

Quantum Hyper-Aether:

From Quantum-Inspired Semantic Dynamics to an Experimental Semantic Operating System

Hirofumi Inomata

September 2026

Abstract

Quantum Hyper-Aether (QHA) is an experimental computing architecture in which meaning, context, intent, purpose, knowledge, constraints, and their relationships are treated as first-class objects of computation and system control.

The term “Quantum” in QHA does not imply that the current implementation requires a physical quantum computer. Instead, QHA uses a quantum-inspired mathematical framework to describe multiple candidate semantic states, semantic transitions, interactions, state selection, entropy, and semantic temperature. These theoretical concepts are mapped onto executable mechanisms implemented on conventional computing infrastructure.

An important conclusion of the QHA investigation is that Semantic Data cannot be represented by a vector alone. A semantic vector is effective for similarity calculation, clustering, routing, and novelty detection, but it does not explicitly and deterministically represent purpose, intent, constraints, provenance, lifecycle state, temporal information, or graph relationships. QHA therefore defines Semantic Data as a hybrid representation consisting of a semantic vector, structured semantic state, and explicit relationships.

The experimental development of QHA progressed from semantic routing and semantic-distance evaluation to Semantic Event-based Asynchronous Graph Neural Networks (SE-AEGNN), Discovery VM generation, autonomous VM lifecycle management, adaptive deletion, knowledge re-acquisition, semantic Business Continuity and Disaster Recovery (BC/DR), and geo-distributed semantic recovery.

In a news-oriented routing experiment, semantic routing achieved 85.60% routing accuracy compared with 64.35% for keyword routing. SE-AEGNN preserved 100% decision agreement with global updates while reducing semantic graph computation by up to 14.32 times at 1,000 VMs.

In adaptive BC/DR experiments, QHA Predictive BC/DR achieved a recovery utility of 0.706 and a semantic recovery time of 30.57 seconds, compared with 0.379 and 58.55 seconds for a fixed recovery policy. Geo-distributed QHA achieved semantic coverage of 0.9184, service continuity of 0.9928, and mean recovery time of 22.984 in the evaluated simulation.

Recent work introduced LLM_SEM as a semantic front end. LLM_SEM generates continuous semantic representations, while the Semantic Shell constructs operational Semantic Data containing explicit meaning, context, intent, purpose, constraints, metadata, and relationships. A learned semantic projection improved leave-one-out semantic routing accuracy from 83.33% to 87.50% in an early controlled experiment.

Taken together, these results suggest an architecture consisting of a quantum-inspired semantic model, a hybrid Semantic Data representation, a Semantic Operating System control plane, and a classical execution infrastructure. QHA therefore provides an experimental framework for investigating computing systems in which semantic state complements conventional machine state.

1 Introduction

Traditional operating systems manage machine-oriented entities such as

$$\{CPU, Memory, Process, Thread, File, Network, Container, VM\}. \quad (1)$$

These abstractions remain essential.

However, AI-native systems increasingly process information for which meaning, context, intent, purpose, and knowledge influence the appropriate execution strategy.

This motivates the central question of QHA:

Can semantic information itself become a unit of computation, scheduling, resource management, learning, and recovery?

QHA introduces the *Semantic Task* as such an abstraction.

A Semantic Task is represented conceptually as

$$T_s = (M, C, I, P, K, R, X), \quad (2)$$

where

- M represents meaning,
- C represents context,
- I represents intent,
- P represents purpose,
- K represents associated knowledge,
- R represents execution requirements,
- X represents explicit constraints.

The QHA research developed through two related paths.

The first investigated a quantum-inspired mathematical description of semantic states and interactions.

The second converted semantic concepts into executable mechanisms such as semantic routing, dynamic VMs, asynchronous semantic graph updates, autonomous learning, and semantic BC/DR.

Recent LLM_SEM experiments connect natural-language interpretation directly to this architecture.

2 Experimental History of QHA

Figure 1 summarizes the principal experimental evolution of QHA.

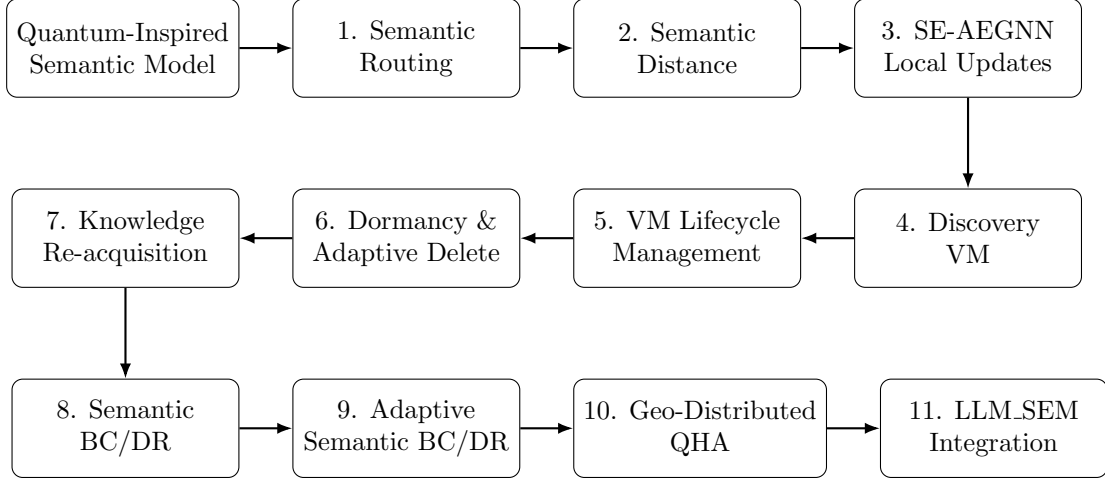


Figure 1: QHA experimental history. The research began with a quantum-inspired semantic model and progressively developed executable mechanisms for semantic routing, asynchronous local updates, VM lifecycle management, autonomous knowledge reconstruction, semantic BC/DR, geo-distributed recovery, and LLM.SEM integration.

3 Quantum-Inspired Foundation

The term “Quantum” in Quantum Hyper-Aether requires careful definition.

The current QHA implementation does not execute quantum gates on a physical quantum computer.

Instead,

$$\boxed{QHA = \textit{Quantum-Inspired Semantic Model} + \textit{Classical Execution}} \quad (3)$$

The quantum-inspired model provides mathematical abstractions for

- multiple candidate semantic states,
- semantic transitions,
- semantic interactions,
- state selection,
- uncertainty,
- entropy,
- and semantic temperature.

3.1 Semantic State

A semantic state with multiple candidate interpretations is represented as

$$|\Psi\rangle = \sum_{i=1}^N c_i |S_i\rangle, \quad (4)$$

where $|S_i\rangle$ represents a candidate semantic state and c_i represents its corresponding weight. The normalized representation satisfies

$$\sum_{i=1}^N |c_i|^2 = 1. \quad (5)$$

This does not represent physical quantum superposition.
It is a mathematical representation of competing semantic possibilities.

3.2 Semantic Superposition

Before execution, a task may be compatible with multiple VMs.

The routing state can therefore be represented as

$$|\Psi_{\text{route}}\rangle = \alpha_1|VM_1\rangle + \alpha_2|VM_2\rangle + \cdots + \alpha_n|VM_n\rangle. \quad (6)$$

A normalized semantic compatibility distribution may be obtained from

$$|\alpha_i|^2 = \frac{\exp(\beta S_i)}{\sum_j \exp(\beta S_j)}, \quad (7)$$

where S_i is semantic compatibility and β controls selection sharpness.

3.3 Semantic State Selection

Execution eventually requires a concrete action.

QHA represents this as a collapse-like transition

$$|\Psi\rangle \longrightarrow |S_k\rangle. \quad (8)$$

In the classical implementation,

$$k = \arg \max_i P(S_i|\mathcal{C}), \quad (9)$$

where $\mathcal{C}$ contains context, purpose, constraints, semantic memory, and resource state.
Thus,

$$\boxed{\textit{Quantum-Inspired Collapse} \approx \textit{Semantic Scheduler Selection}} \quad (10)$$

is the practical implementation mapping.

3.4 Semantic Transition Amplitude

A semantic transition

$$|S_i\rangle \rightarrow |S_j\rangle \quad (11)$$

may be represented by

$$A_{ij} = \langle S_j|\hat{U}|S_i\rangle. \quad (12)$$

The corresponding transition probability is

$$P_{ij} = |A_{ij}|^2. \quad (13)$$

In a classical QHA implementation, this can correspond to a learned transition probability, semantic graph edge weight, or routing score.

4 Action-Based Semantic Dynamics

An action formulation was investigated as a general representation of semantic evolution.

Let

$$S = \int \mathcal{L}(\Psi, \partial_\mu \Psi, C, P, K) d\tau. \quad (14)$$

The semantic trajectory is conceptually selected according to

$$\delta S = 0. \quad (15)$$

A generic QHA semantic Lagrangian can be decomposed as

$$\mathcal{L}_{QHA} = \mathcal{L}_{state} + \mathcal{L}_{interaction} + \mathcal{L}_{purpose} + \mathcal{L}_{knowledge} + \mathcal{L}_{resource}. \quad (16)$$

This representation provides a theoretical mechanism for combining semantic state, purpose, knowledge, interaction, and computational cost.

5 Field-Theoretic Interpretation

QHA also investigated a field-theoretic interpretation.

A semantic field is represented as

$$\Phi(x, t), \quad (17)$$

where x represents a position in semantic space rather than necessarily a position in physical space.

A semantic event modifies a local region:

$$\Phi(x, t) \rightarrow \Phi(x, t + \Delta t). \quad (18)$$

This interpretation became relevant to the later SE-AEGNN experiments:

$$\boxed{\textit{Semantic Field Locality} \rightarrow \textit{Semantic Event Locality} \rightarrow \textit{SE-AEGNN}} \quad (19)$$

5.1 Dirac-Inspired Semantic Dynamics

A Dirac-inspired semantic equation was considered in the form

$$(i\gamma^\mu D_\mu - m_s) \Psi_s = 0. \quad (20)$$

Here m_s is not physical mass.

It represents an abstract semantic inertia.

A large m_s corresponds to a semantic state that is resistant to change, whereas a small m_s corresponds to a state that can be changed more easily by incoming semantic events.

This interpretation can be related to semantic persistence and VM lifecycle behavior.

5.2 QED-Inspired Semantic Interaction

Semantic interactions were also investigated using a QED-inspired interaction term

$$\mathcal{L}_{int} = g_s \bar{\Psi} \gamma^\mu A_\mu^{(s)} \Psi. \quad (21)$$

Here

- Ψ represents a semantic entity,

- $A_\mu^{(s)}$ represents an abstract semantic interaction field,
- g_s represents semantic coupling strength.

This equation is used as a mathematical analogy.

It does not imply that semantic information is a physical quantum field.

6 String- and M-Theory-Inspired Extension

A further theoretical investigation considered whether semantic entities should be regarded as fixed types or as entities whose functions can change according to internal modes.

A string-inspired representation is

$$X^\mu(\sigma, \tau). \quad (22)$$

Different modes may represent different semantic roles:

$$\{Routing, Knowledge, Reasoning, Discovery, Recovery\}. \quad (23)$$

The systems interpretation is more important than the physical analogy.

A semantic execution entity does not need to have one permanently fixed role.

Instead,

$$Role(V, t) \quad (24)$$

may change over time.

This concept later became directly relevant to

$$Discovery \rightarrow Promote \rightarrow Merge \rightarrow Dormant \rightarrow Delete. \quad (25)$$

The M-theory-inspired exploration generalized this concept to a higher-dimensional semantic state space in which several semantic properties can coexist as independent dimensions.

These dimensions may include

$$Meaning, Context, Purpose, Knowledge, Resource, Time. \quad (26)$$

7 Semantic Entropy and Temperature

For semantic-state probabilities p_i , QHA defines semantic entropy as

$$H_s = - \sum_i p_i \log p_i. \quad (27)$$

Low semantic entropy indicates that one or a small number of interpretations dominate.

High semantic entropy indicates ambiguity or competition among several possible semantic interpretations.

Semantic temperature was introduced to represent semantic dynamism, ambiguity, or reasoning demand.

Conceptually,

$$T_s \propto \frac{\partial E_s}{\partial H_s}, \quad (28)$$

where E_s represents semantic processing cost or semantic energy.

7.1 Semantic Temperature Regions

For

$$T_s < T_1, \quad (29)$$

semantic uncertainty is low and deterministic automation may be sufficient:

$$Input \rightarrow Rule \rightarrow Action. \quad (30)$$

For

$$T_1 \leq T_s < T_2, \quad (31)$$

semantic interpretation becomes useful:

$$Input \rightarrow LLM_{SEM} \rightarrow Semantic Scheduler. \quad (32)$$

For

$$T_s \geq T_2, \quad (33)$$

the system may activate

$$Discovery VM + Knowledge Acquisition + LLM + Semantic Reorganization. \quad (34)$$

Thus,

$$\boxed{Semantic Temperature \rightarrow Execution Strategy} \quad (35)$$

connects the theoretical QHA model to the experimental architecture.

8 Mapping Quantum-Inspired Concepts to QHA

Table 1: Mapping Quantum-Inspired Concepts to QHA Mechanisms

Quantum-Inspired Concept	Semantic Interpretation	QHA Mechanism
State vector	Semantic state	Embedding / LLM_SEM
Superposition	Multiple candidate meanings	Candidate VM set
Measurement / collapse	Executable-state selection	Semantic Scheduler
Transition amplitude	Semantic transition likelihood	Routing score
Interaction field	Semantic relationships	Semantic graph
Local field update	Local semantic change	SE-AEGNN
Mode transition	Change of semantic role	VM lifecycle
Entropy	Semantic ambiguity	Unknown detection
Semantic temperature	Reasoning requirement	Automation / LLM / Discovery
State reconstruction	Capability restoration	Semantic BC/DR
Distributed field	Distributed semantic state	Geo-Distributed QHA

9 Semantic Routing Experiment

Four routing methods were evaluated:

- Random Routing,
- Load-Based Routing,

- Keyword Routing,
- Semantic Routing.

Let

$$V = \{V_1, V_2, \dots, V_n\}. \quad (36)$$

For incoming Semantic Task T_s , the scheduler selects

$$V^* = \arg \max_{V_i \in V} S(T_s, V_i). \quad (37)$$

9.1 Initial Routing Experiment

Table 2: Initial Semantic Routing Experiment

Method	Routing Acc.	Task Success	Latency	Similarity
Semantic Routing	100.00%	89.13%	1.892	0.922
Keyword Routing	100.00%	85.11%	1.880	–
Random Routing	14.25%	52.79%	2.579	–
Load-Based	14.12%	54.42%	2.437	–

9.2 News Semantic Task Dataset

Table 3: News Semantic Task Routing Results

Method	Routing Acc.	Task Success	Latency	Similarity
Semantic Routing	85.60%	84.01%	1.857	0.866
Keyword Routing	64.35%	66.46%	–	–
Random Routing	10.76%	45.68%	–	–
Load-Based	10.35%	48.81%	–	–

10 Semantic Distance

One implementation uses cosine similarity:

$$S_{\text{sem}} = \frac{\mathbf{s}_{\text{task}} \cdot \mathbf{s}_{VM}}{\|\mathbf{s}_{\text{task}}\| \|\mathbf{s}_{VM}\|}. \quad (38)$$

Semantic distance is

$$D_{\text{sem}} = 1 - S_{\text{sem}}. \quad (39)$$

A broader QHA model allows

$$D_{\text{sem}} = f(D_{\text{embedding}}, D_{\text{concept}}, D_{\text{ontology}}, D_{\text{context}}). \quad (40)$$

11 Why Semantic Data Cannot Be Represented by a Vector Alone

Semantic vectors are essential to QHA.

However,

$$\boxed{\text{Semantic Vector} \neq \text{Complete Semantic Data.}} \quad (41)$$

A vector

$$\mathbf{v} = (v_1, v_2, \dots, v_d) \in \mathbb{R}^d \quad (42)$$

provides a continuous representation of an object in semantic space.
This is highly useful for

- similarity calculation,
- nearest-neighbor search,
- clustering,
- routing,
- novelty detection,
- and learned semantic projection.

However, QHA requires operational semantic information that cannot reliably be represented only as a position in an embedding space.

11.1 Semantic Similarity Is Not Operational Equivalence

Consider

“Delete the inactive VM.”

and

“Do not delete the inactive VM.”

The sentences share the same topic and much of their lexical and semantic content.
Nevertheless, their operational intents are opposite.
A Semantic OS must preserve this distinction exactly.
Therefore,

$$\textit{Semantic Similarity} \neq \textit{Operational Equivalence}. \quad (43)$$

11.2 Purpose Must Be Explicit

The same semantic content may correspond to several purposes.

Information about a failed VM may be used to

$$\textit{Diagnose}, \quad \textit{Recover}, \quad \textit{Analyze}, \quad \textit{or} \quad \textit{Ignore}. \quad (44)$$

The topic may remain nearly identical while the required execution strategy changes completely.
QHA therefore treats Purpose as an explicit field rather than requiring the scheduler to reconstruct it from an embedding.

11.3 Context Must Be Preserved

Meaning depends on context.

For example,

“migration”

may refer to

- VM migration,
- data migration,
- model migration,
- geographic service migration,
- or semantic capability migration.

An embedding may contain contextual information implicitly, but exact operational context cannot always be reconstructed deterministically from the vector alone.

QHA therefore represents Context explicitly.

11.4 Constraints Require Exact Representation

Execution constraints may include

$$Latency < 100 \text{ ms}, \quad (45)$$

$$Region = Tokyo, \quad (46)$$

$$GPU = Required, \quad (47)$$

or

$$ConfidentialData = LocalOnly. \quad (48)$$

These conditions should not be inferred approximately from semantic distance.

They must be evaluated exactly.

Consequently, QHA separates semantic similarity from execution constraints.

11.5 Relationships Require Graph Structure

Semantic information contains explicit relationships such as

$$VM_A \rightarrow Knowledge_B, \quad (49)$$

$$Task_C \rightarrow Purpose_D, \quad (50)$$

and

$$Knowledge_B \rightarrow Region_E. \quad (51)$$

Vectors may encode relational information implicitly, but QHA requires explicit relationships for

- dependency analysis,

- SE-AEGNN propagation,
- VM Merge,
- knowledge migration,
- failure impact analysis,
- and BC/DR reconstruction.

QHA therefore maintains a semantic graph

$$G_s = (V_s, E_s). \quad (52)$$

11.6 Temporal State and Lifecycle

Semantic objects evolve over time.

A VM may transition through

$$Discovery \rightarrow Active \rightarrow Dormant \rightarrow Deleted. \quad (53)$$

Knowledge may transition through

$$Known \rightarrow Unused \rightarrow Forgotten \rightarrow Reacquired. \quad (54)$$

A vector alone does not explicitly represent this lifecycle.

QHA therefore maintains

$$State(t), \quad Timestamp, \quad LastUsed, \quad Lifetime. \quad (55)$$

11.7 Provenance and Confidence

Two semantic objects may occupy almost the same position in embedding space while having very different reliability.

Operational Semantic Data may therefore require

$$Source, \quad Confidence, \quad Timestamp, \quad ValidationState. \quad (56)$$

This is particularly important when knowledge influences autonomous VM creation, deletion, or disaster recovery.

An embedding alone does not provide an auditable record of provenance.

11.8 Unknown Is an Explicit System State

Suppose

$$\max_i Sim(\mathbf{v}_{task}, \mathbf{v}_{VM_i}) < \theta. \quad (57)$$

The vector similarity provides evidence that the task is outside known semantic domains.

However, it does not itself define the system response.

QHA converts this result into

$$SemanticState = Unknown. \quad (58)$$

The state then triggers

$$Unknown \rightarrow Discovery \rightarrow VM \rightarrow Knowledge \text{ Acquisition}. \quad (59)$$

Thus, a vector contributes evidence to a system decision, while the structured state controls behavior.

11.9 BC/DR Requires Structured Semantic State

Two semantic objects may have similar vectors while having different recovery priorities.
For example,

$$Priority_A = Critical \quad (60)$$

and

$$Priority_B = Low. \quad (61)$$

Semantic BC/DR therefore requires explicit attributes such as

$$Criticality, \quad Dependency, \quad RecoveryPriority, \quad Region, \quad ReplicationState. \quad (62)$$

Recovery can then evaluate

$$RecoveryPriority_i = f(Criticality_i, Dependency_i, Demand_i, SemanticState_i, RecoveryCost_i). \quad (63)$$

11.10 QHA Semantic Data Structure

These observations lead to a hybrid representation.

A QHA Semantic Data object is defined conceptually as

$$D_s = (ID, \mathbf{v}, M, C, I, P, K, R, X, G, Q, T, L) \quad (64)$$

where

- ID is a persistent semantic-object identifier,
- $\mathbf{v}$ is the semantic vector,
- M is explicit meaning or semantic description,
- C is context,
- I is intent,
- P is purpose,
- K is a knowledge reference or payload,
- R is the resource requirement,
- X contains execution constraints,
- G represents graph relationships,
- Q contains confidence, provenance, and quality metadata,
- T contains temporal information,
- L is lifecycle state.

Therefore,

$$\mathbf{v} \subset D_s \quad (65)$$

rather than

$$D_s = \mathbf{v}. \quad (66)$$

11.11 Hybrid Vector–Structured Representation

The vector layer is appropriate for

$$\textit{Similarity, Routing, Clustering, Novelty, Projection.} \quad (67)$$

The structured layer is appropriate for

$$\textit{Intent, Purpose, Constraints, Identity, Lifecycle, Provenance, Policy.} \quad (68)$$

The graph layer is appropriate for

$$\textit{Dependencies, Interactions, Propagation, Recovery.} \quad (69)$$

Thus, the QHA representation is

$$\boxed{\textit{Semantic Data = Vector + Structured Semantic State + Relationships.}} \quad (70)$$

11.12 Connection to the Quantum-Inspired Model

The hybrid representation is consistent with the quantum-inspired QHA model.

A semantic vector can be interpreted as a numerical representation of a semantic state:

$$\mathbf{v} \leftrightarrow |\Psi\rangle. \quad (71)$$

However, a state representation alone does not define the complete system. QHA additionally requires

$$\textit{Context, Purpose, Operators, Relationships, Constraints, State.} \quad (72)$$

Therefore,

$$\boxed{\textit{Semantic Vector} \approx \textit{Semantic State Representation}} \quad (73)$$

but

$$\boxed{\textit{Semantic Vector} \neq \textit{Complete Semantic System.}} \quad (74)$$

12 SE-AEGNN Experiment

QHA represents the semantic system as

$$G = (V, E). \quad (75)$$

Instead of updating the entire graph after every event,

$$\textit{Update}(G), \quad (76)$$

SE-AEGNN updates only

$$\textit{Update}(\mathcal{N}(e_t)), \quad (77)$$

where $\mathcal{N}(e_t)$ is the neighborhood affected by semantic event e_t .

Table 4: SE-AEGNN Local Update Experiment

VMs	Local Fraction	Computation Reduction	Speedup	Decision Agreement
100	89.5%	$3.35\times$	$1.03\times$	100%
300	52.5%	$5.71\times$	$1.05\times$	100%
1000	20.9%	$14.32\times$	$1.19\times$	100%

The maximum semantic decision error was zero.

13 Discovery VM Experiment

When no existing VM adequately represents an incoming Semantic Task,

$$\max_i S(T_s, V_i) < \theta, \quad (78)$$

QHA creates a Discovery VM.

Thus,

$$Unknown\ Semantic\ Region \rightarrow Discovery\ VM. \quad (79)$$

The Discovery VM temporarily represents the new semantic region and collects evidence about whether a persistent execution domain should be created.

14 VM Lifecycle Experiment

The Discovery VM was extended into an autonomous lifecycle:

$$Discovery \rightarrow \{Promote, Merge, Reject, Delete\}. \quad (80)$$

Table 5: Selected Discovery VM Lifecycle Results

Metric	Result
Lifecycle Accuracy	98.51%
Promote Precision	95.09%
Merge Precision	100.00%
Reject Precision	100.00%
Task Success	76.47%
Wrong-Promote Rate	4.91%

Adding a simple Delete mechanism reduced full lifecycle accuracy to 64.16%, while Delete precision was only 57.63%.

This negative result motivated the adaptive deletion experiment.

15 Adaptive Delete and Dormancy

A Dormant state was introduced:

$$Active \rightarrow Dormant \rightarrow \{Reactivate, Merge, Delete\}. \quad (81)$$

Deletion was changed to

$$P_{delete} = f(T_{inactive}, N_{semantic}, P_{reuse}, U). \quad (82)$$

The simple idle-time policy achieved

$$DeletePrecision = 93.33\%, \quad (83)$$

with

$$FalseDeletion = 6.67\%. \quad (84)$$

The semantic-aware configuration achieved

$$DeletePrecision = 100.00\% \quad (85)$$

and

$$FalseDeletion = 0.00\%. \quad (86)$$

16 Autonomous Knowledge Re-acquisition

Deleting a VM can remove knowledge from the active system.

QHA treats a subsequent unanswered Semantic Task as a learning event.

If

$$\max_i S(T_t, V_i) < \theta, \quad (87)$$

the processing path becomes

$$Failure \rightarrow Discovery \rightarrow Knowledge Acquisition \rightarrow VM Reconstruction. \quad (88)$$

Thus,

Missing knowledge can itself become a semantic learning signal.

17 Integrated QHA Experiment

Four architectures were compared:

- Global Update,
- Fixed VM,
- Lifecycle VM,
- QHA + SE-AEGNN.

Table 6: Response Success in the Integrated Experiment

Scenario	Global	Fixed	Lifecycle	QHA+SE-AEGNN
Stationary	86.7%	86.9%	89.2%	94.7%
Preference Drift	84.9%	85.7%	88.2%	94.6%
Knowledge Deletion Shock	86.4%	84.0%	85.7%	93.3%

Approximate update costs were

$$12.0, \quad 1.0, \quad 1.0, \quad 1.30 \quad (89)$$

units/event respectively.

For a knowledge deletion shock, recovery times were approximately

$$29, \quad 64, \quad 78, \quad 34 \quad (90)$$

steps.

18 Semantic BC/DR

Traditional BC/DR primarily attempts to recover infrastructure state.

QHA introduces another objective:

$$\boxed{\textit{Recover Semantic Capability}} \quad (91)$$

rather than requiring every machine to be reconstructed identically.

Semantic recovery priority can be represented as

$$\textit{Priority}(V_i) = f(\textit{Importance}_i, \textit{Demand}_i, \textit{Dependency}_i, \textit{RecoveryCost}_i). \quad (92)$$

19 Adaptive Semantic BC/DR Experiment

Four recovery methods were compared:

- Fixed Policy,
- Rule-Based Adaptive,
- SE-AEGNN Adaptive,
- QHA Predictive BC/DR.

A total of 40,000 simulation trials were performed.

Table 7: Adaptive Semantic BC/DR Results

Method	Utility	Semantic RTO	Semantic RPO	Answer Success	Continuity
Fixed	0.379	58.55	0.398	0.396	0.439
Rule-Based	0.489	49.28	0.321	0.503	0.554
SE-AEGNN	0.634	33.29	0.197	0.649	0.720
QHA Predictive	0.706	30.57	0.157	0.725	0.792

20 Geo-Distributed QHA BC/DR

QHA was extended to three geographic regions:

$$R = \{R_1, R_2, R_3\}. \quad (93)$$

If a region fails,

$$R_k \rightarrow \emptyset, \quad (94)$$

semantic responsibilities are redistributed.

A target region can be selected using

$$R_{\text{target}} = \arg \max_{R_i} [\textit{SemanticFit}(R_i, S_j) - \textit{MigrationCost}(R_i, S_j)]. \quad (95)$$

Table 8: Overall Geo-Distributed QHA Result

Metric	Geo-QHA
Recovery Time	22.984
Semantic Coverage	0.9184
Knowledge Loss	0.0771
Service Continuity	0.9928
Transfer Cost	56.159

21 LLM_SEM and the Semantic Shell

LLM_SEM provides the continuous semantic representation used by QHA.

For input x ,

$$\mathbf{s} = LLM_{SEM}(x). \quad (96)$$

A learned projection may produce

$$\mathbf{z} = P_{\theta}(\mathbf{s}). \quad (97)$$

However,

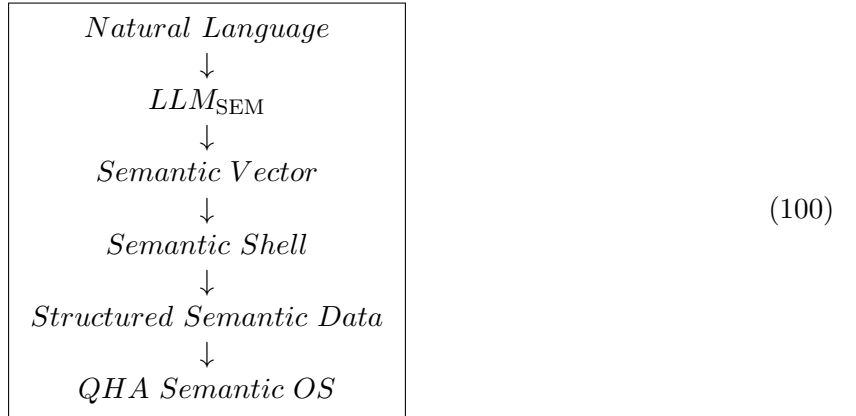
$$\boxed{\mathbf{z} \neq D_s}. \quad (98)$$

The projected semantic vector is only one component of the complete Semantic Data object.

The Semantic Shell combines the original input, the semantic vector, structured interpretation, and system metadata:

$$D_s = SemanticShell(x, \mathbf{z}, \mathcal{C}_{sys}). \quad (99)$$

The complete path is therefore



21.1 LLM_SEM Experimental Results

In an early controlled experiment, leave-one-out routing accuracy changed from

$$83.33\% \rightarrow 87.50\%. \quad (101)$$

Semantic margin changed from

$$0.144574 \rightarrow 0.204620. \quad (102)$$

Known holdout recall changed from

$$63.33\% \rightarrow 66.67\%, \quad (103)$$

while unknown detection changed from

$$80.00\% \rightarrow 83.33\%. \quad (104)$$

Balanced known/unknown accuracy changed from

$$71.67\% \rightarrow 75.00\%. \quad (105)$$

A later 600-sample experiment produced a smaller known-routing improvement:

$$55.00\% \rightarrow 57.50\%. \quad (106)$$

These results indicate that semantic projection can improve separation, while generalization remains an open research problem.

22 Eleven Experiments and Their Findings

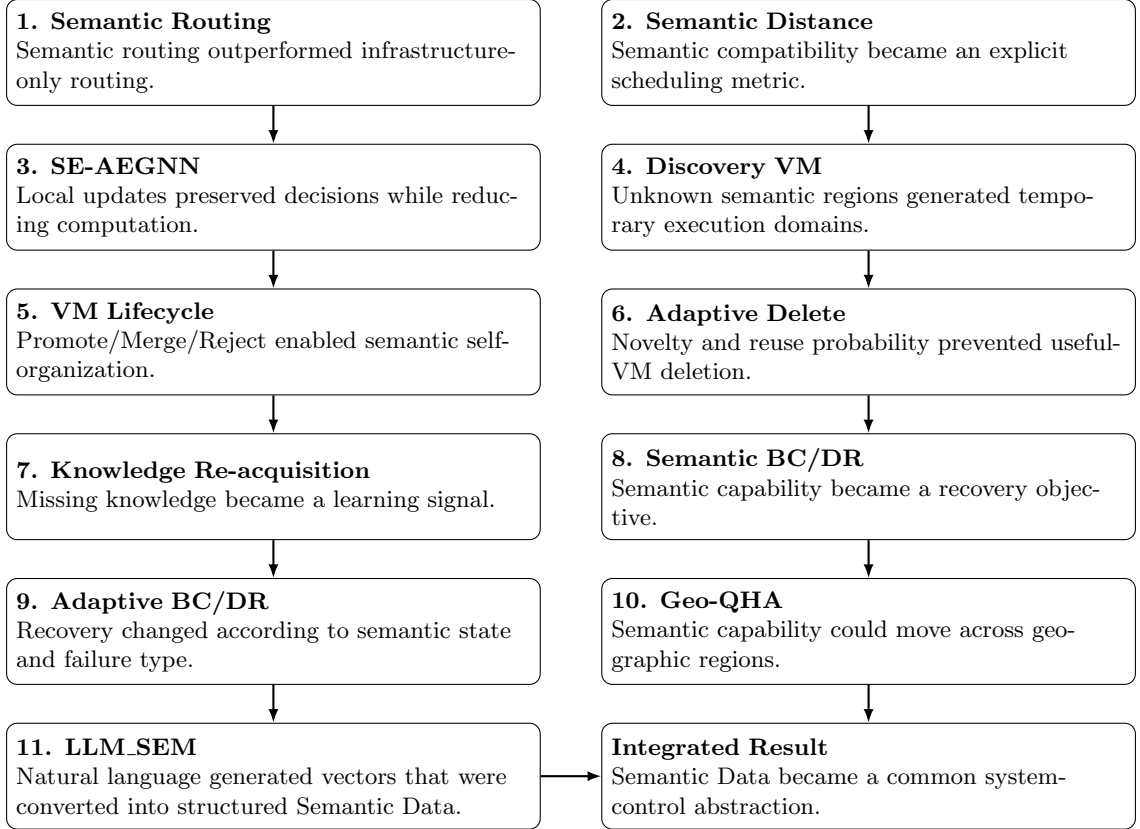


Figure 2: Eleven principal QHA experiments and their findings.

23 Current QHA + LLM_SEM Architecture

The architecture is no longer modeled as a direct

$$Language \rightarrow Vector \rightarrow VM \quad (107)$$

pipeline.

Instead, QHA explicitly separates semantic representation from operational Semantic Data construction.

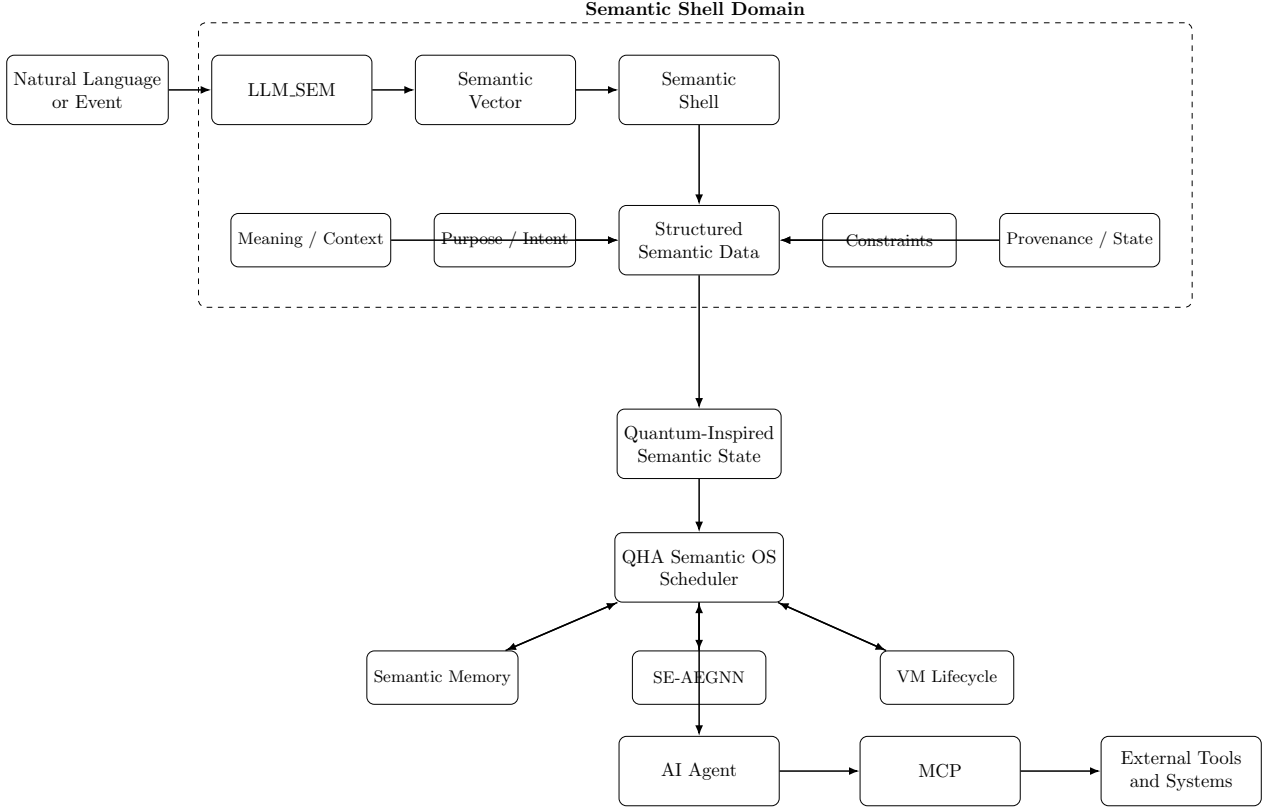


Figure 3: Current QHA + LLM.SEM architecture. LLM.SEM generates a continuous Semantic Vector. The Semantic Shell combines the vector with explicit meaning, context, intent, purpose, constraints, provenance, temporal state, and relationships to construct operational Semantic Data. The Quantum-inspired Semantic State model and QHA Semantic OS then use this hybrid representation for scheduling, memory, event propagation, lifecycle control, and execution.

24 Unified Layered Architecture

The resulting architecture consists of five layers:

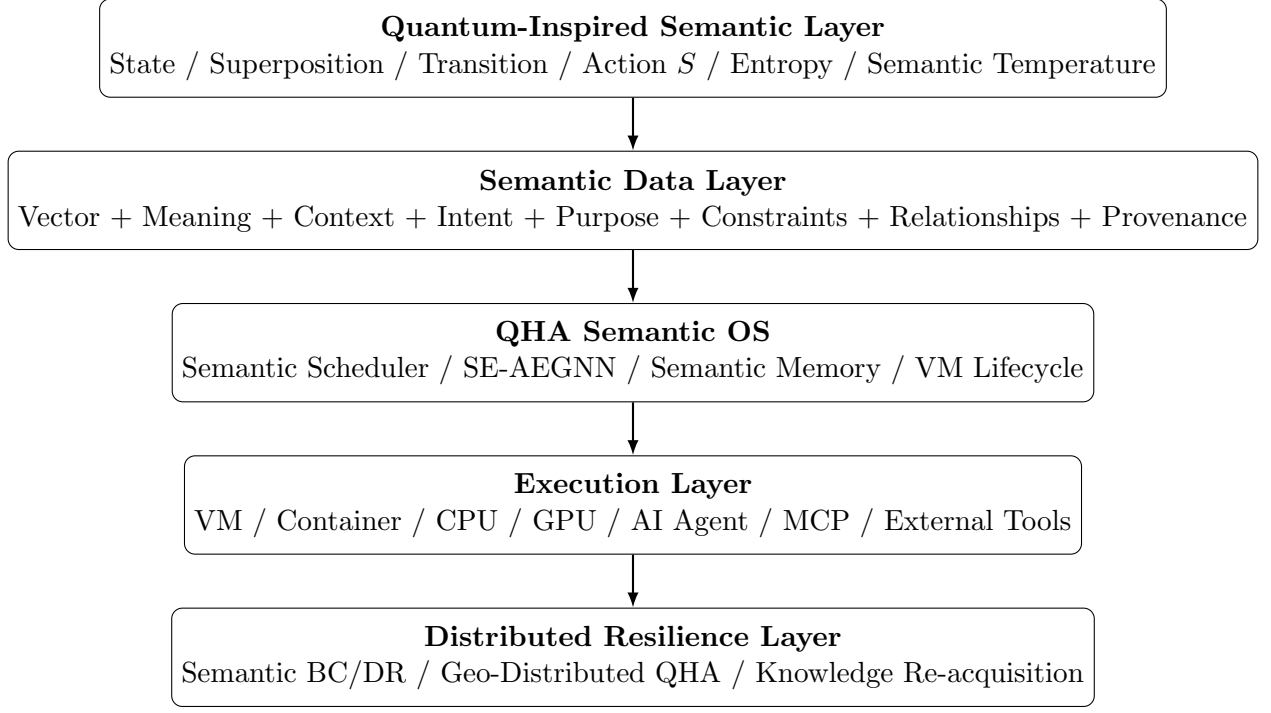


Figure 4: Unified Quantum Hyper-Aether architecture. Semantic Data is explicitly defined as a hybrid representation rather than an embedding vector alone.

25 Consolidated Experimental Results

Table 9: Consolidated QHA Experimental Results

Experiment	Metric	Measured Result
Semantic Routing	Routing accuracy	News dataset: Semantic 85.60%; Keyword 64.35%; Random 10.76%; Load 10.35%.
Semantic Routing	Task success	Semantic 84.01%; Keyword 66.46%; Random 45.68%; Load 48.81%.
SE-AEGNN	Computation reduction	$3.35\times$ at 100 VMs; $5.71\times$ at 300 VMs; $14.32\times$ at 1,000 VMs.
SE-AEGNN	Decision agreement	100% agreement with global-update decisions; maximum observed semantic decision error = 0.
Discovery VM Lifecycle	Lifecycle accuracy	Selected Merge/Reject configuration: 98.51% lifecycle accuracy, 95.09% Promote precision, 100% Merge precision, 100% Reject precision.
Initial Delete	Delete precision	57.63%; result motivated Adaptive Delete.
Adaptive Delete	Delete precision	Semantic-aware configuration: 100.00% Delete precision and 0.00% useful-VM false deletion.
Integrated QHA	Response success	QHA+SE-AEGNN: 94.7% stationary, 94.6% preference drift, 93.3% knowledge deletion shock.
Integrated QHA	Update cost	Global Update approximately 12.0 units/event; QHA+SE-AEGNN approximately 1.30 units/event.
Adaptive BC/DR	Recovery utility	Fixed 0.379; Rule-Based 0.489; SE-AEGNN 0.634; QHA Predictive 0.706.
Adaptive BC/DR	Semantic RTO	58.55, 49.28, 33.29, 30.57 seconds respectively.
Geo-QHA	Semantic coverage	0.9184.
Geo-QHA	Service continuity	0.9928.
Geo-QHA	Knowledge loss	0.0771.
Geo-QHA	Recovery time	22.984 compared with 34.437 for semantic routing alone.
LLM_SEM	LOO routing accuracy	83.33% $\rightarrow$ 87.50%.
LLM_SEM	Semantic margin	0.144574 $\rightarrow$ 0.204620.
LLM_SEM	Unknown detection	80.00% $\rightarrow$ 83.33%.
LLM_SEM	Balanced accuracy	71.67% $\rightarrow$ 75.00%.
LLM_SEM larger experiment	Known routing	55.00% $\rightarrow$ 57.50%.

26 Discussion

The QHA research produced three complementary conclusions.

First, semantic computation can be described using a quantum-inspired state model containing candidate states, transitions, interactions, entropy, and temperature.

Second, Semantic Data cannot be reduced to an embedding vector.

A vector provides an effective continuous representation for similarity and routing, but an operational semantic system additionally requires explicit purpose, context, intent, constraints, identity, relationships, provenance, and lifecycle state.

Third, these semantic structures can be mapped onto conventional executable mechanisms.

For example,

$$\textit{Superposition} \rightarrow \textit{Candidate Semantic States}, \quad (108)$$

$$\textit{Collapse} \rightarrow \textit{Semantic Scheduling}, \quad (109)$$

$$\textit{Field Locality} \rightarrow \textit{SE-AEGNN}, \quad (110)$$

$$\textit{Mode Change} \rightarrow \textit{VM Lifecycle}, \quad (111)$$

and

$$\textit{State Reconstruction} \rightarrow \textit{Semantic BC/DR}. \quad (112)$$

The resulting architecture is therefore not purely neural and not purely symbolic.

It is a hybrid architecture:

$$\boxed{\textit{Continuous Semantic Representation} + \textit{Structured Semantic State} + \textit{Graph Relationships}.} \quad (113)$$

27 From QHA to a Semantic Operating System

A conventional operating system maintains

$$X_{\text{machine}} = \{CPU, Memory, Process, I/O, Storage\}. \quad (114)$$

QHA additionally maintains

$$X_{\text{semantic}} = \{Vector, Meaning, Context, Intent, Purpose, Knowledge, Constraints, Relationships, Lifecycle\}. \quad (115)$$

The complete state becomes

$$X = X_{\text{machine}} \cup X_{\text{semantic}}. \quad (116)$$

The central architectural proposition is therefore

$$\boxed{\textit{Computing State} = \textit{Machine State} + \textit{Semantic State}.} \quad (117)$$

Under this interpretation,

- LLM_SEM generates continuous semantic representations,
- the Semantic Shell constructs operational Semantic Data,

- the Quantum-inspired layer represents candidate states, uncertainty, transitions, and semantic temperature,
- the Semantic Scheduler performs execution selection,
- SE-AEGNN maintains event-driven semantic relationships,
- Semantic Memory maintains knowledge,
- the VM Manager maintains semantic execution domains,
- AI Agents perform specialized execution,
- MCP connects external tools,
- and Semantic BC/DR maintains semantic capability under failure.

28 Limitations

Several limitations must be explicitly recognized.

First, the “quantum” component of the current QHA implementation is quantum-inspired rather than physically quantum.

No experimental result in this paper demonstrates quantum speedup, physical superposition, quantum entanglement, or execution on quantum hardware.

Second, the Dirac-, QED-, string-, and M-theory-inspired formulations are theoretical analogies and modeling tools.

Third, many QHA experiments are simulations or small-scale prototypes.

Fourth, several VM experiments use simplified semantic VMs rather than complete production hypervisor environments.

Fifth, semantic routing quality depends strongly on the quality of the semantic representation.

Sixth, the proposed structured Semantic Data format has not yet been evaluated in a complete end-to-end QHA implementation.

Seventh, LLM.SEM experiments show that improvements in development-space separation do not always generalize proportionally to independent holdout data.

Finally, the complete integrated QHA architecture has not yet been evaluated as a single running system.

29 Next Experimental Stage

The next stage is

$$\boxed{QHA + LLM_{SEM} \text{ Minimum Integrated Prototype v0.1.}} \quad (118)$$

The prototype should contain

$$LLM_{SEM} + \text{Semantic Shell} + \text{Semantic Data} + \text{Quantum-Inspired State} + \text{Semantic Scheduler} + \text{VM Manager} + \dots \quad (119)$$

Initially, VMs may be dummy execution domains whose principal function is to consume controlled execution time.

The prototype should measure

- Semantic Task generation accuracy,
- Purpose and Intent extraction accuracy,

- structured Semantic Data construction accuracy,
- semantic routing accuracy,
- semantic entropy,
- semantic temperature,
- candidate-state distribution,
- unknown-task detection,
- VM creation frequency,
- VM reuse probability,
- Promote/Merge/Reject/Delete accuracy,
- SE-AEGNN update cost,
- scheduler overhead,
- semantic recovery time,
- semantic continuity,
- and total adaptation time.

An especially important experiment is to compare

$$\textit{Vector-Only} \tag{120}$$

against

$$\textit{Vector} + \textit{Structured Semantic Data}. \tag{121}$$

This experiment should test whether explicit Purpose, Intent, Context, Constraints, Provenance, and Relationships improve

- routing correctness,
- constraint compliance,
- unknown detection,
- lifecycle decisions,
- and BC/DR recovery decisions.

Such an experiment would directly validate the hypothesis developed in this paper that a semantic vector alone is insufficient for an operational Semantic OS.

30 Conclusion

Quantum Hyper-Aether began as an investigation into whether semantic information could be represented using concepts inspired by quantum state theory, field theory, entropy, and dynamic modes.

The research subsequently evolved into an experimental computing architecture. The theoretical progression can be summarized as

$$\boxed{\begin{aligned} & \textit{Semantic State} \rightarrow \textit{Superposition} \rightarrow \textit{Transition} \rightarrow \textit{Interaction} \\ & \rightarrow \textit{Entropy} \rightarrow \textit{Semantic Temperature} \end{aligned}} \quad (122)$$

while the experimental progression is

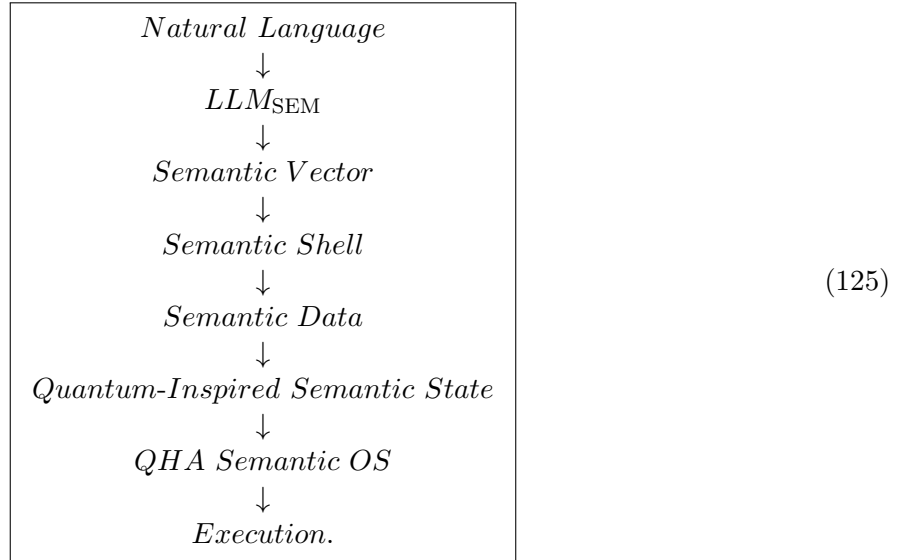
$$\boxed{\begin{aligned} & \textit{Semantic Routing} \rightarrow \textit{Semantic Distance} \rightarrow \textit{SE-AEGNN} \\ & \rightarrow \textit{Discovery VM} \rightarrow \textit{VM Lifecycle} \rightarrow \textit{Adaptive Delete} \\ & \rightarrow \textit{Knowledge Reacquisition} \rightarrow \textit{Semantic BC/DR} \\ & \rightarrow \textit{Geo-Distributed QHA} \rightarrow \textit{LLM}_{\text{SEM}}. \end{aligned}} \quad (123)$$

An additional result of this investigation is that the semantic vector should not be identified with Semantic Data itself.

Instead,

$$\boxed{\textit{Semantic Data} = \textit{Semantic Vector} + \textit{Structured Semantic State} + \textit{Relationships}.} \quad (124)$$

Accordingly, the current QHA processing chain is



The relationship between the theoretical and implementation layers is

$$\boxed{\textit{Quantum-Inspired Model} \rightarrow \textit{Semantic Representation} \rightarrow \textit{Semantic Control} \rightarrow \textit{Classical Execution}.} \quad (126)$$

The significance of QHA is therefore not the replacement of classical computing by quantum computing.

Instead, QHA introduces an additional semantic state layer between human or AI intent and physical computation.

The central architectural proposition becomes

$$\boxed{\textit{Computing State} = \textit{Machine State} + \textit{Semantic State}.} \quad (127)$$

QHA consequently provides an experimental path toward a Semantic Operating System in which meaning, context, intent, purpose, knowledge, relationships, uncertainty, constraints, and lifecycle state become explicit system-level objects.

References

- [1] H. Inomata, “Quantum Hyper-Aether: Semantic Computing Architecture,” QHA Research Notes and Experimental Reports, 2026.
- [2] H. Inomata, “Semantic Routing Experiments for Quantum Hyper-Aether,” QHA Experimental Report, 2026.
- [3] H. Inomata, “Semantic Event-based Asynchronous Graph Neural Network for Quantum Hyper-Aether,” QHA Experimental Report, 2026.
- [4] H. Inomata, “Autonomous Semantic VM Lifecycle Management,” QHA Experimental Report, 2026.
- [5] H. Inomata, “Adaptive Semantic Business Continuity and Disaster Recovery for Quantum Hyper-Aether,” QHA Experimental Report, 2026.
- [6] H. Inomata, “Geo-Distributed Autonomous Semantic BC/DR,” QHA Experimental Report, 2026.
- [7] H. Inomata, “LLM_SEM: Semantic Representation and Projection Experiments for Quantum Hyper-Aether,” Experimental Research Notes, 2026.
- [8] P. A. M. Dirac, “The Quantum Theory of the Electron,” *Proceedings of the Royal Society of London A*, vol. 117, pp. 610–624, 1928.
- [9] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.
- [10] M. B. Green, J. H. Schwarz, and E. Witten, *Superstring Theory*, Cambridge University Press, 1987.