

Union of convex polygons algorithm

This algorithm takes a set of overlapping convex polygons and then returns the union of all of these polygons. It is based on the idea that a concave polygon can be separated into convex sub-polygons. Reversing that process creates a realistic polygon that is usually concave.

Pseudo algorithm

Polygons created by this algorithm will look too regular, since all polygons lay on top of each other. By adding more and more points the result will lead to a circle-like shape. (Assuming the generated random points are constructed uniformly on a disk)

```
Polygon create_random_polygon (int max_number_of_points, int number_of_polygons) {
    List convex_polygons;
    for (int i = 0; i < number_of_polygons; ++i) {
        convex_polygons.push_back(create_random_convex_polygon(max_number_of_points));
    }
    return union_of(convex_polygons);
}
```

```
Polygon create_random_convex_polygon (int max_number_of_points) {
    List vertices = create_random_points(max_number_of_points);
    Polygon polygon = create_convex_hull(vertices);
    return polygon;
}
```

To avoid this problem, after generating a polygon it will be shifted. The shift is based on the position of the previous polygon and an additional vector with a random direction and a fixed length. Therefore, a new parameter needs to be introduced, that is the fixed length of the vector. If the length is smaller the resulting polygon will be more round. The convex polygons will overlap a lot. Increasing the distance that polygons are shifted will lead to bigger gaps. When the union is constructed it is necessary to deal with polygons or sets of polygons that are completely disjunct from other sets of polygons. If the length of the vector that shifts each polygon is bigger, the risk of polygons being disjunct from the rest increases as well.

Pseudocode

Parameters