

# Final Project for The Practice of Mathematics

due 2021 SEP 08, 9:50 a.m.

This final project weighs 100 points in total and serves as the safe-net option for those students who missed the final exam or are not satisfied with their performance in the final exam.

After grasping the theory of splines [2, Chapter 4] and Yin sets [1], you are supposed to write a software package that implements spline fitting of curves, efficient representation of Yin sets and their Boolean algebra. You may earn extra credits by submitting an answer to the assignments in Section 3.

## 1 Representing a cartoon

- (a) Find a black-and-white cartoon figure you like.
- (b) Pick characteristic points on the boundary and fit simple closed cubic splines through these points so that your simple closed splines approximate the original boundary curves very well. Save the coordinates of the spline knots in a single text file.
- (c) Write one or multiple C/C++ subroutines so that the driver program takes as input the text files in (b) and generates a figure for the Yin set, which is supposed to approximate the cartoon figure in (a).
- (d) Write a C/C++ subroutine so that it takes as input a number of pairwise almost disjoint simple closed splines and produces a poset (partially ordered set) of splines with the partial order as the inclusion of Jordan curves in [1, Definition 3.10].
- (e) Plot the Hasse diagram of the poset in (d).  
**Important:** Your Hasse diagram must have at least five levels of oriented Jordan curves, i.e. a chain of four covering relations. Hence the cartoon figure of your choice in (a) must have nontrivial structures.
- (f) Represent the Yin set  $\mathcal{Y}$  plotted in (c) as a realizable  $\text{spadjor } \mathcal{F}$ , i.e.  $\rho(\mathcal{F}) = \mathcal{Y}$ . Then expand the expression  $\rho(\mathcal{F})$  in terms of interiors of oriented Jordan curves.

See Figure 1 for an example.

The total point for this section is 70; the above six questions weigh 5, 10, 20, 20, 10, and 5 points, respectively.

Your C/C++ package must use a `make` system and must compile, run, and finish on `Ubuntu 20.04 LTS`; otherwise you get no points for this section. See [3] for an introduction of these tools.

Your Hasse diagram should have the same format as that of [1, Figure 9]. For your convenience, the  $\text{LATEX}$  source of that figure is attached below.

```
\documentclass[10pt]{article}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{pstricks,pst-eps}
\usepackage{tikz}
\pagestyle{empty}
\begin{document}
\begin{TeXtoEPS}

\begin{tikzpicture}
\node[circle,draw,fill=yellow](n4) at (0,0)
    {\gamma_4~+};
\node[circle,draw](n6) at (-2,-2) {\gamma_6~-};
\node[circle,draw](n5) at (0,-2) {\gamma_5~-};
\node[circle,draw](n7) at (2,-2) {\gamma_7~-};
\node[circle,draw,fill=yellow](n2) at (-1,-4)
    {\gamma_2~+};
\node[circle,draw,fill=yellow](n3) at (+1,-4)
    {\gamma_3~+};
\node[circle,draw,fill=yellow](n1) at (-4,0)
    {\gamma_1~+};
\draw[line width=2] (n4)--(n6);
\draw[line width=2] (n4)--(n5);
\draw[line width=2] (n4)--(n7);
\draw (n5)--(n2);
\draw (n5)--(n3);
\end{tikzpicture}

\end{TeXtoEPS}
\end{document}
```

## 2 Boolean algebra on 2D Yin sets

Write a C/C++ package to implement the Boolean algebra of Yin sets, whose boundaries are represented by linear polygons. Each of your subroutines or classes must be properly documented. Validate your package with cases as exhaustive as possible; the tests in [1, Figure 13] are mandatory.

The total point for this section is 30.

## 3 Boundary of 3D Yin sets

An  $m$ -manifold is a Hausdorff space  $X$  with a countable basis such that each point  $x \in X$  has a neighborhood that is homeomorphic with an open subset of  $\mathbb{R}^m$ . A 1-manifold is often called a curve and a 2-manifold is called a *surface*. The following well known theorem is a classification of orientable compact surfaces.

**Theorem 1.** Every orientable compact surface is homeomorphic either to a sphere or to a connected sum of tori.

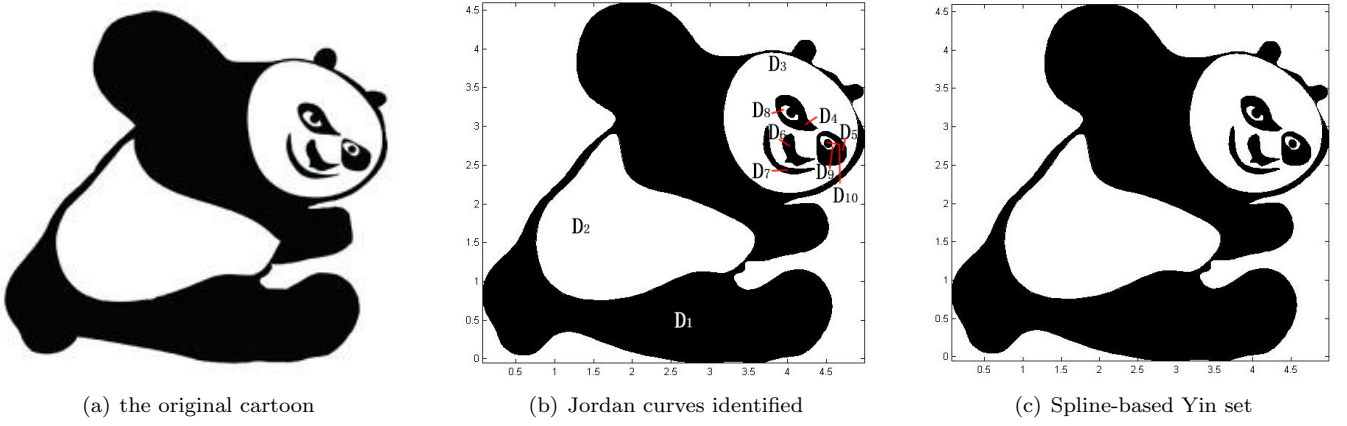


Figure 1: An example for assignments in Section 1. Each  $D_i$  is the bounded complement of the Jordan curve  $\gamma_i$ .



(a) An ox



(b) An jade Buddha

Figure 2: Examples of 3D Yin sets.

An orientable compact surface has the *positive orientation* if the outward norm vector of the surface points from the bounded complement of the surface to the unbounded complement; otherwise it has the negative orientation.

The following theorem is a recent result.

**Theorem 2.** For a nonempty Yin set  $\mathcal{Y} \subset \mathbb{R}^3$ , its boundary  $\partial\mathcal{Y}$  is homeomorphic to the gluing of a collection of orientable compact surfaces along subsets homeomorphic to one-dimensional CW complex.

Like 2D Yin sets, a 3D Yin set can be uniquely represented by its boundary; which is usually an oriented triangular grid that consists of a number of vertices, directed edges, and oriented triangles. In the field of computer graphics, the information of a 3D solid is usually stored in a similar data structure, and a popular format of `.obj` files is widely adopted online; see Figure 2 for two examples.

However, triangular grids in `.obj` files are often not homeomorphic to the boundary of any 3D Yin sets. In

this part, you are supposed to investigate common features of these `.obj` files, design algorithms to modify the triangular grids so that the output is homeomorphic to the boundary of a 3D Yin set and it is close to the boundary of the true 3D Yin set as much as possible.

You should submit a document describing your algorithms and a C/C++ package that implement and test your algorithms.

The total points is 50, all of which count as extra credit.

## References

- [1] Q. Zhang and Z. Li. Boolean algebra of two-dimensional continua with arbitrarily complex topology. *Math. Comput.*, 89: 2333-2364, 2020.
- [2] Q. Zhang. Notes on Numerical Analysis. *preprint*, 2021.
- [3] Q. Zhang. Jump-start Guide for new graduate students. [github.com/Qinghai-Zhang/JumpStartGuide](https://github.com/Qinghai-Zhang/JumpStartGuide) *Github project*, 2021.