Computational Geometry: Project

Optimal Triangulation using Dynamic Programming

BY :

- Sagar Jeevan - 111861945

AGENDA

- Motivation
- Optimization Techniques
- Implementation Illustrations
- Comparison of Techniques
- Demo
- Technologies Used
- Enhancements and Conclusion

MOTIVATION

- Real world applications.
- To understand the extent of optimization achieved.
- To explore the domain further.

OPTIMIZATION TECHNIQUES

- Minimum Weight Triangulation
- Minimizing the Maximum of the Internal Angles
- Minimizing the Longest Edge

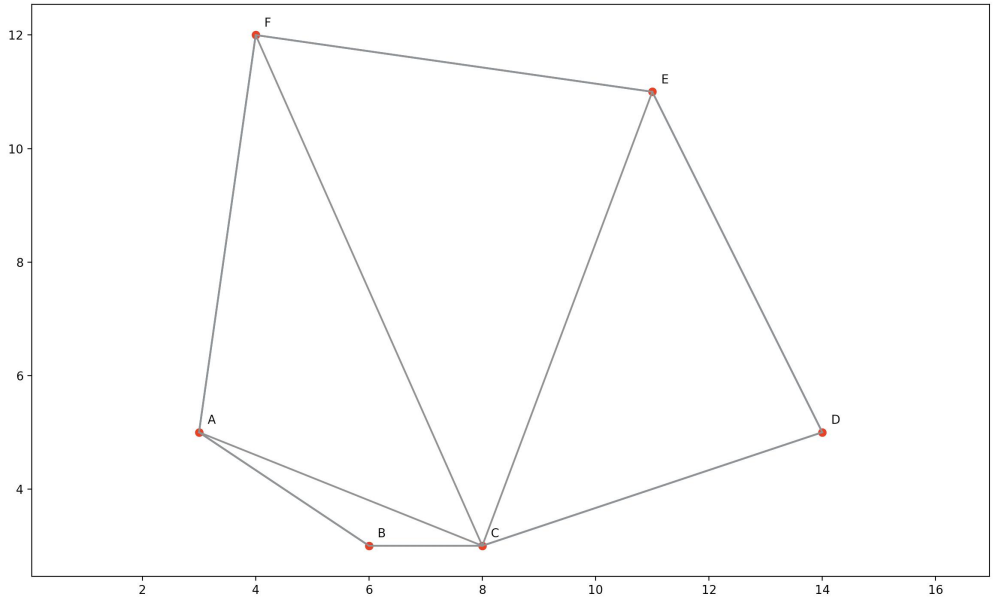
IMPLEMENTATION ILLUSTRATIONS

- *Minimum Weight Triangulation*

  A = (03,05), B = (06,03), C = (08,03)

  D = (14,05), E = (11,11), F = (04,12)

  Total Weight: 80.35 Units



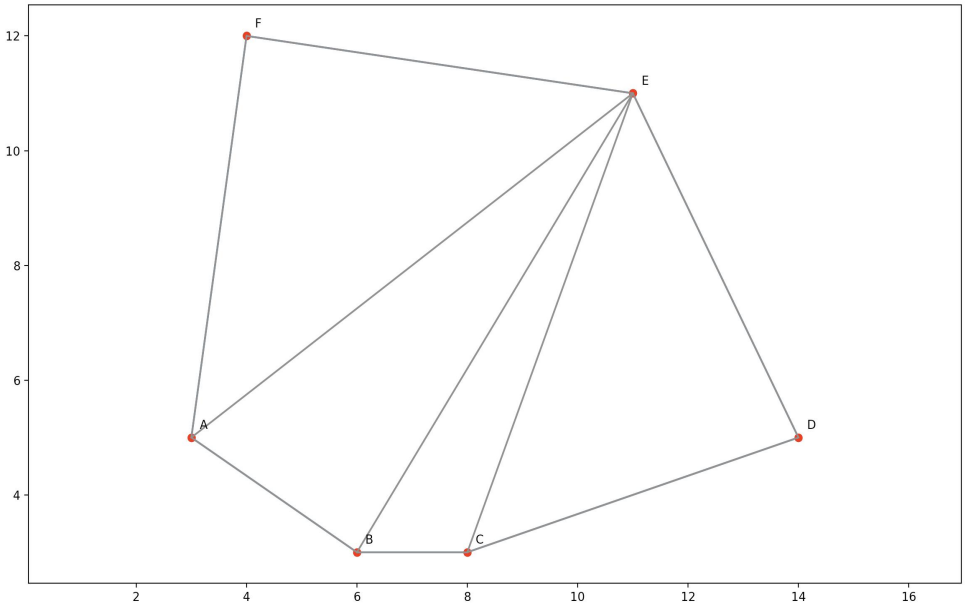
IMPLEMENTATION ILLUSTRATIONS

- *Minimizing the Maximum of the Internal Angles*

A = (03,05), B = (06,03), C = (08,03)

D = (14,05), E = (11,11), F = (04,12)

Maximum Angle (BCE): 110.55 degrees



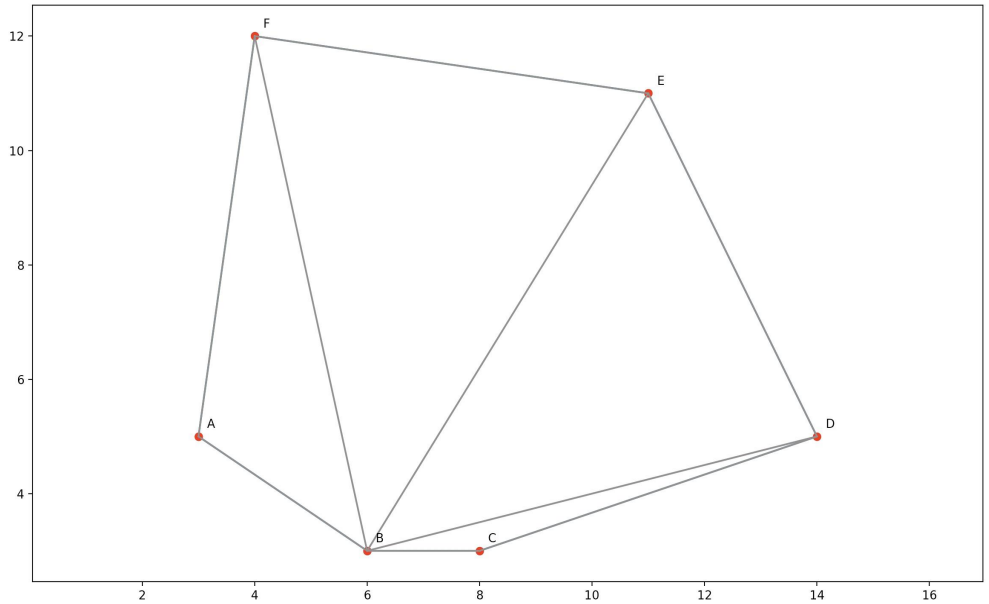
IMPLEMENTATION ILLUSTRATIONS

- *Minimizing the Longest Edge*

A = (03,05), B = (06,03), C = (08,03)

D = (14,05), E = (11,11), F = (04,12)

Longest Edge (BF) = 9.43 Units



COMPARISON BETWEEN TECHNIQUES

	Total Weight	*Maximum of the Internal Angles*	*Longest Edge*
Minimum Weight Triangulation	***80.35***	*146.309/409.425*	*9.848*
Minimizing The Maximum of the Internal Angles	*88.71*	***110.55/370.74***	*10*
Minimizing the Longest Edge	*86.58*	*161.56/423.92*	***9.434***

DEMO

- Input - A text file with two lines
  - First line - Vertices of a polygon
  - Second line - Clockwise listing of x,y coordinates
- Output - A triangulated polygon

TECHNOLOGIES, LIBRARIES

- Python 2.7
- Matplotlib
- TkInter

ENHANCEMENTS AND CONCLUSION

- Implementations on non-convex polygons (using left tests and intersections)
- Comparison of results with other triangulation techniques
- Other similar optimization techniques