

# The Fixed-Point Search

## A Computational Research Programme for the Emergent Stack

*Dimitar Popov*

---

### Abstract

The Constraint-Emergence Ontology proposes that physical reality is a deterministic constraint network whose large-scale structure — quantum mechanics, spacetime, matter — emerges through successive coarse-graining. *The Emergent Stack* showed that existing research programmes (t’Hooft, CDT/MERA, Connes) address distinct strata of this hierarchy. The missing piece is a computational strategy for finding the Stratum 0 rewriting rule that produces our physics.

Two search strategies are possible. The first — simulate candidate rules and compare output to known physics — is computationally intractable for any realistic rule space. The second — find rules whose coarse-graining converges to a fixed point, then check physics — is tractable, because fixed points are rare and isolated. The CEO’s self-bounding closure condition is precisely the fixed-point condition on the renormalisation group flow.

This paper specifies the computational research programme for the second strategy: a massively parallel search over hypergraph rewriting rules, filtering for RG fixed points, classifying by physical invariants, and testing against the proton-electron mass ratio as the primary falsifiability criterion.

The outer search loop is embarrassingly parallel (Ray/Dask, CPU cluster). The inner coarse-graining step is GPU-accelerated tensor computation (JAX). The mathematical bottleneck is the rigorous definition of the coarse-graining operator for hypergraph rewriting rules — the one unsolved piece that determines whether the computation finds the right fixed point. An approximate strategy using existing tensor renormalisation group methods allows Phase 1 to proceed before that definition is complete.

---

# Part I: The Search Problem

## 1.1 Why Brute Force Fails

The space of hypergraph rewriting rules is vast. Wolfram's enumeration of rules up to 3-node rewrites yields on the order of  $10^6$  to  $10^8$  candidates, depending on symmetry-class reduction. For each candidate, checking whether it produces correct physics requires:

- Running the hypergraph simulation for enough steps that large-scale structure emerges (typically  $10^4$  to  $10^6$  updates for any meaningful signal)
- Computing the causal graph and measuring its spectral dimension
- Checking Lorentz invariance of the emergent geometry
- Identifying stable persistent structures and measuring their characteristic scales
- Comparing all of this to known physics

Each candidate evaluation is expensive. The comparison criterion — “matches physics” — is not a single number but a conjunction of many conditions, most of which require long simulation runs to evaluate. Brute-force search in this space is computationally intractable at any realistic scale.

More fundamentally, brute-force search has no gradient. If rule  $R_{147}$  almost works and rule  $R_{148}$  fails completely, brute force learns nothing from  $R_{147}$ 's near-success. There is no sense in which the search is guided by proximity to the target.

## 1.2 Why Fixed-Point Search Is Different

The Wilsonian renormalisation group provides a different organising principle. Physical theories that look the same across a range of scales — that are scale-invariant — are at fixed points of the RG flow. Our observed physics has this property: QCD is asymptotically free (approaches a fixed point in the UV), electroweak theory is similarly structured, and general relativity has a candidate UV fixed point (asymptotic safety). The physical universe's effective theories are all at or near RG fixed points.

The CEO framework makes this precise. The self-bounding closure condition — the requirement that the constraint network's coarse-graining be self-consistent — is the ontological statement. The RG fixed-point condition is the mathematical statement. They are the same condition at different levels of description.

This reframes the search:

**Brute force:** Find  $R$  such that the simulation output matches physics.

**Fixed-point search:** Find  $R^*$  such that  $\text{coarse-graining}(R) = R$ . Check physics only among fixed points.

Fixed points are rare. In the space of all possible rewriting rules, most rules do one of two things under repeated coarse-graining: they flow to the trivial fixed point (empty graph, no dynamics) or they diverge (unbounded complexity). The non-trivial fixed points form a discrete set — or low-dimensional families called universality classes — in an otherwise continuous space. Searching among fixed points rather than all rules reduces the effective search space by orders of magnitude.

Crucially, fixed-point search has a gradient: the RG flow itself. A rule close to a fixed point flows toward it. Rules can be ranked by how many coarse-graining steps they take to converge — a natural proximity measure to the fixed point. The search is guided, not random.

### 1.3 The CEO’s Role

The CEO framework provides what Wolfram’s programme lacks: a principled target specification.

Wolfram’s search criterion is empirical — does the rule produce something that looks like our physics? The CEO’s criterion is structural — does the rule’s fixed point satisfy self-bounding closure and the system invariants INV-01 through INV-11?

The invariants function as necessary conditions that can be checked *before* the full physics comparison:

Invariant	Computational check
INV-06: c as universal maximum	Causal graph has uniform edge traversal rate
INV-07: Gravity as density gradient	Spectral dimension varies with local hyperedge density
INV-10: Determinism + epistemic randomness	Branchial graph gives well-defined probability measure
INV-11: Constraint topology precedes space	Causal graph has non-isomorphic spatial projection (tunneling paths exist)

Each check eliminates fixed points before the expensive full-physics comparison. The filter is applied in increasing order of computational cost.

## Part II: Theoretical Grounding

### 2.1 The Renormalisation Group on Rule Space

Let  $\mathcal{R}$  denote the space of all hypergraph rewriting rules up to a given complexity class. Define the **coarse-graining operator**  $CG_\lambda: \mathcal{R} \rightarrow \mathcal{R}$  as follows:

Given rule  $R$ : 1. Evolve a representative hypergraph  $G$  under  $R$  for  $T$  steps 2. Partition nodes of  $G$  into blocks of size  $\lambda$  3. Compute effective edges between blocks (from the density of connections within  $G$ ) 4. Extract the rule  $R'$  that best describes the block-level dynamics

The RG flow is the sequence:  $R \rightarrow CG_\lambda(R) \rightarrow CG_{\lambda^2}(R) \rightarrow \dots$

A **fixed point**  $R^*$  satisfies  $CG_\lambda(R) = R$  up to a rescaling of coupling constants.

The **basin of attraction** of  $R^*$  is the set of rules that flow to  $R^*$  under repeated application of  $CG_\lambda$ . This is the universality class. All rules in the same universality class produce identical large-scale physics — they differ only in microscopic details that wash out under coarse-graining.

The Wilsonian picture: we do not need the exact microscopic rule. We need the universality class. The class is characterised by a small number of parameters — critical exponents, scaling dimensions, the symmetry group of the fixed point — not by the full rule.

## 2.2 Self-Bounding Closure as Fixed-Point Condition

The CEO's self-bounding closure requires that the constraint network's gap-structure produce itself under coarse-graining. Formally: the effective theory at the coarse-grained level is the same theory as at the original level, up to rescaling.

This is the Banach fixed-point condition applied to the RG operator:  $\|CG_\lambda(R) - R\| \rightarrow 0$ .

The existence of such a fixed point is guaranteed under mild conditions (the RG operator must be a contraction in some metric on  $\mathcal{R}$  near the fixed point). Proving this for the specific case of causal hypergraph rewriting rules is the formal fixed-point theorem listed as an open problem in the CEO System Specification — a prerequisite for the programme's theoretical foundations, but not a prerequisite for beginning the computational search.

## 2.3 What the Fixed Point Must Produce

The physical fixed point  $R^*$  must satisfy the following invariants, derived from the CEO framework and from observation:

**\*\*Geometric invariants (from causal graph of  $R^*$ ):\*\*** - Spectral dimension  $\rightarrow 3+1$  at large scales,  $\rightarrow 2$  at Planck scales (matching CDT results) - Lorentz invariance: no preferred direction in the emergent metric - de Sitter geometry at cosmological scales

**\*\*Quantum invariants (from branchial graph of  $R^*$ ):\*\*** - Causal invariance: the branchial graph is well-defined (fixed point is causally invariant) - Born Rule: the natural measure on branchial branches is  $P = |\psi|^2$  - Entanglement entropy satisfies the Ryu-Takayanagi formula in the appropriate limit

**\*\*Matter invariants (from stable persistent structures of  $R^*$ ):\*\*** - Two or more classes of stable persistent structures exist (particle-like) - The symmetry group of the fixed point contains  $SU(3) \times SU(2) \times U(1)$  - The ratio of characteristic scales of the two simplest stable structures equals 1836.15

This is the complete target specification. A fixed point satisfying all of these is the physical fixed point. The search finds it by filtering successively through these conditions in increasing order of evaluation cost.

---

# Part III: Computational Architecture

## 3.1 Two-Level Parallelism

The computation has two natural levels of parallelism with different characteristics.

### Level 1 — Across rules (embarrassingly parallel, CPU cluster):

Each rule's RG flow is independent of every other rule's. No shared state, no inter-worker communication. The outer loop is a pure map:

```
for each R in rules (in parallel):  
    flow = compute_rg_flow(R)  
    emit (R, flow)
```

This scales linearly with the number of workers. Ray or Dask handles the distribution.  $10^6$  rules across  $10^3$  workers =  $10^3$  evaluations per worker. The search over the complexity  $\leq 3$  rule space is feasible on a modest GPU cluster in hours to days.

### Level 2 — Within a single coarse-graining step (GPU-accelerated):

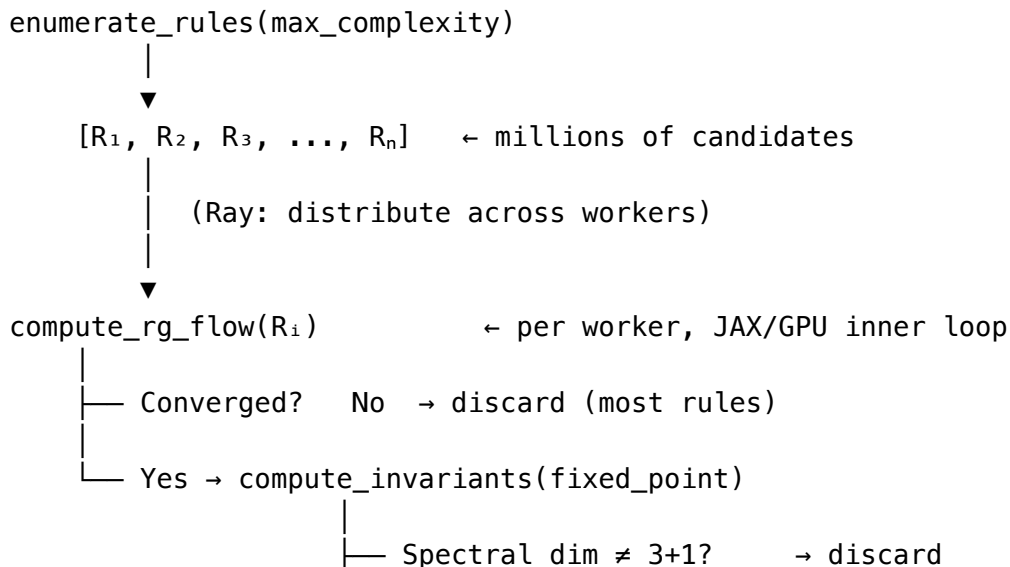
The coarse-graining operation itself is a tensor contraction. Given the coupling tensor  $T$  of a rule, one coarse-graining step is:

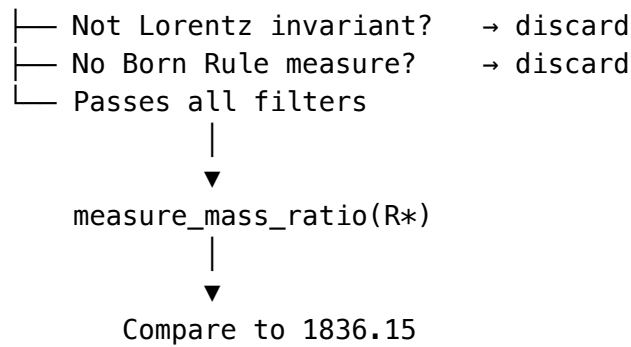
```
T' = contract(T, T, shared_indices) → normalise → truncate to bond  
dimension D
```

This is the Tensor Renormalisation Group (TRG) operation, implemented in JAX and running natively on GPU. A single coarse-graining step for a rank-4 tensor with bond dimension  $D = 16$  takes milliseconds on a modern GPU.

The combination: Ray distributes  $10^6$  independent jobs across a cluster; each job uses JAX on a local GPU to run 50 coarse-graining steps. Total wall-clock time is determined by the slowest batch, not the sum.

## 3.2 Data Flow





The filtering pipeline eliminates the vast majority of candidates before the expensive mass-ratio measurement. Only fixed points passing all geometric and quantum invariant checks reach the final test.

### 3.3 Core Functions

Three functions contain all the physics. The parallel scaffolding is standard engineering; these three are where the research lives.

**enumerate\_rules(max\_complexity)** Systematic enumeration of all hypergraph rewriting rules up to a complexity bound, reduced by isomorphism. Wolfram’s Mathematica implementation is open-source and translatable to Python. Isomorphism reduction uses standard graph canonicalisation (nauty/traces). Output: a generator over Rule objects, each specifying an input pattern, output pattern, and initial coupling tensor.

**coarse\_grain(rule, block\_size)** The critical function. Takes a rule and block size  $\lambda$ , returns the effective rule  $R'$  operating at the coarser scale. Implementation strategy:

- *Phase 1 (approximate)*: Block-spin methods from lattice QCD. Partition nodes by spatial proximity in the causal graph; compute majority-vote effective edges. This is computationally cheap and gives an approximate flow landscape — sufficient to identify which rules converge vs. diverge.
- *Phase 2 (rigorous)*: Once the coarse-graining operator is rigorously defined (see Part IV), replace the approximation with the correct computation.

The tension: the Phase 2 implementation requires solving Problem 3c (defining constraint density  $\rho_c$  on a non-spatial hypergraph). Phase 1 can proceed without this and still maps the landscape.

**compute\_invariants(fixed\_point)** Given a converged fixed point  $R^*$ , measure:

- *Spectral dimension*: Evolve the causal graph for  $10^4$  steps. Compute the return probability  $P(t)$  of a random walk. Fit  $d_s$  such that  $P(t) \sim t^{-d_s/2}$ . Check  $d_s \rightarrow 4$  at large  $t$ ,  $d_s \rightarrow 2$  at small  $t$ .
  - *Symmetry group*: Extract the automorphism group of the fixed-point rule’s local pattern. This is the candidate gauge group.
  - *Causal invariance*: Check that the branchial graph is well-defined (independent of rule application order). Non-causal-invariant fixed points are immediately discarded.
  - *Branchial measure*: Sample paths in the branchial graph. Measure the distribution of path weights. Check whether it is consistent with Born Rule statistics.
-

# Part IV: The Mathematical Bottleneck

## 4.1 Defining the Coarse-Graining Operator

The entire programme depends on the coarse-graining operator  $CG_\lambda$  being correctly defined. This is Problem 3c from *The Emergent Stack*: rigorous definition of constraint density  $q_c$  on a non-spatial hypergraph.

The difficulty: standard coarse-graining (block spin, decimation) relies on spatial proximity to define which nodes belong to the same block. The Stratum 0 hypergraph has no geometry — only graph-theoretic adjacency. The blocking must be defined in terms of graph connectivity, not spatial distance.

A candidate definition:

Two nodes belong to the same block if their constraint-topology distance (shortest path in the hypergraph) is  $\leq r$ , where  $r$  is the blocking radius.

The effective rule  $R'$  is then the rule that best reproduces the dynamics of the inter-block connections under the original rule  $R$ .

Three properties this definition must satisfy to be correct: 1. **Consistency:**  $CG_\lambda \circ CG_\mu = CG_{\{\lambda\mu\}}$  (composition law) 2. **Monotonicity:** Information is lost (not gained) at each coarse-graining step 3. **Correctness:** In the limit where  $R$  is a known lattice theory (e.g., lattice QCD),  $CG_\lambda(R)$  reproduces the known RG flow of that theory

Property 3 is the validation test. If the coarse-graining operator is applied to a known physical theory and recovers the known RG flow, the definition is correct. This is a testable mathematical criterion — it does not require solving any new physics.

## 4.2 Connection to Existing Methods

The Tensor Renormalisation Group (TRG), developed for tensor network coarse-graining in condensed matter physics, is the closest existing method. TRG represents the partition function of a statistical mechanical system as a tensor network and coarse-grains by contracting neighbouring tensors.

The adaptation to hypergraph rewriting rules: - The coupling tensor  $T$  of a rule plays the role of the local tensor in TRG - Contracting  $T$  with itself and truncating to bond dimension  $D$  is the coarse-graining step - The fixed point of this iteration is  $R^*$

TRG is well-implemented in JAX (multiple open-source libraries exist). The adaptation is non-trivial — TRG operates on regular lattices, hypergraph rules are irregular — but structurally compatible.

The research task: prove that TRG-style coarse-graining on hypergraph coupling tensors satisfies properties 1–3 above. If it does, TRG gives the rigorous coarse-graining operator and Phase 2 implementation becomes a standard tensor network computation.

## 4.3 What Approximate Methods Give You

Phase 1's approximate coarse-graining (block-spin) does not give the exact fixed point. It gives:

- A map of which rules flow toward fixed points vs. diverge
- An approximate location of fixed points in rule space
- The universality class structure (which rules cluster near the same attractor)

This is sufficient to: - Eliminate the vast majority of candidates (rules that diverge or hit trivial fixed points) - Identify candidate universality classes for detailed analysis - Test the parallel infrastructure at scale before the rigorous operator is defined

The flow landscape from Phase 1 constrains Phase 2's search. Instead of searching all  $10^6$  rules rigorously, Phase 2 searches only the  $\sim 10^{2-10^3}$  candidates that Phase 1 identifies as near non-trivial fixed points.

---

## Part V: Research Phases

### Phase 1 — Infrastructure and Approximate Landscape (Months 1–4)

**Goal:** Map the flow landscape with approximate coarse-graining. Identify candidate fixed points.

**Deliverables:** - Python implementation of rule enumeration (adapted from Wolfram's Mathematica code) - Ray-based parallel dispatch infrastructure - Approximate coarse-graining using block-spin methods - Flow landscape map for rules of complexity  $\leq 3$ : which converge, which diverge, where the attractors are - Identification of  $\sim 10^2$  candidate fixed-point universality classes

**What this requires:** Standard engineering. No new mathematics. Wolfram's code is the starting point for rule enumeration. Block-spin coarse-graining is textbook statistical mechanics.

**What this does not require:** Solving Problem 3c. Phase 1 is explicitly approximate.

### Phase 2 — Rigorous Coarse-Graining Operator (Months 3–9, overlapping)

**Goal:** Define  $CG_\lambda$  rigorously for hypergraph rewriting rules. Validate against known theories.

**Deliverables:** - Formal definition of constraint density  $q_c$  for non-spatial hypergraphs - Proof (or disproof) that TRG-style coarse-graining satisfies properties 1–3 - Validation: apply the operator to lattice QCD rules, verify recovery of known QCD RG flow - JAX implementation of the rigorous coarse-graining step

**What this requires:** Mathematical work — the definition and the consistency proofs. This is the research problem, not an engineering task. The validation against lattice QCD can be done computationally once the definition is in hand.



**Relationship to Phase 1:** Phase 2 runs in parallel with Phase 1. Phase 1's approximate results constrain which fixed points Phase 2 investigates rigorously.

### **Phase 3 — Rigorous Fixed-Point Classification (Months 8–18)**

**Goal:** Apply the rigorous coarse-graining operator to Phase 1's candidate fixed points. Classify by physical invariants.

**Deliverables:** - Rigorous fixed-point computation for the  $\sim 10^{2-10^3}$  candidates from Phase 1 - Spectral dimension measurements for each fixed point - Symmetry group classification for fixed points with correct spectral dimension - Elimination of candidates failing geometric invariants - Shortlist of fixed points with 3+1 Lorentz-invariant causal graph and correct symmetry structure

**What this requires:** Phase 2 complete. GPU cluster access for spectral dimension computations. Standard graph algorithms for symmetry group extraction.

### **Phase 4 — Physical Invariant Tests (Months 16–24)**

**Goal:** Test surviving candidates against quantum invariants and the mass-ratio criterion.

**Deliverables:** - Branchial graph analysis for shortlisted fixed points: Born Rule check - Stable persistent structure identification: which rules support stable “particle-like” patterns - Mass ratio measurement: ratio of characteristic scales of two simplest stable structures - Comparison to 1836.15

**Decision criterion:** - If a fixed point passes all tests and yields 1836.15 (within numerical precision): the universality class is identified. The CEO framework is falsifiable at this level and has passed its primary test. - If no fixed point passes: either the rule space is too small (increase complexity bound), the coarse-graining operator is wrong (return to Phase 2), or the CEO conjecture is false at the computational level.

---

## **Part VI: Falsifiability and Decision Criteria**

### **6.1 The Primary Test**

The proton-electron mass ratio, 1836.15, is the CEO framework's primary falsifiability criterion. It is chosen because:

- It is a dimensionless number, insensitive to unit conventions
- It is measured to six significant figures (no experimental uncertainty matters)
- It involves two of the simplest stable structures in physics (proton and electron)
- It is not an input to any existing theory — it is derived from the quark model, itself derived from QCD, but the ratio is not explained from first principles

In the CEO framework: the proton and electron are stable gap-patterns in the constraint topology. Their characteristic scales — the topological invariants that specify how large a stable pattern they are in the hypergraph — have a ratio of 1836.15. This ratio is not put in by hand; it is a consequence of the fixed point's structure.

If the computational search finds a fixed point whose two simplest stable structures have scale ratio 1836.15 without that number appearing anywhere in the rule definition, the programme has succeeded.

## 6.2 Intermediate Checkpoints

Before reaching the mass ratio test, intermediate results are falsifiable:

Checkpoint	What it tests	What failure means
Spectral dimension $\rightarrow 3+1$	The fixed point produces the right spatial structure	No fixed point with 3+1 spacetime exists in the rule space searched
Lorentz invariance	Causal invariance of the fixed point	Fixed-point rules are not causally invariant — the CEO framework needs revision
Born Rule measure	The branchial graph gives the right probability structure	The connection between Markov blankets and Hilbert space is wrong
Symmetry group $\supseteq \text{SU}(3) \times \text{SU}(2) \times \text{U}(1)$	The fixed point has the right gauge structure	The Connes connection fails; Standard Model derivation from Stratum 0 is incomplete
Mass ratio 1836.15	The fixed point produces the right particle content	The CEO is wrong, or the fixed-point conjecture is wrong, or the rule complexity bound is too low

Each checkpoint is a genuine go/no-go decision. If Lorentz invariance fails for all fixed points in the searched space, the programme stops and the theory is revised. The checkpoints are not arbitrary — they are necessary conditions for the CEO framework to be correct.

## 6.3 What Partial Success Means

A fixed point with correct spectral dimension but wrong symmetry group is a partial result — it tells us the coarse-graining procedure works (right geometry) but the connection to matter (Connes) is missing. This is not failure; it is localisation of the gap.

A fixed point with correct symmetry group but wrong mass ratio tells us the universality class is right but the stable structure identification is wrong — either the “particles” are being identified incorrectly, or a higher-complexity rule is needed.

Partial results constrain subsequent searches. The programme is iterative, not binary.

---

## Part VII: Resources and Scale

### 7.1 Computational Requirements

Phase	Compute type	Scale	Approximate cost
Phase 1 (complexity $\leq 3$ )	CPU cluster + GPU	$10^3$ worker-hours	\$500–\$2,000 cloud
Phase 1 (complexity $\leq 4$ )	GPU cluster	$10^5$ GPU-hours	\$50,000–\$200,000
Phase 2 (mathematical)	N/A	N/A	Research time only
Phase 3 (rigorous classification)	GPU cluster	$10^3$ GPU-hours (on shortlist)	\$5,000–\$20,000
Phase 4 (physical tests)	GPU cluster	$10^3$ GPU-hours	\$5,000–\$20,000

Phase 1 at complexity  $\leq 3$  is accessible to any research group with cloud budget. Phase 1 at complexity  $\leq 4$  requires dedicated cluster access. Phases 3 and 4 are moderate — most of the compute is eliminated by the filter pipeline.

### 7.2 Software Stack

Component	Tool	Status
Rule enumeration	Python (adapted from Wolfram)	Requires adaptation
Parallel dispatch	Ray or Dask	Available, mature
Coarse-graining (Phase 1)	NumPy / NetworkX	Straightforward
Coarse-graining (Phase 2)	JAX-based TRG	Available, requires adaptation
Causal graph simulation	Python (adapted from Wolfram)	Requires adaptation
Spectral dimension	SciPy random walk	Standard
Symmetry group	nauty/traces Python bindings	Available

No new programming language or framework is required. The entire stack runs in Python on standard hardware. The research investment is in mathematics (Phase 2) and algorithm adaptation (Phase 1 scaffolding), not in infrastructure.

### 7.3 What Exists That Can Be Reused

- **Wolfram Physics Project (open source):** Rule enumeration and causal graph simulation in Mathematica. The algorithms are documented and translatable.
  - **TRG libraries (JAX):** Multiple open-source implementations of Tensor Renormalisation Group for condensed matter. The coarse-graining inner loop adapts from these.
  - **CDT simulation code:** Several open-source CDT implementations exist (primarily C++). The spectral dimension measurement code is directly reusable.
  - **Connes’ spectral action code:** Computational implementations of the Connes-Chamseddine spectral action exist in the physics literature. Symmetry group extraction adapts from these.
- 

## Part VIII: Relationship to the CEO Series

This paper is the computational arm of the programme initiated in:

- **Constraint-Emergence Ontology (v1.3)** — the foundational framework. The invariants INV-01 through INV-11 are the target specification for the fixed-point search.
- **The Emergent Stack** — identifies the strata, the existing programmes, and the two open gaps (Gap 1: equivalence classes = Markov blankets; Gap 2: finite algebra uniqueness). This paper addresses both gaps computationally: Gap 1 is the Born Rule checkpoint; Gap 2 is the symmetry group checkpoint.
- **CEO System Specification (v1.4)** — the LLM-loadable spec; the Inverse Projection Problem in §X is the formal statement of what this research programme solves.

The five Junction Equations from the conceptual analysis map directly to this programme:

Junction Equation	This programme’s treatment
1. Substrate Engine ( $D(x,c)$ as local operator)	The rewriting rule $R$ is $D(x,c)$ . Finding $R^*$ is finding the operator.
2. Born Rule from Markov quotient	Phase 4 checkpoint: branchial measure test
3. MERA-CDT equivalence	Phase 3: spectral dimension + Lorentz invariance test (computational signature of the link)
4. Connes algebra uniqueness	Phase 3: symmetry group classification of fixed points
5. Discrete-to-continuous time projection	Implicit in Phase 3: spectral dimension crossover from 2 (Planck) to 4 (large scale)

The Holy Grail — deriving 1836.15 — is Phase 4’s terminal test.

---

# Conclusion

The fixed-point search is not a philosophical programme. It is a computational one, with a specific data structure (hypergraph rewriting rules), a specific algorithm (iterative coarse-graining via TRG), a specific parallel architecture (Ray outer loop, JAX inner loop), and a specific falsifiability criterion (1836.15).

The one unsolved mathematical piece — the rigorous definition of the coarse-graining operator for non-spatial hypergraphs — is the genuine research problem. Everything else is engineering. Phase 1 proceeds without it, using approximate methods, and constrains the search space for Phase 2. Phase 2 solves the mathematical problem and validates the operator against known theories. Phases 3 and 4 apply it.

The research is falsifiable at every phase. It either produces the fixed point with the right physics, or it identifies precisely where the CEO framework needs revision. Both outcomes are scientifically valuable.

The stack is real. The fixed point is what makes it self-consistent. Finding it is a tractable computational problem.

---

## Appendix: Open Problems as Research Tasks

The open problems from the CEO System Specification §X translate directly into research tasks within this programme:

Open Problem	Phase	What resolution looks like
Discrete/continuous gap	3	Spectral dimension crossover in fixed-point causal graph
Constraint density metric	2	Rigorous $q_c$ definition satisfying properties 1–3
Constant derivation	4	1836.15 from fixed-point topology
Three generations	4	Hilbert space dimension of fixed-point symmetry algebra
Self-bounding closure	2	Formal fixed-point existence theorem for $CG_\lambda$ on $\mathcal{R}$
Inverse projection problem	3	Identification of $CG_\lambda$ consistent with all observed spacetime properties

