INVENTOR(S)

[0001] KIMZEY, Samuel C

TITLE

[0002] Composite Cognition Pools for Emergent Memory-Driven Systems

TECHNICAL FIELD

[0003] This invention relates to collective intelligence and distributed ledger systems, and more particularly to methods for merging, reconciling, and fusing outputs from multiple AI or agent pipelines.

BACKGROUND

[0004] Decentralized systems often employ individual AI agents or proof-of-memory pipelines for specialized tasks, but lack a unified framework to combine these disparate reflections into cohesive meta-insights. Conventional consensus mechanisms rely on tokens or central coordinators, which can limit emergent, organic cognition.

SUMMARY

[0005] Subscribing to reflection output streams from a plurality of autonomous agents or pipelines.

[0006] Organismic fusion of reflections: allowing inter-agent interactions and flow signals to produce emergent meta-insights naturally.

[0007] Algorithmic reconciliation: applying configurable merge rules (e.g., majority vote, weighted averaging) to agent outputs.

[0008] Logging composite outputs as higher-order events in the memory field via one-action-one-mint transactions.
[0009] Feeding composite insights back into downstream modules to influence system behavior adaptively.

DETAILED DESCRIPTION

[00010] In the organismic embodiment, each agent's reflection output is treated as an impulse in the global memory field. Agents propagate their outputs as on-chain events; through natural interactions—such as overlapping memory-flow pulses, pressure point alignments, and resonance effects—the system coalesces a composite cognition signal without explicit merging logic.
[00011] In the algorithmic embodiment, the system collects discrete reflection outputs from agents, applies reconciliation rules (e.g., computing a weighted average based on agent trust scores or flow magnitudes), and produces a single meta-insight event. Both approaches record composite outputs immutably on-chain.
[00012] In some embodiments, agents may include non-generative artificial cognition modules or mesh-anchored LLMs, and composite events may be minted as multi-modal artifacts for direct interface display. Agent weights can adapt based on downstream outcome verification, and composite events may require explicit consent flags. The system can federate reflections from multiple mesh instances, anchoring provenance across networks.

METHOD FLOW

[00013] Step 1: Stream Subscription – Agents emit reflection events to the shared memory field.
[00014] Step 2: Organismic Fusion – Allow reflection events to interact via memory-flow dynamics, generating emergent peaks representing composite insights

[00015] Step 3: Algorithmic Merge – Optionally, collect recent reflection events and apply merge rules to compute a reconciled output.

[00016] Step 4: Composite Event Logging – Mint a one-action-one-mint event capturing the composite insight.

[00017] Step 5: Insight Propagation – Provide the composite event to downstream modules (e.g., VM LLM, flow controllers) for adaptive response.

NARRATIVE WORKED EXAMPLE

[00018] Three agents monitor governance votes, network performance, and user sentiment. Their reflection events coincide around a policy decision. In the organismic mode, overlapping reflection pulses create a natural pressure peak interpreted as a composite governance alert. In algorithmic mode, the system averages the three agent scores to produce a unified recommendation event.

ALGORITHMIC WORKED EXAMPLE

1. Collect recent reflection outputs from all participants (agents and users).
2. For each participant, determine their trust score, contribution weight, or pressure value.
3. Merge the outputs using a defined rule (such as weighted average or majority).
4. Create a composite insight from the merged reflections.
5. Mint the composite insight as a new event in the mesh memory field.
6. Make this composite event available for other modules, users, or agents for further action or adaptive response.

ORGANISMIC/PRESSURE-BASED COMPOSITE PSEUDOCODE

1. Collect recent reflection events from all agents and users.

2. For each, calculate a dynamic “pressure” value (based on recent activity, flow resonance, or mesh alignment).
3. Use these values as weights to combine all reflections into a composite insight.
4. Mint the composite insight as a canonical mesh event for use by any participant or downstream module.

[00019] Pressure for each agent may be calculated as a function of recent memory-flow magnitude, reflection resonance, or alignment with mesh trait activity, thereby allowing agent influence to evolve organically in real time.

[00020] In this embodiment, agent weights are not fixed but are dynamically computed as “pressure” values, derived from memory-flow dynamics and mesh state, enabling truly emergent composite insights.

EMBODIMENTS

[00021] Hierarchical pools where sub-pools fuse regional agent outputs before global aggregation.

[00022] Privacy-preserving pool merges using zero-knowledge proofs of composite correctness.

[00023] Cross-chain cognition pools leveraging anchors across multiple memory meshes.

[00024] Dynamic agent weight adjustment based on reflection accuracy feedback loops.

[00025] Composite pools incorporating outputs from LLM or NGAC-based agents.

[00026] Minting composite events as visual, audio, or multi-modal objects for mesh-native UI rendering.

[00027] Adaptive agent weighting based on downstream feedback loops.

[00028] Consent-gated composite events requiring explicit agent confirmation.

[00029] Cross-mesh composite cognition with federated anchoring and reconciliation.

IMPLEMENTATION NOTES

[00030] Composite cognition events follow the one-action-one-mint paradigm. Organismic fusion requires no extra code modules, while algorithmic merges can be implemented in main application modules or standalone handlers.

CLAIMS

1. A method for generating composite cognition outputs in a memory-driven blockchain system, comprising:
    a. subscribing, by a processor, to reflection event streams from multiple autonomous agents;
    b. allowing, by the processor, organismic interactions of reflection events via memory-flow dynamics to produce emergent composite signals;
    c. optionally applying, by the processor, algorithmic merge rules to reconcile reflection event values into a single output;
    d. recording, by the processor, the composite cognition insight as a one-action-one-mint event on-chain;
    e. propagating, by the processor, the composite insight to downstream modules for adaptive system behavior.
2. The method of claim 1, wherein organismic interactions occur without explicit merge code, driven purely by flow dynamics.
3. The method of claim 1, wherein merge rules include weighted averaging or majority voting based on agent trust scores.
4. The method of claim 1, further comprising hierarchical fusion of sub-pools prior to global composite generation.
5. The method of claim 1, wherein privacy-preserving proofs attest to composite correctness without revealing individual reflections.
6. The method of claim 1, wherein at least one agent or pipeline comprises a mesh-native LLM or NGAC module.
7. The method of claim 1, wherein composite cognition events include rendered visual, auditory, or multi-modal artifacts, minted as canonical display objects within the mesh.

8. The method of claim 1, wherein agent or pipeline weights are dynamically updated in response to feedback or verification loops.
9. The method of claim 1, wherein composite cognition events require explicit consent signals from participating agents.
10. The method of claim 1, wherein reflections are federated from multiple mesh instances and cross-mesh anchors are recorded for auditability.

ABSTRACT

[00031] A method for federating multiple autonomous agents' reflection outputs into composite cognition pools within a decentralized memory-driven system, enabling emergent, organismic insight formation and algorithmic consensus without reliance on tokens. The method supports emergent and algorithmic cognition pooling across mesh and LLM agents, multi-modal output minting, adaptive agent weighting, consent gating, and cross-mesh federation.