6491-2017 P4: Minkowski Morph (MM) of SPCCs

1 Goal

You are given two convex shapes A and B, each bounded by a smooth, piecewise circular curve (SPCC). You may (but do not have to) use your previous projects to construct such shapes, but your sketch must offer some interactive editing tool for designing A and B via click and drag. When the user presses 'a', your software should start an animation of a Minkowski Morph (MM) from A to B that lasts about two seconds.

2 Deliverables

This is an **individual** project. It is due on Oct 31.

Each person must submit the following on T-Square:

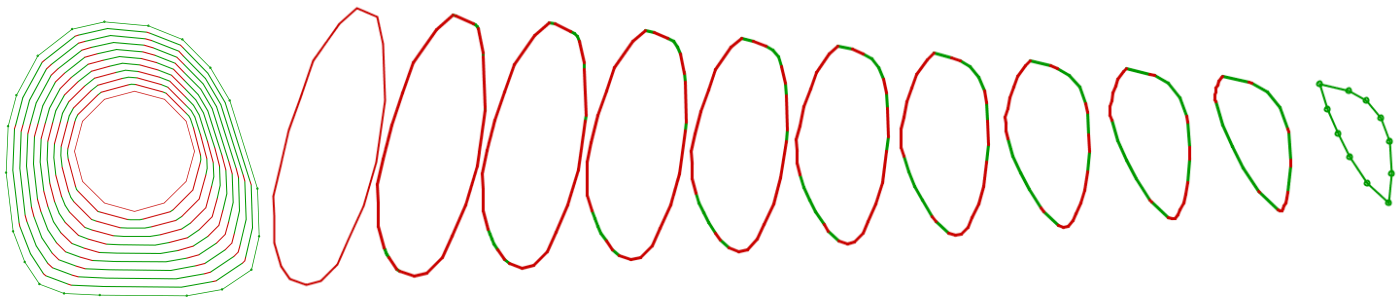
Your complete sketch with a clear indication where the code that you implemented for this project is. Preferably create a class PCC in a separate tab and create a Minkowski Morph function MM(A,t,B) that takes two PCCs (A and B) and time t and that produces a new PCC. At each frame of the animation, or of the stroboscopic display, your algorithm must compute the proper SPCC model of the result—not a polygonal approximation!

A short video of you using your applet that shows you editing A and B several times and morphing between them after each edit.

A short write-up where you provide:

a) full titles
b) your name and picture
c) short problem specification
d) the details of your representation of PCCs (data structure and semantics)
e) a short overview (in your own words) of Minkowski Morph and references to the best of prior art ;-)
f) a clear overview of your algorithm
g) a convincing justification of its correectness
h) a clear explanation of an important, non-trivial details of your algorithm
i) images of several morph animations using stroboscopic superposition of intermediate frames
j) description and explanation of extra credit extensions implemented

For example, the images below show frames from such Minkowski morphs, but for polygonal (rather than SPCC shapes). Produce similar images for SPCCs.



3 Possible extensions (extra credit)

A) Extension to non-convex SPCCs (including proper trimming of portions that are not on the boundary of the result.

B) Combine with a user-provided carrier (user picks two points on bA and two corresponding points on bB and you compute the carrier as a log spiral motion between these pairs) and the MM of the residue difference.

B+) Automatic computation of the above carrier (using centroid, axis of inertia, and area).