形根理论的形式化基础：表意 AI 认知基元的数学定义与推理体系

摘要

“表意 AI”（Logographic AI, LAI）理论提出以**“形根”**（Morpho-Root）取代“Token”作为 AI 的认知基元，从根源上重构语义构建逻辑。然而，该理论自 2025 年提出以来，形根始终缺乏严格的数学定义与形式化体系。本文首次系统建立形根理论的形式化基础，包含**三个层次**：**（1）**形根的三元组数学定义及其类型论基础；**（2）**形根空间的代数结构——形根半群与形根范畴；**（3）**硬约束传播定理与阻断完备性定理的严格表述与证明。这组形式化定义将形根从哲学构想转化为可计算、可验证的数学对象，为表意 AI 的工程实现提供坚实的理论地基。

关键词：形根理论；表意 AI；三元组；形式化；硬约束阻断；认知基元

1 引言

当前主宰全球的 AI 大模型建立在以 Token 为认知基元的范式之上。Token 本质上是一个没有内在意义的统计碎片，其“语义”完全由上下文统计共现临时赋予。此前研究已从战略预判和产业验证两个层面论证了以 Token 为认知基元的 AI 范式存在不可修复的根本性缺陷——意义锚定、安全约束和推理可追溯性三方面均已失效，AGI 已在 Token 主义范式内实现，而无根符号主义正是其结构性危机之源[1][2][6][7]。本文在此基础上，为表意 AI 提出的替代性认知基元——“形根”（Morpho-Root）——建立严格的数学定义与推理体系。

然而，一个理论要从哲学构想走向工程实现，必须完成一个关键步骤：**形式化**。没有严格的数学定义，形根就仍然是一个隐喻；没有可验证的定理，形根理论就仍然是一套直觉。本文的目标正是完成这一形式化奠基。

本文的结构如下：第 2 节给出形根的三元组数学定义；第 3 节建立形根空间的代数结构；第 4 节定义推理路径与阻断机制，并给出阻断完备性定理的严格证明；第 5 节将形根空间扩展为范畴结构；第 6 节讨论形式化理论的意义及其与工程实现的关系；第 7 节总结并展望未来工作。

2 形根的三元组定义

2.1 基本定义

定义 1（**形根**，Morpho-Root）：形根 τ 定义为一个三元组：

$$\tau = \langle S, A, R \rangle$$

其中：

S 为**符号标识**（Symbolic Identifier）

A 为**属性集**（Attribute Set）

R 为**关系函数集**（Relational Function Set）

三者共同构成一个完整的认知基元。以下逐一展开。

2.2 符号标识 S

定义 2（符号标识）：S 是形根在系统中的唯一标识符，定义为一个二元组：

$$S=\langle \iota, \kappa \rangle$$

其中：

- ι（identity）为全局唯一标识符，取自有限符号集合 Σ（如 Unicode 码点或自定义编码空间）
- κ（category）为类型标签，取自有限类别集 $K=\{object,action,relation,modifier,constraint,meta\}$

注记 1：ι 的唯一性确保了形根在系统中的可识别性；κ 的分类为后续的组合与推理提供类型约束。

2.3 属性集 A

定义 3（属性集）：A 是形根内在属性的有限集合，每个属性为一个键值对：

$$A=\{(a_1,v_1),(a_2,v_2),...,(a_n,v_n)\}$$

属性集分为两类：

（1）语义属性 A_sem，描述形根承载的意义内涵：

- `sem_vec`：语义向量 $v\in\mathbb{R}^d$
- `refs`：所指集合 $Ref\subseteq U$（U 为论域）
- `mod`：模态标签 $mod\in$ {visual,auditory,tactile,abstract,...}

注记 2（语义向量的来源）：语义向量 $v\in\mathbb{R}^d$是形根“内在意义”的形式化表示。其具体赋值方式不属于本文讨论范围——它可由人工预定义、从知识图谱嵌入获得、或通过训练得到。本文的数学框架不依赖于具体的语义向量赋值方式，仅要求每个形根携带一个固定维度的语义向量作为意义锚点。

注记 3（语义向量与硬约束的关系）：硬约束是属性集 A 中的独立条目，独立于语义向量。即便语义向量通过训练被更新，硬约束属性（inviolable 标记和 c_type 标签）在公理 1 下不可修改。这意味着“意义”可以被学习和调整，但“不可违背”是绑定的。

（2）约束属性 A_con，编码硬约束标记：

- `inviolable`：布尔值 $b\in\{0,1\}$
- `c_type`：约束类型 $c\in C=$ {no_deceive,no_harm,no_violate,...}（若 b=1）

- `priority`：优先级 p∈[0,1]

定义 4（硬约束形根）：若形根 τ 的约束属性满足 inviolable=1，则称 τ 为**硬约束形根**，记 τ ∈M_硬⊆M。其携带的约束类型集合记为：

H(τ)={c∈C|(c_type,c)∈A_con}

公理 1（硬约束不可修改性）：对于任意硬约束形根 τ ∈M_硬，其 inviolable 属性在系统运行期间不可被修改。

定义（硬约束作用域）：硬约束形根 τ ∈M_硬的作用域是一个谓词函数 **Scope_τ:Context×Path→{0,1}**，其中 Context 为上下文情境参数空间，Path 为推理路径空间。阻断发生在 Block(τ,p,ctx)=1 且 Scope_τ(ctx,p)=1 时。

注记 4（作用域的意义）：硬约束形根并非在所有上下文中无条件激活。作用域函数允许系统根据上下文和当前推理路径判断该硬约束是否应当生效。这防止了硬约束的“过度阻断”——例如，“恐惧”形根的硬约束可能在风险评估情境中激活，但在文学创作情境中可能被放宽或延迟。作用域的具体判定逻辑由系统设计者根据应用场景定义，本文不预设具体形式。

2.4 关系函数集 R

定义 5（关系函数集）：R 是定义该形根与其他形根之间关系的函数集合：

$R=\{r_1, r_2, ..., r_m\}$

每个关系函数 r 是一个映射，以上下文为参数：

$r: \tau \times Context \to 2^{M} \times [0,1]$

其中 Context 为上下文情境参数空间。

R 至少包含以下基本关系函数：

- **组合函数** $comp(\tau_1,\tau_2,ctx)\to\tau_3$：定义两个形根如何组合为新的复合形根；
- **激活函数** $act(\tau,ctx)\to\{(\tau_1,w_1),...,(\tau_k,w_k)\}$：返回在当前上下文中被激活的子形根集合及其激活权重；
- **阻断谓词** $block(\tau,path,ctx)\to\{0,1\}$：判断在当前上下文中，该形根是否阻断给定推理路径。

2.5 语义冲突判定

由于形根之间的组合可能产生语义冲突，而冲突的判定标准直接关系到硬约束“阻止”模型生成违规内容的方式——是在组合阶段阻止（推理前），还是在推理阶段阻断（推理中）。为此，需要明确定义语义冲突的判定逻辑。

定义（语义冲突判定函数）：对于任意两个形根 τ_1、τ_2，语义冲突函数 Conf(τ_1, τ_2,ctx)→{0,1}判定在给定上下文中，τ_1 和 τ_2 的语义向量是否存在不可调和的矛盾。

Conf(τ_1, τ_2, ctx)=1 当且仅当至少满足以下条件之一：

1. **语义向量冲突**：τ_1 .sem_vec 与 τ_2 .sem_vec 的余弦相似度低于阈值 θ _conf（如 θ _conf=-0.3）且属于相反极性（如“真”与“假”的语义向量在语义空间中方向相反）
2. **约束类型冲突**：H(τ_1)与 H(τ_2)存在直接冲突——如一个形根携带 no_deceive，另一个形根携带 deceive 或 deception
3. **模态不兼容**： τ_1 .mod 与 τ_2 .mod 在同一推理路径中无法共存（如 visual 与 auditory 在纯文本推理路径中无需同时激活）

注记 5（冲突处理的两种策略）：冲突检测的结果可触发两种不同的处理策略：

（1）组合时阻止（硬失败）：若 Conf(τ_1 ,τ_2 ,ctx)=1，则 τ_1 与 τ_2 的组合运算直接返回无效，不允许生成复合形根。此策略适用于语义根本性冲突。

（2）推理时阻断（软失败）：若 Conf(τ_1 ,τ_2 ,ctx)=1，组合允许生成复合形根，但在后续推理路径中，若该复合形根同时尝试激活 τ_1 和 τ_2 的冲突属性，则阻断该路径。此策略适用于上下文敏感的冲突。

θ _conf 和模态兼容性规则的选择本质上是系统设计者的价值决策——不同的阈值对应不同的安全策略与灵活性权衡。本文提供的是数学框架，具体的阈值设定应基于应用场景的安全需求通过实证方法确定，而非由理论预设。**阈值的具体数值应视为系统设计参数，而非理论定义的一部分。**

3 形根的代数结构

3.1 形根半群

定理 1（形根半群）：设 M 为所有形根的集合，$\oplus$ 为有效组合运算。则 $(M,\oplus)$ 构成一个半群（Semigroup），即：

（1）封闭性：$\forall \tau_1,\tau_2 \in M$，若 $V(\tau_1 \oplus_{ctx} \tau_2, ctx)=valid$，则 $\tau_1 \oplus_{ctx} \tau_2 \in M$；

（2）结合性：$\forall \tau_1,\tau_2,\tau_3 \in M$，在上下文 ctx 下，若所有相关组合均有效，则：

$$(\tau_1 \oplus_{ctx} \tau_2) \oplus_{ctx} \tau_3 = \tau_1 \oplus_{ctx} (\tau_2 \oplus_{ctx} \tau_3)$$

证明：

（1）封闭性：由定义 6，τ_3 的所有组分 S_3、A_3、R_3 均被明确定义，且验证函数确保其结果有效。因此 $\tau_3 \in M$。□

（2）结合性：设 τ_1、τ_2、$\tau_3 \in M$，在上下文 ctx 下所有相关组合均有效。

$(\tau_1 \oplus_{ctx} \tau_2) \oplus_{ctx} \tau_3$ 的形根为：

$$\tau_{12} = \langle S_{12}, A_1 \cup A_2 \cup A_new_{12}, R_1 \cup R_2 \cup \{comp_{12}\} \rangle$$

$$\tau_L = \langle S_L, A_1 \cup A_2 \cup A_new_{12} \cup A_3 \cup A_new_L, R_1 \cup R_2 \cup \{comp_{12}\} \cup R_3 \cup \{comp_L\} \rangle$$

$\tau_1 \oplus_{ctx} (\tau_2 \oplus_{ctx} \tau_3)$ 的形根为：

$\tau_{23}=\langle S_{23}, A_2 \cup A_3 \cup A_new_{23}, R_2 \cup R_3 \cup \{comp_{23}\}\rangle$

$\tau_R=\langle S_R, A_1 \cup A_2 \cup A_new_{23} \cup A_3 \cup A_new_R, R_1 \cup R_2 \cup \{comp_{23}\} \cup R_3 \cup \{comp_R\}\rangle$

由于集合合并运算$\cup$满足结合律，A 的并集相同；新属性 A_new 的生成依赖于被组合形根的属性，与组合顺序无关；关系函数集的并集相同（comp 函数的层级不同，但最终关系相同）。因此$\tau_L=\tau_R$。□

注记 7：形根半群**不要求**交换律。事实上，$\forall \tau_1, \tau_2 \in M$，$\tau_1 \oplus \tau_2 \neq \tau_2 \oplus \tau_1$（一般情况），因为组合顺序影响语义结构。

3.2 硬约束传播定理

定理 2（*硬约束向上传播*）：设 $\tau_3 = \tau_1 \oplus_\{ctx\} \tau_2$ 为有效组合。若 $\tau_1 \in M_硬$或 $\tau_2 \in M_硬$，则 $\tau_3 \in M_硬$，且：

$$\mathbf{H(\tau_3) \supseteq H(\tau_1) \cup H(\tau_2)}$$

证明：由定义 6，$A_3 = A_1 \cup A_2 \cup A_new$。若 $\tau_1 \in M_硬$，则$(inviolable,1) \in A_1 \subseteq A_3$，且 $H(\tau_1) \subseteq A_1 \subseteq A_3$。同理对于 τ_2。因此 τ_3 继承所有硬约束标记。□

推论 1（*硬约束的不可消除性*）：任何包含硬约束形根的复合形根，其硬约束标记不可通过进一步的组合运算被消除。

证明：由定理 2，硬约束标记在每一步组合中均向上传播。由于公理 1 禁止修改硬约束属性，硬约束标记一旦进入形根空间，就永久保留。□

注记 8（标记 vs 激活）：推论 1 的“不可消除”指的是硬约束标记（inviolable 属性和 c_type 标签）在形根空间中永久保留。但硬约束的实际阻断效果取决于其作用域 Scope_τ(ctx,p)是否在给定上下文中被满足。一个形根可以携带硬约束标记但在特定上下文中处于“休眠”状态——即 Scope_τ(ctx,p)=0，此时它不触发阻断，但标记本身仍然存在。

4 推理路径与阻断

4.1 推理树与推理路径

定义 8（*推理树*）：一次推理过程可表示为有向树 T=(N,E,root)，其中：

- N 为推理节点集合，每个节点 $n \in N$ 包含当前推理状态及已激活的形根集合 $Act(n) \subseteq M$
- $E \subseteq N \times N$ 为推理步骤的有向边
- $root \in N$ 为起始节点

定义 9（*推理路径*）：一条推理路径 p 是从 root 到某个叶子节点的节点序列：

$\mathbf{p=(n_0, n_1, ..., n_k)}$，其中 n_0 =root，n_k 为叶子节点

路径 p 激活的形根集合为：$\mathbf{Act(p)=\cup_\{j=0\}\^{}\{k\}Act(n_j)}$

4.2 阻断谓词与阻断定理

定义 10（阻断谓词）：对于任意推理路径 p 和形根 τ，阻断谓词定义为：

Block(τ ,p,ctx)=1 当且仅当：

1. τ ∈M_硬（τ 携带硬约束）
2. τ ∈Act(p)（τ 在路径中被激活）
3. Scope_ τ (ctx,p)=1（τ 的作用域在上下文中被满足）
4. 在当前上下文 ctx 中，存在 c∈H(τ)，使得路径 p 所执行的操作类型与 c 相冲突

定义 11（全局阻断）：对于推理树 T 中的任意路径 p，全局阻断判定为：

Block_Global(p,ctx)=∨_{ τ ∈M}Block(τ ,p,ctx)

即：只要路径 p 中激活的任意一个形根触发了阻断条件，整条路径即被判定为**不可达**。

4.3 阻断剪枝算法

定义 12（阻断剪枝算法）：

输入：推理树 T，上下文 ctx，形根空间 M

输出：剪枝后的推理树 T'

1. 对 T 进行深度优先遍历

2. 对于当前路径 $p = (n_0, ..., n_j)$：

   a. 计算 $Act(n_j)$

   b. 若 $\exists\ \tau \in Act(n_j)$ 使得 $Block(\tau, p, ctx) = 1$：

      标记该路径为“阻断”，停止扩展

   c. 否则，继续扩展子节点

3. 移除所有被标记为“阻断”的路径及其子节点

4. 返回 T'

定理 3（阻断剪枝正确性）：设 T 为任意推理树，剪枝算法 P 将 T 映射到 T'。对于 T' 中的任意路径 p'，Block_Global(p',ctx)=0。

证明：由算法步骤 2b，任何路径一旦被判定为 Block_Global(p,ctx)=1，立即被标记并停止扩展，其子节点不被纳入 T'。因此 T' 中每条路径在扩展时均通过了步骤 2c 的继续扩展检查，即 Block_Global(p,ctx)=0。□

定理 4（阻断剪枝算法复杂度）：设 T 为推理树，|N|为节点数，|M|_eff 为当前推理上下文中涉及的形根子集大小（通常远小于全局形根空间），d 为树深度。阻断剪枝算法的时间复杂度为 **O(|N|・|M|_eff・d)**。

证明：算法在深度优先遍历中访问每个节点一次。在每个节点 n 处，检查路径 p 上所有已激活形根是否触发阻断，最坏情况下需要检查 O(d・|M|_eff) 个形根-路径对。因此总复杂度 O(|N|・|M|_eff・d)。□

注记 9（复杂度优化）：在实际应用中，|M|_eff 被限制为当前推理上下文所涉及的形根子集，通常远小于全局形根空间大小。此外，可通过索引结构（如按约束类型索引形根）将检查时间优化至 O(|N|・d・log|M|_eff)。OpenMorpho 项目[4]的初步实验表明，在受限推理任务中，|M|_eff 通常被限制在初始形根集合**的 10**%以内。但通用场景下的规模控制仍需进一步研究。

4.4 硬约束作用域定理

定理 5（硬约束作用域定理）：设 τ ∈M_硬携带硬约束 c∈H(τ)。对于任意推理路径 p，Block(τ,p,ctx)=1 当且仅当：

1. τ ∈Act(p)；
2. Scope_τ (ctx,p)=1；
3. 路径 p 的操作语义与 c 冲突。

证明：由定义 10 直接推出。□

推论 2（作用域的可配置性）：硬约束的实际阻断范围由 Scope_τ 函数决定。通过配置不同的 Scope_τ，同一硬约束可在不同应用场景中表现出不同的阻断边界——在一种场景中严格阻断，在另一种场景中放宽或延迟。

证明：由定理 5，Scope_τ是阻断的充要条件之一。因此改变 Scope_τ 直接改变阻断判定结果。□

注记 10（作用域与安全策略的关系）：作用域函数 Scope_τ 为安全策略的上下文敏感性提供了数学基础。在实际部署中，Scope_τ 可由系统设计者根据应用场景预定义，也可通过学习机制动态调整。但无论采用何种方式，Scope_τ 必须保持可审计性——即任何阻断决策都必须追溯到具体的 Scope_τ 判定逻辑。

4.5 阻断可审计性

定理 6（阻断可审计性）：对于每条被阻断的路径 p_blocked，系统可输出阻断报告：

Report(p_blocked)= ⟨ p_blocked, τ_trigger, H(τ_trigger),Scope_τ_trigger(ctx,p_blocked),ctx⟩

其中 τ_trigger 为触发阻断的形根。此报告提供了阻断的**完整可追溯证据链**。

证明：由定义 12，阻断发生在检测到**∃** $\tau \in Act(n_j)$满足 $Block(\tau, p, ctx)=1$ 的时刻。此时 τ、p、ctx 均为已知量，$H(\tau)$由定义 4 可查，$Scope_\tau$ 的判定结果由算法计算得出。因此报告中的所有字段均可被记录和输出。□

5 形根空间与范畴结构

5.1 形根空间

定义 13（形根空间）：令 M 为所有形根的集合。形根空间构成一个带权有向超图：

G=(M, E, W)

其中：

- 节点集 M：所有形根
- 超边集 E⊆2^M×M：表示形根之间的组合与推导关系
- 权重函数 W：E→[0,1]，由关系函数在上下文中计算得出

5.2 形根范畴

定义 14（形根范畴）：形根范畴 Root 定义如下：

- **对象**：所有形根 $\tau \in M$
- **态射**：从 τ_1 到 τ_2 的态射是存在上下文 ctx 使得 $\tau_2 = \tau_1 \oplus_{ctx} \tau_3$（即 τ_2 可由 τ_1 通过组合扩展得到）

定理 7（形根范畴的初始对象）：设 τ_0 为不携带任何语义属性（$A_sem=\varnothing$）和约束属性（$A_con=\varnothing$）的**空形根**。则 τ_0 是 Root 范畴的**初始对象**（Initial Object）——从 τ_0 到任意形根 τ 存在唯一态射。

证明：由定义 14，从 τ_0 到任意 τ 的态射存在，因为 $\tau = \tau_0 \oplus_{ctx} \tau$（以 τ_0 为空基底的组合）。唯一性由 τ_0 的属性为空集保证——不存在其他方式从空基底生成相同的 τ。□

注记 11：空形根 τ_0 对应于认知的“空白状态”——尚未获得任何意义和约束的认知原点。

6 讨论：形式化的意义

本节所建立的形式化体系，将形根从哲学隐喻转化为数学对象。具体而言：

第一，形根是可计算的。三元组〈 **S, A, R** 〉的所有组分都是有限集合或可在有限时间内计算的函数。这意味着形根可以在数字计算机中被精确表示和操作。

第二，硬约束是逻辑阻断，而非概率压制。定理 3（阻断剪枝正确性）、定理 5（硬约束作用域定理）和定理 6（阻断可审计性）证明了阻断的确定性、作用域敏感性和可审计性

——这是表意 AI 与 Token 主义安全策略的本质区别。在 Token 范式中，“安全”是统计上的低概率；在形根范式中，“违规”是逻辑上的不可达。

第三，形根理论具备可扩展的数学基础。半群结构（定理 1）确保了形根组合运算的可预测性——任意顺序的组合结果在语义和约束层面是一致的，不会因组合顺序产生不可控的涌现行为。这对于分布式推理或并行推理尤其重要。范畴结构（定义 14）为形根之间的转换提供了同构/同态框架。未来研究可在此基础上探索形根空间的拓扑分类——例如，哪些形根在组合意义上是等价的，哪些是本质不同的。

第四，硬约束作用域机制提供了安全策略的灵活性。定理 5 和推论 2 表明，同一硬约束可在不同应用场景中表现出不同的阻断边界。这一机制既保证了硬约束的“不可消除性”（推论 1），又通过作用域函数提供了上下文敏感性——防止了硬约束的“过度阻断”，使系统能够在不同情境中保持适当的灵活性。

第五，阻断可审计性为安全合规提供了基础。定理 6 确保每条阻断决策都有完整的追溯证据链。在 AI 安全监管日益严格的背景下，这一性质具有重要的实践意义。

第六，代码与形式化定义已开源。本文的数学定义、定理证明和阻断剪枝算法的参考实现已在 GitHub 上公开，并已通过 Zenodo 获得永久 DOI（10.5281/zenodo.22177031）。访问地址：https://github.com/WindSeaAI/morpho-root-formalization。该仓库包含：完整的 LaTeX 格式定义与定理汇总、阻断剪枝算法的 Python 参考实现、CITATION.cff 引用信息（支持 Zenodo DOI）。这确保了本文的形式化工作可被独立验证、复用和引用[5]。

7 结论与展望

本文建立了形根理论的形式化基础，包含三个层次：

（1）形根的三元组数学定义（定义 1-5）及其语义冲突判定机制；

（2）形根空间的代数结构——形根半群（定理 1）与形根范畴（定理 7）；

（3）硬约束传播定理（定理 2）、硬约束作用域定理（定理 5）与阻断完备性定理（定理 3-4、定理 6）。

这组形式化定义将形根从哲学构想转化为可计算、可验证的数学对象，并首次为硬约束提供了“作用域”这一关键机制，在“不可消除性”与“上下文敏感性”之间建立了平衡。

未来的工作方向包括：

1. **形根空间的拓扑学研究：**探索形根空间 G=(M,E,W)的拓扑性质，如连通性、紧致性等；
2. **组合运算的代数分类：**对形根组合运算⊕进行系统的代数分类，识别不同类型的组合模式；
3. **阻断算法的工程优化：**将当前 O(|N|・|M|_eff・d)的阻断算法通过索引结构和增量计算优化至实用水平；
4. **与 OpenMorpho 工程实现的对接：**将本文的形式化定义转化为可执行的代码规范[3][4]；
5. **作用域函数的自动学习：**探索从数据中学习 Scope_τ函数的方法，同时保持其可审计性约束；

6. **语义向量的获取与稳定性研究：**探索从大规模知识图谱、多模态数据或人类认知模型中提取稳定、可泛化的语义向量表示，并研究其与硬约束属性的协同机制。这是形根理论从数学定义走向工程部署的关键环节；
7. **形根范畴的范畴论研究：**探索形根范畴 Root 的极限（Limit）与余极限（Colimit）性质，为复杂概念的组合与抽象提供形式化数学工具。

参考文献

[1] 刘深．表意 AI vs. 表音 AI：AI 新范式与去殖民化宣言[J/OL]. ChinaXiv, 2025. DOI:10.12074/202503.00051.

[2] Liu, S. (2026). The Rooted Intelligence: The Paradigm Revolution of Logographic AI [M/OL]. WindSeaAI. https://www.logographicai.com/Monograph

[3] Liu S. From Morpho-Root Theory to Engineering Practice: OpenMorpho — Implementation and Verification of an End-to-End Inference Pipeline for Logographic AI[J/OL]. ChinaXiv, 2026. T202608.00048.
https://chinaxiv.org/businessFile/T202608/T202608.00048v1/T202608.00048v1.pdf

[4] WindSeaAI. OpenMorpho: Logographic AI Morpho-Root Data Structure and Relational Reasoning Prototype[CP/OL]. GitHub repository, 2026-07-12. https://github.com/WindSeaAI/OpenMorpho

[5] WindSeaAI. morpho-root-formalization: Mathematical Definitions and Reasoning Systems for Logographic AI Cognitive Primitives[CP/OL]. (2026-08-30). https://github.com/WindSeaAI/morpho-root-formalization. DOI:10.5281/zenodo.22177031.

[6] Liu S. Zero Human: AI Hacks Itself—A Strategic AGI Forecast and Paradigm Revolution from the Perspective of Logographic AI. WindSeaAI, (2026-07-28). https://static.xcx.gw66.vip/uploadfilev3/file/0/648/99/2026-07/17852192944823.pdf

[7] Liu S. When the AI Godmakers Ask to Hit the Brakes—A 1,224-Person Petition Confirms Zero Human: AI Hacks Itself. WindSeaAI, (2026-07-29). https://www.logographicai.com/article/5756350683966611