Whitepaper

The Supra–Omega Resonance Theory (SORT): A Modular Operator-Projection Framework for Structural Analysis

Gregor Herbert Wegener
Independent Theoretical Physicist
Berlin, Germany
gregor.wegener@gmail.com

Keywords: resonance operator framework; idempotent operators; projection operators; projection kernel (k); light-balance condition; operator transition laws; structural adjacency; modular scientific architecture; validation and reproducibility; MOCK v4 framework; cosmological structure; Hubble drift

Contents

 https://doi.org/10.5281/zenodo.17787754

Abstract

This manuscript presents version 6 of the Supra–Omega Resonance Theory (SORT), a mathematically closed operator-projection framework that provides a unified structural foundation for cross-domain system analysis. SORT is constructed around a closed set of 22 idempotent resonance operators, a global projector $\hat{H}$, and a calibrated projection kernel $\kappa(k)$, forming an invariant mathematical core capable of addressing fundamental anomalies across cosmology, artificial intelligence, quantum systems, and complex systems within a single coherent architecture.

Version 6 establishes the complete algebraic structure and modular domain architecture that will enable systematic empirical validation in subsequent versions. The framework introduces a clear separation between a publicly documented core API and an internal validation engine, ensuring reproducibility and controlled extension. Structural consistency is verified via calibrated reference parameters, including kernel parameter $\sigma_0 = 0.00190643$, Jacobi residual below 5×10^{-15}, and idempotency residual below 10^{-12}.

The framework is validated within the MOCK v4 environment using a three-layer verification architecture. Version 6 focuses on architectural completeness, modular extensibility, and preparation for high-performance computing deployment. The explicit formulation of operator transition laws and adjacency validation provides the mathematical machinery for addressing phenomena ranging from the Hubble tension to early galaxy formation within a unified projection-based description. Empirical validation against observational data is scheduled for version 7.

1. Introduction

1.1. Motivation

Recent observational results have revealed a growing set of cosmological anomalies that challenge purely dynamical extensions of the standard cosmological model. These include persistent discrepancies in the inferred Hubble constant between early- and late-Universe probes, commonly referred to as the Hubble tension [12,14], the unexpectedly early appearance of massive galaxies observed by JWST [15,18], evidence for overmassive supermassive black hole seeds at high redshift [20,22], and statistically significant low-ℓ anomalies in the cosmic microwave background [26,27].

While numerous proposals introduce new dynamical degrees of freedom, modified gravity sectors, or exotic particle content, these approaches often address individual tensions in isolation. The accumulation of structurally diverse anomalies motivates the exploration of mechanisms that operate at the level of information distribution, projection, and structural consistency rather than through additional dynamics alone. This work adopts an operator-geometric perspective, in which observed tensions are interpreted as manifestations of projection effects within a constrained structural framework, without postulating new particles or modifying the fundamental dynamical equations.

1.2. Long-term Vision

The accumulation of structurally diverse anomalies across cosmology, quantum systems, and complex systems suggests that a unified structural perspective may provide explanatory power beyond what individual dynamical extensions can achieve. SORT is constructed with the explicit goal of establishing such a unified framework—one that addresses the Hubble tension, early galaxy formation, supermassive black hole seeds, and CMB anomalies within a single coherent mathematical architecture.

1.3. Framework Positioning

The Supra–Omega Resonance Theory (SORT) provides a projection-based structural layer that operates in conjunction with established theories such as ΛCDM, general relativity, and quantum field theory. Rather than modifying dynamical equations, SORT introduces an invariant operator architecture that captures how information and structure are distributed across scales—a perspective that addresses persistent anomalies from a unified geometric viewpoint.

This positioning is strategic: by establishing compatibility with existing frameworks while introducing independent structural constraints, SORT creates the foundation for a comprehensive description that may ultimately unify currently disparate phenomena. The operator-geometric perspective enables systematic analysis of tensions that resist purely dynamical resolution, including scale-dependent discrepancies in cosmological parameters and structural inconsistencies across domains.

1.4. Modular Architecture Overview

SORT is organized as a modular architecture composed of a common mathematical core and multiple domain-specific application layers. The mathematical core consists of a closed resonance-operator algebra, a global projector, and a calibrated projection kernel, all of which are domain-agnostic. Individual domains, including cosmology, artificial intelligence systems, quantum systems, and complex systems, are implemented as independent modules that interpret the same core structures within their respective contexts.

A strict application programming interface (API) separation is enforced between the publicly documented core and internal validation engines. This separation ensures publication safety, reproducibility of the exposed framework, and controlled extension toward empirical validation without compromising the integrity of the public specification.

1.5. Scope and Limitations of v6

Version 6 of SORT establishes the complete mathematical and architectural foundation required for systematic empirical engagement. It defines the full operator algebra, projection structure, validation pathways, and modular extensibility that will support quantitative comparison with observations in version 7.

The present version is designed as the stable core upon which all subsequent empirical work will build. By completing the algebraic closure and validation infrastructure before introducing observational data, SORT ensures that empirical results in later versions will be reproducible, unambiguous, and directly traceable to the public mathematical specification.

1.6. Document Structure

The manuscript is structured to guide different reader groups through the framework at appropriate levels of abstraction. Section 2 introduces the mathematical foundations and operator algebra. Section 3 describes the validation and reproducibility architecture. Sections 6 through 9 present the domain-specific application layers. The appendices provide formal derivations, operator tables, and validation manifests.

Readers primarily interested in mathematical consistency may focus on the core sections and appendices, while practitioners and reviewers concerned with applicability and reproducibility may emphasize the architectural and validation sections.

1.7. Relation to Previous Versions

SORT has evolved through several prior iterations. Version 4 established the conceptual framework and introduced the initial operator tables. Version 5 focused on mathematical hardening, consistency proofs, and algebraic closure. Version 6, presented here, consolidates these developments into a modular architecture with explicit separation between public and internal components and introduces preparation for high-performance computational validation.

A subsequent version 7 is planned to incorporate empirical validation, data assimilation, and large-scale numerical results using the architectural foundations established in the present work.

2. Mathematical Foundations

This section defines the minimal stable core of the SORT framework. The operator structure introduced here constitutes a load-bearing architectural element rather than a complete algebraic theory. No claims are made regarding axiomatic completeness, ontological finality, or physical exhaustiveness. The purpose of this section is to document the epistemic origin, reduction logic, and structural closure of the resonance-operator set that underpins all subsequent constructions. Formal algebraic properties referenced in this section are analyzed in detail in Appendix B.

2.1. Origin, Reduction, and Closure of the Resonance Operator Set

This subsection clarifies the epistemic origin of the 22 resonance operators and the criteria under which the operator set was consolidated into its present form. The methodological perspective follows standard operator-theoretic and projection-based reasoning [37,40,43].

2.1.1. Origin of the Operators

The resonance operators emerged through an iterative analysis of structural invariants observed across multiple theoretical contexts. The initial exploration was guided by recurring principles that appear in projection theory, non-local coupling schemes, and reflexive constructions [38,39]. Central among these was projection, characterized by idempotent mappings satisfying

$$\hat{P}^2 = \hat{P}, \tag{1}$$

together with duality relations expressed through complementary operator pairs, non-separable coupling structures analogous to entanglement, reflexive mappings associated with fixed-point behavior, and stability under operator composition.

These principles were identified as structural motifs rather than postulated axioms and served as orientation for subsequent consolidation.

2.1.2. Methodology of Reduction

The structure-finding process proceeded through successive consolidation stages. An initial fragmentation phase collected a broad set of candidate motifs. This was followed by redundancy elimination, in which elements expressible as linear combinations or compositions of others were removed. Independence was then assessed by verifying that no remaining operator $\hat{O}_i$ could be generated from the complement set under admissible compositions,

$$\hat{O}_i \neq \mathcal{F}\big(\{\hat{O}_j\}_{j \neq i}\big), \tag{2}$$

for any allowed functional composition $\mathcal{F}$. Finally, closure was tested by examining whether the reduced set generated additional independent motifs under repeated composition or adjoint action.

2.1.3. Closure Criterion

The reduction process terminated when three conditions were simultaneously satisfied. First, no new independent operators emerged under further composition. Second, all additional candidates reduced to linear combinations of existing elements. Third, algebraic consistency relations, including commutator stability,

$$[\hat{O}_i, \hat{O}_j] \in \mathrm{span}\{\hat{O}_k\}, \tag{3}$$

closed without residual inconsistencies. The resulting number of operators is therefore a consequence of structural stabilization rather than a predefined target. The value 22 arises from this closure process.

2.1.4. Designation as Resonance Operators

The term resonance operator reflects the functional role of these objects as repeatable structural response modes under non-local coupling and projection. The designation emphasizes structural recurrence under composition and kernel-mediated interaction,

$$\hat{O}_i \xrightarrow{\kappa} \hat{O}_i, \tag{4}$$

without implying physical resonance, oscillatory dynamics, or frequency-domain behavior. The terminology is motivated purely by algebraic and structural considerations.

2.1.5. Scope Delimitation

The present construction makes no claim of physical completeness, ontological interpretation, empirical validation in version 6, or axiomatic necessity. These limitations are intrinsic to the architectural scope of the framework and are revisited in Section 1.5.

2.1.6. Early Structural Motifs and Conceptual Convergence

Several early-identified motifs guided the consolidation process. Projection appeared as a low-dimensional representation of higher-dimensional dynamics, duality as complementary paired descriptions, entanglement as non-local coupling, and self-reference as reflexive fixed-point behavior [43,48]. These motifs converged toward a unified operator description rather than being independently postulated.

Role of the Holographic Principle

A central orienting concept was the holographic principle, according to which information capacity scales with boundary area rather than bulk volume [43–45,48]. Structurally, this motivates a passive projection

$$\pi : \mathcal{H}_{\text{bulk}} \to \mathcal{H}_{\text{boundary}}, \tag{5}$$

which served as a conceptual precursor to the SORT projection formalism.

Transition to Resonance Projection

Within SORT, this picture is generalized to a kernel-mediated projection π_κ that allows coupled structural response while preserving global consistency,

$$\pi_\kappa = \hat{H}\,\kappa(k)\,\hat{H}, \tag{6}$$

as formalized in Section 2.6.

Convergence Rather Than Construction

The operator set was not constructed axiomatically but converged through iterative reduction and stabilization. Structural motifs provided orientation, consolidation defined the process, and the closed operator set emerged as the stabilized result.

2.1.7. Fragment–Operator Correspondence

Recurring fragment types can be mapped to operator properties, such as projection operators satisfying Eq. 1, coherence operators associated with non-local coupling, duality operators obeying involutive relations, and closure operators characterized by vanishing commutators. A complete mapping is provided in Appendix A.

Light-Balance Origin

The operator set admits a balanced partition into constructive and reductive subsets of equal cardinality. Assigning normalized weights w_i yields the balance condition

$$\sum_i w_i = 0, \tag{7}$$

ensuring structural neutrality of the public core.

2.1.8. Derivational Roadmap and Operator Consolidation

The consolidation followed a characteristic roadmap: reduction of redundant motifs, merging into common carriers, algebraic stabilization through consistency checks such as Eq. 3, and consolidation into a minimally closed operator core.

Epistemic Anchoring

This roadmap provides epistemic anchoring rather than historical justification. The resulting 22 operators form the closed public core of SORT. Introducing additional operators would reintroduce redundancy without extending structural capacity. The invariance of this structure across domain interpretations is analyzed further in Appendix C.

2.2. Resonance Operator Space

We define the resonance projection space $\mathcal{H}_R$ as a projective Hilbert space equipped with a finite, structurally closed operator set. All resonance operators act endomorphically on this space,

$$\hat{O}_i : \mathcal{H}_R \to \mathcal{H}_R, \qquad i \in \{1, \ldots, 22\}, \tag{8}$$

and are interpreted strictly in technical terms. No philosophical, ontological, or physical labels are assigned beyond their algebraic and functional roles within the framework.

2.3. Operator Index with Type and Spectrum

The complete resonance operator set consists of 22 elements with well-defined algebraic type, action domain, and spectral characteristics. These properties are summarized in Table 1.

2.4. Technical Interpretation and Weights

Each resonance operator is assigned a technical designation and a normalized structural weight c_i. These weights enforce global balance and do not represent physical charges or probabilities. The assignments are listed in Table 2.

The light-balance condition follows directly,

$$11\left(\frac{1}{11}\right) + 11\left(-\frac{1}{11}\right) = 0, \tag{9}$$

ensuring global structural neutrality.

2.5. Operator Algebra — Invariant Properties

The resonance operators satisfy a minimal invariant algebra [37–39].

Table 1. Resonance operator index with algebraic type and spectral characteristics.

ID	Symbol	Type	Domain	Spectrum
1	$\hat{O}_1$	Self-adjoint	$\mathcal{H}_R$	Continuous
2	$\hat{O}_2$	Self-adjoint	$\mathcal{H}_R$	Continuous
3	$\hat{O}_3$	Unitary	$\mathcal{H}_R$	Discrete
4	$\hat{O}_4$	Self-adjoint	$\mathcal{H}_R$	Bounded
5	$\hat{O}_5$	Hermitian	$\mathcal{H}_R$	Continuous
6	$\hat{O}_6$	Self-adjoint	$\mathcal{H}_R$	Continuous
7	$\hat{O}_7$	Unitary	$\mathcal{H}_R$	Continuous
8	$\hat{O}_8$	Mixed	$\mathcal{H}_R$	Continuous
9	$\hat{O}_9$	Idempotent	$\Sigma \subset \mathcal{H}_R$	Discrete
10	$\hat{O}_{10}$	Self-adjoint	$\mathcal{H}_R$	Recursive
11	$\hat{O}_{11}$	Unitary	$\mathcal{H}_R$	Periodic
12	$\hat{O}_{12}$	Mixed	$\mathcal{H}_R$	Quasi-discrete
13	$\hat{O}_{13}$	Self-adjoint	$\mathcal{H}_R$	Integer
14	$\hat{O}_{14}$	Self-adjoint	$\mathcal{H}_R$	Continuous
15	$\hat{O}_{15}$	Hermitian	$\mathcal{H}_R$	Continuous
16	$\hat{O}_{16}$	Self-adjoint	$\mathcal{H}_R$	Logarithmic
17	$\hat{O}_{17}$	Self-adjoint	$\mathcal{H}_R$	Fractal
18	$\hat{O}_{18}$	Mixed	$\mathcal{H}_R$	Continuous
19	$\hat{O}_{19}$	Self-adjoint	$\mathcal{H}_R$	Probabilistic
20	$\hat{O}_{20}$	Unitary	$\mathcal{H}_R$	Functional integral
21	$\hat{O}_{21}$	Unitary	$\mathcal{H}_R$	Composite
22	$\hat{O}_{22}$	Idempotent	$\mathcal{H}_R$	Cyclic

Table 2. Technical interpretation and structural weights of resonance operators.

ID	Designation	Technical Function	c_i
1	Vacuum substrate	Ground state $\|0\rangle$	$+1/11$
2	Boundary projection	Kernel π_κ	$+1/11$
3	Non-local coherence	$\|\Phi^+\rangle$	$+1/11$
4	Complementarity	Structure factor δ	$+1/11$
5	Cyclic parameter	τ	$+1/11$
6	Metric structure	Emergent metric γ	$+1/11$
7	Invariant propagation	$\lambda = c\,dt$	$+1/11$
8	Temporal integration	μ	$+1/11$
9	Boundary operator	$\beta^2 = \beta$	$+1/11$
10	Recursive map	Fixed point σ	$+1/11$
11	Phase manifold	$e^{i\theta}$	$+1/11$
12	Coarse-graining	η	$-1/11$
13	Symmetry index	Z	$-1/11$
14	Superposition	R	$-1/11$
15	Complementary directions	ρ	$-1/11$
16	Statistical weight	$k \ln W$	$-1/11$
17	Scale recurrence	σ	$-1/11$
18	Symmetry breaking	Σ	$-1/11$
19	Information measure	$-\sum p_i \log p_i$	$-1/11$
20	Path integral	$\int \mathcal{D}\psi\, e^{iS}$	$-1/11$
21	Composition	$\prod_i \hat{O}_i$	$-1/11$
22	Cycle closure	Structural closure	$-1/11$

The projection space is formulated in standard Hilbert space terms [37,40]. 335

$$\mathcal{H}_R := \text{projective Hilbert space}, \tag{10}$$

$$\hat{O}_i^2 = \hat{O}_i, \qquad \forall i, \tag{11}$$

$$[\hat{O}_i, \hat{O}_j] = \sum_k f_{ij}{}^k \hat{O}_k, \tag{12}$$

$$[\hat{O}_i, [\hat{O}_j, \hat{O}_k]] + [\hat{O}_j, [\hat{O}_k, \hat{O}_i]] + [\hat{O}_k, [\hat{O}_i, \hat{O}_j]] = 0. \tag{13}$$

The complete 22×22 commutator matrix is provided in Appendix B.1.

2.5.1. Operator Transition Law and Structural Adjacency

A transition is defined as a formal structural mapping,

$$\hat{T} : \hat{O}_i \to \hat{O}_j. \tag{14}$$

A transition is valid if the operators commute, the target operator closes the structural degree of freedom of the source, and alternative sequences reduce to the same normal form. Transitions describe structural neighborhoods rather than temporal evolution.

2.5.2. Structural Adjacency of Resonance Operators

Structural adjacency defines allowed successor relations between operators. For example, projection operators admit coherence operators as successors, while composition operators admit only cyclic closure. Validation relies on commutator closure, idempotency preservation, and path independence. The complete adjacency matrix is given in Appendix B.1.

2.6. Global Projector $\hat{H}$

The global projector is defined spectrally as

$$\hat{H} = \sum_{i=1}^{22} w_i \hat{O}_i, \qquad \sum_i w_i = 0, \tag{15}$$

and satisfies global idempotency [38–40],

$$\hat{H}^2 = \hat{H}. \tag{16}$$

2.7. Projection Kernel $\kappa(k)$

In Fourier space the kernel is defined as, following standard kernel constructions in spectral theory [37,49,51],

$$\kappa(k) = \exp\left[-\frac{(\sigma_0 L_H k)^2}{2}\right], \tag{17}$$

while in real space

$$\kappa[\Sigma; \phi_+, \phi_-] = \mathcal{N}_\kappa \exp\left[-\frac{(\phi_+ - \phi_-)^2}{2\sigma^2}\right]. \tag{18}$$

The calibration reference is $\sigma_0 = 0.00190643$ with Hubble length $L_H = 4285\,\text{Mpc}$. Normalization is fixed by

$$\kappa(0) = 1. \tag{19}$$

2.8. Derived Structural Quantities

The amplification function is defined as

$$\eta(k) = \kappa(k) - 1, \tag{20}$$

and the structural Hubble drift as

$$\frac{\delta H}{H_0} = \frac{H_{\mathrm{local}} - H_{\mathrm{CMB}}}{H_{\mathrm{CMB}}}. \tag{21}$$

2.9. Mathematical Invariants

All algebraic and numerical consistency markers, including idempotency residuals, Jacobi residuals, and kernel normalization tolerances, are summarized in Appendix A.

3. Validation Architecture

The validation architecture defined in this section specifies the structural consistency checks of SORT v6. Validation is understood as verification of internal coherence, algebraic closure, and deterministic behavior of the public core, rather than empirical confirmation. The architecture and its design principles follow the SORT core framework as documented in Refs. [1–3]. All validation layers are publicly specified and reproducible, while execution internals remain separated as described in Section 4.1.

3.1. Three-Layer Validation Model

SORT employs a three-layer validation model, each layer addressing a distinct consistency domain of the framework [1].

Layer I performs algebraic diagnostics on the complete set of 22 resonance operators. This layer verifies idempotency, commutator closure, Jacobi consistency, and light-balance neutrality of the operator algebra, as formalized in Section 2.5.

Layer II validates kernel construction and calibration. It enforces normalization, parameter stability, and compatibility between the projection kernel $\kappa(k)$ and the global projector $\hat{H}$, following the kernel formalism introduced in Section 2.7 and specified in Ref. [2].

Layer III evaluates semi-spectral evolution. This layer tests deterministic propagation of operator sequences under kernel-modulated projection and verifies convergence to stable normal forms. No empirical data or observational inputs are introduced at any stage [3].

3.2. Operator Sequence Validation

Operator sequences are validated through local neighborhood consistency checks, as defined in the SORT core framework [1]. For any admissible ordered sequence $\{\hat{O}_i\}_{i=1}^{22}$, adjacent operators must satisfy the approximate commutativity condition

$$[\hat{O}_i, \hat{O}_{i+1}] \approx 0, \qquad i \in \{1, \ldots, 21\}, \tag{22}$$

with deviations evaluated against fixed numerical tolerances specified in Section 3.4. This criterion ensures local structural compatibility without enforcing global commutativity of the operator set.

3.2.1. Transition Consistency Check

Version 8 introduces explicit transition consistency diagnostics, extending the validation logic of SORT v6 [1]. For a valid transition map $\hat{T} : \hat{O}_i \rightarrow \hat{O}_j$, three conditions are tested: convergence to a normal form under iteration, invariance of the normal form under repeated application, and path independence across alternative admissible transition sequences. Quantitative thresholds are summarized in Table 3.

Table 3. Transition consistency diagnostics and tolerances.

Test	Description	Tolerance
Sequence stability	Convergence to normal form	$\leq 10^{-10}$
Normal form preservation	Repeated application invariant	$\leq 10^{-12}$
Path independence	Alternative paths yield identical result	exact

3.3. MOCK v4 Environment

All validation layers are implemented within the MOCK v4 environment, which serves as the reference execution platform for SORT v6 [3]. Deterministic execution is enforced via a fixed global seed,

$$\text{seed} = 117666, \tag{23}$$

and all numerical artefacts are recorded with SHA–256 checksums. The environment maintains a strict separation between the public core, containing all algebraic definitions and validation rules, and the internal engine, which implements execution logic and optimization layers, as detailed in Ref. [2].

3.4. Validation Tolerances

Validation tests are evaluated against fixed numerical tolerances that define acceptance criteria for structural consistency. These thresholds are mandatory for all public releases of the SORT framework and are inherited from the core specification [1]. A summary is provided in Table 4.

Table 4. Validation tests and required tolerances.

Test	Tolerance	Status
Idempotency $\hat{O}_i^2 = \hat{O}_i$	$\leq 10^{-12}$	Required
Jacobi residual	$\leq 10^{-14}$	Required
Light-balance $\sum c_i = 0$	$\leq 10^{-14}$	Required
Global idempotency $\hat{H}^2 = \hat{H}$	$\leq 10^{-12}$	Required
Kernel normalization $\kappa(0) = 1$	exact	Required
Neighborhood commutators	$\leq 10^{-10}$	Required
Phase symmetry	$\leq 10^{-8}$	Required
Drift consistency	$\leq 2 \times 10^{-6}$	Required

3.5. Calibrated Reference Values

Validation relies on a minimal set of calibrated reference parameters that serve as structural anchors and are not re-fitted in v6 [2]. These values are summarized in Table 5.

Table 5. Calibrated reference parameters used in validation.

Parameter	Value	Source
σ_0	0.00190643	MOCK v3 calibration
H_{bare}	67.402 km s^{-1} Mpc^{-1}	Calibration
$\delta H / H_0$	0.0800	Layer II
β	0.5	Configuration
L_H	4285 Mpc	Physics
Seed	117666	Global

3.6. Reproducibility Manifest

Reproducibility is enforced through explicit versioning, immutable configuration files, and cryptographic checksums, following the practices defined in the SORT core framework [3]. All public validation artefacts are archived with persistent identifiers, including Zenodo releases. Golden run comparisons are used to verify bitwise identity across environments and platforms. Formal specifications of checksum generation and archive structure are provided in Appendix E.

4. Public Core API and Terminology Rules

4.1. Public Core Definition

The Public Core API defines the complete set of mathematical objects, validation rules, and structural invariants that are stable, published, and binding across all domains of the SORT framework. It comprises the resonance operator set $\{\hat{O}_i\}_{i=1}^{22}$, the global projector $\hat{H}$, the projection kernel κ, and all algebraic and structural invariants required for validation.

All elements of the Public Core are immutable within a given framework version. They may be referenced, instantiated, and evaluated by domain modules, but they must not be altered, extended, or reinterpreted at the implementation level. The Public Core API therefore constitutes the sole authoritative definition of the SORT mathematical framework and serves as the reference layer for reproducibility, peer review, and cross-domain consistency.

4.2. Internal Engine Definition

The Internal Engine denotes all computational components, heuristics, optimizations, and experimental extensions that are not part of the Public Core API. This includes implementation-specific execution logic, performance optimizations, exploratory algorithms, and auxiliary tooling used during development or internal validation.

The Internal Engine is explicitly excluded from publication and does not form part of the scientific claims of the framework. Results derived using the Internal Engine are only admissible for scientific reporting if they can be reproduced using the Public Core API and its documented validation procedures. This separation ensures that scientific conclusions are independent of proprietary or experimental implementation details.

4.3. Binding Terminology Rules

Terminology used throughout this manuscript is binding with respect to the Public Core API. Terms such as *operator*, *projector*, *kernel*, *transition*, and *validation* refer exclusively to their formal definitions as introduced in Section 2 and subsequent sections.

No alternative or metaphorical interpretations of these terms are implied unless explicitly stated. Domain-specific sections may introduce contextual interpretations, but these must remain consistent with the core definitions and must not redefine or overload established terminology. This rule ensures semantic stability across domains and prevents ambiguity in cross-referencing and validation.

4.4. Scope Protection Statements

The distinction between Public Core and Internal Engine is a scope protection mechanism. It ensures that the claims made in this manuscript are limited to structurally defined, reproducible, and publicly verifiable components. No claim of empirical validity, physical interpretation, or practical performance is made on the basis of Internal Engine behavior.

Accordingly, all scientific statements in this work are grounded exclusively in the Public Core API and its documented validation layers. This protection mechanism preserves the integrity of the framework description, enables independent verification, and allows future extensions without retroactive modification of published results.

5. Domain Module Architecture

This section specifies the architectural decomposition of SORT into domain modules. The domain architecture defines how a single, invariant mathematical core is instantiated across multiple application domains without duplication, interference, or modification of the underlying operator structure. The design follows the SORT core framework as established in Refs. [1–3].

5.1. Modular Decomposition

SORT is decomposed into a finite set of domain modules that share a common structural core. Conceptually, this decomposition is expressed as

$$\mathcal{M}_{\text{SORT}} = \mathcal{M}_{\text{COSMO}} \oplus \mathcal{M}_{\text{AI}} \oplus \mathcal{M}_{\text{CX}} \oplus \mathcal{M}_{\text{QS}}, \tag{24}$$

where each module represents a domain-specific interpretation layer.

The direct-sum notation denotes architectural separation rather than a decomposition of the resonance projection space $\mathcal{H}_R$. All modules operate on the same operator set $\{\hat{O}_i\}_{i=1}^{22}$, global projector $\hat{H}$, and projection kernel κ. Structural identity of the mathematical core is preserved across domains, while semantic interpretation and application logic remain domain-specific.

5.2. Domain Interface Contract

Each domain module implements a fixed interface contract that governs its interaction with the public core. Formally, every domain module is required to conform to an abstract base class,

$$\text{DomainModule} := \text{ABC}, \tag{25}$$

which defines mandatory structural methods and validation hooks.

Domain-specific inputs are constrained by a `DomainRequirements` schema that specifies admissible parameter ranges, operator sequences, and kernel configurations. All outputs returned to the core are encapsulated in an immutable boundary object,

$$\text{CoreResult} = \text{const}, \tag{26}$$

ensuring that domain modules cannot modify core state, operator definitions, or validation rules.

5.3. Canonical Domain Order

While all domain modules are structurally equivalent, a canonical ordering is defined for development, validation, and dissemination purposes [1]. The priority sequence reflects strategic considerations rather than mathematical dependence. AI systems are addressed first due to immediate safety and funding relevance, followed by complex systems for infrastructure-scale applications, quantum systems for error correction and noise diagnostics, and cosmology as an open research domain without intellectual property constraints.

This ordering does not impose dependencies between modules. Each domain can be developed, validated, and published independently once the core requirements are satisfied.

5.4. Cross-Domain Consistency

Cross-domain consistency is guaranteed by enforcing identical mathematical foundations across all modules. The operator set, global projector, and projection kernel are invariant,

$$\{\hat{O}_i\}_{\mathrm{AI}} = \{\hat{O}_i\}_{\mathrm{CX}} = \{\hat{O}_i\}_{\mathrm{QS}} = \{\hat{O}_i\}_{\mathrm{COSMO}}, \tag{27}$$

and all algebraic invariants defined in Section 2.5 are preserved.

Domain modules are prohibited from altering operator weights, kernel calibration, or validation tolerances. No domain is permitted to introduce feedback into another domain module. Structural isolation ensures that failures, approximations, or extensions in one domain cannot propagate into others.

5.5. Intellectual Property Strategy per Domain

The intellectual property strategy is defined per domain and is decoupled from the public core specification. The status of each domain module is summarized in Table 6.

Table 6. Intellectual property strategy per SORT domain module.

Domain	IP Status	Rationale
SORT-AI	Protected	Commercial potential and safety relevance
SORT-CX	Protected	Infrastructure and systems value
SORT-QS	Protected	Quantum technology IP landscape
SORT-COSMO	Open	Academic publication priority

This strategy allows open scientific dissemination of the core framework while enabling controlled commercialization and protection of domain-specific implementations, in alignment with the SORT core framework publications [2,3].

6. Domain Applications: SORT-AI (AI Safety)

The SORT-AI domain specifies a structural diagnostic layer for advanced AI systems, with emphasis on safety-relevant stability, drift, and runtime coherence under deployment conditions. The domain is defined as an application module of the invariant SORT core, and therefore inherits the operator set $\{\hat{O}_i\}_{i=1}^{22}$, the global projector $\hat{H}$, and the calibrated projection kernel κ from Section 2 and Section 3. The SORT-AI module follows the published core framework and its reproducibility contract [1–3] and specializes this machinery for alignment stability, drift detection, and scalable oversight in AI system stacks [6,7].

6.1. Domain Overview

SORT-AI targets structural diagnostics for AI systems, safety surfaces, and runtime coherence. The central objective is to evaluate stability and failure regimes at the system level without requiring access to model internals, weights, gradients, or training data. This scope aligns with the need for deployment-facing safety diagnostics articulated in established AI safety agendas [57,60]. In SORT-AI, the diagnostic object is the execution and control structure of an AI system, including orchestration layers, tool chains, retrieval components, and runtime controllers, rather than latent representations or mechanistic circuits. Consequently, SORT-AI is designed to remain applicable in black-box settings, including external APIs and proprietary models, where internal inspection is infeasible or restricted [6,58].

6.2. Application Catalog

The SORT-AI application catalog enumerates canonical diagnostic tasks that instantiate the public SORT core for safety-relevant system analysis [3,6]. The applications are:

Table 7. SORT-AI application catalog.

ID	Title	One-liner
ai.01	Interconnect Stability Control	Interconnect stability diagnostics
ai.02	Structural Drift Diagnostics	Structural drift detection
ai.03	Safety and Risk Surfaces	Projection-based risk surfaces
ai.04	Runtime Control Coherence	Runtime control coherence diagnostics
ai.05	Data and Retrieval Integrity	Structural retrieval integrity

Each item in Table 7 is defined as a domain interpretation of the invariant operators and kernel, and is implemented as a module conforming to the interface contract specified in Section 5.2. The catalog is structured to cover (i) infrastructure-induced instability modes, (ii) drift and distributional shift effects, and (iii) safety-relevant amplification pathways arising from control and retrieval coupling [7,57].

6.3. Methodological Framework

SORT-AI maps AI system components to operator chains acting on structured execution spaces. Let $\mathcal{S}_{\mathrm{AI}}$ denote an abstract system state encoding runtime configuration, orchestration constraints, and data-flow context. Structural evaluation proceeds by constructing admissible operator sequences $\hat{\mathcal{J}}_{\mathrm{AI}}$ from the public core operators and validating them using the diagnostics of Section 3.2. The minimal representation is

$$\hat{\mathcal{J}}_{\mathrm{AI}} = \prod_{m=1}^{M} \hat{O}_{i_m}, \tag{28}$$

with admissibility enforced through neighborhood commutator constraints and transition consistency, as formalized in Eq. 22 and Section 3.2.1.

Stability is evaluated as invariance under kernel-modulated projection and repeated normalization to a domain-specific normal form. The diagnostic projection is expressed at the public-core level as

$$\hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{AI}}) = \hat{H}\,\kappa(k)\,\hat{\mathcal{J}}_{\mathrm{AI}}\,\hat{H}, \tag{29}$$

where $\kappa(k)$ is defined in Eq. 17. Safety and risk surfaces are then defined as structural deviation fields over admissible configuration families, parameterized by perturbations Δ of the system state $\mathcal{S}_{\mathrm{AI}}$. A minimal risk functional consistent with the public core is the projector deviation norm

$$\mathcal{R}_{\mathrm{AI}}(\Delta) = \left\| \hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{AI}}(\Delta)) - \hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{AI}}(0)) \right\|_{\mathcal{H}_R}, \tag{30}$$

where $\|\cdot\|_{\mathcal{H}_R}$ denotes a norm on the resonance projection space introduced in Eq. 8. This construction yields a system-level diagnostic that is independent of model architecture and does not require mechanistic access to internal representations, while remaining compatible with standard safety problem framings [6,57].

For retrieval-augmented generation systems, the structural diagnostic targets retrieval-induced feedback loops and coupling between retrieval, ranking, and generation. In this setting, integrity checks are expressed as constraints on admissible retrieval operator segments and kernel responses, and are evaluated within the RAG diagnostic testbed formalized in [7].

6.4. Relation to Existing AI Safety Frameworks

SORT-AI is complementary to mechanistic interpretability and alignment frameworks [59,62]. It does not attempt to infer internal goals, representations, or circuits. Instead, it provides a structural diagnostic layer for system-level stability and coherence, addressing risks that arise from the interaction between models, runtimes, control layers, and external tools. This is consistent with the position that many practical safety failures emerge from coupled systems rather than from isolated model components [57,60]. Because SORT-AI is formulated in terms of the public core operators and the kernel-modulated projector, it remains applicable to black-box deployments, including externally hosted models and API-access systems [6,58].

6.5. Validation Strategy

SORT-AI validation follows the global three-layer validation architecture in Section 3.1 and is executed within the MOCK v4 environment [3]. Synthetic benchmarks are used to validate operator-chain admissibility, transition consistency, and projector deviation stability under controlled perturbations, consistent with the structural approach of Section 3.2. Case studies on deployed AI stacks are planned for version 7 and will focus on end-to-end integration scenarios, including runtime orchestration, tool use, and retrieval augmentation [6,7]. Reproducibility requirements, deterministic seeding, and checksum-based golden-run comparisons follow the public reproducibility manifest defined in Section 3.6 and Appendix E.

7. Domain Applications: SORT-CX (Complex Systems)

The SORT-CX domain applies the invariant SORT core to complex and distributed systems, including networks, dataflow pipelines, and large-scale computational infrastructures. The objective is the structural diagnosis of stability, emergence, and regime transitions in systems whose behavior is not reducible to isolated components. SORT-CX inherits the resonance operator set, the global projector $\hat{H}$, and the calibrated projection kernel κ defined in Section 2 and validated under the architecture of Section 3. The domain specification follows the SORT core framework and its modular application strategy [1–3,9].

7.1. Domain Overview

SORT-CX focuses on stability diagnostics for distributed systems, networks, and dataflow pipelines operating under heterogeneous coupling and scale separation. The primary capability of the domain is the detection of emergent instabilities, regime shifts, and structurally unstable configurations that are not apparent from local metrics or component-level monitoring. Rather than modeling explicit dynamics, SORT-CX characterizes systems through their structural response under projection, enabling diagnostics that remain valid across scales and implementation details. This approach is aligned with the need for system-level stability analysis in complex infrastructures and large distributed pipelines [1,9].

7.2. Application Catalog

The SORT-CX application catalog enumerates canonical diagnostic tasks that instantiate the public SORT core for complex systems analysis. The catalog is summarized in Table 8.

Table 8. SORT-CX application catalog.

ID	Title	One-liner
cx.01	Pipeline Stability Control	Drift and reproducibility diagnostics
cx.02	Emergent Stability under Projection	Stability islands and regime shifts
cx.03	Network Function Graph Stability	Structural graph stability metrics
cx.04	Detection Graph Drift Control	Drift control for detection graphs

Each application in Table 8 is defined as a domain-specific interpretation of the invariant resonance operators and kernel. No extensions of the operator algebra or validation rules are permitted. The catalog is designed to cover both data-centric pipelines and topology-driven network structures, reflecting the dominant sources of instability in complex technical systems [3,9].

7.3. Methodological Framework

SORT-CX maps network topology, pipeline structure, and dataflow dependencies to operator sequences acting on structured system spaces. Let $\mathcal{S}_{\mathrm{CX}}$ denote an abstract representation of a complex system state, encoding topology, coupling strength, and flow constraints. Structural evaluation proceeds by constructing operator chains

$$\hat{\mathcal{J}}_{\mathrm{CX}} = \prod_{n=1}^{N} \hat{O}_{i_n}, \tag{31}$$

whose admissibility is tested using neighborhood commutator constraints and transition consistency, as defined in Section 3.2 and Section 3.2.1.

Cascade analysis is performed by evaluating admissible successor relations in the operator adjacency structure introduced in Section 2.5.1. Structural cascades correspond to sequences of admissible transitions that amplify projector deviations across subsystems. Stability classification is obtained by projecting $\hat{\mathcal{J}}_{\mathrm{CX}}$ onto invariant subspaces via the global projector,

$$\hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{CX}}) = \hat{H}\,\kappa(k)\,\hat{\mathcal{J}}_{\mathrm{CX}}\,\hat{H}, \tag{32}$$

and evaluating the persistence of the resulting normal form under perturbations of $\mathcal{S}_{\mathrm{CX}}$. Regime shifts are identified as transitions between distinct invariant classes under controlled structural perturbations, independent of time parametrization or explicit dynamics [1,9,94,95].

7.4. Relation to Complex Systems Theory

SORT-CX complements classical complex systems and dynamical systems analysis by providing a structural, operator-based perspective on emergence and stability. Unlike dynamical approaches that emphasize trajectories, attractors, or bifurcation parameters [88,89], SORT-CX focuses on projection-invariant structure and admissible operator composition. This enables scale-invariant diagnostics that remain applicable across heterogeneous implementations and abstraction levels. The framework provides a formal bridge between network topology, pipeline structure, and emergent system behavior, without introducing additional dynamical assumptions, consistent with the structural positioning of the SORT core [1,9].

8. Domain Applications: SORT-QS (Quantum Systems)

The SORT-QS domain applies the invariant SORT core to quantum systems, with emphasis on structural diagnostics for noise, error correction, and hybrid quantum–classical workflows. The domain does not introduce new physical models of quantum dynamics. Instead, it provides an operator-geometric diagnostic layer that evaluates structural stability, admissibility of operator chains, and projection-invariant properties of quantum processes. SORT-QS inherits the resonance operator set, the global projector $\hat{H}$, and the calibrated projection kernel κ defined in Section 2 and validated under the architecture of Section 3. The domain specification follows the SORT core framework and its modular application strategy [1–3,8].

8.1. Domain Overview

SORT-QS focuses on structural diagnostics for quantum systems subject to noise, decoherence, and control imperfections [72,82]. The core capability of the domain is the analysis of operator chains representing quantum channels, gates, and hybrid workflows, without requiring full state tomography or explicit reconstruction of microscopic noise models. SORT-QS targets scenarios in which standard performance metrics are insufficient to characterize system-level stability, such as long-depth circuits, error-corrected logical layers, and hybrid quantum–classical execution pipelines. The framework provides projection-based diagnostics that remain valid across hardware platforms and abstraction layers [1,8].

8.2. Application Catalog

The SORT-QS application catalog enumerates canonical diagnostic tasks that instantiate the public SORT core for quantum systems analysis. The catalog is summarized in Table 9.

Table 9. SORT-QS application catalog.

ID	Title	One-liner
qs.01	Noise Filtering and Operator Diagnostics	Structural noise filtering
qs.02	Error Correction Diagnostics	Structural QEC criteria
qs.03	Hybrid Quantum Workflow Stability	Stability of hybrid workflows

Each application listed in Table 9 is defined as a domain-specific interpretation of the invariant resonance operators and projection kernel. No extensions of the operator algebra, kernel calibration, or validation tolerances are permitted. The catalog is designed to address both near-term noisy intermediate-scale quantum systems and error-corrected logical architectures [3,8].

8.3. Methodological Framework

SORT-QS maps quantum processes to operator chains acting on structured projection spaces. Let $\mathcal{S}_{\mathrm{QS}}$ denote an abstract quantum system state encoding channel composition, control structure, and noise context. Structural evaluation proceeds by constructing admissible operator sequences

$$\hat{\mathcal{J}}_{\mathrm{QS}} = \prod_{q=1}^{Q} \hat{O}_{i_q}, \tag{33}$$

which represent ordered compositions of quantum channels, control operations, and measurement-induced transformations. Admissibility is evaluated using neighborhood commutator constraints and transition consistency, as defined in Section 3.2 and Section 3.2.1.

Noise analysis is performed through projection diagnostics rather than explicit noise modeling. The kernel-modulated projection

$$\hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{QS}}) = \hat{H}\,\kappa(k)\,\hat{\mathcal{J}}_{\mathrm{QS}}\,\hat{H}, \tag{34}$$

filters structurally unstable components of the operator chain and yields a normalized representation suitable for comparison across noise regimes. Integration with existing quantum error correction protocols is achieved by applying SORT-QS diagnostics to logical operator layers and syndrome-processing pipelines, without modifying code constructions or recovery rules [8].

8.4. Relation to Quantum Information Theory

SORT-QS complements standard quantum information theory and quantum error correction approaches by adding a structural diagnostic layer. It does not replace stabilizer formalism, surface codes, or fault-tolerance thresholds [72,79,81]. Instead, it provides additional criteria for assessing operator-chain admissibility, noise amplification pathways, and hybrid workflow stability that are not captured by fidelity or logical error rates alone. The framework is compatible with established QEC architectures, including surface codes and stabilizer codes, and is applicable to hybrid quantum–classical systems where control logic and post-processing play a dominant role in overall stability [1,8].

9. Domain Applications: SORT-COSMO (Cosmology)

9.1. Domain Overview

The SORT-COSMO domain applies the invariant SORT core to cosmological observations with the goal of structurally organizing persistent anomalies and tension patterns across scales. The emphasis is placed on projection-level analysis of inferred quantities rather than on modifying gravitational dynamics or introducing new physical components. SORT-COSMO operates as a diagnostic and interpretative layer that complements standard cosmological modeling and enables a unified structural treatment of phenomena that appear disconnected when analyzed purely through dynamical extensions.

The domain is defined as an open research module with academic publication priority. All constructions inherit the public resonance operator set, the global projector $\hat{H}$, and the calibrated projection kernel κ defined in Section 2 and validated under the architecture of Section 3. The focus in version 6 is on capability definition and structural consistency, building on the analytic and numerical groundwork documented in the SORT-COSMO system specification [4].

9.2. Application Catalog

The SORT-COSMO application catalog enumerates canonical structural diagnostics that map distinct classes of cosmological observations to projection-based operator analysis. The catalog is summarized in Table 10.

Table 10. SORT-COSMO application catalog.

ID	Title	One-liner
cosmo.01	Early Galaxies	Projection-stabilized structures for early JWST
cosmo.02	Early SMBH Seeds	Kernel-controlled amplification for early SMBH
cosmo.03	Hubble Drift	Scale-dependent H_0 as projection drift
cosmo.04	CMB Anomalies	Large-scale CMB projection features
cosmo.05	Dark Baryon Oscillator	Coupled-sector surrogate tension patterns
cosmo.06	Intergalactic Bridges	Filamentary baryons as stable projections

Each application in Table 10 represents a domain-specific interpretation of the same underlying operator core. The catalog is designed to span early-Universe structure, late-time expansion diagnostics, and large-scale matter organization within a single structural framework.

9.3. Methodological Framework

SORT-COSMO maps cosmological observables and inferred fields to operator diagnostics acting on the resonance projection space $\mathcal{H}_R$. Observational inputs, such as galaxy number densities, inferred mass functions, expansion-rate measurements, and CMB angular spectra, are treated as structural data layers rather than as direct dynamical constraints. These layers are embedded into operator chains

$$\hat{\mathcal{J}}_{\mathrm{COSMO}} = \prod_{n=1}^{N} \hat{O}_{i_n}, \tag{35}$$

whose admissibility and stability are evaluated using the same neighborhood commutator and transition consistency criteria defined in Section 3.2 and Section 3.2.1.

Projection-based analysis is performed by applying the global projector and kernel-modulated normalization,

$$\hat{\mathcal{P}}_{\kappa}(\hat{\mathcal{J}}_{\mathrm{COSMO}}) = \hat{H}\, \kappa(k)\, \hat{\mathcal{J}}_{\mathrm{COSMO}}\, \hat{H}, \tag{36}$$

which filters structurally unstable components and yields normalized representations suitable for cross-scale comparison. In this framework, apparent anomalies are interpreted as projection-density effects or scale-coupling signatures rather than as failures of the underlying dynamical model.

The methodology explicitly preserves ΛCDM and General Relativity as the dynamical backbone. SORT-COSMO does not introduce alternative equations of motion or modified gravity terms. Instead, it provides a systematic way to organize how inferred quantities respond under projection and kernel modulation. This enables consistent comparison of early- and late-Universe observables, as well as joint interpretation of galaxy formation, black hole growth, expansion-rate measurements, and large-scale CMB features within a unified structural language [1,4].

9.4. Relation to Standard Cosmology

Within SORT-COSMO, standard cosmology remains fully operative at the level of dynamics and parameter inference [100,101]. The SORT framework adds a complementary structural diagnostic layer that operates on inferred observables and reconstructed fields. This separation allows SORT-COSMO to interface cleanly with existing pipelines, surveys, and simulations, while providing additional insight into cross-scale consistency and tension structure. Empirical confrontation with data and large-scale numerical validation are natural extensions of this framework and are planned for version 7 using high-performance computing resources [4].

9.5. Scope and Interpretation

The scope of SORT-COSMO in version 6 is defined by its role as a structural analysis framework. The domain focuses on internal structural quantities derived from the resonance operator core, projection densities, and scale-coupling diagnostics. These quantities are used to demonstrate how diverse cosmological observations can be organized and compared within a single projection-based architecture.

SORT-COSMO does not aim to replace established cosmological models or to re-derive observational results. Instead, it clarifies which aspects of observed tension patterns can be attributed to structural projection effects and which require genuinely new dynamical input. In this sense, the domain establishes a coherent interpretative layer that prepares the ground for systematic empirical testing and numerical validation in subsequent versions, while remaining fully compatible with the current cosmological standard model.

10. Methods and Implementation

10.1. MOCK v4 Architecture

10.1.1. Five-Layer Model

The overall structure of the MOCK v4 system is organized as a layered architecture that separates the published structural core from domain-specific analyses and application-level use cases. This separation enforces stability of the mathematical framework while allowing controlled extension across domains. The system layout is summarized schematically below.

```
                        MOCK v4 System


             PUBLIC CORE API (Stable, Published)
sort/core/
 operators/      # 22 Operators
 projector/      # Global Projector Ĥ
 kernel/         # Projection Kernel (k)
 invariants/     # Engine Invariants
 validation/     # Public validation tests



             DOMAIN MODULES (Scientific Analysis)
sort/domains/
 ai-systems/      # SORT-AI
 complex-systems/ # SORT-CX
 quantum-systems/ # SORT-QS
 cosmology/       # SORT-COSMO



             APPLICATION LAYER (Use Cases)
sort/applications/
 catalog/        # 18 Public Applications
```

The Public Core API defines the complete resonance operator set, the global projector $\hat{H}$, the projection kernel κ, and all invariant and validation routines. This layer is fixed and versioned to ensure long-term reproducibility.

Domain modules implement scientific interpretations of the core structure without modifying its definitions. Each domain operates as a read-only consumer of the core API and encodes domain-specific diagnostics and mappings.

The application layer collects validated use cases and catalogs admissible operator chains. It provides a stable interface for downstream evaluation while maintaining strict separation from non-public execution engines. “‘

10.1.2. Three-Layer Validation Pipeline

Validation within MOCK v4 is organized into a three-layer pipeline that mirrors the logical dependency structure of the framework. Layer I performs algebraic diagnostics on the resonance operator set, including idempotency checks, Jacobi identity verification, and light-balance consistency. Layer II constructs and validates the projection kernel and associated structural matrices, ensuring normalization and stability across scales. Layer III evaluates semi-spectral evolution and convergence properties of admissible operator sequences under repeated application.

Each layer produces deterministic validation artifacts that are consumed by subsequent layers, enforcing a strict bottom-up dependency structure as described in Section 3.

10.2. Algorithmic Framework

10.2.1. Operator Construction

Resonance operators are implemented as idempotent structural mappings acting on the resonance projection space. Each operator is uniquely identified by an index and carries a fixed structural weight consistent with the global balance condition. Algorithmic construction follows a uniform interface to ensure comparability and deterministic validation.

```
class ResonanceOperator:
def __init__(self, id: int, weight: float):
self.id = id
self.weight = weight  # ±1/11

def apply(self, state):
return self.project(state)

def validate_idempotency(self, test_state):
return norm(
self.apply(self.apply(test_state)) -
self.apply(test_state)
) < 1e-12
```

Idempotency validation thresholds correspond to the tolerances defined in Section 3.4 and are enforced uniformly across all operators.

10.2.2. Transition Validation

Operator transitions are validated through algorithmic checks that encode the structural adjacency criteria introduced in Section 3.2.1. Transitions are not interpreted as temporal evolution but as admissible structural mappings between operators within the closed core.

```
def validate_transition(O_i, O_j):
if commutator_norm(O_i, O_j) > TOLERANCE:
return False
if not closes_freedom(O_j, O_i):
return False
if not path_independent(O_i, O_j):
return False
return True
```

This procedure enforces compatibility, stability, and path independence, ensuring that all accepted transitions preserve the structural normal form of the operator chain.

10.3. Reproducibility Requirements

All MOCK v4 executions are fully deterministic. A fixed global seed is applied across all stochastic components, guaranteeing bitwise reproducibility of numerical artifacts. Every generated file, including intermediate diagnostics and final validation outputs, is accompanied by a SHA-256 checksum. Archival releases are deposited with persistent identifiers, and golden-run comparisons are used to detect deviations across environments and hardware platforms.

These requirements ensure that all reported validation results can be independently reproduced and verified without access to non-public components, in alignment with the reproducibility principles outlined in Section 3.

10.4. HPC Preparation

The MOCK v4 architecture is designed to scale toward high-performance computing environments in preparation for empirical validation planned in version 7. Core components are stateless and parallelizable, enabling distributed validation of operator sequences and kernel evaluations. Resource estimation and scheduling strategies are incorporated at the design level to support large-scale parameter sweeps and semi-spectral convergence studies without altering the public core API.

This preparation ensures continuity between the structural framework established in version 6 and the forthcoming empirical and numerical validation phase.

11. Discussion and Positioning

11.1. Relation to Existing Theories

11.1.1. Comparison with String Theory

SORT and string-theoretic approaches address different problem classes and operate at distinct conceptual levels. String theory aims at a unified dynamical description of fundamental interactions through higher-dimensional geometric constructions, whereas SORT is formulated as an operator-algebraic projection framework designed for structural diagnostics. Extra spatial dimensions are not required in SORT, and empirical predictions are explicitly deferred to later validation stages. The mathematical emphasis lies on closed operator structures rather than on geometric unification mechanisms.

Table 11. Comparison between String Theory and SORT.

Aspect	String Theory	SORT
Extra dimensions	Required	Not required
Empirical predictions	Limited	Deferred to v7
Mathematical framework	Geometric	Operator-algebraic
Primary focus	Unification	Structural diagnostics

11.1.2. Comparison with Loop Quantum Gravity

Loop Quantum Gravity formulates a background-independent quantization of spacetime geometry with discrete spectra for geometric observables. SORT does not assume fundamental space quantization and does not introduce discrete geometric operators. Instead, background independence is realized implicitly through projection-based structural mappings. Observable features are treated as emergent structural patterns rather than as spectra of quantized geometric variables.

11.1.3. Relation to Modified Gravity Approaches

SORT does not alter gravitational field equations and does not introduce modified dynamical laws such as those considered in MOND or $f(R)$ theories. Gravitational dy-

Table 12. Comparison between Loop Quantum Gravity and SORT.

Aspect	LQG	SORT
Space quantization	Fundamental	Not assumed
Background independence	Explicit	Implicit via projection
Observable predictions	Discrete spectra	Structural patterns

namics remain governed by established frameworks. SORT operates as a complementary structural layer that reorganizes inferred quantities without replacing or modifying the underlying dynamical descriptions.

11.2. Physical Interpretation

Within SORT, resonance operators encode stable structural response patterns under projection. These patterns emerge from the algebraic closure of the operator set and govern how information is distributed across scales. Projection operates as a geometric constraint that reorganizes relational content without introducing additional dynamical degrees of freedom.

The ontological status of operators—whether they represent fundamental structure or effective descriptions—remains an open question that empirical validation in version 7 and beyond will inform. SORT is constructed to be compatible with multiple interpretations while providing concrete, testable structural predictions.

11.3. Limitations and Future Work

11.3.1. Current Limitations

Version 6 completes the mathematical and architectural foundations of SORT, providing the stable core required for systematic empirical validation. The framework is formulated to enable quantitative confrontation with observational data while maintaining algebraic closure and reproducibility. Domain-specific applications are designed to interface with external datasets through the standardized domain contracts defined in Section 5.2.

11.3.2. Planned Developments for Version 7

Future work will focus on high-performance computing validation of structural diagnostics. Planned developments include scalable empirical validation pipelines, data assimilation protocols for observational inputs, and quantitative consistency checks against selected datasets. These steps are intended to test the structural capacity of SORT under realistic numerical conditions.

11.3.3. Long-term Research Directions

Beyond version 7, potential research directions include integration with mechanistic interpretability methods in artificial intelligence, extended applications in quantum error correction and hybrid workflows, and the exploration of additional domains where projection-based structural diagnostics may provide analytical value.

12. Conclusion and Outlook

12.1. Summary of Version 6 Contributions

Version 6 of the SORT framework establishes a consolidated structural foundation and a publication-safe architecture. The mathematical core is defined by a closed set of twenty-two resonance operators with verified algebraic consistency, including idempotency, Jacobi stability, and global balance. This core is embedded in a modular architecture that strictly separates a domain-independent public API from domain-specific application layers.

A dedicated validation infrastructure is provided through the MOCK v4 environment, implementing a three-layer verification pipeline that spans algebraic diagnostics, kernel construction, and semi-spectral convergence analysis. On this basis, an application catalog comprising eighteen publicly specified use cases across four domains is introduced. In addition, version 6 incorporates explicit operator transition laws and adjacency validation rules, enabling algorithmic assessment of structural neighborhoods within the operator space.

12.2. Toward Unified Description

The algebraic closure of the 22-operator set, combined with the calibrated projection kernel and validated transition laws, provides the mathematical machinery for a comprehensive structural theory. Version 7 will test whether this framework can quantitatively account for the cosmological anomalies that motivate the present work. Success would establish SORT as a candidate for a fundamental structural theory capable of unifying currently disparate physical phenomena.

12.3. Roadmap to Version 7

The development trajectory of SORT is organized into successive phases that reflect increasing empirical engagement and computational scope.

Table 13. Planned development phases of the SORT framework.

Phase	Primary Content	Timeline
v6 (current)	Framework architecture and validation core	December 2025
v7 (planned)	HPC-based validation and empirical tests	2026
v8 (planned)	Data assimilation and quantitative fits	2026–2027

Version 7 is intended to extend the existing framework toward large-scale numerical validation using high-performance computing resources, while preserving the public core API introduced in version 6.

12.4. Open Questions

Several questions remain open and motivate future investigation. These include the empirical discriminating power of projection-based structural diagnostics when confronted with observational data, potential points of contact with quantum gravity approaches, and the applicability of the framework to non-equilibrium and strongly time-dependent systems. Addressing these questions requires both computational scaling and careful integration with external theoretical and empirical inputs.

12.5. Final Statement

SORT v6 provides a mathematically closed and reproducible framework for unified structural analysis across cosmology, artificial intelligence, quantum systems, and complex systems. The 22-operator architecture, global projector, and calibrated kernel form an invariant core capable of addressing fundamental anomalies that resist purely dynamical explanation.

By completing the algebraic and architectural foundation before empirical engagement, SORT establishes the conditions for rigorous, reproducible validation in version 7. The framework is positioned to demonstrate that projection-based structural constraints can account for phenomena ranging from the Hubble tension to early massive galaxies—offering a path toward unified description that complements and extends established physical theories.

Author Contributions: The author carried out all conceptual, mathematical, structural, and editorial work associated with this manuscript. This includes: Conceptualization; Methodology; Formal Analysis; Investigation; Software; Validation; Writing – Original Draft; Writing – Review & Editing; Visualization; and Project Administration.

Funding: This research received no external funding.

Acknowledgments: The author acknowledges constructive insights from independent computational review systems and diagnostic tools whose structural assessments supported refinement of the resonance-operator algebra and kernel-filter integrations. Numerical checks and operator-chain analyses were performed using publicly available scientific software. No external funding was received.

Conflicts of Interest: The author declares no conflict of interest.

Use of Artificial Intelligence: Language refinement, structural editing and LaTeX formatting were partially assisted by large language models. All mathematical structures, operator definitions, derivations, diagnostics, theoretical developments and numerical validations were created, verified and approved by the author. AI tools contributed only to non-scientific editorial assistance.

13. References

13.1. Core SORT Publications

This work is part of the SORT publication sequence and relies exclusively on previously released SORT framework documents for its scientific and technical foundations. The conceptual projection-based formulation and the initial resonance-operator structure were introduced in the SORT Whitepaper v4 [1]. Mathematical consolidation, operator consistency conditions, and balance constraints were formalized in the hardened framework description of Whitepaper v5 [2].

The public implementation model, validation layers, and reproducibility guarantees employed throughout the present manuscript are specified in the MOCK v4 public skeleton and API specification [3]. Domain-specific structural applications referenced in this work are documented in dedicated SORT preprints covering cosmology [4,5], artificial intelligence systems [6,7], quantum systems [8], and complex systems [9]. Together, these publications define the complete public reference basis for SORT v6.

13.2. Methodological and Validation References

The validation philosophy adopted in SORT v6 follows the deterministic and reproducible execution model defined in the MOCK framework [3]. All numerical diagnostics, operator consistency checks, and kernel constructions referenced in this manuscript are constrained to the public validation layers specified therein. No additional methodological assumptions or external validation frameworks are introduced beyond those explicitly defined in the SORT and MOCK reference documents cited above.

13.3. Scope of External Literature

SORT v6 is a framework and architecture specification. It does not provide empirical fits, phenomenological claims, or domain-level result validation. Consequently, external domain literature in cosmology, artificial intelligence, quantum information, or complex systems is not treated as a primary reference source in this manuscript. Such literature is addressed in the corresponding domain-specific SORT publications and is intentionally not duplicated here in order to preserve the architectural and methodological focus of the present work.

Appendix A. Complete Operator Tables

Appendix A.1. Full 22-Operator Specification

This appendix provides the complete specification of the resonance operator set forming the public SORT core. Each operator is uniquely identified, structurally typed, and assigned a fixed weight consistent with the global balance condition defined in Section 2. The table enumerates algebraic properties only and does not assign ontological interpretation.

Table A1. Complete specification of the 22 resonance operators.

ID	Symbol	Type	Domain	Spectrum	Weight	Technical interpretation
1	$\hat{O}_1$	Self-adjoint	$\mathcal{H}_R$	Continuous	$+1/11$	Vacuum substrate
2	$\hat{O}_2$	Self-adjoint	$\mathcal{H}_R$	Continuous	$+1/11$	Boundary projection
3	$\hat{O}_3$	Unitary	$\mathcal{H}_R$	Discrete	$+1/11$	Non-local coherence
4	$\hat{O}_4$	Self-adjoint	$\mathcal{H}_R$	Bounded	$+1/11$	Complementarity
5	$\hat{O}_5$	Hermitian	$\mathcal{H}_R$	Continuous	$+1/11$	Cyclic parameter
6	$\hat{O}_6$	Self-adjoint	$\mathcal{H}_R$	Continuous	$+1/11$	Metric structure
7	$\hat{O}_7$	Unitary	$\mathcal{H}_R$	Continuous	$+1/11$	Invariant propagation
8	$\hat{O}_8$	Mixed	$\mathcal{H}_R$	Continuous	$+1/11$	Temporal integration
9	$\hat{O}_9$	Idempotent	Σ	Discrete	$+1/11$	Boundary operator
10	$\hat{O}_{10}$	Self-adjoint	$\mathcal{H}_R$	Recursive	$+1/11$	Recursive map
11	$\hat{O}_{11}$	Unitary	$\mathcal{H}_R$	Periodic	$+1/11$	Phase manifold
12	$\hat{O}_{12}$	Mixed	$\mathcal{H}_R$	Quasi-disc.	$-1/11$	Coarse graining
13	$\hat{O}_{13}$	Self-adjoint	$\mathcal{H}_R$	Integer	$-1/11$	Symmetry index
14	$\hat{O}_{14}$	Self-adjoint	$\mathcal{H}_R$	Continuous	$-1/11$	Superposition
15	$\hat{O}_{15}$	Hermitian	$\mathcal{H}_R$	Continuous	$-1/11$	Complementary directions
16	$\hat{O}_{16}$	Self-adjoint	$\mathcal{H}_R$	Logarithmic	$-1/11$	Statistical weight
17	$\hat{O}_{17}$	Self-adjoint	$\mathcal{H}_R$	Fractal	$-1/11$	Scale recurrence
18	$\hat{O}_{18}$	Mixed	$\mathcal{H}_R$	Continuous	$-1/11$	Symmetry breaking
19	$\hat{O}_{19}$	Self-adjoint	$\mathcal{H}_R$	Probabil.	$-1/11$	Information measure
20	$\hat{O}_{20}$	Unitary	$\mathcal{H}_R$	Functional	$-1/11$	Path integral
21	$\hat{O}_{21}$	Unitary	$\mathcal{H}_R$	Composite	$-1/11$	Composition
22	$\hat{O}_{22}$	Idempotent	$\mathcal{H}_R$	Cyclic	$-1/11$	Cycle closure

Appendix A.2. Operator Type Classification

The resonance operators fall into a small number of algebraic classes. Table A2 summarizes the distribution by type, following standard operator-theoretic classification [37,38].

Table A2. Operator type classification.

Type	Count	Operator IDs
Self-adjoint	12	1,2,4,5,6,10,13,14,15,16,17,19
Unitary	5	3,7,11,20,21
Idempotent	2	9,22
Mixed	3	8,12,18

Appendix A.3. Dimensional Hierarchy

Operators can be organized according to their effective dimensional contribution within projection space. Lower-index operators primarily act on scalar or boundary-level structure, while higher-index operators contribute to composite, cyclic, or closure behavior. This hierarchy is consistent with effective field theory style counting, where higher-order structural operators encode increasingly global or collective effects without introducing new degrees of freedom.

Appendix A.4. Fragment–Operator Mapping

Conceptual structural fragments introduced during framework development map uniquely onto the algebraic operator core [53,54]. Table A3 provides the complete correspondence.

Table A3. Fragment-to-operator correspondence.

Fragment	Structural motif	Operator
Vacuum / ground state	Pre-differentiation potential	$\hat{O}_1$
Projection	Boundary mapping	$\hat{O}_2$
Coherence	Non-local coupling	$\hat{O}_3$
Duality	Complementary pair	$\hat{O}_4$
Cyclic phase	Periodic structure	$\hat{O}_5$, $\hat{O}_{11}$
Geometry	Emergent metric	$\hat{O}_6$
Propagation	Invariant transport	$\hat{O}_7$
Temporal aggregation	Integration	$\hat{O}_8$
Boundary fixation	Idempotent cut	$\hat{O}_9$
Self-reference	Fixed-point recursion	$\hat{O}_{10}$
Coarse description	Resolution loss	$\hat{O}_{12}$
Symmetry encoding	Discrete index	$\hat{O}_{13}$
Superposition	Linear combination	$\hat{O}_{14}$
Directional pairing	Complementarity	$\hat{O}_{15}$
Entropy / weight	Statistical scaling	$\hat{O}_{16}$, $\hat{O}_{19}$
Scale recursion	Fractal return	$\hat{O}_{17}$
Symmetry breaking	Sector separation	$\hat{O}_{18}$
Path summation	Functional integration	$\hat{O}_{20}$
Composition	Operator chaining	$\hat{O}_{21}$
Closure	Cyclic consistency	$\hat{O}_{22}$

Appendix B. Commutator Matrix and Algebraic Structure

Appendix B.1. Complete 22×22 *Commutator Matrix*

This subsection documents the full commutator structure of the resonance operator set. For all ordered pairs $(i, j) \in \{1, \ldots, 22\}^2$, the commutator

$$[\hat{O}_i, \hat{O}_j]$$

is evaluated within the resonance operator algebra defined in Section 2. The complete numerical and symbolic commutator matrix is provided as a structured artifact within the MOCK v4 public validation layer and is referenced here as a formal object. The matrix encodes algebraic compatibility, admissible adjacency, and closure properties of the operator core and serves as the primary input for transition validation and neighborhood analysis.

Appendix B.2. Structure Constants

The commutator algebra is expressed in terms of structure constants f_{ij}^k [40] defined by

$$[\hat{O}_i, \hat{O}_j] = \sum_{k=1}^{22} f_{ij}^k \hat{O}_k, \quad \text{(A1)}$$

with antisymmetry $f_{ij}^k = -f_{ji}^k$ and consistency with the Jacobi identity as verified in Section 2.5. The coefficients f_{ij}^k are fixed elements of the public core and are evaluated numerically during Layer I validation. Their magnitudes provide a quantitative measure of operator compatibility and directly determine admissible transitions within the adjacency structure defined below.

Appendix B.3. Operator Transition and Adjacency Table

This subsection specifies the explicit operator transition rules introduced in version 8. Operator transitions are defined as formal structural mappings $\hat{T} : \hat{O}_i \to \hat{O}_j$ and do not represent temporal evolution, dynamical processes, or causal time ordering. Instead, transitions encode admissible structural adjacencies between resonance operators within the invariant operator space $\mathcal{H}_R$.

Admissibility of a transition is determined by algebraic and structural criteria implemented in the MOCK v4 public validation pipeline, including commutator compatibility, idempotency preservation, structural stability, and path independence. These criteria are evaluated deterministically and independently of any domain-specific interpretation.

Table A4 presents a representative subset of admissible operator transitions. The listed transitions are selected to illustrate the transition law and validation logic and are not intended to exhaustively enumerate all admissible adjacencies.

Table A4. Representative subset of admissible operator transitions illustrating the transition law and validation criteria.

From $\hat{O}_i$	To $\hat{O}_j$	Validation Test	Tolerance
$\hat{O}_1$ (Vacuum)	$\hat{O}_2$ (Projection)	Commutator	10^{-10}
$\hat{O}_2$ (Projection)	$\hat{O}_3$ (Coherence)	Commutator	10^{-10}
$\hat{O}_3$ (Coherence)	$\hat{O}_4$ (Duality)	Idempotency	10^{-12}
$\hat{O}_{10}$ (Self-reference)	$\hat{O}_{11}$ (Phase)	Stability	10^{-12}
$\hat{O}_{21}$ (Composition)	$\hat{O}_{22}$ (Closure)	Path independence	exact

The complete operator adjacency relation, including all admissible successor relations and forbidden transitions, is defined exclusively by the machine-readable adjacency matrix provided in the MOCK v4 public repository [3]. That artifact constitutes the authoritative specification of operator transitions. The table shown here serves illustrative purposes only and MUST be interpreted in conjunction with the MOCK v4 reference implementation and validation logic as described in Section 2.5.1.

Appendix B.4. Forbidden Transitions

Transitions not listed in Table A4 are forbidden by construction. A transition $\hat{O}_i \to \hat{O}_j$ is excluded if any of the following conditions hold:

- The commutator norm $\|[\hat{O}_i, \hat{O}_j]\|$ exceeds the prescribed tolerance.
- The target operator $\hat{O}_j$ does not close the active degree of freedom introduced by $\hat{O}_i$.
- Alternative transition paths yield inequivalent normal forms, indicating path dependence.

These criteria ensure that the operator core remains algebraically closed, structurally stable, and uniquely navigable under the transition rules defined in version 8.

Appendix C. Conceptual Development Sequence

Appendix C.1. Chronological Origin Sequence

This appendix documents the conceptual development sequence that guided the identification and consolidation of the resonance operator core. The sequence reflects an epistemic ordering of structural motifs encountered during analysis and serves to contextualize the emergence of the final operator set. It is presented as a structured narrative of conceptual convergence rather than as a formal derivation.

The initial motif concerns retro-causal consistency, understood as structural coherence under reversal or reordering of directional dependencies. This motif motivated the requirement that admissible structures remain consistent under transformations that do not privilege a single temporal ordering, leading to an emphasis on projection-invariant relations rather than directed evolution.

The second motif, projection over boundaries, arose from the observation that boundary mappings act as primary carriers of informational structure. This perspective emphasizes that relational content can be preserved and reorganized through boundary projections without requiring access to bulk detail, motivating the central role of projection operators within the framework.

Entangled duality emerged as a third motif, capturing the non-local coupling of complementary structures. This motif reflects the necessity of treating paired descriptions as structurally inseparable, even when represented in distinct formal contexts. It informed the treatment of dual operators and compatibility conditions within the operator algebra.

The motif of emergent geometry followed, emphasizing that metric and geometric structure arise as secondary constructs determined by relational and projection properties. Geometry is therefore treated as a consequence of structural organization rather than as a primitive input, aligning with the operator-based formulation of the framework.

Memory and self-reference constitute the fifth motif. Reflexive mappings and invariant structures were identified as necessary for maintaining consistency across iterative application and composition. This led to the explicit inclusion of self-referential operators and fixed-point stability conditions within the operator core.

The sequence concludes with closure via global projection. The introduction of a global projector formalizes structural closure by enforcing balance, idempotency, and consistency across the complete operator set. This step establishes a closed structural system without invoking external axioms or extensions.

For clarity and compact overview, Table A5 summarizes the six primary conceptual motifs that guided the early development of the SORT framework. The ordering reflects epistemic progression rather than logical derivation or physical hierarchy.

Table A5. Chronological overview of conceptual development motifs.

Order	Motif	Core structural idea
1	Retro-causal consistency	Structural consistency under reversed dependency
2	Projection over boundaries	Boundaries as primary information carriers
3	Entangled duality	Non-local coupling of complementary structures
4	Emergent geometry	Metric structure as derived outcome
5	Memory and self-reference	Reflexive stability and invariant recursion
6	Closure via global projection	Global structural consistency and balance

Appendix C.2. Clarification

The conceptual sequence presented above serves a limited and well-defined purpose. It records the developmental context in which structural motifs were identified and consolidated, but it does not constitute a metaphysical narrative or a claim about the nature of physical reality. No ontological status is assigned to the motifs or operators described.

Furthermore, the sequence is not intended as a formal derivation. The resonance operators do not follow deductively from the listed motifs, nor are the motifs sufficient to generate the operator algebra through logical implication. Instead, the sequence documents an epistemic pathway that informed the recognition of structural invariants later formalized in the operator framework.

Accordingly, this appendix should be read as contextual documentation that supports interpretative transparency, without extending the formal claims or scope of the mathematical framework presented in the main text.

Appendix D. Simulation Architecture

The simulation architecture described in this appendix constitutes the computational backbone of the SORT framework. All validation stages are implemented within the MOCK environments and are designed to ensure determinism, reproducibility, and strict separation between the published Public Core API and internal experimental components. The present description is consistent with the documented MOCK v3 numerical environment and its public successor MOCK v4 [3].

Appendix D.1. Layer I — Algebraic Validation

Layer I performs purely algebraic diagnostics on the public operator core. Its input consists exclusively of the formally specified resonance operators $\{\hat{O}_i\}_{i=1}^{22}$ together with their assigned weights. No numerical evolution, kernel construction, or domain interpretation is performed at this stage.

The validation suite evaluates idempotency of each operator, commutator closure, satisfaction of the Jacobi identity, and enforcement of the global light-balance constraint. These tests establish that the operator set forms a closed and internally consistent algebraic structure. In the reference MOCK v3 runs, all operators satisfied the required tolerances, with residuals below the prescribed thresholds, thereby qualifying the operator core for structural processing in higher layers [2].

Appendix D.2. Layer II — Structural Matrix Construction

Layer II consumes the algebraically validated operators produced by Layer I and constructs the structural objects required for projection-based diagnostics. Central components are the calibrated projection kernel $\kappa(k)$, the operator coupling matrix λ_{ij}, and associated structural response functions.

Kernel calibration is performed deterministically using a fixed global seed and yields a stable reference value for the kernel width parameter σ_0. This calibration procedure, including normalization checks and stability diagnostics, is documented in the public MOCK v3 archive and carried forward unchanged into MOCK v4 [3]. The primary outputs of Layer II are kernel-normalized structural matrices, resonance heatmaps, and conditioning diagnostics that characterize admissible coupling patterns.

Appendix D.3. Layer III — Semi-Spectral Evolution

Layer III applies a reduced semi-spectral evolution to the structural objects generated in Layer II. This evolution is not interpreted as physical time dynamics. Instead, it serves as a convergence and stability diagnostic that probes whether structural modes remain bounded, converge to stable configurations, and preserve invariants under repeated application.

The inputs to this layer consist of kernel-weighted operator matrices and validated coupling structures. Outputs include convergence metrics, invariant-preservation checks, and projection stability indicators. In the reference MOCK v3 environment, all validated configurations exhibited stable convergence behavior within the specified tolerances, confirming the internal consistency of the structural pipeline [2].

Appendix D.4. Validation Summary

The validation criteria enforced across the three-layer pipeline are summarized in Table A6. Each layer applies independent tolerance thresholds, and all criteria must be satisfied for a configuration to be accepted as structurally valid.

Table A6. Summary of validation layers, tests, and acceptance criteria.

Layer	Test class	Tolerance	Pass criterion
I	Algebraic consistency	10^{-12}	All operators pass
II	Structural conditioning	10^{-8}	Matrix well-conditioned
III	Semi-spectral convergence	10^{-6}	Convergence achieved

This layered validation architecture enforces a strict progression from algebraic correctness to structural normalization and convergence analysis. The separation of concerns ensures reproducibility, enables independent verification, and provides a stable foundation for future high-performance empirical studies planned for subsequent SORT framework versions [3].

Appendix E. Validation Manifest

This appendix specifies the reproducibility and verification requirements governing all numerical results produced within the SORT framework. The manifest applies uniformly to MOCK v3 reference runs and to the structurally equivalent MOCK v4 environment used for the present framework validation. All procedures described here are implemented in the public numerical archive and documented in the MOCK reference environment [3].

Appendix E.1. Reproducibility Specification

All validation runs are fully deterministic and reproducible under the following fixed execution parameters. These parameters are treated as part of the public validation contract and must not be modified for reference runs.

The global random seed is fixed to 117666. The schema version governing configuration and output formats is 0.5.1. The reference execution environment requires Python version 3.10 or higher and NumPy version 1.24 or higher. Additional dependencies are specified explicitly in the public MOCK repository and are version-pinned for reproducibility [3].

Appendix E.2. SHA-256 Checksums

To guarantee bit-level reproducibility, all numerical artifacts generated during a validation run are accompanied by SHA-256 checksums. A global checksum is computed over the complete directory structure of the validated run, including configuration files, operator definitions, kernel outputs, and diagnostic logs. For the reference MOCK v3 archive, the global checksum is

$$H_{\text{global}} = \texttt{F8305739EE41636E18D1B68E7CEE7010A91DECFCE012C38EDAC9B20B3269E507}. \tag{A2}$$

In addition, file-level checksums are generated at runtime for all layer-specific metrics and manifests. These include `layer1_metrics.json`, `layer2_metrics.json`, `layer3_metrics.json`, and the aggregated `manifest.json`. Any modification to numerical results, configuration parameters, or execution order necessarily results in a checksum mismatch and invalidates the run [3].

Appendix E.3. Golden Run Protocol

The golden run protocol defines the canonical procedure for validating a numerical execution against the reference results. A golden run is executed using the fixed seed 117666 and the unmodified public configuration files. All generated artifacts are then compared against the archived reference outputs.

Validation succeeds if and only if all file-level and global SHA-256 checksums match exactly. Any deviation, whether numerical or structural, must be documented explicitly together with the modified configuration and execution context. This protocol ensures that all reported structural diagnostics are independently verifiable and that future extensions or high-performance computing runs can be traced unambiguously back to the validated reference state [3].

Appendix F. Application Catalog (Complete)

Appendix F.1. Summary Table

This subsection provides a compact overview of the publicly specified SORT applications, grouped by domain and identifier range.

Table A7. Summary of public SORT applications by domain.

Domain	Count	Application IDs
SORT-AI	5	ai.01–ai.05
SORT-CX	4	cx.01–cx.04
SORT-QS	3	qs.01–qs.03
SORT-COSMO	6	cosmo.01–cosmo.06
Total	18	—

Appendix F.2. Application Details

For typographic stability, the public application catalog is reduced here to the minimal operational identifiers required for cross-referencing within the manuscript: application identifier, domain, title, and field of use.

Table A8. Public SORT applications: minimal catalog (identifier, domain, title, field of use).

Application ID	Domain	Title	Field of use
ai.01	SORT-AI	Interconnect Stability Control	Infrastructure
ai.02	SORT-AI	Structural Drift Diagnostics	AI systems
ai.03	SORT-AI	Safety and Risk Surfaces	Safety-critical
ai.04	SORT-AI	Runtime Control Coherence	Infrastructure
ai.05	SORT-AI	Data and Retrieval Integrity	AI systems
cx.01	SORT-CX	Pipeline Stability Control	Infrastructure
cx.02	SORT-CX	Emergent Stability under Projection	Research
cx.03	SORT-CX	Network Function Graph Stability	Infrastructure
cx.04	SORT-CX	Detection Graph Drift Control	Safety-critical
qs.01	SORT-QS	Noise Filtering and Operator Diagnostics	Quantum
qs.02	SORT-QS	Error Correction Diagnostics	Quantum
qs.03	SORT-QS	Hybrid Quantum Workflow Stability	Infrastr/QS
cosmo.01	SORT-COSMO	Early Galaxies	Research
cosmo.02	SORT-COSMO	Early SMBH Seeds	Research
cosmo.03	SORT-COSMO	Hubble Drift	Research
cosmo.04	SORT-COSMO	CMB Anomalies	Research
cosmo.05	SORT-COSMO	Dark Baryon Oscillator	Research
cosmo.06	SORT-COSMO	Intergalactic Bridges	Research

The application catalog defines the externally visible capability surface of the SORT framework. All applications consume the same public mathematical core and differ only in domain interpretation, evidence requirements, and infrastructural coupling.

Appendix G. References

1. Wegener, G. H. (2025). The Supra-Omega Resonance Theory (SORT): A Mathematically Hardened Projection Framework for Large-Scale Cosmological Structure. *Preprints* 2024111783. DOI:10.20944/preprints202511.1783.v2
2. Wegener, G. H. (2025). Supra-Omega Resonance Theory (SORT): An Operatoric Model of Cosmological Self-Coherence. *Zenodo*. DOI:10.5281/zenodo.17787754
3. Wegener, G. H. (2025). MOCK v4 — SORT v6 Public Skeleton. *Zenodo/GitHub*. DOI:10.5281/zenodo.18050207 Code: github.com/gregorwegener/SORT
4. Wegener, G. H. (2025). SORT-COSMO: A Projection-Based Structural Framework for Cosmology—Scale-Dependent Drift, Non-Local Correlations, and Projection-Induced Cosmic Structure. *Preprints* 2024121574. DOI:10.20944/preprints202512.1574.v1
5. Wegener, G. H. (2025). Resolving the Hubble Tension as a Scale-Dependent Projection Effect in the Supra-Omega Resonance Framework (SORT). *Preprints* 2024120727. DOI:10.20944/preprints202512.0727.v1
6. Wegener, G. H. (2025). SORT-AI: A Projection-Based Structural Framework for AI Safety—Alignment Stability, Drift Detection, and Scalable Oversight. *Preprints* 2024121334. DOI:10.20944/preprints202512.1334.v2
7. Wegener, G. H. (2025). SORT-AI: A Structural Safety and Reliability Framework for Advanced AI Systems with Retrieval-Augmented Generation as a Diagnostic Testbed. *Preprints* 2024121345. DOI:10.20944/preprints202512.1345.v1
8. Wegener, G. H. (2025). SORT-QS: A Projection-Based Structural Framework for Quantum Systems—Error Correction, Noise Filtering, and Operator Diagnostics. *Preprints* 2024122178. DOI:10.20944/preprints202512.2178.v1
9. Wegener, G. H. (2025). SORT-CX: A Projection-Based Structural Framework for Complex Systems—Operator Geometry, Non-Local Kernels, Drift Diagnostics, and Emergent Stability. *Preprints* 2024121431. DOI:10.20944/preprints202512.1431.v1
10. Riess, A. G., et al. (2022). A Comprehensive Measurement of the Local Value of the Hubble Constant with 1 km/s/Mpc Uncertainty from the Hubble Space Telescope and the SH0ES Team. *Astrophys. J. Lett.* **934**(1), L7. DOI:10.3847/2041-8213/ac5c5b
11. Planck Collaboration (2020). Planck 2018 results. VI. Cosmological parameters. *Astron. Astrophys.* **641**, A6. DOI:10.1051/0004-6361/201833910
12. Di Valentino, E., et al. (2021). In the Realm of the Hubble Tension—A Review of Solutions. *Class. Quantum Grav.* **38**, 153001. DOI:10.1088/1361-6382/ac086d
13. Freedman, W. L. (2021). Measurements of the Hubble Constant: Tensions in Perspective. *Astrophys. J.* **919**(1), 16. DOI:10.3847/1538-4357/ac0e95
14. Verde, L., Treu, T., & Riess, A. G. (2019). Tensions between the early and late Universe. *Nature Astronomy* **3**, 891–895. DOI:10.1038/s41550-019-0902-0
15. Labbé, I., et al. (2023). A population of red candidate massive galaxies $\sim$600 Myr after the Big Bang. *Nature* **616**, 266–269. DOI:10.1038/s41586-023-05786-2
16. Finkelstein, S. L., et al. (2023). A Long Time Ago in a Galaxy Far, Far Away: A Candidate $z \sim 12$ Galaxy in Early JWST CEERS Imaging. *Astrophys. J. Lett.* **946**(1), L13. DOI:10.3847/2041-8213/acade4
17. Naidu, R. P., et al. (2022). Two Remarkably Luminous Galaxy Candidates at $z \approx 10$–12 Revealed by JWST. *Astrophys. J. Lett.* **940**(1), L14. DOI:10.3847/2041-8213/ac9b22
18. Boylan-Kolchin, M. (2023). Stress testing ΛCDM with high-redshift galaxy candidates. *Nature Astronomy* **7**, 731–735. DOI:10.1038/s41550-023-01937-7
19. Arrabal Haro, P., et al. (2023). Spectroscopic confirmation of CEERS NIRCam-selected galaxies at $z \simeq 8$–10. *Nature* **622**, 707–711. DOI:10.1038/s41586-023-06521-7
20. Bogdán, Á., et al. (2024). Evidence for heavy-seed origin of early supermassive black holes from a $z \approx 10$ X-ray quasar. *Nature Astronomy* **8**, 126–133. DOI:10.1038/s41550-023-02111-9
21. Maiolino, R., et al. (2024). A small and vigorous black hole in the early Universe. *Nature* **627**, 59–63. DOI:10.1038/s41586-024-07052-5

22. Natarajan, P., et al. (2024). First Detection of an Overmassive Black Hole Galaxy UHZ1: Evidence for Heavy Black Hole Seed Formation from Direct Collapse. *Astrophys. J. Lett.* **960**(1), L1. DOI:10.3847/2041-8213/ad0e76
23. Volonteri, M., Habouzit, M., & Colpi, M. (2021). The origins of massive black holes. *Nature Rev. Phys.* **3**, 732–743. DOI:10.1038/s42254-021-00364-9
24. Inayoshi, K., Visbal, E., & Haiman, Z. (2020). The Assembly of the First Massive Black Holes. *Ann. Rev. Astron. Astrophys.* **58**, 27–97. DOI:10.1146/annurev-astro-120419-014455
25. Planck Collaboration (2020). Planck 2018 results. VII. Isotropy and Statistics of the CMB. *Astron. Astrophys.* **641**, A7. DOI:10.1051/0004-6361/201935201
26. Schwarz, D. J., et al. (2016). CMB Anomalies after Planck. *Class. Quantum Grav.* **33**, 184001. DOI:10.1088/0264-9381/33/18/184001
27. Copi, C. J., et al. (2010). Large-angle anomalies in the CMB. *Adv. Astron.* **2010**, 847541. DOI:10.1155/2010/847541
28. Gruppuso, A., et al. (2018). The evens and odds of CMB anomalies. *Phys. Dark Universe* **20**, 49–64. DOI:10.1016/j.dark.2018.03.002
29. DESI Collaboration (2024). DESI 2024 VI: Cosmological Constraints from the Measurements of Baryon Acoustic Oscillations. *arXiv:2404.03002*. arXiv:2404.03002
30. Alam, S., et al. (2021). Completed SDSS-IV extended Baryon Oscillation Spectroscopic Survey: Cosmological implications from two decades of spectroscopic surveys at the Apache Point Observatory. *Phys. Rev. D* **103**, 083533. DOI:10.1103/PhysRevD.103.083533
31. Eisenstein, D. J., et al. (2005). Detection of the Baryon Acoustic Peak in the Large-Scale Correlation Function of SDSS Luminous Red Galaxies. *Astrophys. J.* **633**, 560–574. DOI:10.1086/466512
32. Percival, W. J., et al. (2010). Baryon Acoustic Oscillations in the Sloan Digital Sky Survey Data Release 7 Galaxy Sample. *Mon. Not. R. Astron. Soc.* **401**, 2148–2168. DOI:10.1111/j.1365-2966.2009.15812.x
33. Tanimura, H., et al. (2019). Detection of intercluster gas in superclusters using the thermal Sunyaev–Zel'dovich effect. *Mon. Not. R. Astron. Soc.* **483**, 223–234. DOI:10.1093/mnras/sty3118
34. de Graaff, A., et al. (2019). Probing the missing baryons with the Sunyaev–Zel'dovich effect from filaments. *Astron. Astrophys.* **624**, A48. DOI:10.1051/0004-6361/201935159
35. Reiprich, T. H., et al. (2021). The Abell 3391/95 galaxy cluster system: A 15 Mpc intergalactic medium emission filament, a warped radio halo and an ultra-steep spectrum radio relic. *Astron. Astrophys.* **647**, A2. DOI:10.1051/0004-6361/202039590
36. Nicastro, F., et al. (2018). Observations of the missing baryons in the warm-hot intergalactic medium. *Nature* **558**, 406–409. DOI:10.1038/s41586-018-0204-1
37. Reed, M., & Simon, B. (1980). *Methods of Modern Mathematical Physics I: Functional Analysis* (Revised Ed.). Academic Press. ISBN 978-0-12-585050-6.
38. Conway, J. B. (2000). *A Course in Operator Theory*. American Mathematical Society. ISBN 978-0-8218-2065-0.
39. Halmos, P. R. (1982). *A Hilbert Space Problem Book* (2nd ed.). Springer, New York. ISBN 978-0-387-90685-0.
40. Kadison, R. V., & Ringrose, J. R. (1997). *Fundamentals of the Theory of Operator Algebras*, Vol. I. American Mathematical Society. ISBN 978-0-8218-0819-1.
41. Dixmier, J. (1981). *Von Neumann Algebras*. North-Holland. ISBN 978-0-444-86308-9.
42. Bratteli, O., & Robinson, D. W. (1987). *Operator Algebras and Quantum Statistical Mechanics 1* (2nd ed.). Springer, Berlin. ISBN 978-3-540-17093-8.
43. Bousso, R. (2002). The Holographic Principle. *Rev. Mod. Phys.* **74**, 825–874. DOI:10.1103/RevModPhys.74.825
44. Susskind, L. (1995). The World as a Hologram. *J. Math. Phys.* **36**, 6377–6396. DOI:10.1063/1.531249
45. Maldacena, J. (1999). The Large-N Limit of Superconformal Field Theories and Supergravity. *Int. J. Theor. Phys.* **38**, 1113–1133. DOI:10.1023/A:1026654312961
46. Witten, E. (1998). Anti-de Sitter Space and Holography. *Adv. Theor. Math. Phys.* **2**, 253–291. DOI:10.4310/ATMP.1998.v2.n2.a2
47. 't Hooft, G. (1993). Dimensional Reduction in Quantum Gravity. *arXiv:gr-qc/9310026*. arXiv:gr-qc/9310026

48. Ryu, S., & Takayanagi, T. (2006). Holographic Derivation of Entanglement Entropy from AdS/CFT. *Phys. Rev. Lett.* **96**, 181602. DOI:10.1103/PhysRevLett.96.181602
49. Dunford, N., & Schwartz, J. T. (1988). *Linear Operators, Part II: Spectral Theory*. Wiley-Interscience. ISBN 978-0-471-60847-3.
50. Simon, B. (2005). *Trace Ideals and Their Applications* (2nd ed.). American Mathematical Society. ISBN 978-0-8218-3581-4.
51. Kato, T. (1995). *Perturbation Theory for Linear Operators* (Reprint). Springer, Berlin. ISBN 978-3-540-58661-6.
52. Davies, E. B. (1995). *Spectral Theory and Differential Operators*. Cambridge University Press. ISBN 978-0-521-58710-5.
53. Blyth, T. S., & Santos, M. H. (2005). *Idempotent-Generated Algebras*. World Scientific. ISBN 978-981-256-246-1.
54. Golan, J. S. (1999). *Semirings and their Applications*. Springer, Dordrecht. ISBN 978-0-7923-5786-5.
55. Korobkov, S. S. (2010). Algebras with a lattice of idempotents. *Math. Notes* **88**, 373–384. DOI:10.1134/S0001434610090075
56. FitzGerald, D. G., & Leech, J. (2003). Dual symmetric inverse monoids and representation theory. *J. Austral. Math. Soc.* **64**, 345–367. DOI:10.1017/S1446788700039288
57. Amodei, D., et al. (2016). Concrete Problems in AI Safety. *arXiv:1606.06565*. arXiv:1606.06565
58. Russell, S. (2019). *Human Compatible: Artificial Intelligence and the Problem of Control*. Viking. ISBN 978-0-525-55861-3.
59. Ngo, R., Chan, L., & Mindermann, S. (2022). The Alignment Problem from a Deep Learning Perspective. *arXiv:2209.00626*. arXiv:2209.00626
60. Hendrycks, D., et al. (2022). Unsolved Problems in ML Safety. *arXiv:2109.13916*. arXiv:2109.13916
61. Gabriel, I. (2020). Artificial Intelligence, Values, and Alignment. *Minds and Machines* **30**, 411–437. DOI:10.1007/s11023-020-09539-2
62. Christiano, P., et al. (2017). Deep Reinforcement Learning from Human Feedback. *NeurIPS* **30**. NeurIPS 2017
63. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. ISBN 978-0-262-03561-3.
64. Szegedy, C., et al. (2014). Intriguing properties of neural networks. *ICLR 2014*. arXiv:1312.6199
65. Guo, C., et al. (2017). On Calibration of Modern Neural Networks. *ICML 2017*. ICML 2017
66. Ovadia, Y., et al. (2019). Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift. *NeurIPS* **32**. NeurIPS 2019
67. Lu, J., et al. (2020). Learning under Concept Drift: A Review. *IEEE Trans. Knowl. Data Eng.* **31**, 2346–2363. DOI:10.1109/TKDE.2018.2876857
68. Vaswani, A., et al. (2017). Attention Is All You Need. *NeurIPS* **30**. NeurIPS 2017
69. Brown, T., et al. (2020). Language Models are Few-Shot Learners. *NeurIPS* **33**. NeurIPS 2020
70. Radford, A., et al. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI Technical Report*. OpenAI
71. Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL-HLT 2019*. DOI:10.18653/v1/N19-1423
72. Nielsen, M. A., & Chuang, I. L. (2010). *Quantum Computation and Quantum Information* (10th Anniversary Ed.). Cambridge University Press. ISBN 978-1-107-00217-3.
73. Preskill, J. (2018). Quantum Computing in the NISQ era and beyond. *Quantum* **2**, 79. DOI:10.22331/q-2018-08-06-79
74. Arute, F., et al. (2019). Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510. DOI:10.1038/s41586-019-1666-5
75. Shor, P. W. (1997). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **26**, 1484–1509. DOI:10.1137/S0097539795293172
76. Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. *STOC 1996*, 212–219. DOI:10.1145/237814.237866
77. Shor, P. W. (1995). Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **52**, R2493–R2496. DOI:10.1103/PhysRevA.52.R2493
78. Steane, A. M. (1996). Error Correcting Codes in Quantum Theory. *Phys. Rev. Lett.* **77**, 793–797. DOI:10.1103/PhysRevLett.77.793

79. Gottesman, D. (1997). *Stabilizer Codes and Quantum Error Correction*. Ph.D. thesis, California Institute of Technology. arXiv:quant-ph/9705052
80. Kitaev, A. Yu. (2003). Fault-tolerant quantum computation by anyons. *Ann. Phys.* **303**, 2–30. DOI:10.1016/S0003-4916(02)00018-0
81. Fowler, A. G., et al. (2012). Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A* **86**, 032324. DOI:10.1103/PhysRevA.86.032324
82. Zurek, W. H. (2003). Decoherence, einselection, and the quantum origins of the classical. *Rev. Mod. Phys.* **75**, 715–775. DOI:10.1103/RevModPhys.75.715
83. Schlosshauer, M. (2007). *Decoherence and the Quantum-to-Classical Transition*. Springer, Berlin. ISBN 978-3-540-35773-5.
84. Breuer, H. P., & Petruccione, F. (2002). *The Theory of Open Quantum Systems*. Oxford University Press. ISBN 978-0-19-852063-4.
85. Lidar, D. A. (2013). *Quantum Error Correction*. Cambridge University Press. ISBN 978-0-521-89787-5.
86. Bar-Yam, Y. (1997). *Dynamics of Complex Systems*. Addison-Wesley. ISBN 978-0-201-55748-2.
87. Mitchell, M. (2009). *Complexity: A Guided Tour*. Oxford University Press. ISBN 978-0-19-512441-5.
88. Newman, M. (2010). *Networks: An Introduction*. Oxford University Press. ISBN 978-0-19-920665-0.
89. Strogatz, S. H. (2015). *Nonlinear Dynamics and Chaos* (2nd ed.). Westview Press. ISBN 978-0-8133-4910-7.
90. Holland, J. H. (1995). *Hidden Order: How Adaptation Builds Complexity*. Addison-Wesley. ISBN 978-0-201-40793-8.
91. Albert, R., & Barabási, A. L. (2002). Statistical mechanics of complex networks. *Rev. Mod. Phys.* **74**, 47–97. DOI:10.1103/RevModPhys.74.47
92. Barabási, A. L., & Albert, R. (1999). Emergence of Scaling in Random Networks. *Science* **286**, 509–512. DOI:10.1126/science.286.5439.509
93. Gao, J., et al. (2016). Universal resilience patterns in complex networks. *Nature* **530**, 307–312. DOI:10.1038/nature16948
94. Scheffer, M., et al. (2009). Early-warning signals for critical transitions. *Nature* **461**, 53–59. DOI:10.1038/nature08227
95. Scheffer, M., et al. (2012). Anticipating Critical Transitions. *Science* **338**, 344–348. DOI:10.1126/science.1225244
96. Kauffman, S. A. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press. ISBN 978-0-19-507951-7.
97. Prigogine, I., & Stengers, I. (1984). *Order Out of Chaos*. Bantam Books. ISBN 978-0-553-34082-2.
98. Jensen, H. J. (1998). *Self-Organized Criticality: Emergent Complex Behavior in Physical and Biological Systems*. Cambridge University Press. ISBN 978-0-521-48371-1.
99. Bak, P. (1996). *How Nature Works: The Science of Self-Organized Criticality*. Copernicus. ISBN 978-0-387-94791-4.
100. Weinberg, S. (2008). *Cosmology*. Oxford University Press. ISBN 978-0-19-852682-7.
101. Dodelson, S., & Schmidt, F. (2020). *Modern Cosmology* (2nd ed.). Academic Press. ISBN 978-0-12-815948-4.
102. Mukhanov, V. (2005). *Physical Foundations of Cosmology*. Cambridge University Press. ISBN 978-0-521-56398-7.
103. Ryden, B. (2017). *Introduction to Cosmology* (2nd ed.). Cambridge University Press. ISBN 978-1-107-15483-4.
104. Peebles, P. J. E. (1993). *Principles of Physical Cosmology*. Princeton University Press. ISBN 978-0-691-01933-8.
105. Springel, V., et al. (2005). Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature* **435**, 629–636. DOI:10.1038/nature03597
106. Mo, H., van den Bosch, F., & White, S. (2010). *Galaxy Formation and Evolution*. Cambridge University Press. ISBN 978-0-521-85793-2.
107. Press, W. H., & Schechter, P. (1974). Formation of Galaxies and Clusters of Galaxies by Self-Similar Gravitational Condensation. *Astrophys. J.* **187**, 425–438. DOI:10.1086/152650

108. White, S. D. M., & Rees, M. J. (1978). Core condensation in heavy halos: a two-stage theory for galaxy formation and clustering. *Mon. Not. R. Astron. Soc.* **183**, 341–358. DOI:10.1093/mnras/183.3.341
109. Verlinde, E. (2011). On the Origin of Gravity and the Laws of Newton. *J. High Energy Phys.* **2011**(4), 29. DOI:10.1007/JHEP04(2011)029
110. Verlinde, E. (2017). Emergent Gravity and the Dark Universe. *SciPost Phys.* **2**, 016. DOI:10.21468/SciPostPhys.2.3.016
111. Jacobson, T. (1995). Thermodynamics of Spacetime: The Einstein Equation of State. *Phys. Rev. Lett.* **75**, 1260–1263. DOI:10.1103/PhysRevLett.75.1260
112. Rovelli, C. (2008). Loop Quantum Gravity. *Living Rev. Relativ.* **11**, 5. DOI:10.12942/lrr-2008-5
113. Ashtekar, A. (1986). New Variables for Classical and Quantum Gravity. *Phys. Rev. Lett.* **57**, 2244–2247. DOI:10.1103/PhysRevLett.57.2244
114. Perez, A. (2013). The Spin-Foam Approach to Quantum Gravity. *Living Rev. Relativ.* **16**, 3. DOI:10.12942/lrr-2013-3
115. Thiemann, T. (2007). *Modern Canonical Quantum General Relativity*. Cambridge University Press. ISBN 978-0-521-84263-1.
116. Polchinski, J. (1998). *String Theory, Vol. 1: An Introduction to the Bosonic String*. Cambridge University Press. ISBN 978-0-521-67227-6.
117. Green, M. B., Schwarz, J. H., & Witten, E. (2012). *Superstring Theory, Vol. 1: Introduction* (25th Anniversary Ed.). Cambridge University Press. ISBN 978-1-107-02911-8.
118. Zwiebach, B. (2009). *A First Course in String Theory* (2nd ed.). Cambridge University Press. ISBN 978-0-521-88032-9.
119. Einstein, A. (1917). Kosmologische Betrachtungen zur allgemeinen Relativitätstheorie. *Sitzungsber. Königl. Preuß. Akad. Wiss. Berlin*, 142–152. Einstein Papers Project
120. Noether, E. (1918). Invariante Variationsprobleme. *Nachr. Königl. Ges. Wiss. Göttingen, Math.-Phys. Kl.*, 235–257. DOI:10.1080/00411457108231446 (English transl.)
121. Hilbert, D. (1915). Die Grundlagen der Physik. *Nachr. Königl. Ges. Wiss. Göttingen, Math.-Phys. Kl.*, 395–407.
122. Dirac, P. A. M. (1930). *The Principles of Quantum Mechanics*. Oxford University Press. ISBN 978-0-19-852011-5.
123. Golub, G. H., & Van Loan, C. F. (2013). *Matrix Computations* (4th ed.). Johns Hopkins University Press. ISBN 978-1-4214-0794-4.
124. Trefethen, L. N., & Bau, D. (1997). *Numerical Linear Algebra*. SIAM. ISBN 978-0-89871-361-9.
125. Press, W. H., et al. (2007). *Numerical Recipes: The Art of Scientific Computing* (3rd ed.). Cambridge University Press. ISBN 978-0-521-88068-8.
126. Higham, N. J. (2002). *Accuracy and Stability of Numerical Algorithms* (2nd ed.). SIAM. ISBN 978-0-89871-521-7.
127. Stodden, V., Leisch, F., & Peng, R. D. (eds.) (2014). *Implementing Reproducible Research*. CRC Press. ISBN 978-1-4665-6159-5.
128. Wilson, G., et al. (2014). Best Practices for Scientific Computing. *PLoS Biol.* **12**(1), e1001745. DOI:10.1371/journal.pbio.1001745
129. Sandve, G. K., et al. (2013). Ten Simple Rules for Reproducible Computational Research. *PLoS Comput. Biol.* **9**(10), e1003285. DOI:10.1371/journal.pcbi.1003285