Counting Triangulations of a Convex Polygon

Desh Ranjan *†

Introduction

In a 1751 letter to Christian Goldbach (1690–1764), Leonhard Euler (1707–1783) discusses the problem of counting the number of triangulations of a convex polygon. Euler, one of the most prolific mathematicians of all times, and Goldbach, who was a Professor of Mathematics and historian at St. Petersburg and later served as a tutor for Tsar Peter II, carried out extensive correspondence, mostly on mathematical matters. In his letter, Euler provides a "guessed" method for computing the number of triangulations of a polygon that has n sides but does not provide a proof of his method. The method, if correct, leads to a formula for calculating the number of triangulations of an n-sided polygon which can be used to quickly calculate this number [1, p. 339-350] [2]. Later, Euler communicated this problem to the Hungarian mathematician Jan Andrej Segner (1704–1777). Segner, who spent most of his professional career in Germany (under the German name Johann Andreas von Segner), was the first Professor of Mathematics at the University of Göttingen, becoming the chair in 1735. Segner "solved" the problem by providing a proven correct method for computing the number of triangulations of a convex n-sided polygon using the number of triangulations for polygons with fewer than n sides [5]. However, this method did not establish the validity (or invalidity) of Euler's guessed method. Segner communicated his result to Euler in 1756 and in his communication he also calculated the number of triangulations for the n-sided polygons for $n = 1 \ldots 20$ [5]. Interestingly enough, he made simple arithmetical errors in calculating the number of triangulations for polygons with 15 and 20 sides. Euler corrected these mistakes and also calculated the number of triangulations for polygons with up to 25 sides. It turns out that with the corrections, Euler's guessed method gives the right number triangulations of polygons with up to 25 sides.

Was Euler's guessed method correct? It looked like it was but there was no proof. The problem was posed as an open challenge to the mathematicians by Joseph Liouville (1809–1882) in the late 1830's. He received solutions or purported solutions to the problem by many mathematicians (including one by Belgian mathematician Catalan which was correct but not so elegant), some of which were later published in the Liouville journal, one of the primary journals of mathematics at that time and for many decades. The most elegant of these solutions was communicated to him in a paper by Gabriel Lamé (1795-1870) in 1838. French mathematician, engineer and physicist Lamé was educated at the prestigious Ecole Polytéchnique and later at the Ecole des Mines[3, p. 601–602]. From 1832 to 1844 he served as the chair of physics at the Ecole Polytéchnique, and in 1843 joined the Paris Academy of Sciences in the geometry section. He contributed to the fields of differential geometry, number theory, thermodynamics and applied mathematics. Among his publications are textbooks in physics and papers on heat transfer, where he introduced the rather useful technique of curvilinear coordinates. In 1851 he was appointed Professor of Mathematical

*Department of Computer Science; New Mexico State University; Las Cruces, NM 88003; dranjan@cs.nmsu.edu.
†With thanks to David Pengelley, Inna Pivkina and Karen Villaverde.

Physics and Probability at the University of Paris, and resigned eleven years later after becoming deaf. Gauss considered Lamé the foremost French mathematician of his day [3, p. 601–602].

A translated version of this paper by Lamé is the key historical source for this project. This translation is presented in its entirety before the description of the major project tasks in the next section. Interestingly and perhaps somewhat ironically, today these numbers (the number of triangulations of an n sided polygon for $n = 1, 2, 3\ldots$)) are called *Catalan numbers*. Although none of Segner, Euler or Catalan provided any insight into why they were interested in calculating the number of triangulations of convex polygons. However, the related problem of computing optimal polygon triangulations has become a well-studied problem in the realm of algorithm design and is fairly commonly used to illustrate the effectiveness of dynamic programming paradigm for designing efficient algorithms.

Understanding Triangulations

A *diagonal* in a (convex) polygon is a straight line that connects two non-adjacent vertices of the polygon. Two diagonals are different if they have at least one different endpoint. A *triangulation* of a polygon is a division of the polygon into triangles by drawing *non-intersecting* diagonals. For example, the 6-sided polygon $ABCDEF$ below is triangulated into 4 triangles by using the diagonals AD, AE, BD.

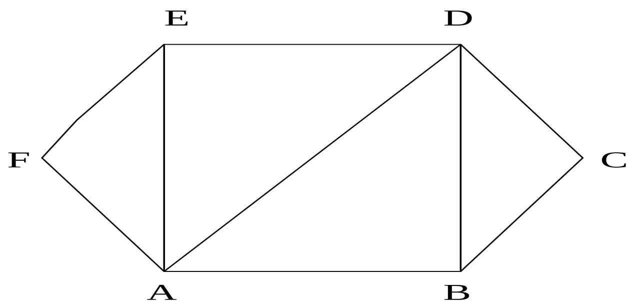

Figure 1: A Triangulation of $ABCDEF$

Two triangulations are different if at least one of the diagonals in a triangulation is different from all diagonals in the other triangulation.

TASKS:

1.1 Draw a triangulation of $ABCDEF$ that is different from the triangulation in Figure 1. How many diagonals does your triangulation have? How many triangles does it divide $ABCDEF$ into?

1.2 Consider an n-sided polygon $A_1A_2\ldots A_n$. How many different possible diagonals does this polygon have? **Note:** We are talking about all possible diagonals, not just diagonals in a triangulation.

1.3 Use mathematical induction to prove that any triangulation of an n sided polygon has $n-2$ triangles and $n-3$ diagonals.

Let's now read from Lamé's letter to Liouville [4].

Excerpt from a letter of Monsieur Lamé to Monsieur Liouville on the question: Given a convex polygon, in how many ways can one partition it into triangles by means of diagonals?[1]

The formula that you communicated to me yesterday is easily deduced from the comparison of two methods leading to the same goal.

Indeed, with the help of two different methods, one can evaluate the number of decompositions of a polygon into triangles: by consideration of the sides, or of the vertices.

I.

Let $ABCDEF\ldots$ be a convex polygon of $n+1$ sides, and denote by the symbol P_k the total number of decompositions of a polygon of k sides into triangles. An arbitrary side AB of $ABCDEF\ldots$ serves as the base of a triangle, in each of the P_{n+1} decompositions of the polygon, and the triangle will have its vertex at C, or D, or $F\ldots$; to the triangle CBA there will correspond P_n different decompositions; to DBA another group of decompositions, represented by the product P_3P_{n-1}; to EBA the group P_4P_{n-2}; to FBA, P_5P_{n-3}; and so forth, until the triangle ZAB, which will belong to a final group P_n. Now, all these groups are completely distinct: their sum therefore gives P_{n+1}. Thus one has

$$P_{n+1} = P_n + P_3P_{n-1} + P_4P_{n-2} + P_5P_{n-3} + \cdots + P_{n-3}P_5 + P_{n-2}P_4 + P_{n-1}P_3 + P_n. \quad (1)$$

II.

Let $abcde\ldots$ be a polygon of n sides. To each of the $n-3$ diagonals, which end at one of the vertices a, there will correspond a group of decompositions, for which this diagonal will serve as the side of two adjacent triangles: to the first diagonal ac corresponds the group P_3P_{n-1}; to the second ad corresponds P_4P_{n-2}; to the third ae, P_5P_{n-3}, and so forth until the last ax, which will occur in the group P_3P_{n-1}. These groups are not totally different, because it is easy to see that some of the partial decompositions, belonging to one of them, is also found in the preceding ones. Moreover they do not include the partial decompositions of P_n in which none of the diagonals ending in a occurs.

But if one does the same for each of the other vertices of the polygon, and combines all the sums of the groups of these vertices, by their total sum

$$n\,(P_3P_{n-1} + P_4P_{n-2} + \cdots + P_{n-2}P_4 + P_{n-1}P_3)$$

one will be certain to include all the partial decompositions of P_n; each of these is itself repeated therein a certain number of times.

[1] See a Memoir of Segner (*Novi Commentarii Acad. Petrop.*, vol. VII, p. 203). The author found equation (1) of M. Lamé; but formula (3) presents a much simpler solution. Formula (3) is no doubt due to Euler. It is pointed out without proof on page 14 of the volume cited above. The equivalence of equations (1) and (3) is not easy to establish. M. Terquem proposed this problem to me, achieving it with the help of some properties of factorials. I then communicated it to various geometers: none of them solved it; M. Lamé has been very successful: I am unaware of whether others before him have obtained such an elegant solution. J. Liouville

Indeed, if one imagines an arbitrary such decomposition, it contains $n-2$ triangles, having altogether $3n-6$ sides; if one removes from this number the n sides of the polygon, and takes half of the remainder, which is $n-3$, one will have the number of diagonals appearing in the given decomposition. Now, it is clear that this partial decomposition is repeated, in the preceding total sum, as many times as these $n-3$ diagonals have ends, that is $2n-6$ times: since each end is a vertex of the polygon, and in evaluating the groups of this vertex, the diagonal furnished a group including the particular partial decomposition under consideration.

Thus, since each of the partial decompositions of the total group P_n is repeated $2n-6$ times in $n\,(P_3P_{n-1}+P_4P_{n-2}+\cdots+P_{n-2}P_4+P_{n-1}P_3)$, one obtains P_n upon dividing this sum by $2n-6$. Therefore one has

$$P_n = \frac{n\,(P_3P_{n-1}+P_4P_{n-2}+\cdots+P_{n-2}P_4+P_{n-1}P_3)}{2n-6}. \qquad (2)$$

III.

The first formula (1) gives

$$P_3P_{n-1}+P_4P_{n-2}+\cdots+P_{n-2}P_4+P_{n-1}P_3 = P_{n+1}-2P_n,$$

and the second (2) gives

$$P_3P_{n-1}+P_4P_{n-2}+\cdots+P_{n-2}P_4+P_{n-1}P_3 = \frac{2n-6}{n}P_n;$$

so finally

$$P_{n+1}-2P_n = \frac{2n-6}{n}P_n,$$

or

$$P_{n+1} = \frac{4n-6}{n}P_n. \qquad (3)$$

This is what was to be proven.

Paris, 25 August, 1838

Optimal Triangulation and Counting Triangulations

The *Optimal Polygon Triangulation* Problem is the following: Given an n-sided polygon $A_1A_2\ldots A_n$ and a weight $w_{i,j}$ for each diagonal A_iA_j, find a triangulation of the polygon such that the sum of the weights of the diagonals in the triangulations is minimized. A naïve way to solve the problem is to generate all possible triangulations one by one, calculate their weight (i.e. sum of weights of all the diagonals in the triangulation) and keep the best. The efficiency of this naïve method depends on the number of possible triangulations of a polygon with n sides. Thus, we would like to count how many different triangulations an n-sided polygon has. As mentioned in the introduction, the problem of counting the number of triangulations of an n-sided convex polygon was already being discussed in the mid eighteenth century by well-known figures in mathematics like Euler and Segner and an elegant solution was provided by Lamé in 1838. As noted before, the numbers P_n in Lamé's paper are now called Catalan numbers.

TASKS:

2.1 Read Section I in Lamé's paper. Explain what Lamé is saying in your own words and derive the general recursive formula for P_{n+1}, i.e., formula (1) in Lamé's paper.

2.2 Use the recursive formula to calculate P_i for $i = 2, 3, 4, 5, 6, 7, 8$ by hand and display it as a table.

2.3 Draw all triangulations of polygons with n sides for $n = 4, 5$.

2.4 Lamé's recurrence relation in his section 1 for $n = 5$ yields

$P_6 = P_5 + P_3P_4 + P_4P_3 + P_5.$

Draw all triangulations of a 6-sided polygon classified into groups according to the idea of the recurrence relation, i.e., the triangulations should be classified into four groups with each group corresponding to a term on the right-hand side of the recurrence above.

2.5 Write a simple recursive function $SRCAT(n)$ (for "Simple Recurrence CATalan") in Java that given an input n calculates P_n using the recurrence relation (1) in Lamé's paper directly.

2.6 Write another Java program that repeatedly uses $SRCAT$ to calculate P_i for $i = 3, 4, 5\ldots$ Restrict the total time your program uses to 10 minutes. What is the largest value N_0 of i for which your program calculates P_i? Print out a table with i and the time required in seconds by $SRCAT$ to calculate each of the P_i values. Your table should have a row for each $i = 3, 4, 5\ldots, N_0$.

2.7 From your calculations you may observe that it seems that for all n, if $n \geq 3$ then $P_{n+1} \geq 2 * P_n$. Give a simple mathematical argument that establishes the truth of this statement.

2.8 Prove that for all n, if $n \geq 3$ then $P_n \geq 2^n/8$.

2.9 What does this tell you about the efficiency of the naïve algorithm for solving the optimal polygon triangulation problem?

2.10 Write a Java program that repeatedly uses the recurrence given in formula (1) in Lamé's paper to calculate P_i for $i = 3, 4, 5\ldots$ but that stores the computed values in an array systematically and uses them as needed. Restrict the total time your program uses to 10 minutes. What is the largest value M_0 of i for which your program calculates P_i?

2.11 Extend your program to print out a table of values of i and time required in seconds to compute P_i for $i = 3, 4, \ldots, M_0$.

2.12 Graph the tables obtained in 2.6 and 2.11. Analyse these graphs and write down your observations.

Lamé's Method for deriving a formula for P_n

Section II of Lamé's paper gives an alternative way of counting triangulations of a polygon. Read this section carefully.

TASKS:

Consider a 6-sided polygon $ABCDEF$.

3.1 Draw all triangulations of the polygon where:

- AC is one of the diagonals in the triangulation.
- AD is one of the diagonals in the triangulation.
- AE is one of the diagonals in the triangulation.

How many total triangulations did you draw?

3.2 Repeat the same with vertex B as the "special" vertex, i.e., draw all triangulations where:

- BD is one of the diagonals in the triangulation.
- BE is one of the diagonals in the triangulation.
- BF is one of the diagonals in the triangulation.

How many total triangulations did you draw?

3.3 Do the same with vertices C, D, E, F being "special".

3.4 Consider the triangulation of $ABCDEF$ in figure 1 (of section 1). How many times is that triangulation repeated in all the triangulations that you drew for $ABCDEF$ in this section? Identify the diagonals in whose group it was drawn.

3.5 Do the same for the different triangulations of $ABCDEF$ that you drew in section 1.

3.6 What would you guess about the number of times any triangulation of $ABCDEF$ is repeated? Argue why your guess is correct.

3.7 Consider the n-sided polygon $A_1A_2 \dots A_n$. Let P_i denote the number of different triangulations of a polygon with i sides.

(*a*) Calculate, in terms of P_i's, the number of triangulations of this polygon that have A_1A_3 as a diagonal, that have A_1A_4 as a diagonal, that have A_1A_j as a diagonal.

(*b*) Consider drawing triangulations treating A_1 as the "special" vertex. That is, draw all triangulations where A_1A_3 is a diagonal, then draw all triangulations where A_1A_4 is a diagonal, etc. all the way up to where A_1A_{n-1} is a diagonal. What is the number of triangulations you draw (in terms of P_i's) when A_1 is treated as a special vertex?

(*c*) Suppose we repeat the above process with with another vertex (say A_2) being the special vertex instead of A_1. What can you say about the number of triangulations drawn as compared to the number of triangulations drawn when A_1 was chosen as the special vertex? Explain in your own words why this is true.

(*d*) Consider doing what you did for A_1 in (*b*) successively for each vertex. That is, enumerate all triangulations treating A_1 as a special vertex, treating A_2 as a special vertex, ...treating A_n as a special vertex. Now consider the specific triangulation of $A_1, A_2 \dots A_n$ obtained by drawing the diagonals $A_1A_3, A_1A_4, \dots A_1A_{n-1}$. How many times is this triangulation enumerated? What about the triangulation obtained by drawing the diagonals $A_1A_4, A_1A_5 \dots A_1A_{n-2}$ and the two diagonals $A_2A_4, A_{n-2}A_n$? Justify your answer.

(*e*) What is your guess as to how many times any specific triangulation is enumerated? Explain in your own words why this is the case.

3.8 Combine (*b*) and (*e*) to derive the formula (2) in Lamé's paper. Explain in your own words how this formula is obtained.

3.9 Combine formulas (1) and (2) in Lamé's paper to obtain the formula (3) in Lamé's paper. Show all the steps in your calculation. Explain why this formula is better for calculating P_n.

3.10 Using formula (3) in Lamé's paper, show that $P_{n+2} = \frac{1}{n+1}\binom{2n}{n}$ where $\binom{2n}{n} = \frac{(2n)!}{n!n!}$.

3.11 Write a simple recursive function $ASRCAT(n)$ (for "Another Simple Recurvise CATalan") in Java that given an input n calculates P_n using the recurrence relation (3) in Lamé's paper directly.

3.12 Write another Java program that repeatedly uses $ASRCAT$ to calculate P_i for $i = 3, 4, 5 \ldots$ Restrict the total time your program uses to 10 minutes. What is the largest value L_0 of i for which your program calculates P_i?

3.13 Extend your program to print out a table of values of i and time required in seconds by $ASRCAT(n)$ to compute each of the P_i values for $i = 3, 4, \ldots, L_0$. Your table should have a row for each $i = 3, 4, \ldots, L_0$.

3.14 Write a better Java program using the ideas from dynamic programming ("store and re-use") that repeatedly calculates P_i for $i = 3, 4, \ldots$ Restrict the total time your program uses to 10 minutes. What is the largest value L_1 of i for which your program calculates P_i.

3.15 Extend your program to print out a table of values of i and the time required in seconds to calculate each of the P_i values. Your table should have a row for each $i = 3, 4, \ldots, L_1$.

3.16 Graph the tables obtained in 3.13 and 3.15. Analyse all four graphs obtained and write down your observations. How do the results for the second two programs compare with your first two programs? How fast does the running time of the last two programs grow?

3.17 Discuss how the choice of Lamé's formulas (1) or (3), or using dynamic versus naïve recursive programming influences the effectiveness of computation.

Notes to the Instructor

This project is most suitable for use in an upper-division undergraduate algorithm design and analysis course. In particular, it is best used at the time when the students are learning the *dynamic programming* paradigm for algorithm design. It allows them to see why a naïve solution is infeasible for solving the Optimal Polygon Triangulation Problem for polygons with large number of sides and how the dynamic programming technique allows one to do the same calculation much more efficiently.

Some of thet tasks, as stated in the project, ask the students to write JAVA programs. Use of JAVA is not critical. Any programming language that allows for recursion and use of arrays (e.g. C, C++) can be substituted for JAVA without affecting the project.

References

[1] Euler, L., *Leonhard Euler und Christian Goldbach, Briefwechsel 1729–1764*, Juskevic, A. P., Winter, E. (editors), Akademie Verlag, Berlin, 1965.

[2] Euler, L., *Novi Commentarii Academiae Scientarium Imperialis Petropolitanque* **7** (1758–59), p. 9–28.

[3] Gillispie, C. C., Holmes, F. L., (editors) *Dictionary of Scientific Biography*, Scribner, New York, 1970.

[4] Lamé, G., "Un polygone convexe étant donné, de combien de manières peut-on le partager en triangles au moyen de diagonales?," *Journal de Mathématiques Pures et Appliquées*, **3** (1838), 505–507.

[5] Segner, A., "Enumeratio Modorum Quibus Figurae Planae Rectilinae per Diagonales Dividuntur in Triangula", *Novi Commentarii Academiae Scientarium Imperialis Petropolitanque* **7** (1758-59), 203–209.