Zep-CAD

Mini-CAD Software for Zeppelin Modeling
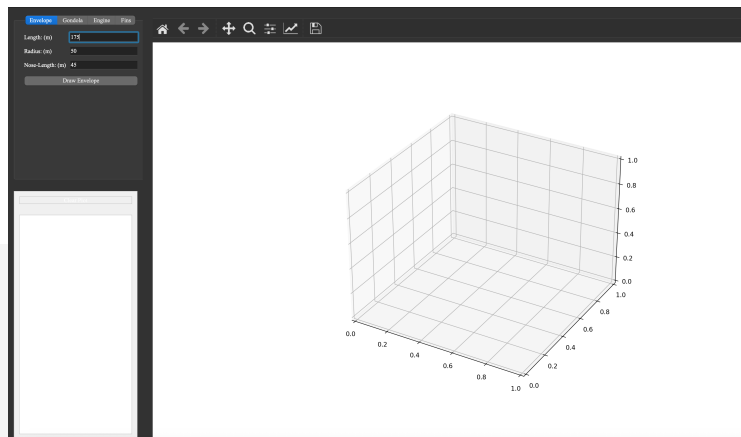
Evan Blosser, Matthew Dobbs, Blake Johnson

University of Oklahoma

AME 5193 - Spring 2025

April 29, 2025

Project Objective

- Develop lightweight Python-based CAD tool to support conceptual Zeppelin design.
- Enable parametric modeling and real-time surface regeneration.
- Provide user-friendly GUI interface using PySide6 and Matplotlib.

GUI Layout

Tabbed interface with Envelope, Gondola, Engine, and Fins control panels.

Code Structure


```python
import sys
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import CubicSpline

from PySide6 import QtCore, QtWidgets, QtGui
from PySide6.QtWidgets import (QApplication, QMainWindow, QVBoxLayout, QHBoxLayout,
                               QWidget, QLabel, QLineEdit, QPushButton, QTextEdit, QTabWidget,
                               QMenuBar, QComboBox, QScrollArea, QListView, QFileDialog,
                               QFileSystemModel,QGridLayout, QSlider, QMessageBox)
from PySide6.QtGui import QPainter, QPen, QColor, QBrush
from PySide6.QtCore import Qt, QRectF
from matplotlib.backends.backend_qtagg import FigureCanvasQTAgg as FigureCanvas
from matplotlib.backends.backend_qtagg import NavigationToolbar2QT as NavigationToolbar
from mpl_toolkits.mplot3d.art3d import Poly3DCollection

sys.dont_write_bytecode = True
from config import clear_plot, save_as_stl, merge_wireframes
from Envelope import draw_envelope
from Engine import draw_engine
from Gondola import draw_canoe_gondola
from Fins import build_fin, Ro_x


class PlotCanvas(FigureCanvas):
    def __init__(self, parent=None):
```
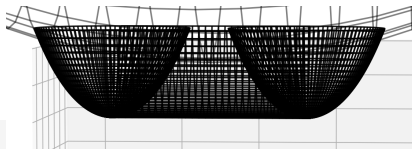
Modular design using main.py and independent geometry modules

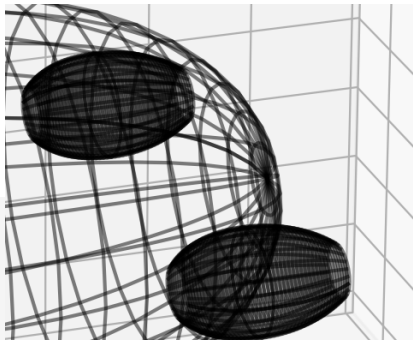# Curve and Surface Types
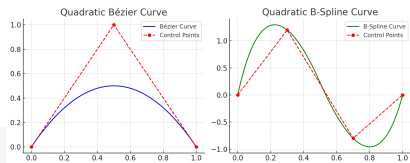
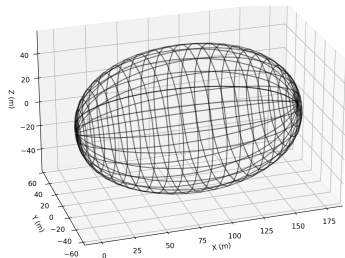Bézier, B-spline, cubic, and quadratic spline curves.

Revolved, ruled, and lofted surface examples.
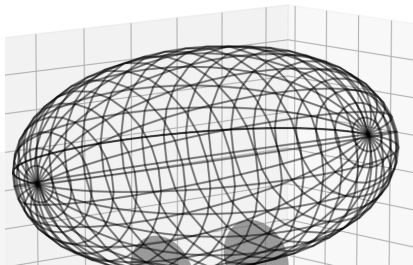
Transformations

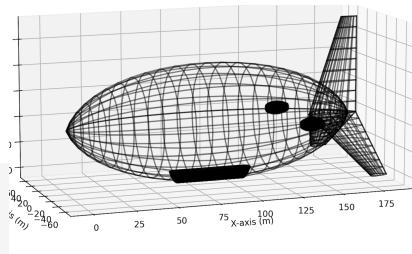Engine translated laterally for spacing.

Fins rotated around envelope axis.

Model Variations: Nose Profiles

Standard nose geometry.

Sharpened nose via control point adjustment.

Planned Enhancements

- Implement gondola-envelope intersection trimming.
- Add STL export capabilities for 3D printing.
- Introduce shaded rendering and interactive rotation.

Demonstration Video

(Time for a Demonstration Video)

Lessons Learned

- Integration of symbolic and numerical math for CAD.
- Importance of modular GUI architecture in engineering tools.
- Applying CAD concepts like lofting and revolving programmatically.

Self-Assessment

- ▶ Delivered a complete modular CAD prototype.
- ▶ Achieved real-time visual updates and user interactivity.
- ▶ Set groundwork for advanced features like trimming and export.

Thank You!

Questions?