however, the segment $\overline{WU}$ causes us to leave the polygon (by crossing the exterior), then that means that within the triangle ΔWVU there must be some other vertices in the way. In this case, we can choose the interior vertex V' furthest from the segment $\overline{VU}$ that is still within the triangle ΔWVU. Now if we draw a diagonal from V to this further vertex, V', then this line won't cross any existing edge, creating a valid diagonal.



In both of these cases, the diagonal splits the original polygon P into two smaller polygons P_a and P_b, each with fewer than n sides. By the inductive hypothesis, both P_a and P_b can be triangulated. Hence, so can P.



So we've triangulated the room. What now? Well, as before, we could stop here and put a ghostbuster in the vertex of every triangle. That *would* ensure full coverage, but it wouldn't be efficient. There must be a better way!

2.4 Minimising the number of Ghostbusters

We know from earlier that a single Ghostbuster can cover an entire triangle by standing on any of its three vertices. We also know that positioning one ghostbuster on one vertex can sometimes cover multiple triangles. So, we want to place as few Ghostbusters as possible, each covering as many triangles as they can.

One way to approach this is by using a method known as 3-colouring. We can assign one of three colours to each vertex of our triangulated polygon (e.g Red, purple and yellow) so that no triangle has two vertices of the same colour.

Now imagine each colour as a potential formation of Ghostbusters. Because every triangle contains all three colours, selecting any one colour group ensures that each triangle has at least one Ghostbuster. And because we want to minimise the number of Ghostbusters, we can simply choose the colour group with the fewest vertices.

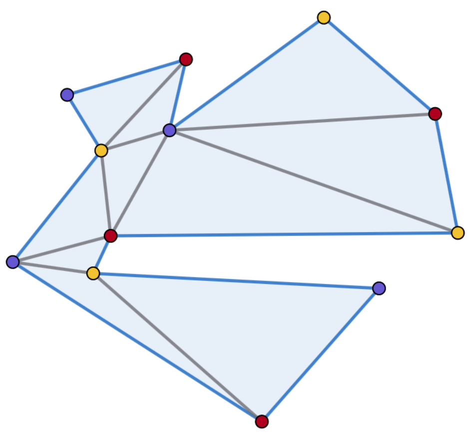


Figure 8 - Our triangulated example room with 3-colouring

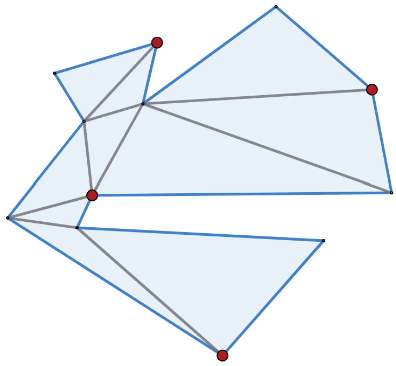


Figure 9 - For our example room, all three colours appear in equal number. So we can choose any one of these colour groups to represent our Ghostbuster arrangement, resulting in a solution where $\frac{n}{3}$ Ghostbusters are necessary.

x

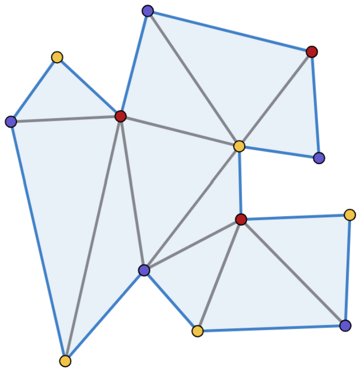


Figure 10 - However for this room, one colour group appears fewer times than the others. Choosing that group, gives us a more efficient solution where fewer than $\frac{n}{3}$ Ghostbusters are sufficient.

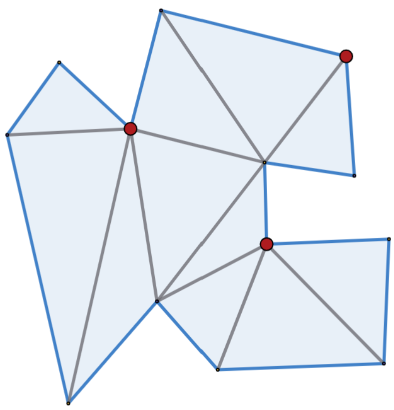


Figure 11 - The optimal solution for this room

We can generalise this using the pigeonhole principle. If we split n vertices into 3 colour groups, then at least one group must have no more than $\frac{n}{3}$ items. If each group had more than $\frac{n}{3}$ vertices, the total number of vertices would exceed n which is a contradiction to the number of vertices being n.

Therefore, we can guarantee that there will always be a way to position the Ghostbusters such that at most $\frac{n}{3}$ are needed to guide a room of size n sides.

We can express this as: For any simple polygon with n vertices, $\frac{n}{3}$ Ghostbusters will always be sufficient and sometimes necessary.

2.5 Case closed

So there we go – case closed!

With this understanding, we've now cracked the mathematical puzzle behind the optimal placement of proton packs in a room. We've also proven that no matter the complexity of the room, a solution always exists that requires no more than $\frac{n}{3}$ Ghostbusters.

So, next time you're faced with a haunted house, you'll know exactly how many Ghostbusters to call – and it's probably fewer than you think!

References:

All diagrams own work

Chesnokov, N. (2018, May 16). *The art gallery problem: An overview and extension to chromatic coloring and mobile guards*. Massachusetts Institute of Technology. https://math.mit.edu/~apost/courses/18.204_2018/Nicole_Chesnokov_paper.pdf

Polygon triangulation. Wikipedia, The Free Encyclopedia. Last modified 29 July 2024 at 13:45 UTC. Retrieved 6 April 2025 from https://en.wikipedia.org/w/index.php?title=Polygon_triangulation&oldid=1237377295. Page Version ID: 1237377295

Wikipedia contributors. (2025, April 2). Proton pack (image). Wikipedia. https://en.wikipedia.org/wiki/Proton_pack

de Berg, M., Cheong, O., van Kreveld, M., & Overmars, M. (2008). *Computational Geometry: Algorithms and Applications* (3rd ed.). Berlin, Heidelberg: Springer-Verlag.
ISBN: 978-3-540-77973-5 (Print)

https://cimec.org.ar/foswiki/pub/Main/Cimec/GeometriaComputacional/DeBerg_-_Computational_Geometry_-_Algorithms_and_Applications_2e.pdf