Solid Angles and Trigonometry in Space

Tey Yi Fan and Tan Yan Zhong

Hwa Chong Institution
Project Work 2018 Group 8-13

August 16, 2018

Nomenclature

First, we will introduce some terminology and notation used throughout the paper.

$\hat{i}$ — $\langle x, y, z\rangle = \langle 1, 0, 0\rangle$ (in $\mathcal{R}^3$)
The unit vector in the positive x direction

$\hat{j}$ — $\langle x, y, z\rangle = \langle 0, 1, 0\rangle$
The unit vector in the positive y direction

$\hat{k}$ — $\langle x, y, z\rangle = \langle 0, 0, 1\rangle$
The unit vector in the positive z direction

$\vec{u} \cdot \vec{v}$ — $u_1v_1 + u_2v_2 + \cdots + u_nv_n$
The dot product of vectors $\vec{u}$ and $\vec{v}$

$\vec{u} \times \vec{v}$ — $\det\left(\hat{i} + \hat{j} + \hat{k}, \vec{u}, \vec{v}\right)$
The cross product of the vectors $\vec{u}$ and $\vec{v}$

θ and ϕ	Any Planar Angle
Ω and Φ	Any Solid Angle
$\vec{P}$	The vector $\overrightarrow{OP}$
$\triangle ABC$	The triangle with vertices A, B and C
$ABCD$	The tetrahedron with vertices A, B, C, D
$[ABC]$	The area of $\triangle ABC$
$\angle AB$	The dihedral angle at the edge AB
$\mathfrak{t}_i$	The i^{th} minor 3D trigonometric function
$\mathcal{T}_i$	The i^{th} major 3D trigonometric function
τ	Any 3D trigonometric function in general
$\operatorname{Br}\tau(\Omega)$	The branch of the function τ

Authors' Note

This is a concise version of the report on all of the authors' research findings throughout the project. Many results, their proofs, and their discussions have been omitted completely or otherwise shortened. Algebraic detail for all proofs in this paper will be omitted for the sake of brevity and left as exercises for the reader, but the idea will be presented. Sections which are less important towards the whole project are especially shortened, such as the Introduction section, and so motivation for the project may seem less developed and the rigorous foundation may be compromised in favour of more intuitive discussions.

Contents

I. Introduction

Solid angles are a formal measurement of the separation between three planes intersecting at one point. It is often seen as the three-dimensional (3D) analogue of the usual planar angle in two-dimensional (2D) space. It is measured in *steradians*, which span throughout the range $[0, 4\pi)$.

Perhaps the greatest application of planar angles is in the field of Trigonometry. In Trigonometry, the 2D trigonometric functions—mainly sine, cosine, tangent, their multiplicative and functional inverses—relate planar angles to ratios of side lengths of triangles. These functions, since they were defined in Ancient India in the 4th century CE [Boyer, 1991], have become widely applicable in almost every field of pure and applied mathematics, to the extent that without the existence of these functions, many fields of mathematics could not progress..

It is therefore surprising that, despite the overwhelming utility of trigonometry, no three-dimensional generalisation of it involving three-dimensional 'trigonometric' functions exist. This paper will propose such generalisations and investigate basic functional properties of these functions, as well as discussions on their utility in other fields of pure and applied mathematics. We will call these generalised functions the **hedronometric functions**. We will now present rigorous definitions of these hedronometric functions.

Formally, we will be investigating the tetrahedron $OVAB$ in $\mathcal{R}^3$ where point O is the origin, point $V(x, y, z)$ is the *variable vertex*, and points $A(x, 0, 0)$ and $B(0, y, 0)$ are the two last vertices of the tetrahedron. In trigonometry, the product of the length of the longest side and the trigonometric functions give the lengths of the other sides. Similarly, we want to achieve this effect.

Definition 1 (Minor Hedronometric Functions)**.** For any vector $\vec{v}\langle x, y, z\rangle$, define three functions

$$\mathfrak{t}_1 = \frac{x}{v} \quad \mathfrak{t}_2 = \frac{y}{v} \quad \mathfrak{t}_3 = \frac{z}{v},$$

where $v = OV$ is the length of the longest side.

These functions achieve the desired effect. We want to also define three functions to represent ratios of side areas.

Definition 2 (Major Hedronometric Functions)**.** For any tetrahedron $OVAB$, define

$$\mathcal{T}_1 = \frac{[OVA]}{[VAB]} \quad \mathcal{T}_2 = \frac{[OVB]}{[VAB]} \quad \mathcal{T}_3 = \frac{[OAB]}{[VAB]}.$$

Since we defined the 3D generalisation of the trigonometric functions, of course we would need a 3D analogue of the angle. For this, we utilise the solid angle. The hedronometric functions are then functions of the solid angle Ω. Because the newly defined functions are not single-valued as formal functions should be, we will introduce the notation ${}_\theta\tau(\Omega)$ to dictate a relationship between the x and y coordinates of the vector $\vec{v}$ which τ describes. In particular,

$$y = x \tan \theta.$$

We call the angle θ the *branch* of the hedronometric function τ. Now, every function ${}_\theta\tau(\Omega)$ is single-valued. We will investigate all branches of the functions in general, but if some problems are too difficult to discuss in the general case, we will reduce the investigation to only include the family of hedronometric functions with branch $\theta = 0$. We call this branch the *primitive branch*, where properties are trivial due to it being reduced to the two-dimensional case.

Research Questions

The project has investigated a total of four research questions.

1. What are the analytic relations between the hedronometric functions?
2. What is a numerical method of computing values the hedronometric functions?
3. What are the graphs of the hedronometric functions?
4. What is an application of the hedronometric functions in Astrophysics?

The first question aims to establish fundamental theoretical facts about the hedronometric functions, which will be instrumental in later investigations. Then, the second question will establish the means to numerically compute the value of any hedronometric function given any branch and any solid angle. The third question will provide a method to visualise the functions, and lastly the fourth question will allow us to see a practical application of the theoretical constructs built throughout this paper.

II. Results

i. Analytic Relations

We will come to demonstrate several important facts about the hedronometric functions, each providing insight into the nature of the functions and how they relate to each other. First, we consider the functions on their own and discuss two important properties—their domains and continuity.

The *domain* of a function is the set of all values for which it is defined. The minor hedronometric functions are always defined for all real numbers, as regardless of the choice of the solid angle Ω, we can always find a vector $\vec{v}$ of any given branch such that it makes a solid angle Ω with the xy-plane. This vector then corresponds to one tetrahedron, which gives the values of the minor functions.

Usually, all the major functions are also always defined as the exact same reasoning holds. The unique exception is when the denominator, $[VAB]$, is equal to 0. Clearly, this occurs precisely when the solid angle Ω is 0, regardless of the value of θ. Thus, the major functions are defined everywhere except $\Omega = 0$.

Now that we have this fundamental fact established, we may investigate some more interesting results.

One of the most fundamental results of trigonometry is the identity $\sin^2 x + \cos^2 x \equiv 1$ (this is called the Pythagorean identity). This result can be derived from the Pythagorean identity $a^2 + b^2 = c^2$, where a and b are the lengths of shorter sides of any right-angle triangle, and c is the length of the longest side. There is a corresponding version of this, which we will call the *Sum-of-Squares Identity*.

Theorem (Sum-of-Squares Identity). *For any given branch θ and solid angle Ω, the identity*

$${}_\theta \mathrm{t}_1^2(\Omega) + {}_\theta \mathrm{t}_2^2(\Omega) + {}_\theta \mathrm{t}_3^2(\Omega) \equiv 1.$$

is always true.

The proof of this identity is rather simple: just substitute the definitions of the functions, then utilise the formula for the magnitude of a vector in 3 dimensions. This identity provides a natural generalisation of the trigonometric Pythagorean identity.

This correlation is even more obvious if one considers the fact that ${}_\theta \mathrm{t}_1(0) \equiv \cos\theta$, ${}_\theta \mathrm{t}_2(0) \equiv \sin\theta$ and ${}_\theta \mathrm{t}_3(0) \equiv 0$. Hence in the special case that $\Omega = 0$, the Sum-of-Squares Identity easily reduces to the trigonometric Pythagorean Identity.

Some other identities using the same idea also arise easily by definition. Some of these are presented in Appendix A, but they are generally not as important as the Sum-of-Squares Identity.

At this point, it becomes rather difficult to find more analogues of the trigonometric identities regarding the hedronometric functions. The hedronometric angle addition identity, for example, is difficult to find as it is near-impossible to describe the change in the solid angle Ω in relation to the change in the vector $\vec{v}$.

An important observation to allow us to discover more relations is the dependence of the solid angle Ω to the two geodesics from V to $(1,0,0)$ and $(0,1,0)$. If we have a vector $\vec{v}_0$ with a corresponding solid angle Ω_0, and a vector $\vec{v}$ so that V lies on the geodesic through V_0 and $(1,0,0)$, we can say that $\vec{v}$ satisfies

$$cy = bz$$

as this is the plane which the geodesic lies on. The importance of this result lies in the fact that it allows for relations between functions of different branches to be discovered. If we have a branch of functions with well-known properties, we can now find information about all branches of functions. It is easy to show that

$$\begin{aligned} {}_0\mathrm{t}_1(\Omega) &= {}_{\pi/2}\mathrm{t}_2(\Omega) = \cos\Omega \\ \text{and } {}_0\mathrm{t}_3(\Omega) &= {}_{\pi/2}\mathrm{t}_3(\Omega) = \sin\Omega. \end{aligned}$$

With these facts as a reference point, in combination with the method we have found above, we can arrive at an important fact.

Theorem. *Let $\vec{v}$ lie on $\widehat{B'V_0}$ where $B' = (0,1,0)$ and $y_0 = 0$. The first minor function[1] may be evaluated as*

$$_\theta\mathrm{t}_1(\Omega) = \sqrt{\frac{1}{\sec^2\theta + \tan^2\Omega_0}}.$$

It is a trivial corollary of this theorem that

$$_\theta\mathrm{t}_2(\Omega) = \sqrt{\frac{1}{\csc^2\theta + \cot^2\theta\tan^2\Omega_0}}$$

and

$$_\theta\mathrm{t}_3(\Omega) = \sqrt{\frac{1}{\sec^2\theta\cot^2\Omega_0 + 1}}.$$

These results are crucial in considering the relations between the hedronometric functions, as it is written in only trigonometric functions, whose properties are well known. For example, with this, we can easily show that the minor functions are continuous almost everywhere on $\mathcal{R}$. With any value of Ω and θ, it is easy to evaluate the minor functions, since it is possible to use a computer to find the value of Ω_0 numerically.

[1]The complementary results for the second and third minor functions may be found in Appendix A.

There is also an important fact which concerns the major functions. In 1803, the French Lazare Carnot discovered the hedronometric Law of Cosines, which states that for any tetrahedron with faces W, X, Y and Z opposite to vertices D, A, B and C respectively, the identity

$$\begin{aligned} W^2 \equiv X^2 + Y^2 + Z^2 - 2XY\cos(\angle DC) \\ - 2YZ\cos(\angle DA) - 2XZ\cos(\angle DB), \end{aligned}$$

where $\angle DC$ is the dihedral angle at the edge DC (and so on), always holds. If we let D be the origin, C be $(x,0,0)$, A be $(0,y,0)$ and B be (x,y,z), we return to the family of tetrahedra we are interested in.

Dividing both sides of the hedronometric Law of Cosines by W^2, we can easily rewrite the statement as

$$\begin{aligned} 1 \equiv \mathcal{T}_1^2 + \mathcal{T}_2^2 + \mathcal{T}_3^2 - 2\mathcal{T}_1\mathcal{T}_2\cos(\angle DC) \\ - 2\mathcal{T}_2\mathcal{T}_3\cos(\angle DA) - 2\mathcal{T}_1\mathcal{T}_3\cos(\angle DB). \end{aligned}$$

By vector methods, it is easy to see that $\cos(\angle DA) = \frac{1}{\sqrt{1+z^2/x^2}}$ and so on. It is also possible to demonstrate that this is the x-coordinate of the intersection between the positive x-plane and the geodesic passing through $(0,1,0)$ and V. Let us call the solid angle which is associated with this point of intersection Ω_0, and let that associated with the point of intersection between the positive y-plane and the geodesic through $(1,0,0)$ and V be Φ_0.

Direct computation shows that $\cos\Omega_0 = \frac{1}{\sqrt{1-z^2/x^2}}$ and $\cos\Phi_0 = \frac{1}{\sqrt{1-z^2/y^2}}$, and thus we can rewrite the Law of Cosines with the trigonometric and hedronometric functions.

Theorem (Functional Law of Cosines)**.** *For any tetrahedron OVAB, we have the identity*

$$\begin{aligned} 1 \equiv \mathcal{T}_1^2 + \mathcal{T}_2^2 + \mathcal{T}_3^2 - 2\mathcal{T}_1\mathcal{T}_2\cos\Phi_0 \\ - 2\mathcal{T}_2\mathcal{T}_3\cos\Omega_0 - 2\mathcal{T}_1\mathcal{T}_3\cos\Phi_0\cos\Omega_0. \end{aligned}$$

Notice that comparing terms with the original statement shows that $\angle DA = \Omega_0$ and $\angle DC = \Phi_0$. In addition, $\cos\angle DB = \cos\angle DA \cdot \cos\angle DC$. These are surprising results which have become easy corollaries of the functional restatement of the Law of Cosines.

ii. Numerical Values

Our goal in this research question is to evaluate any hedronometric function given the branch and solid angle. To evaluate some numerical values of the hedronometric functions, the most obvious approach will be to approximate it with a computer program. We will be using the Python programming language.

Due to L'Huilier, we know an extremely important formula for calculating any solid angle given the vertex V. L'Huilier's formula [Zwillinger, 1995] states that

> Given any spherical triangle on a sphere with vertices A, B and C, the spherical excess angle $E = \angle A + \angle B + \angle C - \pi$ can be given by
>
> $$4\tan^{-1}\sqrt{\tan\frac{s}{2}\tan\frac{s-a}{2}\tan\frac{s-b}{2}\tan\frac{s-c}{2}}$$
>
> where a, b and c are the angles at the origin between $\vec{B}$ and $\vec{C}$, $\vec{A}$ and $\vec{C}$, and $\vec{A}$ and $\vec{B}$ respectively, and where $s = \dfrac{a+b+c}{2}$.

In combination with the fact that on any sphere of radius R, a spherical triangle's area is ER^2, we know that the expression in L'Huilier's formula also gives the area of the spherical triangle we want. In fact, this is an expression for the solid angle.

Finding the Solid Angle

It is easy to know that $\cos a = x$, $\cos b = y$ and $\cos c = 0$, and thus $s = \frac{\cos^{-1} x + \cos^{-1} y + \pi/2}{2}$. We now have all the values needed in the formula above to calculate the solid angle. Obviously, the formula is extremely complicated[2] and nearly impossible to calculate by hand, but it may be approximated numerically by a computer. Thus, given any vector $\vec{v}$, this allows us to find the value of the solid angle Ω.

To determine the solid angle given any vector, we may use the following three-step approach. Here, we use `(x,y,z)` to represent the coordinates of the variable vertex, `k` to represent the solid angle, and `b` to represent the branch. (We will assume that the `math` module has been imported.)

Before substituting into the formula, we want to create a function to 'initialise' the coordinates. That is, to ensure that our coordinates do indeed describe a vector lying on the unit sphere, such that the input does not necessarily need to satisfy $x^2 + y^2 + z^2 = 1$, but only be in the right ratio.

[2] For the full formula, see Appendix B. We will call this formula $\Omega(x,y)$.

```
def init(x,y,z):
    f = math.sqrt(x*x+y*y+z*z)
    x = x/f
    y = y/f
    z = z/f
    return (x,y,z)
```

This code takes the user input of x, y and z and divides each by the magnitude of the vector $\vec{v}$, given by $\sqrt{x^2+y^2+z^2}$. This provides the coordinates of $\vec{v}_0$, the unit vector in the direction of $\vec{v}$. Now, we have no more use of the value z (we need just two pieces of information), and so we will focus on the variables x and y.

Finally, we will evaluate the solid angle.

```
def solidangle(x,y,z):
    (x,y,z) = init(x,y,z)
    a = math.acos(x)
    b = math.acos(y)
    c = math.acos(0)
    s = (a+b+c)/2
    def tann(n):
        return math.tan(n/2)
    return 4*math.atan(math.sqrt(tann(s)*tann(s-a)*
     ↪ tann(s-b)*tann(s-c)))
```

This code first defines the values `a`, `b` and `c`, which are angles in the formula for the solid angle which we have found. It then substitutes these values into the formula to give the desired result (the `tann` function is only defined to simplify notation).

Now, we can add all these pieces into the full program, which prints the solid angle associated with any variable vertex.

Full Program

```
import math

x = float(input('Enter x-coordinate: '))
y = float(input('Enter y-coordinate: '))
z = float(input('Enter z-coordinate: '))

def init(x,y,z):
    f = math.sqrt(x*x+y*y+z*z)
    x = x/f
    y = y/f
    z = z/f
    return (x,y,z)

def solidangle(x,y,z):
    (x,y,z) = init(x,y,z)
    a = math.acos(x)
    b = math.acos(y)
    c = math.acos(0)
    s = (a+b+c)/2
    def tann(n):
        return math.tan(n/2)
    return 4*math.atan(math.sqrt(tann(s)*tann(s-a)*
     ↪ tann(s-b)*tann(s-c)))

print(solidangle(x,y,z))
```

Finding the Vector

For any given vector, we have developed a program to find the associated solid angle. Now, we want the program to work the other way round—given any solid angle and branch, we want a program to find the associated vector. Though it is extremely tedious and perhaps impossible to analytically find the inverse to $\Omega(x, y)$, it is comparatively easy to approximate it with a program.

The easiest way to do so would be to run through all possible values of `x` between `0` and $\cos\theta$ (if $x > \cos\theta$, it must lie outside the circle). Of course, we cannot test all possible real numbers in this range, but we can test all of them to an increment of `0.00001` to obtain a reasonable approximation to three decimal points. For each possible `x`, we can generate the corresponding `y` and `z` values, making a tuple `(x,y,z)` for which we may evaluate the solid angle. Now, we can just pick the tuple which generates the closest solid angle to the desired value, which will be our approximation to three decimal points.

We need to first define a function which returns the minimum item in any list.

```
def minitem(list):
    min = list[0]
    for n in list:
        if n < min:
            min = n
    return min
```

Now, we will define the main function, which provides the vector corresponding to any given solid angle (`k`) and branch (`b`).

```
def vec(k,b):
    xrandlist = [n*math.cos(b)/1000000 for n in
     ↪ range(1000000)]
    yrandlist = [n*math.tan(b) for n in xrandlist]
    zrandlist = [math.sqrt(1-(x/math.cos(b))**2) for x
     ↪ in xrandlist]
    resultslist = []
    for i in range(1000000):
        (x,y,z)=(xrandlist[i],yrandlist[i],zrandlist[i])
        resultslist.append(abs(solidangle(x,y,z)-k))
    ri = resultslist.index(minitem(resultslist))
    (x,y,z) =
     ↪ (xrandlist[ri],yrandlist[ri],zrandlist[ri])
    return (x,y,z)
```

Now, we can define the other interesting properties relating to the vector associated with the given solid angle. These will be the hedronometric functions of the solid angle Ω. Note that the minor functions are just the coordinates of the vector, so we need not define new functions to represent them. The algebraic definitions are easy to see as equivalent to Definition 2 by vector methods.

```
def den(x,y,z):
    return math.sqrt((xy)**2 + (yz)**2 + (xy)**2)

def T1(x,y,z):
    return (x*math.sqrt(1-x*x))/den(x,y,z)

def T2(x,y,z):
    return (y*math.sqrt(1-y*y))/den(x,y,z)

def T3(x,y,z):
    return x*y/den(x,y)
```

Now, we can compile this code into a full program, which prints the values of all six hedronometric functions once given the branch and solid angle.

Full Program

```
import math

k = float(input('Enter solid angle: '))
b = float(input('Enter branch: '))

def solidangle(x,y,z):
    (x,y,z) = init(x,y,z)
    a = math.acos(x)
    b = math.acos(y)
    c = math.acos(0)
    s = (a+b+c)/2
    def tann(n):
        return math.tan(n/2)
    return 4*math.atan(math.sqrt(tann(s)*tann(s-a)*⌋
     ↪ tann(s-b)*tann(s-c)))

def vec(k,b):
    xrandlist = [n*math.cos(b)/1000000 for n in
     ↪ range(1000000)]
    yrandlist = [n*math.tan(b) for n in xrandlist]
    zrandlist = [math.sqrt(1-(x/math.cos(b))**2) for x
     ↪ in xrandlist]
    resultslist = []
    for i in range(1000000):
        (x,y,z)=(xrandlist[i],yrandlist[i],zrandlist[i])
        resultslist.append(abs(solidangle(x,y,z)-k))
    ri = resultslist.index(minitem(resultslist))
    (x,y,z) =
     ↪ (xrandlist[ri],yrandlist[ri],zrandlist[ri])
    return (x,y,z)

def den(x,y,z):
    return math.sqrt((xy)**2 + (yz)**2 + (xy)**2)

def T1(x,y,z):
    return (x*math.sqrt(1-x*x))/den(x,y,z)

def T2(x,y,z):
    return (y*math.sqrt(1-y*y))/den(x,y,z)

def T3(x,y,z):
    return x*y/den(x,y)

v = vec(k,b)
print(v[0], v[1], v[2], T1(v), T2(v), T3(v))
```

It is easy to modify this program slightly to yield Ω_0 and Φ_0 given Ω and θ in the formula of the functional law of cosines, which provides a relation between the major functions. One only has to find the two vectors which each lie on one of the same geodesics as $\vec{v}$, and substitute their coordinates into the `solidangle(x,y,z)` function.

iii. Graphs

With the code we have just developed in the last section, we can easily evaluate many values of the trigonometric functions and plot a graph with them. We will still be using the Python programming language, and using a library to do the plotting.

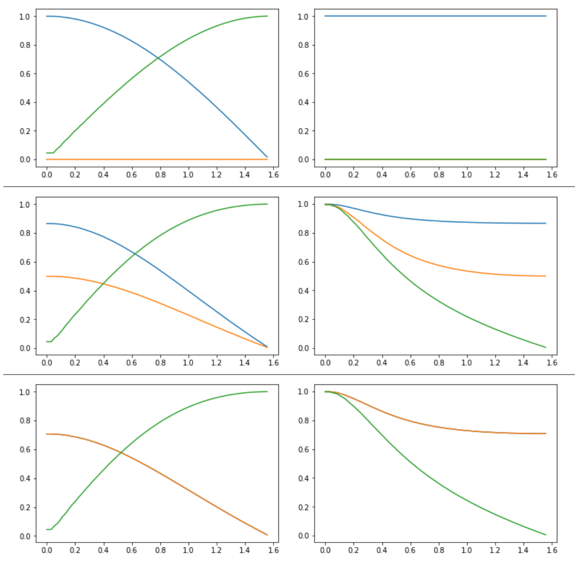
```
import matplotlib.pyplot as pyplot
```

To create a function which plots the first minor function, we can use the following code.

```
def kplott1(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(vec(k,b)[0])
    pyplot.plot(xlist,ylist)
```

Note that this function only executes a command, so it does not have a **return** statement. Executing this function gives us a plot of the first minor function given any branch. The exact same method can be applied to give plots of the other hedronometric functions. We will be forsaking precision in exchange for speed, since the exactness of the values are irrelevant compared to the general trends which we are interested in.

In the following graphs, blue represents the first (minor or major) function. Orange represents the second function, while green represents the third function. The plots of the minor functions are on the left, and the plots of the major functions are on the right.



In order: $\theta = 0°, 30°, 45°$.

Full Modified Program

```
import matplotlib.pyplot as pyplot

[program for functions from previous section, with the
 ↪ output of vec(k,b) being x,y,z instead of
 ↪ (x,y,z).]

def kplott1(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(vec(k,b)[0])
    pyplot.plot(xlist,ylist)

def kplott2(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(vec(k,b)[1])
    pyplot.plot(xlist,ylist)

def kplott3(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(vec(k,b)[2])
    pyplot.plot(xlist,ylist)

def kplotT1(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(T1(vec(k,b)))
    pyplot.plot(xlist,ylist)

def kplotT2(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(T2(vec(k,b)))
    pyplot.plot(xlist,ylist)

def kplotT3(b):
    xlist=[]
    ylist=[]
    for i in range(100):
        k = i*math.pi/200
        xlist.append(k)
        ylist.append(T3(vec(k,b)))
    pyplot.plot(xlist,ylist)

b = float(input('Enter branch: '))

# Minor Functions
kplott1(b)
kplott2(b)
kplott3(b)
pyplot.show()

# Major Functions
kplotT1(b)
kplotT2(b)
kplotT3(b)
pyplot.show()
```

Now, we can successfully plot any hedronometric function of any branch. Note that due to Theorem 4 (see Appendix A), the plots of $\theta = 30°$ and $60°$, as well as $0°$ and $90°$ are identical for both minor and major functions, only with "the colours switched over". When $\theta = 45°$, we know that $t_1 = t_2$ and $\mathcal{T}_1 = \mathcal{T}_2$, so their graphs overlap, which is the reason why only two graphs are displayed when there really are three.

Importantly, observe that the first and second minor functions are always constant multiples of the graph of $\cos\Omega$. This is not at all obvious from any form of analytic investigation. Peculiarly, no matter the value of θ, the third minor function is always identically equal for all values of Ω. This means that, moving V along any plane parallel to the xy-plane on the unit sphere, the area of the spherical triangle with vertices V, $(1,0,0)$ and $(0,1,0)$ is always constant.

The major functions are also constant multiples of each other when varying the branch, though the authors have not been able to find a formula for these. The three functions appear to be usually concurrent at the point $(0,1)$, except when $\theta = 0°$, in which case the third major function does not pass through $(0,1)$.

iv. Practical Application

An astronomical practical application of all the theory discussed in this report so far is in the investigation of fusion reactions in stars. All stars undergo a dramatic nuclear fusion reaction, occurring usually in its core. The most common types of fusion reactions include the fusion of hydrogen into helium, and that of helium into beryllium and carbon.

The most common method in which this reaction occurs is the proton-proton chain, where hydrogen is fused into helium in a chain of reactions, producing the net result:

$$6\,{}^1\mathrm{H} \longrightarrow {}^4\mathrm{He} + 2\,{}^1\mathrm{H} + 26.72\,\mathrm{MeV}$$

where MeV denotes mega electronvolts, a unit of energy approximately equal to 1.6×10^{-13} joules. Every such reaction produces 26.72 MeV of energy, among which 2% is carried away into space by neutrinos, and the rest is converted to either kinetic energy or radiative energy in gamma rays.

It is the gamma rays which we are interested in. In order to reach the emptiness of space, each ray has to pass through the internal diameter of the star. Every time it collides with a particle, it changes direction in an erratic manner, causing

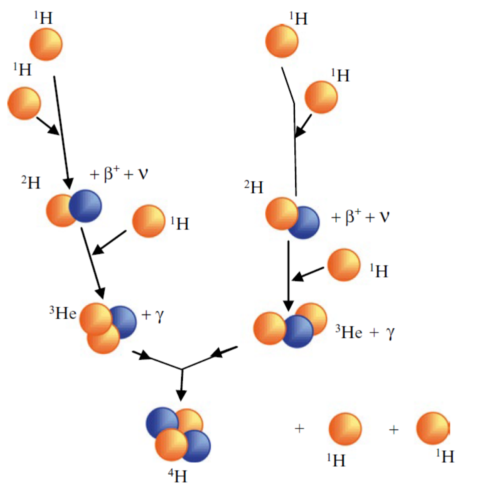


Figure 1: *The proton-proton chain: six ^1H atoms fuse to produce one ^4He atom and two excess ^1H atoms. Orange particles are protons, blue particles are neutrons, β^+ represent positrons, ν represent neutrinos and γ indicates photons. [Inglis, 2015]*

it to participate in a *random walk*[3] until it reaches the surface. This process can take up to a million years [Odenwald, n.d.].

It is currently extremely difficult to model a random walk for fairly obvious reasons. The estimates for the duration of the process, for example, vary by up to 5 orders of magnitude. The physical theory for this is not yet fully developed. However, our newly developed concept of the hedronometric functions might provide crucial insight into this aspect, in acting as a bridge between the coordinates of any particle and its associated solid angle. The reason this would be useful is that a solid angle holds a special significance in astrophysics—it is related to the radiative flux (the 'amount of light') passing through the surface of any sphere like a star.

III. Conclusion

In conclusion, this report has defined the six hedronometric functions as effective generalisations of the two-dimensional trigonometric functions. In relation to these hedronometric functions, we have proven several important but surprising identities, and developed a numerical method to approximating their values. Using this knowledge, we have successfully con-

[3] A random walk is a path of motion which is entirely random and unpredictable. It is a mathematical object often studied in relation to the theory of Markov chains, and especially probability theory.

structed their graphs. Finally, we have seen how all these knowledge can come together to contribute to the solution of an astronomical problem.

Further Extensions

It has also left some important questions unanswered:

1. What are the hedronometric analogues of the trigonometric angle-addition identities, product-to-sum and sum-to-product formulas, and the R-formula?
2. How can a vector triple-product, similar to the dot product, involving the hedronometric functions be defined?
3. What are the derivatives of the hedronometric functions?
4. What are the exact values of some of the hedronometric functions?
5. Does there exist explicit formulas for the hedronometric functions?
6. What does the plot of the hedronometric functions against their branches look like?
7. How can Hedronometry be further applied in other fields of mathematics?

These are questions which are pivotal in the development of this field, and crucial further extensions to this project.

Appendices

A. Analytic Relations: Proofs

Here we present the proofs of several facts previously used without proof, as well as some additional basic results not previously presented.

Theorem 1. *In the case that the solid angle $\Omega = 0$,*

$$_\theta\mathrm{t}_1(0) \equiv \cos\theta \quad _\theta\mathrm{t}_2(0) \equiv \sin\theta \quad _\theta\mathrm{t}_3(0) \equiv 0.$$

Proof. Write the vector $\vec{v} = \langle x, x\tan\theta, 0\rangle$ and then consider the definitions of both the LHS and the RHS. The full proof is left to the reader. □

Remark. Using this and substituting into the Sum-of-Squares Identity will give the realisation that the Sum-of-Squares Identity is indeed a generalised version of the trigonometric Pythagorean Identity, when $\Omega = 0$.

Theorem 2. *For any branch θ and solid angle Ω, we have the relations*

$$_\theta\mathrm{t}_1^2(\Omega)\sec^2\theta + {_\theta\mathrm{t}_3^2}(\Omega) = 1, \textit{and}$$

$$_\theta\mathrm{t}_2^2(\Omega)\csc^2\theta + {_\theta\mathrm{t}_3^2}(\Omega) = 1.$$

Proof. Substituting the definition of branches, $\mathrm{t}_2 = \mathrm{t}_1\tan\theta$, into the Sum-of-Squares Identity gives the conclusion easily. □

Theorem 3. *For any branch θ,*

$$\frac{_\theta\mathrm{t}_1^2}{\cos^2\theta} + \frac{_\theta\mathrm{t}_2^2}{\sin^2\theta} = {_\theta\mathrm{t}_1^2} + {_\theta\mathrm{t}_2^2}.$$

Proof. Sum both equations of Theorem 2 and write $2 = 2(\mathrm{t}_1^2 + \mathrm{t}_2^2 + \mathrm{t}_3^2)$ (due to the Sum-of-Squares Identity). Simplify the equation to yield the desired result. □

Theorem 4 (Complementary branches)**.** *For any branch θ and given solid angle Ω, we have*

$$_\theta\mathrm{t}_1(\Omega) = {_{\theta_c}\mathrm{t}_2}(\Omega) \quad \textit{and} \quad _\theta\mathcal{T}_1(\Omega) = {_{\theta_c}\mathcal{T}_2}(\Omega),$$

where θ_c is the complement of angle θ.

Proof. Simply reflect the geometric interpretations of the LHS along the plane $x = y$ to obtain the RHS. □

B. Formula for the Solid Angle

In Section II.ii, we investigated the formula for the solid angle, and used L'Huilier's formula to deduce that, for any vector $\vec{V}$, the corresponding solid angle is given by

$$4\tan^{-1}\sqrt{\tan\frac{s}{2}\tan\frac{s-a}{2}\tan\frac{s-b}{2}\tan\frac{s-c}{2}},$$

where $a = \cos^{-1}x$, $b = \cos^{-1}y$, $c = \pi/2$ and $s = \frac{a+b+c}{2}$. We can write the formula down in full. Let $V = (x, y, z)$. Direct substitution yields

$$\Omega = 4\tan^{-1}\sqrt{\tan\left(\frac{\cos^{-1}x+\cos^{-1}y}{2}+\frac{\pi}{4}\right)\tan\left(\frac{\cos^{-1}x}{2}+\frac{\pi}{4}\right)\tan\left(\frac{\cos^{-1}x+\cos^{-1}y}{2}\right)\tan\left(\frac{\cos^{-1}y}{2}+\frac{\pi}{4}\right)},$$

which is the full formula we desire.

References

[Alvarez, n.d.] Alvarez, A. S. (n.d.). Note on an n-dimensional Pythagorean theorem. *Carnegie Mellon University*, Centre for Nonlinear Analysis and Department of Mathematical Sciences.

[Boyer, 1991] Boyer, C. B. (1991). *A History of Mathematics (2nd edition).* New York: John Wiley & Sons, Inc. ISBN 0-471-54397-7.

[Inglis, 2015] Inglis, M. (2015). *Astrophysics is Easy! An Introduction for the Amateur Astronomer, 2nd ed.* Long Island, New York: Springer.

[Odenwald, n.d.] Odenwald, S. (n.d.). How long does it take light to get out from the inside of the Sun? Retrieved from https://image.gsfc.nasa.gov/poetry/ask/a11354.html.

[Zwillinger, 1995] Zwillinger, D. (1995). *CRC Standard Mathematical Tables and Formulae, 31st ed.* Boca Raton, Florida: CRC Press.