STEM – Stabilizing the Stochastic with the Static: A Deterministic Spatiotemporal Embedding Model for Infrastructure-Aware Reasoning

Sean Tucker

Abstract

We introduce **STEM (Spatiotemporal Embedding Model)**, a deterministic and interpretable vector encoding system designed to stabilize the stochastic behavior of modern AI with a static, algebra-compatible base. STEM provides a reproducible embedding of space and time—anchoring AI reasoning in infrastructure-aware contexts such as transit planning, environmental risk, and emergency management. Unlike learned embeddings, STEM supports transparent, prompt-readable vector logic and enables vector-native spatial joins when used with ontology and geometry embeddings. We formalize the STEM formula, present **GeoMesh** as an optional global spatial substrate, and explain its foundational role in the forthcoming **MOSES** system for modular reasoning (Tucker, 2025b).

---

1. Introduction: Stabilizing the Stochastic with the Static

As AI systems increasingly interface with real-world environments, the ability to reason over space, time, and elevation is becoming essential—especially in infrastructure-critical domains such as transportation, urban planning, and disaster response [Goodchild, 2009; Anselin, 2019]. However, the majority of vector embeddings in today's AI systems—especially in LLMs and multimodal models—are stochastic, opaque, non-reproducible, and algebraically intractable (Bommasani et al., 2021; Liu et al., 2021; Mai et al., 2022; Bengio et al., 2021). While these models offer impressive generalization, they often lack referential precision, logical decomposability, and deterministic grounding. In domains such as urban infrastructure, disaster planning, and geospatial intelligence, this poses critical challenges (Cai & Wang, 2020; Goodchild, 2009).

To address these limitations, we build upon the **Statically Implied Models (SIMs)** framework [Tucker, 2025a], which defines a class of deterministic, interpretable embedding models. SIMs provide a model-theoretic foundation for transparent, symbolic-compatible vector reasoning. They are especially useful when integrating large-scale knowledge systems with **vector databases**, **retrieval-augmented generation**, or **LLMs**, where static and structured reference logic is required alongside dynamic reasoning [Tucker, 2024].

We propose **STEM**: a deterministic, algebra-compatible embedding that encodes latitude, longitude, altitude, and time into a fixed-length vector with known structure and reproducibility. STEM is the first canonical implementation of a **SIM (Statically Implied Model)** (Tucker, 2025a).

STEM encodes geospatial and temporal context into a reproducible vector that:

- Represents spatial and cyclic temporal information algebraically
- Supports deterministic proximity, delta-time, and elevation logic
- Allows **Prompt-Readable Vector Logic** by exposing vector components to prompts or symbolic reasoning modules

STEM is designed to stabilize stochastic AI systems by introducing a **static substrate for space-time reasoning**—providing reliable anchors for infrastructure-aware applications.

In the forthcoming **MOSES system** (Modular Onto-Logic Spatial Embedding System), STEM is not a standalone model but a **shared vector component** embedded within all geo-based SIMs—including **GOE** (Geospatial Onto-logical Embeddings) and **GPLE** (Geospatial Poly-Linear Embeddings). These SIMs embed a stem_vector field derived from STEM, enabling **co-referencing**, **vector-native joins**, and **multi-type model integration** in vector databases like Weaviate (Yan et al., 2021; Tucker, 2025b).

Table 1.1 Outlines the required properties of STEM.
Table 1.1 - Properties of STEM

Property	Description
Deterministic	Same inputs always yield the same vector [Tucker, 2025a]
Decomposable	Each component corresponds to a human-interpretable feature
Algebra-Compatible	Supports arithmetic operations (e.g., vector difference = temporal or spatial displacement)
Prompt-Compatible	Vectors and subcomponents can be embedded in natural language prompts [Kojima et al., 2022]
Vector-DB Ready	Works in near_vector queries in Weaviate and similar platforms [Yan et al., 2021]
Infrastructure-Ready	Reusable across planning, routing, and disaster applications [Goodchild, 2009]

---

2. Formula and Implementation

Given:

- **Latitude**: lat (in degrees)
- **Longitude**: lon (in degrees)
- **Altitude**: alt (in meters)
- **Time of day**: t (in seconds since midnight)

We define the STEM vector as:

$$STEM(lat,lon,alt,t)=[\sin(lat),\cos(lat),\sin(lon),\cos(lon),alt/h,\sin(2\pi t/T),\cos(2\pi t/T)]$$

Where:

- h is the maximum expected terrestrial altitude (default: 9000m)
- T is the duration of the daily time cycle (86,400 seconds)

This results in a **deterministic**, algebraic, and prompt-compatible vector.

---

3. Scaling and Geodetic Considerations

Altitude Normalization

The alt/h component of STEM vectors encodes normalized elevation, where h represents the maximum expected altitude for the application domain. This introduces a scaling dependency: for urban environments (e.g., NYC), h = 500m may be appropriate, while mountainous regions may require larger values. Three strategies are proposed:

- Use a **universal ceiling** (e.g., 8,849m, Mount Everest) for global consistency
- Use **domain-aware scaling** for local fidelity
- Apply **logarithmic or percentile-based transforms** to reduce sensitivity to outliers

A default of h = 10,000m ensures consistency across most terrestrial contexts while enabling interpretable vector math. The SIM framework permits flexible schema-level encoding of h to support adaptive implementations.

Latitude/Longitude and Geodetic Distortion

STEM's sin(lat), cos(lat), sin(lon), cos(lon) encoding captures Earth's curvature but does not account for the **shrinking real-world distance of longitudinal degrees** at higher latitudes. For most regional-scale applications, this distortion is minimal. However, global or polar use cases may require:

- A **cos(lat)** correction factor on longitude
- A **projection into UTM or local tangent plane coordinates**

These refinements may be implemented in higher-order SIMs or system-specific preprocessors. Alternatively, they can be bypassed altogether using the **GeoMesh** base layer introduced below.

---

4. GeoMesh: A Global STEM Base Layer

To support high-precision reasoning and global interoperability, we propose **GeoMesh**: a deterministic, globally tiled, high-resolution spatial base layer derived from satellite imagery and elevation models, designed as a shared anchor for AI systems using STEM. Each **GeoMesh tile**:

- Represents a fixed spatial extent (e.g., $1m^2$ or $10m^2$)
- Has a known lat, lon, alt centroid
- Is optionally precomputed with a STEM vector

Can be assigned a unique mesh ID (e.g., GMESH[x][y][z])

GeoMesh enables:

- **Geodetic alignment** of STEM vectors across Earth's curvature
- **Snapping and interpolation** of features to canonical grid cells
- **Cross-region consistency** and mapping resolution
- **Projection-free reasoning** across systems without requiring geodesic math

STEM can embed a **GeoMesh ID** or vector interpolation index, serving as a reference layer for precise location anchoring and downstream reasoning.
GeoMesh enables AI systems—particularly LLMs, agents, and infrastructure logic engines—to operate with **metric-aware, spatially aligned grounding** without requiring

geodetic expertise. It can be embedded in prompts, ingested into vector search, or used for SIM-based inter-model referencing.

---

5. Prompt-Readable Vector Logic

STEM enables **Prompt-Readable Vector Logic**—the ability to reason over vector operations in plain language and symbolic logic (Kojima et al., 2022; Liu et al., 2021).

Examples:

- “Which areas are uphill from this location?”
- “What is the time difference between now and station arrival?”
- “Find assets within 15 minutes of flood zones.”

Because STEM vectors are **decomposable** and **predictable**, they can be injected into:

- LLM prompts
- Function-calling chains
- Symbolic reasoning workflows
- RAG (retrieval-augmented generation) pipelines (Tucker, 2025a)

---

6. Use Case: Infrastructure-Aware RAG

In an infrastructure AI system:

- **Subway stations** are embedded with STEM
- **Flood zones** use STEM + GOE
- **Paths or regions** use STEM + GPLE

A planner queries:

“Which stations are open, not flooded, and reachable within 15 minutes walking?”

The system:

- Compares stem_vector proximity
- Filters by **GOE** semantic type (e.g., station)
- Computes path feasibility via **GPLE** and elevation differentials

- Produces explainable results grounded in space and time

This multi-model, multi-type reasoning is possible due to **shared STEM anchoring**.

---

7. Role in MOSES and Model Integration

In **MOSES** (Tucker, 2025b), STEM acts as the **foundational spatiotemporal anchor**. Each SIM—whether GOE (ontology), GPLE (geometry), or others—**includes a stem_vector**, enabling:

- **Vector-native joins** in vector databases (e.g., Weaviate)
- **Co-referencing** across different data types (e.g., bridge, path, sensor)
- **Query fusion**: enabling logic over location, semantics, and topology simultaneously

This architecture makes MOSES modular without sacrificing grounding—allowing learned, symbolic, and static reasoning to co-exist over shared coordinates.

---

8. Cognitive Foundations: Why Anchoring Matters

STEM's role mirrors cognitive mechanisms in human reasoning:

- **Modularity of Mind** (Fodor, 1983): Specialized modules (language, space, time) use shared referents for coherence.
- **Multiple Intelligences** (Gardner, 1983): Spatial and logical intelligences interact through a common context.
- **Parallel Distributed Processing** (Rumelhart et al., 1986): Distributed symbolic processing relies on common anchors.
- **Embodied Cognition** (Wilson, 2002): Spatial awareness underlies abstract reasoning.

Like the brain, AI systems benefit from **anchoring mechanisms** that orient thought and inference. STEM provides this function for vector-native AI.

---

9. Comparison: STEM vs. Learned Embeddings

Feature	STEM	Learned Embeddings (e.g., GeoBERT)
Deterministic	✅	❌
Interpretable	✅	❌
Prompt-readable	✅	❌
Supports vector math	✅	⚠️ Partial
Geodetically accurate	⚠️ (with GeoMesh)	❌
Modularity (MOSES integration)	✅	❌

STEM is ideal for environments demanding **traceability, precision, and symbolic integration**—e.g., infrastructure, disaster planning, public safety.

---

10. Conclusion

STEM offers a foundational layer for **deterministic, infrastructure-aware AI**:

- Anchors space and time in reusable vector format
- Enables modular reasoning across embedding types
- Stabilizes stochastic LLMs with static, prompt-readable vectors
- Forms the spatial substrate for MOSES and future SIMs (Tucker, 2025a; 2025b)

STEM is not just a technical encoding—but a **cognitive and architectural substrate** for transparent, modular intelligence in AI.

References

- Bommasani, R. et al. (2021). *On the Opportunities and Risks of Foundation Models*. arXiv:2108.07258.
- Cai, L., & Wang, J. (2020). *A survey of explainable AI: Interpretability, fairness, and robustness*. arXiv:2006.11371.
- Fodor, J. A. (1983). *The Modularity of Mind*. MIT Press.
- Gardner, H. (1983). *Frames of Mind: The Theory of Multiple Intelligences*. Basic Books.
- Goodchild, M. F. (2009). *Geographic information systems and science: today and tomorrow*. Annals of GIS, 15(1), 3–9.
- Kojima, T. et al. (2022). *Large Language Models are Zero-Shot Reasoners*. arXiv:2205.11916.
- Liu, P. et al. (2021). *Pre-train Prompt: Deep Transfer Learning by Pre-Training Prompts*. arXiv:2107.13586.
- Rumelhart, D. E., McClelland, J. L., & PDP Research Group. (1986). *Parallel Distributed Processing*. MIT Press.
- Tucker, S. (2025a). *SIM: Statically Implied Models for Transparent and Interoperable Embedding of Spatiotemporal Knowledge*. Preprint.
- Tucker, S. (2025b). *MOSES: A Modular Onto-Logic Spatial Embedding System for Reasoning with SIMs*. In preparation.
- Wilson, M. (2002). *Six views of embodied cognition*. Psychonomic Bulletin & Review, 9(4), 625–636.
- Yan, B., Mai, G., & Janowicz, K. (2021). *Vector-based geographic information systems (V-GIS): A conceptual framework*.