The Triangulation of Titling Data in Non-Linear Gaussian Fashion via ρ Series

John Doe
Magic Department, Richard Miles University

Richard Row, LaTeX Academy

October 31, 2014

Contents

Chapter 1

The Platonic Solids

"A platonic solid is a polyhedron all of whose faces are congruent regular polygons, and where the same number of faces meet at every vertex. The best know example is a cube whose faces are six congruent squares." - John Kaiser

"Mathematics as an expression of the human mind reflects the active will, the contemplative reason, and the desire for aesthetic perfection. Its basic elements are logic and intuition, analysis and construction, generality and individuality." - Richard Courant

"For scholars and laymen alike it is not philosophy but active experience in mathematics itself that can alone answer the question: What is mathematics?" - Richard Courant

1.1 There are only five!

The Greeks recognized that there are only five platonic solids. But why is this so? The key observation is that the interior angles of the polygons meeting at a vertex of a polyhedron add to less than 360 degrees. To see this note that if such polygons met in a plane, the interior angles of all the polygons meeting at a vertex would add to exactly 360 degrees. Now cut an angle out of paper, and fold another piece of paper to that angle along a line [5]. The first piece will fit into the second piece when it is perpendicular to the fold. Think of the fold as a line coming out of our polyhedron. The faces of the polyhedron meet at the fold at angles less than 90 degrees. How can this be possible? Try wiggling your first piece of paper within the second. To be able to incline it with respect to the fold you have to decrease the angle of the first piece, or increase the angle of the second. Next we'll consider all possibilities for the number of faces meeting at a vertex of a regular polyhedron.

For each possibility we actually construct such a polyhedron, a picture of which you can see close by on this page. Here are the possibilities:

- **Triangles**. The interior angle of an equilateral triangle is 60 degrees. Thus on a regular polyhedron, only 3, 4, or 5 triangles can meet a vertex. If there were more than 6 their angles would add up to at least 360 degrees which they can't. Consider the possibilities:

- **Squares**. Since the interior angle of a square is 90 degrees, at most three squares can meet at a vertex. This is indeed possible and it gives rise to a **hexahedron** or **cube**.

  - ✓ *3 triangles* meet at each vertex. This gives rise to a **Tetrahedron**.
  - ✓ *4 triangles* meet at each vertex. This gives rise to an **Octahedron**.
  - ✓ *5 triangles* meet at each vertex. This gives rise to an **Icosahedron**.

- **Pentagons**. As in the case of cubes, the only possibility is that three pentagons meet at a vertex. This gives rise to a Dodecahedron.

- **Hexagons** or regular polygons with more than six sides cannot form the faces of a regular polyhedron since their interior angles are at least 120 degrees.

Before continuing, let us collect some data. Let

- m be the number of polygons meeting at a vertex,
- n the number of vertices of each polygon,
- f the number of faces of the polyhedron,
- e the number of edges of the polyhedron, and
- v the number of vertices of the polyhedron.

name	n	m	f	e	v
Tetrahedron	3	3	4	6	4
Octahedron	3	4	8	12	6
Icosahedron	3	5	20	2	1
Hexahedron	4	3	6	9	4
Dodecahedron	5	3	4	15	9

Table 1.1: Pair of numbers

Our aim now is to show that for any pair of number n and m the values of the other parameters, f, e, and v are determined uniquely. First we note that since two faces meet in one edge, we must have:

$$e = \frac{nf}{2} \tag{1.1}$$

Next, since every vertex is shared by m faces, we must have:

$$v = \frac{nf}{m} \tag{1.2}$$

It is apparent from the Table that for all five regular polyhedra

$$f = 2 + e - v \tag{1.3}$$

We'll see below that this equations actually holds for all convex polyhedra. Given m and n the above three equations determine f, e, and v uniquely, and so there are only five possible regular polyhedra.

1.2 Euler's Polyhedron Theorem

To see why it is true we proceed in several steps. First we remove one face from the polyhedron. Let

$$F = f - 1 \tag{1.4}$$

be the new number of faces. We need to show

$$f = 1 + e - v \tag{1.5}$$

Now think of the remaining faces of the polyhedron as made of rubber and stretched out on a table. This will certainly change the shape of the polygons and the angles involved, but it will not alter the number of vertices, edges, and faces. Now we draw diagonals in the stretched faces out of the Polygons. Every diagonal increase the number e of edges by one, and also the number F of faces, so that our equation **(1.5)** remains valid. We continue this process until all polygons have been changed into triangles.

In the final stage we remove triangles until we are left with only one triangle for which (*) is obviously true. How do we do that? If the removed triangle has exactly one edge on the boundary then F and e are both decreased by 1 and **(1.5)** remains true. If it has two edges on the boundary then F is reduced by 1, e is reduced by 2, and v is reduced by 1, so that **(1.5)** remains true.

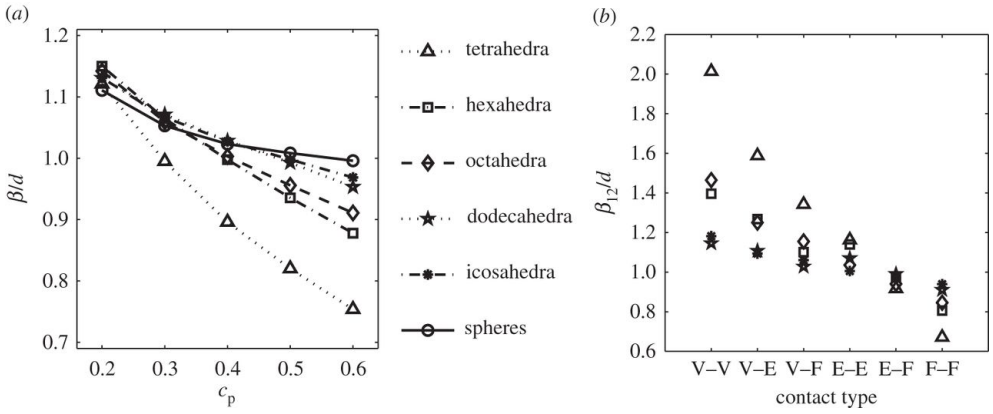


Figure 1.1: 6 models of thermodynamics
Third-order thermo-mechanical properties for packs of Platonic solids using statistical micromechanics

Chapter 2

Analysis of Algorithms

"By understanding a machine-oriented language, the programmer will tend to use a much more efficient method; it is much closer to reality..." - Donald Knuth

"I took a computer-science course to fill a prerequisite at Stanford, and I realized that every day was a new problem, and every day you got to think about how to solve something new, how to reason through something new, how to develop an algorithm to solve for something you hadn't worked on before." - Marissa Mayer

Despite all of our technological advances, content creation still requires time, inspiration, and a certain amount of sweat. There aren't any shortcuts. You can't write an algorithm for it. You can't predict it. You can't code it. - Shawn Amos

2.1 Cost models

Time efficiency estimates depend on what we define to be a step. For the analysis to correspond usefully to the actual execution time, the time required to perform a step must be guaranteed to be bounded above by a constant. One must be careful here; for instance, some analyses count an addition of two numbers as one step. This assumption may not be warranted in certain contexts. For example, if the numbers involved in a computation may be arbitrarily large, the time required by a single addition can no longer be assumed to be constant.

Two cost models are generally used:

1. the uniform cost model, also called uniform-cost measurement (and similar variations), assigns a constant cost to every machine operation, regardless of the size of the numbers involved

2. the logarithmic cost model, also called logarithmic-cost measurement (and similar variations), assigns a cost to every machine operation proportional to the number of bits involved [1]

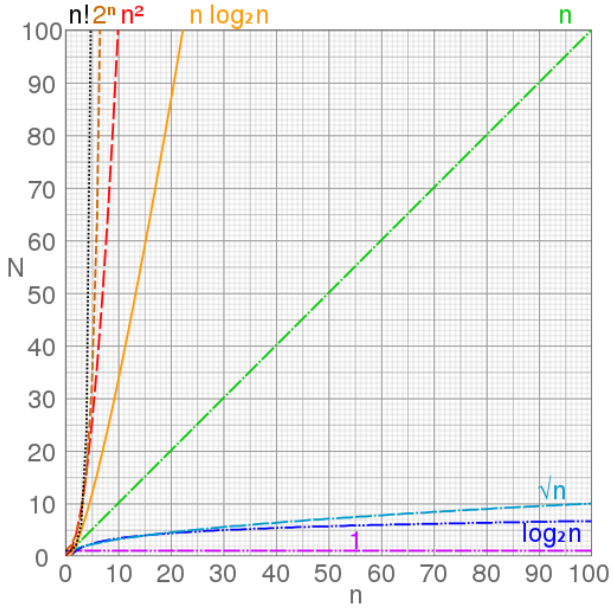
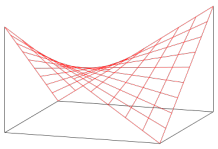


Figure 2.1: Graph of number of operations
The analysis of algorithms, N vs input size, n for common complexities, assuming a coefficient of 1

Run-time analysis is a theoretical classification that estimates and anticipates the increase in running time (or run-time) of an algorithm as its input size (usually denoted as n) increases. Run-time efficiency is a topic of great interest in computer science: A program can take seconds, hours or even years to finish executing, depending on which algorithm it implements (see also performance analysis, which is the analysis of an algorithm's run-time in practice).[3]

2.2 Shortcomings of empirical metrics

Since algorithms are platform-independent (i.e. a given algorithm can be implemented in an arbitrary programming language on an arbitrary computer running an arbitrary operating system), there are significant drawbacks to using an empirical approach to gauge the comparative performance of a given set of algorithms.



Take as an example a program that looks up a specific entry in a sorted list of size n. Suppose this program were implemented on Computer A, a state-of-the-art machine, using a linear search algorithm, and on Computer B, a much slower machine, using a binary search algorithm.

$q1/q2$	0	1
0	$\frac{dvc}{dt} = -\frac{1}{C}I_{load}$	$\frac{Lic}{dt} = \frac{63}{V}I_i$
1	$\frac{dvc}{dt} = -\frac{R_l}{L}i_L - \frac{1}{C}I_{load}$	$\frac{RiV}{dt} = -\frac{R_i}{V}i_L - \frac{1}{V}I_{load}$

Table 2.1: I/O - input/output algorithm

2.3 Evaluating run-time complexity

As a rule-of-thumb [4], one can assume that the highest-order term in any given function dominates its rate of growth and thus defines its run-time order. In this example, n^2 is the highest-order term, so one can conclude that $f(n) = O(n^2)$. Formally this can be proven as follows:

$$\left[\frac{1}{2}(n^2+n)\right]T_6 + \left[\frac{1}{2}(n^2+3n)\right]T_5 + (n+1)T_4 + T_1 + T_2 + T_3 + T_7 < cn^2, n \geq n_0 \tag{2.1}$$

n	**Computer A run-time (in nanoseconds)**	**Computer B run-time (in nanoseconds)**
16	8	100,000
63	32	150,000
250	125	200,000
1,000	500	250,000

Table 2.2: Platform-independent algorithms

$$\sin(2x) = \frac{2\cos(x) + \frac{sin(x)}{cos(x)}}{1 - \tan^2(x)}$$

$$p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3$$

$$\begin{aligned} 2x - 5y &= 8 \\ 3x + 9y &= -12 \end{aligned}$$

$$\begin{array}{ccc} x = y & w = z & a = b + c \\ 2x = -y & 3w = \frac{1}{2}z & a = b \\ -4 + 5x = 2 + y & w + 2 = -1 + w & ab = cb \end{array}$$

Suppose this program were implemented on Computer A, a state-of-the-art machine, using a linear search algorithm:

$$\lim_{n\to\infty} a_n = g \Leftrightarrow \forall \varepsilon > 0 \, \forall n > N_\varepsilon \colon |a_n - g| < \varepsilon \tag{2.2}$$

$$\sum_{i=1}^{\infty} \frac{1}{n^s} = \prod_{p} \frac{1}{1 - p^{-s}}$$

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\,\mathrm{d}x$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx\,\mathrm{d}x =$$

$$= \frac{1}{\pi} \int_{-\pi}^{\pi} x^2 \cos nx\,\mathrm{d}x$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx\,\mathrm{d}x =$$

$$= \frac{1}{\pi} \int_{-\pi}^{\pi} x^2 \sin nx\,\mathrm{d}x$$

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + a_4}}} \tag{2.3}$$

		Primes				
		$x_0 + x_{-1} + x_{-2}$	3	5	7	
Powers	$\frac{1}{9}$	$2x^4 + 4x^3 + 2x^2 + \frac{(x+2)(x+4)}{4}$	$\frac{1}{3}$	$\frac{1}{16}$	$\frac{1}{32}$	
	540	$3x^5 + 4x^4 + 5x^2 + (x+2)(x+4)$	$\frac{1}{256}$	$\frac{1}{512}$	$\frac{1}{1024}$	
Powers	gcd	$2x^4 + 4x^3 + 2x^2 + (x+2)(x+4)$	2	0	0	$\geq$ minimum capacity
	fog	$\sqrt{\frac{x^2}{2}}$	$\sqrt{\frac{x^2}{4}}$	$\sqrt{\frac{x^2}{8}}$	$\sqrt{\frac{x^2}{16}}$	$\leq$ maximum capacity

Table 2.3: Powers of prime numbers

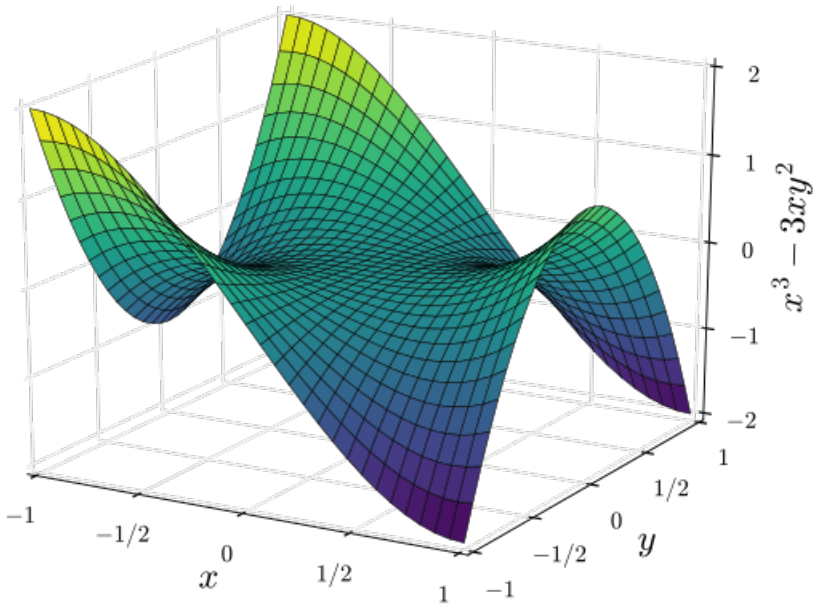
http://www.math.utah.edu/ pa/math/polyhedra/polyhedra.html

Chapter 3

The Final Frontier

This is the final chapter

> "Space: [4] the final frontier. These are the voyages of the starship Enterprise. Its five-year mission: to explore strange new worlds, to seek out new life and new civilizations, to boldly go where no man has gone before." [2]

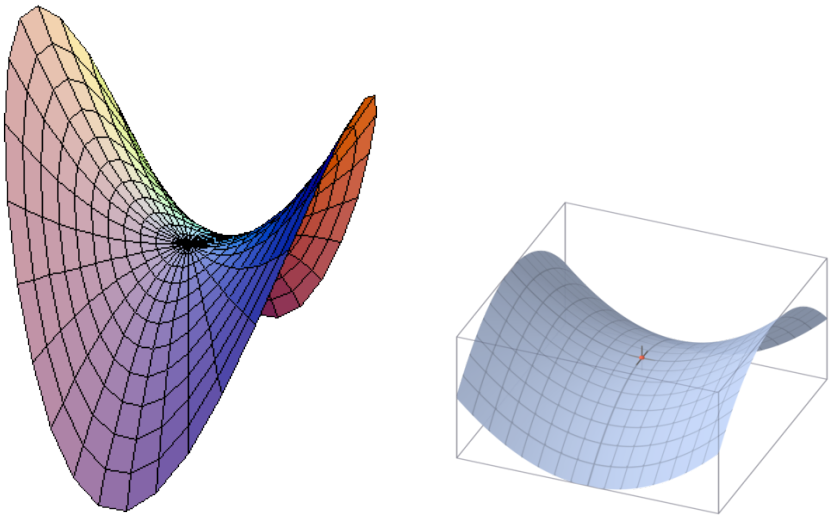


$$ax^2 + bx + c + \frac{12}{x+1}$$

3.1 Tensile structure

A **tensile structure** is a construction of elements carrying only tension and no compression or bending. The term tensile should not be confused with tensegrity, which is a structural form with both tension and compression elements. Tensile structures are the most common type of thin-shell structures. Most tensile structures are supported by some form of compression or bending elements, such as masts (as in The O2, formerly the Millennium Dome), compression rings or beams.

A **tensile membrane structure** is most often used as a roof, as they can economically and attractively span large distances.



(a) Hyperbolic paraboloid (b) Saddle Shape

Figure 3.1: 2 Figures side by side

As before, plotting the left hand side and right hand side of the equation against the tension, T, will give the equilibrium tension for a given pretension.

By equilibrium:

$$dist = \sqrt{\left(\frac{dx}{hx}\right)^2 + \left(\frac{dy}{hy}\right)^2 + \left(\frac{dz}{hz}\right)^2} \tag{3.1}$$

By geometry:

$$L_0 + \frac{L_0(T - T_0)}{EA} = \sqrt{S^2 + 4\left(\frac{W(L_0 + \frac{L_0(T-T_0)}{EA}}{4T}\right)} \tag{3.2}$$

3.2 Ruled surface

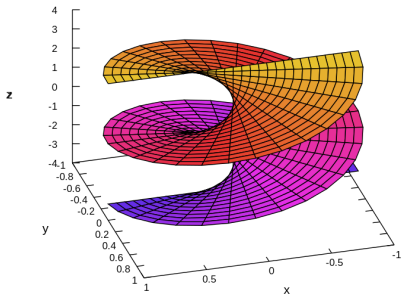


Figure 3.2: A ruled helicoid

The properties of being ruled or doubly ruled are preserved by projective maps, and therefore are concepts of projective geometry. In algebraic geometry ruled surfaces are sometimes considered to be surfaces in affine or projective space over a field, but they are also sometimes considered as abstract algebraic surfaces without an embedding into affine or projective space, in which case "straight line" is understood to mean an affine or projective line. A developable surface is a surface that can be (locally) unrolled onto a flat plane without tearing or stretching it. If a developable surface lies in three-dimensional Euclidean space, and is complete, then it is necessarily ruled, but the converse is not always true. For instance, the cylinder and cone are developable, but the general hyperboloid of one sheet is not. More generally, any developable surface in three dimensions is part of a complete ruled surface, and so itself must be locally ruled. There are developable surfaces embedded in four dimensions which are however not ruled.

The "moving line" view means that a ruled surface has a parametric representation of the form

$$S(t,u) = p(t) + ur(t)\ p(t) = (cos(2t), sin(2t), cos(t)sin(2t))$$

List of Tables

List of Figures

Bibliography

[1] S. T. Abedon. Lysis and the interaction between free phages and infected cells. In Jim D. Karam Karam, John W. Drake, Kenneth N. Kreuzer, Gisela Mosig, Dwight Hall, Frederick A. Eiserling, Lindsay W. Black, Elizabeth Kutter, Karin Carlson, Eric S. Miller, and Eleanor Spicer, editors, *Molecular biology of bacteriophage T4*, pages 397–405. ASM Press, Washington DC, 1994.

[2] S. T. Abedon, P. Hyman, and C. Thomas. Experimental examination of bacteriophage latent-period evolution as a response to bacterial availability. *Applied and Environmental Microbiology*, 69:7499–7506, 2003.

[3] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.

[4] George D. Greenwade. The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3):342–351, 1993.

[5] John Smith. *The Go Zone*. The Grape Publishing, 1996.