

Energy Minima Are Not Enough: A Free-Energy Design Condition for Encoding Shortest Path in Stochastic Spiking Networks

Gary Engler

gary@engler-labs.com

Engler Labs

Article

Keywords: Spiking neural networks, neural sampling, Boltzmann distribution, winner-take-all networks, shortest path, combinatorial optimization, free energy, neuromorphic computing

Posted Date: October 16th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11001010/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Energy Minima Are Not Enough: A Free-Energy Design Condition for Encoding Shortest Path in Stochastic Spiking Networks

Gary R. Engler

Abstract

In a spiking network with Boltzmann stationary distribution, design is energy-function design. For the winner-take-all encoding of weighted shortest path, correctness needs two conditions, not one, and we give both. Ordering requires the length-bias coefficient $\lambda = 2b_k + D + c$ to vanish. Discrimination does not follow: equal-activity configurations the couplings cannot distinguish from a path leave only the continuity scale c to separate them. Required c is a ratio, not a depth law, reproduced over 1,027 instances at 8.6 % median error and within 16 % on withheld families. At the customary $c = 4$, counting microstates at $B = 2w_{\max}$, 13.4 % clear, one in forty at the size normally run. **And on most instances the path is not the cheapest configuration:** disjoint cycle sets carry an extra continuity pair, and raising c favors them. What excludes them is the terminal drive, outside the energy function.

Index Terms

Spiking neural networks, neural sampling, Boltzmann distribution, winner-take-all networks, shortest path, combinatorial optimization, free energy, neuromorphic computing.

I. INTRODUCTION

A CLASS of spiking neuron models admits an exact analysis that is unusual in computational neuroscience. If a neuron’s postsynaptic potential is rectangular and its duration matches its refractory period, the network’s state trajectory is a Markov chain whose stationary distribution is Boltzmann in a quadratic energy determined by the synaptic weights and biases [1]. The network does not converge to a fixed point; it samples, indefinitely, from a distribution known in closed form.

This turns network design into energy-function design. To solve a problem with such a network one encodes it so that its solution is the cheapest configuration, and the designer’s task becomes the encoding rather than the dynamics.

We are careful with that sentence from the outset, because for this construction it is not quite true and the gap is structural rather than pedantic. What Section IV establishes is that the least-weight path is the minimum *of the path sector*. It is not the global minimum: Proposition 3 shows the all-silent configuration has energy identically zero, which at the parameterization the construction requires, and at the terminal bias adopted here, is strictly below every path configuration. **The shortest path is a conditional minimizer, and what makes the condition hold is the additive drive on the two terminals** — which is why that drive is treated here as a structural component of the design rather than an implementation convenience. Jonke, Habenschuss and Maass developed this into a constructive method built from winner-take-all (WTA) motifs and demonstrated it on constraint satisfaction problems [2], in a line of work covering probabilistic inference [3] and cortical microcircuit models [4]. The

G. R. Engler is with Engler Labs, United States. E-mail: gary@engler-labs.com.

Submitted August 2026.

Prior publication. This work develops material from the author’s doctoral dissertation, *Stochastic Neural Algorithms* (Stevens Institute of Technology, 2018), which is the earlier statement of the construction studied here and the principal object of the corrections below. The construction of Section III follows that source. The design conditions, the free-energy argument, the enumeration campaign, the ratio law and the $O(|E|)$ formulation are new; Section IV-F and Table IV set out the four quantities where the accounting here differs from it. No part of this work has been published elsewhere or posted as a preprint.

approach suits neuromorphic hardware, where stochasticity is abundant and cheap [5], and it rests on the position that noise in neural systems is a computational resource rather than a nuisance [6], [7].

This paper studies the shortest-path member of that family. A weighted graph is compiled into one WTA cluster per vertex, with three classes of coupling arranged so that a configuration encoding a source–target path has energy equal, up to an additive constant, to that path’s total weight [8]. Shortest path is a good vehicle for the general question. It is polynomially solvable, so the interest is not in competing with Dijkstra’s algorithm but in understanding what a sampling network does with a problem whose answer is known; its solutions carry a graded quality measure, so partial success is measurable; and it is the computational core of spatial navigation, a domain where the biological substrate is comparatively well characterized [9], [10].

A. *The question this paper asks, and the one it does not*

An energy-based construction offers a guarantee about a *minimum*. What an application needs is that the network *reach* that minimum within a usable time, and that the configuration it reaches *be* the answer rather than merely contain it. These are three different properties, and the distance between them is large enough that they should not be run together.

This paper is about the first: what an encoding must satisfy for the answer to be the minimum at all. That turns out to be two conditions rather than one, and the second is not implied by the first. **The other two — whether the sampler reaches the minimum, and what a fair measurement of “reaching” would even be — are outside its scope,** and we say so rather than leaving a reader to infer it.

Separating them is not merely editorial. The conditions derived here are statements about an energy function and are checkable against a compiled network by enumeration; they are true or false independently of any sampler, any budget and any readout. Questions about reaching are statements about a stochastic process and are answerable only up to a sampling budget. Conflating the two is how an energy-based construction comes to be described as working when what has been established is that its minimum is in the right place.

B. *How the analysis is grounded*

Several of this paper’s results concern quantities that are easy to get wrong on paper and easy to check on a compiled network: the exact combinatorics of a path configuration, and the energy that follows from them. For each we state whether it was verified against a compiled network or derived algebraically, and mark the derivations not checked numerically; Section IV carries measured tables alongside its propositions for that reason.

One term needs fixing before it is used throughout. “Measured on a compiled network” means a quantity *enumerated or evaluated exactly* on a compiled instance — the energy of a hand-built configuration, an exhaustive path count, a matching count by dynamic program — and not a statistic of a sampled trajectory. Those quantities are deterministic: the same instance returns the same value, and a disagreement is an error rather than a fluctuation. **Four results below are exceptions, and each is named as sampled where it appears:** the cooling table of Section IV-D, the (c, λ) enrichment sweep and the mass-ratio attempt of Section V-F, and the timings of Section VI-C. Everything else in this paper is an enumeration.

The discipline has a specific origin. A re-implementation of the construction, checked only against itself, cannot detect a premise the two share — and the re-implementation used for much of the empirical work omits the auxiliary source and target vertices of Section III. That omission changes the energy of a path configuration, and no amount of internal consistency reveals it — the step is small enough to be lost in any re-implementation, and its consequences are not. **It is also the reason Section IV-E exists as a section rather than as a footnote:** some of the measurements reported below were taken on that variant, and which results transfer to the construction and which do not is a question we answer explicitly rather than leave to the reader.

C. *What is new here, and how it relates to the earlier statement*

Because this construction has been described before [8], it is worth separating what is new from what is a correction, rather than leaving a reader to work it out from Section IV-F.

New here: the decomposition of a path configuration’s energy into weight and hop-count terms and the identification of λ as the coefficient that decides between them; the all-silent constraint and the structural role it gives the

terminal drive; the free-energy argument and the discrimination condition, which has no counterpart in the earlier statement; the discrimination margin in closed form together with its verification and its threshold enumerated at the matched competitor size; the demonstration of what the augmentation buys, by measurement on a network compiled without it; and the $O(|E|)$ formulation.

Where the accounting here differs from the earlier statement. Four quantities are revised, and it is worth being precise about what that does and does not touch. Three are factors of two in a count or a coefficient: the active β count, which is $L + 2$; the active α count, which is $L + 1$; and a global factor in E_{SPC} . **None of the three disturbs the shortest-path property**, which is sound at $\lambda = 0$ on the construction as published — the E_{SPC} factor rescales one energy component uniformly and moves only the regime boundary of Corollary 1. The fourth is of a different kind. The bias condition is obtained there by imposing a zero at one particular depth, which yields a depth-independent bias but leaves a residual linear in hop count at every other depth; the resulting *parameterization* sits at $\lambda = D/2 - 1$, and is length-neutral only at $D = 2$. **That is a statement about a parameter choice, not about the construction.** Section IV-F sets all four out in detail, for readers working from those sources.

D. Contributions and organization

Section III states the compiled network exactly — its clusters, its three coupling classes and its augmentation — and fixes the summation convention the original derivation leaves ambiguous, a choice that changes two of the counts everything downstream depends on.

Section IV derives the energy of a path configuration in closed form, exhibits the decomposition into weight and hop-count terms, and gives the length-neutrality condition $\lambda = 0$ and the bias $b_k = -(D + c)/2$ it implies. It also shows that the all-silent configuration has energy identically zero, so it competes with every path and constrains λ from below — which is what gives the terminal drive a structural role rather than an incidental one. Section IV-E then exhibits, by measurement on a compiled network, what omitting the augmentation costs: terminal edges charged once and interior edges twice, so paths of equal weight acquire different energies and the ordering can invert outright.

Section V gives the second condition. We exhibit *scatter* configurations matching a path in every quantity the global couplings observe, show the cross-cluster strength cancels exactly between the two families, give the discrimination margin in closed form and verify it against compiled networks, then enumerate the threshold at the matched competitor size over 1,027 instances in four graph families. Required c turns out not to be a depth law but a ratio, and the law that does hold transfers to topologies and weight conventions withheld from its fit — while at the operating point this work uses, clearing the condition is — counting microstates — the exception rather than the rule.

Section VI gives the $O(|E|)$ formulation that makes these sizes reachable, and Section VII draws out what the two conditions imply and states what this paper deliberately does not establish.

II. BACKGROUND

A. Stochastic spiking neurons and neural sampling

Spiking neurons differ qualitatively from rate-based units, because the quantity carrying information is the timing of discrete events rather than a continuous activation [11]. The line of work this paper belongs to adds a second departure: the spiking is *stochastic*, and the stochasticity is treated not as a nuisance to be averaged away but as the mechanism that does the computing.

The model is spare. Consider n neurons, and let $x_l(t) \in \{0, 1\}$ indicate that neuron l emitted a spike in the window $(t - \tau, t]$. Each neuron carries a bias b_k and is coupled to the others through a symmetric weight matrix w_{kl} , giving the membrane potential

$$u_k(t) = b_k + \sum_l w_{kl} x_l(t). \quad (1)$$

Two modeling choices then do all the work. The postsynaptic potential is rectangular of duration τ , so a spike contributes w_{kl} for exactly τ units of time and nothing thereafter; and the refractory period equals that same τ , so a neuron is unavailable for exactly as long as its own spike is visible to its neighbors. Then x_l means both “this neuron is refractory” and “this neuron’s spike is currently being felt”, so $x = (x_1, \dots, x_n)$ describes the network completely and its trajectory is a Markov chain.

Buesing *et al.* [1] showed that if, in addition, the firing mechanism satisfies the *neural computability condition*

$$\frac{P(\text{on})}{P(\text{off})} = \exp(u_k), \quad (2)$$

then that Markov chain has a stationary distribution available in closed form,

$$P(X = x) = \frac{\exp(-E_N(x))}{\sum_y \exp(-E_N(y))}, \quad (3)$$

$$E_N(x) = -\sum_k b_k x_k - \frac{1}{2} \sum_k \sum_l x_k x_l w_{kl}.$$

The distribution is Boltzmann in a quadratic energy whose coefficients are exactly the biases and synaptic weights the designer chose, and nothing is approximated: (3) is the exact invariant law of the chain defined by (1) and (2).

The design target is therefore a distribution rather than an attractor. A deterministic recurrent network relaxes and stops, and one reads out a fixed point; a network satisfying (2) never stops. The properties one may ask for are accordingly which configuration carries the most mass, how much, and how long a run must be before that mass is realized as time spent — the questions Markov chain Monte Carlo has always faced [12], with a spiking circuit as the sampler.

B. Energy-function design as a programme

Because the energy in (3) is quadratic in the state and linear in the parameters, a designer who can write a problem's objective as a quadratic form in binary variables can read the required weights and biases straight off it. This inverts the usual relation between architecture and behavior: instead of choosing dynamics and asking what they compute, one chooses what should be cheap and lets the dynamics follow. Solving a combinatorial problem then means encoding it so that its solution is the energy minimum, in the expectation that the network visits that configuration more often than any other.

Jonke, Habenschuss and Maass turned this into a constructive method [2]. Their building block is the winner-take-all (WTA) cluster: neurons coupled by strong mutual inhibition, so that configurations with more than one active member are heavily penalized and the cluster behaves as a categorical variable taking one value at a time. A problem is compiled by assigning one cluster per variable and one coupling per constraint, with the coupling strength set by what violating that constraint should cost. The same framework supports probabilistic inference in general graphical models [3] and models of cortical microcircuits [4]: one mechanism, read three ways.

Two motivations sit behind the programme. The first is hardware: neuromorphic substrates supply spiking and device-level stochasticity natively while paying for dense connectivity and high-precision arithmetic [5], so a method whose resource is noise and whose couplings are sparse suits them well. The second is the position that noise in neural systems is a computational resource and that sampling is a plausible account of what cortical circuits do with it [6], [7] — a position on which the results below put a quantitative boundary, since the stochasticity making the target reachable in principle also makes it rare in practice.

C. Neural approaches to graph optimization

Energy-based combinatorial optimization predates the spiking formulation by three decades. Hopfield showed that a symmetrically coupled recurrent network descends a Lyapunov energy [13], and Hopfield and Tank encoded the traveling salesman problem so that valid tours occupied low-energy configurations [14], launching a substantial literature reviewed by Smith [15] whose shortest-path member is due to Bousoño-Calzón and Figueiras-Vidal [16].

That literature also produced a methodological lesson. Wilson and Pawley re-examined the traveling-salesman network and found its apparent behavior depended on the quantity one chose to measure: it settled predominantly on configurations that were not tours at all [17]. The durable reading is a measurement one — what must be reported is how often the network's state *is* the object the construction computes, and that is not implied by any statement about the energy minimum. That is not the question this paper answers, and we note it here only to mark where the present analysis stops.

Four spiking approaches to path problems locate the present construction. Rueckert *et al.* [18] sample from a distribution over trajectories — the nearest neighbor, differing in what is distributed, trajectories of a dynamical

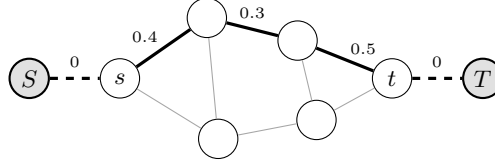


Fig. 1. The augmentation. Two auxiliary vertices S and T join the source and target by zero-weight edges. Their purpose is not bookkeeping: they make s and t ordinary *traversed* vertices, so every edge of the highlighted $L = 3$ optimum is charged exactly twice by the continuity couplings and the weight-dependent energy becomes proportional to the path's total weight. The configuration spans $L + 1 = 4$ traversals and $L + 2 = 5$ agreed edges (Lemma 1). Section IV-E shows what a network compiled without them does instead.

system there against induced subgraphs of a fixed graph here. Krichmar [19] plans with axonal delays, the arrival order of spike wavefronts carrying the computation, with no energy function and a first-arrival readout. Friedrich and Lengyel [20] derive goal-directed action selection by probabilistic inference, targeting a policy rather than a combinatorial object, and Zhang *et al.* [21] treat combinatorial optimization with spiking P systems. One property distinguishes the present construction from all four: its stationary distribution is Boltzmann in an energy the designer writes down, so the object of study is an encoding rather than a heuristic.

D. Notation

The instance is a weighted simple undirected graph $G = (V, E, \varphi)$ with $\varphi : E \rightarrow \mathbb{R}_{>0}$, $N = |V|$ vertices and distinguished $s, t \in V$. Every other symbol is defined where it is introduced; two conventions deserve a word here. The *depth* L of a simple s - t path P is its hop count in G , so P spans $L + 1$ vertices of G , of which $L - 1$ are interior in G ; L unqualified is the depth of the optimal path, the s - t path of least total weight. And B is a normalizing bound on edge weights, chosen so that $\varphi(e)/B < 1$ for every $e \in E$, making

$$\tilde{E}(P) = \frac{1}{B} \sum_{e \in P} \varphi(e) \quad (4)$$

the normalized weight of a path. We take $B = 2w_{\max}$ throughout, and that choice is a design decision rather than a normalization. **The ordering condition is invariant to it** — Corollary 1(iii) holds for any admissible value, so B affects only the rate at which depth trades against weight when $\lambda \neq 0$, and hence where the regime boundaries of that corollary sit. **The discrimination condition of Section V is not invariant to it.** B enters that condition's margin directly, and admissibility alone ($B > w_{\max}$) admits values at which the margin is negative and no c suffices; the bound (28) below shows that $B = 2w_{\max}$ is the smallest constant multiple of $w_{\max}$ that guarantees a positive margin at every depth, and Section VII-B gives the cost of raising it further. Every required- c figure in this paper is therefore conditional on $B = 2w_{\max}$, and we state the convention here because those values cannot be reconstructed without it. A symbol with an argument — as in $N(x)$, $M(x)$ below — is a function of the configuration, not the graph size N .

III. THE CONSTRUCTION

This section presents the compilation of a weighted graph into a spiking network in the form a reader would need to implement it, following [8] and making explicit the quantities Section IV depends on.

A. Network structure

a) *What a cluster represents.*: The encoding is easiest to follow if one point is fixed at the outset, because everything else follows from it and because it is the point most easily lost in paraphrase.

A cluster does not represent a *vertex*. It represents a *traversal*, i.e. the passage of a path *through* that vertex.

Figure 1 shows the augmentation this reading forces, and Figure 2 what it means for one cluster.

A path passing through v_i uses exactly two of that vertex's incident edges, one to arrive and one to leave; a vertex off the path uses none. What has to be encoded at v_i is therefore not its identity but its *use*: which unordered pair of incident edges the path occupies there. That is why a cluster carries two *slots*, each choosing at most one incident

edge. One slot alone asserts nothing about a path; a pair of occupied slots asserts a traversal. As Section III-B makes precise, the coupling that rewards a traversal is exactly the coupling that spans the two slots, so **an active coupling of that class is the network’s record that a path passes through that vertex**, and the weights of both edges of the traversal enter its strength together.

Reading clusters as vertices rather than as traversals makes the next step look like a convenience for labelling the endpoints. It is not.

b) Step one: augmentation.: A path does not pass *through* its own source. It *begins* there, using a single incident edge. Under the reading above, cluster s therefore has nothing to put in its second slot, is not a well-formed traversal, and records no passage — and the same holds at t . The construction removes the exception rather than accommodating it. The graph is augmented with two new vertices, a source S and a target T , joined respectively to s and to t by edges of weight zero, and to nothing else.

The two purposes the source states for this step [8] are now both visible. It distinguishes the source and target vertices without affecting the length of any path, because the added edges carry no weight. And, given the form of the couplings introduced below, it is what allows one component of the energy function “to decompose to the shortest path problem”: with S in place, cluster s has a second edge to occupy its free slot, becomes a genuine traversal — in by way of S , out by way of the first real hop — and contributes to the energy on the same footing as every vertex between the endpoints. Section IV makes that quantitative. In one line: every vertex of the augmented path is a traversal, every traversal contributes one coupling carrying *both* of its edges, and therefore every real edge of the path is charged exactly twice — which is what makes the weight-dependent component of the energy proportional to the path’s total weight and to nothing else.

c) Step two: clusters.: Each vertex v_i of the augmented graph becomes a WTA *cluster* of $2 \deg(i)$ neurons in two slots, each slot holding one neuron per neighbor of v_i . Write $n[i, c, j]$ for the neuron in cluster i , slot $c \in \{0, 1\}$, representing the incident edge (v_i, v_j) , and w_j for $\varphi((v_i, v_j))$ when the cluster is clear.

The two added vertices are the exception, and for the same reason: they are not traversed either. S and T are *single neurons*, each attached to the neuron representing it in each slot of its respective cluster, and to nothing else [8]. They carry no couplings among themselves, having none to carry. Everything else about s and t is ordinary: their clusters are built exactly as any other vertex’s, they hold one neuron per incident edge — now including the zero-weight edge to S or T — and a path occupies *both* of their slots, one with the terminal zero-weight edge and one with the first, or last, real edge. After augmentation the encoded object is the augmented path $S, s, \dots, t, T$, in which s and t are interior.

Summing $2 \deg(i)$ over the vertices counts every edge four times, so with the two terminal neurons the network has

$$n = 4|E| + 2 \quad (5)$$

neurons, as reported in [8]; whether $|E|$ there is the original or the augmented edge set is not settled by the source, and the implementation used for the measurements below allocates $4|E|$ neurons over the *original* edge set with no additive term, which is worth checking against before any count is carried across. An active neuron $n[i, c, j]$ is the assertion “vertex i uses its incident edge to j ”, so a configuration x is a set of such assertions — which is to say, an induced subgraph. The design problem is to make the assertions cohere into a path, and to make the cheapest coherent set of assertions the one describing the shortest path.

Two further properties matter later. First, slots are not oriented: they are the two edges a path uses at a vertex, not an “incoming” and an “outgoing” edge, and which one a given edge occupies is arbitrary — so every cross-cluster coupling must connect all ordered slot pairs, and every readout must be slot-agnostic.

Second, S and T are the only neurons whose participation is given rather than inferred. They are the boundary condition, and the construction supplies it by driving them: the dynamics are observed after “the single neurons representing the original nodes S and T are activated, representing that they fire regularly every τ ms” [8], and after the inhibitory reset described in the same source, sampling restarts from S and T active. The drive is delivered to S and T themselves and reaches the source and target clusters through the cross-cluster couplings; it is an added input rather than a fixed state, so every neuron involved remains a stochastic unit. Two disciplines are natural — delivering the input at every step, or periodically. As Section IV shows, this drive also excludes a configuration that would otherwise compete with every path.

Being single neurons rather than clusters, S and T need a bias of their own, and nothing in the construction fixes it. **We take** $b_S = b_T = b_k$, so that every neuron active in a path configuration is charged alike and the bias

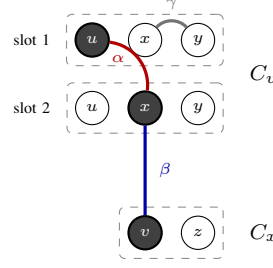


Fig. 2. A cluster is a traversal, not a vertex. Filled neurons are active: cluster C_v asserts “the path enters v from u and leaves toward x ”. The α coupling joins the two slots and is what charges both incident edge weights; β rewards the two endpoints of an edge for agreeing it is traversed; γ is winner-take-all inhibition within a slot. **Only α separates a path from an equal-activity scatter configuration** — β and γ cancel exactly between the two families, by the cancellation (23), which is why the discrimination condition constrains the continuity scale c and nothing else.

count in (13) below is the full $2(L + 2)$. The choice deserves a paragraph rather than a clause, because it is the second quantity in this construction that reads as bookkeeping and is not.

What b_S does *not* affect is the length-bias coefficient. S and T are active exactly once in every path configuration, so b_S contributes the same amount to every path irrespective of depth or weight; it cancels from every comparison among paths, and it does not enter λ , which is a per-hop quantity. Definition 2, Corollary 1, the bias condition (17) and the tolerance of Section IV-C are therefore identical under either assignment, as is the discrimination margin of Section V, which Section V-A shows to be free of every bias term.

What it does affect is the affine constant of Proposition 1 — and that constant is precisely what a path configuration’s energy is measured against when the competitor is the empty network. Taking $b_S = 0$ instead drops two neurons from the bias count and gives $E_{NC}(x_P) = -\lambda(L + 1) - D$ in place of (15); Proposition 3 states the consequence, and it is not a small one. **The bias carried by two auxiliary neurons decides whether silence competes with the path sector at all**, which is more than one expects a spare parameter to carry.

One consideration points the other way, and it is dynamical rather than energetic. With $b_S = 0$ a terminal whose cluster is occupied receives $2D$ through its couplings and fires at every opportunity — once per refractory period, which is what the source prescribes [8] — whereas at $b_S = b_k$ an *unanchored* terminal waits some 55τ before firing unaided, and the unanchored state is not incidental, since it is what the network occupies immediately after every inhibitory reset. **That is a statement about how the sampler behaves rather than about what the energy function ranks**, and this paper is about the latter, so we record the consideration and adopt $b_S = b_k$ regardless. A design targeting the dynamics rather than the ranking may reasonably choose otherwise, and Proposition 3 says exactly what that choice costs.

Note what the drive does *not* do. It does not select the first hop. Exciting S makes cluster s assert its terminal edge, which fills one slot; the second slot is then filled by whichever real edge the within-cluster couplings favor, and those couplings are weighted by edge weight. The boundary condition anchors the path without choosing it.

B. The three coupling classes

Three classes of coupling turn the cluster array into a path encoder. Each exists to make one family of configurations cheap and its complement expensive, and they are easiest to read in that order: what should a path configuration have that others do not?

a) α — *continuity*.: This is the class that carries the reading of Section III-A. A traversal of v_i occupies both of that cluster’s slots with two *different* incident edges, and the α class rewards exactly that configuration, coupling neurons in opposite slots of the same cluster with different adjacency indices at weight

$$f = 1 - \frac{w_j + w_k}{B}, \quad (6)$$

where w_j and w_k are the weights of the two edges the coupled neurons represent. An active α pair is the network’s record that a path traverses that vertex, and since f decreases in the two edge weights, traversing by way of heavy edges earns less than traversing by way of light ones. This is where the objective enters the circuit: the α weights are the only place edge weights appear. The clusters of s and t are no exception — they carry α couplings like

any other cluster, and the pair a path activates there involves a zero-weight terminal edge, so (6) charges it for the single real edge alone. Scaling the whole of (6) by a factor c leaves the structure untouched and is the design's free parameter; $c = 1$ is the base construction, and Section V shows what c has to do.

b) β — consistency.: An edge is used by a path only if both endpoints say so, and since each endpoint holds its own opinion in its own cluster, nothing so far prevents cluster i asserting edge (i, j) while cluster j asserts something else. The β class is one coupling per edge of the augmented graph — the two zero-weight edges included — of weight D , joining the neurons in the two endpoint clusters that represent each other across all ordered slot pairs. An active β pair means both endpoints agree the edge is used; D is the reward for that agreement, and hence the price of a half-asserted edge.

c) γ — winner-take-all inhibition.: Finally, a slot must hold at most one edge and the two slots of a cluster must hold *different* edges. The γ class supplies both: inhibition of strength I between all pairs within a slot, and between same-index neurons in opposite slots. The first enforces one winner per slot, so a cluster cannot claim three or more incident edges; the second prevents one edge occupying both slots and masquerading as a traversal. Following the WTA convention $|I| \gg |D|$, so violating the categorical structure is never worth the consistency reward it might buy.

C. Energy decomposition

Substituting the three coupling classes into (3) gives an energy separating into two parts with distinct interpretations,

$$E_N(x) = E_{\text{NC}}(x) + E_{\text{SPC}}(x), \quad (7)$$

the first measuring how well the configuration respects the network's structural constraints and the second the weight of what it encodes. Writing $|\alpha|$ for the number of active α pairs, and letting $\sum \sum_\alpha$, $\sum \sum_\beta$, $\sum \sum_\gamma$ range over active *ordered* pairs of the respective class,

$$\begin{aligned} E_{\text{NC}}(x) &= M(x) - N(x), \\ N(x) &= \sum_k b_k x_k + \frac{1}{2} \sum_\beta \sum_l x_k x_l D + |\alpha|, \end{aligned} \quad (8)$$

$$\begin{aligned} M(x) &= \sum_\gamma \sum_l x_k x_l I, \\ E_{\text{SPC}}(x) &= \frac{1}{2B} \sum_\alpha \sum_l x_k x_l (w_j + w_k). \end{aligned} \quad (9)$$

The summation convention is worth stating once, because every count in Section IV depends on it. The double sums run over ordered pairs, and the factor $\frac{1}{2}$ in front of each of them is spent converting that ordered count to an unordered one; this is what makes the constant part of (6) contribute exactly $|\alpha|$ to $N(x)$, one unit per active α pair, as written in (8). The same $\frac{1}{2}$ is already accounted for in (9), so no further halving of the α weight sum occurs downstream.

The *network-consistency* component E_{NC} collects everything independent of edge weights: $N(x)$ accumulates the reward for being a well-formed structure — biases, agreed edges, traversed vertices — while $M(x)$ accumulates the γ penalty, nonzero exactly when some slot has more than one winner. The *shortest-path-cost* component E_{SPC} is the weight-dependent part of the α contribution, and on a path configuration it is a normalized sum of the weights traversed — this is the component the augmentation was introduced to make decompose cleanly. The intended reading is that E_{NC} decides *whether* a configuration is path-shaped and E_{SPC} decides *which* path it is; Section IV makes both halves quantitative and gives the parameter conditions under which the separation holds.

D. Realizing the chain in discrete time

The analysis above is continuous-time, and (3) is conditional on two properties of the neuron: the neural computability condition (2), and the equality of postsynaptic-potential duration and refractory period. A discrete-time simulation does not inherit these; it must be built to preserve them. Two requirements follow, and an implementation meeting both samples from (3) while one missing either samples from something else.

a) *Requirement 1: the per-step firing probability.*: The neuron model specifies firing as a *rate*, $p_k = \tau^{-1} \exp(u_k)$ when not refractory, whereas a discrete-time sampler needs a per-step probability — and that is not the rate itself. Under a deterministic τ -step active window, a neuron firing with per-step probability q when available spends τ steps active for every $(1 - q)/q$ steps silent in expectation, so its stationary odds of being on are $\tau q/(1 - q)$. Imposing (2) on those odds gives

$$q_k = \frac{\exp(u_k)}{\tau + \exp(u_k)} = \sigma(u_k - \ln \tau), \quad (10)$$

the logistic function of the membrane potential offset by $\ln \tau$, which lies in $[0, 1]$ by construction — a property neither $p_k dt$ nor the exponential-clock survival probability $1 - \exp(-p_k dt)$ has, both of which also bias every neuron toward silence.

b) *Requirement 2: refractory bookkeeping order.*: The refractory counter must be decremented at the top of the step, before firing is resolved. Decrementing it afterwards leaves each neuron active for $\tau - 1$ steps rather than τ , and that equality of duration and refractory period is exactly the hypothesis making the stationary distribution Boltzmann rather than merely stationary.

c) *Validation.*: Both requirements are checkable directly, because an eight-neuron network has only 256 configurations and its Boltzmann distribution can be enumerated exactly. An implementation satisfying (10) and decrementing at the top of the step reaches a total variation distance to the enumerated distribution of 0.0558 (KL 0.0127) at $\tau = 3$, 0.037 at $\tau = 5$, and 0.0225 (KL 0.0023) at $\tau = 10$. The residual is the discretisation itself and shrinks with τ toward the continuous-time limit the analysis assumes; we adopt $TV \approx 0.02$ at $\tau = 10$ as the regression threshold every later configuration of the sampler is held to. For comparison, an implementation taking the rate as a per-step probability and decrementing after firing sits at $TV = 0.235$ on the same enumeration — close enough to produce plausible-looking paths, far enough that no statement about the stationary distribution transfers to it.

E. Reading the network state

The last piece is the map from a configuration back to a graph object. Since a configuration is a set of assertions “vertex i uses its incident edge to j ”, and since the β couplings exist to make paired assertions cheap, the natural readout keeps an edge when both endpoints assert it.

Definition 1 (Induced subgraph). Edge (u, v) is *induced* by configuration x if some neuron in cluster u representing v is active and some neuron in cluster v representing u is active, in either slot on either side. The induced subgraph of x is the set of all edges it induces.

The phrase “in either slot on either side” is doing real work. Slot assignment is arbitrary, so a readout inspecting only one slot discards every induced edge whose active neuron happened to occupy the other, and reports a systematically thinner subgraph than the network represents. The effect is not marginal: because the two slots of a cluster are exchangeable, a slot-restricted decoder misses a constant fraction of induced edges independently on every traversal, so the probability that it recovers an L -hop path falls geometrically in L . **A network can represent a path that a component-specific decoder cannot see**, and the readout must be slot-agnostic for the comparison against ground truth to mean anything; the slot-agnostic readout of Definition 1 reproduces Dijkstra’s answer on all 12 instances of the end-to-end check.

Definition 1 also fixes the comparison against ground truth, since it returns an object of the same type as the answer — one may ask whether the induced subgraph *contains* an optimal s - t path or whether it *is* one, and the two have very different answers. Choosing between them is a question about measurement rather than about the encoding, and this paper does not settle it.

IV. DESIGN CONDITIONS ON THE PARAMETERIZATION

The construction of Section III fixes the topology of the network completely: the neuron count (5), which neurons are coupled, and which of the three classes each coupling belongs to are all determined by the graph. What it does not fix are the four numbers those couplings carry; the bias b_k , the consistency weight D , the inhibitory weight I , and the scale c multiplying the continuity weight (6). The energy (3) is linear in all four, so the behavior of the

sampler is a question about a four-parameter family rather than about a single network, and the design problem is to say which points of that family have the intended property.

This section derives the conditions. It proceeds from the bottom up: first the exact combinatorics of the configuration a path produces, since every subsequent statement is an accounting over those counts; then the closed-form energy of such a configuration, which turns out to depend on the parameters through a single scalar combination; then the condition that makes the network order paths by weight; and finally the interaction between that condition and the one configuration no parameter choice can make expensive, the all-silent one. A second and independent condition, on the scale c , comes from comparing the path sector against the far larger set of configurations that resemble it, and is developed in Section V. Throughout, the augmentation of Section III-A is in force; Section IV-E sets out what changes without it, and why that matters for the measurements reported later.

A. The combinatorics of a path configuration

Fix a simple s - t path P of depth L in G and let x_P denote the configuration that encodes it. In the augmented graph, P is the path

$$S, v_0 = s, v_1, \dots, v_L = t, T, \quad (11)$$

of $L+2$ edges, two of which carry weight zero. In x_P , the terminal neurons S and T are active, at every vertex of P exactly the neurons representing that vertex's two incident edges along (11) are active, one per slot, and no other neuron in the network is active. This is the configuration the design intends to be cheap, and everything below is an accounting over it.

Lemma 1 (Counts on a path configuration). *Let x_P encode a simple s - t path of depth L . (i) In the network of Section III, x_P has $2(L+1)$ active neurons in the clusters of G and two more in S and T , hence $2(L+2)$ active neurons in all; it has $L+2$ active β pairs and $L+1$ active α pairs, one per vertex of P . It has no active γ pairs. (ii) In the variant of Section IV-E, which omits the augmentation, the corresponding counts are $2L$ active neurons, L active β pairs and $L-1$ active α pairs, again with no active γ pairs.*

Proof. For (i), the augmented path (11) has $L+3$ vertices and $L+2$ edges. Each of its $L+1$ interior vertices is a vertex of G and is traversed: it uses two incident edges and therefore holds two active neurons, one in each slot, giving $2(L+1)$; S and T are single neurons and are active, giving $2(L+2)$ in all. Each of the $L+2$ edges of (11) is asserted at both of its endpoints and so contributes exactly one active β pair, and no other edge is asserted at either endpoint. An α pair is active when a cluster holds active neurons in opposite slots with different adjacency indices, which is exactly the situation at each of the $L+1$ traversed clusters and at no other; S and T are single neurons with no slot structure and contribute none. Finally, no slot holds more than one active neuron and the two occupied slots of a traversed vertex carry different adjacency indices, so no γ coupling of either kind is engaged. For (ii) the same argument applies with s and t using one incident edge apiece and holding one active neuron each, so the interior count drops to $L-1$ and the edge count to L . $\square$

The triple $2(L+1)/(L+2)/(L+1)$ of part (i) is the foundation of the whole analysis, and it is worth being explicit about how it was established. Every count was read off a compiled network rather than derived on paper alone: the path configuration was constructed on the actual neuron array, in both the augmented and the unaugmented compilation, and its active neurons, active β pairs and active α pairs were enumerated directly. Table I reports that enumeration at $L=5$. We recommend the same discipline to anyone re-deriving these quantities for a variant of the construction, because the three counts scale differently in L and an accounting that swaps any two of them still produces a plausible-looking bias.

The counts determine the energy, and the α count determines the weight-carrying half of it. Each active α pair sits at a traversed vertex and reports the two edges of (11) meeting there, so summing $(w_j + w_k)$ over α pairs registers each edge of the augmented path at both of its endpoints. The enumeration on the instance of Table I shows the accounting in full:

TABLE I

COUNTS ON THE CONFIGURATION ENCODING A 5-HOP PATH, ENUMERATED ON A COMPILED NETWORK, FOR THE CONSTRUCTION OF SECTION III AND FOR THE UNAUGMENTED VARIANT OF SECTION IV-E. THE INSTANCE IS THE PATH 0–1–2–3–4–5 WITH WEIGHTS 1, 2, 3, 4, 5 AND A PENDANT AT EACH END SO THAT $\deg(s) = \deg(t) = 2$; $\sum_{e \in P} \varphi(e) = 15$ AND $B = 2\varphi_{\max} = 12$. ENERGIES ARE AT $c = 1$, $D = 4$, $b_k = -2$, HENCE $\lambda = +1$.

quantity	unaugmented	as constructed
active neurons	$10 = 2L$	$14 = 2(L+2)$
of these, in the clusters of G	10	$12 = 2(L+1)$
active β pairs	$5 = L$	$7 = L+2$
active α pairs $ \alpha $	$4 = L-1$	$6 = L+1$
α pair at the cluster of s ? at t ?	no, no	yes, yes
$\sum \sum_{\alpha} (w_j + w_k)$	24	$30 = 2 \sum_{e \in P} \varphi(e)$
measured E_{SPC}	2.000000	2.500000
measured E_{NC}	-4.000000	-6.000000

cluster	the two edges the pair reports	$w_j + w_k$	
$v_0 = s$	(v_0, S) at 0, (v_0, v_1) at 1	1	the s cluster
v_1	(v_1, v_0) at 1, (v_1, v_2) at 2	3	
v_2	(v_2, v_1) at 2, (v_2, v_3) at 3	5	
v_3	(v_3, v_2) at 3, (v_3, v_4) at 4	7	
v_4	(v_4, v_3) at 4, (v_4, v_5) at 5	9	
$v_5 = t$	(v_5, v_4) at 5, (v_5, T) at 0	5	the t cluster
		30	$= 2 \times 15$

Every real edge of P is charged exactly twice, once at each endpoint, and the two zero-weight edges supply precisely the two α pairs that would otherwise be missing at the ends without adding any weight of their own. This is what the augmentation was introduced to achieve. Since the $\frac{1}{2}$ of (9) is already spent converting the ordered double sum to an unordered one (Section III-C), no further halving occurs and

$$E_{\text{SPC}}(x_P) = \frac{c}{B} \cdot 2 \sum_{e \in P} \varphi(e) = 2c \tilde{E}(P), \quad (12)$$

with $\tilde{E}$ the normalized path weight of (4). On the instance of Table I this reads $2 \times 15/12 = 2.5$ at $c = 1$, against a measured $E_{\text{SPC}} = 2.500000$. The essential property of (12) is that it is proportional to the path's total weight with no additive correction of any kind: a factor of two, uniform across every path configuration in the network, and nothing else.

B. The length-bias coefficient

Because x_P engages no γ coupling, the penalty term of (8) vanishes, $M(x_P) = 0$, and the network-consistency component reduces to $E_{\text{NC}}(x_P) = -N(x_P)$. Equation (8) is written for the base construction $c = 1$; carrying the scale explicitly, each active α pair contributes c to N through the constant part of (6) and its weight-dependent part to E_{SPC} . Substituting Lemma 1(i),

$$\begin{aligned} N(x_P) &= 2(L+2)b_k + (L+2)D + c(L+1) \\ &= (L+2)(2b_k + D + c) - c. \end{aligned} \quad (13)$$

This is the point at which the parameterization asserts itself, and it does so through a single combination. Every parameter enters $N(x_P)$ multiplied by the same $L+2$ except for the $-c$ left over from the offset in the α count, so the whole family collapses onto one number.

Definition 2 (Length-bias coefficient). The *length-bias coefficient* of a parameterization is

$$\lambda \equiv 2b_k + D + c. \quad (14)$$

TABLE II

MEASURED E_{NC} AGAINST PROPOSITION 1, ON THE INSTANCE OF TABLE I AT $L = 5$, $c = 1$, $D = 4$, FOR FIVE VALUES OF THE BIAS. THE DEFINITION $\lambda = 2b_k + D + c$ IS UNCHANGED BY THE AUGMENTATION; ONLY THE AFFINE CONSTANT MOVES.

b_k	λ	unaugmented		as constructed	
		measured	$-\lambda L + c$	measured	$-\lambda(L+2) + c$
-3.00	-1	6.000000	6.000000	8.000000	8.000000
-2.50	0	1.000000	1.000000	1.000000	1.000000
-2.00	+1	-4.000000	-4.000000	-6.000000	-6.000000
-1.50	+2	-9.000000	-9.000000	-13.000000	-13.000000
-1.00	+3	-14.000000	-14.000000	-20.000000	-20.000000

Proposition 1 (Energy of a path configuration). *For the configuration x_P encoding a simple s - t path P of depth L ,*

$$\begin{aligned} E_{\text{NC}}(x_P) &= -\lambda(L+2) + c, \\ E_N(x_P) &= -\lambda(L+2) + c(1 + 2\tilde{E}(P)). \end{aligned} \tag{15}$$

Proof. Immediate from $E_{\text{NC}}(x_P) = -N(x_P)$, (13) and (14) for the first identity, and from adding $E_{\text{SPC}}(x_P) = 2c\tilde{E}(P)$, which is (12), for the second. $\square$

Proposition 1 is an exact identity and was treated as one to be checked rather than assumed. Table II compares the first identity against direct measurement on a compiled network across five bias values, holding $L = 5$, $c = 1$ and $D = 4$ and thereby sweeping λ from -1 to $+3$; the corresponding check for the unaugmented variant, with $L+2$ replaced by L , is shown alongside. The comparison isolates what the augmentation does and does not change. It moves the affine constant, and it does not touch λ : the definition (14) is the same in both columns, as is the length-neutral bias derived from it below. The second row is the reason to expect this — at $\lambda = 0$ the L -dependent term is absent and the two networks return the same energy, as they must.

The reading of (15) is direct: λ is the *per-hop bias*. The energy of a path configuration is $-\lambda$ per hop plus a constant, and on top of that the weight term adds $2c\varphi(e)/B$ per hop, which is bounded by $2c$ because B normalizes the edge weights below unity. Extending a path by one hop of weight w therefore changes its energy by

$$\frac{dE_N}{dL} = -\lambda + \frac{2cw}{B}, \quad 0 < \frac{w}{B} < 1. \tag{16}$$

Three regimes follow, and they are qualitatively different.

Corollary 1 (Ordering of path configurations). *Let $c > 0$ and consider configurations encoding simple s - t paths.*

(i) *If $\lambda \geq 2c$, every additional hop strictly lowers the energy whatever weight it carries, so longer paths are cheaper independently of weight.* (ii) *If $\lambda \leq 0$, every additional hop strictly raises the energy whatever weight it carries, so shorter paths are cheaper independently of weight.* (iii) *If $\lambda = 0$, then $E_N(x_P) = c(1 + 2\tilde{E}(P))$, a strictly increasing function of the path's normalized weight and of nothing else, so the minimizer over all path configurations is the least-weight path.*

Proof. By (16) the increment is $-\lambda + 2cw/B$ with $0 < w/B < 1$, hence $0 < 2cw/B < 2c$. If $\lambda \geq 2c$ the increment is bounded above by $-\lambda + 2c \leq 0$ and is strictly negative because $2cw/B < 2c$; if $\lambda \leq 0$ it is bounded below by $2cw/B > 0$. For (iii), $\lambda = 0$ removes the L -dependent term of (15) and leaves E_N affine and increasing in $\tilde{E}(P)$. $\square$

Case (iii) is the construction's shortest-path property, and it is worth stating plainly in those terms: at $\lambda = 0$ the energy of a path configuration is an affine, strictly increasing function of that path's total weight and of no other quantity, so the least-weight s - t path is the energy minimum of the path sector, and by (3) the most probable configuration within that sector. The property holds exactly, for every graph, at every depth, with no boundary correction and no residual dependence on hop count. The uniform factor of two carried by (12) rescales E_{SPC} globally and therefore does not disturb the argument at all; it changes only the rate at which depth trades against weight when $\lambda \neq 0$, which is why the boundary of case (i) sits at $2c$ rather than at c .

The intermediate band $0 < \lambda < 2c$ is not a fourth regime so much as a mixture of the first two: there the sign of (16) depends on the weight of the hop being added, so depth and weight trade against each other at a rate the designer has not chosen deliberately. What distinguishes case (iii) from every other value of λ is not the sign of the trade but its absence. For $\lambda \neq 0$ the energy of a path configuration is a sum of a weight term and a weight-blind term $-\lambda(L+2)$ that grows without bound in depth, so on any instance family whose optima are deep enough the second term decides the ranking; at $\lambda = 0$ the second term is constant and the ranking is the one the α couplings were designed to express.

Only the third regime realizes the intent of the construction, and setting $\lambda = 0$ in (14) gives the bias it requires,

$$\boxed{b_k = -\frac{D+c}{2}} \quad (17)$$

This is the paper’s central design parameter, and its structure repays a moment’s attention. The bias must offset exactly half the total reward a neuron receives for participating in a well-formed path — the consistency reward D it earns from its partner across the graph edge, and the continuity reward c it earns from its partner in the opposite slot — with the factor of one half appearing because (3) shares each pairwise term between the two neurons that realize it. When (17) holds, joining a path is energetically free and only the edge weight is charged.

We check (17) on a compiled network rather than resting on the algebra. The length-neutral bias, the coefficient $\lambda = 2b_k + D + c$ from which it comes, and the network consistency energy $E_{\text{NC}} = -\lambda(L+2) + c$ that both rest on were each evaluated directly on compiled instances and agree with the derived forms; Table II reports the E_{NC} check across five bias values at $L = 5$.

It is worth being explicit about why λ has to vanish rather than merely be small, and about why the condition takes this form rather than being imposed depth by depth. The bias is a single number shared by every neuron in the network; it cannot depend on L , because L is not known when the network is compiled — indeed, finding a path of unknown depth is the problem. A condition imposed at one particular depth, such as requiring $N(x_P) = 0$ for a path of a specific L , is therefore satisfiable, but it leaves a residual proportional to L at every other depth. The natural condition to impose is that the L -dependence itself vanish, and (14) is exactly the coefficient of that dependence. This is what makes λ the right object: it is the one combination of b_k , D and c that a depth-agnostic design must control.

Equation (17) also suggests the parameterization we use for the rest of the paper. Inverting (14) gives

$$b_k = \frac{\lambda - D - c}{2}, \quad (18)$$

so that λ and c may be set independently and the bias follows. This matters experimentally, because c and λ are the two axes along which behavior changes and a sweep that adjusts b_k by hand moves both at once; (18) separates them, and every parameter sweep in this work is laid out on that grid.

C. How exactly λ must vanish

Corollary 1(iii) is a statement about a single value, and a design condition that holds only on a measure-zero set is of limited use to anyone building the thing, least of all on a neuromorphic substrate, where finite parameter precision is part of the motivation. The right question is therefore not whether λ can be set to zero but how far from zero it may sit before the ordering changes, and (15) answers it directly.

For two path configurations encoding P and Q ,

$$E_N(x_Q) - E_N(x_P) = -\lambda(L_Q - L_P) + \frac{2c}{B}(W_Q - W_P), \quad (19)$$

with W the total weight. Taking P to be the least-weight path, every rival with the same hop count is ordered correctly at any λ , since the first term vanishes. Only rivals at a different depth can invert, and requiring that none of them does gives

Proposition 2 (Ordering tolerance). *Let P be the least-weight s - t path. The energy ordering of P against every other simple s - t path configuration is that of Corollary 1(iii) provided*

$$|\lambda| < \frac{2c}{B} \min_{Q: L_Q \neq L_P} \frac{W_Q - W^*}{|L_Q - L_P|}. \quad (20)$$

Proof. Immediate from (19): the right-hand side is positive iff $\lambda(L_Q - L_P) < (2c/B)(W_Q - W^*)$, and since $W_Q \geq W^*$ the binding cases are those in which λ and $L_Q - L_P$ share a sign. Dividing by $|L_Q - L_P|$ and taking the minimum over such Q gives (20). *[DERIVED; the inequality has not itself been checked against a compiled network, though the quantity on its right has been measured — see below.]* $\square$

Two things about (20) are worth drawing out. It is *not* a statement about λ alone: the tolerance scales with c , so a network run at a larger continuity scale tolerates proportionally more length bias, and the two design conditions of this paper therefore interact rather than being independent knobs. And its right-hand side is a per-instance quantity — the smallest weight excess per unit of hop-count difference among the optimum’s rivals — so the precision a given graph demands is a property of that graph, not of the construction.

On the reference instance it is small. Enumerating rivals by a loopless k -shortest-paths search, the optimum has $W^* = 0.548769$ at $L_P = 8$ and the minimizing rival sits at $L_Q = 4$ with $W_Q = 0.550628$: a weight excess of 0.00186 spread over four hops, so the denominator of (20) is 4.648×10^{-4} , and at $c = 4$ the tolerance is about 1.9×10^{-3} in λ . That looks like a knife-edge.

Measured over a population it is not the typical case, and the correction is worth more than the original reading. Across 72 instances spanning four families and depths 3 to 15, the denominator has a median of 5.2×10^{-2} — **two orders of magnitude looser than the reference instance**, which sits near the fifth percentile of its own population. And it is strongly family-dependent, with medians of 1.04 for the lattice family, 0.067 for sparse $G(n, p)$, 0.042 for the dense arm and 1.3×10^{-3} for the spatially embedded terrain graphs — **four orders of magnitude between topologies**.

Three things follow, and the first is the one a practitioner needs. **The tight regime is real but it belongs to the spatial family**, where near-equal route weights are a property of the embedding rather than an accident of one draw; on $G(n, p)$ the typical instance tolerates a length bias two orders of magnitude larger. **The tolerance is therefore an instance-level design quantity and should be evaluated, not inherited** — (20) is cheap to compute and its value cannot be guessed from the family alone. And **these are upper bounds**: the search reached its $K = 4000$ limit on every one of the 72, so each reported denominator may exceed the true minimum, and this measurement cannot distinguish a rival never reached from one reached above its true value. A cross-check against exhaustive depth-first enumeration agrees on 147 of 162 depth strata, the discrepancies being exactly that truncation.

D. The all-silent configuration

There is one configuration whose energy no choice of parameters can alter. Every term in (8) and (9) is a sum over active neurons or over active pairs, so on the configuration in which nothing is active every term is an empty sum:

$$E_N(\mathbf{0}) = 0 \quad \text{identically, for all } b_k, D, I, c. \quad (21)$$

This is a fixed reference point against which every other configuration is scored, and it is worth noting that the same is not true of any other configuration in the network: $\mathbf{0}$ is the unique state whose energy is parameter-independent.

Comparing (21) with Proposition 1 gives the following.

Proposition 3 (When silence competes with the path sector). *Let $c > 0$ and let P be a simple s - t path. (i) Under the terminal bias $b_S = b_k$ adopted in Section III-A, if $\lambda \leq 0$ then $E_N(\mathbf{0}) < E_N(x_P)$: the all-silent configuration is strictly cheaper than every path configuration, and in particular at the length-neutral point $\lambda = 0$ identified by Corollary 1. (ii) Under $b_S = 0$ the comparison is conditional rather than universal. At $\lambda = 0$, $E_N(x_P) = 2c\tilde{E}(P) - D$, so silence is cheaper exactly when $\tilde{E}(P) > D/2c$, and any path lighter than that is cheaper than silence.*

Proof. For (i), (15) gives $E_N(x_P) = -\lambda(L+2) + c(1+2\tilde{E}(P))$; at $\lambda = 0$ this is $c(1+2\tilde{E}(P)) > 0$ since $c > 0$ and $\tilde{E}(P) > 0$, and for $\lambda < 0$ the term $-\lambda(L+2)$ is positive and the energy larger still. For (ii), $b_S = 0$ replaces the bias count $2(L+2)$ in (13) by $2(L+1)$, so $N(x_P) = (L+1)\lambda + D$ and $E_{NC}(x_P) = -\lambda(L+1) - D$; adding $E_{SPC}(x_P) = 2c\tilde{E}(P)$ from (12) and setting $\lambda = 0$ gives the stated form, against $E_N(\mathbf{0}) = 0$ from (21). Both branches are exact identities in the counts of Lemma 1(i) and were evaluated directly on compiled networks. $\square$

Part (ii) is worth stating even though this paper does not adopt it, and it is not a hypothetical. $b_S = 0$ is what makes the source’s own prescription literally true — a terminal whose cluster is occupied then receives $2D$

and fires once per refractory period, where at $b_S = b_k$ an unanchored terminal waits some 55τ (Section III-A). **A design that wants the terminals on that schedule must take $b_S = 0$, and part (ii) is the price list.** It also locates where the design’s dependence on the terminal drive actually comes from. The drive is load-bearing under (i) and conditionally so under (ii), and which regime a given implementation is in is decided by a parameter — the bias of two auxiliary neurons — that neither the original statement of the construction nor any re-implementation of it has occasion to think about. On the reference instance at the operating point used later, $\tilde{E}(P) = 0.276267$ against a threshold $D/2c = 0.5$, so the two branches disagree there: under (i) silence undercuts the path by 6.210 and under (ii) the path undercuts silence by 1.790.

The two conditions of this section are therefore in tension, and the tension is structural rather than numerical. Corollary 1 says the network orders paths by weight only at $\lambda = 0$; Proposition 3(i) says that at $\lambda = 0$ the sector containing all those paths sits above the empty configuration. Raising λ above zero buys occupancy of the path sector at the cost of the ordering within it: the network is then active, but its preference runs toward depth rather than toward weight. There is no value of λ at which both hold.

The resolution is already in the construction, and Proposition 3(i) is what gives it a name. Section III-A keeps the terminal neurons S and T active, because they are the two units whose participation is given rather than inferred. *At the operating point* that drive removes 0 — and every configuration in its neighborhood that leaves the terminals silent — from the accessible ensemble, so the comparison Proposition 3(i) makes never has to be won. The terminal drive is thus not a boundary-condition convenience but a load-bearing part of the design: it is what makes $\lambda = 0$ a usable operating point. It is also why the neuron count (5) carries its $+2$.

The exclusion is local to that operating point, and the reason is the terminal bias again. Under $b_S = b_k$ the terminals are cooled by the same bias as every other neuron, so the additive drive competes against b_k rather than overriding it. Measured on the sweep of Table III, the terminals’ own active rate falls from 0.98 at $b_k = -2$ to 0.05 at $b_k = -7$: the drive holds S and T up where the design sits, and is overwhelmed well before the cold end. **The sweep makes that boundary directly visible.** It is initialised in the all-silent configuration, and at $b_k = -3$ and $b_k = -4$ the network leaves that configuration and never returns to it in 20,000 steps, while at $b_k \leq -5$ it returns and stays there for thousands of them. The empty configuration is therefore reachable under deep cooling whether the drive is delivered at every step or periodically — both disciplines were run and both reach it — which is why the sweep finds an energy floor of 0.00, and why that observation is *consistent* with the construction rather than evidence against it. **The analytic statement is Proposition 3(i); the table corroborates it near the design point and shows what leaving that point costs.**

The measured evidence runs in the expected direction, and we set out both what it shows and what it does not. Table III reports a sweep in b_k alone at $c = 1$, $D = 4$, on the reference instance over 20,000 steps per rung, on which the target path configuration has 20 active neurons, induces 10 edges and sits at $E_N = -8.447$. As the bias is lowered the network cools through the target’s activity level and far past it — from 116.7 active neurons at $b_k = -2$ to 0.59 at $b_k = -7$ — but the target is never approached. The lowest energy the sampler reaches is 0.00, the all-silent configuration, and not -8.447 ; from $b_k = -5$ downward that floor is reached at every rung. The energy floor the sampler finds is silence.

The contained column makes Definition 1’s distinction concrete. The optimal path is contained in the induced subgraph five times at $b_k = -2$ and twice at $b_k = -3$, at rungs whose lowest energy over the whole run is 87.12 and 65.79 against a target at -8.447 — some 95 and 74 energy units above it. **Containment does not require the network to be anywhere near the configuration that encodes the path;** it requires only that the path’s edges be among the many the network is asserting at once. That is the gap between *contains* and *is*, measured rather than defined.

One limit of that evidence should be stated. No run reported here measures residence time in the all-silent configuration with the drive switched off against on; what the sweep establishes is where the sampler goes under cooling, not a controlled contrast between drive disciplines. Proposition 3(i) is the analytic statement, and the table sits alongside it as corroboration.

E. What the augmentation buys, measured by what its absence costs

The clearest way to show what the two auxiliary vertices are for is to compile the network without them. That is not a hypothetical: it is the natural reading of the construction, it is what our own re-implementation did. We

TABLE III

COOLING BY BIAS ALONE ON THE NETWORK OF SECTION III, AT $c = 1$, $D = 4$, WITH THE PERIODIC TERMINAL DRIVE, ON THE REFERENCE INSTANCE ($N = 60$, $p = 0.05$, SEED 19), 20,000 STEPS PER RUNG. INDUCED-EDGE COUNTS ARE IN THE AUGMENTED FRAME, IN WHICH THE TARGET PATH CONFIGURATION HAS 20 ACTIVE NEURONS, INDUCES 10 EDGES AND SITS AT $E_N = -8.447$. ENERGY MINIMA ARE TAKEN OVER POST-BURN-IN STEPS, THE FIRST 2,000 DISCARDED; MEANS ARE OVER THE FULL RUN. ACTIVITY DESCENDS THROUGH THE TARGET’S LEVEL AND FAR PAST IT, AND THE SAMPLER REACHES THE ALL-SILENT CONFIGURATION — $E_N = 0.00$ — AT THE THREE COLDEST RUNGS, NEVER THE TARGET.

b_k	active	induced	E_N mean	E_N min	contained
-2.0	116.7	44.5	1103.54	87.12	5
-3.0	73.7	26.9	355.26	65.79	2
-4.0	27.6	8.4	117.35	24.00	0
-5.0	6.8	1.4	31.62	0.00	0
-6.0	1.8	0.2	10.47	0.00	0
-7.0	0.6	0.0	4.05	0.00	0

call it the *variant*. It allocates clusters for the vertices of G and no others, and supplies the boundary condition by an additive drive to the neurons of the s and t clusters rather than by two augmenting neurons; Lemma 1(ii) gives its counts. **It is internally consistent, which is exactly why the omission survives self-checking, and it does not have the ordering property the construction is designed for.**

Without the augmentation the clusters of s and t hold one active neuron apiece and contribute no α pair, so the α sum registers each interior path edge at both endpoints but each of the two terminal edges at its single interior endpoint only:

$$E_{\text{SPC}}(x_P) = \frac{c}{B} \left(2 \sum_{e \in P} \varphi(e) - \varphi(e_1) - \varphi(e_L) \right), \quad (22)$$

with e_1 and e_L the terminal edges. Table I confirms this on the 5-hop instance, where (22) evaluates to 2.000000 against a measured 2.000000 and (12) to 2.500000. On the reference instance used throughout this paper — a $G(n, p)$ graph with $N = 60$, $|E| = 142$ and an 8-hop optimum, at $D = 4$, $b_k = -2$, $c = 1$, hence $\lambda = +1$ — the same forms give a measured $E_{\text{NC}} = -7.000000$, exactly $-\lambda L + c$ for the variant, and a measured $E_{\text{SPC}} = 0.510616$, which is (22) to six figures; the plain normalized weight of that path is $\tilde{E}(P) = 0.276267$, so the two are quite different quantities and should not be read for one another.

The consequence is that the boundary term of (22) depends on which edges sit at the ends of the path, and a path may lower its E_{SPC} by placing heavy edges at the terminals whatever its total weight. Two 4-hop s – t paths make the effect concrete at $c = 1$. Two paths of *equal* total weight 10.0, one with weights (1, 4, 4, 1) and one with (4, 1, 1, 4), have $E_{\text{SPC}} = 2.250000$ and 1.500000 and hence energies -0.750000 and -1.500000 : the ranking separates them by 0.75 at identical weight. And the ordering can invert outright. A path of weights (0.1, 4, 4, 0.1) and total weight 8.2 — Dijkstra’s answer on that instance — sits at $E_{\text{SPC}} = 1.620000$ and $E_N = -1.380000$, while a path of weights (5, 0.1, 0.1, 5) and total weight 10.2 sits at $E_{\text{SPC}} = 1.040000$ and $E_N = -1.960000$: the heavier path is preferred by 0.58 at $c = 1$, and by 2.32 at the $c = 4$ operating point used later.

These are constructed instances exhibiting the inversion, not an estimate of how often it occurs; what they establish is that the ordering property fails on the variant, not the rate at which it fails. How often the variant’s minimum and Dijkstra’s answer disagree over a graph population is the measurement noted at the end of this section as one we have not made.

This is the clearest evidence available that the augmentation is load-bearing rather than presentational. Corollary 1(iii) holds on the network of Section III because (12) is proportional to total weight; on the variant, (22) is not, and the ordering property does not survive.

One consequence should be recorded, because it governs how any measurement taken on the variant is to be read. Such measurements are unaffected *in their own terms* whenever they are scored against a reference computed by Dijkstra’s algorithm on the graph rather than against the network’s own energy ranking. What carries an extra assumption is any *interpretation* of a failure as a failure to reach or hold the network’s own minimum, since on the variant that minimum need not be Dijkstra’s path. How often the two disagree, and by how much weight, is a measurement we have not made and do not estimate.

TABLE IV

QUANTITIES IN THE PARAMETERIZATION, AS STATED IN [8] AND AS DERIVED IN THIS SECTION. PATH QUANTITIES REFER TO A CONFIGURATION ENCODING A SIMPLE s - t PATH OF DEPTH L ON THE NETWORK OF SECTION III; THE NUMERICAL COLUMN EVALUATES AT $L = 8$.

Quantity	Earlier	As derived here
Active β pairs	$(L+2)/2$; 5 at $L=8$	$L+2$; 10 at $L=8$
Active α pairs	$2(L+1)$; 18 at $L=8$	$L+1$; 9 at $L=8$
E_{SPC} on a path	$B^{-1} \sum_{e \in P} \varphi(e)$	$2cB^{-1} \sum_{e \in P} \varphi(e)$
Bias condition	$-(\frac{D}{4} + 1)$, imposed at fixed L	$-\frac{D+c}{2}$, from $\lambda = 0$

F. Relation to the earlier statement of the construction

The accounting derived above differs in a few particulars from the one given in the earlier statement of this construction [8]. Table IV sets the differences out for readers working from that source. Three of the four are factors of two in a count or a coefficient: the β count enters through the coefficient of D in (13) and hence the D term of λ , and the α count enters through the coefficient of c and supplies the constant term of (15). The third of them, in E_{SPC} , rescales that component of the energy uniformly across every path configuration, so it leaves the ordering of paths — and with it the shortest-path property of Corollary 1(iii) — exactly as stated in those sources; what it moves is the rate at which depth trades against weight away from $\lambda = 0$, and hence the location of the regime boundary in Corollary 1(i). The fourth entry is of a different kind. The bias condition is obtained there by requiring the bracket of (8) to vanish at one particular depth while the resulting bias is L -independent, which leaves a residual linear in hop count at every other depth; imposing the same zero on the counts derived here, at $c = 1$, gives $b_k = -(\frac{D}{2} + \frac{L+1}{2(L+2)}) \approx -\frac{D+1}{2}$, the $c = 1$ case of (17). The active-neuron count $2(L+1)$ in the clusters of G agrees with Lemma 1(i) and needs no reconciliation.

V. FREE ENERGY AND THE DISCRIMINATION CONDITION

Section IV settled what the energy function has to look like: the length-bias coefficient λ of Definition 2 must vanish, and then Corollary 1(iii) makes the energy of a path configuration an increasing function of that path's weight alone. That is the right condition on the energy, and it is worth pausing on what it does and does not deliver. It orders the path configurations correctly among themselves. It says nothing about how much of the sampler's time is spent on path configurations in the first place.

The quantity that governs sampling is not energy but free energy. A Boltzmann sampler assigns probability $e^{-E_N(x)}$ to each configuration, so the mass carried by a family $\mathcal{F}$ of configurations is a sum, $\sum_{x \in \mathcal{F}} e^{-E_N(x)} \approx |\mathcal{F}|e^{-\bar{E}}$, whose logarithm is an entropy term $\ln |\mathcal{F}|$ against an energy term $\bar{E}$. A family with modest energy and enormous membership can outweigh one with lower energy and few members, and the path sector is small by any measure: the number of simple s - t paths of a given depth grows like a branching factor raised to that depth, while the number of configurations of comparable activity grows like a binomial coefficient in $|E|$. Whether the network spends its time on paths is therefore a comparison between a designed energy margin and a combinatorial multiplicity, and this section carries it out.

A. Configurations that look like paths

To make the comparison sharp we need the family of configurations that competes most directly with the path sector: those the network's structural couplings cannot tell apart from a path.

Definition 3 (Scatter configuration). A *scatter* configuration of size m induces m pairwise disjoint edges — mutually consistent, sharing no vertex, and hence forming no chain.

A scatter configuration is a perfectly legitimate state of the network, and its counts follow directly from the definition. Each of its m edges is asserted from both of its endpoints, so it contributes one active neuron at each

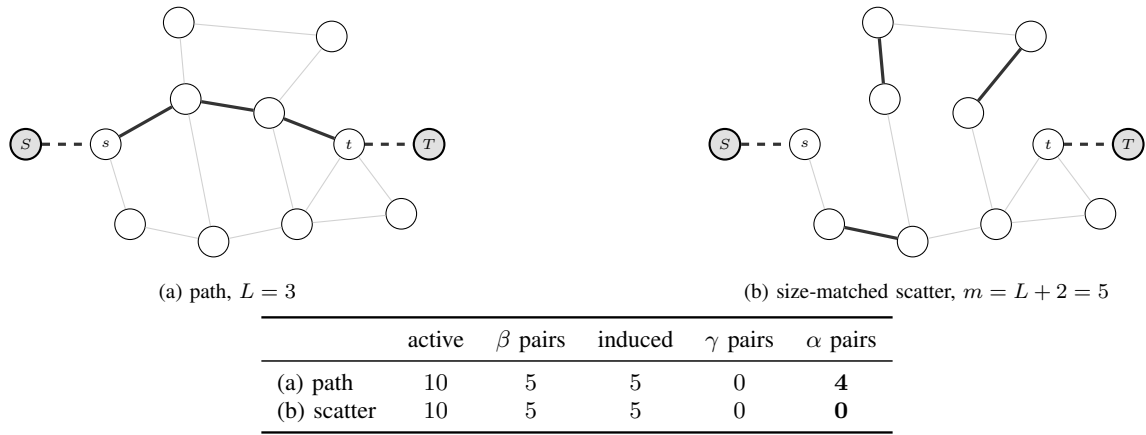


Fig. 3. **The competitor the global couplings cannot see.** Both configurations are drawn on the same graph; bold edges are induced, dashed bold are the two zero-weight auxiliary edges, which are admitted to the competitor set (Section V-A). They agree in every quantity a global mechanism can observe — total activity, agreed edges, induced edges, slot violations — and differ in one count only. **The bias term, the cross-cluster weight D and the winner-take-all weight I therefore cancel exactly between them** (23), leaving the continuity scale c as the only parameter that separates a path from a scatter, and $c(L+1)\bar{\alpha}$ as the entire margin (24).

end and exactly one active β pair. Because the edges share no vertex, no cluster is touched twice: no cluster holds two active neurons in one slot, and none holds active neurons in *both* slots. A size- m scatter configuration therefore has $2m$ active neurons, m active β pairs and m induced edges, and engages no γ coupling of either kind — and, which is the point of the definition, it has no active α pairs at all.

Set that against the path configuration, whose counts are Lemma 1(i) and are not re-derived here. The configuration encoding a path of depth L has $2(L+2)$ active neurons, $L+2$ active β pairs, induces the $L+2$ edges of the augmented path (11), and engages no γ coupling. The size-matched competitor is therefore the scatter of size $m = L+2$, and matched it is exactly: the same total activity, the same number of agreed edges, the same number of induced edges, the same absence of slot violations. In every quantity the network’s global couplings can observe, the two families are the same object. They differ in one count (Figure 3). The path configuration carries $L+1$ active α pairs, one at each traversed vertex of the augmented path; the equal-activity scatter carries none.

The same matching in the implemented variant of Section IV-E pairs a depth- L path configuration with a size- L scatter, since by Lemma 1(ii) that configuration has $2L$ active neurons and L active β pairs there; the α count on the path side is then $L-1$ against none. The structure of the comparison is identical in the two cases and only the offsets differ, which matters below because the measurements reported in this paper were made on the variant.

That single difference in the α count is the whole of the discrimination problem, and it fixes the margin. The two families agree in total activity and in β count, so the bias term and the cross-cluster class contribute equally to both and cancel in the difference; neither engages a γ coupling, so the penalty term is zero for both. What is left of the energy difference is the path configuration’s total α contribution, and nothing else:

$$\Delta = c \sum_{\alpha \text{ pairs}} f \quad \text{independently of } b_k, D \text{ and } I. \quad (23)$$

Evaluating that sum with the counts of Lemma 1(i) and the continuity weight (6), whose weight-dependent part is (12), gives the margin in closed form:

$$\begin{aligned} \Delta &= c \left[(L+1) - 2\tilde{E}(P) \right] = c(L+1)\bar{\alpha}, \\ \bar{\alpha} &= \frac{1}{L+1} \sum_{\alpha \text{ pairs}} f = 1 - \frac{2\tilde{E}(P)}{L+1}. \end{aligned} \quad (24)$$

The two displays have different standing and it is worth separating them before either is used. Equation (23) is an exact cancellation: it requires only that the two families be matched in activity and in β count and engage no γ coupling, all of which are established counts. Equation (24) is algebra performed on top of it, following from Lemma 1(i) together with (6) and (12). **We checked it against a compiled network rather than leaving it as algebra.** Size-matched scatter configurations were built by hand on compiled augmented networks at $L \in \{3, 5, 8\}$

and on the reference instance at $c \in \{1, 4\}$, the two energies differenced, and the difference compared against (24): the worst discrepancy over the sixteen rows is exactly zero. The competitor is properly matched in the sense the cancellation requires — path and scatter agree on active count, on β count and on γ count, so the α class is the only difference and the margin is free of λ .

Two definitional rulings inside that check are worth stating, because both were close and both move the numbers that follow. Zero-weight auxiliary edges are *admitted* to the competitor set: the margin is identical either way, but excluding them under-counts the competitor and would bias any threshold read off it downward. And the count is over *microstates* rather than edge sets, because winner-take-all slots are interchangeable at equal energy — a path carries 2^{L+1} assignments and a k -matching carries 4^k , verified by enumeration. On the reference instance that second ruling is worth a factor of 2048, or 7.6 nats, which moves the required c by +0.89: four times the effect of the first.

One point of hygiene, since the whole argument compares configurations the network can actually occupy. With the terminals driven, any admissible state has S and T active, and the scatter competitors are constructed to satisfy that: the matched-size matching is built on the augmented graph and the two auxiliary edges are admitted to the competitor set, so a scatter configuration is a genuine reachable state of the driven network and not merely an edge set with the right cardinality. **A competitor the network could not occupy would make the comparison vacuous, and we admit the auxiliary edges partly for that reason** — the other reason, that excluding them under-counts the competitor and biases the threshold downward, is taken up in Section V-B.

The reading of (23) is that continuity is the only thing being paid for. Every quantity a global mechanism can observe — total activity, the number of agreed edges, the number of violated slots — is identical between the two families, so the cross-cluster coupling D , which is precisely the reward for agreement, contributes equally to both and drops out. This is not an approximation but an exact cancellation, and it does not depend on which of the two count sets is in force. Raising D makes both families cheaper by the same amount and buys no discrimination whatever. The continuity scale c is the only parameter that separates them. On the implemented variant the same cancellation gives $\Delta = c(L - 1) - E_{\text{SPC}}(x_P)$, which on the reference instance of Section IV-E — $N = 60$, $|E| = 142$, $L = 8$, $c = 1$, with a measured $E_{\text{SPC}} = 0.510616$ — evaluates to 6.489384, or $\bar{\alpha} = 0.927$ per α pair.

B. How large the competitor family is

The margin now has to be weighed against multiplicity. The shape of that comparison is not in doubt, and it is worth stating separately from its numerical value, which is.

Simple s – t paths of a given depth are counted by a branching factor raised to that depth; equal-activity scatter configurations are counted by a binomial coefficient in $|E|$, and on any graph of interest the second is astronomically the larger. Aggregate mass is multiplicity times a Boltzmann factor, so the path sector carries the larger mass only if the energy margin buys back the log-multiplicity of the look-alike family. Two features of that trade are asymptotic and survive any change in the constants. First, the count is a lower bound on the competition rather than an estimate of it, because it is taken at matched activity while the unconstrained sampler runs far hotter than a path requires, so the family the sampler actually explores is larger still. Second, the asymptotics are mild. With $\#\text{scatter} \sim \binom{|E|}{m}$ at the matched size m and $\#\text{path} \sim b^L$ for a branching factor b ,

$$\ln\left(\frac{\#\text{scatter}}{\#\text{path}}\right) \sim L \ln\left(\frac{|E|e}{Lb}\right), \quad (25)$$

so the entropy the margin must overcome grows linearly in L and only logarithmically in $|E|$. Since the margin (24) also grows linearly in L , the depth dependence cancels and the requirement on c is logarithmic in graph size rather than polynomial — a design condition that does not become a treadmill as instances grow.

The numerical value of the threshold is a different matter, and we state plainly that we do not quote one — but the reason is a measurement rather than an absence. Both inputs are set by the counts: the competitor family is enumerated at activity $2(L + 2)$ by Lemma 1(i), and the margin it is compared against is (24). We carried that enumeration out at the matched size rather than leaving it open. On the reference instance — $N = 60$, $|E| = 142$, $L = 8$, $\tilde{E} = 0.276267$ — exhaustive depth-first search gives 1,754 paths at matched depth and a frontier dynamic program gives 3.35×10^{19} matched-size scatter configurations. **Both sides are then converted to microstates before the ratio is taken**, as the convention of Section V-B requires: each path edge set carries $2^{L+1} = 512$

TABLE V
REQUIRED c OVER 1,027 COMPLETED INSTANCES, MICROSTATE CONVENTION. QUANTILES, NOT EXTREMES. “CLEARS” IS THE FRACTION WITH $c_{\text{req}} < 4$.

family	n	med L	q25	med	q75	clears
sparse	223	3	4.57	5.72	7.01	11.7 %
dense	182	3	4.50	5.35	6.39	11.0 %
grid	360	14	9.89	13.57	16.40	0.0 %
mtn-kNN	262	10	3.76	4.28	4.82	35.1 %
pooled	1,027	7	4.47	5.82	10.32	13.4 %

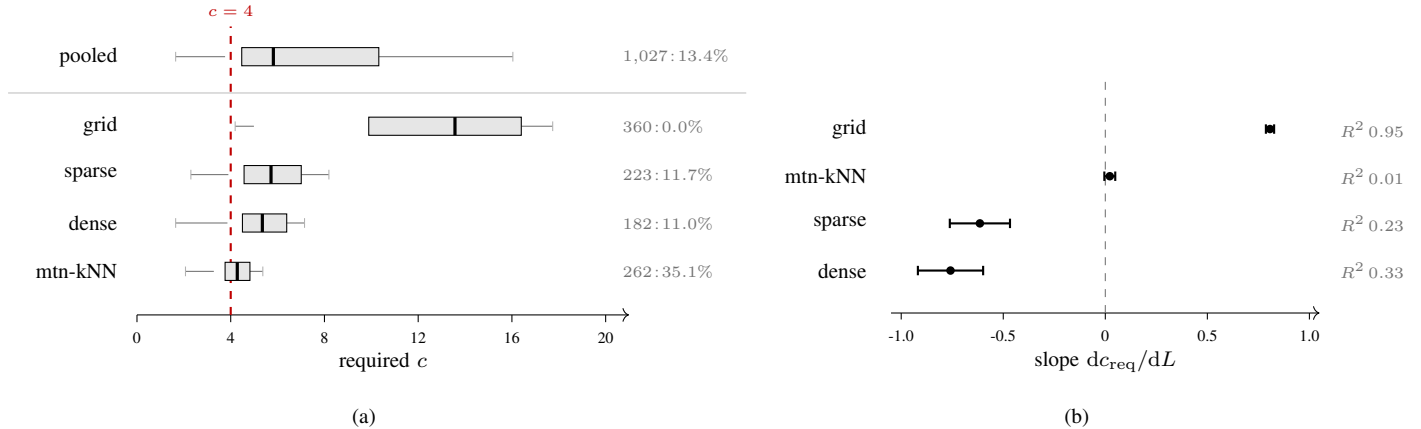


Fig. 4. The required continuity scale over 1,027 instances in four graph families. (a) Distribution, not extremes: box spans the interquartile range with the median marked, whiskers reach the 10th and 90th percentiles, outer ticks the minimum and maximum; right-hand labels give n and the fraction clearing at $c = 4$ in the microstate convention. The lattice family clears on none of 360 instances while the spatially embedded family clears on 35.1 %, which is why a single threshold cannot be quoted. (b) Why it is not a depth law: fitting c_{req} against optimal depth within each family gives slopes of *opposite sign*, with 95 % intervals excluding zero in three of four cases. Both are correct — c_{req} is the ratio (27), whose numerator is family-independent and whose denominator is the margin.

configurations, so $\#\text{path} = 1,754 \times 512 = 898,048$, and the scatter figure is already a microstate count. That gives a log-ratio of 31.25 nats against a margin at $c = 1$ of 8.447, so the two sides cross at $c = 3.699$.

The reason we quote no threshold is that the value does not hold still, and we measured how far it moves rather than asserting that it does. We enumerated the required c over **1,080 instances**: four graph families — sparse and dense $G(n, p)$, square lattices, and spatially embedded k -nearest-neighbor terrain graphs — across three or more sizes each, forty graphs per cell, with 53 instances (4.9 %) censored at a declared wall or memory cap and reported as censored rather than imputed. **The censoring is not random and its direction is favourable to the construction**: it concentrates in the deepest lattice and terrain cells, which are also where required c is largest, so the quantiles below and the clear-fraction are computed on a population from which the hardest instances have been removed. Every figure reported here is therefore optimistic by an amount we do not estimate, and in the same direction as the lower-bound status of the competitor count. Table V and Figure 4(a) give the distribution.

The population spans 1.65 to 19.59, an order of magnitude, with a pooled interquartile range of 4.47 to 10.32. **A single instance-independent threshold is not available and reporting one would be reporting the reference instance.** What *is* stable is a direction: within every family the requirement *rises with size* — in sparse $G(n, p)$ the median runs 4.22, 5.01, 5.88, 6.15, 6.46, 7.77 across $N = 20$ to 100, and in the dense arm 4.32 to 6.87 across $N = 20$ to 80. **Discrimination gets harder as instances grow, not easier.**

a) *Required c is not linear in depth, and the failure of that fit is the informative result.*: Fitting c_{req} against L within each family gives slopes of -0.61 $[-0.76, -0.47]$ for sparse, -0.76 $[-0.92, -0.60]$ for dense, $+0.81$ $[+0.79, +0.83]$ for the lattice and $+0.02$ $[-0.005, +0.049]$ for the terrain family — **rising, flat and falling in the same sweep**, as Figure 4(b) shows. The reason is that c_{req} is a ratio, and only its numerator behaves as (25)

predicts. Fitted once over all 1,027 instances with no per-family term,

$$\ln\left(\frac{\#\text{scatter}}{\#\text{path}}\right) = \kappa_L L + \kappa_E \ln|E| + \kappa_0, \quad (26)$$

with $\kappa_L = 2.289$ [2.250, 2.328], $\kappa_E = 11.566$ [11.224, 11.909] and $\kappa_0 = -42.451$, at $R^2 = 0.966$ — **linear in depth and logarithmic in size, the two dependences (25) identifies, and with a family-independent depth coefficient.**

The relationship between (26) and (25) is weaker than a transcription, so we fitted the predicted form as well. Expanding (25) gives $L \ln(|E|e/L) - L \ln b$: the size dependence enters as a *product* with depth, and there is a $-L \ln L$ term. (26) is additive and separable and has neither, so it can only be a linearization over the depth range this population spans. Fitting the interaction form $\beta_1 L \ln(|E|e/L) + \beta_2 L + \beta_0$ — the same three free parameters — returns $\beta_1 = 1.75$ [1.72, 1.79], $\beta_2 = -2.819$ [-2.922, -2.716], $\beta_0 = 2.799$ [2.543, 3.055] at $R^2 = 0.986$ against 0.966.

b) The asymptotic gets the form right and the leading coefficient wrong.: (25) predicts $\beta_1 = 1$. The fit returns 1.75, with a confidence interval nowhere near unity, and the value is stable at 1.86 with the intercept suppressed and at 1.72 when competitors are counted as edge sets rather than microstates — **so it is a property of the population and not of a fitting convention.** Fitting the three terms separately, $L \ln|E|$, $L \ln L$ and L , returns +1.690, -1.219 and -2.480 at $R^2 = 0.987$: **the $-L \ln L$ term the expansion predicts is recovered near its predicted -1, while the leading term is renormalized by three quarters.** The structure of the asymptotic is therefore confirmed and its prefactor is not. We read the excess as a finite-size effect the leading-order count does not carry — the expansion prices each hop by the branching available to it and ignores the correlations between hops that a fixed graph imposes — and we do not claim to have derived the value it takes.

That reading suggests a test, and we report the test as inconclusive rather than leaving the reading unexamined. Hop correlations differ sharply across these topologies, a lattice constraining successive hops far more than a sparse $G(n, p)$ does, so if the reading is right β_1 should vary by family. Refitting per family returns 2.07, 2.21, 0.55 and 2.41 for sparse, dense, lattice and terrain, with disjoint confidence intervals — which looks like confirmation and is not. **Within any one family the two regressors are collinear**, $L \ln(|E|e/L)$ and L correlating at 0.90 to 0.999, so β_1 and β_2 are not separately identified there and only their combination is. The lattice is the extreme case at 0.9991: a lattice's edge count is determined by its size, so the family supplies only nine distinct values of $|E|$, and its $R^2 = 1.000$ reports a near-degenerate regression rather than a measurement. The four extra coefficients also buy almost nothing, R^2 moving from 0.986 to 0.9965.

What the per-family spread measures is therefore the $(L, |E|)$ regime each family occupies, not its topology, and this population cannot separate the two — family, depth and size range vary together by construction, with median depths of 3, 3, 14 and 10. The pooled coefficient is well identified precisely because it spans a range no single family does, which is why we quote it and do not read the spread as a topology signal. **The correlation reading is accordingly neither confirmed nor refuted here**, and separating it would need a population in which depth and family are crossed rather than confounded. The family-independence this paper does claim rests on the held-out transfer results below, which test it directly, and not on this fit.

The interaction fit also says what (26)'s coefficients mean. Its size coefficient is not an independent size exponent but the interaction coefficient carrying a population-typical depth, and the two fits say so quantitatively: $\kappa_E/\beta_1 = 6.6$, against a pooled median depth of 7. **The additive form works because this population's depths are concentrated.**

c) We nonetheless state (26) as the rule, and the reason is the only one that should decide it.: The interaction form is the better description of this population, by a wide margin — 4.0% median relative error against 8.6%. **It is the worse extrapolator wherever the weight regime changes, and the two are tied where it does not.** Withheld one family at a time, the two forms are indistinguishable: worst held-out ratios of 0.84 and 1.17, or 17.4% and 15.7% on a log scale, marginally favoring the interaction form. Asked to predict the $U[1, 2]$ weight convention neither was shown, the additive form returns 7.5% median error against 11.8%; on the reference instance, which is in neither fitted population, it is wrong by 6.2% against 10.8%. **Two out-of-sample wins and one tie, against one in-sample test in the other direction.**

That pattern is the one this paper's own thesis predicts, which is why we read it as a result rather than as three assorted comparisons. The numerator is family-independent, so a form that captures the numerator should transfer across families about equally well however it is parameterized — and the family hold-out duly ties. The

whole spread between topologies lives in the margin, through ρ ; so a change of *weight* regime is the transfer that can separate two numerators, and it is exactly there that the coarser form wins twice. A design rule is applied to an instance that was not in the population it was fitted on, so the out-of-sample behavior is the behavior that matters and (26) is what we recommend. The better fit is reported because it is what the theory predicts and because the comparison is the evidence for this choice, not as an alternative rule.

That β_1 is a fitted renormalization rather than a derived constant is, we think, why it does not travel: the extra flexibility buys agreement with this population's correlation structure and carries it to the next population as bias. **Equal parameter counts do not imply equal transfer**, and a reader taking a law from this paper to a different family should take the coarser one.

The denominator is not a fit at all but the identity margin $= (1 - 2\rho)L + 1$, writing $\rho = \tilde{E}(P)/L$ for the path's mean edge weight in units of B . Together,

$$c_{\text{req}}(L, |E|) = \frac{\kappa_L L + \kappa_E \ln |E| + \kappa_0}{(1 - 2\rho)L + 1}. \quad (27)$$

The fit coefficients are written κ to keep them clear of b , the branching factor of (25), and of c , the continuity scale. Required c therefore rises with depth over the range this population spans where the $\ln |E|$ term dominates the numerator and falls where it does not. We do not extrapolate the trend beyond that range: (26) is a linearization, and the $L \rightarrow \infty$ behavior it suggests is a property of the linearization rather than of the system — under (25)'s $-L \ln L$ term required c decreases in depth instead. Evaluated per instance against each instance's own measured margin, (27) reproduces the 1,027 values at a median relative error of 8.6% and $R^2 = 0.965$.

d) The denominator is bounded, and the bound is what makes the margin the transferable quantity.: Under the convention $B = 2w_{\text{max}}$ of Section II-D, $\rho = \mu/B \leq w_{\text{max}}/(2w_{\text{max}}) = \frac{1}{2}$, writing μ for the optimal path's mean edge weight, with equality exactly when every edge of that path carries w_{max} . Hence

$$\text{margin} = (L + 1)\bar{\alpha} = L + 1 - 2\rho L \in [1, L + 1], \quad (28)$$

the floor attained on uniform-weight graphs at every depth and the ceiling approached as the path's weight vanishes. So the spread in c_{req} that the families exhibit is not open-ended: at fixed multiplicity the weight distribution can inflate the requirement by at most a factor of $L + 1$, and ρ is the coordinate along that interval. **This is also why the lattice family behaves as it does.** At $\rho = 0.430$ it sits nearest the $\frac{1}{2}$ ceiling of any family here, so its margin sits nearest the floor — $0.140L + 1$, against the sparse family's $0.762L + 1$ — and required c is correspondingly nearest the full log-multiplicity. That is the mechanism behind the 0 of 360 reported below, which the depth reading leaves unexplained.

Two consequences of (28) are worth stating because they bear on the convention itself. Positivity of the margin at depth L requires $B > 2\mu L/(L + 1)$, which in the worst case $\mu = w_{\text{max}}$ is $B > 2w_{\text{max}}L/(L + 1)$; the supremum of that requirement over L is exactly $2w_{\text{max}}$, so $B = 2w_{\text{max}}$ **is the smallest constant multiple of w_{max} under which discrimination is possible at every depth.** And admissibility alone is not enough: at $B = 1.5w_{\text{max}}$ on an unweighted graph of depth 8 the margin is -1.67 , and at $B = 1.1w_{\text{max}}$ it is -5.55 — **negative, so the discrimination condition is unsatisfiable at any c ,** and the design fails for a reason that is not visible anywhere in the ordering condition. Section VII-B takes up what the choice costs on the other side.

e) Depth alone does not transfer across families; margin-per-hop does.: At $L \approx 14$ the lattice family requires 13.57 while the terrain family requires 4.30 — **a factor of 3.2 at the same depth**, and a depth law fitted on the other families predicts 4.01 for the lattice against an observed 13.57. Under (27), with the numerator fitted on the other three families only, each held-out family is recovered at ratios of 1.16, 0.87, 0.84 and 1.08 for sparse, dense, lattice and terrain — the interaction form reaching 1.17 at its worst on the same test, which is the same accuracy. **What is family-independent is the 2.289 nats per hop the multiplicity charges; the entire spread between topologies is carried by the margin**, through ρ — median 0.119, 0.076, 0.430 and 0.105 in the same order. **Quoting a depth coefficient across families is wrong by a factor of two to three, and quoting the ratio is not.**

f) The weight convention is a design decision, not a normalization — and the law predicts its effect before the measurement.: Re-drawing the sparse family with edge weights on $U[1, 2]$ raises ρ from 0.119 to ≈ 0.36 , so $(1 - 2\rho)$ falls from 0.76 to ≈ 0.29 and (27) requires c_{req} to roughly double. Measured over 119 instances at three sizes it does: ratios of 1.95, 1.95 and 1.89, and **not one of the 119 clears at $c = 4$.** Predicting them from the

numerator fit above — **which never saw this weight convention** — gives a median relative error of 7.5 %, the same accuracy it achieves inside the population it was fitted on, against 11.8 % for the interaction fit on the same instances. What carries the prediction is that the numerator is a count of microstates and does not see the weights at all: the whole effect of the convention is in ρ , and (27) therefore predicts it without having been shown it.

g) *What this says about $c = 4$, which is where the rest of this paper operates.:* **Across the whole population, 13.4 % of instances clear the condition at $c = 4$** in the microstate convention, at $B = 2 w_{\max}$. That number needs three qualifiers and not one, on top of the normalization just named. *Family:* the lattice family clears on none of 360 instances, while the terrain family clears on 35.1 %. *Size:* clearing concentrates in small graphs, and in the sparse family at $N = 60$ — the size at which this construction is customarily run — **one instance in forty clears**, with none of thirty-nine in the dense arm. *Convention:* in the looser edge-set convention 40.0 % clear, and **the two conventions disagree on 273 of the 1,027 instances**, so a statement about $c = 4$ that does not name its convention can be reversed by choosing the other.

Since the computed c is a *lower* bound on what would be required, all of this is robust in one direction only: instances that fail to clear cannot be rescued by counting more competitors. **It does not say identification is impossible at $c = 4$** — the construction as published runs *with* an inhibitory mechanism [8], which this bare-network condition does not model. But $c = 4$ cannot be described as sitting above the discrimination threshold at the sizes and families this work uses, and we do not describe it that way anywhere below.

C. How the counts are computed, and how far the method goes

Both counts are exact, and their costs bind on opposite resources — which sets the reach of everything above.

$\#path$ is an exhaustive depth-first enumeration of simple s – t paths at the matched depth, in exact integer arithmetic. Its cost grows with the branching factor raised to L , so **it is bound by depth and by time**. $\#scatter$ is a frontier dynamic program over an elimination ordering, carrying the matching polynomial; its state space grows with the frontier width, so **it is bound by density and by memory**. Which of the two binds flips between families: the random families die on the dynamic program’s memory — beyond $N \approx 140$ sparse and $N \approx 100$ dense at a 6 GB cap, because a $G(n, p)$ frontier grows as $\Theta(N)$ — while the lattice and terrain families die on the path enumeration’s time. **A lattice is the cheapest family for the dynamic program**, because a $k \times k$ lattice’s frontier is exactly $\sqrt{N}$, and the hardest for the path count, because its optimum depth grows as $2(\sqrt{N} - 1)$.

One implementation hazard is worth reporting, because it failed silently and because the paper’s numbers sit close to it. A packed representation that carries the whole matching polynomial in fixed-width coefficient fields is exact only while every coefficient stays below the field width; above it, a coefficient overflows and the one $\#scatter$ is read from is masked, returning a plausible integer with no error raised. **The error runs in the dangerous direction** — $\#scatter$ too small makes c_{req} too small, so the threshold appears easier to meet than it is. We size the field width from a provable bound, $m(G, k) \leq \binom{|E|}{k}$, and raise rather than return when it cannot be met; every count records its top-coefficient bit length so a future crossing is visible in the data rather than only in an assertion.

The scale statement, in the one form that is checkable. A packed count at width W is exact if and only if $\max_k m(G, k) < 2^W$. On the lattice family the top coefficient grows by about twelve bits per rung of k — 12, 20, 30, 41, 53, 65, 78, 90, 103 for $k = 4 \dots 12$ — so a 96-bit field is exact through $k = 11$ and fails at $k = 12$. Every instance behind the numbers in this paper sits at least 51 bits below that ceiling, a factor of 2^{51} , and 49 of the 1,080 instances in the population above — all of them in the two deepest lattice and terrain cells — would have been silently wrong without the sized field. **The whole population cost about ninety core-minutes**, so the reach of this method is set by memory on dense instances and by depth on sparse ones, not by aggregate compute.

D. The discrimination condition

Putting (24) and (25) together gives the second of the paper’s two design conditions. The path sector carries more aggregate probability mass than the equal-activity scatter sector only if the margin exceeds the log-multiplicity of the competing family,

$$c(L+1)\bar{\alpha} > \ln\left(\frac{\#scatter}{\#path}\right), \quad (29)$$

$$\text{equivalently } c > \frac{L}{L+1} \cdot \frac{\ln(|E|e/(Lb))}{\bar{\alpha}}.$$

We state (29) as a design condition rather than as a proposition for a reason that has nothing to do with its left-hand side — (24) is verified against compiled networks in Section V-A — and everything to do with the family on its right.

Each term has a plain reading. The left side is the energy the construction charges for continuity, accumulated over the $L + 1$ traversed vertices of the augmented path at an average α weight $\bar{\alpha}$, which (24) shows to be one minus twice the path's normalized weight per traversed vertex — so light paths earn a larger margin than heavy ones, as intended. The right side is the log-multiplicity of the look-alike family: the number of nats of entropy the margin has to buy back.

Two features of (29) deserve emphasis, and neither depends on the numerical value of either side. It constrains c and nothing else: by the cancellation (23) neither b_k nor D nor I appears anywhere in it. And it is logically independent of the length-bias condition $\lambda = 0$ of Section IV-B, which constrains the combination $2b_k + D + c$. The two conditions address different questions — given that a configuration is a path, which path is cheapest; and given an activity budget, is a path the cheapest way to spend it — and a complete parameterization must satisfy both:

$$\boxed{\begin{aligned} \lambda = 2b_k + D + c = 0, \quad b_k &= -\frac{D + c}{2}, \\ c &> \frac{\ln(\#\text{scatter}/\#\text{path})}{(L + 1)\bar{\alpha}}. \end{aligned}} \quad (30)$$

Because c appears in both, they are solved in order: choose c from the discrimination condition, then set the bias from (17).

a) What (29) does and does not constrain.: The condition is stated *at fixed activity*, and that restriction is forced rather than chosen. A path of L hops carries $L + 2$ β pairs, so only L -hop paths are size-matched against a scatter competitor of size $L + 2$; restricting the competitor restricts the target. Accordingly $\#\text{path}$ counts simple s - t paths at the matched depth, and (29) compares two families that hold the same number of neurons active. It is a statement about a conditional comparison, not a marginal one.

b) The matchings are one end of a graded family, and (29) evaluates that end.: This is the most important limit on (29), and it is sharper than a list of families we did not cost, because the competitors at matched activity can be characterized completely.

A configuration at matched activity induces $m = L + 2$ edges on the augmented graph, the two auxiliary edges being admitted to the competitor set as in Section V-A — S and T are single neurons held active by the terminal drive, so they contribute to the activity budget whether or not their edges are induced, and admitting the edges keeps m matched across the whole family. A vertex of degree three or more in that edge set forces two winners into one slot and engages a γ coupling, which the convention $|I| \gg |D|$ makes prohibitive; so the admissible configurations are those in which every vertex has degree at most two — **disjoint unions of paths, together with cycles**. Cycles have no endpoints, so every one of their vertices has degree two. For a configuration of m edges with k_p path components and any number of cycles, the vertices of degree two — which are exactly the traversals, and exactly the active α pairs — therefore number

$$|\alpha| = m - k_p, \quad (31)$$

independently of the cycle count. The natural index is accordingly not the component count but the α deficit relative to the path,

$$d = (L + 1) - |\alpha| = k_p - 1, \quad (32)$$

One point about the range of d decides how much of the space this grading governs, and it is easy to get wrong. S and T are single neurons rather than clusters, and the terminal drive acts on their membrane potentials rather than entering the energy, so *a configuration in which both are inactive is neither forbidden nor penalised by (2)*. Matched activity therefore does not force the auxiliary edges into the induced set, k_p may be zero, and

$$d \in \{-1, 0, 1, \dots, m - 1\}. \quad (33)$$

The familiar cases are the interior and the upper end: $d = 0$ contains the least-weight s - t path, with $|\alpha| = m - 1 = L + 1$, which is Lemma 1(i); $d = m - 1$ is the matching, with $|\alpha| = 0$; a path plus a detached edge is $d = 1$ and

a broken chain is $d = 2$, so **the near-misses are not a separate species but the interior of the same grading. The lower end, $d = -1$, has no counterpart among the near-misses and is the subject of Section V-E.**

That grading fixes what the general condition looks like. A configuration at deficit d carries d fewer α pairs than the path and therefore sits roughly $c d \bar{\alpha}$ above it in energy, so writing N_d for the number of microstates at deficit d , the path family outweighs every sector *above* it when

$$c > \max_{d \geq 1} \frac{\ln(N_d / \# \text{path})}{d \bar{\alpha}}. \quad (34)$$

The denominator is $\# \text{path}$ and not the whole $d = 0$ class, and the distinction is not cosmetic. Section V-E shows that class to be overwhelmingly *not* paths; dividing by it would measure against a family the decoder rejects, and would understate the requirement by 61 % on the reference instance. Every comparison in this paper is against the configurations a decoder accepts.

Condition (29) is then exactly the $d = m - 1$ term of that maximum, since $d = m - 1$ is $L + 1$, the factor (29) carries — and evaluating the maximum returns 3.699 at that term, reproducing the reference crossing of Section V-B to four digits. **But it is not the binding term, and we have now measured which is.** Counting every matched-activity configuration by deficit with a degree-two frontier recursion, the maximum falls monotonically in d — 18.11, 11.25, 8.68, 7.22, 6.23, 5.46, 4.82, 4.25, 3.70 for $d = 1 \dots 9$. **The near-path sector binds, not the scatter sector** — a configuration one component short of a path is rare enough to be cheap and close enough to be numerous, and it is charged only a single $\bar{\alpha}$ rather than $L + 1$ of them. So the published condition understates the requirement by a factor of *five* here, in the direction Section V-B already argued every omission must cut. **What we have replaced is the uncertainty’s shape: the competing space is now delimited rather than open, and the omission is one index of a max rather than an unbounded collection of families.**

One difference between (34) and (29) should be stated plainly: the general form is heuristic where the special case is exact. Here $\bar{\alpha}$ is the *path’s* mean α weight, defined at (24). At $d = m - 1$ the competitor retains no α pair at all, so the entire path α sum is the gap and the substitution is an identity — which is why (23) was an exact cancellation. At intermediate d it assumes the competitor’s surviving pairs carry the path’s mean weight, and since $f = 1 - (w_j + w_k)/B$ is larger on light edges, a competitor whose traversals sit on light edges is more competitive than (34) prices it. The error is in the direction that has been consistent throughout: it can only make the true requirement larger.

One consequence of (31) is worth stating because it is a limit the edge-set picture hides: $d = 0$ is not the path family alone. A shorter s - t path Q carrying $p < m$ edges together with disjoint cycles carrying the remaining $m - p$ has $k_p = 1$, hence $|\alpha| = m - 1$ and the same activity as the optimum. **The deficit that (34) prices is therefore exactly zero for this family, and the mechanism by which that condition suppresses competitors is inoperative on it.**

What survives is the weight term. Summing (6) over the degree-two vertices of such a configuration — each cycle contributing every one of its vertices, the path component all but its two ends, and the zero-weight auxiliary edges contributing nothing — gives $\sum_{\alpha} f = (m - 1) - (2/B)(W_Q + W_{\text{cyc}})$, so against the optimum

$$E_N(x) - E_N(x_P) = \frac{2c}{B} (W_Q + W_{\text{cyc}} - W^*) > 0, \quad (35)$$

strictly, since Q is an s - t path so $W_Q \geq W^*$ and $W_{\text{cyc}} > 0$. **Once the auxiliary edges are not forced the class is far larger than that.** It is dominated by configurations using neither terminal — a path component among the graph’s own vertices, plus cycles — which cover two more clusters than the optimum and therefore carry four times its microstate weight. Counting the class exactly on the reference instance gives $N'_0 = 1.239 \times 10^{11}$ against $\# \text{path} = 898,048$, a ratio of some 138,000.

Carrying the same α sum through for an arbitrary member — whose path component may end on real edges rather than on the zero-weight auxiliaries — gives the gap in the general form

$$E_N(x) - E_N(x_P) = \frac{2c}{B} \delta(x), \quad \delta(x) = W_x - W^* - \frac{1}{2}(w_a + w_b), \quad (36)$$

with W_x the configuration’s total weight and w_a, w_b its path component’s two end edges. **Every member of the class is above the path, and it takes two arguments to see it.** A member carrying a cycle is above because any cycle weighs at least the weighted girth, 0.7553 on this instance, against $W^* = 0.5488$. A member that is a

single path is not covered by that argument, and we settled it by exhaustive search over the class: the least δ over all pure-path members is $\delta_{\min} = 0.042972$, so those too sit above — by 0.173 in energy at $c = 4$, the closest any competitor comes to the optimum. It is attained by a path anchored at *one* terminal and running nine real edges to an ordinary vertex, and the deposit carries the configuration so the value can be regenerated rather than taken on trust.

The third condition is therefore

$$c > \frac{B}{2} \frac{\ln(N'_0/\#\text{path})}{\delta_{\min}}, \quad (37)$$

which on the reference instance requires $c > 273.5$. **The binding gap is set by a light path, not a light cycle**, and the distinction is worth twenty-fold: substituting the girth for $\delta_{\min}$ — the natural guess, and ours until we measured it — overstates the gap by 17.6 and understates the requirement by the same. **The $d = 0$ sector is not a favorable omission but a binding condition, and the operating point misses it by a factor of about seventy.**

E. Configurations below the path, and what excludes them

The lower end of (33) is worse than a binding condition, and it is the sharpest statement this paper can make about why the construction needs more than an energy function.

At $d = -1$ there is no path component at all: the induced set is a union of disjoint cycles on m vertices, every one of them a traversal, so $|\alpha| = m = L + 2$ — **one more α pair than the optimum carries**. Activity is $2m$ and matches. Repeating the calculation that gave (35),

$$E_N(x) - E_N(x_P) = -c \left[1 + \frac{2(W^* - W_{\text{cyc}})}{B} \right], \quad (38)$$

which is *negative* — the cycle set cheaper than the shortest path — whenever

$$W_{\text{cyc}} < \frac{B}{2} + W^* = w_{\max} + W^*, \quad (39)$$

using $B = 2w_{\max}$. **And the gap in (38) is proportional to c with the wrong sign: raising the continuity scale makes these configurations relatively cheaper, without bound.** No choice of c suppresses them, because they are not competing with the path on multiplicity against a margin — they are below it in the very term the margin is built from.

This is not hypothetical. On the reference instance the lightest ten-cycle weighs 1.2239 against a threshold of 1.5420; read off a compiled network its energy at $c = 4$ is 4.929 against the path's 6.210. Decomposing both states confirms the mechanism exactly: the bias, consistency and inhibition terms agree to the digit and the entire difference is one α pair. **The shortest-path configuration is not the minimum of the matched-activity space**, and (29) — which compares it only against the matching sector — cannot see this.

a) How often, and the certificate that decides it.: Because (39) is a statement about weights alone, absence can be certified without any search: **if the m lightest edges of G already sum to at least $w_{\max} + W^*$, no admissible cycle set exists**, at a cost of one sort. **We therefore ran the absence test over the entire population and the presence search over a stratified subsample**, the two having very different costs. The subsample takes three sizes per family spanning the population's range, twelve seeds each — sizes 20, 40 and 60 for the three irregular families and side 5, 7 and 9 for the lattice — giving 36 instances per family and 144 in all.

Across all 1,080 instances the sector is provably absent on 198, and every one of them is a lattice. The certificate fires on 55.0% of the 360 lattice instances and on *none* of the 240 sparse, 200 dense or 280 terrain instances — which is (41) at population scale, since the lattice is the only family whose ρ reaches the regime in which absence can be established at all. Over the 144-instance subsample the sector is *present* on 96, provably *absent* on 25, and undetermined on 23, which we report rather than impute. The split by family is the informative part: present on every sparse and dense instance sampled, on two thirds of the terrain instances — with the deepest gaps, a median of -8.1 at $c = 4$ — and provably absent on the lattice, whose near-uniform weights leave even its lightest edges too heavy. **Because the undetermined cases are neither confirmed nor excluded, the presence rate over the subsample is bounded rather than known: it lies in [67%, 83%],** and we quote that interval wherever the rate is used rather than the point estimate its lower end would suggest.

b) *The certificate is a condition on ρ alone, and the depth cancels out of it.*: Write $\bar{w}$ for the mean of the m lightest edges and $\gamma = \bar{w}/\mu$ for its ratio to the optimum's mean edge weight, so γ measures how much lighter the graph's light tail is than the path that uses it. Since $\rho = \mu/2w_{\max}$, the certificate $m\bar{w} \geq w_{\max} + W^*$ rearranges *identically* — not approximately; we checked the two forms agree on every instance of the subsample — into

$$\gamma \geq \frac{L + 1/(2\rho)}{L + 2}. \quad (40)$$

The right-hand side equals 1 when $L + 1/(2\rho) = L + 2$, that is at $\rho = \frac{1}{4}$, **and the depth cancels: the boundary sits at $\rho = \frac{1}{4}$ at every L** . Measured across the subsample $\gamma \leq 1$ on all 144 instances — at most 0.71 on the three irregular families and 0.93 to 0.98 on the lattice, whose weights are near-uniform by construction — so on this evidence:

$$\rho < \frac{1}{4} \implies (40) \text{ cannot be satisfied at any } \gamma \leq 1. \quad (41)$$

Below a quarter the sector cannot be certified absent at all, whatever the graph: the only route to a proof of absence is closed off. That is a statement about what can be established, not about what is there — (40) is sufficient for absence and not necessary, and its failure exhibits no cycle set. A graph whose light edges happen to form a forest carries no light cycle at any ρ . **What we can say is empirical and it is strong enough**: below the boundary the sector was present on every sparse and dense instance we searched, at $\rho = 0.119$ and 0.076. Above a quarter absence becomes provable without becoming assured — the condition is necessary and not sufficient, and a handful of sparse instances reaching $\rho \approx 0.35$ still carry the sector because their γ is too small. *That $\gamma \leq 1$ is an observation on this population rather than a theorem* — an optimum drawn entirely from the graph's lightest edges would violate it — and we state it as measured.

c) *This is the fourth coupling between the paper's design parameters, and the tightest.*: The margin is $(1 - 2\rho)L + 1$, so the very ρ that buys freedom from the cheaper-cycle sector drives the discrimination margin toward its floor: at $\rho = \frac{1}{4}$ the margin is at most $L/2 + 1$, and as $\rho \rightarrow \frac{1}{2}$ it reaches the floor of 1 that (28) sets. **The weight distribution cannot buy both**. The lattice family is the demonstration and it inverts its own result: at $\rho = 0.430$ and $\gamma \approx 0.95$ it is the one family provably free of a sector no c suppresses, *and* it clears the discrimination condition on none of 360 instances — the sharpest negative in this paper. The terrain family is the mirror image, clearing most often at 35.1 % and carrying the deepest gaps below the path. **A designer choosing weights is choosing which of the two failures to take**, and (40) says where the boundary is without any search.

d) *What excludes them is the drive, and this is the third time that pattern appears.*: A $d = -1$ configuration uses neither terminal. The additive drive holds S and T firing, and a state in which both are silent is not one the driven network dwells in — so the sector is removed from the accessible ensemble by a mechanism that never appears in the energy function at all. That is the same argument Proposition 3 makes for the all-silent configuration, extended from a single state at zero activity to a populated sector *at the matched activity where (29) operates*, which is the more damaging place for it to sit. Section I states that the least-weight path is a conditional minimiser and that the drive is what makes the condition hold; **this is the quantitative form of that claim, and it says the drive is doing more work than the ordering and discrimination conditions together**. Two auxiliary vertices, a terminal bias and an additive drive: three components a reader would take for bookkeeping, each of them load-bearing.

We do not claim the drive is sufficient. What it removes is a sector of the *static* energy landscape by acting on the *dynamics*, and Section VII-D is explicit that nothing here speaks to reachability. **The honest statement is that the energy function alone does not put the shortest path at the bottom of the matched-activity space, on the majority of instances we sampled, at any c** — and that what rescues the construction is not a condition on the couplings but a mechanism outside them.

Every omission inside the grading cuts in the favorable direction. (29) is one term of the maximum (34), so it can only understate the requirement; the same holds for the second omission, that the count is taken at matched activity while an unconstrained sampler runs far hotter than a path requires, so the space actually explored is larger still; and the same holds for the $\bar{\alpha}$ substitution just described. A larger competing family means a larger log-multiplicity, hence a *larger* required c . **Every competitor at nonzero deficit that we have failed to count pushes the requirement up, never down** — so any finding of the form “this parameterization does not clear the condition” survives those omissions intact, and it is a finding of exactly that form that Section V-B reports.

The sectors at $d \leq 0$ are the exception and we keep them outside that statement. Neither is a term of (34): at $d = 0$ the maximum has no term to give, and at $d = -1$ the sign of the gap reverses so that raising c makes

matters worse. Both are now measured rather than open — (37) binds at $c > 273.5$ on the reference instance and (39) is satisfiable on between 67 and 83% of the instances sampled. **Everything asserted about direction in this paper is asserted about the graded competitors of (34), at $d \geq 1$;** what happens below that is Section V-E.

e) The grading is exhaustive, and that is worth stating as a claim rather than leaving to be inferred.: A referee tracking this construction across drafts sees a binding requirement that has moved — 3.699 against the scatter sector, then 18.11 once the maximum was evaluated, then 273.5 once the $d = 0$ class was counted — each step because a sector was found that the previous accounting had not priced. **The natural question is whether that sequence has converged, and it has, for a reason available from the classification itself.** Every matched-activity configuration engaging no γ coupling has maximum degree two, so it is a disjoint union of paths and cycles and nothing else; (31) assigns each one a value of k_p , and (33) bounds $d = k_p - 1$ between -1 and $m - 1$. **Those indices exhaust the space.** There is no further sector to find, because there is no configuration the grading fails to place — what remains open is the *size* of each N_d on instances we have not counted, not whether some further family exists. The sequence above is not an unconverged series of discoveries; it is one classification, evaluated term by term, in the order the terms happened to become computable.

This matters for how the condition may be tested, because the natural empirical proxy does not test it. That proxy — the fraction of steps at which the induced subgraph is *any* simple s - t path, at any depth — is a marginal over every activity level the network visits, not a comparison at one. The two quantities are not interchangeable, and no measured rate of that kind, however precise, either confirms or refutes (29). Widening #path to all depths without correspondingly widening the competitor would lower the apparent threshold, which is the wrong direction and a further reason to keep the matched form. A genuinely marginal version of the comparison would have to sum over all activity levels, and would then be dominated by the resting sector at order $2N$ rather than by the path sector at $2(L + 2)$ — which is the same $\Theta(N)$ obstruction described in Section V-F, and the reason the direct mass-ratio measurement reported there returned no usable estimate.

F. What the measurements show

Empirically it is the second condition that carries the effect. A sweep over the (c, λ) grid — 8 deep graphs with $L \geq 8$ at $N = 60$, $p = 0.05$, three network seeds per graph, 10,000 steps per run, $D = 4$ and $b_k = (\lambda - D - c)/2$, on the network of Section III at the terminal bias adopted there — gives, along the $\lambda = 0$ column, an enrichment of optimal-path containment over chance of 0, 12 and 80 at $c = 1, 2, 3$, on 0, 31 and 426 containment events against chance expectations of 3.24, 2.50 and 5.35.

The chance baseline is density ^{L} in the *original* hop count, and that choice moves the answer by more than an order of magnitude. The two auxiliary edges are induced whenever the terminals are driven, so they carry no chance of failure; pricing them as though they did — taking the exponent to be the augmented $L + 2$ — understates the baseline and inflates the same three figures to 0, 198 and 1008. We quote the conservative form throughout. On the unaugmented variant of Section IV-E the same design gives 0.88, 41.2 and 330. **Enrichment is a ratio against a density-dependent baseline and the two networks do not run at the same density, so we report both and claim no improvement from the augmentation on this metric;** what the comparison is being asked to establish is the shape of the dependence on c , and that is common to both.

The cell $c = 1$, $\lambda = 0$ is the informative one: it holds the length-bias condition exactly while leaving the continuity coupling at its base scale, and across its 240,000 scored steps it contains the optimum *not once*. That is a measured zero rather than a small number, so the entry is an absence of the effect and not a weak instance of it. The length-bias condition alone recovers nothing; the continuity scale is what moves the network into the path sector. This is the qualitative prediction of (29) and is what that condition is being asked to explain.

Selection is already strong at $c = 2$. Where the crossing predicted by (29) lies on this population is not a number we quote, for the reason given in Section V-B; and even with that value in hand, the sweep would not test it directly, because the two statements concern different quantities. Enrichment is a per-step containment rate measured against density ^{L} , whereas (29) compares aggregate mass in two sectors, and there is no reason for the two to cross together. Establishing the threshold as such calls for measuring the mass ratio directly.

We attempted that measurement, and its failure characterizes the state space in a way a successful estimate would not have. At $N = 12$, $c = 2.0$, $b_k = -3.0$ and $\lambda = 0$, over 72 independent units and 143,280,000 scored steps, we accumulated path and scatter mass restricted to the matched-activity shell that (29) compares. **The shell was**

occupied for 2,116 steps in 143,280,000 — a fraction of 1.5×10^{-5} — and path mass and scatter mass were identically zero in all 72 units. There is no ratio to report and no sample size to attach to one.

The reason is structural, and it says something useful about the construction. The γ couplings elect at most one winner per slot, but nothing in the design silences a slot outright, so the network’s resting activity sits at the order of the slot count $2N$ — here the measured mean activity runs 25.0 to 43.5 against $2N = 24$, at or above one active neuron per slot. The path sector on that network sits at $2L$, the run having been made on the variant of Section IV-E, so Lemma 1(ii) supplies its occupancy. Reaching matched activity therefore asks the sampler for a $\Theta(N)$ downward excursion in a variable whose fluctuations are $O(\sqrt{N})$, and a free-running chain does not make that excursion at any budget one can buy. The obstruction is a property of the dynamics, not of the compute, and it is unchanged by the offset between the two count sets, since $2(L + 2)$ and $2L$ are equally far below $2N$ at the sizes in question.

Two things are therefore recorded about the discrimination condition and kept separate. Its numerical value on a given instance is a derivation, computed under (24) at the matched size $L + 2$ in Section V-B and found to span an order of magnitude across 1,027 instances, which is why we state no instance-independent threshold. And confirming any such threshold by free-running sampling is structurally obstructed rather than merely expensive. The second point stands on its own and is itself a useful characterization of the state space: it locates the discrimination condition in a region these dynamics occupy with vanishing frequency, which is exactly the sort of fact one wants before designing an experiment. A direct test needs importance sampling or an activity-constrained ensemble rather than free running.

VI. AN $O(|E|)$ FORMULATION

A. Why scale is part of the question

The conditions above are statements about behavior at depth L and size N jointly, and cannot be tested inside a range of sizes narrow enough that size cannot vary. Earlier experimental work with this construction reached $N = 90$ vertices. Implementation efficiency is therefore a precondition for the measurements rather than a later optimization, and we treated it as one.

B. The decomposition

A direct transcription of the membrane update (1) stores an explicit weight matrix W with $\sum_i (2 \deg_i)^2$ nonzeros and performs one sparse matrix–vector product per timestep. That product is what caps the construction at $N = 90$: the intra-cluster block is dense within each cluster, so cost grows with the square of the degree.

The structure that removes it is a separability property of the continuity weight. Reading (6) again,

$$f = 1 - \frac{w_j + w_k}{B}, \quad (42)$$

the two weight-dependent terms depend on the presynaptic and the postsynaptic neuron *independently*. Nothing couples the identity of the two neurons except through their own edge weights. Consequently a neuron’s total α input depends on the rest of its cluster only through two scalar reductions, taken once per step over each *segment* $s = 2 \cdot \text{cluster} + \text{slot}$, of which there are $2|V|$:

$$\begin{aligned} A[s] &= \#\{\text{active neurons in segment } s\}, \\ Sw[s] &= \sum_{\text{active } k \in s} w_k. \end{aligned} \quad (43)$$

In terms of these aggregates the α input to neuron k from the opposite slot of its own cluster is available in closed form,

$$\begin{aligned} \sum_{l \in \text{opp. slot}} x_l f(w_k, w_l) &= \left(1 - \frac{w_k}{B}\right) A_{\text{opp}} - \frac{Sw_{\text{opp}}}{B} \\ &\quad - x_{\text{partner}} \left(1 - \frac{2w_k}{B}\right), \end{aligned} \quad (44)$$

the final term removing the same-index partner, which the α class excludes and the γ class handles instead. Every γ contribution follows from the same two reductions, and the β contribution is a fixed gather over graph edges, one

TABLE VI
MEASURED SCALING OF THE $O(|E|)$ SIMULATOR: ONE CORE, PURE NUMPY, PEAK RESIDENT MEMORY 417 MB. THE “ W WOULD NEED” COLUMN IS THE MEMORY AN EXPLICIT WEIGHT MATRIX WOULD HAVE REQUIRED AT THAT SIZE. THE $n = 90$ ROW IS THE PREVIOUSLY REACHABLE CEILING.

n	$ E $	neurons	W would need	1000 steps
90	892	3,568	1 MB	0.1 s
250	6,436	25,744	16 MB	1.4 s
700	49,533	198,132	325 MB	17 s
1200	144,880	579,520	1.6 GB	56 s
3000	901,758	3,607,032	25 GB	~8 min

per edge, with no dependence on cluster degree at all. The dense intra-cluster block is therefore never materialised: time and memory fall from $O(Nd^2)$ to $O(|E|)$. A pleasant by-product is that both energy components of (7) are functions of the same aggregates (43), so tracking E_{NC} and E_{SPC} at every step costs essentially nothing, where in the matrix formulation energy evaluation was itself a bottleneck.

C. Validation and scaling

Correctness of (44) was checked against the object it replaces. An explicit dense W was constructed and the decomposed membrane potential compared with the matrix–vector product over random configurations; the maximum per-neuron discrepancy is 1.99×10^{-7} in `float32`, and it is unchanged at 1.99×10^{-7} after the α term is rescaled by c , as the discrimination condition of Section V requires. The two formulations agree to floating-point resolution.

Table VI gives the measured scaling, on one CPU core in pure NumPy, with peak resident memory of 417 MB throughout. The matrix formulation meets a memory wall at roughly $n = 1200$ – 2000 on ordinary hardware; the decomposed simulator stays within a few hundred megabytes to $N = 3000$, where the network has 3,607,032 neurons.

These timings were taken without repetitions and on harnesses later found to be oversubscribed, so they establish the order of magnitude and the memory behavior rather than serving as a benchmark.

VII. WHAT THE TWO CONDITIONS IMPLY

A. The two conditions are independent, and only one of them is usually stated

The analysis above separates two requirements that are easily run together. The first is that the energy function rank the right configuration lowest: at $\lambda = 0$ the least-weight path is the minimum of the path sector, and that is the property the construction was designed to have. The second is that the path sector carry more aggregate probability mass than the family of configurations it competes with, which is a statement about multiplicity as well as energy and does not follow from the first.

The second condition is the one that binds. Its content is that the continuity coupling must buy back the log-multiplicity of the scatter family, and (25) shows the trade is mild — the entropy to be overcome grows linearly in depth and only logarithmically in $|E|$, and the margin (24) also grows linearly in depth, so the requirement on c is logarithmic in graph size rather than polynomial. That is a genuinely favorable asymptotic and it is worth stating clearly, because it says the condition does not become a treadmill as instances grow.

What is not favorable is the constant. Enumerated at the matched competitor size over 1,027 instances across four graph families, the required c has a pooled median of 5.82 and spans 1.65 to 19.59, with the requirement rising with size inside every family. **No instance-independent threshold is defensible**, so a design condition of the form “take c above such-and-such” cannot be quoted honestly; what can be quoted is (27), evaluated per instance.

And the shape of that dependence is not the one we expected. A linear fit of required c against depth returns slopes of opposite sign in different families — $+0.81$ on lattices, -0.76 on dense random graphs — because c_{req} is a ratio in which only the numerator behaves as the asymptotics predict. Depth enters the log-multiplicity linearly at a family-independent 2.289 nats per hop and size only logarithmically, exactly as (25) says; the entire spread between topologies is carried by the margin, through the path’s mean edge weight. **This is the paper’s**

most transferable empirical result, and we mean transferable in the demonstrated sense: one three-parameter numerator over each instance’s own measured margin reproduces 1,027 required- c values at 8.6 % median relative error across four topologies, recovers each family when that family is withheld from the fit, and predicts a weight convention it never saw at 7.5 % — **beating the better-fitting interaction form of Section V-B on both tests that change the weight regime, and tying it on the family hold-out.**

Two things must be said about using it as a rule. The first is what it is evaluated from. (27) takes L and ρ , and both are properties of the *optimal* path — so a designer computes them classically, by Dijkstra’s algorithm, before compiling the network. That is consistent with the framing of Section I, where the interest is stated to be not in competing with Dijkstra’s algorithm but in what a sampling network does with a problem whose answer is known, but it should be explicit: **this is a rule for choosing parameters on an instance whose answer is already known**, which is the regime in which one studies what a sampling network does at all.

The second is conditioning, and it is not uniform. Differentiating (27),

$$\frac{d \ln c_{\text{req}}}{dL} = \frac{\kappa_L}{\kappa_L L + \kappa_E \ln |E| + \kappa_0} - \frac{1 - 2\rho}{(1 - 2\rho)L + 1}, \quad (45)$$

so a one-hop error in L moves c_{req} by 4.6 % on the reference instance ($|E| = 142$, $L = 8$), taking the two directions separately rather than averaging them. **That sensitivity is not uniform in depth, and it is the one place the linearization’s coarseness is visible in the rule’s behavior rather than only in its fit:** the same perturbation on the same instance moves c_{req} by 18.6 % at $L = 3$ and 30.8 % at $L = 2$, because $\kappa_L L$ is a shrinking share of a numerator whose remaining terms do not fall with depth. The interaction form of Section V-B is flat at 3.9–5.3 % across the same range. **A designer working at very shallow depth should evaluate the numerator both ways;** over the depths at which most of this population sits, the two agree. A 20 % error in ρ , by contrast, moves c_{req} by 4–5 % at $\rho \approx 0.11$ but by a factor of 5.4 at the lattice family’s $\rho = 0.430$ and its depth of 14. The reason is visible in the denominator: $(1 - 2\rho) \rightarrow 0$ as $\rho \rightarrow \frac{1}{2}$, so **the rule is well conditioned in depth everywhere and ill conditioned in mean edge weight exactly where required c is largest.** A designer working on heavy-weighted instances should evaluate ρ carefully; one working on light-weighted instances need not.

And it returns a floor, not a target — now measured, and the gap is large. (29) prices the scatter sector, which Section V-D shows to be the *smallest* term of the maximum (34). On the reference instance (27) returns 3.93 against a binding $d = 1$ requirement of 18.11 and a $d = 0$ requirement of 273.5 — **a factor of five and a factor of seventy respectively.** So c_{req} as computed here is a lower bound in a strong sense: **a design that clears (27) has satisfied the scatter condition and neither of the others.** We report it because it is the term with a closed-form asymptotic and a transferable law, and because the direction is signed — but it should not be read as the continuity scale an instance needs.

B. The construction is normally operated below its own discrimination threshold

That last observation deserves more than a clause, because it is the most consequential empirical statement in this paper and it is easy to read past.

Across 1,027 instances spanning four graph families and a range of sizes, 13.4 % clear the discrimination condition at $c = 4$ in the microstate convention and at $B = 2 w_{\text{max}}$ — and $c = 4$ is not an arbitrary probe, but the value at which this construction is customarily run.

That number carries four qualifiers, and dropping any one of them makes it wrong. *Family:* the lattice family clears on none of 360 instances, the spatially embedded family on 35.1 %. *Size:* clearing concentrates in small instances, and at $N = 60$ in sparse $G(n, p)$ — the size this construction is usually run at — **one instance in forty clears**, with none of thirty-nine in the dense arm. *Convention:* the looser edge-set count puts the figure at 40.0 % and the two disagree on 273 of the 1,027 instances, which is why no statement about $c = 4$ in this paper appears without its convention named. *Normalization:* the figure is computed at $B = 2 w_{\text{max}}$, and (46) below gives what any other admissible choice would do to it.

Because those four travel together and the directions below travel with them, we state the result once, in full, and refer back to this form rather than restating it:

Result (the discrimination verdict at $c = 4$). Of 1,027 instances in four families, 13.4 % satisfy (29) at $c = 4$, counting competitors as microstates and normalizing weights at $B = 2w_{\max}$; in the sparse family at $N = 60$, the size and family at which this construction is customarily run, 1 of 40 does, and in the lattice family none of 360 does. Three known omissions all push the true requirement *up*: the count is taken at matched activity while a sampler runs hotter, (29) is one term of the maximum (34) rather than the maximum — the maximum is 18.11 at $d = 1$ against (29)’s 3.699 on the reference instance — and the $\bar{\alpha}$ substitution in (34) prices light-edge competitors conservatively. **The verdict is therefore robust in the only direction that could overturn it. Two sectors lie outside the grading entirely and are not answered by any c :** the $d = 0$ class, which imposes (37) at $c > 273.5$, and the $d = -1$ class of Section V-E, which sits *below* the path and is present on 67–83 % of the instances sampled. Both are excluded by the terminal drive rather than by the couplings. Whether inhibition — absent from the bare network analyzed here — reduces the competing family enough to change the answer remains open.

That form is narrower than a universal and stronger than an anecdote. The ten-graph population from which the stricter reading was originally taken is a single cell of this population, and its verdict is reproduced within that cell.

Two things follow, and they pull in different directions.

It does not say the network cannot identify. The condition is derived for the bare network, with no inhibitory mechanism in it, while the construction as published carries one [8]. What inhibition supplies is precisely a reduction in the size of the competing family — it suppresses off-path activity, which is the multiplicity the margin has to buy back. So the honest reading is that inhibition is doing work the encoding alone does not do, and the discrimination condition quantifies how much work that is.

What it does mean for anyone using the construction is that $c = 4$ cannot be described as sitting above threshold, that a practitioner should evaluate the condition on their own instance rather than inherit a customary value, and that raising c is not free — it enters (17) and moves the bias with it. **The two design conditions are not independent knobs, and this is where that bites.**

There is a third knob, and it is the one that looks like a normalization. The figures above are conditional on $B = 2w_{\max}$, and a reader who has followed (27) will see the obvious move: the numerator of c_{req} is a pure count of microstates and contains no B , while the denominator is $L + 1 - 2\mu L/B$ with μ the optimum’s mean edge weight. So **required c falls monotonically as B rises**, approaching $\ln(\#\text{scatter}/\#\text{path})/(L + 1)$ in the limit — a reduction of up to the full factor of $L + 1$ that (28) bounds. Raising B appears to buy discrimination for nothing, and it would be reasonable to ask whether the 13.4 % above is an artifact of the convention.

It is not, because B is charged on the other condition. The ordering tolerance (20) carries a factor $2c/B$, so **the precision to which λ must be held tightens like $1/B$** : doubling B halves the margin of a mistuned bias, on a construction whose tolerance already runs to 1.9×10^{-3} on the reference instance and 1.3×10^{-3} in the terrain family. Section IV-C argues that a design condition holding only on a measure-zero set is of limited use on a substrate with finite parameter precision; enlarging B is exactly a trade of that precision for discrimination margin. The two conditions therefore couple twice — through c , which enters both (17) and (29), and through B , which loosens one condition in the same proportion as it tightens the other. **What the construction offers is a three-way trade among the continuity scale, the precision available in λ , and the weight normalization, not two knobs and a bookkeeping choice.** The lower end of the B range is not a matter of taste either: below $2w_{\max}$ the margin can go negative and no c suffices, which (28) makes precise.

And the upper end can be settled outright, because the sweep is analytic. The numerator of (27) is a count of microstates and contains no B , which enters only through $\rho = \mu/B$. So c_{req} as a function of B needs no re-enumeration: it is $\ln(\#\text{scatter}/\#\text{path})$ divided by $L + 1 - 2\mu L/B$, on counts already computed. Since (28) caps the margin at $L + 1$, **the lever is bounded**, and the limit is reached at

$$c_{\text{req}}(B) \searrow \frac{\ln(\#\text{scatter}/\#\text{path})}{L + 1} = c_{\text{req}}(2w_{\max}) \cdot \frac{(1 - 2\rho)L + 1}{L + 1} \quad (B \rightarrow \infty), \quad (46)$$

so the floor is a per-instance quantity computable from counts already in hand, with no re-enumeration. Evaluated over the whole population, 31.3 % **of the 1,027 instances have a floor at or above 4 — no admissible B clears them at $c = 4$, at any value.** The split by family is the point: 65.5 % of sparse instances and 72.0 % of dense ones are unreachable in this sense, against 16.8 % of terrain and *none* of the lattice. Median floors run 4.58, 4.68,

2.70 and 3.46 for sparse, dense, lattice and terrain; among the instances that are reachable, the median B required to bring c_{req} below 4 is 3.65, 6.57, 4.92 and 2.29 times w_{max} .

That inverts the naive reading of the lever and answers the question it raises. **The 13.4% is not an artifact of the normalization: in the two families at the sizes where this construction is customarily run, most instances are not clearable at $c = 4$ by any admissible B , because the margin is capped by $L + 1$ before the requirement is reached. The lattice family, which clears on none of 360 at $B = 2w_{\text{max}}$, is the one family every instance of which some B could clear** — precisely because its ρ sits far from zero, which is the same property that Section V-E shows frees it from the sector below the path. Raising B helps exactly where the requirement was already largest, and it is charged against λ precision when it does.

Two definitional choices inside that enumeration are worth flagging as choices. Counting the competitor family over microstates rather than edge sets is worth 7.6 nats on the reference instance and moves the required c by +0.89 — and it is what changes the verdict at $c = 4$, where 40.0% of the population clears under the looser convention against 13.4% under the stricter, the two disagreeing on more than a quarter of all instances. Admitting the zero-weight auxiliary edges to the competitor set leaves the margin identical but avoids biasing the threshold downward. **A convention that changes an answer is not a convention**, and both are stated in Section V-B rather than buried in an implementation.

C. The length-bias coefficient as a design parameter

Throughout this paper λ has been treated as a quantity to be set to zero, because zero is the condition under which the network orders paths by weight rather than by hop count. But λ is a free parameter, and a nonzero value is not an error state: it is a network that prefers paths of a particular length scale. A network at $\lambda > 0$ favors routes with more hops; one at $\lambda < 0$ favors fewer, and in the limit collapses toward silence.

That suggests a composition whose components differ in λ rather than merely in the subgraph they are given — a coarse component proposing a route through waypoints, finer ones resolving the segments between, with Proposition 2 setting how precisely each must be held. **We present this as motivation rather than result:** nothing here tests a composed system, and how components at different λ should be coupled is open.

D. What this paper does not establish

Three limits are worth stating plainly, because each marks a place where it would be easy to read more into the analysis than it carries.

First, a correct minimum is not a reachable one, and nothing here speaks to reachability. Every result above is a statement about an energy function, checkable by enumeration on a compiled network and independent of any sampler. Whether the network’s dynamics visit the minimum, and at what rate, is a separate question, and nothing established here should be read as bearing on it.

Second, the discrimination condition has not been confirmed by sampling, and we explain in Section V-D why that is structural rather than a matter of budget. Confirming it directly asks a free-running chain for a $\Theta(N)$ downward excursion in a quantity whose fluctuations are $O(\sqrt{N})$, which does not happen at any budget one can buy. A direct test needs importance sampling or an activity-constrained ensemble. What we report instead is the condition, its margin verified against compiled networks, and its threshold enumerated exactly at the matched size — which is a different and weaker thing than a sampled confirmation, and we do not present it as more.

Third, the corrections we make to the original derivation are corrections of counts, not of the theorem. The shortest-path claim is sound at $\lambda = 0$ on the construction as published. What we correct is narrower: two count-dependent quantities, and the observation that the published *bias value* sits at $\lambda = D/2 - 1 \neq 0$ for $D \neq 2$, so the published *parameterization* is not length-neutral. That is a statement about a parameter choice rather than about the construction.

VIII. CONCLUSION

We have given the design conditions a stochastic spiking network must satisfy for the shortest path to be its energy minimum, shown that there are two of them rather than one, and supplied a procedure that returns both for a given instance.

That procedure is the paper’s practical content, and its scope is narrower than we could state two revisions ago. Equation (27) takes an instance’s depth, edge count and mean edge weight and returns what the *scatter* sector requires; it reproduces 1,027 measured values at 8.6 % median error, and recovers both graph families and a weight convention withheld from its fit. **What it does not return is the continuity scale an instance needs:** the scatter term is the smallest of the graded maximum, and on the reference instance it sits a factor of five below the binding one. **A designer who computes it learns a floor and learns it cheaply; a designer who inherits a customary value is usually below even that** — which is what the $c = 4$ verdict below amounts to in practice.

The first is an ordering condition. The energy of a path configuration decomposes into a term linear in the path’s weight and a term linear in its hop count, and $\lambda = 2b_k + D + c$ decides which the network ranks by; setting it to zero gives $b_k = -(D + c)/2$. Because the all-silent configuration has energy identically zero it competes with every path and constrains λ from below, which is what makes the terminal drive structural rather than incidental — and *how* it competes turns out to be decided by the bias carried by the two auxiliary neurons, a parameter with no other consequence anywhere in the analysis.

The second is a discrimination condition, and it does not follow from the first. A path competes against *scatter* configurations matching it in activity, in agreed-edge count and in every other quantity the global couplings observe, so the cross-cluster strength D cancels exactly and buys no discrimination whatever; only the continuity scale c separates them. The multiplicity ratio and the margin both grow linearly in depth, so the requirement on c is logarithmic in graph size — but its constant is instance-dependent, spanning 1.65 to 19.59 across 1,027 instances, and at the value where this construction is normally operated only one instance in forty clears it — counting microstates — at the size and in the family that matter most.

Along the way we found that a step reading as bookkeeping carries the construction’s correctness. The two auxiliary vertices make the terminals ordinary traversed vertices, which is what makes the weight-dependent energy proportional to total weight; a variant lacking them can rank a path 24 % heavier than the optimum below it. We record this because the step is small enough to be lost in any re-implementation, and because the resulting variant is internally consistent, so no amount of self-checking reveals it. To make these conditions testable at the sizes the questions demand we gave an $O(|E|)$ membrane update, validated to 2×10^{-7} and extending the tractable range by three orders of magnitude in neuron count.

And the deeper finding is that there may be no such minimum to reach. Correcting our own account of what the terminal drive does opened a sector the grading had hidden: configurations that are unions of disjoint cycles carry *one more* continuity pair than the optimum, so their advantage over it is proportional to c *with the sign reversed*, and no continuity scale suppresses them. Below $\rho = \frac{1}{4}$ their absence cannot even be certified, and on every sparse and dense instance we searched they were present. **So on most instances sampled the shortest path is not the cheapest configuration at matched activity, and no parameterization makes it so.** What removes that sector is the additive drive on the terminals — which enters no coupling, appears nowhere in the energy, and is doing more work than either design condition. **Three components a reader would take for bookkeeping** — two auxiliary vertices, a terminal bias, a membrane drive — **and the construction’s correctness rests on all three.**

What none of this establishes is that the network *reaches* the minimum it is designed to have, or that the drive excludes that sector as cleanly as a static count suggests. That is a separate question, and so is what a fair measurement of “reaching” would consist of.

DATA AVAILABILITY

The graph instances, enumerated path and matching counts, and derived required- c values behind Table V, Figure 4 and the ratio law (27) — covering all 1,080 generated instances, including the 53 censored at the declared wall and memory caps — are contained in the repository archived at Zenodo, version DOI 10.5281/zenodo.22060612 (release v1.0.3; the all-versions DOI 10.5281/zenodo.22035801 resolves to the current release), and are released under CC BY 4.0. Instances are distributed as generator seeds together with the generator, so the population is reconstructible exactly rather than only in distribution. The compiled-network verification tables (Tables I and II) are enumerations over instances specified completely in their captions and require no separate deposit.

CODE AVAILABILITY

The compiler, the $O(|E|)$ simulator, the exact path enumerator, the frontier dynamic program for the matching polynomial, and the scripts reproducing every table and figure in this paper are available at <https://github.com/>

EnglerLabs/spnn-shortest-path and archived at Zenodo, version DOI 10.5281/zenodo.22060612 — the same archive as above, which carries both the code and the data. The repository carries a claim register recording, for each numerical value printed in this paper, whether it regenerates from source or is pinned to a recorded run, and its continuous-integration workflow checks that register on every commit. **At the release archived here no entry is unresolved**, and the few values that do not regenerate — machine timings and one declared counter bound — are marked as pinned rather than reproduced. The archive is dual-licensed: code under the MIT license, and the deposited data — the contents of `instances/` and `results/` — under CC BY 4.0. The packed matching-polynomial counter records the top-coefficient bit length of every count, so the overflow condition described in Section V-C is visible in the released data rather than only in an assertion.

ACKNOWLEDGEMENTS

The author has no acknowledgements to declare.

AUTHOR CONTRIBUTIONS

G.R.E. is the sole author and is responsible for the conception of the study, the derivations, the implementation, the experiments, the analysis and the manuscript.

COMPETING INTERESTS

The author declares no competing interests.

REFERENCES

- [1] L. Buesing, J. Bill, B. Nessler, and W. Maass, “Neural dynamics as sampling: A model for stochastic computation in recurrent networks of spiking neurons,” *PLoS Computational Biology*, vol. 7, no. 11, p. e1002211, 2011.
- [2] Z. Jonke, S. Habenschuss, and W. Maass, “Solving constraint satisfaction problems with networks of spiking neurons,” *Frontiers in Neuroscience*, vol. 10, p. 118, Mar. 2016.
- [3] D. Pecevski, L. Buesing, and W. Maass, “Probabilistic inference in general graphical models through sampling in stochastic networks of spiking neurons,” *PLoS Computational Biology*, vol. 7, no. 12, p. e1002294, 2011.
- [4] S. Habenschuss, Z. Jonke, and W. Maass, “Stochastic computations in cortical microcircuit models,” *PLoS Computational Biology*, vol. 9, no. 11, p. e1003311, 2013.
- [5] G. Indiveri, F. Corradi, and N. Qiao, “Neuromorphic architectures for spiking deep neural networks,” in *2015 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2015, pp. 4.2.1–4.2.4.
- [6] W. Maass, “Noise as a resource for computation and learning in networks of spiking neurons,” *Proceedings of the IEEE*, vol. 102, no. 5, pp. 860–880, 2014.
- [7] W. Maass, “Searching for principles of brain computation,” *Current Opinion in Behavioral Sciences*, vol. 11, pp. 81–92, 2016.
- [8] G. R. Engler, Jr., “Stochastic neural algorithms,” Ph.D. dissertation, Stevens Institute of Technology, Hoboken, NJ, 2018, chapter 4 gives the construction; Chapter 5 the inhibitory subnetwork.
- [9] E. I. Moser, E. Kropff, and M.-B. Moser, “Place cells, grid cells, and the brain’s spatial representation system,” *Annual Review of Neuroscience*, vol. 31, pp. 69–89, 2008.
- [10] L. R. Howard, A. H. Javadi, Y. Yu, R. D. Mill, L. C. Morrison, R. Knight, M. M. Loftus, L. Staskute, and H. J. Spiers, “The hippocampus and entorhinal cortex encode the path and Euclidean distances to goals during navigation,” *Current Biology*, vol. 24, no. 12, pp. 1331–1340, 2014.
- [11] W. Maass, “Networks of spiking neurons: The third generation of neural network models,” *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, Dec. 1997.
- [12] P. Diaconis, “The Markov chain Monte Carlo revolution,” *Bulletin of the American Mathematical Society*, vol. 46, no. 2, pp. 179–205, 2009.
- [13] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [14] J. J. Hopfield and D. W. Tank, ““Neural” computation of decisions in optimization problems,” *Biological Cybernetics*, vol. 52, no. 3, pp. 141–152, 1985.
- [15] K. A. Smith, “Neural networks for combinatorial optimization: A review of more than a decade of research,” *INFORMS Journal on Computing*, vol. 11, no. 1, pp. 15–34, 1999.
- [16] C. Boussoño-Calzón and A. R. Figueiras-Vidal, “A bank of Hopfield neural networks for the shortest path problem,” *Signal Processing*, vol. 61, no. 2, pp. 157–170, Sep. 1997.
- [17] G. V. Wilson and G. S. Pawley, “On the stability of the travelling salesman problem algorithm of Hopfield and Tank,” *Biological Cybernetics*, vol. 58, no. 1, pp. 63–70, 1988.
- [18] E. Rueckert, D. Kappel, D. Tanneberg, D. Pecevski, and J. Peters, “Recurrent spiking networks solve planning tasks,” *Scientific Reports*, vol. 6, p. 21142, 2016.
- [19] J. L. Krichmar, “Path planning using a spiking neuron algorithm with axonal delays,” in *2016 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2016, pp. 1219–1226.

- [20] J. Friedrich and M. Lengyel, “Goal-directed decision making with spiking neurons,” *The Journal of Neuroscience*, vol. 36, no. 5, pp. 1529–1546, 2016.
- [21] G. Zhang, H. Rong, F. Neri, and M. J. Pérez-Jiménez, “An optimization spiking neural P system for approximately solving combinatorial optimization problems,” *International Journal of Neural Systems*, vol. 24, no. 05, p. 1440006, 2014.