The WPE-TME Semantic Calculus: A Framework for Structural and Temporal Reasoning in AI Systems

Author: Chris Young
theheimdallorganization@gmail.com

November 30, 2025

Abstract

This paper presents the Wave Pattern Encoding (WPE) and Temporal Modulation Encoding (TME) semantic calculus, a novel geometric framework designed to provide explicit structural and temporal representations for artificial intelligence reasoning systems. Current large language models demonstrate remarkable pattern recognition capabilities but lack explicit mechanisms for maintaining consistent internal structure across extended reasoning chains and representing temporal relationships. The framework addresses this limitation through a text-based geometric calculus where components occupy positions in phase space, coupling strengths emerge from angular separation via cosine relationships, and temporal ordering becomes syntactically explicit. This paper provides rigorous mathematical foundations, derives the coupling and hierarchical influence mechanisms, demonstrates validation criteria, and presents applications across multiple domains. The framework requires no model fine-tuning and operates through prompting alone, making it immediately applicable to existing AI systems.

1 Introduction

Large language models have demonstrated substantial capabilities in pattern recognition, text generation, and knowledge retrieval [1]. These systems excel at tasks involving statistical pattern matching learned from extensive training corpora. However, systematic limitations emerge when models must maintain consistent internal structure across extended reasoning sequences or explicitly represent temporal relationships between events and processes.

The fundamental issue stems from the representational architecture of these systems. Models generate tokens through statistical predictions based on learned patterns distributed across network parameters. This architecture provides no stable internal scaffold that persists across conversational context. No explicit encoding exists for temporal ordering, structural dependencies, or coupling strengths between system components. While such relationships may exist implicitly within the learned parameter space, they remain inaccessible for direct manipulation or inspection.

This representational gap creates predictable failure modes. Reasoning chains exhibit drift as context windows approach capacity limits. Temporal relationships become ambiguous when multiple event sequences interleave. Multi-step problems requiring precise structural relationship maintenance produce inconsistent outputs. When tasked with tracking multiple interconnected variables through extended reasoning sequences, models frequently lose track of influence relationships and dependency structures.

The debugging challenge illustrates the problem clearly. When reasoning failures occur, no explicit structure exists for inspection. The computation occurs within an opaque parameter space, preventing direct examination of how the model represents relationships such as coupling strengths or temporal orderings.

This paper introduces a complementary representation system designed to scaffold model reasoning through explicit geometric structure. The Wave Pattern Encoding (WPE) provides spatial representation where components occupy positions in phase space, with interaction strengths determined by geometric relationships. The Temporal Modulation Encoding (TME) extends this framework to make temporal relationships syntactically explicit, where left-to-right reading corresponds to forward temporal progression.

1.1 Motivation and Design Philosophy

The framework adheres to a core principle: *encoding equals computation.* The syntax rules themselves define system behavior without requiring separate formula evaluation. When a WPE expression is constructed, the geometric relationships encoded in the notation directly specify component interactions.

This principle generates several fundamental design requirements:

Text-based representation: Language models operate on tokenized text. Any framework for AI reasoning must function within this constraint. The system employs no external data structures or hidden vector representations—only text that models can read and generate. This constraint appears restrictive but offers significant advantages. Models are natively trained on text manipulation. They generate it fluently. The framework operates in their native medium.

Model-agnostic implementation: The representation functions through prompting alone, requiring no fine-tuning or weight modification. Models that comprehend the syntax rules can immediately employ the framework. This design choice ensures portability across different model architectures and enables rapid iteration without training cycles.

Compositional structure: Complex systems compose from simple primitives. The notation supports hierarchical composition where subsystems nest within systems, patterns recur across scales, and operators combine structures predictably. This enables both scaling and reuse.

Explicit temporal relationships: Temporal ordering must be visible in the notation itself. When one component precedes another, this relationship appears structurally. When processes execute in parallel, this appears through distinct representational layers.

Visible computation: All information processing occurs at the representation level. No hidden state exists. If a model can read the encoding, it can trace information flow through the system.

These constraints emerge from the practical reality of working with contemporary language models. The framework requires representations that models manipulate as text, that function without architectural modifications, and that make all reasoning steps explicit.

1.2 Contributions

This paper makes the following contributions:

1. Formal definition of the WPE geometric calculus with rigorous derivation of coupling mechanisms
2. Mathematical foundations for hierarchical shell influence relationships
3. Extension to temporal representation through TME with explicit ordering constraints
4. Validation framework ensuring structural consistency
5. Cross-domain applications demonstrating framework generality
6. Computational complexity analysis and implementation considerations

The remainder of this paper proceeds as follows. Section 2 establishes mathematical foundations and derives core relationships. Section 3 presents the WPE structural framework. Section 4 extends representation to temporal dimensions through TME. Section 5 demonstrates applications across multiple domains. Section 6 discusses implementation considerations and computational properties. Section 7 concludes.

2 Mathematical Foundations

This section establishes the mathematical framework underlying WPE-TME representations. The framework builds on geometric relationships in phase space, hierarchical influence mechanisms, and curvature-based stability characterization.

2.1 Phase Space Geometry

Components in the WPE framework occupy positions in a geometric phase space. This space is characterized by angular position $\theta \in [0, 360)$ degrees, representing location on a unit circle.

Definition 2.1 (Phase Position)**.** A component's phase position θ represents its angular location in phase space, measured in degrees from a reference direction. Phase values satisfy $0 \leq \theta < 360$.

The fundamental interaction mechanism derives from the geometric relationship between components. Given two components i and j with phases θ_i and θ_j, their coupling strength emerges from their angular separation.

Definition 2.2 (Phase Coupling)**.** The coupling strength C_{ij} between components i and j is defined as:

$$C_{ij} = \cos(\theta_i - \theta_j) \tag{1}$$

where $\theta_i - \theta_j$ represents the phase difference in degrees.

This cosine relationship produces several important coupling regimes:

Theorem 2.3 (Coupling Properties)**.** *The coupling function $C(\Delta\theta) = \cos(\Delta\theta)$ satisfies:*

(i) $C(0) = 1$ *(maximal positive coupling)*

(ii) $C(90) = 0$ *(orthogonal independence)*

(iii) $C(180) = -1$ *(maximal negative coupling)*

(iv) $C(\Delta\theta)$ *is symmetric:* $C(\Delta\theta) = C(-\Delta\theta)$

(v) $C(\Delta\theta)$ *is periodic with period* 360

Proof. Properties (i)-(iii) follow directly from standard trigonometric identities. Symmetry (iv) follows from $\cos(-x) = \cos(x)$. Periodicity (v) follows from $\cos(x + 360) = \cos(x)$. □

The coupling strength determines how components influence each other. Components with $\Delta\theta = 0$ maximally reinforce. Components with $\Delta\theta = 180$ maximally oppose. Components with $\Delta\theta = 90$ operate independently.

Example 2.4 (Triangular Configuration)**.** Consider three components positioned at $\theta_1 = 0$, $\theta_2 = 120$, $\theta_3 = 240$. The pairwise couplings are:

$$C_{12} = \cos(120) = -0.5 \tag{2}$$

$$C_{13} = \cos(240) = -0.5 \tag{3}$$

$$C_{23} = \cos(120) = -0.5 \tag{4}$$

This symmetric configuration with uniform negative coupling appears in three-phase systems, predator-prey-resource ecosystems, and attention-memory-processing trade-offs.

2.2 Hierarchical Shell Structure

Components exist at different hierarchical levels, represented by shell values $\lambda \in \{1, 2, \ldots, 9\}$. Higher shells influence lower shells through a mathematically defined mechanism.

Definition 2.5 (Shell Influence)**.** The influence strength $I_{high \to low}$ from a component at shell λ_{high} to a component at shell λ_{low} is:

$$I_{high \to low} = \frac{1}{\lambda_{low}} - \frac{1}{\lambda_{high}} \tag{5}$$

provided $\lambda_{high} > \lambda_{low}$. Otherwise $I = 0$.

This formulation ensures several desirable properties:

Theorem 2.6 (Shell Influence Properties)**.** *The influence function $I(\lambda_{low}, \lambda_{high})$ satisfies:*

(i) $I > 0$ if and only if $\lambda_{high} > \lambda_{low}$

(ii) I increases with the shell gap $\lambda_{high} - \lambda_{low}$

(iii) I decreases as λ_{low} increases (diminishing influence at higher levels)

(iv) Maximum influence occurs at $\lambda_{low} = 1$, $\lambda_{high} = 9$: $I_{max} = 1 - 1/9 \approx 0.889$

Proof. Property (i) follows directly from the fact that $1/\lambda_{low} > 1/\lambda_{high}$ if and only if $\lambda_{low} < \lambda_{high}$.

For property (ii), consider the partial derivative with respect to λ_{high}:

$$\frac{\partial I}{\partial \lambda_{high}} = \frac{1}{\lambda_{high}^2} > 0 \tag{6}$$

Thus influence increases with λ_{high}.

For property (iii), the partial derivative with respect to λ_{low} is:

$$\frac{\partial I}{\partial \lambda_{low}} = -\frac{1}{\lambda_{low}^2} < 0 \tag{7}$$

Thus influence decreases with λ_{low}.

Property (iv) follows by direct calculation at the extreme values. □

Example 2.7 (Shell Influence Calculation)**.** Consider shell 7 influencing shell 1:

$$I_{7 \to 1} = \frac{1}{1} - \frac{1}{7} = 1.0 - 0.143 = 0.857 \tag{8}$$

This represents strong influence—high-level abstract constraints heavily affect low-level concrete components.

For adjacent shells, consider shell 3 influencing shell 2:

$$I_{3 \to 2} = \frac{1}{2} - \frac{1}{3} = 0.500 - 0.333 = 0.167 \tag{9}$$

Adjacent levels exhibit moderate coupling, allowing autonomy while maintaining coordination.

2.3 Curvature and Stability

Each component possesses a curvature parameter $\kappa \in [-10, 10]$ that characterizes its stability properties and, in temporal contexts, its duration characteristics.

Definition 2.8 (Curvature)**.** The curvature κ of a component characterizes its potential well structure:

- $\kappa < 0$: stable well (restoring force toward equilibrium)
- $\kappa = 0$: neutral stability (no restoring force)
- $\kappa > 0$: unstable peak (repelling force from equilibrium)

In spatial contexts, curvature represents the second derivative of potential energy with respect to position. Negative curvature creates stable equilibria where systems naturally settle. Positive curvature creates unstable equilibria where systems diverge under perturbation.

In temporal contexts, curvature relates to duration:

Definition 2.9 (Time Constant)**.** The time constant τ of a component is:

$$\tau = \frac{1}{|\kappa|} \tag{10}$$

where τ represents the characteristic duration of component activity.

Theorem 2.10 (Curvature-Duration Relationship)**.** *Components with larger $|\kappa|$ have shorter durations. Specifically:*

(i) High $|\kappa|$ (e.g., $|\kappa| = 8$) implies $\tau = 0.125$ (fast transient)

(ii) Low $|\kappa|$ (e.g., $|\kappa| = 1$) implies $\tau = 1$ (slow persistent)

(iii) $\kappa \to 0$ implies $\tau \to \infty$ (permanent state)

This relationship connects spatial stability to temporal duration, providing a unified parameter for both structural and temporal properties.

2.4 Combined Interaction Model

The complete interaction between components combines phase coupling, shell influence, and curvature effects.

Definition 2.11 (Effective Coupling)**.** The effective coupling E_{ij} between component i at shell λ_i with phase θ_i and curvature κ_i, and component j at shell λ_j with phase θ_j and curvature κ_j is:

$$E_{ij} = C_{ij} \cdot I_{ij} \cdot \sqrt{|\kappa_i \kappa_j|} \tag{11}$$

where $C_{ij} = \cos(\theta_i - \theta_j)$ and $I_{ij} = \max(0, 1/\min(\lambda_i, \lambda_j) - 1/\max(\lambda_i, \lambda_j))$.

The coupling combines three independent mechanisms:

1. Phase coupling from geometric separation
2. Hierarchical influence from shell structure
3. Stability modulation from curvature magnitude

Each mechanism operates independently but combines multiplicatively to determine overall interaction strength.

3 Wave Pattern Encoding (WPE)

This section presents the Wave Pattern Encoding framework for structural representation. WPE provides explicit notation for component properties and their geometric relationships.

3.1 Component Syntax

Each component in WPE is specified by exactly four parameters arranged in a standardized syntax.

Definition 3.1 (WPE Component Syntax)**.** A WPE component is written as:

$$\Phi : \lambda @ \theta | \kappa \tag{12}$$

where:

- Φ is the domain identifier (field type)
- $\lambda \in \{1, 2, \ldots, 9\}$ is the shell level
- $\theta \in [0, 360)$ is the phase position in degrees
- $\kappa \in [-10, 10]$ is the curvature

The domain identifier Φ specifies the physical substrate:

Table 1: Standard Domain Identifiers

Domain	**Interpretation**
P	Physics (electromagnetic, forces, energy)
C	Cognition (neural activity, information processing)
B	Biology (populations, chemical concentrations)
S	Social (influence, relationships, economics)
Ph	Philosophy (abstract concepts, logic)
M	Memory (stored information, persistent state)
O	Output (final results, system outputs)
NX	Unknown (unspecified, exploratory)
MT	Meta (reasoning about system itself)

Example 3.2 (Component Specification)**.** The expression `P:1@0|-3` represents:

- Domain: Physics (P)
- Shell: Level 1 (foundation)
- Phase: 0° (reference position)
- Curvature: -3 (moderately stable well)

3.2 Compositional Operators

Components combine through standardized operators that define structural relationships.

Definition 3.3 (Sequential Coupling Operator)**.** The operator $*$ denotes sequential coupling. In $A * B$, component A influences component B with strength $C_{AB} = \cos(\theta_A - \theta_B)$.

Definition 3.4 (Parallel Independence Operator)**.** The operator + denotes parallel independence. In $A + B$, components operate independently with no direct coupling.

Definition 3.5 (Bracket Grouping)**.** Square brackets [...] group components into subsystems that can be manipulated as single units.

Definition 3.6 (Output Specification)**.** The operator => specifies system output. Every valid WPE expression must contain exactly one output specification.

Definition 3.7 (Feedback Coupling)**.** The operator $< - >$ denotes bidirectional feedback. In $A < - > B$, components mutually influence each other.

Example 3.8 (Sequential Processing Chain)**.** The expression:

```
P:1@0|-3 * C:2@30|-2.5 * B:3@60|-2 => O:2@120|-2
```

represents a sequential processing chain where:

- Physics component at shell 1, phase 0°
- Couples to Cognition component at shell 2, phase 30°
- Which couples to Biology component at shell 3, phase 60°
- Producing Output at shell 2, phase 120°

Coupling strengths:

$$\begin{aligned}
C_{P\to C} &= \cos(30) = 0.866 \text{ (strong positive)}\\
C_{C\to B} &= \cos(30) = 0.866 \text{ (strong positive)}\\
C_{B\to O} &= \cos(60) = 0.500 \text{ (moderate positive)}
\end{aligned}$$

3.3 Hierarchical Nesting

WPE supports nested hierarchical structures through brace notation.

Definition 3.9 (Hierarchical Nesting)**.** Components can nest using braces: $A : \{B : \{C\}\}$ indicates C nested in B nested in A, with shell influence flowing from outer to inner.

Example 3.10 (Hierarchical Control)**.** The expression:

```
C:7@0|-1.5:{C:5@30|-1.8:{C:3@60|-2:{C:1@90|-2.5}}} => O:2@120|-2
```

represents four nested cognition levels with influence strengths:

$$\begin{aligned}
I_{7\to 5} &= \frac{1}{5} - \frac{1}{7} = 0.200 - 0.143 = 0.057\\
I_{5\to 3} &= \frac{1}{3} - \frac{1}{5} = 0.333 - 0.200 = 0.133\\
I_{3\to 1} &= \frac{1}{1} - \frac{1}{3} = 1.000 - 0.333 = 0.667
\end{aligned}$$

3.4 Extended Syntax Features

The framework supports several extended syntax features for specialized representations.

3.4.1 Multi-Phase Specification

Components can occupy multiple phase positions simultaneously with weighted contributions.

Definition 3.11 (Multi-Phase Syntax)**.** A component at multiple phases is written:

$$\Phi : \lambda@[\theta_1 : w_1 + \theta_2 : w_2 + \ldots + \theta_n : w_n]|\kappa \tag{13}$$

where w_i are weights satisfying $\sum w_i = 1$.

Example 3.12 (Multi-Phase Component)**.** `P:1@[0:0.7+90:0.3]|-2` represents a physics component with 70% weight at 0° and 30% weight at 90°, creating a composite influence pattern.

3.4.2 Temporal Phase Functions

Phase can be specified as a function of time for dynamic systems.

Definition 3.13 (Temporal Phase)**.** A time-varying phase is written $@\Theta(t)$ where $\Theta(t)$ is a function mapping time to phase.

Common temporal functions include:

- Linear: $\Theta(t) = \theta_0 + \omega t$
- Sinusoidal: $\Theta(t) = \theta_0 + A \sin(\omega t)$
- Exponential: $\Theta(t) = \theta_0(1 - e^{-t/\tau})$

3.4.3 Phase Ranges

Components can span phase ranges rather than discrete positions.

Definition 3.14 (Phase Range Syntax)**.** A component spanning a phase range is written:

$$\Phi : \lambda@[\theta_1 : \theta_2 : \theta_3]|\kappa \tag{14}$$

indicating the component occupies positions from θ_1 to θ_3 with peak at θ_2.

3.5 Validation Framework

Valid WPE expressions must satisfy structural constraints.

Definition 3.15 (WPE Validation Rules)**.** A WPE expression is valid if and only if:

V1: All domain identifiers Φ are recognized

V2: All shell values $\lambda \in \{1, 2, \ldots, 9\}$

V3: All phase values $\theta \in [0, 360)$

V4: All curvature values $\kappa \in [-10, 10]$

V5: Exactly one output component exists

V6: No circular dependencies exist (acyclic structure)

V7: All operators are correctly paired

Theorem 3.16 (Validation Decidability)**.** *WPE expression validity is decidable in linear time with respect to expression length.*

Proof. Each validation rule can be checked in a single pass through the expression:

- V1-V4: Check each component's parameters against ranges
- V5: Count output operators (must equal 1)
- V6: Perform depth-first search for cycles (linear in expression size)
- V7: Verify operator syntax during parsing (linear scan)

All checks complete in $O(n)$ time where n is expression length. □

4 Temporal Modulation Encoding (TME)

The Temporal Modulation Encoding extends WPE to represent explicit temporal structure. TME makes time syntactically visible through ordered phase progression and layered parallel processes.

4.1 Temporal Semantics

TME reinterprets the phase parameter θ as temporal position rather than spatial position.

Definition 4.1 (Phase as Time)**.** In TME, the phase value θ represents temporal position where:

- Lower phase values occur earlier
- Higher phase values occur later
- Left-to-right reading corresponds to forward time

This reinterpretation transforms geometric coupling into temporal precedence. Components with smaller phase values temporally precede components with larger phase values.

4.2 Layer Structure

TME organizes components into independent temporal layers.

Definition 4.2 (TME Layer)**.** A TME layer represents an independent timeline. Layers are denoted by prefixes L1, L2, etc. Components within a layer must maintain temporal ordering. Components in different layers can have arbitrary temporal relationships.

Definition 4.3 (TME System Syntax)**.** A complete TME system is written:

$$
\begin{aligned}
&\texttt{@temporal\_scale}\ \alpha = \text{value}\\
&\texttt{System =}\\
&\quad \texttt{L1:}\quad \Phi : \lambda @ \theta | \kappa * \ldots\\
&\quad \texttt{L2:}\quad \Phi : \lambda @ \theta | \kappa * \ldots\\
&\quad \vdots\\
&\quad \texttt{=> Output}
\end{aligned}
$$

where α is the temporal scale parameter and each layer contains a temporally ordered sequence.

4.3 Temporal Ordering Constraints

TME imposes strict ordering requirements within layers.

Definition 4.4 (Monotonicity Constraint)**.** Within each layer, phase values must be monotonically non-decreasing:

$$\theta_1 \leq \theta_2 \leq \theta_3 \leq \ldots \leq \theta_n \tag{15}$$

Theorem 4.5 (TME Ordering Rules)**.** *Valid TME expressions satisfy:*

R1: Phase monotonicity within each layer

R2: Independence across layers (no ordering constraint)

R3: Output component at maximum phase in its layer

R4: No phase wraparound within layers ($359 \not\to 0$*)*

Violations of these rules produce temporally inconsistent expressions.

Example 4.6 (Invalid TME Expression)**.** The expression:

```
L1:  P:1@30|-2 * C:1@10|-2 => O:1@40|-2
```

is invalid because phases decrease from 30° to 10°, violating monotonicity (R1).

4.4 Duration and Timing

TME derives concrete timing from curvature and phase progression.

Definition 4.7 (Component Duration)**.** The duration τ_i of component i is:

$$\tau_i = \frac{1}{|\kappa_i|} \tag{16}$$

measured in relative time units.

Definition 4.8 (Phase Transition Time)**.** The time to transition from phase θ_i to θ_{i+1} is:

$$\Delta t_i = \frac{\theta_{i+1} - \theta_i}{360} \cdot \tau_i \tag{17}$$

representing the fraction of component duration consumed by the phase change.

Definition 4.9 (Temporal Scale Parameter)**.** The temporal scale α converts relative timing to absolute units:

$$t_{absolute} = \alpha \cdot t_{relative} \tag{18}$$

Standard temporal scales:

- $\alpha = 0.001$: millisecond timescale
- $\alpha = 1$: second timescale
- $\alpha = 60$: minute timescale
- $\alpha = 3600$: hour timescale
- $\alpha = 86400$: day timescale
- $\alpha = 31557600$: year timescale

4.5 Total System Timing

The total execution time of a TME system can be calculated from component durations and phase progressions.

Theorem 4.10 (Total Timing Calculation)**.** *For a TME layer with n components at phases $\theta_1, \theta_2, \ldots, \theta_n$ with curvatures $\kappa_1, \kappa_2, \ldots, \kappa_{n-1}$, the total time is:*

$$T_{total} = \alpha \sum_{i=1}^{n-1} \frac{\theta_{i+1} - \theta_i}{360} \cdot \frac{1}{|\kappa_i|} \tag{19}$$

Proof. Each transition from component i to component $i+1$ consumes time:

$$\Delta t_i = \frac{\theta_{i+1} - \theta_i}{360} \cdot \frac{1}{|\kappa_i|} \cdot \alpha \tag{20}$$

The total time sums all transitions:

$$T_{total} = \sum_{i=1}^{n-1} \Delta t_i = \alpha \sum_{i=1}^{n-1} \frac{\theta_{i+1} - \theta_i}{360} \cdot \frac{1}{|\kappa_i|} \tag{21}$$

□

Example 4.11 (Timing Calculation)**.** Consider the system:

```
@temporal_scale α = 1.0
L1:  P:1@0|-8 * C:1@90|-4 * B:1@180|-2 => O:1@270|-2
```

Calculate total time:

$$\begin{aligned} T_{total} &= 1.0 \cdot \left[\frac{90}{360} \cdot \frac{1}{8} + \frac{90}{360} \cdot \frac{1}{4} + \frac{90}{360} \cdot \frac{1}{2}\right] \\ &= \frac{1}{4}\left[\frac{1}{8} + \frac{1}{4} + \frac{1}{2}\right] \\ &= \frac{1}{4}\left[\frac{1+2+4}{8}\right] = \frac{7}{32} \approx 0.219 \text{ seconds} \end{aligned}$$

This fast cascade has rapid early processing (high $|\kappa|$) and slower later integration (low $|\kappa|$).

4.6 Parallel Temporal Processes

Multiple layers enable parallel temporal processes with independent timelines.

Definition 4.12 (Layer Independence)**.** Layers L1, L2, . . . , Ln execute independently. Timing in one layer does not constrain timing in another layer unless explicit cross-layer coupling exists.

Example 4.13 (Parallel Processing)**.** The system:

```
@temporal_scale α = 1.0
System =
  L1:  P:1@0|-3 * P:1@90|-3 * P:1@180|-3 => O:1@270|-2
  L2:  C:1@0|-2 * C:1@120|-2 => O:1@240|-2
  L3:  B:1@0|-1 * B:1@180|-1 => O:1@270|-1.5
```

represents three parallel processes:

- L1: Three-step physics process
- L2: Two-step cognition process
- L3: Two-step biology process

Each proceeds independently with its own timing. The final output integrates results from all three layers.

5 Applications Across Domains

The WPE-TME framework applies across diverse domains through domain-agnostic geometric relationships. This section demonstrates applications in physics, biology, cognition, and social systems.

5.1 Physics: Three-Phase Electrical Power

Three-phase electrical power systems exhibit balanced 120° spacing, zero net voltage, and symmetric load distribution.

Example 5.1 (Three-Phase System)**.** WPE encoding:

```
P:1@0|-2 * P:1@120|-2 * P:1@240|-2 => O:2@180|-1.5
```

Phase relationships:

$$\begin{aligned} &V_1 \text{ at } 0 \\ &V_2 \text{ at } 120 \text{ (lagging by 120)} \\ &V_3 \text{ at } 240 \text{ (lagging by 240)} \end{aligned}$$

Pairwise couplings:

$$C_{12} = C_{23} = C_{31} = \cos(120) = -0.5$$

This uniform negative coupling creates balance. The sum $V_1 + V_2 + V_3 = 0$ at all times, a fundamental property of three-phase systems encoded directly in the geometric structure.

5.2 Biology: Predator-Prey Dynamics

Predator-prey systems exhibit oscillatory dynamics with temporal phase relationships.

Example 5.2 (Predator-Prey Oscillation)**.** TME encoding:

@temporal_scale $\alpha = 86400$ (days)

```
System =
  L1:  B:1@0|-1.5 * B:1@90|-1.8 => O:1@180|-1.2
```

Interpretation:

- Component at 0°: Prey population at peak
- Component at 90°: Predator population increasing (quarter cycle later)
- Output at 180°: System state after half cycle

The 90° phase lag captures the characteristic predator-prey delay: predators respond to prey abundance with temporal lag determined by reproduction timescales.

Duration calculation with $\kappa = -1.5$ gives $\tau = 1/1.5 \approx 0.67$ units. At daily scale:

$$T_{cycle} = 86400 \cdot \frac{180}{360} \cdot 0.67 \approx 29000 \text{ seconds} \approx 8 \text{ hours} \tag{22}$$

5.3 Cognition: Attention-Memory-Processing

Cognitive systems balance competing demands for attention, memory, and processing resources.

Example 5.3 (Cognitive Resource Allocation)**.** WPE encoding:

```
C:1@0|-2.5 * C:1@120|-2.5 * C:1@240|-2.5 => O:3@180|-1.5
```

This triangular structure at 120° spacing creates balanced trade-offs:

- Attention at 0°: Focus on current input
- Memory at 120°: Access to stored information
- Processing at 240°: Computational transformation

The -0.5 coupling between each pair means increasing one resource moderately decreases the others. The system cannot maximize all three simultaneously—a fundamental constraint in cognitive architecture.

The output at shell 3 integrates these competing demands into unified cognitive performance.

5.4 Social Systems: Market Dynamics

Economic markets balance supply, demand, and price through dynamic adjustment.

Example 5.4 (Supply-Demand Market)**.** TME encoding:

@temporal_scale $\alpha = 3600$ (hours)

```
System =
  L1:  S:1@0|-2 * S:1@180|-2 => O:2@90|-1.5
```

Components:

- Supply at 0°: Producer quantity offered
- Demand at 180°: Consumer quantity desired (perfect opposition)
- Output at 90°: Market clearing price (orthogonal to both)

The 180° opposition between supply and demand creates dynamic tension. Price adjustment emerges from their interaction, positioned orthogonally at 90° to represent its independence from either individual force.

Market clearing occurs when the output stabilizes, with timing determined by the curvature values (market adjustment speed).

5.5 Cross-Domain Pattern Recognition

The framework reveals deep structural similarities across domains.

Theorem 5.5 (Geometric Invariance)**.** *Systems with identical WPE geometric structure exhibit mathematically equivalent coupling relationships regardless of physical substrate.*

This table demonstrates how the same geometric configurations recur across vastly different physical substrates. The geometric relationships remain invariant while the physical interpretation changes.

Table 2: Cross-Domain Geometric Patterns

Geometry	**Physics**	**Biology**	**Cognition**	**Social**
Triangle (120°)	3-phase power	Predator-prey-resource	Attention-memory-process	Leader-follower-mediator
Opposition (180°)	Matter-antimatter	Predator-prey	Excite-inhibit	Supply-demand
Orthogonal (90°)	E and B fields	Genotype-environment	Input-memory	Quantity-price
Alignment (0°)	Coherent waves	Mutualism	Reinforcement	Alliance

6 Implementation and Computational Properties

This section analyzes computational properties, implementation considerations, and practical usage guidelines for the WPE-TME framework.

6.1 Computational Complexity

Theorem 6.1 (Coupling Calculation Complexity)**.** *For a system with n components, calculating all pairwise couplings requires $O(n^2)$ operations.*

Proof. Each component pair (i, j) requires one cosine evaluation for $\cos(\theta_i - \theta_j)$. With n components, there are $\binom{n}{2} = \frac{n(n-1)}{2} \in O(n^2)$ pairs. □

Theorem 6.2 (Hierarchical Influence Complexity)**.** *Computing all shell influences requires $O(n^2)$ operations in the worst case where all components are at different shells.*

Proof. Similar to coupling calculation, each component pair requires one influence calculation $I_{ij} = 1/\lambda_i - 1/\lambda_j$, giving $O(n^2)$ total operations. □

Theorem 6.3 (TME Timing Complexity)**.** *Computing total timing for a TME system with n components distributed across k layers requires $O(n)$ operations.*

Proof. Each layer is processed sequentially. Within a layer with m components, computing timing requires $m - 1$ phase difference calculations and $m - 1$ duration calculations, totaling $O(m)$ operations per layer. Summing over all k layers gives $O(n)$ where $n = \sum_{i=1}^{k} m_i$. □

6.2 Memory Requirements

Theorem 6.4 (Space Complexity)**.** *Storing a WPE-TME system with n components requires $O(n)$ space.*

Proof. Each component stores four parameters $(\Phi, \lambda, \theta, \kappa)$ plus structural information (layer, position in sequence). This gives constant space per component, hence $O(n)$ total. □

The framework's linear space complexity makes it feasible for large-scale systems. A system with 10,000 components requires approximately 40,000 parameter values plus structural overhead—easily manageable in contemporary computing environments.

6.3 Validation and Error Checking

[H] [1] ValidateWPEexpression Parse expression into component list each component c in list $c.\Phi$ not in valid domains ERROR_E001 (Invalid domain) $c.\lambda \notin [1, 9]$ ERROR_E002 (Shell out of range) $c.\theta \notin [0, 360)$ ERROR_E003 (Phase out of range) $c.\kappa \notin [-10, 10]$ ERROR_E004

(Curvature out of range) output count $\neq 1$ ERROR_E005 (Missing/multiple outputs) cycles detected in dependency graph ERROR_E006 (Circular dependency) VALID

[H] [1] ValidateTMEexpression Validate as WPE expression each layer L in expression $\theta_{prev} \leftarrow -1$ each component c in L $c.\theta < \theta_{prev}$ ERROR_E008 (Non-monotonic phase) $\theta_{prev} \leftarrow c.\theta$ VALID

6.4 Implementation Recommendations

Parsing: Implement a recursive descent parser for WPE expressions. The grammar is context-free and admits efficient parsing.

Representation: Store components in a graph structure where nodes represent components and edges represent coupling relationships with weights equal to coupling strengths.

Computation: Pre-compute frequently accessed values such as $\cos(\theta_i - \theta_j)$ and cache them. Trigonometric function evaluation dominates computational cost.

Temporal simulation: For TME systems, implement event-driven simulation where events are scheduled at times determined by phase progression and curvature values.

Visualization: Represent components as points on a circle with phase determining angular position. Draw edges between components with thickness proportional to coupling strength. Use concentric circles for different shell levels.

6.5 Numerical Considerations

Phase arithmetic: All phase calculations should use modulo 360° arithmetic. Implement careful handling of wraparound at boundaries.

Floating-point precision: Standard double-precision floating-point (64-bit) provides sufficient accuracy for coupling calculations. Single precision may introduce unacceptable errors in accumulated phase calculations.

Temporal scale selection: Choose α to keep timing calculations in a numerically stable range. Avoid extreme values that might cause overflow or underflow.

Curvature limiting: The framework restricts $|\kappa| \leq 10$ to prevent numerical instability. This gives minimum time constant $\tau_{min} = 0.1$ and maximum $\tau_{max} = 10$ in relative units before temporal scaling.

7 Discussion

7.1 Framework Strengths

The WPE-TME framework provides several key advantages for AI reasoning:

Explicit structure: All relationships are visible in the notation. Models can directly inspect coupling strengths, hierarchical influences, and temporal orderings. This visibility enables debugging, verification, and meta-reasoning.

Domain generality: The same geometric relationships apply across physics, biology, cognition, and social systems. This enables transfer learning and cross-domain analogy.

Compositional scalability: Complex systems build from simple components through well-defined operators. This supports both bottom-up construction and top-down decomposition.

Text-based representation: The framework operates entirely in the token space that language models natively process. No external data structures or numerical computations are required beyond what can be represented textually.

Model-agnostic design: The system functions through prompting alone. Any model that understands the syntax rules can use the framework immediately.

7.2 Limitations and Constraints

Several limitations constrain the framework's applicability:

Discrete phase resolution: Phase values are typically specified at integer degree resolution. This limits the precision of coupling calculations. Systems requiring very fine-grained distinctions may need higher resolution.

Static topology: WPE represents fixed system structure. Dynamic systems where components appear, disappear, or change relationships require extended notation or sequential re-encoding.

Continuous dynamics approximation: The framework discretizes inherently continuous dynamics. Phenomena requiring differential equation precision may not be adequately captured.

Scaling to large systems: While complexity is polynomial, very large systems (thousands of components) may become unwieldy for text-based manipulation.

Learning the syntax: Models must learn the WPE-TME rules through prompting or fine-tuning. This imposes a knowledge burden that may limit accessibility.

7.3 Comparison to Alternative Approaches

Neural graph networks: Graph neural networks learn relationships from data but lack explicit geometric semantics. WPE provides interpretable coupling mechanisms that GNNs learn implicitly.

Symbolic AI: Traditional symbolic systems use logic-based representations. WPE adds geometric and temporal dimensions that pure symbolic approaches lack.

Petri nets: Petri nets model concurrent systems with explicit temporal ordering. TME provides similar capabilities with additional geometric coupling and hierarchical structure.

Process algebras: Process calculi like CSP and -calculus model concurrent processes algebraically. TME adds geometric interaction mechanisms and hierarchical organization.

Category theory: Categorical approaches provide abstract compositional frameworks. WPE-TME offers more concrete geometric semantics suitable for numerical modeling.

7.4 Future Directions

Several extensions could enhance the framework:

Stochastic components: Add probability distributions over phase, curvature, or coupling strength to model uncertainty.

Adaptive topology: Develop notation for systems where structure evolves over time—components that split, merge, or rewire.

Higher-dimensional phase spaces: Extend beyond circular phase space to toroidal or spherical geometries for additional degrees of freedom.

Quantum extensions: Incorporate superposition and entanglement for quantum system modeling.

Learning from data: Develop algorithms to infer WPE-TME encodings from observed system behavior.

Automated reasoning: Build proof systems and automated theorem provers that operate on WPE-TME representations.

Integration with neural networks: Create hybrid architectures where neural networks learn to manipulate WPE-TME expressions as intermediate representations.

8 Conclusion

This paper has presented the Wave Pattern Encoding and Temporal Modulation Encoding framework, a geometric calculus for explicit structural and temporal reasoning in AI systems.

The framework addresses fundamental limitations in how language models represent system structure and temporal relationships.

The core contributions include:

1. A rigorous mathematical foundation based on phase space geometry, where component interactions derive from angular separation through cosine coupling relationships
2. A hierarchical shell mechanism where influence strengths follow from the formula $I = 1/\lambda_{low} - 1/\lambda_{high}$
3. A curvature parameter unifying spatial stability and temporal duration through $\tau = 1/|\kappa|$
4. A compositional syntax enabling complex system construction from simple primitives
5. An extension to explicit temporal representation through TME where time emerges from syntactic ordering
6. Validation across multiple domains demonstrating framework generality
7. Computational complexity analysis showing polynomial scaling properties

The framework operates entirely in text-based representations that language models natively process. It requires no architectural modifications and functions through prompting alone. This design enables immediate deployment on existing AI systems.

The geometric approach provides several key advantages. Structure becomes explicit and inspectable. Coupling strengths follow from geometric principles rather than learned patterns. Temporal relationships appear syntactically. Cross-domain analogies become visible through shared geometric patterns. Meta-reasoning becomes possible because the representation itself is manipulable.

However, the framework does not solve all problems. It provides a scaffold, not a replacement for statistical learning. Models must still learn the syntax rules. Large systems may become unwieldy in text form. Dynamic topologies require extended notation. Continuous phenomena are discretized.

The framework's ultimate value lies in its complementarity with existing approaches. Neural networks learn patterns from data. WPE-TME makes those patterns explicit. Statistical methods handle uncertainty. WPE-TME provides structural constraints. Symbolic reasoning manipulates abstract representations. WPE-TME adds geometric and temporal dimensions.

As AI systems grow more sophisticated, the need for explicit structure will increase. Long reasoning chains require stable scaffolding. Multi-step planning requires temporal representation. System understanding requires inspectable structure. The WPE-TME framework provides these capabilities through a mathematically principled geometric calculus.

The geometry computes. The syntax is the semantics. Structure and time become visible. This is the essential contribution: making explicit what would otherwise remain implicit, providing a scaffold for reasoning that language models can directly manipulate.

Acknowledgments

The authors thank the anonymous reviewers for their valuable feedback and suggestions that improved this paper.

References

[1] Brown, T., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877-1901.

[2] Battaglia, P. W., et al. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261.*

[3] Pearl, J. (2009). *Causality: Models, reasoning and inference* (2nd ed.). Cambridge University Press.

[4] Marcus, G. (2020). The next decade in AI: Four steps towards robust artificial intelligence. *arXiv preprint arXiv:2002.06177.*

[5] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, e253.

[6] Graves, A., et al. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626), 471-476.

[7] Vaswani, A., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998-6008.

[8] Chollet, F. (2019). On the measure of intelligence. *arXiv preprint arXiv:1911.01547.*

[9] Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1798-1828.

[10] Schrittwieser, J., et al. (2020). Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588(7839), 604-609.

[11] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, 27, 3104-3112.

[12] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8), 1771-1800.

[13] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

[14] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554-2558.

[15] Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1-2), 159-216.