

# QuickHull algorithm

The QuickHull algorithm is an algorithm for computing the [convex hull](#) of a set of points in two or more dimensions.

## Two dimensions

This is a variation on the [incremental algorithm](#) where we associate each unprocessed point with an edge that it can see. If the point can't see any edges it can be thrown away.

E  
X  
P  
A  
N  
D



At each step we select an edge (shown in light blue), find the associated point that is furthest away (point A) and add it to the hull. Then we delete the the selected edge.

E  
X  
P  
A  
N  
D



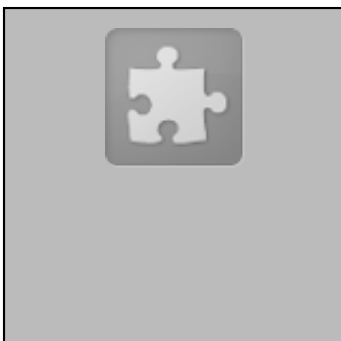
Add an edge to connect the new point to the remainder of the old hull. The points associated with the deleted edge are reassigned to the new edge if they can see it. (Point B in this case.)

E  
X  
P  
A  
N  
D



Add the second edge to finish connecting the new point to the remainder of the old hull. The remaining points associated with the deleted edge are reassigned to this new edge if they can see it. (In this case there are no such points.) Points that can't see any of the new edges are inside the hull and can be discarded. (In this case, just point C).

E  
X  
P  
A  
N  
D



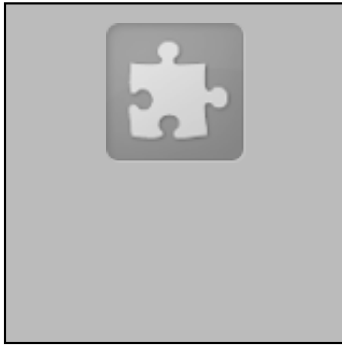
Select a new edge and repeat.

## Three dimensions

This is a variation on the incremental algorithm where we associate each unprocessed point with a triangle that it can see. If the point can't see any triangles it can be thrown away.

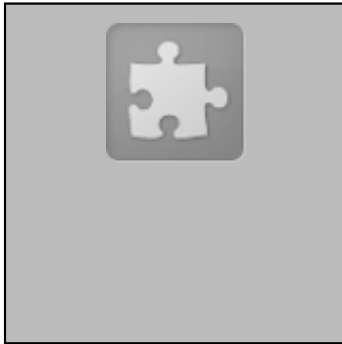
You can drag with the left mouse button down to look at any of the images below from a different direction.

E  
X  
P  
A  
N  
D



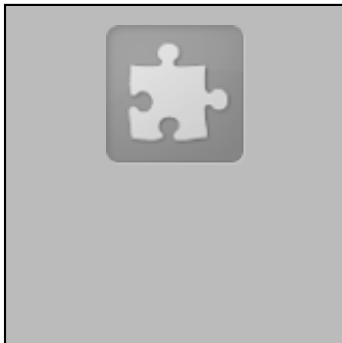
At each step we select a triangle (shown in light blue), find the associated point that is furthest away (coloured differently) and add it to the hull. Then we delete the the selected triangle and any other triangles (shown in yellow) that can see this point.

E  
X  
P  
A  
N  
D



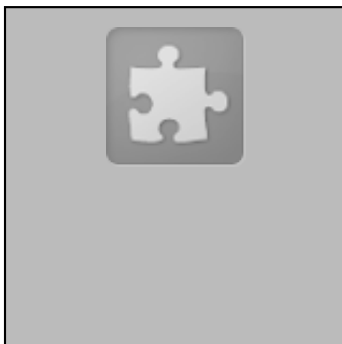
Deleting the triangles makes a hole in the convex hull. Add a triangle to connect the new point to an edge of the hole. The points associated with the deleted triangles are reassigned to the new triangle if they can see it.

E  
X  
P  
A  
N  
D



Repeat for each edge of the hole to finish connecting the new point to the remainder of the old hull. The points associated with the deleted triangles are reassigned to the first new triangle they can see. Points that can't see any of the new triangle are inside the hull and can be discarded.

E  
X  
P  
A  
N  
D



Select a new triangle and repeat.

In the worst case this algorithm is  $O(n^2)$ , but in practice it is no worse than  $O(n \log n)$ .

## More dimensions

We can generalise this to any number of dimensions.

[lambert@cse.unsw.edu.au](mailto:lambert@cse.unsw.edu.au)

Last modified: Thu Sep 24 18:30:59 AET 1998