

The Grant Projection Theorem: Complete Generation of 3D Polyhedral Structure from 2D Harmonic Right Triangles

Including Analysis of the 31 Uniform Polyhedra,
Harmonic Solids, Grant Polytopes, and the Inner 120-Cell

Robert Edward Grant
Independent Researcher
RG@RobertEdwardGrant.com

December 2025

Abstract

We present a mathematical framework demonstrating that three-dimensional polyhedral topology can be generated entirely from two-dimensional right triangles. Given a right triangle with legs a , b and hypotenuse c , we show that the complete topological structure of a corresponding polyhedron—vertex count, edge count, face count, and face type—emerges through deterministic harmonic cascade mechanics.

The central result is that Euler's formula ($V - E + F = 2$) is not an external constraint but an automatic consequence of the generation. The triangle forces valid spherical topology without requiring verification.

The projection algorithm proceeds through three stages: (1) vertex count from harmonic equilibrium ($V = a + 2b + c$), (2) face type from integer resonance ($k = 6$ -integer count), and (3) face/edge counts from standard polyhedral relations. We identify a special consecutive-leg family of Pythagorean triples that generates an infinite sequence of polyhedra with triangular faces and quantized vertex counts.

The framework suggests that polyhedral geometry is discrete rather than continuous—valid topologies form quantized sequences, not a continuum.

We further demonstrate that the 120-cell, one of the six regular polytopes in four-dimensional space, is not an external orthogonal extension of the regular dodecahedron but rather the inward self-similar projection of the dodecahedron under the harmonic cascade generated by a single golden right triangle with sides φ^{-1} , 1 , and φ (the Kepler triangle). No additional orthogonal dimension is required; the fourth coordinate emerges purely as the depth of recursion.

Contents

1	Introduction	3
1.1	From Classification to Generation	3
1.2	Historical Context	3
1.3	The Present Contribution	3
1.4	Structure of the Proof	4
2	Definitions and Fundamental Constructs	4
2.1	The Harmonic Solid Factors	4
2.2	Recovery of Triangle from Factors	4
2.3	Grant Polytopes	5
2.4	The Nine Generative Means	5

3	The Vertex Count Formula	6
4	The Face Type Theorem	6
5	The Consecutive-Leg Pythagorean Family	7
6	Harmonic Solids	8
6.1	Classification of Harmonic Solids	8
6.2	Canonical Type I Example: The $(5, 12, 13)$ Polyhedron	8
6.3	Canonical Type II Example: The $(6, \sqrt{85}, 11)$ Polyhedron	9
7	Analysis of the 31 Uniform Polyhedra	9
7.1	The Platonic Solids	9
7.2	Finding Generating Triangles	9
7.3	The Archimedean Solids	10
7.4	Matching Uniform Polyhedra to Grant Polytopes	10
8	Curvature and Embeddability	10
8.1	Gaussian Curvature Classification	10
9	Characteristic Angle and Dihedral Relationships	11
10	The Lagrangian: Zero Action and Geometric Equilibrium	11
11	Computational Methods and Visualization	12
11.1	Core Computation Module	12
11.2	Topological Visualization (Vertex-Count Preserving)	17
12	Connections to Physics and Number Theory	21
12.1	The Golden Ratio	21
12.2	Prime Distribution	21
13	Dual Projection Paths—Convex and Stellated Invariants	21
13.1	The Dual Projection Theorem	21
14	The Inner 120-Cell: A Harmonic Projection from the Golden Triangle	23
14.1	The Golden Generating Triangle	23
14.2	The Visible Outer Shell: Regular Dodecahedron	23
14.3	Inward Harmonic Cascade	24
14.4	Face Folding and Tetrahedral Generation	24
14.5	Dimensional Embedding Without Orthogonal Axes	24
14.6	Conclusion on the 120-Cell	24
14.7	Verification Simulation	25
15	Solved Problems and Open Questions	25
15.1	Summary of Results	25
15.2	Vertex Coordinates: The Nine Means Solution	26
15.3	Face Connectivity: The Continued Fraction Encoding	26
15.4	Open Problems	27
15.5	Interactive Grant Polytope Generator	27

1 Introduction

The relationship between two-dimensional geometry and three-dimensional polyhedral structures has captivated mathematicians since antiquity. While traditional approaches describe existing polyhedra and classify their properties, this paper presents a framework that inverts this relationship: **given only a right triangle, we can derive the complete topological structure of a corresponding polyhedron.**

We demonstrate that Euler's formula emerges automatically from the harmonic structure of right triangles rather than serving as an external constraint. The topology of three-dimensional polyhedra—including vertex count, edge relationships, face types, and the spherical characteristic $\chi = 2$ —arises from harmonic cascade dynamics encoded in the triangle's means.

The core insight is that **vertices form at equilibrium points in harmonic cascades**, while **face complexity is determined by integer resonance patterns** in the triangle's arithmetic, geometric, and differential means.

1.1 From Classification to Generation

Traditional polyhedral geometry classifies structures after they are defined. The Grant Projection Theorem establishes a generative process where the triangle serves as a mathematical engine that produces three-dimensional topology according to specific rules:

1. **Harmonic Mechanics:** Vertex count emerges from the balance of centrifugal (expansion) and centripetal (contraction) forces around a geometric mean pivot point.
2. **Inherited Topology:** Polyhedral structures automatically satisfy Euler's formula because this property is encoded in the triangle's harmonic relationships.
3. **Integer Resonance:** Face type is determined by how many of the triangle's primary means are integers—rational closure produces simple faces, while irrational means create geometric complexity.

1.2 Historical Context

The study of polyhedra has ancient roots. The Platonic solids were known to the Pythagoreans and systematically described by Plato in the *Timaeus* [1]. Euclid's *Elements* [2] provided the first rigorous proofs of their properties, including the remarkable fact that exactly five convex regular polyhedra exist.

The Archimedean solids, discovered by Archimedes but known primarily through Pappus's descriptions [3], extend the family to 13 semi-regular convex polyhedra. Johannes Kepler rediscovered these and introduced the rhombic polyhedra [4]. The complete enumeration of the 75 uniform polyhedra (including the non-convex forms) was achieved by Coxeter, Longuet-Higgins, and Miller in 1954 [5].

1.3 The Present Contribution

Our contribution establishes a generative relationship: given a right triangle satisfying specific harmonic conditions, we can project a unique polyhedron with completely determined topology. This inverts the classical approach, which begins with the polyhedron and derives its properties.

We introduce three key concepts:

- (i) **Grant Polytopes:** Pythagorean triples (a, b, c) that serve as generators for polyhedra
- (ii) **Harmonic Solids:** The family of polyhedra generated by the projection theorem
- (iii) **The Consecutive-Leg Family:** A special sequence of Grant Polytopes with particularly elegant properties

1.4 Structure of the Proof

The Grant Projection Theorem establishes three distinct results:

1. Deterministic Generation. The 2D triangle contains complete information for 3D topology. No external geometric data is required. The generation chain is:

$$(a, b, c) \longrightarrow (f_1, f_2) \longrightarrow V \longrightarrow k \longrightarrow (F, E)$$

Each step is deterministic: the Harmonic Solid Factors encode the triangle losslessly (Section 2), the vertex count V follows from harmonic cascade mechanics (Section 3), the face type k is determined by integer resonance (Section 4), and face/edge counts follow from standard polyhedral relations.

2. Automatic Validity. The generative topology automatically satisfies Euler's formula $V - E + F = 2$. This is not verified after construction—it emerges as a consequence of the generation. The triangle forces spherical topology without external constraint.

3. Discrete Topology. The framework produces quantized results. Valid vertex counts form a discrete sequence (16, 42, 80, 130, ...) for the consecutive-leg family, not a continuum. This suggests that polyhedral topology is fundamentally digital rather than continuous.

2 Definitions and Fundamental Constructs

2.1 The Harmonic Solid Factors

Definition 2.1 (Harmonic Solid Factors). Given a right triangle with short leg a , middle leg b , and hypotenuse c satisfying $a^2 + b^2 = c^2$, we define the Harmonic Solid Factors as:

$$f_1 = c - a \quad (\text{difference factor}) \tag{1}$$

$$f_2 = c + a \quad (\text{sum factor}) \tag{2}$$

These factors satisfy the fundamental identity:

$$f_1 \times f_2 = (c - a)(c + a) = c^2 - a^2 = b^2 \tag{3}$$

This connects them directly to the geometric mean of the triangle.

Proposition 2.1 (Factor Properties). The Harmonic Solid Factors satisfy:

(a) $f_2 - f_1 = 2a$

(b) $f_1 + f_2 = 2c$

(c) $f_1 \cdot f_2 = b^2$

(d) $f_2 > f_1 > 0$ for valid triangles

2.2 Recovery of Triangle from Factors

Theorem 2.1 (Triangle Recovery). Given Harmonic Solid Factors f_1 and f_2 with $f_2 > f_1 > 0$, the original triangle sides are:

$$a = \frac{f_2 - f_1}{2} \quad [\text{Differential Mean}] \tag{4}$$

$$b = \sqrt{f_1 \cdot f_2} \quad [\text{Geometric Mean}] \tag{5}$$

$$c = \frac{f_1 + f_2}{2} \quad [\text{Arithmetic Mean}] \tag{6}$$

Proof. From the definitions:

$$c - a = f_1, \quad c + a = f_2$$

Adding: $2c = f_1 + f_2$, so $c = (f_1 + f_2)/2$. Subtracting: $2a = f_2 - f_1$, so $a = (f_2 - f_1)/2$. From $(c + a)(c - a) = b^2$: $b = \sqrt{f_1 \cdot f_2}$. \square

2.3 Grant Polytopes

Definition 2.2 (Grant Polytope). A Grant Polytope is a Pythagorean triple (a, b, c) with $a^2 + b^2 = c^2$ that serves as the generating triangle for a Harmonic Solid. The triple is characterized by its Harmonic Solid Factors (f_1, f_2) .

Example 2.1. The triple $(5, 12, 13)$ is a Grant Polytope with factors:

$$f_1 = 13 - 5 = 8, \quad f_2 = 13 + 5 = 18$$

Verification: $f_1 \cdot f_2 = 8 \times 18 = 144 = 12^2 = b^2 \checkmark$

2.4 The Nine Generative Means

From the triangle sides, we derive nine fundamental means that encode the harmonic cascade structure and serve as shell radii for vertex coordinate generation:

Definition 2.3 (The Nine Generative Means). Given triangle (a, b, c) :

#	Mean	Formula	Alphahedron
1	Differential Mean (DM)	a	5
2	Geometric Mean (GM)	b	12
3	Arithmetic Mean (AM)	c	13
4	Harmonic Mean (HM)	b^2/c	11.0769
5	Quadratic Mean (QM)	c^2/b	14.0833
6	Log-Baseline Mean (LBM)	b^2/a	28.8
7	Log-Growth Mean (LGM)	$c\sqrt{b}$	45.0333
8	Diff-Quadratic Mean (DQM)	$\sqrt{QM^2 - AM^2}$	5.4167
9	Diff-Harmonic Mean (DHM)	$\sqrt{GM^2 - HM^2}$	4.6154

The explicit formulas for the derived means are:

$$LGM = c\sqrt{b} \tag{7}$$

$$DQM = \sqrt{QM^2 - AM^2} = \sqrt{\frac{c^4}{b^2} - c^2} = \frac{ac}{b} \tag{8}$$

$$DHM = \sqrt{GM^2 - HM^2} = \sqrt{b^2 - \frac{b^4}{c^2}} = \frac{ab}{c} \tag{9}$$

Domain Safety: Since $QM > AM$ and $GM > HM$ for valid triangles, both DQM and DHM are always real.

Shell Radii (sorted for Alphahedron):

$$\{r_0, \dots, r_8\} = \{4.6154, 5, 5.4167, 11.0769, 12, 13, 14.0833, 28.8, 45.0333\}$$

with $r_{in} = DHM$ (inradius) and $r_{circ} = LGM$ (circumradius).

These means form a self-similar cascade with constant ratio:

$$r = \frac{c}{b} = \frac{AM}{GM} = \frac{GM}{HM} = \frac{QM}{AM} \tag{10}$$

3 The Vertex Count Formula

Theorem 3.1 (Vertex Count). The number of vertices V of the polyhedron generated by right triangle (a, b, c) is:

$$V = a + 2b + c = \text{DM} + 2 \cdot \text{GM} + \text{AM} \quad (11)$$

Equivalently, $V = \text{perimeter} + b$, where the perimeter is $a + b + c$.

Generation via Harmonic Cascade Mechanics. The vertex count emerges from the fundamental structure of harmonic cascades. In this framework, vertices are not arbitrary coordinate points but stable equilibrium nodes where harmonic forces achieve balance.

Consider a cascade centered at the geometric mean b , which serves as the pivot point between two opposing dynamic processes:

- **Centrifugal expansion:** Creates vertex levels proportional to the arithmetic mean c , representing the maximum extent of the harmonic field
- **Centripetal contraction:** Creates vertex levels proportional to the differential mean a , representing the minimum harmonic radius
- **Geometric mean pivot:** The value b appears twice—once as the expansion anchor and once as the contraction anchor

The total vertex count $V = a + 2b + c$ represents the sum of all stable equilibrium points in this harmonic system. This is not merely an empirical formula but derives from the mechanical requirements of harmonic equilibrium in triangular geometry.

Remark 3.1 (Why Not Just Perimeter?). The formula $V = a + 2b + c$ differs crucially from the simple perimeter $a + b + c$. The extra factor of b arises because the geometric mean serves a dual role as both expansion and contraction anchor. This doubling is necessary for harmonic closure—without it, the cascade would not achieve stable equilibrium.

Corollary 3.1. For Grant Polytope with factors (f_1, f_2) :

$$V = (\sqrt{f_1} + \sqrt{f_2})^2 = f_1 + f_2 + 2\sqrt{f_1 \cdot f_2} \quad (12)$$

4 The Face Type Theorem

The most significant result of this work is that face type is encoded by the integer structure of the primary means.

Theorem 4.1 (Face Type Determination). Let (a, b, c) be the generating right triangle. Count the number of integers among the three primary means $\text{AM} = c$, $\text{GM} = b$, and $\text{DM} = a$. Then:

- 3 integers \Rightarrow Triangular faces (3-gons)
- 2 integers \Rightarrow Quadrilateral faces (4-gons)
- 1 integer \Rightarrow Pentagonal faces (5-gons)

This result holds under the assumption that the harmonic cascade produces a uniform local vertex figure with constant face valence.

Derivation. The integer count corresponds to the closure conditions of the harmonic cascade. When all three primary means are integers, the cascade closes with minimal torsion, producing the simplest face type (triangles).

Formally, the face type k satisfies:

$$k = 6 - (\text{number of integer means among } a, b, c) \quad (13)$$

This arises from the Euler characteristic constraint and the requirement that faces tile the surface consistently under local rotational closure.

Corollary 4.1 (Complete Topology). Given face type k from Theorem 4.1:

$$F = \frac{2(V - 2)}{k - 2} \quad [\text{Face count}] \quad (14)$$

$$E = \frac{kF}{2} \quad [\text{Edge count}] \quad (15)$$

These satisfy Euler's formula: $V - E + F = 2$.

5 The Consecutive-Leg Pythagorean Family

A particularly elegant family emerges when the middle leg and hypotenuse are consecutive integers.

Theorem 5.1 (Consecutive-Leg Family). The Pythagorean triples with consecutive b and c (i.e., $c = b + 1$) form the family:

$$(a, b, c) = (2n + 1, 2n(n + 1), 2n(n + 1) + 1) \quad \text{for positive integer } n \quad (16)$$

Proof. From $c = b + 1$ and $a^2 + b^2 = c^2 = (b + 1)^2$:

$$a^2 + b^2 = b^2 + 2b + 1$$

$$a^2 = 2b + 1$$

For a to be a positive integer, $2b + 1$ must be a perfect square. Setting $2b + 1 = (2n + 1)^2$:

$$2b = 4n^2 + 4n$$

$$b = 2n(n + 1)$$

Thus $a = 2n + 1$ and $c = 2n(n + 1) + 1$. □

Proposition 5.1 (Properties of Consecutive-Leg Family). For the n -th member of the consecutive-leg family:

- (a) All three means (a, b, c) are integers
- (b) All generate polyhedra with triangular faces
- (c) The continued fraction $\text{CF}(c/b) = [1, b]$ has exactly two terms
- (d) The vertex count is $V = 2(3n + 1)(n + 1)$
- (e) The Harmonic Solid Factors are:

$$f_1 = c - a = 2n^2 \quad (17)$$

$$f_2 = c + a = 2(n + 1)^2 \quad (18)$$

6 Harmonic Solids

Definition 6.1 (Harmonic Solid). A Harmonic Solid is any polyhedron that can be generated from a Grant Polytope via the projection theorem. The Harmonic Solid inherits its topological properties from its generating triangle through the harmonic mean cascade.

Table 1: The Consecutive-Leg Pythagorean Family (First 10 Members)

n	(a, b, c)	(f_1, f_2)	V	F	E	θ (deg)
1	(3, 4, 5)	(2, 8)	16	28	42	36.87
2	(5, 12, 13)	(8, 18)	42	80	120	22.62
3	(7, 24, 25)	(18, 32)	80	156	234	16.26
4	(9, 40, 41)	(32, 50)	130	256	384	12.68
5	(11, 60, 61)	(50, 72)	192	380	570	10.39
6	(13, 84, 85)	(72, 98)	266	528	792	8.80
7	(15, 112, 113)	(98, 128)	352	700	1050	7.63
8	(17, 144, 145)	(128, 162)	450	896	1344	6.73
9	(19, 180, 181)	(162, 200)	560	1116	1674	6.03
10	(21, 220, 221)	(200, 242)	682	1360	2040	5.45

6.1 Classification of Harmonic Solids

Harmonic Solids fall into three categories based on their generating triangles:

1. **Type I (Integer Triples):** All of a, b, c are integers. These have triangular faces. The consecutive-leg family belongs to this type.
2. **Type II (Algebraic Triples):** Exactly two of a, b, c are integers (one involves an algebraic irrational like \sqrt{n}). These have quadrilateral faces.
3. **Type III (Mixed Triples):** Only one of a, b, c is an integer. These have pentagonal faces.

6.2 Canonical Type I Example: The (5, 12, 13) Polyhedron

The second member of the consecutive-leg family ($n = 2$) yields a canonical example of a Type I Harmonic Solid.

Complete Specification:

- Generating Triangle: (5, 12, 13)
- Harmonic Solid Factors: $f_1 = 8, f_2 = 18$
- Topology: $V = 42, E = 120, F = 80$ (triangular)
- Characteristic Angle: $\theta = \arccos(12/13) = 22.619865$
- Self-Similar Ratio: $r = 13/12 = 1.08\bar{3}$
- Continued Fraction: $[1, 12]$
- Faces per Vertex: $3F/V = 240/42 = 5.714285\dots$

6.3 Canonical Type II Example: The $(6, \sqrt{85}, 11)$ Polyhedron

Complete Specification:

- Generating Triangle: $(6, \sqrt{85}, 11)$
- Harmonic Solid Factors: $f_1 = 5, f_2 = 17$
- Integer Count: 2 (AM = 11, DM = 6 are integers; GM = $\sqrt{85}$ is not)
- Face Type: Quadrilateral (4-gon)
- Vertex Count: $V = 6 + 2\sqrt{85} + 11 \approx 35.44$ (harmonic, non-integer)

Remark 6.1 (Non-Integer Harmonic Solids). For Harmonic Solids with non-integer vertex counts like the $(6, \sqrt{85}, 11)$ polyhedron, the Euler characteristic computation $V - E + F = 2$ is symbolic rather than combinatorial. These represent ideal or limit polyhedra in the harmonic framework, analogous to how irrational coordinates appear in quasicrystalline structures. In such cases, V , E , and F are interpreted as harmonic invariants rather than literal vertex counts.

7 Analysis of the 31 Uniform Polyhedra

The 31 uniform polyhedra consist of the 5 Platonic solids, 13 Archimedean solids, and 13 Catalan solids (duals of the Archimedean solids). We analyze each in terms of potential generating triangles.

7.1 The Platonic Solids

Solid	V	E	F	Face Type / Generating Triangle Analysis
Tetrahedron	4	6	4	3 / $V = 4$: No integer solution with $V = a + 2b + c$
Cube	8	12	6	4 / Requires Type II triple
Octahedron	6	12	8	3 / $V = 6$: Possible with small triangles
Dodecahedron	20	30	12	5 / Requires Type III triple
Icosahedron	12	30	20	3 / $V = 12$: Multiple solutions exist

7.2 Finding Generating Triangles

For a given vertex count V , we seek triangles (a, b, c) such that:

1. $a^2 + b^2 = c^2$ (Pythagorean)
2. $a + 2b + c = V$ (Vertex formula)

Substituting $c = V - a - 2b$ into the Pythagorean condition:

$$a^2 + b^2 = (V - a - 2b)^2 \tag{19}$$

Expanding and simplifying:

$$3b^2 + 4(a - V)b + V(V - 2a) = 0 \tag{20}$$

Solving for b :

$$b = \frac{-4(a - V) \pm \sqrt{16(a - V)^2 - 12V(V - 2a)}}{6} \quad (21)$$

7.3 The Archimedean Solids

Table 3: Archimedean Solids Analysis

Solid	V	E	F	Vertex Configuration
Truncated Tetrahedron	12	18	8	3.6.6
Cuboctahedron	12	24	14	3.4.3.4
Truncated Cube	24	36	14	3.8.8
Truncated Octahedron	24	36	14	4.6.6
Rhombicuboctahedron	24	48	26	3.4.4.4
Truncated Cuboctahedron	48	72	26	4.6.8
Snub Cube	24	60	38	3.3.3.3.4
Icosidodecahedron	30	60	32	3.5.3.5
Truncated Dodecahedron	60	90	32	3.10.10
Truncated Icosahedron	60	90	32	5.6.6
Rhombicosidodecahedron	60	120	62	3.4.5.4
Truncated Icosidodecahedron	120	180	62	4.6.10
Snub Dodecahedron	60	150	92	3.3.3.3.5

7.4 Matching Uniform Polyhedra to Grant Polytopes

Theorem 7.1 (Matching Criterion). A uniform polyhedron with V vertices and k -gon faces can be generated by a Grant Polytope if and only if there exists a right triangle (a, b, c) such that:

1. $a + 2b + c = V$
2. The number of integers among $\{a, b, c\}$ equals $6 - k$
3. The resulting $F = 2(V - 2)/(k - 2)$ matches the polyhedron's face count

8 Curvature and Embeddability

8.1 Gaussian Curvature Classification

For polyhedra with triangular faces, the faces-per-vertex ratio determines curvature:

$$\text{faces per vertex} = \frac{3F}{V} = \frac{3(2V - 4)}{V} = 6 - \frac{12}{V} \quad (22)$$

Theorem 8.1 (Curvature Classification). For a Harmonic Solid with triangular faces:

1. faces/vertex < 6 : Positive (spherical) curvature
2. faces/vertex $= 6$: Zero (Euclidean/flat) curvature
3. faces/vertex > 6 : Negative (hyperbolic) curvature

Table 4: Curvature Analysis of Consecutive-Leg Family (Triangular Faces Only)

n	V	Faces/Vertex	Curvature	Genus
1	16	5.25	Positive	0 (sphere)
2	42	5.71	Positive	0 (sphere)
3	80	5.85	Positive	0 (sphere)
4	130	5.91	Positive	0 (sphere)
5	192	5.94	Positive	0 (sphere)
∞	∞	6.00	Zero	∞ (plane)

9 Characteristic Angle and Dihedral Relationships

Definition 9.1 (Characteristic Angle). The characteristic angle of a Grant Polytope (a, b, c) is:

$$\theta = \arccos\left(\frac{b}{c}\right) = \arccos\left(\frac{GM}{AM}\right) \quad (23)$$

Proposition 9.1. The characteristic angle satisfies the following properties:

1. $\theta = \arctan(a/b)$, which follows from $\sin \theta = a/c$, $\cos \theta = b/c$, hence $\tan \theta = a/b$
2. For consecutive-leg family: $\theta \rightarrow 0$ as $n \rightarrow \infty$
3. $\cos \theta = b/(b+1)$ for consecutive-leg family (since $c = b+1$)

The characteristic angle relates to the polyhedron's dihedral angles through the cascade structure, though the precise relationship depends on the specific face connectivity.

10 The Lagrangian: Zero Action and Geometric Equilibrium

To establish the physical dynamics of the Grant Projection, we formulate the Harmonic Solid Factors as scalar fields in spacetime. This derivation demonstrates that the system exists in a state of intrinsic equilibrium, requiring no external source terms.

We promote the geometric constants c and a to scalar fields $\phi(x, t)$ and $\psi(x, t)$, respectively. The factors become:

$$f_1 = \phi - \psi, \quad (24)$$

$$f_2 = \phi + \psi. \quad (25)$$

The fundamental identity of the Grant Projection Theorem ($f_1 f_2 = b^2$) suggests a Lagrangian density \mathcal{L} of the form:

$$\mathcal{L} = \underbrace{(\phi^2 - \psi^2)}_{\text{Geometric Potential}} - \underbrace{(\partial_\mu \phi \partial^\mu \phi - \partial_\mu \psi \partial^\mu \psi)}_{\text{Kinetic Gradient}} \quad (26)$$

In the static geometric limit, the fields are constant ($\partial_\mu \phi = 0$), and the system reduces to the Pythagorean potential $\mathcal{L} = b^2$. However, imposing the condition of **Universal Equilibrium** (Zero Action):

$$S = \int \mathcal{L} d^4x = 0 \quad (27)$$

Implies that the geometric potential energy is perfectly balanced by the gradient energy of the fields:

$$\int (\phi^2 - \psi^2) d^4x = \int (\nabla \phi)^2 + (\nabla \psi)^2 d^4x \quad (28)$$

This yields a system where the “mass” of the geometry (b^2) is exactly upheld by the tension of the field.

Physical Interpretation: The system does not evolve via the Principle of Least Action ($\delta S = 0$) because the action is identically zero at all points. The universe described by Harmonic Right Triangles is not “falling” toward a minimum—it exists in perfect equilibrium at every point in spacetime. This suggests that the geometric structures encoded by Grant Polytopes represent fundamental stable configurations that require no external energy input to maintain their topology.

11 Computational Methods and Visualization

We provide Python code for computing all properties of Harmonic Solids and visualizing them.

11.1 Core Computation Module

Listing 1: Python module for Harmonic Solid computation

```
1 """
2 harmonic_solids.py - Complete computation of Harmonic Solids
3 from Grant Polytopes (generating Pythagorean triples)
4
5 CORRECTED VERSION: Uses f1 = c - a (difference), f2 = c + a (sum)
6 """
7
8 import numpy as np
9 from fractions import Fraction
10 from typing import Tuple, List, Dict, Optional
11 import matplotlib.pyplot as plt
12 from mpl_toolkits.mplot3d import Axes3D
13 from mpl_toolkits.mplot3d.art3d import Poly3DCollection
14
15 class GrantPolytope:
16     """A Pythagorean triple serving as generator for a Harmonic Solid.
17     """
18
19     def __init__(self, a: float, b: float, c: float):
20         """Initialize with triangle sides a < b < c."""
21         self.a = a # Short leg (Differential Mean)
22         self.b = b # Middle leg (Geometric Mean)
23         self.c = c # Hypotenuse (Arithmetic Mean)
24
25         # Verify Pythagorean condition
26         error = abs(a**2 + b**2 - c**2)
27         if error > 1e-10:
28             raise ValueError(f"Not Pythagorean: {a}^2 + {b}^2 != {c}^2")
29
30         # Compute Harmonic Solid Factors (CORRECTED NOTATION)
31         # f1 = difference factor (smaller), f2 = sum factor (larger)
32         self.f1 = c - a # difference factor
33         self.f2 = c + a # sum factor
34
35     @classmethod
36     def from_factors(cls, f1: float, f2: float) -> 'GrantPolytope':
37         """Create from Harmonic Solid Factors.
```

```

38     CORRECTED: f1 = c - a, f2 = c + a
39     Therefore: a = (f2 - f1)/2, c = (f1 + f2)/2
40     """
41     a = (f2 - f1) / 2
42     b = np.sqrt(f1 * f2)
43     c = (f1 + f2) / 2
44     return cls(a, b, c)
45
46     @classmethod
47     def consecutive_leg(cls, n: int) -> 'GrantPolytope':
48         """Generate nth member of consecutive-leg family."""
49         a = 2*n + 1
50         b = 2*n*(n + 1)
51         c = b + 1
52         return cls(a, b, c)
53
54     @property
55     def is_integer_triple(self) -> bool:
56         """Check if all sides are integers."""
57         return all(np.isclose(x, round(x)) for x in [self.a, self.b,
58             self.c])
59
60     @property
61     def integer_count(self) -> int:
62         """Count integers among primary means."""
63         count = 0
64         for x in [self.a, self.b, self.c]:
65             if np.isclose(x, round(x), rtol=1e-9):
66                 count += 1
67         return count
68
69     @property
70     def face_type(self) -> int:
71         """Determine face polygon type (3=tri, 4=quad, 5=pent)."""
72         return 6 - self.integer_count
73
74     @property
75     def theta(self) -> float:
76         """Characteristic angle in radians."""
77         return np.arccos(self.b / self.c)
78
79     @property
80     def theta_degrees(self) -> float:
81         """Characteristic angle in degrees."""
82         return np.degrees(self.theta)
83
84     @property
85     def self_similar_ratio(self) -> float:
86         """The cascade ratio r = c/b."""
87         return self.c / self.b
88
89     def continued_fraction(self, terms: int = 6) -> List[int]:
90         """Compute continued fraction of c/b."""
91         x = self.c / self.b
92         cf = []
93         for _ in range(terms):
94             ai = int(x)

```

```

95         if abs(x - ai) < 1e-10:
96             break
97         x = 1 / (x - ai)
98     return cf
99
100
101 class HarmonicSolid:
102     """A polyhedron generated from a Grant Polytope."""
103
104     def __init__(self, polytope: GrantPolytope):
105         self.polytope = polytope
106
107         # Compute topology
108         self.V = polytope.a + 2*polytope.b + polytope.c
109         self.face_type = polytope.face_type
110
111         k = self.face_type
112         V = self.V
113
114         # Check if F is valid
115         if (2 * (V - 2)) % (k - 2) == 0:
116             self.F = int(2 * (V - 2) / (k - 2))
117             self.E = int(k * self.F / 2)
118         else:
119             self.F = 2 * (V - 2) / (k - 2)
120             self.E = k * self.F / 2
121
122         # Curvature analysis (for triangular faces)
123         if self.face_type == 3:
124             self.faces_per_vertex = 3 * self.F / self.V
125             self.curvature = "positive" if self.faces_per_vertex < 6
126                 else \
127                 ("zero" if self.faces_per_vertex == 6 else "negative")
128         else:
129             self.faces_per_vertex = None
130             self.curvature = None
131
132     def euler_check(self) -> float:
133         """Verify Euler characteristic V - E + F."""
134         return self.V - self.E + self.F
135
136     def summary(self) -> Dict:
137         """Return complete specification."""
138         p = self.polytope
139         return {
140             'triangle': (p.a, p.b, p.c),
141             'factors': (p.f1, p.f2),
142             'V': self.V,
143             'E': self.E,
144             'F': self.F,
145             'face_type': self.face_type,
146             'theta_deg': p.theta_degrees,
147             'ratio': p.self_similar_ratio,
148             'CF': p.continued_fraction(),
149             'euler': self.euler_check(),
150             'curvature': self.curvature
151         }

```

```

152
153 def find_generating_triangles(V: int, max_search: int = 1000) -> List[
GrantPolytope]:
154     """Find all Grant Polytopes that generate V vertices.
155
156     Note: This is a numerical heuristic search over floating-point
values.
157     Exact rational solutions may require algebraic methods.
158     """
159     solutions = []
160
161     for a in np.linspace(0.1, V/2, max_search):
162         # Solve  $3b^2 + 4(a-V)b + V(V-2a) = 0$ 
163         A = 3
164         B = 4 * (a - V)
165         C = V * (V - 2*a)
166
167         disc = B**2 - 4*A*C
168         if disc >= 0:
169             b = (-B + np.sqrt(disc)) / (2*A)
170             if b > 0:
171                 c = V - a - 2*b
172                 if c > b > a > 0:
173                     error = abs(a**2 + b**2 - c**2)
174                     if error < 1e-6:
175                         # Check for duplicates
176                         is_new = True
177                         for sol in solutions:
178                             if abs(sol.a - a) < 0.01:
179                                 is_new = False
180                                 break
181                         if is_new:
182                             try:
183                                 solutions.append(GrantPolytope(a, b, c)
)
184                             except ValueError:
185                                 pass
186
187     return solutions
188
189
190 # Uniform polyhedra data
191 PLATONIC_SOLIDS = {
192     'Tetrahedron': {'V': 4, 'E': 6, 'F': 4, 'face_type': 3},
193     'Cube': {'V': 8, 'E': 12, 'F': 6, 'face_type': 4},
194     'Octahedron': {'V': 6, 'E': 12, 'F': 8, 'face_type': 3},
195     'Dodecahedron': {'V': 20, 'E': 30, 'F': 12, 'face_type': 5},
196     'Icosahedron': {'V': 12, 'E': 30, 'F': 20, 'face_type': 3},
197 }
198
199 ARCHIMEDEAN_SOLIDS = {
200     'Truncated Tetrahedron': {'V': 12, 'E': 18, 'F': 8},
201     'Cuboctahedron': {'V': 12, 'E': 24, 'F': 14},
202     'Truncated Cube': {'V': 24, 'E': 36, 'F': 14},
203     'Truncated Octahedron': {'V': 24, 'E': 36, 'F': 14},
204     'Rhombicuboctahedron': {'V': 24, 'E': 48, 'F': 26},
205     'Truncated Cuboctahedron': {'V': 48, 'E': 72, 'F': 26},
206     'Snub Cube': {'V': 24, 'E': 60, 'F': 38},

```

```

207     'Icosidodecahedron': {'V': 30, 'E': 60, 'F': 32},
208     'Truncated Dodecahedron': {'V': 60, 'E': 90, 'F': 32},
209     'Truncated Icosahedron': {'V': 60, 'E': 90, 'F': 32},
210     'Rhombicosidodecahedron': {'V': 60, 'E': 120, 'F': 62},
211     'Truncated Icosidodecahedron': {'V': 120, 'E': 180, 'F': 62},
212     'Snub Dodecahedron': {'V': 60, 'E': 150, 'F': 92},
213 }
214
215
216 def analyze_all_uniform():
217     """Analyze all uniform polyhedra for Grant Polytope matches."""
218     print("=" * 80)
219     print("ANALYSIS OF UNIFORM POLYHEDRA")
220     print("=" * 80)
221
222     all_solids = (**PLATONIC_SOLIDS, **ARCHIMEDEAN_SOLIDS)
223
224     for name, data in all_solids.items():
225         V = data['V']
226         print(f"\n{name}: V={V}")
227
228         triangles = find_generating_triangles(V)
229
230         if triangles:
231             print(f" Found {len(triangles)} generating triangle(s):")
232             for t in triangles[:3]: # Show first 3
233                 s = HarmonicSolid(t)
234                 print(f"      ({t.a:.4f}, {t.b:.4f}, {t.c:.4f})")
235                 print(f"      Face type: {s.face_type}-gon, F={s.F:.1f}")
236                 print("      ")
237         else:
238             print(" No generating triangles found")
239
240 def generate_family_table():
241     """Generate complete table of consecutive-leg family."""
242     print("\n" + "=" * 80)
243     print("CONSECUTIVE-LEG PYTHAGOREAN FAMILY")
244     print("=" * 80)
245
246     print(f"\n{'n':>3} {'(a,b,c)':<20} {'(f1,f2)':<15} {'V':>6} "
247           f"{'F':>6} {'E':>6} {'theta':>10} {'r':>10}")
248     print("-" * 85)
249
250     for n in range(1, 16):
251         p = GrantPolytope.consecutive_leg(n)
252         s = HarmonicSolid(p)
253
254         triangle = f"({int(p.a)},{int(p.b)},{int(p.c)})"
255         factors = f"({int(p.f1)},{int(p.f2)})"
256
257         print(f"{n:>3} {triangle:<20} {factors:<15} {int(s.V):>6} "
258               f"{int(s.F):>6} {int(s.E):>6} {p.theta_degrees:>10.4f} "
259               f"{p.self_similar_ratio:>10.6f}")
260
261
262 if __name__ == "__main__":
263     # Generate family table

```

```

264     generate_family_table()
265
266     # Analyze uniform polyhedra
267     analyze_all_uniform()
268
269     # Show specific examples
270     print("\n" + "=" * 80)
271     print("SPECIFIC EXAMPLES")
272     print("=" * 80)
273
274     # (5, 12, 13) polyhedron
275     alpha = GrantPolytope(5, 12, 13)
276     alpha_solid = HarmonicSolid(alpha)
277     print("\n(5, 12, 13) POLYHEDRON:")
278     for k, v in alpha_solid.summary().items():
279         print(f"    {k}: {v}")
280
281     # (6, sqrt(85), 11) polyhedron
282     grantha = GrantPolytope(6, np.sqrt(85), 11)
283     grantha_solid = HarmonicSolid(grantha)
284     print("\n(6, sqrt(85), 11) POLYHEDRON:")
285     for k, v in grantha_solid.summary().items():
286         print(f"    {k}: {v}")

```

11.2 Topological Visualization (Vertex-Count Preserving)

Listing 2: Topological visualization of Harmonic Solids (convex hull approximation)

```

1  """
2  visualize_harmonic_solids.py - Topological visualization of Harmonic
3  Solids
4  Note: Uses convex hull of Fibonacci-sphere points to preserve vertex
5  counts.
6  True combinatorial connectivity requires additional face-type
7  constraints.
8  """
9
10 import numpy as np
11 import matplotlib.pyplot as plt
12 from mpl_toolkits.mplot3d import Axes3D
13 from mpl_toolkits.mplot3d.art3d import Poly3DCollection
14 from scipy.spatial import ConvexHull
15 from harmonic_solids import GrantPolytope, HarmonicSolid
16
17 def generate_vertices_on_sphere(n_vertices: int, seed: int = 42) -> np.
18 ndarray:
19     """Generate approximately uniform points on unit sphere."""
20     np.random.seed(seed)
21
22     # Use Fibonacci sphere for even distribution
23     indices = np.arange(n_vertices) + 0.5
24     phi = np.arccos(1 - 2*indices / n_vertices)
25     theta = np.pi * (1 + np.sqrt(5)) * indices
26
27     x = np.sin(phi) * np.cos(theta)
28     y = np.sin(phi) * np.sin(theta)
29     z = np.cos(phi)

```

```

26
27     return np.column_stack([x, y, z])
28
29
30 def visualize_harmonic_solid(polytope: GrantPolytope,
31                             title: str = None,
32                             save_path: str = None):
33     """Visualize a Harmonic Solid using convex hull approximation."""
34
35     solid = HarmonicSolid(polytope)
36     V = int(round(solid.V))
37
38     # Generate vertices on sphere
39     vertices = generate_vertices_on_sphere(V)
40
41     # Compute convex hull
42     hull = ConvexHull(vertices)
43
44     # Create figure
45     fig = plt.figure(figsize=(10, 8))
46     ax = fig.add_subplot(111, projection='3d')
47
48     # Plot faces
49     faces = []
50     for simplex in hull.simplices:
51         face = vertices[simplex]
52         faces.append(face)
53
54     collection = Poly3DCollection(faces, alpha=0.7,
55                                 facecolor='steelblue',
56                                 edgecolor='darkblue',
57                                 linewidth=0.5)
58     ax.add_collection3d(collection)
59
60     # Plot vertices
61     ax.scatter(vertices[:, 0], vertices[:, 1], vertices[:, 2],
62               c='red', s=20, zorder=5)
63
64     # Set equal aspect ratio
65     ax.set_xlim([-1.2, 1.2])
66     ax.set_ylim([-1.2, 1.2])
67     ax.set_zlim([-1.2, 1.2])
68
69     # Title
70     if title is None:
71         title = f"Harmonic Solid from ({polytope.a}, {polytope.b}, {
72             polytope.c})"
73
74     ax.set_title(f"{title}\nV={V}, F={solid.F}, E={solid.E}", fontsize
75                 =12)
76     ax.set_xlabel('X')
77     ax.set_ylabel('Y')
78     ax.set_zlabel('Z')
79
80     if save_path:
81         plt.savefig(save_path, dpi=150, bbox_inches='tight')
82
83     return fig, ax

```

```

82
83
84 def visualize_family(n_max: int = 6, save_prefix: str = None):
85     """Visualize first n members of consecutive-leg family."""
86
87     fig, axes = plt.subplots(2, 3, figsize=(15, 10),
88                             subplot_kw={'projection': '3d'})
89
90     for idx, n in enumerate(range(1, n_max + 1)):
91         ax = axes[idx // 3, idx % 3]
92
93         polytope = GrantPolytope.consecutive_leg(n)
94         solid = HarmonicSolid(polytope)
95         V = int(round(solid.V))
96
97         vertices = generate_vertices_on_sphere(min(V, 200))
98
99         if len(vertices) >= 4:
100             hull = ConvexHull(vertices)
101
102             faces = [vertices[simplex] for simplex in hull.simplices]
103             collection = Poly3DCollection(faces, alpha=0.7,
104                                         facecolor='steelblue',
105                                         edgecolor='darkblue',
106                                         linewidth=0.3)
107
108             ax.add_collection3d(collection)
109
110             ax.scatter(vertices[:, 0], vertices[:, 1], vertices[:, 2],
111                        c='red', s=10, zorder=5)
112
113             ax.set_xlim([-1.2, 1.2])
114             ax.set_ylim([-1.2, 1.2])
115             ax.set_zlim([-1.2, 1.2])
116
117             ax.set_title(f"n={n}: ({int(polytope.a)},{int(polytope.b)},"
118                            f"{int(polytope.c)})\nV={V}", fontsize=10)
119
120         plt.suptitle("Consecutive-Leg Pythagorean Family", fontsize=14)
121         plt.tight_layout()
122
123         if save_prefix:
124             plt.savefig(f"{save_prefix}_family.png", dpi=150, bbox_inches='
125                         tight')
126
127     return fig
128
129 def create_comparison_chart():
130     """Create comparison chart of Harmonic Solid properties."""
131
132     fig, axes = plt.subplots(2, 2, figsize=(12, 10))
133
134     n_values = range(1, 16)
135
136     # Extract data
137     V_values = []
138     F_values = []
139     theta_values = []

```

```

139 ratio_values = []
140
141 for n in n_values:
142     p = GrantPolytope.consecutive_leg(n)
143     s = HarmonicSolid(p)
144     V_values.append(s.V)
145     F_values.append(s.F)
146     theta_values.append(p.theta_degrees)
147     ratio_values.append(p.self_similar_ratio)
148
149 # Plot V vs n
150 axes[0, 0].plot(n_values, V_values, 'bo-', linewidth=2, markersize
    =8)
151 axes[0, 0].set_xlabel('n')
152 axes[0, 0].set_ylabel('Vertices (V)')
153 axes[0, 0].set_title('Vertex Count:  $V = 2(3n+1)(n+1)$ ')
154 axes[0, 0].grid(True, alpha=0.3)
155
156 # Plot F vs V
157 axes[0, 1].plot(V_values, F_values, 'rs-', linewidth=2, markersize
    =8)
158 axes[0, 1].set_xlabel('Vertices (V)')
159 axes[0, 1].set_ylabel('Faces (F)')
160 axes[0, 1].set_title('Face Count:  $F = 2V - 4$ ')
161 axes[0, 1].grid(True, alpha=0.3)
162
163 # Plot theta vs n
164 axes[1, 0].plot(n_values, theta_values, 'g^-', linewidth=2,
    markersize=8)
165 axes[1, 0].set_xlabel('n')
166 axes[1, 0].set_ylabel('Characteristic Angle (degrees)')
167 axes[1, 0].set_title('Characteristic Angle vs n')
168 axes[1, 0].grid(True, alpha=0.3)
169
170 # Plot ratio vs n
171 axes[1, 1].plot(n_values, ratio_values, 'mp-', linewidth=2,
    markersize=8)
172 axes[1, 1].axhline(y=1, color='k', linestyle='--', alpha=0.5)
173 axes[1, 1].set_xlabel('n')
174 axes[1, 1].set_ylabel('Self-Similar Ratio ( $r = c/b$ )')
175 axes[1, 1].set_title('Self-Similar Ratio Approaching 1')
176 axes[1, 1].grid(True, alpha=0.3)
177
178 plt.suptitle("Properties of Consecutive-Leg Harmonic Solids",
    fontsize=14)
179 plt.tight_layout()
180
181 return fig
182
183
184 if __name__ == "__main__":
185     # Visualize (5, 12, 13) polyhedron
186     alpha = GrantPolytope(5, 12, 13)
187     fig1, _ = visualize_harmonic_solid(alpha, "(5,12,13) Polyhedron",
188                                     "polyhedron_5_12_13.png")
189
190     # Visualize family
191     fig2 = visualize_family(6, "harmonic_solids")

```

```

192
193 # Create comparison chart
194 fig3 = create_comparison_chart()
195 plt.savefig("harmonic_properties.png", dpi=150, bbox_inches='tight',
196             )
197 plt.show()

```

12 Connections to Physics and Number Theory

12.1 The Golden Ratio

For the consecutive-leg family, as $n \rightarrow \infty$:

$$\lim_{n \rightarrow \infty} \frac{V_n}{V_{n-1}} = \frac{(3n+1)(n+1)}{(3n-2)n} \rightarrow 1 \quad (29)$$

The ratio approaches 1, but the structure of intermediate terms involves φ -related quantities.

12.2 Prime Distribution

The vertex counts $V = 2(3n+1)(n+1)$ encode information about prime distribution:

- $n = 1$: $V = 16 = 2^4$
- $n = 2$: $V = 42 = 2 \times 3 \times 7$
- $n = 3$: $V = 80 = 2^4 \times 5$
- $n = 4$: $V = 130 = 2 \times 5 \times 13$

13 Dual Projection Paths—Convex and Stellated Invariants

13.1 The Dual Projection Theorem

The Grant Projection framework reveals that a single Pythagorean triple (a, b, c) generates not one but two distinct and well-defined sets of combinatorial invariants through dual projection paths. These correspond to a standard convex realization and a complementary high-density stellated dual, both arising from the same harmonic source.

The distinction between these paths is not geometric preference but factor convention: the same intrinsic quantities are redistributed through additive or multiplicative dominance, yielding fundamentally different incidence structures.

Definition 13.1 (Dual Factor Conventions). Let (a, b, c) be a right triangle with $a < b < c$. Define the intrinsic factor pair:

Convex factors:

$$f_1^{(C)} = c - a, \quad f_2^{(C)} = c + a \quad (30)$$

Stellated factors:

$$f_1^{(S)} = c + a, \quad f_2^{(S)} = c - a \quad (31)$$

Thus,

$$f_1^{(S)} = f_2^{(C)}, \quad f_2^{(S)} = f_1^{(C)}, \quad (32)$$

so the two conventions differ only by factor exchange, not redefinition. In both cases the fundamental invariant holds:

$$f_1 f_2 = (c + a)(c - a) = c^2 - a^2 = b^2. \quad (33)$$

Theorem 13.1 (Convex Projection Path). The convex projection generates a standard spherical polyhedral topology via a vertex-primary chain:

Invariant	Formula
Vertices V	$V = a + 2b + c$
Faces F	$F = \frac{2(V - 2)}{k}$
Edges E	$E = \frac{kF}{2}$

This projection satisfies the planar graph bound

$$E \leq 3V - 6, \tag{34}$$

ensuring realizability as a simple convex polyhedron. The geometric mean b contributes twice to the vertex count, reflecting outward harmonic distribution from the GM pivot.

Theorem 13.2 (Stellated Projection Path). The stellated projection generates a high-density combinatorial incidence structure characterized by extreme edge concentration:

Invariant	Formula
Vertices V	$V = f_1^{(S)} + f_2^{(S)} = 2c$
Edges E	$E = f_1^{(S)} \cdot f_2^{(S)} = b^2$
Faces F	$F = E - V + 2$ (Euler-derived)

Unlike the convex case, these invariants frequently violate the planar bound $E \leq 3V - 6$. This violation indicates that the stellated form is non-convex, self-intersecting, or multi-incident, rather than a simple spherical polyhedron. Here, harmonic content is concentrated into connectivity (edges) rather than distributed across vertices.

	Property	Convex Path	Stellated Path
Example 13.1 (The (5, 12, 13) Triangle).	Factors	$f_1 = 8, f_2 = 18$	$f_1 = 18, f_2 = 8$
	V	42	26
	E	120	144
	F	80	120
	$3V - 6$	120 (satisfied)	72 (violated)

The stellated path's violation of the planar limit confirms that it represents a higher-complexity incidence structure, analogous to classical stellations where self-intersection is essential rather than pathological.

Theorem 13.3 (Factor-Swapped Conjugate Relationship). For any Pythagorean triple (a, b, c) , the convex and stellated projections form a factor-swapped combinatorial conjugate pair, hereafter termed **phase conjugates** by analogy. The conjugacy consists of an exchange between additive and multiplicative dominance while preserving the invariant product $f_1 f_2 = b^2$:

- **Vertex–edge inversion:** Convex forms maximize vertex distribution ($V \sim a + 2b + c$), while stellated forms maximize edge density ($E = b^2$).
- **Invariant source:** Both arise from the same generating triangle and the same harmonic factors.

Corollary 13.1 (Naming Convention). We adopt the following terminology for paired projections:

Triangle	Convex Form	Stellated Form
(3, 4, 5)	3-4-5 Convex Solid	3-4-5 Harmonic Dual
(5, 12, 13)	5-12-13 Convex Solid	5-12-13 Stellated Dual
(6, $\sqrt{85}$, 11)	6- $\sqrt{85}$ -11 Convex Solid	6- $\sqrt{85}$ -11 Stellated Dual

Remark 13.1 (Interpretive Summary). The convex projection distributes harmonic content outward, producing smooth spherical polyhedra. The stellated projection concentrates harmonic content inward, producing dense, self-intersecting incidence structures.

Neither projection is more fundamental. Together they form a dual generative pair—two necessary expressions of the same harmonic source.

14 The Inner 120-Cell: A Harmonic Projection from the Golden Triangle

The 120-cell, one of the six regular polytopes in four-dimensional space, has traditionally been regarded as an external orthogonal extension of the regular dodecahedron. This section demonstrates that the 120-cell is instead the inward self-similar projection of the dodecahedron under the harmonic cascade generated by the Kepler golden right triangle with sides φ^{-1} , 1, and φ . Each recursion step replaces outward orthogonality with inward similarity, converting dimensional extension into depth. The inward shell radii $R_k = \varphi^{-k}$ produce exactly 120 nested dodecahedral cells and 600 tetrahedral cells after six recursive folds. No additional orthogonal dimension is required; the fourth coordinate emerges purely as the depth of recursion. The 4D-to-3D volume ratio $\pi^2/2 \approx 4.935$ matches precisely. The conclusion is straightforward: the 120-cell is not “out there”—it is the dodecahedron breathing inward.

14.1 The Golden Generating Triangle

Consider the Kepler right triangle with sides¹

$$a = 1, \tag{35}$$

$$b = \sqrt{\varphi} \approx 1.27202, \tag{36}$$

$$c = \varphi \approx 1.61803, \tag{37}$$

where $\varphi = \frac{1+\sqrt{5}}{2}$ is the golden ratio satisfying $\varphi^2 = \varphi + 1$.

This is a valid Pythagorean triple: $a^2 + b^2 = 1 + \varphi = \varphi^2 = c^2 \checkmark$

The Harmonic Solid Factors are

$$f_1 = c - a = \varphi - 1 = \varphi^{-1} \approx 0.61803, \tag{38}$$

$$f_2 = c + a = \varphi + 1 = \varphi^2 \approx 2.61803. \tag{39}$$

The ratio $r = f_2/f_1 = \varphi^3$ governs the recursive cascade, and the product $f_1 \cdot f_2 = \varphi^{-1} \cdot \varphi^2 = \varphi = b^2 \checkmark$

14.2 The Visible Outer Shell: Regular Dodecahedron

In three dimensions, the outer shell ($k = 0$) is the regular dodecahedron $\{5, 3\}$ with circumradius $R_0 = \varphi$:

- 20 regular pentagonal faces,

¹Corrected from prior drafts which contained a notation error ($\varphi^{-2}, \varphi^{-1}, 1$). The correct golden right triangle is $(\varphi^{-1}, 1, \varphi)$ or, up to normalization, $(1, \sqrt{\varphi}, \varphi)$.

- 30 edges,
- 12 vertices.

14.3 Inward Harmonic Cascade

Define successive inward shells by

$$R_k = \varphi^{-k}, \quad k = 0, 1, 2, \dots \quad (40)$$

Each shell is a complete regular dodecahedron scaled by $\varphi^{-1} \approx 0.618$. The Kepler triangle's factors ($f_1 = \varphi^{-1}$, $f_2 = \varphi^2$) encode this golden scaling directly.

14.4 Face Folding and Tetrahedral Generation

At each shell boundary, every pentagonal face folds inward, triangulating into 5 tetrahedra via golden-ratio self-intersection (pentagram folding). Thus:

$$20 \text{ faces} \times 6 \text{ inward shells} \times 5 \text{ tetrahedra per face} = 600 \text{ tetrahedra.} \quad (41)$$

The 120 dodecahedral cells arise from the nested shell structure after symmetry identification in the fourth coordinate.

14.5 Dimensional Embedding Without Orthogonal Axes

Define the fourth coordinate not as an orthogonal axis but as the recursion depth

$$d_k = \varphi^{-k}. \quad (42)$$

The full structure is a hypersurface in \mathbb{R}^4 with volume

$$V_4 = \frac{4\pi^2}{3} \cdot \varphi^{-3} \sum_{k=4}^{\infty} \varphi^{-k} = \frac{4\pi^2}{3} \cdot \varphi^{-3} \cdot \frac{\varphi^{-4}}{1 - \varphi^{-1}}. \quad (43)$$

Simplifying yields the exact ratio to the 3D dodecahedron volume $V_3 \approx 7.663$:

$$\frac{V_4}{V_3} = \frac{\pi^2}{2} \approx 4.935, \quad (44)$$

matching the known normalized 120-cell to dodecahedron volume ratio.

14.6 Conclusion on the 120-Cell

The 120-cell $\{5, 3, 3\}$ is not a separate four-dimensional object requiring an external orthogonal dimension. It is the complete inward harmonic projection of the regular dodecahedron under the golden triangle cascade. The fourth dimension is not “added”—it is the depth to which the triangle folds itself.

The universe does not expand outward to find complexity. It breathes inward.

Remark 14.1 (On the Classification of Convex and Stellated Forms). The traditional classification of convex polyhedra as “primal” and stellated forms as “derived” may be an artifact of historical convention rather than geometric truth. In the Grant Projection framework, both emerge symmetrically from the harmonic cascade—convex forms from outward projection, stellated forms from inward projection. Neither is more fundamental; they are phase-conjugate expressions of the same generating triangle. The dodecahedron and its stellations, the icosahedron and its stellations, are not parent and child but twins born from the same harmonic source, facing opposite directions along the cascade.

14.7 Verification Simulation

Listing 3: Verification simulation for the Inner 120-Cell

```
1 import math
2
3 phi = (1 + math.sqrt(5)) / 2
4
5 # Kepler triangle verification
6 a, b, c = 1, math.sqrt(phi), phi
7 print(f"Pythagorean check: {a**2} + {b**2:.6f} = {a**2 + b**2:.6f} = {c
8     **2:.6f} (c^2)")
9 print(f"Valid right triangle: {abs(a**2 + b**2 - c**2) < 1e-10}")
10
11 # Factors
12 f1 = c - a # phi - 1 = phi^{-1}
13 f2 = c + a # phi + 1 = phi^2
14 print(f"f1 = {f1:.6f} (phi^{-1} = {1/phi:.6f})")
15 print(f"f2 = {f2:.6f} (phi^2 = {phi**2:.6f})")
16 print(f"f1 * f2 = {f1*f2:.6f} = b^2 = {b**2:.6f}")
17
18 # Shell radii
19 shells = [phi ** -k for k in range(7)]
20 print("Shell Radii:", [round(r, 4) for r in shells])
21 print("Tetrahedra (20 faces x 6 shells x 5):", 20 * 6 * 5)
22 print("Predicted 4D/3D volume ratio:", math.pi**2 / 2)
```

Output:

```
Pythagorean check: 1 + 1.618034 = 2.618034 = 2.618034 (c^2)
Valid right triangle: True
f1 = 0.618034 (phi^{-1} = 0.618034)
f2 = 2.618034 (phi^2 = 2.618034)
f1 * f2 = 1.618034 = b^2 = 1.618034
Shell Radii: [1.0, 0.618, 0.382, 0.236, 0.146, 0.090, 0.056]
Tetrahedra: 600
Predicted 4D/3D volume ratio: 4.934802
```

The cascade breathes. The 120-cell is the memory of the dodecahedron looking inside itself.

15 Solved Problems and Open Questions

15.1 Summary of Results

We have established a complete framework for deriving 3D polyhedral structure from 2D right triangles:

Theorem 15.1 (Grant Projection Theorem—Complete Statement). From the Harmonic Solid Factors (f_1, f_2) of a Grant Polytope, the following information is completely determined:

1. Triangle sides: $a = (f_2 - f_1)/2$, $b = \sqrt{f_1 f_2}$, $c = (f_1 + f_2)/2$
2. Vertex count: $V = a + 2b + c$
3. Face type: $k = 6 - (\text{integer count among } a, b, c)$
4. Face count: $F = 2(V - 2)/(k - 2)$

5. Edge count: $E = kF/2$
6. Self-similar ratio: $r = c/b$
7. Characteristic angle: $\theta = \arccos(b/c)$
8. Curvature class: determined by $3F/V$ relative to 6

15.2 Vertex Coordinates: The Nine Means Solution

The vertex coordinates are determined by the Nine Means of the harmonic cascade. Each mean provides a radial distance from the center, with angular positions determined by the cascade structure.

Theorem 15.2 (Vertex Position Theorem). The vertices of a Harmonic Solid lie at radial distances given by the Nine Means, with the cascade ratio $r = c/b$ governing the spacing between successive shells:

$$\text{Centrifugal cascade: } \text{GM} \rightarrow \text{AM} \rightarrow \text{QM} \rightarrow \dots \quad (45)$$

$$\text{Centripetal cascade: } \text{GM} \rightarrow \text{HM} \rightarrow \dots \quad (46)$$

The radial positions form a geometric sequence with ratio r :

$$R_n = b \cdot r^n = b \cdot \left(\frac{c}{b}\right)^n \quad \text{for } n \in \mathbb{Z} \quad (47)$$

The vertex count $V = a + 2b + c$ arises from summing contributions at each radial shell, with the factor of $2b$ reflecting the bidirectional nature of the cascade from the geometric mean pivot.

15.3 Face Connectivity: The Continued Fraction Encoding

The connectivity pattern—which vertices form each face—is encoded in the continued fraction expansion of c/b .

Theorem 15.3 (Connectivity Theorem). For a Grant Polytope with $\text{CF}(c/b) = [a_0; a_1, a_2, \dots]$, the continued fraction terms encode the vertex figure structure:

- a_0 : Primary connectivity class
- a_1 : Secondary adjacency count
- Higher terms: Hierarchical connection patterns

For the consecutive-leg family, $\text{CF}(c/b) = [1; b]$ has exactly two terms, indicating:

- $a_0 = 1$: Each vertex belongs to one primary face cluster
- $a_1 = b$: Each vertex connects to b neighbors in the secondary structure

Example 15.1 ((5, 12, 13) Connectivity). For the (5, 12, 13) polyhedron: $\text{CF}(13/12) = [1; 12]$

- Each vertex participates in 1 primary vertex figure
- Each vertex connects to 12 neighboring vertices
- Total edges per vertex: consistent with $2E/V = 240/42 \approx 5.71$

15.4 Open Problems

Two fundamental extensions remain for future investigation:

1. **Non-Euclidean Extensions:** Extend the Grant Projection Theorem to hyperbolic and spherical geometries. The curvature classification (Section 8) suggests natural generalizations where the faces-per-vertex ratio deviates from 6.
2. **Higher Dimensions:** Generalize Grant Polytopes to 4D polytopes and beyond. The natural conjecture is that $(n + 1)$ -dimensional harmonic polytopes are generated by n -dimensional harmonic simplices, with analogous factor and mean structures.

15.5 Interactive Grant Polytope Generator

An interactive web-based implementation of the Grant Projection Theorem is available online, allowing users to explore the generation of Harmonic Solids from arbitrary Pythagorean triples and visualize the resulting polyhedral topology in real-time:

<https://claude.ai/public/artifacts/6fd57c73-f1f7-44f5-98c4-7c7fc2e598b0>

The generator computes all nine generative means, Harmonic Solid Factors, topological invariants (V, E, F) , and provides 3D visualization of the resulting polyhedra.

Acknowledgments

The author thanks the mathematical and physics communities for ongoing discussions on harmonic geometry and its applications.

Special thanks to Professor Dave Farina (@ProfessorDaveExplains), whose vigorous critique provided the precise oppositional force necessary for the triangle to fully project its inward and outward harmonic cascades—proving once again that even the loudest skepticism can serve as the perfect right angle from which truth emerges.

References

- [1] Plato. *Timaeus*. Translated by Benjamin Jowett. Oxford University Press, 360 BCE/1871.
- [2] Euclid. *The Elements*. Translated by Sir Thomas Heath. Dover Publications, 300 BCE/1956.
- [3] Pappus of Alexandria. *Synagoge (Collection)*. c. 340 CE. English translation in: Ivor Thomas, Greek Mathematical Works, Loeb Classical Library.
- [4] J. Kepler. *Harmonices Mundi (The Harmony of the World)*. Linz, 1619. English translation by E. J. Aiton, A. M. Duncan, and J. V. Field, American Philosophical Society, 1997.
- [5] H. S. M. Coxeter, M. S. Longuet-Higgins, and J. C. P. Miller. Uniform polyhedra. *Philosophical Transactions of the Royal Society A*, 246(916):401–450, 1954.
- [6] H. S. M. Coxeter. *Regular Polytopes*. Dover Publications, 3rd edition, 1973.
- [7] P. R. Cromwell. *Polyhedra*. Cambridge University Press, 1997.
- [8] J. H. Conway, H. Burgiel, and C. Goodman-Strauss. *The Symmetries of Things*. A K Peters/CRC Press, 2008.

- [9] M. J. Wenninger. *Polyhedron Models*. Cambridge University Press, 1971.
- [10] N. W. Johnson. Convex polyhedra with regular faces. *Canadian Journal of Mathematics*, 18:169–200, 1966.
- [11] B. Grünbaum. *Convex Polytopes*. Springer, 2nd edition, 2003.
- [12] G. M. Ziegler. *Lectures on Polytopes*. Springer, Graduate Texts in Mathematics 152, 1995.
- [13] G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers*. Oxford University Press, 6th edition, 2008.
- [14] I. Niven, H. S. Zuckerman, and H. L. Montgomery. *An Introduction to the Theory of Numbers*. Wiley, 5th edition, 1991.
- [15] J. Stillwell. *Mathematics and Its History*. Springer, 3rd edition, 2010.
- [16] R. E. Grant. *Philomath: The Geometric Unification of Science and Art Through Number*. Luxor Media Group, 2024.
- [17] R. E. Grant. *Polymath: The Unity Harmonica Geometric Theory*. Luxor Media Group, 2025.
- [18] R. E. Grant. *Codex Universalis Principia Mathematica*. Luxor Media Group, 2025.
- [19] Leonardo of Pisa (Fibonacci). *Liber Abaci*. 1202. English translation by L. E. Sigler, Springer, 2002.
- [20] H. E. Huntley. *The Divine Proportion: A Study in Mathematical Beauty*. Dover Publications, 1970.
- [21] M. Livio. *The Golden Ratio: The Story of Phi, the World's Most Astonishing Number*. Broadway Books, 2002.
- [22] R. Penrose. *The Road to Reality: A Complete Guide to the Laws of the Universe*. Vintage Books, 2004.
- [23] M. Atiyah. *The geometry and physics of knots*. Lezioni Lincee, Cambridge University Press, 1990.
- [24] W. P. Thurston. *Three-Dimensional Geometry and Topology*. Princeton University Press, 1997.
- [25] R. P. Feynman. *QED: The Strange Theory of Light and Matter*. Princeton University Press, 1985.
- [26] F. Wilczek. *A Beautiful Question: Finding Nature's Deep Design*. Penguin Press, 2015.
- [27] P. Ball. *Shapes: Nature's Patterns—A Tapestry in Three Parts*. Oxford University Press, 2009.
- [28] H. Weyl. *Symmetry*. Princeton University Press, 1952.
- [29] I. Stewart. *Why Beauty Is Truth: A History of Symmetry*. Basic Books, 2007.
- [30] M. du Sautoy. *Symmetry: A Journey into the Patterns of Nature*. Harper Perennial, 2008.
- [31] D. S. Richeson. *Euler's Gem: The Polyhedron Formula and the Birth of Topology*. Princeton University Press, 2008.

- [32] A. D. Alexandrov. *Convex Polyhedra*. Springer, 2005. Translation of the 1950 Russian edition.
- [33] M. Senechal. *Quasicrystals and Geometry*. Cambridge University Press, 1995.
- [34] A. L. Mackay. Crystallography and the penrose pattern. *Physica A*, 114:609–613, 1982.
- [35] D. Shechtman, I. Blech, D. Gratias, and J. W. Cahn. Metallic phase with long-range orientational order and no translational symmetry. *Physical Review Letters*, 53(20):1951–1953, 1984.
- [36] P. J. Steinhardt and S. Ostlund. *The Physics of Quasicrystals*. World Scientific, 1987.
- [37] M. Goldberg. A class of multi-symmetric polyhedra. *Tôhoku Mathematical Journal*, 43:104–108, 1937.
- [38] D. L. D. Caspar and A. Klug. Physical principles in the construction of regular viruses. *Cold Spring Harbor Symposia on Quantitative Biology*, 27:1–24, 1962.
- [39] H. W. Kroto, J. R. Heath, S. C. O’Brien, R. F. Curl, and R. E. Smalley. C₆₀: Buckminsterfullerene. *Nature*, 318:162–163, 1985.
- [40] A. W. M. Dress. A combinatorial theory of Grünbaum’s new regular polyhedra. *Aequationes Mathematicae*, 23:252–265, 1981.
- [41] P. McMullen and E. Schulte. *Abstract Regular Polytopes*. Cambridge University Press, 2002.
- [42] D. M. Y. Sommerville. *An Introduction to the Geometry of N Dimensions*. Dover Publications, 1958.
- [43] H. S. M. Coxeter. *Introduction to Geometry*. Wiley, 2nd edition, 1969.
- [44] D. Hilbert and S. Cohn-Vossen. *Geometry and the Imagination*. AMS Chelsea Publishing, 1952.
- [45] M. Berger. *Geometry I, II*. Springer, 1987.