Crystallon: Learning by the Growth and Dissolution of a Discrete Knowledge Carrier

Anonymous authors

Abstract

Symbolic and connectionist learners share an assumption: the knowledge carrier is fixed—rules over a given vocabulary, or weights in a fixed architecture—and learning fills in values. We study *Crystallon*, a gradient-free learning system that drops this assumption. Its carrier is dynamic: a population of discrete structural units that are created, deepened, and dissolved by access-driven consistency dynamics, with no value-fitting inner loop, so that what is learned is which units exist, how deeply they crystallize, and how long they persist. None of the individual mechanisms is new—each descends from adaptive resonance, synaptic consolidation, disuse theory, or program induction—and the claim we defend is narrow and architectural rather than a new paradigm. We show that this commitment yields two *structural existence* properties, stated as facts about what an architecture can represent rather than as performance wins. (i) The number of independently representable concepts is not bounded by an embedding dimension: a fixed d-dimensional representation has its independent capacity capped at d by a rank argument, whereas Crystallon's is set by its dynamics and exceeds d. (ii) Growth is closure-gated: on a pure-noise stream Crystallon instantiates no structure, whereas assign-every-point methods such as CF-trees and k-means are forced to. We further show that forgetting is driven by the content of what recurs rather than by elapsed time or access count, yielding a recall/recognition dissociation and relearning savings, and we give a proof of concept in which representational *form* accretes from cross-stimulus recurrence without search. We also characterize a structural *boundary*: marginal growth is blind to cross-map feature binding, which we locate precisely and tie to feature-integration theory. Crystallon is a structure-discovery system rather than a classifier, and we evaluate it as such. Its contribution is not a new learning mechanism but an architectural commitment—that the knowledge carrier is itself dynamic—together with a characterization, in part by provable argument, of what that commitment makes possible and where it reaches a precise boundary.

1 Introduction

Symbolic and connectionist learning both *fix the knowledge carrier*—rules over a given vocabulary, or weights in a fixed architecture—and learn by *filling in values*. *Crystallon* makes the carrier itself dynamic: discrete units are born, deepen, compete, and dissolve through access dynamics, with no gradient and no value-fitting inner loop.

The shared assumption is so basic it is easy to overlook. A symbolic learner is handed a vocabulary and a space of rules and searches within it; a connectionist learner is handed an architecture and adjusts the values of its weights. In both, the *carrier* of knowledge—what can be represented at all—is committed at design time, and learning rearranges values on top of it. This is enormously effective when the carrier is well matched to the task, and it underwrites most of modern machine learning. It also means the shape of what can be learned is decided in advance: a fixed-width representation has a fixed capacity, a fixed vocabulary cannot name what it has no symbol for, and the act of admitting a genuinely new unit is either taken by the designer or smoothed into a differentiable relaxation. Crystallon asks what learning looks like when that act is returned to the system and taken discretely—when the carrier may grow a unit, deepen one, or let one dissolve, and these acts of growth and dissolution *are* the learning rather than scaffolding

around it. The question is not new in spirit; what we contribute is an architecture that commits to it fully and a characterization of the consequences. We are deliberately modest about the size of this claim. The rhetoric of an organizing principle "beyond" symbolism and connectionism is itself well populated—adaptive resonance has long been framed this way, as have unifications of competitive learning—and every mechanism we use descends from prior work (Section 3). The novelty we defend is therefore architectural rather than atomic, and the orthogonality we claim is narrow and at a single level: *what the carrier is*, not a new account of learning at large.

Concretely, a Crystallon model is a population of discrete structural units, each carrying a position in a feature space, a crystallization depth, and a durability. Learning proceeds one *access* (one input) at a time. When no existing unit lies close enough to an input, a new unit is *seeded*—a literal branch on whether the best match's similarity clears a threshold, not a relaxed or differentiable surrogate. When an input is corroborated by existing units, the matching unit is *deepened*; when a unit goes uncorroborated, it *dissolves* at a rate set by its durability. Readout is a logical OR over the active units rather than a weighted sum or softmax. Nothing in this loop fits values by gradient descent: what is learned is *which units exist, how deeply they have crystallized, and how long they persist*—the topology and depth of the carrier, not numbers placed into a fixed one.

Crystallon is a structure-discovery system, not a classifier, and we are deliberate about the basis on which we ask to be judged. We do not chase benchmark accuracy: when a structure-discovery system is scored as a classifier the result is a category check rather than a fair test, a point we make precise in Section 8. The claims that matter here are about *structural properties*—existence facts about what a dynamic carrier can represent, scoped to the architecture classes against which they hold—and these are where the paper places its weight. Two are sharp (Figure 1). First, the number of concepts that can be represented *independently* is, for any fixed d-dimensional representation, capped at d by a rank argument; Crystallon's independent capacity is set by its own dynamics and exceeds d across the whole range we test. Second, Crystallon's growth is *closure-gated*: presented with a pure-noise stream it instantiates no structure at all, whereas *assign-every-point* methods such as CF-trees and k-means—which have no operation for refusing to grow—are forced to manufacture clusters from noise. Neither property is a tuning artifact; the first is a rank argument and the second follows from what the baseline architectures lack. We are careful, in Section 4, to say exactly what each property is *not*: in particular we do not claim that fixed-d classification is worse, nor that this holds against all clustering methods.

Beyond these two, Crystallon exhibits a forgetting whose driver is the *content* of what recurs rather than elapsed time or access count—producing a recall/recognition dissociation (structure intact while confidence has decayed) and relearning savings—and, as a proof of concept, a regime in which the representational *form* itself accretes from cross-stimulus recurrence without any search over a hypothesis space.

Who should care. (i) For *continual and online learning*, Crystallon is a concrete point in design space where forgetting is content-driven and structure is added only on corroboration—useful as a contrast case, even though it is not competitive on accuracy. (ii) For *clustering and nonparametric methods*, closure-gated growth and the capacity-beyond-d result sharpen what separates "grow on demand" methods from assign-every-point ones. (iii) For *memory-inspired and biologically grounded learning*, Crystallon shows one way to make graded forgetting and form discovery first-class, gradient-free representational operations rather than failure modes.

Contributions.

- **(C1) A dynamic-carrier organizing principle and architecture.** We present Crystallon and show that its mechanisms—seeding, deepening, metastable dissolution, a confidence/durability split, form precipitation, and a hierarchical-OR readout—follow from one commitment rather than being independently chosen parts.

- **(C2) Structural-existence results.** Independent-representation capacity is not bounded by the embedding dimension, and growth is closure-gated, so no structure is instantiated on pure noise—

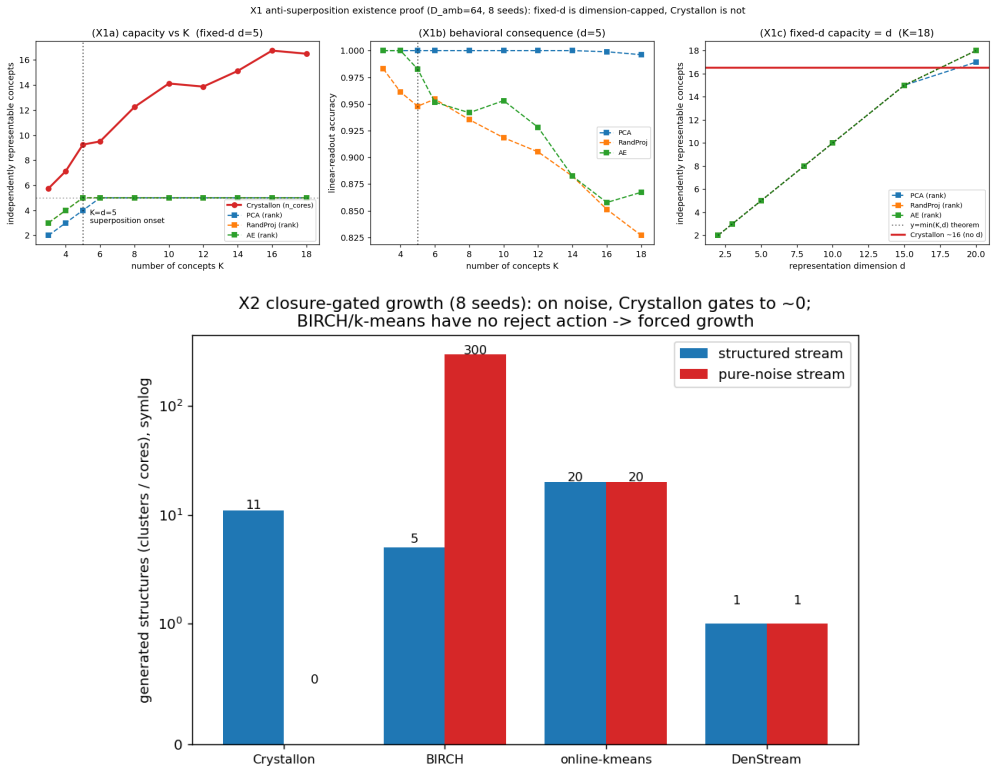

Figure 1: Two structural-existence results. **Top (capacity beyond the embedding dimension):** number of independently representable concepts versus the number of concepts K; fixed-d representations (PCA, random projection, autoencoder) saturate at $\min(K, d)$ while Crystallon's count is set by its dynamics and exceeds d. **Bottom (closure-gated growth):** structures instantiated on a structured stream versus a pure-noise stream; Crystallon grows nothing on noise, whereas assign-every-point baselines (CF-tree/BIRCH, k-means) are forced to. These are properties a fixed-carrier method cannot have, not performance comparisons.

properties unavailable to fixed-dimension and assign-every-point methods, stated as structural properties rather than performance wins.

- **(C3) Content-driven forgetting.** Forgetting is governed by the content of what recurs rather than by elapsed time or access count, producing a recall/recognition dissociation and relearning savings.
- **(C4) Form precipitation (proof of concept).** Representational form can accrete from cross-stimulus recurrence on a continuous substrate without search.
- **(C5) A precisely characterized capability boundary.** Marginal (single-feature) growth is, by construction, blind to cross-map feature binding; we locate the boundary exactly (single-feature statistics $z \leq 0.5$ vs. a per-element joint primitive $z = 128.8$, robust over 20 seeds $\times$ three noise levels) and connect it to feature-integration theory and the neural binding problem.

Claim ledger (stated up front). Because the structural results are easy to over-read, we fix their scope here, once, and hold to it throughout (the same ledger is repeated where the evidence is presented, Section 4). **We claim:** (X1) for any fixed d-dimensional code, independent-representation capacity is rank-capped at d, whereas Crystallon's exceeds d; (X2) assign-every-point methods (CF-trees, k-means) are architecturally forced to instantiate structure on pure noise, whereas Crystallon's recurrence gate is not. **We do *not* claim:** that fixed-d classification accuracy is worse (it is not—a d=5 PCA code classifies our well-separated concepts perfectly); that Crystallon's capacity is unbounded (it has its own dynamic ceiling, ≈16 here); or

that *all* online clustering is forced to grow on noise (density-based methods have their own gate). These are existence results—properties a fixed-carrier / assign-every-point architecture *cannot* have—not performance comparisons. Crucially, none of the individual mechanisms is claimed as new; each is grounded in prior work (Section 3), and the contribution is the architectural commitment that no prior system adopts in full together with a characterization of what it makes structurally available.

2 The Crystallon architecture

A Crystallon model is a population of discrete structural units sharing a feature space $\mathbb{R}^{\dim}$. Each unit i carries a position $\boldsymbol{p}_i \in \mathbb{R}^{\dim}$, a crystallization depth $d_i \geq 0$, a confidence $c_i \in [0,1]$, a durability M_i, a lability r_i, and a distortion f_i. Units are organized into trees by a parent pointer—a unit with no parent is a *trunk* (root), a child is a *twig*—so structure can be of arbitrary depth. The clock is the access count: one step is one input presentation, never physical time, and all parameters are dimensionless. Learning is the event loop below; there is no objective minimized by gradient descent and no inner optimization—updates are closed-form increments triggered by each access. (Units live in fixed-size arrays with an `active` mask; a dissolved unit is never hard-deleted but goes silent as its depth decays and the budget reclaims it. Default parameter values are in Appendix A.)

Matching. For an input $\boldsymbol{x}$, each active unit responds with a Gaussian similarity

$$\mathrm{sim}_i = \exp\big(-\|\boldsymbol{x} - \boldsymbol{p}_i\|^2/2\ell^2\big).$$

Two summaries drive the discrete decision. The *anchor* $a = \max_i c_i$ is an *activation*—how confident the currently present structure is—deliberately not a similarity. The *coverage* $\mathrm{cov} = \mathrm{mean}\{\mathrm{sim}_i : \mathrm{sim}_i > \mathrm{cov_thresh}\}$ is the consistency of the responders; an access is *referenced* when coverage clears a threshold (broad mutual corroboration) and *unreferenced* otherwise. Referenced accesses receive a high cross-check weight η, unreferenced a low one—internal coverage-consistency stands in for a supervised target.

Seed vs. deepen (the discrete commitment). The decision to instantiate a new unit is a literal branch on *space*, not on confidence:

$$\text{if } \max_i \mathrm{sim}_i < \mathrm{sim_seed} \;\Rightarrow\; \text{seed a new trunk at } \boldsymbol{p} = \boldsymbol{x}.$$

This spatial criterion is what later yields the recall/recognition split (Section 5): a concept whose confidence has cooled but whose structure remains still has high similarity, so it is recognized and deepened, not mistaken for something new and re-seeded. A new unit starts shallow (d=0, c=0.1) with its durability fixed at formation from the novelty of the content,

$$M \leftarrow M_{\mathrm{floor}} + w_{\mathrm{surprise}}\,(1 - \max_i \mathrm{sim}_i) + \mathrm{priority}.$$

When the input instead matches existing structure, the matched unit j is *deepened*. Confidence rises by corroboration density with diminishing returns,

$$c_j \leftarrow c_j + \kappa\,\eta\,(1 - c_j);$$

durability ratchets upward by the *span* over which the content recurs—a function of span, never of raw repetition count,

$$M_j \leftarrow \max\big(M_j,\; M_{\mathrm{base},j} + w_{\mathrm{span}} \tanh(\mathrm{span}_j)\big), \qquad \mathrm{span}_j = \tfrac{t - \mathrm{first_seen}_j}{\mathrm{span_unit}};$$

and depth crystallizes only once confidence clears a soft gate,

$$d_j \leftarrow d_j + \alpha_d\,\sigma\big((c_j - c^\star)/w_{\mathrm{gate}}\big)\,(1 - f_j).$$

Confidence and durability are therefore driven by different aspects of the same event—density vs. span—so a unit can occupy any cell of a confidence×durability grid, including "important but not yet confident."

Continuous dynamics and two-timescale forgetting. After each access every active unit relaxes. Confidence cools on a fast scale,

$$c_i \leftarrow c_i \, e^{-1/\tau_c},$$

while depth undergoes *metastable* dissolution on a slow scale whose rate is set by the unit's own depth and durability,

$$d_i \leftarrow d_i - \beta_i \, d_i, \qquad \beta_i = \beta_0 \, e^{-d_i M_i / D_c}.$$

A hard invariant, asserted at construction, ties the two: $\tau_c < 1/\beta_0$—the most stubborn confidence decay is still faster than the most fragile structural dissolution. Without it the model would confabulate (structure gone, confidence remaining); with it, the reachable intermediate state is "structure present, confidence gone"—recognition without recall. A finite budget rescales depths when their sum exceeds B, and within a multi-branch tree a fraction ϕ of the depth dissolving off the smaller branches condenses back onto the root trunk (an Ostwald-ripening reflux), consolidating shared cores at the expense of incidental detail.

Readout. Expression is a logical OR over active units rather than a weighted sum or softmax. With effective evidence $e_i = d_i(1 - f_i)$,

$$P = 1 - \prod_{i \text{ active}} \big(1 - \sigma\big((e_i - \theta)/\tau_r\big)\big).$$

Optionally the readout is hierarchical: a twig's evidence is multiplied by its root trunk's presence, so a twig whose trunk has dissolved falls silent—a detached detail is meaningless on its own. Empirically, when a trunk dissolves, a flat OR still fires on the now-orphaned twig ($P = 1.00$) while hierarchical trunk-gating silences it ($P = 0.35$).

One commitment, not six parts. Each mechanism above expresses a single commitment: the carrier is dynamic, and the birth, depth, and decay of its units *are* the learning. Seeding is the birth of carrier; deepening is depth accruing on corroboration; metastable dissolution is the death of carrier on disuse; the OR readout reads structure rather than summing values; the confidence/durability split lets "how sure" and "how durable" move independently. None is a value-fitting step, and it is the absence of a fixed carrier that makes them cohere. Section 3 places each mechanism against the prior work it descends from.

3 Positioning: one principle, many precedents

Crystallon's novelty is architectural, not atomic. Each mechanism of Section 2 rests on an established foundation, and we make those lineages explicit—not to bargain the contribution away, but because a synthesis is only as strong as the ground it stands on, and naming the ground is how we show it is solid. The contribution is the organizing commitment that no prior system instantiates in full (Table 1) together with the structural properties it provably yields (Section 4). We write each lineage as "in the spirit of X, but unlike X," and the very breadth of these foundations—spanning machine learning, computational neuroscience, and cognitive science—is itself a sign that the synthesis is principled rather than assembled ad hoc.

Gated, non-gradient growth. Seeding a unit by a non-gradient threshold test is in the spirit of ART's vigilance-gated category creation (Carpenter & Grossberg, 1987; Carpenter et al., 1991; Carpenter & Grossberg, 2016) and DP-means' hard-threshold cluster creation (Kulis & Jordan, 2012). The closure gate of Section 4.2 is the same family of move. But unlike ART, which tunes adaptive prototype weights during resonance, and DP-means, which fits centroids to their members, Crystallon's seed carries no value-fitting inner loop, and its units may later dissolve. Knowledge held in a grown topology, learned without gradients, is shared with incremental decision trees (Domingos & Hulten, 2000); unlike VFDT's greedy, supervised, axis-aligned splits, Crystallon grows unsupervised generative units that compete and dissolve. Constructive networks (Fahlman & Lebiere, 1990) likewise grow structure, but train it by backpropagation.

Table 1: No prior system shares Crystallon's full commitment; each shares a proper subset. The contribution is the conjunction, not any single row.

Neighbour	Shared subset	Where it departs from the full commitment
ART (Carpenter & Grossberg, 2016)	gated growth, no backprop	tunes adaptive weights; built for *permanence* (no dissolution)
Fusi cascade / Benna–Fusi (Fusi et al., 2005)	metastability, lifespans	carrier is *fixed* (synapse-internal state)
Hoeffding tree / VFDT (Domingos & Hulten, 2000)	non-gradient, knowledge in topology	greedy *supervised* splits; discriminative; no dissolution
DP-means (Kulis & Jordan, 2012)	threshold-gated growth	fits centroids (value-fitting); assign-every-point; no dissolution
Constructive NN (Fahlman & Lebiere, 1990)	architecture grows	trained by *backprop* (value-fitting)
CCM (Ororbia, 2021)	non-gradient competition, lifelong	fixed unit count; no dissolution; no metastability

Metastable durability and lifespans. Durability that governs a slow, graded decay is in the spirit of cascade and multi-timescale models of synaptic consolidation (Fusi et al., 2005; Benna & Fusi, 2016). We are explicit that the overlap runs deep: the cascade model already sets durability in a single exceptional event (a neuromodulator that drives a synapse into resistant states in one shot), so we do not claim formation-time durability as novel. The delta is the *level* at which it acts. In the cascade the synaptic carrier is fixed and metastability is an internal state of each synapse; in Crystallon durability is an explicit per-unit scalar that governs whether a structural unit persists or is removed—the carrier itself lives and dies.

Confidence versus durability. Splitting "how sure" from "how durable" is in the spirit of Bjork and Bjork's distinction between retrieval and storage strength, including the dissociation between them (Bjork & Bjork, 1992). Unlike that account—where the two are scalars on a fixed trace and storage strength is monotonic—Crystallon's confidence and durability are two separately driven quantities (corroboration density vs. span), confidence relaxing on a fast timescale while durability sets the slow dissolution rate.

Form precipitation. The idea that structure can be induced by detecting recurrence in data, rather than searched over a hypothesis space, is established—data-driven structure induction and anti-unification (Rule et al., 2024; Ellis et al., 2023). We concede the concept. What we claim (as a proof of concept, Section 6) is the instantiation: form accreting on a *continuous* representational substrate through a non-search precipitation dynamic, with no domain-specific language and no Bayesian search.

The "third principle" rhetoric. Claims of an organizing principle beyond symbolism and connectionism are not new: ART has long been framed this way (Carpenter & Grossberg, 2016), as have unifications of competitive learning (Ororbia, 2021). We therefore make the orthogonality claim narrowly and at one level only—*what the carrier is*—and support it by the observation that no neighbour adopts the full commitment.

It is the conjunction—a dynamic carrier, no value-fitting inner loop, units that are born and dissolve, the confidence/durability split, and the OR readout, under one commitment—together with the structural properties it makes available (Section 4), that constitutes the contribution. No row of Table 1 does this on its own.

4 Structural-existence results

We do not argue that Crystallon performs better than its baselines; we argue that it has properties they *cannot* have. The distinction matters: a claim of the form "method A outperforms method B" is fragile to configuration and can be overturned by a fairer baseline, whereas a claim that "architecture B cannot, by construction, exhibit property P" is a structural statement. We report two such properties. Throughout,

Table 2: Independently representable concepts vs. number of concepts K (ambient D_{amb}=64, embedding d=5, 8 seeds). Fixed-d capacity (matrix rank) locks at $\min(K, d)$=5; Crystallon's core count is set by its dynamics and exceeds d across the whole range.

K	Crystallon (`n_cores`)	PCA rank	RandProj rank	AE rank
3	5.8	2.0	3.0	3.0
5	9.2	4.0	5.0	5.0
6	9.5	**5.0**	**5.0**	**5.0**
10	14.1	5.0	5.0	5.0
14	15.1	5.0	5.0	5.0
18	16.5	**5.0**	**5.0**	**5.0**

`crystallon/core.py` is byte-identical across all runs (it is only observed, never tuned per experiment), and every baseline uses its official implementation and default configuration.

Before characterizing limits, we confirm the prerequisite holds: the system self-organizes. On interleaved unlabeled streams it forms the correct number of well-separated concepts—across 7/7 runs there is no collapse and no run-away, the inter-core distance stays near the true value (≈2.0) and never approaches zero, and injecting a new concept raises the core count by exactly one. We note honestly that this is demonstrated for the fixed, hand-given kernel, under which representational collapse is in fact structurally unreachable; testing self-organization under a *learnable* representation belongs to the deferred encoder work (Section 8).

4.1 Capacity beyond the embedding dimension

Proposition. Fix a d-dimensional representation and let it encode K concepts by their code-space means. Those K means span a subspace of dimension at most $\min(K, d)$; once $K > d$ at most d of them are linearly independent. Independent-concept capacity is therefore capped at d by a rank argument, for *any* fixed d-dimensional code. Crystallon allocates a unit (a *core*) per concept, and the number of cores it sustains is set by its access dynamics, not by an ambient dimension.

Setup. We generate K well-separated synthetic concepts in ambient dimension D_{amb}=64 and sweep K across d. Baselines are official `PCA`(d), `GaussianRandomProjection`(d), and a d-bottleneck `MLPRegressor` autoencoder. The primary measure is the number of *independently representable* concepts: for a fixed-d code, the matrix rank of the K concept means in the code; for Crystallon, its core count `n_cores`. We deliberately calibrate d=5, small enough that Crystallon's own dynamic ceiling (≈16 in this setting) sits well above d so that the comparison is not confounded by that ceiling.

Result. Fixed-d capacity locks at d as soon as $K > d$, for all three baselines, while Crystallon rises to ≈16 (Table 2, Figure 1 left). Varying d at fixed K=18 confirms the mechanism directly: fixed-d capacity equals $\min(K, d)$ (PCA/RandProj/AE rank $= 2, 5, 10, 15$ at $d = 2, 5, 10, 15$, and 17/18/18 at d=20), whereas Crystallon's count is flat at ≈16 and has no d to vary.

What the property is, and is not. The cost of superposition falls on *independent representational capacity* (rank), not on single-label classification accuracy. A linear 1-of-K readout from the d=5 PCA code stays at 1.00 even at K=18 (well-separated blobs remain separable by hyperplanes even when their means are linearly dependent; RandProj/AE drift to 0.83/0.87). We therefore claim only that the *rank* is dimension-capped, not that fixed-d classification is worse. Two further boundaries are explicit. (i) Crystallon has its own *dynamic* ceiling (here ≈16; higher under other settings), so the property is "not bounded by d," not "unbounded." (ii) `n_cores` is not a faithful count of K (it over-segments at small K—K=3→5.8—and saturates at large K); it faithfully supports only the inequality $>d$. The difference bites in the open-world / lifelong setting where K is unknown: a fixed-d method must commit to a dimension in advance and is forced to superpose once K exceeds it, whereas Crystallon allocates on demand within its dynamic ceiling.

Table 3: Structures instantiated on a structured stream vs. a pure-noise stream (8 seeds). Assign-every-point methods (BIRCH, k-means) are forced to grow on noise; Crystallon does not. DenStream has its own (density) gate and is shown for honesty (see text).

Method	structured stream	noise stream	has a "refuse" op?
Crystallon	10.9	**0.0**	yes (recurrence gate)
BIRCH (CF-tree)	5.0	299.8	no
online k-means (k=20)	20.0	20.0	no
DenStream (density)	1.0	1.0	own gate (degenerate here)

4.2 Closure-gated growth: refusing to represent noise

Proposition. Facing a pure-noise stream (every input novel, nothing recurring), Crystallon instantiates essentially no structure, because its growth is gated on recurrence. Assign-every-point methods have no "refuse to grow" operation: a CF-tree must place each point in a leaf, and k-means must assign each point to a centroid.

Setup. Two equal-length streams: (a) structured (5 concepts repeated) and (b) pure noise (each point a fresh random draw). We count instantiated structures: Crystallon's `n_cores`; `Birch(threshold=0.5)` subclusters; `MiniBatchKMeans(k=20)` non-empty clusters; and `DenStream` clusters. All baselines use official defaults.

Result. On noise, Crystallon grows nothing while assign-every-point baselines are forced to manufacture structure (Table 3, Figure 1 right). BIRCH opens a leaf for almost every isolated noise point (299.8); k-means returns its fixed budget of 20 regardless of whether real structure is present. Crystallon's 0.0 is a gate, not an inability to grow—the same model grows 10.9 structures on the structured stream.

Why this is structural, and its scope. "Add a leaf" (CF-tree) and "assign to the nearest centroid" (k-means) are constructive operations; neither architecture contains a "this is noise—do not instantiate" operation, so the behaviour cannot be removed by tuning. Crystallon's gate is its seeding rule (recurrence-then-crystallize). The honest scope: this is a property relative to *assign-every-point* methods, not to all online clustering. Density-based streaming (DenStream) has its own outlier gate and does not explode on noise (1.0). On this data, however, DenStream is degenerate—it also finds only 1.0 structure on the *structured* stream, failing to recover the 5 concepts—so Crystallon's recurrence gate is the one mechanism here that simultaneously rejects noise and recovers the structured concepts.

At MNIST scale, with official baselines. Both properties replicate beyond synthetic data (Figure 2). On full MNIST (20 seeds, with official `FAISS`, `river` DenStream, and `scikit-learn` BIRCH and PCA), Crystallon's independent unit count grows with the number of concepts (to ≈9–10) while a 5-dimensional PCA code's measured concept rank caps at 5 (the X1 property); and on a noise stream Crystallon instantiates 0 structures against BIRCH's 200 (12 vs. 87 on the structured stream—the X2 property). Scaling to MNIST required recalibrating the similarity length scale to the higher input dimension (a single per-dataset constant), which we note as an honest dependency rather than hide.

Claim ledger. We can state: (X1) fixed-d independent capacity is rank-capped at d while Crystallon's exceeds d; (X2) assign-every-point methods are architecturally forced to grow on noise while Crystallon's recurrence gate is not. We do not claim: that fixed-d classification accuracy is worse; that Crystallon's capacity is unbounded; or that all online clustering is forced to grow on noise. These are existence results—properties a fixed-carrier / assign-every-point architecture *cannot* have—and not performance comparisons.

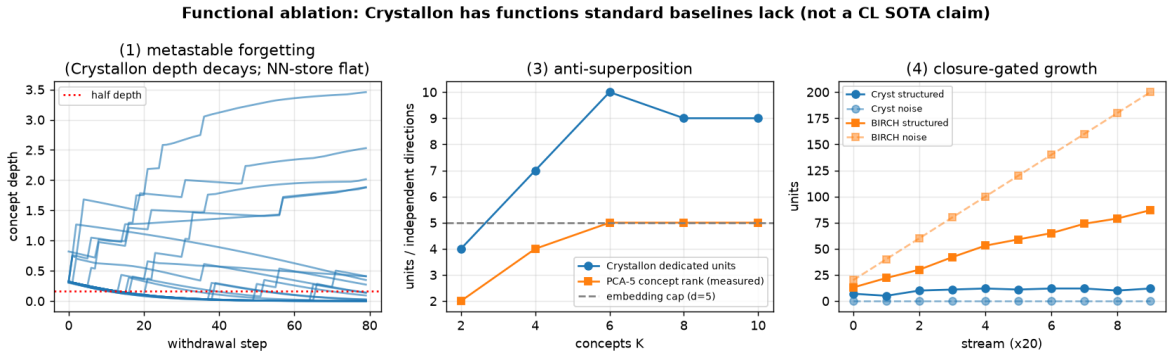

Figure 2: The functional properties replicate at MNIST scale (20 seeds) against official baselines (`FAISS`, `river` DenStream, `scikit-learn` BIRCH and PCA): the anti-superposition and closure-gated-growth properties of Section 4, and the relearning behaviour of Section 5, all hold on full MNIST.

5 Content-driven forgetting and the recall/recognition split

Crystallon's dissolution is metastable (Section 2); here we show what that buys behaviourally. The phenomena below are properties of the mechanism described on their own terms—an account of *how this memory forgets*, not an entry in the catastrophic-interference contest, which Section 8 explains is a different game on a different axis.

Forgetting is driven by content, not by time or use. Because dissolution is applied only inside an access (it is access-gated, not wall-clock), a unit that receives no accesses does not decay at all ($\Delta d = +0.000$); this is a logical consequence of the mechanism, not a finding. The non-trivial result appears under *equal* numbers of accesses: holding a crystallized concept and then presenting 80 further accesses of a *corroborating* (similar) stream deepens it ($\Delta d = +19.7$), whereas the same number of accesses of a *non-corroborating* (dissimilar or noise) stream dissolves it ($\Delta d = -0.33$ from an initial depth $\approx$0.33, i.e. essentially to zero). Same access count, same elapsed clock, opposite outcomes: the driver is the *content* of what recurs, not the amount of use or time (Figure 3, left; synthetic, D=64, 30 seeds).

The effect is graded by similarity. The deepen/dissolve outcome varies monotonically with the similarity between the held concept and the withdrawing stream. On MNIST (10 seeds), the depth change rises from -0.33 (pure noise) through -0.27 (low-similarity other class) to $+0.96$ (high-similarity other class) and $+5.66$ (same class; Figure 3, right); Spearman $\rho = 0.55$ ($p = 3.7 \times 10^{-87}$; digits agree, $\rho = 0.41$). This also delimits where metastable forgetting reads cleanly: on well-separated concepts (digits) a withdrawing stream is non-corroborating and depth decays, whereas on MNIST—whose classes are mutually similar—a withdrawing class often corroborates the held one and deepens it. The result is a sharp statement about *when* forgetting occurs; controlling *how much* is a separate question we do not take up here.

Recall/recognition dissociation and relearning savings. The two-timescale invariant $\tau_c < 1/\beta_0$ (Section 2) makes one intermediate state reachable and another not. After a long rest a concept reaches "structure present, confidence gone": its depth is intact (recognition, d=1.70) while its confidence has cooled to zero (recall failure, $c \to 0.00$). A single re-exposure restores confidence with savings ($c : 0.00 \to 0.27$) and—because seeding is spatial, not confidence-based—does *not* open a new trunk: the cooled-but-present concept is recognized as the same thing rather than learned afresh. The opposite ordering of timescales would instead produce confabulation (confidence without structure); the invariant rules it out. The same savings appear at MNIST scale (20 seeds, official baselines): re-acquisition takes 1.3 accesses against 2.7 for first acquisition, reusing the same unit 95% of the time.

Spacing and consolidation. The durability dynamics also reproduce the spacing effect. Massed repetition (no rest) yields a single deep-but-narrow unit (durability M=2.45, one node), whereas spaced

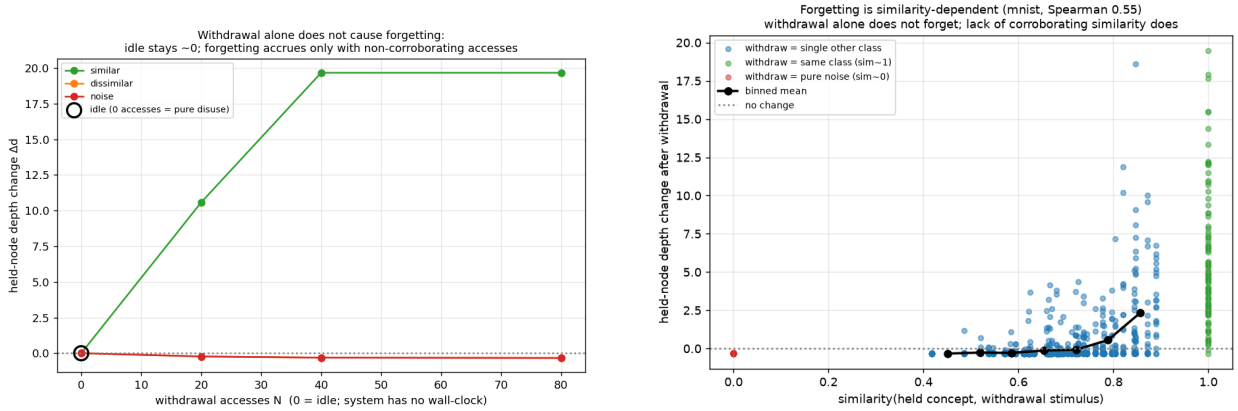

Figure 3: Forgetting is content-driven. **Left:** under equal numbers of accesses, a corroborating (similar) stream deepens a held concept while dissimilar and noise streams dissolve it, and an idle unit does not change—same access count, opposite outcomes. **Right:** the depth change varies monotonically with the similarity between the held concept and the withdrawing stream (MNIST; Spearman $\rho = 0.55$), from dissolution under noise to deepening under same-class input.

repetition—which the span term rewards—yields higher durability and breadth (M=2.97, three nodes); and a spaced core deepens review by review (1.10 $\rightarrow$ 1.34 $\rightarrow$ 2.11 $\rightarrow$ 2.44 over 6 to 40 reviews) as the Ostwald reflux consolidates depth onto the trunk, which remains the deepest node throughout.

6 Form precipitation: growing representational form without search

The mechanisms so far grow and remove units of a fixed representational *form*. As a proof of concept we show that the form itself can accrete from recurrence. A unit holds a parametric form (a circle) and can grow a correction term (an eccentricity, making it an ellipse) when residual harmonics recur across stimuli; the precipitation criterion is a cross-epoch mean significance, not a fit score searched over candidates.

Structure precipitates only from recurrence. On a structured stream—a stretch of fixed direction, repeatedly presented—the stretch atom precipitates in 12/12 runs, with an axis error of 0.003 rad; on a scrambled stream—the stretch direction re-randomized each epoch, destroying recurrence—it precipitates 0/12. The separation is large in the precipitation statistic itself: cross-epoch mean significance is 57–95 on the structured stream versus 0.3–1.2 on the scrambled one. Atoms come from recurrence in the stimulus, not from anything preset in the mechanism.

Recurrence, not search, selects the form. Given a stream mixing a recurrent order-2 deformation with a one-shot order-3 deformation, the recurrent type precipitates (12/12) and the one-shot type does not (0/12): both produce a signal every epoch, but only the recurrent one has a non-zero cross-epoch mean. The candidate order count stays bounded (two) and the decision is local to the unit. This is the sense in which growth can stand in for search—the stimulus, by recurring, supplies the constraint a hypothesis-space search would otherwise have to.

Reach of the proof of concept. We are precise about what this establishes: that a single recurrent deformation can be precipitated, and that recurrence—not search—selects among a small bounded set of types. It does not yet reach multiple interacting atoms, an enlarged form space, or structure the stimulus never repeats; that last item is not a gap but the principle itself, since there is no precipitation without recurrence. (One point of method: an early-transient latch initially produced 2/12 false positives on the scrambled stream; restoring the metastable dissolution the design already specifies—re-evaluating each epoch and dissolving unsupported atoms—returned it to 0/12 with the core unchanged, a small confirmation that the mechanism, not a tuned exception, is doing the work.)

Table 4: Cross-map feature binding: BOUND vs. UNBOUND separation z. Single-feature marginals are blind by construction (their distributions are identical across classes); a per-element joint primitive discriminates.

Statistic	Kind	z (BOUND vs. UNBOUND)
orientation concentration	single-feature marginal	0.4
auxiliary concentration	single-feature marginal	0.0
orientation $\lvert H_4 \rvert$	single-feature marginal	0.5
$(o-a)$ coherence	per-element joint	**128.8**
$(o+a)$ concentration	per-element joint	29.6

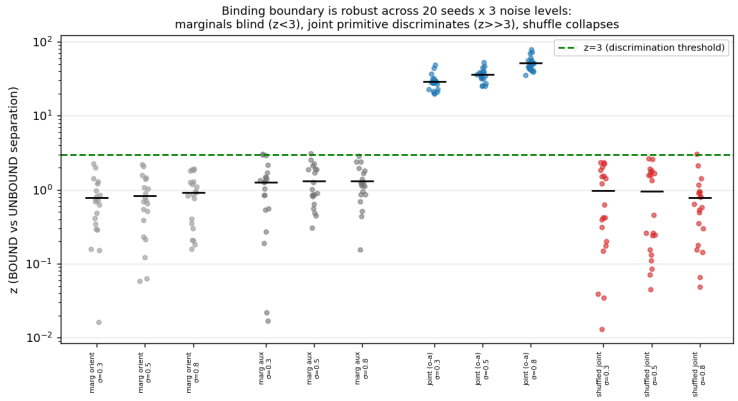

Figure 4: The binding boundary is robust, not a single-seed artifact. Across 20 seeds × three noise levels, the per-element joint statistic's BOUND-vs-UNBOUND separation (z) stays well above the marginal and shuffled-pairing null band; the two never overlap (a $z=3$ line separates them cleanly).

7 A characterized capability boundary: cross-map binding

A structural-properties paper should be as precise about what the mechanism cannot do as about what it can. Crystallon's translation invariance comes from precipitating *marginal* (single-feature) regularities, and this is, by construction, blind to *conjunctions*. We make that boundary exact.

Task. Each element carries two features, an orientation and an auxiliary value. In the BOUND class the two are correlated per element; in the UNBOUND class they are independent. Crucially, the single-feature *marginals* are identical across the two classes ($|\text{diff}| \approx 0$), so any statistic that separates BOUND from UNBOUND must be relational—it must pair the two features of the same element.

Result. Every single-feature marginal is blind to the distinction, while a single per-element joint primitive separates it decisively (Table 4). A hand-specified joint primitive yields a separation of $z = 128.8$, and shuffling the within-element pairing collapses it back toward UNBOUND ($z = 0.38$): the mechanism is intact—what is missing is a *basis* able to express the conjunction. The boundary is robust, not a single-seed artifact: across 20 seeds × three noise levels the joint statistic's separation (19.9–79.5) never falls below 6.5× the top of the marginal/shuffled null band (0.0–3.1; Figure 4).

Interpretation. The limit is one of *basis expressibility*, not of mechanism, and it lands precisely on the step of pairing the two features of a single element—a step marginal precipitation never performs. This is the computational echo of Feature Integration Theory (Treisman & Gelade, 1980): features register in parallel, separable maps, and binding them into objects requires a dedicated conjunction step (there, focal attention; its absence yields illusory conjunctions). It is also a clean, localized instance of the binding problem in neural networks (Greff et al., 2020). The boundary sharpens what Crystallon does and does not do: it can associate

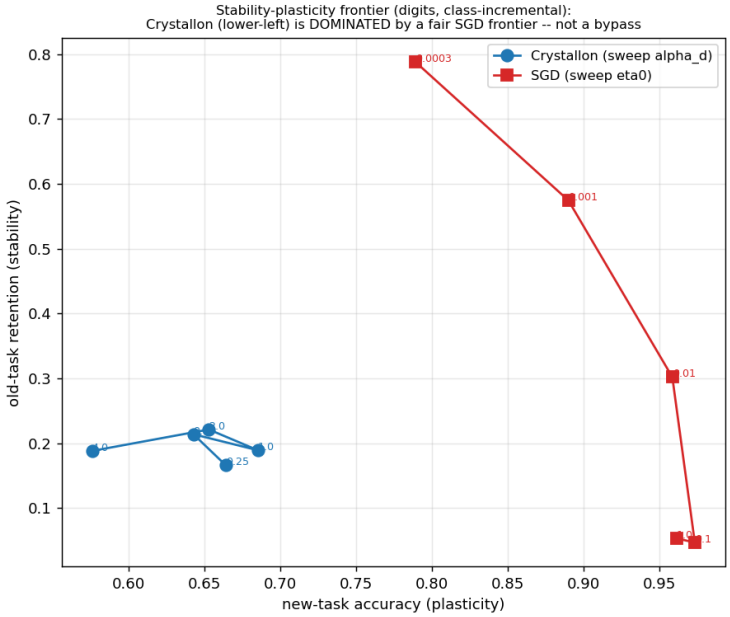

Figure 5: Measured as a classifier, Crystallon sits *inside* the stability–plasticity trade-off. A fairly configured SGD baseline traces a new-task-vs-old-task frontier that dominates Crystallon on both axes (digits, class-incremental, 10 seeds)—the expected result when a structure-discovery system is scored on discrimination, and the reason we do not evaluate Crystallon this way.

a feature and its label when they are *co-present* on the same element, but marginal growth alone cannot bind features carried on *separate* maps. We claim neither that binding is impossible (the hand-added primitive shows it is expressible) nor that we have solved it; whether the binding primitive can be *grown* rather than supplied is the same open grounding question raised in Section 8.

8 Scope: what Crystallon is, and is not

A precise account of scope is part of the contribution, not an apology for it. The boundaries below mark where Crystallon sits in the design space—a different place from a discriminative learner—rather than a shortfall against one; read as deficits, they mistake the system for something it is not trying to be.

A structure-discovery system, not a classifier. Scored as a classifier, Crystallon is weak—and it should be, because its units are local sub-group centres formed by unsupervised structure discovery, not class-optimal decision prototypes. The numbers make the category distinction concrete rather than embarrassing. On a standard class-incremental benchmark (digits, 10 seeds), a fairly configured SGD baseline (standardized inputs, equal epochs, swept learning rate) reaches $(\text{new}, \text{old}) = (0.79, 0.79)$ and dominates Crystallon's $(0.65, 0.20)$ on both axes (Figure 5); even i.i.d. and fully covered, Crystallon's readout reaches 0.66 against a 0.90 nearest-centroid ceiling. That gap is exactly what one expects when a structure-discovery system is asked to draw decision boundaries: the comparison is a category check, which Crystallon passes by failing in the predicted way, not a competition it sets out to win. (For completeness, MNIST-scale continual-learning accuracy is $\approx$0.18, on par with a FAISS store and below DenStream's 0.23.) Per-label pooling would lift the readout to $\sim$0.7–0.8 without changing that verdict, so we make no accuracy or continual-learning claim and direct the evaluation to where the contribution actually lies.

Bounded capacity and its costs. Crystallon retains on the order of 20 concepts simultaneously—far below its slot budget—and this ceiling is set by the depth dynamics, not by the number of slots or the budget parameter (varying either has no measurable effect). Nor can it simply be raised through the depth-gain

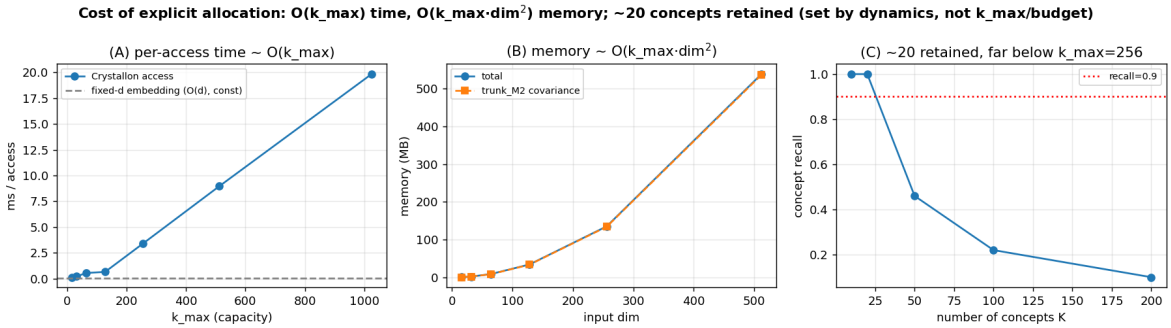

Figure 6: The costs of explicit allocation, stated rather than hidden. Per-access time grows with the slot count (left), memory with the square of the input dimension via per-unit covariance (centre), and the number of concepts retained simultaneously saturates near 20 (right)—a ceiling set by the depth dynamics, not by the slot count or budget.

α_d: simultaneous capacity is non-monotonic in α_d, peaking near $\alpha_d \approx 1$ and falling when pushed higher as competition evicts concepts; and under capacity pressure, raising α_d trades away old-concept retention ($0.73 \to 0.47 \to 0.00$ as $\alpha_d : 0.5 \to 1.0 \to 2.0$), placing the system squarely inside the stability–plasticity trade-off, consistent with its domination by SGD above. Bounded simultaneous retention with graceful forgetting is by design: a working memory, not an unbounded store. The costs are explicit (Figure 6). Per-access time is $O(k_{\max})$ from the similarity scan ($\sim$20 μs per slot); memory is $O(k_{\max} \cdot \dim^2)$, dominated by per-unit covariance, extrapolating to $\sim$1.3 GB at MNIST resolution and $\sim$19 GB at CIFAR resolution—high input dimension is the hard limit, to be mitigated by encoding to a lower-dimensional feature space. Against a fixed embedding's $O(d)$ cost (constant in the number of concepts), Crystallon trades higher cost for explicit, disentangled, inspectable allocation.

Controlled evidence and a staged basis. Our evidence is on synthetic data, scikit-learn digits, MNIST, and small image patches—deliberately controlled settings in which a structural property can be isolated and stated cleanly, rather than large-scale benchmarks (which we do not claim). At this stage the feature basis on which units grow is supplied rather than learned: Crystallon grows structure *on* a representation it does not yet build. We read this as the seam between the present stage and the next (Section 10), with form precipitation (Section 6) the first step across it, not as a concealed weakness.

9 Related work

Section 3 placed each *mechanism* against the specific prior work it descends from; here we situate Crystallon among the broader *research areas* its claims touch, and mark in each case how it relates rather than competes. None of this is a performance comparison: Crystallon is a structure-discovery system, and the contrast is one of design commitment.

Continual and lifelong learning. The problem of preserving old knowledge while acquiring new is the subject of a large literature, surveyed by Parisi et al. (2019). Regularization approaches such as elastic weight consolidation (Kirkpatrick et al., 2017) and metric-and-metricked incremental learning (Chaudhry et al., 2018) protect a *fixed* carrier by penalizing changes to important weights; replay and gradient-projection methods such as GEM (Lopez-Paz & Ranzato, 2017) do so by constraining updates against stored examples. A second family changes the architecture itself: progressive networks (Rusu et al., 2016) and dynamically expandable networks (Yoon et al., 2018) *grow* capacity as tasks arrive, the closest neighbours in spirit to a dynamic carrier. Crystallon differs from all of these on three axes at once: its growth is gradient-free and unsupervised, it requires no task boundaries (in the task-free spirit of Aljundi et al., 2019), and its units not only appear but *dissolve*, so the carrier shrinks as well as grows. We use this literature's evaluation tools—the average-accuracy and backward-transfer metrics of Lopez-Paz & Ranzato (2017) and the forgetting measure

of Chaudhry et al. (2018)—only as an appendix-level characterization, and are explicit (Section 8) that Crystallon is dominated by a fairly configured gradient baseline when scored this way. The point of contact is not accuracy but the shape of forgetting: where these methods treat forgetting as a failure to be prevented, Crystallon's is a graded, content-driven operation (Section 5).

Streaming and growing clustering. Crystallon's closure-gated growth (Section 4.2) is measured against online clusterers that add structure on the fly: the CF-tree of BIRCH (Zhang et al., 1996), density-based stream clustering with an explicit outlier buffer (DenStream; Cao et al., 2006), and topology-growing methods such as growing neural gas (Fritzke, 1995), alongside the threshold-gated creation of DP-means (Kulis & Jordan, 2012) already discussed in Section 3. These differ in whether they can *refuse* to instantiate structure. Assign-every-point methods (a CF-tree, k-means) cannot, and are forced to manufacture clusters from noise; growing neural gas inserts a node on a fixed schedule rather than on recurrence, so it too grows on noise; density-based methods have their own (density) gate, which is why we report DenStream honestly as a method with a different gate rather than as a foil. Crystallon's gate is recurrence: structure is instantiated only when content recurs, which is what yields the noise-rejection result.

Superposition, distributed codes, and sparse coding. Our capacity-beyond-the-embedding-dimension result (Section 4.1) is the converse of a well-studied phenomenon. Elhage et al. (2022) formalize how networks pack more features than they have dimensions into *superposition*, tolerating interference because features are sparse; classical sparse coding (Olshausen & Field, 1996) likewise represents many features in an overcomplete dictionary whose atoms share a fixed-dimensional space. That work shows fixed-dimensional codes *can* encode many concepts by sharing directions; our rank argument is simply the other face of the same coin—those shared directions are not linearly independent, so the number of *independently* representable concepts is still capped at the dimension. We take no position that superposition is undesirable (it is an efficient code); we observe only that a dynamic carrier, by allocating a dedicated unit per concept, occupies the opposite corner of that trade-off—inspectable separateness at the cost of compactness.

Prototype and metric learning. Crystallon's readout compares an input to a set of unit positions, which invites comparison to prototype methods. Prototypical networks (Snell et al., 2017) learn an embedding in which a single per-class mean is a near-optimal classifier; nearest-centroid classification is the non-learned version we use as a ceiling (Section 8). The diagnosis there is precisely that Crystallon's units are *not* class-optimal prototypes: they are unsupervised local sub-group centres, often several per class and off the class mean, so scored as a classifier the system is weak by construction. Metric-learning methods optimize the embedding so that means become discriminative; Crystallon performs no such optimization, which is why we evaluate it as a structure-discovery system rather than a classifier.

Memory consolidation in biology. Finally, the carrier's two-timescale dynamics connect to systems-level accounts of memory. Complementary learning systems theory (McClelland et al., 1995) separates a fast hippocampal learner from a slow neocortical one; cascade and metaplastic synapse models (Fusi et al., 2005; Benna & Fusi, 2016) give single synapses a spectrum of timescales; and the new theory of disuse (Bjork & Bjork, 1992) separates retrieval from storage strength. Crystallon's contribution relative to these is not a new biological claim but an architectural transposition: it makes the fast/slow split a per-unit *confidence*/*durability* pair that governs whether a discrete structural unit is recognized, recalled, or removed (Section 5), so that consolidation and forgetting act on the existence of carrier units rather than on values within a fixed substrate.

10 Discussion

The thread tying the paper together is one axis: *what the knowledge carrier is.* Symbolic and connectionist systems both fix it and learn by filling in values; Crystallon makes it dynamic and lets the birth, depth, and dissolution of its units be the learning. We have been deliberate about the size of this claim. We do not argue a new paradigm in the grand sense—the rhetoric of a "third" organizing principle is well populated (Section 3), and every mechanism we use descends from prior work. The orthogonality we claim is narrow

and structural: along the carrier axis no prior system adopts the full commitment, and that commitment makes available properties a fixed carrier provably cannot have (Section 4).

Read this way, the results indicate where a dynamic carrier earns its cost: open-world or lifelong settings where the number of concepts is unknown in advance (so any fixed dimension must eventually superpose), settings that must refuse to over-fit noise, and settings where forgetting should be a graded, content-driven operation rather than a failure mode. They equally indicate where it does not: tasks that are fundamentally discriminative and bounded, where gradient learning on a fixed carrier is both cheaper and better.

One boundary is deliberate, and worth naming as a direction rather than a deficit. Crystallon grows structure on a hand-given basis, and is grounded only as far as that basis is. This is the natural seam of a staged program: first establish, on a fixed basis, that a dynamic carrier yields the structural properties of Sections 4–7; then let the basis itself be grown. Section 6 is the first evidence that the second stage is reachable—representational form already precipitates from recurrence without search. Growing the full basis (symbol grounding), attaching a learned encoder front-end, and replacing the monotonic span-to-durability map with an independent maturation variable capable of the abrupt overlearning-to-permastore transition are the program's next stage, not obstacles to the present claims.

More broadly, a dynamic carrier is a way of putting a discrete commitment back into learning. A fixed carrier never has to declare that something new has appeared—it absorbs novelty by adjusting values—whereas Crystallon's seed is an all-or-none act: *this is new; instantiate it.* That commitment is what gives the structural properties their bite. A system that must decide whether to grow is one that can also refuse—hence closure-gated growth; a system whose units are individuated can hold them independent past an embedding dimension—hence capacity beyond d; a system that can mark a unit's importance at birth can let importance, not elapsed time, govern its fate—hence content-driven forgetting. We do not claim Crystallon settles the long-standing wish for learning systems that make discrete, inspectable commitments rather than only smooth interpolations; we claim it is a concrete, working point in that direction whose consequences can be measured rather than asserted.

11 Conclusion

The point of this paper is not a benchmark number but a characterization of what a dynamic carrier makes structurally possible: representational capacity beyond an embedding dimension, growth that refuses to instantiate structure on noise, content-driven forgetting with a recall/recognition dissociation, form that precipitates from recurrence without search, and a precisely located binding boundary. The carrier itself is dynamic—units are born, deepen, and dissolve, and those acts are the learning—and every mechanism that realizes it stands on established ground in adaptive resonance, synaptic consolidation, disuse theory, and program induction; the contribution is the commitment that unifies them, which no prior system makes in full, and the properties it provably yields. Crystallon is not, and does not try to be, a competitive classifier. Judged on the axis it is built for, it does what a fixed carrier cannot.

References

Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11254–11263, 2019. doi: 10.1109/CVPR.2019.01151.

Marcus K. Benna and Stefano Fusi. Computational principles of synaptic memory consolidation. *Nature Neuroscience*, 19(12):1697–1706, 2016. doi: 10.1038/nn.4401.

Robert A. Bjork and Elizabeth L. Bjork. A new theory of disuse and an old theory of stimulus fluctuation. In Alice F. Healy, Stephen M. Kosslyn, and Richard M. Shiffrin (eds.), *From Learning Processes to Cognitive Processes: Essays in Honor of William K. Estes*, volume 2, pp. 35–67. Erlbaum, Hillsdale, NJ, 1992.

Feng Cao, Martin Ester, Weining Qian, and Aoying Zhou. Density-based clustering over an evolving data stream with noise. In *Proceedings of the 2006 SIAM International Conference on Data Mining (SDM)*, pp. 328–339, 2006. doi: 10.1137/1.9781611972764.29.

Gail A. Carpenter and Stephen Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37(1):54–115, 1987.

Gail A. Carpenter and Stephen Grossberg. Adaptive resonance theory. In Claude Sammut and Geoffrey I. Webb (eds.), *Encyclopedia of Machine Learning and Data Mining*. Springer, 2016. doi: 10.1007/978-1-4899-7502-7_6-1.

Gail A. Carpenter, Stephen Grossberg, and David B. Rosen. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4(6):759–771, 1991.

Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 556–572, 2018. doi: 10.1007/978-3-030-01252-6_33.

Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 71–80, 2000. doi: 10.1145/347090.347107.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. arXiv:2209.10652.

Kevin Ellis, Lionel Wong, Maxwell Nye, Mathias Sablé-Meyer, Luc Cary, Lore Anaya Pozo, Luke Hewitt, Armando Solar-Lezama, and Joshua B. Tenenbaum. DreamCoder: Growing generalizable, interpretable knowledge with wake–sleep Bayesian program learning. *Philosophical Transactions of the Royal Society A*, 381(2251):20220050, 2023. doi: 10.1098/rsta.2022.0050.

Scott E. Fahlman and Christian Lebiere. The cascade-correlation learning architecture. In *Advances in Neural Information Processing Systems 2 (NeurIPS)*, pp. 524–532, 1990.

Bernd Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems 7 (NeurIPS)*, pp. 625–632, 1995.

Stefano Fusi, Patrick J. Drew, and L. F. Abbott. Cascade models of synaptically stored memories. *Neuron*, 45(4):599–611, 2005. doi: 10.1016/j.neuron.2005.02.001.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Brian Kulis and Michael I. Jordan. Revisiting k-means: New algorithms via Bayesian nonparametrics. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pp. 6467–6476, 2017.

James L. McClelland, Bruce L. McNaughton, and Randall C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995. doi: 10.1037/0033-295X.102.3.419.

Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996. doi: 10.1038/381607a0.

Alexander Ororbia. Continual competitive memory: A neural system for online task-free lifelong learning. *arXiv preprint arXiv:2106.13300*, 2021.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. doi: 10.1016/j.neunet.2019.01.012.

Joshua S. Rule, Steven T. Piantadosi, Andrew Cropper, Kevin Ellis, Maxwell Nye, and Joshua B. Tenenbaum. Symbolic metaprogram search improves learning efficiency and explains rule learning in humans. *Nature Communications*, 15:6847, 2024. doi: 10.1038/s41467-024-50966-x.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pp. 4077–4087, 2017.

Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12 (1):97–136, 1980.

Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: an efficient data clustering method for very large databases. *ACM SIGMOD Record*, 25(2):103–114, 1996. doi: 10.1145/235968.233324.

A Reproducibility and implementation

The core is a single zero-dependency NumPy module (`crystallon/core.py`) realizing the state, event loop, dynamics, and readout of Section 2. Across every experiment in this paper the core is *byte-identical* (`sha256` prefix `561ae6bf`): it is only ever observed, never re-tuned per experiment, so any reported behaviour is a property of the one model, not of an experiment-specific variant. The model's RNG is seeded (default `seed=0`), so single-run demonstrations are deterministic and reproducible. The default dimensionless parameters are listed in Table 5; the clock is the access count and there is no physical-time constant. For double-blind review the code is provided through an anonymized link; the public repository will be linked on de-anonymization.

Artifacts, seeds, and configurations. Table 6 maps every result in the paper to the script that produces it, the number of seeds, the key configuration, and the in-paper artifact. All baselines use official `scikit-learn`, `river`, or `FAISS` implementations at default settings (BIRCH `threshold=0.5`, `MiniBatchKMeans` k=20, `DenStream` defaults, `PCA`/`GaussianRandomProjection` at the stated d). Each figure's source file in `figures/` shares the name of its experiment; "1 (det.)" marks a deterministic single-run demonstration under the seeded RNG.

B Additional results

Conservation of the reflux. Because the Ostwald reflux adds depth to a trunk, we audited that it never manufactures depth. Recording total active depth before dissolution (s_0) and after reflux (s_2) over 200 continuous steps of a spaced run (196 of them with an active multi-branch reflux), we find $\max(s_2 - s_0) = +0.000$ (never positive) and $\text{mean}(s_2 - s_0) = -0.092$ (net dissipative, 18.44 total depth dissipated): the reflux redistributes and dissipates depth, it does not create it, so the consolidation reported in Section 5 is real rather than book-keeping.

Table 5: Default parameters (`crystallon/core.py`). All dimensionless; the construction-time invariant $\tau_c < 1/\beta_0$ ($2.5 < 12.5$) enforces two-timescale forgetting.

Symbol	Value	Role
ℓ	0.6	similarity length scale
cov_thresh / cov_star	0.30 / 0.25	responder threshold / referenced threshold
η_{ref} / η_{unref}	0.9 / 0.1	cross-check weight (referenced / unreferenced)
sim_seed / sim_trunk	0.50 / 0.80	spatial seed threshold / same-trunk threshold
$a^\star$	0.15	anchor floor (twig logic)
κ	0.45	corroboration jump
$c^\star$ / w_{gate}	0.40 / 0.08	crystallization gate centre / softness
α_d	0.55	deepening rate
β_0 / D_c	0.08 / 1.2	base dissolution rate / durability scale
τ_c	2.5	confidence cooldown (fast)
τ_R / ρ / λ_F	3.0 / 0.6 / 0.05	lability relaxation / drive / distortion anneal
M_{floor} / w_{surprise} / w_{span}	0.5 / 1.6 / 0.9	durability floor / surprise / span weights
span_unit	12.0	one rest-interval worth of span
ϕ	0.5	Ostwald reflux fraction
B	20.0	depth budget
θ / τ_r	0.25 / 0.12	readout threshold / softness
$k_{\max}$	256	slot capacity (with active mask)

Working-point plateau. A sweep of the geometric knobs shows the active-unit count is on a plateau, not a phase-transition edge: for $\ell \geq 0.6$ the count is flat at a structural floor of $\approx 3 \times (1 + \text{max_breadth})$ on a three-concept stream, the concepts never merge, and the seed threshold $a^\star$ has no effect (consistent with seeding being spatial).

Table 6: Reproducibility map: result → script, seeds, configuration, and artifact. "det." = deterministic single run (seeded). Dimensions are feature-space dimensions; D_{amb} is the synthetic ambient dimension.

Result	Script (`experiments/`)	Seeds	Configuration	Artifact
X1 capacity	`anti_superposition_existence.py`	8	synthetic, D_{amb}=64, d=5; PCA / RandProj / AE	Tab. 2, Fig. 1 (L)
X2 closure-gate	`closure_gated_existence.py`	8	structured (5 concepts) vs. noise; BIRCH / k-means / DenStream	Tab. 3, Fig. 1 (R)
MNIST functional	`bench_functional.py`	20	full MNIST (784-d), ℓ=0.8; FAISS / DenStream / BIRCH / PCA	Fig. 2
Forgetting (fork)	`forgetting_disuse_control.py`	30	synthetic, 64-d; idle / similar / dissimilar / noise	Fig. 3 (L)
Forgetting (similarity)	`forgetting_similarity.py`	10 (MNIST), 5 (digits)	MNIST ℓ=0.8; per-(held, other) pairs	Fig. 3 (R)
Recall/recognition	`recall_recognition.py`	1 (det.)	synthetic streaming; long rest + re-exposure	§5
Spacing / ripening	`massed_vs_spaced.py`, `spaced_ripening.py`	1 (det.)	synthetic; massed vs. spaced, 6–40 reviews	§5
Trunk-gating OR	`hierarchical_or.py`	1 (det.)	synthetic; trunk intact vs. dissolved	§2
Form precipitation	`stage4_atoms.py`	12 runs	structured vs. scrambled stream; order-2 vs. order-3	§6
Binding boundary	`stage8_object_relation.py`, `binding_robustness.py`	20×3	feature binding BOUND/UNBOUND; noise $\sigma \in \{.3, .5, .8\}$	Tab. 4, Fig. 4
Stability–plasticity	`stability_plasticity_frontier.py`	10	digits class-incremental (5×2); SGD swept η_0	Fig. 5
Classifier diagnosis	`classifier_readout_diagnosis.py`	5	digits; readout decomposition	§8
Cost scaling	`cost_scaling.py`	1 (det.)	synthetic, 64-d; sweep $k_{\max}$, dim, K	Fig. 6
Reflux conservation	`conservation_check.py`	1 (det.)	spaced run; 200 continuous steps audited	§B
Working-point plateau	`param_sweep.py`, `refine_working_point.py`	6–8 / point	3-concept stream; sweep ℓ, sim_seed, $a^\star$, cov_star	§B