Computing the intersections of the plane parametric curves

We are given parametrizations of two curves, K and L, in the plane $\mathbb{R}^2$. Our task is to find all points of intersection of K and L. Let $\mathbf{p}(t)$ and $\mathbf{q}(t)$ be the parametrizations of K and L defined on intervals $I = [a, b]$ and $J = [c, d]$, respectively. Find the points of intersection using the following procedure:

1. Divide the intervals I and J into subintervals of length $h > 0$, where h is sufficiently small.

2. Approximate K and L with polygonal chains determined by evaluating the parametrizations at subdivision points of I and J and find the intersections of the polygonal chains.

3. Use the points of intersection of the polygonal approximations as an initial guess for Newton's iteration, which will determine the actual intersection points of K and L more accurately.

For the Newton iteration you will need also the parametrizations of $\dot{\mathbf{p}}$ and $\dot{\mathbf{q}}$.

Task

Write an Octave/Matlab function, which for given curves K and L determines all their intersection points following the procedure above. Draw two figures: on the first figure plot the polygonal chains K', L' and clearly mark the intersection points, while on the second do the same for the curves K, L.

Detailed instructions for the Octave/Matlab function

Write an Octave/Matlab function `intersectionCurves`, which computes the intersection points P of the curves K, L with given parametrizations. Function call should be of the form:

```
[P,Q] = intersectionCurves(p, pdot, intp, q, qdot, intq, h),
```

where:

- p (q) is a function handle, describing the first (second) plane curve,

- pdot (qdot) is a function handle, describing the derivative of the first (second) plane curve,

1

- `intp` (`intq`) is an interval, on which the first (second) curve is parametrized,

- `h` is the length of the subintervals dividing the intervals `intp` and `intq` (the last subinterval can be shorter),

- `Q` is the list of the intersection points of K' and L'.

The function should also plot both curves K, L on one figure, both polygonal chains K', L' on the second figure, and on both figures also mark the points in the lists P and Q.

Example

For the curves K and L parametrized by

$$
t \mapsto \begin{bmatrix} t \\ \cos t \end{bmatrix} \qquad \text{in} \qquad u \mapsto \begin{bmatrix} u \\ \sin u \end{bmatrix}
$$

on the intervals $[-2\pi, 2\pi]$ and a step $h = 0.1$, we would use

```
p = @(t) [t;cos(t)];
pdot=@(t) [1;-sin(t)];
intp=[-2*pi,2*pi];
q = @(u) [u;sin(u)];
qdot=@(u) [1;cos(u)];
intq=[-2*pi,2*pi];
h=0.1
[P,Q] = intersectionCurves(p,pdot,intp,q,qdot,intq,h)
```

Tests

At the end of the file with a function `intersectionCurves` add test cases, which check correctness of your programme. Tests should be made for a few pairs of curves (K, L).

Submission

Use the online classroom to submit the following:

1. file **intersectionCurves.m** which should be well-commented and contain at least one test,

2. a report file **solution.pdf** which contains the necessary derivations and answers to questions.

While you can discuss solutions of the problems with your colleagues, the programs and report must be your own work. You can use all Octave/Matlab functions from problem sessions/lectures.