THE RADIAL SYSTEM - A Geometric and Harmonic Framework for state Identity and Infinite Convergence. "STATE COMPUTER"

Inventor: Seven Abraham

Publication date: April 2-2026

ABSTRACT

A first of its kind computational system for generating, assigning, and enforcing unique identities to discrete states using angular transformation. The system computes identity values using derived angular relationships and enables deterministic interaction signatures between states. The invention includes a software-based engine, visualization interface, and interaction analysis module.

FIELD OF THE INVENTION

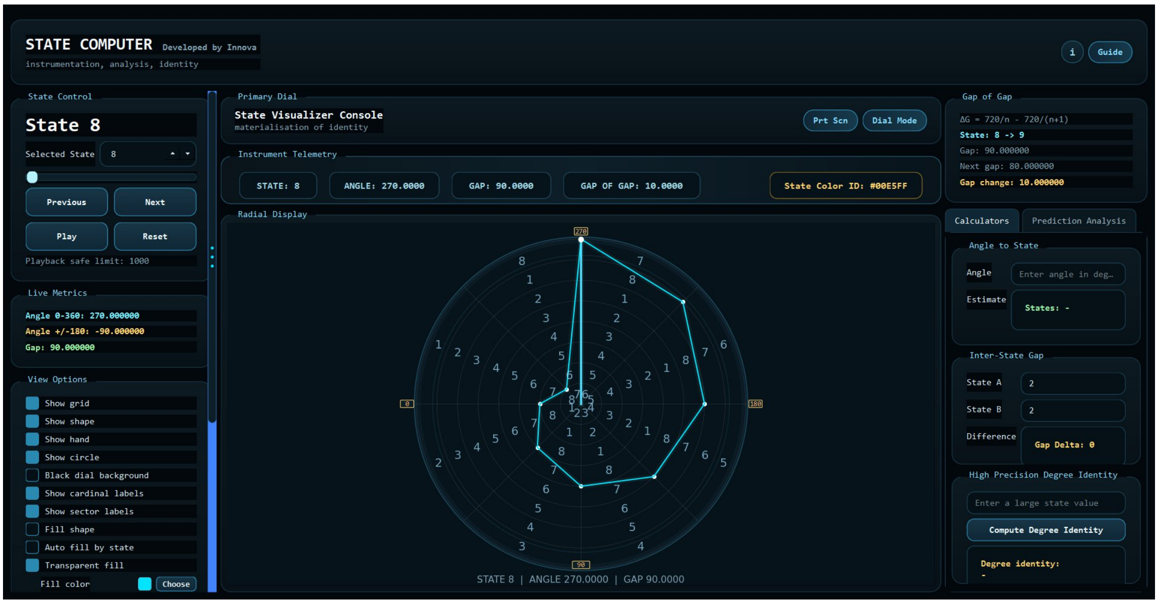
This invention relates to computational systems, identity encoding, mathematical modeling, and software-based analytical instruments.

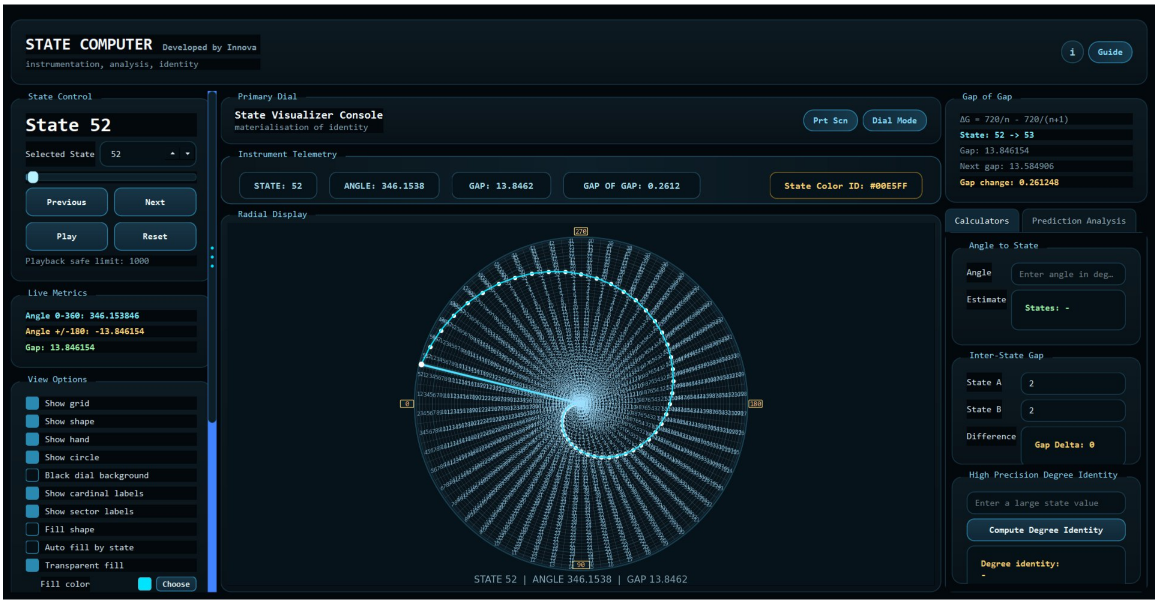
BACKGROUND

Existing systems rely on direct numerical comparison or hashing techniques for identity and interaction analysis. These approaches lack geometric identity representation and deterministic interaction uniqueness derived from angular transformation.

SUMMARY OF THE INVENTION

The present invention introduces a State Computer system that transforms discrete numerical states into angular identities (space). Each state is assigned a unique identity using a computational formula, and interactions between states produce unique signatures.





<u>System Interfaces</u>

Mathematics

In Summary, The system includes:

1. Identity Engine:
 Computes angular identity using:
 $G_n = 720/n$
 $\theta_n = 360 - (720/n)$

2. Interaction Engine:
 Computes interaction signature:
 $\Sigma(a,b) = |\theta_a - \theta\_b|$

3. Visualization Module:
 Displays radial representation of identities.

4. Control System:
 Enables state traversal, prediction, and telemetry.

APPLICATION

Prediction engine: In established physics and mathematics, achieving consistent accuracy in prediction remains a significant challenge. This limitation is not primarily due to the unavailability of techniques or a shortage of data, but rather arises from contextual deficiencies specifically what may be described as an *"error of ambiguity"* in the classification and categorization of data.
As simple as this may seem, it has been a long-standing issue across multiple scientific disciplines. This ambiguity plays a major role in why future events cannot be precisely determined or predicted, often referred to as the *"black box" nature* of reality and complex systems.

The State Computer, along with its associated extensions, addresses this limitation at a fundamental level. The system not only enables the accuracy at prediction but also allows such outcomes to be replicated and visualized within a dedicated virtual environment, even when operating with minimal data input.

Research instrument: The State Computer serves as a powerful instrument for scientific research, particularly in the study of the universe's behavior. It computationally reflects how the universe may be structured, including how it generates continuity, progression, and expansion.
Additionally, the system provides a framework for understanding how the universe permutes possibilities across space and time. It offers a new lens through which complex interactions and dynamic states can be analyzed with greater clarity.
For further theoretical context, refer to the paper on *Simultaneity and "Intervality" Theory*, also published on innova.ng

Data compression: In the domain of data compression, efficiency, and management, the State Computer introduces a novel and forward-looking approach based on identity-driven encoding and reconstruction.
Rather than relying solely on conventional redundancy-based compression techniques, the system operates by transforming data into structured identity representations derived from its internal classification and permutation framework.
The system generates an effectively infinite range of unique identities, which can be used to represent complex datasets in a significantly reduced and structured form. These identities do not function as simple hashes, but as interpretable identity states within the State Computer environment.
When instructed to perform compression, the system analyzes the input data, performs deep classification and contextual categorization, and determines the true structural identity of the dataset. This identity is then encoded into a compact representation, resulting in a substantial reduction in data size.
For example, large-scale datasets (e.g., on the order of hundreds of terabytes) would be significantly reduced in size potentially by a very large margin. Importantly, this process preserves the integrity of the data, as reconstruction is not dependent on the identity alone, but on the identity interpreted within the system.

This approach shifts the paradigm from traditional storage to identity-based data regeneration, where data is not merely stored but can be recreated from its underlying natural structural identity.

A Detailed Tutorial Manual on Matrix Construction, Radial Mapping, State Angle Derivation, Infinite Behavior, and the Natural Compression Effect.

Prepared from the development of the system itself, organized as a teachable manual for researchers, builders, and readers approaching the method from first principle to pre-coding aspect..

This document is intentionally instructional rather than brief. It explains the system from the seed-vector stage through matrix arrangement, radial placement, angle identity, the 720-based shortcut, the infinite attribute of the system, and the compression / slowdown behavior observed as states rise.

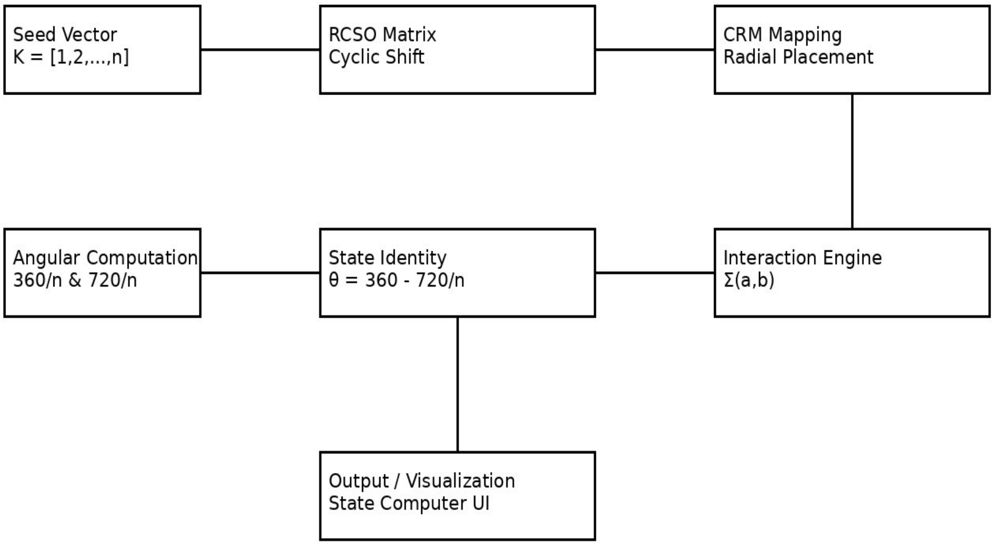
Working vocabulary used in this manual: seed vector, state, RCSO-style (radial cyclic shift operator) matrix, CRM-style (canonical radial mapping), state angle, gap, identity, infinity, compression, and natural deceleration.

1. Orientation to the System

The radial system is a method of converting a finite ordered seed into a structured matrix and then into circular geometry. Its importance is not merely that a pattern can be drawn, but that number order, cyclic movement, angle, and behavior can all be tied together in one framework. A state is the size of the seed and the size of the corresponding system. Thus a 7-state system starts from the seed vector 1, 2, 3, 4, 5, 6, 7 and grows into a 7 x 7 matrix and then into a radial field with 7 rings and 7 sectors. This is confirmed to be the first of its kind.

The system can be learned in three major passes. First, construct the seed and the cyclic matrix. Second, transfer that matrix into a radial map. Third, read angle behavior from the arrangement so that the state can eventually be identified without relying on the visual grid itself. This manual follows that progression in detail.

- Matrix stage: how the values are arranged and why the rows shift cyclically.
- Radial stage: how rows become rings, columns become sectors, and cells become placements.
- Identity stage: how angle, gap, and the 720-based relation allow larger states to be studied without always drawing them.
- always drawing them.



Illustrative flow diagram

2. Core Definitions

2.1 Seed vector

The seed vector is the ordered list that starts the state. For state n, the seed is the natural run from 1 to n. For example, state 7 begins with the seed vector 1, 2, 3, 4, 5, 6, 7. The seed tells us the size of the system and fixes the first row of the matrix.

Seed for state n: K = [1, 2, 3, ..., n]

2.2 State

A state is the cardinal size of the system. If the seed contains 7 values, then the system is state 7. The state simultaneously determines the number of rows, the number of columns, the number of rings, and the number of sectors. This one number therefore governs both the algebraic and the geometric form of the construction.

2.3 Matrix placement meaning

Once the seed is expanded into the matrix, each matrix cell has a row index r and a column index c. During radial conversion, the row index becomes ring position and the column index becomes sector position. Therefore, each cell M(r,c) is not only a numerical value; it is also an address in the radial map.

Row r -> Ring r   Column c -> Sector c

2.4 Angle language

Every state divides one circle into n equal angular parts. The ordinary sector angle of the state is therefore 360 degrees divided by n. This is the structural angle. Later in the manual we explain a second angle framework based on 720 divided by n, used as a behavioral shortcut for identity work.

Basic sector angle: 360 / n

3. Building the Matrix from the Seed

3.1 The simple hand method

For hand construction, the most direct method is not to compute every cell separately. Begin with the seed as the first row. Then, to form each next row, move the last entry of the previous row to the front. This is the fastest manual rule and it is especially useful when sketching by hand or teaching the system to another person.

- Write the seed as row 1.
- Take the last value of the row and move it to the front.
- Repeat until you have n rows.

3.2 State 7 worked directly from the seed

Using the rule above, state 7 expands as follows. Notice that the seed is preserved as the first row and every further row is a one-step cyclic right shift of the row before it.

r/c	A	B	C	D	E	F	G
A	1	2	3	4	5	6	7
B	7	1	2	3	4	5	6
C	6	7	1	2	3	4	5
D	5	6	7	1	2	3	4
E	4	5	6	7	1	2	3
F	3	4	5	6	7	1	2
G	2	3	4	5	6	7	1

3.3 The formal formula

The cyclic arrangement can also be written as a formula. This is useful for coding, proving, checking exact cells, and expressing the arrangement in a formal mathematical way. The formula below gives the value in row r and column c for a state n system.

$$M(r,c) = ((c - r) \bmod n) + 1$$

This formula means: take the column index, subtract the row index, reduce the result by modulo n so the value wraps around the state, and finally add 1 so that the matrix values lie in the set 1 to n rather than 0 to n - 1.

3.4 Why the formula reproduces the seed

When r = 1, the formula becomes M(1,c) = ((c - 1) mod n) + 1. As c runs from 1 to n, this gives 1, 2, 3, ..., n. In other words, the seed is not foreign to the formula. The seed is already built into the formula as the first row.

$$\text{For } r = 1: \quad M(1,c) = ((c - 1) \bmod n) + 1 = 1, 2, 3, \ldots, n$$

3.5 State 7 formula substitution

For state 7, the matrix formula becomes M(r,c) = ((c - r) mod 7) + 1. A few cells are enough to see its operation.

$$\text{State 7 formula:} \quad M(r,c) = ((c - r) \bmod 7) + 1$$

- M(1,1) = ((1 - 1) mod 7) + 1 = 1
- M(2,1) = ((1 - 2) mod 7) + 1 = (-1 mod 7) + 1 = 6 + 1 = 7
- M(4,2) = ((2 - 4) mod 7) + 1 = (-2 mod 7) + 1 = 5 + 1 = 6
- M(7,7) = ((7 - 7) mod 7) + 1 = 0 + 1 = 1

Interpretation. Working by hand , the row-shift method is usually easier. For formal work, coding, or checking exact placements, the matrix formula is the right tool. Both describe the same structure.

4. Converting Matrix into the Radial Map

4.1 The governing translation

The radial step of the method is a controlled translation from matrix coordinates into circular coordinates. The map is simple but powerful: rows become rings, columns become sectors, and each cell is placed at the intersection of its ring and sector. A value therefore preserves its matrix identity while gaining a geometric address.

Matrix -> Radial mapping: row r -> ring r, column c -> sector c

4.2 Radial subdivision

For a state n system, draw one master circle. Divide the full 360 degrees into n equal sectors. Then divide the radius into n equal concentric rings. The result is an n by n polar lattice. Because the same state number governs both directions, the radial grid and the matrix have the same total number of placement cells.

$$\text{Sector size} = 360 / n \qquad \text{Ring thickness} = R / n$$

4.3 State 7 radial quantities

If n = 7, the state is divided into 7 sectors and 7 rings. The sector angle is 360 / 7 = 51.428571 degrees. If a chosen working radius is R, then each ring thickness is R / 7. For example, if R = 70 mm, then each ring has thickness 10 mm.

For state 7: sector angle = 360 / 7 = 51.428571 degrees

If R = 70 mm, then ring thickness = 70 / 7 = 10 mm

4.4 Cell-center

When precision is required, one may describe each placement by a radial center and an angular center. The ring center of row r is at the midpoint of ring r. The angular center of column c is at the midpoint of sector c. This is not mandatory for all manual work, but it becomes useful in exact plotting and later software implementation.

$$r\_center = (r - 1/2) * (R / n)$$

$$theta\_center = theta0 + (c - 1/2) * (360 / n)$$

In the author's preferred convention, angle zero is taken from the left side of the circle and the sectors progress anticlockwise. Any other convention could be used if it is applied consistently, but one fixed reference is essential for cross-state comparison.

4.5 Manual transfer procedure

- Write or compute the matrix first.
- Draw a master circle and locate the center.
- Mark the n sector divisions around the circle.
- Draw n concentric rings from the center to the outer radius.
- Copy row 1 of the matrix into ring 1, one value per sector.
- Copy row 2 into ring 2, and continue outward until all rows are placed.

5. Tracing a Value and Harvesting State Pattern

5.1 A value appears once in each row

Because of the cyclic matrix structure, each number appears exactly once in every row and exactly once in every column. This means a chosen value can be followed as a path through all rings. In the radial map, that path becomes a geometric trajectory.

Position of value k in row r: c(r) = ((r + k - 2) mod n) + 1

5.2 State 7 example for the value 7

Substitute n = 7 and k = 7. Then the column position of the value 7 in each row is c(r) = ((r + 5) mod 7) + 1. Computing row by row gives the sequence of sectors in which the value 7 appears. (Ring 1 = closest to center of circle. Sector 1 = sector directly below diameter line)

Row r	Sector c(r)	Placement meaning
1	7	Ring 1, Sector 7
2	1	Ring 2, Sector 1
3	2	Ring 3, Sector 2
4	3	Ring 4, Sector 3
5	4	Ring 5, Sector 4
6	5	Ring 6, Sector 5
7	6	Ring 7, Sector 6

Once the points are marked in the radial grid, connect them in row order. The resulting line is the visible identity path of the chosen value within that state. Repeating the same procedure for other values reveals parallel identities and symmetries inside the same system.

6. From One State to State Angle

6.1 Structural angle for any state

At the first level, each state n has a direct structural angle: the circle is divided into n equal parts, so one sector occupies 360 / n degrees. This value explains the size of one sector and the basic angular step of the radial grid.

Structural angle of state n: A_structural = 360 / n

6.2 State 7 structural angle

A_structural(7) = 360 / 7 = 51.428571 degrees

This number is correct and useful example, but it does not by itself capture the wider identity logic that later emerged from comparing many states. The next step came from observing state behavior, gap behavior, wrap-around, and return relations across the system.

6.3 Why larger-state identity needed a shortcut

When only a single state is being drawn, the grid gives all visible information. But once many states are compared, and especially when larger states are involved, repeated visual construction becomes heavy and angular spacing becomes very small. A more direct route was needed: one that could read the state's identity angle without always relying on the radial grid.

The search for that shortcut is what led to the 720-based relation. In the language of the system, 360 described structural partition, but the doubled frame of 720 exposed a

State n	360 / n	720 / n gap	State angle 360 - 720 / n

stronger gap relation that was more useful for identity work.

7. The 720-Based Shortcut and the Easy Derivation of State Angle

7.1 The hidden relation

The 720 constant did not come from arbitrary choice. It emerged from repeated comparison of angular behavior across states. The key realization was that one full cycle of 360 degrees described only the immediate partition, while a doubled frame made the gap behavior clearer and easier to compare from state to state.

$$\textit{Gap quantity:}\quad G(n) = 720 / n$$

This quantity was treated as the gap arising from the doubled angular frame (individual state's angular Gap from 360). Once that gap was known, the state angle could be taken as the complement of that gap inside 360 degrees.

$$\textit{State angle:}\quad A\_state(n) = 360 - (720 / n)$$

7.2 Why the expression is written this way

The expression can be read in two pieces. First compute the doubled-frame gap 720 / n. Then subtract that gap from 360 to obtain the state angle. This makes the relation easy to execute by hand, easy to code, and easy to apply to large states without drawing the grid.

7.3 Angular identity of State 7 without using the grid

$$\textit{Gap for state 7:}\quad G(7) = 720 / 7 = 102.857143 \textit{ degrees}$$

$$\textit{State angle for state 7:}\quad A\_state(7) = 360 - 102.857143 = 257.142857 \textit{ degrees}$$

7.4 General procedure for any state

- Take the state number n.
- Compute the gap G(n) = 720 / n.
- Subtract that result from 360.
- The answer is the state angle A_state(n) = 360 - (720 / n).

7.5 Example table across several states

4	90	180	180
5	72	144	216
6	60	120	240
7	51.428571	102.857143	257.142857
8	45	90	270

Interpretive caution. Within this manual, the 720 relation is presented exactly as it was developed and used in the system discussion. It is treated as the shortcut identity rule that helped move from grid-based discovery to direct state-angle computation.

8. How Unique State Identity Was Recognized Across Many States

8.1 From one state to many states

After learning how to plot one state, the next step was to compare state after state while holding the geometric frame consistent. The same master circle idea, the same center, the same angular reference, and the same general method of placement were used across different states. This consistency is one of the strongest reasons identity became visible.

8.2 Why fixed radius mattered

Using the same circular radius across different states meant the boundary of the system did not keep changing. If radius had changed from state to state, the eye would have mixed two different effects: scale change and structural change. By fixing the radius, the comparison was purified. What changed from state to state was not the outer size, but the internal angular subdivision and the behavior of the placements.

With fixed radius R, ring thickness changes as R / n, but the total circle stays constant.

8.3 Other reasons identity became possible

- The same 360-degree boundary was used for all states, so each state lived in the same total angular space.
- The same origin and direction convention were maintained, so cross-state angles were comparable.
- The matrix rule remained cyclic, so movement was not random but structured.
- Rows became rings consistently, which created a progression axis across states.

- A chosen value could be traced from ring to ring, making behavior visible rather than merely static.

Together these choices created a normalized measurement framework. That normalized framework is what allowed state identity to emerge as an intrinsic property rather than an artifact of drawing scale or inconsistent plotting.

8.4 The role of angle sequences and gap behavior

Once multiple states had been built, comparison shifted from simple shape viewing to behavioral reading. The questions became: What is the angular gap of this state. How does it compare with that of another state. Does the pattern close quickly or stretch into finer motion. Does the state behave in a coarse, symmetric way or a finer, more compressed way. This is the stage at which identity ceased to mean merely a picture and began to mean a rule of behavior.

9. The Infinite Attribute of the System

9.1 Infinite states in a fixed framework

The radial system has an infinite attribute because the state number n has no final upper bound. For every positive integer, another state can be created. Each new state produces a new matrix, a new radial partition, a new set of trajectories, and a new identity relation. This makes the collection of all states countably infinite.

n = 1, 2, 3, 4, ...   with no final state

9.2 Infinity inside finite space

A particularly strong insight of the system is that this infinite progression unfolds inside a finite spatial frame. The circle remains bounded, yet the state count can rise without bound. As n increases, the sector size decreases and the ring thickness decreases, allowing more and more structure to be compressed into the same circular area.

As n increases: 360 / n -> 0

9.3 Infinite resolution and asymptotic behavior

Because 360 / n becomes smaller as n grows, the system tends toward ever finer angular resolution. High states therefore approximate a continuous field more closely than low states. In the shortcut identity expression, A_state(n) = 360 - (720 / n), the gap term 720 / n tends toward zero as n rises, which means the state angle tends toward 360 degrees without any finite state ever actually becoming the infinite limit itself.

As n -> infinity: 720 / n -> 0,   so A_state(n) -> 360

9.4 Why this matters conceptually

The infinite attribute is not a mere statement that counting never stops. It means the method can continue generating new structure, new angle behavior, and new pattern classes indefinitely. Each state remains finite in itself, but the family of states is unbounded, and its geometric behavior becomes progressively finer.

10. Compression and the Natural Slowdown Effect

10.1 The observed phenomenon

A strong practical observation from the system was that as the shapes rotate through successively higher states, the movement appears to slow down. Early states show wide angle gaps and broad jumps. Higher states show thinner angle gaps, tighter motion, and a gradual compression effect. Importantly, this was recognized as not merely a processor limitation. The system itself naturally produces this visual deceleration.

10.2 The mathematical cause

The root cause is simple: the angular displacement per state decreases inversely with the state number. Whether one uses the structural division 360 / n or the doubled-frame gap 720 / n, the same tendency appears. As n rises, the amount of angle covered in one step becomes smaller.

Angular step decreases with state: step is proportional to 1 / n

As n increases: 360 / n and 720 / n both become smaller

10.3 Why the motion looks slower

Visual rotational pace depends on how much angular distance is covered from one state to the next or from one layer to the next. If the angle difference is large, the pattern seems to swing broadly and quickly. If the angle difference is small, the motion seems tight and restrained. Thus the apparent slowdown is an intrinsic geometric result of the shrinking angular gaps.

10.4 Compression in the same circle

Because the circle radius remains fixed while the state number rises, more subdivisions are packed into the same area. This is a compression effect in both space and motion. Spatially, the sectors and rings become finer. Dynamically, the angular displacement narrows. The visual result is that the rotating conical-hand style motion does not simply continue at the same expressive pace; it compresses and decelerates.

10.5 Research statement of the effect

A suitable formal statement is the following: as the state number increases, the angular gap associated with the state decreases inversely with n. This compresses the rotational displacement and causes the apparent motion of the state trajectory to slow progressively. The slowdown therefore has a computational side in actual software, but it also has a deeper geometric origin in the natural system behavior itself.

11. Closing Perspective

The radial system begins with a simple seed and grows into a matrix, a radial field, and an angle-based identity method. Its power lies in the fact that one ordered finite list can be transformed into geometric behavior, and that the behavior remains comparable across states because the construction is disciplined: the same circle, the same center, the same directional convention, and the same cyclic logic are preserved.

From this base, several key features emerge. First, the matrix-to-radial translation makes behavior visible. Second, the 720-based relation gives a non-visual route to the state angle. Third, the family of states extends indefinitely, giving the system an infinite attribute. Fourth, increasing state number naturally compresses angular gaps and produces a real slowdown effect that belongs to the geometry itself, not only to computation.

For teaching, manual reproduction, and research development, the method can therefore be approached in two complementary ways. One may build from the grid upward and let the geometry reveal its behavior. Or, once the system has been understood, one may use the angle shortcut to move directly from state number to state angle. Both routes belong to the same structure.

CONCLUSION

The State Computer provides a novel method of encoding identity and interaction using angular computation, enabling deterministic and scalable identity systems. A new paradigm where numbers are not merely values but dynamic identities. The system provides a robust framework for comparing, understanding and simulating behavior.