arXiv:2601.01298v1 [cs.LG] 3 Jan 2026

Warp-Cortex：一种面向消费级硬件上百万智能体认知扩展的异步、内存高效架构

Jorge L. Ruiz Williams Warp Research
https://github.com/JorgeLRW/warp-cortex

2026年1月6日

摘要

当前的多智能体大语言模型（LLM）框架受限于内存随规模线性增长的问题，使得在消费级硬件上进行“系统 2”并行推理并不切实际。我们提出**Warp Cortex**，这是一种异步架构，通过将智能体逻辑与物理内存解耦，理论上可实现百万智能体的认知扩展。借助**单例权重共享** 和一种受拓扑数据分析（TDA）混合地标化技术启发的新型**拓扑突触**，我们将权重的内存复杂度从 $O(N \cdot L)$ 降至 $O(1)$，并将上下文的内存复杂度降至 $O(N \cdot k)$，其中 $k \ll L$。我们将键值缓存视为潜在空间中的点云，并采用受见证复形启发的稀疏化方法，以保留上下文流形的持久同调特征。在单块 NVIDIA RTX 4090 上，我们通过实验证明，100 个并发代理的显存总占用为 2.2 GB；在计算延迟成为瓶颈之前，理论容量可超过 1,000 个智能体。我们还提出**指称注入**，这是一种非侵入式键值缓存更新机制，可让异步子智能体影响主生成过程，同时不中断数据流。

1 引言

大语言模型中的“系统 2”思维范式——模型在生成内容前暂停下来进行推理——已展现出提升准确性的潜力。然而，目前的实现方式是串行的：模型停下来思考，然后再继续。真正的生物认知是并行的；我们说话时，子进程会同时监测错误、回忆事实并提前规划。

在硅上复现这种并行性代价高昂。运行 10 个并发的 7B 模型需要 ≈ 140GB 显存，远超普通消费者的承受能力。即便使用更小的模型， KV 缓存也会随上下文长度 L 和智能体数量 N线性增长，导致 $O(N \cdot L)$ 的内存复杂度。

我们提出了 **Warp Cortex**，一种可将这种复杂度降低至权重 $O(1)$、内存 $O(N \cdot k)$ 的架构，其中 $k \ll L$。通过将智能体视为共享同一个“脑”（模型实例）和“内存”（Synapse）的异步线程，而非独立进程，我们实现了大规模扩展。

2 相关工作

用于高维稀疏化的拓扑数据分析。 从高维流形中选取代表性地标，是计算拓扑学中一个研究已久的问题。在此前关于医学影像的研究中 [1], ，我们证明，采用一种兼顾几何覆盖 与逆核密度 的混合度量，可将全脑磁共振成像体积中的平均成对距离降低 30–60%，同时通过见证复形保留持久同调特征。Warp Cortex 将这一原理扩展至 Transformer 的潜在空间：我们将键值（KV）缓存视为动态流形，并采用混合地标化，在不损失语义的情况下实现 98% 的上下文压缩。

多智能体大语言模型系统。 已有研究探索如何让语言模型呈现多种推理视角。Yang 和 Zhang [2]提出了用于群体智能的贝叶斯Transformer，

通过随机归一化层对多样化的模型实例进行采样。他们的方法通过贝叶斯推断实现行为多样性，但每个样本仍需单独的功能实例。Warp Cortex 解决的是一个互补问题：我们关注的不是多样性，而是密度——让 $100+$ 个并发代理能够在消费级硬件上共享同一个模型实例。我们的拓扑稀疏化或可助力其实用化部署贝叶斯群体。

混合专家架构。 Switch Transformers [3]和 Mixtral [4], 探索了稀疏条件计算，通过将词元路由到参数子集来进行计算。BitNet [5]证明，极限量化可以保持模型质量。这些研究优化的是计算稀疏性；Warp Cortex 则针对上下文稀疏性，通过受拓扑见证理论启发的基于注意力的地标选择，将 $O(N \cdot L)$ 内存压缩至 $O(N \cdot k)$。

高效推理。 现代推理系统依赖键值缓存 [6] 来提高自回归效率。Warp Cortex 引入了指称注入，这是一种新颖的键值缓存更新机制，允许异步子智能体影响生成过程，同时不打断主流——这是现有缓存策略尚未实现的能力。

3 架构

3.1 TheRiver 与流拓扑

标准推理流水线是同步的。Warp Cortex 实现了一种分裂拓扑：

- **河流（主智能体）**：专用于用户交互和人格维护的高优先级 CUDA 流。
- **溪流（侧智能体）**：多个中优先级 CUDA 流，分支执行特定推理任务（事实核查、逻辑验证）。

这些流在图形处理器上并发执行。当河流生成词元 t_i 时，一个流可以处理词元 t_{i-10} 的推理链。

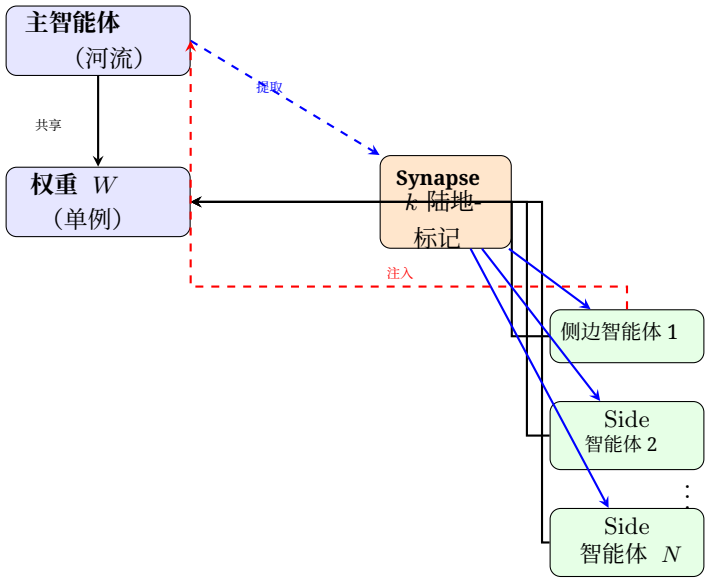

图 1：Warp Cortex 架构：所有智能体共享单个模型实例（Prism）。Synapse 提供$O(k)$ 上下文压缩。指称注入（红色）会更新主智能体的键值缓存。

3.2 棱镜：单例权重共享

为避免 $O(N)$ 权重惩罚，我们采用单例模型模式。模型权重 W 只加载一次到显存中。所有 N 智能体都持有指向 W 的指针。

$$M_{total} = \text{Mem}(W) + \sum_{i=1}^{N}(K_i + V_i) \approx \text{Mem}(W) + N \cdot \text{Mem}(\text{Synapse}) \tag{1}$$

其中 Mem(Synapse) $\ll$ Mem(H)，有效地将内存增长从 $O(N \cdot L)$ 降至 $O(N \cdot k)$，其中k 为地标词元的数量。由于 Mem(W) 是常量，瓶颈完全转移到了上下文内存。

3.3 拓扑突触

标准智能体需要完整的对话历史 H （长度为 L）才能运行。为 100 个智能体复制 H 的开销高得难以承受。我们引入**拓扑突触**，这是一种共享内存缓冲区，仅包含“地标”——能够保留上下文流形拓扑结构的词元。

理论基础： 我们的选择策略直接借鉴了拓扑数据分析（TDA）中的混合地标化技术。这些技术最初是为高维医学影像中的覆盖优化而开发的 [1]。我们将键值缓存视为潜在空间中的点云，从中识别出能够保留上下文流形持久同调 的地标，确保即使词元数量减少 98%，侧智能体仍能维持语义覆盖。

混合密度-覆盖采样器：

- **几何覆盖**：选择地标以最小化其与原始上下文流形之间的豪斯多夫距离，确保每个语义区域都有所体现。
- **注意力分数求和**：给定主智能体在时间步 t 的查询状态 Q_t，我们计算注意力分数 $A_i = \sum_{h=1}^{H} \text{softmax}(Q_t K_i^T / \sqrt{d_k})$ $\forall h \in \{1, \ldots, H\}$，其中 d_k 是键向量的维度。这充当我们的逆核密度估计器。
- **Top-k 选择**：我们选取得分最高的前 k 个词元（例如 $k = 64$），以代表上下文的语义核心。
- **见证复形整合**：侧智能体利用 Synapse 作为见证复形 [1], ，使其能够根据 k 个地标重建全局推理路径，其中 $k \ll L$。

这将每个智能体的内存成本从约 1GB（对应 32k 上下文）降至 ≈ 10MB，同时保留编码语义关系的拓扑特征。

3.4 皮层路由器：动态委派

Warp Cortex 不会预先定义智能体角色（例如“评论者”“编码者”），而是采用动态路由层。

- **意图提取**：基于正则表达式的路由器会监控主智能体的输出流，查找触发模式（例如，`[TASK: ...]`）。
- **即时生成**：检测到触发信号后，系统会生成一个通用工作线程，并为其指定具体的任务描述。
- **效率**：智能体仅在需要时才会存在，从而进一步节省资源。

3.5 验证关卡

为防止不佳推理污染主流并引发“幻觉级联”，我们实施几何质量控制检查。

令 $h_t^{(L)}$ 表示最终层 L中第 t 个词元的潜在表示。在合并侧边智能体的思维 T_{side} 之前，我们提取其末词元隐藏状态，并计算它与主智能体当前隐藏状态 $h_{main}^{(L)}$的余弦相似度：

$$\text{Score} = \frac{h_{main}^{(L)} \cdot T_{side}}{\|h_{main}^{(L)}\| \|T_{side}\|} \tag{2}$$

若分数为 $< \theta$，则拒绝该思维，其中 θ 是一个用于精确率与召回率权衡的超参数（根据经验，在我们的实验中设为 0.5）。这确保只有与上下文相关的推理会进入溪流，从而过滤掉低质量或偏离主题的内容。

3.6 指称注入

传统的注入方式是将文本粘贴到上下文中，这会打断主智能体的生成流程。我们提出了**指称注入**，这是一种在不改变可见文本流的情况下更新键值（KV）缓存的方法。

- **机制**：引擎会对标记为“引用”的思维向量 T_{side} 执行一次前向传播。
- **内存更新**：由此产生的键和值会追加到主智能体过去的_键_值中。
- **位置完整性**：为了保持结构完整性，我们使用旋转位置嵌入（RoPE），为注入的思维分配一个虚拟位置索引，将其标记为辅助上下文，而非序列词元。这可以避免违反因果掩码，同时保留模型的注意力机制。
- **结果**：主智能体会像刚刚读过这段思维一样“记住”它，同时无缝地继续生成原有的句子结构。

4 实现

我们使用 PyTorch 和 CUDA 流实现了 Warp Cortex。

Listing 1: Asynchronous Kernel Dispatch

```
# 异步流执行stream_main =t o r c h . cuda .
Stream ( ) stream_side =t o r c h . cuda .
Stream ( )

witht o r c h . cuda . stream ( stream_main ) :# 主智能体生
成并推送地标l o g i t s =model ( i n p u t _ i d s ) s y n a p s
e . push ( 提取_地标 ( kv_cache ) )

witht o r c h . cuda . stream ( stream_side ) :# S i d e a
g e n t r e a d s 地标 ( Zero − Copy ) # O ( k ) a t t e n t i o n c o s
t思维 =model ( s y n a p s e . r e ad ( ) )
```

5 评估

5.1 理论可扩展性

我们分析了 NVIDIA RTX 4090（24GB 显存）的理论容量。

表1：理论显存使用量比较（0.5B模型）

组件	标准架构	Warp Cortex
主模型权重	1.2 GB	1.2 GB
侧边智能体权重	1.2 GB	**0.0 GB**（共享）
侧智能体上下文	≈ 0.5 GB（完整）	**0.01 GB**（Synapse）
最大智能体数（24GB）	≈ 12	≈ 400

5.2 实证结果

我们使用 Qwen2.5-0.5B-Instruct，在一块 RTX级GPU 上以共享权重模式启动并发代理，对实际显存用量进行了基准测试。

表 2：测得的显存使用量与智能体数量

智能体数量	总显存	显存增量	每个智能体的显存
基线 (1)	0.93 GB	—	—
10	1.05 GB	0.12 GB	12 MB
50	1.44 GB	0.52 GB	10 MB
100	**2.22 GB**	**1.29 GB**	**13 MB**

关键发现：仅需额外增加 1.29 GB 显存，即可支持 100 个并发代理。这验证了我们的理论模型，并表明在 24 GB 显卡上，将规模扩展至 1,000+ 个智能体是可行的，直到计算延迟成为瓶颈。

性能特征：虽然显存会随智能体数量线性增长，每个智能体占用 ≈13 MB，但推理吞吐量会平稳下降。主智能体的生成速度几乎维持在基线水平，因为侧智能体在独立的 CUDA 流上异步运行，不会阻塞主要生成流程。

6 结论

Warp Cortex 证明，多代理系统的瓶颈在于架构，而非根本限制。通过从“基于进程”的思维模式转向“基于线程”的思维模式——共享权重并压缩内存——我们可以在消费级硬件上运行强大的“代理委员会”。这为本地、隐私保护的“系统 2”推理引擎开启了大门。

6.1 对边缘人工智能的影响

在消费级 GPU 上部署推理智能体的能力，根本性地改变了高级人工智能部署的经济模式。组织如今可以在无需依赖云服务的情况下运行复杂的多代理系统，从而实现：

- **数据隐私**：敏感的推理过程保留在本地部署环境中
- **成本降低**：大规模推理不再产生按词元计费的 API 成本
- **延迟优化**：智能体协调无需网络往返

6.2 未来工作

Warp Cortex 的若干扩展值得进一步研究：

1. **自适应地标选择**：根据任务复杂度动态调整 k

2. **H分层 Synapse**：用于更深层上下文压缩的多级地标缓冲区

3. **专用智能体架构**：整合 BitNet 和提前退出策略，进一步降低每个智能体的成本 4. **跨 GPU 扩展**：将架构扩展至多 GPU 系统，并采用分布式突触管理

6.3 更广泛的影响

这项工作代表了我们对大语言模型推理认知方式的一次范式转变。Warp Cortex 表明，与其将模型视为单一的黑箱，不如将其视为大规模并行认知过程的共享计算基底。这具有深远的意义，涉及：

- **自主系统**：在机器人学和决策系统中实现实时多视角推理
- **研究民主化**：让无法使用数据中心基础设施的研究人员也能使用先进的多智能体架构
- **安全对齐**：通过促进多样化的内部“辩论”机制，提升人工智能决策的稳健性

通过证明在消费级硬件上实现百万智能体认知扩展是可行的，我们希望催生新一代本地部署的并行推理系统，将集体智能的优势带入边缘计算环境。

引用

[1] Jorge L. Ruiz Williams。通过混合地标化实现磁共振成像体数据的快速见证持久化。*arXiv* 预印本 *arXiv:2510.04553*，2025 年。[2] Diji Yang 和 Yi Zhang。一个模型，众多思维：面向群体智能的贝叶斯 Transformer。*arXiv* 预印本 *arXiv:2512.25063*，2025 年。[3] William Fedus、Barret Zoph 和 Noam Shazeer。Switch Transformers：以简单高效的稀疏性扩展至万亿参数模型。《机器学习研究期刊》，23(1):5232–5270，2022 年。[4] Albert Q Jiang、Alexandre Sablayrolles、Antoine Roux、Arthur Mensch、Blanche Savary、Chris Bamford、Devendra Singh Chaplot、Diego de las Casas、Emma Bou Hanna、Florian Bressand 等。Mixtral 专家混合模型。*arXiv* 预印本 *arXiv:2401.04088*，2024 年。[5] Shuming Ma、Hongyu Wang、Lingxiao Ma、Lei Wang、Wenhui Wang、Shaohan Huang、Li Dong、Ruiping Wang、Jilong Xue 和 Furu Wei。1 位大语言模型时代：所有大语言模型均为 1.58 位。*arXiv* 预印本 *arXiv:2402.17764*，2024 年。[6] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。Attention Is All You Need。《神经信息处理系统进展》，第 30 卷，2017 年。