A Hexagon Cutting Problem and its Generalisations

Joel Tan, Matthew Chia, Daniel Leong
NUS High School of Mathematics and Science

February 19, 2017

1 Introduction

We explore a geometrical problem as follows: Start with a regular n-gon, $n \geq 6, n \in \mathbb{N}$. In an *operation*, choose two adjacent edges of the current polygon. The line joining the midpoints of these two edges divides the polygon into a triangle and an $(n+1)$-gon. We remove the triangle. After some operations, we wish to determine the smallest possible fraction of the original area left.

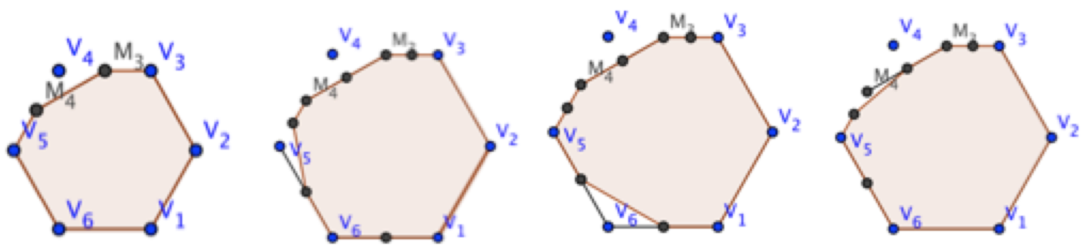


Figure 1: The first diagram shows possible polygons after an operation (in brown); the next three show some of the possible polygons after two operations (also in brown).

2 Motivation and Aim of Research

This problem originated from Problem 4 of the USA Mathematics Olympiad (USAMO) 1997, where the question requires one to prove the area left is at least $\frac{1}{3}$. In this project, we found a better lower bound for the area left starting with a regular hexagon, and extend the results to polygons of more sides in general.

3 Methodology and Results

We created a C++ program that does operations in two ways: randomly, and greedily (the best possible operation at each step). From this, we found that there exists a sequence of operations giving less than 0.80216 of the area, and that Greedy algorithm works best most of the time.

3.1 Geometric Methods

By showing each edge of the hexagon cannot be completely removed, we obtained a bound of $\frac{1}{2}$ of the original area. Devising a system to classify the operations allowed us to bound the final polygon outside a fixed dodecagon as shown in Figure 2, which is $\frac{49}{85}$ of the original area.

3.2 Algebraic Methods

We introduced the idea of *subareas*: In a polygon $P_1P_2...P_n$ (with indices taken modulo n), the subareas S_i are areas of the triangles in the form $\Delta P_iP_{i+1}P_{i+2}$.

We then established the *Polygon Array* of an n-gon (which loops around):

$[-log_2(S_1)+2, -log_2(S_2)+2, ..., -log_2(S_n)+2]$.

We proved an operation is equivalent to replacing some part of this array $[a, b, c]$ with $[a+1, b+2, b+2, c+1]$ and subtracting $\frac{1}{2^b}$ from the area (e.g. $[0,0,0,0,0,0] \rightarrow [0,0,1,2,2,1]$)

We then managed to prove, with much effort, that the remaining area is always at least $\frac{3}{4}$ of the initial area of the hexagon for any finite number of operations.

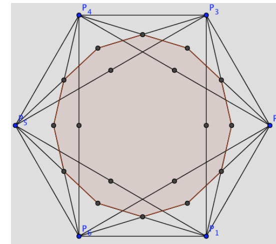
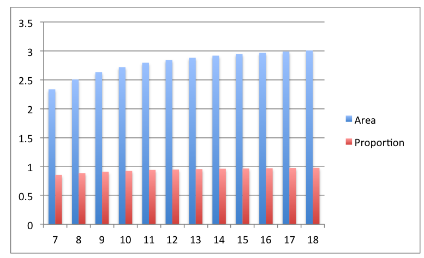


Figure 2: Dodecagon

3.3 Computer Approximations

The C++ program allowed us to approximate the lower bounds for polygons with more sides. The results are shown in the following graph and table: The graph shows both the area (in an n-gon inscribed in a unit circle) and the proportion (expressed as a decimal less than 1.)



Number of Sides	Random	Greedy	Area of Initial Polygon	Minimum Proportion
7	2.34039	2.33738	2.73641	0.85417
8	2.50551	2.501	2.82847	0.88422
9	2.63194	2.62626	2.89254	0.90794
10	2.7192	2.72527	2.93892	0.92523