Notes on Unified-EPA (UEPA)

T. Linneweber, June 2021

Unified-EPA combines the GJK (Gilbert–Johnson–Keerthi) algorithm and the EPA (Expanding polytope algorithm). The algorithm operates on the Minkowski sum (difference) of two convex shapes. For non-intersecting shapes it returns the closest distance. For intersecting shapes it returns the penetration depth which is defined by the minimal displacement needed to separate these shapes. It also returns two corresponding points on the surface of the shapes, as well as the displacement vector, compare Figure 1.

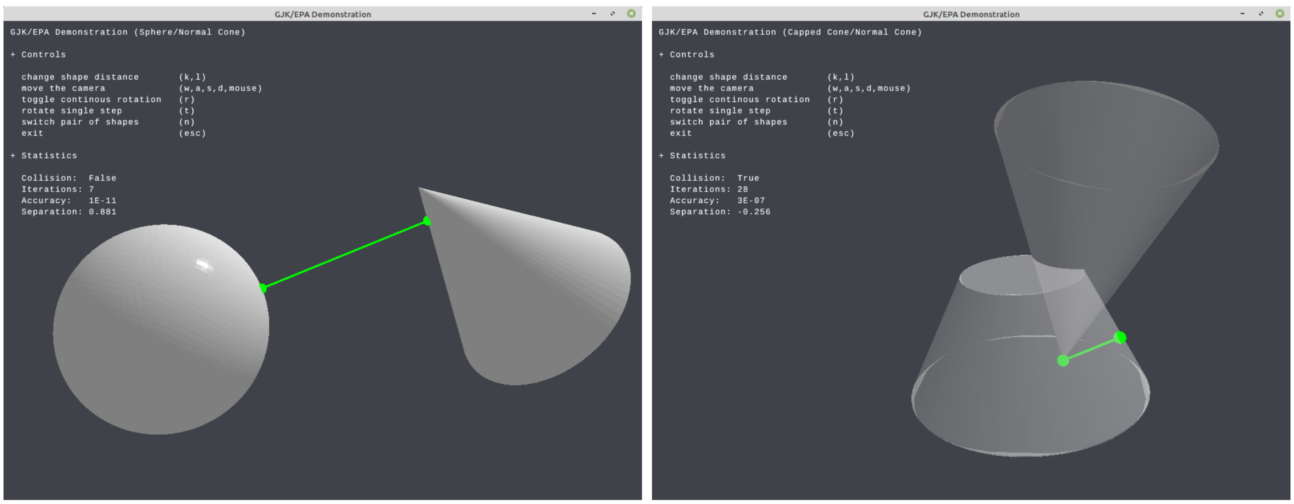

Figure 1: Demo implementation of the unified-EPA algorithm. Left: Separated shapes. Right: Intersecting shapes. Displacement vector shown as green line.

The collision information can be derived from the knowledge of the relative position of the origin to the Minkowski difference (Md) of the two shapes: The shapes are intersecting if the origin is contained within the Md, otherwise they are non-intersecting. For both cases the separation (and the displacement vector) can be determined by finding the smallest distance of the origin to the surface of the Md.

The Md is implicitly defined through a support function. The algorithm described on the next page constructs a polytope within the Md to extract this information.

Demo implementation

https://github.com/notgiven688/unified_epa

Outlined below are the steps of UEPA for an example case in two dimensions (see Figure 2):

(a) The Md is outlined in black. In this example the origin (red cross) lies within the Md. Blue cross: point known to be deep within the Md. A small initial triangle (the starting polytope) is created around this point. It is guaranteed that the triangle (tetrahedron in 3d) lies within the Md.

(b) The point of closest distance (green cross) from the polytope to the origin is found. The support point (green circle) of the Md is found by searching in the negative direction of the position vector of the closest point, i.e. the search direction (green arrow) points from the green cross to the red cross.

(c) The polytope (outlined in blue) was expanded by adding the newly found support point (green circle). Note that the expansion is done in such a way that the resulting polytope is convex. In the example here, two edges of the initial triangle got replaced.

(d) Step (b) is repeated.

(e) Step (c) is repeated. Note that from now on the origin is enclosed by the polytope.

(f) Step (b) is repeated with a twist: The search direction is reversed, i.e. the green direction vector points from the red cross to the green cross. This is done when the origin is already enclosed by the polytope.

(g) Step (c) is repeated.

(h) Step (b) is repeated. However the found support point is already known and we can not further expand the closest edge (light green line) of the polytope. This edge must be the closest to the origin. We terminate the algorithm here.

Additional notes:

- A point deep within the Md can be found by subtracting the center of mass of both shapes.
- Additional numerical stability of the algorithm originates from the finite volume of the initial triangle (tetrahedron). The size of the initial polytope is an upper bound of how small the Md of shapes can be.
- If the origin is outside the Md the algorithm resembles GJK.
- Once the origin is enclosed by the polytope the algorithm resembles EPA.
- Once the origin is enclosed it is sufficient to rely on line-point (plane-point in 3d) calculations to determine the closest point of the polytope to the origin. Otherwise segment-point (triangle-point in 3d) calculations are needed.

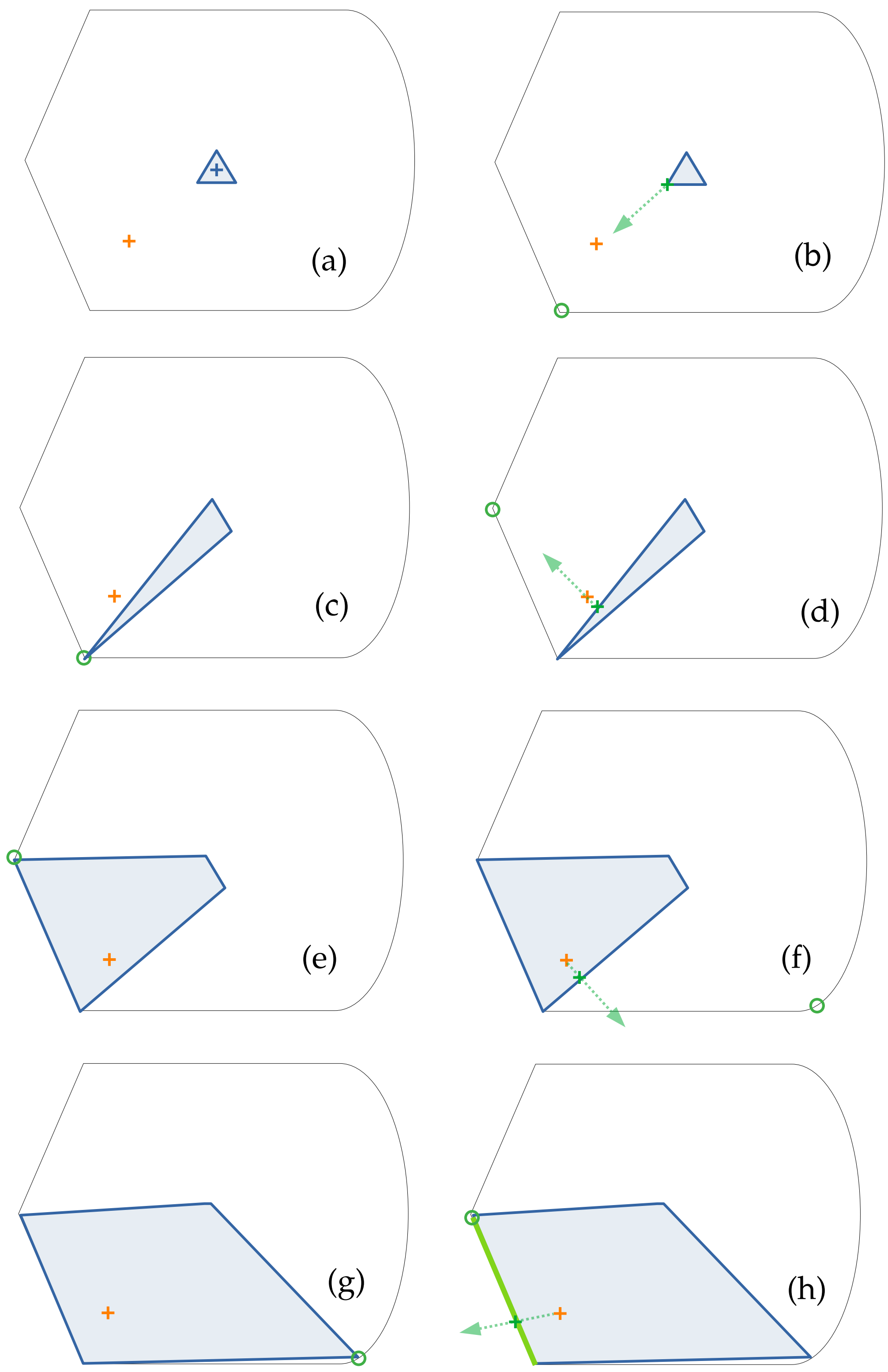

Figure 2: Steps of unified EPA. In this example the algorithm converges after four calls to the support function.